



TAMPEREEN TEKNILLINEN YLIOPISTO

HENRIK PAMMO
FURUTA-KÄÄNTÖHEILURIN SUMEA SÄÄDIN
OPETUSKÄYTTÖÖN
Diplomityö

Tarkastajat:
professori Hannu Koivisto,
yliopisto-op. Mikko Laurikkala

Tarkastajat ja aihe hyväksytty
Teknisten tieteiden tiedekuntaneu-
voston kokouksessa
8. huhtikuuta 2015

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO

Automaatiotekniikan koulutusohjelma

Pammo, Henrik: Furuta-kääntöheilurin sumea säädin opetuskäyttöön

Diplomityö, 45 sivua, 18 liitesivua

Toukokuu 2015

Pääaine: Oppivat järjestelmät

Tarkastajat: professori Hannu Koivisto, yliopisto-opettaja Mikko Laurikkala

Avainsanat: simulaatio, mallinnus, sumea logiikka, sumea säätö, parallel distributed control, sector nonlinearity, Furuta, kääntöheiluri

Sumea säätö on säätötekniikan alue, jota voidaan soveltaa, kun ongelma on helpommin kielellisesti kuin matemaattisesti ilmaistavissa tai kun säädettävä prosessi on vaikeasti mallinnettava tai epälineaarinen. Sumea säädin laskee ulostulonsa asetusarvon ja mittausdatan pohjalta käyttäen sääntökantaansa, joka testaa sumeiden sisääntulojen osuudesta sumeisiin jäsenyysfunktioihin. Lopuksi tulos selkeytetään ja ulostulo muodostetaan. Sääntökannan luominen ja sopivan resoluution valinta on aikaavievää ja joskus myös vaikeaa. Myös jäsenyysfunktioiden valinta ja viritys ja selkeytysmenetelmän valinta kuuluvat säätimen rakentamisen haasteisiin. Sumeita säätimiä voidaan käyttää paitsi suoraan säätöön, myös perinteisemmän, esimerkiksi PID-säädön, apuna. Tällöin niiden avulla valitaan toimintapiste, optimoidaan parametreja tai reagoidaan poikkeustilanteisiin, joista tavallinen säädin ei selviydy.

Työssä perehdytään klassiseen säätöteorian esimerkkiongelmiaan, Furuta-kääntöheiluriin. Tutkittava laitteisto koostuu opetuskäyttöön tarkoitetuista Quanser Qube rotating inverted pendulum -kääntöheilurista ja NI:n cRIO -datankeräyslaitteesta. Tavoitteena on luoda sumea säädin kääntöheilurille käytettäväksi myöhemmin opetustarkoituksessa. Tätä tarkoitusta varten työssä pohditaan opiskelijoille sopivia säätömenetelmään liittyviä harjoituksia. Säätömenetelmää ja työssä tehtyä säädintä voidaan käyttää esiteltäessä opiskelijoille kehittyneitä säätöratkaisuja.

Käytettävät ohjelmistot ovat Matlab ja Simulink, joilla luodaan simulaattori heilurille heilurin liikeyhtälöiden perusteella ja simuloidaan sumeaa parallel distributed -säätöä. Säädin perustuu Takagi-Sugeno-tyyppiseen sumeeseen järjestelmään, jolla tulkitaan järjestelmän epälineaarisuuksia. Parallel distributed -säädössä säädettävää prosessia kuvataan usealla lokaalilla mallilla ja sumea järjestelmä tekee näistä jäsenyysfunktioiden painottaman keskiarvoisen järjestelmän. Työssä säädin suunnitellaan LQR-periaatteilla tavallisilla lineaarisilla työkaluilla, mikä on tämäntapaisen sumean järjestelmän vahvuus. Samalla tavalla kuin mallikin, myös säädin on lokaalien säätimien sumean keskiarvon lopputulos. Lokaalit mallit saadaan local sector nonlinearity -menetelmän avulla. Menetelmässä etsitään paikallisia sektoreita, joilla järjestelmän keskeiset parametrit liikkuvat ja muodostetaan mallit näiden ääripäistä.

Työn tuloksena saadaan sumea epälineaarinen tilatakaisinkytketty säädin, joka on tavallista linearisoidun mallin kautta suunniteltua lineaarista LQR-säädintä nopeampi. Lisäksi sumeaa epälineaarista säädintä käyttämällä heilurin liikkeet ovat pienemmät ja värähtelyä on vähemmän. Työn ohessa tunnistetaan useita opiskelijoille sopivia sumeeseen säätöön ja valittuun säätömenetelmään liittyviä tehtäviä.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY

Master's Degree Programme in Automation Science and Engineering

PAMMO, HENRIK: Fuzzy Controller of a Furuta Pendulum for Educational Use

Master of Science Thesis, 45 pages, 18 Appendix pages

May 2015

Major: Automation and Control Engineering

Examiners: Professor Hannu Koivisto, University Teacher Mikko Laurikkala

Keywords: simulation, modeling, fuzzy logic, fuzzy control, parallel distributed control, sector nonlinearity, Furuta, rotating inverted pendulum

Fuzzy control is an area of control engineering that can be applied when a problem is easier to express linguistically than mathematically or when a process is nonlinear or otherwise difficult to model. A fuzzy controller's output is determined from evaluating its fuzzy inputs in relation to a set of fuzzy rules, a knowledge base. The inputs are fuzzified by giving them a membership value that relates to a membership function. Lastly, the fuzzy output must be defuzzified to produce a crisp output.

Selecting the fuzzy rulebase and finding appropriate membership functions can be very time consuming and sometimes also difficult. Other difficulties include the needed resolution of the membership functions and the right method of defuzzification. Tuning the controller is sometimes very laborious as there are no standard methods to achieve a good tuning.

Fuzzy controllers can be used to directly control process actuators at the lowest level of control but they are normally more useful at a higher control level, providing for example, optimization of PID control parameters or maybe acting as a failsafe in abnormal process conditions.

In this thesis, a classic control theory problem, the Furuta rotating inverted pendulum, is studied. The equipment is meant for educational use. It consists of a NI cRIO data logger and a Quanser Qube motor unit with a rotating inverted pendulum addition. The goal is to create a balancing fuzzy controller for this Furuta pendulum for later use in control engineering education. For this purpose, several potential exercises for students are considered throughout the thesis. The chosen control method can be shown to students as an example of advanced control.

Matlab and Simulink are used to create a simulator based on the pendulum's nonlinear equations of motion and to design a fuzzy parallel distributed controller. It's based on a Takagi-Sugeno type fuzzy system that interpretes the system's nonlinearities. Parallel distributed systems are described with several local models that are combined by the fuzzy inference system. The controller also uses this approach and consists of several local LQR controllers designed for the local models. The local models are found using the local sector nonlinearity method that uses locally valid parameters to form a space of possible control designs.

The resulting fuzzy state feedback controller is faster than the conventional LQR controller. It is also more accurate with smaller pendulum movements and less oscillation. Several potential exercises for students on the topic of fuzzy control and the chosen PDC control with local sector nonlinearity are discovered.

ALKUSANAT

Suuri kiitos Koulutusrahastolle ja aikuiskoulutustukijärjestelmälle, joita ilman tämän työn tekeminen ei olisi ollut mahdollista.

Vähintään yhtä suuri kiitos kuuluu puolisololleni Petralle, joka on ollut opiskelun tukena ja jonka ajoittain koeteltu kärsivällisyys diplomityöprojektini aikana onneksi riitti tarpeeksi pitkälle.

Kiitos TTY:n henkilökunnalle tuesta.

Tampereella 10.4.2015

Henrik Pammo

SISÄLLYS

1	Johdanto.....	1
1.1	Tavoitteet.....	1
1.2	Työn rakenne.....	2
2	Sumea logiikka.....	4
2.1	Keskeiset käsitteet ja perusteet.....	4
2.1.1	Sääntökanta.....	7
2.2	Sumeat järjestelmätyypit.....	7
2.2.1	Mamdani-järjestelmät.....	7
2.2.2	Takagi-Sugeno -järjestelmät.....	9
2.2.3	Mamdani versus T-S.....	10
2.3	Sumea säätö.....	10
2.3.1	Parallel distributed control (PDC).....	11
2.3.2	Local sector nonlinearity.....	12
2.3.3	Viritys.....	13
3	Koejärjestely.....	15
3.1	Laitteet ja prosessi.....	16
3.1.1	Quanser QUBE.....	17
3.1.2	National Instruments CompactRIO.....	17
3.2	Ohjelmistot.....	19
4	Järjestelmän malli.....	21
4.1	Sumea malli.....	24
5	Toteutus.....	31
5.1	Säätömenetelmät.....	31
5.2	Simulaatiotuloksia.....	34
5.3	Havaitut ongelmat.....	40
5.4	Hyviä ratkaisuja.....	41
6	Tulokset ja johtopäätökset.....	42
	Lähteet.....	44
	Liitteet.....	46
	A. Mallin tekemisessä hyödynnettyä Matlab-koodia.....	47
	B. Heilurimallin FIS-tiedosto.....	52
	C. Säätimen FIS-tiedosto.....	57
	D. Sumean järjestelmän stabiiliuden todistaminen.....	60
	E. Simulaattori swing-upilla ja sumealla säätimellä.....	62

TERMIT JA NIIDEN MÄÄRITELMÄT

$\mu_A(x)$	Sumean joukon A jäsenyysfunktio
$A \cup B$	A:n ja B:n unioni
$A \cap B$	A:n ja B:n leikkaus
\tilde{x}, \tilde{X}	Lingvistinen muuttuja tai arvo
${}^y x, x_y$	X:n indeksointi
\dot{x}	X:n derivaatta
x^T	X:n transpoosi
A_i, B_i, C_i, D_i	Järjestelmän tilaesityksen matriisit
B_r, B_p	Kitka
D_r, D_p	Viskoosinen vaimennuskerroin
g	Putoamiskiihtyvyyys
J	Hitausmomentti (inertia)
K	Tilatakaisinkytkennän vahvistusmatriisi
k_m	Vastasähkömotorinen voima
k_t	Moottorin ominaisvääntö
L	Pituus
L, T, V, Q_i	Voima
m	Massa
M_{ij}	Sumea joukko
R_m	Resistanssi (moottorin)
V_m	Moottorin jännite
$z_i(t)$	Sector nonlinearity -apumuuttuja
α	Heilurin kulma
θ	Varren kulma
τ	Vääntömomentti

(Local) sector nonlinearity	PDC-säädön toteutustapa, missä lokaalimallit muodostetaan mahdollisen säätöavaruuden rajoista
CA	Center Average, selkeytystapa
CoG	Center of Gravity, painopiste, selkeytystapa
FIS	Fuzzy Inference System, sumea päättelyjärjestelmä
Furuta-heiluri	Eräs kääntöheilurimalli, myös ”rotating inverted pendulum”
Gain scheduling	Tapa muuttaa säätöä vallitsevien olosuhteiden mukaan
I/O	Input/Output
LMI	Linear matrix inequality
LQR	Linear-quadratic regulator, säädintyyppi
Mamdani	Sumean järjestelmän tyyppi, missä seurausosa on sumea joukko
MoM	Mean of Maximum, selkeytystapa
Partition spaces	PDC-säädön toteutustapa, missä lokaalimallit muodostetaan toiminta-avaruuden mukaan
PDC, parallel distributed compensation tai control	Sumea säätötapa, missä järjestelmän epälinearisuuksia kuvataan lineaaristen osajärjestelmien sumealla yhdistelmällä
PID	Proportional-integral-derivative, säädintyyppi
PWM	Pulse width modulation
Swing-up	Heilurin heilautus alhaalta vakaasta tasapainoasemastaan ylös pystyyn
Takagi-Sugeno, T-S	Sumean järjestelmän tyyppi, missä seurausosa on funktio
TCP/IP	Verkkoprotokolla
UDP	Verkkoprotokolla

1 JOHDANTO

1.1 Tavoitteet

Työn tavoitteena on esitellä sumeaa säätöä automaatioalan opiskelijoiden kurssikäyttöön. Opetushenkilökunta voi käyttää sisältöä kurssimateriaalin pohjana ja kokeita esimerkkinä. Säätöratkaisuja esitellään perinteisiin säätömenetelmiin tutustuneille opiskelijoille. Opetuskäytössä on tarkoitus uusia vanha hidas esimerkkiprosessi nopeammalla ja helpommin seurattavalla. Opiskelijoille työ tarjoaa perehdytyksen heilurilaitteistoon, sen malliin ja muutamaa mahdolliseen säätötapaan.

Kirjoitushetkellä saatavilla olevan tiedon mukaan työssä esiteltyä Furuta-heilurilaitteistoa käytetään demo-tyylisesti nuorempien opiskelijoiden kursseilla säädön esittelyssä ja syventävillä kursseilla laboratorioharjoituksena, jossa opiskelijat pääsevät itse toteuttamaan säätimen joko simulaationa tai oikealla prosessilla. Furuta-heiluri on laajalti koulutuskäytössä käytetty prosessi ja sitä on tutkittu paljon, muun muassa [1][2][4][6][19] ja sen säätöä myös [3][13][22][23][24].

Työssä aikaansaatu säädin voi toimia pohjana esitellessä opiskelijoille kehittyneitä säätöratkaisuja, *linear matrix inequality* (LMI) -matematiikkaan liittyvää optimisäätöä, sumeaa säätöä yleisesti tai epälineaarista säätöä. Tavoitteena ei ole niinkään esittää jostain yhtä loppuun saakka hiottua simulaatiomallia ja sen säädintä, vaan osoittaa *parallel distributed* -säädön toimivuus ja epälineaarisen säädön edut linearisoituun vaihtoehtoon nähden. Sumeiden säätimien virityksen vaikeuden tähden optimaaliseen viritykseen tarvittava aika aiheuttaisi reilun ylityksen työn aikatauluun. Tämän lisäksi sumean mallin ja säätimen optimointi on hyvä harjoitustyö opiskelijoille, sillä työn pohjana olevan säätimen ymmärtäminen ja kokeileminen vaativat monenlaista osaamista liittyen sumeaan säätöön ja säätötekniikkaan yleisesti. Lähtökohdan, sumea tilatakaisinkytkentä, voi katsoa olevan selkeä ja opiskelijoille tuttu.

Opetuskäytön lisäksi tavoitteena on hyvä sumea säädin. Valittu menetelmä kytketään opiskelijoille tuttuihin asioihin, kuten napojen asetteluun ja gain scheduling -menetelmään. Tähän tarkoitukseen muun muassa Lillyn [9] kirjassa esitelty *parallel distributed pole placement control*, eli eräänlainen Takagi-Sugeno -mallinen sumea napojen asetteluun perustuva järjestelmä on hyvä. Vastaava LQR-säätö (linear-quadratic regulator) on esitetty Tanaka et al. [17], ja tässä työssä onkin päädytty käyttämään LQR-säätöä sen antaman hyvän vertailukohdan ansiosta, sillä laitteistovalmistajan valmiiseen laboratorioharjoitusvalikoimaan kuuluu linearisoitu LQR-säätö.

Kokonaisuutena valittu tapa on käyttötarkoitukseensa melko monimutkainen, sillä tasapainottavan säätimen tekemiseen on yksinkertaisempiakin ratkaisuja ja sumeasta säätimestä tulee opiskelijalle jokseenkin epäintuitiivinen ja järjestelmän mallinnuksessa käytettävät yhtälöt ovat opetuskäyttöön melko hankalia. Oletuksena kuitenkin on, ettei opiskelija itse johda järjestelmämallia, vaan demolaitteiston valmistajan toimittama tai esimerkiksi tästä työstä saatu on tarpeeksi hyvä opiskelijan tarpeisiin ja valmiina saatavilla. Kirjallisuudesta on mahdollista löytää myös tarpeeksi tukea oman mallin luomiseen, mikäli se on opiskeltavan asian kannalta tarpeellista.

1.2 Työn rakenne

Johdantoluvun jälkeen toisessa luvussa keskitytään sumeaan teoriaan. Sumean teorian periaatteiden ymmärtäminen on olennaista työn seuraamisen kannalta, sillä teoria muodostaa tietenkin pohjan, jonka päälle uudet tekniikat pystytetään. Työhön valitut menetelmät astuvat myös sumean teorian ulkopuolelle, esimerkiksi säätöteoriaan ja LMI-matematiikkaan. Näitä käsitellään työkaluina tarpeellisella laajuudella työn tekemiseksi.

Sumeasta teoriasta käsitellään lyhyesti sen kehitystä ja historiaa, sekä näihin kytkettyä erilaisten sumeiden järjestelmien rakenteita. Peruskäsitteet liittyvät näiden ymmärtämiseen olennaisesti. Työssä on käytetty kahta valtavirran ulkopuolista menetelmää, *parallel distributed* -mallinnusta ja -säätöä sekä tähän liittyen *local sector nonlinearity* -mallinnustapaa, joka on esitetty alkujaan jo 1992 Kawamoto et al.:ssa [8]. Yhdessä nämä muodostavat aluksi hieman epäintuitiivisen, mutta lopulta selkeän ja tehokkaan tavan käsitellä vahvasti epälineaarista Furuta-heilurijärjestelmää. Parallel distributed -menetelmässä [9][17] on kyse epälineaarisen mallin hajottamisesta useaan linearisoi-tuun osaan, joista sumea päättelyjärjestelmä muodostaa sumean epälineaarisen mallin. Näille linearisoiduille malleille luodaan sitten lineaariset säätimet, joista myös muodostetaan sumealla päättelyllä epälineaarinen sumea säädin.

Kolmannessa luvussa esitellään laboratoriolaitteisto, jonka simulointiin työ perustuu. Varsinaiset opetuskäyttöön tulevat laitteistot eivät olleet vielä työn kirjoitushetkellä käyttövalmiina, vaan Quanserin heilurisarjasta oli saatavilla toinen laitteistoversio, jonka ohjelmisto ei ollut enää yhteensopiva uudempien tietokoneiden kanssa.

Laitteiston esittelyn jälkeen voidaan siirtyä mallin tekemiseen. Järjestelmän epälineaarinen malli on tunnettava, jotta voidaan rakentaa sen pohjalta sumea malli. Epälineaarinen malli tehdään heilurijärjestelmän liikeyhtälöiden ja laitteistovalmistajan materiaaleista kerättyjen parametrien avulla. Sumean mallin rakentamista varten ratkaistaan sopivat lokaalimallit, jotka vuorostaan saadaan *local sector nonlinearity* -menetelmän avulla. Tuloksena on epälineaarista järjestelmää approksimoiva sumea Takagi-Sugeno -tyyppinen järjestelmä. Järjestelmien eroja tutkitaan testifunktion vasteiden avulla.

Viidennessä luvussa on luontevaa siirtyä mallin olemassaolon jälkeen säätötoteutukseen. Säädin on parallel distributed -tyyppinen, eli se on epälineaarinen sumea säädin, joka koostuu lineaarisia lokaaleja säätimiä painotettuna keskiarvona yhdistävästä sumeasta päättelyjärjestelmästä. Lokaalit säätimet suunnitellaan tavallisilla lineaarisen säädön työkaluilla. Tässä työssä ne toteutetaan LQR-säätiminä, jotta saadaan hyvä vertailu laitevalmistajan linearisoidulle LQR-säädölle. Lohkokaaviomuodossa säätöratkaisu näyttää lähes tulkoon tavalliselta tilatakaisinkytkennältä. Lisänä on vain sumea lohko muuttamassa tilatakaisinkytkennän vahvistusta. Tilahavaitsijana käytetään Quanserin ratkaisua.

Simulaatiotuloksissa verrataan lineaarista LQR-säädintä ja epälineaarista sumeaa säädintä keskenään eri tilanteissa. Kiinnostavia asioita ovat säädinten suoriutuminen poikkeustilanteista ja suorituskyky. Epälineaarinen säädin osoittautuu odotetusti paremmaksi molemmissa tapauksissa.

Työtä tehdessä havaitut ongelmat ja niiden ratkaisuehdotukset on myös koottu viidenteen lukuun. Kun on tiedossa, että Furuta-heiluri on mahdollista pitää tasapainoasemassaan myös linearisoidulla yksinkertaisella säätimellä, tuntuu vaativan sumean järjestelmän tekeminen ehkä jopa liioittelulta.

Viimeisessä luvussa kootaan yhteen työn tulokset ja pohditaan niiden vaikutuksia. Sumea epälineaarinen säätö voidaan todeta hyväksi ratkaisuksi tilanteisiin, joissa prosessi on rakenteeltaan niin vaativa, osittain tuntematon tai vahvasti epälineaarinen, ettei yksinkertaisempi linearisoitu perusratkaisu käy säätöongelman ratkaisuksi.

2 SUMEA LOGIIKKA

Sumea logiikka on 1960-luvulla kehitetty [20] matemaattisen logiikan laajennus, jossa totuusarvot ilmaistaan diskreetin *tosi/epätosi*-asetelman asemesta suljetulla välillä nol-lasta yhteen $[0,1]$. Sumea logiikka syntyi tarpeesta ilmaista sanallisen epätarkkaa maail-maa tarkemmin ja käsiteltävämmin. Sumea järjestelmä kuitenkin käsittelee sisäänme-noinaan täsmällisiä lukuja ja tuottaa myös täsmällisen ulostulon. Sumea säätö on suosit-tu sovellusalue, kuten myös nykyisin tärkeä ”big datan” analyysi, signaalinkäsittely ja konenäkö.[20]

Tässä työssä sumeaa logiikkaa sovelletaan epälineaarisen järjestelmän säätöön. Juuri epälineaariset ja sellaisina vaikeasti mallinnettavat ja säädettävät järjestelmät ovat su-meän säädön vahvuusaluetta ja suosittu soveltamiskohde.

Sumean logiikan isäksi on yleisesti tunnustettu Lotfi Zadeh, joka kehitti sumeiden joukkojen teorian 1965. Sumean logiikan soveltaminen käytännössä kehittyi 1970-lu-vulla Mamdani-tyyppisten järjestelmien myötä. Niissä ideana oli käyttää sumeaa järjes-telmää kuvaamaan kokenutta järjestelmävalvojaa (ihmistä) ja kuvata hänen toimintaansa järjestelmän säädössä. Kokeneen ihmisen tietotaito järjestelmän ohjailussa kuvattiin su-meina päättelysääntöinä ja sumea järjestelmä linkitti numeerisen ja sanallisen maailman. Sumea logiikka ymmärrettiin alkujaan osaksi tekoälyä. Nykyisin kuitenkin sumeat jär-jestelmät nähdään enemmän perinteisinä säätöjärjestelminä, joissa sumeus edustaa funk-tioiden approksimaatioita. [20]

T.Takagi ja M.Sugeno esittelivät T-S sumean mallin vuonna 1985[16]. Siinä sumei-den sääntöjen seurausosan sumea joukko oli korvattu lineaarisella sisääntulojen funk-tiolla, mikä mahdollisti lineaaristen järjestelmien teorian soveltamisen sumeisiin järjes-telmiin kaikkine työkaluineen.

Tällä hetkellä sumeiden järjestelmien suosio on taas kasvamassa hyvän soveltamis-menestyksen siivittämänä. [17]

2.1 Keskeiset käsitteet ja perusteet

Sumean logiikan perusta rakentuu sumeille joukoille, joissa alkion kuulumista joukkoon ilmaistaan totuusarvolla $[0,1]$. Sumea totuusarvo voidaan nähdä niin, että se ilmaisee millä vahvuudella alkio kuuluu joukkoon. Tämä tekee sumeista joukoista erinomaisia kuvaamaan lingvistisiä muuttujia, esimerkiksi ”virtaus on kova”. Lingvistinen luonne tekee sumeasta logiikasta intuitiivisen tavan luokitella tai järjestellä asioita.[9]

Joukoista muodostetaan jäsenyysfunktioita, joiden perusteella alkion joukkoon kuuluminen määritellään. Avaruuden X alkion x jäsenyys ilmaistaan

$$\mu_A(x): X \rightarrow [0,1] \quad (1)$$

Tässä $\mu_A(x)$ on X -avaruudessa olevan sumean joukon A jäsenyysfunktio. Mitä lähempänä se on arvoa 1, sitä vahvemmin alkio x on joukon A jäsen. Joukko A voidaan ilmaista myös [7]

$$A = \{(x, \mu_A(x)); x \in X, \mu_A(x) \in [0,1]\} \quad (2)$$

Jäsenyysfunktioiden muodolla ja resoluutiolla vaikutetaan sumean järjestelmän käyttäytymiseen. Muoto voi olla esimerkiksi Gaussin käyrä, puolisuunnikas, kolmiomainen tai yksikköjoukko (singleton). Kaksi viimeksimainittua saattavat helpottaa laskentaa, mikä tekee niistä suositun valinnan. Järjestelmän jäsenyysfunktioiden on muodostettava aukoton kokonaisuus, jotta kaikki mahdolliset sisäänmenot saadaan sumeutettua ja sijoitettua sopivaan joukkoon. Funktion muodolla voidaan vaikuttaa siihen, miten tarkasti muuttujan on toteuduttava. Kolmionmallinen funktio määrittelee erittäin tarkan pisteen, kun taas vaikkapa puolisuunnikkaan muotoinen sopivan välin. Tästä esimerkkinä ”veden lämpötila on 20 astetta” ja ”veden lämpötila on sopiva”. Funktioiden tiuhempi resoluutio taas mahdollistaa lingvistisen muuttujan tarkemman kuvauksen, esimerkiksi ”veden lämpötila on *hieman* liian lämmin”. [9][7]

Selkeässä eli täsmällisessä muodossa oleva data sumeutetaan jäsenyysfunktioiden, mutta myös normalisoinnin ja suodatuksen kautta sopivaksi sumealle päättelyjärjestelmälle. Päättely suoritetaan annettujen IF-THEN -muotoisten sääntöjen pohjalta. Ne muodostavat sumean sääntökannan. Ehtoihin voidaan soveltaa matemaattisen logiikan toimintoja, kuten AND, OR tai NOT. Niiden toiminta ei kuitenkaan sumean totuusarvon ääripäitä lukuunottamatta vastaa perinteistä, vaan sumeaa logiikkaa. Sumea NOT määritellään sumean joukon A komplementtina A' :

$$\mu_{A'}(x) = 1 - \mu_A(x), \quad x \in X \quad (3)$$

Sumea unioni OR toteutetaan S-normilla, yleinen valinta on ”max”:

$$\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)] \quad \forall x \in X \quad (4)$$

Vastaavasti leikkaus AND usein T-normilla, joista yleinen valinta on ”min”:

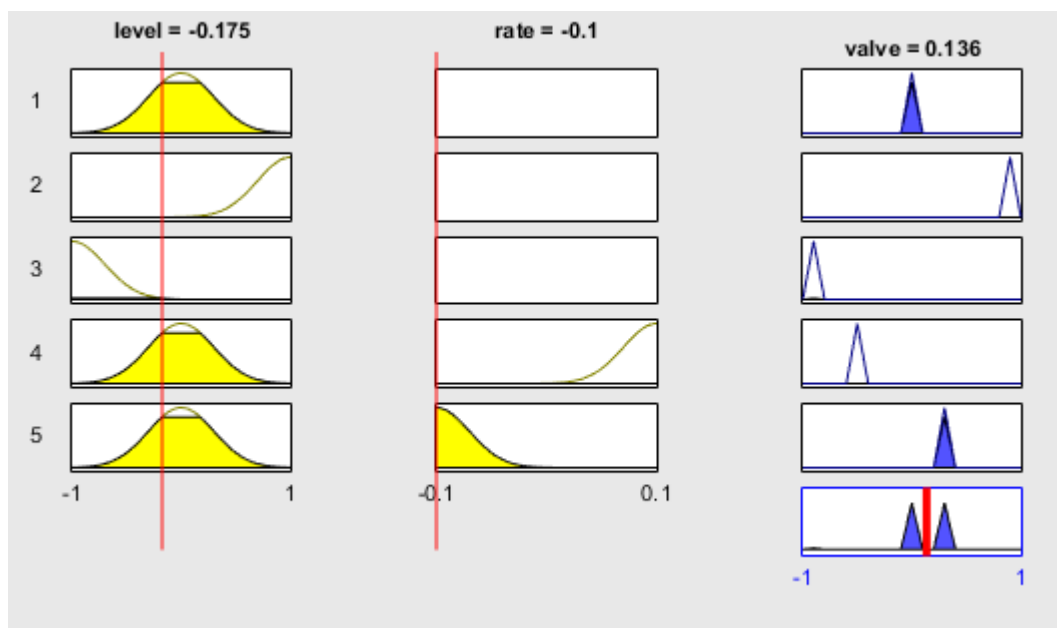
$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)] \quad \forall x \in X \quad (5)$$

Nämä perusoperaatiot olivat jo Zadeh'n 1960-luvun sumeiden joukkojen määrittelyssä mukana. [7]

Päätelyn jälkeen seurausosuuden toiminta riippuu sumean järjestelmän tyypistä. Kaksi tavallisinta järjestelmätyyppiä ovat Mamdani ja Takagi-Sugeno (T-S). Näistä Mamdani-tyyppiset järjestelmät palauttavat sumean päätelyn tuloksena sumean joukon, joka pitää selkeyttää. T-S -järjestelmät toteuttavat jonkin funktion päätelyn seurauksena, ja sen tulos on valmiiksi selkeytetty. Tämä on T-S -järjestelmän toiminnan sisäänrakennettu ominaisuus.[9][20]

Mamdani-järjestelmien selkeytysmetodi valitaan halutun toiminnan mukaan. Mikäli tavoitteena on esimerkiksi säätää epälineaarista järjestelmää, painopisteselkeytys (CoG, center of gravity) tuottaa sopivan approksimaation ulostulosta säädön käyttöön. Jos tarvitaan edellämainitun asemesta jonkin tietyn jäsenyysfunktion liipaisema tulos, on käytössä yleensä MoM-selkeytys (mean of maximum). Tämä sopii vaikkapa sumeaaan vianetsintään tai muihin ei-interpoloiiviin tunnistustehtäviin. [9][20]

Sumeaa päätelyä on havainnollistettu kuvassa 1, joka on Matlabin sumeiden järjestelmien päätelyä esittävästä *fuzzy rule viewer* -työkalusta. Vasemmalla ja keskellä on esimerkkijärjestelmän suureita, joiden avulla ohjataan sumeasti venttiiliä tankkijärjestelmässä. Suureiden tilaa esitetään punaisin pystyviivoin. Vaakariveillä 1-5 on sumeita sääntöjä, joiden voimassaoloa ja voimassaolon astetta tutkitaan jäsenyysfunktioiden avulla. Näistä säännöistä saadaan muodostettua seurauksia, jotka esitetään kuvan oikeassa reunassa. Oikeanpuoleisen rivin alareunassa seuraukset yhdistetään, jolloin venttiilin ulostulo selkeytyy. Selkeytetty ulostulo on merkitty paksulla punaisella pystyviivalla.



Kuva 1: Esimerkki sumeasta päätelystä

2.1.1 Sääntökanta

Sääntökanta liittyy sumeaan päätöksentekoon. Se muodostuu päättelysäännöistä, jotka astuvat voimaan tietyn ehdon ollessa voimassa, ja niiden seurauksista. Karray et al [7] luettelevat viisi ehtoa, jotka sääntökannan tulisi täyttää: Sääntökannan on oltava kokonainen, jatkuva ja johdonmukainen. Lisäksi säännöt eivät saa käyttää toisiaan (interact) ja kannan on johdettava robustiin ja vakaaseen toimintaan.

Kokonainen (complete) sääntökanta tuottaa päättelyn kaikissa tilanteissa. Kaikki sumean järjestelmän mahdolliset sisäänmenot tuottavat siis jonkin tuloksen, eikä aukkoja päättelyssä ole. Tämä tapahtuu niin, että minkä tahansa sisäänmenon on liipaistava ainakin yksi sääntö eli jonkin jäsenyysfunktion arvo poikkeaa nollassa.

Jäsenyysfunktiot on muotoiltava niin, ettei aukkoja päättelyyn tule. Tämä varmistaa päättelyn jatkuvuuden. Johdonmukaisuudella tarkoitetaan sitä, etteivät säännöt taistele toisiaan vastaan loogisesti.

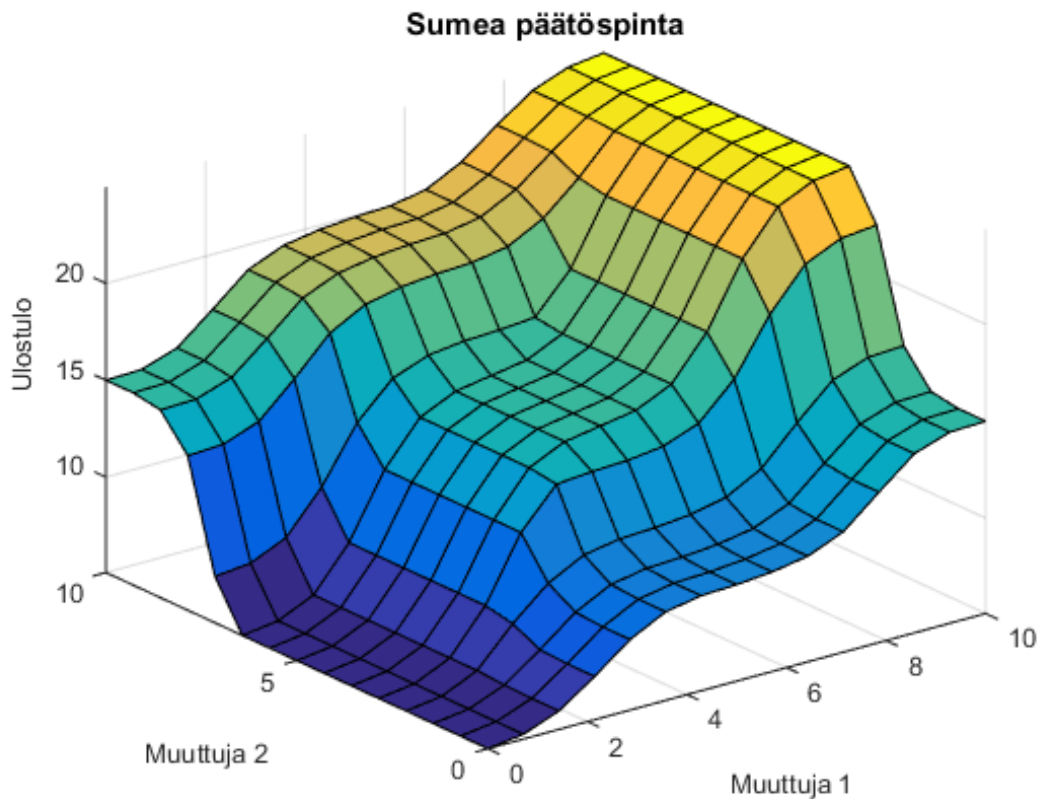
Jos sääntökannan säännöt käyttävät toisiaan tuloksen muodostamiseen, voi päätöksenteosta tulla hitaampaa ja monimutkaista. Ehdoton kielto tämä siis ei ole, ja toiminta voi olla joissain tapauksissa tarkoituksenmukaista[7].

2.2 Sumeat järjestelmätyypit

Sumea logiikka on ollut teollisuuden käytössä hyvin varhain keksinnön ikään nähden. Sumea säätö sopii parhaiten monimutkaisten, epälineaaristen, hankalasti mallinnettavien järjestelmien kanssa käytettäväksi [9]. Sumeiden järjestelmien soveltaminen on ollut Japanissa ja muualla Aasiassa suosittumpaa kuin länsimaissa [7]. Syyksi tähän arvellaan [7] muun muassa sanaan sumea (fuzzy) länsimaissa yhdistettyä huonoa mielikuvaa. Suosituimmat sumean järjestelmän rakennetyypit ovat kehittäjiensä mukaan nimetyt Mamdani ja Takagi-Sugeno. Näistä ensin mainittu rakenne on hieman vanhempi.

2.2.1 Mamdani-järjestelmät

Mamdani-tyyppiset järjestelmät ovat niitä, joiksi sumeat järjestelmät yleensä käsitetään asiaan vihkiytymättömien mielessä. Kuten aiemmin todettiin, Mamdani-järjestelmät kehittyivät sumean logiikan käytön alkuaikoina. Niiden toiminta kuvaa ihmisoperaattoria ainakin periaatetasolla: operoidaan suuretta X , tapahtuu Y . Näistä toisiinsa liittyvistä tapahtumista saadaan kehitettyä kuvan 2 esimerkin kaltainen sumea päätöspinta, joka kuvaa tapahtumien epälineaaristakin yhteyttä. Esimerkkikuvassa sumean järjestelmän päättelyyn vaikuttaa kaksi suuretta (ehtomuuttujaa). Automaattisena tietokonepohjaisena järjestelmänä sumea järjestelmä on tietenkin massiivisesti ihmiskäyttäjää nopeampi ja tarkempi, ja mahdollisti aikanaan monien hankalasti mallinnettavien järjestelmien automatisoinnin[7]. Järjestelmä koostuu kuvan 3 mukaisesti kolmesta osasta: sumeutus, sumea päättely ja selkeytys.



Kuva 2: Esimerkki kahden ehtomuuttujan sumeasta päätöspinnasta.

Sumeutusosaan tulevat selkeät sisäänmenot ja niille määritellään jäsenyysfunktioiden perusteella jäsenyysaste. Tuloksena on sumean joukon alkio, jolla on paitsi arvo, myös joukkoon kuulumisen vahvuuden kertova jäsenyysfunktion arvo.

Sumeutettuja sisäänmenoja käsitellään päättelyosassa. Se sisältää sääntökannan, joka kertoo IF-THEN tyyliisesti seurauksen voimassa olevalle säännölle R_i kaavojen (7) ja (8) mukaisesti tai yleisemmin [9]:

$$R_i: \text{ IF } \tilde{x} \text{ is } \tilde{P}, \text{ THEN } \tilde{y} \text{ is } \tilde{Q} \quad (6)$$

Tässä \tilde{x} on lingvistinen muuttuja X -avaruudessa, \tilde{P} on sumean joukon P lingvistinen arvo X -avaruudessa, \tilde{y} Y -avaruuden lingvistinen muuttuja ja \tilde{Q} Y -avaruudessa olevan sumean joukon Q lingvistinen arvo. Säännöllä on siis ehto-osa ja seurausosa. Kumpikin osa voi sisältää useita osia, toisin sanoen monella yhtäaikaaisella ehdolla voi olla yksi seuraus ja yhdellä ehdolla monta seurausta. Sääntöjen seuraukset tuottavat sumeita arvoja, jotka on selkeytettävä käyttökelpoiseksi ulostuloksi. Seurauksien yhdistelyyn ja jäsenyysarvon ratkaisemiseen käytettäviä operaatioita on esitelty aiemmin luvus-

sa 2.1. Niiden (T-normit) valinta ei ole yksiselittäistä ja johtaa hieman erilaisiin lopputuloksiin. [9]

Seurausosista saatujen sumeiden arvojen selkeytykseen on niiden tyypistä (jäsenyysfunktion muoto) riippuen erilaisia menetelmiä. Yleisimmät menetelmät ovat painopistiselkeytys (COG, Center of Gravity) ja keskipiste (CA, Center Average) [9]. Painopistiselkeytyksessä lopputuloksen määrää liipaistujen sääntöjen seurausten jäsenyysfunktioiden muodostaman pinta-alan keskipiste. Vastaavasti keskipiste- tai keskiarvoselkeytyksessä lasketaan sumeista joukoista keskiarvo. Jälkimmäinen tapa on varsinkin singleton-tyyppisten seurausten kanssa käyttökelpoinen ja helpottaa laskentaa.[9][7]

2.2.2 Takagi-Sugeno -järjestelmät

Takagi-Sugeno -tyyppisten sumeiden järjestelmien tulo 1980-luvulla [16] mahdollisti perinteisten säätömenetelmien käytön vaikeissa, epälineaarisisissa prosesseissa, joiden tarkka malli on tuntematon [9]. Tämä perustuu rakenteeseen, jossa sumean päättelyn jälkeinen seurausosuus toteuttaa jonkin, yleensä järjestelmän sisäänmenojen, funktion. Järjestelmästä kerätystä datasta voidaan sovittaa T-S -tyyppinen malli. Rakenne sisältää samalla selkeytyksen, jota ei siis tarvitse tehdä enää erikseen.

Sumean järjestelmän IF-THEN -tyyppiset päättelysäännöt muodostavat edelleen järjestelmän epälineaarisen kuvauksen. T-S -järjestelmissä sääntöjen muodostamaa lokaalia dynamiikkaa kuvaa lineaarinen malli. Epälineaarisuus muodostuu lineaaristen mallien yhdistelmästä, Tanaka et.al.[17] kuvauksen mukaan sumeana sekoituksena. T-S -tyyppiset sumeat mallit ovat yleisapproksimaattoreita.[17]

Jatkuva-aikaiselle järjestelmälle säännöt ja niiden seuraukset voidaan esittää[17]:

$$\begin{aligned} \text{IF } z_1(t) \text{ is } M_{i1} \text{ AND } \cdots \text{ AND } z_p(t) \text{ is } M_{ip}, \\ \text{THEN } \begin{cases} \dot{x}_i(t) = A_i x(t) + B_i u(t), & i=1,2,\dots,r \\ y_i(t) = C_i x(t), \end{cases} \end{aligned} \quad (7)$$

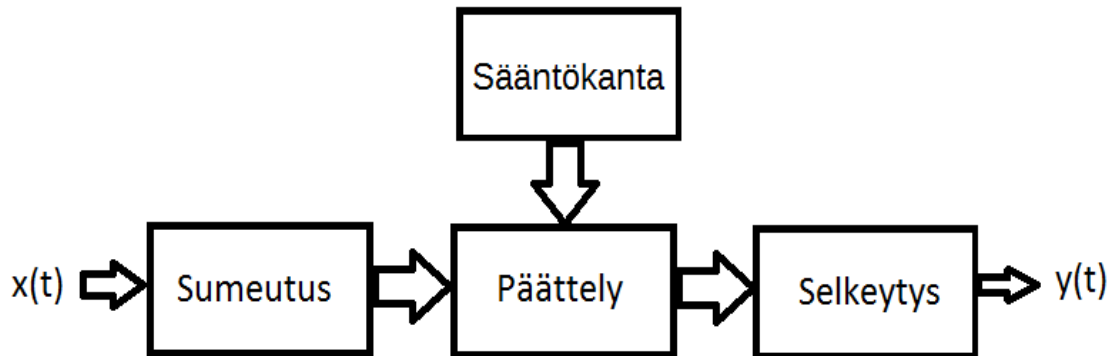
Ja vastaavasti diskreettiaikaiselle sumealle järjestelmälle:

$$\begin{aligned} \text{IF } z_1(k) \text{ is } M_{i1} \text{ AND } \cdots \text{ AND } z_p(k) \text{ is } M_{ip}, \\ \text{THEN } \begin{cases} x_i(k+1) = A_i x(k) + B_i u(k), & i=1,2,\dots,r \\ y_i(k) = C_i x(k), \end{cases} \end{aligned} \quad (8)$$

Esityksessä M_{ij} on sumea joukko ja r sääntöjen lukumäärä. Vektorit x , u ja y esittävät järjestelmän tilaa, sisäänmenoja ja ulostuloja. Matriisit A_i , B_i ja C_i kuvaavat järjestelmää muodostaen lokaaleja lineaarisia kuvauksia. Ehtolausekkeen muuttujat z_i eivät saa olla sisäänmenovektorin $u(t)$ funktioita selkeytyksen vaikeutumisen takia [17]. Yleensä

ehtolausekkeen muuttujat ovat tilojen, häiriöiden tai ajan funktioita [17]. Muuttujat on esitetty tässä merkinnällä z_i , jotta esitetty kaava kytkeytyy selkeämmin local sector non-linearity -menetelmään kappaleessa 4.1.

Järjestelmän lopullinen ulostulo lasketaan kaavan (23) avulla painotettuna keskiarvona. Tässä diskreetin järjestelmän kaavaa ei ole esitetty erikseen, mutta se on helppo päätellä jatkuva-aikaisen järjestelmän kaavasta.



Kuva 3: Sumean järjestelmän osat

2.2.3 Mamdani versus T-S

Toisin kuin Mamdani-järjestelmissä, T-S -järjestelmissä säännön seuraus ei ole sumea joukko, vaan matemaattinen, tyypillisesti järjestelmän sisäänmenojen, funktio. Tämä mahdollistaa muun muassa adaptiivisen säädön ja suljetun järjestelmän stabiiliuden todistamisen [9], mutta käyttö ei ole samalla lailla intuitiivista kuin Mamdani-tyyppisissä sumeissa järjestelmissä [9]. Mamdani-järjestelmien voidaan osoittaa olevan T-S sumean järjestelmän erityistapaus [9]. Nykyaikaisten lineaarisen säädön työkalujen käyttö on mahdollista T-S -järjestelmien kanssa, mikä onkin suuri motivaatiotekijä niiden käytössä.

Jos säädettävästä järjestelmästä ei ole olemassa kunnollista mallia, mutta sen toiminnasta voidaan muodostaa lingvistisiä sääntöjä, on sille mahdollista tehdä Mamdani-tyyppinen sumea säädin. Tämä ominaisuus muodostaa erään parhaista syistä käyttää sumeita järjestelmiä. Niitä käytetäänkin varsinkin epälineaaristen vaikeasti mallinnettavien tai tuntemattomien kohteiden takaisinkytketyssä säädössä. [9][7]

2.3 Sumea säätö

Sumean säädön käytön motiivina on monesti sen tuoma mahdollisuus epälineaariseen säätöön. Sumealla säätimellä saadaan muodostettua epälineaarinen kuvaus sisäänmenojen ja ulostulojen välille (kuten esimerkiksi kuvassa 2). Sumean logiikan käyttö säädössä keskittyi aluksi kuvastamaan kokenutta ihmisoperaattoria. Vaikeutena sekä alkuaikoina että edelleen nykyäänkin on paitsi suunnittelu, myös virittäminen, jotka molemmat

joudutaan tekemään useimmiten hankalasti käsityönä. Yleispäteviä viritysmenetelmiä ei ole, vaikkakin niitä on kyllä yritetty kehittää [21]. Sumean järjestelmän stabiilius ei myöskään aina ole helposti todettavissa.[9]

Sumeaa säätöä voidaan tehdä usealla eri tasolla. Sumea säädin voi toimia alhaisella säätötasolla, kuten tämän työn heilurin suorassa säädössä, tai korkeamman tason säätimenä joko alhaisen tason säätimien parametrien optimoinnissa (automaattinen viritys), virheiden, suorituskyvyn ja vikatilanteiden tarkkailussa tai vielä korkeammalla tasolla. Tästä esimerkkinä vaikkapa tuotannonohjaus.

Sumean säädön avulla saadaan hankalat ja monimutkaiset prosessit käsiteltyä pienempinä, helpommin käsiteltävinä palasina. Se myös yhdistää palaset takaisin kokonaisuudeksi. Näin ollen sumea säätö on hyvä hajoita ja hallitse -tyyppinen ratkaisu säätöongelmiin. Jos säädössä käytetään tässäkin työssä käytettyä T-S parallel distributed -tyyppistä sumeaa säädintä, voidaan säädössä hyödyntää tuttuja lineaarisia työkaluja ja menetelmiä.[9] [17]

2.3.1 Parallel distributed control (PDC)

T-S -tyyppisen sumean mallin kanssa voidaan käyttää parallel distributed -säätöä, missä jokaiselle mallin sääntökannan säännölle, ja sitä myöten lokaalille mallille, luodaan oma säädin. Tuloksena on epälineaarista järjestelmää approksimoiva lineaaristen mallien keskiarvon keskiarvoinen säädin. Tätä menetelmää käytetään tässä työssä Furuta-heilurin mallintamiseen ja tasapainosäätöön.[9][17]

Säädettävän järjestelmän sumea T-S -malli interpoloi funktioiden (lokaalien mallien) välillä. Mallin rakentaminen ja toiminta on esitetty tarkemmin kappaleessa 4.1. Saadulla sumealla mallilla on mahdollista esittää järjestelmän epälineaarisuus lineaaristen lokaalien mallien avulla. Jokaiselle lokaalille mallille luodaan oma säädin, jotka yhdessä muodostavat toisen T-S -mallin säätimelle. Tämä T-S -mallinen sumea säädin tuottaa lineaarisilla menetelmillä epälineaarisen säätöratkaisun. Säätömenetelmiä on useita, yhtenä esimerkkinä tavallinen napojen asettelu. Tässä työssä käytetään LQR-säätöä, missä optimaalinen tilatakaisinkytkentä muodostetaan ratkaisemalla neliöllisen kustannusfunktion minimi (quadratic cost function) järjestelmän mallin suhteen. LQR-säätimen optimiratkaisu on olemassa vain, jos malli on stabiili. Saatu ratkaisu tuottaa myöskin stabiilin suljetun lokaalin järjestelmän. Kustannusfunktio on muotoa

$$J(u) = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad \text{järjestelmälle} \quad \dot{x} = Ax + Bu .$$

Kustannusfunktiossa Q ja R ovat tilojen x ja sisäänmenojen u painomatriisit. Ohjelmistotyökaluilla saadaan ratkaistua pienimmän kustannuksen aiheuttava tilatakaisinkytkennän vahvistusmatriisi K , joka lasketaan

$$K = R^{-1} B^T S ,$$

missä S on tasapainotilan ratkaisu Riccatin yhtälöön

$$A^T S + SA - SBR^{-1}B^T S + Q = 0$$

ja A ja B ovat järjestelmän kuvaus.

Sumeiden järjestelmien stabiiliuden tarkasteleminen ei ole aina ongelmaton. Parallel distributed -menetelmän T-S mallisen järjestelmän globaali stabiilius voidaan kuitenkin todeta, jos löydetään yksi matriisi G , joka toteuttaa kaikille i ja j yhdistelmille

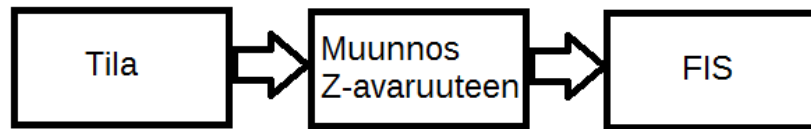
$$G({}^i A - {}^i B {}^j K) + ({}^i A - {}^i B {}^j K)^T G < 0 .$$

Kappaleen 4.1 mukaan järjestelmä jaetaan kuuteentoista lokaaliin malliin ja säätimien. Tämän LMI-ongelman (linear matrix inequality) ratkaisemiseen tarvitaan siis tutkittavan järjestelmän tapauksessa 16 lokaalille järjestelmälle (${}^i A$ ja ${}^i B$) ja 16 takaisinkytkentämatriisille (${}^j K$) yhteensä 256 epäyhtälöä. [9]

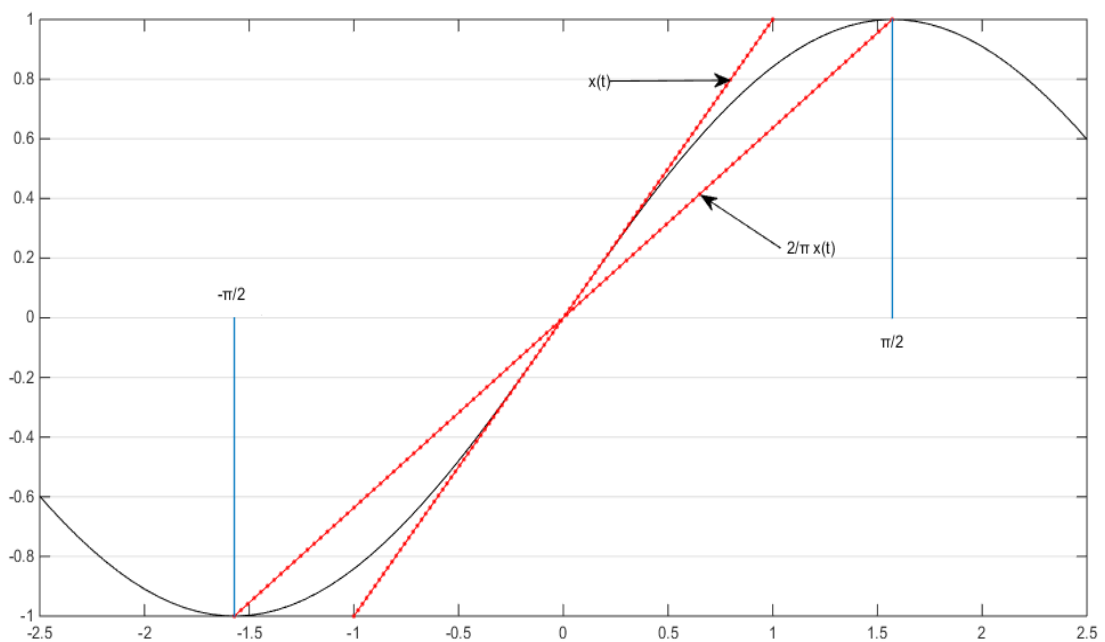
2.3.2 Local sector nonlinearity

Vuonna 1992 esitelty sector nonlinearity -konsepti [8] perustuu järjestelmien epälineaaristen osien sumeaan tulkkaukseen. Se syntyi sumeiden järjestelmien stabiiliustarkastelemissa sivutuotteena. Epälineariselle järjestelmälle $\dot{x} = f(x(t))$, $f(0) = 0$ yritetään löytää globaali sektori $\dot{x} = f(x(t))$, $x(t) \in [a_1 a_2]$, jolloin sumea T-S -järjestelmä vastaa epälineaarista järjestelmää. Globaalin sektorin löytämisen hankaluuden vuoksi on usein tarpeen käyttää lokaalia vastinetta. Koska fyysisillä järjestelmillä on aina rajansa, tämä on mahdollista. Luodaan siis pienempi lokaali alue $-d \dots d$ ja $-d < x(t) < d$. Sumea järjestelmä vastaa epälineaarista tällä pienemmällä alueella. Vaikeudeksi jää vain sopivan alueen löytäminen, mutta alueen löytäminen ja rajaaminen on kuitenkin mahdollista ja sitä voidaan käyttää säätösuunnittelun lähtökohtana [8]. Kuvassa 4 esitetään menetelmän kytkeytyminen parallel distributed -säätöön. Sektoreiden muodostama Z-avaruus määrittää mahdollisen säätöavaruuden. Sijainti avaruudessa lasketaan järjestelmän tiloista ja syötetään sumealle päättelyjärjestelmälle sisäänmenona. [17]

Esimerkkifunktio $\sin(x)$ ja tämän lokaali sektori alueella $-\frac{\pi}{2} < x(t) < \frac{\pi}{2}$ on esitetty kuvassa 5. Tästä on helppo havaita sektorin suurin mahdollinen yksikäsitteinen alue. Sektoria rajaavat suorat ovat $x(t)$ ja $\frac{2}{\pi} x(t)$.



Kuva 4: Järjestelmän tiloista lasketaan sijainti säätöavaruudessa Z , minkä jälkeen sumealla päättelyllä valitaan sopiva säädin (PDC)



Kuva 5: $\sin(x)$:n lokaali sektori esitettynä punaisilla katkoviivoilla. Kuva perustuu [17]

Local sector nonlinearity -periaatetta on käytetty tässä työssä järjestelmän sumean T-S mallin rakentamisessa luvussa 4, mikä toimii samalla selventävänä esimerkkinä. Myös säädin hyödyntää samaa periaatetta. Menetelmän etuna on sumean sääntökannan mahdollisesti rajukin pieneneminen varsinkin järjestelmillä, joissa on monta säädettävää tilaa. Mikäli sopiva toiminta-alue löytyy, on lokaalimallien valinta helppoa verrattuna esimerkiksi sumeiden alueiden (partition spaces) menetelmään nähden. Toiminta-alueen rajaamisesta ovat kirjoittaneet muun muassa Ohtake et al. [12]. Heidän tuloksensa sumean mallin ja epälineaarisen mallin vastaavuuden välillä ovat erittäin hyviä. Sääntökannan pienentämisen periaate ja siten sumean järjestelmän monimutkaisuuden raju vähentäminen on esitetty Taniguchi et al. [18]. [17]

2.3.3 Viritys

Kuten edellä mainittiin, yleispäteviä viritysmenetelmiä ei ole. Sumean säätäjän viritys tapahtuu monesti yritys-erehdys -menetelmällä ja voi olla hyvinkin työlästä. Säädettäviä parametreja on paljon ja järjestelmän rakenteessa on vaihtoehtoisia matemaattisia mene-

telmiä. Lisäksi säädettävät prosessit eivät ole aina täysin tunnettuja. Viritystä on kuitenkin tutkittu, ja esimerkiksi Viljamaa [21] yrittää rakentaa systemaattisen viritysmenetelmän käyttäen prosessista tiedettyjä ominaisuuksia viritysparametrien perustana.

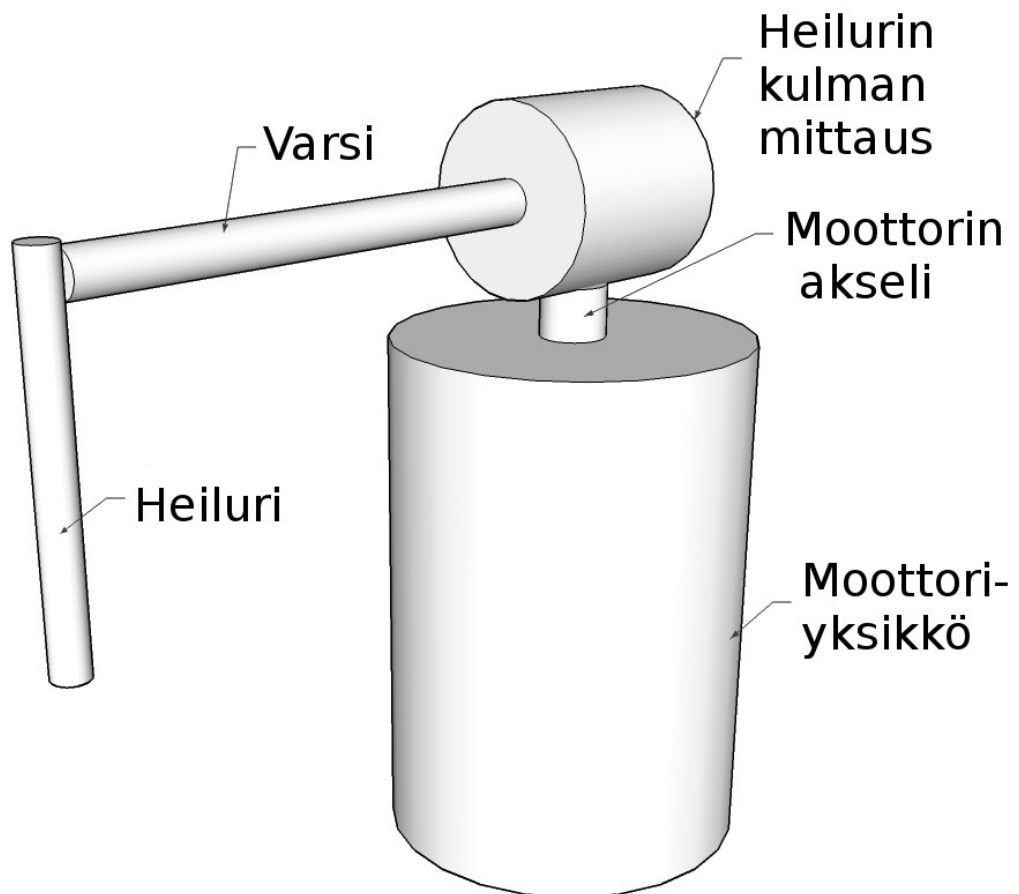
Mikäli sumea säädin korvaa tavallisen PID-säätimen, on virittämisen lähtökohdaksi mahdollista käyttää vanhan PID-säätimen hyväksi havaittuja parametreja. Tämän voi tehdä tietyn tyyppisellä sumealla säätimellä, joka luodaan PID-säätimen pohjalta sitä täysin vastaavaksi [9].

Jos malli on tunnettu ja halutaan rakentaa aikajatkuva säätöjärjestelmä, voidaan säätösuunnitella myös järjestelmän luonnollisen taajuuden kautta ja ikään kuin pienemmän kertaluvun järjestelmänä käyttämällä järjestelmän dominoivia napoja suunnittelun lähtökohtana.[19]

3 KOEJÄRJESTELY

Kokeet perustuvat kanadalaisen Quanserin opetuskäyttöön suunniteltuun QUBE-laitteistoon ja sen rotary inverted pendulum -lisäosaan. Siihen sisältyy kuution muotoiseen koteloon paketoitu moottoriyksikkö ja moottorin akselille varren päähän kiinnitetty Furuta-tyyppinen kääntöheiluri. Jatkossa laitteistoon viitataan Furuta-heilurina. Datan keruuseen ja laitteiston ohjaukseen on erillinen laitteisto. Paketti on suunniteltu alunperinkin opetuskäyttöön, muun muassa säätötekniikan ja servomoottoreiden esittelyyn.[14]

Heilurin rakennetta hahmotetaan kuvissa 7 ja 6 ja QUBE-laitteiston teknisiä ja fyysisiä ominaisuuksia Taulukossa 1.

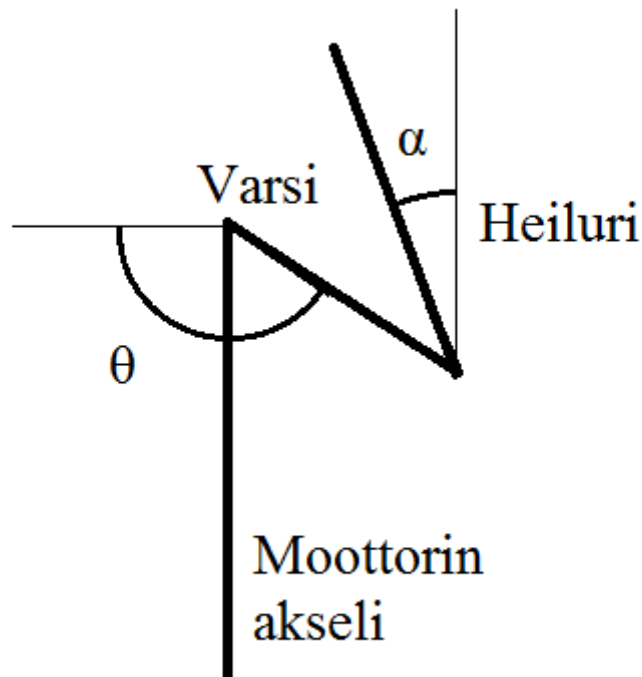


Kuva 6: Furuta-kääntöheilurin rakenne

3.1 Laitteet ja prosessi

Rotating inverted pendulum, tai kääntöheiluri, kuuluu säätötekniikan klassikkopulmiin. Keksijänsä Katsuhisa Furutan mukaan sitä nimitetään myös Furuta-heiluriksi. Se kehitettiin Tokiossa Tokyo Institute of Technologyssa vuonna 1992 [2]. Heiluri on epälineaarinen järjestelmä ja laitteisto muodostaa rajoitteita muun muassa moottoriyksikön varren nopeudelle ja maksimiliikkeelle. Pulmalla on käytännön sovelluksia esimerkiksi avaruusraketin asennon säädössä ja robotiikassa. Haastavuus ja monipuolisuus tekee kääntöheilurista suositun tutkimuskohteen.

Heilurin säädössä on kaksi vaihetta, heilurin heilautus ylös ja sen tasapainottelu. Näitä voidaan tutkia myös erikseen ja monesti säätötoteuksissa vaiheita hoitavatkin erilliset säätimet, jolloin toinen heilauttaa heilurin tarpeeksi ylös ja tästä tasapainottava säädin ottaa ohjat ja hienosäätää heilurin epävakaiseen tasapainotilaansa yläasentoon. Furuta-heilurilla voidaan opetustilanteessa demonstroida sekä lineaarista että epälineaarista säätöä. Säätökohteena Furuta-heilurin mielenkiintoa lisää myös se, että sillä on enemmän vapausasteita kuin ohjausmuuttujia (*underactuated*).



Kuva 7: Furuta-heilurin periaatekuva

3.1.1 Quanser QUBE

Qube on kuution muotoiseen metallikuoreen paketoitu moottoriyksikkö. Se sisältää sähkömoottorin, moottorin asennon mittauksen, PWM-vahvistimen ja tarvittavat liitännät anturien lukemiseen ja moottorin ohjaamiseen. Moottorin akselille on kiinnitetty heilurin kulman mittaustaite ja varsi, jonka päässä heiluri pääsee kääntymään vapaasti. Quben malli on kolmen laitteen valikoimasta Direct I/O, ja se vaatii parikseen erillisen säätimen ja tiedonkeruuyksikön. Moottoriyksikön voiman luo Allied Motionin CL40-sarjan 16705-mallinen tasavirtamoottori. Moottori tuottaa 22 mNm nimellisvääntömomentin ja 7W maksimitehon. Rakenteeltaan se on harjallinen ja inertia on valmistajan mukaan pyritty minimoimaan ($4 \times 10^{-6} \text{ kgm}^2$). [14]

Kääntöheilurin kulman mittaamiseen laitteistossa on US Digitalin E8P optinen anturi. Toiminta perustuu pyörivän levyn merkintöjen optiseen havaitsemiseen. Se tuottaa ulostulona pulseja anturin pyörimissuunnan ja -nopeuden mukaan, joten varsinainen haluttu kulman arvo on laskettava pulsseista. Anturin käyttöresoluutio on 1/512 osaa kierroksesta, eli se havaitsee noin 0,7 asteen muutoksen. Prosessi palautuu aina samaan tasapainotilaan ilman ohjausta, joten anturin pulseille saadaan sopiva referenssipiste. [14]

3.1.2 National Instruments CompactRIO

Kirjoitushetkellä tiedonkeruulaitteistona ja säätimenä on National Instruments CompactRIO. Se on modulaarinen itsenäinen säädin, joka voidaan ohjelmoida Labview-ohjelmiston avulla. CompactRIOssa on I/O-moduuli, josta löytyvät sopivat liitännät Quanser Quben anturitietojen lukemiseen ja moottorin ohjaamiseen. Laitetta on myyty pakettina Quben kanssa. Yhteys CompactRIOsta eteenpäin PC-koneelle hoidetaan Ethernet-väylän yli.

Toistaiseksi ei ole selvää, toteutetaanko opiskelijoiden laboratorioharjoitukset CompactRIO:n kanssa, vai korvataanko se uudemmalla tai edullisemmalla vastaavalla säätimellä. Qube-heilurilaitteistosta on myös suoraan USB2-väylän yli toimiva versio, joka kytkeytyy ilman välilaitteita PC-koneeseen. Tällöin tosin menetetään Ethernet-väylän kätevä mahdollisuus puuttua laboratoriolaitteiden toimintaan myös etäämpää. Kuvassa 8 vasemmalla on CompactRIO:n säädinosa cRIO-9024 ja oikealla Quanserin I/O-moduuli. PC-tietokone on liitetty laitteisiin ethernet-väylän yli.

Taulukko 1: Quanser QUBEn ominaisuudet ja parametrit

Parametri	Lyhenne	Arvo
Varren pituus	L_r	0,085 m
Varren massa	m_r	0,095 kg
Varren inertia	J_r	$5,7198 \cdot 10^{-5} \text{ kgm}^2$
Varren viskoosinen vaimennuskerroin	D_r	0,0015 Nm s/rad
Varren (moottorin akselin) kulma	θ	Kasvaa vastapäivään
Heilurin pituus	L_p	0,129 m
Heilurin massa	m_p	0,024 kg
Heilurin inertia	J_p	$3,3282 \cdot 10^{-5} \text{ kgm}^2$
Heilurin viskoosinen vaimennuskerroin	D_p	0,0005 Nm s/rad
Heilurin kulma	α	0-piste ylhäällä, kulma kasvaa vastapäivään
Vastasähkömotorinen voima (Back-EMF)	k_m	0,042 Vs/rad
Moottorin ominaisvääntö	k_t	0,042 Nm/A
Moottorin resistanssi	R_m	8,4 Ω



Kuva 8: National Instruments CompactRIO ja Quanser I/O-moduuli

3.2 Ohjelmistot

Laitteiston valmistaja Quanser tarjoaa sitä käytettäväksi Matlab/Simulink- tai Labview-ohjelmistojen kanssa. Työssä käytetään Mathworksin Matlabia ja sen Simulink-lisäosaa. Matlab-käyttöön tarvitaan sen osista Symbolic Toolbox, Control System Toolbox ja Robust Control Toolbox. Valmistajan käyttämää Matlab-versiota ei Quanserin tarjoamassa Courseware-materiaalissa kerrota, mutta työssä käytettiin versiota R2014b. Työssä käytettyjä Matlab-koodeja ja sumeiden järjestelmien FIS-rakenteet on esitetty työn Liitteet-osassa.

Laitteistovalmistaja tukee siis sekä Labview- että Matlab-ympäristöjä. Kuitenkin jos laitteisto on tutkitun kaltainen National Instrumentsin CompactRIO:n sisältävä yhdistelmä, painopiste siirtyy vahvasti Labview'n puolelle. Tämä käy ilmi esimerkiksi National Instrumentsin verkkosivuilta tai Quanserin courseware-materiaalin virheestä. Eräs Quanserin Matlab-laboratoriotyön tiedostopaketti sisältää vain Labview-muotoisia tiedostoja.

Labview'n etuna tässä on mahdollisuus ajaa ilman PC-konetta itsenäisesti CompactRIO:ssa Labview'lla tehtyä säädintä. TTY:llä käytössä on ensisijaisesti Matlab ja Simulink, joten tämä ei ole olennaista. Sen sijaan laitteiston käyttö verkon yli on tarpeellinen ominaisuus. Quanserin tuote tähän on QUARC, joka luo yhteyden Simulinkin ja laitteiston välille. Se toimii muun muassa TCP/IP:n ja UDP:n yli, eikä Quben USB-version kanssa tarvitse CompactRIOa, vaan PC Simulinkilla varustettuna riittää. [14]

Mikäli CompactRIO:n itsenäistä toimintaa halutaan hyödyntää, voidaan Simulink-mallit tulkata Labview'n Model Interface Toolkitin avulla Labview-muotoon. Muunnos vaatii tietyt ehdot täyttävän mallin. National Instrumentsin materiaalien perusteella on epäselvää, selviytyykö tulkkaus esimerkiksi Simulinkin sumeasta lohkokosta. Jos ei, niin sumean toiminnon voisi mahdollisesti tehdä erillisenä Labview-sovelluksena. Tämän menetelmän kokeilu jää tulevien oppinäytetöiden tekijöiden tehtäväksi. [11] [10]

Jos Simulinkia ei voida tai haluta käyttää ollenkaan, on mahdollista toteuttaa sumeiden lohkojen toiminta myös ilman sitä. Työssä käytetyssä sumeassa järjestelmässä laskutoimitukset eivät ole ylivoimaisen monimutkaisia toteutettaviksi komentosarjana. Tämä mahdollistaisi Matlab-koodin käytön Labview-ympäristössä Labview'n MathScript-lisäosan avulla. National Instruments tarjoaa myös mahdollisuuden OPC:n käyttöön, mikä sopii automaatiotekniikan opetustarpeisiin. [11] [10]

Quanser QUARCin toiminnot ovat opetuskäyttöön ylimitoitettuja, sillä se on tehty käytettäväksi muun muassa teollisuusrobottien kanssa ja ammattimaisessa prototyyppitestauksessa [14]. Simulinkiin on mahdollista tehdä kevyempi verkon yli UDP:lla tai TCP/IP:llä laboratoriolaitteiston kanssa keskusteleva osa. TTY:n automaatio- ja säätötekniikan laitoksella on aiemmin toimittu tällaisen projektin parissa.

4 JÄRJESTELMÄN MALLI

Furuta-heiluri on laajasti tutkittu järjestelmä. Lähdeaineistoja etsiessä törmää kuitenkin monesti siihen, että mallin johtaminen on jätetty pois ja tutkimus nojaa vahvasti jonkun muun johtamaan malliin. Tämä voi johtaa kertautuviin virheisiin, mutta toisaalta on otettava huomioon, ettei pyörän keksiminen uudestaan ole aina tarkoituksenmukaista. Mallia ovat tutkineet muun muassa Cazzolato ja Prime (2011) [2], jotka ovat myös löytäneet muiden tekemistä yhtälöistä virheitä. Demolaitteiston valmistaja Quanser käyttää maineikkaan ruotsalaisen Lundin yliopiston K.J.Åströmin yhtälöitä [25]. Åström on tehnyt myös Furuta-heilurin keksijän kanssa tutkimuksia ja julkaisuja. Mallia on tutkinut myös Gäfvert (1998) [6] Lundissa.

Yleisesti järjestelmän dynamiikkaa voidaan kuvata tilaesityksenä järjestelmämatriisien A, B, C , ja D avulla:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (9)$$

Vektori x sisältää järjestelmän tilamuuttujat, joita ovat heilurin tapauksessa luonnollisesti (vertaa kuva 7) heilurin kulma α , varren kulma θ , sekä näiden aikaderivaatat eli kulmanopeudet. Näistä kulmat voidaan mitata, mutta nopeudet täytyy laskea. Nopeuksien laskenta on työssä toteutettu heilurin laitevalmistajan käyttämällä tavalla yksinkertaisena suodattimena.

$$\begin{aligned} x^T &= [\theta \quad \alpha \quad \dot{\theta} \quad \dot{\alpha}] \\ y^T &= [x_1 \quad x_2] \end{aligned} \quad (10)$$

Tilaesitykseen pääsemiseksi laitteiston valmistaja on päätenyt mallintamaan heilurin Euler-Lagrange -menetelmän avulla [14] ja linearisoimaan mallin sen epävakaiseen tasapainotilaan (heiluri ylhäällä). Samaa mallinnusmenetelmää käyttävät myös muun muassa Cazzolato et al.[2]. Menetelmä perustuu yhtälöön

$$\frac{\partial^2 L}{\partial t \partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i \quad (11)$$

Jossa q_i ovat yleistettyjä koordinaatteja, Q_i yleistettyjä voimia ja L on Lagrangen funktio. Määritellään $q(t)$:

$$q(t)^T = [\theta(t) \quad \alpha(t)]$$

$$\dot{q}(t)^T = \left[\frac{\partial \theta(t)}{\partial t} \quad \frac{\partial \alpha(t)}{\partial t} \right] \quad (12)$$

Lyhennetään merkinnästä pois aika t . Tällöin Furuta-heilurille

$$\frac{\partial^2 L}{\partial t \partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = Q_1$$

$$\frac{\partial^2 L}{\partial t \partial \dot{\alpha}} - \frac{\partial L}{\partial \alpha} = Q_2 \quad (13)$$

Lagrangen järjestelmälle pätee $L = T - V$, missä T on kokonaisliike-energia ja V kokonaispotentiaalienergia. Heilurin tapauksessa Q_i muodostuvat varteen ja heiluriin vaikuttavista voimista, eli moottorin väännöstä τ ja kitkoista B_r ja B_p .

$$Q_1 = \tau - B_r \dot{\theta}$$

$$Q_2 = -B_p \dot{\alpha} \quad (14)$$

Epälineaarisen mallin liikeyhtälöt ovat [14][25]:

$$\left(m_p L_r^2 + \frac{1}{4} m_p L_p^2 - \frac{1}{4} m_p L_p^2 \cos(\alpha)^2 + J_r \right) \ddot{\theta} - \left(\frac{1}{2} m_p L_p L_r \cos(\alpha) \right) \ddot{\alpha}$$

$$+ \left(\frac{1}{2} m_p L_p^2 \sin(\alpha) \cos(\alpha) \right) \dot{\theta} \dot{\alpha} + \left(\frac{1}{2} m_p L_p L_r \sin(\alpha) \right) \dot{\alpha}^2 = \tau - B_r \dot{\theta} \quad (15)$$

$$- \frac{1}{2} m_p L_p L_r \cos(\alpha) \ddot{\theta} + \left(J_p + \frac{1}{4} m_p L_p^2 \right) \ddot{\alpha} - \frac{1}{4} m_p L_p^2 \cos(\alpha) \sin(\alpha) \dot{\theta}^2$$

$$- \frac{1}{2} m_p L_p g \sin(\alpha) = -B_p \dot{\alpha} \quad (16)$$

Ja moottorin vääntö τ varren juuressa:

$$\tau = \frac{k_m (V_m - k_m \dot{\theta})}{R_m} \quad (17)$$

Työssä käytettävän säätimen verrokiksi tarvitaan tavallinen lineaarinen säädin, joka suunnitellaan tasapainopisteeseen linearisoidun mallin avulla. Linearisoitu kahden muuttujan funktio muodostetaan [14] yleisessä tapauksessa

$$f_{lin} = f(x_0) + \left. \left(\frac{\partial f(x)}{\partial x_1} \right) \right|_{x=x_0} (x_1 - a) + \left. \left(\frac{\partial f(x)}{\partial x_2} \right) \right|_{x=x_0} (x_2 - b) \quad (18)$$

Tällöin funktio linearisoidaan pisteen $x_0^T = [a \ b]$ ympärille. Laitteistovalmistaja on päättänyt linearisoituun tilaesitykseen, jossa systeemimatriisit A ja B ovat

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ b_3 \\ b_4 \end{bmatrix}$$

ja mitattujen tilojen ja järjestelmän rakenteen perusteella

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Systeemimatriiseissa A ja B esitetyt alkiot a_{ij} ja b_k muodostuvat seuraavasti kun järjestelmä on linearisoitu toimintapisteeseen heilurin yläasentoon kaikkien alkutilojen ollessa $x_i = 0$:

$$\begin{aligned} a_{32} &= m_p^2 (L_p/2)^2 L_r g / J_t \\ a_{33} &= -D_r (J_p + m_p (L_p/2)^2) / J_t \\ a_{34} &= m_p (L_p/2) L_r D_p / J_t \\ a_{42} &= -m_p g (L_p/2) (J_r + m_p L_r^2) / J_t \\ a_{43} &= m_p (L_p/2) L_r D_r / J_t \\ a_{44} &= -D_p (J_r + m_p L_r^2) / J_t \\ b_3 &= (J_p + m_p (L_p/2)^2) / J_t \\ b_4 &= -m_p (L_p/2) L_r / J_t \end{aligned}$$

Tässä J_t on järjestelmän kokonaisinertia, joka muodostuu:

$$J_t = J_r J_p + m_p (L_p/2)^2 J_r + J_p m_p L_r^2,$$

jolloin lopulta päädytään moottorin ominaisuuksien R_m ja k_m B -matriisiin lisäämisen jälkeen esitykseen:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 149.2751 & -14.9287 & 4.9149 \\ 0 & -261.6091 & 14.7551 & -8.6136 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 49.7275 \\ -49.1493 \end{bmatrix}$$

Järjestelmän parametrit selityksineen on esitetty Taulukossa 1. Laitteistovalmistajan linearisoidun mallin navat ovat:

$$\begin{aligned} &0.0000 + 0.0000i \\ &-7.1921 + 11.5854i \\ &-7.1921 - 11.5854i \\ &-9.1581 + 0.0000i \end{aligned}$$

4.1 Sumea malli

Järjestelmätoimittaja Quanserin käyttämä malli riittää hyvin tasapainottavan säädön suunnitteluun ja simulointiin mukana toimitetuilla PID- ja LQR-säätimillä. Kuitenkin työn sumean säätimen rakentamiseen järjestelmälle on luotava sumea Takagi-Sugeno -tyyppinen malli. Tavoitteena on jakaa epästabiilin tasapainoaseman ympäristö useampaan malliin, millä saavutetaan sumeassa toiminnassa järjestelmän epälineaarisuuden mallinnus. Jako voidaan toteuttaa joko *sector nonlinearity*- tai *fuzzy partition spaces* -järjestelmällä [17].

Erään jälkimmäisellä tyylillä toimivan sopivan ratkaisun on kehittänyt C.W. Tao et al. [19], jotka jakoivat heilurin tasapainottelusäädön toiminta-alueen (sumeisiin) alueisiin heilurin kulman ja kulmanopeuden perusteella. Järjestelmä on linearisoitava useampaan pisteeseen kuin vain totuttuihin tasapainopisteisiin. Alueille sopivat mallivaihtoehdot luodaan tekemällä heilurin kulman ja nopeuden arvoista ristikko, jota verrataan taukukon 2 alueisiin. Alueille luodaan tämän jälkeen eri vaihtoehdoista sopivat mallit, yh-

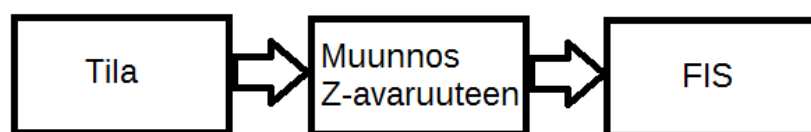
teensä seitsemän eri mallia, jotka valitaan kokeellisesti. Keskialueelle yksi, molemmille sivuille (heilurin kulma poikkeaa nolasta) kolme. Sumeat alueet on esitetty Taulukossa 2. Kulma ja kulmanopeus on jaettu myös seitsemään osaan, NB, NM, NS, ZO, PS, PM, PB. Nämä muodostuvat sanoista Negative Big, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium ja Positive Big. Nämä tulevat vastaamaan suurille luotavia sumeita jäsenyysfunktioita.

Taulukko 2: Osa sääntökannasta, sumeiden alueiden havainnollistaminen [19]

Sumeat alueet		$x_2 = \alpha$						
		NB	NM	NS	ZO	PS	PM	PB
$x_4 = d/dt \alpha$	NB	R1	R1	R1	R1	R1	R1	R1
	NM	R1	R2	R2	R2	R2	R2	R1
	NS	R1	R2	R3	R3	R3	R2	R1
	ZO	R1	R2	R3	R4	R3	R2	R1
	PS	R1	R2	R3	R3	R3	R2	R1
	PM	R1	R2	R2	R2	R2	R2	R1
	PB	R1	R1	R1	R1	R1	R1	R1

Tässä työssä yritettiin ensin soveltaa fuzzy partition spaces -menetelmää, mutta lopulta tämä vaihtoehto oli kuitenkin hylättävä monimutkaisuutensa vuoksi. Jo pelkästään sääntökannan koko olisi ollut kaksi kertaa Taulukon 2 osoittamaa suurempi, toisin sanoen $2 \times 7 \times 7 = 98$ sääntöä, jotka muodostuisivat heilurin kulman θ , varren kulmanopeuden ja varren kulman α sumeiden jäsenyysfunktioiden kautta. Tiloista heilurin kulmanopeutta ei ole käytetty sääntökannan muodostamiseen. Sääntökannan suuruusongelman lisäksi mallien valinta ei ole mitenkään yksiselitteistä saati sitten helppoa ja yksinkertaista. Valinta tehdään tyypillisesti kokeellisesti. Ilman pääsyä fyysiseen laboratorio-laitteistoon tämä ei ole mielekästä.

Sääntökannan koko järkevöityy, kun siirrytään hyödyntämään Tanaka et al. [17] esittämää ja Fairus et al. [4] käyttämää ”local sector nonlinearity” -konseptin menettelyä, missä valitaan apumuuttujiksi epälinearisuuksia aiheuttavat osat liikeyhtälöistä muodostetusta järjestelmämallista ja suoritetaan epälinearisuuksien tulkkauksella sumealla järjestelmällä näiden muodostamilta väleiltä. Kuvassa 9 esitetään asia lohkokaaavana. Sen sijaan, että käytettäisiin järjestelmän tiloja suoraan sumean järjestelmän (FIS) sisäänmenoina, ne muunnetaan sopiviksi local sector nonlinearity -menetelmän sektoreiden muodostamaan Z-avaruuteen. Muunnos mahdollistaa sääntökannan pienentämisen.



Kuva 9: Muunnos local sector nonlinearity -konseptin Z-avaruuteen

Säätökannan pienentämisen ja apumuuttujien valinnan teoriaa on esitetty Taniguchi et al. [18]. Apumuuttujiksi vastaavassa heilurijärjestelmässä Fairus et al. valitsivat:

$$\begin{aligned} z_1 &= \sin x_2 / x_2 \\ z_2 &= \cos x_2 \\ z_3 &= x_3 \sin x_2 \cos x_2 \\ z_4 &= x_4 \sin x_2 \end{aligned}$$

Tässä z_i ovat uudet apumuuttujat ja x_i järjestelmän tiloja. Apumuuttujat on ainakin työssä tarkastellun heilurin tapauksessa loogista valita juuri näin liikeyhtälöiden perusteella. Nyt järjestelmän tilat voidaan esittää kaavoista (15) ja (16) työläästi laskemalla

$$\begin{aligned} \dot{x}_1 &= x_3 \\ \dot{x}_2 &= x_4 \end{aligned}$$

$$\begin{aligned} \dot{x}_3 &= \left[\frac{-P2L_p^2 L_r M_p^2 R_m z_1 z_2 g}{N_T} \right] x_2 + \left[\frac{-P8J_p k_m^2 - P2L_p^2 M_p k_m^2}{N_T} \right] x_3 \\ &+ \left[\frac{-P8D_r J_p R_m - P2D_r L_p^2 M_p R_m - PL_p^3 L_r M_p^2 R_m z_2 z_3}{N_T} \right] x_3 \\ &\quad + \left[\frac{-P4D_p L_p L_r M_p R_m z_2}{N_T} \right] x_3 \\ &\quad + \left[\frac{-PL_p^4 M_p^2 R_m z_3 - PL_p^3 L_r M_p^2 R_m z_4}{N_T} \right] x_4 \\ &\quad + \left[\frac{-P4J_p L_p^2 M_p R_m z_3 - P4J_p L_p L_r M_p R_m z_4}{N_T} \right] x_4 \\ &\quad + \left[\frac{P8J_p k_m + P2L_p^2 M_p k_m}{N_T} \right] u \end{aligned} \tag{19}$$

$$\begin{aligned} \dot{x}_4 &= \left[\frac{-1/2L_p M_p g z_1}{N_A} \right] x_2 \\ &+ \left[\frac{1/4L_p^2 M_p z_3 - D_p + 1/2L_p L_r M_p P z_2 D_r + 1/2L_p L_r M_p P z_2 k_m^2 / R_m}{N_A} \right] x_3 \\ &+ \left[\frac{1/2L_p L_r M_p P z_2 1/2M_p L_p^2 z_3 + 1/2L_p L_r M_p z_2 1/2L_r M_p L_p z_4}{N_A} \right] x_4 \\ &\quad + \left[\frac{-k_m / R_m 1/2L_p M_p P z_2}{N_A} \right] u \end{aligned} \tag{20}$$

Joissa

$$P = \frac{1}{1/16 L_p^4 M_p^2 z_2^2 + L_r^2 M_p + J_r}$$

$$N_T = 2(R_m(PL_p^2 L_r^2 M_p^2 z_2^2 + L_p^2 M_p + 4J_p))$$

$$N_A = 1/4 PL_p^2 L_r^2 M_p^2 z_2^2 + 1/4 L_p^2 M_p + J_p$$

Kaavoissa käytetyt järjestelmän fyysiset parametrit on esitetty aiemmin Taulukossa 1. Fairus et al. [4] jatkaa tästä varsinaisen sumean mallin luomiseen. Sumea malli approksimoi epälineaarista alueella $x_2 \in [-a \ a]$, $x_3 \in [-b \ b]$, $x_4 \in [-c \ c]$, mistä muodostuvat apumuuttujien z_i maksimi- ja minimiarvot. Quanser käyttää esimerkissäädöissään parametrille $a = 0.3 \text{ rad}$, mutta Fairus et al. $a = \pi/4 \text{ rad}$, eli toiminta-alue on yli kaksinkertainen. Tavalliselle kääntöheilurille on kirjallisuudessa [17] esiintynyt jopa $a = \pm 88^\circ$ eli noin $\pm 1.535 \text{ rad}$. Furuta-heiluri on kuitenkin hieman tätä vaativampi säätökohde. Local sector nonlinearity -menetelmällä esimerkiksi sinifunktion sektori voi ulottua $\pm \pi/2 \text{ rad}$ [17]. Tämän voi todeta myös kuvasta 5.

Tavoitteena on mahdollisimman monissa tilanteissa toimiva säädin. Parametrin b täytyy olla tietenkin c :tä pienempi, jotta säätö olisi mahdollista. Valitaan ilman sumeaa mallia tehtyjen simulaatioiden perusteella $b = 6.2$ ja $c = 8$. Tällöin apumuuttujan z_i arvoiksi tulevat

$$z_{1min} = 0.9003, \quad z_{1max} = 1, \quad z_{2min} = 0.7071, \quad z_{2max} = 1, \\ z_{3min} = -3.1, \quad z_{3max} = 3.1, \quad z_{4min} = -5.657, \quad z_{4max} = 5.657$$

Maksimi- ja minimiarvojen perusteella $z_i(t)$ voidaan esittää jäsenyysfunktioina $M_{j1}(z_j(t))$ ja $M_{j2}(z_j(t))$ [4]:

$$\begin{aligned} z_1(t) &= M_{11}(z_1(t)) z_{1max} + M_{12}(z_1(t)) z_{1min} \\ z_2(t) &= M_{21}(z_2(t)) z_{2max} + M_{22}(z_2(t)) z_{2min} \\ z_3(t) &= M_{31}(z_3(t)) z_{3max} + M_{32}(z_3(t)) z_{3min} \\ z_4(t) &= M_{41}(z_4(t)) z_{4max} + M_{42}(z_4(t)) z_{4min} \end{aligned} \quad (21)$$

Missä

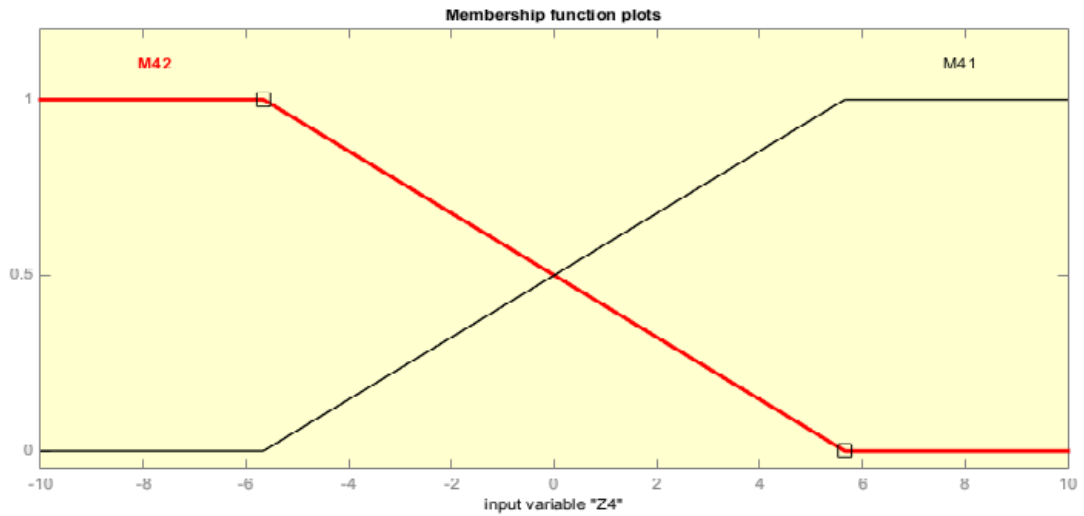
$$M_{j1}(z_j(t)) + M_{j2}(z_j(t)) = 1$$

Tämän perusteella jäsenyysfunktioiksi voidaan laskea

$$M_{j1}(z_j(t)) = \frac{z_j(t) - z_{jmax}}{z_{jmax} - z_{jmin}} \quad \text{ja} \quad M_{j2}(z_j(t)) = \frac{z_{jmax} - z_j(t)}{z_{jmax} - z_{jmin}}$$

Esimerkki jäsenyysfunktioista esitetään kuvassa 10, mutta kaikki voidaan muodostaa samalla tavalla.

Fairus et al. [4] saivat tämän jälkeen muodostettua Takagi-Sugeno -tyyppisen sumean mallin järjestelmälle vain 16 säännöllä, sillä jokaisella apumuuttuja z_i :llä on kaksi jäsenyysfunktiota, toisin sanoen $2^4 = 16$ sääntöä muodostaa täyden sääntökannan. Nämä säännöt voidaan esittää jäsenyysfunktioiden ja lokaalimallien avulla [17][4]



Kuva 10: Muuttujan z_4 jäsenyysfunktiot

$$\begin{aligned} \text{IF } z_1(t) \text{ is } {}^1M_{12} \text{ AND } z_2(t) \text{ is } {}^1M_{22} \text{ AND } z_3(t) \text{ is } {}^1M_{32} \\ \text{AND } z_4(t) \text{ is } {}^1M_{42} \\ \text{THEN } \dot{x} = {}^1Ax(t) + {}^1Bu(t) \end{aligned}$$

$$\begin{aligned} \text{IF } z_1(t) \text{ is } {}^2M_{12} \text{ AND } z_2(t) \text{ is } {}^2M_{22} \text{ AND } z_3(t) \text{ is } {}^2M_{32} \\ \text{AND } z_4(t) \text{ is } {}^2M_{41} \end{aligned} \quad (22)$$

$$\text{THEN } \dot{x} = {}^2Ax(t) + {}^2Bu(t)$$

⋮

$$\begin{aligned} \text{IF } z_1(t) \text{ is } {}^{16}M_{11} \text{ AND } z_2(t) \text{ is } {}^{16}M_{21} \text{ AND } z_3(t) \text{ is } {}^{16}M_{31} \\ \text{AND } z_4(t) \text{ is } {}^{16}M_{41} \end{aligned}$$

$$\text{THEN } \dot{x} = {}^{16}Ax(t) + {}^{16}Bu(t)$$

Tällöin sääntökannaksi muodostuu kuvan 11 tapainen sääntökanta, tosin sillä erotuksella, että sumea malli muodostaa seurauksenaan A - ja B -matriisien parametreja eikä tilatakaisinkytkennän vahvistusarvoja. Kaavassa (22) jäsenyysfunktioiden yläindeksillä ilmaistaan käsiteltävä sääntö.

Nyt voidaan laskea järjestelmämatriisien iA ja iB arvot. Saaduista $x(t)$ ja $u(t)$ saadaan laskettua sumean järjestelmän ulostulo eli selkeytettyä tulos painotettuna keskiarvona

$$\dot{x}(t) = \frac{\sum_{i=1}^{16} {}^i w(z(t)) ({}^i A x(t) + {}^i B u(t))}{\sum_{i=1}^{16} {}^i w(z(t))} \quad (23)$$

Missä

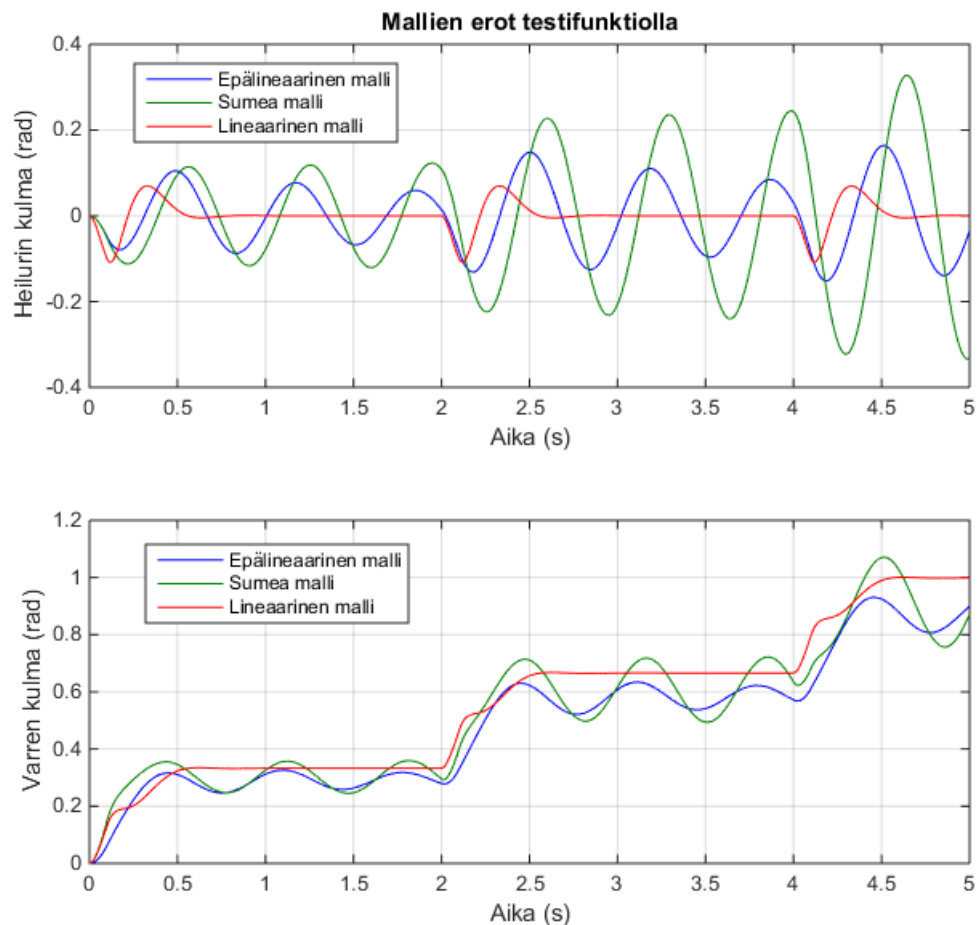
$$\begin{aligned} {}^1 w(z(t)) &= {}^1 M_{12} \times {}^1 M_{22} \times {}^1 M_{32} \times {}^1 M_{42} \\ {}^2 w(z(t)) &= {}^2 M_{12} \times {}^2 M_{22} \times {}^2 M_{32} \times {}^2 M_{41} \\ {}^3 w(z(t)) &= {}^3 M_{12} \times {}^3 M_{22} \times {}^3 M_{31} \times {}^3 M_{42} \\ {}^4 w(z(t)) &= {}^4 M_{12} \times {}^4 M_{22} \times {}^4 M_{31} \times {}^4 M_{41} \\ {}^5 w(z(t)) &= {}^5 M_{12} \times {}^5 M_{21} \times {}^5 M_{32} \times {}^5 M_{42} \\ {}^6 w(z(t)) &= {}^6 M_{12} \times {}^6 M_{21} \times {}^6 M_{32} \times {}^6 M_{41} \\ {}^7 w(z(t)) &= {}^7 M_{12} \times {}^7 M_{21} \times {}^7 M_{31} \times {}^7 M_{42} \\ {}^8 w(z(t)) &= {}^8 M_{12} \times {}^8 M_{21} \times {}^8 M_{31} \times {}^8 M_{41} \\ {}^9 w(z(t)) &= {}^9 M_{11} \times {}^9 M_{22} \times {}^9 M_{32} \times {}^9 M_{42} \\ {}^{10} w(z(t)) &= {}^{10} M_{11} \times {}^{10} M_{22} \times {}^{10} M_{32} \times {}^{10} M_{41} \\ {}^{11} w(z(t)) &= {}^{11} M_{11} \times {}^{11} M_{22} \times {}^{11} M_{31} \times {}^{11} M_{42} \\ {}^{12} w(z(t)) &= {}^{12} M_{11} \times {}^{12} M_{22} \times {}^{12} M_{31} \times {}^{12} M_{41} \\ {}^{13} w(z(t)) &= {}^{13} M_{11} \times {}^{13} M_{21} \times {}^{13} M_{32} \times {}^{13} M_{42} \\ {}^{14} w(z(t)) &= {}^{14} M_{11} \times {}^{14} M_{21} \times {}^{14} M_{32} \times {}^{14} M_{41} \\ {}^{15} w(z(t)) &= {}^{15} M_{11} \times {}^{15} M_{21} \times {}^{15} M_{31} \times {}^{15} M_{42} \\ {}^{16} w(z(t)) &= {}^{16} M_{11} \times {}^{16} M_{21} \times {}^{16} M_{31} \times {}^{16} M_{41} \end{aligned}$$

Tällä mallilla päädytään siis tilanteeseen, missä epälineaarista järjestelmää approksimoidaan lokaalien mallien avulla. Sumea malli on painotettu keskiarvo lineaarisista lokaaleista malleista, eikä vain jokin yksittäinen lineaarinen malli. Näin saadaan järjestelmän epälineaarisuus esitettyä helposti käsiteltävien lokaalien mallien avulla ja siis tietenkin myös kuvattua järjestelmää tarkemmin kuin yhdellä lineaarisella mallilla, joka toimisi vain jonkin yhden toimintapisteen rajoitetussa ympäristössä. Koska malleja ei käytetä vain yhtä kerrallaan, ei lopullinen järjestelmä ole täysin intuitiivinen toiminnaltaan, eikä sitä voi käsittää suoraviivaisesti heilurin liikkeen jakamisena alueisiin, sillä z_i ovat osittain päällekkäisiä. Kuvasta 12 voidaan nähdä mallien eroja testifunktion vasteinä. Sumea malli pystyy pääosin seuraamaan epälineaarisen mallin tuottaman vasteen muotoa. Paremmalla virityksellä toimintaa on vielä mahdollista parantaa. Kuvaa tulki-

nessa on syytä ottaa huomioon myös se, että sumea malli on luotu rajoitetulle toiminta-alueelle.

1. If (Z1 is M12) and (Z2 is M22) and (Z3 is M32) and (Z4 is M42) then (K1 is mf1)(K2 is mf1)(K3 is mf1)(K4 is mf1) (1)
2. If (Z1 is M12) and (Z2 is M22) and (Z3 is M32) and (Z4 is M41) then (K1 is mf2)(K2 is mf2)(K3 is mf2)(K4 is mf2) (1)
3. If (Z1 is M12) and (Z2 is M22) and (Z3 is M31) and (Z4 is M42) then (K1 is mf3)(K2 is mf3)(K3 is mf3)(K4 is mf3) (1)
4. If (Z1 is M12) and (Z2 is M22) and (Z3 is M31) and (Z4 is M41) then (K1 is mf4)(K2 is mf4)(K3 is mf4)(K4 is mf4) (1)
5. If (Z1 is M12) and (Z2 is M21) and (Z3 is M32) and (Z4 is M42) then (K1 is mf5)(K2 is mf5)(K3 is mf5)(K4 is mf5) (1)
6. If (Z1 is M12) and (Z2 is M21) and (Z3 is M32) and (Z4 is M41) then (K1 is mf6)(K2 is mf6)(K3 is mf6)(K4 is mf6) (1)
7. If (Z1 is M12) and (Z2 is M21) and (Z3 is M31) and (Z4 is M42) then (K1 is mf7)(K2 is mf7)(K3 is mf7)(K4 is mf7) (1)
8. If (Z1 is M12) and (Z2 is M21) and (Z3 is M31) and (Z4 is M41) then (K1 is mf8)(K2 is mf8)(K3 is mf8)(K4 is mf8) (1)
9. If (Z1 is M11) and (Z2 is M22) and (Z3 is M32) and (Z4 is M42) then (K1 is mf9)(K2 is mf9)(K3 is mf9)(K4 is mf9) (1)
10. If (Z1 is M11) and (Z2 is M22) and (Z3 is M32) and (Z4 is M41) then (K1 is mf10)(K2 is mf10)(K3 is mf10)(K4 is mf10) (1)
11. If (Z1 is M11) and (Z2 is M22) and (Z3 is M31) and (Z4 is M42) then (K1 is mf11)(K2 is mf11)(K3 is mf11)(K4 is mf11) (1)
12. If (Z1 is M11) and (Z2 is M22) and (Z3 is M31) and (Z4 is M41) then (K1 is mf12)(K2 is mf12)(K3 is mf12)(K4 is mf12) (1)
13. If (Z1 is M11) and (Z2 is M21) and (Z3 is M32) and (Z4 is M42) then (K1 is mf13)(K2 is mf13)(K3 is mf13)(K4 is mf13) (1)
14. If (Z1 is M11) and (Z2 is M21) and (Z3 is M32) and (Z4 is M41) then (K1 is mf14)(K2 is mf14)(K3 is mf14)(K4 is mf14) (1)
15. If (Z1 is M11) and (Z2 is M21) and (Z3 is M31) and (Z4 is M42) then (K1 is mf15)(K2 is mf15)(K3 is mf15)(K4 is mf15) (1)
16. If (Z1 is M11) and (Z2 is M21) and (Z3 is M31) and (Z4 is M41) then (K1 is mf16)(K2 is mf16)(K3 is mf16)(K4 is mf16) (1)

Kuva 11: Sumean säätimen sääntökanta Matlabin Fuzzy Toolboxilla esitettyinä



Kuva 12: Mallien vasteet testifunktioon

5 TOTEUTUS

Työn edetessä kävi lopulta selväksi, ettei oikeaa fyysistä laboriolaitteistoa ehditä saamaan käyttöön aikataulun sallimissa rajoissa. Tämän rajoitteen takia työ perustuukin Matlab/Simulink -simulaatioihin ja -laskentaan ja järjestelmän implementointi jää tulevien opinnäytetekijöiden vastuulle.

5.1 Säättömenetelmät

Järjestelmän toimittaja on valinnut esimerkkiratkaisuissaan tasapainottaviksi säättömenetelmiksi aikajatkuvan lineaarisen säädön PID-säätimellä ja LQR-säätimellä. Heilurin ylösheilautus on toteutettu heilurin energiataseeseen pohjautuvalla erillisellä säätimellä. Valmistajan säättömenetelmistä erityisesti LQR on työn kannalta mielenkiintoinen. Kun epälineaarinen parallel distributed -tyyppinen sumea järjestelmä suunnitellaan myös LQR-työkaluilla samoilla parametreilla, voidaan vertailla linearisoidun ja epälineaarisen säädön eroja suoraan. Lähellä tasapainopistettä säätötuloksien pitäisi olla tietenkin yhtenevät. Kun liikutaan kauemmas linearisointipisteestä, saadaan selville monimutkaiseman ratkaisun oletettu etu. Säättömenetelmänä LQR tarjoaa hyvän keskitien suorituskyvyn ja säätösuoritteen välille [5].

Matlabin LQR-työkalua käyttäen säätöratkaisu on nopeasti käsillä. Työkalu minimoi kustannusfunktion

$$J(u) = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (24)$$

järjestelmälle $\dot{x} = Ax + Bu$.

Funktiossa Q ja R ovat painomatriiseja, joilla voidaan määrittää tilojen ja ohjauksien painoarvo ja tätä kautta luoda halutunkaltainen säätö. Q vaikuttaa tiloihin x ja R ohjausmuuttujiin u , joita heilurin tapauksessa on vain yksi, joten R on skalaari. Valmistaja käyttää arvoa $R=1$. Q :n valinta ei ole yksiselitteistä, eikä yhtä oikeaa vastausta ole – säädölle asetetut tavoitteet ratkaisevat osaltaan valinnan suunnan. Järjestelmän kaistanleveys kasvaa Q :n arvojen kasvaessa, Q :n nopeustermien painotus suurentaa järjestelmän vaimennusta ja järjestelmän osa, jonka nopeutta halutaan kasvattaa, nopeutuu, kun siihen liittyvien tilojen painoa kasvatetaan [5]. Laittevalmistajan valmiissa laboratorioharjoituksissa käytetään $Q = \text{diag}([5 \ 1 \ 1 \ 1])$, mikä asettaa heilurin vaakavarren liikkeelle suurimman kustannuksen. Näin ajatellen säätötavoitteeseen pyritään mahdollisimman pienellä säätöliikkeellä, mikä taas johtaa suureen kiihtyvyyksivaatimukseen

moottorin osalta. Fairus et al. [4] ovat kokeellisesti päätyneet arvoon $Q = \text{diag}([10 \ 20 \ 10 \ 1])$. Sääötavoitteille sopivien parametrien löytäminen on hyvä laboratorioharjoitus alan opiskelijoille. Niiden löytämisessä linearisoitu malli ja lineaarinen säädin ovat työtaakkaa helpottavia, sillä Matlabin sumeiden työkalujen (Fuzzy Toolbox) rajoitteista johtuen PDC-menetelmän sumealla säätimellä tehdyt kokeet ovat erittäin työläitä.

Aikajatkuvan järjestelmän tilatakaisinkytkennän vahvistusmatriisi K lasketaan yhtälöstä

$$K = R^{-1} B^T S, \quad (25)$$

missä S on tasapainotilan ratkaisu Riccatin yhtälöön

$$A^T S + SA - SBR^{-1}B^T S + Q = 0 \quad (26)$$

Työkalun toimintaedellytyksenä on, että järjestelmä on ylipäättään stabiloitavissa, $R > 0$ ja $Q \geq 0$. Järjestelmä tulee olla saavutettava, ohjattava ja havaittava. Saavuttavuudella tarkoitetaan, että järjestelmä on mahdollista ajaa äärellisessä ajassa tilasta toiseen. Ohjattavuudella tarkoitetaan, että järjestelmän tilat voidaan ajaa origoon jostain toisesta tilasta äärellisessä ajassa. Täten kaikki saavutettavat järjestelmät ovat myös ohjattavia, mutta ei toisinpäin. Ohjattavuutta tarkastellaan ohjattavuusmatriisin avulla. Se lasketaan tilamatriisien avulla $M_c = [BAB \ \dots \ A^{n-1}B]$. Mikäli ohjattavuusmatriisin rangi vastaa järjestelmän dimensiota n eli $\text{rank}(M_c) = n$, on järjestelmä saavutettava ja ohjattava. Mikäli järjestelmä on ohjattava, on vahvistus K mahdollista löytää. [5]

Havaittavuudella ja siitä seuraavalla tarkkailtavuudella kuvataan mahdollisuutta tunnistaa järjestelmän alkutila tietämällä ohjauksen ja lähtösuureiden arvot jollakin ajanjaksoilla. Tähän tarvitaan havaittavuusmatriisi, joka lasketaan $M_o = [CCA \ \dots \ CA^{n-1}]^T$. Mikäli tämän matriisin rangi on täysi, on järjestelmä tarkkailtava ja havaittava. Kaikki tilamuuttujat vaikuttavat lähtösuureeseen. [5]

Heilurijärjestelmän osalta voidaan todeta sen olevan näiden vaatimusten mukainen, kun käytetään aiemmin esitettyä linearisoitua mallia. LQR-säädössä vahvistusmatriisi $K = [2.2361 \ -4.1633 \ 0.9071 \ -0.2326]$ tilatakaisinkytkennälle $u = -Kx$.

Sumealla järjestelmällä kaikkien kuudentoista lokaalimallin on toteutettava nämä edellä mainitut vaatimukset. Jokaiselle lokaalille mallille luodaan oma säädin täsmälleen samalla menetelmällä, kuin tavalliselle linearisoidulle mallillekin. Tämä on PDC-menetelmän vahvuus: säätimen suunnittelussa voidaan käyttää tuttuja hyvin toimivia ohjelmistotyökaluja. Samoilla Q - ja R -parametreilla lokaalimalleille LQR-työkalulla luodut tilatakaisinkytkennän vahvistukset on esitetty taulukossa 3 ja järjestelmän rakenne on hieman yksinkertaistettuna esitetty kuvassa 14. Matlab sisältää myös aikajatkuvan järjestelmämallin pohjalta toimivan työkalun, joka ratkaisee diskreettijärjestelmälle vas-

taavan tilatakaisinkytkennän vahvistuksen. Työssä käytetään aikajatkuvaa ratkaisua, mutta säätimen soveltamisessa oikeaan fyysiseen maailmaan diskreettiaikainen ratkaisu olisi todennäköinen valinta.

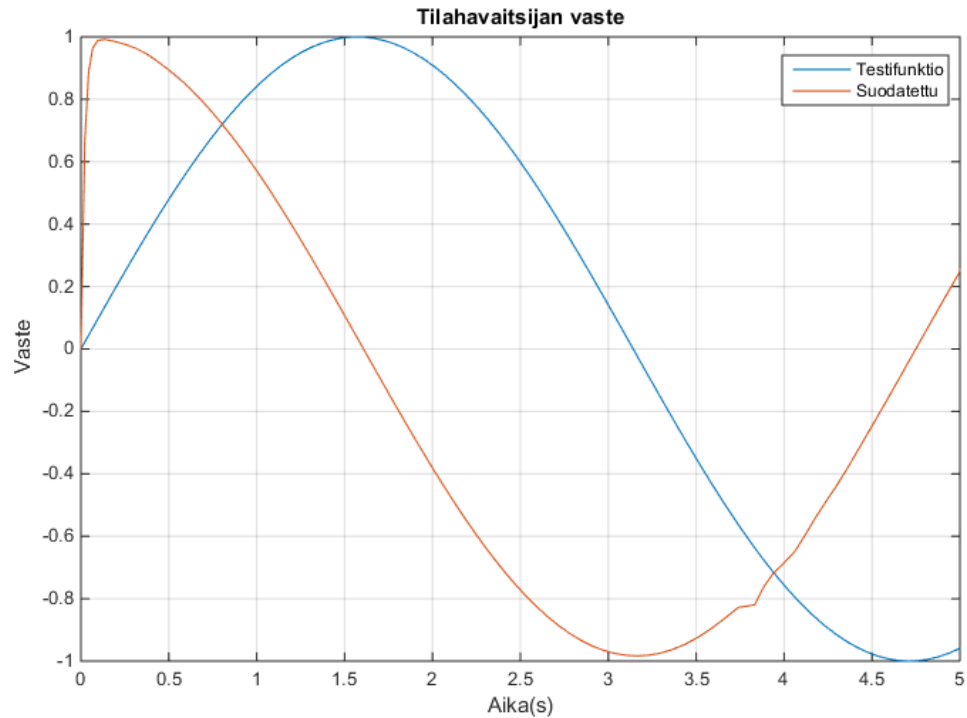
Työn luvun 3.1.1 perusteella tiedetään, että vaikka havaittavuusmatriisin rangi on täysi, säätöön käytetyistä järjestelmän tiloista mitataan vain kahta. Mitattuja suureita ovat heilurin ja sen varren kulmat θ ja α . Kulmanopeudet täytyy siis laskea näistä, jotta tilat voidaan ajaa haluttuihin arvoihin. Tämä toteutetaan tilahavaitsijalla. Työssä käytetty tilahavaitsija on myös laitevalmistajan käyttämä, yksinkertainen suodin, joka toteuttaa siirtofunktion

$G(s) = \frac{50s}{s+50}$. Tällöin kulmanopeudet lasketaan varrelle $\dot{\theta} = G(s)\theta$

ja heilurille $\dot{\alpha} = G(s)\alpha$. Kuvan 13 vasteen perusteella tämä karkea ratkaisu hoitaa tehtävänsä tarpeeksi hyvin. Laitteistovalmistaja ei perustele ratkaisuaan courseware-materiaalissa, vaan opiskelijoita vain kehoitetaan käyttämään sitä laboratoriotöissä. Paremman tilahavaitsijan tekeminen järjestelmälle sopii hyvin harjoitustyöksi, mutta on laajuutensa vuoksi rajattava tämän työn ulkopuolelle. Järjestelmän malli tunnetaan, mikä on hyvä lähtökohta tilahavaitsijan tekemiselle. Häiriöiden suuruus ja malli ovat kuitenkin tuntemattomia. [5]

Taulukko 3: Sumean säätimen vahvistusarvot

$$\begin{aligned}
 {}^1K &= [2.2361 \quad -11.7114 \quad 0.6717 \quad -0.9777] \\
 {}^2K &= [2.2361 \quad -11.1063 \quad 0.6784 \quad -0.6909] \\
 {}^3K &= [2.2361 \quad -6.5337 \quad 0.5445 \quad -1.0145] \\
 {}^4K &= [2.2361 \quad -6.6484 \quad 0.5420 \quad -0.7498] \\
 {}^5K &= [2.2361 \quad -10.6180 \quad 0.6637 \quad -0.9726] \\
 {}^6K &= [2.2361 \quad -10.0422 \quad 0.6704 \quad -0.6814] \\
 {}^7K &= [2.2361 \quad -4.7673 \quad 0.5917 \quad -1.0188] \\
 {}^8K &= [2.2361 \quad -4.9552 \quad 0.5846 \quad -0.7531] \\
 {}^9K &= [2.2361 \quad -10.9004 \quad 0.6595 \quad -0.9804] \\
 {}^{10}K &= [2.2361 \quad -10.3307 \quad 0.6671 \quad -0.6925] \\
 {}^{11}K &= [2.2361 \quad -6.1668 \quad 0.5455 \quad -1.0072] \\
 {}^{12}K &= [2.2361 \quad -6.2775 \quad 0.5430 \quad -0.7426] \\
 {}^{13}K &= [2.2361 \quad -9.7998 \quad 0.6542 \quad -0.9894] \\
 {}^{14}K &= [2.2361 \quad -9.2624 \quad 0.6615 \quad -0.6958] \\
 {}^{15}K &= [2.2361 \quad -4.5252 \quad 0.5942 \quad -1.0147] \\
 {}^{16}K &= [2.2361 \quad -4.7023 \quad 0.5871 \quad -0.7490]
 \end{aligned}$$



Kuva 13: Quanser tilahavaintsijan vaste testifunktioon. Suodatinta sovelletaan kulmanopeuksien laskentaan, joten testifunktio edustaa kulman arvoa ja suodatettu tulos tämän aikaderivaattaa.

Kun säätäjälle tuleva tieto järjestelmän tiloista on peräisin tilahavaintsijalta, pitää huomioida, että kyseessä on tilaestimaatti. Tätä merkitään usein \hat{x} , jolloin ohjaussignaali muodostuu tietysti $u = -K\hat{x}$.

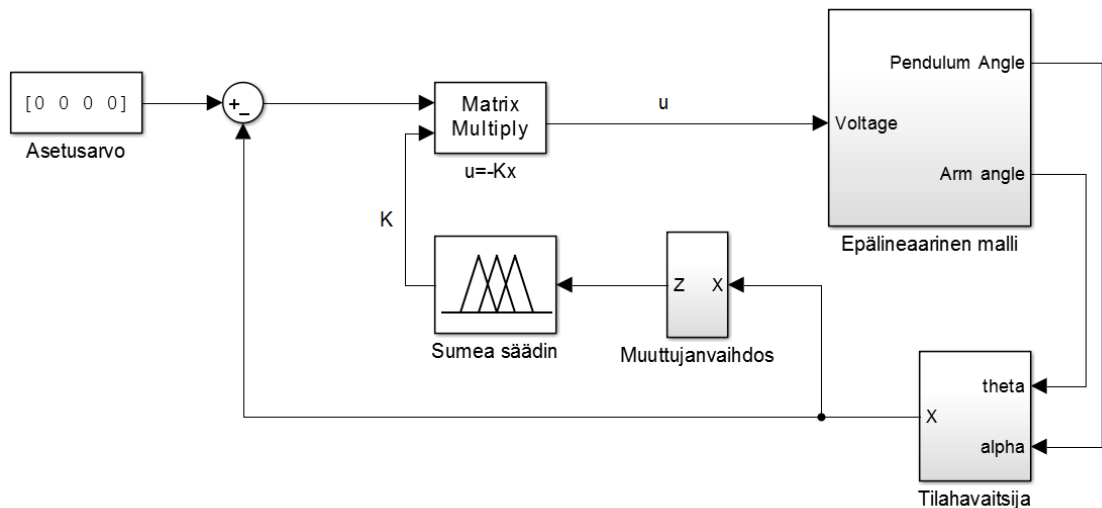
Kappaleen 2.3.1 perusteella järjestelmä on stabiili, kun kaikille 256:lle LMI:lle löydetään yhteinen ratkaisu G . Tämä voidaan laskea Matlabin Robust Control Toolboxin avulla. LMI:t ja tarvittava koodi on esitetty Liitteet-osiossa. Ratkaisu odotetusti löytyykin ja järjestelmän voidaan todeta täyttävän edellämainitun stabiiliusehdon:

$$G = \begin{bmatrix} 0.0001 & -0.0000 & 0.0000 & 0.0000 \\ -0.0001 & 0.0001 & 0.0000 & 0.0000 \\ -0.0002 & 5.7007 & -0.0000 & -0.0000 \\ -5.7007 & -0.0002 & 0.0000 & 0.0000 \end{bmatrix} \times 10^4$$

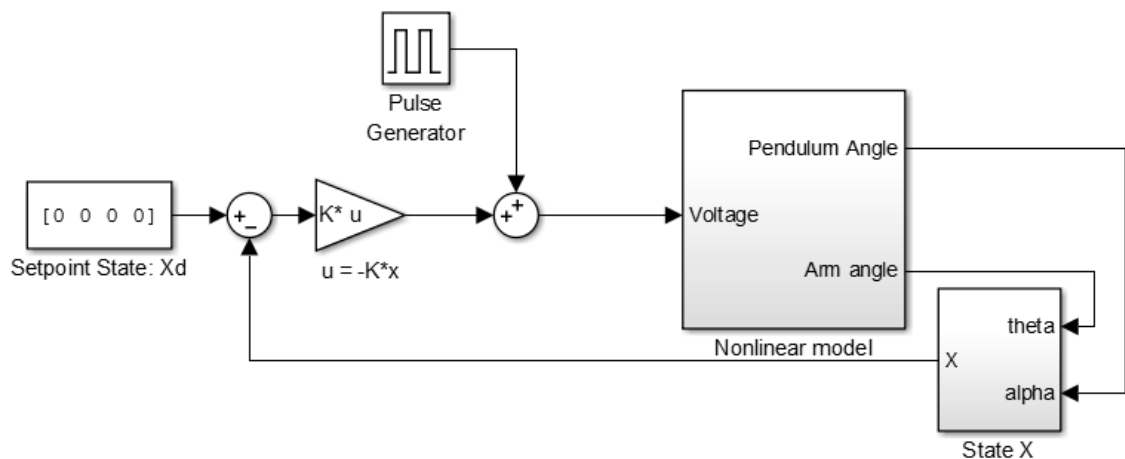
5.2 Simulaatiotuloksia

Ensimmäisessä kokeessa simuloitua heiluria häiritään pulssigeneraattorilla, jotta saadaan selville, pystyykö säädin ajamaan tilat haluttuihin arvoihin. Yksinkertaistettu esitys lineaarisista säädintä käyttävän simulaattorin rakenteesta on esitetty kuvassa 15. Aloitetaan linearisoidulla mallilla ja sille suunnitellulla lineaarisella LQR-tilasäätimellä: Ku-

vasta 16 voidaan nähdä heilurin kulman palautuvan pienen häiriöpulssin jälkeen n. 2 sekunnin päästä takaisin nollaan (heilurin yläasento). Oletuksena on, että sumea säädin suoriutuu tehtävästään vastaavalla tavalla, sillä heilurin yläasennon lähellä säädinten pitäisi olla lähes täysin toisiaan vastaavat. Tämä johtuu siitä, että molemmat on suunniteltu LQR-periaatteilla ja linearisoitu malli on tehty heilurin yläasentoon.



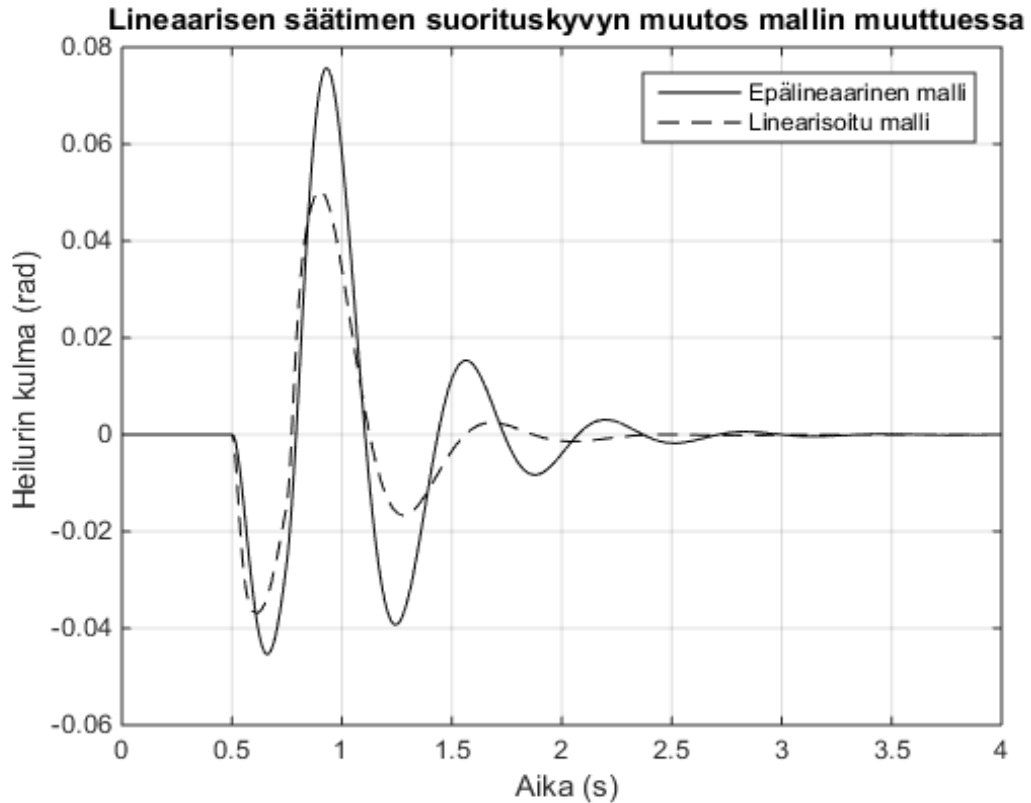
Kuva 14: Tilatakaisinkytkentä sumealla säätimellä



Kuva 15: Yksinkertaistettu simulaattori lineaarisella säätimellä

Epälineaarilla mallilla tilanne muuttuu hieman. Muutosta havainnollistetaan edelleen kuvassa 16. Linearisoidun mallin avulla suunniteltu säädin toimii tietenkin nyt huommin. Samanlaisella häiriöllä kokeiltuna heilurin liikkeitä on hieman suurempia ja sen lopullinen rauhoittuminen tapahtuu vasta noin sekuntia myöhemmin lineaariseen

malliin verrattuna. Säädin pystyy kuitenkin edelleen vakauttamaan heilurin, mutta liike ei ole yhtä määrätietoisesti siisti, vaan sisältää hieman oskillointia.



Kuva 16: Heilurin kulman vaste häiriöpulssiin lineaarisella LQR-tilasäätimellä

Edellisissä kokeissa oltiin selvästi vielä laitteistovalmistajan säätimellä määrittämän sallitun heilurikulman ± 0.3 rad sisäpuolella. Tästä huolimatta epälineaarisuuden tuoma ero oli erittäin selkeästi nähtävillä ja säätimen suorituskyky huonontui selvästi, kun se joutui selviämään tilanteesta, johon sitä ei ole suunniteltu. Tarkastellaan seuraavaksi lineaarisen LQR-säätimen ja sumean PDC-säätimen eroja.

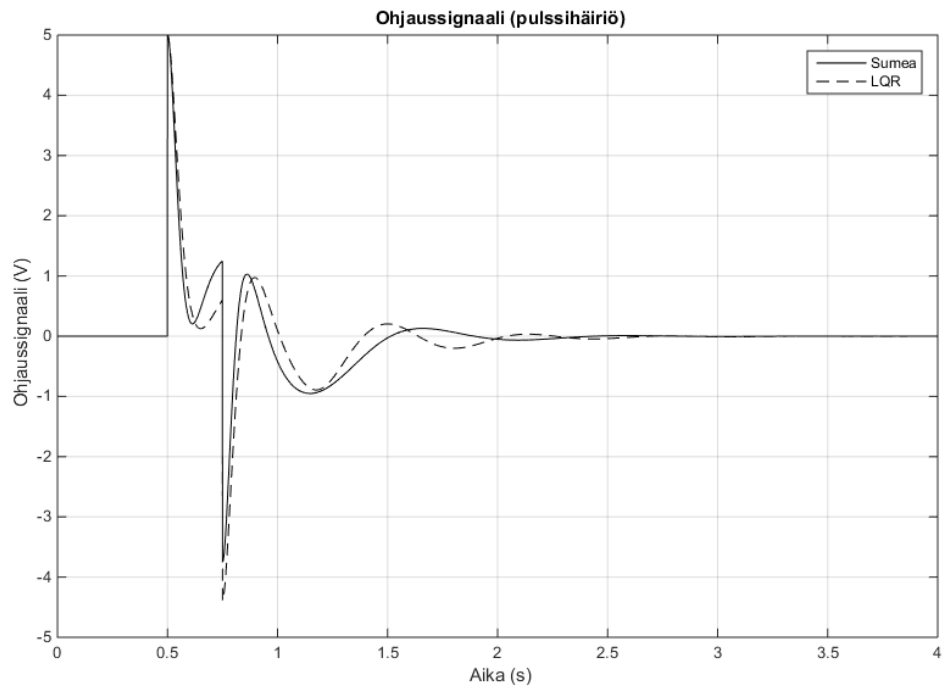
Säätimen ja prosessin väliin ohjausjännitteeseen luodaan häiriöpulssi. Tällä saadaan aikaiseksi liike koko heiluriin tai voidaan tutkia äkillistä häiriötä ohjaimessa, moottoris- sa tai jännitelähteessä. Kuvassa 17 on esitetty järjestelmän ohjaussignaali tällaisessa tilanteessa. 5V häiriö on vielä molemmille säätimille hallittavissa oleva ilmiö. Sumea säädin reagoi hieman voimakkaammin ja on täten hieman nopeampi. Sumean järjestelmän vaste on myös siistimpi eikä sisällä lineaarisen järjestelmän ylimääräisiä liikkeitä. Toistettaessa koe suuremmalla 10V häiriöpulssilla sumean järjestelmän reagoinnin voimakkuus tuli näkyviin entistä paremmin. Mikäli ohjaussignaalin maksimi asetettiin samalla 10V:iin, saturoitui ohjaus hetkeksi sumealla säätimellä. Lineaarinen säädin toimi laiskemmin eikä pystynyt tällöinkään vastaamaan sumean järjestelmän suorituskykyyn.

Sumean säätimen reagointi on ”oikeampi”, sillä järjestelmän häiriön viedessä sen kauemmas tasapainopisteestä epälineaarisuus ja muuttunut toimintapiste voidaan huomioida.

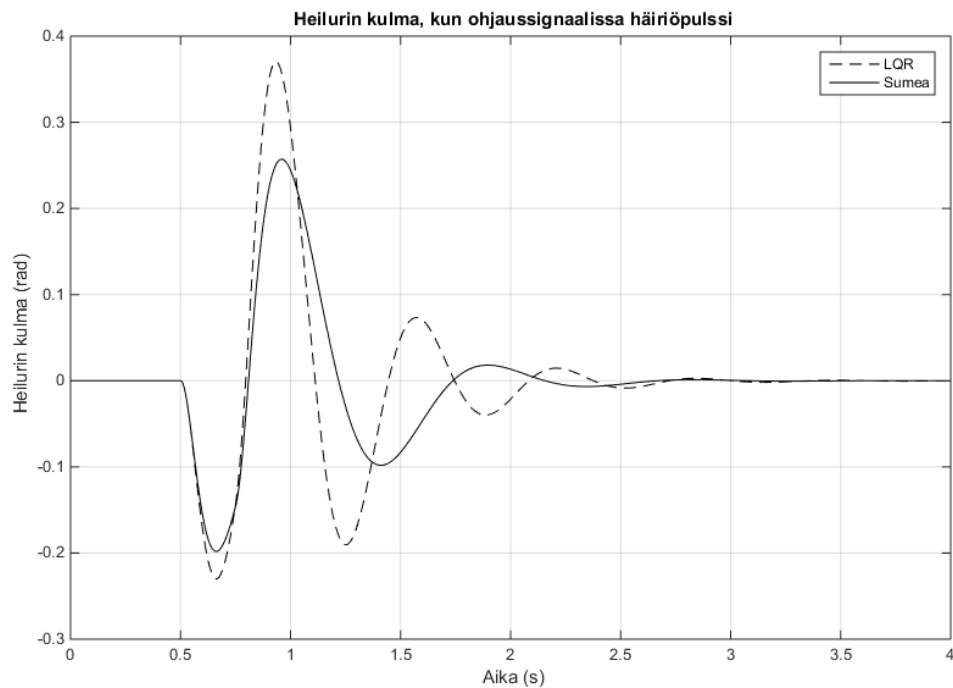
Kun vastaavalla tavalla tehdään askelkoe, ollaan säätimen ohjaussignaalin suhteen kuvan 19 tilanteessa. Tällöin häiriötilanne vastaa fyysisesti heilurin varren siirtämistä toiseen asentoon. Pienelläkin häiriöllä voidaan havaita sumean järjestelmän parempi suorituskyky, haluttu tila saavutetaan noin puoli sekuntia nopeammin. Sumea järjestelmä reagoi voimakkaammin ja hieman nopeammin.

Samoissa tilanteissa, eli pulssikokeessa ja askelkokeessa, heilurin kulma vaihtelee kuvien 18 ja 20 mukaisesti. Sumealla järjestelmällä heilurin liikkeet ovat pienempiä ja värähtelyä ei juurikaan esiinny. Tarkemmalla virittämisellä vastetta olisi myös edelleen mahdollista parantaa, kun taas lineaarinen säädin tekee jo kaiken mihin pystyy. Askelkokeen häiriö on melko pieni, ja heilurin liike onkin selvästi pienempi kuin $\pm 0,3\text{rad}$.

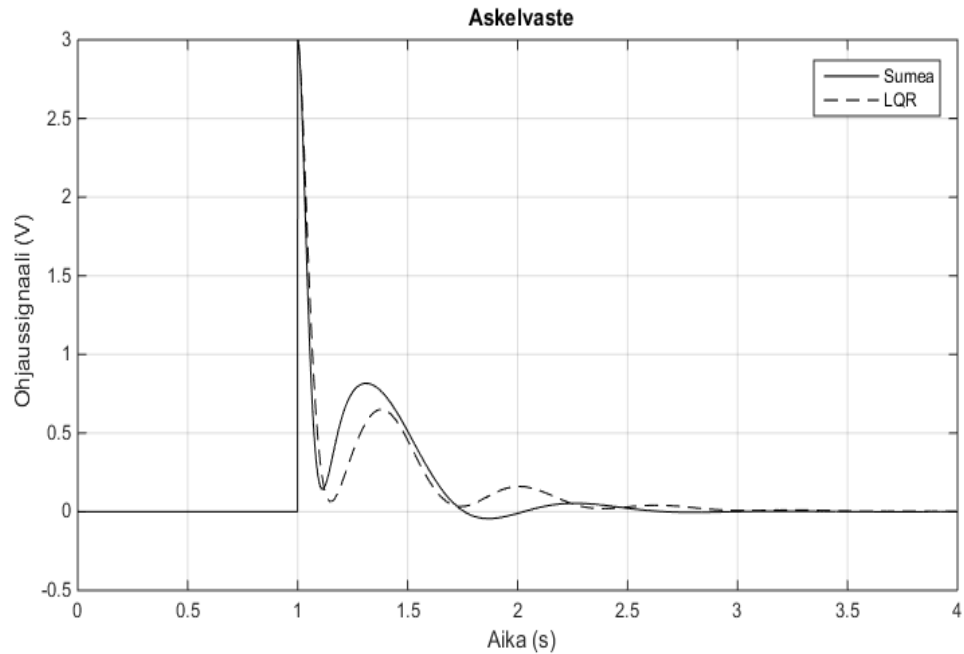
Pulssikokeen (5V) seurauksena heilurin liikkeiden erot ovat erittäin selkeät. Kuvan 18 vasteista voidaan nähdä kuinka jälleen kerran lineaarista säädintä käyttävä järjestelmä päästää heilurin liikkumaan suuremman matkan. Kuten aiemmin, myös tässäkin tapauksessa sumea järjestelmä tuottaa myös vähemmän värähtelyä. Jos sumean säätimen kuvaajaa vertaa kuvan 16 tilanteeseen, missä oli kyse lineaarisen säätimen ja linearisoidun järjestelmän tilanteesta, voidaan nähdä niissä hyvin paljon samankaltaisuutta. Tästä voidaan päätellä sumean järjestelmän tulkitsevan epälineaarisuuksia melko hyvin.



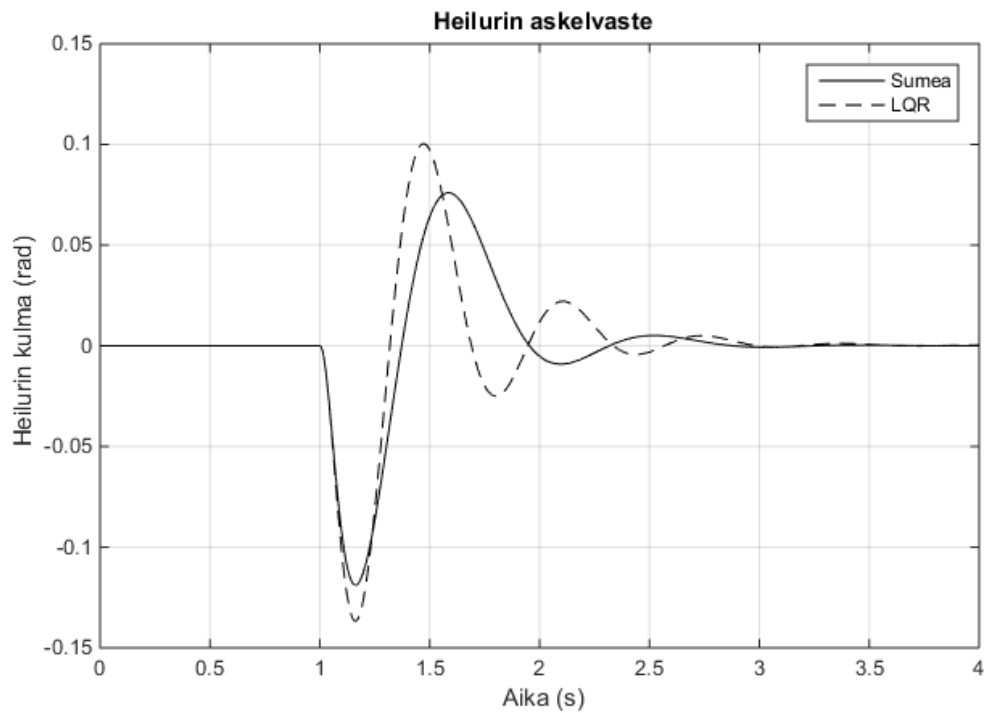
Kuva 17: Säätimen ohjaussignaali, kun järjestelmää häiritään pulssilla. Pulssi alkaa 0.5 s kohdalla ja on 5V suuruinen



Kuva 18: Heilurin vaste 5V häiriöpulssiin. Yli $\pm 0,3\text{rad}$ liike aiheuttaisi heilurin tippumisen ala-asentoon ja uuden ylösheilautuksen, jos myös swing-up -sääto on käytössä



Kuva 19: Säätimen ohjaussignaali, kun varren asentoa muutetaan askelmaisesti. Sumea säädin reagoi nopeammin ja voimakkaammin



Kuva 20: Heilurin vaste askelmuotoiseen häiriöön (varren asennon muutos). Sumea järjestelmä tuottaa siistin vasteen ilman värähtelyä. Heiluri liikkuu vähemmän kuin lineaarisella säätimellä.

5.3 Havaitut ongelmat

Työtä tehdessä useita ongelmia ilmaantui eteen ratkaistavaksi. Etukäteen tiedossa oli sumeiden järjestelmien virittämisen hankaluus, sillä tiettyä, aina toimivaa metodia ei ole. Parallel distributed -menetelmään liittyvien lokaalimallien löytäminen oli ensimmäisenä eteen tullut suuri ongelma. PDC-menetelmän valintaan vaikuttanut lähdekirja [9] ei juurikaan ottanut kantaa asiaan. Ratkaisu näytti löytyvän C.W.Tao et al. [19] artikkelista, jossa säädettiin myös Furuta-heiluria sumealla PDC-säätimellä. Heidän ratkaisunsa oli asiaa tarkemmin tuntemattomalle looginen ja intuitiivinen: heilurin toiminta-alue jaettiin sumeisiin osiin (fuzzy partition spaces), mikä vastaa melko hyvin opiskelijoille tutumpaa gain scheduling -ajattelua. Samanlaista ratkaisua käyttivät myös H.Ding et al. [3].

Partition spaces -ratkaisussa sääntökannasta muodostui valtava, lähes sadan säännön hirviö. Kuitenkin suurin ongelma oli edelleen lokaalimallien löytäminen. C.W.Tao et al. [19] käyttämä menetelmä vaati mallien kokeellista löytämistä ehdokasmallien joukosta. Samaan kokeelliseen tapaan ratkaista tai varmistaa asia olivat päätyneet muutkin. Selvästi ongelma oli suuruusluokaltaan sellainen, että se mielellään lakaistiin maton alle.

Ratkaisu tilanteeseen löytyi lopulta K.Tanaka et al. [17], missä partition spaces -ajattelulle esitettiin vaihtoehdoksi ”local sector nonlinearity”. Tässä ratkaisussa lokaalimallit muodostettiin mahdollisen säätöavaruuden reunoilta ja ilmeni, että ratkaisua oli käytetty menestyksekkäästi Furuta-heilurin säätöön [4].

Ongelmaksi jäi nyt enää sopivan avaruuden löytäminen. Säätötavoitteena oli muun muassa suurempi säätimen toiminta-alue, joten jonkinlainen lähtökohta oli olemassa. Muut rajat löytyivät simulaatioiden perusteella.

Säädintä tehdessä paljastui Matlabin Fuzzy Toolboxin työkalujen kankeus ja vanhanaikaisuus, mikä teki kokeiluista vianetsintätilanteissa erittäin työläitä. Väärin toimivan prototyypin vikaa etsiessä oli vaikea kohdistaa huomio oikeaan paikkaan. Oireena oli, että säädin ajautui jatkuvasti säätöavaruuden äärioloihin eikä tilatakaisinkytkennän vahvistus muuttunut odotetusti. Ensireaktio oli muuttaa avaruutta, eli oletuksena oli, että local sector -parametrit olivat väärät. Tästä ei kuitenkaan ollut apua ja lopulta koko sumea malli piti hylätä ja tehdä alusta asti uusiksi. Lopulta vika paljastui epälineaarisen mallin liikeyhtälöistä. Vian löydyttyä sumea säädin syntyi nopeasti.

Säädin on tehtävänsä melko monimutkainen, mutta olennaista on, että se osoitti kykynsä epälineaarisen prosessin säädössä. Vastaavaa konseptia voi soveltaa moneen paikkaan, missä tarvitaan joko tarkempaa (värähtelyt) tai nopeampaa säätöä, kuin mihin lineaarinen säädin pystyy. Aluksi Takagi-Sugeno -tyyppinen sumea järjestelmä tuntuu epäintuitiivisemmalta kuin Mamdani-järjestelmä, kun sumeat järjestelmät mielletään ihmisoperaattoria matkiviksi. Vaikutelma kuitenkin hälvenee nopeasti ja T-S on jopa helpokäyttöinen työkalu.

5.4 Hyviä ratkaisuja

Työtä tehtäessä ei ollut mahdollisuutta kerätä dataa oikeasta fyysisestä laitteistosta. Mikäli käytössä olisi hyvälaatuista mitattua dataa, voisi sillä todentaa mallin oikeellisuuden ja jopa virittää sitä paremmaksi. Nyt ei ole myöskään takeita siitä, että laitevalmistajan kurssimateriaaleista kerätyt laitteen ominaisuudet ja arvot vastaisivat todellisuutta. Virheellisiin laitteistovalmistajan ilmoittamiin arvoihin on törmätty esimerkiksi TTY:n Roinilan diplomityössä [15]. Mikäli simuloitu ja oikea heiluri eivät vastaa toisiaan, on simulaation parametrit yksinkertaista korjata. Tämän asian varmistaminen voisi kuulua opiskelijoiden tekemiin laboratorioharjoituksiin.

Local sector nonlinearity -menetelmä on erittäin tehokas pienentämään sumean järjestelmän sääntökantaa. Mikäli sopivia säätöavaruuden muodostavia sektoreita ei löydy tai niitä ei voida päätellä, on menetelmän käyttäjä pulassa. Parhaiten se sopiikin esimerkiksi mekaanisten järjestelmien, kuten juuri Furuta-heilurin sumeaan mallintamiseen, sillä mekaanisilla järjestelmillä on aina rajat jonkin asian suhteen. Menetelmän käyttäminen sallii opiskelijoille ymmärrettävän systemaattisen etenemistavan.

Säädön onnistumisen kannalta mallin on oltava mahdollisimman hyvä. Sumea järjestelmä voi olla yksinkertaista saada toimimaan joten kuten, mutta hyvin toimivan järjestelmän aikaansaaminen edellyttää hyvää mallia. Aina kuitenkin ei ole selvää, miten tähän voidaan päästä, ja monesti sumeita järjestelmiä viritetään kokeellisesti. Sector nonlinearity ja parallel distributed compensation ovat menetelmiä, jotka ovat kytköksissä LMI-ongelmiin. Näiden ratkaisuun on tehokkaita ja kehittyneitä ohjelmistotyökaluja [18][12]. Näin ollen näillä menetelmillä rakennettujen järjestelmien virittämiseen ja stabiiliustarkasteluun voi olla myös hyvinkin selkeä numeerisesti saavutettava ratkaisu.

6 TULOKSET JA JOHTOPÄÄTÖKSET

Työn tuloksena saatiin luotua tasapainottava epälineaarinen sumea säädin, joka on lineaarista nopeampi ja tarkempi. Sumea säädin pystyy myös toimimaan laajemmalla alueella. Tältä osin tavoitteet voidaan katsoa saavutetuiksi.

Säädin pyrittiin työssä tekemään valmistajan lineaarista LQR-säädintä vastaavaksi, mikä vaikutti muun muassa parametrien valintaan. Tavoitteena oli saada tällä tavalla epälineaarisen säätimen ero normaaliin lineaariseen paremmin esille ilman muita apuja tai eroja. Tämä onnistui hyvin ja epälineaarinen sumea säädin oli huomattavasti parempi samoilla LQR-parametreilla. Näiden parametrien valinnassa lineaarinen malli ja säädin voisivat ehkä toimia kevyempänä apuvälineenä, kun otetaan huomioon Fuzzy Toolboxin rajoitteet ja halutaan kaikkien lokaalisäätimien toimivan samoilla kustannuspainoparametreilla. PDC-säätöhän mahdollistaa näiden parametrien muuttamisen lokaalimallikohdaisesti. Tällainen lähestymistapa sopii kuitenkin paremmin fuzzy partition spaces -ajatteluun ja täten fyysisen liikkeen jakamiseen alueisiin. Local sector nonlinearity -menetelmän avaruutta ei voida mieltää samankaltaiseksi.

Säätimiä verrattiin samanlaisissa tilanteissa samalla epälineaarilla liikeyhtälöihin perustuvalla mallilla. Vielä ei voida tietää, miten hyvin käytetty malli vastaa oikeaa heiluria. Tällä ei kuitenkaan työn lopputuloksen kannalta ole merkitystä, sillä malli ja säädin saadaan kyllä tehtyä helposti uudestaan, kun fyysiset parametrit saadaan varmistettua.

Sumean säätimen parametreissa on varmasti vielä optimoitavaa ja sen todellinen suorituskyky paljastuu mahdollisesti vasta kokeellisen, työlään virittämisen kautta. Sumea ja epälineaarinen säätö tuovat kumpikin lisäkerroksen monimutkaisuutta ja valittavia parametreja ja menetelmiä säädön suunnitteluun. Yleisesti ajatellen säädettävän prosessin on oltava todella vaativa säädön suhteen, jotta lisääntynyt monimutkaisuus olisi perusteltua. Furuta-heiluri ei tarvitse välttämättä epälineaarista säädintä, vaikka säätötulos sellaisella toki parempi onkin. Epälineaarisen säädön tutkimisen ja opetuskäytön motiiveilla monimutkainen säädin voidaan kyllä perustella.

Opetustavoite oli hieman epämääräisempi suorituskykytavoitteisiin verrattuna, sillä tarkat yksityiskohdat laitteiston, mallin ja säätimen käytöstä opetuksessa puuttuvat vielä. Kuitenkin työn edetessä tunnistettiin useita perinteisesti harjoituksena tehtäviä, sopivan haastavia töitä opiskelijoille. Näitä olivat muun muassa sumean säätimen LQR-parametrien valinta ja local sector nonlinearity -menetelmän parametrien muodostaman avaruuden löytäminen. Myös tilahavaitsijan parantamisessa on ainesta harjoitukseksi. Peruskursseilla opiskelijoiden osaamistaso riittää lineaariseen säätöön, esimerkiksi na-

pojen asetteluun. Pidemmälle ehtineille myös LQR ja sumeiden järjestelmien perusteet ovat tuttuja. Työn local sector nonlinearity ja parallel distributed -säätö ovat syventävien kurssien konsepteja.

Opetuskäyttö jättää tilaa järjestelmän kehittämislle edelleen. Mitä tarkemmaksi mallin saa tehtyä, sitä paremman säätimen saa myös tehtyä. Muun muassa Ohtake et al. [12] tulokset osoittavat, että sumea järjestelmä vastaa epälineaarista erittäin hyvin. Tämän työn malliin jäi jonkin verran parantamisen varaa teoriaan [17] verrattuna.

Säädintä voisi myös kehittää edelleen niin, että swing-up integroidaan osaksi sumeaa säätöjärjestelmää. Nykyisellään swing-up -säädin on erillinen ja perustuu heilurin energiataseeseen. Swing-up -säätimen ja sumean säätimen yhdistelmä on esitetty työn Liitteet-osiossa kuvassa 21. Tämän yksinkertaisen ja tehokkaasti toimivan swing-upin kehittivät Åström ja Furuta [26]. Mikäli swing-up halutaan kuitenkin integroida tasapainottavan säätimen kanssa samaan järjestelmään, on sumea PDC-säädin hyvä vaihtoehto. Kunhan heilurin toiminta-alue tunnustetaan oikein, voi lokaalin säätimen suunnitella juuri sellaiseksi kuin swing-upin kannalta on parasta. Tällöin on kuitenkin pidettävä huolta siitä, että sumea päättely pystyy eristämään swing-up -säädön ja tasapainottavan säädön toisistaan esimerkiksi sopivilla jäsenyysfunktioilla, muuten säätötulos on huono.

Sumea PDC-säätö osoittautui mielenkiintoiseksi konseptiksi työtä tehdessä. Kun vielä lopputulos oli hyvä ja soveltamismahdollisuuksia on paljon, voi sumealle PDC-säädölle ennustaa hyvää menestystä. Aina hyvät menetelmät eivät kuitenkaan saa tuulta alleen, ja sumeat järjestelmät ovat olleet teknisen kehityksen muodin ulkopuolella, minkä osoittaa muun muassa Mathworksin Matlabin sumeiden työkalujen vanhanaikaisuus.

Sector nonlinearity -konsepti osoittautui erittäin toimivaksi. Sen mahdollistama sumean sääntökannan raju pienentäminen laajentaa sumean säädön sovelluspiiriä sellaisiin järjestelmiin, joiden lukuisat säädettävät tilat ovat merkinneet epäkäytännöllisen suurta sääntökantaa.

Työn tekeminen vaati hyvin laaja-alaista osaamista. Jo pelkästään heilurin dynamiikassa on tarpeeksi tekemistä lopputyöhön, minkä osoittaa aiheesta kirjoitetun tutkimuksen määrä. Kun tähän lisätään sumean logiikan osaaminen ja tutustuminen epälineaariin sumeeseen säätöön ja sivutaan optimisäätöteoriaa ja LMI-ongelmia, on henkinen kuormitus melko suuri. Ajoittain tämä johti työn tekemisessä ongelmiin, kuten ”sokeutumiseen” liikeyhtälöiden virheisiin. Lopulta kaikki kuitenkin selvisi ja tehty simulaattori ja mielenkiintoinen sumea säätöratkaisu olivat varsin antoisia kohde työlle.

LÄHTEET

- [1]: Acosta, J.A., 2009, Furuta's Pendulum: A Conservative Nonlinear Model for Theory and Practise, *Mathematical Problems in Engineering* , Volume 2010, 29p
- [2]: Cazzolato, B. et al., 2011, On the Dynamics of the Furuta Pendulum, *Journal of Control Science and Engineering* , Volume 2011, Article ID 52834, 8p
- [3]: Ding, H. et al., 2006, Dynamic Switching Control for the Swing-up and Stabilizing Control of the Furuta Pendulum, *IEEE Conference on Industrial Electronics and Applications* , Volume 2006 1st, 5p
- [4]: Fairus, M.A. et al., 2013, Fuzzy modeling and control of rotary inverted pendulum using LQR technique, *IOP Conf. Series: Materials Science and Engineering* , Volume 53, 11p
- [5]: Franklin, Gene F. et al., 1998, *Digital Control of Dynamic Systems*, Addison-Wesley, pp.1-52,279-419,479-642
- [6]: Gäfvert, M., *Modelling the Furuta Pendulum*, Tutkimusraportti, Lund Institute of Technology, 1998
- [7]: Karray, F., de Silva, C., 2004, *Soft Computing and Intelligent Systems Design*, Pearson, 560p
- [8]: Kawamoto, S. et al., 1992, An Approach to Stability Analysis of Second Order Fuzzy Systems, *IEEE intl. conference on Fuzzy Systems* , Volume 1992, pp.1427-1434
- [9]: Lilly, John H., 2010, *Fuzzy control and identification*, Wiley, 189p
- [10]: National Instruments Corp., *LabVIEW 2014 Model Interface Toolkit Help*, Verkkodokumentti, Haettu 8.5.2015 , <http://zone.ni.com/reference/en-XX/help/374160B-01/>, National Instruments
- [11]: National Instruments Corp., *LabVIEW 2014 Help*, Verkkodokumentti, Haettu 8.5.2015 , <http://zone.ni.com/reference/en-XX/help/371361L-01/>, National Instruments
- [12]: Ohtake et al., 2001, Fuzzy Modeling via Sector Nonlinearity Concept, *IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th* , Volume 1, pp.127-132
- [13]: Park, M. et al., 2011, Swing-up and LQR stabilization of a rotary inverted pendulum, *Artificial Life and Robotics* , Volume 16, Issue 1, pp. 94-97
- [14]: Quanser, Quanser QUBE-servo, Verkkodokumentti, Haettu 15.10.2014 , http://www.quanser.com/Products/qube_servo, Quanser
- [15]: Roinila, T., 2006 , *Kaksoiskäntöheilurin dynaaminen mallinnus ja säätö tilamenetelmiä käyttäen*, Diplomityö, TTY, 106p
- [16]: Takagi, T. et al., 1985, *Fuzzy Identification of Systems and Its Applications to*

Modeling and Control , IEEE Transactions on Systems, Man, and Cybernetics , Volume smc-15, no.1., pp.116-132

[17]: Tanaka, K. et al., 2001, Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach, John Wiley & Sons. inc, 305p

[18]: Taniguchi, T. et al., 2001, Model Construction, Rule Reduction, and Robust Compensation for Generalized Form of Takagi–Sugeno Fuzzy Systems, IEEE Transactions on Fuzzy Systems , Volume 9, No.4, pp.525-538

[19]: Tao, C.W. et al., 2008, Design of a Fuzzy Controller With Fuzzy Swing-up and Parallel Distributed Pole Assignment Schemes for an Inverted Pendulum and Cart System, IEEE Transactions on Control Systems Technology , Volume 16, No. 6, pp.1277-1288

[20]: Verbruggen, H.B. et al., 1999, Fuzzy Algorithms for Control, Kluwer Academic Publishers, 352p

[21]: Viljamaa, P., Fuzzy-säätäjän viritysmenetelmiä, Lisensiaatintutkimus, TTY Sähkötekniikan osasto, 1995

[22]: Wang, Z. et al., 2004, Minimum-Time Swing-up of A Rotary Inverted Pendulum by Iterative Impulsive Control, Proceeding of the 2004 American Control Conference , Volume 2004, pp. 1335-1340

[23]: Yurkovich, S. et al., 1996, Fuzzy Controller Synthesis for an Inverted Pendulum System, Control Eng. Practice , Volume 4, No.4, pp. 455-469

[24]: Åkesson, J. et al., 2001, Safe Manual Control of the Furuta Pendulum, IEEE Intl.Conference on Control Applications , Volume 2001, pp.890-895

[25]: Åström, K.J., Pendulum Equations, Verkkodokumentti, Haettu 15.10.2014 , http://www.quanser.com/Content/CoursewareNavigators/qubeservo_matlab/Courseware/Ancillary%20Material/Profiles/Astrom/Pendulum%20Equations.pdf, Quanser

[26]: Åström, K.J. et al., 2000, Swinging up a pendulum by energy control, Automatica , Volume 36, pp.287-295

LIITTEET

A. MALLIN TEKEMISESSÄ HYÖDYNNETTYÄ MATLAB-KOODIA

Alla esitetyn Matlab-koodin avulla on luotu järjestelmälle tilaesitys. Matlabin Symbolic Toolboxia on hyödynnetty, mutta termien järjestely lopulliseksi esitykseksi on tehty käsin. Matlab-käytön vaatimat järjestelmän fyysiset parametrit määritellään ja sijoitetaan esitykseen.

Local sector nonlinearity -konseptin mukaisesti järjestelmän osamallit lasketaan säätovaruuden rajoilta. Näin saadut 16 mallin ja säätimen parametrit sijoitetaan Fuzzy Toolboxin käyttämiin FIS-tiedostoihin. Lopuksi luodaan vielä laboratoriolaitteiston valmistajan käyttämän linearisoidun järjestelmän malli ja säädin [14].

```
%% Sumea malli ja säädin
%
syms Rm kt km Mr Lr Dr Jr Mp Lp Dp Jp g;
syms x1 x2 x3 x4 dotx3 dotx4 Vm;
syms Z1 Z2 Z3 Z4 P;
tau=km*(Vm-km*x3)/Rm;
eqn1=(Mp*Lr^2+0.25*Mp*Lp^2*0.25*Mp*Lp^2*(cos(x2))^2+Jr)*dotx3-
(0.5*Mp*Lp*Lr*cos(x2))*dotx4+0.5*Mp*Lp^2*sin(x2)*cos(x2)*x3*x4+0.5*Mp*
Lp*Lr*sin(x2)*x4^2==tau-Dr*x3;
dotdottheta=solve(eqn1,dotx3);
eqn2=0.5*Mp*Lp*Lr*cos(x2)*dotdottheta+(Jp+0.25*Mp*Lp^2)*dotx4-
0.25*Mp*Lp^2*cos(x2)*sin(x2)*x3^2+0.5*Mp*Lp*g*sin(x2)==-Dp*x3;
dotdotalfa=solve(eqn2,dotx4);
eqn3=(Mp*Lr^2+0.25*Mp*Lp^2*0.25*Mp*Lp^2*(cos(x2))^2+Jr)*dotx3-
(0.5*Mp*Lp*Lr*cos(x2))*dotdotalfa+0.5*Mp*Lp^2*sin(x2)*cos(x2)*x3*x4+0.
5*Mp*Lp*Lr*sin(x2)*x4^2==tau-Dr*x3;
dotdottheta=solve(eqn3,dotx3);
sievadotdotalfa=simplify(dotdotalfa);
sievadotdottheta=simplify(dotdottheta);
%
dotdotthetaz=subs(dotdottheta,x3*sin(x2)*cos(x2),Z3);
dotdotthetaz=subs(dotdotthetaz,x3*cos(x2)*sin(x2),Z3);
dotdotthetaz=subs(dotdotthetaz,cos(x2)*sin(x2)*x3,Z3);
dotdotthetaz=subs(dotdotthetaz,cos(x2),Z2);
dotdotthetaz=subs(dotdotthetaz,x4*sin(x2),Z4);
dotdotthetaz=subs(dotdotthetaz,sin(x2),Z1);
%
[abbrdotdotthetaz,P1]=subexpr(dotdotthetaz,P);
sievadotdotthetaz=simplify(abbrdotdotthetaz);
%pretty(sievadotdotthetaz);
%
dotdotalfaz=subs(dotdotalfa,x3*sin(x2)*cos(x2),Z3);
dotdotalfaz=subs(dotdotalfaz,x3*cos(x2)*sin(x2),Z3);
dotdotalfaz=subs(dotdotalfaz,cos(x2)*sin(x2)*x3,Z3);
dotdotalfaz=subs(dotdotalfaz,cos(x2),Z2);
```

```

dotdotalfaz=subs(dotdotalfaz,x4*sin(x2),Z4);
dotdotalfaz=subs(dotdotalfaz,sin(x2),Z1);
%
[abbrdotdotalfaz,P2]=subexpr(dotdotalfaz,P);
sievadotdotalfaz=simplify(abbrdotdotalfaz);
%pretty(sievadotdotalfaz);
nim1=2*(Rm*P1*Lp^2*Lr^2*Mp^2*Z2^2+Lp^2*Mp+4*Jp);
nim2=0.25*P2*Lp^2*Mp^2*Lr^2*Z2^2+0.25*Lp^2*Mp+Jp;
%

% Parametrit Quanserilta:
% Motor
% Resistance
Rm = 8.4;
% Current-torque (N-m/A)
kt = 0.042;
% Back-emf constant (V-s/rad)
km = 0.042;
%
% Rotary Arm
% Mass (kg)
Mr = 0.095;
% Total length (m)
Lr = 0.085;
% Moment of inertia about pivot (kg-m^2)
Jr = Mr*Lr^2/12;
% Equivalent Viscous Damping Coefficient (N-m-s/rad)
Dr = 0.0015;
%
% Pendulum Link
% Mass (kg)
Mp = 0.024;
% Total length (m)
Lp = 0.129;
% Moment of inertia about pivot (kg-m^2)
Jp = Mp*Lp^2/12;
% Equivalent Viscous Damping Coefficient (N-m-s/rad)
Dp = 0.0005;
% Gravity Constant
g = 9.81;
%

```

```

% Sumea malli
% Parametrit
a=pi/4; % säätötavoite -pi/4 .. pi/4
b=6.2; % prosessikokeista
c=8; % prosessikokeista
% Sumeat ehtomuuttujat z_i
minz1 = sin(-a)/(-a);
maxz1 = 1;
minz2 = cos( a);
maxz2 = 1;
syms x y;
f(x,y)=y*sin(x)*cos(x);
%ezmesh(f, [(-a), a, (-b), b]);
minz3 = -3.1; % from figure
maxz3 = 3.1; % from figure
f(x,y)=y*sin(x);
%figure
%ezmesh(f, [(-a), a, (-c), c]);
minz4 = -5.657 % from figure
maxz4 = 5.657; % from figure
%
% Lokaalimallit:
Aat=cell(16,1);
Beet=cell(16,1);
vaihtoehdot=[maxz1 maxz2 maxz3 maxz4;
              maxz1 maxz2 maxz3 minz4;
              maxz1 maxz2 minz3 maxz4;
              maxz1 maxz2 minz3 minz4;
              maxz1 minz2 maxz3 maxz4;
              maxz1 minz2 maxz3 minz4;
              maxz1 minz2 minz3 maxz4;
              maxz1 minz2 minz3 minz4;
              minz1 maxz2 maxz3 maxz4;
              minz1 maxz2 maxz3 minz4;
              minz1 maxz2 minz3 maxz4;
              minz1 maxz2 minz3 minz4;
              minz1 minz2 maxz3 maxz4;
              minz1 minz2 maxz3 minz4;
              minz1 minz2 minz3 maxz4;
              minz1 minz2 minz3 minz4];
for k=1:16
    Z1=vaihtoehdot(k,1);
    Z2=vaihtoehdot(k,2);
    Z3=vaihtoehdot(k,3);
    Z4=vaihtoehdot(k,4);
    dotx1=[0 0 1 0];
    dotx2=[0 0 0 1];
    dotx3=[0 (-P1*2*Lp^2*Lr*Mp^2*Rm*Z1*Z2*g)/nim1 (-P1*8*Jp*km^2-
P1*2*Lp^2*Mp*km^2-P1*8*Dr*Jp*Rm-
P1*2*Dr*Lp^2*Mp*Rm+P1*Lp^3*Lr*Mp^2*Rm*Z1*Z3-
P1*4*Dp*Lp*Lr*Mp*Rm*Z2)/nim1 (-P1*Lp^4*Mp^2*Rm*Z3-
P1*Lp^3*Lr*Mp^2*Rm*Z4-P1*4*Jp*Lp^2*Mp*Rm*Z3-
P1*4*Jp*Lp*Lr*Mp*Rm*Z4)/nim1];
    b3=(P1*8*Jp*km+P1*2*Lp^2*Mp*km)/nim1;
    dotx4=[0 (0 -0.5*Lp*Mp*g*Z1)/nim2 (0.25*Lp^2*Mp*Z3-
Dp+0.5*Lp*Lr*Mp*P2*Z2*Dr+0.5*Lp*Lr*Mp*Z2*km^2/Rm)/nim2
(0.5*Lp*Lr*Mp*P2*Z2*0.5*Mp*Lp^2*Z3+0.5*Lp*Lr*Mp*P2*Z2*0.5*Lr*Mp*Lp*Z4)
/nim2];

```



```

b4=(-0.5*Lp*Lr*Mp*P2*Z2*km/Rm)/nim2;
dotx3=subs(dotx3);
dotx4=subs(dotx4);
dotx3=eval(dotx3);
dotx4=eval(dotx4);
dotx=[dotx1; dotx2; dotx3; dotx4];
b3=subs(b3);
b4=subs(b4);
b3=eval(b3);
b4=eval(b4);
bee=[0 0 b3 b4]';
Aat{k}=dotx;
Beet{k}=bee;
end
% Mallien parametrit syötetään FIS-tiedostoon
malli=readfis('Heilurimalli.fis')
%
% Sumeat säätimet
Koot=cell(16,1);
Riccatit=cell(16,1);
eigs=cell(16,1);
% 16 lokaalia LQR-säädintä
for k=1:16
    A=Aat{k};
    B=Beet{k};
    Q=diag([5 1 1 1]); % Quanser
    % Q=diag([10 20 10 1]); % Fairus et al.
    R=1; % yksi säätösuure
    [K,S,e]=lqr(A,B,Q,R); % u=-Kx
    Koot{k}=K;
    Riccatit{k}=S;
    eigs{k}=e;
end
saadin=readfis('LQRSaadin.fis')
%
```

```

% Vertailuksi laitteistovalmistajan malli

% Quanser State-Space Model

% Find Total Inertia
Jt = Jr*Jp + Mp*(Lp/2)^2*Jr + Jp*Mp*Lr^2;

% State Space Representation
A = [0 0 1 0;
      0 0 0 1;
      0 Mp^2*(Lp/2)^2*Lr*g/Jt -Dr*(Jp+Mp*(Lp/2)^2)/Jt
      Mp*(Lp/2)*Lr*Dp/Jt;
      0 -Mp*g*(Lp/2)*(Jr+Mp*Lr^2)/Jt Mp*(Lp/2)*Lr*Dr/Jt
      -Dp*(Jr+Mp*Lr^2)/Jt];

B = [0; 0; (Jp+Mp*(Lp/2)^2)/Jt; -Mp*(Lp/2)*Lr/Jt];
C = eye(2,4);
D = zeros(2,1);

% Add actuator dynamics
B = km * B / Rm;
A(3,3) = A(3,3) - km*km/Rm*B(3);
A(4,3) = A(4,3) - km*km/Rm*B(4);

% Build optimal gain K, Riccati solution S, eigenvalues e
Q=diag([5 1 1 1]); % Arm angle change is punished
R=1;
[K,S,e]=lqr(A,B,Q,R); % u=-Kx
% K=[-2 35 -1.5 3]; % from Quanser

```

B. HEILURIMALLIN FIS-TIEDOSTO

Heilurimallin sisältävä tiedosto esitetään opiskelijoiden avuksi. Tiedosto määrittelee su-
mean järjestelmän, joka käyttää sisäänmenoina sector nonlinearity -apumuuttujia z_i ja
tuottaa ulostuloinaan järjestelmän tilaesityksen $\dot{x} = Ax + Bu$ A - ja B -matriisien muuttu-
vat parametrit. Jokaiselle sisäänmenolle on kaksi puolisuunnikkaan muotoista jäsenyys-
funktioita.

```
[System]
Name='Heilurimalli'
Type='sugeno'
Version=2.0
NumInputs=4
NumOutputs=8
NumRules=16
AndMethod='prod'
OrMethod='probor'
ImpMethod='prod'
AggMethod='sum'
DefuzzMethod='wtaver'

[Input1]
Name='Z1'
Range=[0.9 1]
NumMFs=2
MF1='M12': 'trapmf', [0.8045 0.8888 0.9 1]
MF2='M11': 'trapmf', [0.9 0.999867724867725 1.02 1.1]

[Input2]
Name='Z2'
Range=[0.701 1]
NumMFs=2
MF1='M22': 'trapmf', [0.4316 0.6714 0.701 1]
MF2='M21': 'trapmf', [0.701 1 1.03 1.266]

[Input3]
Name='Z3'
Range=[-10 10]
NumMFs=2
MF1='M32': 'trapmf', [-28.4 -12.31 -3.1 3.1]
MF2='M31': 'trapmf', [-3.1 3.1 12 28]

[Input4]
Name='Z4'
Range=[-10 10]
NumMFs=2
MF1='M42': 'trapmf', [-28 -12 -5.657 5.657]
MF2='M41': 'trapmf', [-5.657 5.657 12 28]
```

```
[Output1]
Name='A32'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [-95.2951]
MF2='mf2': 'constant', [-95.2951]
MF3='mf3': 'constant', [-95.2951]
MF4='mf4': 'constant', [-95.2951]
MF5='mf5': 'constant', [-114.7647]
MF6='mf6': 'constant', [-114.7647]
MF7='mf7': 'constant', [-114.7647]
MF8='mf8': 'constant', [-114.7647]
MF9='mf9': 'constant', [-85.7958]
MF10='mf10': 'constant', [-85.7958]
MF11='mf11': 'constant', [-85.7958]
MF12='mf12': 'constant', [-85.7958]
MF13='mf13': 'constant', [-103.3245]
MF14='mf14': 'constant', [-103.3245]
MF15='mf15': 'constant', [-103.3245]
MF16='mf16': 'constant', [-103.3245]
```

```
[Output2]
Name='A33'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [-12.0522]
MF2='mf2': 'constant', [-12.0522]
MF3='mf3': 'constant', [-15.9369]
MF4='mf4': 'constant', [-15.9369]
MF5='mf5': 'constant', [-18.9615]
MF6='mf6': 'constant', [-18.9615]
MF7='mf7': 'constant', [-25.5777]
MF8='mf8': 'constant', [-25.5777]
MF9='mf9': 'constant', [-12.2458]
MF10='mf10': 'constant', [-12.2458]
MF11='mf11': 'constant', [-15.7432]
MF12='mf12': 'constant', [-15.7432]
MF13='mf13': 'constant', [-19.2913]
MF14='mf14': 'constant', [-19.2913]
MF15='mf15': 'constant', [-25.2479]
MF16='mf16': 'constant', [-25.2479]
```

```
[Output3]
Name='A34'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [-8.6563]
MF2='mf2': 'constant', [0.7956]
MF3='mf3': 'constant', [-0.7956]
MF4='mf4': 'constant', [8.6563]
MF5='mf5': 'constant', [-14.7429]
MF6='mf6': 'constant', [1.3550]
MF7='mf7': 'constant', [-1.3550]
MF8='mf8': 'constant', [14.7429]
MF9='mf9': 'constant', [-8.6563]
MF10='mf10': 'constant', [0.7956]
MF11='mf11': 'constant', [-0.7956]
MF12='mf12': 'constant', [8.6563]
MF13='mf13': 'constant', [-14.7429]
```

```
MF14='mf14': 'constant', [1.3550]
MF15='mf15': 'constant', [-1.3550]
MF16='mf16': 'constant', [14.7429]
```

```
[Output4]
```

```
Name='A42'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [-72.9372]
MF2='mf2': 'constant', [-72.9372]
MF3='mf3': 'constant', [-72.9372]
MF4='mf4': 'constant', [-72.9372]
MF5='mf5': 'constant', [-88.9795]
MF6='mf6': 'constant', [-88.9795]
MF7='mf7': 'constant', [-88.9795]
MF8='mf8': 'constant', [-88.9795]
MF9='mf9': 'constant', [-65.6666]
MF10='mf10': 'constant', [-65.6666]
MF11='mf11': 'constant', [-65.6666]
MF12='mf12': 'constant', [-65.6666]
MF13='mf13': 'constant', [-80.1097]
MF14='mf14': 'constant', [-80.1097]
MF15='mf15': 'constant', [-80.1097]
MF16='mf16': 'constant', [-80.1097]
```

```
[Output5]
```

```
Name='A43'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [3.1960]
MF2='mf2': 'constant', [3.1960]
MF3='mf3': 'constant', [0.2227]
MF4='mf4': 'constant', [0.2227]
MF5='mf5': 'constant', [2.4301]
MF6='mf6': 'constant', [2.4301]
MF7='mf7': 'constant', [-1.1971]
MF8='mf8': 'constant', [-1.1971]
MF9='mf9': 'constant', [3.1960]
MF10='mf10': 'constant', [3.1960]
MF11='mf11': 'constant', [0.2227]
MF12='mf12': 'constant', [0.2227]
MF13='mf13': 'constant', [2.4301]
MF14='mf14': 'constant', [2.4301]
MF15='mf15': 'constant', [-1.1971]
MF16='mf16': 'constant', [-1.1971]
```

```
[Output6]
```

```
Name='A44'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [3.7363]
MF2='mf2': 'constant', [-0.3434]
MF3='mf3': 'constant', [0.3434]
MF4='mf4': 'constant', [-3.7363]
MF5='mf5': 'constant', [3.2232]
MF6='mf6': 'constant', [-0.2962]
MF7='mf7': 'constant', [0.2962]
MF8='mf8': 'constant', [-3.2232]
MF9='mf9': 'constant', [3.7363]
```

```
MF10='mf10': 'constant', [-0.3434]
MF11='mf11': 'constant', [0.3434]
MF12='mf12': 'constant', [-3.7363]
MF13='mf13': 'constant', [3.2232]
MF14='mf14': 'constant', [-0.2962]
MF15='mf15': 'constant', [0.2962]
MF16='mf16': 'constant', [-3.2232]
```

[Output7]

```
Name='B3'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [31.7454]
MF2='mf2': 'constant', [31.7454]
MF3='mf3': 'constant', [31.7454]
MF4='mf4': 'constant', [31.7454]
MF5='mf5': 'constant', [54.0671]
MF6='mf6': 'constant', [54.0671]
MF7='mf7': 'constant', [54.0671]
MF8='mf8': 'constant', [54.0671]
MF9='mf9': 'constant', [31.7454]
MF10='mf10': 'constant', [31.7454]
MF11='mf11': 'constant', [31.7454]
MF12='mf12': 'constant', [31.7454]
MF13='mf13': 'constant', [54.0671]
MF14='mf14': 'constant', [54.0671]
MF15='mf15': 'constant', [54.0671]
MF16='mf16': 'constant', [54.0671]
```

[Output8]

```
Name='B4'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [-13.7023]
MF2='mf2': 'constant', [-13.7023]
MF3='mf3': 'constant', [-13.7023]
MF4='mf4': 'constant', [-13.7023]
MF5='mf5': 'constant', [-11.8203]
MF6='mf6': 'constant', [-11.8203]
MF7='mf7': 'constant', [-11.8203]
MF8='mf8': 'constant', [-11.8203]
MF9='mf9': 'constant', [-13.7023]
MF10='mf10': 'constant', [-13.7023]
MF11='mf11': 'constant', [-13.7023]
MF12='mf12': 'constant', [-13.7023]
MF13='mf13': 'constant', [-11.8203]
MF14='mf14': 'constant', [-11.8203]
MF15='mf15': 'constant', [-11.8203]
MF16='mf16': 'constant', [-11.8203]
```

[Rules]

```
1 1 1 1, 1 1 1 1 1 1 1 1 (1) : 1
1 1 1 2, 2 2 2 2 2 2 2 2 (1) : 1
1 1 2 1, 3 3 3 3 3 3 3 3 (1) : 1
1 1 2 2, 4 4 4 4 4 4 4 4 (1) : 1
1 2 1 1, 5 5 5 5 5 5 5 5 (1) : 1
1 2 1 2, 6 6 6 6 6 6 6 6 (1) : 1
1 2 2 1, 7 7 7 7 7 7 7 7 (1) : 1
1 2 2 2, 8 8 8 8 8 8 8 8 (1) : 1
```

2 1 1 1, 9 9 9 9 9 9 9 9 (1) : 1
2 1 1 2, 10 10 10 10 10 10 10 10 (1) : 1
2 1 2 1, 11 11 11 11 11 11 11 11 (1) : 1
2 1 2 2, 12 12 12 12 12 12 12 12 (1) : 1
2 2 1 1, 13 13 13 13 13 13 13 13 (1) : 1
2 2 1 2, 14 14 14 14 14 14 14 14 (1) : 1
2 2 2 1, 15 15 15 15 15 15 15 15 (1) : 1
2 2 2 2, 16 16 16 16 16 16 16 16 (1) : 1

C. SÄÄTIMEN FIS-TIEDOSTO

Säätimen sisältävä tiedosto on samantyyppinen kuin mallin tiedosto. Säätimen sisäänmenoina ovat samat z_i kuin mallillakin. Nyt ulostuloina ovat kuitenkin tilatakaisinkytketyn säätimen vahvistusparametrit K_j . Jäsenyysfunktiot ovat puolisuunnikkaan muotoisia ja niitä on kaksi jokaista sisäänmenoa kohden kuten mallillakin.

```
[System]
Name='LQRSaadin'
Type='sugeno'
Version=2.0
NumInputs=4
NumOutputs=4
NumRules=16
AndMethod='prod'
OrMethod='probor'
ImpMethod='prod'
AggMethod='sum'
DefuzzMethod='wtaver'

[Input1]
Name='Z1'
Range=[0.9 1]
NumMFs=2
MF1='M12': 'trapmf', [0.8045 0.8888 0.9 1]
MF2='M11': 'trapmf', [0.9 0.999867724867725 1.02 1.1]

[Input2]
Name='Z2'
Range=[0.701 1]
NumMFs=2
MF1='M22': 'trapmf', [0.4316 0.6714 0.701 1]
MF2='M21': 'trapmf', [0.701 1 1.03 1.266]

[Input3]
Name='Z3'
Range=[-10 10]
NumMFs=2
MF1='M32': 'trapmf', [-28.4 -12.31 -3.1 3.1]
MF2='M31': 'trapmf', [-3.1 3.1 12 28]

[Input4]
Name='Z4'
Range=[-10 10]
NumMFs=2
MF1='M42': 'trapmf', [-28 -12 -5.657 5.657]
MF2='M41': 'trapmf', [-5.657 5.657 12 28]

[Output1]
```



```
Name='K1'  
Range=[0 1]  
NumMFs=16  
MF1='mf1': 'constant', [2.2361]  
MF2='mf2': 'constant', [2.2361]  
MF3='mf3': 'constant', [2.2361]  
MF4='mf4': 'constant', [2.2361]  
MF5='mf5': 'constant', [2.2361]  
MF6='mf6': 'constant', [2.2361]  
MF7='mf7': 'constant', [2.2361]  
MF8='mf8': 'constant', [2.2361]  
MF9='mf9': 'constant', [2.2361]  
MF10='mf10': 'constant', [2.2361]  
MF11='mf11': 'constant', [2.2361]  
MF12='mf12': 'constant', [2.2361]  
MF13='mf13': 'constant', [2.2361]  
MF14='mf14': 'constant', [2.2361]  
MF15='mf15': 'constant', [2.2361]  
MF16='mf16': 'constant', [2.2361]
```

[Output2]

```
Name='K2'  
Range=[0 1]  
NumMFs=16  
MF1='mf1': 'constant', [-11.7114]  
MF2='mf2': 'constant', [-11.1063]  
MF3='mf3': 'constant', [-6.5337]  
MF4='mf4': 'constant', [-6.6484]  
MF5='mf5': 'constant', [-10.6180]  
MF6='mf6': 'constant', [-10.0422]  
MF7='mf7': 'constant', [-4.7673]  
MF8='mf8': 'constant', [-4.9552]  
MF9='mf9': 'constant', [-10.9004]  
MF10='mf10': 'constant', [-10.3307]  
MF11='mf11': 'constant', [-6.1668]  
MF12='mf12': 'constant', [-6.2775]  
MF13='mf13': 'constant', [-9.7998]  
MF14='mf14': 'constant', [-9.2624]  
MF15='mf15': 'constant', [-4.5252]  
MF16='mf16': 'constant', [-4.7023]
```

[Output3]

```
Name='K3'  
Range=[0 1]  
NumMFs=16  
MF1='mf1': 'constant', [0.6717]  
MF2='mf2': 'constant', [0.6784]  
MF3='mf3': 'constant', [0.5445]  
MF4='mf4': 'constant', [0.5420]  
MF5='mf5': 'constant', [0.6637]  
MF6='mf6': 'constant', [0.6704]  
MF7='mf7': 'constant', [0.5917]  
MF8='mf8': 'constant', [0.5846]  
MF9='mf9': 'constant', [0.6595]  
MF10='mf10': 'constant', [0.6671]  
MF11='mf11': 'constant', [0.5455]  
MF12='mf12': 'constant', [0.5430]  
MF13='mf13': 'constant', [0.6542]  
MF14='mf14': 'constant', [0.6615]
```

```
MF15='mf15': 'constant', [0.5942]
MF16='mf16': 'constant', [0.5871]
```

```
[Output4]
```

```
Name='K4'
Range=[0 1]
NumMFs=16
MF1='mf1': 'constant', [-0.9777]
MF2='mf2': 'constant', [-0.6909]
MF3='mf3': 'constant', [-1.0145]
MF4='mf4': 'constant', [-0.7498]
MF5='mf5': 'constant', [-0.9726]
MF6='mf6': 'constant', [-0.6814]
MF7='mf7': 'constant', [-1.0188]
MF8='mf8': 'constant', [-0.7531]
MF9='mf9': 'constant', [-0.9804]
MF10='mf10': 'constant', [-0.6925]
MF11='mf11': 'constant', [-1.0072]
MF12='mf12': 'constant', [-0.7426]
MF13='mf13': 'constant', [-0.9894]
MF14='mf14': 'constant', [-0.6958]
MF15='mf15': 'constant', [-1.0147]
MF16='mf16': 'constant', [-0.7490]
```

```
[Rules]
```

```
1 1 1 1, 1 1 1 1 (1) : 1
1 1 1 2, 2 2 2 2 (1) : 1
1 1 2 1, 3 3 3 3 (1) : 1
1 1 2 2, 4 4 4 4 (1) : 1
1 2 1 1, 5 5 5 5 (1) : 1
1 2 1 2, 6 6 6 6 (1) : 1
1 2 2 1, 7 7 7 7 (1) : 1
1 2 2 2, 8 8 8 8 (1) : 1
2 1 1 1, 9 9 9 9 (1) : 1
2 1 1 2, 10 10 10 10 (1) : 1
2 1 2 1, 11 11 11 11 (1) : 1
2 1 2 2, 12 12 12 12 (1) : 1
2 2 1 1, 13 13 13 13 (1) : 1
2 2 1 2, 14 14 14 14 (1) : 1
2 2 2 1, 15 15 15 15 (1) : 1
2 2 2 2, 16 16 16 16 (1) : 1
```

D. SUMEAN JÄRJESTELMÄN STABIILIUDEN TODISTAMINEN

Aiemmin luvussa 2.3.1 on esitetty, että parallel distributed -menetelmän T-S mallisen järjestelmän globaali stabiilius voidaan todeta, jos löydetään yksi matriisi G , joka toteuttaa kaikille $i=1,\dots,16$ ja $j=1,\dots,16$ yhdistelmille

$$G({}^iA-{}^iB{}^jK)+({}^iA-{}^iB{}^jK)^T G < 0,$$

missä matriisit A ja B ovat järjestelmän tilaesityksen matriisit ja K säätimen vahvistus. Tämä johtaa 16 lokaalin mallin ja 16 lokaalin säätimen tapauksessa 256:een epäyhtälöön. Osa ratkaisun Matlab-koodista on esitetty alla.

```
% Furuta-simulaatiomallin stabiiliuden todistaminen

% Ratkaistaan kaikille G*(A_i-B_i*K_j)+(A_i-B_i*K_j)'*G<0
% yhteinen G. Mikäli G on olemassa, suljetun järjestelmän stabiilius
on
% todistettu.
setlmis([]);

G=lmivar(2,[4 4]);

lmiterm([1 1 1 G],1,(A1-B1*K1),'s');% LMI #1: G*(A1-B1*K1)+(A1-B1*K1)'*G
lmiterm([2 1 1 G],1,(A1-B1*K2),'s');% LMI #2: G*(A1-B1*K2)+(A1-B1*K2)'*G
lmiterm([3 1 1 G],1,(A1-B1*K3),'s');% LMI #3: G*(A1-B1*K3)+(A1-B1*K3)'*G
lmiterm([4 1 1 G],1,(A1-B1*K4),'s');% LMI #4: G*(A1-B1*K4)+(A1-B1*K4)'*G
lmiterm([5 1 1 G],1,(A1-B1*K5),'s');% LMI #5: G*(A1-B1*K5)+(A1-B1*K5)'*G
lmiterm([6 1 1 G],1,(A1-B1*K6),'s');% LMI #6: G*(A1-B1*K6)+(A1-B1*K6)'*G
lmiterm([7 1 1 G],1,(A1-B1*K7),'s');% LMI #7: G*(A1-B1*K7)+(A1-B1*K7)'*G
lmiterm([8 1 1 G],1,(A1-B1*K8),'s');% LMI #8: G*(A1-B1*K8)+(A1-B1*K8)'*G
lmiterm([9 1 1 G],1,(A1-B1*K9),'s');% LMI #9: G*(A1-B1*K9)+(A1-B1*K9)'*G
lmiterm([10 1 1 G],1,(A1-B1*K10),'s');% LMI #10: G*(A1-B1*K10)+(A1-B1*K10)'*G
lmiterm([11 1 1 G],1,(A1-B1*K11),'s');% LMI #11: G*(A1-B1*K11)+(A1-B1*K11)'*G
lmiterm([12 1 1 G],1,(A1-B1*K12),'s');% LMI #12: G*(A1-B1*K12)+(A1-B1*K12)'*G
lmiterm([13 1 1 G],1,(A1-B1*K13),'s');% LMI #13: G*(A1-B1*K13)+(A1-B1*K13)'*G
lmiterm([14 1 1 G],1,(A1-B1*K14),'s');% LMI #14: G*(A1-B1*K14)+(A1-B1*K14)'*G
lmiterm([15 1 1 G],1,(A1-B1*K15),'s');% LMI #15: G*(A1-B1*K15)+(A1-B1*K15)'*G
lmiterm([16 1 1 G],1,(A1-B1*K16),'s');% LMI #16: G*(A1-B1*K16)+(A1-B1*K16)'*G
lmiterm([17 1 1 G],1,(A2-B2*K1),'s');% LMI #17: G*(A2-B2*K1)+(A2-B2*K1)'*G
lmiterm([18 1 1 G],1,(A2-B2*K2),'s');% LMI #18: G*(A2-B2*K2)+(A2-B2*K2)'*G
lmiterm([19 1 1 G],1,(A2-B2*K3),'s');% LMI #19: G*(A2-B2*K3)+(A2-B2*K3)'*G
lmiterm([20 1 1 G],1,(A2-B2*K4),'s');% LMI #20: G*(A2-B2*K4)+(A2-B2*K4)'*G
lmiterm([21 1 1 G],1,(A2-B2*K5),'s');% LMI #21: G*(A2-B2*K5)+(A2-B2*K5)'*G
lmiterm([22 1 1 G],1,(A2-B2*K6),'s');% LMI #22: G*(A2-B2*K6)+(A2-B2*K6)'*G
lmiterm([23 1 1 G],1,(A2-B2*K7),'s');% LMI #23: G*(A2-B2*K7)+(A2-B2*K7)'*G
lmiterm([24 1 1 G],1,(A2-B2*K8),'s');% LMI #24: G*(A2-B2*K8)+(A2-B2*K8)'*G
lmiterm([25 1 1 G],1,(A2-B2*K9),'s');% LMI #25: G*(A2-B2*K9)+(A2-B2*K9)'*G
lmiterm([26 1 1 G],1,(A2-B2*K10),'s');% LMI #26: G*(A2-B2*K10)+(A2-B2*K10)'*G
```

```

lmiterm([27 1 1 G],1,(A2-B2*K11),'s');% LMI #27: G*(A2-B2*K11)+(A2-B2*K11)'*G
lmiterm([28 1 1 G],1,(A2-B2*K12),'s');% LMI #28: G*(A2-B2*K12)+(A2-B2*K12)'*G
lmiterm([29 1 1 G],1,(A2-B2*K13),'s');% LMI #29: G*(A2-B2*K13)+(A2-B2*K13)'*G

      :

lmiterm([113 1 1 G],1,(A8-B8*K1),'s');% LMI #113: G*(A8-B8*K1)+(A8-B8*K1)'*G
lmiterm([114 1 1 G],1,(A8-B8*K2),'s');% LMI #114: G*(A8-B8*K2)+(A8-B8*K2)'*G
lmiterm([115 1 1 G],1,(A8-B8*K3),'s');% LMI #115: G*(A8-B8*K3)+(A8-B8*K3)'*G
lmiterm([116 1 1 G],1,(A8-B8*K4),'s');% LMI #116: G*(A8-B8*K4)+(A8-B8*K4)'*G
lmiterm([117 1 1 G],1,(A8-B8*K5),'s');% LMI #117: G*(A8-B8*K5)+(A8-B8*K5)'*G
lmiterm([118 1 1 G],1,(A8-B8*K6),'s');% LMI #118: G*(A8-B8*K6)+(A8-B8*K6)'*G
lmiterm([119 1 1 G],1,(A8-B8*K7),'s');% LMI #119: G*(A8-B8*K7)+(A8-B8*K7)'*G
lmiterm([120 1 1 G],1,(A8-B8*K8),'s');% LMI #120: G*(A8-B8*K8)+(A8-B8*K8)'*G
lmiterm([121 1 1 G],1,(A8-B8*K9),'s');% LMI #121: G*(A8-B8*K9)+(A8-B8*K9)'*G

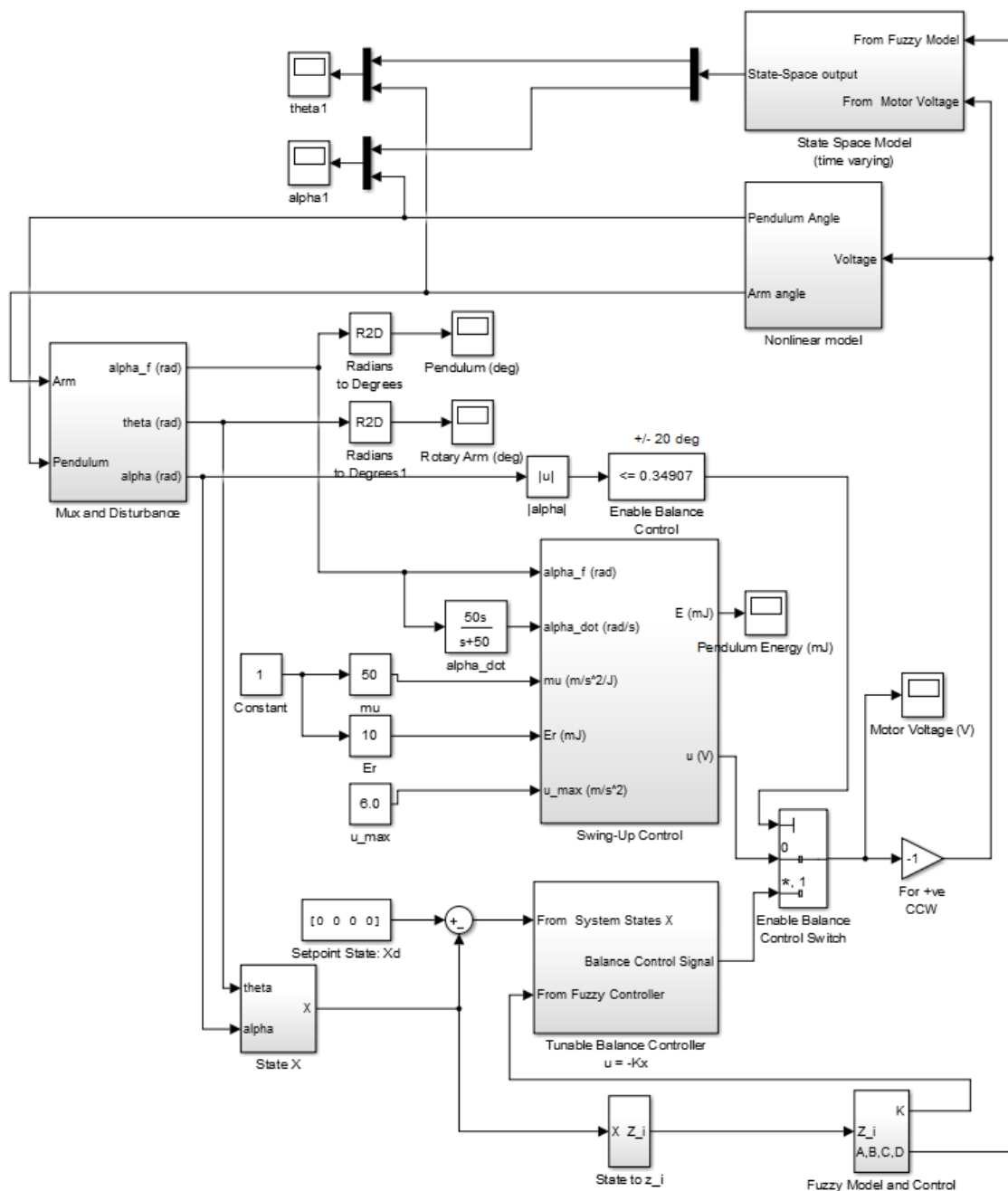
      :

lmiterm([253 1 1 G],1,(A16-B16*K13),'s');% LMI #253: G*(A16-B16*K13)+(A16-
B16*K13)'*G
lmiterm([254 1 1 G],1,(A16-B16*K14),'s');% LMI #254: G*(A16-B16*K14)+(A16-
B16*K14)'*G
lmiterm([255 1 1 G],1,(A16-B16*K15),'s');% LMI #255: G*(A16-B16*K15)+(A16-
B16*K15)'*G
lmiterm([256 1 1 G],1,(A16-B16*K16),'s');% LMI #256: G*(A16-B16*K16)+(A16-
B16*K16)'*G

sumeastab=getlmis;
[tmin,xfeas] = feasp(sumeastab);
G = dec2mat(sumeastab,xfeas,G)

```

E. SIMULAATTORI SWING-UPILLA JA SUMEALLA SÄÄTIMELLÄ



Kuva 21: Simulaattori sumealla säätimellä ja swing-upilla. Sumean ja epälineaarisen mallin vertailu.

Kuvassa 21 esitetään Simulink-malli, jossa yhdistetään tasapainottava sumea säädin ja järjestelmän energiataseeseen perustuva swing-up-säädin. Kytkennässä on lisäksi sumean ja epälineaarisen mallin vertailun mahdollisuus.