



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Mikko Kansikas

EUREF-FIN-koordinaattikonversioiden ja N2000-korkeusmuunnoksen ohjelmointi Matlabilla

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Maanmittaustekniikka

Insinöörityö

30.3.2025

Tekijä Otsikko Sivumäärä Aika	Mikko Kansikas EUREF-FIN-koordinaattikonversioiden ja N2000-korkeusmuunnoksen ohjelmointi Matlabilla 58 sivua + 1 liite 30.3.2025
Tutkinto	insinööri (AMK)
Tutkinto-ohjelma	maanmittaustekniikka
Ohjaaja	lehtori Ilkka Partonen
<p>Insinöörityön tarkoituksena oli ohjelmoida Matlab-ohjelmistoa käyttäen joukko funktioita, joiden avulla voidaan suorittaa ETRS89-koordinaattijärjestelmän kansalliseen realisaatioon EUREF-FIN liittyvät koordinaattikonversiot sekä N2000-normaalikorkeuksien ja ellipsoidisten EUREF-FIN-korkeuksien väliset korkeusmuunnokset. Tavoitteena oli, että valmista työtä voidaan yhdessä siihen liittyvän kirjallisen osuuden kanssa käyttää opiskelun apuvälineenä, joka johdattaa lukijan aihepiiriin pariin tehokkaammin kuin aiemmin saatavilla ollut oppimateriaali.</p> <p>Työ toteutettiin kahdessa vaiheessa. Ensin kirjallisuustutkimuksena perehdyttiin entisen Geodeettisen laitoksen nykyisen Maanmittauslaitoksen paikkatietokeskuksen julkaisuihin sekä julkisen hallinnon suosituksiin. Aineistosta koostettiin työn tueksi kattava tietopaketti. Tämän jälkeen tarvittava ohjelmointityö suoritettiin käyttäen apuna ohjelmistovalmistajan julkaisemaa Matlab-dokumentaatiota sekä kolmansien osapuolien kustantamia opaskirjoja.</p> <p>Työn tuloksena syntyi helppokäyttöinen ohjelma, jolla voidaan toteuttaa sanotut konversiot ja muunnokset. Kirjallisessa osuudessa ohjelman rakenne ja toiminta selvitetään seikkaperäisesti ja annetaan ohjeet sen muokkaamiseen lukijan omiin tarpeisiin. Lisäksi tekstissä tuodaan ilmi työn edetessä esiin nousseita kehitysehdotuksia nyt valmistuneen materiaalin laajentamiseksi ja jatkokehittämiseksi.</p>	
Avainsanat	ETRS89, EUREF-FIN, N2000, ohjelmointi, Matlab
Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.	

Author Title Number of Pages Date	Mikko Kansikas Programming EUREF-FIN Coordinate Conversions and N2000 Height Transformation with Matlab 58 pages + 1 appendix 30 March 2024
Degree	Bachelor of Engineering
Degree Programme	Land Surveying
Instructors	Ilkka Partonen, Senior Lecturer
<p>The purpose of this final year project was by using Matlab to program a set of functions with which one can perform coordinate conversions related to ETRS89 coordinate reference system's national realisation EUREF-FIN and height transformations between N2000 normal heights and ellipsoidal EUREF-FIN heights. The objective was that these functions could be assembled as a conversion program to be used with the project's written part as an aid to studying that would introduce a reader to the subject matter more efficiently than previously available study materials.</p> <p>The study was carried out in two phases. First as literary research in which publications by the Finnish Geospatial Research Institute and public administration recommendations were thoroughly studied. A comprehensive information package was compiled to support programming work, which was then performed utilising the software manufacturer's Matlab documentation and third party manuals.</p> <p>The project's end result was an easy-to-use program with which the said conversions and transformations can be performed. In the written part of the project the program's structure and operational principle are explained in detail and the reader is provided with instructions to modify the program's source code to better suit their needs. Furthermore, the text highlights new ideas to extend and further develop the now completed material that arose during the work process.</p>	
Keywords	ETRS89, EUREF-FIN, N2000, programming, Matlab
The originality of this thesis has been checked using Turnitin Originality Check service.	

Sisällys

Lyhenteet

1	Johdanto	1
2	Geodesian perusteita	3
2.1	Geoidi ja ellipsoidi	3
2.2	Koordinaattijärjestelmä ja koordinaatistot	7
2.3	Korkeusjärjestelmä ja korkeudet	11
2.4	Muunnokset ja konversiot	15
3	Johdatus Matlab-ohjelmointiin	19
3.1	Ohjelmiston käyttötarkoitus	19
3.2	Päänäkymä	20
3.3	Muuttujat ja taulukot	24
3.4	Taulukko- ja matriisioperaatiot	26
3.5	Skriptit ja funktiot	29
3.6	Ehtolauseet if, elseif ja else	32
3.7	For- ja while-silmukat	35
4	Konversio-ohjelman rakenne ja toiminta	39
4.1	Taustatietoja ohjelmasta	39
4.2	Ohjelman rakenne	40
4.3	Ohjelman käyttöönotto	43
4.3.1	Valmistelevat toimenpiteet	43
4.3.2	Matlabin asetukset	45
4.3.3	Esimerkkikonversion suorittaminen	46
4.3.4	Lopputuloksen tarkastelua	47
4.4	Ohjelman laajentaminen	48
4.5	Vertailu Paikkatietoikkunan koordinaattimuunnospalvelun kanssa	52
5	Yhteenveto	56
	Lähteet	59
	Kuvalähteet	61
	Liite: Konversio-ohjelman lähdekoodi	

Lyhenteet

BeiDou	Běidǒu Wèixīng Dǎoháng Xitǒng on Kiinan puolustusministeriön hallinnoima satelliittipaikannusjärjestelmä, joka on kehitetty alun perin sotilaskäyttöön.
BLR	Baltic Levelling Ring oli Itämeren ympäri tehty korkeustasoitus, johon sisältyi vaaitushavaintoja Hollannista, Pohjois-Saksasta, Puolasta, Liettuasta, Latviasta, Virosta, Tanskasta, Norjasta, Ruotsista ja Suomesta. Tasoituksen korkeusdatumi oli NAP ja sillä määritettiin Suomen valtakunnallisen korkeusjärjestelmän N2000 lähtötaso Kirkkonummella.
CTRS	Conventional Terrestrial Reference System on globaalissa geodesiassa suorakulmainen koordinaattijärjestelmä, jonka origo on Maan massakeskipisteessä, z-akseli on Maan pyörimisakselin suuntainen, x-akseli osoittaa kohti Greenwichin meridiaania ja y-akseli on kohtisuorassa edellisiin nähden. Esimerkiksi ITRS on CTRS-järjestelmä.
EPN	EUREF Permanent Network on Euroopan laajuinen pysyvien GNSS-tukiasemien verkko, jolla ylläpidetään ETRS89-koordinaattijärjestelmää. Verkon tukiasemat ovat jatkuvasti mitattavia pisteitä, joille tunnetaan tarkat järjestelmän mukaiset koordinaatit. Niiden avulla realisoidaan ETRF-koordinaatisto. Verkkoon kuuluvia asemia hallinnoivat eri maiden yliopistot, tutkimuskeskukset ja karttalaitokset. Osa Suomen FinnRef-verkon tukiasemista kuuluu EPN-verkkoon, ja vastaavasti osa EPN-verkon tukiasemista kuuluu maailmanlaajuiseen IGS-verkkoon.
EUREF	Reference Frame Sub-Commission for Europe on kansainvälisen geodeettisen assosiaation alakomissio. Järjestön tehtäviin kuuluu ETRS89-koordinaattijärjestelmän, EVRS-korkeusjärjestelmän ja EPN-tukiasemaverkon kehittäminen.
EUREF89	Vuonna 1989 järjestetty mittauskampanja, jolla realisoitiin ensimmäistä kertaa ETRS89-koordinaattijärjestelmä. Kampanjan mittausverkko koostui 92 GPS-pisteestä sekä useista VLBI- ja satelliittilaserpisteistä. Suomesta kampanjassa olivat mukana Nisulan, Jänhiälän ja Kaunispään 1. luokan kolmiopisteet sekä Metsähovin/Sjökullan VLBI-piste.

- EUREF-FIN ETRS89-koordinaattijärjestelmän realisaatio Suomessa eli kolmiulotteinen suorakulmainen koordinaatisto, jonka määrittelee joukko maastossa olevia kiintopisteitä ja pysyviä mittausasemia, joilla on tarkat ETRS89-järjestelmän mukaiset koordinaatit.
- ETRS-GKn EUREF-FIN-koordinaatistoon liittyvä tasokoordinaatisto. Muodostetaan oikeakulmaisella lieriöprojektiolla, joka sivuaa GRS80-vertausellipsoidia pituuspiiriä pitkin. Tietystä koordinaatistosta puhuttaessa n-kirjain korvataan kokonaisluvulla väliltä 19–31, jotka ovat projektion Suomessa käytössä olevia projektiokaistoja. Luvuilla viitataan kaistojen keskimeridiaaneina käytettäviin pituuspiireihin. GK on lyhenne nimestä Gauss–Krüger.
- ETRS89 Yleiseurooppalainen koordinaattijärjestelmä, joka on kiinnitetty Euraasian mantereen deformatiiviseen osaan eli siihen osaan, joka ei laatta-tektoniikan vaikutuksesta muuta muotoaan. ETRS89 yhtyy ITRS-järjestelmään epookissa 1989.0 eli mitatun kohteen ETRS89-järjestelmässä olevat koordinaatit vastaavat sen ITRS-järjestelmässä olevia koordinaatteja vuodelta 1989. Järjestelmien myöhempi eroaminen toisistaan johtuu mannerlaattojen liikkeestä.
- ETRS-TM35FIN EUREF-FIN-koordinaatistoon liittyvä tasokoordinaatisto. Muodostetaan oikeakulmaisella lieriöprojektiolla, joka leikkaa GRS80-vertausellipsoidia pituuspiirejä pitkin. Tasokoordinaatisto kattaa koko Suomen. TM on lyhenne nimestä Transverse Mercator eli poikittainen Mercator-projektiio. Luku 35 viittaa projektion keskimeridiaaniin eli pituuspiiriin 35.
- EUVN-DA European Vertical reference Network - Densification Action oli satelliittimittauskampanja, jolla tihennettiin aiemman EUVN-kampanjan pisteistöä. Mittauksilla pyrittiin yhdistämään eurooppalaiset korkeusjärjestelmät. Pisteistöä on yhdessä Suomen kolmannen tarkkavaaituksen tulosten kanssa käytetty FIN2005N00-kvasigeoidimallin määrittämiseen.
- EVRS European Vertical Reference System on Euroopan korkeusvertausjärjestelmä eli korkeusdatumi, jonka suomalainen realisaatio eli korkeusvertauskehys on N2000-korkeusjärjestelmä. Vertausjärjestelmän mukaiset korkeudet ovat normaalikorkeuksia ja sen lähtötasona on NAP.

- FIN2005N00 Suomalaisen N2000-korkeusjärjestelmän kanssa käytettävä kvasigeoidimalli, jonka sisältämien kvasigeoidin korkeuksien eli korkeusanomalioiden avulla voidaan muuntaa EUREF-FIN-koordinaatistossa olevat ellipsoidiset korkeudet GRS80-vertausellipsoidin pinnalta normaalikorkeuuksiksi.
- GLONASS GLObalnaja NAVigatsionnaja Sputnikovaja Sistema on Venäjän puolustusministeriön hallinnoima satelliittipaikannusjärjestelmä, joka on kehitetty alun perin sotilaskäyttöön.
- GNSS Global Navigation Satellite System on yhteisnimitys maailmanlaajuisesti toimiville satelliittipaikannusjärjestelmille.
- GOCE Gravity Field and Steady-State Ocean Circulation Explorer oli Euroopan avaruusjärjestön kaukokartoitussatelliitti, joka mittasi Maan painovoimakenttää. Mittausten avulla voitiin mallintaa Maan geoidi aiempaa tarkemmin. Satelliitti oli toiminnassa vuosina 2009–2013.
- GPS Global Positioning System on Yhdysvaltain puolustusministeriön hallinnoima satelliittipaikannusjärjestelmä, joka on kehitetty alun perin sotilaskäyttöön.
- GRS80 ETRS89-koordinaattijärjestelmän vertausellipsoidi eli pyörähdysellipsoidi, jonka keskipiste on järjestelmän origossa ja pyörähdysakseli järjestelmän Z-akselin suuntainen. Pyörähdysellipsoidi on matemaattinen malli, joka pallosta poiketen huomioi Maan litistyneisyyden navoilta. Mitatun kohteen geodeettiset koordinaatit määritetään vertausellipsoidin avulla.
- IERS International Earth Rotation and Reference Systems Service on kansainvälinen Maan pyörähdysliikettä, sen parametreja sekä koordinaattijärjestelmiä ja niiden realisaatioita määrittävä järjestö. IERS on määrittänyt ITRS-koordinaattijärjestelmän ja sen realisaation ITRF-koordinaatiston.
- IGS International GNSS Service on maailmanlaajuinen pysyvien GNSS-tukiasemien verkko, jolla ylläpidetään ITRS-koordinaattijärjestelmää. Verkon tukiasemat ovat jatkuvasti mitattavia pisteitä, joiden avulla realisoidaan ITRF-koordinaatisto. Osa Suomen FinnRef-verkon ja eurooppalaisen EPN-verkon tukiasemista kuuluu myös IGS-verkkoon.

ITRF	International Terrestrial Reference Frame on kolmiulotteinen suorakulmainen koordinaatisto, joka on toteutettu satelliittilaser-, kuulaser-, VLBI-, Doris- sekä GPS- ja GLONASS-havaintojen avulla. ITRF-koordinaatisto on ITRS-koordinaattijärjestelmän realisaatio.
ITRS	International Terrestrial Reference System on IERS:n ylläpitämä maailmanlaajuinen CTRS:n mukainen kolmiulotteinen suorakulmainen koordinaattijärjestelmä.
KKJ	KartastoKoordinaattiJärjestelmä oli Suomessa vuonna 1970 käyttöönotettu koordinaattijärjestelmä, joka perustui kolmiomittauksiin. Järjestelmään kuului kuusi erillistä kolmen asteen välein jaoteltua tasokoordinaatistoa ja sen kanssa käytettiin Gauss–Krüger-karttaprojektiota. Järjestelmän käytöstä luovuttiin pääosin 2010-luvun aikana.
MATLAB	MATrix LABoratory on suosittu numeerisessa laskennassa käytettävä ohjelmisto ja ohjelmointikieli, jota hyödynnetään erityisesti insinööri- ja luonnontieteiden parissa. Ohjelmisto on alkujaan suunniteltu lineaarialgebraa tarvittavaa matriisilaskentaa varten.
N2000	Valtakunnallinen korkeusjärjestelmä, joka perustuu vuosina 1978–2006 tehtyyn Suomen kolmanteen tarkkavaaitukseen. Järjestelmän lähtötason määrittävä kiintopiste PP2000 sijaitsee Metsähovin observatorion alueella Kirkkonummella. N2000-järjestelmä on sidottu NAP:n määrittämään korkeustasoon. Sen korkeudet ovat normaalikorkeuksia.
NAP	Normaal Amsterdams Peil on korkeusdatumien lähtötaso, jota käytetään useissa Euroopan Unionin maissa. Sen nollakorkeus on IJ-joen keskimääräinen vuoroveden huippukorkeus Amsterdamissa vuonna 1684. Korkeus on määritelty kaupungissa sijaitsevalla kiintopisteellä 13600.
NKG	Nordic Geodetic Commission, ruotsiksi Nordiska Kommissionen för Geodesi eli Pohjoismainen Geodeettinen Komissio on pohjoismaisten ja baltialaisten geodesian tutkijoiden järjestö, jonka tehtävä on edistää sen jäsenten ammatillista yhteistyötä. Osana komissiota toimivat sen geodynaamiikan ja Maan havainnoinnin, korkeuksien ja geoidin, satelliittipaikannuksen sekä vertauskehysten työryhmät.

- NKG2004 Nordic Geodetic Commission 2004 on pohjoismainen kvasigeoidimalli. Sovittamalla malli pisteisiin, joille tunnettiin N2000-järjestelmän mukaiset korkeudet ja EUREF-FIN-järjestelmän mukaiset koordinaatit, luotiin suomalainen FIN2005N00-kvasigeoidimalli.
- NKG2005LU Nordic Geodetic Commission 2005 Land Uplift on pohjoismainen maannousumalli, jonka avulla on määritetty maannousukorjaukset vuosien 1978–2006 Suomen kolmannen tarkkavaaitukseen vaihtelevista vaitusvuosista N2000-korkeusjärjestelmän epookkiin vuoteen 2000.
- PZ-90 Parametry Zemli 1990 on Venäjän puolustusministeriön kehittämä koordinaattijärjestelmä sekä samanniminen koordinaatisto ja vertausellipsoidi. Sitä käytetään GLONASS-satelliittipaikannusjärjestelmän kanssa.
- UTM Universal Transverse Mercator on Yhdysvalloissa vuonna 1947 käyttöön otettu karttaprojektiojärjestelmä, joka käyttää poikittaista leikkaavaa lieeriöprojektiota, jonka mittakaavakerroin projektion keskimeridiaanina käytettävällä pituuspiirillä on 0,9996. Järjestelmä jakaa Maa-planeetan 60:een kuusi astetta leveään projektiokaistaan, joista Suomen alueella ovat käytössä kaistat 34–36. Koko Suomen kattava tasokoordinaatisto ETRS-TM35FIN vastaa muuten UTM-kaistaa 35, mutta on n. 13 ° leveä.
- VLBI Very-Long-Baseline Interferometry on radioastronomian tekniikka, jossa eri puolilla maailmaa sijaitsevat teleskoopit havainnoivat yhtä aikaa samaa lähdettä samalla taajuudella, toimien suurena virtuaaliteleskooppina.
- WGS-84 World Geodetic System 1984 on Yhdysvaltain puolustusministeriön kehittämä koordinaattijärjestelmä sekä samanniminen koordinaatisto ja vertausellipsoidi. Sitä käytetään GPS-satelliittipaikannusjärjestelmän kanssa. Monet internetin karttapalvelut, kuten Google Maps, käyttävät WGS84-koordinaatteja, mutta eivät vertausellipsoidia karttaprojektioissa.
- YKJ Yhtenäiskoordinaattijärjestelmä tai yhtenäiskoordinaatisto oli koko Suomen kattava tasokoordinaatisto, joka ei ole enää aktiivisessa käytössä. Sen karttaprojektion keskimeridiaanina oli 27. pituuspiiri, ja se vastasi muuten kartastokoordinaattijärjestelmän koordinaatistoa KKJ3, mutta sen projektiokaista oli koko maan levyinen.

1 Johdanto

Kohteiden sijainnin määrittäminen suhteessa toisiinsa ja ympäröivään maailmaan on maanmittarin ammattitaidon kovaa ydintä, joka suoraan tai välillisesti liittyy lähes kaikkiin alan opintoihin ja työtehtäviin. Sijaintitietojen täsmällisyydestä huolehtiminen onkin välttämätöntä osa sekä käytännön mittauksia maastossa suorittavien että mittausten avulla saatuja paikkatietoaineistoja käsittelevien alan ammattilaisten työtä. Vaikka tähän tarvittava tekniikan ja ammattikielen tuntemus muodostuu tavallisesti lukuisten toistojen kautta nopeasti osaksi aloittelevan maanmittarin ammatillisia rutiineja, saattaa käsitys taustalla vaikuttavista tieteellisistä ja matemaattisista lainalaisuuksista helposti jäädä vajavaiseksi.

Satelliittipaikannuksen tultua erottamattomaksi osaksi maanmittausalan arkea, on koordinaatistojen hallinnan merkitys kasvanut yhä suuremmaksi. Jokapäiväisissä työtehtävissään maanmittari toimii yleisimmin pituuspiireittäin jaoteltujen valtakunnallisten, ja siten vain rajatulla maantieteellisellä alueella käyttökelpoisten, kaksiulotteisten tasokoordinaatistojen kanssa, joihin on kolmiulotteisen todellisuuden kuvaamiseksi erikseen liitetty ihmiselle helposti hahmotettavissa oleva korkeus merenpinnasta. Tällöin puhutaan niin sanotuista 2D+1D-koordinaatistoista. Satelliittipaikannuksella suoritettuja mittauksia ei kuitenkaan tehdä tasokoordinaatistossa, vaan kolmiulotteisessa, maakeskeisessä koordinaatistossa, joka kattaa planeettamme kokonaisuudessaan. Sijaintitietojen esitystavan muuttaminen jälkimmäisestä ensin mainittuun on monivaiheinen tapahtumasarja, jonka kokonaisvaltainen ymmärtäminen ei ole itsestäänselvyys alalla pidempäänkään työskennelleiden kohdalla.

Tämä työ on syntynyt tarpeesta tuottaa aineistoa, jota apuna käyttäen lukija voi perehtyä kyseiseen, koordinaattikonversioita ja korkeusmuunnoksen käsittävään, tapahtumasarjaan nopeammin ja tehokkaammin kuin aiemmin saatavilla olleen kirjallisen materiaalin avulla on ollut mahdollista, ja joka toisaalta voisi toimia luonnoksena siitä mihin suuntaan vastaavan kaltaista ainesta käsitteleviä oppimateriaaleja olisi mahdollista kehittää. Innoituksena työlle on ollut kirjoittajalle opintojen aikana muodostunut kokemus siitä, kuinka taustaltaan matemaattisten käsitteiden omaksuminen on useimmiten huomattavasti vaivattomampaa itse laskemalla ja ohjelmoimalla kuin kulloiseenkin aihepiiriin tutustuminen lukemiseen keskittymällä.

Työ muodostuu kahdesta erillisestä osasta, jotka on tarkoitettu yhdessä käytettäviksi. Ensimmäinen on Matlab-ohjelmistoa käyttäen ohjelmoitu konversio- ja muunnosohjelma, jonka avulla lukija voi itse kokeilemalla ja ohjelmakoodia muokkaamalla tutustua aiheeseen. Toinen on käsillä oleva kirjallinen osuus, jonka tehtävä on toimia ohjelmaa täydentävänä aineistona ja tarjota tiivis, mutta riittävän kattava johdatus sen ominaisuuksiin. Ohjelmaa on mahdollista käyttää myös ilman kirjallista osuutta, edellyttäen että aihepiiri on lukijalle ennestään tuttu.

Kirjallinen osuus on jaettu kolmeen osaan: Tekstissä käydään ensin läpi keskeisiä Maa-planeetan muodon määrittämisessä tarvittavia käsitteitä, joiden tunteminen on välttämätöntä konversio- ja muunnosohjelman toiminnan ymmärtämiseksi tai siihen läheisesti liittyvät. Tämän jälkeen lukija tutustutetaan Matlab-ohjelmistoon ja sen keskeisiin toimintaperiaatteisiin, mihin kuuluvat sekä ohjelmiston graafisen käyttöliittymän toiminnot että samannimisen ohjelmointikielen tärkeimmät ominaisuudet. Lopuksi paneudutaan itse konversio- ja muunnosohjelmaan käyden läpi sen rakenne ja toimintaperiaate. Tähän sisältyy paitsi opastus ohjelman käyttöönottamiseen myös havainnolliset ohjeet sen toiminnan laajentamiseen.

Tämän tekstin myöhemmissä vaiheissa konversio- ja muunnosohjelmaan viitataan lyhyemmin ja hieman harhaanjohtavasti vain nimellä konversio-ohjelma. Lukijan käyttöön konversio-ohjelma on saatavilla zip-pakattuna tiedostona (Konversiot.zip) Matlab-ohjelmistoa valmistavan Math Works -yhtiön tiedostonvaihtopalvelusta nimellä *EUREF-FIN coordinate conversions and N2000 height transformation*. Lisäksi tämän tekstin liitteenä ovat lähes kaikki konversio-ohjelman osat: konversio- ja muunnosfunktiot, niiden yhteistoiminnasta vastaava pääohjelma, skriptit, joiden avulla ohjelman asetukset määritetään ja esimerkkikonversiot suoritetaan sekä esimerkkikonversioiden tarvitsemat koordinaattitiedot. Liitteeseen ei kuitenkaan sisälly Maanmittauslaitoksen paikkatietokeskukselta saatu FIN2005N00-kvasigeoidimalli, joka on satojentuhansien rivien mittaisena liian suuri liitettäväksi tähän tekstiin. Malli (FIN2005N00_list.zip) on noudettavissa Maanmittauslaitoksen verkkosivuilta. (1; 2.)

2 Geodesian perusteita

2.1 Geoidi ja ellipsoidi

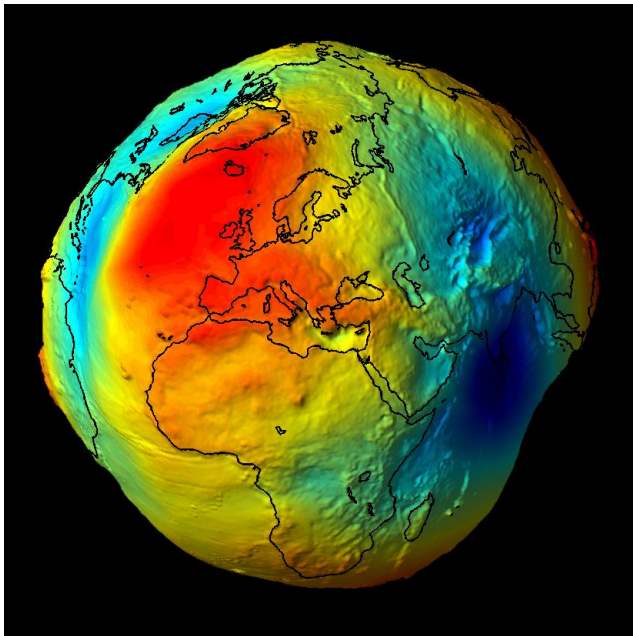
Maanmittarit ovat perinteisesti tottuneet toimimaan kaksiulotteisten tasokoordinaatistojen ja niihin liitettyjen meren keskivedenpintaan sidottujen korkeuksien kanssa. Kun nämä yhdistetään, voi mittaaja hahmottaa maailmaa suorakulmaisten koordinaatistojen kautta. Tällaiset suorakulmaiset koordinaatistot ovat kuitenkin käyttökelpoisia vain paikallisesti, ja satelliittimittauksen hyödyntämiseksi ne on liitettävä osaksi globaalia, koko maailman kattavaa koordinaatistoa, jollaisten varassa myös satelliittipaikannusjärjestelmät toimivat. Tätä varten tarvitaan Maa-planeetan muotoa kuvaavia matemaattisia malleja, joita kutsutaan geoidiksi ja ellipsoidiksi. (3, s. 133–137.)



Kuva 1. Maa avaruudesta käsin nähtynä. Astronautti Harrison Schmitt otti kuuluisan valokuvan "The Blue Marble" Apollo 17 -kuulennolla vuonna 1972.

Ihmiselle on luontevaa ajatella Maata pallona, jolta se muiden planeettojen tapaan avaruudesta otetuista kuvista näyttää (kuva 1). Todellisuudessa asia on kuitenkin huomattavasti monimutkaisempi. Maan pinnalla, ja myös merenpinnalla, on oma muotonsa eli topografia, sen massa on jakautunut epätasaisesti ja se on pyörähdysliikkeensä vaikutuksesta litistynyt navoilta. Näistä Maan pallonmuodosta poikkeamisen aiheuttavista seikoista topografiaa ei tarvitse mallintaa, sitä mittaaja havainnoi maastossa. Epätasainen massan jakautuminen ja sen vaikutus painovoimaan huomioidaan geoidin avulla, kun taas napojen litistymisen kuvaamiseen käytetään ellipsoidia. (3, s.135–136.)

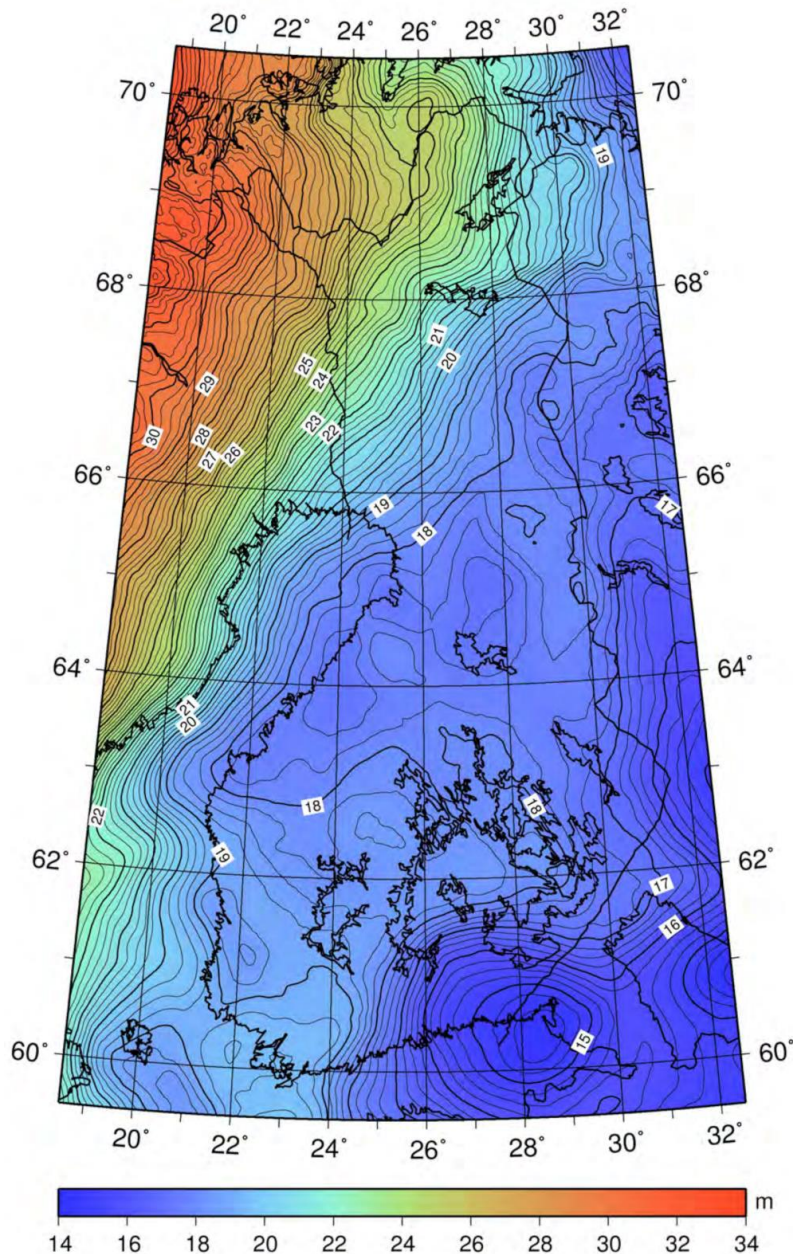
Vakiintuneen määritelmän mukaan geoidi on Maan painovoimakentän tasa-arvopinta eli ekvipotentiaalipinta, joka yhtyy likimäärin valtameren keskivedenpintaan. Tarkalleen ottaen tämä ei pidä paikkaansa, johtuen merivirtojen ja lämpölaajenemisen kaltaisten geofysikaalisten ilmiöiden aiheuttamista poikkeamista geoidin ja keskivedenpinnan välillä. Painovoimakentän tasa-arvopinnalla tarkoitetaan sitä, että painovoima on kaikkialla geoidin pinnalla yhtä suuri. Painovoiman suunta eli ns. luotiviiva puolestaan on aina kohtisuorassa geoidia vasten, eli toisin sanoen geoidin pinta on siis vaakatasossa. Jos geoidi olisi kiinteä pinta, sille asetettu kuula ei näin vierisi mihinkään suuntaan. Maanmittarin näkökulmasta geoidi on tärkeä käsite, koska sen avulla satelliittimittauksella saadut ellipsoidiset korkeudet voidaan muuntaa samaan järjestelmään keskimerenpintaan sidottujen vaaittujen korkeuksien kanssa. (3, s.135–136; 4, s. 34; 5, s. v, 2.)



Kuva 2. ESA:n GOCE-kaukokartoitussatelliitin mittausten perusteella tehty geoidimalli, joka käsittää koko Maan.

Tavallisesti geoidi kuvataan korkeuksina vertausellipsoidista, jolloin kyseessä on geoidimalli. Kuvassa 2 olevan globaalin geoidimallin pintaan on sovitettu vertausellipsoidi, ja mallin korkeuden vaihteluja on liioiteltu. Punaiset alueet ovat ellipsoidin pinnan yläpuolella ja siniset alueet alapuolella. Intian valtamerellä geoidi on selvästi kuopalla, kun taas Pohjois-Atlantilla se on koholla. Geoidimallin ja vertausellipsoidin välinen ero eli geoidin korkeus on siis Intian valtamerellä negatiivinen ja Pohjois-Atlantilla positiivinen. Geoidin pinta vastaa sitä tasoa, jossa vaaitsemalla mitattu korkeus olisi nolla, eli geoidin korkeus on ellipsoidisten ja vaaittujen korkeuksien välinen poikkeama. (5, s. 2.)

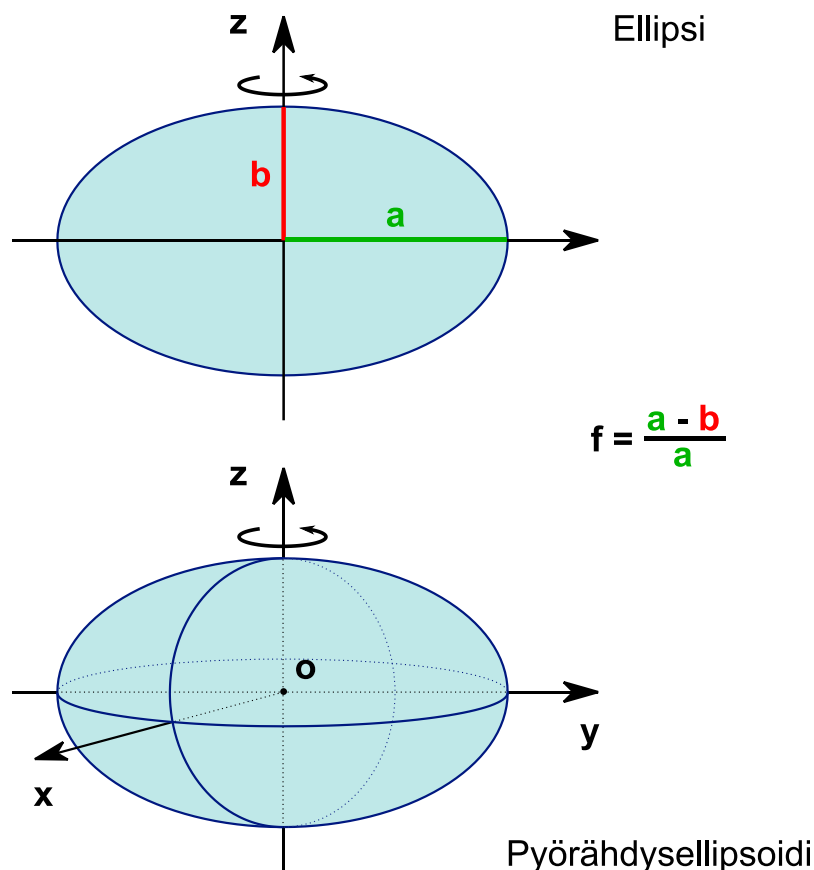
Geoidimallin ei tarvitse välttämättä olla globaali, vaan se voi olla myös paikallisesti rajattu, kuten asia yleensä onkin. Paikallinen geoidimalli voi olla esimerkiksi Euroopan tai Pohjoismaiden laajuinen, tai se voi kattaa vain yhden valtion lähialueineen, kuten konversio-ohjelmaan liittyvä Suomessa nykyisin käytettävä FIN2005N00-kvasigeoidimalli (kuva 3), jonka avulla satelliittimittauksin GRS80-vertausellipsoidin pinnalta saadut ellipsoidiset korkeudet muunnetaan N2000-korkeusjärjestelmän mukaisiksi. (5, s. 2, 7.)



Kuva 3. Maanmittauslaitoksen paikkatietokeskuksen tuottama kvasigeoidimalli FIN2005N00. Kuvassa on esitetty kvasigeoidin korkeudet (korkeusanomaliat) GRS80-vertausellipsoidin pinnalta metreinä. Korkeuskäyrät on piirretty 0,25 metrin välein.

Geodesiassa ellipsoidilla tarkoitetaan Maan muotoa karkeasti kuvaavaa pyörähdysellipsoidia, joka syntyy, kun ellipsi pyörähtää pikkuakselinsa ympäri. Matematiikassa pyörähdysellipsoidi voi syntyä myös ellipsin pyörähtäessä isoakselinsa ympäri, ja pyörähdysellipsoidi itsessään taas on ellipsoidin erikoistapaus, jonka kolmesta akselista kaksi ovat yhtä pitkiä. Kuvan 4 mukaisesti ellipsin ja pyörähdysellipsoidin muotoa voidaan kuvata iso- ja pikkuakselin pituuden sekä litistyssuhteen avulla. (3, s. 137; 6; 7, s. 5.)

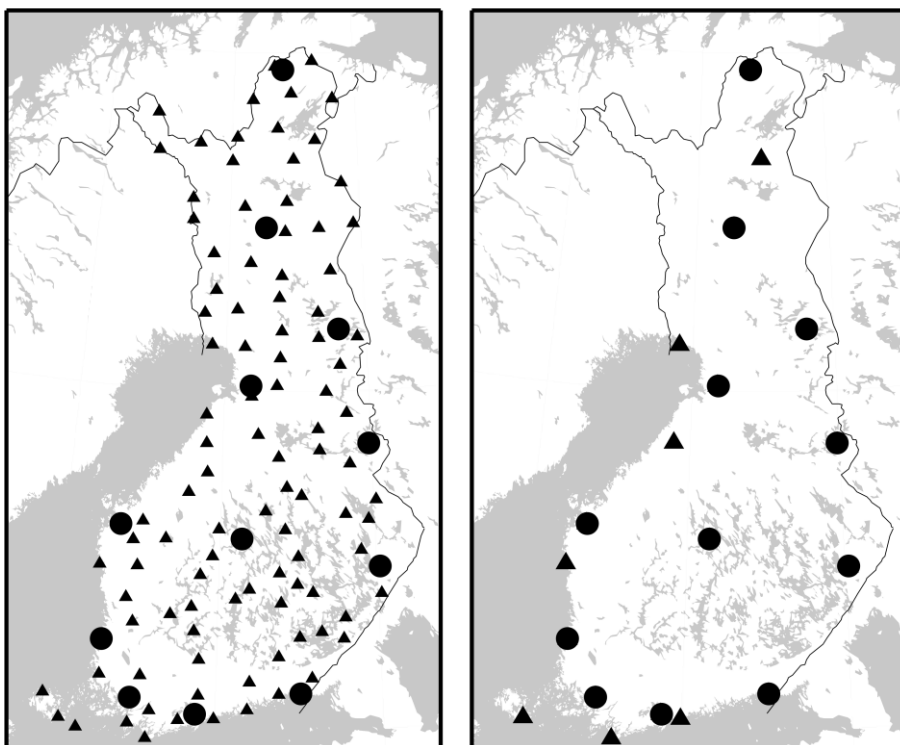
Kun ellipsoidi on osa jotakin koordinaattijärjestelmää, sitä kutsutaan vertausellipsoidiksi. GRS80-vertausellipsoidi on osa ETRS89-koordinaattijärjestelmää. Vertausellipsoidin tehtävänä on toimia vertailupintana, jonka avulla suorakulmaisessa koordinaatistossa mitatun kohteen sijainti esitetään suhteessa Maan pintaan eli maantieteellisinä koordinaatteina sekä etäisyytenä vertailupinnasta eli ellipsoidisena korkeutena. Yhdessä näitä kutsutaan geodeettisiksi koordinaateiksi, jotka taas voidaan edelleen projisoida tasokoordinaateiksi ja muuntaa vaaittuja korkeuksia vastaaviksi. (3, s. 138; 5, s. 3.)



Kuva 4. Ellipsi ja ellipsoidi sijoitettuna koordinaatistoon. a on ellipsin isoakselin puolikas ja b pikkuakselin puolikas. f on ellipsoidin litistyssuhde. GRS80-pyörähdysellipsoidin keskipiste on Maan massakeskipisteessä, z -akseli Maan pyörimisakselin suuntainen, x -akseli osoittaa Greenwichin meridiaania ja y -akseli on kohtisuorassa edellisiin nähden.

2.2 Koordinaattijärjestelmä ja koordinaatitot

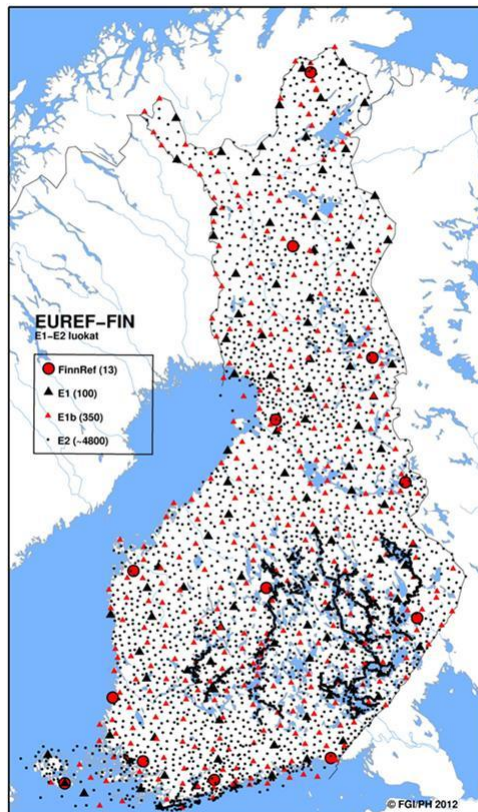
Maailmanlaajuisia satelliittipaikannusjärjestelmiä kutsutaan yhteisnimityksellä GNSS. Näitä järjestelmiä ovat yhdysvaltalainen GPS, venäläinen GLONASS, eurooppalainen Galileo ja kiinalainen BeiDou. Kaikki satelliittipaikannusjärjestelmät toimivat omissa koordinaatistoissaan, esim. GPS:n kanssa käytetään WGS-84-koordinaatistoa ja GLONASS:n kanssa vastaavasti PZ-90-koordinaatistoa, joten niiden yhteiskäyttö samalla mittalaitteella vaatii mittaustulosten muuntamista samaan koordinaatistoon. Tällainen koordinaatisto on kansainvälisen IERS-järjestön ylläpitämä ITRF, joka perustuu



Kuva 5. Vuosien 1996–1997 EUREF-FIN-mittauskampanja. Vasemmalla on esitetty kaikki kampanjan aikana mitatut 112 pistettä, jotka määrittelevät EUREF-FIN-koordinaatiston ja oikealla niistä valittu 19 pisteen alijoukko, jonka EUREF-komissio on hyväksynyt Suomen viralliseksi ETRS89-realisaatioksi. FinnRef-verkon tukiasemat (12 kpl) on merkitty mustilla ympyröillä ja mitatut kiintopisteet (100 kpl) mustilla kolmioilla. Kuva on vuodelta 2009, FinnRef-tukiasemien määrä on sittemmin kasvanut jo 47:ään.

ITRS-koordinaattijärjestelmään. Käsite koordinaattijärjestelmä eli terrestrinen vertausjärjestelmä tarkoittaa datumin avulla Maahan kiinnitettyä koordinaatistoa, ja datumi taas niitä parametreja, joilla määritellään järjestelmän koordinaattiakselien ja vertausellipsoidin suhde planeettaan. ITRS-järjestelmän vertausellipsoidi on GRS80. ITRS on niin sanottu CTRS-järjestelmä, mikä merkitsee, että se on suorakulmainen koordinaatti-

järjestelmä, jonka origo on Maan massakeskipisteessä, z-akseli on Maan pyörimisakselin suuntainen, x-akseli osoittaa kohti Greenwichin meridiaania ja y-akseli on kohtisuorassa edellisiin nähden. ITRS on realisoitu ITRF-koordinaatistoksi satelliittilaser-, kuulaser-, VLBI-, Doris- sekä GPS- ja GLONASS-havaintojen avulla. Realisaatiolla tarkoitetaan, että Maan pinnalle on mitattu joukko kiintopisteitä, joille on määritetty koordinaatit ITRF-koordinaatistossa. Mannerlaattojen liikkeen takia realisaatio on suori-



Kuva 6. EUREF-FIN-pistetihennys. EUREF-FIN-koordinaatiston määrittelevät E1-luokan pisteet eli pysyvät FinnRef-verkon GNSS-asetat (13 kpl) sekä vuosien 1996–1997 EUREF-FIN-mittauskampanjan aikana mitatut pisteet (100 kpl) on esitetty kuvassa punaisilla ympyröillä ja mustilla kolmioilla. Geodeettisen laitoksen vuosina 1998–1999 mitaamat E1b-luokan pisteet (350 kpl) on esitetty kuvassa punaisilla kolmioilla ja Maanmittauslaitoksen mitaamat E2-luokan pisteet mustilla pisteillä. Kuva on vuodelta 2012, esimerkiksi FinnRef-asettien määrä on sittemmin kasvanut jo 47:ään.

tettava uudelleen muutaman vuoden välein, ja ITRF-koordinaatistosta on siksi olemassa useita eri versioita. Jatkuvasti muuttuvat ITRF-koordinaatit olisivat jokapäiväisessä mittaustoiminnassa varsin epäkäytännöllisiä, joten eri puolilla maailmaa on luotu omia mannerlaattojen kiinteisiin osiin sidottuja koordinaatistoja ja koordinaattijärjestelmiä. Euroopassa on käytössä ETRS89-koordinaattijärjestelmä, joka yhtyy ITRS-järjestelmään epookkina 1989.0. Järjestelmät ovat siis tänä ajankohtana samat, mikä tarkoit-

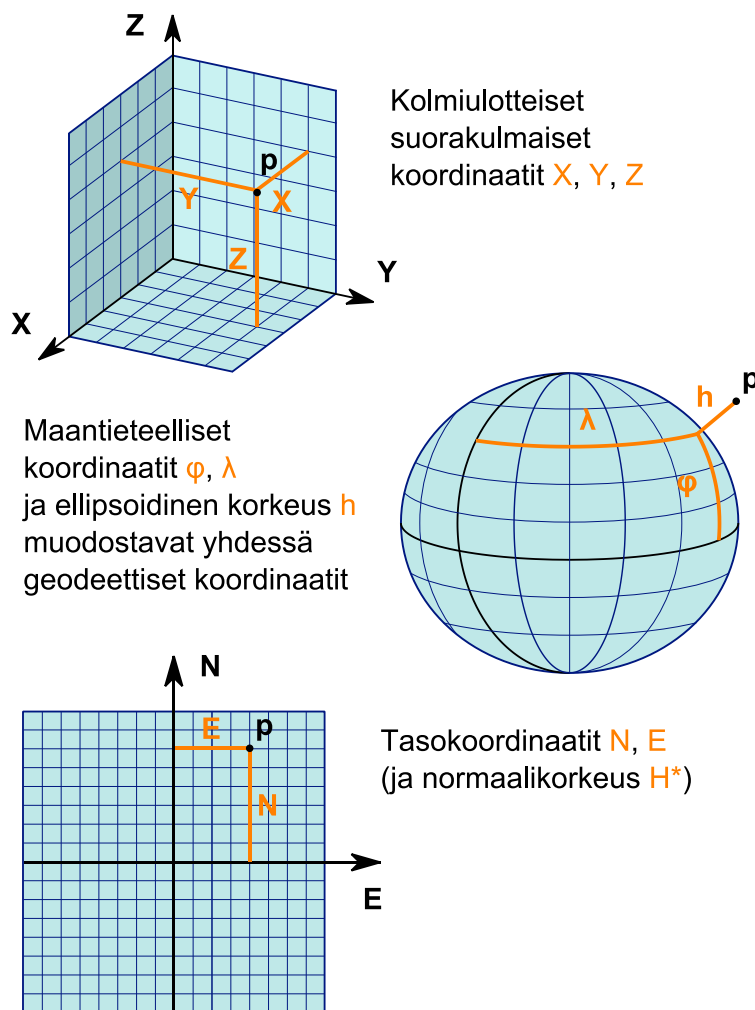
taa, että myös järjestelmien mukaisissa koordinaatioissa eli terrestrisissä vertauskehysissä tänä ajankohtana mitatun pisteen koordinaatit ovat samat. ETRS89-järjestelmästä on olemassa sekä yleiseurooppalainen, että monia paikallisia realisaatioita. Järjestelmän realisaatio koko Euroopan alueella on ETRF-koordinaatio, ja paikallisesti Suomessa EUREF-FIN-koordinaatio. Koordinaatit on liitetty toisiinsa yhteisten pisteiden avulla. Konversio-ohjelmalla voi nimensä mukaisesti suorittaa koordinaattikonversioita ETRS89-järjestelmän määrittävään datumiin perustuvien koordinaatioiden välillä (luku 2.4). Ennen tätä vaihetta suoritettavat koordinaattimuunnokset eivät kuulu tämän työn aihepiiriin. (3, s. 154–155; 4, s. 7, 19–22, 59, 62; 8, s. 4.)

ETRS89-järjestelmän historia ulottuu vuoteen 1989, jolloin sen realisoimiseksi toteutettiin Euroopan laajuinen mittauskampanja EUREF89. Mittausverkkoon kuului 92 GPS-pistettä sekä useita VLBI- ja satelliittilaserpisteitä. Kampanjassa käytettiin myös neljää pistettä Suomesta: Metsähovin/Sjökullan VLBI-pistettä sekä Nisulan, Jänhiälän ja Kainispään I luokan kolmiopisteitä. Verkon tasotuslaskennassa VLBI-pisteiden koordinaatteja pidettiin kiinteinä, mistä seuraa, että koordinaation orientointi ja mittakaava ovat peräisin VLBI-havainnoista. Seuraavana vuonna 1990 ETRS89 luotiin virallisesti EUREF-järjestön kokouksen päätöslauselmalla Firenzessä, ja vuonna 1999 Euroopan komissio järjesti kokouksen, joka suositteli ETRS89:n hyväksymistä yleiseurooppalaiseksi koordinaattijärjestelmäksi. (4, s. 22–23; 8, s. 1.)

Suomessa ETRS89-järjestelmän kansallista realisaatiota alettiin valmistella 1990-luvun alussa, ja Geodeettinen laitos rakensi pohjoismaisen NKG-järjestön suosituksesta vuosina 1993–1996 Manner-Suomeen 12 pysyvän GPS-tukiaseman FinnRef-verkon. Myöhemmin vuonna 2005 verkkoon liitettiin 13. asemana Ahvenanmaan Degerby ja 2020-luvulla verkko on kasvanut jo 47:n GNSS-aseman laajuiseksi. Alkuperäisestä verkosta Joensuun, Metsähovin, Sodankylän ja Vaasan neljä FinnRef-asemaa kuuluvat myös eurooppalaiseen EPN-verkkoon, jolla ylläpidetään ETRS89-järjestelmää, ja Metsähovin asema maailmanlaajuiseen IGS-verkkoon, jolla ylläpidetään ITRS-järjestelmää. Näihin kansainvälisiin verkkoihin kuuluvien suomalaisasemien määrä on sittemmin kasvanut. (4, s. 23.)

Geodeettinen laitos realisoi ETRS89-järjestelmän EUREF-FIN-koordinaatistoksi vuosina 1996–1997 mittauskampanjalla, johon kuuluivat FinnRef-verkon 12 tukiasemaa sekä sata kampanjan aikana mitattua kiintopistettä (kuva 5). Nämä pisteet muodostavat Suomen kiintopisteverkon korkeimman E1-luokan, ja ne määrittelevät EUREF-FIN-

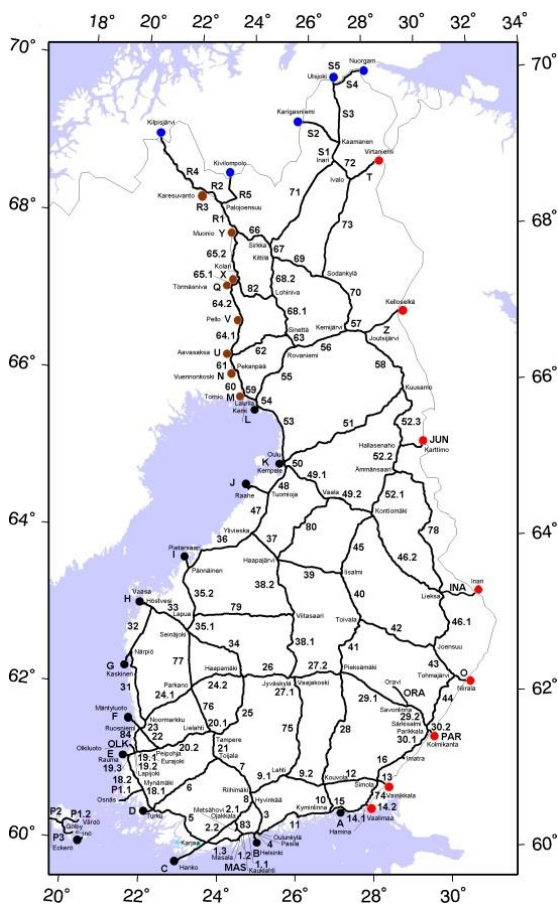
koordinaatiston. Johtuen siitä, että pisteitä on hyvin harvassa ja monesti vaikeasti saatavissa paikoissa, mittasi Geodeettinen laitos vuosina 1998–1999 lisäksi 350 uutta E1b-luokan pistettä, jotka sidottiin E1-luokan pisteisiin. Maanmittauslaitos puolestaan on mitannut n. 2500 E2-luokan pistettä, jotka on sidottu E1- ja E1b-luokan pisteisiin (kuva 6). EUREF-FIN-koordinaatit ovat kolmiulotteisia suorakulmaisia koordinaatteja, jotka voidaan esittää myös geodeettisina koordinaatteina (kuva 7). Käytännön mittauksissa ne projisoidaan tasokoordinaateiksi, joihin liitetään N2000-järjestelmän mukainen normaalikorkeus. (4, s. 23–24, 27; 8, s. 8.)



Kuva 7. ETRS89-koordinaattijärjestelmän kansallisen realisaation EUREF-FIN kanssa käytettävät koordinaatistotyypit. Ylhäältä alas: kolmiulotteinen suorakulmainen koordinaatisto, geodeettinen koordinaatisto ja tasokoordinaatisto. Mitatun pisteen (p) koordinaatit on esitetty kuvassa oranssilla: suorakulmaiset koordinaatit (X, Y, Z); maantieteelliset koordinaatit leveyskulma (ϕ) ja pituuskulma (λ), jotka yhdessä ellipsoidisen korkeuden (h) kanssa muodostavat geodeettiset koordinaatit; tasokoordinaatit pohjoinen (N) ja itä (E), joihin liitetään mukaan N2000-järjestelmän mukainen normaalikorkeus (H^*), jolloin muodostuvat mittaajalle havainnolliset, suorakulmaiset 2D+1D-koordinaatit.

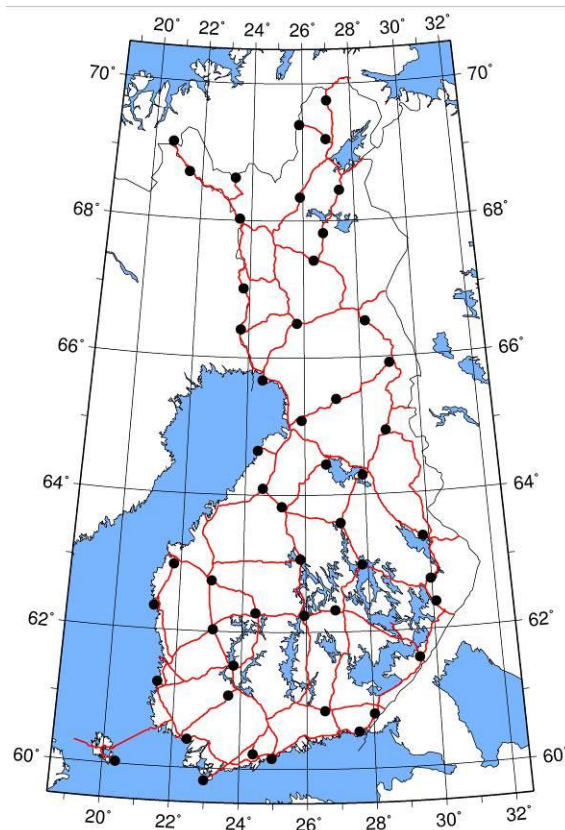
2.3 Korkeusjärjestelmä ja korkeudet

Korkeusjärjestelmä eli korkeusvertauskehys tarkoittaa korkeusdatumin eli korkeusvertausjärjestelmän realisaatiota maastossa. Suomessa on nykyisin käytössä N2000-korkeusjärjestelmä, joka perustuu Euroopan korkeusvertausjärjestelmään (EVRS). Korkeusdatumi määrittää realisoinnin lähtötason eli lähtökorkeuden. Realisointi tapahtuu vaaitsemalla ja tasoituslaskemalla korkeuksia järjestelmän kattamalla alueella sijaitseville korkeuskiintopisteille. N2000-korkeusjärjestelmä perustuu vuosina 1978–2006 tehtyyn Suomen kolmanteen tarkkavaaitukseen. Järjestelmän lähtötason määrittävä kiintopiste PP2000 sijaitsee Metsähovin observatorion alueella Kirkkonummella. Pisteen korkeus (54,4233 m) on johdettu Itämeren ympäri tehdyllä tasoituksella (BLR), jonka lähtötasona oli EVRS:n mukaisesti Normaal Amsterdams Peil eli NAP, joka kuvaa merenpinnan tasoa Amsterdamissa. (4, s. 32–33; 9, s. 3; 10, s. 16, 19; 11, s. 9.)



Kuva 8. Suomen kolmas tarkkavaaitusverkko. Vaaituslinjat, jotka ovat osana sulkeutuvaa silmukkaa on numeroitu. Haaralinjoilla, jotka eivät ole osana sulkeutuvaa silmukkaa, on kirjaintunnukset. Mustat pallot ovat mareografeja eli meriveden korkeudenmittausasemia ja värilliset pallot liitoksia naapurimaiden vaaitusverkkoihin.

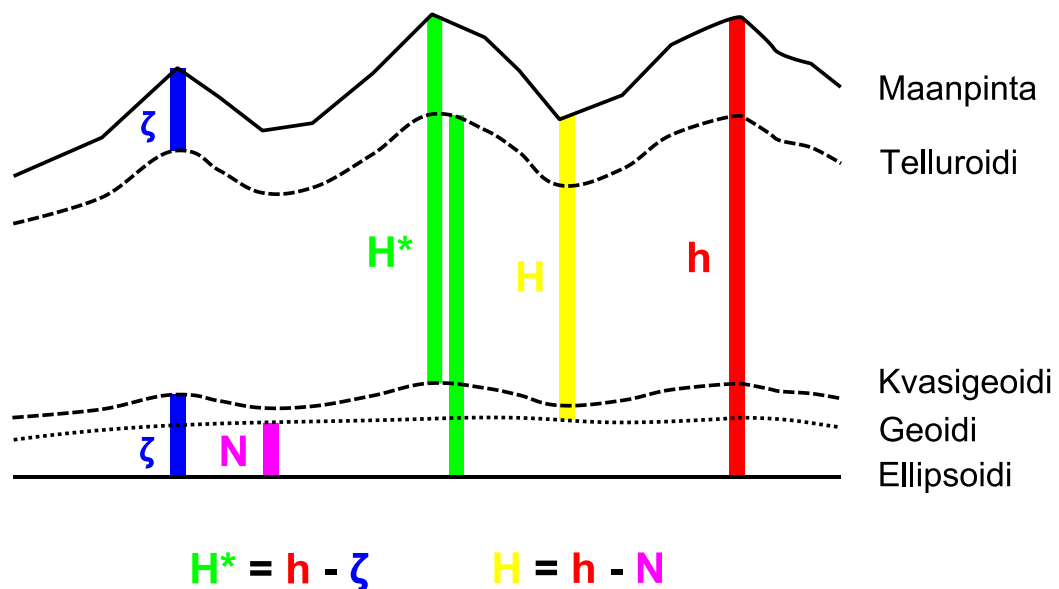
Suomen kolmannen tarkkavaaituksen suoritti Geodeettinen laitos eli nykyinen Maanmittauslaitoksen paikkatietokeskus. Vaaituksella oli pituutta yhteensä 9158 km ja se käsitti 29 suljettua silmukkaa haaralinjoihin sekä 6103 korkeuskiintopistettä keskimäärin 1,5 km välein. Vaaitusverkosta yhteensä 48 % sijoittui rautateille ja 52 % maanteille. Vaaitusverkko on esitetty linjoittain kuvassa 8. N2000-järjestelmää varten tarkkavaaituksen tuloksista laskettiin kansallinen tasoitus, johon otettiin Suomen rajan läheisyydestä lisäksi mukaan Ruotsin ja Norjan tarkkavaaitusten tuloksia, millä varmistettiin valtioiden korkeusjärjestelmien yhteensopivuus. Tasoituksen ajankohdaksi (epookki) valittiin vuosi 2000, mistä juontuu myös korkeusjärjestelmän nimi. Maannousukorjaukset vaihtelevista vaaitusvuosista vuoteen 2000 määritettiin yhteispohjoismaisella maannousumallilla NKG2005LU. (11, s. 7, 9, 19, 20.)



Kuva 9. FIN2005N00-kvasigeoidimallin laskennassa käytetyt GPS- ja vaaituspisteet (mustalla) sekä Suomen kolmas tarkkavaaitusverkko (punaisella).

Satelliittimittausten avulla saatavien ellipsoidisten korkeuksien muuntamiseksi uuden N2000-järjestelmän mukaisia vaaituja korkeuksia vastaaviksi tarvittiin Suomen alueelle uusi geoidimalli, jota ryhdyttiin kehittämään vuonna 2005. Tätä geoidimallia kutsutaan FIN2005N00-kvasigeoidiksi, joka on esitetty kuvassa 3. Mallin laskemiseen käytettiin

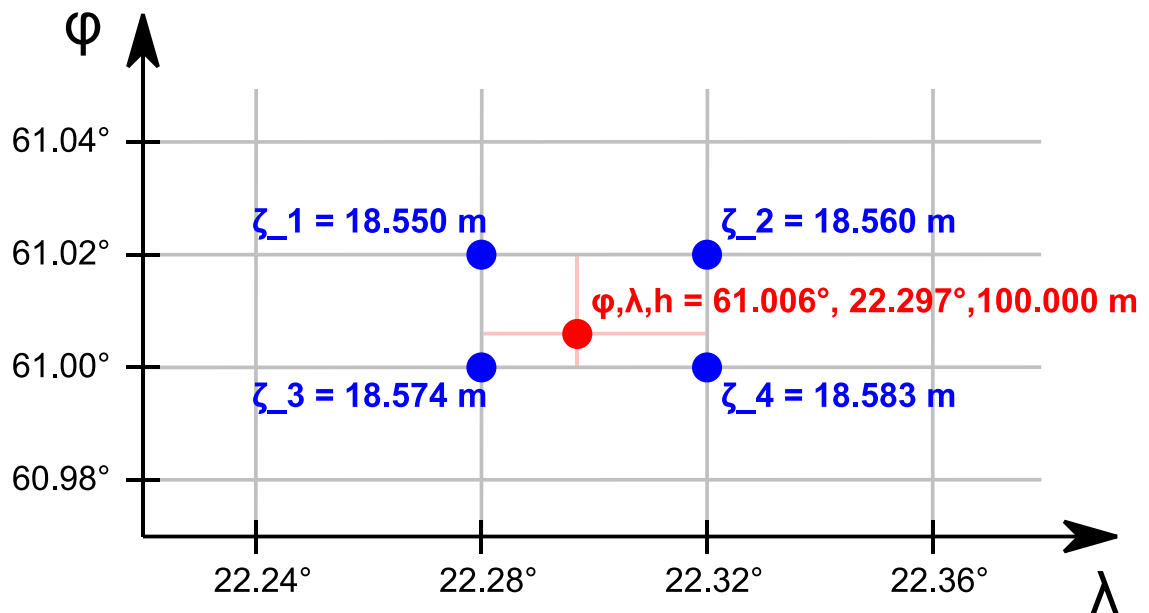
50 GPS- ja vaaituspistettä, joille tunnettiin sekä N2000-järjestelmän mukaiset korkeudet että EUREF-FIN-järjestelmän mukaiset suorakulmaiset koordinaatit, joista esitystapaa muuttamalla saadaan ellipsoidiset korkeudet. GPS-mittaukset olivat peräisin vuoden 2005 EUVN-DA-mittauskampanjasta (30 pistettä) sekä vuosien 1996–1997 EUREF-FIN-mittauskampanjasta (20 pistettä). Pisteiden sijainnit on esitetty kuvassa 9. FIN2005N00-kvasigeoidi määritettiin laskemalla näistä pisteistä pienimmän neliösumman kollokaatiomenetelmällä korjauspinta pohjoismaisesta NKG2004-kvasigeoidista, ja yhdistämällä lopuksi korjauspinta samaiseen NKG2004-kvasigeoidiin. Jonkin matkaa GPS- ja vaaituspisteiden kattaman alueen ulkopuolella korjauspinnan korkeus painuu nolnaan, jolloin kvasigeoidit vastaavat toisiaan. FIN2005N00-kvasigeoidin avulla voidaan muuntaa korkeuksia EUREF-FIN- ja N2000-järjestelmien välillä alle 2 cm tarkkuudella. (5, s. 13–15; 12, s. 77–78, 86.)



Kuva 10. Normaalikorkeudet ja ortometriset korkeudet. Normaalikorkeus (H^*) lasketaan ellipsoidisen korkeuden (h) ja korkeusanomalian (ζ) erotuksena. Ortometrinen korkeus (H) lasketaan ellipsoidisen korkeuden (h) ja geoidin korkeuden (N) erotuksena.

N2000-järjestelmän korkeudet ovat normaalikorkeuksia, kun taas Suomen aiemmissa korkeusjärjestelmissä on käytetty ortometrisia korkeuksia. Tarkkavaaituksesta saatavat ortometriset korkeudet lasketaan jakamalla mitattavan pisteen geopotentiaaliluku pisteen ja geoidin välisellä keskimääräisellä painovoimalla. Normaalikorkeuksien tapauksessa jakajana käytetään normaalipainovoiman keskiarvoa pisteen kohdalla. Ellipsoidisten ja normaalikorkeuksien erotuksia sanotaan korkeusanomalioksi, jotka muodostavat telluroidiksi kutsutun maanpinnan muotoja seurailevan pinnan, joka voidaan

tulkita myös lähes geoidin kaltaiseksi kvasigeoidiksi (kuva 10), joka ei kuitenkaan ole aivan Maan painovoimapotentiaalin todellinen tasa-arvopinta. Suomessa normaalikorkeuksien ja ortometrinen korkeuksien ero on tavallisesti millimetriluokkaa, koska maanpinta on yleensä matalalla. Vuorilla erot voivat kuitenkin olla kymmeniä senttejä ja Lapissakin päästään suurimmillaan noin kahdeksaan senttiin. Merellä normaalikorkeudet ja ortometriset korkeudet ovat samoja. (5, s. 31–33; 7, s. 80–82, 393–397.)



$$\varphi_{\text{norm}} = (\varphi - \varphi_{\text{min}}) / (\varphi_{\text{max}} - \varphi_{\text{min}})$$

$$\lambda_{\text{norm}} = (\lambda - \lambda_{\text{min}}) / (\lambda_{\text{max}} - \lambda_{\text{min}})$$

$$\zeta = \varphi_{\text{norm}} * (1 - \lambda_{\text{norm}}) * \zeta_1 + \varphi_{\text{norm}} * \lambda_{\text{norm}} * \zeta_2 + \dots \\ (1 - \varphi_{\text{norm}}) * (1 - \lambda_{\text{norm}}) * \zeta_3 + (1 - \varphi_{\text{norm}}) * \lambda_{\text{norm}} * \zeta_4$$

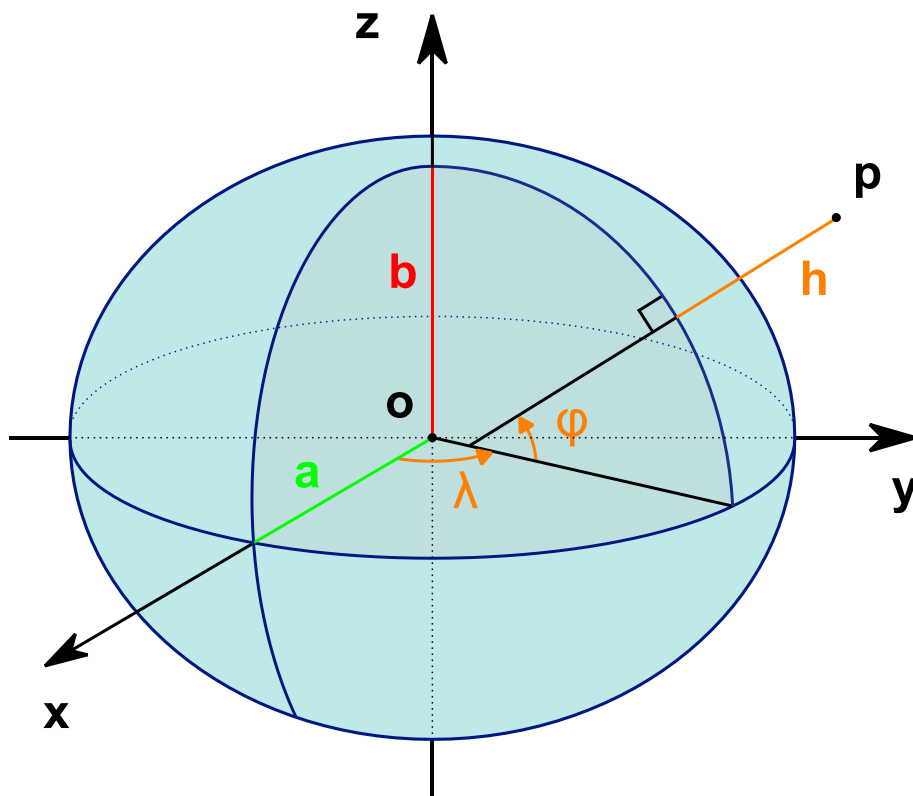
$$H^* = h - \zeta = 81.4292... \text{ m}$$

Kuva 11. Ellipsoidisten korkeuksien muuntaminen vaaittuja korkeuksia vastaaviksi. Pistein korkeus GRS80-ellipsoidin pinnalta (h) N2000-järjestelmän mukaisiksi normaalikorkeuksiksi (H*) lasketaan bilineaarisen interpoloinnin avulla.

FIN2005N00-kvasigeoidi on saatavilla Maanmittauslaitoksen verkkosivuilta korkeushilana, joka sisältää tiedot kvasigeoidin korkeuksista maantieteellisin koordinaatein erotettuihin väleihin. Mitatun pisteen ellipsoidinen korkeus muunnetaan normaalikorkeudeksi etsimällä neljä sitä ympäröivää hilapistettä ja laskemalla tämän jälkeen mitatulle pisteelle kvasigeoidin korkeus bilineaarisella interpoloinnilla. Normaalikorkeus saadaan ellipsoidisen korkeuden ja kvasigeoidin korkeuden erotuksena (kuva 11). (5, s. 20, 34.)

2.4 Muunnokset ja konversiot

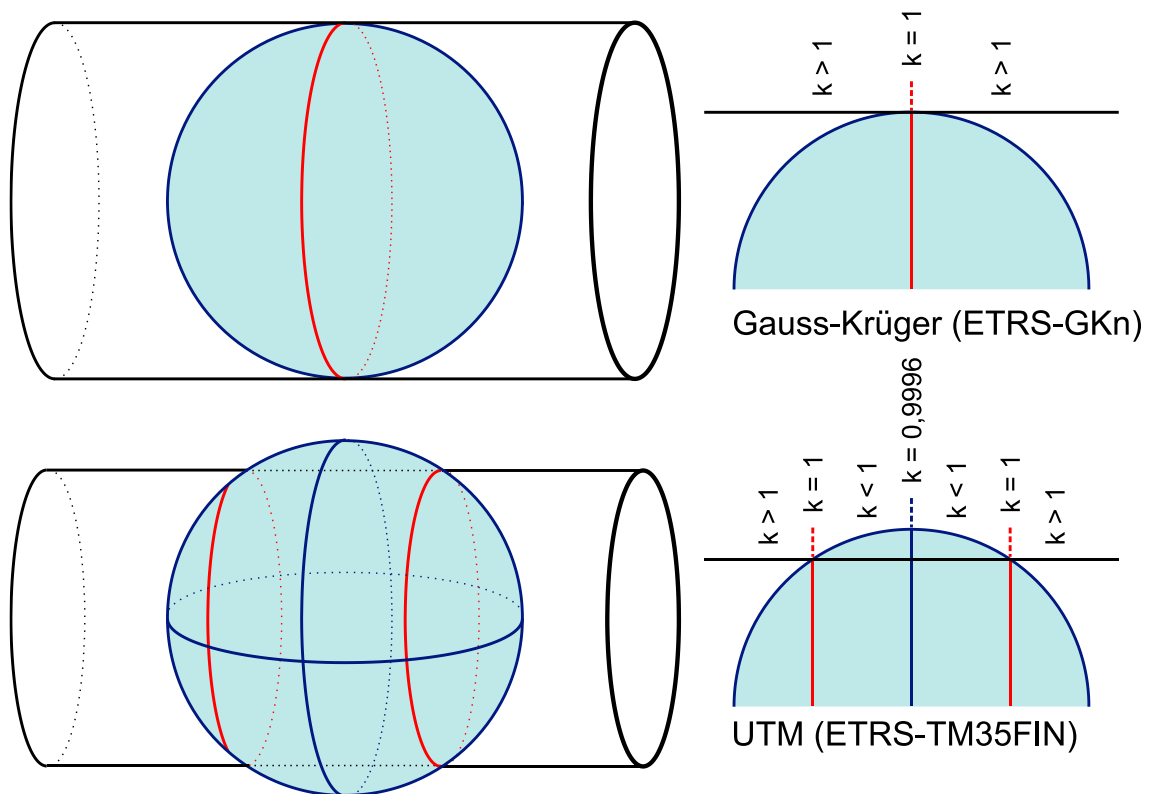
Satelliittipaikannuksella saatavat suorakulmaiset koordinaatit ovat mittaustuloksia tarkastelevan käyttäjän näkökulmasta varsin epäkäytännöllisiä, koska niiden sijoittumista maaston suhteen on vaikea hahmottaa. Tämän vuoksi koordinaatteja onkin tarpeen muuttaa havainnollisempaan muotoon, käytännössä maantieteellisiksi koordinaateiksi tai tasokoordinaateiksi. Muutosten tekemiseen on olemassa kaksi toisistaan poikkeavaa tapaa, koordinaattikonversio ja koordinaattimuunnos. Näistä koordinaattikonversio



Kuva 12. Geodeettiset koordinaatit suorakulmaisessa koordinaatistossa. Mitatun pisteen (p) geodeettiset koordinaatit koostuvat maantieteellisistä koordinaateista eli leveyskulmasta (ϕ) ja pituuskulmasta (λ) sekä ellipsoidisesta korkeudesta (h). Isoakselin puolikas (a) ja pikkuakselin puolikas (b) määrittävät vertausellipsoidin koon.

suoritetaan kahden samaan datumiin perustuvan koordinaatiston välillä, kun taas koordinaattimuunnos tehdään kahden eri datumiin perustuvan koordinaatiston välillä. Datumilla tarkoitetaan tässä niitä koordinaattijärjestelmän parametreja, joilla koordinaatisto kiinnitetään Maan pintaan. ETRS89-järjestelmän kanssa Suomessa käytettäviä koordinaattikonversioita ovat esimerkiksi suorakulmaisten EUREF-FIN-koordinaattien konvertoiminen geodeettisiksi EUREF-FIN-koordinaateiksi sekä geodeettisiin koordi-

naatteihin sisältyvien maantieteellisten koordinaattien projisoiminen ETRS-TM35FIN-tasokoordinaateiksi. Koordinaattimuunnoksia taas ovat esimerkiksi suorakulmaisten EUREF-FIN-koordinaattien muuntaminen suorakulmaisiksi KKJ-koordinaateiksi kolmiulotteisella yhdenmuotoisuusmuunnoksella eli 7-parametrisella Helmert-muunnoksella ja ETRS-TM35FIN-tasokoordinaattien muuntaminen YKJ-tasokoordinaateiksi kolmiulotteisella affiinisella muunnoksella. Koordinaattimuunnosten muunnosparametrit on aina



Kuva 13. Gauss-Krüger- ja UTM-karttaprojektioiden periaate. Gauss-Krüger on poikittainen lieriöprojektiio, jonka avulla luodaan ETRS-GK n -tasokoordinaatistot. Siinä projektioitasona toimiva lieriö sivuaa GRS80-vertausellipsoidia projektioikaistan keskimeridiaanina toimivaa pituuspiiriä pitkin. UTM-karttaprojektioilla luodaan ETRS-TM35FIN-tasokoordinaatisto. Se vastaa muuten Gauss-Krüger-projektiota, mutta nyt lieriö leikkaa vertausellipsoidia. Sivuamis- ja leikkauskohdat on merkitty kuvaan punaisella. Mittakaavakerroin (k) kertoo, miten tasokoordinaatistojen mittakaavavirhe riippuu lieriön sijoitumisesta vertausellipsoidin suhteen.

laskettava molemmissa koordinaatistoissa mitattujen samojen kiintopisteiden avulla. Tämän vuoksi koordinaattimuunnoksiin sisältyy aina muunnosvirhettä, kun taas koordinaattikonversiot ovat muunnosvirheestä vapaita. On tärkeää huomata, että koordinaattimuunnokset suoritetaan aina kahden samantyyppisen koordinaatiston (esim. tasokoordinaatistot) välillä, kun taas koordinaattikonversiot voidaan tehdä myös keskenään erityyppisten koordinaatistojen välillä, kuten lähes aina tehdäänkin. (4, s. 35–38, 59.)

Konversio-ohjelmaan sisältyy nimensä mukaisesti koordinaattikonversioita sekä korkeusmuunnokset ellipsoidisten EUREF-FIN-korkeuksien ja N2000-järjestelmän normaalikorkeuksien välillä, mutta ei lainkaan edellä tarkoitettuja koordinaattimuunnoksia. Ohjelmassa käytettyjä koordinaattikonversioita on kolmea tyyppiä: suorakulmaisen ja geodeettisen koordinaatiston väliset konversiot, karttaprojektio geodeettisen koordinaatiston ja tasokoordinaatistojen välillä sekä projektiotason vaihdot tasokoordinaatistojen ETRS-TM35FIN ja ETRS-GK27 välillä.

Taulukko 1. ETRS89-järjestelmän kanssa Suomessa käytettävien tasokoordinaatistojen ominaisuuksia.

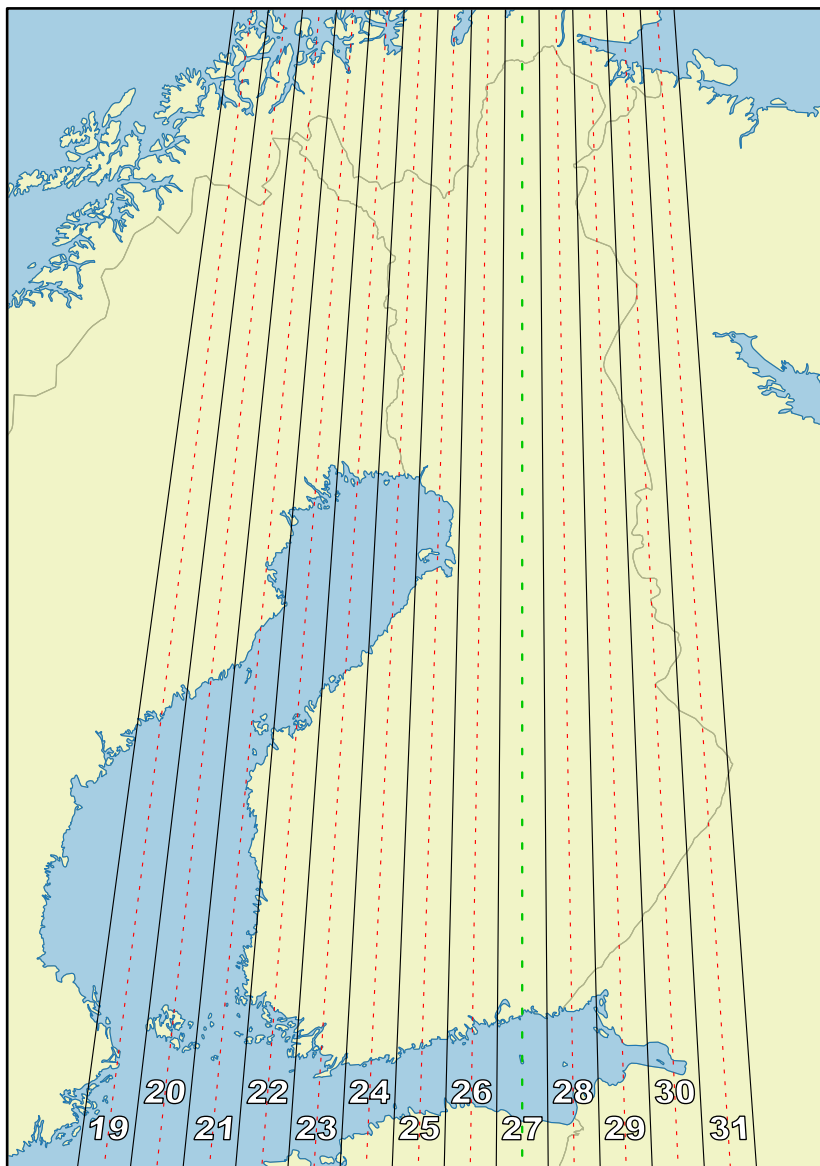
	ETRS-TM35FIN	ETRS-GK _n
Karttaprojektio	UTM	Gauss-Krüger
Datumi	ETRS89	ETRS89
Vertausellipsoidi	GRS80	GRS80
Keskimeridiaani	27°	19°...31°
Meridiaanikaistoja	1	13
Kaistan leveys (suhteessa keskimeridiaaniin)	n. 13° (-8°...+5°)	1° / tarkoituksenmukainen
Itäkoordinaatin arvo keskimeridiaanilla	500 000 m	n 500 000 m, missä n on keskimeridiaanin asteluku
Mittakaava keskimeridiaanilla	0,9996	1,000

Suorakulmaisten EUREF-FIN-koordinaattien (X,Y,Z) konvertoinnissa geodeettisiksi EUREF-FIN-koordinaateiksi (φ, λ, h) on käytännössä kysymys vain koordinaattien esitystavan muutoksesta. Vertausellipsoidin keskipiste on suorakulmaisen koordinaatiston keskipisteessä, ja molemmat sijaitsevat Maan massakeskipisteessä. Samoin vertausellipsoidin pikkuakseli on suorakulmaisen koordinaatiston Z-akselin suuntainen, ja molemmat ovat Maan pyörähdysakselin suuntaisia. Toisin sanoen mitattu piste sijaitsee kolmiulotteisessa tilassa samalla kohdalla (kuva 12). (4, s. 19–20; 7, s. 43–45.)

Maantieteelliset EUREF-FIN-koordinaatit puolestaan projisoidaan ETRS-GK_n-tasokoordinaateiksi (N,E) Gauss-Krüger-karttaprojektion avulla, ja ETRS-TM35FIN-tasokoordinaatit (N,E) saadaan käyttämällä UTM-projektiota. Molemmat ovat poikittaisia lieriöprojektiota, jotka poikkeavat toisistaan siten, että ensimmäisessä projektiossa toimiva lieriö sivuaa vertausellipsoidia projektioikaistan keskimeridiaanilla ja jälkimmäisessä lieriö taas leikkaa ellipsoidia (kuva 13). (4, s. 7–8, 36–37; 7, s. 54–56.)

Karttaprojektioissa vertausellipsoidin pinnalta lieriön pinnalle syntyy mittakaavavirhettä, jonka suuruus riippuu mitatun pisteen etäisyydestä pituuspiiriin, jolla ellipsoidi ja lieriö sivuavat tai leikkaavat toisiaan. Itäkoordinaatin arvoksi tasokoordinaatistojen

keskimeridiaanilla annetaan suuri ennalta sovittu luku, ns. valeitä tai itäsiirtymä, jotta vältetään negatiivisten lukujen käytöltä. Mitatun pisteen itäkoordinaatin arvo riippuu sen etäisyydestä keskimeridiaaniin. Pohjoiskoordinaatin arvo taas on pisteen etäisyys päiväntasaajasta lieriön pintaa pitkin. Konversio ETRS-TM35FIN- ja ETRS-GK27-koordinaatistojen välillä voidaan suorittaa myös yksinkertaisemmilla laskukaavoilla käyttämättä karttaprojektioita. Koska molemmilla on sama keskimeridiaani, 27. pituuspiiri (kuva 14), riittää pelkkä projektiotason vaihtaminen. Tasokoordinaatistojen ominaisuuksia esitellään taulukossa 1. (3, s. 143–151; 4, s. 36–37; 7, s. 54–56.)

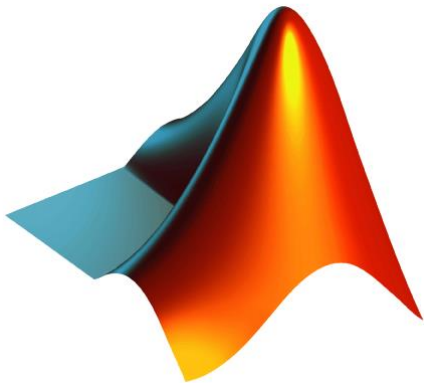


Kuva 14. EUREF-FIN-koordinaatistoon liittyvät karttaprojektiot. ETRS-GKn-tasokoordinaatiston projektiokaistojen keskimeridiaanit on esitetty punaisilla katkoviivoilla ja kaistojen rajat yhtenäisillä mustilla viivoilla. Vihreä katkoviiva 27. pituuspiirillä kuvaa sekä ETRS-GK27- että ETRS-TM35FIN-tasokoordinaatiston keskimeridiaania.

3 Johdatus Matlab-ohjelmointiin

3.1 Ohjelmiston käyttötarkoitus

Matlab (Matrix Laboratory) on suosittu numeerisessa laskennassa käytettävä ohjelmisto ja ohjelmointikieli, jota hyödynnetään erityisesti insinööri- ja luonnontieteiden parissa. Ohjelmisto on alkujaan suunniteltu lineaarialgebrassa tarvittavaa matriisilaskentaa varten, mutta se on sittemmin kasvanut erittäin kattavaksi järjestelmäksi, jonka avulla voidaan ratkoa lähes mitä tahansa tekniikan kehittämiseen liittyviä ongelmia. Yksin Matlabin perusversio ilman laajennusosia (toolkit) käsittää yli tuhat sisäänrakennettua funktiota ja laajennusosien avulla sen kattamien toimintojen piiriä voidaan laajentaa lukuisien erityisalojen tai esimerkiksi symbolisen laskennan alueelle. (13, s. 1; 14, s. 3.)

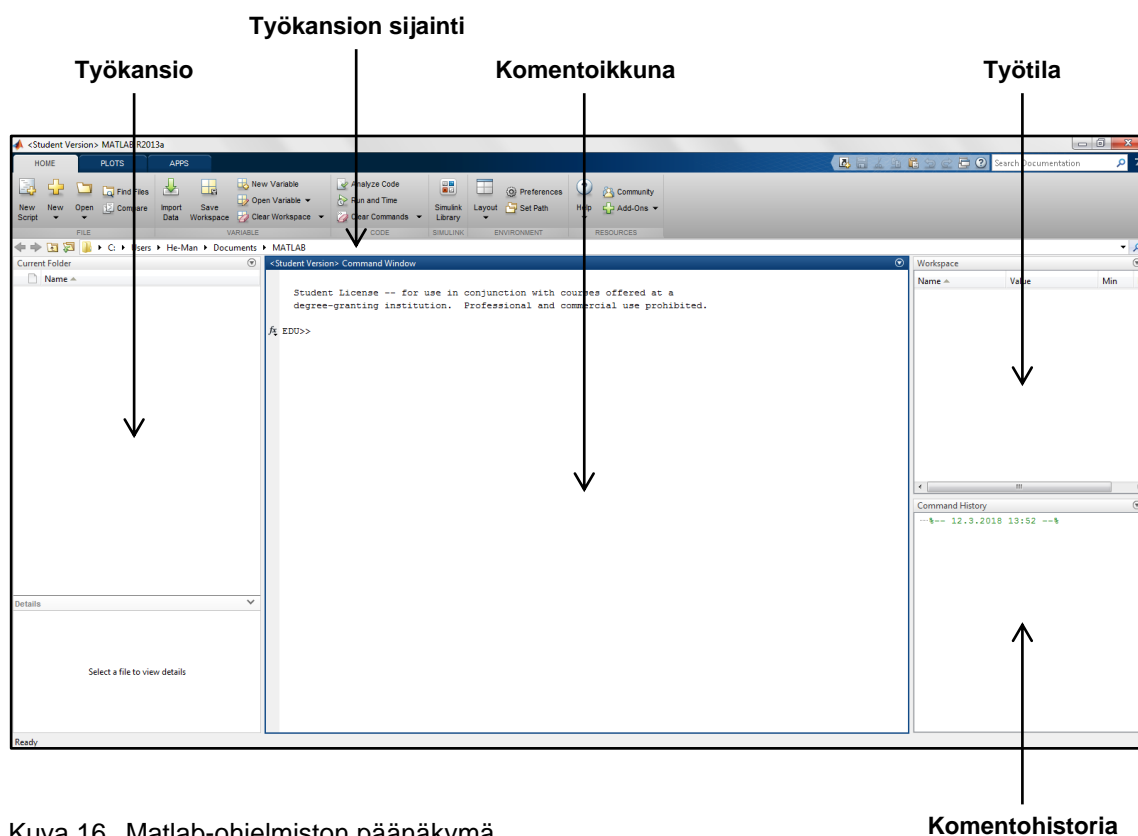


Kuva 15. Matlab-ohjelmistoa kehittävä Math Works -yhtiön liikemerkki.

Tämän osion tarkoitus on tehdä tutuksi Matlab-ohjelmiston keskeisiä toimintoja siltä osin, kun niitä tarvitaan konversio-ohjelman toiminnan ymmärtämiseen. Tästä syystä useita ohjelmiston perusominaisuuksia, kuten kuvaajien piirto, sivuutetaan kokonaan. Kattavaa käyttöopasta etsivän on suositeltavaa tutustua Math Works -yhtiön (kuva 15) julkaisemaan *MATLAB Primer* -verkkodokumenttiin tai jonkin kolmannen osapuolen kirjoittamaan opaskirjaan, kuten *Essential Matlab* (Hahn, Valentine) tai *Matlab Programming for Engineers* (Chapman), joita on hyödynnetty tämän tekstin kirjoittamisessa. Muita käyttökelpoisia resursseja ovat ainakin ohjelmointiin liittyvien kysymysten ympärille rakennettu verkkosivusto Stack Exchange sekä huomattavan määrän Matlab-aiheisia opasvideoita sisältävä YouTube. (13; 14; 15.)

3.2 Päänäkymä

Windows-ympäristössä Matlab voidaan avata työpöydällä tai muualla graafisessa käyttöliittymässä sijaitsevasta kuvakkeesta sekä kirjoittamalla komentoriville (cmd) teksti `matlab` ja painamalla enter-näppäintä. Kun ohjelmisto on käynnistynyt, näytölle avautuu alla olevaa (kuva 16) vastaava näkymä, jonka tarkemmat yksityiskohdat riippuvat ohjelmiston versiosta, joka tässä työssä on R2013a.

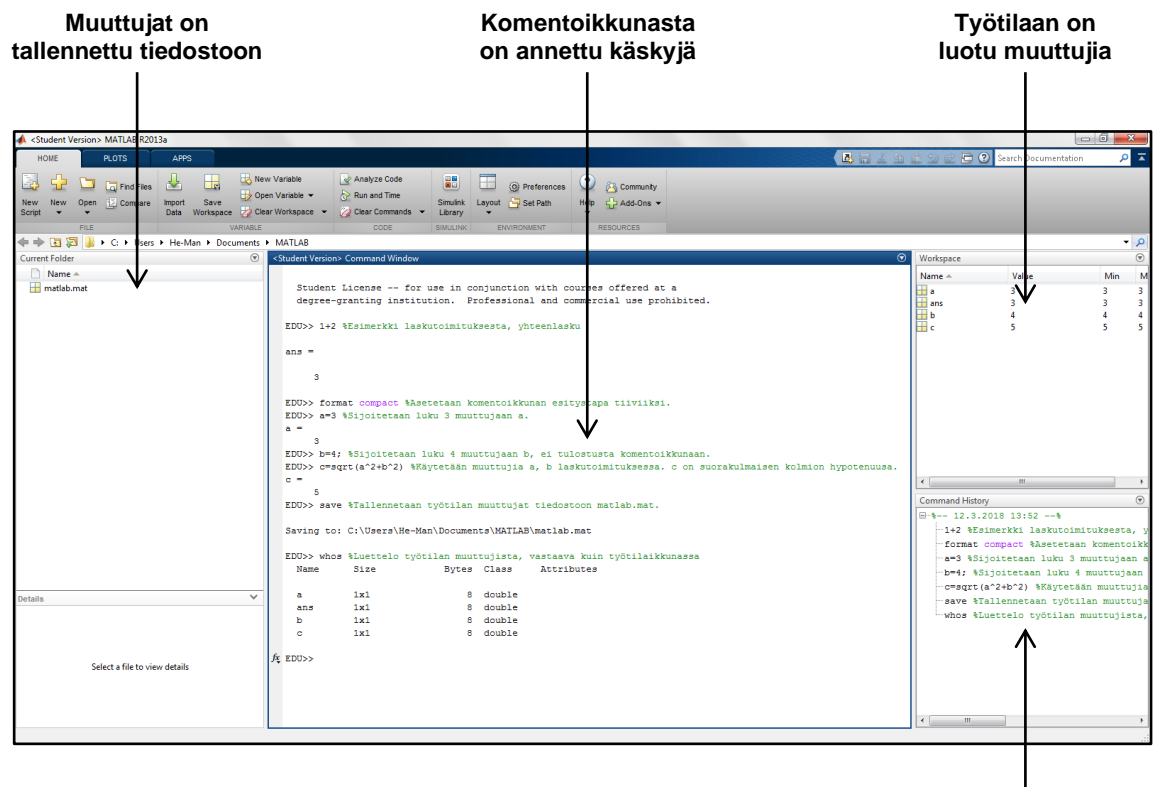


Kuva 16. Matlab-ohjelmiston päänäkymä.

Keskellä päänäkymää on komentoikkuna (command window), josta ohjelmistolle voidaan antaa käskyjä, kuten suoritettavia laskutoimituksia. Käskyt kirjoitetaan komentoriville (command line), joka on tunnistettavissa merkinnästä `"EDU >>"`. Merkintä on nimeltään komentokehote (prompt). Teksti `"EDU"` viittaa opiskelijalisenssillä hankittuun ohjelmistoon, yleensä komentokehote esiintyy lyhyemmässä muodossa `">>"`. (13, s 4.)

Päänäkymän vasemmassa laidassa sijaitsee työkansioikkuna ja heti sen yläpuolella on nähtävissä työkansion (current folder) sijainti Windowsin kansiorakenteessa. Oletuksena työkansio on nimeltään MATLAB ja se luodaan Windowsissa käyttäjän Tiedostokansion alle ohjelmiston ensimmäisellä käynnistyskerralla. Työkansioon olennaisesti liittyvä käsite on hakupolku (search path), jolla tarkoitetaan niiden kansioden joukkoa,

joista Matlab etsii ohjelmiston käyttämiä tiedostoja. Kun käyttäjä antaa Matlabille käskyn, joka sisältää tiedoston nimen, on työkansio ensimmäinen sijainti, josta tiedostoa etsitään. Tästä seuraa, että työkansion sisältäessä nimeltään etsittävää vastaavan tiedoston, muualla hakupolussa sijaitsevia samannimisiä tiedostoja ei huomioida. Työkansio on myös paikka, johon Matlab oletuksena tallentaa käyttäjän luomat tiedostot, jos niille ei erikseen anneta toista sijaintia. (13, s. 15–17; 14, s. 12, 29.)



Kuva 17. Suoritettujen käskyjen vaikutus Matlabin päänäkymään.

Annetut käskyt tallentuvat komentohistoriaan

Päänäkymän oikeassa laidassa ovat työtila- ja komentohistoriaikkunat. Käsitteellä työtila (workspace) viitataan tässä yhteydessä Matlabin hallitsemaan osaan tietokoneen muistista. Työtilaikkunassa näkyvät kaikki ohjelmiston komentoikkunasta suoritettujen käskyjen seurauksena syntyneet muuttujat (ks. 3.3). Työtila tyhjenee, kun ohjelmisto sammutetaan. Komentohistoria (command history) taas on luettelo ohjelmistolle viiemeksi annetuista käskyistä. Luettelo ei tyhjene ohjelmistoa sammutettaessa, vaan sisältää myös ohjelmiston aiemmilla käyttökertoilla annetut käskyt siitä saakka, kun käyttäjä on itse sen edellisen kerran tyhjentänyt tai komentohistorian tallennustiedoston (history.m) kokorajoitus tulee täyteen. (13, s. 6–7, 11–12.)

Kun Matlab on ensimmäisen kerran käynnistetty, ovat päänäkymän ikkunat vielä tyhjillään (kuva 16). Yllä on esitetty (kuva 17) miten komentoikkunasta annettujen käskyjen

vaikutus ilmenee ohjelmiston päänäkymässä. Komentoikkunaan on ensin syötetty lauseke $1+2$, jonka arvo on tulostunut samaan ikkunaan. Tulosteesta nähdään, että vastaus on sijoitettu muuttujaan "ans". Näin tapahtuu aina, kun käyttäjä ei erikseen sijoita lukuarvoa, lauseketta tai funktion (ks. 3.5) tulostetta johonkin itse valitsemaansa muuttujaan. Laskutoimituksen jälkeen komentoikkunan esitystapa on muutettu tiiviimmäksi komennolla "format compact", joka poistaa ikkunasta tarpeettomat rivivälit. Vastaavilla komennoilla (ks. 3.4) voidaan vaikuttaa mitä moninaisimpiin asioihin Matlabin ulkoasussa ja toiminnassa sekä saada tietoja ohjelmistosta ja sen käyttämisestä tiedostoista. (13. s 17, 40–41; 14, s. 6.)

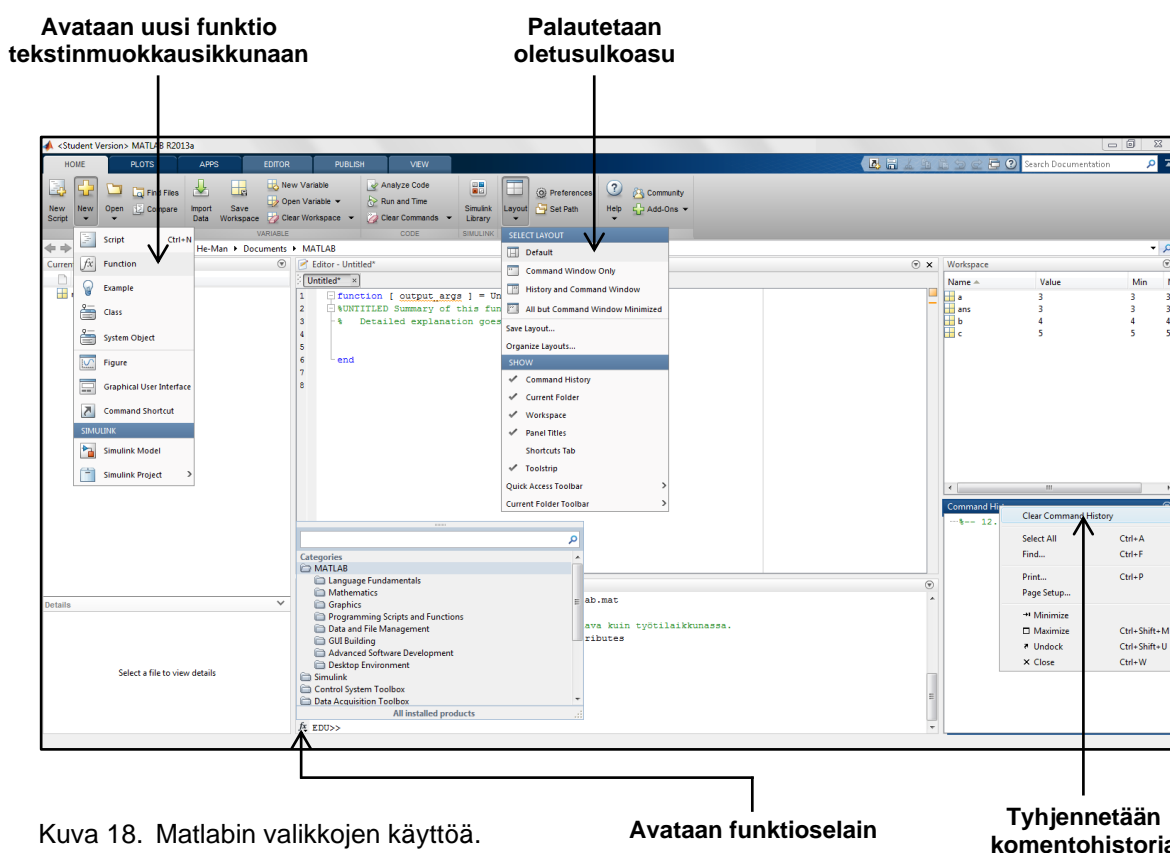
Seuraavilla riveillä muuttujiin "a" ja "b" on sijoitettu luvut 3 ja 4. Sijoitusoperaattorina käytetään yhtäsuuruusmerkkiä "=". On tärkeää huomata, että Matlab-ohjelmoinnissa yhtäsuuruusmerkkinä puolestaan toimii merkintä "==", jota tarvitaan erityisesti ehtolauseiden (ks. 3.6) kanssa. Muuttujista "a" ja "b" ainoastaan "a" on lukuarvoineen tulostunut komentoikkunaan. Tämä johtuu siitä, että sijoitettaessa lukua 4 muuttujaan "b", on käskyn loppuun lisätty puolipiste ";" Tulostuksen estäminen puolipisteellä on hyödyllistä, kun halutaan säästää tilaa näytöllä ja pitää komentoikkuna paremmin luettavana. Muuttujia voidaan käyttää lausekkeissa samoin kuin lukuarvoja. Esimerkkinä tästä on muuttujia "a" ja "b" käyttäen ratkaistu suorakulmaisen kolmion hypotenuusan pituus ja sijoitettu vastaus muuttujaan "c". Lauseke sisältää Matlabin sisäänrakennetun funktion "sqrt()" eli neliöjuurifunktion. (13, s. 11; 14, s. 7–8.)

Kaikki komentoikkunasta luodut muuttujat ovat tämän jälkeen nähtävissä päänäkymän oikeassa yläkulmassa sijaitsevasta työtilaikkunasta. Kun komentoriville kirjoitetaan käsky "save", tallentuvat muuttujat arvoineen työkansioon tiedostoon matlab.mat, mikä voidaan havaita myös päänäkymän vasemmassa laidassa olevasta työkansioikkunasta. Viimeisenä komentoikkunaan on annettu käsky "whos", joka tulostaa luettelon kaikista työtilan muuttujista. Luettelo vastaa työtilaikkunassa näkyvää. Matlabille onkin tyypillistä, että ohjelmistoa voidaan hallita sekä komentoikkunasta annetuilla käskyillä että hiirellä graafisesta käyttöliittymästä. (13, s. 11, 43, 362; 14, s. 35.)

Päänäkymän oikeassa alakulmassa sijaitsevasta komentohistoriaikkunasta voidaan havaita, että kaikki komentoikkunasta annetut käskyt ovat tallentuneet komentohistoriaan. Komentoikkunaan kirjoitettujen käskyjen perässä olevat %-merkillä alkavat vihreät tekstit ovat kommentteja, jotka eivät vaikuta ohjelmiston toimintaan, vaan ne on tarkoitettu muistiinpanoiksi, joita lukemalla käyttäjä voi jälkikäteen päätellä, mikä oli annetun

käskyn tarkoitus. Kommentteja käytetään tavallisesti skriptien ja funktioiden kanssa (ks. 3.5), mutta niistä on hyötyä myös suoraan komentoikkunasta annettujen käskyjen yhteydessä, sillä myös kommentit tallentuvat komentohistoriaan. (13, s. 6–7; 14, s. 15.)

Komentoikkunaan syötettävien käskyjen ohella Matlabin tehokas käyttö edellyttää sen valikoiden tuntemista. Alla (kuva 18) on esitetty muutamia ohjelmiston käytön aloittamisen kannalta tärkeimpiä. Päänäkymän vasempaan yläkulmaan on aukaistu valikko, josta voidaan edelleen avata tekstinmuokkausikkunaan (editor) esimerkiksi uusi skripti tai funktio. Kuvassa uuden funktion sisältävä tekstinmuokkausikkuna näkyy päänäkymän keskellä taustalla. (13, s. 7–8; 14, s. 14.)



Kuva 18. Matlabin valikkojen käyttöä.

Päänäkymän ulkoasu (layout) voi muuttaa sulkemalla ikkunoita ja avaamalla uusia tarpeen mukaan sekä raahaamalla niitä eri puolille näkymää yläpalkista tarttumalla. Oletusulkoasun (default layout) voi palauttaa keskelle päänäkymän yläosaa avatusta valikosta. Samasta valikosta voi myös ottaa käyttöön muita valmiiksi määritellyjä ulkoasuja, vaikuttaa siihen mitkä ikkunat ovat auki ja tallentaa itse luomiaan ulkoasuja.

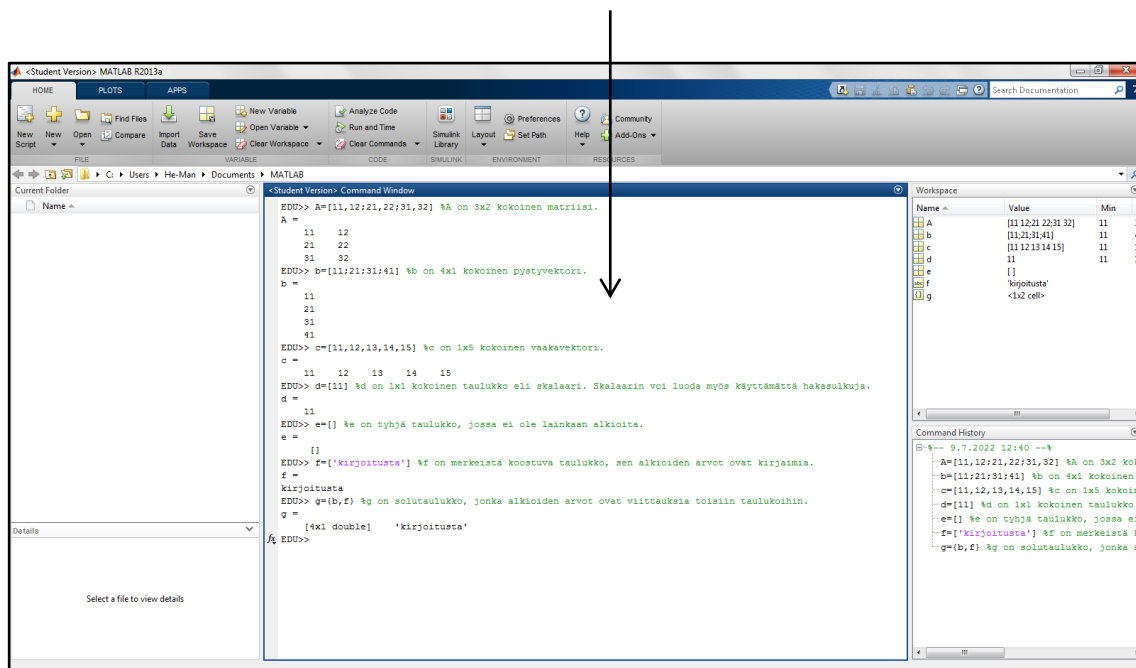
Komentoikkunan vasemmassa reunassa, komentokehotteen vieressä on merkintä "fx", josta voidaan avata Matlabin funktioselain (function browser). Kuvassa se näkyy kes-

kellä alhaalla. Funktiotiedoston avulla käyttäjä voi etsiä tarvitsemaansa funktiota poistumatta komento- tai tekstinmuokkausikkunasta. Käytettäessä ohjelmiston tekstinmuokkausikkunaa, funktiotiedosto löytyy ohjelmiston ylävalikosta välilehden ”editor” alta.

3.3 Muuttujat ja taulukot

Tietojenkäsittelyn perusyksikkö Matlab-ohjelmoinnissa on nimeltään taulukko (array), jolla tarkoitetaan luku- tai muista data-arvoista koostuvaa yksikäsitteisesti nimettyä koelmaa, joka on järjestetty riveiksi ja sarakkeiksi. Kaikki yleisimmin käytössä olevat taulukot ovat muotoa **RxS**, jossa R merkitsee rivien ja S sarakkeiden lukumäärää. Rivien ja sarakkeiden järjestysluvuilla yksilöidään taulukon alkioita (elementtejä), jotka sisältävät yksittäiset luku- tai muut arvot. Muuttuja (variable) on tietokoneen muistissa oleva käyttäjän nimeämä alue, joka sisältää taulukon. (13, s. 21–22.)

Komentoikkunassa annetuilla
käskyillä on luotu taulukoita

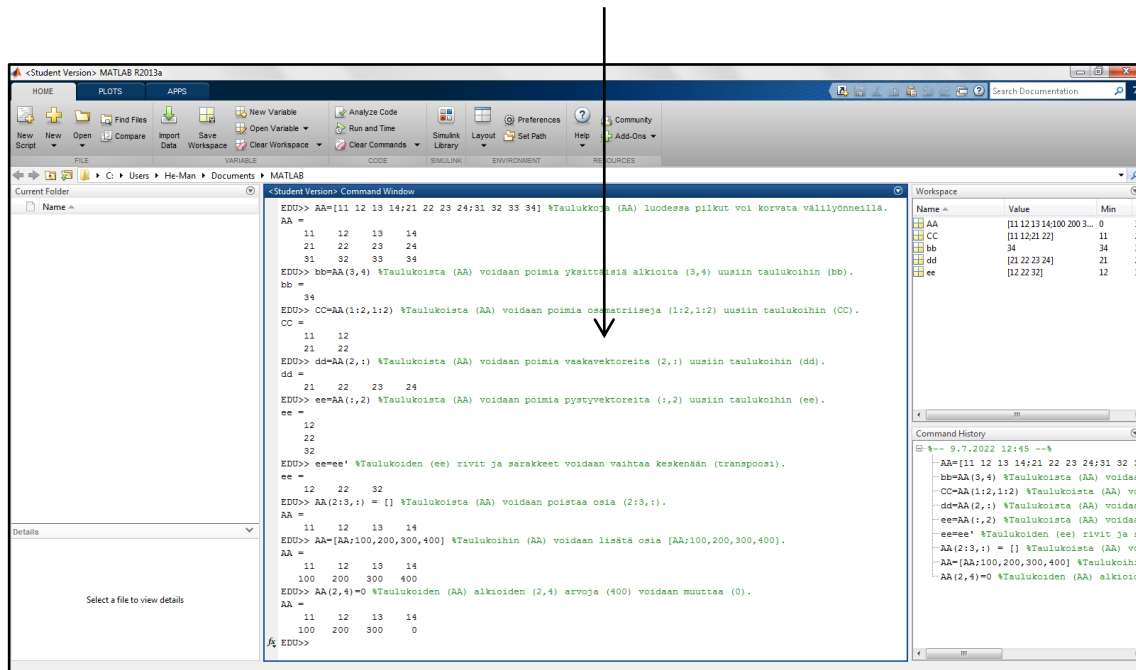


Kuva 19. Erialaisten taulukoiden luominen Matlabissa.

Taulukot voidaan edelleen jaotella matriiseiksi (matrix), vektoreiksi (vector), skalaareiksi (scalar) ja tyhjiksi taulukoiksi (empty array). Matriisi on taulukko, jossa on vähintään kaksi riviä ja saraketta. Vektoreita on kahdenlaisia, pystyvektoreita (column vector) ja vaakavektoreita (row vector). Ensimmäisessä on vähintään kaksi riviä ja yksi sarake,

toisessa on yksi rivi ja vähintään kaksi saraketta. Skalaarissa on yksi rivi ja yksi sarake eli vain yksi alkio. Tyhjässä taulukossa ei ole yhtään alkioita, sitä käytettäessä taulukoon on yleensä tarkoitus sijoittaa myöhemmin lisää skalaareita, vektoreita tai matriiseja. Edellä mainittujen lisäksi on olemassa myös taulukoita, joilla on kolme tai useampia ulottuvuuksia, mutta ne eivät kuulu tämän työn aihepiiriin. Taulukoiden luominen ohjelmiston komentoikkunassa on esitetty kuvassa 19. (13, s. 21–22.)

**Komentoikkunassa annetuilla
käskyillä on muokattu taulukoita**



Kuva 20. Taulukoiden käsittely ja muokkaaminen Matlabissa.

Matlabissa on taulukoihin kohdistuvaan tiedonhakuun ja muokkaamiseen erittäin monipuoliset toiminnot, joista kaikkein yleisimpiä on esitetty kuvassa 20. Konversio-ohjelmassa on toiminnoista käytetty yksittäisten alkioiden poimintaa taulukosta ja kokonaisen rivin eli vaakavektorin lisäämistä taulukkoon. (13, s. 35–37; 14, s. 1.10–1.12.)

Ohjelmistossa on taulukoiden data-arvoja varten useita erilaisia datatyyppäjä. Näistä yleisimmät ovat lukuarvoille oletuksena käytettävä ”double”, merkeille kuten kirjaimille oletuksena käytettävä ”char” ja looginen (tosi/epätosi) datatyyppi ”logical”. Yksi taulukko ei kuitenkaan voi sisältää datatyypeiltään erilaisia arvoja. Tähän tarvitaan solutaulukoita (cell array), jonka alkioita ovat soluja, eräänlaisia säiliöitä, joiden arvot viittaavat toisiin taulukoihin. Konversio-ohjelmassa solutaulukkoa on käytetty koordinaattitietoja ohjelmaan luettaessa. (13, s. 323–325; 15, s. 2.28–2.30.)

3.4 Taulukko- ja matriisioperaatiot

Laskutoimituksia eli aritmeettisiä operaatioita (arithmetic operation) voidaan Matlabissa suorittaa kahdella eri tavalla, joita ovat taulukko-operaatiot (array operation) ja matriisioperaatiot (matrix operation). Ne eroavat toisistaan siten, että taulukko-operaatiot suoritetaan kahden taulukon samalla rivillä ja sarakkeessa sijaitsevien alkioiden kesken. Jos yksi taulukoista on skalaari, käytetään sitä erikseen jokaisen toisen taulukon alkion kanssa. Matriisioperaatiot taas suoritetaan käyttäen tavanomaisia lineaarialgebran laskusääntöjä. Mikäli laskutoimitus suoritetaan taulukoille, jotka ovat molemmat skalaareja, tuottavat taulukko- ja matriisioperaatiot saman tuloksen (kuva 21). Yhteen- ja vähennyslaskulle ei ole erikseen matriisioperaatioita, koska ne tuottaisivat aina saman tuloksen kuin taulukko-operaatiot, riippumatta taulukon koosta. (13, s. 46–47.)

Komentoikkunassa on vertailtu taulukko- ja matriisioperaatioita skalaareilla

```

EDD> a=3 %Sijoitetaan skalaari 3 muuttujan a.
a =
     3
EDD> b=5 %Sijoitetaan skalaari 5 muuttujan b.
b =
     5
EDD> c=a.*b %Skalaarien a ja b tulo alkiittain...
c =
    15
EDD> c=a*b %...on sama kuin a:n ja b:n matriisitulo.
c =
    15
EDD> d=a./b %Skalaarien a ja b oikealta jako alkiittain...
d =
    0.6000
EDD> d=a/b %...on sama kuin a:n ja b:n käänteismatriisin tulo.
d =
    0.6000
EDD> e=a\b %Skalaarien a ja b vasemmalta jako alkiittain...
e =
    1.6667
EDD> e=a\b %...on sama kuin a:n käänteismatriisin ja b:n tulo.
e =
    1.6667
EDD> f=a.^b %Skalaari a potenssiin skalaari b alkiittain...
f =
    243
EDD> f=a^b %...on sama kuin a matriisipotenssiin b.
f =
    243
EDD>
  
```

Name	Value	Min	Max
a	3	3	3
b	5	5	5
c	15	15	15
d	0.6000	0.6000	0.6000
e	1.6667	1.6667	1.6667
f	243	243	243

Kuva 21. Taulukko- ja matriisioperaatiot tuottavat skalaareilla saman tuloksen.

Taulukko- ja matriisioperaatiot erotetaan toisistaan pisteoperaattorilla (dot operator) eli pisteellä "." siten, että matriisioperaatioita merkitään tavanomaisilla aritmeettisillä operaattoreilla, joiden eteen taas taulukko-operaatioissa sijoitetaan piste. Yhteen- ja vähennyslaskun tapauksessa pistettä ei käytetä, koska taulukko-operaatioita ei tarvitse erottaa vastaavista matriisioperaatioista, joita ei ole olemassa. Kaikki Matlabin taulukko- ja matriisioperaatiot on esitetty kuvassa 23. (13, s. 46–47, 49.)

Sen lisäksi, että skalaareilla suoritettavat taulukko- ja matriisioperaatiot tuottavat saman tuloksen, saadaan operaatioilla sama tulos silloin, kun skalaaria suurempaa taulukkoa kerrotaan tai jaetaan skalaarilla. Muissa tapauksissa taulukko- ja matriisioperaatiot tuottavat eri tuloksen tai ainakin toinen operaatioista on mahdoton. Konversio-ohjelmassa kaikki laskutoimitukset on suoritettu skalaareilla, joten siinä on voitu käyttää taulukko-operaatioiden sijaan lyhyemmin merkittyjä matriisioperaatioita. Yhteen- ja vähennyslaskut on suoritettu taulukko-operaatioilla. (13, s. 47, 49.)

Komentoikkunassa on esitelty taulukko- ja matriisioperaatioita neliömatriiseilla

```

EDU>> aa=2 %Sijoitetaan skalaari 2 muuttuun aa.
aa =
     2

EDU>> BB=[2,3;5,7] %Sijoitetaan (neliö-) matriisi [2,3;5,7] muuttuun BB.
BB =
     2     3
     5     7

EDU>> CC=[11,13;17,19] %Sijoitetaan (neliö-) matriisi [11,13;17,19] muuttuun CC.
CC =
    11    13
    17    19

EDU>> DD=aa*BB %Skalaarin (aa) ja matriisin (BB) summa lasketaan aina alkioittain.
DD =
     4     6
     7     9

EDU>> EE=BB*CC %Matriisien (BB ja CC) summa lasketaan aina alkioittain.
EE =
    13    16
    22    26

EDU>> FF=aa*BB %Skalaarin aa ja matriisin BB tulo alkioittain...
FF =
     4     6
    10    14

EDU>> GG=BB*CC %Matriisien BB ja CC tulo alkioittain...
GG =
    22    39
    85   133

EDU>> GG=BB*CC %...on eri laskutoimitus kuin BB:n ja CC:n matriisitulo.
GG =
    73    83
   174   198

EDU>>
  
```

Name	Value	Min	Max
BB	[2 3; 5 7]	2	7
CC	[11 13; 17 19]	11	19
DD	[4 6; 7 9]	4	9
EE	[13 16; 22 26]	13	26
FF	[4 6; 10 14]	4	14
GG	[73 83; 174 198]	73	198
aa	2	2	2

Kuva 22. Eräiden taulukko- ja matriisioperaatioiden esittelyä ja vertailua.

Kuvassa 22 on esitetty kuinka taulukko- ja matriisioperaatiot poikkeavat toisistaan kertolaskun tapauksessa. Yhden taulukon (aa) ollessa skalaari, tulokset (FF) ovat samoja. Kun molemmissa taulukoissa (BB, CC) on useampi alkio, tulokset (GG) poikkeavat toisistaan. Esimerkiksi taulukko-operaatiossa taulukon BB rivillä 1, sarakkeessa 1 sijaitsevan alkion arvo "2" on kerrottu taulukossa CC samalla rivillä ja sarakkeessa sijaitsevan alkion arvolla "11" ja niiden tulo "22" on sijoitettu samalle paikalle taulukkoon GG. Matriisioperaatiossa taulukon GG vastaavan alkion arvo (73) on saatu taulukon BB ensimmäisen rivin ja taulukon CC ensimmäisen sarakkeen pistetulona ($2 \cdot 11 + 3 \cdot 17$). Molemmat operaatiot ovat nyt mahdollisia, koska taulukot BB ja CC ovat samankokoisia eli niissä on yhtä monta riviä ja saraketta (taulukko-operaatio), ja taulukossa BB on yhtä monta saraketta, kuin taulukossa CC on rivejä (matriisioperaatio).

1	%Matlabin taulukko-operaatiot
2	
3	A + B %plus, A:n ja B:n summa
4	A - B %minus, A:n ja B:n erotus
5	A .* B %times, A:n ja B:n alkioittainen kertolasku
6	A .^ B %power, A potenssiin B alkioittain
7	A ./ B %rdivide, A:n ja B:n osamäärä alkioittain ("oikealta jako")
8	A .\ B %ldivide, B:n ja A:n osamäärä alkioittain ("vasemmalta jako")
9	A.' %transpose, A:n transpoosi
10	
11	%Matlabin matriisioperaatiot
12	
13	A * B %mtimes, A:n ja B:n matriisitulo
14	A / B %mrdivide, A:n ja B:n käänteismatriisin tulo ("oikealta jako")
15	A \ B %mldivide, A:n käänteismatriisin ja B:n tulo ("vasemmalta jako")
16	A ^ B %mpower, A matriisipotenssiin B
17	A' %ctranspose, reaalityyppillä A:n transpoosi, kompleksityyppillä A:n
18	%hermitoitu matriisi

Kuva 23. Matlabin taulukko- ja matriisioperaatiot. Kommenteista on linkki operaatioiden dokumentaatioon Math Works -yhtiön verkkosivuilla, joten ne ovat kuvassa osittain erivärisiä kuin Matlabin tekstinmuokkausikkunassa.

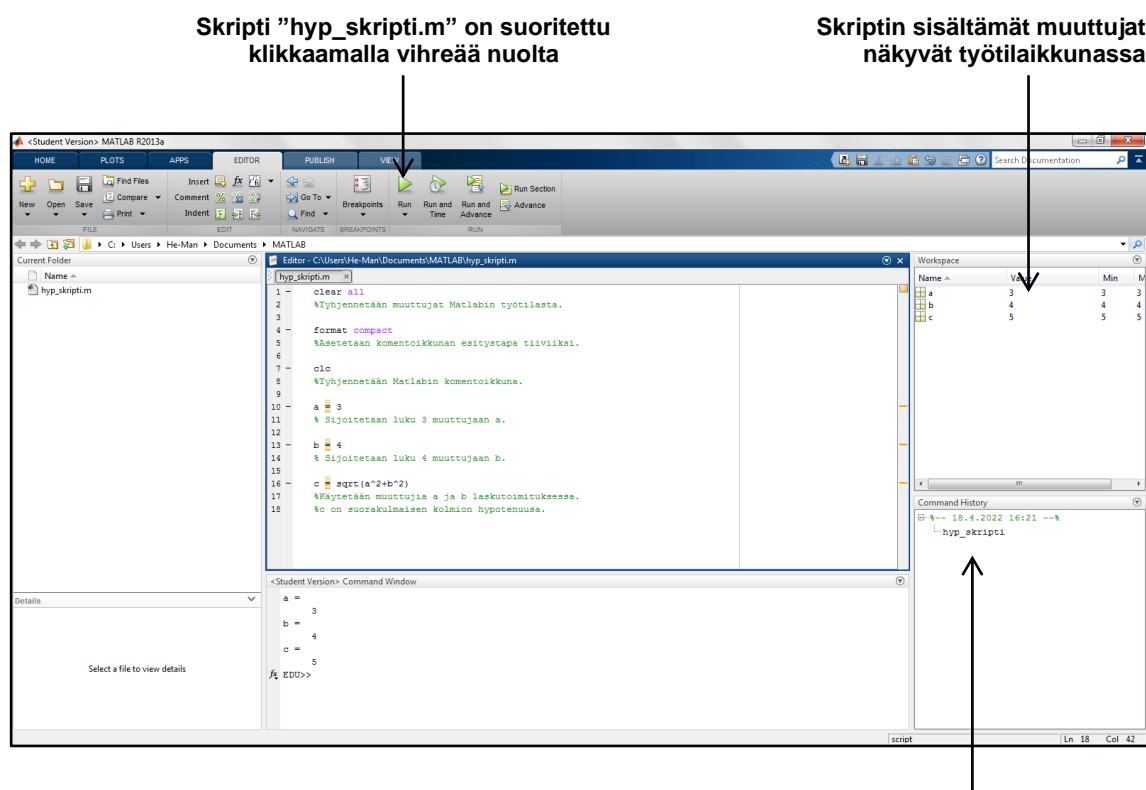
1	%Matlabin vertailuoperaatiot, joille on oma symbolinsa
2	
3	A == B %eq, A on yhtä suuri kuin B
4	A ~= B %ne, A on eri suuri kuin B
5	A > B %gt, A on suurempi kuin B
6	A >= B %ge, A on suurempi tai yhtä suuri kuin B
7	A < B %lt, A on pienempi kuin B
8	A <= B %le, A on pienempi tai yhtä suuri kuin B
9	
10	%Matlabin loogiset operaatiot, joille on oma symbolinsa
11	
12	A & B %and, A ja B
13	A && B %short-circuit and, A ja B, B:tä ei arvioida, jos A on epätosi
14	A B %or, A tai B
15	A B %short-circuit or, A tai B, B:tä ei arvioida, jos A on tosi
16	~A %not, ei A

Kuva 24. Matlabin vertailu- ja loogiset operaatiot, joille on oma symbolinsa. Kommenteista on linkki operaatioiden dokumentaatioon Math Works -yhtiön verkkosivuilla, joten ne ovat kuvassa osittain erivärisiä kuin Matlabin tekstinmuokkausikkunassa.

Aritmeettisten operaatioiden lisäksi Matlabissa on myös vertailuoperaatioita (relational operation) ja loogisia operaatioita (logical operation). Vertailuoperaatioilla voidaan tutkia lausekkeiden erilaisuussuhdetta. Lausekkeina tai niiden osina voivat olla vakiot, muuttujat ja funktiot. Operaation vastaus on loogisen datatyyppin arvo tosi (true) tai epätosi (false). Loogisilla operaatioilla taas voidaan verrata esimerkiksi useiden erilaisuussuhteiden toteutumista samaan aikaan. Myös tällöin vastauksena on tosi tai epätosi. Matlabin vertailu- ja loogiset operaatiot on esitetty kuvassa 24. Konversio-ohjelmassa niitä on käytetty silmukoiden ja ehtolauseiden (ks. 3.6 ja 3.7) kanssa. (13, s. 92–99.)

3.5 Skriptit ja funktiot

Edellä käsitellyissä esimerkeissä (ks. luvut 3.2–3.4) Matlabia on käytetty tavanomaisen laskukoneen tapaan, syöttämällä käskyjä suoraan komentoikkunaan. Ohjelmiston todellinen tehokkuus tulee kuitenkin ilmi vasta, kun käskyjä aletaan koostaa kokonaisiksi ohjelmiksi. Tähän tarvitaan skriptejä (script) ja funktioita (function). Ne ovat Matlabkoodia sisältäviä tekstitiedostoja, joita yhdessä kutsutaan m-tiedostoiksi (m file). Nimitys tulee niiden kanssa käytetystä tiedostopäätteestä ".m". (14, s. 28.)



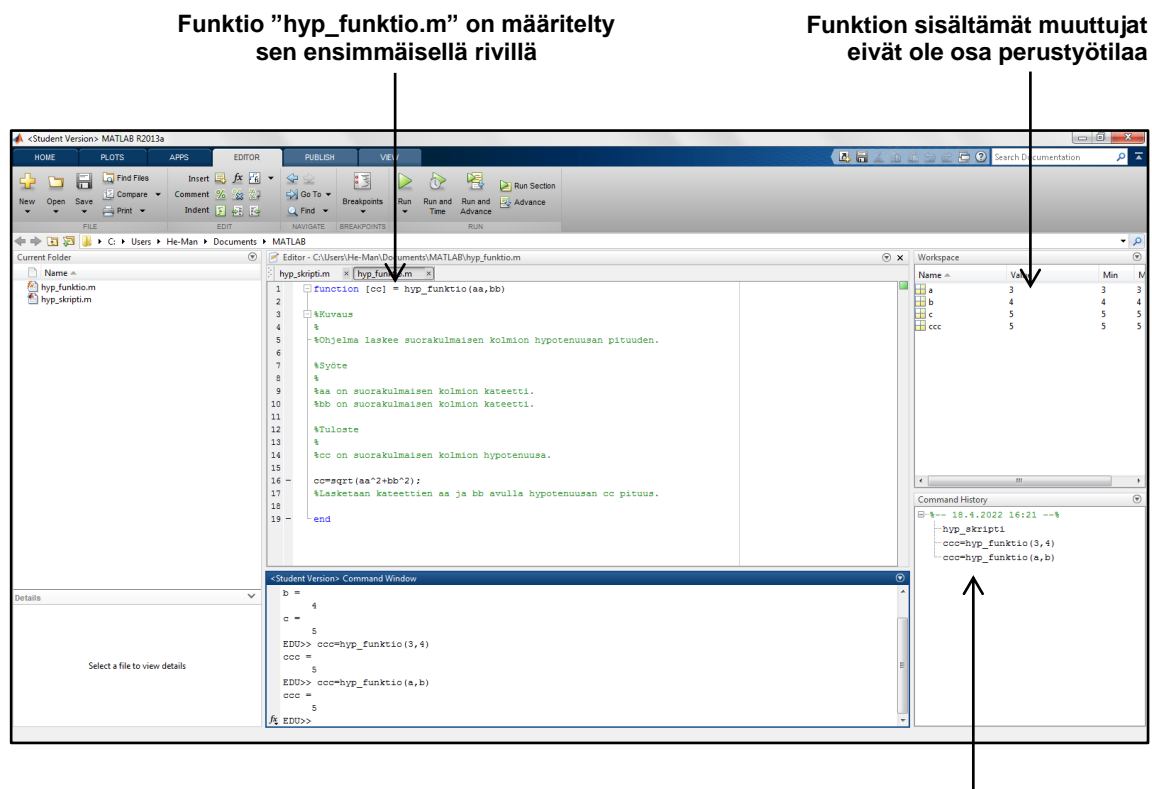
Kuva 25. Skriptin kirjoittaminen ja suorittaminen Matlabissa.

Suoritettu skripti näky komentohistoriassa

Skriptit ja funktiot eroavat toisistaan siinä, miten ne ovat yhteydessä Matlabin komentoikkunasta hallittavaan perustyötilaan (base workspace). Skriptien sisältämät käskyt vastaavat ohjelmiston komentoikkunasta suoritettuja käskyjä, ne on vain tallennettu tiedostoon, jolloin ne ovat helposti uudelleen käytettävissä. Jokaisella yksittäisellä funktiolla taas on oma työtilansa (function workspace), ja ne ovat yhteydessä Matlabin perustyötilaan niille annetun syöteen (input) ja niiden palauttaman tulosteen (output) avulla. (13, s. 201; 15, s. 5.11.)

Käskyjen suorittamista suoraan komentoikkunasta voi verrata käskyjen suorittamiseen skriptin avulla tarkastelemalla kuvaa 17 (ks. 3.2) ja yllä olevaa kuvaa 25. Molemmissa

kuvissa on ratkaistu suorakulmisen kolmion hypotenuusan "c" pituus, kun kateettien "a" ja "b" pituudet ovat 3 ja 4. Molemmissa tapauksissa vastaus on 5. Jälkimmäisessä kuvassa käskyt on käsin kirjoittamisen sijaan annettu suorittamalla kuvan keskellä näkyvä skripti "hyp_skripti.m" klikkaamalla Matlabin yläpalkista vihreää nuolta, jonka alla on teksti "Run". Vaihtoehtoisesti skriptin voi suorittaa myös kirjoittamalla sen nimen suoraan komentoikkunaan. Kuvien oikeissa yläkulmissa olevista työtilaikkunoista nähdään, että molemmissa tapauksissa työtilassa ovat käytettävissä muuttujat "a", "b" ja "c", mikä osoittaa skriptin käyttävän Matlabin perustyötilaa. Oikeissa alakulmissa olevista komentohistoriaikkunoista taas voidaan havaita, että komentohistoriaan tallentuvat vain komentoikkunasta suoritettut käskyt, eli käsin kirjoitettaessa kaikki käskyt ja suoritettun skriptin tapauksessa vain skriptin nimi. (13, s. 201.)



Kuva 26. Funktion kirjoittaminen ja suorittaminen Matlabissa.

Suoritettu funktio näkyy komentohistoriassa

Skriptien ja funktioiden eroavaisuuksia puolestaan voi tarkastella vertaamalla edellisellä sivulla olevaa kuvaa 25 ja yllä olevaa kuvaa 26. Molemmissa ratkaistaan jälleen sama suorakulmisen kolmion hypotenuusan pituus. Matlab erottaa funktion ja skriptin toisistaan funktion ensimmäiselle riville kirjoitettavasta määrittelylauseesta (kuva 26), joka alkaa avainsanalla "function". Sitä seuraa hakasulkeissa funktion tuloste "[cc]", sijoitusoperaattori "=", funktion nimi "hyp_funktio" ja kaarisulkeissa funktion syöte (aa,bb). Funktio on suoritettu komentoikkunasta kahteen kertaan, jotta nähdään, kuinka

se on yhteydessä perustyötilaan. Ensin funktiolle "hyp_funktio.m" on annettu syötteeksi luvut 3 ja 4, ja seuraavaksi aiemmin skriptillä määritellyt muuttujat "a" ja "b", joilla on samat arvot 3 ja 4. Molemmissa tapauksissa funktion tuloste (5) on sijoitettu muuttujaan "ccc". Funktion sisällä, sen omassa työtilassa, muuttujien "a" ja "b" arvot kuuluvat muuttujille "aa" ja "bb". Funktion tuloste annetaan muuttujalla "cc". Kuvan oikeassa yläkulmassa olevassa työtilaikkunassa funktion sisäiset muuttujat eivät näy, koska ne eivät kuulu perustyötilaan, vaan ovat käytettävissä ainoastaan funktion omassa työtilassa. Oikeassa alakulmassa olevasta komentohistoriaikkunasta taas nähdään, että komentohistoriaan tallentuvat vain komentoikkunasta funktion suorittamiseksi annetut käskyt, mutta ei funktion sisäisiä käskyjä. (13, s. 201–202; 15, s. 5.11–5.12.)

1	abs(X)	%Palauttaa luvun X itseisarvon.
2	addpath('X')	%Lisää kansion 'X' hakupolkuun.
3	asin(X)	%Palauttaa luvun X arkussin radiaaneina.
4	asinh(X)	%Palauttaa luvun X hyperbolisen arkussin radiaaneina.
5	atan(X)	%Palauttaa luvun X arkustangentin radiaaneina.
6	atanh(X)	%Palauttaa luvun X hyperbolisen arkustangentin radiaaneina.
7	cd('X')	%Vaihtaa kansion 'X' työkansioksi.
8	ceil(X)	%Palauttaa luvun X pyöristettynä ylöspäin kokonaislukuun.
9	cos(X)	%Palauttaa radiaaneina olevan kulman X kosinin.
10	cosh(X)	%Palauttaa radiaaneina olevan kulman X hyperbolisen kosinin.
11	edit('X')	%Avaa tiedoston 'X' tekstinmuokkausikkunaan.
12	fclose('X')	%Sulkee tiedoston 'X' komennoilla muokkaamiselta.
13	floor(X)	%Palauttaa luvun X pyöristettynä alaspäin kokonaislukuun.
14	fopen('X')	%Avaa tiedoston 'X' komennoilla muokattavaksi.
15	format('X')	%Muuttaa komentoikkunan esitystapaa määritelmän 'X' mukaan.
16	fprintf(X, 'Y', Z)	%Kirjoittaa tietoja Z tavalla 'Y' tiedostoon tunnisteella X.
17	pi()	%Palauttaa ympyrän kehän ja halkaisijan osamäärän likiarvon.
18	round(X)	%Palauttaa luvun X pyöristettynä lähimpään kokonaislukuun.
19	sech(X)	%Palauttaa radiaaneina olevan kulman X hyperbolisen sekantin.
20	sin(X)	%Palauttaa radiaaneina olevan kulman X sinin.
21	sinh(X)	%Palauttaa radiaaneina olevan kulman X hyperbolisen sinin.
22	size(X)	%Palauttaa taulukon X rivien ja sarakkeiden lukumäärän.
23	sqrt(X)	%Palauttaa luvun X neliöjuuren.
24	strcmp('X', 'Y')	%Vertaa merkkijonoja 'X' ja 'Y'. Palauttaa totuusarvon.
25	tan(X)	%Palauttaa radiaaneina olevan kulman X tangentin.
26	tanh(X)	%Palauttaa radiaaneina olevan kulman X hyperbolisen tangentin.
27	textscan(X)	%Lukee tietoja avoimesta tekstitiedostosta tunnisteella X.

Kuva 27. Konversio-ohjelmassa käytetyt Matlabin sisäänrakennetut funktiot. Funktioista on linkki niiden dokumentaatioon Math Works -yhtiön verkkosivuilla, joten ne ovat kuvassa erivärisiä kuin Matlabin tekstinmuokkausikkunassa.

Skriptejä on tarkoituksenmukaista käyttää silloin, kun niiden sisältämien muuttujien alkuarvot pysyvät samoina. Funktiot taas sopivat käytettäväksi samankaltaisten ongelmien ratkaisuun silloin, kun muuttujien alkuarvot (syöte) vaihtuvat. Tähän tarkoitukseen Matlabissa on itse kirjoitettujen funktioiden lisäksi myös suuri joukko sisäänrakennettuja funktioita, joista konversio-ohjelmassa käytetyt on esitetty kuvassa 27.

3.6 Ehtolauseet if, elseif ja else

On erittäin tavallista, että ohjelman halutaan toimivan vaihtoehtoisin tavoin riippuen sille annetusta syötteestä. Tämä on mahdollista käyttämällä ehtolauseita *if*, *elseif* ja *else*. Täsmällisemmin ilmaistuna ne ovat avainsanoja, joista *if* ja *elseif* muodostavat yhdessä niiden perään kirjoitetun ehdon kanssa varsinaisen ehtolauseen. Jos ehtolause on tosi, suoritetaan sitä seuraavien lauseiden, esimerkiksi laskutoimitusten muodostama koodi-

```

1 %If-elseif-else-rakenne on muotoa:
2 %
3 %if ehto
4 %   lauseet
5 %elseif ehto
6 %   lauseet
7 %else
8 %   lauseet
9 %end
10 %
11 %ehto on lausekkeiden erilaisuussuhteen testi. Lausekkeina tai niiden osana
12 %voivat olla sekä vakiot, muuttujat että funktiot. If- ja elseif-avainsana
13 %muodostavat niitä seuraavan ehdon kanssa ehtolauseen. Else-avainsana on
14 %ehtolause tapaukselle, jossa mikään aiemmista ehdoista ei ole tosi.
15 %Jokaisessa ehtolauseiden muodostamassa rakenteessa on oltava yksi ja vain
16 %yksi if-avainsana. Elseif-avainsanoja voi olla rajaton määrä ja else-
17 %avainsanoja korkeintaan yksi. Rakenne päättyy aina end-avainsanaan.
18 %lauseet tarkoittaa yhtä tai useampaa tehtävää, jotka voivat olla
19 %esimerkiksi laskutoimituksia. Kunkin ehtolauseen jälkeen seuraavat lauseet
20 %muodostavat koodilohkon, joka suoritetaan jos ehto on tosi.
21 %
22 %Esimerkkejä erilaisista tavoista käyttää ehtolauseita alla.
23
24 a = 7
25 if a < 10
26     disp('a on pienempi kuin 10')
27 end
28 %Tässä rakenteessa on vain yksi ehtolause "if a < 10". Teksti "a on
29 %pienempi kuin 10" tulostuu Matlabin komentoikkunaan, koska ennen
30 %rakennetta määritellyn muuttujan a arvo on seitsemän.
31
32 b = 13
33 if b < 10
34     disp('b on pienempi kuin 10')
35 else
36     disp('b on suurempi tai yhtäsuuri kuin 10')
37 end
38 %Tässä rakenteessa on kaksi ehtolauseetta "if b < 10" ja "else" eli kaikki
39 %muut tapaukset. Teksti "b on suurempi tai yhtä suuri kuin 10" tulostuu
40 %Matlabin komentoikkunaan, koska ennen rakennetta määritellyn muuttujan b
41 %arvo on 13.
42

```

Kuva 28. Yksinkertaisia esimerkkejä ehtolauseiden käytöstä.

lohko. Avainsana *else* taas on itsessään ehtolause tapaukselle, jossa yksikään aiemista ehdoista ei ole tosi. Jos useampi kuin yksi ehtolause on tosi, suoritetaan vain niistä rakenteessa ensimmäisenä olevaa seuraava koodilohko. Kuvissa 28 ja 29 on esitetty yksinkertaisia esimerkkejä ehtolauseiden eri käyttötavoista. (13, s 102–112.)

```

43 c = 10
44 if c < 10
45     disp('c on pienempi kuin 10')
46 elseif c == 10
47     disp('c on yhtä suuri kuin 10')
48 else
49     disp('c on suurempi kuin 10')
50 end
51 %Tässä rakenteessa on kolme ehtolauseetta "if c < 10", "elseif c==10" ja
52 %else eli kaikki muut tapaukset. Teksti "c on yhtä suuri kuin 10" tulostuu
53 %Matlabin komentoikkunaan, koska ennen rakennetta määritellyn muuttujan c
54 %arvo on 10.
55
56 d = 23
57 if (-1)^d == -1
58     disp('d on pariton')
59     if isprime(d) == 1
60         disp('d on alkuluku')
61     else
62         disp('d ei ole alkuluku')
63     end
64 else
65     disp('d on parillinen')
66     disp('d ei ole alkuluku')
67 end
68 %Tässä rakenteessa on käytetty sisäkkäisiä ehtolauseita. Ulommassa
69 %if-else-rakenteessa on ehtolauseet "if (-1)^d == -1" ja else eli kaikki
70 %muut tapaukset. Sisemmässä if-else-rakenteessa on ehtolauseet
71 %"isprime(d)== 1" ja else eli kaikki muut tapaukset. Sisempi if-lause on
72 %esimerkki funktion käyttämisestä ehtolauseen lausekkeena. Rakenne
73 %selvittää onko kahta (2) suurempi luku parillinen vai pariton, ja onko se
74 %alkuluku. Tekstit "d on pariton" ja "d on alkuluku" tulostuvat Matlabin
75 %komentoikkunaan, koska ennen rakennetta määritellyn muuttujan d arvo on
76 %23.

```

Kuva 29. Lisää yksinkertaisia esimerkkejä ehtolauseiden käytöstä.

Yksinkertaistettujen esimerkkien lisäksi on alla olevassa kuvassa 30 esitetty käytännöllisempi tapaus ehtolauseiden hyödyntämisestä matemaattisen ongelman ratkaisemisessa. Funktio "tay.m" ratkaisee käyttäjän antaman toisen asteen yhtälön reaali juuret. Ohjelmalle annetaan syötteenä yhtälön normaalimuodon kertoimet, joista toisen asteen termin kerroin ei voi olla nolla, koska tällöin kyseessä ei olisi toisen asteen yhtälö. Ohjelma palauttaa tulosteena taulukon, joka on kahden alkion vaakavektori, kun reaali juuria on kaksi. Jos juuria on kahden sijasta yksi, taulukko on skalaari, ja jos juuria ei ole lainkaan, palauttaa ohjelma tyhjän taulukon.

```

1 function [J] = tay(a,b,c)
2
3 %Ohjelma selvittää käyttäjän antaman toisen asteen yhtälön reaali juuret.
4 %
5 %Syöte
6 %
7 %a,b,c ovat toisen asteen yhtälön normaalimuodon a*x^2+b*x+c = 0 kertoimet.
8 %a ei voi olla 0.
9 %
10 %Tuloste
11 %
12 %J on vektori, joka sisältää yhtälön juuret, joiden lukumäärä voi olla
13 %0, 1, 2.
14
15 D = b^2-4*a*c;
16 %Ratkaistaan diskriminantti.
17
18 if D < 0
19     J = [];
20 elseif D == 0
21     J = -b/(2*a);
22 else
23     J1 = (-b+sqrt(b^2-4*a*c))/(2*a);
24     J2 = (-b-sqrt(b^2-4*a*c))/(2*a);
25     if J1 < J2
26         J = [J1,J2];
27     else
28         J = [J2,J1];
29     end
30 end
31 end

```

Kuva 30. Esimerkki ehtolauseiden käytöstä matemaattisen ongelman ratkaisussa.

Ehtolauseiden *if*, *elseif* ja *else* lisäksi Matlab-ohjelman muuttujan arvosta riippuvia vaihtoehtoisia toimintatapoja voidaan toteuttaa myös *switch-case*-rakenteen ja *try-catch*-rakenteen avulla. *Switch-case*-rakenne on tarkoitettu käytettäväksi erityisesti silloin, kun ehtoina on yhden muuttujan yksittäisiä arvoja tai äärellisiä arvojoukkoja. Tällaisessa tapauksessa *switch-case*-rakenne on käyttäjälle helppolukuisempi ja vaihtoehtojen määrän ollessa suuri myös tietokoneelle nopeampi suorittaa kuin *if-elseif-else*-rakenne. Sitä voidaan kuitenkin totuusarvojen avulla käyttää myös paljolti samaan tapaan kuin *if-elseif-else*-rakennetta. *Try-catch*-rakenne on tarkoitettu poikkeusten käsittelyyn. Ohjelman kohdatessa virheen, sen toiminta keskeytyy. *Try-catch*-rakenteen avulla ohjelman keskeytyminen voidaan välttää siten, että *try*-avainsanaa seuraavan koodilohkon kohdatessa virheen, Matlab keskeyttää sen suorittamista, mutta ei koko ohjelman suorittamista, vaan jatkaa suoraan *catch*-avainsanaa seuraavasta koodilohkosta. Kumpaakaan rakennetta ei ole käytetty konversio-ohjelmassa, eikä niitä siksi käsitellä tässä enempää. Lukija voi halutessaan tutustua niihin tarkemmin esimerkiksi luvussa 3.1 lueteltujen lähteiden avulla. (13, s. 113–115; 14, s. 72.)

3.7 For- ja while-silmukat

Vähänkään pidemmän ohjelman kanssa syntyy yleensä myös tarve suorittaa sama käsky useamman kerran peräkkäin. Tähän käytetään silmukoita. Ne ovat toistorakenteita, joiden avulla valittua käskyä voidaan toistaa haluttu määrä kertoja tai kunnes jokin ehto ei täyty. Silmukoita on kahdenlaisia: *for*- ja *while*-silmukoita. Ensimmäistä käytetään, kun tarvittavien toistokertojen määrä tunnetaan ennalta. Jälkimmäistä taas toistetaan siihen saakka, että käyttäjän määrittämä ehto ei enää täyty. (13, s. 147.)

```

1  %For-silmukka on muotoa:
2  %
3  %for indeksi = arvot
4  %   lauseet
5  %end
6  %
7  %indeksi on muuttuja, johon tallennetaan "lennosta" silmukan kullakin
8  %toistokerralla käytettävä arvo. Se saadaan vektorista arvot.
9  %arvot on vektori, johon on tallennettu ennalta kaikkien mahdollisten
10 %toistokertojen arvot. Se annetaan muodossa alku:loppu tai
11 %alku:lisäys:loppu. Jos lisäystä ei ole annettu, se on oletuksena 1.
12 %Toistokertojen arvot voidaan antaa myös yksitellen tai funktioiden avulla.
13 %lauseet tarkoittaa yhtä tai useampaa tehtävää, jotka silmukan kullakin
14 %toistokerralla voidaan suorittaa. Ne ovat esimerkiksi laskutoimituksia.
15 %
16 %Esimerkkejä erilaisista for-silmukoista alla.
17
18 for i = 1:5
19     a = 2*i
20 end
21 %Tämä silmukka kertoo arvot 1-5 (eli 1,2,3,4,5) kahdella. Arvot on annettu
22 %muodossa alku:loppu (eli 1:5). Tulot (2,4,6,8,10) tallennetaan muuttujaan
23 %a, joka tulostuu Matlabin komentoikkunaan. Muuttujan a arvo
24 %ylikirjoitetaan joka toistokerralla.
25
26 for i = 4:2:12
27     b = i^2
28 end
29 %Tämä silmukka laskee kahden numeron välein arvojen 4-12 (eli 4,6,8,10,12)
30 %neliöt. Arvot on annettu muodossa alku:lisäys:loppu (eli 4:2:12). Neliöt
31 %(16,36,64,100,144) tallentaa muuttujaan b, jonka arvo ylikirjoitetaan joka
32 %toistokerralla.
33
34 c = []
35 for i = [2,3,5,7,11]
36     i = i^3
37     c = [c,i]
38 end
39 %Tämä silmukka laskee yksitellen annettujen alkulukujen (2,3,5,7,11)
40 %kuutiot (8,27,125,343,1331) ja lisää ne ennen silmukkaa luotuun tyhjään
41 %vektoriin c.

```

Kuva 31. Yksinkertaisia esimerkkejä for-silmukan käytöstä.

Konversio-ohjelmassa on hyödynnetty ainoastaan *for*-silmukoita, joiden perustoiminnallisuutta on esitelty kuvissa 31 ja 32. Kuvien esimerkeissä ei ole huomioitu tapauksia, joissa indeksimuuttuja ei ole skalaari.

```

42
43 d = [13,17,19,23,29]
44 e = []
45 for i = d
46     i = i+1
47     e = [e,i]
48 end
49 %Tämä silmukka korottaa ennen silmukkaa määritellystä vektorista löytyviä
50 %alkulukuja (13,17,19,23,29) yhdellä lähimpään, suurempaan parilliseen
51 %lukuun (14,18,20,24,30) ja lisää ne ennen silmukkaa luotuun tyhjään
52 %vektoriin e.
53
54 f = [11,22,33,44,55]
55 g = 0
56 for i = 1:length(f)
57     g = g+10
58 end
59 %Tämä silmukka saa toistokertojensa määrän funktion length avulla vektorin
60 %f alkioden lukumäärästä. Toistokertojen arvot on annettu muodossa
61 %alku:loppu (eli 1:length(f) eli 1:5). Silmukka korottaa joka
62 %toistokerralla ennen silmukkaa määritellyn muuttujan g arvoa (aluksi 0)
63 %kymmenellä (lopuksi 50). Toistokertojen indeksiarvoja ei ole käytetty
64 %laskutoimituksissa.
65
66 H = []
67 for i = 1:10
68     k = []
69     for j = 1:10
70         k = [k,j*i]
71     end
72     H = [H;k]
73 end
74 %Tämän silmukan sisällä on toinen for-silmukka. Sisempi silmukka laskee
75 %kutakin ulomman silmukan toistokerran arvoa (1-10) vastaavan kertotaulun
76 %(1-10) ja lisää sen ennen silmukkaa luotuun tyhjään matriisiin H.
77
78 l = [31,37,41,43,47]
79 m = []
80 for i = 1
81     if i > 42
82         break
83     end
84     m = [m,i]
85 end
86 %Tämä silmukka lisää ennen silmukkaa määritellyn vektorin l alkion
87 %(31,37,41,43,47) ennen silmukkaa luotuun tyhjään vektoriin m. Mikäli
88 %vektorista l kuitenkin löytyy alkio, joka on suurempi kuin 42, silmukan
89 %suorittaminen keskeytetään break-komennolla.

```

Kuva 32. Hieman monimutkaisempia esimerkkejä *for*-silmukan käytöstä.

Yksinkertaistettujen esimerkkien lisäksi on alla olevassa kuvassa 33 esitetty käytännöllisempi tapaus *for*-silmukan hyödyntämisestä matemaattisen ongelman ratkaisemisessa. Funktio "kolnel.m" osaa luvusta yksi alkaen selvittää käyttäjän antaman määrän lukuja, jotka ovat sekä kolmio- että neliölukuja eli luonnollisia lukuja, joiden osoittamasta määrästä pisteitä voidaan muodostaa sekä tasasivuisen kolmion että neliön muotoinen kuvio. Ohjelmalle annetaan syötteeksi lukujen määrä, ja se palauttaa matriisin, jonka sarakkeina ovat kolmion sivun pituus, neliön sivun pituus sekä itse luku. Ohjelmassa käytetyn algoritmin toiminta on selitetty matemaatikko Alexander Bogomolnyn kirjoittamassa artikkelissa *There exist triangular numbers that are also square* (16).

```

1 function [M] = kolnel(lkm)
2
3 %Ohjelma selvittää luvusta yksi (1) alkaen käyttäjän antaman määrän
4 %kokonaislukuja, jotka ovat sekä kolmio- että neliölukuja.
5 %
6 %Syöte
7 %
8 %lkm on käyttäjän antama kokonaislukujen määrä.
9 %
10 %Tuloste
11 %
12 %M on matriisi, jonka ensimmäiseen sarakkeeseen tallennetaan kolmion sivun
13 %pituus, toiseen neliön sivun pituus ja kolmanteen etsittävä luku.
14
15 format longG
16
17 M = [1 1 1];
18 k = M(1);
19 n = M(2);
20
21 for luku = 2:lkm
22     ks = 3 * k + 4 * n + 1;
23     ns = 2 * k + 3 * n + 1;
24     k = ks;
25     n = ns;
26     M = [M; ks ns ns^2];
27 end

```

Kuva 33. Esimerkki *for*-silmukan käytöstä matemaattisen ongelman ratkaisussa.

Konversio-ohjelmassa ei ole tarvittu lainkaan *while*-silmukoita, koska kaikkien siinä käytettävien silmukoiden toistokertojen määrä tunnetaan käytettyjen laskentakaavojen osalta ennalta, ja voidaan käyttäjän antaman syöteen kohdalla erikseen laskea. Tästä huolimatta on yleisen mielenkiinnon vuoksi seuraavalla sivulla kuvassa 34 esitelty lyhyesti myös *while*-silmukan käyttöä. Toisin kuin *for*-silmukkaa koskeneissa esimerkeissä, on *while*-silmukkaa käsittelevien esimerkkien kohdalla käytetty erillisten skalaari-muuttujien lisäksi myös vektorin alkioista poimittua lukuarvoa.

```

1 %While-silmukka on muotoa:
2 %
3 %while vertailu
4 %   lauseet
5 %end
6 %
7 %vertailu on lausekkeiden erilaisuussuhteen testi. Silmukkaa toistetaan
8 %niin kauan, kuin vertailu on tosi, ellei silmukkaa keskeytetä muulla
9 %tavalla. Lausekkeina tai niiden osana voivat olla sekä vakiot, muuttujat
10 %että funktiot. Muuttujille on määriteltävä alkuarvo ennen silmukkaa tai
11 %ensimmäistä vertailua ei voida suorittaa. Jonkin muuttujan (ja lausekkeen)
12 %arvon on myös muututtava silmukan toistokertojen välillä, jottei silmukka
13 %jää toistumaan ikuisesti.
14 %lauseet tarkoittaa yhtä tai useampaa tehtävää, jotka silmukan kullakin
15 %toistokerralla voidaan suorittaa. Ne ovat esimerkiksi laskutoimituksia.
16 %
17 %Esimerkkejä erilaisista while-silmukoista alla.
18
19 a=1
20 while a < 10
21     a = a+1
22 end
23 %Tämä silmukka kasvattaa muuttujan a arvoa yhdellä (1), niin kauan kuin sen
24 %arvo on pienempi kuin 10. Muuttujan kasvatetut arvot (2-10) tulostuvat
25 %Matlabin komentoikkunaan. Arvo 10 tulostuu, koska se on ehditty tallentaa
26 %muuttujaan a ennen silmukan keskeyttävää vertailua. Myös alkuarvo (1)
27 %tulostuu komentoikkunaan.
28
29 b=1
30 c=b
31 while b+10 >= b^2-10
32     b=b+1
33     c=[c,b]
34 end
35 %Tämä silmukka kasvattaa muuttujan b arvoa yhdellä (1), niin kauan kuin
36 %lausekkeen b+10 arvo on suurempi tai yhtä suuri kuin lausekkeen b^2-10.
37 %Muuttujan b kasvatetut arvot (2-5) tulostuvat Matlabin komentoikkunaan ja
38 %ne tallennetaan vektorin c alkioiksi. Myös vektori c tulostuu ohjelman
39 %komentoikkunaan silmukan jokaisella toistokerralla.
40
41 x=1
42 y=2
43 while y(x) < 1024
44     x=x+1
45     y=[y,2^x]
46 end
47 %Tämä silmukka kasvattaa muuttujan x arvoa yhdellä (1) ja laskee luvun
48 %kaksi (2) potensseja (2,4,8,16,32,64,128,256,512,1024) niin kauan, kuin ne
49 %ovat pienempiä kuin luku 1024 eli 2^10. Ennen silmukkaa on luotu vektori
50 %y, jonka alkioiksi potenssit tallennetaan. Silmukan vertailu suoritetaan
51 %käyttämällä suoraan viimeisimpään vektorin y alkioon tallennettua
52 %potenssia, ei erillistä lauseketta tai skalaarimuuttujaa. Tästä syystä
53 %myös viimeisen alkion arvon 1024 on löydyttävä vektorista.

```

Kuva 34. Yksinkertaisia esimerkkejä while-silmukan käytöstä.

4 Konversio-ohjelman rakenne ja toiminta

4.1 Taustatietoja ohjelmasta

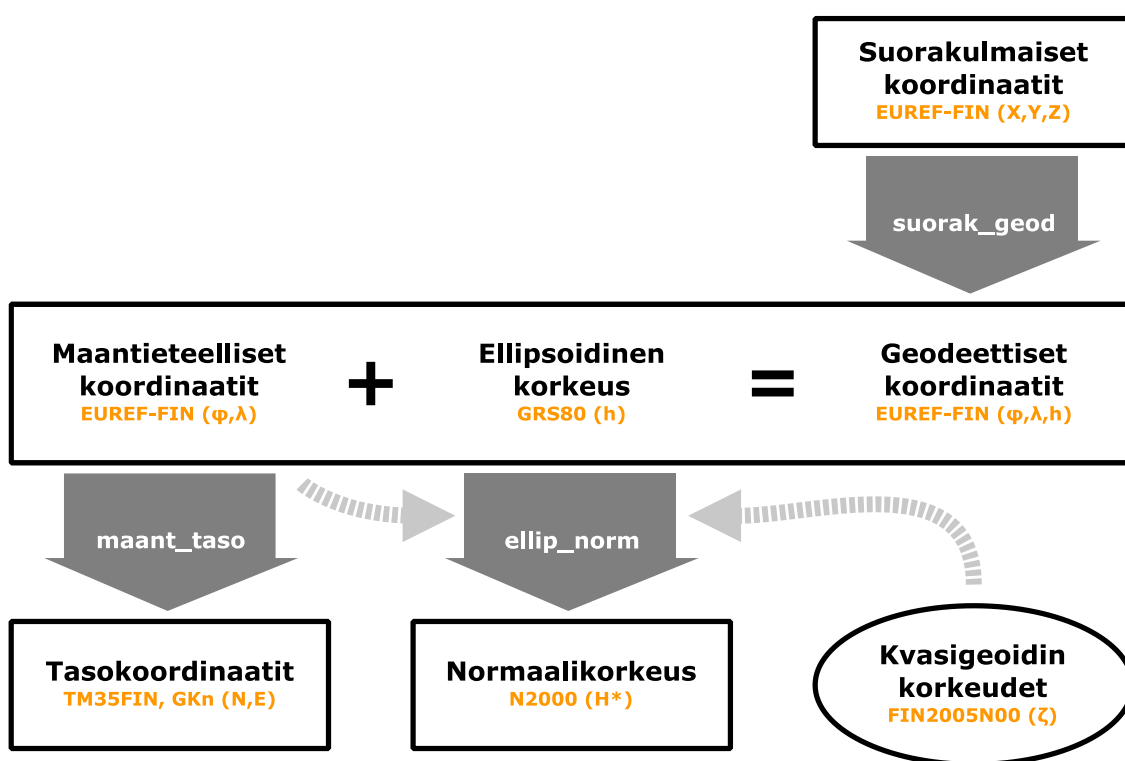
Seuraavassa esiteltävällä konversio-ohjelmalla voidaan suorittaa ETRS89-koordinaattijärjestelmän kansalliseen realisaatioon EUREF-FIN liittyvät koordinaattikonversiot sekä N2000-normaalikorkeuksien ja ellipsoidisten EUREF-FIN-korkeuksien väliset korkeusmuunnokset. Ohjelman voi ladata Math Works -yhtiön tiedostonvaihtopalvelusta, jossa sen käyttöönottoon on annettu myös lyhyet englanninkieliset ohjeet lukijoille, joiden kielitaito ei riitä tämän tekstin ymmärtämiseen. (1.)

Matemaattiset kaavat joihin ohjelman toiminta perustuu, ovat peräisin Maanmittauslaitoksen paikkatietokeskuksen julkaisuista *Suomen geodeettiset koordinaatitot ja niiden väliset muunnokset* sekä *Suomen geoidimallit ja niiden käyttäminen korkeuden muunnoksissa*. Korkeusmuunnosten kanssa tarvittava FIN2005N00-kvasigeoidimalli (FIN2005N00.lst) on ladattavissa Maanmittauslaitoksen verkkosivuilta (FIN2005N00_list.zip). Malli kattaa pituuspiirien 17,48–33,00 ja leveyspiirien 59,00–70,70 välisen alueen. Malli on sovitettu Suomen N2000-korkeusjärjestelmään, joten vaikka se ulottuu Suomen rajojen ulkopuolelle, sitä on tarkoitus käyttää vain rajojen sisäpuolella. Ohjelmaa varten laadittujen esimerkkikonversioiden koordinaattitiedot on saatu vapaan aineiston Wikimedia Commons -verkkosivustolta maantieteellisinä koordinaatteina (WGS84), ja niille on laskettu korkeudet Maanmittauslaitoksen Avoimien aineistojen tiedostopalvelun 10 m ruutukoon korkeusmallin avulla. Korkeusmalli on päivittyvä, ja esimerkkikonversioita varten korkeudet on laskettu loppuvuoden 2015 mallilla, jota ei ole enää saatavilla. (2; 3; 4; 17.)

Konversio-ohjelman ja tämän tekstin tarkoituksena on auttaa lukijaa ymmärtämään etenkin satelliittimittausten kanssa tarvittavien koordinaattikonversioiden ja korkeusmuunnoksen käyttöä. Ohjelman rakenteeseen ja toimintaan voi tutustua luvussa 4.2. Ohjelmaa on tarkoitus käyttää oppimateriaalina koostamalla sen avulla omia koordinaattikonversioita luvun 4.2 kaavakuvien 35, 36 ja 37 avulla. Jotta tämä olisi mahdollisimman helppoa, on ohjelmaa varten laadittu kaksi esimerkkikonversiota (liite 1, s. 1–4, 27), joiden käyttötapaan voi tutustua luvussa 4.3. Esimerkkikonversioiden lähdekoodia voi käyttää pohjana omien konversioiden laadinnassa, mihin opastetaan luvussa 4.4. Lisäksi luvussa 4.5 annetaan ohjeet konversioiden oikeellisuuden tarkistamiseen.

4.2 Ohjelman rakenne

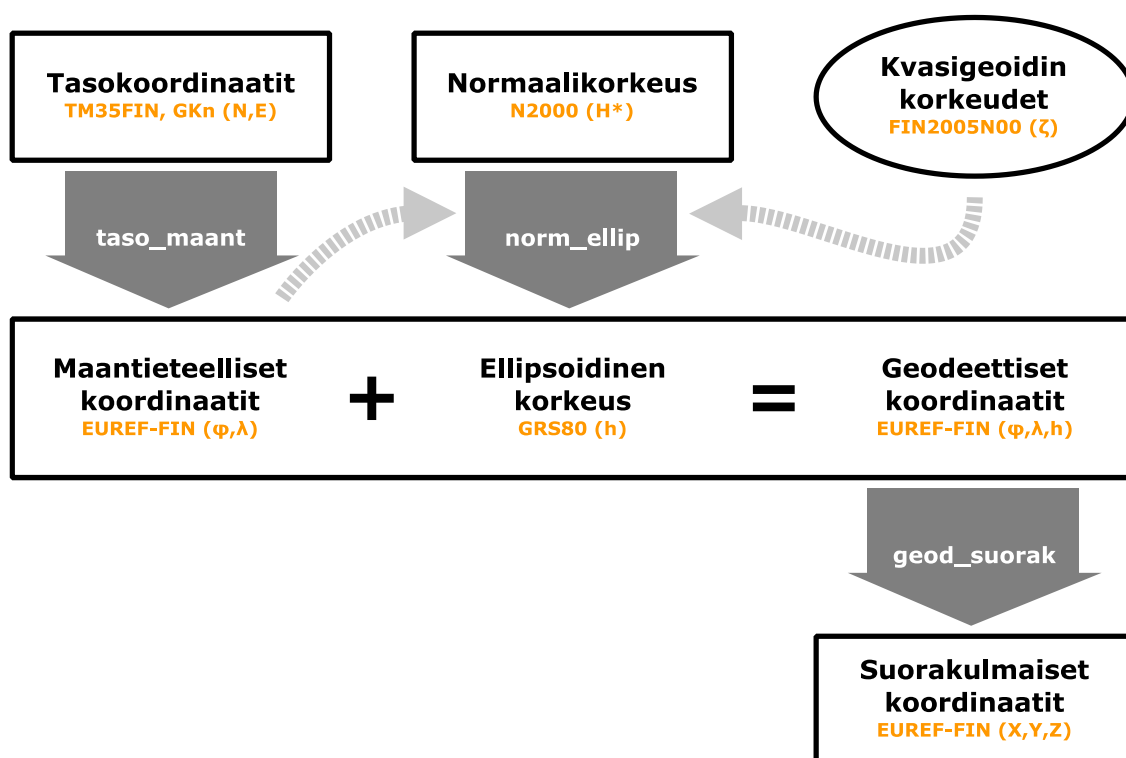
Konversio-ohjelma koostuu kolmista erilaisista rakenneosista. Ensimmäisiä näistä ovat konversioskriptit (esimerkit, liite 1, s. 27), joiden avulla ohjelman toimintaa ohjataan. Niillä annetaan ohjelmalle tieto siitä millainen konversio ja mitä lähtötiedostoa käyttäen on suoritettava. Jokaista konversiota varten tarvitaan yksi skripti. Toisia rakenneosia ovat konversio- ja korkeusmuunnosfunktiot (8 kpl, liite 1, s. 5–25), joiden avulla suoritetaan varsinainen koordinaattitietoihin liittyvä laskenta. Funktioille annetaan syötteenä koordinaattitietoja yhdessä koordinaatistossa ja ne taas palauttavat tulosteena koordinaattitietoja toisessa koordinaatistossa. Funktioita voidaan ketjuttaa, ja muodostaa näin



Kuva 35. Konversiopolun kaaviokuva kolmiulotteisista suorakulmaisista koordinaateista tasokoordinaateiksi ja normaalikorkeuksiksi.

konversiopolku, jolla suorittaa yhtä funktiota monimutkaisempia konversioita. Yhtä konversiota varten tarvitaan yhdestä kolmeen funktiota riippuen konversiopolun pituudesta. Tämän tässä konversiopoluksi kutsutun käsitteen ymmärtäminen, ja sellaisten itse muodostaminen on ohjelman oppimateriaalina käyttämisen ydin, ja asiaa käsitellään siksi tarkemmin luvussa 4.4. Kolmannen rakenneosan muodostaa yksin pääohjelma (paaohjelma.m, liite 1, s. 1–4), jonka sisällä konversiopolkujen kokoaminen tapahtuu.

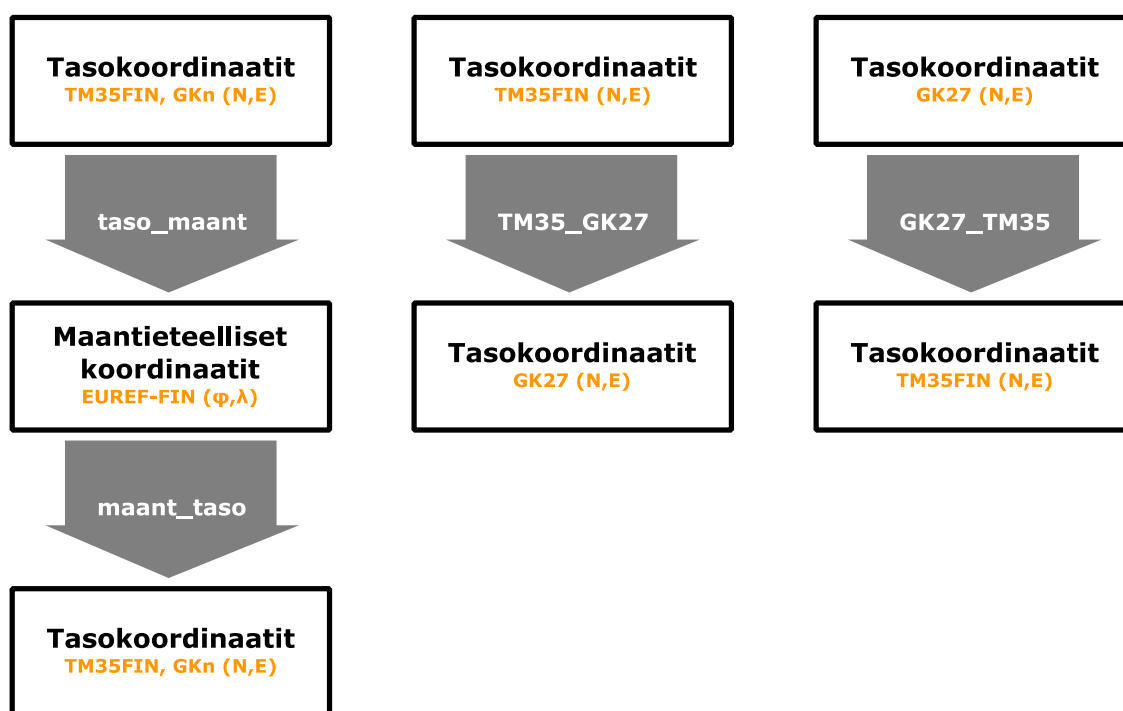
Pääohjelma on funktio kuten konversio- ja korkeusmuunnosfunktiotkin, mutta toisin kuin ne, se ei palauta tulosteena koordinaattitietoja muiden funktioiden käsiteltäväksi, vaan kirjoittaa ne tiedostoon, joka on konversio- lopputuote. Näiden rakenneosien lisäksi ohjelmaan kuuluvat vielä aloitusskripti (aloitus.m, liite 1, s. 26), jolla muutetaan Matlabin asetuksia sellaisiksi, että ohjelman käyttö on mahdollista, sekä korkeushilana listamuodossa oleva FIN2005N00-kvasigeoidimalli, joka mahdollistaa korkeusmuunnosten suorittamisen. Ohjelman käyttämät koordinaattitiedot luetaan ja kirjoitetaan csv-tiedostoista.



Kuva 36. Konversiopolun kaaviokuva tasokoordinaateista ja normaalikorkeuksista kolmiulotteiseksi suorakulmaisiksi koordinaateiksi.

Kuvan 35 kaaviossa on esitetty konversiopolku kolmiulotteisista suorakulmaisista koordinaateista tasokoordinaateiksi ja normaalikorkeuksiksi. Lukija voi poimia kuvasta omiin tarpeisiinsa myös muut samaan suuntaan etenevät konversiopolut. Kuvassa valkoiset, mustareunaiset laatikot kuvaavat erilaisia koordinaatistoja, ja myös normaalikorkeuksilla on oma laatikkonsa. Samanvärinen soikio kuvaa kvasigeoidimallia. Musta teksti laatikoiden sisällä kertoo koordinaatiston tyyppin ja oranssi teksti tämän alapuolella koordinaatiston tarkemman nimen. Tasokoordinaatit-laatikossa on useita koordinaatistoja. Oranssin tekstin perässä ovat koordinaatiston koordinaattiakselien lyhenteet, jotka ovat peräisin Maanmittauslaitoksen paikkatietokeskuksen julkaisuista, ja joita käytetään

myös konversio-ohjelman lähdekoodissa, sillä erolla, että kreikkalaiset kirjaimet on korvattu kirjainten latinalaisin aakkosin kirjoitetuilla nimillä ja normaalikorkeuksien lyhenteestä on jätetty pois asteriski (*), joka on Matlabissa kertomerkki. Tummanharmaat nuolet ovat konversio- ja korkeusmuunnosfunktioita, joiden suunta kuvaa sitä mistä koordinaatistosta mihin konversio tai muunnos suoritetaan. Nuolien sisällä olevat valkoiset tekstit ovat funktioiden nimiä ilman tiedostopäätettä. Harmaat katkoviivanuolet kertovat mitä tietoja funktio "ellip_norm.m" tarvitsee ellipsoidisten korkeuksien lisäksi, jotta se voi suorittaa muunnoksen normaalikorkeuksiksi.



Kuva 37. Konversiopolun kaaviokuva tasokoordinaateista toisiksi tasokoordinaateiksi.

Kuvan 36 konversiopolkuun pätevät vastaavat selitykset, nyt polun suunta vain on kääntynyt tasokoordinaateista ja normaalikorkeuksista kolmiulotteisiksi suorakulmaisiksi koordinaateiksi, ja funktiot ovat vaihtuneet. Kuva 37 poikkeaa edellisistä merkittävästi, siinä on esitetty kolme erilaista konversiopolkua tasokoordinaatistojen välillä. Ensin vasemmalla on tasokoordinaatistojen välisten konversioiden, kuten kaistanvaihtojen, peruserä: konversio on tehtävä maantieteellisten koordinaattien kautta. Tähän on kuitenkin kaksi poikkeusta, jotka näkyvät keskellä ja oikealla. ETRS-TM35FIN- ja ETRS-GK27-koordinaatistojen väliset konversiot voidaan suorittaa suoraan tasokoordinaatistosta toiseen, sillä niillä on sama keskimeridiaani, 27. pituuspiiri, joten pelkkä projektiotason vaihto riittää. Kuvassa ei ole huomioitu mahdollisia korkeusmuunnoksia.

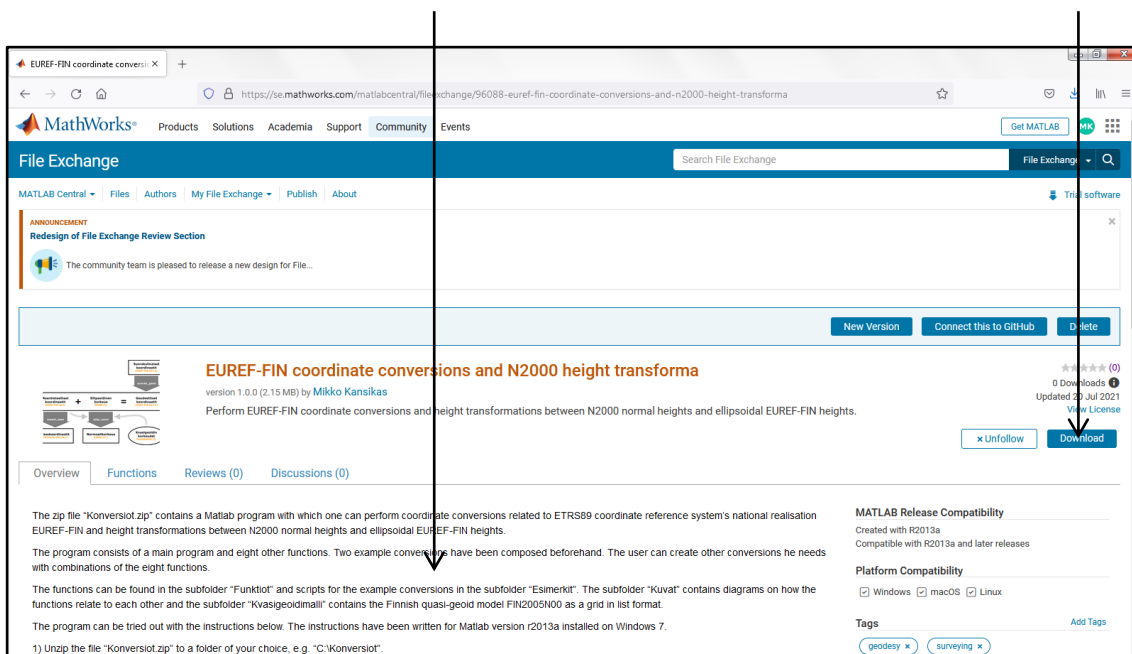
4.3 Ohjelman käyttöönotto

4.3.1 Valmistelevat toimenpiteet

Konversio-ohjelma on ladattavissa zip-pakattuna tiedostona (Konversiot.zip) Matlab-ohjelmistoa valmistavan Math Works -yhtiön tiedostonvaihtopalvelusta nimellä *EUREF-FIN coordinate conversions and N2000 height transformation*. Siltä varalta, että ohjelma ei jostain syystä olisi enää tulevaisuudessa saatavilla samasta paikasta, tämän tekstin liitteenä ovat lähes kaikki osat, joista se koostuu: konversio- ja muunnosfunktiot, niiden yhteistoiminnasta vastaava pääohjelma, skriptit, joiden avulla ohjelmaa käytetään sekä esimerkkikonversioiden tarvitsemat koordinaattitiedot. Ainoa puuttuva osa, FIN2005N00-kvasigeoidimalli (FIN2005N00.lst), on ladattavissa Maanmittauslaitoksen verkkosivuilta (FIN2005N00_list.zip). Tässä tekstissä esitetyn kuvauksen avulla lukijan on mahdollista koostaa ohjelma tarvittaessa itse uudelleen. (1; 2.)

Konversio-ohjelman englanninkielinen käyttöohje

Konversio-ohjelma ladataan tästä

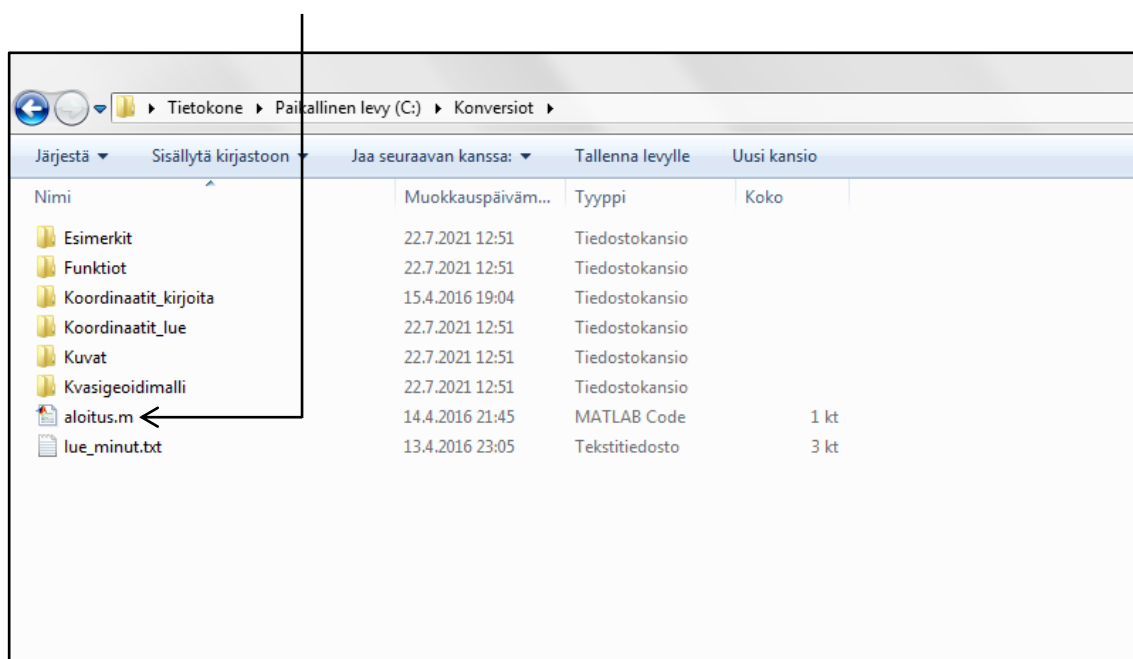


Kuva 38. Konversio-ohjelma on ladattavissa Math Works -yhtiön tiedostonvaihtopalvelusta.

Käytännössä konversio-ohjelma ladataan tiedostonvaihtopalvelusta klikkaamalla sivun oikeassa laidassa (kuva 38) olevaa painiketta, jossa on teksti "Download". Tätä varten sivustolle on rekisteröidyttävä ja kirjauduttava. Lataamisen jälkeen zip-tiedosto on purettava, suositeltavaa on käyttää hakemistoa "C:\Konversiot", joka on ohjelman lähde-

koodissa, tiedostossa "aloitus.m", määritelty Matlabin työkansion sijainniksi. Muussa tapauksessa työkansion sijainti on myöhemmin vaihdettava lukijan valitsemaksi. Kun zip-tiedosto on purettu, kannattaa ennen ohjelman käyttöönottoa tutustua sen kansiorakenteeseen (kuva 39). Kansiossa "Esimerkit" on kaksi skriptiä: "esimerkki_1.m" ja "esimerkki_2.m", joiden avulla suoritetaan ohjelmaa varten laaditut esimerkkikonversiot. Kansiossa "Funktiot" on pääohjelma "paaohjelma.m" sekä konversio- ja muunnosfunktio "ellip_norm.m", "geod_suorak.m", "GK27_TM35.m", "maant_taso.m", "norm_ellip.m", "suorak_geod.m", "taso_maant.m" ja "TM35_GK27.m". Kansio "Koordinaatit_kirjoita" on tyhjä, sitä käytetään ohjelman kirjoittamien uusien tiedostojen sijoituspaikkana. Kansio "Koordinaatit_lue" sisältää esimerkkikonversioissa käytetyt lähtötiedot eli tiedostot "kaupungit_FINsuorak.csv" ja "kaupungit_TM35FIN.csv". Kansiossa "Kuvat" ovat luvussa 4.2 esitetyt kaavakuvat "suorak_taso.png", "taso_suorak.png" sekä "taso_taso.png", ja kansiossa "Kvasigeoidimalli" kvasigeoidimalli "FIN2005N00.lst".

**Tuplaklikkaa tiedostoa aloitus.m
aloittaaksesi ohjelman käytön**

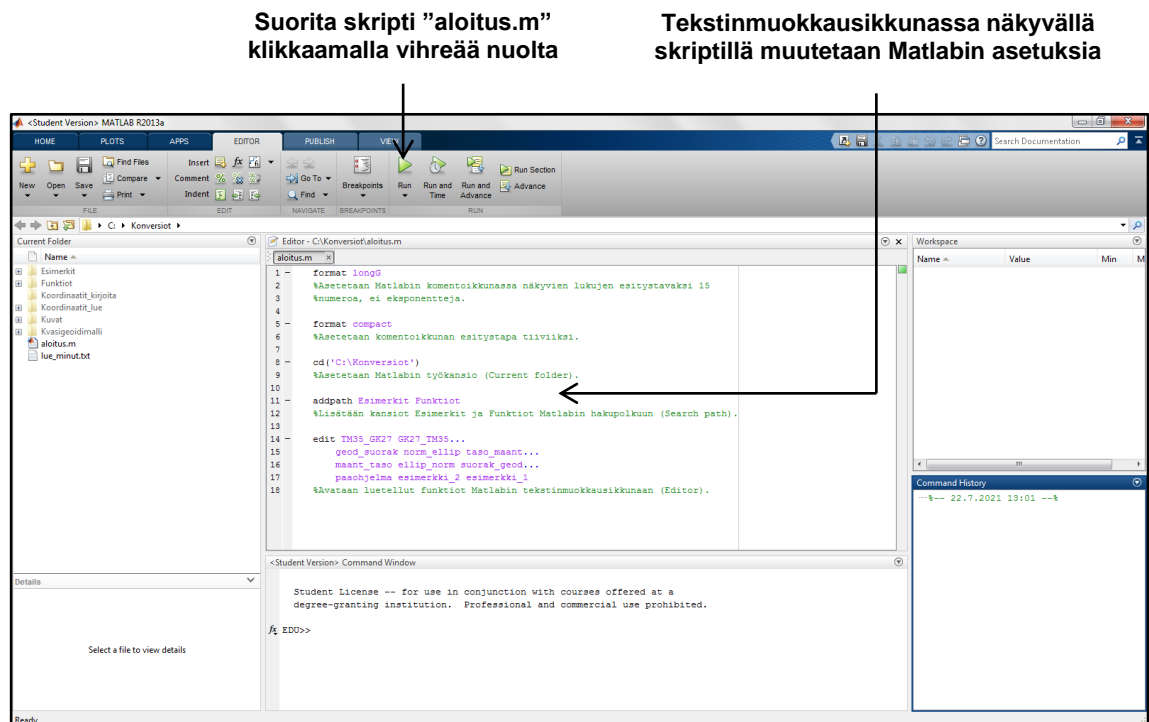


Kuva 39. Purettu zip-tiedoston "Konversiot.zip" hakemistorakenne.

Tiedosto "aloitus.m" sisältää skriptin, jonka avulla vaihdetaan Matlabin asetuksia sekä avataan funktiot ja esimerkit Matlabin tekstinmuokkausikkunaan. Konversio-ohjelman käytön voi aloittaa tuplaklikkaamalla sitä. Tiedostossa "lue_minut.txt" on ohjelman käyttöohje, joka on lyhennelmä tästä tekstistä. Saman käyttöohjeen englanninkielinen käännös on nähtävissä kuvassa 38.

4.3.2 Matlabin asetukset

Kun tiedostoa "aloitus.m" on tuplaklikattu, on samalla käynnistetty Matlab ja avattu tiedosto sen tekstinmuokkausikkunaan, joka näkyy alla keskellä (kuva 40). Skriptin voi suorittaa klikkaamalla Matlabin yläpalkista vihreää nuolta, jossa on teksti "Run". Mikäli alla olevassa kuvakaappauksessa näkyvää koodia on vaikea lukea, löytyy skripti myös tekstiversiona liitteen sivulta 26.



Kuva 40. Tiedosto aloitus.m on avattu Matlabin tekstinmuokausikkunaan.

Skriptin rivillä 1 vaihdetaan komentoikkunaan tulostuvien lukujen esitystapa mahdollisimman tarkaksi ja helppolukaiseksi, ja rivillä 5 ikkunan esitystapa asetetaan mahdollisimman tiiviiksi. Nämä asetukset eivät vaikuta konversio-ohjelman toimintaan, vaan ne on tarkoitettu helpottamaan virheiden etsintää, kun ohjelmaa muokataan. Rivillä 8 asetetaan Matlabin työkansioksi "C:\Konversiot". Asetus vaikuttaa erityisesti siihen, miten eri tiedostoihin voidaan jatkossa viitata. Jos zip-tiedosto on purettu johonkin muuhun kansioon, tämä tieto on vaihdettava. Rivillä 11 lisätään kansion "Konversiot" alikansiot "Esimerkit" ja "Funktiot" Matlabin hakupolkuun, jotta se osaa etsiä niistä skriptejä ja funktioita, ja rivillä 14 näistä kansioista löytyvät skriptit ja funktiot avataan tekstinmuokausikkunaan. Rivillä 11 kansiot "Esimerkit" ja "Funktiot" voitiin lisätä hakupolkuun pelkillä nimillään ilman täyttä kansiopolkua, koska ne sijaitsevat työkansion alla.

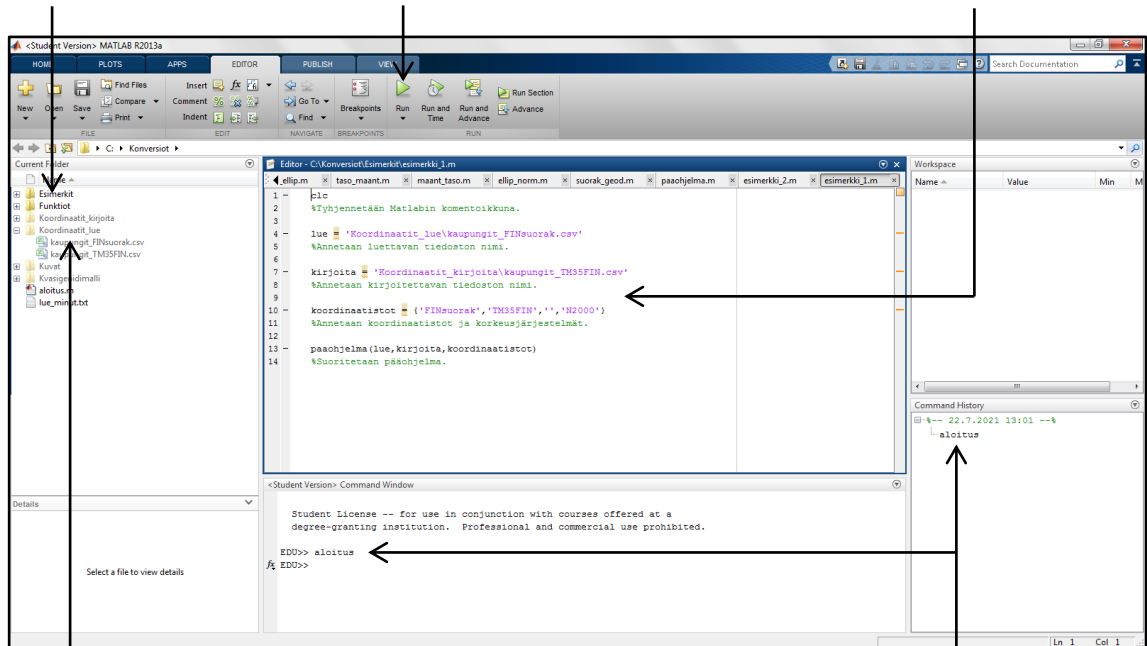
4.3.3 Esimerkkikonversion suorittaminen

Tiedoston "aloitus.m" suorittamisen jälkeen kansioista "Esimerkit" ja "Funktiot" luetut tiedostot ovat avautuneet tekstinmuokkausikkunaan (kuva 41), näkyvillä on skripti "esimerkki_1.m". Kuvasta huomataan myös, kuinka hakupolkuun lisätyt kansiot ovat aktiivisina vasemmalla työkansiossa, ja kuinka skriptin "aloitus.m" suorittaminen näkyy komentoikkunassa ja komentohistoriassa. Skriptin "Esimerkki_1.m" (liite 1, s. 27) voi suorittaa painamalla jälleen yläpalkista vihreää nuolta, jossa on teksti "Run".

**Kansiot "Esimerkit" ja "Funktio-
t" on lisätty hakupolkuun**

**Suorita skripti "esimerkki_1.m"
klikkaamalla vihreää nuolta**

**Funktioit ja skriptit on avattu
tekstinmuokkausikkunaan**



Kuva 41. Tiedosto aloitus.m on suoritettu, funktioit ja esimerkkikonversioiden skriptit ovat avautuneet tekstinmuokkausikkunaan.

**Ohjelma lukee tiedoston
"kaupungit_FINsuorak.csv"**

**Skriptin "aloitus.m" suorittaminen näkyy
komentoikkunassa ja komentohistoriassa**

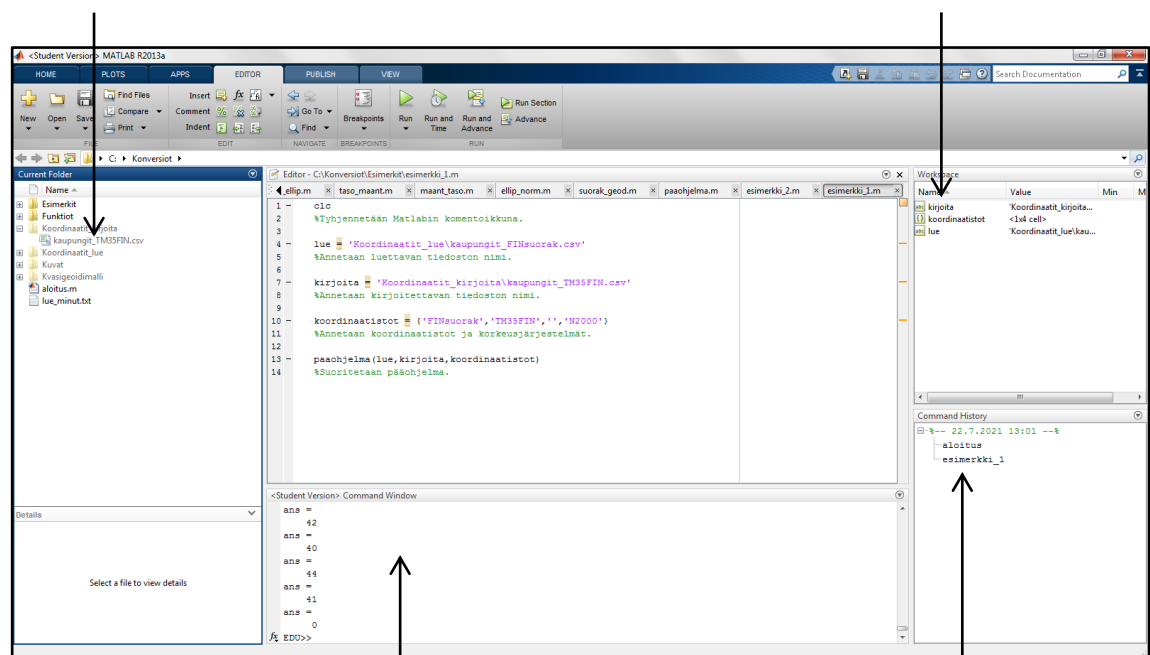
Skriptin rivillä 1 tyhjennetään Matlabin komentoikkuna aiemmin suoritetuista käskyistä, mikä helpottaa konversion aloittavien käskyjen löytämistä skriptin suorittamisen jälkeen. Rivillä 4 sijoitetaan muuttujaan "lue" alkuperäiset koordinaattitiedot sisältävän tiedoston "kaupungit_FINsuorak.csv" nimi, ja rivillä 7 muuttujaan "kirjoita" konvertoidut ja muunnetut koordinaatit sisältävän tiedoston "kaupungit_TM35FIN.csv" nimi. Rivillä 10 sijoitetaan solutaulukkaan "koordinaatit" konvertoitavien ja muunnettavien koordinaattien alkuperäiset ja tulevat koordinaatit ja korkeusjärjestelmät. Lopuksi rivillä 13 muuttujat ja solutaulukko annetaan syötteeksi pääohjelmalle.

4.3.4 Lopputuloksen tarkastelua

Kun tiedosto ”esimerkki_1.m” on suoritettu, huomataan kuvan 42 vasemmasta laidasta, että työkansion ”Konversiot” alikansioon ”Koordinaatit_kirjoita” on luotu uusi tiedosto ”kaupungit_TM35FIN.csv”. Tiedostoon kirjoitetut konvertoidut ja muunnetut koordinaatit ovat nähtävillä luvun 4.5 kuvassa 50, jossa niitä on verrattu Maanmittauslaitoksen paikkatietokeskuksen koordinaattimuunnospalvelussa vastaavalla tavalla laskettuihin koordinaatteihin. Alkuperäiset koordinaatit tiedostosta ”kaupungit_FINsuorak.csv” taas löytyvät liitteen sivulta 28.

Ohjelma on kirjoittanut tiedoston ”kaupungit_TM35FIN.csv”

Työtilaan on luotu muuttujat ”kirjoita” ja ”lue” sekä solutaulukko ”koordinaatitot”



Kuva 42. Konversio esimerkki_1.m on suoritettu, kirjoitettu tiedosto ”kaupungit_TM35FIN.csv” näkyy työkansion alikansiossa ”Koordinaatit_kirjoita”.

Laskennan välivaiheet ja funktioiden tulosteet näkyvät komentoikkunassa

Skriptin ”esimerkki_1.m” suorittaminen näkyy komentohistoriassa

Muita kuvassa 42 näkyviä tietoja ovat oikeassa yläkulmassa komentoikkunan työtilaan luodut muuttujat ”kirjoita” ja ”lue” sekä solutaulukko ”koordinaatitot”, oikean alakulman komentohistoriassa näkyvä suoritettu skripti ”esimerkki_1.m”, ja alhaalla keskellä olevaan komentoikkunaan ilmestyneet laskennan välivaiheet ja funktioiden tulosteet. Näkyvillä olevat luvut ”42, 40, 44, 41, 0” eivät liity laskentaan, vaan tarkoittavat pääohjelmassa käytetyn Matlabin sisäänrakennetun funktion ”fprintf” tiedoston ”kaupungit_TM35FIN.csv” viimeisille riveille kirjoittamien tavujen määrää (18).

4.4 Ohjelman laajentaminen

Sen jälkeen, kun lukija on tutustunut konversio-ohjelman toimintaperiaatteeseen ja uskoo ymmärtävänsä sen, on ohjelmaa mahdollista muokata omien tarpeiden mukaiseksi lisäämällä siihen uusia koordinaattikonversioita. Tämä tehdään yksinkertaisesti hankkimalla ensin jostain lähtökoordinaatistossa olevat konvertoitavat koordinaattitiedot ja kirjoittamalla ne csv-tiedostoon. Seuraavaksi katsotaan luvun 4.2 kuvasta 35, 36 tai 37 mitä funktioita tarvitaan, jotta konversio haluttuun koordinaatistoon voidaan suorittaa. Lopuksi laaditaan skripti, jolla syötetään tarvittavat lähtötiedot pääohjelmalle, ja luodaan pääohjelmaan uusi konversiopolku. Seuraavassa on esitetty kahden eri kaavakuviin perustuvan uuden konversion luominen. Konversiot on tehty muokkaamalla ohjelman mukana tulevia valmiita skriptejä ja konversiopolkuja.

```

1  clc
2  %Tyhjennetään Matlabin komentoikkuna.
3
4  lue = 'Koordinaatit_lue\Helsinki_FINsuorak.csv'
5  %Annetaan luettavan tiedoston nimi.
6
7  kirjoita = 'Koordinaatit_kirjoita\Helsinki_GK25.csv'
8  %Annetaan kirjoitettavan tiedoston nimi.
9
10 koordinaatistot = {'FINsuorak', 'GK25', '', 'N2000'}
11 %Annetaan koordinaatistot ja korkeusjärjestelmät.
12
13 paaohjelma(lue, kirjoita, koordinaatistot)
14 %Suoritetaan pääohjelma.

```

Kuva 43. Konversioskripti esimerkki_3.m, suorakulmaisten EUREF-FIN-koordinaattien konvertoiminen ja muuntaminen ETRS-GK25-tasokoordinaateiksi ja N2000-korkeuksiksi.

Ensin käydään läpi suorakulmaisten EUREF-FIN-koordinaattien konvertoiminen ja muuntaminen ETRS-GK25-tasokoordinaateiksi ja N2000-korkeuksiksi. Ohjelman mukana tullut tiedosto "kaupungit_FINsuorak.csv" (liite 1, s. 28) sisältää kymmenen kaupungin koordinaatit, joista ainoastaan Helsingin koordinaatit ovat GK25-karttaprojektion alueella. Poistetaan tiedostosta muut koordinaatit ja tallennetaan uusi tiedosto kansioon "Koordinaatit_lue" nimellä "Helsinki_GK25.csv". Luvun 4.2 kuvasta 35 nähdään, että konversion suorittamiseen tarvitaan funktiot "suorak_geo.m", "maant_taso.m" ja "ellip_norm.m", jotka ovat täsmälleen samat ja samassa järjestyksessä kuin esimerkissä 1. Konversiota varten tarvittava skripti ja pääohjelman konversiopolku voidaan laatia muokkaamalla skriptiä "esimerkki_1.m" (liite 1, s. 27) ja pääohjelman konversiopolkua "Esimerkki 1" (liite 1, s. 1–2). Kuvassa 43 nähdään skriptiin tehtävät muutokset, jotka

on korostettu keltaisella. Ne ovat rivillä 4 oleva luettava tiedosto "Helsinki_FINsuorak.csv", rivillä 7 oleva kirjoitettava tiedosto "Helsinki_GK25.csv" ja rivillä 10 oleva taso-

```

1 %Esimerkki 3, suorakulmaisten EUREF-FIN-koordinaattien konvertoiminen ja
2 %muuntaminen ETRS-GK25-tasokoordinaateiksi ja N2000-korkeuksiksi.
3
4 if strcmp(koordinaatistot,{'FINsuorak','GK25','','N2000'})
5
6     tasok = koordinaatistot(2)
7     %tasok on tasokoordinaatisto, johon konvertoidaan (GK25).
8
9     for rivi = 1:size(koordinaatit,1)
10
11         X = koordinaatit(rivi,1)
12         Y = koordinaatit(rivi,2)
13         Z = koordinaatit(rivi,3)
14         %X, Y ja Z ovat suorakulmaiset EUREF-FIN-koordinaatit.
15
16         [fii,lambda,h] = suorak_geod(X,Y,Z)
17         %fii ja lambda ovat geodeettinen leveys ja pituus. h on
18         %ellipsoidinen korkeus. Koordinaatit konvertoi funktio suorak_geod.
19
20         [N,E] = maant_taso(fii,lambda,tasok)
21         %N ja E ovat GK25-koordinaatiston pohjois- ja itäkoordinaatti,
22         %jotka konvertoi funktio maant_taso.
23
24         H = ellip_norm(fii,lambda,h)
25         %H on N2000-järjestelmän normaalikorkeus, jonka muuntaa funktio
26         %ellip_norm.
27
28         uusi_rivi = [N,E,H]
29         %Kerätään konvertoidut koordinaatit ja muunnetut korkeudet uuteen
30         %vektoriin.
31
32         uudet = [uudet;uusi_rivi]
33         %Lisätään uusi rivi kaikki konvertoidut koordinaatit ja muunnetut
34         %korkeudet sisältävään matriisiin uudet.
35
36     end
37
38     tiedosto = fopen(kirjoita,'w')
39     %Avataan uusi tiedosto kirjoitettavaksi.
40
41     for rivi = 1:size(uudet,1)
42
43         fprintf(tiedosto,'%s,%.4f,%.4f,%.4f\r\n',...
44             pistetunnukset{rivi},uudet(rivi,:))
45         %Kirjoitetaan uusi tiedosto
46
47     end
48     fclose(tiedosto)
49     %Suljetaan tiedosto.
50
51 end

```

Kuva 44. Esimerkki 3 pääohjelmassa: suorakulmaisten EUREF-FIN-koordinaattien konvertoiminen ja muuntaminen ETRS-GK25-tasokoordinaateiksi ja N2000-korkeuksiksi.

koordinaatisto ”GK25”, johon koordinaatit konvertoidaan. Uusi konversioskripti on tallennettu nimellä ”esimerkki_3.m”. Kuvassa 44 puolestaan näkyvät pääohjelman konversiopolkuun tehtävät muutokset, jotka riviä 4 lukuun ottamatta koskevat vain kommentteja, eivätkä vaikuta ohjelman toimintaan. Riville 4 on vaihdettu tieto ”GK25” eli tasokoordinaatisto johon koordinaattitiedot konvertoidaan, jotta pääohjelma osaa yhdistää suoritettavan skriptin oikeaan konversiopolkuun.

```

1  clc
2  %Tyhjennetään Matlabin komentoikkuna.
3
4  lue = 'Koordinaatit_lue\Helsinki_TM35FIN.csv'
5  %Annetaan luettavan tiedoston nimi.
6
7  kirjoita = 'Koordinaatit_kirjoita\Helsinki_GK25.csv'
8  %Annetaan kirjoitettavan tiedoston nimi.
9
10 koordinaatistot = {'TM35FIN', 'GK25', 'N2000', 'N2000'}
11 %Annetaan koordinaatistot ja korkeusjärjestelmät.
12
13 paaohjelma(lue, kirjoita, koordinaatistot)
14 %Suoritetaan pääohjelma.

```

Kuva 45. Konversioskripti esimerkki_4.m, TM35FIN-tasokoordinaattien konvertoiminen ETRS-GK25-tasokoordinaateiksi.

Toisena esimerkkinä tarkastellaan ETRS-TM35FIN-tasokoordinaattien konvertoimista ETRS-GK25 tasokoordinaateiksi, mikä vaatii hieman enemmän muokkaustoimenpiteitä. Koordinaattitiedot saadaan nyt tiedostosta ”kaupungit_TM35FIN.csv” (liite 1, s. 28), josta poistetaan jälleen muiden kaupunkien kuin Helsingin koordinaatit ja tallennetaan uusi tiedosto nimellä ”Helsinki_TM35FIN.csv”. Luvun 4.2 kuvasta 37 nähdään, että konversion suorittamiseen tarvitaan funktiot ”taso_maant.m” ja ”maant_taso.m”. Kuvasta 36 taas nähdään, että esimerkki 2 alkaa funktiolla ”taso_maant.m”, joten skriptiä ”esimerkki_2.m” (liite 1, s. 27) ja pääohjelman konversiopolkua ”Esimerkki 2” (liite 1, s. 3–4) voidaan käyttää uuden konversion aihiona. Kuvassa 45 näkyvät skriptiin tehtävät muutokset, jotka vastaavat edellistä konversiota, tiedot vain vaihtuvat. Uusi konversioskripti on tallennettu nimellä ”esimerkki_4.m”. Kuvassa 46 näkyvät pääohjelman konversiopolkuun tehtävät muutokset, joita on nyt enemmän kuin aiemmin. Konversiopolusta on riviltä 27 alkaen poistettu funktiokutsut funktioille ”norm_ellip.m” sekä ”geod_suorak.m”, ja korvattu ne funktiokutsulla funktiolle ”maant_taso.m”. Riville 9 taas on lisätty uusi muuttuja ”tasok2”, jonka avulla tälle funktiolle annetaan syötteenä skriptissä ”esimerkki_4.m” kerrottu koordinaatisto ”GK25” johon konvertoidaan. Korkeuksia ei muunneta lainkaan, vaan rivillä 31 ne kopioidaan suoraan uuteen tiedostoon muuttujan ”H” avulla.

```

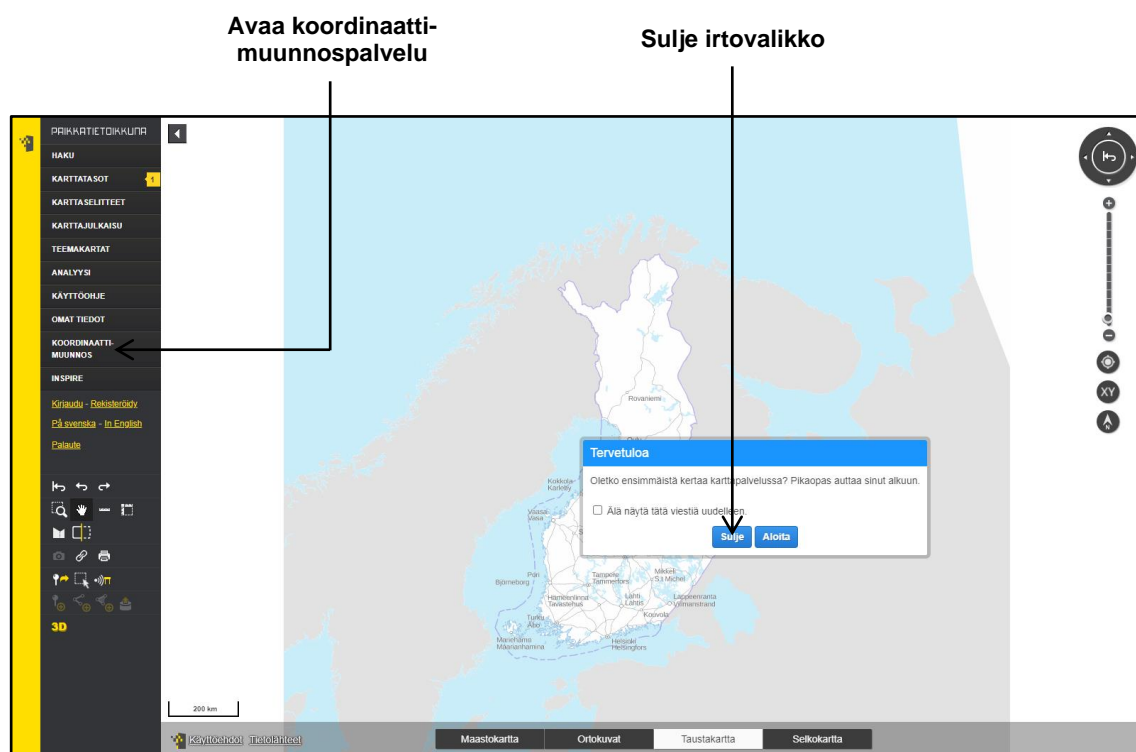
1 %Esimerkki 4, ETRS-TM35FIN-tasokoordinaattien konvertoiminen
2 %ETRS-GK25-tasokoordinaateiksi. N2000-korkeudet pysyvät muuttumattomina.
3
4 if strcmp(koordinaatitot,{'TM35FIN','GK25','N2000','N2000'})
5
6     tasok = koordinaatitot(1)
7     %tasok on tasokoordinaatisto, josta konvertoidaan (TM35FIN).
8
9     tasok2 = koordinaatitot(1)
10    %tasok2 on tasokoordinaatisto, johon konvertoidaan (GK25).
11
12    for rivi = 1:size(koordinaatit,1)
13
14        N = koordinaatit(rivi,1)
15        %N on TM35FIN-koordinaatiston pohjoiskoordinaatti.
16
17        E = koordinaatit(rivi,2)
18        %E on TM35FIN-koordinaatiston itäkoordinaatti.
19
20        H = koordinaatit(rivi,3)
21        %H on normaalikorkeus N2000-järjestelmässä.
22
23        [fii, lambda] = taso_maant(N,E,tasok)
24        %fii ja lambda ovat maantieteelliset koordinaatit, jotka konvertoi
25        %funktio taso_maant.
26
27        [N2,E2] = maant_taso(fii,lambda,tasok2)
28        %N2 ja E2 ovat GK25-koordinaatiston pohjois- ja itäkoordinaatti,
29        %jotka konvertoi funktio maant_taso.
30
31        uusi_rivi = [N2,E2,H]
32        %Kerätään konvertoitua koordinaatit uuteen vektoriin.
33
34        uudet = [uudet;uusi_rivi]
35        %Lisätään uusi rivi kaikki konvertoitua koordinaatit sisältävään
36        %matriisiin uudet.
37
38    end
39
40    tiedosto = fopen(kirjoita,'w')
41    %Avataan uusi tiedosto kirjoitettavaksi.
42
43    for rivi = 1:size(uudet,1)
44
45        fprintf(tiedosto,'%s,%.4f,%.4f,%.4f\r\n',...
46            pistetunnukset(rivi),uudet(rivi,:))
47        %Kirjoitetaan uusi tiedosto.
48
49    end
50
51    fclose(tiedosto)
52    %Suljetaan tiedosto.
53
54 end

```

Kuva 46. Esimerkki 4 pääohjelmassa: TM35FIN-tasokoordinaattien konvertoiminen ETRS-GK25-tasokoordinaateiksi.

4.5 Vertailu Paikkatietoikkunan koordinaattimuunnospalvelun kanssa

Aina sen jälkeen, kun lukija on luonut uuden konversiopolun pääohjelmaan ja suorittanut konversion haluamilleen koordinaateille, on lopputulos hyvä tarkistaa jossakin muussa ohjelmistossa. Tässä tarkistusta varten käytetään Paikkatietoikkunan koordinaattimuunnospalvelua, joka on itse asiassa verkkosivua vaihtanut vanha Geodeettisen laitoksen koordinaattimuunnospalvelu (coordtrans.fgi.fi). Tarkistuksen voisi hyvin tehdä tämän verkkopalvelun sijaan jollakin lukijan koneelle asennetulla paikkatieto-ohjelmistolla, kuten 3D-Win, QGIS tai ArcGIS. Nyt käytetty palvelu on kuitenkin valittu, koska toisin kuin työpöytäohjelmistot, se on varmasti kaikkien saatavilla, ja sen on tuottanut sama taho, jonka julkaisuihin konversio-ohjelma perustuu. (19.)



Kuva 47. Paikkatietoikkunan kotisivu (kartta.paikkatietoikkuna.fi).

Palvelun käytön voi aloittaa kirjoittamalla verkkoselaimen osoiteriville palvelun osoitteen (kartta.paikkatietoikkuna.fi). Selaimeen aukeaa palvelun kotisivu (kuva 47), jolla kylläkin "Tervetuloa"-irtovalikon voi sulkea, ja valita sivun vasemmassa laidassa olevasta valikosta linkin "koordinaattimuunnos". Selaimeen aukeaa nyt "Koordinaattimuunnos"-valikko, johon syötetään tarvittavat tiedot muunnosta varten riippuen koordinaattien geodeettisesta koordinaatti- ja korkeusjärjestelmästä sekä lähteen muodosta.

valita käytettävä tiedosto ja antaa täsmällisemmät tiedot sen muodosta. Tässä tapauksessa tiedostossa käytetty desimaalierotin on piste, sarake-erotin pilkku ja tiedosto sisältää tunnisteet eli kaupunkien nimet. Koordinaattitiedot voitaisiin tarvittaessa syöttää myös näppäimistöltä, mutta se on vaivalloista, kun kyse on useamman kuin yhden tai kahden pisteen koordinaateista. Koordinaattien sijainnin valitseminen kartalta taas on tarkoitettu tilanteeseen, jossa niitä ei ennalta tunneta lainkaan.

**Koordinaattien muuntaminen/
konvertointi selainikkunassa**

**Kirjoitettavan tiedoston
asetukset**

**Koordinaattien muuntaminen/
konvertointi tiedostoon**

Kuva 49. Koordinaattien tiedostosta lukeminen ja tiedostoon kirjoittaminen.

Kun tiedosto on valittu irtovalikosta ”Lähtöaineiston ominaisuudet”, ja sen tiedot syötetty, klikataan painiketta ”Valmis”, jolloin koordinaatit luetaan ”Koordinaattimuunnos”-valikon vasemmanpuoleiseen taulukkoon (kuva 49). Tämän jälkeen koordinaatit voidaan muuntaa kahdella tapaa: suoraan selainikkunassa klikkaamalla painiketta ”Muunna >>”, jolloin muunnetut koordinaatit tulevat näkyviin ”Koordinaattimuunnos”-valikon oikeanpuoleiseen taulukkoon, tai klikkaamalla painiketta ”Muunna tiedostoon”, jota käyttämällä koordinaatit voidaan arvatenkin kirjoittaa tiedostoon. Tälle kirjoitettavalle tiedostolle on annettava nimi ja tiedot sen muodosta samoin kuin luettavalle tiedostolle. Tämä tapahtuu ”Aineiston muodostaminen”-irtovalikosta, joka siis aukeaa ”Muunna

tiedostoon”-painikkeesta. Nyt kirjoitetaan tiedosto nimeltä ”Tarkistus.csv”, nimellä ei ole muunnoksen kannalta merkitystä, joten lukija voi valita omansa, miten haluaa. Tiedoston desimaalien tarkkuudeksi määritellään 0.1 mm, desimaalierottimeksi piste ja sarake-erottimeksi pilkku. Koordinaateille lisätään tunnisteet eli kaupunkien nimet, ja koordinaatit ovat käänteisessä järjestyksessä eli pohjoiskoordinaatti ensin. Rivierottimeksi on valittu oletuksena Windows/DOS, sitä ei tarvitse muuttaa. Palvelu haluaa tiedostoa tallentaessa lisätä nimeen päätteen ”.txt”, mutta sen voi poistaa, jolloin tallennetun tiedoston voi avata normaalisti taulukkolaskentaohjelmaan kuten MS Exceliin, mikäli Windowsin desimaali- ja sarake-erotin on määritelty samoin kuin edellä.

Tarkituslaskennassa koordinaatit ovat samat

Konversio-ohjelma				Paikkatietoikkuna				Erotus			
Kaupunki	N	E	H	Kaupunki	N	E	H	Kaupunki	ΔN	ΔE	ΔH
Helsinki	6672223.8515	385564.6371	6.9580	Helsinki	6672223.8515	385564.6371	6.9580	Helsinki	0.0000	0.0000	0.0000
Tampere	6822549.2002	327606.7107	89.8269	Tampere	6822549.2002	327606.7107	89.8269	Tampere	0.0000	0.0000	0.0000
Oulu	7211188.5050	427743.7217	0.8200	Oulu	7211188.5050	427743.7217	0.8200	Oulu	0.0000	0.0000	0.0000
Turku	6711043.0156	239742.7070	5.4870	Turku	6711043.0156	239742.7070	5.4870	Turku	0.0000	0.0000	0.0000
Jyväskylä	6901580.3989	434758.8535	96.8100	Jyväskylä	6901580.3989	434758.8535	96.8100	Jyväskylä	0.0000	0.0000	0.0000
Kuopio	6973793.6879	534485.1301	100.5080	Kuopio	6973793.6879	534485.1301	100.5080	Kuopio	0.0000	0.0000	0.0000
Joensuu	6944062.6335	641878.3710	80.9820	Joensuu	6944062.6335	641878.3710	80.9820	Joensuu	0.0000	0.0000	0.0000
Vaasa	7007660.6525	228381.1231	10.7430	Vaasa	7007660.6525	228381.1231	10.7430	Vaasa	0.0000	0.0000	0.0000
Rovaniemi	7376368.1566	443706.1761	85.2570	Rovaniemi	7376368.1566	443706.1761	85.2570	Rovaniemi	0.0000	0.0000	0.0000
Kokkola	7084826.7361	309804.7951	4.7760	Kokkola	7084826.7361	309804.7951	4.7760	Kokkola	0.0000	0.0000	0.0000

Kuva 50. Konversio-ohjelman ja koordinaattimuunnospalvelun tulosten vertailua MS Excelissä.

Kun muunnospalvelussa tehty tiedosto on tallennettu, avataan se Exceliin, ja tehdään samoin luvun 4.3 esimerkissä 1 kirjoitetulle tiedostolle ”kaupungit_TM35FIN.csv”. Siirretään tiedostojen sisältämät koordinaatit samaan laskentataulukkoon ja suoritetaan tarkistuslaskenta konversio-ohjelmalla tuotetuille koordinaateille (kuva 50). Kuvassa vasemmalla ovat konversio-ohjelman koordinaatit ja keskellä muunnospalvelun koordinaatit. Oikealla näkyvät koordinaattien erotukset (0.0000), jotka osoittavat koordinaattien olevan tismalleen samat, kuten pitääkin. Mikäli jonkin lukijan itse luoman konversiopolun tuottamat koordinaatit poikkeavat muunnospalvelun tuottamista koordinaateista pyöristysvirhettä enemmän, on jotakin tehty väärin, ja virhe on korjattava.

5 Yhteenveto

Tämän insinööriyön tarkoituksena oli ohjelmoida Matlab-ohjelmistolla joukko funktioita, joilla voidaan suorittaa ETRS89-koordinaattijärjestelmän kansalliseen realisaatioon EUREF-FIN liittyvät koordinaattikonversiot sekä N2000-normaalikorkeuksien ja ellipsoidisten EUREF-FIN-korkeuksien väliset korkeusmuunnokset. Työssä laadittiin opiskelun apuvälineeksi soveltuva helppokäyttöinen ohjelma ja siihen liittyvä kirjallinen osuus, jotka auttavat lukijaa ymmärtämään erityisesti satelliittimittausten kanssa tarvittavien koordinaattikonversioiden ja korkeusmuunnosten käyttöä.

Kun ajatus työn toteuttamisesta syntyi alkuvuodesta 2016, se muodostui vain koordinaattimuunnosohjelmasta ja mahdollisuudesta käyttää ohjelmaa opetusvälineenä. Työn tarkempi laajuus ja muoto eivät olleet vielä hahmottuneet. Kuten usein käy, tämänkin työn aiottu aihepiiri kutistui sitä pienemmäksi mitä pidemmälle lähdemateriaaliin tutustuminen eteni. Nopeasti kävi selväksi, että käsite koordinaattimuunnokset muodostaa valtavan laajan kokonaisuuden, ja aihepiiriä olisi pakko rajata. Lopulta työn aiheeksi tarkentuivat kaikkein yleisimmin käytettävät konversiot ja muunnokset kaikkein yleisimmässä asiayhteydessä: EUREF-FIN-koordinaattikonversiot sekä N2000-normaalikorkeuksien ja ellipsoidisten EUREF-FIN-korkeuksien väliset korkeusmuunnokset, joita jokainen maanmittari tulee huomaamattaan käyttäneeksi nykyisin lähes aina kun tarttuu satelliittimittalaitteeseen.

Työn toteutus alkoi lähdemateriaalin koostamisella kirjoittajan omaksi tietopakettiksi, jonka pohjalta muotoutui konversio-ohjelman lisäksi myöhemmin myös tämän kirjallisen osuuden toinen luku. Tärkeintä lähdemateriaalia olivat entisen Geodeettisen laitoksen nykyisen Maanmittauslaitoksen paikkatietokeskuksen julkaisut *Suomen geodeettiset koordinaatitot ja niiden väliset muunnokset*, kirjoittajina Pasi Häkli, Jyrki Puupponen, Hannu Koivula ja Markku Poutanen, sekä *Suomen geoidimallit ja niiden käyttäminen korkeuden muunnoksissa*, kirjoittajina Mirjam Bilker-Koivula ja Matti Ollikainen. Myöhemmin merkittäväksi lähteeksi muodostui myös Rovaniemen ammattikorkeakoulun julkaisu *Mittaus- ja kartoitustekniikan perusteet*, kirjoittajana Pasi Laurila. (3; 4; 5.)

Kun keskeinen lähdemateriaali oli koottu, oli mahdollista aloittaa työn käytännöllinen osuus, konversio-ohjelman toteutus. Ensimmäiseksi oli valittava käytettävä ohjelmointikieli. Varteenotettavia vaihtoehtoja oli kaksi: Python ja Matlab. Ensimmäinen on tulkattava ohjelmointikieli, jonka maanmittareille tutuin käyttökohte on skriptien ajaminen

paikkatieto-ohjelmistoissa kuten ArcGIS ja QGIS. Jälkimmäinen on numeeriseen laskentaan käytettävä ohjelmisto ja ohjelmointikieli, jota hyödynnetään erityisesti insinööri- ja luonnontieteiden parissa. Valinta kohdistui jälkimmäiseen, koska se oli saatavilla useammilla Metropolia ammattikorkeakoulun Leppävaaran toimipisteen tietokoneilla, olihan konversio-ohjelmaa tarkoitus käyttää oppimateriaalina. Ohjelmointityötä tukeväksi lähdemateriaaliksi valikoitui Matlab-ohjelmistoa valmistavan Math Works -yhtiön julkaisu *Matlab Primer* sekä Brian H. Hahnin ja Daniel T. Valentinin kirja *Essential Matlab* ja Stephen J. Chapmanin kirja *Matlab Programming for Engineers*. (13; 14; 15.)

Pian ohjelmointikielen valinnan jälkeen syntyi tärkein työn vaikeustasoa oppimateriaalina määrittävä päätös tehdä valmiiksi asti kaikki yksittäiset konversio- ja korkeusmuunnosfunktiot, mutta jättää ne toisiinsa liittävä pääohjelma keskeneräiseksi kahta esimerkkitapausta lukuun ottamatta. Lukijan tehtäväksi jäisi siis soveltaa tämän kirjallisen osuuden sisältämien kaavakuvien tietoja pääohjelmaan, toteuttaa näin tarvitsemansa konversiot ja samalla oppia mitä matemaattisesti tapahtuu, kun maastossa mitattavat kohteet siirretään kartalle.

Kattavan lähdemateriaalin ansiosta konversio-ohjelma valmistui kohtuullisen nopeassa aikataulussa, ja vastasi tekijänsä sille asettamia tavoitteita sekä matemaattisen oikeellisuuden että ohjelmallisen toimivuuden suhteen. Konversio-ohjelman lähdekoodi ja sen funktioiden suhteita kuvaavat kaavakuvat ovat vapaasti saatavilla internetistä Math Works -yhtiön tiedostonvaihtopalvelusta, mistä on toivottavasti hyötyä kotitehtävien ratkaisussa useammankin vuosikurssin opiskelijoille. Lähdekoodi löytyy myös tämän tekstin liitteestä, lukuun ottamatta Maanmittauslaitoksen paikkatietokeskuksen tuottamaa kvasigeoidimallia, jota tarvitaan korkeusmuunnosten suorittamiseen. (1; 2.)

Ohjelmointityön valmistuttua oli aika siirtyä tämän kirjoitetun osuuden pariin. Ylimääräistä haastetta työn loppuun saattamiseen toi kirjoittajan siirtyminen työelämään pian kirjoitustyön alettua, mikä erinäisten vaiheiden jälkeen johti tekstin viimeistelyn lykkäytymiseen aina kevääseen 2025 asti. Keskeinen kirjoitustyötä hidastanut seikka oli kirjoittajan havainto, että painettu teksti on väärä formaatti konversio-ohjelman toiminnan ja sen teoreettisen taustan esittelyyn. Yhtäältä siksi, että teoria on jo kirjoitettu kattavammin alkuperäisiin lähteisiin, mutta erityisesti siksi, että työn lähtökohtana oli kirjoittajan näkemys itse tekemisen paremmuudesta lukemiseen verrattuna.

Mikäli konversio-ohjelmaa tukeva kirjoitettu osuus pitäisi tehdä nyt uudestaan, se olisi-kin parempi toteuttaa mallioppimista edistävässä videomuodossa. Erityisesti tämä koskee lukua 4, jossa konversio-ohjelman toimintaa on kuvattu käytännön esimerkein. Ne olisi kertojäänellä täydennettynä voitu tallentaa suoraan tietokoneen näytöltä, sen sijaan, että sama esitetään nyt kömpelömmiin kuvina ja tekstinä. Sama koskee myös lukua 3, jossa on käsitelty Matlab-ohjelmiston perustoiminnallisuutta. Videoformaatin käyttö oli kuitenkin nyt mahdotonta geodesian perusteisiin keskittyvän luvun 2 vuoksi. Vaikka myös tältä osin olisi voitu päästä alkuperäislähteitä havainnollisempaan lopputuloksen, asian toteuttaminen olisi vaatinut kirjoittajalta tietokoneanimaation opettelua, mikä olisi tehnyt työmäärästä yhdelle hengelle aivan liian suuren.

Kirjoitustyön aikana nousi esiin myös useita muita kehityskohteita ja parannusehdotuksia, joita ei ollut nyt ajankäytöllisistä syistä mahdollista toteuttaa. Työn aihepiirin rajauksen sisältä näistä selvästi merkittävin oli konversio-ohjelmassa käytettyjen laskenta-kaavojen johtaminen yksinkertaisemmasta aineksesta, mikä palvelisi työn alkuperäistä tarkoitusta toimia aiempia oppimateriaaleja tehokkaampana johdatuksena aihepiiriin. Rajauksen ulkopuolelta työtä olisi järkevää laajentaa kattamaan eri satelliittipaikannusjärjestelmillä erilaisissa suorakulmaisissa koordinaatistoissa tehtyjen havaintojen muuntaminen samaan suorakulmaiseen koordinaatistoon, mistä ei kirjoittajan tietojen mukaan ole tällä hetkellä olemassa helposti omaksuttavia suomenkielisiä lähteitä.

Lähteet

- 1 Kansikas, Mikko. 2016. EUREF-FIN coordinate conversions and N2000 height transformation. Verkkoaineisto.
<se.mathworks.com/matlabcentral/fileexchange/96088-euref-fin-coordinate-conversions-and-n2000-height-transforma>. Luettu 22.3.2025.
- 2 FIN2005N00 geoidimalli. Maanmittauslaitos. Verkkoaineisto.
<www.maanmittauslaitos.fi/kartat-ja-paikkatieto/koordinaatit-ja-paikannus/koordinaattimuunnokset/fin2005n00-geoidimalli>. Luettu 22.3.2025.
- 3 Laurila, Pasi. 2012. Mittaus- ja kartoitustekniikan perusteet, 4. uudistettu painos. Verkkoaineisto. Rovaniemen ammattikorkeakoulu.
<ka.ramk.fi/eJulkaisut/D3_Mittaus%20ja%20kartoitustekniikan%20perusteet/RAMK_D3_lowress.pdf>. Luettu 1.4.2016.
- 4 Häkli, Pasi; Puupponen, Jyrki; Koivula, Hannu & Poutanen, Markku. 2009. Suomen geodeettiset koordinaatit ja niiden väliset muunnokset. Verkkoaineisto. Geodeettinen laitos.
<www.fgi.fi/fgi/sites/default/files/publications/gltiedote/GLtiedote30.pdf>. 10.12.2009. Luettu 1.4.2016.
- 5 Bilker-Koivula, Mirjam & Ollikainen, Matti. 2009. Suomen geoidimallit ja niiden käyttäminen korkeuden muunnoksissa. Verkkoaineisto. Geodeettinen laitos.
<www.fgi.fi/fgi/sites/default/files/publications/gltiedote/GLtiedote29.pdf>. Luettu 1.4.2016.
- 6 Albrecht, Jochen. 2005. The earth's shape is an ellipsoid. Verkkoaineisto.
<www.geo.hunter.cuny.edu/~jochen/gtech201/lectures/lec6concepts/Datums/The%20earths%20shape%20is%20an%20ellipsoid.htm>. 2005. Luettu 14.4.2023.
- 7 Vermeer, Martin. 2018. Geodesy. Verkkoaineisto.
<users.aalto.fi/~mvermeer/geobook.pdf>. 12.2.2018. Luettu 12.7.2021.
- 8 JHS 196 ETRS89-järjestelmän mukaiset koordinaatit Suomessa. 2016. Verkkoaineisto. JUHTA - Julkisen hallinnon tietohallinnon neuvottelukunta.
<www.suomidigi.fi/sites/default/files/2020-07/JHS196.doc>. 7.2.2016. Luettu 2.7.2023.
- 9 JHS 163 Suomen korkeusjärjestelmä N2000. 2019. Verkkoaineisto. JUHTA - Julkisen hallinnon tietohallinnon neuvottelukunta.
<www.suomidigi.fi/sites/default/files/2020-06/JHS163.doc> 10.10.2019. Luettu 8.1.2023.

- 10 Geoinformatiikan sanasto. 2014. Verkkoaineisto. Helsinki: Sanastokeskus TSK. <www.tsk.fi/tiedostot/pdf/GeoinformatiikanSanasto.pdf>. 17.4.2014. Luettu 2.4.2016.
- 11 Lehmuskoski, Pekka; Saaranen, Veikko; Takalo, Mikko & Rouhiainen, Paavo. 2008. Suomen kolmannen tarkkavaaituksen kiintopisteluetelo. Verkkoaineisto. Geodeettinen laitos. <helda.helsinki.fi/bitstream/handle/10138/346564/FGI_Publ_139.pdf> 2008. Luettu 2.7.2023.
- 12 Bilker-Koivula, Mirjam. 2010. Development of the Finnish Height Conversion Surface FIN2005N00. Verkkoaineisto. Nordic Journal of Surveying and Real Estate Research. <<https://journal.fi/njs/article/view/3663/3432>>. 27.10.2010. Luettu 15.1.2024.
- 13 Chapman, Stephen J. 2004. MATLAB Programming for Engineers, 3rd edition. Canada: Nelson, Thomson Canada.
- 14 Hahn, Brian H & Valentine, Daniel T. 2013. Essential MATLAB, 5th edition. United States of America: Academic Press, Elsevier
- 15 MATLAB Primer, 27th printing. 2016. Verkkoaineisto. Natick, Massachusetts. The Math Works Inc. <www.mathworks.com/help/pdf_doc/matlab/getstart.pdf>. Luettu 30.9.2016.
- 16 There exist triangular numbers that are also square. Bogomolny, Alexander. Verkkoaineisto. <www.cut-the-knot.org/do_you_know/triSquare2.shtml>. Luettu 30.3.2025.
- 17 Bilker-Koivula, Mirjam. 2007. FIN2005N00.Ist-geoiditiedoston selitys. Verkkoaineisto. Maanmittauslaitos. <www.maanmittauslaitos.fi/sites/maanmittauslaitos.fi/files/attachments/2019/02/IN2005N00_ist_lyhyt2.pdf>. 1.10.2007. Luettu 6.8.2021.
- 18 Math Works Help Center fprintf. Verkkoaineisto. The MathWorks Inc. <se.mathworks.com/help/matlab/ref/fprintf.html>. Luettu 28.7.2021.
- 19 Koordinaattimuunnospalvelu siirtynyt Paikkatietoikkunaan. Verkkoaineisto. Maanmittauslaitos. <www.maanmittauslaitos.fi/ajankohtaista/koordinaattimuunnospalvelu-siirtynyt-paikkatietoikkunaan>. Luettu 19.7.2021.

Kuvalähteet

1. Schmitt, Harrison. 1972. The Earth seen from Apollo 17.jpg. Verkkoaineisto. NASA. <upload.wikimedia.org/wikipedia/commons/9/97/The_Earth_seen_from_Apollo_17.jpg>. Luettu 8.4.2021.
2. Earth_s_geoid_as_seen_by_GOCE.tif. 2004. Verkkoaineisto. ESA. <www.esa.int/ESA_Multimedia/Images/2008/05/Earth_s_geoid_as_seen_by_GOCE>. Luettu 31.7.2021.
3. Bilker-Koivula, Mirjam & Ollikainen, Matti. 2009. Suomen geoidimallit ja niiden käyttäminen korkeuden muunnoksissa. Verkkoaineisto. Geodeettinen laitos. <www.fgi.fi/fgi/sites/default/files/publications/gltiedote/GLtiedote29.pdf>. Luettu 1.4.2016.
4. Kansikas, Mikko. 2023.
5. Häkli, Pasi; Puupponen, Jyrki; Koivula, Hannu & Poutanen, Markku. 2009. Suomen geodeettiset koordinaatistot ja niiden väliset muunnokset. Verkkoaineisto. Geodeettinen laitos. <www.fgi.fi/fgi/sites/default/files/publications/gltiedote/GLtiedote30.pdf>. 10.12.2009. Luettu 1.4.2016.
6. Koordinaatti- ja korkeusjärjestelmät. Verkkoaineisto. Maanmittauslaitos. <www.maanmittauslaitos.fi/tutkimus/tutkimustoiminta/tutkimusryhmat/paattyneet-tutkimusryhmat/koordinaattijarjestelmat-ja-0>. Luettu 27.1.2024.
7. Kansikas, Mikko. 2024.
8. JHS 163 Suomen korkeusjärjestelmä N2000. 2019. Verkkoaineisto. JUHTA - Julkisen hallinnon tietohallinnon neuvottelukunta. <www.suomidigi.fi/sites/default/files/2020-06/JHS163.doc> 10.10.2019. Luettu 8.1.2023.
9. Bilker-Koivula, Mirjam & Ollikainen, Matti. 2009. Suomen geoidimallit ja niiden käyttäminen korkeuden muunnoksissa. Verkkoaineisto. Geodeettinen laitos. <www.fgi.fi/fgi/sites/default/files/publications/gltiedote/GLtiedote29.pdf>. Luettu 1.4.2016.
10. Kansikas, Mikko. 2023.
11. Kansikas, Mikko. 2023.
12. Kansikas, Mikko. 2024.

13. Kansikas, Mikko. 2024.
14. Kansikas, Mikko. 2016.
15. The Math Works, Inc. Verkkoaineisto. Luettu 15.4.2016.
16. Kansikas, Mikko. 2018.
17. Kansikas, Mikko. 2018.
18. Kansikas, Mikko. 2018.
19. Kansikas, Mikko. 2022.
20. Kansikas, Mikko. 2022.
21. Kansikas, Mikko. 2022.
22. Kansikas, Mikko. 2022.
23. Kansikas, Mikko. 2022.
24. Kansikas, Mikko. 2022.
25. Kansikas, Mikko. 2022.
26. Kansikas, Mikko. 2022.
27. Kansikas, Mikko. 2021.
28. Kansikas, Mikko. 2022.
29. Kansikas, Mikko. 2022.
30. Kansikas, Mikko. 2018.
31. Kansikas, Mikko. 2021.
32. Kansikas, Mikko. 2021.
33. Kansikas, Mikko. 2018.
34. Kansikas, Mikko. 2022.

35. Kansikas, Mikko. 2016.
36. Kansikas, Mikko. 2016.
37. Kansikas, Mikko. 2016.
38. Kansikas, Mikko. 2021.
39. Kansikas, Mikko. 2021.
40. Kansikas, Mikko. 2021.
41. Kansikas, Mikko. 2021.
42. Kansikas, Mikko. 2021.
43. Kansikas, Mikko. 2021.
44. Kansikas, Mikko. 2021.
45. Kansikas, Mikko. 2021.
46. Kansikas, Mikko. 2021.
47. Kansikas, Mikko. 2021.
48. Kansikas, Mikko. 2021.
49. Kansikas, Mikko. 2021.
50. Kansikas, Mikko. 2021.

Pääohjelma (paaohjelma.m)

```

1 function [] = paaohjelma(lue,kirjoita,koordinaatistot)
2
3 %Kuvaus
4 %
5 %Ohjelmalla voidaan suorittaa ETRS89-koordinaattijärjestelmän kansalliseen
6 %realisaatioon EUREF-FIN liittyvät koordinaattikonversiot sekä
7 %N2000-normaalikorkeuksien ja ellipsoidisten EUREF-FIN-korkeuksien väliset
8 %korkeusmuunnokset.
9 %
10 %Konversioiden ja muunnosten suorittamiseen tarvitaan funktioita, jotka
11 %löytyvät tämän pääohjelman kanssa samasta kansiossa (Funktioit). Ohjelmaa
12 %varten on tehty valmiiksi kaksi esimerkkiä (Esimerkki 1, Esimerkki 2),
13 %joita voi kokeilla toisesta kansiossa (Esimerkit) löytyvien
14 %komentosarjojen eli skriptien avulla (esimerkki_1, esimerkki_2).
15
16 %Syöte
17 %
18 %lue on luettavan tiedoston nimi.
19 %kirjoita on kirjoitettavan tiedoston nimi.
20 %koordinaatistot sisältää tiedot konversioiden ja muunnosten lähtö- ja
21 %päätekoordinaatistoista sekä korkeusjärjestelmistä.
22
23
24
25
26
27 tiedosto = fopen(lue)
28 %Avataan käyttäjän antama tiedosto luettavaksi.
29
30 luettu = textscan(tiedosto,'%s %f %f %f','delimiter',' ','CollectOutput',1)
31 %Luetaan tiedosto.
32
33 pistetunnukset = luettu{1}
34 %Erotetaan pistetunnukset omaksi solutaulukokseen (cell array).
35
36 koordinaatit = luettu{2}
37 %Erotetaan koordinaatit omaksi matriisikseen.
38
39 fclose(tiedosto)
40 %Suljetaan tiedosto.
41
42 uudet = []
43 %Luodaan matriisi, johon kerätään uudet, konvertoidut koordinaatit ja
44 %muunnetut korkeudet.
45
46
47
48
49
50 %Esimerkki 1, suorakulmaisten EUREF-FIN-koordinaattien konvertoiminen ja

```

```

51 %muuntaminen ETRS-TM35FIN-tasokoordinaateiksi ja N2000-korkeuksiksi.
52
53 if strcmp(koordinaatistot,{'FINSuorak','TM35FIN',' ','N2000'})
54
55     tasok = koordinaatistot(2)
56     %tasok on tasokoordinaatisto, johon konvertoidaan (TM35FIN).
57
58     for rivi = 1:size(koordinaatit,1)
59
60         X = koordinaatit(rivi,1)
61         Y = koordinaatit(rivi,2)
62         Z = koordinaatit(rivi,3)
63         %X, Y ja Z ovat suorakulmaiset EUREF-FIN-koordinaatit.
64
65         [fii,lambda,h] = suorak_geod(X,Y,Z)
66         %fii ja lambda ovat geodeettinen leveys ja pituus. h on
67         %ellipsoidinen korkeus. Koordinaatit konvertoi funktio suorak_geod.
68
69         [N,E] = maant_taso(fii,lambda,tasok)
70         %N ja E ovat TM35FIN-koordinaatiston pohjois- ja itäkoordinaatti,
71         %jotka konvertoi funktio maant_taso.
72
73         H = ellip_norm(fii,lambda,h)
74         %H on N2000-järjestelmän normaalikorkeus, jonka muuntaa funktio
75         %ellip_norm.
76
77         uusi_rivi = [N,E,H]
78         %Kerätään konvertoidut koordinaatit ja muunnetut korkeudet uuteen
79         %vektoriin.
80
81         uudet = [uudet;uusi_rivi]
82         %Lisätään uusi rivi kaikki konvertoidut koordinaatit ja muunnetut
83         %korkeudet sisältävään matriisiin uudet.
84
85     end
86
87     tiedosto = fopen(kirjoita,'w')
88     %Avataan uusi tiedosto kirjoitettavaksi.
89
90     for rivi = 1:size(uudet,1)
91
92         fprintf(tiedosto,'%s,%.4f,%.4f,%.4f\r\n',...
93             pistetunnukset{rivi},uudet(rivi,:))
94         %Kirjoitetaan uusi tiedosto
95
96     end
97     fclose(tiedosto)
98     %Suljetaan tiedosto.
99
100 end

```

```

101
102
103
104
105
106 %Esimerkki 2, ETRS-TM35FIN-tasokoordinaattien ja N2000-korkeuksien
107 %konvertoiminen ja muuntaminen suorakulmaisiksi EUREF-FIN-koordinaateiksi.
108
109 if strcmp(koordinaatistot,{'TM35FIN','FINsuorak','N2000',''})
110
111     tasok = koordinaatistot(1)
112     %tasok on tasokoordinaatisto, josta konvertoidaan (TM35FIN).
113
114     for rivi = 1:size(koordinaatit,1)
115
116         N = koordinaatit(rivi,1)
117         %N on TM35FIN-koordinaatiston pohjoiskoordinaatti.
118
119         E = koordinaatit(rivi,2)
120         %E on TM35FIN-koordinaatiston itäkoordinaatti.
121
122         H = koordinaatit(rivi,3)
123         %H on normaalikorkeus N2000-järjestelmässä.
124
125         [fii, lambda] = taso_maant(N,E,tasok)
126         %fii ja lambda ovat maantieteelliset koordinaatit, jotka konvertoi
127         %funktio taso_maant.
128
129         h = norm_ellip(fii,lambda,H)
130         %h on ellipsoidinen korkeus, jonka muuntaa funktio norm_ellip.
131
132         [X,Y,Z] = geod_suorak(fii,lambda,h)
133         %X, Y ja Z ovat suorakulmaiset EUREF-FIN-koordinaatit, jotka
134         %konvertoi funktio geod_suorak.
135
136         uusi_rivi = [X,Y,Z]
137         %Kerätään konvertoitua koordinaatit uuteen vektoriin.
138
139         uudet = [uudet;uusi_rivi]
140         %Lisätään uusi rivi kaikki konvertoitua koordinaatit sisältävään
141         %matriisiin uudet.
142
143     end
144
145     tiedosto = fopen(kirjoita,'w')
146     %Avataan uusi tiedosto kirjoitettavaksi.
147
148     for rivi = 1:size(uudet,1)
149
150         fprintf(tiedosto,'%s,%.4f,%.4f,%.4f\r\n',...

```

```
151         pistetunnukset{rivi},uudet(rivi,:))
152         %Kirjoitetaan uusi tiedosto.
153
154     end
155
156     fclose(tiedosto)
157     %Suljetaan tiedosto.
158
159 end
160
161 end
```

Funktio (suorak_geod.m)

```

1 function [fii,lambda,h] = suorak_geod(X,Y,Z)
2
3 %Kuvaus
4 %
5 %Ohjelma konvertoi suorakulmaiset EUREF-FIN-koordinaatit geodeettisiksi
6 %EUREF-FIN-koordinaateiksi.
7
8 %Syöte
9 %
10 %X on x-akselin suuntainen suorakulmainen koordinaatti.
11 %Y on y-akselin suuntainen suorakulmainen koordinaatti.
12 %Z on z-akselin suuntainen suorakulmainen koordinaatti.
13
14 %Tuloste
15 %
16 %fii on geodeettinen leveys.
17 %lambda on geodeettinen pituus.
18 %h on ellipsoidinen korkeus.
19
20
21
22
23
24 %GRS80-vertausellipsoidin parametrit
25
26 a = 6378137.0
27 %a on ellipsoidin isoakselin puolikas.
28
29 f = 1/298.257222101
30 %f on ellipsoidin litistysuhde.
31
32
33
34
35
36 %Apusuureet
37
38 e = sqrt(2*f-f^2)
39 %e on ellipsoidin ensimmäinen eksentrisyys eli epäkeskisyyys.
40
41 fii = atan(Z/((1-e^2)*sqrt(X^2+Y^2)))
42 %Lasketaan geodeettisen leveyden likiarvo iterointia varten.
43
44
45
46
47
48 %Laskenta
49
50 if abs(fii) < 45*(pi()/180)

```

```

51
52     for i = 1:4
53
54         N = a/sqrt(1-e^2*(sin(fii))^2)
55         %Lasketaan ellipsoidin poikittaiskaarevuussäteen likiarvo.
56
57         h = sqrt(X^2+Y^2)/cos(fii)-N
58         %Lasketaan ellipsoidisen korkeuden likiarvo.
59
60         fii = atan((Z/sqrt(X^2+Y^2))*(1/(1-(e^2*N)/(N+h))))
61         %Lasketaan geodeettisen leveyden likiarvo.
62
63     end
64
65 else
66
67     for i = 1:4
68
69         N = a/sqrt(1-e^2*(sin(fii))^2)
70         %Lasketaan ellipsoidin poikittaiskaarevuussäteen likiarvo.
71
72         h = Z/sin(fii)-(1-e^2)*N
73         %Lasketaan ellipsoidisen korkeuden likiarvo.
74
75         fii = atan((Z/sqrt(X^2+Y^2))*(1/(1-(e^2*N)/(N+h))))
76         %Lasketaan geodeettisen leveyden likiarvo.
77
78     end
79
80 end
81
82 %Lasketaan geodeettinen pituus.
83
84 if X == 0 & Y > 0
85     lambda = pi()/2
86 elseif X == 0 & Y < 0
87     lambda = -pi()/2
88 elseif X < 0 & Y >= 0
89     lambda = atan(Y/X)+pi()
90 elseif X < 0 & Y < 0
91     lambda = atan(Y/X)-pi()
92 else
93     lambda=atan(Y/X);
94 end
95
96
97
98
99
100 %Yksikkömuunnos
101
102 fii = fii*(180/pi())
103 %Muunnetaan geodeettinen leveys radiaaneista asteiksi.
104
105 lambda = lambda*(180/pi())
106 %Muunnetaan geodeettinen pituus radiaaneista asteiksi.
107
108 end

```

Funktio (maant_taso.m)

```

1 function [N,E] = maant_taso(fii,lambda, tasok)
2
3 %Kuvaus
4 %
5 %Ohjelma konvertoi maantieteelliset EUREF-FIN-koordinaatit ETRS-TM35FIN-
6 %tai ETRS-GKn-tasokoordinaateiksi.
7
8 %Syöte
9 %
10 %fii on maantieteellinen leveys.
11 %lambda on maantieteellinen pituus.
12 %tasok on tasokoordinaatiston nimi.
13
14 %Tuloste
15 %
16 %N on projektion pohjoiskoordinaatti.
17 %E on projektion itäkoordinaatti.
18
19
20
21
22
23 %GRS80-vertausellipsoidin parametrit
24
25 a = 6378137.0
26 %a on ellipsoidin isoakselin puolikas.
27
28 f = 1/298.257222101
29 %f on ellipsoidin litistyssuhde.
30
31
32
33
34
35 %Karttaprojektion parametrit, jotka säilyvät samana kaikkien
36 %tasokoordinaatistojen kanssa.
37
38 E_0 = 500000
39 %E_0 on itäkoordinaatin arvo keskimeridiaanilla.
40
41
42
43
44
45 %Karttaprojektion parametrit, jotka vaihtelevat tasokoordinaatistoittain.
46
47 if strcmp(tasok, 'TM35FIN')
48
49     k_0 = 0.9996
50     %k_0 on mittakaavakerroin keskimeridiaanilla.

```

```
51
52     lambda_0 = 27*(pi()/180)
53     %lambda_0 on projektion keskimeridiaani.
54
55 elseif strcmp(tasok, 'GK19')
56
57     k_0 = 1
58     %k_0 on mittakaavakerroin keskimeridiaanilla.
59
60     lambda_0 = 19*(pi()/180)
61     %lambda_0 on projektion keskimeridiaani.
62
63 elseif strcmp(tasok, 'GK20')
64
65     k_0 = 1
66     %k_0 on mittakaavakerroin keskimeridiaanilla.
67
68     lambda_0 = 20*(pi()/180)
69     %lambda_0 on projektion keskimeridiaani.
70
71 elseif strcmp(tasok, 'GK21')
72
73     k_0 = 1
74     %k_0 on mittakaavakerroin keskimeridiaanilla.
75
76     lambda_0 = 21*(pi()/180)
77     %lambda_0 on projektion keskimeridiaani.
78
79 elseif strcmp(tasok, 'GK22')
80
81     k_0 = 1
82     %k_0 on mittakaavakerroin keskimeridiaanilla.
83
84     lambda_0 = 22*(pi()/180)
85     %lambda_0 on projektion keskimeridiaani.
86
87 elseif strcmp(tasok, 'GK23')
88
89     k_0 = 1
90     %k_0 on mittakaavakerroin keskimeridiaanilla.
91
92     lambda_0 = 23*(pi()/180)
93     %lambda_0 on projektion keskimeridiaani.
94
95 elseif strcmp(tasok, 'GK24')
96
97     k_0 = 1
98     %k_0 on mittakaavakerroin keskimeridiaanilla.
99
100    lambda_0 = 24*(pi()/180)
```

```
101     %lambda_0 on projektion keskimeridiaani.
102
103 elseif strcmp(tasok, 'GK25')
104
105     k_0 = 1
106     %k_0 on mittakaavakerroin keskimeridiaanilla.
107
108     lambda_0 = 25*(pi()/180)
109     %lambda_0 on projektion keskimeridiaani.
110
111 elseif strcmp(tasok, 'GK26')
112
113     k_0 = 1
114     %k_0 on mittakaavakerroin keskimeridiaanilla.
115
116     lambda_0 = 26*(pi()/180)
117     %lambda_0 on projektion keskimeridiaani.
118
119 elseif strcmp(tasok, 'GK27')
120
121     k_0 = 1
122     %k_0 on mittakaavakerroin keskimeridiaanilla.
123
124     lambda_0 = 27*(pi()/180)
125     %lambda_0 on projektion keskimeridiaani.
126
127 elseif strcmp(tasok, 'GK28')
128
129     k_0 = 1
130     %k_0 on mittakaavakerroin keskimeridiaanilla.
131
132     lambda_0 = 28*(pi()/180)
133     %lambda_0 on projektion keskimeridiaani.
134
135 elseif strcmp(tasok, 'GK29')
136
137     k_0 = 1
138     %k_0 on mittakaavakerroin keskimeridiaanilla.
139
140     lambda_0 = 29*(pi()/180)
141     %lambda_0 on projektion keskimeridiaani.
142
143 elseif strcmp(tasok, 'GK30')
144
145     k_0 = 1
146     %k_0 on mittakaavakerroin keskimeridiaanilla.
147
148     lambda_0 = 30*(pi()/180)
149     %lambda_0 on projektion keskimeridiaani.
150
```

```

151 elseif strcmp(tasok, 'GK31')
152
153     k_0 = 1
154     %k_0 on mittakaavakerroin keskimeridiaanilla.
155
156     lambda_0 = 31*(pi()/180)
157     %lambda_0 on projektion keskimeridiaani.
158
159 end
160
161
162
163
164
165 %Apusuureet
166
167 n = f/(2-f)
168 %n on toinen litistyssuhde.
169
170 e = sqrt(2*f-f^2) %mikä on f?
171 %e on ensimmäinen epäkeskisyyys eli eksentrisyys.
172
173 h_1p = (1/2)*n-(2/3)*n^2+(5/16)*n^3+(41/180)*n^4 %mikä on n?
174 h_2p = (13/48)*n^2-(3/5)*n^3+(557/1440)*n^4
175 h_3p = (61/240)*n^3-(103/140)*n^4
176 h_4p = (49561/161280)*n^4
177
178 A_1 = a/(1+n)*(1+n^2/4+n^4/64) %mikä on a?
179 %A_1 on meridiaanin pituisen ympyrän säde.
180
181
182
183
184
185 %Laskenta
186
187 fii = fii*(pi()/180)
188 %Muunnetaan maantieteellinen leveys asteista radiaaneiksi.
189
190 lambda = lambda*(pi()/180)
191 %Muunnetaan maantieteellinen pituus asteista radiaaneiksi.
192
193 Q_p = asinh(tan(fii))
194 Q_pp = atanh(e*sin(fii))
195 Q = Q_p-e*Q_pp
196
197 l = lambda-lambda_0
198 beeta = atan(sinh(Q))
199 eeta_p = atanh(cos(beeta)*sin(l))
200 ksii_p = asin(sin(beeta)/sech(eeta_p))

```

```
201
202 ksii_1 = h_1p*sin(2*ksii_p)*cosh(2*eeta_p)
203 ksii_2 = h_2p*sin(4*ksii_p)*cosh(4*eeta_p)
204 ksii_3 = h_3p*sin(6*ksii_p)*cosh(6*eeta_p)
205 ksii_4 = h_4p*sin(8*ksii_p)*cosh(8*eeta_p)
206
207 eeta_1 = h_1p*cos(2*ksii_p)*sinh(2*eeta_p)
208 eeta_2 = h_2p*cos(4*ksii_p)*sinh(4*eeta_p)
209 eeta_3 = h_3p*cos(6*ksii_p)*sinh(6*eeta_p)
210 eeta_4 = h_4p*cos(8*ksii_p)*sinh(8*eeta_p)
211
212 ksii = ksii_p+ksii_1+ksii_2+ksii_3+ksii_4
213 eeta = eeta_p+eeta_1+eeta_2+eeta_3+eeta_4
214
215 N = A_1*ksii*k_0
216 %Lasketaan projektion pohjoiskoordinaatti.
217
218 E = A_1*eeta*k_0+E_0
219 %Lasketaan projektion itäkoordinaatti.
220
221
222
223
224
225 %Lisätään projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
226 %alkuun.
227
228 if strcmp(tasok,'GK19')
229     E = E+19000000
230 elseif strcmp(tasok,'GK20')
231     E = E+20000000
232 elseif strcmp(tasok,'GK21')
233     E = E+21000000
234 elseif strcmp(tasok,'GK22')
235     E = E+22000000
236 elseif strcmp(tasok,'GK23')
237     E = E+23000000
238 elseif strcmp(tasok,'GK24')
239     E = E+24000000
240 elseif strcmp(tasok,'GK25')
241     E = E+25000000
242 elseif strcmp(tasok,'GK26')
243     E = E+26000000
244 elseif strcmp(tasok,'GK27')
245     E = E+27000000
246 elseif strcmp(tasok,'GK28')
247     E = E+28000000
248 elseif strcmp(tasok,'GK29')
249     E = E+29000000
250 elseif strcmp(tasok,'GK30')
251     E = E+30000000
252 elseif strcmp(tasok,'GK31')
253     E = E+31000000
254 end
255
256 end
```

Funktio (ellip_norm.m)

```

1 function [H] = ellip_norm(fii,lambda,h)
2
3 %Kuvaus
4 %
5 %Ohjelma muuntaa ellipsoidiset korkeudet eli korkeudet
6 %GRS80-vertausellipsoidin pinnalta N2000-korkeuksiksi. Muunnosta varten
7 %tarvitaan hilaruudukko, jonka pisteiden koordinaatit löytyvät tiedostosta
8 %FIN2005N00.lst.
9
10 %Syöte
11 %
12 %fii on maantieteellinen leveys.
13 %lambda on maantieteellinen pituus.
14 %h on ellipsoidinen korkeus.
15
16 %Tuloste
17 %
18 %H on normaalikorkeus N2000-korkeusjärjestelmässä.
19
20
21
22
23
24 %Lasketaan hilaruudun kulmapisteiden maantieteelliset koordinaatit asteina.
25
26 fii_max = ceil(fii*100/2)/100*2
27 %Suurin maantieteellinen leveys.
28
29 fii_min = floor(fii*100/2)/100*2
30 %Pienin maantieteellinen leveys.
31
32 lambda_max = ceil(lambda*100/4)/100*4
33 %Suurin maantieteellinen pituus.
34
35 lambda_min = floor(lambda*100/4)/100*4
36 %Pienin maantieteellinen pituus.
37
38
39
40
41
42 %Lasketaan niiden rivien numerot, joilta hilaruudun pisteiden koordinaatit
43 %löytyvät tiedostosta FIN2005N00.lst. Rivejä on yhteensä neljä, mutta niitä
44 %on aina kaksi peräkkäin, joten vain kahden rivin numerot tarvitsee laskea.
45
46 delta_fii = 70.70-fii_max
47 delta_lambda = lambda_min-17.48
48 rivi_1 = (389/0.02)*delta_fii+(1/0.04)*delta_lambda+1
49 %Sen rivin numero, josta löytyvät hilaruudun luoteiskulman eli vasemman
50 %ylänurkan koordinaatit.

```

```

51
52 delta_fii = 70.70-fii_min
53 rivi_3 = (389/0.02)*delta_fii+(1/0.04)*delta_lambda+1
54 %Sen rivin numero, josta löytyvät hilaruudun lounaiskulman eli vasemman
55 %alanurkan koordinaatit.
56
57
58
59
60
61 %Haetaan hilaruudun kulmapisteiden kvasigeoidin korkeudet tiedostosta
62 %FIN2005N00.lst.
63
64 tiedosto = fopen('Kvasigeoidimalli\FIN2005N00.lst')
65 rivit_12 = textscan(tiedosto,'%f %f %f',2,'MultipleDelimsAsOne',1,...
66     'HeaderLines',round(rivi_1-1))
67
68 zeeta_1 = rivit_12{3}(1)
69 %Hilaruudun vasemman yläkulman kvasigeoidin korkeus.
70
71 zeeta_2 = rivit_12{3}(2)
72 %Hilaruudun oikean yläkulman kvasigeoidin korkeus.
73
74 rivit_34 = textscan(tiedosto,'%f %f %f',2,'MultipleDelimsAsOne',1,...
75     'HeaderLines',round(rivi_3-rivi_1-1))
76
77 zeeta_3 = rivit_34{3}(1)
78 %Hilaruudun vasemman alakulman kvasigeoidin korkeus.
79
80 zeeta_4 = rivit_34{3}(2)
81 %Hilaruudun oikean alakulman kvasigeoidin korkeus.
82
83 fclose(tiedosto)
84
85
86
87
88
89 %Bilineaarinen interpolointi hilassa.
90
91 fii_norm = (fii-fii_min)/0.02
92 lambda_norm = (lambda-lambda_min)/0.04
93 %Lasketaan normalisoitu sijainti hilaruudun sisällä. Sijainnit fii_norm ja
94 %lambda_norm ovat suhteellisia arvoja välillä 0-1.
95
96 zeeta = fii_norm*(1-lambda_norm)*zeeta_1...
97     +fii_norm*lambda_norm*zeeta_2...
98     +(1-fii_norm)*(1-lambda_norm)*zeeta_3...
99     +(1-fii_norm)*lambda_norm*zeeta_4
100 % Lasketaan kvasigeoidin korkeus zeeta.
101
102
103
104
105
106 H = h-zeeta
107 %Lasketaan normaalikorkeus H.
108
109 end

```

Funktio (taso_maant.m)

```
1 function [fii,lambda] = taso_maant(N,E,tasok)
2
3 %Kuvaus
4 %
5 %Ohjelma konvertoi ETRS-TM35FIN- ja ETRS-GKn-tasokoordinaatit
6 %maantieteellisiksi EUREF-FIN-koordinaateiksi.
7
8 %Syöte
9 %
10 %N on projektion pohjoiskoordinaatti.
11 %E on projektion itäkoordinaatti.
12 %tasok on tasokoordinaatiston nimi.
13
14 %Tuloste
15 %
16 %fii on maantieteellinen leveys.
17 %lambda on maantieteellinen pituus.
18
19
20
21
22
23 %GRS80-vertausellipsoidin parametrit
24
25 a = 6378137.0
26 %a on ellipsoidin isoakselin puolikas.
27
28 f = 1/298.257222101
29 %f on ellipsoidin litistyssuhde.
30
31
32
33
34
35 %Karttaprojektion parametrit, jotka säilyvät samana kaikkien
36 %tasokoordinaatistojen kanssa.
37
38 E_0 = 500000
39 %E_0 on itäkoordinaatin arvo keskimeridiaanilla.
40
41
42
43
44
45 %Karttaprojektion parametrit, jotka vaihtelevat tasokoordinaatistoittain.
46
47 if strcmp(tasok,'TM35FIN')
48
49     k_0 = 0.9996
50     %k_0 on mittakaavakerroin keskimeridiaanilla.
```

```
51
52     lambda_0 = 27*(pi()/180)
53     %lambda_0 on projektion keskimeridiaani.
54
55 elseif strcmp(tasok, 'GK19')
56
57     k_0 = 1
58     %k_0 on mittakaavakerroin keskimeridiaanilla.
59
60     lambda_0 = 19*(pi()/180)
61     %lambda_0 on projektion keskimeridiaani.
62
63     E = E-19000000
64     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
65     %alusta.
66
67 elseif strcmp(tasok, 'GK20')
68
69     k_0 = 1
70     %k_0 on mittakaavakerroin keskimeridiaanilla.
71
72     lambda_0 = 20*(pi()/180)
73     %lambda_0 on projektion keskimeridiaani.
74
75     E = E-20000000
76     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
77     %alusta.
78
79 elseif strcmp(tasok, 'GK21')
80
81     k_0 = 1
82     %k_0 on mittakaavakerroin keskimeridiaanilla.
83
84     lambda_0 = 21*(pi()/180)
85     %lambda_0 on projektion keskimeridiaani.
86
87     E = E-21000000
88     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
89     %alusta.
90
91 elseif strcmp(tasok, 'GK22')
92
93     k_0 = 1
94     %k_0 on mittakaavakerroin keskimeridiaanilla.
95
96     lambda_0 = 22*(pi()/180)
97     %lambda_0 on projektion keskimeridiaani.
98
99     E = E-22000000
100    %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
```

```
101     %alusta.
102
103 elseif strcmp(tasok, 'GK23')
104
105     k_0 = 1
106     %k_0 on mittakaavakerroin keskimeridiaanilla.
107
108     lambda_0 = 23*(pi()/180)
109     %lambda_0 on projektion keskimeridiaani.
110
111     E = E-23000000
112     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
113     %alusta.
114
115 elseif strcmp(tasok, 'GK24')
116
117     k_0 = 1
118     %k_0 on mittakaavakerroin keskimeridiaanilla.
119
120     lambda_0 = 24*(pi()/180)
121     %lambda_0 on projektion keskimeridiaani.
122
123     E = E-24000000
124     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
125     %alusta.
126
127 elseif strcmp(tasok, 'GK25')
128
129     k_0 = 1
130     %k_0 on mittakaavakerroin keskimeridiaanilla.
131
132     lambda_0 = 25*(pi()/180)
133     %lambda_0 on projektion keskimeridiaani.
134
135     E = E-25000000
136     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
137     %alusta.
138
139 elseif strcmp(tasok, 'GK26')
140
141     k_0 = 1
142     %k_0 on mittakaavakerroin keskimeridiaanilla.
143
144     lambda_0 = 26*(pi()/180)
145     %lambda_0 on projektion keskimeridiaani.
146
147     E = E-26000000
148     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
149     %alusta.
150
```

```
151 elseif strcmp(tasok, 'GK27')
152
153     k_0 = 1
154     %k_0 on mittakaavakerroin keskimeridiaanilla.
155
156     lambda_0 = 27*(pi()/180)
157     %lambda_0 on projektion keskimeridiaani.
158
159     E = E-27000000
160     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
161     %alusta.
162
163 elseif strcmp(tasok, 'GK28')
164
165     k_0 = 1
166     %k_0 on mittakaavakerroin keskimeridiaanilla.
167
168     lambda_0 = 28*(pi()/180)
169     %lambda_0 on projektion keskimeridiaani.
170
171     E = E-28000000
172     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
173     %alusta.
174
175 elseif strcmp(tasok, 'GK29')
176
177     k_0 = 1
178     %k_0 on mittakaavakerroin keskimeridiaanilla.
179
180     lambda_0 = 29*(pi()/180)
181     %lambda_0 on projektion keskimeridiaani.
182
183     E = E-29000000
184     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
185     %alusta.
186
187 elseif strcmp(tasok, 'GK30')
188
189     k_0 = 1
190     %k_0 on mittakaavakerroin keskimeridiaanilla.
191
192     lambda_0 = 30*(pi()/180)
193     %lambda_0 on projektion keskimeridiaani.
194
195     E = E-30000000
196     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
197     %alusta.
198
199 elseif strcmp(tasok, 'GK31')
200
```

```

201     k_0 = 1
202     %k_0 on mittakaavakerroin keskimeridiaanilla.
203
204     lambda_0 = 31*(pi()/180)
205     %lambda_0 on projektion keskimeridiaani.
206
207     E = E-31000000
208     %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
209     %alusta.
210
211 end
212
213
214
215
216
217 %Apusuureet
218
219 n = f/(2-f)
220 %n on toinen litistyssuhde.
221
222 e = sqrt(2*f-f^2)
223 %e on ellipsoidin ensimmäinen eksentrisyys eli epäkeskisyyys.
224
225 h_1 = (1/2)*n-(2/3)*n^2+(37/96)*n^3-(1/360)*n^4
226 h_2 = (1/48)*n^2+(1/15)*n^3-(437/1440)*n^4
227 h_3 = (17/480)*n^3-(37/840)*n^4
228 h_4 = (4397/161280)*n^4
229
230 A_1 = a/(1+n)*(1+n^2/4+n^4/64)
231 %A_1 on meridiaanin pituisen ympyrän säde.
232
233
234
235
236
237 %Laskenta
238
239 ksii = N/(A_1*k_0)
240 eeta = (E-E_0)/(A_1*k_0)
241
242 ksii_1p = h_1*sin(2*ksii)*cosh(2*eeta)
243 ksii_2p = h_2*sin(4*ksii)*cosh(4*eeta)
244 ksii_3p = h_3*sin(6*ksii)*cosh(6*eeta)
245 ksii_4p = h_4*sin(8*ksii)*cosh(8*eeta)
246
247 eeta_1p = h_1*cos(2*ksii)*sinh(2*eeta)
248 eeta_2p = h_2*cos(4*ksii)*sinh(4*eeta)
249 eeta_3p = h_3*cos(6*ksii)*sinh(6*eeta)
250 eeta_4p = h_4*cos(8*ksii)*sinh(8*eeta)

```

```
251
252 ksii_p = ksii-(ksii_1p+ksii_2p+ksii_3p+ksii_4p)
253 eeta_p = eeta-(eeta_1p+eeta_2p+eeta_3p+eeta_4p)
254
255 beeta = asin(sech(eeta_p)*sin(ksii_p))
256 l = asin(tanh(eeta_p)/cos(beeta))
257
258 Q = asinh(tan(beeta))
259 Q_p = Q+e*atanh(e*tanh(Q))
260
261 for i = 1:3
262     Q_p = Q+e*atanh(e*tanh(Q_p))
263 end
264
265 fii = atan(sinh(Q_p))
266 %Lasketaan maantieteellinen leveys radiaaneina.
267
268 lambda = lambda_0+l
269 %Lasketaan maantieteellinen pituus radiaaneina.
270
271
272
273
274
275 %Yksikkömuunnos
276
277 fii = fii*(180/pi())
278 %Muunnetetaan maantieteellinen leveys radiaaneista asteiksi.
279
280 lambda = lambda*(180/pi())
281 %Muunnetaan maantieteellinen pituus radiaaneista asteiksi.
282
283 end
```

Funktio (norm_ellip.m)

```

1 function [h] = norm_ellip(fii,lambda,H)
2
3 %Kuvaus
4 %
5 %Ohjelma muuntaa N2000-korkeudet ellipsoidisiksi korkeuksiksi eli
6 %korkeuksiksi GRS80-vertausellipsoidin pinnalta. Muunnosta varten tarvitaan
7 %hilaruudukko, jonka pisteiden koordinaatit löytyvät tiedostosta
8 %FIN2005N00.lst.
9
10 %Syöte
11 %
12 %fii on maantieteellinen leveys.
13 %lambda on maantieteellinen pituus.
14 %H on normaalikorkeus N2000-korkeusjärjestelmässä.
15
16 %Tuloste
17 %
18 %h on ellipsoidinen korkeus.
19
20
21
22
23
24 %Lasketaan hilaruudun kulmapisteiden maantieteelliset koordinaatit asteina.
25
26 fii_max = ceil(fii*100/2)/100*2
27 %Suurin maantieteellinen leveys.
28
29 fii_min = floor(fii*100/2)/100*2
30 %Pienin maantieteellinen leveys.
31
32 lambda_max = ceil(lambda*100/4)/100*4
33 %Suurin maantieteellinen pituus.
34
35 lambda_min = floor(lambda*100/4)/100*4
36 %Pienin maantieteellinen pituus.
37
38
39
40
41
42 %Lasketaan niiden rivien numerot, joilta hilaruudun pisteiden koordinaatit
43 %löytyvät tiedostosta FIN2005N00.lst. Rivejä on yhteensä neljä, mutta niitä
44 %on aina kaksi peräkkäin, joten vain kahden rivin numerot tarvitsee laskea.
45
46 delta_fii = 70.70-fii_max
47 delta_lambda = lambda_min-17.48
48 rivi_1 = (389/0.02)*delta_fii+(1/0.04)*delta_lambda+1
49 %Sen rivin numero, josta löytyvät hilaruudun luoteiskulman eli vasemman
50 %ylänurkan koordinaatit.

```

```

51
52 delta_fii = 70.70-fii_min
53 rivi_3 = (389/0.02)*delta_fii+(1/0.04)*delta_lambda+1
54 %Sen rivin numero, josta löytyvät hilaruudun lounaiskulman eli vasemman
55 %alanurkan koordinaatit.
56
57
58
59
60
61 %Haetaan hilaruudun kulmapisteiden kvasigeoidin korkeudet tiedostosta
62 %FIN2005N00.lst.
63
64 tiedosto = fopen('Kvasigeoidimalli\FIN2005N00.lst')
65 rivit_12 = textscan(tiedosto,'%f %f %f',2,'MultipleDelimsAsOne',1,...
66     'HeaderLines',round(rivi_1-1))
67
68 zeeta_1 = rivit_12{3}(1)
69 %Hilaruudun vasemman yläkulman kvasigeoidin korkeus.
70
71 zeeta_2 = rivit_12{3}(2)
72 %Hilaruudun oikean yläkulman kvasigeoidin korkeus.
73
74 rivit_34 = textscan(tiedosto,'%f %f %f',2,'MultipleDelimsAsOne',1,...
75     'HeaderLines',round(rivi_3-rivi_1-1))
76
77 zeeta_3 = rivit_34{3}(1)
78 %Hilaruudun vasemman alakulman kvasigeoidin korkeus.
79
80 zeeta_4 = rivit_34{3}(2)
81 %Hilaruudun oikean alakulman kvasigeoidin korkeus.
82
83 fclose(tiedosto)
84
85
86
87
88
89 %Bilineaarinen interpolointi hilassa.
90
91 fii_norm = (fii-fii_min)/0.02
92 lambda_norm = (lambda-lambda_min)/0.04
93 %Lasketaan normalisoitu sijainti hilaruudun sisällä. Sijainnit fii_norm ja
94 %lambda_norm ovat suhteellisia arvoja välillä 0-1.
95
96 zeeta = fii_norm*(1-lambda_norm)*zeeta_1...
97     +fii_norm*lambda_norm*zeeta_2...
98     +(1-fii_norm)*(1-lambda_norm)*zeeta_3...
99     +(1-fii_norm)*lambda_norm*zeeta_4
100 % Lasketaan kvasigeoidin korkeus zeeta.
101
102
103
104
105
106 h = zeeta+H
107 %Lasketaan ellipsoidinen korkeus h.
108
109 end

```

Funktio (geod_suorak.m)

```
1 function [X,Y,Z] = geod_suorak(fii,lambda,h)
2
3 %Kuvaus
4 %
5 %Ohjelma konvertoi geodeettiset EUREF-FIN-koordinaatit suorakulmaisiksi
6 %EUREF-FIN-koordinaateiksi.
7
8 %Syöte
9 %
10 %fii on geodeettinen leveys.
11 %lambda on geodeettinen pituus.
12 %h on ellipsoidinen korkeus.
13
14 %Tuloste
15 %
16 %X on x-akselin suuntainen suorakulmainen koordinaatti.
17 %Y on y-akselin suuntainen suorakulmainen koordinaatti.
18 %Z on z-akselin suuntainen suorakulmainen koordinaatti.
19
20
21
22
23
24 %Yksikkömuunnos
25
26 fii = fii*(pi()/180)
27 %Muunnetaan geodeettinen leveys asteista radiaaneiksi.
28
29 lambda = lambda*(pi()/180)
30 %Muunnetaan geodeettinen pituus asteista radiaaneiksi.
31
32
33
34
35
36 %GRS80-vertausellipsoidin parametrit
37
38 a = 6378137.0
39 %a on ellipsoidin isoakselin puolikas.
40
41 f = 1/298.257222101
42 %f on ellipsoidin litistyssuhde.
43
44
45
46
47
48 %Apusuureet
49
50 e = sqrt(2*f-f^2)
```

```
51 %e on ellipsoidin ensimmäinen eksentrisyys eli epäkeskisyyys.  
52  
53 N = a/sqrt(1-e^2*(sin(fii))^2)  
54 %N on ellipsoidin poikittaiskaarevuussäde.  
55  
56  
57  
58  
59  
60 %Laskenta  
61  
62 X = (N+h)*cos(fii)*cos(lambda)  
63 Y = (N+h)*cos(fii)*sin(lambda)  
64 Z = (N*(1-e^2)+h)*sin(fii)  
65  
66 end
```

Funktio (TM35_GK27.m)

```
1 function [N,E] = TM35_GK27(N,E)
2
3 %Kuvaus
4 %
5 %Ohjelma konvertoi ETRS-TM35FIN-koordinaatit ETRS-GK27-koordinaateiksi.
6
7 %Syöte
8 %
9 %N on projektion pohjoiskoordinaatti.
10 %E on projektion itäkoordinaatti.
11
12 %Tuloste
13 %
14 %N on projektion pohjoiskoordinaatti.
15 %E on projektion itäkoordinaatti.
16
17
18
19
20
21 N = N/0.9996
22 %Lasketaan projektion pohjoiskoordinaatti.
23
24 E = (E-200)/0.9996+27000000
25 %Lasketaan projektion itäkoordinaatti.
26
27 end
```

Funktio (GK27_TM35.m)

```
1 function [N,E] = GK27_TM35(N,E)
2
3 %Kuvaus
4 %
5 %Ohjelma konvertoi ETRS-GK27-koordinaatit ETRS-TM35FIN-koordinaateiksi.
6
7 %Syöte
8 %
9 %N on projektion pohjoiskoordinaatti.
10 %E on projektion itäkoordinaatti.
11
12 %Tuloste
13 %
14 %N on projektion pohjoiskoordinaatti.
15 %E on projektion itäkoordinaatti.
16
17
18
19
20
21 E = E-27000000
22 %Poistetaan projektiokaistan keskimeridiaanin asteluku itäkoordinaatin
23 %alusta.
24
25
26
27
28
29 N = N*0.9996
30 %Lasketaan projektion pohjoiskoordinaatti.
31
32 E = (E-500000)*0.9996+500000
33 %Lasketaan projektion itäkoordinaatti.
34
35 end
```

Skripti (aloitus.m)

```
1 format longG
2 %Asetetaan Matlabin komentoikkunassa näkyvien lukujen esitystavaksi 15
3 %numeroa, ei eksponentteja.
4
5 format compact
6 %Asetetaan komentoikkunan esitystapa tiiviiksi.
7
8 cd('C:\Konversiot')
9 %Asetetaan Matlabin työkansio (Current folder).
10
11 addpath Esimerkit Funktiot
12 %Lisätään kansiot Esimerkit ja Funktiot Matlabin hakupolkuun (Search path).
13
14 edit TM35_GK27 GK27_TM35...
15     geod_suorak norm_ellip taso_maant...
16     maant_taso ellip_norm suorak_geod...
17     paaohjelma esimerkki_2 esimerkki_1
18 %Avataan luetellut funktiot Matlabin tekstinmuokkausikkunaan (Editor).
```

Skripti (esimerkki_1.m)

```
1 clc
2 %Tyhjennetään Matlabin komentoikkuna.
3
4 lue = 'Koordinaatit_lue\kaupungit_FINsuorak.csv'
5 %Annetaan luettavan tiedoston nimi.
6
7 kirjoita = 'Koordinaatit_kirjoita\kaupungit_TM35FIN.csv'
8 %Annetaan kirjoitettavan tiedoston nimi.
9
10 koordinaatistot = {'FINsuorak','TM35FIN','','N2000'}
11 %Annetaan koordinaatistot ja korkeusjärjestelmät.
12
13 paaohjelma(lue,kirjoita,koordinaatistot)
14 %Suoritetaan pääohjelma.
```

Skripti (esimerkki_2.m)

```
1 clc
2 %Tyhjennetään Matlabin komentoikkuna.
3
4 lue = 'Koordinaatit_lue\kaupungit_TM35FIN.csv'
5 %Annetaan luettavan tiedoston nimi.
6
7 kirjoita = 'Koordinaatit_kirjoita\kaupungit_FINsuorak.csv'
8 %Annetaan kirjoitettavan tiedoston nimi.
9
10 koordinaatistot = {'TM35FIN','FINsuorak','N2000',''}
11 %Annetaan koordinaatistot ja korkeusjärjestelmät.
12
13 paaohjelma(lue,kirjoita,koordinaatistot)
14 %Suoritetaan pääohjelma.
```

Koordinaatit (kaupungit_FINsuorak.csv)

1	Helsinki,2884084.7243,1341042.6054,5509990.4569
2	Tampere,2792862.7490,1229519.3791,5582133.5732
3	Oulu,2438813.8215,1161513.3651,5758511.4725
4	Turku,2918311.3091,1194903.7810,5525472.9655
5	Jyvaskyla,2682928.7996,1293770.5778,5621139.1943
6	Kuopio,2580605.6838,1353603.6668,5654633.8990
7	Joensuu,2554787.3617,1461002.9908,5639686.3410
8	Vaasa,2690331.1401,1066007.3633,5664845.0001
9	Rovaniemi,2297405.2143,1107381.7039,5826560.8382
10	Kokkola,2593025.0524,1107728.0448,5701839.1947

Koordinaatit (kaupungit_TM35FIN.csv)

1	Helsinki,6672223.8515,385564.6371,6.9580
2	Tampere,6822549.2002,327606.7107,89.8270
3	Oulu,7211188.5050,427743.7217,0.8200
4	Turku,6711043.0156,239742.7070,5.4870
5	Jyvaskyla,6901580.3989,434758.8535,96.8100
6	Kuopio,6973793.6879,534485.1301,100.5080
7	Joensuu,6944062.6335,641878.3710,80.9820
8	Vaasa,7007660.6525,228381.1231,10.7430
9	Rovaniemi,7376368.1567,443706.1761,85.2570
10	Kokkola,7084826.7361,309804.7951,4.7760