

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Hendrig Sellik 142438IAPB

**KUUPSATELLIIDI ASENDIMÄÄRAMINE
PÄIKESE- JA MAGNETVEKTORIGA
KASUTADES SVD MEETODIT**

Bakalaureusetöö

Juhendaja: Martin Rebane
MSc

Kaasjuhendaja: Alar Leibak
PhD
Dotsent

Tallinn 2018

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Hendrig Sellik

20.05.2018

Annotatsioon

Käesoleva töö esmaseks eesmärgiks oli välja pakkuda esialgne lahendus TTÜ100 satelliidi asendi määramiseks koos vajaliku raudvara kirjeldusega ning täpsemalt kirjeldada päikesevektori arvutusi ning SVD meetodil põhinevat asendi määramist.

Kuna SVD meetodi kirjeldamine käsitlemata pealiskaudselt magnetomeetrit, satelliidi positsiooni määramist orbiidil ja Maa magnetvälja mudelit oleks segadust tekitav, on käsitletud ka neid ning antud viited nende teostustele MATLAB-is või C-koodis.

Töö tulemusena arendati välja efektiivne ja valideeritud mudel päikesevektori arvutamiseks. Samuti kirjeldati kogu esialgse asendimääramise protsessi piisavalt detailselt ja terviklikult koos asjakohaste viidetega, et lihtsustada oluliselt uute arendustiimi liikmete projekti sisse-eramist.

Lisaks kontakteeruti Aalborgi ülikooli satelliidi AAUSAT3 arendajatega, kes jagasid satelliidi jaoks 2010. aastal välja töötatud asendimääramise tervikliku mudelit, mis on realiseeritud MATLAB-is ja Simulinkis.

Töös mainitud koodid ja mudelid on kättesaadavad GitLabi repositooriumis <https://gitlab.com/martinrebane/ADCS>.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 49 leheküljel, 3 peatükki, 18 joonist, 2 tabelit.

Abstract

Cube satellite attitude determination with sun and magnetic field vectors using SVD method

The aim of this thesis was to offer an initial solution for TTÜ100 cube satellite attitude determination with the necessary hardware, description of sun vector calculation and SVD method in more detail.

Because describing attitude determination with the help of SVD method without addressing the topics of magnetometers, IGRF model, orbit propagation and Kalman filtering would be confusing, these topics were also discussed in this thesis.

As a result of this thesis, an effective and validated model of Sun vector calculation in Earth Centered Inertial frame was developed. In addition, the initial attitude determination of TTÜ100 cube satellite was described in detail with simple to understand references, so new members of the TTÜ100 ADCS team will be able to adapt to the project quickly.

In addition, developers of AAUSAT3 from Aalborg were contacted and they provided the author of this thesis with Simulink environment which completely simulates attitude determination, including sensor errors.

All the mentioned code and models in this thesis are available through ADCS team GitLab repository <https://gitlab.com/martinrebane/ADCS>.

The thesis is in Estonian and contains 49 pages of text, 3 chapters, 18 figures, 2 tables.

Lühendite ja mõistete sõnastik

ARM	ARM (esialgses variandis ingl k. <i>Acorn RISC Machine</i>) on odavate ja väikese energiakuluga 32-bitiste RISC-protsessorite sari manussüsteemide, andmetöötluse, digitaalse signaalitöötluse, arvutimängude, laiatarbe-multimeedia ja portatiivsete rakenduste jaoks. [1]
ECEF	Ingl k. <i>Earth Centered Earth Fixed</i> Lühend koordinaadistiku kohta, mida on kirjeldatud peatükis 2.3.3.
ECI	Ingl k. <i>Earth Centered Inertial</i> . Lühend koordinaadistiku kohta, mida on kirjeldatud peatükis 2.3.1.
git	Versioonihaldussüsteem, mis jälgib failimuudatusi ning seeläbi aitab mitmel inimestel töötada samade failidega korraga.
kvaternioon	Neljandat järku hüperkompleksarv. Erinevalt Euleri nurkadest, milles kirjeldatakse objekti pööret 3 erineva telje abil, teeb kvaternioon seda ühe pöördega. Lähemalt saab lugeda viitest [2, p. 17].
SBF	Ingl k. <i>Satellite Body Frame</i> Lühend koordinaadistiku kohta, mida on kirjeldatud peatükis 2.3.2.
SRAM	Staatiline muutmälu, mis ei vaja pidevat andmete värskendamist, kuid pole siiski püsिमälu, st andmed hävivad toite väljalülitamisel. [1]

Sisukord

1 Sissejuhatus	10
2 Üldine taustauuring ja analüüs	12
2.1 Raudvara	12
2.1.1 Ümbrusvalguse andur VEML6030X01	12
2.1.2 <i>Pinhole</i> päikeseandur	13
2.1.3 Magnetomeeter HMC5983	13
2.1.4 Mikrokontroller STM32F303VD	14
2.2 Asendimääramise põhiolemus	15
2.3 Koordinaatsüsteemid	16
2.3.1 ECI koordinaadistik	16
2.3.2 SBF koordinaadistik	17
2.3.3 ECEF koordinaadistik	17
2.4 Juulise kuupäev	18
2.4.1 Juulise kuupäeva arvutamine	18
3 Satelliidi asendimääramise tarkvara arendus	20
3.1 Päikese asendi määramine	20
3.1.1 Päikesevektori arvutamine ECI koordinaadistikus	20
3.1.2 Parallaksi efekt	21
3.1.3 Päikesevektori arvutamine SBF koordinaadistikus	23
3.1.4 Päikese Albedo mudel	23
3.1.5 Päikesevarjutuse mudel	24
3.2 Satelliidi asendi määramine orbiidil	26
3.2.1 TLE	26
3.2.2 SGP4 mudel	27
3.3 Magnetväli	28
3.3.1 IGRF mudel	29
3.4 SVD meetodil põhinev asendimääramine	30
3.4.1 SVD meetodi sisendid	30
3.4.2 SVD meetodil põhinev asendimääramise teostus	30

3.5 Kalmani filter.....	31
3.6 Satelliidi esialgse asendimääramise komponentide sisendid ja väljundid ning visuaalne esitlus.....	32
4 Tulemuste valideerimine	35
4.1 Päikese mudeli valideerimine	35
Kokkuvõte	36
Kasutatud kirjandus	38
Lisa 1 – GitLab’i link	41
Lisa 1 – MATLAB-i koodid.....	42
4.2 Päikese mudeli MATLAB-i kood.....	42
4.3 Päikesevarjutuse mudeli MATLAB-i kood.....	44
4.4 Juuliuse kuupäeva arvutamise MATLAB-i koodi.....	45
4.5 SVD meetodi MATLAB-i kood	47
4.6 Maatriksist kvaterniooni teisendamise MATLAB-i kood	49

Jooniste loetelu

Joonis 1. VEML6030X01 ümbrusvalguse andur	12
Joonis 2. VEML6030X01 reageerimine erineva lainepikkusega valgusele	13
Joonis 3. Magnetomeeter HMC5983	13
Joonis 4. Mikrokontroller STM32F3discovery, mis sisaldab STM32F303VD mikrokontrollerit	14
Joonis 5 ECI koordinaadistiku telgede suunad.....	16
Joonis 6. Satelliidi keha koordinaadistiku telgede suunad (autori modifitseeritud joonis)	17
Joonis 7. ECEF koordinaadistiku telgede suunad	17
Joonis 8. Parallaksi efekt	22
Joonis 9. Vektorid, nurk ja kaugus, mida kasutatakse päikesevarjutuse mudelis	25
Joonis 10. Maa peamine magnetväli kujutatud kui dipoolmagnet	28
Joonis 11. Magnetvektori arvutamine ECI koordinaadistikus	29
Joonis 12. Asendimääramise süsteemi lihtsustatud mudel	34
Joonis 13. Päikese mudeli vektori vahe võrreldes Aerospace Toolbox'i arvutustega aastal 2019	35
Joonis 14. MATLAB-i kood päikese vektori arvutamiseks ECI koordinaadistikus	43
Joonis 15. Päikesevarjutuse mudeli MATLAB-i kood.....	44
Joonis 16. Juuliuse kuupäeva arvutamise MATLAB-i kood.....	46
Joonis 17. SVD meetodi MATLAB-i kood.....	48
Joonis 18. Maatriksist kvaterniooni teisendamise MATLAB-i kood.....	50

Tabelite loetelu

Tabel 1. TLE 1. rea kirjeldus.....	27
Tabel 2. TLE 2. rea kirjeldus.....	27

1 Sissejuhatus

TTÜ100 satelliit on TTÜ Mektory satelliidi programmi osa, mille eesmärgiks on saata orbiidile võimekas ning töökindel kuupsatelliit. Satelliidi arendamine on rahvusvaheline ja interdistsiplinaarne tegevus, mida aitavad läbi viia õpilased ja professorid erinevatest rahvustest peamiselt TTÜ-st, mille lõpptulemuseks on plaanitud töötava satelliidi orbiidile saatmine 2019. aastal. Selle tulemusena saavad tudengid praktilise kogemuse, mis aitab neil valmistuda töötamiseks kosmose- ja tehnoloogiavaldkondades ning suuremahulistes ja rahvusvahelistes tiimiprojektides [3].

Satelliit arendatakse vastavalt kuupsatelliidi standardile suurusega 10x10x10cm, mis kaalub vähem kui 1.33 kilogrammi, millest tulenevad ka piirangud satelliidile paigaldavate andurite suurustele ning asendimääramise süsteemi arvutusjõudlusele [4].

Selle bakalaureusetöö eesmärgiks oli välja töötada efektiivne päikesemudel, mida saab kasutada asendimääramise süsteemis, kuna eelmine lahendus ei olnud satelliidi jaoks sobiv. Samuti määrata ära töö kirjutamise hetkel päikeseanduritega seotud kitsaskohad.

Töö kirjutamise hetkel ei olnud keegi detailsemalt kirjutanud TTÜ100 satelliidi asendimääramise süsteemist tervikuna, ega arvestanud oma lõputööde kirjutamisel teiste asendimääramise tarkvara komponentidega. See tulenes sellest, et satelliidi asendimääramise teadmiste õppimiskõver on üsna järsk ning seda on erinevate tööde põhjal raske hoomata. Sellest tulenevalt otsustas töö autor uurida ka esialgselt asendimääramist, panna kirja selle komponentide olemus ja detailsem kirjeldus koos asjalike viidetega, mis peaks oluliselt lihtsustama projekti uute liikmete sisse elamist. Samuti luua viitepunkt, mida edasises arenduses kasutada.

Töö esimeses osas kirjeldatakse satelliidis kasutatavat raudvara, antakse ülevaade töös kasutatavatest koordinaadistikest ja näidatakse kuidas arvutada Juuliuse kuupäeva.

Töö teises osas kirjeldatakse detailsemalt asendimääramise tarkvarasse kuuluvaid komponente, milleks on magnetvälja ja päikesevektori määramine ECI koordinaadistikus, päikese- ja magnetvektori arvutamine SBF koordinaadistikus ning

satelliidi asendi määramine orbiidil. Lõpuks pakutakse lahendus nende andmete abil satelliidi asendi arvutamiseks viisil, mis põhineb SVD meetodil. Lisaks on kirjeldatud kõikide komponentide sisendeid ja väljundeid ning loodud detailne skeem, mis visualiseerib komponentide koostööd. Teises osas on kirjeldatud pealiskaudselt ka Kalmani filtrit, kuid see ei ole selle töö skoobis ning seetõttu praktilisi teostusi sellele ei pakuta, vaid viidatakse eelmise magistr töö kitsaskohtadele ja antakse head vihjed nende parendamiseks.

Töö kolmandas osas valideeritakse päikese mudelist saadud vektori tulemus.

2 Üldine taustauuring ja analüüs

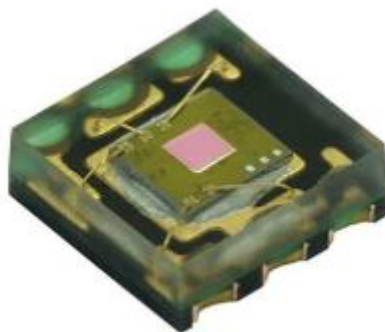
Selles peatükis käsitletakse TTU100 satelliidi raudvara ja tutvustatakse satelliidi asendimääramise tarkvara põhilist olemust ning selle juurde kuuluvaid põhimõisteid.

2.1 Raudvara

Selles alapeatükis kirjeldatakse TTU100 satelliidi asendimääramiseks kasutatavat raudvara.

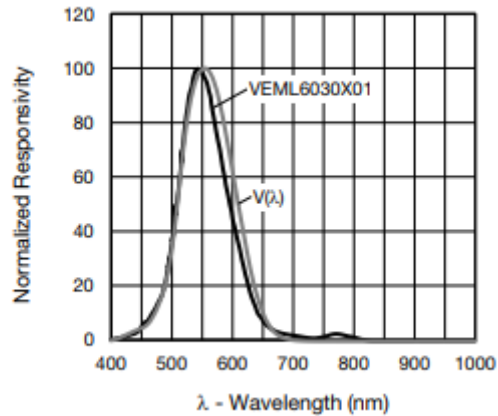
2.1.1 Ümbrusvalguse andur VEML6030X01

VEML6030X01 on Vishay Intertechnology poolt toodetud ümbrusvalguse andur [5].



Joonis 1. VEML6030X01 ümbrusvalguse andur [5]

Ümbrusvalguse andureid kasutatakse, et tajuda valgust või heledust sarnaselt inimsilmale. Neid võib tihti leida tööstuslikus valgustuses, tarbeelektronikas (telefonid jms), kus nad aitavad seada sätteid vastavalt ümbritseva valguse oludele. Selle tulemusena suudavad ümbrusvalguse andurid säästa energiat või pakkuda turvalisust, samal ajal eemaldades vajaduse manuaalselt sätteid muuta [6].



Joonis 2. VEML6030X01 reageerimine erineva lainepikkusega valgusele [5]

Jooniselt (2) on näha, et VEML6030X01 poolt tuvastatav valguse lainepikkus on väga sarnane inimese poolt nähtavale valguse lainepikkusele $V(\lambda)$.

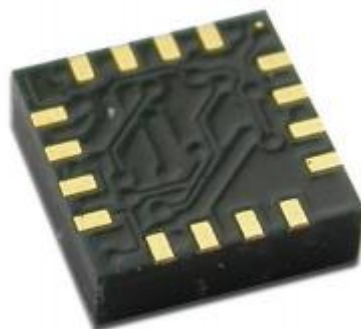
Kuupsatelliidil on 6 päikeseandurit, mis asuvad igal satelliidi küljel. Anduri väljundiks on lugem vastavalt nurgale koosinus funktsiooni järgi, kus makismaalne lugem on otse päikeseandurile langev 90° kraadine valgus.

2.1.2 Pinhole päikeseandur

Tegemist on Eiko Priideli poolt arendatava *pinhole* tüüpi eksperimentaalse päikeseanduriga, mis annab tulemuse kahes teljes. Tarkvara sellele praeguse seisuga kirjutatud ei ole ning selle mõõtetulemuste kvaliteeti hinnata ei ole võimalik.

2.1.3 Magnetomeeter HMC5983

Magnetomeetrit kasutatakse, et tuvastada magnetilisi välju ümber anduri. Satelliidil kasutatakse Honeywell International Inc. poolt toodetud HMC5983 magnetomeetrit [7].



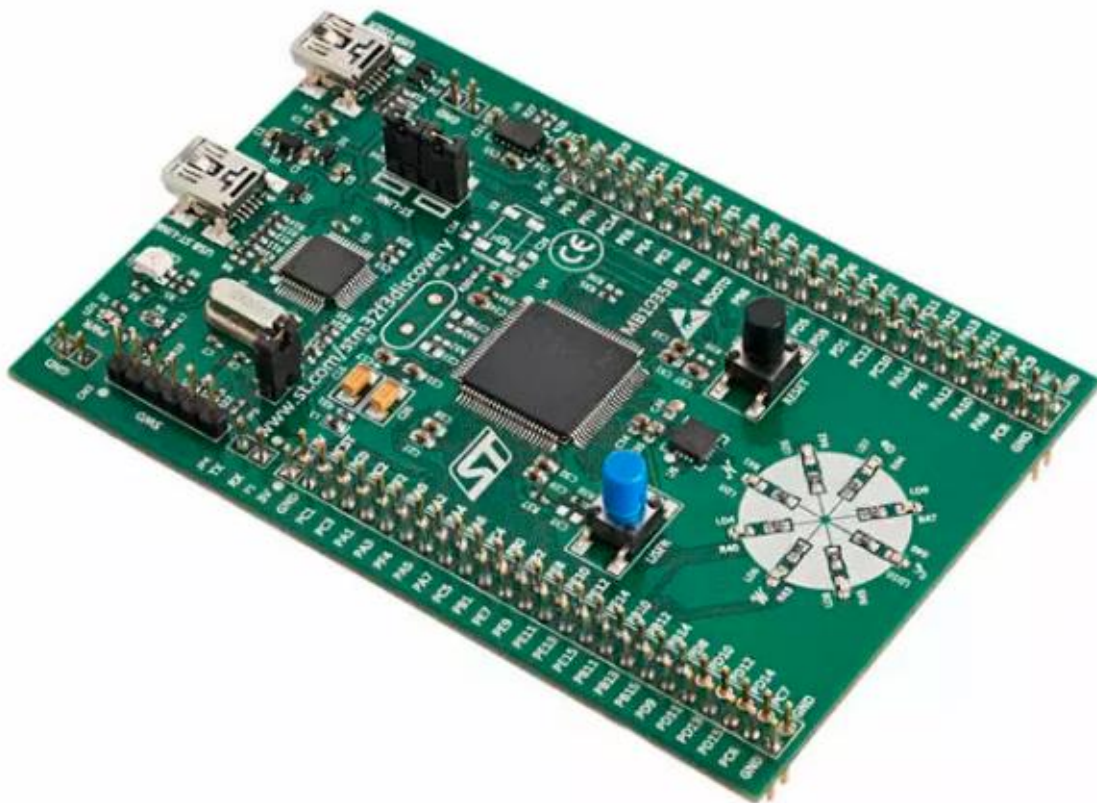
Joonis 3. Magnetomeeter HMC5983 [7]

Tegu on madala voolutarbe, suuruselt väikse ning väga odava anduriga, mis tagastab magnetvälja tugevuse 3 teljel läbi I²C liidese. Selle 220Hz väljundsagedus on satelliidi nõuete jaoks enam kui piisav [7].

On tähtis, et magnetomeetrit hoitaks eemal metallidest nagu teras, raud ja nikkel, kuna need moonutavad magnetvälju. Samuti on tähtis hoida magnetomeetrit eemal vooluallikatest ja võimalikult kaugel magnetitest ning mähistest. Anduril saab seadistada nihet, mille abil saab kompenseerida rauda sisaldavate materjalide magnetväljasid satelliidi pardal (*hard iron calibration*) [7].

2.1.4 Mikrokontroller STM32F303VD

Satelliidi asendimääramise kontrolleriiks on STM32F303VD, mille tootjaks on STMicroelectronics [8].



Joonis 4. Mikrokontroller STM32F3discovery, mis sisaldab STM32F303VD mikrokontrollerit [9]

Kuna kosmoses peab süsteem tarbima vähe voolu, on mikrokontrolleril võrreldes tavakasutuses olevate arvutitega oluliselt väiksem arvutusjõudlus. Näiteks on sellel vaid 512 kilobaiti *flash* mälu ja 64 kilobaiti SRAM-i. Mikrokontrolleris kasutatakse ARM

Cortex M4 32-bitist protsessorit koos ujukomaplokiga, mis on mõeldud matemaatiliste arvutuste kiiremaks sooritamiseks [8].

Sellest tulenevalt peab asendimääramise tarkvara arendamisel silmas pidama jõudlusele seatud piiranguid. Samuti peab arvestama manusrakenduse C-koodi kirjutamisel protsessori arhitektuuriga.

2.2 Asendimääramise põhiolamus

Satelliidi asendimääramise süsteem kasutab andureid, et saada vektori komponendid satelliidi ristkoordinaatteljestikus (SBF) ja matemaatilisi mudeleid, et saada vektori komponendid Maa inertsiaalses ristkoordinaatteljestikus (ECD). Neid komponente kasutatakse koos erinevate algoritmidega, et määrata satelliidi asend mingi koordinaadistiku suhtes näiteks kvaternioonide või Euleri nurkade kujul. Käesolevas töös kasutatakse satelliidi asendi määramiseks satelliidi koordinaatteljestiku suhet Maa inertsiaalse koordinaadistiku suhtes, mis on väljendatud kvaternioonides [10].

On vaja vähemalt kahte andurite mõõtetulemusel saadud vektorit, et määrata satelliidi asend. Näiteks saab kasutada satelliidi koordinaatteljestikus päikeseanduriga teatud ajahetkel mõõdetud päikesevektorit s_b ja magnetomeetriga mõõdetud magnetvälja vektorit m_b . Samal ajal kasutatakse päikese asendi matemaatilist mudelit, et arvutada välja vektori komponendid Maa inertsiaalses koordinaatteljestikus s_i . Samamoodi kasutatakse Maa magnetvälja mudelit, et arvutada välja maa magnetvälja vektori komponendid m_i Maa inertsiaalses koordinaatteljestikus satelliidi asukohal. Seejärel kasutatakse asendimääramise algoritmi, et määrata pöörlemismatriks R^{bi} valemitega (1) ja (2) [10].

$$s_b = R^{bi} s_i \quad (1)$$

$$m_b = R^{bi} m_i \quad (2)$$

Tuleb tähele panna, et nii päikeseanduri kui ka magnetomeetri mõõtetulemustest saadud ühikvektorid suudavad mõlemad määrata kaks asendi määramiseks vajalikku vektori komponenti kolmest. Seega tekib olukord, kus vaja on kolme komponenti, aga teada on neli. Sellest saab järeldada, et probleem on üle määratud ning tegelikkuses

asendimääramise süsteemis kasutatavad algoritmid saavad asendit vaid ligikaudselt hinnata [10, p. 2].

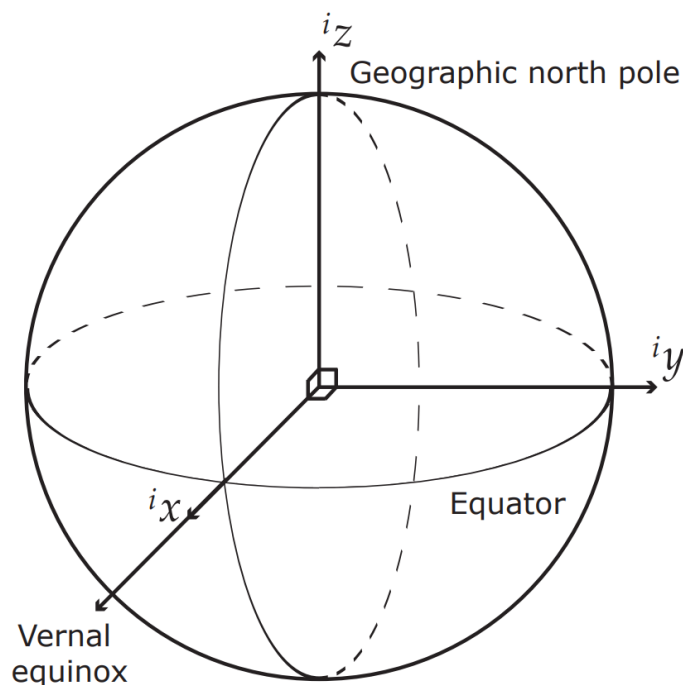
2.3 Koordinaatsüsteemid

Kuna selles töös vaadeldavad objektid, näiteks satelliit ja Päike, on maapinnast tunduvalt kõrgemal või lausa miljoneid kilomeetreid eemal, on osasid arvutusi lihtsam teha koordinaadistikes, mis ei ole seotud Maa pöörlemisega.

Järgnevalt on detailsemalt lahti kirjutatud 3 selles töös kasutatud koordinaadistikku.

2.3.1 ECI koordinaadistik

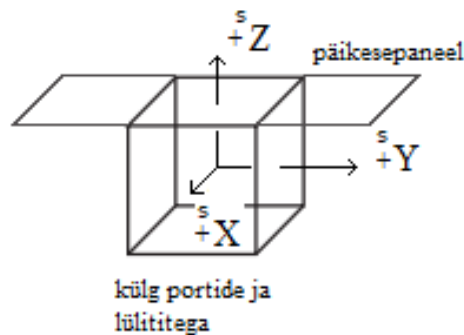
ECI (ingl k. *Earth Centered Inertial Frame*) ehk Maa keskse inertsiaalse koordinaatsüsteemi keskmeks on Maa geomeetiline keskpunkt. Z-teljeks on selles süsteemis geograafiline põhjapoolus. X-teljeks on päikese suund kevadisel pööripäeval ja Y-telg asetseb nii, et moodustuks parempoolne ristkoordinaatsüsteem. Tuleb tähele panna, et selles töös on X-telje suund on fikseeritud epohhil J2000 (01.01.2000 kell 12:00) ja ei liigu maa pöörlemisega kaasa [11].



Joonis 5 ECI koordinaadistiku telgede suunad [12]

2.3.2 SBF koordinaadistik

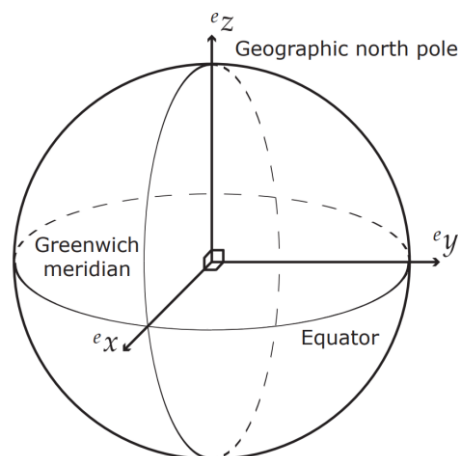
Et kirjeldada satelliidi asendit on vaja satelliidi põhise koordinaatsüsteemi. Seejärel saab satelliidi asendit kirjeldada pöördega satelliidi põhise koordinaatsüsteemi ja taustsüsteemi vahel (selles töös ECI koordinaadistik). SBF (*Satellite Body Frame*) ehk satelliidi kere koordinaatsüsteemi kese on satelliidi masskeskmes. TTÜ100 puhul on X-telg portide ja lülitite suunas, Z-telg päikesepaneelide suunas ja Y-telg selliselt, et moodustuks parempoolne ristkoordinaatsüsteem [12].



Joonis 6. Satelliidi keha koordinaadistiku telgede suunad (autori modifitseeritud joonis) [12]

2.3.3 ECEF koordinaadistik

ECEF koordinaadistik (ingl k. *Earth Centered Inertial Frame*) on fikseeritud Maa suhtes, ehk pöörleb Maaga kaasa. Sellises koordinaadistikus on mugav määratleda näiteks Maa magnetvälja ning kontrollkeskuse asukohta. ECEF-i keskpunktiks on Maa massikeske. Selle X-telg läheb läbi punkti, kus Greenwichi meridiaan läbib ekvaatorit (0 laiuskraad ja 0 kõrguskraad). Z-telg läbib geograafilise põhjapooluse ning Y-telg on kahe eelneva telje ristkorutus, moodustades nendega paremakäelise ristkoordinaadistiku, mida on kujutatud joonisel (7) [12].



Joonis 7. ECEF koordinaadistiku telgede suunad [12]

2.4 Juulise kuupäev

Väga levinud ajaformaad astronoomilistes arvutustes ja satelliidi asendi määramisel on Juulise kuupäev ehk JDN või JD (ingl k. *Julian day number*). Juulise kuupäeva täisarvuline osa koosneb päevade arvust pärast kindlat kuupäeva, milleks on 1. jaanuar kell 12:00 4713 e.m.a.. Ujuvkomakohad tähistavad fraktsiooni, mis on möödunud täispika päeva lõpust sekundi täpsusega. Tuleb tähele panna, et Juulise kuupäev algab ja lõpeb keskpäeval kell 12:00. Näiteks on Gregoriuse kalendri kuupäev 13. märts 2018 kell 15:21 Juulise kuupäevas 2458191.139583 [13].

Juulise kuupäeva loomise ajalooliseks põhjuseks oli aja arvamise lihtsustamine erinevate kalendrite vahel, kuid see leidis hilisemalt teistsugust kasutust, peamiselt astronoomias. Juulise kuupäeva ajaloost võib lähemalt lugeda viitest [14, p. 591].

Satelliidi asendimääramise tarkvara arvutustes kasutatakse tihti Juulise kuupäeva alates 1. jaanuarist kell 12:00 aastal 2000 ehk JD2000. See tähendab, et tavalisest Juulise kuupäevast on lahutatud 2451545.0 päeva ehk J2000 [15].

2.4.1 Juulise kuupäeva arvutamine

Kõik järgnevad valemid on viitest [16]. Gregoriuse kalendrist arvutatakse Juulise päeva valemiga (3).

$$JDN = day + \left\lfloor \frac{153 * m + 2}{5} \right\rfloor + 365 * y + \left\lfloor \frac{y}{4} \right\rfloor - \left\lfloor \frac{y}{100} \right\rfloor + \left\lfloor \frac{y}{400} \right\rfloor - 32045 \quad (3)$$

Kus a , y ja m on arvutatud valemitega (4), (5) ja (6).

$$a = \left\lfloor \frac{14 - month}{12} \right\rfloor \quad (4)$$

$$y = year + 4800 - a \quad (5)$$

$$m = month + 12 * a - 3 \quad (6)$$

Tuleb tähele panna, et $y(x) = \lfloor x \rfloor$ tähendab, et rakendama peab MATLAB-i *floor* funktsiooni, ehk arv ümardatakse järgmiseks väiksemaks täisarvuks. Kuna arvud ei saa olla väiksemad kui null, pole vahet, kas kasutatakse *fix* funktsiooni nagu Kristi Pilleti töös

[16], mis ümardab arvu lähima täisarvuni nulli poole või *floor* funktsiooni, mis ümardab arvu lähima väikseima täisarvuni.

Ujuvkomakoht ja lõplik Juuliusse kuupäev arvutatakse valemiga (7).

$$JD = JDN + \frac{hour - 12}{24} + \frac{minute}{1440} + \frac{second}{86400} \quad (7)$$

3 Satelliidi asendimääramise tarkvara arendus

Selles peatükis käsitletakse asendimääramise põhikomponente detailsemalt ning pakutakse teostus otse töös või antakse viide mõnele välisele teostusele.

3.1 Päikese asendi määramine

Ühtedeks satelliidi olulisemateks asendimääramise anduriteks on päikeseandurid. Päikeseandurite abil saadud vektorit SBF raamistikus ning päikese mudeli abil arvutatavat päikesevektorit ECI koordinaadistikus selles peatükis arutataksegi.

3.1.1 Päikesevektori arvutamine ECI koordinaadistikus

Päikese asendi määramiseks ECI koordinaatsüsteemis kasutatakse päikese mudelit, mille abil saab satelliidi kontrollsüsteemis arvutada päikese vektori maa keskpunktist. Kuna paljud parameetrid, mida kasutatakse päikese asendi arvutamiseks ajas väga palju ei muutu, saab mudelit kõvasti lihtsustada ja eemaldada vastavad arvutused.

Antud on päikese keskmine anomaalia (ingl k. *mean anomaly*) $M_{SunEpoch}$ 1. jaanuaril aastal 2000 kell 12:00 (UTC). Kuna Päikese keskmine liikuvus (ingl k. *mean motion*) n_{sun} on samuti teada, saame me kirjeldada hetke Päikese keskmist anomaaliat M_{Sun} valemi (8) abil [15, pp. 62-63] [17].

$$M_{Sun} = M_{SunEpoch} + n_{Sun} * JD_{2000} = 357.528 + 0.9856003 * JD_{2000} \quad (8)$$

Kus $JD_{2000} = JD - J_{2000}$. Kuna parameetrid, mis iseloomustavad Päikese liikumist orbiidil, on teada, on Päikese positsiooni arvutamiseks orbiidi tasandil (ingl k. *ecliptic longitude*) vaja ainult Juuliuse kuupäeva ja hetke keskmist anomaaliat. Seega saab valemi (9) abil arvutada päikese asukohta ECI koordinaatsüsteemis orbiidi tasandil [15, pp. 62-63] [10].

$$V_{Sun} = 280.461 + 0.9856474 * JD_{2000} + 1.915 * \sin(M_{Sun}) + 0.02 * \sin(2 * M_{Sun}) \quad (9)$$

Et muuta asend orbiidi tasandilt ECI koordinaatteljestikule, on vaja kallutada maa pöördetelge, mis on ECI koordinaatsüsteemis Z telg. See arvutatakse valemiga (10) [15, pp. 62-63].

$$\varepsilon = 23.4393 - 0.0000004 * JD_{2000} \quad (10)$$

Kus on antud maa kaldenurk ajahetkel J_{2000} . Juurde liidetakse 0.0000004 kraadi iga sellest möödunud päeva kohta.

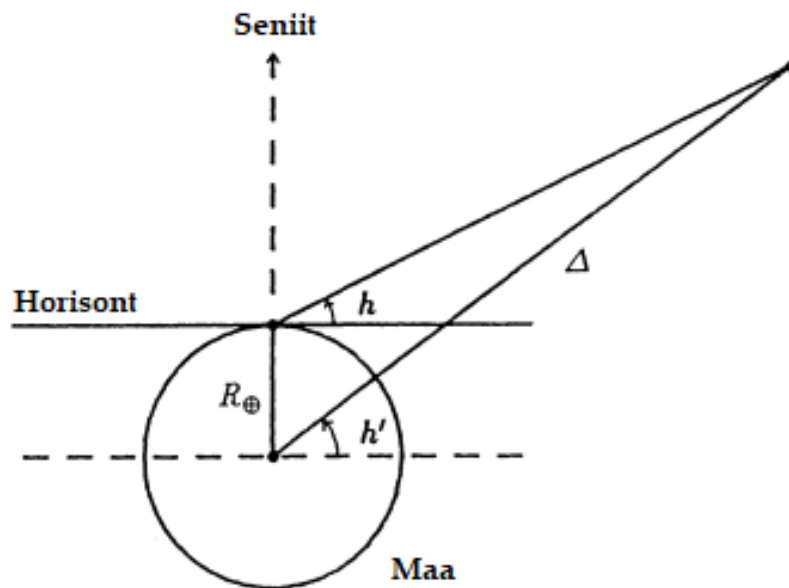
Päikese asend määratakse ühikvektoriga ECI koordinaatsüsteemis. See toimub valemiga (11), kus Päikese asend orbiidi tasandil V_{Sun} teisendatakse ristkoordinaatideks ECI koordinaadistikus kasutades Maa kaldenurka ε [10].

$$\begin{aligned} X_{Sun} &= \cos(V_{Sun}) \\ Y_{Sun} &= \cos(\varepsilon) \sin(V_{Sun}) \\ Z_{Sun} &= \sin(\varepsilon) \sin(V_{Sun}) \end{aligned} \quad (11)$$

Realiseeritud MATLAB-i koodi võib leida lisast (1). Tulenevalt sellest, et mudelis ei kasutata keerukaid maatrikseid, soovib autor C-koodi loomiseks kasutada MATLAB-i generaatorit, mis suudab MATLAB-i skriptidest genereerida C-koodi.

3.1.2 Parallaksi efekt

Kui jälgida fikseeritud asendiga tähti erinevatest kohtadest, näiteks maa keskpunktist ja maa pealsest punktist, ei ole praktilist vahet tähe asendis ja see tundub mõlemast punktist vaadatuna sama. Selle põhjuseks on tähe väga suur vahemaa tähe ja maa vahel võrreldes maa keskpunkti ja vaatleja vahelise distantsiga. Kuid vaadeldes lähemaid kosmilisi objekte, nagu planeedid, Päike ja kuu, siis võib tähele panna erinevusi. Järgnevad matemaatilised arvutused ja joonised on pärit viitest [18, p. 45].



Joonis 8. Parallaksi efekt [18, p. 45]

Seda erinevust tuntakse kui parallaksi efekti ja selle tulemusena näib vaadeldav objekt asuvat maapinnalt vaadatuna veidi madalamal kui ta oleks Maa keskpunktist vaadatuna. Maapealne kõrgus h on väiksem kui maakeskne kõrgus h' . Kui vaadeldav objekt ja vaatleja asuvad seniidis, siis parallaksi efekt kaob. Kui aga vaatleja kõrguskraad väheneb, suureneb ka parallaksi efekt ning see saavutab maksimumi kõrguskraadil $h = 0^\circ$. See on tuntud kui horisontaalne parallaksi efekt ja see arvutatakse maapinnal vaadeldes valemiga (12).

$$\pi = \sin^{-1} \frac{R_{\oplus}}{\Delta} \quad (12)$$

kus $R_{\oplus} \sim 6378$ km, ehk vahemaa vaatleja ja Maa keskpunkti vahel ja Δ vaadeldava objekti kaugus Maa keskpunktist. On näha, et mida kaugemal asub vaatleja ja lähemal vaadeldav objekt, seda suurem erinevus tekib, seega vaatleme halvimat olukorda. Kui me lisame maapinnal asuva vaatleja kaugusele satelliidi kauguse maapinnalt, mis on ~ 700 km ja võtame päikese kauguseks 147 000 000 km (lähispunkt ehk periheel), siis saame $\pi = 0^\circ 0' 9.932''$ ehk $0,0028^\circ$, mis on piisavalt väike, et jätta see arvestamata.

3.1.3 Päikesevektori arvutamine SBF koordinaadistikus

Järgnevat arvutustes eeldatakse, et päike paistab satelliidile vaid kolmelt küljelt korraga. Praktiliselt on olukord keerulisem, kuna Maalt peegelduva valguse tõttu võivad valgust saada kõik satelliidi 6 külge.

Kuna igal satelliidi küljel on päikeseandur, tähendab see, et igal satelliidi küljel asub üks andur ja igal teljel seega 2 andurit. Sõltuvalt sellest, kas andur asub telje positiivsel või negatiivsel küljel, on selle väärtus positiivne või negatiivne ning see määratakse valemitega (13), (14) ja (15).

$$\underline{S}_X^B = \pm \cos \alpha \quad (13)$$

$$\underline{S}_Y^B = \pm \cos \beta \quad (14)$$

$$\underline{S}_Z^B = \pm \cos \gamma \quad (15)$$

Teades kolme vektori komponenti saab arvutada päikese ühikvektori satelliidi keha koordinaadistikus valemitega (16) ja (17) [19, p. 15].

$$\underline{S}^B = \underline{S}_X^B + \underline{S}_Y^B + \underline{S}_Z^B \quad (16)$$

$$\underline{\hat{S}}^B = \frac{\underline{S}^B}{|\underline{S}^B|} \quad (17)$$

3.1.4 Päikese Albedo mudel

Satelliidini jõuab valgus ka muudest allikates kui otse Päikeselt. Üheks suurimaks valgusallikaks peale otsese päikesevalguse on Albedo valgus. Ehk päikesevalgus, mis peegeldub maalt satelliidini. Selle tugevus oleneb erinevatest teguritest, näiteks vee- ja lumekattest maapinnal ning pilvkattest, ulatudes kuni 30%ni otsesest päikesevalgusest [12]. Seega tuleb Maalt tuleva valgusega kindlasti arvestada, kui kasutatakse VEML6030X01 päikeseandurit, sest see ei ole võimeline tegema vahet valguse intensiivsusel.

Päikese Albedo valgust arvutatakse tavaliselt mudeli abil, kus Maa on jagatud teatud suurusega võrgustikuks. Mida väiksem on võrgustik, seda täpsemini on peegelduv valgus arvutatav. Päikese, satelliidi ning Maa asendit teades arvutatakse välja kiirguse hulk igalt võrgustiku ruudult, mis on satelliidi vaateväljal ning kätte saadaksegi Maalt peegelduv valgus. Albedo mudeli kohta võib lähemalt uurida viitest [20].

Selline arvutusviis on aga äärmiselt ressursimahukas ning nõuab palju arvutusjõudlust, mida TTÜ100 kuupsatelliidil ei ole. Samuti on efektiivse, kuid vähe ressursse nõudva mudeli loomine keerukas, ning seega käesoleva töö raames seda ei tehtud.

On mainimist väärt, et päikese Albedo valguse mudeli loomine või efektiivse mudeli leidmine ning teostamine on üsna keerukas tegevus, mis täidaks potentsiaalselt ühe bakalaureusetöö mahu.

Mõttekas on Albedo valguse arvutamist sootuks vältida ning proovida teisi lahendusi, mille abil saaks Albedo valguse mõju hoida sellisel tasemel, et see satelliidi asendi määramist ei mõjuta.

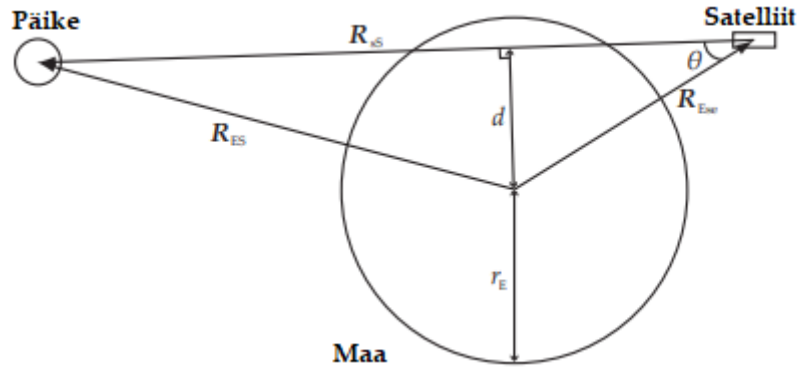
Esimeseks variandiks oleks VEML6030X01 anduri asemel kasutada *pinhole* tüüpi andureid, mida arendab Eiko Priidel, ning mis on tõenäoliselt TTÜ100 satelliidi peal.

Kuigi seda pole ESTCube'il proovitud, soovitas ESTCube'i Hendrik Ehrpais ADCS-i arendusmeeskonnast teise variandina paigaldada andurite ette *bandpass* filtrid, mille lainepikkustes on Albedo valguse mõju väiksem.

3.1.5 Päikesevarjutuse mudel

Kui satelliit asub Päikese suhtes kaugemal kui maa, on tõenäoline, et see on Maa poolt varjutatud. Päikesevarjutuse olekut on vaja teada, et rakendada seda Kalmani filtris. Selle alalõigu joonised ja arvutused on viitest [12, pp. 66-67].

Arvutuste jaoks eeldame, et Päike on punkt ja Maal on ühtlane raadius. Sel juhul saame määrata päikese olekut kui vaatlеме kaugust d ja nurka θ . Joonis (9) näitab olukorda, kus satelliit on päikese eest varjus.



Joonis 9. Vektorid, nurk ja kaugus, mida kasutatakse päikesevarjutuse mudelis

Vahemaa d saab määrata valmiga (18).

$$d = \sin \theta \cdot \|R_{Ese}\| \quad (18)$$

Kus θ on nurk vektorite R_{Ese} (Maa-satelliidi vektor) ja R_{SS} (satelliidi-Päikese vektor) vaheline.

Vektor R_{Ese} on määratud SGP4 mudeli abil ja vektor R_{SS} on määratud valemiga (19) .

$$R_{SS} = -R_{Ese} + R_{ES} \quad (19)$$

Kus R_{ES} on vektor Maa keskmest Päikese keskpunkti. See on arvatud Päikese mudeli abil.

Nurk θ on normaliseeritud vektorite R_{SS} ja R_{ES} skalaarkorrutise arkuskoosinus, mis väljendub valmiga (20).

$$\theta = \cos^{-1} \left(\frac{R_{SS}}{\|R_{SS}\|} \cdot \frac{-R_{ES}}{\|R_{ES}\|} \right) \quad (20)$$

Satelliit on päikesevarjutuses kui täidetud on kaks kriteeriumit: vahemaa d peab olema väiksem kui maa raadius ning nurk θ väiksem kui 90° .

3.2 Satelliidi asendi määramine orbiidil

Satelliidi asendi määramiseks on vaja teada satelliidi asendit orbiidil. Seda satelliit ise määrama ei ole võimalik ning see saadetakse satelliidile mõnel ajahetkel TLE (ingl k. *Two Line Element*) andmeformaadis. SGP4 (ingl k. *Simplified General Perturbations*) mudeliga, mis kirjeldab satelliidi liikumist, saab seejärel arvutada satelliidi asendit orbiidil ka hetkedel, mis leiavad aset peale satelliidile esialgsete andmete saatmist. Tuleb tähele panna, et TLE määrab satelliidi kui punkti asukoha, mitte ei kirjelda satelliidi rotatsiooni Maa suhtes [21].

3.2.1 TLE

Üks peamisi TLE formaadi loojaid on NORAD (North American Aerospace Defense Command), kelle üheks funktsiooniks on jälgida kõiki Maa ümber tiirlevaid objekte. Informatsioon objektide kohta väljastatakse TLE formaadis. Selle alalõigu koostamiseks on kasutatud viidet [13, pp. 8-9].

TLE on andmeformaad, millega edastatakse informatsiooni Maa ümber tiirlevate objektide kohta teatud ajahetkel ehk epohhil. Kasutades sobivat orbiidi hälbe mudelit on TLE abil võimalik arvutada satelliidi positsiooni teatava täpsusega minevikus ja tulevikus. Väga levinud on selleks SGP4, millest tuleb juttu järgmises peatükis.

TLE tekstifail koosneb kahest 69 tähelisest reast. Olemas võib olla ka esimene rida, kus on kirjas satelliidi nimi. TLE formaat on järgnev:

ISS (ZARYA)

```
1 25544U 98067A 08264.51782528 -.00002182 00000-0 -11606-4 0 2927
2 25544 51.6416 247.4627 0006703 130.5360 325.0288 15.72125391563537
```

Rida 1		
Veerg	Kirjeldus	Näide
01	Rea number	1
03-07	Satelliidi number	25544
08	Klassifikatsioon (U= <i>unclassified</i> ehk avalik)	U
10-11	Orbiidile saatmise aasta kaks viimast arvkohta	98
12-14	Eelnimetatud aasta stardinumbr	067
15-17	Stardile viitav tähis	A

19-20	Mõõtmise aasta viimased 2 arvkohta	08
21-32	Päevade arv mõõtmise aasta algusest koos murdosaga)	264.51782528
34-43	Pool keskmise liikumise ajalisest tuletisest	-.00002182
45-52	Kuuendik keskmise liikumise ajalisest tuletisest	00000-0
54-61	BSTAR hõõrdumise tegur	-11606-4
63	Number 0, mis pidi algselt näitama info tüüpi	0
65-68	Kokku koostatud TLE-de hulk	292
69	Kontrollsumma	7

Tabel 1. TLE 1. rea kirjeldus

Rida 2		
Veerg	Kirjeldus	Näide
01	Rea number	2
03-07	Satelliidi number	25544
09-16	Inklinatsioon	51.6416
18-25	Sõlmpunkti otsetõus	247.4627
27-33	Ektsentrilisus	0006703
35-42	Periheeli nurk [kraadides]	130.5360
44-51	Keskmine anomaalia	325.0288
53-63	Keskmine liikuvus [orbiiti päevas]	15.72125391
64-68	Mõõtmise hetkel ümber maa tehtud tiirude arv	56353
69	Kontrollsumma	7

Tabel 2. TLE 2. rea kirjeldus

Tuleb tähele panna, et ridades on tühikutel tähtis koht ning veerud, mis ei ole tabelis märgitud on täidetud tühikutega.

3.2.2 SGP4 mudel

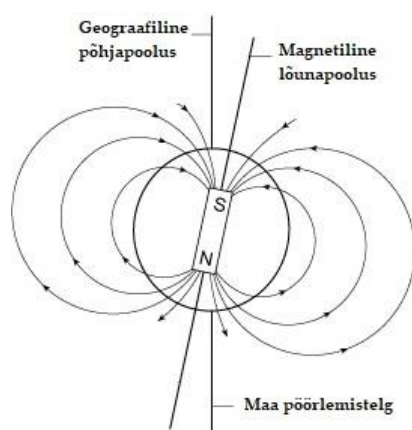
Kõige enam kasutatud matemaatiline mudel satelliidi asendi määramiseks, mille orbitaalperiood ei ületa 225 minutit, on SGP4 mudel. See on täpsem kui Kepleri orbiidi mudel, sest arvestab erinevaid jõude nagu atmosfääri hõõrdejõud, Päikese radiatsioon ning ka teiste taevakehade gravitatsioon, mis avaldab mõju tehiskaaslase orbiidile [13].

SGP4 abil saab arvutada satelliidi asendit TLE-s määratud ajahetkest edasi, kuid samas tuleb arvestada, et TLE määramise ajahetkel on viga umbes 1km, mis kasvab ~1-3km päevas. Seega on vaja TLE andmeid pidevalt uuendada [17].

SGP4 lähtekood on avaldatud FORTRAN-is aastal 1988 [21]. Peale seda on avaldatud lähtekoode ka näiteks Pythonis, PHP5-s, Javas, C++-is, Pascalis, C#-s ja MATLAB-i koodis, mille võib leida viitest [22]. C-koodi on võimalik kätte saada viitest [23]. SGP4 sisendiks on andmed TLE-st ning aeg, mis on kulunud peale TLE väljastamist. Tuleb tähele panna, et enamasti on see aeg erinevatel SGP4 mudelite teostustel minutites [17]. SGP4 väljundiks on satelliidi vektor ECI koordinaadistikus, mille komponendid on esitatud kilomeetrites [24].

3.3 Magnetväli

Maa magnetväli ehk geomagneetiline väli on magnetväli, mis algab Maa sisemusest ning ulatub kosmosesse. See teadmine võimaldab satelliidil Maa magnetvälja abil oma asendit määrata ning kontrollsüsteemi abil ka muuta. Geomagneetilist välja võib kaudselt kirjeldada kui dipoolmagnetit (Joonis 10), mis on maa pöörlemistelje suhtes 11° viltu. Magnetiline põhjapoolus, mis asub Gröönimaa lähedal on füüsiliselt hoopis kujuteldava dipoolmagneti lõunapoolus, sest tõmbab enda poole magnetite põhjapoolust [25].



Joonis 10. Maa peamine magnetväli kujutatud kui dipoolmagnet [26]

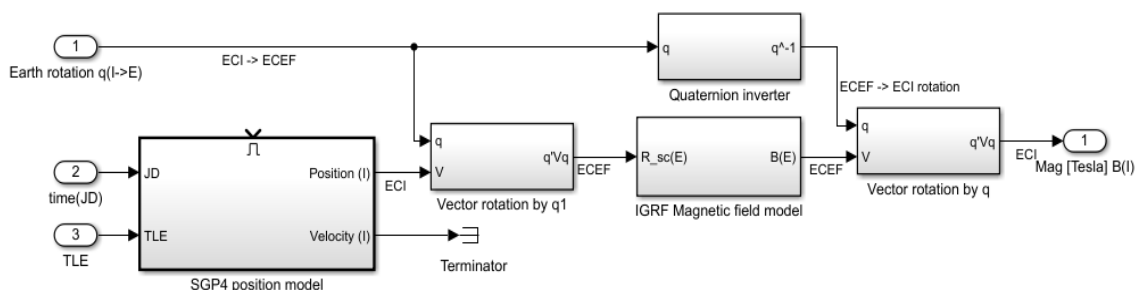
Maa peamine magnetväli moodustab rohkem kui 90% Maa kogu magnetväljast ja tekib Maa ülavahevööst, mis liigub aeglaselt. Maa magnetosfäär ja maakoore aga tekitavad selles magnetväljas varieeruvusi, mis toimuvad kiiremini kui peaväljas [25].

Maa peamist magnetvälja saab kirjeldada IGRF (ingl k. *International Association of Geomagnetism*) ning WMM (ingl k. *World Magnetic Model*) matemaatiliste mudeliga, mis on täpsuselt umbes samad [12]. Antud töös kasutatakse IGRF mudelit.

3.3.1 IGRF mudel

IGRF (ingl k. *International Association of Geomagnetism*) mudel on aastast 1965 IAGA (ingl k. *International Association of Geomagnetism and Aeronomy*) poolt loodud matemaatiline mudel, mis kirjeldab Maa magnetvälja. See on loodud kasutades maa magnetvälja andmeid observatooriumidest Maal ja satelliitidest Maa orbiidil [27].

IGRF saab uued koefitsiendid iga viie aasta tagant. Hetke 12nda generatsiooni IGRF kasutab koefitsiente aastast 2015, mis asendatakse uutega aastal 2020. Mudeli sisendiks on SGP4 mudeliga arvutatud satelliidi asend ECEF koordinaadistikus ning aeg. Mudeli väljundiks on maa magnetvälja vektor antud ECEF koordinaadistiku punktis etteantud ajal. Kuna satelliidi asend on SGP4 väljundis ECI koordinaadistikus, tuleb see kvaternionide abil pöörata ECEF koordinaadistiku. Peale IGRF-i lugemi saamist saab selle pöördkvaternioniga pöörata tagasi ECI koordinaadistikku. Protsess on visualiseeritud Simulink'i mudeli abil joonisel (11).



Joonis 11. Magnetvektori arvutamine ECI koordinaadistikus (autori joonis)

Käesolevas bakalaureusetöös IGRF-i mudeli tööpõhimõtteid detailselt lahti ei seletata, sest mudel on piisavalt keeruline, et sellele keskenduda täiesti iseseisvas bakalaureusetöös. Soovi korral saab IGRF-i kohta lähemalt lugeda viitest [27]. Küll aga saab kasutada seda mudelit satelliidi tarkvara arendamisel, sest eksisteerib MATLAB-i kood, mille ümberkirjutamisega C-koodiks tegeleb TTÜ100 ADCS tiimis Shu Taya.

Töö kirjutamise hetkel on IGRF-i mudeli C koodi ümberkirjutamine valideerimise faasis.

3.4 SVD meetodil põhinev asendimääramine

SVD (ingl k. *Single Value Decomposition*) meetodil põhinev asendimääramine annab kvaterniooni, mis aitab kontrollida Kalmani filtri õigsust kui satelliit ei ole päikesevarjus. SVD meetodi abil on võimalik teha ka väga hea esialgne asendi hinnang, mis on Kalmani filtri sisendiks.

3.4.1 SVD meetodi sisendid

SVD meetodi 1 põhineva asendimääramise sisenditeks on viite [12] järgi

- Anduritega mõõdetud Päikesevektor satelliidi keha koordinaadistikus (SBF) ${}^sV_{sun}$
- Anduriga mõõdetud magnetvektor satelliidi keha koordinaadistikus (SBF) ${}^sV_{mag}$
- Päikese mudeliga arvatud päikese asend ECI koordinaadistikus ${}^iV_{sun}$
- IGRF mudeliga arvatud magnetvälja vektor ECI koordinaadistikus satelliidi asendil ${}^iV_{mag}$
- Konstant σ_{sun}^{-2}
- Konstant σ_{mag}^{-2}

3.4.2 SVD meetodil põhinev asendimääramise teostus

Esimesena leitakse maatriks \underline{B} valemiga (21), mis põhineb eelmises alapeatükis käsitletud sisenditel [12].

$$\underline{B} = \sigma_{sun}^{-2} {}^sV_{sun} {}^iV_{sun} + \sigma_{mag}^{-2} {}^sV_{mag} {}^iV_{mag} \quad (21)$$

Järgmiseks rakendame SVD meetodit valemis (22) [12].

$$SVD(\underline{B}) = [\underline{U}, \underline{S}, \underline{V}] \quad (22)$$

Maatriksid \underline{U} ja \underline{V} on ortogonaalsed ning \underline{S} on diagonaalmaatriks, mille diagonaalelemendid on järjestatud kahanevalt. SVD meetodi teostusi on võimalik leida internetist ning selles töös kasutatakse meetodit AAUSAT3-st. Järgmisena on vaja leida optimaalne pöördmaatriks valemiga (23) [12].

$${}^{\mathcal{I}}\underline{A} = \underline{U}diag[1 \ 1 \ \det(\underline{U}) \ \det(\underline{V})]\underline{V}^T \quad (23)$$

Optimaalne pöördmaatriks ${}^{\mathcal{I}}\underline{A}$ tuleb seejärel teisendada kvaterniooniks ${}^{\mathcal{I}}q$. Kirjeldatud teostus on realiseeritud MATLAB-is [12].

3.5 Kalmani filter

See peatükk pakub Kalmani filtrile pealiskaudse seletuse, kuna see ei mahu selle töö skoopi, kuid on asendimääramise süsteemi väga tähtis osa.

Kalmani filter on korduv algoritm, mis on võimeline optimaalselt ennustama reaallajas süsteemi seisundit, kusjuures ei nõuta eelnevate andmete salvestamist ega ümber arvutamist kuni uus mõõtmine on tehtud. Kalmani filter kasutab asendi määramiseks teadmisi algse asendi, müra ja mõõtmiste vigade ning süsteemi ja mõõtmiste dünaamika kohta [28].

Kui selles töös kasutatav SVD meetodil põhinev teostus arvutab satelliidi esialgse asendi, siis Kalmani filtri abil määratakse seda pidevalt ja kaudselt läbi mõõtmiste ja satelliidile mõjuvate jõudude abil. See võimaldab näiteks määrata satelliidi asendit päikesevarjutuses baseerudes satelliidi esialgsele asendile ja hilisematele güroskoobi mõõtetulemustele [29].

Kalmani filtri matemaatiliste aluste, täpsema kirjelduste ning ainult güroskoobi andmetel põhinevate simulatsioonide kohta saab lugeda Caroliina Laantee tööst [29]. Oluline on aga tähele panna, et antud töö kasutab vaid güroskoobi andmeid ja töötab Euleri nurkadega. Tulevikus vaja kasutada Kalmani filtris aga lisaks güroskoobi andmetele ka päikese- ja magnetvälja vektorit nii mõõdetult SBF koordinaadistikus kui arvutatult ECI

koordinaadistikus satelliidi asendilt, samuti päikesevarju olekut ja satelliidi kontrollsüsteemi mähiste poolt rakendatud jõude satelliidi pööramiseks. Lisaks tuleb Euleri nurgad asendada kvaternioonidega, et elimineerida Euleri nurkadega seotud nähtust, *gimbal lock*'i [12, p. 92].

On näha, et Kalmani filtri realiseerimine pole triviaalne ülesanne, kuid selle teostamiseks soovib autor võtta näiteks AAUSAT3-e MATLAB-i koodid ja Simulink'i skeemid, mille teooria leiab viitest [12, p. 92] ja koodi TTÜ100 ADCS tiimi git'i repositooriumist.

3.6 Satelliidi esialgse asendimääramise komponentide sisendid ja väljundid ning visuaalne esitus

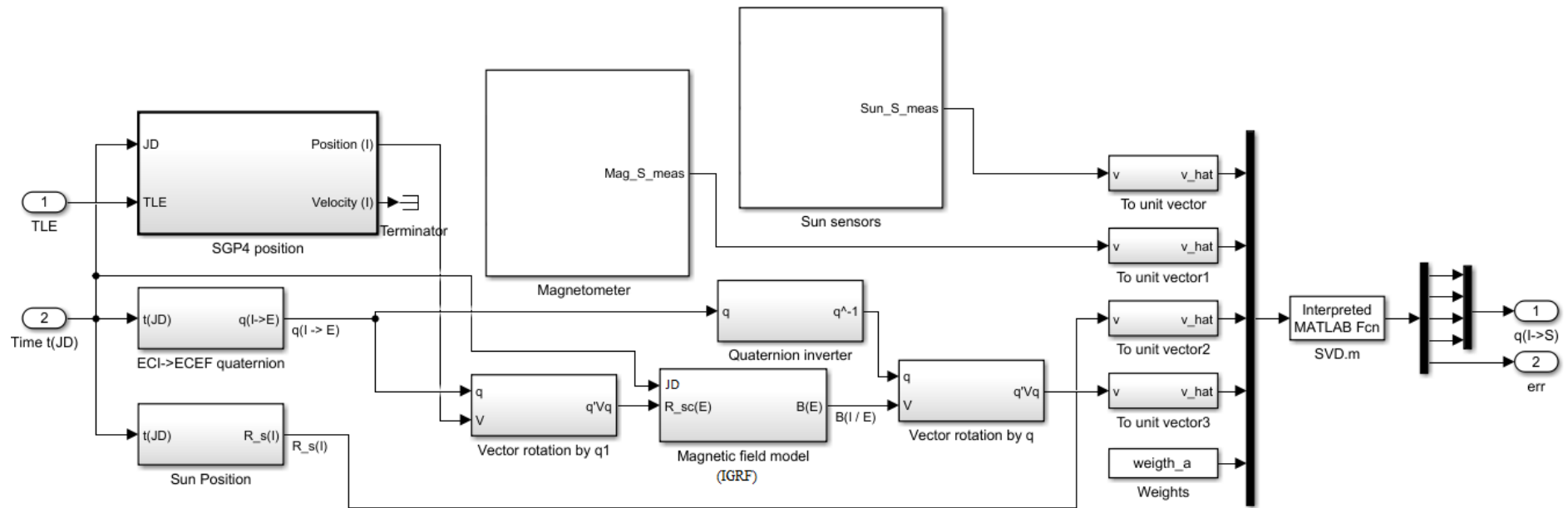
Satelliidi esialgse asendi määramiseks ja selle bakalaureusetöö paremaks mõistmiseks on selles peatükis välja toodud kõikide satelliidi esialgse asendimääramise komponentide sisendid ja väljundid koos visuaalse skeemiga.

Satelliidi asendimääramise erinevateks komponentideks koos sisendite ja väljunditega on:

- Päikese mudel, mis arvutab Päikese asendi ECI koordinaadistikus
 - Sisend: aeg Juuliuse kuupäeva (JD) formaadis
 - Väljund: vektor päikese asendiga ECI koordinaadistikus kilomeetrites
- Päikese andur, mis väljastab päikese asendi SBF koordinaadistikus ühikvektorina
 - Sisend: puudub
 - Väljund: Päikese suunaline ühikvektor SBF koordinaadistikus
- IGRF mudel, mis arvutab magnetvälja satelliidi asukohal ECEF koordinaadistikus
 - Sisend: aeg (JD), satelliidi positsioon ECEF koordinaadistikus vektorina
 - Väljund: magnetvälja vektor ECEF koordinaadistikus satelliidi asukohal
- Magnetomeeter

- Sisend: puudub
- Väljund: magnetvälja vektor satelliidi asukohal SBF koordinaadistikus
- SGP4 mudel, mis arvutab satelliidi positsiooni
 - Sisend: aeg (minutites) peale TLE aega ning TLE ise, milles sisaldub satelliidi positsioon orbiidil ja muud andmed. Tuleb tähele panna, et kuigi TLE-s on antud satelliidi positsioon, saab SGP4 mudeli abil arvutada satelliidi positsioone ajas edasi.
 - Väljund: satelliidi positsioon ECI koordinaadistikus kilomeetrites
- SVD meetodil põhinev asendimääramine
 - Sisend: Päikese asend ECI koordinaadistikus Maa keskpunktist ja SBF koordinaadistikus satelliidi keskpunktist ühikvektorites. Samuti magnetvälja vektor ECI koordinaadistikus ja SBF koordinaadistikus, mõlemad satelliidi keskpunktist ühikvektorites.
 - Väljund: Satelliidi asendit määrav kvaternioon ECI koordinaadistiku suhtes.

Järgnevalt esitatakse joonis (12), kus on välja toodud erinevad komponendid ja nende vaheline ühendus.



Joonis 12. Asendimääramise süsteemi lihtsustatud mudel (autori joonis)

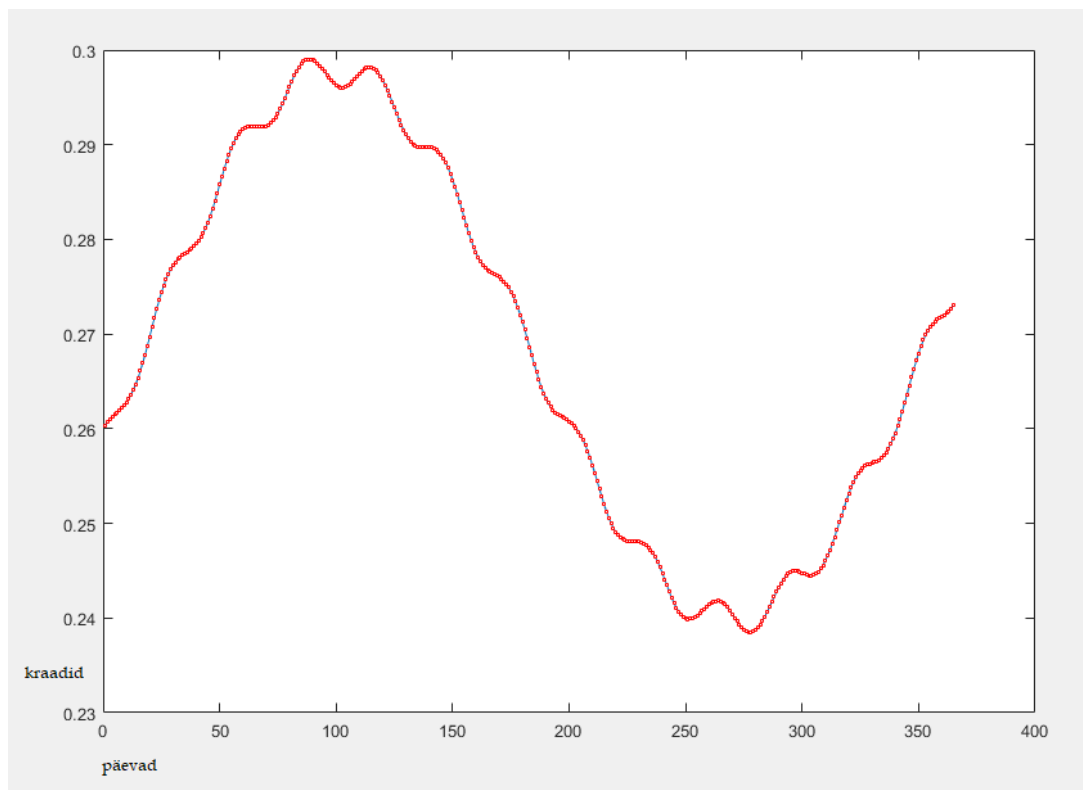
4 Tulemuste valideerimine

Selles peatükis valideeritakse päikese mudeli abil saadud päikesevektor.

4.1 Päikese mudeli valideerimine

Päikese mudeli valideerimiseks on kasutatud MATLAB-i *Aerospace Toolbox*'i koodi, täpsemalt *planetEphemeris* meetodit, mis tagastab päikesevektori ECI koordinaadistikus [30].

Päikese asendit on vaadeldud 2019. aasta kõigil päevadel. ning satelliidi lihtsustatud ning *Aerospace Toolbox*'i pakutud mudelitest saadud vektorite vahe on $\sim 0.23^\circ$ kuni $\sim 0.3^\circ$, mis on ADCS süsteemi jaoks piisava täpsusega.



Joonis 13. Päikese mudeli vektori vahe võrreldes Aerospace Toolbox'i arvutustega aastal 2019 (autori joonis)

Kokkuvõte

Bakalaureusetöö peamiseks eesmärgiks oli luua päikesemudel TTÜ100 satelliidi jaoks, mis oleks kiire täitmisajaga ja piisavalt täpne. Selle jaoks loeti läbi hulgaliselt kirjandust ja töid ning valiti välja üks mudel, mille tulemusena saadi päikesevektor ECI koordinaadistikus. Mudeli tulemus valideeriti MATLAB-is, mille tulemusena selgus, et välja valitud mudeli vektor erineb *Mathworks*'i poolt loodud *Aerospace Toolbox*'i meetodi *Planet Ephemeris* omast vaid 0.23 – 0.3 kraadi, mis on asendimääramise süsteemi jaoks piisava täpsusega.

Kuna tervikliku esialgse asendimääramise kohta varasemad tööd TTÜ-s puuduvad, siis otsustas autor selle luua ning lisas töösse ka teiste komponentide detailsemad kirjeldused.

Kirjeldatud on TTÜ100 satelliidi asendimääramiseks kasutatavat raudvara, mis koosneb ümbrusvalguse andurist VEML6030X01, Eiko Priideli arendatavast *pinhole* tüüpi päikeseandurist, magnetomeetrist HMC5983 ning mikrokontrollerist STM32F303VD. Samuti on välja toodud päikeseandurite kitsaskohad.

Läbi on töötatud satelliidi asendimääramise peamised koordinaadistikud, milleks on ECI, SBF ja ECEF koordinaadistikud. Nende juurde on lisatud illustreerivad pildid hoomamise lihtsustamiseks. Lisaks on loodud MATLAB-i kood ECI ja ECEF koordinaadistike vahel asendite konverteerimiseks.

Töös on kirjeldatud ka päikesevektori arutamist satelliidi koordinaadistikus päikeseanduritest saadud mõõtetulemuste abil. Adresseeritud on ka Albedo mudeli arvutusliku keerukust ning tõdeti, et selle teostamine satelliidil oleks arvutuslikult ebaotstarbekas. Selle asemel soovitati kasutada Eiko Priidel *pinhole* sensorit või asetada sensorite ette Albedo valgust filtreerivad füüsilised *bandpass* filtrid.

Lisaks on töös kirjutatud IGRF magnetvälja mudelist, mille implementeerimisega tegeleb hetkel Shu Taya.

Autor käsitles ka satelliidi asendi määramist orbiidil, mille jaoks analüüsiti detailselt TLE andmeformaati, et seejärel kirjeldada SGP4 mudelit. Välja pakuti SGP4 mudeli

teostus C-s, mille sisend on TLE, aeg TLE määramisest minutites ning väljund satelliidi asend ECI raamistikus, mille vektori komponendid on kilomeetrites.

Töös vaadeldi ka asendimääramise algoritmi, mis põhineb SVD meetodil. Selle teostus põhineb AAUSAT3-e asendimääramise süsteemil ning autor tegi selle kättesaadavaks git'i repositooriumis.

Lisaks on autor pealiskaudselt vaadelnud ka Kalmani filtrit. Autor järeldas, et Caroliina Laantee pakutud lahendus güroskoobiga vajab ajaliselt mahukat edasiarendust ning lisisisendeid päikese- ja magnetvälja vektorite, satelliidi asendi, samuti päikesevarju oleku ja satelliidi kontrollsüsteemi mähiste poolt rakendatud jõudude näol. Samuti peab lahendus kasutama Euleri nurkade asemel kvaternioone.

Lõpetuseks koostas autor esialgse asendimääramise komponentide nimekirja, koos sisendite ja väljundite nimekirjaga, ning kujutas neid omavahel ühendatuna joonisel, mis lihtsustab süsteemist tervikuna arusaamist.

Suur panus TTÜ100 ADCS meeskonnale on autori suhtlus Taani satelliidi AAUSAT3 arendusmeeskonnaga, kes andis autorile MATLAB-i ja Simulinki failid, mis simuleerivad tervikliku asendimääramist ja selle kontrollimist. Sellega lõi autor suurepäraseid eeldused tulevastele ADCS meeskonna liikmetele nii tarkvara realiseerimiseks kui ka selle valideerimiseks.

Töö aitas nii autoril kui ADCS meeskonnal tervikuna saada teadmisi asendimääramise tarkvara kohta ning aitas autoril juurde saada rahvusvahelises meeskonnas töötamise kogemust.

Kõik töös kirjeldatud algoritmide teostused on saadaval vastavas alapeatükis välja toodud viites või GitLab-is, mille aadress on olemas lisas (1).

Kasutatud kirjandus

- [1] V. Heikki, „Vallaste e-teatmik,“ [Võrgumaterjal]. Available: <http://www.vallaste.ee/>.
- [2] M. Veske, „SPINN-PEIERLSI ÜHENDI TIPO4 LOKAALSE STRUKTUURI UURINGUD 31P TMR ABIL,“ Tartu-Tallinn, 2013.
- [3] TTÜ, „TTÜ100 satelliidi koduleht,“ [Võrgumaterjal]. Available: <https://www.ttu.ee/projects/mektory-eng/satellite-programme-3/satellite-programme/ttu-mektory-satellite-programme/>. [Kasutatud 15 mai 2018].
- [4] California Polytechnic State University, „CubeSat Design Specification REV 13,“ 14 veebruar 2014. [Võrgumaterjal]. Available: https://static1.squarespace.com/static/5418c831e4b0fa4ecac1bacd/t/56e9b62337013b6c063a655a/1458157095454/cds_rev13_final2.pdf. [Kasutatud 16 mai 2018].
- [5] Vishay, „Mouser Electronics,“ [Võrgumaterjal]. Available: http://www.mouser.com/ds/2/427/vishay_VEML6030X01_84273_20151125_preliminary_cons-888264.pdf. [Kasutatud 18 mai 2018].
- [6] Vishay, „Arrow Electronics,“ [Võrgumaterjal]. Available: http://static6.arrow.com/aropdfconversion/740d0e0ec8edc4e1dfca55c3ce50b0527db4ab13/49670_pl0366.pdf. [Kasutatud 18 mai 2018].
- [7] Honeywell International Inc., „3-Axis Digital Compass IC HMC5983 Datasheet,“ jaanuar 2012. [Võrgumaterjal]. Available: https://aerocontent.honeywell.com/aero/common/documents/myaerospacecatalog-documents/Defense_Brochures-documents/HMC5983_3_Axis_Compass_IC.pdf. [Kasutatud 18 mai 2018].
- [8] STMicroelectronics, „STM32F303VD tehnilised andmed,“ [Võrgumaterjal]. Available: http://www.st.com/content/st_com/en/products/microcontrollers/stm32-32-bit-arm-cortex-mcus/stm32-mainstream-mcus/stm32f3-series/stm32f303/stm32f303vd.html. [Kasutatud 18 mai 2018].
- [9] STMicroelectronics, „STMicroelectronics koduleht,“ [Võrgumaterjal]. Available: <http://www.st.com/en/evaluation-tools/stm32f3discovery.html>. [Kasutatud 18 mai 2018].
- [10] C. Hall, „Spacecraft Dynamics and Control,“ 2003. [Võrgumaterjal]. Available: <http://www.dept.aoe.vt.edu/~cdhall/courses/aoe4140/attde.pdf>.
- [11] European Space Agency, „Navipidea,“ 23 veebruar 2012. [Võrgumaterjal]. Available: http://www.navipedia.net/index.php/Conventional_Celestial_Reference_System.
- [12] K. F. Jensen ja K. Vinther, „Attitude Determination and Control System for AAUSAT3,“ 2010. [Võrgumaterjal]. Available: http://projekter.aau.dk/projekter/files/32312420/thesis_10gr1035.pdf.

- [13] R. Raabe, „ESTCUBE-1 ENERGIATOOTLUSE SIMULAATOR JA TELEMEETRIA ANDMETE,“ 2014. [Võrgumaterjal]. Available: http://dSPACE.ut.ee/bitstream/handle/10062/41604/Raabe_Riho.pdf.
- [14] E. G. Richards ja K. P. Seidelmann, „Explanatory Supplement to the Astronomical Almanac,“ University Science Books, 1992. [Võrgumaterjal]. Available: http://aa.usno.navy.mil/publications/docs/c15_usb_online.pdf.
- [15] K. Krogh ja E. Schreder, „Attitude Determination for AAU CubeSat,“ 2002. [Võrgumaterjal]. Available: <http://www.space.aau.dk/cubesat/dokumenter/ADC-report.pdf>.
- [16] K. Pillet, „Kuubik-satelliidi asendi määramine päikesesensoritega. Attitude Determination of a Cube Satellite Using Sun Sensors,“ TTÜ, 6 juuni 2017. [Võrgumaterjal]. Available: <https://digi.lib.ttu.ee/i/?8448>.
- [17] D. A. Vallado, Fundamentals of Astrodynamics and Applications, 4th ed., Microcosm Press, 2013.
- [18] T. P. O. Montenbruck, „Astronomy on the Personal Computer,“ 1998. [Võrgumaterjal]. Available: https://ia600203.us.archive.org/23/items/springer_10.1007-978-3-662-03349-4/10.1007-978-3-662-03349-4.pdf.
- [19] J. Gaebler, „Coarse Sun Sensing for Attitude Determination of a CubeSat,“ 2007. [Võrgumaterjal]. Available: <http://ufdc.ufl.edu/AA00062245/00001>.
- [20] D. Bhandari ja T. Bak, „Modeling Earth Albedo for Satellites in Earth Orbit,“ 2005. [Võrgumaterjal]. Available: [http://vbn.aau.dk/en/publications/modeling-earth-albedo-for-satellites-in-earth-orbit\(0c432530-9c2d-11db-8ed6-000ea68e967b\).html](http://vbn.aau.dk/en/publications/modeling-earth-albedo-for-satellites-in-earth-orbit(0c432530-9c2d-11db-8ed6-000ea68e967b).html).
- [21] F. R. Hoots ja R. L. Roehrich, „SPACETRACK REPORT NO. 3 Models for propagation of norad element sets,“ 1988.
- [22] D. A. Vallado, „Astrodynamics Software for Fundamentals of Astrodynamics and Applications,“ 15 märts 2018. [Võrgumaterjal]. Available: <https://celestrak.com/software/vallado-sw.asp>. [Kasutatud 12 mai 2018].
- [23] P. Crawford, „The Dundee SGP4 code,“ 2010. [Võrgumaterjal]. Available: <http://www.sat.dundee.ac.uk/psc/sgp4.html>. [Kasutatud 12 mai 2018].
- [24] T. S. Kelso, „More Frequently Asked Questions,“ Satellite Times, jaanuar 1998. [Võrgumaterjal]. Available: <https://celestrak.com/columns/v04n03/>. [Kasutatud 6 mai 2018].
- [25] R. Merrill, The magnetic field of the earth: paleomagnetism, the core, and the deep mantle, Academic Press, 1996.
- [26] K. R. Lang, „Tufts University NASA's Cosmos,“ 2010. [Võrgumaterjal]. Available: https://ase.tufts.edu/cosmos/view_picture.asp?id=326. [Kasutatud 2 mai 2018].
- [27] E. Thebault, C. C. Finlay ja ..., „International Geomagnetic Reference Field: the 12th generation,“ 2015.
- [28] M. S. Peter, „Stochastic models, estimation, and control VOLUME 1,“ 1979. [Võrgumaterjal]. Available: https://www.cs.unc.edu/~welch/kalman/media/pdf/maybeck_ch1.pdf.
- [29] C. Laantee, „The suitability analysis and simulations of the state estimation algorithm for the TTU Mektory nanosatellite,“ 2017. [Võrgumaterjal]. Available: <https://digi.lib.ttu.ee/i/?8009>.

- [30] MathWorks, „Aerospace Toolbox Documentation,“ MATLAB, [Võrgumaterjal]. Available: <https://se.mathworks.com/help/aerotbx/ug/planetephemeris.html>.
- [31] D. A. Vallado, P. Crawford, R. Hujsak ja T. S. Kelso, „Revisiting Spacetrack Report #3: Rev 2,“ 2006.

Lisa 1 – GitLab’i link

<https://gitlab.com/martinrebane/ADCS/>

Lisa 1 – MATLAB-i koodid

4.2 Päikese mudeli MATLAB-i kood

```
function [Seci] = sun_eci(JD)
%-----
%-----
% Build Sun Vector in Earth Centered Intertial Frame
%-----
%-----
%
% -----
% Inputs
% -----
% JD      (:)      Time of determination in Julian Date
% rSc     (3,:)    Spacecraft vector in the ECI frame (km)
%
% -----
% Outputs
% -----
% Seci    (3,:)    Sun vector (accounted for parallax if satellite
position given)
%                               ECI frame Cartesian coordinates in km
% Example
% sun_eci(2458480.40277) %ECI Sun position 27th of December 2018 21:40
%-----
%-----
%-----
% References: [1] K. Krogh ja E. Schreder, „Attitude Determination for AAU
CubeSat“,
%           Aalborg University, Aalborg, 2002. p62
%
%           [2] C. Hall, „Spacecraft Dynamics and Control,“ 2003. p.4-7
%           Available:
http://www.dept.aoe.vt.edu/~cdhall/courses/aoe4140/attde.pdf.
%
%-----
%-----

J2000 = 2451545;          %Jan 1st 2000 12:00:00.0

%Mean Anomaly of Sun [1]
Msun = 357.528 + 0.9856003 * (JD-J2000);

%Ecliptic longitutde of Sun [1]
Vsun = 280.461 + 0.9856474 * (JD-J2000) + 1.95 * sin(Msun*pi/180) + 0.020 *
sin(2 * Msun*pi/180);
```

```

%Obliquity of the ecliptic [1]
E = 23.4393 - 0.0000004 * (JD-J2000);

%Transform to Cartesian coordinates [1]
XSun = cos(Vsun*pi/180);
Ysun = cos(E*pi/180) * sin(Vsun*pi/180);
ZSun = sin(E*pi/180) * sin(Vsun*pi/180);

Seci = [ XSun; Ysun; ZSun ];

%Calculate distance to Sun in AU-s [2]
r = 1.000140612 - 0.016708617 * cos(Msun) - 0.000139589 * cos(2 *Msun);

%Convert AU to km
%1 Astronomical Unit = 149.6 million km
r = r * 149.6e6;

% Place results in structure
Seci(1) = Seci(1) * r;
Seci(2) = Seci(2) * r;
Seci(3) = Seci(3) * r;

end

```

Joonis 14. MATLAB-i kood päikese vektori arvutamiseks ECI koordinaadistikus

4.3 Päikesevarjutuse mudeli MATLAB-i kood

```
function result = eclipse(Seci, seci)
%-----
%-----
% Eclipse model for cube satellite
%-----
%-----
%
% -----
% Inputs
% -----
% Seci      (3,:)  Sun vector ECI frame, Cartesian coordinates in km
% seci      (3,:)  Sun vector SBF (satellite body frame, Cartesian
coordinates in km
%
% -----
% Outputs
% -----
% result    integer  1 if satellite is in eclipse, 0 if it is not
%
%-----
%-----
%
%-----
%-----
% References: [1]K. F. Jensen ja K. Vinther,
%             „Attitude Determination and Control System for AAUSAT3,“
2010. pp.66-67
%             Available:
%
http://projekter.aau.dk/projekter/files/32312420/thesis_10gr1035.pdf.
%
%
%-----
%-----

rEarth = 6371000; %meters
Ssbf = -seci + Seci;
theta = acos( dot((Ssbf/norm(Ssbf)), (Seci/norm(Seci)) ));
d = sqrt(1 - ( dot((-seci + Seci)/norm(-seci + Seci), (-Seci /
norm(Seci))))^2) * norm(seci);
if (theta < 90 && d < rEarth)
    result = 1;
else
    result = 0;
end
```

Joonis 15. Päikesevarjutuse mudeli MATLAB-i kood

4.4 Juulise kuupäeva arvutamise MATLAB-i koodi

```
function JD = calculate_julian_day(year, month, day, hour, minute, second)
%-----
%-----
% Calculate Julian Day from Gregorian calendar date
%-----
%-----
%
% -----
% Inputs
% -----
% year      (:)    Year
% month     (:)    Month
% day       (:)    Day
% hour      (:)    Hour in 24h system
% minute    (:)    Minute
% second    (:)    Second
%
% -----
% Outputs
% -----
% JD        (:)    Time in Julian Date
%
% Example
% calculate_julian_day(2018,12,27,21,40,12) %Calculates julian date for
% 27th of
% December 2018 21:40:12
%-----
%-----
%-----
% References: [1] K. Pillet, „Kuubik-satelliidi asendi määramine
päikesesensoritega.
% Attitude Determination of a Cube Satellite Using Sun
Sensors,“
% TTÜ, 6 juuni 2017. [Web]. Available:
https://digi.lib.ttu.ee/i/?8448
%
% [2] http://calendars.wikia.com/wiki/Julian_day_number
%-----
%-----
%Expalantion from [2], but because it's wikipedia, reference [1] is also
%provided for "academic" correction purposes
a = fix((14 - month) / 12);
y = fix(year + 4800 - a);
m = fix(month + 12 * a - 3);

J = day+ fix((153 * m + 2)/5) + 365 * y +fix(y/4) - fix(y/100) + fix(y/400) -
32045;
```

```
JD = J + (hour-12)/24 + minute/1440 + second/86400;
```

```
%This code below works as well and is faster, because it doesn't declare  
%unnecessary variables, but is not very readable
```

```
%JD = 367*year-  
floor(7*(year+floor((month+9)/12))/4)+floor(275*month/9)+day+1721013.5+hour/2  
4+minute/1440+second/86400;  
end
```

Joonis 16. Juuliuse kuupäeva arvutamise MATLAB-i kood

4.5 SVD meetodi MATLAB-i kood

```

function [quaternion] = SVD(a1,b1_i,b1_s,a2,b2_i,b2_s)
% Solve Wahba's problem by Singular Value Decomposition (originally by
% Markley)
%-----
%
% -----
% Inputs
% -----
% a1          float      Weight on first vector observation (usually
1/sigma^2)
% a2          float      Weight on second vector observation (usually
1/sigma^2)
% b1_i        (1,3)      First vector observation in ECI.
% b1_s        (1,3)      First vector observation in SBRF.
% b2_i        (1,3)      Second vector observation in ECI.
% b2_s        (1,3)      Second vector observation in SBRF.
%
% -----
% Outputs
% -----
% quaternion (1,4)      The quaternion based on the the optimal rotation
matrix A.
%      Error indicator (1=quaternion cannot be trusted, 0=succes).
%-----
%-----
%
% References:  K. Vinther and K. F. Jensen, Aalborg University, 2010.
%-----
%-----

B=a1*b1_s*(b1_i)'+a2*b2_s*(b2_i)';

% Singular Value Decompositionn, B=U*S*V'
[U,S,V]= svdmex(B);
temp=det(U)*det(V);

% Check covariance P (of the rotation angle error vector)
s1=S(1,1);
s2=S(2,2);
s3=temp*S(3,3);
P=U*diag([(s2+s3)^(-1),(s3+s1)^(-1),(s1+s2)^(-1)])*U';
temp2=sum(P);
P_sum=sum(temp2);

% If the covariance is big then the attitude is unobservable
if P_sum < 0.1
    % Calculation of optimal attitude matrix

```

```

Aopt=U*diag([1,1,temp])*V';

% Attitude matrix to quaternion conversion
quaternion(1:4)=A_to_q(Aopt);

% invert quaternion
%output(1:3)=-output(1:3);
else
    quaternion(1:4)=[0 0 0 1];
end

% output must be finite in order to pass to Matlab function blok
if isinf(P_sum) == 1 || P_sum > 1000 || isnan(P_sum) == 1
    quaternion(5)=1000;
else
    quaternion(5)=P_sum;
end
end

```

Joonis 17. SVD meetodi MATLAB-i kood

Tuleb tähele panna, et $[U,S,V]=\text{svdmex}(B)$; on MATLAB-i mex funktsioon, mille leiab git'i repositooriumist (sisaldab 368 rida).

4.6 Maatriksist kvaterniooni teisendamise MATLAB-i kood

```
function [q] = A_to_q(A)
%-----
%
% % Attitude matrix to quaternion conversion
%-----
%
% -----
% Inputs
% -----
% A      (3,3)      Attitude 3x3 matrix
%
% -----
% Outputs
% -----
% q      (1,4)      quaternion
%
%-----
%-----
%
% References:
http://www.euclideanspace.com/maths/geometry/rotations/conversions/matrixToQuaternon/index.htm
%-----
%-----

a = A;
q = zeros(4,1);
% Find i for largest denominator
q(4) = 0.5*sqrt(1+a(1,1)+a(2,2)+a(3,3));
dmax = [q(4), a(1,1), a(2,2), a(3,3)];
[nymax i] = max(dmax);

if (i==1)
q(4) = 0.5*sqrt(1+a(1,1)+a(2,2)+a(3,3));
q(1) = 0.25*(a(2,3)-a(3,2)) / q(4);
q(2) = 0.25*(a(3,1)-a(1,3)) / q(4);
q(3) = 0.25*(a(1,2)-a(2,1)) / q(4);
elseif (i==2)
q(1) = 0.5*sqrt(1+a(1,1)-a(2,2)-a(3,3));
q(2) = 0.25*(a(1,2)+a(2,1)) / q(1);
q(3) = 0.25*(a(1,3)+a(3,1)) / q(1);
q(4) = 0.25*(a(2,3)-a(3,2)) / q(1); % Here Sidi has a sign error
elseif (i==3)
q(2) = 0.5*sqrt(1-a(1,1)+a(2,2)-a(3,3));
q(1) = 0.25*(a(1,2)+a(2,1)) / q(2);
q(3) = 0.25*(a(2,3)+a(3,2)) / q(2);
q(4) = 0.25*(a(3,1)-a(1,3)) / q(2);
```

```
elseif (i==4)
q(3) = 0.5*sqrt(1-a(1,1)-a(2,2)+a(3,3));
q(1) = 0.25*(a(1,3)+a(3,1)) / q(3);
q(2) = 0.25*(a(2,3)+a(3,2)) / q(3);
q(4) = 0.25*(a(1,2)-a(2,1)) / q(3);
end
%if (q(4) < 0)
%q = - q;
%end
end
```

Joonis 18. Maatriksist kvaterniooni teisendamise MATLAB-i kood