

Niklas Auranen

# SERVOMOOTTORIEN AUTOMAATTIVIRITYK- SEN TOIMINTAPERIAATE KAUPALLISISSA SERVOMOOTTORIOHJAIMISSA

Kandidaatintutkielma

Tekniikan ja luonnontieteiden tiedekunta

Tarkastaja: Pauli Mustalahti

Tarkastaja: Jouni Mattila

12/2024

# TIIVISTELMÄ

Niklas Auranen: Servomoottorien automaattivirityksen toimintaperiaate kaupallisissa servomoottoriohjaimissa  
Kandidaatintutkielma  
Tampereen yliopisto  
Teknisten tieteiden kandidaatin tutkinto-ohjelma  
Joulukuu 2024

---

Servomoottorien ohjaukseen käytettävät servomoottoriohjaimet vastaavat siitä, että moottori liikuttaa kuorman oikeaan paikkaan oikealla vauhdilla, käyttötarkoituksensa mukaisesti. Jotta systeemi voi toimia vaadittujen tavoitteiden mukaan, sille täytyy toteuttaa viritysprosessi, jossa määritetään servomoottoriohjaimen säätimien parametrit. Parametrit tulee määrittää siten, että tavoitedynamiikka saavutetaan. Koska ohjainlaitteen viritettäviä säätöparametrejä on usein kymmeniä, on viritysprosessi vaativa ja aikaa vievä. Ratkaisuna tähän on kehitetty automaattivirittämiä, kuten Simulinkin closed-loop PID-autotuner, jotka kykenevät toteuttamaan viritysprosessin itsenäisesti.

Aluksi työssä perehdytään servomoottoriohjaimen rakenteeseen yleisellä tasolla. Tehdään katsaus ohjaimen silmukkarakenteeseen ja sen ominaisuuksiin. Seuraavaksi tarkennetaan PID-säätimen haarojen rakennetta ja toimintaa. Luvussa kolme käydään läpi erilaisten automaattiviritysmenetelmien historiaa ja kehitystä. Lisäksi luvussa käsitellään ohjattavan servosysteemin mallintamista ja sen avulla taustoitetaan keskeisimpiä virityssäätöjä.

Kaupallisten tuotteiden osiossa esitellään viisi yleisessä käytössä olevaa servomoottoriohjainta, joissa on mahdollista suorittaa automaattiviritys. Ohjainten automaattiviritysprosessien rakennetta ja toimintaa tutkitaan. Luvussa pyritään määrittämään, onko viritys toteutettu aikatazon vai taajuustason menetelmiä käyttäen.

Lopuksi työssä tehdään simulointi MathWorksin Simulink-ohjelmalla. Simulink-ohjelmaan toteutettua PID closed-loop autotuning-työkalua hyödynnetään yleisen PID-säätimen virityksessä. Ohjailtavana laitteena on kaksi eri kokoluokan DC-moottoria A ja B. Systeemien ohjaukseksi valitaan moottoriin syötettävä jännite, ulostuloksi valitaan moottorin pyörittävän akselin kulmanopeus. Systeemin herätteenä käytetään yksikköaskelfunktiota. Simulointituloksista selviää, että automaattiviritys tuottaa kahden simulointikierroksen aikana molemmille systeemeille toisiaan muistuttavat vasteet. Ensimmäisen simulointikierroksen aikana automaattivirittimen tuottamat säätöparametrit aiheuttavat värähtelyä tavoitearvon molemmin puolin kummankin moottorin vasteessa. Toisen kierroksen jälkeen molemmat vasteet ylittävät asetusarvon vain kerran.

Avainsanat: Automaattiviritys, PID-säätö, servomoottoriohjain, servosysteemi

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin Originality Check -ohjelmalla.

Opinnäytteessäni käytetyt tekoälytyökalut ja niiden käyttötarkoitukset on kuvailtu alla:

**Työkalun nimi (ja versio):** Scopus AI

**Käyttötarkoitus ja osio, jossa työkalua käytettiin:** Lähteiden etsintä luvuissa 2.1 Ohjaussilmukat ja 2.4 P-, I- ja D-säätö

Olen tietoinen siitä, että olen täysin vastuussa koko opinnäytteeni sisällöstä, mukaan lukien tekoälyllä tuotetut osat, ja hyväksyn vastuun mahdollisista eettisten ohjeiden rikkomuksista.

# SISÄLLYSLUETTELO

1. JOHDANTO .....	1
2. SERVOMOOTTORIOHJAIMEN RAKENNE JA TOIMINTA .....	3
2.1. Ohjaussilmukat .....	3
2.1.1. Virtaohjain.....	4
2.1.2. Nopeusohjain.....	5
2.1.3. Asento-ohjain.....	5
2.2. Taajuusvaste ja kaistanleveys .....	5
2.3. Pulssinleveysmodulaatio PWM.....	6
2.4. P-, I- ja D-säätö .....	6
2.4.1. P-säädin .....	6
2.4.2. PI-säädin .....	7
2.4.3. PID-säädin.....	7
3. SERVOMOOTTORIOHJAIMEN VIRITYS.....	9
3.1. Automaattiviritys .....	10
3.2. Systemin mallintaminen .....	12
3.3. Virityksen menetelmät .....	13
3.3.1. Ziegler–Nichols ja askelvastemenetelmä .....	13
3.3.2. Ziegler–Nichols-taajuusvastekoe .....	15
3.3.3. Automatisoitu taajuusvastemenetelmä.....	16
4. RATKAISUT KAUPALLISISSA TUOTTEISSA.....	18
4.1. Beckhoff TE5960 .....	18
4.2. Maxon motor EPOS Studio Auto tune.....	19
4.3. Mitsubishi MR-J4 Auto-tuning .....	20
4.4. Schneider electric autotuning.....	21
4.5. Automationdirect SureServo2 autotuning.....	21
4.6. Kaupallisten ratkaisujen yhteenveto.....	22
5. SIMULAATIO .....	23
5.1. DC-moottori .....	23
5.2. Simulaation PID-säädin .....	27
5.3. Tulokset.....	28
5.3.1. Kierros 1 .....	28
5.3.2. Kierros 2 .....	30
6. YHTEENVETO .....	32

LÄHTEET.....	33
6.1. LIITE A: DC-MOOTTORIEN A JA B MÄÄRITYS SEKÄ FOPDT- JA SOPDT-SYSTEEMIEN ASKELVASTEET, MATLAB .....	36

# LYHENTEET JA MERKINNÄT

FOPDT	First-Order Plus Dead Time
SOPDT	Second-Order Plus Dead Time
D-haara	Derivointihaara
PID	Proportional-integral-derivative
P-haara	Proportionaalihaara
PRBS	Pseudorandom binary sequence
PWM	Pulse Width Modulation
I-haara	Integraalihaara
ZN	Ziegler–Nichols
ÅH	Åström–Hägglund
$a$	Järjestelmän amplitudi
$b$	Kitkakerroin
$C_{par}(s)$	PID-säätimen siirtofunktio
$D$	D-vahvistusparametri
$d$	Systeemin viive
$e$	Vastasähkömotorinen voima
$e_r$	Erosuure
$G(s)$	Avoimen silmukan siirtofunktio
$h$	Releohjauksen perusvärähtelyn amplitudi
$H(s)$	Takaisinkytkennän siirtofunktio
$I$	I-vahvistusparametri
$i$	sähkövirta
$I(s)$	Sähkövirran Laplace-muunnosfunktio
$J$	Hitausmomentti
$K$	Vääntömomenttivakio
$K_G$	Siirtofunktion vahvistus
$K_p$	Proportionaalivahvistus
$K_u$	Kriittinen vahvistus
$K_{emf}$	Vastasähkömotorisen voiman ja kulmanopeuden verrannollisuuskerroin
$L$	Käämin induktanssi
$N$	Derivaattorisuodattimen kerroin
$\omega_n$	Luonnollinen kulmataajuus
$P$	P-vahvistusparametri
$P_u$	Kriittinen jaksonaika
$R$	Resistanssi
$\tau$	Aikavakio
$\tau_d$	Derivointiaika
$\tau_I$	Integrointiaika
$T$	Vääntömomentti
$\dot{\theta}$	Kulmanopeus
$\ddot{\theta}$	Kulmakiihtyvyys

$\theta(s)$	Moottorin aseman Laplace-muunnosfunktio
$\dot{\theta}(s)$	Moottorin kulmanopeuden Laplace-muunnosfunktio
$s$	Laplace-muuttuja
$u$	Sisäänmeno
$U(s)$	Sisäänmenon siirtofunktio
$V_{in}$	Jännitelähteen jännite
$V_L$	Jännite käämin yli
$V_r$	Jännite vastuksen yli
$V(s)$	Jännitteen Laplace-muunnosfunktio
$Y(s)$	Ulostulon siirtofunktio
$\zeta$	Vaimennussuhde

# 1. JOHDANTO

Servomoottorien ohjaukseen käytettävät servomoottoriohjaimet vastaavat siitä, että moottori toteuttaa käyttötarkoituksensa mukaista dynamiikkaa. Jotta moottorille saavutettaisiin tavoiteltava liike, tulee servomoottoriohjaimelle suorittaa viritys. Viritys tarkoittaa prosessia, jossa pyritään valitsemaan systeemin ohjainlaitteelle suorituskyvyn kannalta mahdollisimman optimaaliset säätöparametrit. Säätöparametrit voivat olla esimerkiksi PID-säätimen P-, I- ja D-haarojen vahvistuksia, jotka vaikuttavat säätimen luoman ohjaussignaalin luonteeseen. Ohjainlaitteen optimaaliset parametrit ovat aina systeemikohtaisia, ja niihin vaikuttaa esimerkiksi suoritettavan tehtävän luonne ja systeemin rakenne. Viritys on tunnetusti vaativa ja aikaa vievä prosessi, ja systeemin mukaan optimoitavana voi olla kymmeniä eri parametreja.

Viritys käsitteenä tarkoittaa systeemin käyttötarkoituksen kannalta optimaalisten parametrien hakua moottorin ohjainlaitteelle. Parametrit muodostetaan moottorille suoritettavien testien tuloksien perusteella. [1, s.51] Lisäksi viritykseen voi liittyä käyttäjän asettamia vaatimuksia, kuten esimerkiksi asetuseron ylitysten määrä, tai asetumisaika. Tyypilliseen viritykseen kuuluu myös systeemin värinää ja äänihaittoja aiheuttavien taajuuksien suodattaminen sekä hitauden ja kitkan kompensointi. Automaattisen viritysohjelman tarkoitus on löytää järjestelmälle käyttötarkoitukseen sopiva viritys ilman käyttäjän apua.

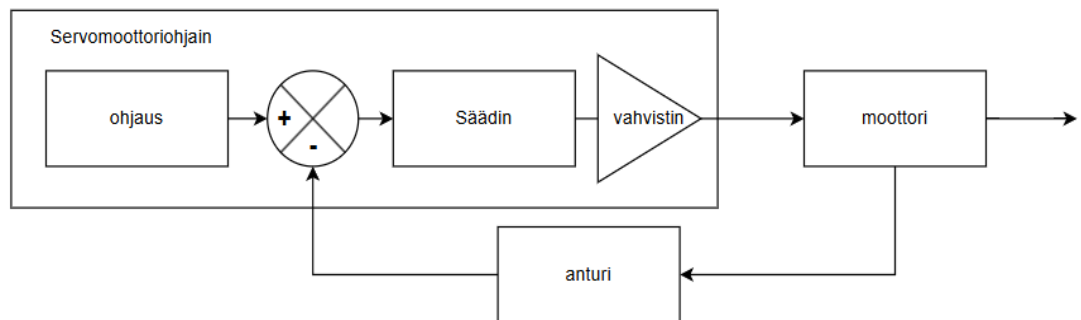
1970-luvulta lähtien on kehitetty ohjelmistoja, joilla on pyritty luomaan järjestelmän virityksestä itsenäinen ja helposti toistettavissa oleva prosessi, joka ei vaadi käyttäjältä tietämystä säätötekniikasta [1, s.50]. Sähkämoottorien käytön yleistyminen monilla tekniikan aloilla on entisestään lisännyt tarvetta nopean ja tarkan dynamiikan systeemeille. Pitkän kehityskaaren seurauksena nykyajan automaattiviritysohjelmat kykenevät virittämään useimmat järjestelmät vastaamaan tehokkuusvaatimuksia.

Tällä hetkellä servomoottoriohjainten automaattisten viritysohjelmien toimintaperiaatteista löytyy vain vähän julkaisuja suomeksi. Tämän työn tavoitteena on vertailla eri ohjainten viritysratkaisuja ja tarkastella automaattivirityksen toimintaa servomoottoriohjaimissa. Lisäksi yleisen DC-moottorin automaattista viritystä simuloidaan Mathworksin Simulink-ohjelman avulla.

Toisessa luvussa esitellään servomoottoriohjaimen eri ohjaintyyppien rakennetta ja toimintaa sekä perehdytään PID-säätimen rakenteeseen. Kolmannessa luvussa tehdään katsaus automaattisen viritysohjelman toimintaan ja tarkastellaan servomoottoriohjaimen viritysprosessia yleisesti. Neljännessä luvussa esitellään erityyppisissä kaupallisissa ohjaimissa käytössä olevia automaattiviritysratkaisuja. Viidennessä luvussa perehdytään Mathworksin Simulink-ohjelmalla simuloidun DC-moottorin automaattiseen viritykseen.

## 2. SERVOMOOTTORIOHJAIMEN RAKENNE JA TOIMINTA

Servomoottorien käyttöympäristöön kuuluu usein systeemin ohjaussignaalin eli asetusarvon vastaanottava servomoottoriohjain, joka koostuu säätimestä ja servovahvistimesta. Ohjaimen välittyä servomoottoriin kytketyn anturin mittaama informaatio moottorin todellisesta tilanteesta, systeemin ulostulosta, jota verrataan annettuun ohjaussignaaliin. Kuvassa 2.1 on yksinkertaistetun servomoottoriohjaimen rakenne servosysteemissä.



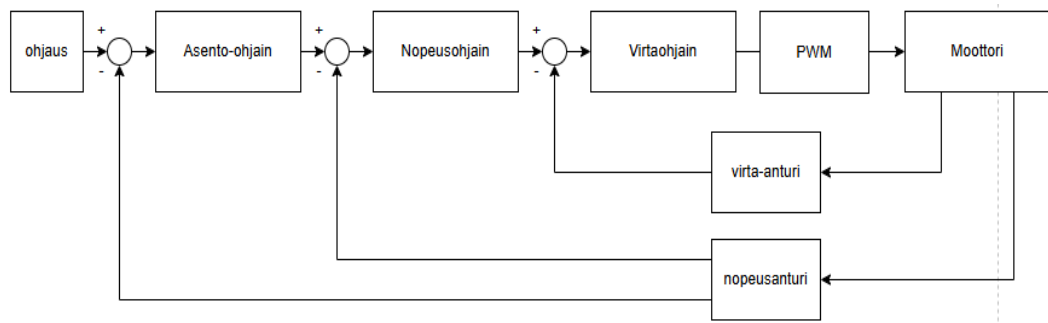
**Kuva 2.1.** Servosysteemin lohkokaavio [2]

Ohjaimen sisällä oleva säädin laskee asetusarvon ja systeemin ulostulon erotuksen, jota kutsutaan erosuureeksi. Säädin muuntaa erosuureen servomoottorin ohjaussignaaliiksi ja välittää sen servovahvistimelle. Servovahvistin vahvistaa ohjaussignaalia ja välittää moottorille ohjauskäskyn erosuureen minimoimiseksi. [2]

### 2.1. Ohjaussilmukat

Yleisimmät servomoottorin ohjauksessa käytettävät ohjaussilmukat ovat virtaohjaus, nopeusohjaus ja asento-ohjaus. Systeemin vaatimuksista ja toiminnasta riippuen servomoottoriohjaimella voi olla käytössä kaikki kolme ohjaustyyppiä tai joku niiden yhdistelmä. [3]

Useamman ohjaussilmukan servosysteemissä silmukat on kytketty sisäkkäin. Sisäkkäisten silmukoiden kaistanleveyksien tulee pienentyä sisältä ulospäin, jotta sisimmät ohjaussilmukat toimivat nopeimmalla dynamiikalla [3]. Kaikki ohjaussilmukat vaikuttavat samanaikaisesti moottorin syötettävän virran suuruuteen tavoitedynamiikan saavuttamiseksi [4]. Kuvassa 2.2 on servomoottorihjaimen silmukkarakenteet yksinkertaistettuna.



**Kuva 2.2** kolmen ohjaussilmukan servomoottorihjaimen servosysteemissä [5]

### 2.1.1. Virtaohjain

Virtaohjaimella säädellään servomoottoriin kulkeutuvaa virtaa, mikä vaikuttaa moottorin vääntömomentin suuruuteen. Virtaohjain on takaisinkytketty servomoottoriin, mikä mahdollistaa ohjaimen reagoinnin vääntömomentin virheeseen. Mikäli moottorin anturin välittämän virran suuruus ei vastaa asetettua arvoa, virtaohjaimen ulostulo päivittyy. Virtaohjaimen tulee pystyä reagoimaan muutoksiin nopeasti, joten ohjaimen käytettävissä olevan kaistanleveyden tulee olla suuri verrattuna muihin ohjaimiin. Tästä syystä virtaohjain kytketään ohjaussilmukoista sisimmäksi. [3]

Moottorin vääntömomentin suuruuden kasvattaminen lisää moottorin pyörimisnopeutta ja mahdollistaa sitä kautta myös nopean määrätyn asennon saavuttamisen. Virtaohjain on siten ominaisuuksiltaan hyödyllinen nopeaa vasteaikaa ja tarkkaa asento-ohjausta vaativissa tehtävissä. Rakenteeltaan virtaohjain on tyypillisesti PI-säädin. [3]

### 2.1.2. Nopeusohjain

Nopeusohjaus on servomoottorien yleisin ohjaustyyppi. Se säätelee servomoottorin nopeutta vertaamalla moottorin todellista nopeutta asetusarvoon. Nopeusohjain välittää asetusarvon poikkeaman käskynä virtaohjaimelle. [3]

Nopeussilmukka on tyypillisesti PI-säädin, eli se hyödyntää sekä P-vahvistusta, että I-vahvistusta ohjaussignaalin muodostamisessa. Nopeusohjain kytketään ohjaussilmukoiden keskelle. [3]

### 2.1.3. Asento-ohjain

Asento-ohjain kytketään ohjaussilmukoista uloimmaksi. Sisäkkäin kytketyt ohjaussilmukat muodostavat niin kutsutun kaskadiohjainrakenteen, missä ulompi silmukka välittää ohjaussignaalin sisemmälle silmukalle. [3]

Ohjain vertailee todellista moottorin asentoa asetettuun arvoon takaisinkytkennän avulla. Asento-ohjain on tyypillisesti P-säädin, eli se hyödyntää vain P-vahvistusta ohjaussignaalin muokkauksessa. Asento-ohjain välittää asentokäskyn nopeusohjaimelle. [3]

## 2.2. Taajuusvaste ja kaistanleveys

Servomoottoriohjaimen ohjaussilmukat kytketään yleensä sisäkkäin. Sisäkkäinen kytkentämenetelmä aiheuttaa tiettyjä rajoituksia ohjaussilmukoiden kaistanleveydelle.

Younkin määrittelee teoksessaan "Industrial servo control systems: fundamentals and applications" [6] systeemin taajuusvasteen ja kaistanleveyden yhteyden seuraavasti: Taajuusvaste kertoo, kuinka tarkasti järjestelmän ulostulo mukailee sinimuotoista sisäänmenoa sisäänmenon taajuuden noustessa. Korkeammilla taajuuksilla

ulostulo ei enää mukaile sisäänmenoa. Mitä korkeampi servomoottoriohjaimen kais-  
tanleveys on, sitä paremmin se kykenee seuraamaan nopeita muutoksia sisäänme-  
nossa. [6, s. 42–43]

Taajuusvasteen vaiheviive sisäänmenoon verrattuna on oltava vähemmän kuin 180  
astetta, jotta järjestelmä pysyy stabiilina. Tyypillisesti 135 asteen vaiheviive on ase-  
tettu takarajaksi jättäen vaihemarginaaliksi 45 astetta. Asentoa ohjaavissa järjestel-  
missä, joissa on sisäänrakennettu nopeussilmukka, on oltava nopeuden takaisinkyt-  
kennän integrointi asennon takaisinkytkennäksi. Tämä integrointiprosessi aiheuttaa  
aina 90 asteen vaiheviiveen kaikilla taajuuksilla. Tämä rajoittaa vaiheviitettä entises-  
tään 135 asteesta 45 asteeseen. [6, s. 42–43]

### **2.3. Pulssinleveysmodulaatio PWM**

Virtaohjain säätelee moottorille kulkeutuvaa virtaa pulssinleveysmoduloinnin (PWM,  
pulse width modulation) avulla. Tieteen termipankki määrittelee Pulssinleveysmodu-  
loinnin seuraavasti: ”Pulssisuhdesäädön eli pulssinleveysmodulaation (PWM)  
ideana on katkoa suurella taajuudella vakioasuinen syöttötasajännite kestoaltaan ja  
leveydeltään vaihtelevanleveysiksi pulsseiksi siten, että näistä pulsseista integroi-  
malla muodostuu mahdollisimman hyvin siniaaltoja muistuttava syöttöjännite.” [7]

Pulssinleveysmodulaation suorittaa tyypillisesti erillinen virtasilmukkaan kytketty  
vahvistin, joka välittää moduloidun ohjaussignaalin moottorille. Pulssinleveysmodu-  
laation etuna muihin säätelymenetelmiin verrattuna on pieni tehohäviö. [8].

### **2.4. P-, I- ja D-säätö**

#### **2.4.1. P-säädin**

Rakenteeltaan yksinkertaisin säädin on Proportional- eli P-säädin. P-säätimessä on  
vain P-haara, joka vahvistaa ohjaussignaalia erosuureen minimoimiseksi. P-säätii-  
men rakenne on

$$u = K_p e_r, \quad (2.1)$$

missä  $u$  on sisäänmeno,  $K_p$  säätimen proportionaalivahvistus ja  $e_r$  erosuure. [9, s. 66–67] P-säädin ei aina kykene eliminoimaan stationääritilan virhettä, siksi se on hyödyllinen järjestelmissä, joissa systeemin ulostulolle riittävä tarkkuus on tietyn virhemarginaalin sisällä.

### 2.4.2. PI-säädin

Proportional-Integral-säädin eli PI-säädin koostuu P-haarasta sekä I-haarasta. PI-säätimen rakenne on

$$u = K_p \left( e_r + \frac{1}{\tau_i} \int e_r dt \right), \quad (2.2)$$

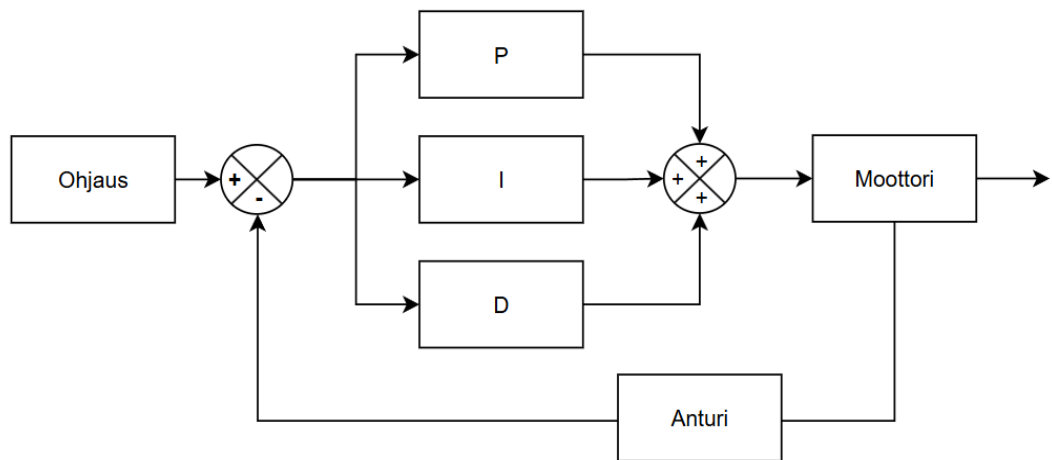
missä  $\tau_i$  on integroimisaika. I-haara ohjaa ulostulosignaalin kohti asetuservoa, kun systeemi on vakaassa tilassa. I-haara kykenee vahvistamaan pienet säätövirheet ohjaussignaaliin ja eliminoi vakaan tilan virheen. [9, s. 67–68]

### 2.4.3. PID-säädin

Proportional-Integral-Derivative-säädin koostuu P-, I- ja D-haaroista, ja se on yleisesti käytössä teollisuuden prosesseissa. PID-säätimen rakenne on

$$u = K_p \left( e_r + \frac{1}{\tau_i} \int e_r dt + \tau_d \frac{d}{dt} e_r \right). \quad (2.3)$$

D-haarassa järjestelmän stabiiliutta pyritään parantamaan tekemällä ohjaussignaali suhteelliseksi virheen ennustukseen. Ennustus tehdään ekstrapoloimalla virhe-käyrän tangenttia.  $\tau_d$  on derivoimisaika. [9, s. 69] Kuvassa 3.3 on havainnollistettu PID-säädin systeemissä.



**Kuva 2.3** PID-säädin systeemissä [2]

PID-säätimen kaikki kolme ohjaustyyppiä vaikuttavat systeemiin samanaikaisesti, mikä tekee siitä joustavan tavan hallita prosesseja. Kolme ohjaustyyppiä mahdollistaa laajasti säädeltävän ohjauksen sekä säätimen mukauttamisen monenlaisiin järjestelmiin.

### 3. SERVOMOOTTORIOHJAIMEN VIRITYS

Servomoottorin vakioparametrit ovat tyypillisesti valmistajan ilmoittamat. Nämä vakioparametrit tulee syöttää servomoottoriohjaimelle ennen viritysprosessin alkamista. Taulukossa 3.1 on listattu joitain tyypillisiä parametrejä servomoottorille.

**Taulukko 3.1.** Servomoottorin vakioparametrit

Vakioparametri	yksikkö
Nimellisteho	$W$
hitausmomentti	$kgm^2$
hyötysuhde	%
vääntömomenttivakio	$\frac{Nm}{A}$
jännitevakio	$\frac{V}{rpm}$
resistanssi	$\Omega$
induktanssi	$H$

Viritysprosessin aikana määriteltävät parametrit ovat servomoottoriohjaimen sisäisten säätimien parametreja, esimerkiksi P-, I-, ja D-vahvistuksia. Tässä luvussa perehdytään edellä mainittujen vahvistuksien määrittämiseen moottorin vakioparametreille sopivaksi.

### 3.1. Automaattiviritys

Automaattinen viritysprosessi on yksinkertaistettuna lista vaiheita, jonka viritysohjelma toteuttaa itsenäisesti. Prosessin tavoitteena on saavuttaa mahdollisimman optimaalinen viritys ohjainlaitteelle, käyttäjän mahdolliset vaatimukset huomioiden. Tyypillisesti automaattiviritysohjelma teettää sarjan liikkeitä moottorille arvioiden systeemin kitkaa, resonanssitaajuuksia sekä liikuteltavaa kuormaa. Lisäksi viritysjärjestelmä ottaa huomioon käyttäjän asettamat vaatimukset, kuten vasteen ylitysten maksimimäärän ja vasteajan. Näiden tekijöiden perusteella automaattiviritysjärjestelmä luo systeemille mallin, arvioi ohjainlaitteelle moottoria vastaavat optimaaliset vahvistusparametrit sekä asettaa tarvittavat kaistanestosuodattimet. Hägglund [1, s. 51] määrittelee säätimen automaattivirityksen kiteytyvän kolmeen vaiheeseen:

1. Prosessidynamiikan määrittämiseksi säätösignaalia muutetaan hieman, jotta saadaan aikaan häiriö.
2. Tutkimalla prosessin ulostuloa, prosessin dynamiikasta voidaan määrittää matemaattinen prosessimalli.
3. Prosessimallin perusteella säätimelle voidaan määrittää vahvistusparametrit.

Säädin on mahdollista virittää prosessimallin perusteella monella eri tavalla. Toisin sanoen prosessimalli ei tarjoa yhtä oikeaa ratkaisua ohjainlaitteen viritykselle. [1]

Garrido ja Miranda esittivät vuonna 2006 julkaisussaan ”Autotuning of a DC Servomechanism Using Closed Loop Identification” kaksiosaisen näkökulman automaattiviritysprosessin rakenteelle. Julkaisussa automaattiviritysprosessi on jaettu kahteen peräkkäiseen osaan, joista ensimmäinen on moottorin mallin parametrisointi, ja toinen on säätimen viritys perustuen tunnistettuihin parametreihin. Jälkimmäiselle osuudelle löytyy laajasti sekä lineaarisia, että epälineaarisia menetelmiä useammalle säädintyypille. Ensimmäisen osuuden parametrien tunnistaminen etenkin suljetun silmukan järjestelmille on vähemmän tutkittu aihe. [10]

Automaattiviritystä ei ole vielä ratkaistu kaikkia järjestelmiä kattavaksi, vaan tietyt ehdot pätevät. K. Papadopoulos listaa teoksessaan ”PID controller tuning using the

magnitude optimum criterion” [11] kolme vaatimusta, jotka automaattisesti viritetyn ohjaimen tulee täyttää.

1. Automaattiviritysohjelman tulee valita P-, PI- ja PID-ohjauksista tehokkain ohjaustyyppi.
2. Automaattiviritysohjelman tulee päätyä sellaisiin ohjaimen parametreihin, jotta järjestelmä pysyy vakaana ohjauksen jäljittelyyn, sekä ulostulon häiriön torjumiin liittyvissä tilanteissa.
3. Automaattiviritysohjelman tulee kyetä virittämään järjestelmä uudelleen, mikäli systeemin dynamiikka muuttuu.

Yllä olevien ehtojen täyttymiseen vaikuttaa suurelta osin prosessin mallintamisen onnistuminen. Automaattivirityksen haasteet näkyvät erityisesti järjestelmissä, joiden dynamiikan mallintaminen onnistuneesti on lähes mahdotonta. [11, s. 8–9]

Automaattisesti viritetty järjestelmä ei siis aina takaa parhaita tuloksia kaikkiin järjestelmiin. Manuaalinen viritys mahdollistaa säätimen parametrien valinnan siten, että systeemi on optimoitu toimimaan tietyllä tavalla. Automaattiviritys ei esimerkiksi aina pyri minimoimaan systeemin vasteaikaa, vaikka se olisi systeemille olennaista. Manuaalisesti viritettynä parametrit voidaan valita mahdollisimman nopean vasteajan mukaan. Automaattiviritys ei siten takaa järjestelmästä riippumatta tehokkainta mahdollista viritystä vaan yleispätevästi toimivan ohjauksen.

Automaattivirityksen etuna on sen toistettavuus ja säännönmukaisuus. Tyypillisesti automaattiviritysalgoritmit perustuvat tutkittuihin viritysmenetelmiin, mikä tekee viritysprosessista tieteellisemmän. Useimmat automaattiviritysohjelmat tuottavat toimivat alkuarvot, joita voidaan manuaalisesti säätää prosessin mukaan. [12]

Äärimmäistä tarkkuutta ja nopeutta vaativissa tehtävissä automaattinen viritys ei siten välttämättä ole paras vaihtoehto. Manuaalisen virityksen mahdollistama valinnaisuus optimointisuunnan suhteen tekee siitä tietyille järjestelmille paremman ratkaisun.

### 3.2. Systemin mallintaminen

Systemin viritykseen liittyy poikkeuksetta viritettävän järjestelmän mallintaminen jollain tasolla. Matemaattisen mallin tarkoitus on jäljitellä todellisen systemin reagointia ohjaussignaaliin. Ohjaitavan systemin dynamiikka määrittää minkä kertaluokan mallia mallinnuksessa kannattaa hyödyntää. Mallin parametrien valitseminen systemiä kuvaavaksi on virityksen kannalta avainasemassa.

Tarkastellaan ensimmäisen kertaluokan viiveellistä systemimallia, myöhemmin FOPDT (first order plus dead time). Yleisen FOPTD-systemin siirtofunktio malli on muotoa

$$G(s) = \frac{K_G}{\tau s + 1} e^{-ds}, \quad (3.1)$$

missä  $K_G$  on systemin vahvistus,  $\tau$  aikavakio, ja  $d$  viive.  $s$  on Laplace-muunnosoperaattori. Edellä mainituille parametreille pyritään määrittämään arvot siten, että malli kuvaa systemin todellista dynamiikkaa. [13]

Yleisen toisen kertaluokan viiveellisen systemin (SOPDT, Second Order Plus Dead Time) siirtofunktio malli on muotoa

$$G(s) = \frac{K_G}{as^2 + bs + c} e^{-ds}, \quad (3.2)$$

missä parametrit  $a$ ,  $b$  ja  $c$  määrittävät siirtofunktion napojen sijainnit kompleksitasossa ja siten vaikuttavat askelvastekäyrän muotoon. [13]

Parametrien määrittämiselle on useita eri tapoja, jotka jakautuvat joko taajuustason tai aikataason menetelmiin. Taajuustasossa prosessin dynamiikkaa tarkastellaan saattamalla systemi värähtelevään tilaan. Värähtelevästä taajuusvasteesta voidaan päätellä systemin ominaisuuksia joko laskennallisilla menetelmin, tai tarkastelemalla vasteen käyrän muotoa.

Aikataason menetelmissä systemin sisäänmenoksi asetetaan yksinkertainen askelfunktio, joka tuottaa systemistä askelvasteen. Yksinkertainen sisäänmeno mahdol-

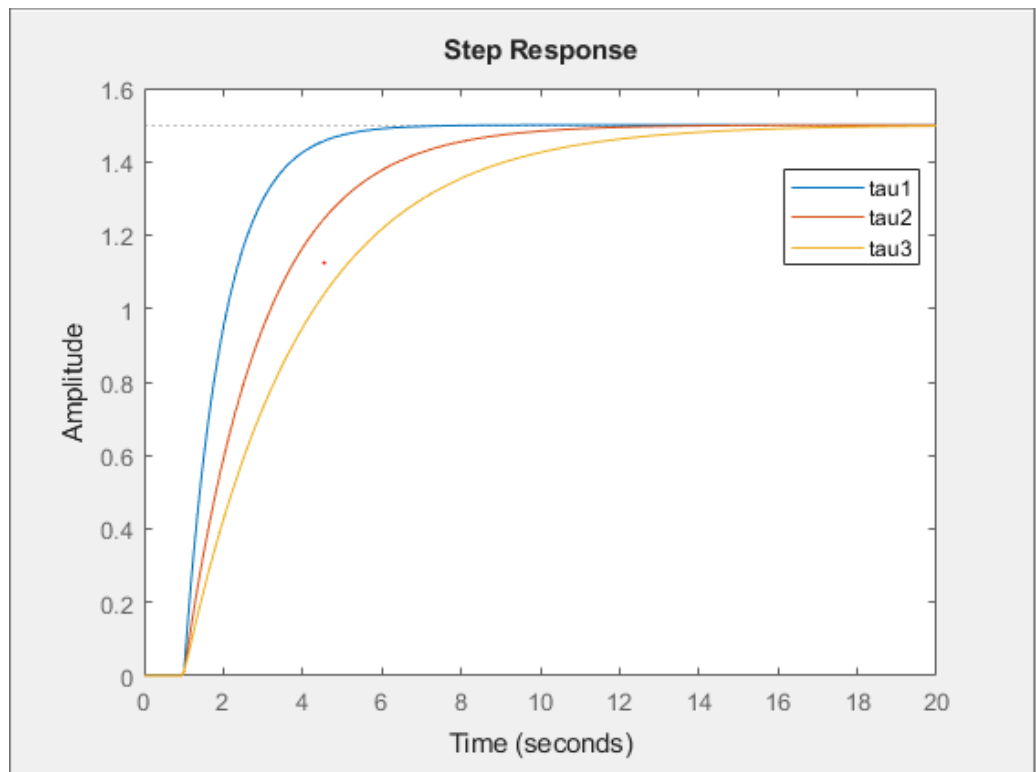
listaa systeemin ominaisuuksien päättelemisen suoraan kuvaajasta, sekä laske-  
malla. Kummankin tason menetelmät ovat laajasti käytössä. Luvussa 3.3 perehdy-  
tään tarkemmin menetelmien yksityiskohtiin.

### **3.3. Virityksen menetelmät**

#### **3.3.1. Ziegler–Nichols ja askelvastemenetelmä**

Ziegler ja Nichols (ZN) kehittivät vuonna 1942 menetelmän prosessin mallin määrit-  
tämiseksi, sekä laskusäännöt säätimen parametreille hyödyntämällä järjestelmän as-  
kelvastetta. Mallin 3.1 mukaisen prosessin parametrien määrittäminen onnistuu syöt-  
tämällä järjestelmän sisäänmenoksi askelfunktio. Prosessin ulostulosta eli askelvas-  
teesta voidaan määrittää FOPTD-mallille tarvittavat parametrit.

Mikäli askelfunktioksi valitaan yksikköaskelfunktio, askelvasteen loppuarvo on aina  
FOPDT-systeemissä vahvistus  $K_G$ . Systeemin viive  $d$  voidaan määrittää mittaamalla  
järjestelmän reagointiaika askelfunktion sisäänmenohetkestä alkaen. Aikavakio  $\tau$  ku-  
vaa sitä ajankohtaa, kun vaste on saavuttanut 63.2 % loppuarvostaan. Toisin sanoen  
 $\tau$  kuvaa systeemin transienttivaiheen nopeutta. [13] Kuvassa 3.1 on FOPDT-systeey-  
min askelvaste kolmella eri aikavakiolla. ZN-askelvastemenetelmän mukaiset PID-  
vahvistusparametrit määrittävät  $d:n$ , sekä  $\tau:n$  taulukon 3.2 mukaan.



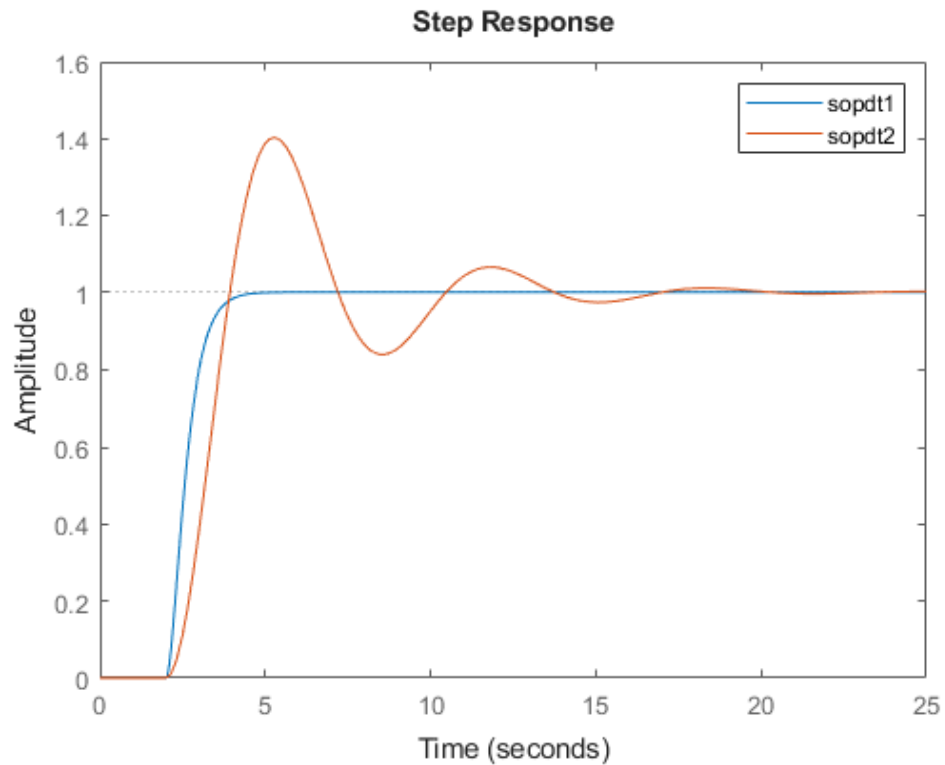
**Kuva 3.1** kolmen eri aikavakion FOPDT-systeemin askelvaste.

**Taulukko 3.2** ZN-askelvastemenetelmän virityssäännöt [14].

Ziegler–Nichols	$K_p$	$\tau_I$	$\tau_d$
P	$\frac{\tau}{d}$	$\infty$	0
PI	$0.9 \frac{\tau}{d}$	$\frac{d}{0.3}$	0
PID	$1.2 \frac{\tau}{d}$	2s	0.5d

SOPDT-systeemeissä askelvaste ei tyypillisesti ole enää monotoninen, vaan käyrässä ilmenee värähtelyä. SOPDT-systeemien tarkastelussa ilmenee edellä mainittujen ominaisuuksien lisäksi vaimennussuhde  $\zeta$  ja luonnollinen kulmataajuus  $\omega_n$ , jotka määräytyvät nimittäjän napojen ominaisuuksista.  $\zeta$  suuruus määrittää värähte-

lyn voimakkuuden. [13] Pieni  $\zeta$ -arvo näyttäyty askelvasteessa voimakkaana värähtelynä. Kuvassa 3.2 on sekä kriittisesti vaimennetun (sopdt1), että alivaimennetun (sopdt2) SOPDT-systeemin askelvasteet.



**Kuva 3.2** Kriittisesti vaimennetun (sopdt1) ja alivaimennetun (sopdt2) SOPDT-systeemin askelvasteet.

### 3.3.2. Ziegler–Nichols-taajuusvastekoe

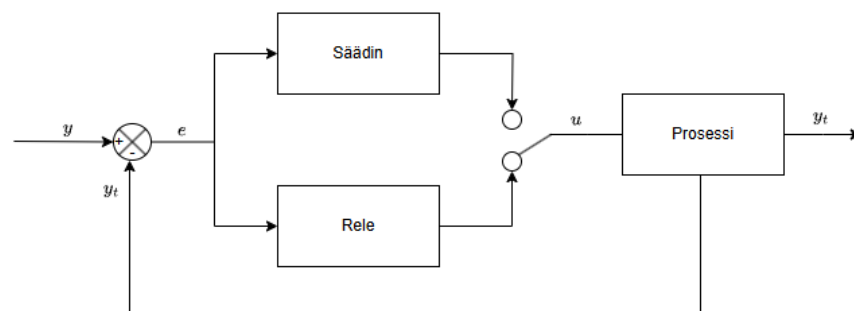
Säätimen parametrit voidaan määrittää selvittämällä piste, jossa Nyquist-käyrä leikkaa negatiivisen reaaliakselin. Pistettä kuvataan kriittisellä jaksonajalla  $P_u$  ja kriittisellä vahvistuksella  $K_u$ . ZN-taajuusvastemenetelmän mukaan järjestelmän kriittinen vahvistus ja kriittinen jaksonaika voidaan määrittää seuraavasti: P-säätimen vahvistusta nostetaan, kunnes saavutetaan värähtelyn tila. Värähtelyn aiheuttava vahvistus on  $K_u$ , ja värähtelyn jaksonaika on systeemin  $P_u$ . [15, s.17–18] Taulukossa 3.3 on listattu P-, PI- ja PID-säätimen parametrien laskentakaavat ZN-taajuusvastemenetelmän mukaisesti.

**Taulukko 3.3** ZN-taajuusvastemenetelmän virityssäännöt [15]

Ziegler–Nichols	$K_p$	$\tau_I$	$\tau_d$
P	$\frac{K_u}{2}$	0	0
PI	$\frac{K_u}{2.2}$	$\frac{P_u}{1.2}$	0
PID	$\frac{K_u}{1.7}$	$\frac{P_u}{2}$	$\frac{P_u}{8}$

### 3.3.3. Automatisoitu taajuusvastemenetelmä

Teoksessaan [16] Åström ja Hägglund esittelevät vaihtoehtoisen menetelmän ZN-menetelmälle. Alkuperäinen ZN-menetelmä saattaa aiheuttaa epävakautta alivaimennetuilla systeemeillä [15 s.18] ja sen toteuttaminen automaattisena prosessina on hankalaa [16, s. 2]. Åströmin ja Hägglundin (ÅH) menetelmä perustuu releohjatun (kuvassa 3.3) taajuusvasteen tarkasteluun.

**Kuva 3.3** Releohjattu järjestelmä

Vaiheviiveen ollessa  $-180$  astetta, releohjattu järjestelmä värähtelee kriittisellä jaksonajalla  $P_u$ . Jos releen amplitudi on  $h$ , Fourierin sarjakehitelmän perusteella releohjauksen tuottaman perusvärähtelyn amplitudi on  $\frac{4h}{\pi}$ . Järjestelmän amplitudin ollessa  $a$  kriittisen vahvistuksen voidaan approksimoida olevan

$$K_u = \frac{4h}{\pi a}. \quad (3.2)$$

ZN virityssääntöjä voidaan soveltaa vastaavalla tavalla ÅH-menetelmällä ratkaistuihin termeihin. ÅH-menetelmän etuna verrattuna ZN-menetelmään on se, että manuaalisesti ohjatun P-vahvistuksen sijaan systeemiä ohjataan releellä. Tämä poistaa riskin epästabiiliudesta, mikäli releen amplitudi on valittu oikein. [16, s. 3]

## 4. RATKAISUT KAUPALLISISSA TUOTTEISSA

Tässä luvussa esitellään teollisuuden järjestelmissä käytössä olevia automaattiviritimiä ja niiden toimintaperiaatteita. On yleistä, että kaupallisten servomoottoriohjaimien automaattivirtimien toimintaa tai rakennetta ei ole tarkennettu, mikä rajoittaa katsauksen laajuuden tunnistettavissa oleviin menetelmiin.

Todennäköisesti yleisin hajautetuissa ohjausjärjestelmissä käytössä oleva automaattivirtin on Åströmin ja Hägglundin vuoden 1984 julkaisun "Automatic tuning of simple regulators with specifications on phase and amplitude margins" mukainen. ABB:n ohjausjärjestelmät ABB ECA600, ABB 800 XA, sekä Emersonin DeltaV perustuvat kaikki ÅH-menetelmään. [17]

ÅH-menetelmän mukaiset virttimet vaikuttavat olevan yleisesti käytössä prosessitekniikan alalla, mikä viittaa siihen, että menetelmä toimii parhaiten hitaampaa ja vaakaampaa ohjausta vaativissa järjestelmissä. Yleisesti taajuustasoon perustuvat menetelmät ovat sovellettavissa nopeampaan dynamiikkaan.

Tässä luvussa pyritään täsmentämään, toteutetaanko kaupallisen automaattivirtimen identifiointivaihe vasteen perusteella taajuustasossa vai aikatasossa. Lisäksi ratkaisujen helppokäyttöisyyttä ja mahdollisia käyttäjältä vaadittuja ennakkotietoja arvioidaan.

### 4.1. Beckhoff TE5960

Beckhoffin TwinCAT 3 ohjelmaan sisäänrakennettu TE5960 automaattivirtitys perustuu stabiilisuuskriteerien ja systeemin ominaisuuksien, kuten vaiheen ja amplitudin tulkintaan Bode-diagrammista. TwinCAT 3 on servomoottoriohjaukseen suunniteltu

ohjelma, joka tarjoaa käyttäjälle laajan valikoiman ohjaukseen liittyviä asetuksia. Automaattiviritysprosessi määrittää systeemin kuorman hitauden, ohjaimen vahvistusparametrit, sekä tarvittavat taajuussuodattimet. [18]

Automaattiviritysprosessin toimintaa on kuvattu vain yleisellä tasolla, Bode-diagrammin tulkinta, sekä vaiheen ja amplitudin määrittäminen kuitenkin viittaavat taajuustason toteutukseen.

## 4.2. Maxon motor EPOS Studio Auto tune

Maxon motorin kehittämä EPOS Studio-ohjelma tarjoaa käyttäjälle laajan kattauksen vaihtoehtoja EPOS4-ohjaimien viritykseen. Ohjelman Auto tune prosessi ei vaadi käyttäjältä tietoa säätötekniikasta, vaan kaikki tarvittavat parametrit arvioidaan systeemin takaisinkytkennän avulla. EPOS4-sarjan ohjaimissa on käytössä nopeus-, asento- ja virtaohjaus. Lisäksi ohjaimissa on sovellettu nopeuden ja kiihtyvyyden etusyöttöä. [19, s.14–15]

Auto Tune-prosessi jakautuu omaksi vaiheekseen jokaiselle ohjaussilmukalle.

1. Virtaohjain. Ensimmäisenä prosessi suorittaa identifioinnin systeemin sähköiselle vasteelle. Seuraavaksi virtaohjaimen parametrit lasketaan identifioinnin perusteella. Lopuksi systeemiin syötetään testisignaali, jolla varmistetaan ohjauksen toimivuus.
2. Nopeusohjain. Ensimmäisenä ohjelma suorittaa systeemin identifioinnin saattamalla systeemin värähtelevään tilaan erilaisilla taajuuksilla ja amplitudeilla. Systeemin taajuusvasteesta ratkaistaan parametrit säätimelle. Lopuksi systeemiin syötetään testisignaali.
3. Asento-ohjain. Vastaavasti aiempiin vaiheisiin, systeemi identifioidaan taajuustason menetelmiä käyttäen. Identifioinnin perusteella säätimelle lasketaan parametrit. Lopuksi systeemiin syötetään testisignaali.

EPOS Studion Auto Tune-prosessin laskennallisia menetelmiä ei ole julkaistu tarkemmin. Vaiheissa 2 ja 3 on käytössä taajuustason menetelmä, jolla systeemin identifiointi suoritetaan. Lasketun mallin parametrejä sovelletaan virityssääntöihin, joiden avulla säätimen vahvistukset ratkaistaan. Virityssääntöjä ei ole tarkennettu.

Z. Zeroual esittelee julkaisuussaan ”maxon Auto Tuning” [20] maxonin EPOS2-sarjan ohjaimien rakennetta. Säätämällä alipäästösuotimen aikavakiota, sekä vahvistusta, voidaan luoda järjestelmän taajuusvastetta vastaava värähtely. Värähtelystä voidaan määrittää malliin tarvittavat parametrit. [20]

### **4.3. Mitsubishi MR-J4 Auto-tuning**

Mitsubishin MR-J4 Servo amplifier ohjaimen toteutettu Auto-tuning-funktio on erityisesti resonanssitaajuuksien ja äänihaittojen eliminointiin erikoistunut automaattiviritysohjelma. MR-J4 tarjoaa käyttäjälle kolmen eri skaalan automaattiviritysvaihtoehtoa: Off-line auto-tuning, reactive continuous auto-tuning ja proactive online auto-tuning.

Off-line auto-tuning on lyhimmän aikaskaalan ratkaisu, missä ohjain teettää moottorille ennalta määritellyn sarjan liikkeitä, jonka perusteella systeemille luodaan malli. Mallin avulla lasketaan säätimelle parametrit. [21] Systeemin identifiointi perustuu kuorman ja moottorin hitauden suhteen arviointiin. Moottoria ohjataan kiihtymään ja hidastumaan sarjoissa, mikä mahdollistaa suhteen arvioinnin systeemin ulostulosta. ohjausparametrit valitaan automaattisesti sisäisen vahvistustaulukon arvojen mukaisesti. [22, 6. 9–10]

Mitsubishin automaattiviritysratkaisussa viritettävää systeemiä ei saateta värähtelävään tilaan, mikä viittaa aikatazon ratkaisuun.

#### **4.4. Schneider electric autotuning**

Tässä luvussa perehdytään Schneider electricin Lexium32-sarjan servomoottoriohjaimiin toteutettuun automaattiviritysohjelmaan. Automaattiviritys selvittää ensimmäisenä järjestelmän mekaaniset resonanssitaajuuudet ja asettaa niille suodattimen. Seuraavana tuottamalla sarjan liikkeitä moottorille, automaattiviritys arvioi systeemissä olevan kitkan, sekä kuorman massan. Moottoria ohjataan joko positiiviseen tai negatiiviseen suuntaan tietty matka, jonka jälkeen moottoria ohjataan vastakkaiseen suuntaan kaksinkertainen matka. Ohjaussykliä toistetaan useita kertoja. Koko järjestelmän hitaus arvioidaan kiihdyttämällä systeemiä. [23] Näiden mittausten perusteella systeemille luodaan todennäköisesti malli, jonka perusteella viritetään ohjaussilmukat.

Ratkaisussa käyttäjältä ei vaadita mitään ennakkotietoa viritettävästä systeemistä. Systeemiä ei saateta värähtelevään tilaan, mikä viittaa siihen, että taajuustason menetelmiä ei sovelleta virityksen toteutuksessa.

#### **4.5. Automationdirect SureServo2 autotuning**

Automationdirectin automaattiviritys perustuu moottorin ja kuorman hitauden arviointiin aikatasossa. Ennen automaattivirityksen aloittamista SureServo2-ohjelmaan tulee syöttää moottorin arvioitu pyörimisnopeus, sekä kiihdytyksen ja hidastuksen suuruudet työkuormassa. Syötettyjen arvojen on tarkoitus vastata moottorin käyttötarkoituksessa ilmenevää liikettä, mikä edesauttaa automaattivirityksen soveltuvuutta juuri käyttäjän tarkoituksiin. Lisäksi systeemille tulee asettaa kaksi asentopistettä, joiden välillä automaattiviritin liikuttaa moottoria.

Alkuarvojen syöttämisen jälkeen systeemin voi viritellä. Automaattiviritys ohjaa moottoria asentopisteiden välillä asetettujen arvojen mukaan. Systeemin vasteesta arvioidaan iteratiivisesti hitautta, kitkaa, kuorman suuruutta sekä resonanssitaajuuksia. [24]

Automaattiviritysohjelma vaatii muihin tutkittaviin ratkaisuihin verrattuna käyttäjältä enemmän tietämystä viritettävän systeemin ominaisuuksista. On myös mahdollista, että näiden ennakkotietojen perusteella kyseinen ohjelma kykenee verrattaessa parempiin tuloksiin. Systeemin mallin identifiointivaiheessa ei todennäköisesti sovelleta taajuustason menetelmiä, sillä systeemiä ei saateta värähtelevään tilaan.

#### **4.6. Kaupallisten ratkaisujen yhteenveto**

Sekä maxon motorsin, että Beckhoffin automaattiviritysratkaisut perustuivat selvästi taajuustason menetelmiin ja siten taajuusvasteen arviointiin. Molemmat automaattiviritysohjelmat ohjasivat moottoria värisemään useassa jaksossa eri amplitudeilla ja taajuuksilla. Viritettävät servomoottorihajaimet sisälsivät virta-, nopeus- ja asentosilmukat.

Mitsubishin, schneider electricin, sekä AutomatioDirectin ratkaisut perusutivat moottorin ja kuorman hitauden arviointiin aikatasossa. Automaattivirityksiä yhdisti moottorin ajaminen vastakkaisiin suuntiin sykleissä, ja sen perusteella suoritettavan kitkan ja pysyvän kuorman arviointi. Lisäksi kiihdytyksien ja hidastamisen hyödyntäminen hitauden arvioinnissa oli yhtenäinen ominaisuus. Kaikissa aikatason ratkaisuissa ohjaimissa oli käytössä virta-, nopeus- ja asentosilmukat.

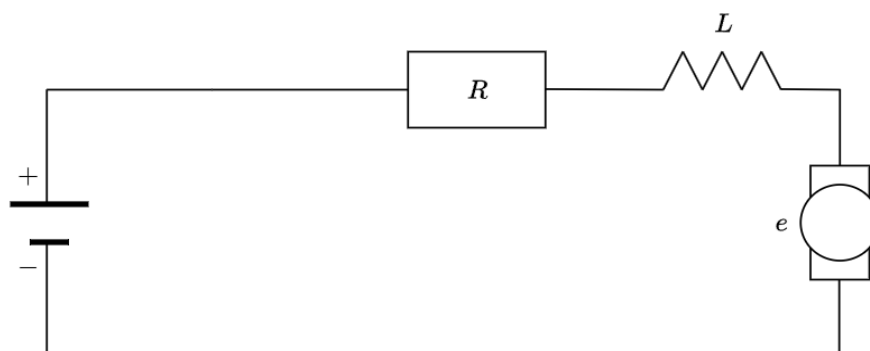
## 5. SIMULAATIO

Closed-loop PID autotuner on Mathworksin Simulink-ohjelmaan rakennettu suljetun silmukan PID-säätimen viritystoiminto. Toiminnon avulla mallin säätimeen voidaan hakea sopivat vahvistusparametrit  $P$ ,  $I$  ja  $D$ -haaroihin automaattisesti. Automaattivirittäjän toiminta perustuu taajuusvastemenetelmään, joka voidaan suorittaa simuloimalla, tai vastaavasti reaaliajassa fyysisen järjestelmän ollessa kytkettynä.

Simuloinnin tarkoituksena on tutkia automaattivirittäjän toimintaa yksinkertaisen DC-moottorin siirtofunktion, sekä PID-säätimen avulla. Seuraavassa luvussa johdetaan DC-moottorille siirtofunktio ja luodaan Simulink-ohjelmaan järjestelmän malli.

### 5.1. DC-moottori

Simulaatiossa mallinnetaan DC-sähkömoottoria, joka on kytkettynä PID-säätimeen. Kuvassa 5.1 on yksinkertaistettu DC-moottorin piiri.



**Kuva 5.1** DC-moottorin virtapiiri

Simulaatiossa halutaan mallintaa moottorin ohjausjännitteen suhdetta akselin kulmanopeuteen. Luodaan systeemille sopiva siirtofunktio. DC-moottorin tuottama vääntömomentti  $T$  ( $Nm$ ) on

$$T = Ki, \quad (5.1)$$

missä  $i$  (A) on systeemissä kulkeva sähkövirta, ja  $K$  ( $\frac{Nm}{A}$ ) vääntömomenttivakio. Newtonin II laista saadaan vääntömomentille säilymlakiyhtälö

$$Ki = J\ddot{\theta} + b\dot{\theta}, \quad (5.2)$$

missä  $J$  ( $kgm^2$ ) on moottorin hitaus,  $\dot{\theta}$  (rpm) akselin kulmanopeus ja  $\ddot{\theta}$  ( $rpm^2$ ) kulmakiiktyvyys.  $b$  (Nms) on kitkakerroin. Lisäksi Kirchhoffin jännitelaista saadaan

$$V_{in} = V_r + V_L + e \quad (5.3)$$

$$V_{in} = Ri + L\frac{di}{dt} + e, \quad (5.4)$$

missä  $V_{in}$  (V) on jännitelähteen jännite,  $R$  ( $\Omega$ ) on vastuksen resistanssi ja  $L$  (H) käämin induktanssi.  $e$  (V) on moottorin vastasähkömotorinen voima, joka on verrannollinen moottorin akselin kulmanopeuteen seuraavasti:

$$e = K_{emf}\dot{\theta}. \quad (5.5)$$

missä  $K_{emf}$  ( $\frac{V}{rpm}$ ) on kulmanopeuden ja vastasähkömotorisen voiman välinen verrannollisuuskerroin. Laplace-muuntamalla yhtälöt 5.2, 5.4 ja 5.5 saadaan

$$KI(s) = Js^2\theta(s) + bs\theta(s)$$

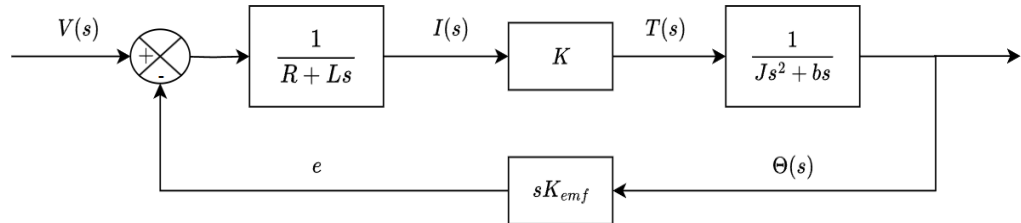
$$\Rightarrow \theta(s) = \frac{KI(s)}{Js^2 + bs} \quad (5.6)$$

ja

$$V(s) = RI(s) + LsI(s) + sK_{emf}\theta(s)$$

$$\Rightarrow I(s) = \frac{1}{R+Ls}(V(s) - sK_{emf}\theta(s)) \quad (5.7)$$

Systeemin sisäänmenoksi on valittu jännitelähteen jännite  $U(s) = V(s)$  ja ulostuloksi moottorin akselin kulmanopeus  $Y(s) = \dot{\Theta}(s)$ . Kuvassa 5.2 systeemiä on kuvattu lohkokaaviolla.



Kuva 5.2 Prosessin lohkokaavio.

Nyt moottorin avoimen silmukan siirtofunktioksi saadaan

$$G(s) = \frac{1}{R+Ls} \cdot K \cdot \frac{1}{Js+b} \quad (5.8)$$

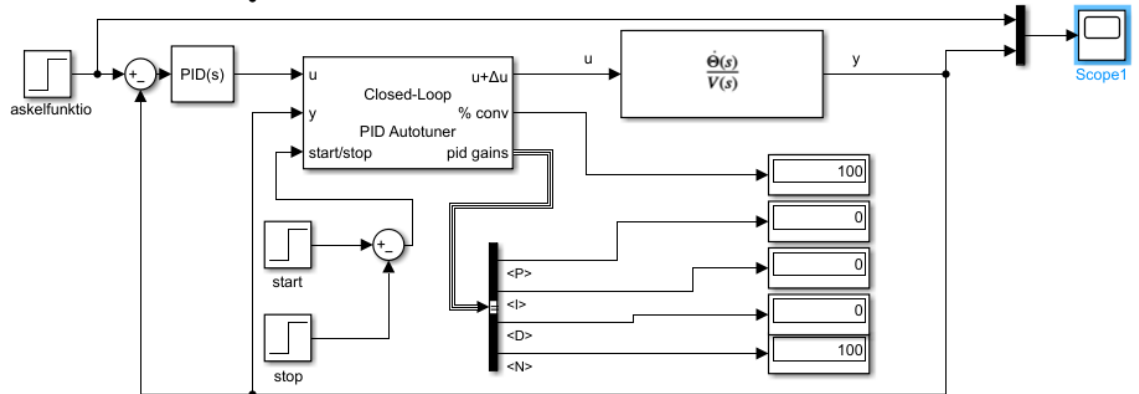
ja takaisinkytketyn systeemin siirtofunktioksi saadaan

$$\frac{Y(s)}{U(s)} = \frac{\dot{\Theta}(s)}{V(s)} = \frac{G(s)}{1+G(s) \cdot H(s)}, \quad (5.9)$$

missä  $H(s) = K_{emf}$  ja  $\dot{\Theta}(s) = s\Theta(s)$ . Seuraa

$$\begin{aligned} \frac{\dot{\Theta}(s)}{V(s)} &= \frac{\frac{1}{R+Ls} \frac{1}{Js+b} K}{1 + \frac{1}{R+Ls} \frac{1}{Js+b} K \cdot K_{emf}} \\ \Rightarrow \frac{\dot{\Theta}(s)}{V(s)} &= \frac{K}{(R+Ls)(Js+b) + KK_{emf}} \\ \Rightarrow \frac{\dot{\Theta}(s)}{V(s)} &= \frac{K}{RJs + Rb + LJs^2 + Lsb + KK_{emf}} \\ \Rightarrow \frac{\dot{\Theta}(s)}{V(s)} &= \frac{K}{LJs^2 + (RJ+Lb)s + Rb + KK_{emf}} \end{aligned} \quad (5.10)$$

Yhtälöä 5.10 käytetään prosessin siirtofunktiona Simulink-simulaatiossa. Parametreille valitaan esimerkkiarvot kuvaamaan kahta tyypillistä moottoria.



Kuva 5.3. Järjestelmän simulaattori

Kuvassa 5.3 on simulaattori Simulink-ohjelmassa. Automaattivirittäjä on kytketty PID-säätimen ja prosessin väliin. Herätteenä käytetään yksikköaskelfunktiota. Simuloinnin käynnistyessä automaattivirittäjä lisää askelfunktioon sinimuotoisen häiriösignaalin, jonka perusteella voidaan arvioida prosessin taajuusvaste. Automaattivirittäjä laskee asetetun tavoitekaistanleveyden läheisyydessä olevalle taajuusvasteelle P, I ja D-vahvistukset. [25]

**Taulukko 5.1.** Moottorien A ja B parametrit

	moottori A	moottori B	Yksikkö
kitkakerroin $b$	0.10	0.50	Nms
hitaus $J$	0.01	0.05	$kgm^2$
Vääntömomenttivakio $K$	0.10	0.10	$Nm/A$
Verrannollisuuskerroin $K_{emf}$	0.10	0.5	$V/rpm$
Resistanssi $R$	1.00	2.00	$\Omega$
Käämin induktanssi $L$	0.50	0.80	$H$

Kahden eri moottorin parametrit ovat listattu taulukkoon 5.1. Moottori A on ominaisuuksiltaan pienempi ja nopeampi kuin moottori B.

## 5.2. Simulaation PID-säädin

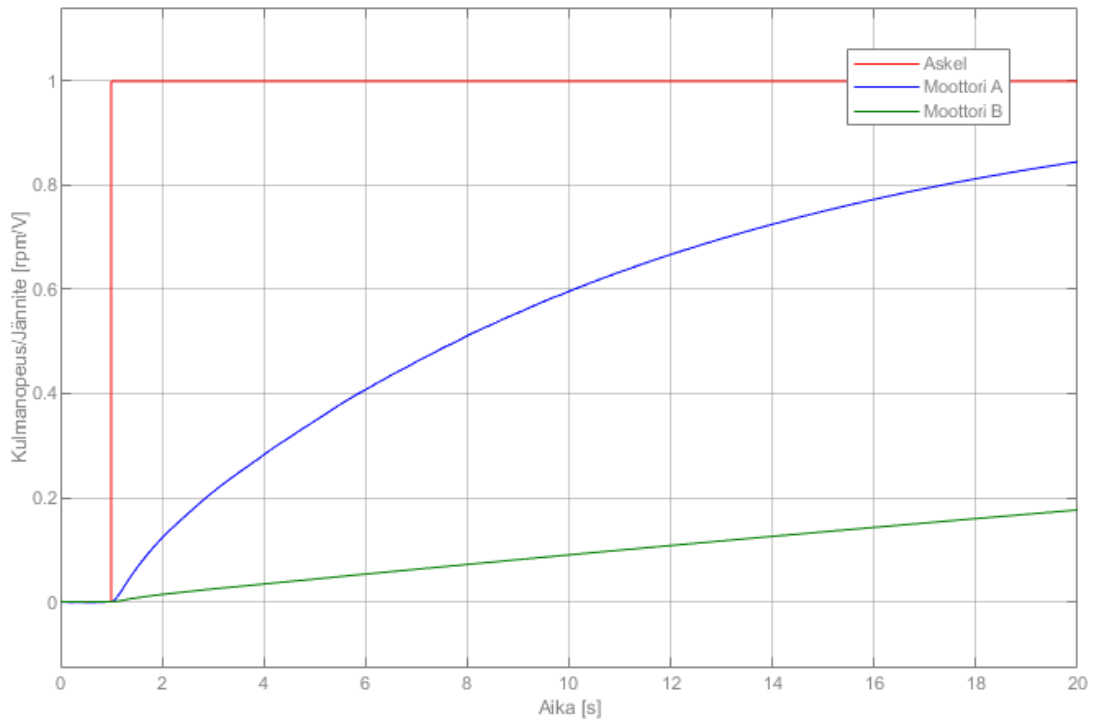
Simulaatiossa käytössä oleva PID-säädin on ominaisuuksiltaan rinnakkainen ja jatkuva-aikainen. Säätimen siirtofunktio on muotoa

$$C_{par}(s) = P + I \frac{1}{s} + D \frac{Ns}{s+N},$$

missä  $P$ ,  $I$  ja  $D$  ovat säätimen vahvistusparametrit, ja  $N$  on säätimen derivaattorisuodattimen kerroin. Simulink käyttää kerrointa apumuuttujana säätimen derivointitoteutuksessa, sekä mahdollistaa taajuusmuokatun säädön toteuttamisen. [26]

### 5.3. Tulokset

Molemmat moottorit ajettiin aluksi virittämättömien PID-säätimien kanssa. Simulinkin PID-säätimen oletusparametrit ovat  $P = 1$ ,  $I = 1$ ,  $D = 0$ ,  $N = 100$  [26]. Kuvassa 9. on moottorien A ja B askelvasteet, sekä ohjaussignaali ennen automaattiviritystä.

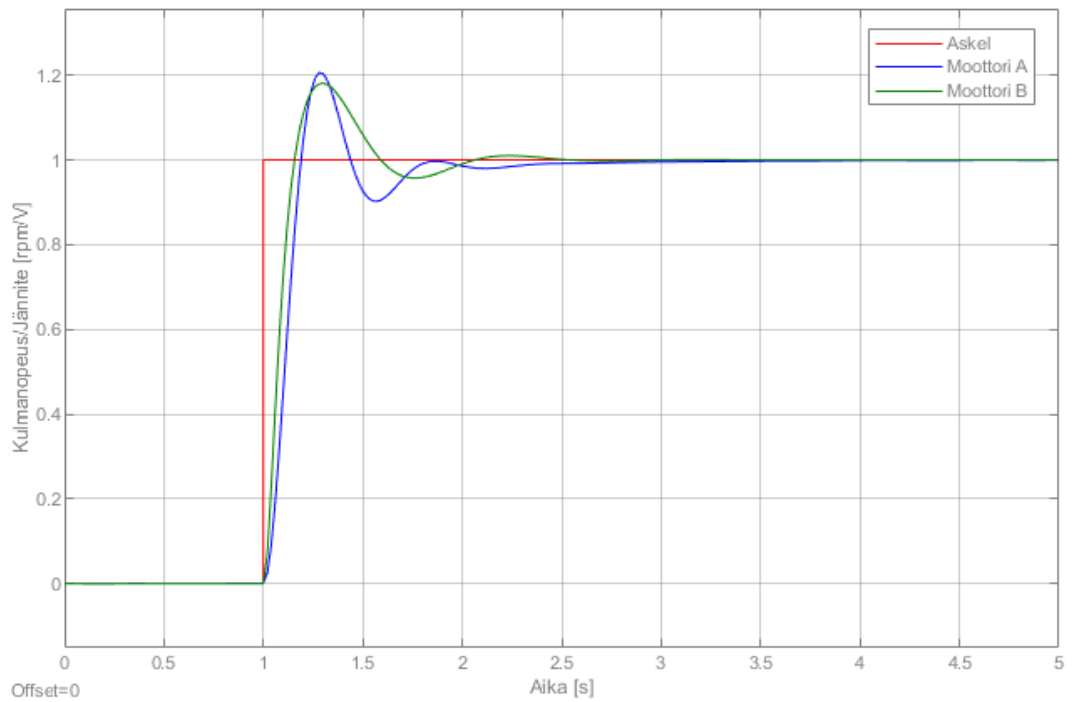


**Kuva 5.4** Moottorien A ja B askelvasteet ennen automaattiviritystä.

Molempien moottorien askelvasteet ovat monotonisia virittämättömien säätimien kanssa. Moottorin A askelvaste on kuitenkin huomattavasti jyrkempi moottorin B askelvasteeseen verrattuna, mikä johtuu A-moottorin pienemmästä inertiaasta.

#### 5.3.1. Kierros 1

1. Simulointikierroksen jälkeen molemmat systeemit oskilloivat hieman ennen asetusta tavoitearvoon. Kuvasta 5.5 voidaan katsoa moottorilla A olevan kaksi suurempaa ylitystä, moottorilla B vain yksi. Molemmat systeemit saavuttavat asetuksen hetkellä  $t = 2s$ .



**Kuva 5.5.** systeemien A ja B askelvasteet 1. simulointikierröksen jälkeen

Taulukkoon 5.2 on listattu automaattivirittäjän tuottamat parametrit ensimmäisen kierroksen jälkeen.

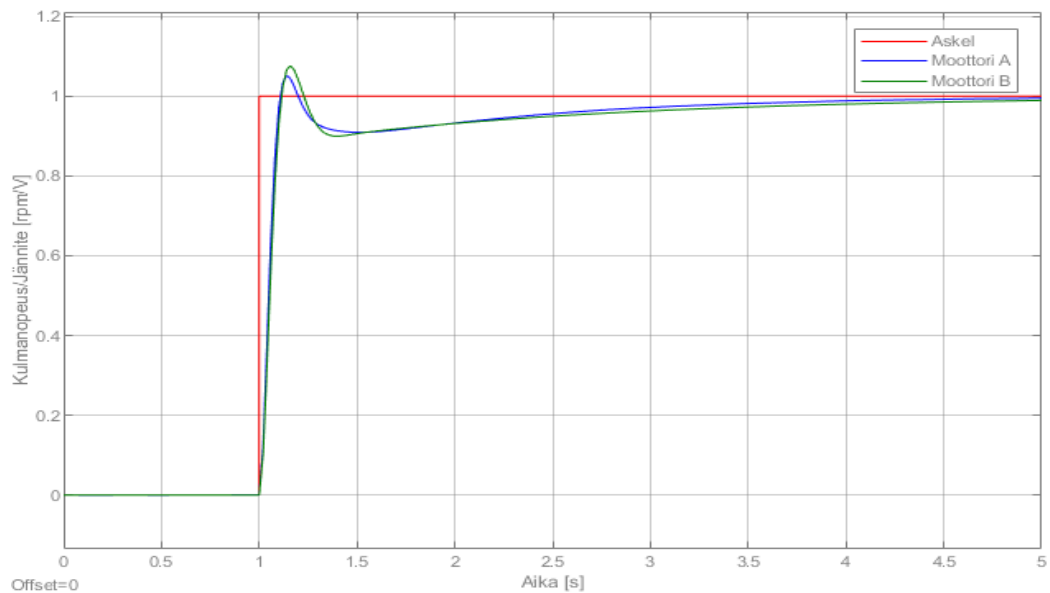
**Taulukko 5.2** Automaattivirittäjän parametrit A- ja B-systeemeille 1. kierroksen jälkeen.

Vahvistukset	Moottori A	Moottori B
P	69.19	410.41
I	91.03	2774.03
D	0.00	26.05
N	100.00	44.97

Koska molemmat systeemit värähtelevät suhteellisen voimakkaasti, suoritetaan toinen simulointikierrös ja otetaan käyttöön ensimmäisen kierroksen parametrit.

### 5.3.2. Kierros 2

Toisella simulointikierroksella molemmat systeemit ylittävät tavoitearvon vain kerran. Vastekäyrät muistuttavat toisiaan. Kuvassa 5.6 on molempien systeemien vastekäyrät, sekä ohjaussignaali.



**Kuva 5.6** Moottorien A ja B askelvasteet 2. simulointikierroksen jälkeen

Kummankin moottorin vaste nousee aggressiivisesti asetusarvon yli. Ylitystä kompensoidaan reilusti, jonka jälkeen vaste nousee loivasti kohti asetusarvoa. Virityksen ansiosta molemmat järjestelmät reagoivat muutokseen nopeasti, mutta ylitys on kohtuullisen suuri. Lisäksi asettumisaika on pitkä ylimenon liiallisen kompensoinnin vuoksi ja verrattuna 1. kierrokseen, molemmat käyrät asettuvat hitaammin. Toisen simulointikierroksen jälkeen automaattivirittäjän tulokset eivät enää muuttuneet merkittävästi. Taulukossa 5. on listattu automaattivirittäjän parametrit systeemien A ja B säätimille.

**Taulukko 5.3** Säästöparametrit A- ja B-systeemeille 2. kierroksen jälkeen.

Vahvistukset	Moottori A	Moottori B
P	61.81	683.20
I	53.59	447.26
D	6.02	35.20
N	44.97	44.97

Huomioitavaa on, että B-moottorin vahvistukset ovat huomattavasti A-moottoria suuremmat. B-moottorin raskaampi ja hitaampi rakenne vaativat säätimeltä aggressiivisempaa ohjausta, mikä näkyy säätimen suuremmissa vahvistusparametreissä. Derivaattorisuodattimen N arvo on sama molemmille systeemeille.

Closed-loop Autotuner tarjoaa käyttäjälle vaihtoehtoja tavoitedynamiikan ominaisuuksille. Tässä simulaatiossa Autotuneria ajettiin PRBS-testiasetuksella, sillä sen havaittiin tuottavan johdonmukaisia ja säännöllisiä tuloksia simulointikierröksillä. Muilta osin automaattiviritintä, sekä PID-säädintä ajettiin oletusasetuksilla.

## 6. YHTEENVETO

Kaupallisten servomoottoriohjainten automaattiviritysratkaisuissa löytyi joitain yhtäläisyyksiä. Kaksi viidestä tutkituista ohjaimesta sovelsi taajuustason viritysmenetelmiä joko täysimittaisena viritysratkaisuna, tai yhdistettynä aikatason viritysmenetelmiin. Loput kolme viritysratkaisua perustuivat aikatason menetelmiin. Kaikille ohjaimille yhteistä oli viritysprosessin karkea rakenne, sekä useamman eri viritysvaihtoehdon tarjonta käyttäjälle. Yleistä oli, että ohjain oli mahdollista virittää joko ennalta määritetyn liikesarjan perusteella tiettyihin asetuksiin, tai pitkäaikaisemmin systeemin käyttöjaksoon soveltuen. Lisäksi etenkin Mitsubishiin MR-J4-ohjaimen viritystoteutuksessa haitallisten resonanssitaajuuksien automaattiseen eliminointiin oli kehitetty edistyneitä ratkaisuja.

Suurin osa esitellyistä automaattiviritysratkaisuista ei vaatinut käyttäjältä tietämystä säätötekniikasta, vaan viritysprosessi tapahtui täysin itsenäisesti. Vain yksi viidestä ohjaimesta vaati käyttäjältä tietoja moottorin käyttöjakson luonteesta, vapaaehtoinen ennakkotietojen syöttö oli mahdollista useammalle ohjaimelle. Automaattivirityksen voidaan sanoa kasvaneen osaksi alan standardia säätötekniikassa. Yhä edistyneempiä ratkaisuja kehitetään jatkuvasti systeemin mallin luomisen, sekä virityssääntöjen toteutukseen.

Tämä katsaus pyrki selvittämään automaattivirityksen toimintaa nimenomaan servomootoreihin liittyvässä ohjauksessa. Kuten luvussa 5 esitettiin, automaattiviritysratkaisut ovat servomoottoriohjauksen lisäksi laajasti käytössä prosessitekniikan sovelluksissa. Jatkotutkimusaiheena voisi olla viritysratkaisujen tutkimisen laajentaminen eri sovellusalueille.

## LÄHTEET

[1] T. Hägglund, Autotuning, In. J. Baillieul & T. Samad (ed.) Encyclopedia of Systems and Control, Springer London, 2015, p. 50–55.

[2] J. McKay, What is a servo drive: basics and working mechanisms, Advanced Motion Controls, 2024, päivitetty 26.6.2024. Saatavissa (viitattu 5.11.2024): <https://www.a-m-c.com/servo-drive-basics/>

[3] D. Collins, FAQ: What are servo motor current, velocity and position loops and bandwidths? Motion Control Tips, 2016, päivitetty 9.6.2016. Saatavissa (viitattu 24.10.2023): <https://www.motioncontroltips.com/faq-servo-motor-current-velocity-position-loops-bandwidths/#:~:text=to%20optimal%20performance.,Current%20loop,speed%2C%20and%20therefore%2C%20position.>

[4] R. Mallik, Robust control approaches for minimizing the bandwidth ratio in multi-loop control, arXiv, 2022. Saatavissa (viitattu 6.11.2024): <https://arxiv.org/abs/2203.09022>

[5] Kollmorgen, What is a closed loop system? 2022. Saatavissa (viitattu 6.11.2024): <https://www.kollmorgen.com/en-us/blogs/what-closed-loop-system>

[6] G.W. Younkin, Industrial Servo Control Systems: Fundamentals and Applications. 2nd ed., Marcel Dekker, 2003.

[7] Tieteen Termipankki, Pulssisuhdesäätö, Pulssinleveysmodulaatio, 2014. Saatavissa (viitattu 13.11.2024): <https://tieteentermi-pankki.fi/wiki/S%C3%A4hk%C3%B6tekniikka:pulssisuhdes%C3%A4%C3%A4t%C3%B6>

[8] BYJU'S, Pulse width modulation. 2024. Saatavissa (viitattu 3.12.2024): <https://byjus.com/physics/pulse-width-modulation/>

[9] K. Åström, T. Hägglund, PID Controllers - Theory, Design, and Tuning (2nd Edition). ISA, 1995.

- [10] R. Garrido, R. Miranda. Autotuning of a DC Servomechanism Using Closed Loop Identification, IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics, IEEE, 2006, pp. 258–63.
- [11] K. Papadopoulos, PID controller tuning using the magnitude optimum criterion, Springer, 2014.
- [12] C. Lewin, Tuning servomotors, Motion system design, v. 48, n. 6, 2006, pp.22–26.
- [13] V.-P. Pyrhönen, Mittaus: Luento 1, Tampereen yliopisto, 2024.
- [14] Skedbooks, Ziegler-Nichols first method for tuning PID controllers, 2024. Saatavissa (viitattu 3.12.2024): <https://skedbooks.com/books/control-systems-1/ziegler-nichols-first-method-for-tuning-pid-controllers/>
- [15] C. Yu, Autotuning of PID Controllers, A Relay Feedback Approach, 2<sup>nd</sup> Edition, Springer, 2006.
- [16] K. Åström, T. Hägglund, Automatic Tuning of Simple Regulators with Specifications on Phase and Amplitude Margins. Automatica, v. 20, n. 5, 1984, pp. 645–651.
- [17] J. Berner, K. Soltesz, T. Hägglund, K. Åström, An experimental comparison of PID autotuners, Control Engineering Practice, v. 73, 2018, pp. 124–133.
- [18] Beckhoff, TE5960, TwinCAT 3 Autotuning, 2024. Saatavissa (viitattu 18.11.2024): <https://www.beckhoff.com/fi-fi/products/automation/twincat/texxxx-twincat-3-engineering/te5960.html>
- [19] maxon motor, EPOS4 application notes, 2017. Saatavissa (viitattu 18.11.2024): [https://www.maxongroup.net.au/medias/sys\\_master/8825356812318.pdf](https://www.maxongroup.net.au/medias/sys_master/8825356812318.pdf)
- [20] Z. Zeroual, Automatic Controller Setting for Electric Drives, in iNaCoMM2013, 2013.
- [21] Mitsubishi, MR-J4-\_TM\_SERVO AMPLIFIER INSTRUCTION MANUAL, 2015

[22] Mitsubishi, Auto-Tuning Servos Optimize Machine Performance Effortlessly, 2017

[23] Schneider electrics, How to autotune a LXM32 servo drive through SoMove? 2016, päivitetty 2024. Saatavissa (viitattu 26.11.2024):  
<https://www.se.com/in/en/faqs/FA292680/>

[24] AutomationDirect, SureServo2 User Manual – 2<sup>nd</sup> Ed. Rev A – 01/30/2024, chapter 5: Tuning, 2024

[25] MathWorks, Closed-loop PID autotuner, 2024. Saatavissa (viitattu 4.12.2024):  
<https://www.mathworks.com/help/slcontrol/ug/closedlooppidautotuner.html>

[26]: Mathworks, pid, 2010. Saatavissa (viitattu 4.12.2024):  
[https://www.mathworks.com/help/control/ref/pid.html?s\\_tid=doc\\_ta](https://www.mathworks.com/help/control/ref/pid.html?s_tid=doc_ta)

## 6.1. LIITE A: DC-MOOTTORIEN A JA B MÄÄRITYS SEKÄ FOPDT- JA SOPDT-SYSTEEMIEN ASKELVASTEET, MATLAB

```

%DC-moottori A
b = 0.1;
J = 0.01;
K = 0.01;
K_emf = 0.095;
R = 1;
L = 0.5;

%DC-moottori B
b_b = 0.5;
J_b = 0.05;
K_b = 0.01;
K_emf_b = 0.477;
R_b = 2;
L_b = 0.8;

%siirtofunktiot A ja B
s = tf('s');
P_motor_A = K/(L*J*s^2+(R*J+L*K)*s+R*b+K*K_emf)
P_motor_B = K_b/(L_b*J_b*s^2+(R_b*J_b+L_b*K_b)*s+R_b*b_b+K_b*K_emf_b)

%FOPDT-askelvasteet
tau1 = 1/(s+1)
tau2 = 1/(2*s+1)
tau3 = 1/(3*s+1)
step(tau1)
hold on
step(tau2)
step(tau3)
hold off

%SOPDT-askelvasteet
sopdt1 = 9/(s^2+6*s+9)*exp(-2*s)
sopdt2 = 9/(9*s^2+5*s+9)*exp(-2*s)
step(sopdt1)
hold on
step(sopdt2)
hold off

```