

**LEGO – ROBOTTI SÄÄTÖTEKNIIKAN OPETUKSEN
DEMONSTRAATIOVÄLINEENÄ**

**A Lego robot implemented as an educational device for
control engineering**

Pekka Vanhanen

TIIVISTELMÄ

Lappeenrannan teknillinen yliopisto

LUT School of Energy Systems

Sähkötekniikka

Pekka Vanhanen

Lego -robotti säätötekniikan opetuksen demonstraatiovälineenä

2019

Kandidaatintyö.

23 s.

Tarkastajat: TkT Niko Nevaranta, DI Pekko Jaatinen

Säätötekniikan teorianopetusta tukemassa on hyödyllistä käyttää rinnalla havainnollistavia ja konkreettisia esimerkkejä. Tässä työssä tarkastellaan säätötekniikan opetukseen sopivaa, luonnostaan epästabiilia kaksipyöräistä Lego Mindstorms EV3 – robottia ja työn tavoitteena on kehittää balansoiva säätö Lego - robotin rungon sivuttaissuuntaiselle kallistuskulmalle estäen robotin kaatumisen. Robotin dynamiikka mallinnetaan tilayhtälöiden avulla, jotka sisältää systeemi -, tulo -, lähtö - ja myötäkytkennän matriisit. Työn tuloksena on Matlab – koodi, joka sisältää Lego – robotin mallinnuksen sekä työssä analysoidaan järjestelmää tunnetuilla menetelmillä; avoimen – ja suljetun – sekä säädetyn piirin askelvasteilla ja napa – nollakohtadiagrammin avulla. Työssä havainnollistetaan Simulink - simulointimalli, joka koostuu mm. edellä mainituista Matlabin matriiseja hyödyntävästä tilayhtälöt – lohkoista, PID - säätimestä ja takaisinkytkennästä. Työn tarkastelu rajataan robotin stabiloivaan säätöön ja kaksi muuta liikeakselia, joilla käännetään robottia ja liikutetaan sitä eteen – tai taaksepäin, jätetään tarkastelun ulkopuolelle.

ABSTRACT

Lappeenranta University of Technology

LUT School of Energy Systems

Electrical Engineering

Pekka Vanhanen

A Lego robot implemented as an educational device for control engineering

2019

Bachelor's Thesis.

23 p.

Examiners: D.Sc. Niko Nevaranta, M.Sc. Pekko Jaatinen

It is useful to use illustrative and concrete examples alongside supporting theoretical teaching in control technology. This thesis covers the unstable two-wheeled Lego Mindstorms EV3 robot, which is suitable for the teaching of control technology, and aims to develop balancing closed loop control law for the lateral tilt angle of the Lego robot body, preventing the robot from falling. Robot dynamics are modelled by state space equations that include system, input, output, and positive feedback matrices. The result of the thesis is the Matlab code, which includes the modelling of the Lego robot, and this thesis also analyzes the system with known methods; open - and closed - as well as closed loop step responses and pole - zero diagram. The thesis illustrates the Simulink simulation model, which consists of e.g. the above-mentioned Matlab matrix-based state space equations - block, PID controller and feedback. Scope of this thesis is limited to stabilizing the robot. Two other motion axes – one that rotates the robot and another that drives it forward or backward - are excluded.

SISÄLLYSLUETTELO

Käytetyt merkinnät ja lyhenteet

| | | |
|-----|--------------------------------------|----|
| 1. | Johdanto..... | 7 |
| 1.1 | Tavoite ja tutkimuskysymykset..... | 8 |
| 2. | Lego mindstorms Ev3..... | 9 |
| 2.1 | Mallinnus..... | 9 |
| 2.2 | Järjestelmän parametrit..... | 13 |
| 2.3 | Analysointi | 14 |
| 3. | Säätimen suunnittelu ja testaus..... | 16 |
| 3.1 | Simulointimalli | 16 |
| 3.2 | Tulokset | 17 |
| 4. | Yhteenveto/johtopäätökset | 21 |
| | Lähteet | 23 |

Liitteet

Liite 1. Matlab - koodi Lego – robotin mallinnuksesta.

KÄYTETYT MERKINNÄT JA LYHENTEET

| | |
|----------|--|
| ARM | Advanced RISC Machines |
| PID | Proportional – Integral – Derivative |
| PWM | Pulse Width Modulation |
| RAM | Random Access Memory |
| SD | Secure Digital |
| USB | Universal Serial Bus |
| | |
| A_1 | systemimatriisi |
| A_2 | systemimatriisi |
| B_1 | tulomatriisi |
| B_2 | tulomatriisi |
| C | lähtömatriisi |
| D | myötä kytkennän matriisi |
| E | matriisi |
| f_m | kitkakerroin rungon ja tasavirtamoottorin välillä |
| f_w | kitkakerroin maan ja renkaan välillä |
| g | putoamiskiihtyvyys |
| i | imaginääriyksikkö |
| I | vakio |
| j | matriisin rivi |
| J | vakio |
| J_m | tasavirtamoottorin roottorin hitausmomentti |
| J_w | renkaan hitausmomentti |
| J_Φ | rungon kaltevuuskulman (yx – tasossa) hitausmomentti |
| J_Ψ | hitaismomentti rungon kallistuskulman suhteen |
| k | matriisin sarake |
| K | vakio |
| K_b | tasavirtamoottorin nopeusvakio |
| K_D | säätimen derivointivahvistus |
| K_I | säätimen integrointivahvistus |
| K_P | säätimen vahvistus |

| | |
|---------------------|--|
| K_t | tasavirtamoottorin vääntömomenttivakio |
| L | etäisyys renkaan akselilta robotin painopisteeseen |
| m | renkaan massa |
| M | robotin rungon massa |
| n | vaihteen välityssuhde |
| R | renkaan säde |
| R_m | tasavirtamoottorin resistanssi |
| t | aika |
| u | tulo, tulovektori |
| W | rungon leveys |
| v_l | vasemman tasavirtamoottorin jänniteohje |
| v_r | oikean tasavirtamoottorin jänniteohje |
| x_1 | tilavektori |
| \dot{x}_1 | tilavektorin derivaatta ajan suhteen |
| x_2 | tilavektori |
| \dot{x}_2 | tilavektorin derivaatta ajan suhteen |
| y | lähtö |
| α | vakio |
| β | vakio |
| θ | renkaan kiertymiskulma |
| $\dot{\theta}$ | renkaan kulmanopeus |
| ϕ | rungon kiertymiskulma |
| $\dot{\phi}$ | rungon kulmanopeus kiertymiskulman suhteen |
| Ψ | rungon kallistuskulma |
| Ψ_{ref} | rungon kallistuskulman referenssiarvo |
| $\dot{\Psi}$ | rungon kulmanopeus kallistuskulman suhteen |

1. JOHDANTO

Säätötekniikka on erityisen tärkeässä roolissa erilaisten prosessien ja järjestelmien ohjauksessa. Opetettaessa säätötekniikkaa teoriapainotteisen opetuksen tueksi on tärkeää saada havainnollistavia esimerkkejä. Nykysuuntaus on, että liikkeenohjauksen havainnollistamiseen perustuvia säädönopetusvälineitä, kuten erilaiset robotit, käytetään usein koulutustarkoituksessa niiden visuaalisuuden takia. Esimerkiksi Legolla on tähän tarkoitukseen Lego Mindstorms – robotti, jota on mahdollista muokata ja ohjelmoida erilaiseen tarpeisiin. Kuvassa 1.1 on esitettynä Lego – robotteja, joilla on tarkoitus havainnollistaa epästabiilin systeemin säätöä, tässä tapauksessa robotin tasapainotusta.



Kuva 1.1 Esimerkkejä kaksipyöräisistä ja itsebalansoituvista Lego Mindstorms EV3 – roboteista. (Valk 2016).

Lego - robotteja on käsitelty laajasti opinnäytetöissä sekä akateemisissa artikkeleissa. Yhdessä opinnäytetyössä on samaa aihetta tutkittu laajasti ensin johtamalla liikeyhtälöt robotin eri osille esim. renkaille, jonka jälkeen niistä kehitettiin siirtofunktio analyysia ja säädön viritystä varten. Epästabiili järjestelmä stabiloitiin PID – säätimellä, jota testattiin myös simulaatioilla sekä käytännön kokeilla. (Silfver 2018).

Eräissä aikaisemmassa tutkimuksessa on perehdytty laajasti EV3 – tyyppisen Lego - robotin siirtofunktion määrittämiseen kokeellisesti eri menetelmin. Tutkimuksen tuloksena on johdettu systeemiä kuvaava toisen- ja kolmannen kertaluokan siirtofunktio huomioimatta säätimen vaikutusta. (Norouzi Kandalan 2016).

Yksi tutkimus selvittää miten robotiikkaa käytetään opetuksessa ts. Lego Mindstorms – robotin käyttämisestä opetuksessa Yhdistyneen arabiemiirikunnan kouluissa ja kuinka opettajat voivat yhdistää tai lisätä robotiikkaa opetussuunnitelmaan. Tutkimus myös ehdottaa tehokkaita keinoja robotiikan käytöstä opetusvälineenä sekä antaa suosituksia oppimiskokemusten parantamiseksi ja kuinka se vaikuttaa oppilaiden mielenkiintoon tieteseen, teknologiaan sekä tekniikkaan ja matematiikkaan liittyvissä aineissa. (Afari 2017). Vastaavasti (Alimisis et al. 2016) käsittelee robotiikkaan liittyviä aiheita laajasti ja monipuolisesti mm. robotiikan soveltamisesta teoriassa ja käytännössä koulutuksessa.

Tässä kandidaatintyössä tarkastellaan kuvan 1.1 mukaista Lego – robottia sekä mallinnetaan robotti Matlab – ympäristössä ja suunnitellaan sille PID – pohjainen säätörakenne.

1.1 Tavoite ja tutkimuskysymykset

Työn tarkoitus on mallintaa balansoiva säätö EV3 – Mindstorms Lego – robotin rungon kallistuskulmalle ψ eli käytännössä kehittää säätörakenne, jonka avulla voidaan tasapainottaa liikkumaton ja pystyssä renkaiden päällä seisova robotti. Säätö mallinnetaan Matlab- / Simulink ohjelmilla ja työssä käsiteltyä matemaattista mallia voidaan käyttää säätötekniikan opetuksen tukivälineenä sekä mahdollisesti visualisoimaan säädön vaikutusta.

Tarkastelun alla olevan Lego – robotin mallinnuksessa hyödynnetään tilamalliesitystä. Tilamallina voidaan hyödyntää jo olemassa olevaa, toisen mallisen Lego – robotin tilamallia (NXTway – GS), jonka parametrit määritetään tutkimuksessa käytettävään robottiin sopivaksi simulointimalliin.

2. LEGO MINDSTORMS EV3

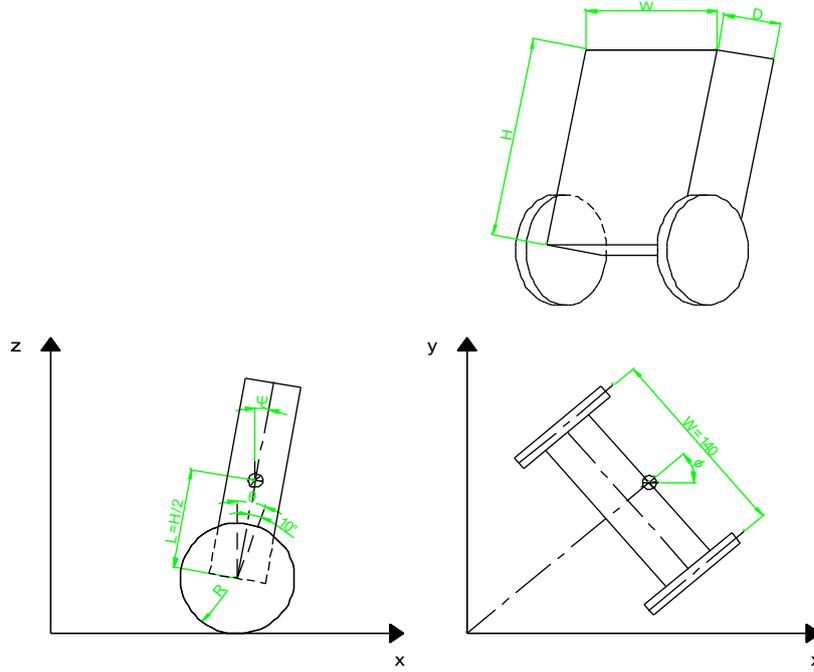
Mindstorms EV3 on tanskalaisen Legon kehittänyt kolmannen sukupolven robottien rakennussarja, joka julkaistiin syyskuussa 2013. Sen uusi ohjelmoitava ydin eli keskusyksikkö, on teknisesti kehittyneempi kuin edeltäneessä Mindstorms NXT – robotissa sekä ytimellä on mikro SD – kortinlukija ja USB – portti. Tässä uuden sukupolven ytimessä, käyttöjärjestelmänä toimii Linux, sekä siinä hyödynnetään ARM9 - prosessoria verrattuna edellisen sukupolven ARM7 - prosessoriin. RAM - ja Flash muistien kokoa on kasvatettu 64 ja 256 kilotavusta 64 ja 16 megatavuun. Mindstorms EV3 – robotti on varusteltu kahdella suuremmalla ja yhdellä pienemmällä tasavirtamoottorilla sekä erityyppisillä antureilla: kosketus -, väri -, gyro – ja ultraäänianturilla. (Norouzi Kandalan 2016).

Säätötekniikan opetuksen kannalta robotin ohjelmointi on suoraviivaista, koska se tukee Matlab/Simulink ympäristöä omalla lohkokirjastollaan.

(<https://se.mathworks.com/hardware-support/lego-mindstorms-ev3-matlab.html>)

2.1 Mallinnus

Mallinnuksen lähtökohtana Lego – robotin geometrinen ulkomuoto on yksinkertaistettu ja oletettu laataksi ja kahdeksi kiekoksi, joiden tiheys on vakio, matemaattisen mallin laskentaa varten kuvassa 2.1. Kyseinen ns. heilurimalli on riittävän tarkka kuvaamaan massan kiertymistä toisin sanoen kulmaa (θ , Ψ , ϕ) ja sen kulmanopeutta ($\dot{\theta}$, $\dot{\Psi}$, $\dot{\phi}$) kuvassa 2.1 määritettyjen akseleiden suhteen ja mallinnuksen tuloksena ovat tilaesitykset. Mallinnus perustuu kulmiin (θ , Ψ , ϕ) sekä niiden kulmanopeuksiin ($\dot{\theta}$, $\dot{\Psi}$, $\dot{\phi}$), jotka ovat myös järjestelmän säädettäviä suureita.



Kuva 2.1 Lego – robotin yksinkertaistettu heilurimalli, jossa säätöjärjestelmän lähtösuureet ovat zx – tasossa rungon kallistuskulma Ψ sekä renkaan kiertymiskulma θ ja yx – tasossa ϕ on rungon kiertymiskulma. Lisäksi kuvassa ovat laskennassa käytetyt parametrit: L , H , R , W ja D . (Yamamoto 2009).

Robotin dynamiikka voidaan esittää tilamallien avulla, (Yamamoto 2009):

$$\dot{x}_1 = \mathbf{A}_1 x_1 + \mathbf{B}_1 u, \quad (2.1)$$

$$\dot{x}_2 = \mathbf{A}_2 x_2 + \mathbf{B}_2 u, \quad (2.2)$$

jossa \mathbf{A}_1 , \mathbf{A}_2 , \mathbf{B}_1 ja \mathbf{B}_2 ovat tilamallin parametrit sisältävät matriisit ja jotka sisältävät matriisin \mathbf{E} , (Yamamoto 2009):

$$\mathbf{E} = \begin{bmatrix} (2m + M)R^2 + 2J_w + 2n^2J_m & MLR - 2n^2J_m \\ MLR - 2n^2J_m & ML^2 + J_\Psi + 2n^2J_m \end{bmatrix}, \quad (2.3)$$

jossa m on renkaan massa, M on robotin rungon massa, R on renkaan säde, J_w on renkaan hitausmomentti, n on vaihteen välityssuhde, J_m on tasavirtamoottorin roottorin hitausmomentti, L on etäisyys renkaan akselilta robotin painopisteeseen ja J_Ψ on hitausmomentti rungon kallistuskulman suhteen. Yhtälöiden (2.1) ja (2.2) tilavektori x_1 voidaan esittää, (Yamamoto 2009):

$$x_1 = [\theta \quad \Psi \quad \dot{\theta} \quad \dot{\Psi}]^T, \quad (2.4)$$

jossa θ on renkaan kiertymiskulma ja Ψ on rungon kallistuskulma tunnetun pisteen suhteen pystytasossa sekä niiden derivaatat eli kulmanopeudet. Vastaavasti tilavektori x_2 voidaan esittää, (Yamamoto 2009):

$$x_2 = [\phi \quad \dot{\phi}]^T, \quad (2.5)$$

jossa ϕ on rungon kiertymiskulma ja $\dot{\phi}$ on sen kulmanopeus tunnetun pisteen suhteen vaakatasossa. Tulovektori u voidaan esittää, (Yamamoto 2009):

$$u = [v_l \quad v_r]^T, \quad (2.6)$$

jossa v_l ja v_r ovat vasemmalle ja oikealle tasavirtamoottorille syötetyt jänniteohjeet. Systemimatriisi \mathbf{A}_1 on muotoa, (Yamamoto 2009):

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-gMLE(1,2)}{\det(\mathbf{E})} & \frac{-2[(\beta+f_w)\mathbf{E}(2,2)+\beta\mathbf{E}(1,2)]}{\det(\mathbf{E})} & \frac{2\beta[\mathbf{E}(2,2)+\mathbf{E}(1,2)]}{\det(\mathbf{E})} \\ 0 & \frac{gMLE(1,1)}{\det(\mathbf{E})} & \frac{2[(\beta+f_w)\mathbf{E}(1,2)+\beta\mathbf{E}(1,1)]}{\det(\mathbf{E})} & \frac{-2\beta[\mathbf{E}(1,1)+\mathbf{E}(1,2)]}{\det(\mathbf{E})} \end{bmatrix}, \quad (2.7)$$

jossa β on vakio ja se lasketaan yhtälöllä, (Yamamoto 2009):

$$\beta = \frac{nK_t K_b}{R_m} + f_m, \quad (2.8)$$

jossa K_t on tasavirtamoottorin vääntömomenttivakio, K_b on tasavirtamoottorin nopeusvakio, R_m on tasavirtamoottorin resistanssi ja f_m on kitkakerroin rungon ja tasavirtamoottorin välillä. Matriisissa (2.7) $\mathbf{E}(j, k)$ viittaa matriisiin (2.3) kyseiseen alkioon ($j = 1, 2, k = 1, 2$), g on putoamiskiihtyvyyden ja f_w on kitkakerroin renkaan ja lattian välillä. Matriisi \mathbf{A}_2 on muotoa, (Yamamoto 2009):

$$\mathbf{A}_2 = \begin{bmatrix} 0 & 1 \\ 0 & -J/I \end{bmatrix}, \quad (2.9)$$

jossa J on vakio ja se lasketaan yhtälöllä, (Yamamoto 2009):

$$J = \frac{W^2}{2R^2} (\beta + f_w), \quad (2.10)$$

jossa W on rungon leveys. Matriisissa (2.9) I on vakio ja se lasketaan yhtälöllä, (Yamamoto 2009):

$$I = \frac{1}{2}mW^2 + J_\Phi + \frac{W^2}{2R^2}(J_w + n^2J_m). \quad (2.11)$$

jossa J_Φ on rungon kaltevuuskulman (yx – tasossa) hitausmomentti. Matriisi \mathbf{B}_1 ts. tulomatriisi on muotoa, (Yamamoto 2009):

$$\mathbf{B}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{\alpha[\mathbf{E}(2,2)+\mathbf{E}(1,2)]}{\det(\mathbf{E})} & \frac{\alpha[\mathbf{E}(2,2)+\mathbf{E}(1,2)]}{\det(\mathbf{E})} \\ -\frac{\alpha[\mathbf{E}(1,1)+\mathbf{E}(1,2)]}{\det(\mathbf{E})} & -\frac{\alpha[\mathbf{E}(1,1)+\mathbf{E}(1,2)]}{\det(\mathbf{E})} \end{bmatrix}, \quad (2.12)$$

jossa α on vakio ja se lasketaan yhtälöllä, (Yamamoto 2009):

$$\alpha = \frac{nK_t}{R_m}. \quad (2.13)$$

Matriisissa (2.12) $\mathbf{E}(j, k)$ viittaa matriisin (2.3) kyseiseen alkioon. Matriisi \mathbf{B}_2 ts. tulomatriisi on muotoa, (Yamamoto 2009):

$$\mathbf{B}_2 = \begin{bmatrix} 0 & 0 \\ -K/I & K/I \end{bmatrix}, \quad (2.14)$$

jossa K on vakio, joka lasketaan yhtälöllä, (Yamamoto 2009):

$$K = \frac{W}{2R}\alpha. \quad (2.15)$$

Matriiseihin (2.7) ja (2.12) sijoitettu determinantti lasketaan:

$$\det(\mathbf{E}) = [(2m + M)R^2 + 2J_w + 2n^2J_m] \cdot (ML^2 + J_\Psi + 2n^2J_m) - (MLR - 2n^2J_m)^2. \quad (2.16)$$

Järjestelmän lähtö y on yleisesti muotoa, (Savolainen 1998):

$$y = \mathbf{C}x_{1,2} + \mathbf{D}u, \quad (2.17)$$

jossa \mathbf{C} on lähtömatriisi, $x_{1,2}$ on joko tilavektori (2.4) tai (2.5) ja \mathbf{D} on myötäkytkennän matriisi, joka tässä tapauksessa oletetaan nolaksi koska lähtö y ei riipu tulosta u , (Savolainen 1998), niin yhtälö (2.17) sievenee muotoon, kun tarkasteltava lähtönä y on rungon kallistuskulma Ψ ja sen derivaatta $\dot{\Psi}$:

$$y = \mathbf{C}x_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 \\ \psi \\ 0 \\ \dot{\psi} \end{bmatrix}. \quad (2.18)$$

Säädössä vain paikkaa käytetään takaisinkytkentänä. Tilamalleissa (2.1), (2.2) ja (2.18) lähtö y ja tilavektoreiden derivaatat ajan suhteen \dot{x}_1 sekä \dot{x}_2 riippuvat suoraviivaisesti tulosta u ja tilavektoreista $x_{1,2}$ ts. y ja \dot{x}_1 sekä \dot{x}_2 ovat lineaarisia niiden suhteen, joten edellä mainittuja tilamalleja voi kutsua lineaariseksi vakio kertomiseksi differentiaaliyhtälömalliksi ja sen sisältämät kerroinmatriisit \mathbf{A}_1 , \mathbf{A}_2 , \mathbf{B}_1 ja \mathbf{B}_2 eivät riipu ajasta t . (Savolainen 1998).

2.2 Järjestelmän parametrit

Lego – robotin mallinnus perustuu edellä johdettuihin yhtälöihin sekä mainittuihin parametreihin. Taulukossa 2.1 on esitetty laskennassa sekä simuloinnissa käytetyt parametrit, jotka löytyvät myös liitteessä 1 olevasta Matlab - koodista. Liitteenä 1 oleva Matlab – koodi sisältää myös järjestelmän tilamallin.

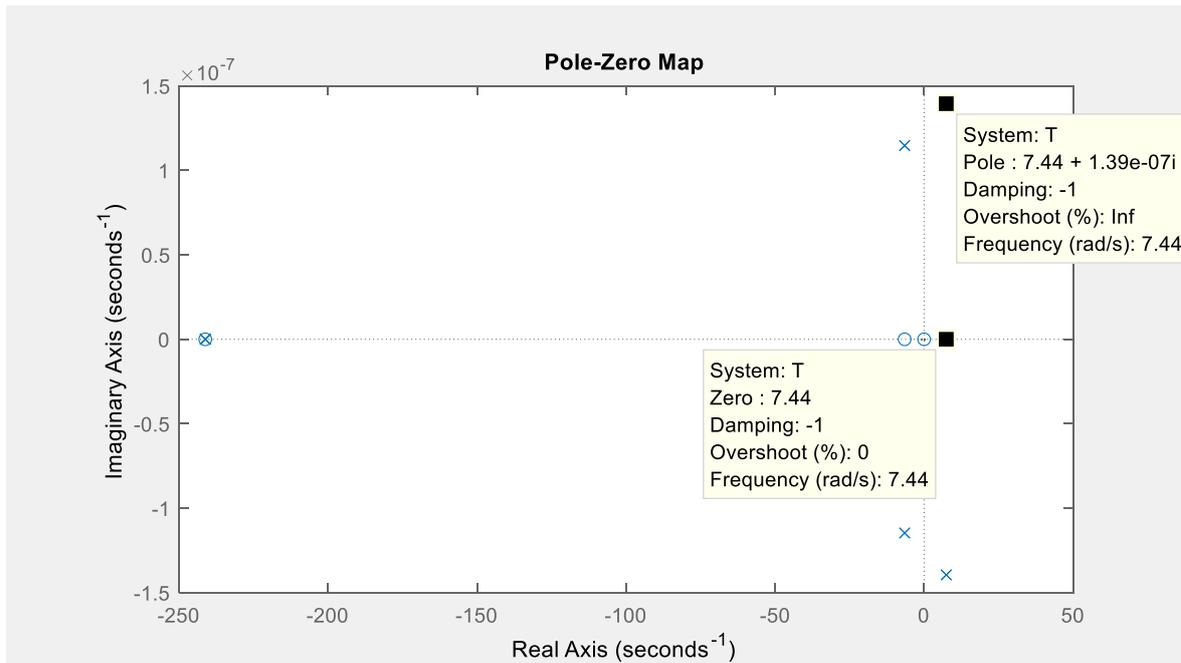
Taulukko 2.1 Matlab - koodissa käytetyt parametrit ja niiden arvot. Hitausmomentit J_Ψ ja J_Φ on laskettu taulukon yhtälöillä kuvan 2.1 mukaisen kulman pyörähdysakselin suhteen. Renkaan hitausmomentti J_w on laskettu keskiakselinsa suhteen. (Yamamoto 2009).

| Suure | Arvo | Yksikkö |
|--------------------------|---------------------|------------------|
| g | 9,81 | m/s ² |
| m | 0,03 | kg |
| R | 0,04 | m |
| $J_w = mR^2/2$ | $2,4 \cdot 10^{-5}$ | kgm ² |
| M | 0,6 | kg |
| W | 0,14 | m |
| D | 0,04 | m |
| H | 0,144 | m |
| $L = H/2$ | 0,072 | m |
| $J_\Psi = ML^2/3$ | 0,001 | kgm ² |
| $J_\Phi = M(W^2+D^2)/12$ | 0,0011 | kgm ² |
| J_m | $1 \cdot 10^{-5}$ | kgm ² |
| R_m | 6,69 | Ω |
| K_b | 0,468 | Vs/rad |
| K_t | 0,317 | Nm/A |
| n | 1 | |
| f_m | 0,0022 | |
| f_w | 0 | |

2.3 Analysointi

Heilurimalliin perustuva Lego - robotti ja sen tilayhtälö ovat luonnostaan epästabiileja, koska robotti pyrkii kaatumaan eteen - tai taaksepäin pystyasennosta luonnollisesti vaakatasoon eikä rungon kallistumista saada rajoitettua haluttuun asetusarvoon ilman takaisinkytkentään perustuvaa säätöä.

Säätämättömän – ja myös säädetyin järjestelmän dynaamista käyttäytymistä voidaan tutkia napa – nollakohtadiagrammilla kun tunnetaan järjestelmän siirtofunktio, kuva 2.2. Siirtofunktion osoittajan nollakohdat ovat järjestelmän nollia ja nimittäjän nollakohdat ovat järjestelmän napoja, jotka näkyvät kuvassa 2.2 merkinnöillä (o) ja (x). (Savolainen 1998).



Kuva 2.2 Säättämättömän järjestelmän napa – nollakohta – diagrammi kompleksitasossa, jossa rastit (x) ovat järjestelmän napoja (pole) ja ympyrät (o) nollia (zero). Kuvaan on merkitty toinen napa kompleksikonjugaattiparista ($7,44 + 1,39 \cdot 10^{-7} i$) [1/s] oikeaan puolitasoon eli järjestelmä on epästabiili.

Yleisesti ehto järjestelmän stabiilisuudelle on, että kaikki navat sijaitsevat vasemmassa puolitasossa, kun keskellä on imaginääriakseli. Kun johonkin napaan (x) kuuluu imaginääriosia esim. kuvassa 2.2, niin se aiheuttaa askelvasteeseen värähtelyä ja mitä suurempi on imaginääriosan itseisarvo niin sitä suurempi on värähtelyn kulmataajuus. Reaaliakselilla olevat navat (x) eivät siis aiheuta värähtelyä ja yleisesti mitä kauempana napa (x) on origosta (ns. nopea napa eli pieni aikavakio), sitä pienempi on sen vaikutus järjestelmään, joten lähimpänä origoa olevat navat (x) ovat dominoivimpia. Sama pätee myös nolliin (o) eli kauimmaisilla on vähäisin merkitys ja oikeassa puolitasossa oleva reaaliosaltaan positiivinen nolla (o) aiheuttaa askelvasteeseen notkahduksen, sitä suuremman mitä lähempänä nolla (o) on origoa. Niiden sijainti ei vaikuta stabiilisuuteen. (Savolainen 1998).

Tämän tulkinnan sekä kuvan 2.2 perusteella järjestelmä on epästabiili ja vaatii siten takaisinkytketyn säädön.

3. SÄÄTIMEN SUUNNITTELU JA TESTAUS

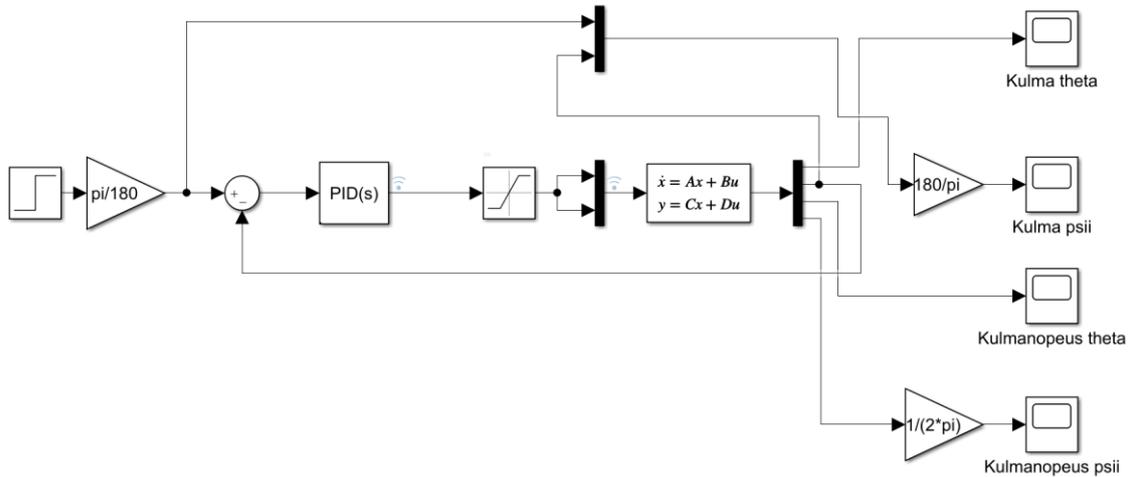
Lego – robotin sivusuuntaista kallistuskulmaa säädetään pyörittämällä molempia renkaita tasavirtamoottoreilla samanaikaisesti ts. järjestelmän lähtö eli säädettävä suure on rungon kallistuskulma Ψ , jota mitataan gyroilla eli kallistuskulmasensorilla. Systemin tuloina toimii yksinkertaistettuna molemmille tasavirtamoottoreille syötettävät jännitteet. Balansoivalla kallistuskulman säädöllä pyritään estämään robotin kaatumisen. Tässä työssä mallinnuksessa ja analyysissä ei oteta huomioon robotin vapaata liikkumista eteen- tai taaksepäin eikä kääntymistä paikallaan tai liikkeessä.

Matlabin avulla voidaan tarkastella kuinka stabiili takaisinkytketty järjestelmä eli samalla itse säätö on.

3.1 Simulointimalli

Edellä kuvattujen matriisien (2.3), (2.7), (2.9), (2.12) ja (2.14) alkioiden arvot sekä determinantti (2.16) lasketaan Matlabin avulla. Laskennassa hyödynnetään parametrejä, kuten robotin massa M sekä fyysisiä dimensioita kuten etäisyys L . Lisäksi robotin dynamiikkaa kuvataan mekaniikan yhtälöillä, ts. tunnettujen hitausmomenttien J_m , J_w ja J_Ψ avulla. Lasketut matriisit \mathbf{A}_1 , \mathbf{A}_2 , \mathbf{B}_1 ja \mathbf{B}_2 sijoitetaan tilamallia kuvaavaan yhtälöön/yhtälöihin (2.1) ja (2.2), joista seuraavaksi kehitetään järjestelmän siirtofunktio. Se mallinnetaan Simulink – ohjelmaan yhdessä sopivan säätimen kanssa.

Kuvassa 3.1 on esitetty Simulink - ohjelmalla mallinnettu Lego – robotin säätöjärjestelmä, johon kuuluu askeleena syötettävä ohjearvo, eroelin, säädin, saturaatio – lohko, järjestelmän dynamiikkaa kuvaava tilayhtälöt – lohko sekä siitä lähtösuureena saatavat neljä tilasuureta matriisissa (2.4), joista tarkastellaan vain rungon kallistumiskulmaa Ψ ja sen kulmanopeutta $\dot{\Psi}$. Jotta säädettävästä avoimesta järjestelmästä saataisiin stabiili, niin siihen on lisättävä takaisinkytkentä, jolloin säätöjärjestelmästä tulee suljettu. Takaisinkytkentä on toteutettu mitaamalla kulmaa Ψ Lego – robotin gyro – anturilla, kuva 3.1. Tarkastelussa rungon kallistumiskulma Ψ on rajattu korkeintaan ± 10 asteeseen, jolloin toimitaan realistisella toiminta - alueella ts. säätöjärjestelmä pystyy estämään Lego – robotin kaatumisen varmuudella.



Kuva 3.1 Lohkokaavio Lego – robotin takaisinkytketystä säätöjärjestelmästä, jossa mittasuureena gyrolla mitattu rungon kallistuskulma Ψ sekä lähtö - ja mittasuureen erotuksesta laskettava ja PID - säätimelle syötettävä eroosuure.

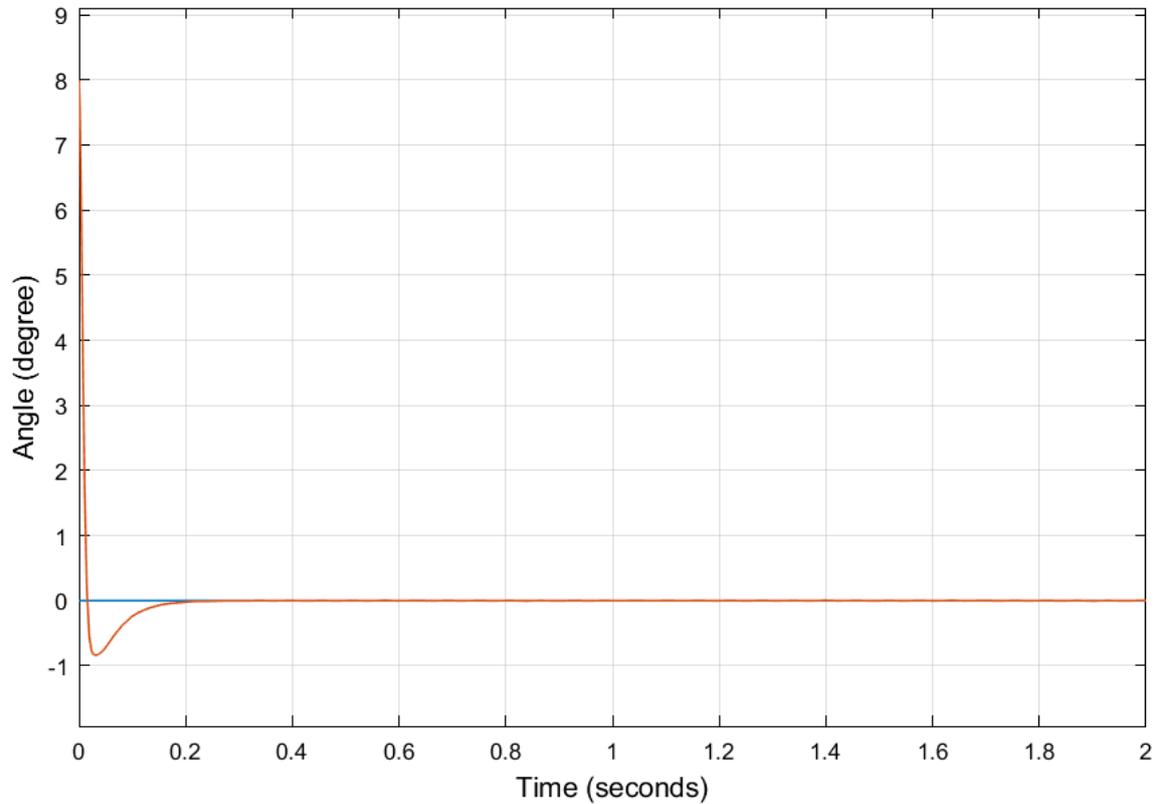
Järjestelmän erosuure PID - säätimelle lasketaan mitta - ja tulosuureen erotuksesta ja säätimestä välitetään molempien tasavirtamoottoreiden jänniteohjeet ohjaussuureena tilayhtälöt – lohkolle, joka voidaan käsittää järjestelmän toimilaitteena Simulinkissa. Säätöjärjestelmä käyttää rungon kallistuskulman Ψ ja sen kulmanopeuden $\dot{\Psi}$ yksikkönä radiaania ja rad/s, jotka skaalataan asteina syötetystä ohjearvosta. Lähtösuureet skaalataan takaisin asteiksi ja pyörimisnopeus lasketaan kierroksia sekunnissa.

Kuvassa 3.1 säätimen ja tilayhtälöt – lohkon välissä on saturaatio – lohko, jonka tehtävänä on määrätä pulssileveysmoduloidun (PWM) tulossignaalin eli jänniteohjeen ylä - ja alarajat, jotka on asetettu + 100:aan ja – 100:aan. Kun jänniteohje on ylä – ja alarajan välissä niin se läpäisee lohkon muuttumattomana, mutta jos se on näiden rajojen ulkopuolella niin se rajataan eli saturoidaan joko ylempään tai alempaan rajaan, (Nise 2000).

3.2 Tulokset

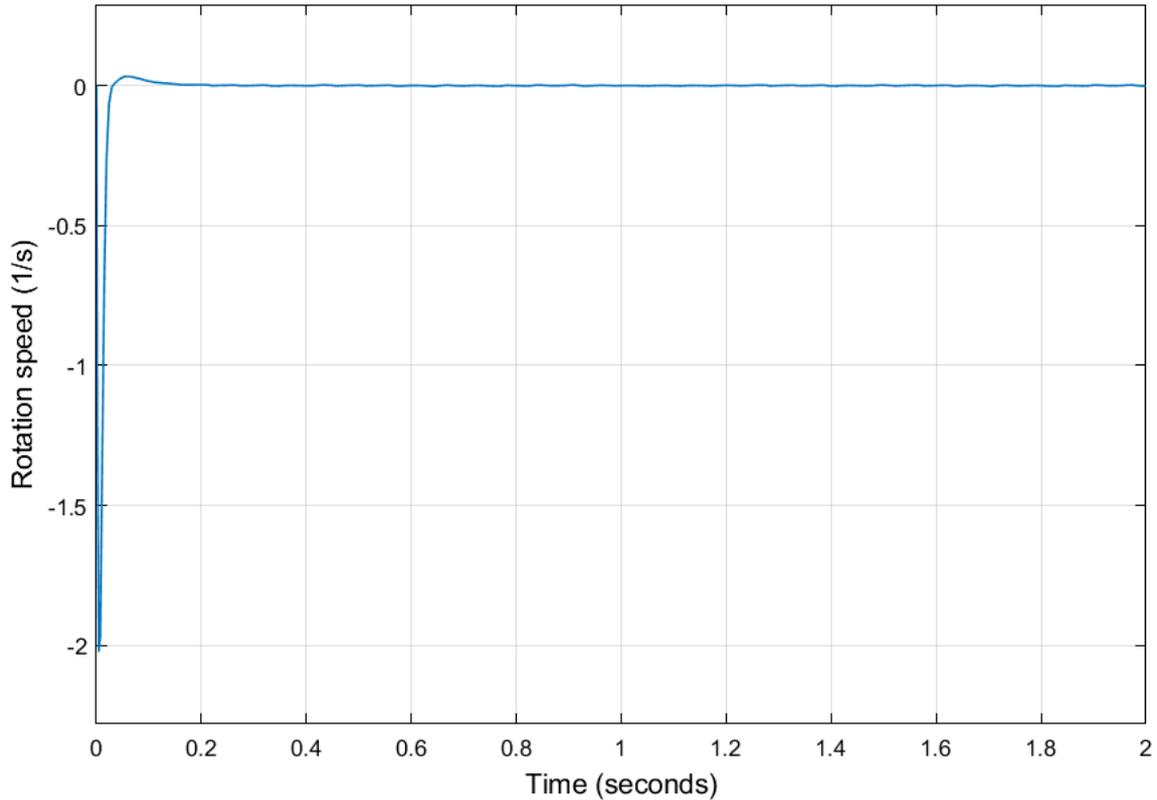
Kun kuvan 3.1 säätöjärjestelmän tulosuureena (referenssinä) Ψ_{ref} on 0 astetta, joka kuvaa robotin pystyasentoa. Stabilointia kuvataan asettamalla tilayhtälöt – lohkon alkukulmaksi 8 astetta. Muutostilanteessa rungon kallistuskulman Ψ ja sen kulmanopeuden $\dot{\Psi}$ askelvasteet ovat kuvien 3.2 ja 3.3 mukaiset. Suunnittelun lähtökohdaksi on valittu PID – säädin sen monipuolisuuden takia (Savolainen 1998) ja johon on optimoitu parametrit käyttäen apuna

Simulinkin autotune - ominaisuutta: $K_P = -193,1$, $K_I = -3595$ ja $K_D = -0,4541$. Johtuen parametrien optimoimisesta PID - säädin ei jätä jatkuvuustilan virhettä asetusarvon 0 astetta ja rungon kallistumiskulman Ψ välille, kuva 3.2.



Kuva 3.2 Rungon kallistumiskulma Ψ [°] ajan t [s] funktiona (punainen käyrä) kun asetusarvoksi on syötetty 0° (sininen käyrä) ja kun alkukulma on 8° . PID – säätimen parametrit ovat: $K_P = -193,1$, $K_I = -3595$ ja $K_D = -0,4541$.

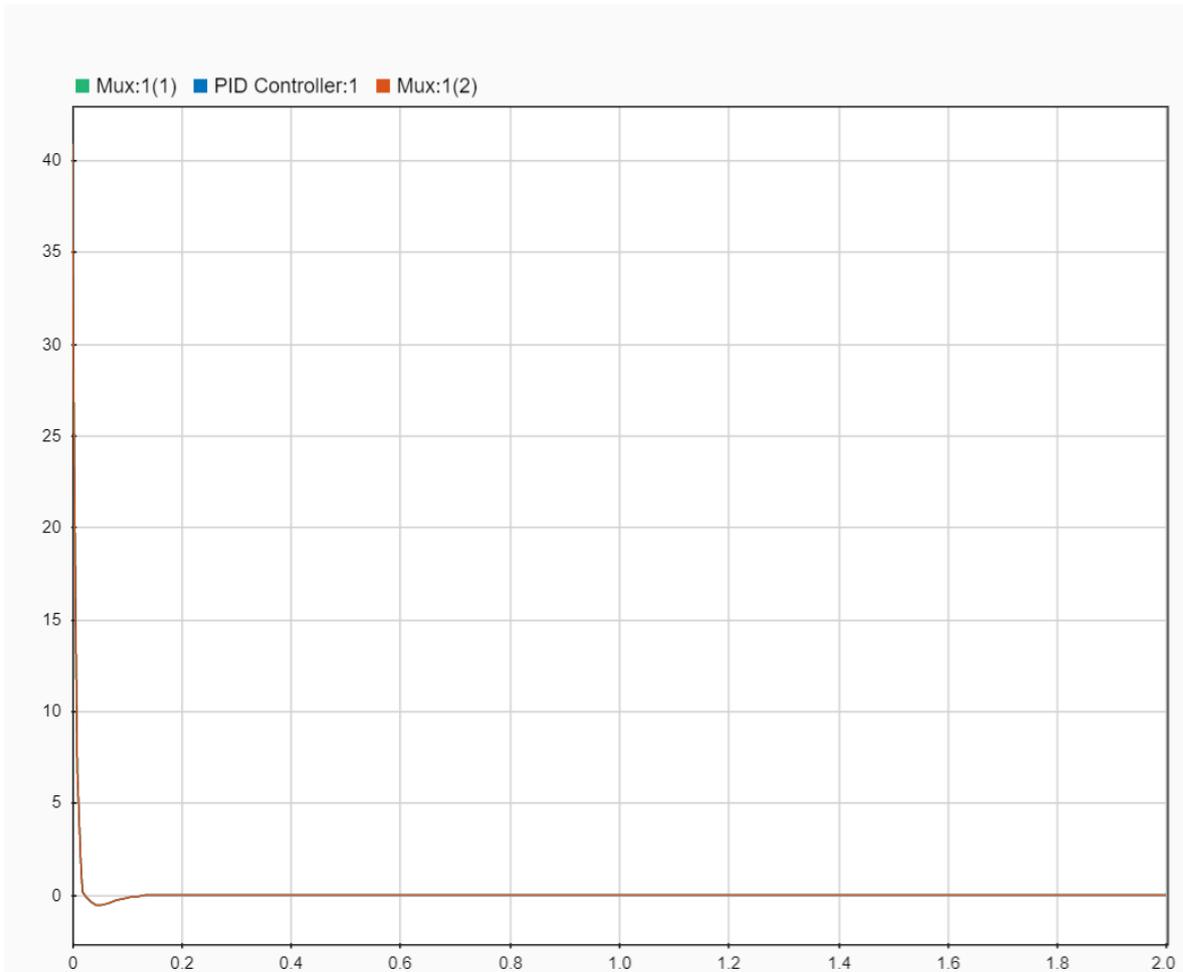
Kulmanopeus $\dot{\Psi}$ laskee kiihtyvästi heti arvoon noin -2 1/s eli moottorit pyörittävät renkaita negatiiviseen ts. päinvastaiseen suuntaan kuin 8 asteen kallistuma ja tämän jälkeen kulmanopeus $\dot{\Psi}$ nousee nollaan samalla hidastuen eli moottorit pysähtyvät, kuva 3.3.



Kuva 3.3 Rungon kallistumiskulman kulmanopeus $\dot{\Psi}$ laskettuna pyörimisnopeudeksi [1/s] ajan t [s] funktiona kun asetusarvoksi on syötetty 0° ja kun alkukulma on 8° . PID – säätimen parametrit ovat: $K_P = -193,1$, $K_I = -3595$ ja $K_D = -0,4541$.

Kuvan 3.2 perusteella robotin paikoitus on mahdollista tarkasti pysty – eli tasapaino - asentoon kun liike lähtee paikaltaan ($\dot{\Psi} = 0$) ja alkukulmasta 8 astetta.

PID – säädin syöttää ohjaussuureen tilayhtälöt – lohkolle eli käytännössä vasemman – ja oikean tasavirtamoottorin jänniteohjeen (v_l , v_r), jotka ovat samat koska molempia renkaita pyöritetään synkronoidusti. Kun alussa rungon kallistuskulma Ψ ja myös mittasuure on 8 astetta, niin PID - säätimen syöttämät jänniteohjeet ovat suurimmillaan. Erosuureen pienentyessä ja kallistumiskulman Ψ lähestyessä tasapainoasemaa eli 0 astetta, niin jänniteohjeetkin ensin pienentyvät jyrkästi ja kun kallistumiskulma Ψ ylittää tasapainoaseman eli renkaiden pyörimissuunta vaihtuu, niin jänniteohjeet muuttuvat negatiivisiksi. Lopulta jänniteohjeet asettuvat nolnaan moottoreiden pysähtyttyä tasapainoasemassa, kuva 3.4. Erosuure on nolla, kun mitta – ja tulosuureen erotus on nolla.



Kuva 3.4 PID - säätimen jänniteohje ajan t [s] funktiona kun rungon kallistumiskulma Ψ ja sen kulmanopeus $\dot{\Psi}$ ovat kuvien 3.2 ja 3.3 mukaiset. Kuvassa kanavien signaalit (Mux) ovat päällekkäin eli samat ja vastaavat jänniteohjeita eli ohjaussuureta tilayhtälöt – lohkolle.

4. YHTEENVETO/JOHTOPÄÄTÖKSET

Kaksipyöräisen Lego Mindstorms EV3 – robotin dynamiikan kuvaamiseen tarvitaan matriisimuotoisia tilamalleja, jotka sisältävät tulovektorin u ja tilavektorit (x_1, x_2) sekä tilamallin parametrit sisältävät matriisit ($\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1$ ja \mathbf{B}_2). Tilavektorit esittävät järjestelmän säädettäviä suureita, joista mallinnuksessa tarvitaan käytännössä vain kahta ($\Psi, \dot{\Psi}$). Tilamalliin kuuluu myös järjestelmän lähtö y , joka sisältää lähtömatriisin \mathbf{C} ja tilavektorin x_1 .

Lego – robotti on luonnostaan epästabiili eikä pysy ilman säätöä tasapaino – eli pystyasennossa eli pelkästään matriisimuotoiset tilamallit ja – lähtö muodostavat epästabiilin avoimen piirin. Avoimesta – sekä suljetusta piiristä voi muodostaa myös s – tason siirtofunktion tilayhtälöiden matriiseilla ja esimerkiksi avoimen piirin siirtofunktiosta voi muodostaa napa – nollakohtadiagrammin, jonka avulla tutkitaan stabiilisuutta Matlabissa. Järjestelmän stabiloimiseksi siihen lisätään takaisinkytkentä, joka on myös mittasuure robotin kallistuskulmasensorilta eli gyroilta, ja sopiva säädin tilayhtälöt – lohkon kanssa sarjaan Simulinkissa, jolloin säätöjärjestelmästä tulee suljettu. Säätöjärjestelmässä käytetään PID – säädintä, jonka parametrit voi syöttää joko manuaalisesti tai optimoida Simulinkin avulla automaattisesti.

Tehdyn mallinnuksen avulla voidaan tutkia ja visualisoida erilaisen säätimen vaikutusta järjestelmän askelvasteeseen eli lähtöön sekä kuinka parametrien esim. robotin eri osien massojen tai fyysisten mittojen muuttaminen Matlabissa vaikuttaa tilamalleihin ts. järjestelmän dynamiikkaan ja itse säätöön Simulinkin puolella. Lisäksi säätimen vapaan valinnan lisäksi sen parametrejä voi myös vapaasti itse kokeilla vaikka Simulinkissa onkin autotune – ominaisuus parametrien optimoimiseksi ja näin havainnoida niiden vaikutusta rungon kallistuskulmaan Ψ ja sen kulmanopeuteen $\dot{\Psi}$ eli lähtöihin. Näin ollen voidaan kokeilla manuaalisesti erilaisia säätimiä eri parametreilla ja hakea parasta kombinaatiota, vaikka se ei ole nopein tapa.

Säädön todettiin olevan tarkka, jos liike lähtee paikaltaan aloituskulmasta 8 astetta ja tulosuurena 0 astetta eli pystyasento.

Jatkokehittelyssä voidaan myös tarkastella kahta muutakin lähtösuuretta (θ, ϕ). Jälkimmäisen kulman tapauksessa pitäisi kuvan 3.1 mukaisessa lohkokaaviossa vaihtaa matriisit \mathbf{A}_1 sekä \mathbf{B}_1 \mathbf{A}_2 :een sekä \mathbf{B}_2 :een ja muokata lähtömatriisia \mathbf{C} tilayhtälöt – lohkoissa. Renkaan

kiertymiskulman θ tapauksessa riittäisi vain lähtömatriisin \mathbf{C} muokkaus sekä takaisinkytkennän lisääminen tälle lähdölle. Kaikkien kolmen kulman yhtäaikainen säätäminen vaatisi jo suurempia lisäyksiä lohkokaavioon mm. uuden tilayhtälöt .- lohkon ja sille säätimen takaisinkytkentöineen samoin kuin kahden uuden tulon lisäyksen nykyiseen verrattuna. Mallintamisessa ei otettu huomioon ns. häiriösuuretta esim. robottia tönäistään sormella sen ollessa liikkeessä, mutta sen lisääminen onnistuu Simulinkissa. Lisäksi säädön ja sen virityksen verifiointi on hankalaa pelkästään simulointimallin avulla, koska oikea robotti on epälineaarinen ja sisältää useita epäideaalisuuksia.

LÄHTEET

Afari, E., Khine, M. S. 2017. Robotics as an Educational Tool: Impact of Lego Mindstorms. IJIT, Vol. 7, No. 6, s. 437 – 442.

Alimisis, D., Menegatti, E., Moro, M. 2016. Educational Robotics in the Makers Era. EDUMOTIVA. International Conference: Educational Robotics 2016. November 25, Athens, Greece.

LEGO MINDSTORMS EV3 Support from MATLAB [verkkodokumentti]. [viitattu 19.5.2019]. Saatavissa

<https://se.mathworks.com/hardware-support/lego-mindstorms-ev3-matlab.html>

Nise, N. S. 2000. Control Systems Engineering. Third Edition. New York: Wiley.

Norouzi Kandalan, R. 2016. The Modeling and Simulation of EV3 Motor Dynamics. University of North Texas. [verkkodokumentti]. [viitattu 8.10.2018]. Saatavissa

<https://digital.library.unt.edu/ark:/67531/metadc862751/>

Savolainen, J., Vaittinen, R. 1998. Sääntötekniikan perusteita. 2. painos. Saarijärvi: Gummerus Kirjapaino Oy

Silfver, T. 2018. Control Synthesis for Balancing Robots. Aalto University. Espoo.

Valk, L. 2016. Tutorial: Self – Balancing EV3 Robot [verkkodokumentti]. [Viitattu 4.5.2019]. Saatavissa:

<http://robotsquare.com/2014/07/01/tutorial-ev3-self-balancing-robot/>

Yamamoto, Y. 2009. NXTway-GS Model-Based Design – Control of self-balancing two-wheeled robot built with LEGO Mindstorms NXT. First Edition. Cybernet Systems Co., Ltd.

LIITTEET**Matlab – koodi Lego – robotin mallinnuksesta**

```

clc
clearvars
close all

%vakiot
g=9.81;%putoamiskiihtyvyyys, [m/s^2]
m=0.03;%renkaan massa, [kg]
R=0.04;%renkaan säde, [m]
Jw=m*R^2/2;%renkaan hitausmomentti, [kgm^2]
M=0.6;%rungon massa, [kg]
W=0.14;%rungon leveys, [m]
D=0.04;%rungon syvyys, [m]
H=0.144;%rungon korkeus, [m]
L=H/2;%massakeskipisteen etäisyys renkaan akselilta, [m]
Jksii=M*L^2/3;%rungon kaltevuuskulman (zx -tasossa) hitausmo-
mentti, [kgm^2]
Jfii=M*(W^2+D^2)/12;%rungon kaltevuuskulman (yx -tasossa) hi-
tausmomentti, [kgm^2]
Jm=1e-5;%tasavirtamoottorin hitausmomentti, [kgm^2]
Rm=6.69;%tasavirtamoottorin resistanssi, [ohm]
Kb=0.468;%tasavirtamoottorin nopeusvakio (back EMF constant),
[V/s/rad]
Kt=0.317;%tasavirtamoottorin vääntömomenttivakio (torque
constant), [Nm/A]
n=1;%välityssuhde
fm=0.0022;%kitkakerroin rungon ja tasavirtamoottorin välillä
fW=0;%kitkakerroin renkaan ja lattian välillä
alfa=n*Kt/Rm;%vakio
beta=(n*Kt*Kb)/Rm+fm;%vakio
I=0.5*m*W^2+Jfii+W^2/(2*R^2)*(Jw+n^2*Jm);%vakio
J=W^2/(2*R^2)*(beta+fW);%vakio
K=W/(2*R)*alfa;%vakio

%matrxiisit
E=[(2*m+M)*R^2+2*Jw+2*n^2*Jm M*L*R-2*n^2*Jm;M*L*R-2*n^2*Jm
M*L^2+Jksii+2*n^2*Jm];%E matriisi
A1_32=-g*M*L*E(1,2)/det(E);%A1 matriisin 3. rivi, 2. sarake
A1_42=g*M*L*E(1,1)/det(E);%A1 matriisin 4. rivi, 2. sarake
A1_33=-2*((beta+fW)*E(2,2)+beta*E(1,2))/det(E);%A1 matriisin
3. rivi, 3. sarake
A1_43=2*((beta+fW)*E(1,2)+beta*E(1,1))/det(E);%A1 matriisin
4. rivi, 3. sarake
A1_34=2*beta*(E(2,2)+E(1,2))/det(E);%A1 matriisin 3. rivi, 4.
sarake
A1_44=-2*beta*(E(1,1)+E(1,2))/det(E);%A1 matriisin 4. rivi,
4. sarake
B1_3=alfa*(E(2,2)+E(1,2))/det(E);%B1 matriisin 3. rivi
B1_4=-alfa*(E(1,1)+E(1,2))/det(E);%B1 matriisin 4. rivi

```

```

C1=[0 0 0 0;0 1 0 0;0 0 0 0;0 0 0 1];%y1=C*x1
C2=[1 0;0 1];%y2=C2*x2
D1=[0 0;0 0;0 0;0 0];%D1 matriisi
A1=[0 0 1 0;0 0 0 1;0 A1_32 A1_33 A1_34;0 A1_42 A1_43
A1_44];%A1 matriisi
B1=[0 0;0 0;B1_3 B1_3;B1_4 B1_4];%B1 matriisi
A2=[0 1;0 -J/I];%A2 matriisi
B2=[0 0;-K/I K/I];%B2 matriisi
I1=eye(4);%I1 matriisi eli 4x4 yksikkömatriisi

sys=ss(A1,B1,C1,0);%x1' eli matriiseista A1, B1 ja C1 luotu
tilamalli
% (state-space)
step(sys);%x1' askelvaste
figure(2);
sys2=ss(A2,B2,C2,0);%x2' eli matriiseista A2, B2 ja C2 luotu
tilamalli
% (state-space)
step(sys2);%x2' askelvaste
G=tf(sys);%x1' tilamallista luotu siirtofunktio (transfer
function)

Kp = -193.1;%proportionaalinen vahvistus
Ki = -3595;%integrointivahvistus
Kd = -.4541;%derivointivahvistus
s = tf('s');%Laplace muuttuja s
PID = Kp+Ki/s+Kd*s;%PID-säätimen siirtofunktio
T=C1*((s*I1-A1)^-1)*B1%järjestelmän siirtofunktio
Gavoin = T*Kp;%lähtö T ilman takaisinkytkentää

G_suljettu=feedback(T(2,1)*PID,1)%takaisinkytketty suljettu
piiri: input 1,
%output 2
figure(3);
step(G_suljettu);%suljetun piirin askelvaste
title('Askelvaste');
figure(4);
pzmap(T);%luo kuvaajan,jossa ovat säätämättömän piirin navat
ja nollat

```