



TUGAS AKHIR - TF 141581

PERANCANGAN *PASSIVE FAULT TOLERANT CONTROL* (PFTC) PADA PENGENDALIAN KECEPATAN SISTEM MOTOR DC DENGAN KESALAHAN PADA AKTUATOR

DIYAJENG LULUK KARLINA
NRP. 0231144000012

Dosen Pembimbing:
Dr. Katherin Indriawati, S.T., M.T.

DEPARTEMEN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2018

Halaman ini sengaja dikosongkan



FINAL PROJECT - TF 141581

**PASSIVE FAULT TOLERANT CONTROL (PFTC)
DESIGN FOR VELOCITY CONTROL OF DC
MOTOR SYSTEM WITH FAULT ACTUATOR**

DIYAJENG LULUK KARLINA
NRP. 0231144000012

Supervisors :
Dr. Katherin Indriawati, S.T., M.T.

ENGINEERING PHYSICS DEPARTMENT
Faculty of Industrial Technology
Sepuluh Nopember Institute Technology
Surabaya 2018

Halaman ini sengaja dikosongkan

PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini:

Nama : Diyajeng Luluk Karlina
NRP : 0231144000012
Departemen/ Prodi : Teknik Fisika/ S1 Teknik Fisika
Fakultas : Fakultas Teknologi Industri
Perguruan Tinggi : Institut Teknologi Sepuluh Nopember

Dengan ini menyatakan bahwa Tugas Akhir dengan judul "Perancangan *Passive Fault Tolerant Control* Pada Pengendalian Kecepatan Sistem Motor DC Dengan Kesalahan Pada Aktuator" adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain. Apabila di kemudian hari terbukti terdapat plagiat pada tugas akhir ini, maka saya bersedia menerima sanksi sesuai ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.

Surabaya, 8 Januari 2018
Yang membuat pernyataan,



Diyajeng Luluk Karlina
NRP.0231144000012

Halaman ini sengaja dikosongkan

**LEMBAR PENGESAHAN
TUGAS AKHIR**

**PERANCANGAN *PASSIVE FAULT TOLERANT
CONTROL* (PFTC) PADA PENGENDALIAN
KECEPATAN SISTEM MOTOR DC DENGAN
KESALAHAN PADA AKTUATOR**

Oleh:

Divajeng Luluk Karlina

NRP. 02311440000012

Surabaya, 8 Januari 2018

Menyetujui,
Dosen Pembimbing

Dr. Katherin Indriawati, S.T., M.T.

NIPN. 19760523 200012 2 001

Mengetahui,
Kepala Departemen
Teknik Fisika FTI ITS



Agus M. Huda, S.T., M.Si., Ph.D

NIPN. 19780907 200312 1 002

Halaman ini sengaja dikosongkan

**PERANCANGAN *PASSIVE FAULT TOLERANT CONTROL*
(PFTC) PADA PENGENDALIAN KECEPATAN SISTEM
MOTOR DC DENGAN KESALAHAN PADA AKTUATOR**

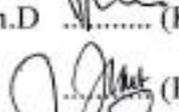
TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
pada
Bidang Studi Rekayasa Instrumentasi
Program Studi S-1 Departemen Teknik Fisika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Oleh:

DIYAJENG LULUK KARLINA
NRP. 0231144000012

Disetujui oleh Tim Penguji Tugas Akhir:

1. Dr. Katherin Indriawati, S.T., M.T.  (Pembimbing)
2. Totok Ruki Biyanto, S.T., M.T., Ph.D.  (Ketua Penguji)
3. Dr. Imam Abadi, S.T., M.T.  (Penguji)
4. Ir. Harsono, M.Sc., Ph.D.  (Penguji)

Halaman ini sengaja dikosongkan

PERANCANGAN *PASSIVE FAULT TOLERANT CONTROL* (PFTC) PADA PENGENDALIAN KECEPATAN SISTEM MOTOR DC DENGAN KESALAHAN PADA AKTUATOR

Nama : Diyajeng Luluk Karlina
NRP : 0231144000012
Departemen : Teknik Fisika FTI-ITS
Dosen Pembimbing : Dr. Katherin Indriawati, S.T., M.T.

Abstrak

Hingga saat ini gangguan yang terjadi pada motor DC masih dapat ditangani dengan menggunakan sistem pengendali umpan balik (konvensional). Jika gangguan berupa kesalahan pada komponen aktuator maka sistem pengendali konvensional tidak dapat mengatasi gangguan tersebut. Pada tugas akhir ini dilakukan perancangan sistem *Passive Fault Tolerant Control* (PFTC) yang mampu bekerja ketika terdapat kesalahan pada aktuator pada pengendalian kecepatan sistem motor DC sehingga performansi sistem tetap terjaga. Langkah pertama yang dilakukan adalah merancang sistem kontrol *state feedback* dengan *integrator* menghasilkan nilai K_p sebesar 224,99 dan 19 sedangkan K_i sebesar 2,5. Langkah berikutnya adalah merancang *observer* yang berfungsi mengestimasi kesalahan. Sistem PFTC bekerja dengan melakukan kompensasi terhadap sinyal kontrol dengan menggunakan hasil estimasi *observer*. Penerapan sistem PFTC menghasilkan nilai *maximum undershoot* dan nilai *settling time* lebih kecil dibandingkan dengan sistem tanpa PFTC yaitu 10,16% dan 4,4 detik. Untuk sistem dengan PFTC dan sistem tanpa PFTC tidak terdapat *error steady state*. Sistem PFTC mampu menangani kesalahan *bias* sebesar 100% dan kesalahan *loss of effectiveness* sebesar 95% pada aktuator. Selain itu PFTC mampu menangani ketidakpastian sebesar 10%.

Kata Kunci: Aktuator, *State Feedback* dengan *Integrator*, Sistem Motor DC, Sistem PFTC

Halaman ini sengaja dikosongkan

PASSIVE FAULT TOLERANT CONTROL DESIGN FOR VELOCITY CONTROL OF DC MOTOR SYSTEM WITH FAULT ACTUATOR

Name : Diyajeng Luluk Karlina
NRP : 0231144000012
Department : Teknik Fisika FTI-ITS
Supervisors : Dr. Katherin Indriawati, S.T., M.T.

Abstract

Currently, the disturbance that occurs in DC motors can still be handled using conventional feedback controller systems. If the disturbance is in a form of fault occurring in the actuator component, the conventional control system will not be able to overcome the disturbance. In this research, the Passive Fault Tolerant Control (PFTC) system that is able to work when there is an fault on the actuator on DC motor system speed control is designed so that the system performance is maintained. The first step is designing a state feedback control system with integrators yielding the K_p values of 224,99 and 19 while the K_i value is 2,5. The next step is to design an observer that works to estimate the fault. The PFTC system works by compensating the control signal by using the observer estimation results. The implementation of PFTC system results in smaller value of maximum undershoot and settling time compared to system without PFTC that is 10,16% and 4,4 seconds. For systems with PFTC and systems without PFTC there is no steady state error. The PFTC system is able to handle 100% bias fault and 95% loss of effectiveness in actuator. In addition, PFTC is able to handle uncertainty by 10%.

Keywords: Actuator, State Feedback with Integrator, Motor DC System, PFTC system

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang senantiasa melimpahkan rahmat serta hidayah-Nya, serta sholawat serta salam kepada Nabi Muhammad SAW, hingga terselesaikannya tugas akhir beserta laporan tugas akhir yang berjudul

PERANCANGAN *PASSIVE FAULT TOLERANT CONTROL* PADA PENGENDALIAN KECEPATAN SISTEM MOTOR DC DENGAN KESALAHAN PADA AKTUATOR

Penulis telah banyak memperoleh bantuan dari berbagai pihak dalam penyelesaian tugas akhir dan laporan Tugas Akhir ini. Penulis mengucapkan terimakasih kepada :

1. Bapak Agus Muhamad Hatta, S.T., M.Si., Ph.D. selaku ketua departemen teknik fisika yang telah memberikan petunjuk, ilmu, serta bimbingan selama menempuh pendidikan di teknik fisika.
2. Ibu Dr. Katherin Indriawati, S.T., M.T. selaku dosen pembimbing yang telah dengan sabar memberikan petunjuk, ilmu, serta bimbingan yang sangat bermanfaat.
3. Bapak Totok Ruki Biyanto, S.T., M.T., Ph.D. selaku kepala Laboratorium Rekayasa Instrumensi yang telah memberikan ilmu, petunjuk, nasihat, serta kemudahan perizinan.
4. Bapak Dr. Ir. Ali Musyafa' M.Sc. selaku dosen wali yang telah membimbing penulis selama perkuliahan.
5. Kedua orang tua (Bapak Agus Riyanto dan Ibu Nurlia). Terimakasih atas segala cinta, kasih sayang, doa, perhatian, serta dukungan moril dan materiil yang telah diberikan
6. Seluruh teman – teman Departemen Teknik Fisika angkatan 2014, terima kasih untuk semuanya
7. Seluruh dosen, karyawan dan civitas akademik Teknik Fisika, terimakasih atas segala bantuan dan kerjasamanya.
8. Semua pihak yang tidak dapat disebutkan satu persatu, terimakasih atas bantuannya.

Penulis sadar bahwa penulisan laporan tugas akhir ini tidak sempurna, namun semoga laporan ini dapat memberikan kontribusi yang berarti dan menambah wawasan yang bermanfaat bagi pembaca, keluarga besar Teknik Fisika khususnya, dan civitas akademik ITS pada umumnya. Semoga laporan tugas akhir ini dapat bermanfaat sebagai referensi pengerjaan laporan tugas akhir bagi mahasiswa yang lain.

Surabaya, 8 Januari 2018

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
<i>TITLE PAGE</i>	iii
LEMBAR PENGESAHAN I.....	Error! Bookmark not defined.
LEMBAR PENGESAHAN II.....	Error! Bookmark not defined.
Abstrak.....	xi
<i>Abstract</i>	xiii
KATA PENGANTAR.....	xv
DAFTAR ISI.....	xvii
DAFTAR GAMBAR.....	xix
DAFTAR NOTASI.....	xxiii
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Batasan Masalah.....	2
1.5 Sistematika Laporan.....	3
TINJAUAN PUSTAKA.....	5
2.1 Motor DC.....	5
2.2 Model Matematik Motor DC.....	7
2.3 <i>State Feedback</i> dengan <i>Integrator</i>	11
2.4 Konsep Kestabilan dan Performansi <i>Robust</i>	12
2.5 Sistem <i>Passive Fault Tolerant Control</i>	23
2.6 <i>Observer State</i>	24
2.7 Optimisasi dengan LMI.....	27
METODOLOGI PENELITIAN.....	29
3.1 Pemodelan Motor DC.....	30
3.2 Perancangan Sistem Kontrol <i>State Feedback</i>	32
3.3 Perancangan <i>Robust Observer</i>	34
3.4 Uji Performansi PFTC.....	44
ANALISIS DATA DAN PEMBAHASAN.....	45
4.1 Uji <i>Tracking</i> Kontroler <i>State Feedback</i>	45

4.2	Hail Estimasi <i>Observer</i>	46
4.3	Uji Performanis PFTC.....	47
KESIMPULAN DAN SARAN		55
5.1	Kesimpulan.....	55
5.2	Saran.....	55
DAFTAR PUSTAKA.....		57
LAMPIRAN		
BIODATA PENULIS		

DAFTAR GAMBAR

Gambar 2.1	Motor DC	6
Gambar 2.2	Rangkaian motor DC	8
Gambar 2.3	Diagram blok kontrol state feedback dengan integrator	11
Gambar 2.4	Diagram sistem kontrol secara umum	13
Gambar 2.5	Konfigurasi gangguan dalam bentuk penambahan	14
Gambar 2.6	Konfigurasi gangguan dalam bentuk perkalian input	15
Gambar 2.7	Sistem kontrol robust secara umum	16
Gambar 2.8	Sistem dengan Struktur $M-\Delta$	16
Gambar 2.9	Kestabilan robust	18
Gambar 2.10	Diagram kontrol robust	19
Gambar 2.11	Ilustrasi grafik luasan H_2 -norm	22
Gambar 2.12	Ilustrasi grafik luasan H_∞ norm	23
Gambar 3.1	Diagram alir penelitian tugas akhir	29
Gambar 3.2	Diagram blok sistem pengendalian	30
Gambar 3.3	Struktur extended state observer	37
Gambar 4.1	Respon sistem perancangan sistem kontrol state feedback dengan integrator secara simulasi	45
Gambar 4.2	Hasil estimasi observer tanpa kesalahan	46
Gambar 4.3	Hasil estimasi observer dengan kesalahan	46
Gambar 4.4	Respon sistem dengan setpoint 0,5 rad/detik dan kesalahan bias 100% secara simulasi	48
Gambar 4.5	Respon sistem dengan setpoint 2 rad/detik dan kesalahan bias 100% secara simulasi	49
Gambar 4.6	Respon sistem dengan setpoint 0,5 rad/detik dan kesalahan loss of effectiveness 95% secara simulasi	50

- Gambar 4.7** Respon sistem dengan setpoint 2 rad/detik dan kesalahan loss of effectiveness 95% secara simulasi51
- Gambar 4. 8** Respon sistem dengan PFTC dan setpoint 0,5 rad/detik dan ketidakpastian 10% secara simulasi53
- Gambar 4. 9** Respon sistem dengan PFTC dan setpoint 2 rad/detik dan ketidakpastian 10% secara simulasi53

DAFTAR TABEL

- Tabel 4.1** Performansi sistem dengan setpoint 0,5 rad/detik dan kesalahan bias 100% secara simulasi48
- Tabel 4.2** Performansi sistem dengan setpoint 0,5 rad/detik dan kesalahan loss of effectiveness 95% secara simulasi 50
- Tabel 4.3** Performansi sistem dengan setpoint 2 rad/detik dan kesalahan loss of effectiveness 95% secara simulasi 52

Halaman ini sengaja dikosongkan

DAFTAR NOTASI

b	= Koefisien gesekan viskos ekivalen dari motor atau beban pada poros motor (N-m/rad/s)
Ψ	= Fluks
K_f	= Konstanta fluks
i_f	= Arus fluks (ampere)
T	= Torsi (N-m)
K_t	= Konstanta torsi motor
K_b	= Konstanta emf balik
R	= Tahanan kumparan jangkar, (ohm)
J	= Momen inersia eqivalen dari motor, (Kg.m ²)
ω	= Kecepatan rotasi, (rpm)
V	= Tegangan input servo (volt)
e_a	= Tegangan yang dikenakan pada jangkar (volt)
e_b	= Gaya gerak listrik balik (volt)
L	= Induktansi kumparan jangkar (henry)
i_a	= Arus kumparan jangkar (ampere)
A	= Matriks keadaan (<i>state</i>)
B	= Matriks kendali
C	= Matriks keluaran
D	= Matriks gangguan
x	= Vektor keadaan (<i>state</i>)
$u(t)$	= Masukan kendali
$y(t)$	= Vektor keluaran
$x(t)$	= vektor keadaan (<i>state</i>)
A	= Matriks keadaan
B	= Matriks <i>input</i>
C	= Matriks <i>output</i>
D_d	= Matriks gangguan
D_ω	= Matriks <i>noise</i>
F_a	= Matriks kesalahan actuator
$Gm(s)$	= <i>Plant</i> Nominal
(Δ)	= Gangguan

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Motor DC (*Direct Current*) merupakan sebuah perangkat elektromagnetis yang berfungsi untuk mengubah energi listrik menjadi energi mekanik. Motor DC banyak digunakan dalam bidang industri, peralatan rumah tangga dan komponen elektronik lainnya (Koksal, 2007). Selain itu, penggunaan motor DC sudah diterapkan dalam dunia otomatis berupa penggunaan motor DC untuk mobil listrik.

Permasalahan yang sering terjadi di dunia industri tentang penerapan motor DC yaitu berupa gangguan secara mendadak, yang disebabkan oleh *disturbance* dan *noise*. Hingga saat ini gangguan tersebut masih dapat ditangani dengan menggunakan sistem pengendali umpan balik (konvensional). Namun, jika gangguan berupa kesalahan pada komponen seperti sensor dan aktuator maka sistem pengendali konvensional tidak dapat mengatasi gangguan tersebut (Zhang, Y & Jiang, J, 2008). Untuk mengatasi kelemahan pada sistem pengendali PID, dilakukan pendekatan baru dalam desain sistem pengendalian, dengan cara mentolerir kerusakan komponen sehingga performa sistem tetap stabil. Sistem pengendalian yang mempertahankan tingkat kestabilan dan performansi sistem pada saat terjadi kesalahan pada komponen sensor dan aktuator disebut *Fault Tolerant Control System* (FTCS) (Indriawati, Agustinah, & Jazidie, 2015).

Secara umum terdapat dua metode dalam *Fault Tolerant Control* (FTC) yaitu *Passive Fault Tolerant Control* (PFTC) dan *Active Fault Tolerant Control* (AFTC) (Zhang, Y & Jiang, J, 2008). Pada *Passive Fault Tolerant Control* (PFTC) didesain untuk menjadi sistem pengendali yang *robust* dari kesalahan suatu komponen. Pada awalnya teori *robust* tidak memperhitungkan kesalahan sensor dan aktuator, tetapi memperhitungkan ketidakpastian (*uncertainty*). Sedangkan untuk teori *Passive Fault Tolerant Control* (PFTC) pada awalnya diperlukan perhitungan

kesalahan sensor dan aktuator dan juga memperhitungkan ketidakpastian (*uncertainty*).

Pada penelitian sebelumnya mengenai *Passive Fault Tolerant Control* (PFTC) dijelaskan bahwa penelitian tersebut menggunakan kontroler yang *robust* terhadap kesalahan aktuator tanpa menggunakan *observer* (Noboru Sebe, 2012). Oleh karena itu pada tugas akhir ini dilakukan perancangan *Passive Fault Tolerant Control* (PFTC) pada pengendalian kecepatan sistem motor DC dengan kesalahan pada aktuator dengan menggunakan *observer* yang *robust* terhadap kesalahan aktuator sebagai pengembangan dari penelitian sebelumnya. Sehingga penggunaan *Passive Fault Tolerant Control* pada sistem motor DC mampu menoleransi kesalahan aktuator yang terjadi pada sistem motor DC agar sistem dapat bekerja dengan stabil.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, rumusan masalah dari penulisan tugas akhir ini adalah:

1. Bagaimana mendesain *observer* yang dapat mengestimasi kesalahan dan menghasilkan *state* yang benar ?
2. Bagaimana merancang sistem *Passive Fault Tolerant Control* (PFTC) dengan menggunakan *observer* ?

1.3 Tujuan

Berdasarkan latar belakang dan perumusan masalah, tujuan penelitian yang ingin dicapai dalam pengerjaan tugas akhir ini adalah merancang sistem pengendalian kecepatan pada sistem motor DC menggunakan *Passive Fault Tolerant Control* yang mampu menangani kesalahan pada aktuator sehingga performansi sistem dapat terjaga.

1.4 Batasan Masalah

Berdasarkan identifikasi masalah maka batasan masalah dari tugas akhir ini adalah :

- a. Kesalahan yang terjadi adalah kesalahan minor berupa kesalahan *bias* dan *loss of effectiveness*.

- b. Komponen aktuator yang digunakan adalah *servo amplifier* dengan kesalahan pada suplai tegangan dan penyumbatan yang menyebabkan kesalahan *bias* dan *loss off effectiveness*.
- c. Variabel yang dikontrol adalah kecepatan motor DC.
- d. Variabel yang dimanipulasi adalah tegangan motor DC.
- e. Motor DC yang digunakan adalah jenis magnet permanen..
- f. Ketidakpastian pada motor DC terjadi pada tahanan kumparan jangkar (R).

1.5 Sistematika Laporan

Sistematika penulisan laporan tugas akhir adalah sebagai berikut :

- a. **BAB I PENDAHULUAN**
Pada bab I ini terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan dan sistematika laporan.
- b. **BAB II TINJAUAN PUSTAKA**
Pada bab II ini dibahas mengenai teori-teori yang berkaitan dengan penelitian yang akan dilakukan, seperti teori kontrol *robust*, kontrol *feedback*, ketidakpastian model.
- c. **BAB III METODOLOGI PENELITIAN**
Pada bab III ini berisi mengenai rancangan dari penelitian yang dilakukan, metode dan langkah-langkah dalam penelitian.
- d. **BAB IV ANALISA DATA DAN PEMBAHASAN**
Pada bab IV ini berisi tentang analisis hasil perancangan sistem *passive fault tolerant control*.
- e. **BAB V KESIMPULAN DAN SARAN**
Pada bab V ini diberikan kesimpulan tentang tugas akhir yang telah dilakukan berdasarkan hasil yang diperoleh, serta diberikan saran sebagai penunjang maupun pengembangan tugas akhir selanjutnya.

Halaman ini sengaja dikosongkan

BAB II

TINJAUAN PUSTAKA

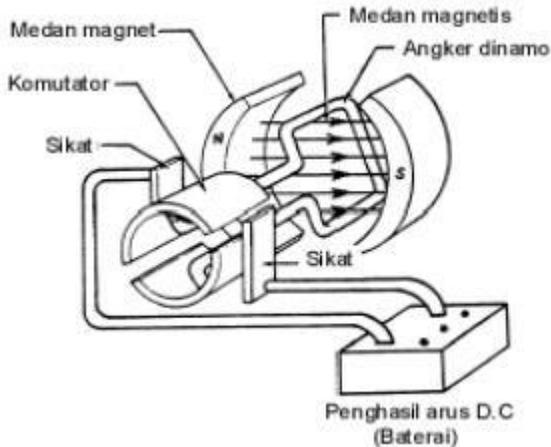
2.1 Motor DC

Motor DC merupakan perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik (Sugiono, 2015). Pengaplikasian motor DC sangat luas, seperti penggerak utama beberapa fungsi mesin industri, penggerak mobil listrik, penggerak peralatan rumah tangga (mesin jahit, kipas angin, pompa). Jenis motor DC yang sering digunakan yaitu motor DC jenis magnet permanen.

Motor DC magnet permanen yang ditunjukkan pada Gambar 2.1 terdiri atas beberapa bagian yaitu kumparan jangkar atau angker dinamo (*armature*), magnet permanen, komutator (*commutator*) dan sikat (*brush*). Prinsip kerja motor DC yaitu magnet permanen sebagai stator motor tersusun dari dua magnet dengan kutub yang berbeda (kutub utara dan kutub selatan), yang saling berhadapan dan berfungsi menghasilkan medan magnet. Komutator merupakan bagian dari rotor motor yang terletak pada kedua ujung kumparan jangkar yang berfungsi untuk mengumpulkan arus induksi dari jangkar dan mengubahnya menjadi arus searah. Sikat berfungsi menyalurkan arus listrik dari sumber diluar motor ke dalam kumparan jangkar (*armature*) dan kumparan jangkar sebagai rotor motor memiliki fungsi untuk merubah energi listrik menjadi energi mekanik dalam bentuk gerak putar.

Dengan adanya arus di dalam kumparan jangkar yang terletak didalam medan magnet yang mengandung fluks, maka akan menghasilkan suatu gaya, yang biasa disebut gaya Lorentz (F), selanjutnya gaya tersebut akan menggerakkan kumparan jangkar untuk berputar. Komponen komutator yang selalu ikut berputar dengan rotor dan sikat karbon yang selalu diam, menjadi komponen yang akan menjaga arah arus listrik untuk selalu tetap yakni mengalir dari sisi kiri kawat angker ke kanan. Arah arus listrik yang selalu tetap di setiap setengah putaran rotor inilah yang akan

membuat rotor motor listrik selalu berputar selama masih ada arus listrik yang mengalir ke kumparan jangkar (Setyaningrum, D, 2012).



Gambar 2.1 Motor DC (Hudaya, 2013)

Sensor yang digunakan pada sistem servo motor DC berupa sensor *tachometer* dan aktuator yang digunakan berupa *servoamplifier*.

2.1.1 *Tachometer*

Tachometer adalah sensor kecepatan yang umum digunakan pada sistem pengendalian motor, yang merupakan suatu perangkat elektronik yang mengubah energi mekanik ke energi listrik. Perangkat ini bekerja sebagai pembangkit tegangan dengan tegangan keluaran sebanding dengan magnituda kecepatan sudut dari batan masukan. Pada sistem pengendalian, kebanyakan menggunakan tachometer jenis DC (misalnya: tegangan keluarannya adalah sinyal DC). Tachometer alat ini biasanya menampilkan revolutions per-minute (RPM) pada sebuah pengukur skala analog, namun yang versi tampilan digital juga sudah semakin populer (Sari, 2009).

2.1.2 *Servo amplifier*

Merupakan aktuator yang digunakan untuk menggerakkan motor DC. *Servo Amplifier* adalah sebuah penguat yang mengolah sinyal error untuk mengoreksi perbedaan antara sinyal input (referensi) dengan sinyal umpan balik (*feedback*) sehingga menghasilkan output yang diharapkan (Setyaningrum, D, 2012).

Kesalahan merupakan penyimpangan yang tak diinginkan dari sifat karakteristik atau parameter sistem. Dalam sistem kontrol, kesalahan dapat terjadi pada aktuator, sensor, dan proses. Kesalahan aktuator dapat terjadi karena suplai tegangan, penurunan kinerja *amplifier* dan penyumbatan. Kesalahan tersebut berkaitan dengan kesalahan *loss of effectiveness* (hilangnya sebagian tekanan hidrolis atau pneumatik) dan kesalahan *bias* pada aktuator (Lee & Lyou, 2001).

Kesalahan *bias* dapat dituliskan dalam bentuk persamaan sebagai berikut,

$$b = b + \omega^b \quad (2.1)$$

Kemudian kesalahan aktuator dapat dituliskan sebagai berikut,

$$B(f)u = Bu + Fb \quad (2.2)$$

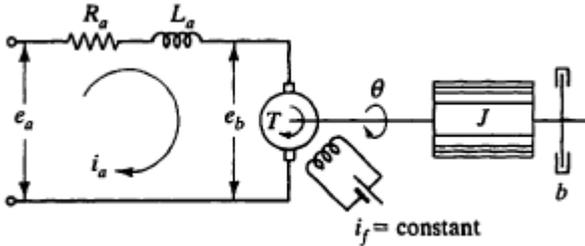
Dimana $F=B$ menunjukkan pengaruh matriks kesalahan aktuator pada vektor keadaan.

2.2 Model Matematik Motor DC

Sistem merupakan kesatuan bagian komponen-komponen yang saling berhubungan dalam menjalankan suatu aksi. Sesuai dengan komponen yang digunakan, suatu sistem dapat berupa sistem mekanik, elektrik, pneumatik, atau elektro-mekanik. Motor DC merupakan salah satu contoh sistem elektro-mekanik.

Salah satu syarat dasar pada kebanyakan strategi yang digunakan dalam merancang suatu sistem kendali adalah

kemampuan dalam memodelkan secara matematika dari sistem yang akan dikendalikan. Model matematika dari suatu sistem dapat berupa persamaan differensial, fungsi transfer, atau ruang keadaan (*state space*) (Ogata, K, 1997).



Gambar 2.2 Rangkaian motor DC (Ogata, 2004)

Berdasarkan Gambar 2.2, tegangan (V) untuk sistem motor DC adalah masukan yang dipasang pada armatur (R , L , θ motor). Tegangan tersebut diubah oleh motor menjadi gerak putar, gerak putar yang dimaksudkan adalah keluaran sistem dan sebagai kecepatan rotasi dari motor. Secara umum, torsi yang dibangkitkan oleh motor DC sebanding terhadap arus motor dan kuat medan magnet, dengan asumsi bahwa medan magnet konstan maka torsi motor sebanding dengan arus motor i dengan faktor konstanta K_t yang dapat dituliskan sebagai berikut,

$$T = K_t i_a \quad (2.3)$$

Jika kumparan magnet berputar maka tegangan menjadi sebanding dengan hasil kali fluks dan kecepatan sudut. Untuk fluks yang konstan, gaya gerak listrik balik e_b berbanding lurus dengan kecepatan rotasi motor ω seperti dituliskan dalam persamaan berikut,

$$e_b = K_b \omega \quad (2.4)$$

Dalam unit SI, konstanta torsi motor dan konstanta gaya gerak listrik emf balik adalah sama, sehingga $K_t = K_b$, dapat disimbolkan dengan konstanta K untuk menyatakan keduanya.

Untuk bagian mekanik, momen inersia rotor (J) dan gesekan viskositas (b) menghasilkan torsi dengan persamaan sebagai berikut,

$$J \frac{d\omega}{dt} + b\omega = T = Ki_a \quad (2.5)$$

Pada sistem motor dc, kecepatan kumparan jangkar dikendalikan oleh tegangan kumparan jangkar e_a , agar kumparan jangkar dapat bergerak, maka e_a melalui rangkaian kumparan jangkar yang terdiri dari hambatan jangkar (R) dan induktansi jangkar (L), sehingga persamaan diferensial rangkaian kumparan jangkar adalah,

$$L \frac{di_a}{dt} + Ri_a + e_b = e_a \quad (2.6)$$

Pada persamaan (2.4) disubstitusikan kedalam persamaan (2.6), menjadi

$$L \frac{di_a}{dt} + Ri_a = V - K\omega \quad (2.7)$$

Persamaan (2.5) dan (2.7) dapat diubah dengan menggunakan transformasi Laplace sebagai berikut,

$$Js\omega(s) + b\omega(s) = KI(s) \quad (2.8)$$

$$LsI(s) + RI(s) = V(s) - K\omega(s) \quad (2.9)$$

Persamaan (2.9) dapat dituliskan dalam bentuk $I(s)$ menjadi,

$$I(s) = \frac{V(s) - K\omega(s)}{R + Ls} \quad (2.10)$$

Substitusi persamaan (2.11) ke persamaan (2.8) menghasilkan persamaan sebagai berikut,

$$Js\omega(s) + b\omega(s) = K \frac{V(s) - K\omega(s)}{R + Ls} \quad (2.11)$$

Fungsi transfer dari masukan tegangan $V(s)$ berbanding dengan kecepatan motor $\omega(s)$ menghasilkan persamaan sebagai berikut,

$$\frac{\omega(s)}{V(s)} = \frac{K}{(R + Ls) + (Js + b) + K.Kb} \quad (2.12)$$

Nilai L terkadang tidak dianggap karena nilainya yang sangat kecil jika dibandingkan dengan nilai R, sehingga

$$\frac{\omega(s)}{V(s)} = \frac{K}{RJs + Rb + K.Kb} \quad (2.13)$$

Persamaan (2.5) dan (2.7) dituliskan dalam bentuk persamaan *state space* untuk kecepatan motor DC dengan kecepatan rotasi dan arus motor sebagai variabel keadaan, dan tegangan sebagai masukan (*input*) serta kecepatan rotasi sebagai keluaran (*output*):

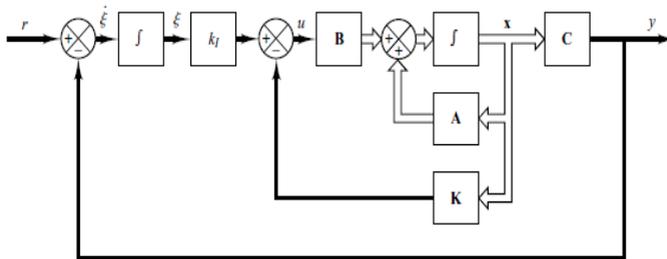
$$\frac{d}{dt} \begin{bmatrix} \omega \\ i_a \end{bmatrix} = \begin{bmatrix} \frac{-b}{j} & \frac{K}{j} \\ -\frac{K}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \omega \\ i_a \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} V \quad (2.14)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ i_a \end{bmatrix} \quad (2.15)$$

Keterangan

- b = koefisien gesekan viskos ekivalen dari motor atau beban pada poros motor (N-m/rad/s)
 Ψ = Fluks
 i_f = arus fluks (ampere)
 T = Torsi (N-m)
 K_t = Konstanta torsi motor
 K_b = Konstanta emf balik
 R = Tahanan kumparan jangkar, (ohm)
 J = Momen inersia eqivalen dari motor, (Kg.m²)
 ω = Kecepatan rotasi, (rpm)
 V = Tegangan input servo (volt)
 e_a = Tegangan yang dikenakan pada jangkar (volt)
 e_b = Gaya gerak listrik balik (volt)
 L = Induktansi kumparan jangkar (henry)
 i_a = Arus kumparan jangkar (ampere)

2.3 State Feedback dengan Integrator



Gambar 2.3 Diagram blok kontrol *state feedback* dengan *integrator* (Ogata, K, 1997)

Sistem pengendalian kecepatan pada sistem motor DC menggunakan sistem kontrol *state feedback* dengan *integrator*. Jika dalam suatu sistem ditambahkan dengan *integrator* maka dapat menghilangkan *error steady state* (Ogata, K, 1997).

Dari Gambar 2.3 dapat dituliskan dalam bentuk persamaan untuk kontrol *state feedback* dengan *integrator* sebagai berikut,

$$u = K_p \cdot x + K_i \int e \cdot dt \quad (2.16)$$

dengan

u = Input dari motor DC

K_p = Gain propotional

K_i = Gain integral

x = State

Beberapa karakteristik sistem dalam pengendali *state feedback* yaitu *maximum overshoot*, *settling time*, *time constant* dan *error steady state* (Ogata, K, 2010)

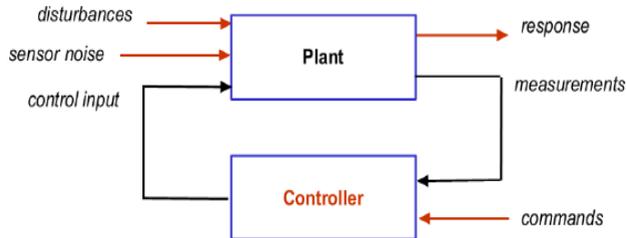
- a. *Maximum (percent) overshoot* adalah nilai puncak tertinggi respon sistem pengukuran terhadap keadaan yang diinginkan. Jika nilai *steady state* dari respon berbeda dengan nilai yang diinginkan, umumnya menggunakan *maximum percent overshoot* dengan persamaan

$$\text{Maximum percent overshoot} = \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\% \quad (2.17)$$

- b. *Settling time* adalah waktu yang dibutuhkan respon sistem untuk mencapai dan berada direntang $\pm 5\%$ atau $\pm 2\%$ dari keadaan yang diinginkan atau nilai akhir (Kresna, 2017).
- c. *Error steady state* adalah nilai *error* yang dihasilkan dari keadaan *steady* ketika belum mencapai *setpoint* yang diinginkan (Ardhiantama, 2016).

2.4 Konsep Kestabilan dan Performansi *Robust*

Sistem pengendalian memiliki tujuan untuk menghasilkan respon keluaran sistem sesuai dengan referensi yang diinginkan. Secara umum blok diagram dari sistem kontrol sebagai berikut (Susanto & Wibawa, 2015):



Gambar 2.4 Diagram sistem kontrol secara umum (Susanto & Wibawa, 2015)

Pada umumnya untuk memodelkan suatu sistem diperlukan asumsi-asumsi untuk mendapatkan model matematis dari sistem sebenarnya. Dengan adanya asumsi-asumsi ini, model yang dirancang akan mengandung ketidakpastian terhadap model sistem yang sesungguhnya. Bentuk ketidakpastian diantaranya efek gangguan pada *plant*, pengukuran kebisingan, error pemodelan karena nonlinearitas, dan error pemodelan karena parameter waktu bervariasi (Burn, 2001). Oleh karena itu perlu dirancang suatu sistem kontrol yang dapat mengantisipasi ketidakpastian model tersebut. Selain itu, kontroler harus mampu mengatasi perubahan parameter dan gangguan.

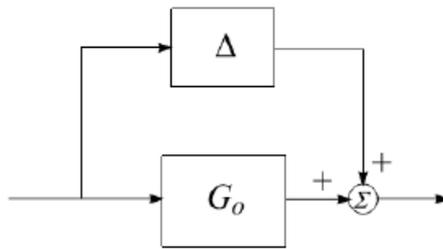
Suatu sistem dikatakan memiliki sifat *robust* jika sistem tersebut memiliki kemampuan bertahan terhadap keadaan disekitar. Jadi, suatu sistem kontrol yang *robust* adalah sistem kontrol yang mampu bekerja dengan baik (mempertahankan kestabilan), walaupun terdapat gangguan dalam pemodelan yang tidak diketahui karakteristiknya.

2.4.1 Ketidakpastian (*Uncertainty*)

Ketidakpastian dapat dibagi ke dalam dua jenis ketidakpastian *nonparametric*, termasuk dalam kategori ini adalah kesalahan pemodelan dan gangguan; dan ketidakpastian *parametric* yang dikarakterisasi dalam bentuk parameter yang tak-diketahui.

Gangguan dinamik yang terjadi pada sistem dapat dibangun menjadi satu blok gangguan (Δ). Diberikan juga sistem actual (sistem dengan gangguan) ($G_P(s)$) dan sistem nominal plant ($G_0(s)$) sehingga dapat dicari fungsi alih gangguan yang terjadi dalam suatu sistem. Fungsi alih gangguan dapat dibagi menjadi dua yaitu:

1. Gangguan dalam bentuk penambahan (*additive*)



Gambar 2.5 Konfigurasi gangguan dalam bentuk Penambahan (Susanto & Wibawa, 2015)

Sehingga gangguan dalam bentuk penambahan dapat dimodelkan sebagai berikut:

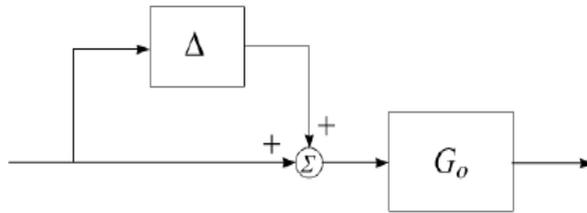
$$G_P(s) = G_0(s) + \Delta(s) \quad (2.18)$$

Ketidakpastian additive dapat disimbolkan dengan $l_a(s)$ sehingga,

$$G(s) = Gm(s) + l_a(s) \quad (2.19)$$

Dimana $Gm(s)$ adalah plant nominal dan l_a adalah ketidakpastian *additive*.

2. Gangguan dalam bentuk perkalian (*multiplicative*)



Gambar 2.6 Konfigurasi gangguan dalam Bentuk Perkalian Input (Susanto & Wibawa, 2015)

Sehingga gangguan dalam bentuk perkalian input dapat dimodelkan sebagai berikut

$$G_p(s) = G_o(s)[I + \Delta(s)] \quad (2.20)$$

Ketidakpastian multiplicative dinotasikan dengan $l_m(s)$ sehingga persamaan menjadi :

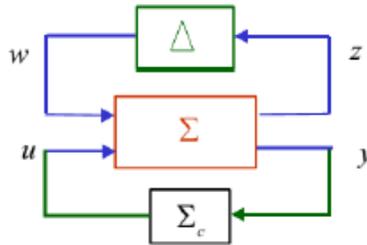
$$G(s) = (1 + l_m(s))G_m(s) \quad (2.21)$$

Substitusikan persamaan (2.20) dan (2.21) didapatkan persamaan sebagai berikut:

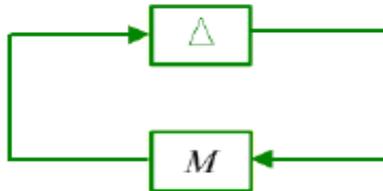
$$l_a(s) = l_m(s)G_m(s) \quad (2.22)$$

2.4.2 Kestabilan *Robust*

Gangguan (Δ) atau ketidakpastian adalah salah satu dari permasalahan stabilitas. Cara memodelkan sistem secara umum dapat dimisalkan dengan sistem yang dihubungkan dengan struktur tiga blok yang ditunjukkan gambar 2.15 dan mengandung blok ketidakpastian (Δ), sedangkan bentuk sistem yang dikoneksikan dalam tiga blok dapat disederhanakan menjadi struktur dua blok M- Δ yang ditunjukkan pada gambar 2.16.



Gambar 2.7 Sistem kontrol robust secara umum
(Susanto & Wibawa, 2015)



Gambar 2.8 Sistem dengan struktur M- Δ
(Susanto & Wibawa, 2015)

Dengan menggunakan teorema *small-gain*, Jika Δ stabil dan $\|\Delta\|_{\infty} \cdot \|M\|_{\infty} < 1$, maka sistem yang terhubung akan stabil. Diasumsikan $\|T_{zw}\|_{\infty} < \gamma$ sistem dengan ketidakpastian tak struktur jika

$$\|T_{zw}\|_{\infty} \cdot \|\Delta\|_{\infty} < \gamma \cdot \|\Delta\|_{\infty} < 1 \rightarrow \|\Delta\|_{\infty} < \frac{1}{\gamma} \quad (2.23)$$

Kestabilan *robust* bisa dicari di domain frekuensi dengan menggunakan kriteria kestabilan *nyquist*. Untuk nominal sistem loop terbuka $G_m(j\omega)C(j\omega)$ dengan model ketidakpastian yang dijelaskan pada persamaan (2.19). Jika diasumsikan $\bar{l}_a(\omega)$ sebagai ketidakpastian *additive* dan menjadi *radius disk* pada nominal *nyquist*, menjelaskan bahwa $G(j\omega)$ terletak pada anggota *plant* $\pi(G(j\omega) \in \pi)$ yang dijelaskan oleh *disk*, yang didefinisikan dalam bentuk persamaan sebagai berikut :

$$\pi = \{G: |G(j\omega) - G_m(j\omega)| \leq \pi) \bar{l}_a(\omega)\} \quad (2.23)$$

Dan selanjutnya,

$$|l_a(j\omega)| \leq \bar{l}_a(\omega) \quad (2.24)$$

Jika ketidakpastian *multiplicative* pada persamaan (2.21) dan (2.22) dituliskan sebagai berikut,

$$lm(j\omega) = \frac{la(j\omega)}{Gm(j\omega) - C(j\omega)} \quad (2.25)$$

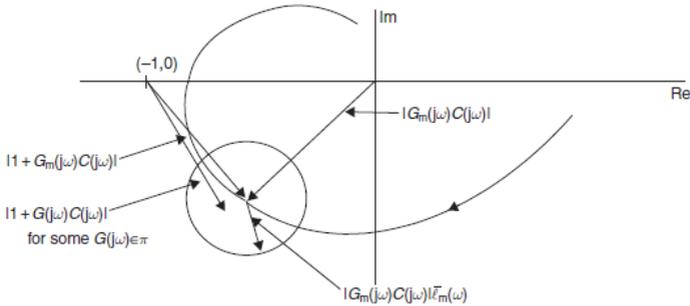
Dan batasan dari ketidakpastian *multiplicative* menjadi persamaan sebagai berikut,

$$\bar{lm}(j\omega) = \frac{\bar{la}(j\omega)}{Gm(j\omega) - C(j\omega)} \quad (2.26)$$

Dari persamaan (2.26) radius *disk* (batasan ketidakpastian) adalah sebagai berikut,

$$\bar{l}_a(\omega) = |G_m(j\omega) - C(j\omega)| \bar{l}_m(\omega) \quad (2.27)$$

Pada kriteria stabilitas *nyquist*, jika dimisalkan $N(k, G(j\omega))$ diartikan jumlah dari penghitung searah jarum jam dari suatu titik $(k, 0)$ dari kontur *nyquist*. Jika diasumsikan bahwa semau plant di anggota π yang dinyatakan dalam persamaan 2, memiliki jumlah yang sama (n) dari right-hand pole (RHP) pole.



Gambar 2.9 Kestabilan *robust* (Burn, 2001)

Terdapat kestabilan *robust* dari *controller* $C(j\omega)$ jika dan hanya jika

$$N(-1, G(j\omega)C(j\omega)) = -n ; G(j\omega) \in \pi \quad (2.28)$$

Untuk menjadikan nominal plant $G_m(j\omega)$ menjadi stabil

$$N(-1, G_m(j\omega)C(j\omega)) = -n \quad (2.29)$$

Pada Gambar 2.9 dijelaskan kestabilan *robust* terjadi bila besaran vector $|1 + G_m(j\omega)C(j\omega)|$ melebihi radius *disk* $|G_m(j\omega)C(j\omega)|\bar{l}_m(\omega)$ dapat dituliskan dalam bentuk persamaan sebagai berikut,

$$|1 + G_m(j\omega)C(j\omega)| > |G_m(j\omega)C(j\omega)|\bar{l}_m(\omega) \quad (2.30)$$

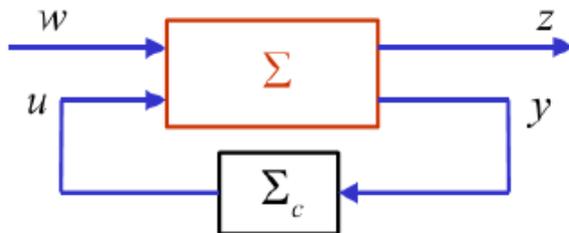
$$\left| \frac{G_m(j\omega)C(j\omega)}{1 + G_m(j\omega)C(j\omega)} \right| \bar{l}_m(\omega) < 1 \quad (2.31)$$

Pada persamaan (2.31) digunakan besarnya fungsi sensitifitas komplemen $T(j\omega)$ yang dituliskan dalam persamaan sebagai berikut:

$$|T(j\omega)|\bar{l}_m(\omega) < 1 \text{ for all } \omega \quad (2.32)$$

2.4.3 Performansi H_2 dan H_∞

Masalah kontrol untuk sistem linier invariant waktu dengan Σ diasumsikan plant dan Σ_c diasumsikan controller (asumsi sistem *stabilizable* dan *detectable*).



Gambar 2.10 Diagram kontrol *robust*
(Susanto & Wibawa, 2015)

Pada Gambar 2.10 dijelaskan bahwa w merupakan *exogenous input* berupa gangguan, sensor *noise*, dan sinyal referensi. Komponen z merupakan semua sinyal yang ingin di kontrol berupa *tracking error* antara sinyal referensi dan output dari plant. Komponen y merupakan output dari dari sensor yang akan masuk ke controller. Sedangkan untuk komponen u berisi semua kontrol input yang masuk ke plant.

Masalah kontrol untuk sistem dengan ketidakpastian adalah mendesain sistem kontrol yang tepat (Σ_c) sehingga ketika dipasangkan ke plant (Σ) dengan gangguan ketidakpastian diperoleh sistem loop tertutup stabil secara internal (kondisi diperlukan untuk sembarang desain sistem kontrol).

Fungsi alih loop tertutup, dari gangguan w terhadap kontroller *output* z , T_{zw} dapat sekecil mungkin, sehingga pengaruh gangguan terhadap keluaran yang dikontrol akan minimum. Fungsi alih memiliki frekuensi berkisar dari 0 sampai ∞ , sehingga cukup sulit menentukan pengaruh tersebut besar atau kecil. Metode praktis yang dapat digunakan yaitu dengan mengukur nilai normnya yaitu H_2 -norm dan H_∞ -norm.

Dari Gambar 2.10 didapatkan persamaan ruang keadaan sistem dinamik *plant* sebagai berikut:

$$\Sigma \begin{cases} \dot{x} = Ax + Bu + Ew \\ y = C_1x + 0u + D_1w \\ z = C_2x + D_2u + 0w \end{cases} \quad (2.33)$$

Dan persamaan sistem dinamik pengendali atau kontroler dapat dituliskan sebagai berikut:

$$\Sigma \begin{cases} \dot{v} = A_c v + B_c y \\ u = C_c v + D_c y \end{cases} \quad (2.34)$$

dengan

x = variabel *state*

v = kontroler *state*

Sehingga \dot{v} dapat dituliskan sebagai berikut:

$$\dot{v} = A_c v + B_c(C_1x + D_1w) \quad (2.35)$$

Selanjutnya persamaan (2.34) disubstitusikan ke persamaan (2.33) sehingga \dot{x}, y, z menjadi:

$$\dot{x} = Ax + B(C_c v + D_c y) + Ew \quad (2.36)$$

$$y = C_1x + 0u + D_1w \quad (2.37)$$

$$z = C_2x + D_2(C_c v + D_c y) \quad (2.38)$$

Sehingga jika persamaan (2.37) dimasukkan ke persamaan (2.36) dan (2.38) persamaan menjadi sebagai berikut:

$$\dot{x} = Ax + BC_c v + BD_c(C_1 x + D_1 w) + Ew \quad (2.39)$$

$$z = C_2 x + D_2 C_c v + D_2 D_c(C_1 x + D_1 w) \quad (2.40)$$

Sehingga dapat dikelompokkan menjadi sebagai berikut:

$$\dot{x} = (A + BD_c C_1)x + BC_c v + (E + BD_c D_1)w \quad (2.41)$$

$$z = (C_2 + D_2 D_c C_1)x + D_2 C_c v + D_2 D_c D_1 w \quad (2.42)$$

Kemudian jika disusun dalam persamaan bentuk matriks sebagai berikut:

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = \begin{bmatrix} A + BD_c C_1 & BC_c \\ B_c C_1 & A_c \end{bmatrix} \begin{pmatrix} x \\ v \end{pmatrix} + \begin{bmatrix} E + BD_c D_1 \\ B_c D_1 \end{bmatrix} w \quad (2.43)$$

$$z = [C_2 + D_2 D_c C_1 \quad D_2 C_c] \begin{pmatrix} x \\ v \end{pmatrix} + [D_2 D_c D_1] w \quad (2.44)$$

Dimana persamaan (2.43) dan (2.44) dapat ditulis sebagai berikut,

$$\begin{pmatrix} \dot{x} \\ \dot{v} \end{pmatrix} = A_{cl} \tilde{x} + B_{cl} w \quad (2.45)$$

$$z = C_{cl} \tilde{x} + D_{cl} w \quad (2.46)$$

Sehingga fungsi alih closed loop w-z dituliskan sebagai berikut:

$$T_{zw}(s) = C_{cl}(sI - A_{cl})^{-1} B_{cl} + D_{cl} \quad (2.47)$$

Dari sistem secara closed loop akan stabil secara internal jika dan hanya jika nilai eigen dari $A_{cl} = \begin{bmatrix} A + BD_c C_1 & BC_c \\ B_c C_1 & A_c \end{bmatrix}$ terletak di sebelah kiri bidang kompleks.

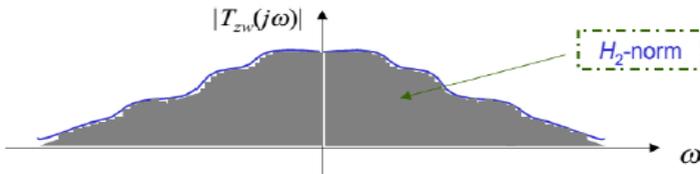
Kontrol optimal terdiri dari kontrol optimal H_2 dan kontrol optimal H_∞ yang biasa disebut dengan norm H_2 dan norm H_∞ .

- *Norm H_2*

Norm H_2 adalah semua energi yang berkaitan dengan tanggapan impuls dari $T_{zw}(s)$, sehingga meminimalkan norm H_2 suatu fungsi alih $T_{zw}(s)$ yang berarti meminimalkan energi dari gangguan w menuju output terkontrol z . Fungsi alih $T_{zw}(s)$ yang stabil dan sesuai, maka norm H_2 dapat dituliskan sebagai berikut:

$$\|T_{zw}\|_2 = \left(\left(\frac{1}{2\pi} \right) \text{trace} \left[\int_{-\infty}^{\infty} T_{zw}(j\omega) T_{zw}^*(j\omega) d\omega \right] \right)^{\frac{1}{2}} \quad (2.48)$$

Dari persamaan diatas dapat diilustrasikan sebagai berikut :



Gambar 2.11 Ilustrasi grafik luasan $H_2\text{-norm}$
(Susanto & Wibawa, 2015)

Infimum norm H_2 suatu matriks alih closed loop $T_{zw}(s)$ yang secara keseluruhan menstabilkan kontrol yang tepat, dituliskan sebagai berikut:

$$\gamma_2: \inf\{\|T_{zw}\|_2 \mid \Sigma \text{ internally stabilizes } \Sigma\} \quad (2.49)$$

Kontrol Σ_c dikatakan sebagai performa H_2 jika secara internal menstabilkan Σ dan $\|T_{zw}\|_2 = \gamma_2$.

- *Norm H_∞*

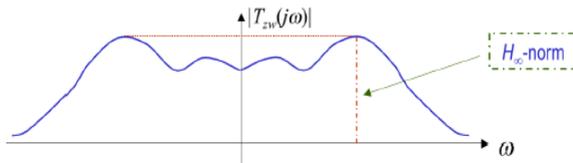
Norm H_∞ adalah energi penguatan untuk keadaan terburuk dalam $T_{zw}(s)$, sehingga meminimalkan norm H_∞ suatu fungsi alih $T_{zw}(s)$ berarti meminimalkan situasi (penguatan) terburuk, dari gangguan w menuju output terkontrol z . Fungsi alih $T_{zw}(s)$

yang stabil dan sesuai, maka $norm H_\infty$ dapat dituliskan sebagai berikut:

$$\|T_{zw}\|_\infty = \sup \sigma \max[T_{zw}(j\omega)] \quad (2.50)$$

$$\|T_{zw}\|_\infty = \sigma_{maks} \quad (2.51)$$

Dari persamaan diatas dapat diilustrasikan sebagai berikut



Gambar 2.12 Ilustrasi grafik luasan H_∞ norm (Susanto & Wibawa, 2015)

Infimum norm H_∞ suatu matriks alih closed loop $T_{zw}(s)$ yang secara keseluruhan menstabilkan kontrol yang tepat, dapat dituliskan sebagai berikut:

$$\gamma_2: \inf\{\|T_{zw}(s)\|_\infty \mid \Sigma_c \text{ internally stabilizes } \Sigma\} \quad (2.52)$$

Kontrol Σ_c dikatakan sebagai performa H_∞ jika secara internal menstabilkan Σ dan $\|T_{zw}\|_\infty < \gamma$ ($> \gamma$).

2.5 Sistem *Passive Fault Tolerant Control*

Pada sistem pengendalian dibutuhkan suatu peningkatan performa dan keamanan (*safety*) yang cukup tinggi. Desain sistem pengendalian umpan balik (*feedback control design*) yang sering digunakan pada sistem yang kompleks dapat mengakibatkan kinerja yang kurang stabil apabila terjadi beberapa kerusakan pada aktuator, sensor, ataupun komponen penyusun sistem tersebut. Untuk mengatasi kerusakan tersebut, dilakukan desain pengendalian yang mampu menolerir kerusakan yang terjadi sehingga stabilitas sistem tetap terjaga sesuai yang diinginkan. Sistem *Fault Tolerant Control* adalah suatu sistem pengendali yang

mampu mentoleransi kesalahan dalam suatu sistem untuk meningkatkan kinerja yang diinginkan (Zhang & Jiang, 2003).

FTCS dapat diklasifikasikan pada 2 tipe, yaitu *Passive Fault Tolerant Kontrol Sistem* (PFTCS) dan *Active Fault Tolerant Kontrol Sistem* (AFTCS). PFTCS dirancang untuk mempertahankan sistem dari kegagalan yang terjadi. Sedangkan AFTCS bereaksi terhadap kegagalan komponen dengan merekonfigurasi aksi kontrol sehingga kestabilan dan kinerja pada sistem dapat dijaga. Perbedaan antara sistem PFTC dan AFTC adalah terletak pada perancangan *Reconfigurable Control* (RC) dan FDD dimana perancangan tersebut diperlukan dalam sistem AFTC sedangkan sistem PFTC tidak memerlukan perancangan tersebut.

Pada *Passive Fault Tolerant Control* (PFTC) parameter kontroler dibuat tetap dan didesain menggunakan algoritma kontrol *robust* untuk memastikan sistem kontrol tetap mampu mengatasi kesalahan dari komponen sistem. Kelebihan sistem PFTC yaitu karena besarnya kesalahan sudah dapat ditentukan sebelumnya sehingga sistem dapat didesain menjadi sistem pengendali yang *robust* terhadap kesalahan yang terjadi ada suatu komponen dan tidak terjadi penundaan waktu karena tidak memerlukan prosedur RC dan FDD seperti pada sistem AFTC. Namun konsekuensinya, jika besaran kesalahan diluar nilai yang ditentukan maka kinerja sistem ini akan memburuk.

2.6 Observer State

Observer adalah suatu algoritma yang digunakan untuk mengestimasi keadaan (*state*) dari variabel yang tidak terukur berdasarkan model matematis sistem, sehingga observer juga dapat digunakan untuk mengestimasi kesalahan yang terjadi pada suatu sistem jika kesalahan tersebut masuk dalam pemodelan. Suatu sistem umumnya dimodelkan dalam bentuk *state space*, yang dapat didefinisikan secara umum seperti persamaan sistem dibawah ini (Indriawati, 2015):

$$\dot{x} = Ax + Bu \quad (2.53)$$

$$y = Cx + Du \quad (2.54)$$

dengan

A	= matriks keadaan (<i>state</i>)
B	= matriks kendali
C	= matriks keluaran
D	= matriks gangguan
x	= vektor keadaan (<i>state</i>)
u	= sinyal kontrol
y	= vektor keluaran

Persamaan ruang keadaan yang mengandung kesalahan aktuator dapat dituliskan sebagai berikut

$$\dot{x} = Ax + Bu + F_a f_a \quad (2.55)$$

$$y = Cu + Du \quad (2.56)$$

Selanjutnya *state* baru, ditambahkan untuk memindahkan kesalahan aktuator sebagai berikut,

$$\dot{x}_z(t) = A_z(y - x_z) \quad (2.57)$$

dari persamaan (2.55) dapat disubstitusikan ke dalam persamaan (2.57) sehingga diperoleh persamaan:

$$\dot{x}_z = A_z(Cx + Du - x_z) \quad (2.58)$$

Untuk A_z merupakan matriks identitas (I) untuk $A_z \in I^q$, sedangkan matriks dari kesalahan aktuator didefinisikan sebagai $f = [f_a]$.

Dari persamaan (2.55) dapat didefinisikan matriks yang akan digunakan untuk mendapatkan matriks yang sudah diekspansi yang ditunjukkan pada persamaan berikut

$$\dot{x}_a = A_a x_a + B_a u + E_a f \quad (2.59)$$

$$y_a = C_a x_a \quad (2.60)$$

Sehingga persamaan (2.58) dan (2.59) dapat diubah ke dalam bentuk persamaan ruang keadaan menjadi,

$$\begin{bmatrix} \dot{x} \\ \dot{x}_z \end{bmatrix} = \begin{bmatrix} A & 0 \\ AzC & -Az \end{bmatrix} \begin{bmatrix} x \\ x_z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} F_a \\ 0 \end{bmatrix} [f_a] \quad (2.61)$$

$$y = \dot{x}_z = \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} x \\ x_z \end{bmatrix} \quad (2.62)$$

Dengan

$$x_a = \begin{bmatrix} x \\ x_z \end{bmatrix}, A_a = \begin{bmatrix} A & 0 \\ AzC & -Az \end{bmatrix}, B_a = \begin{bmatrix} B \\ 0 \end{bmatrix}, C_a = \begin{bmatrix} 0 & I \end{bmatrix}$$

$$E_a = \begin{bmatrix} F_a \\ 0 \end{bmatrix}$$

Untuk memperoleh nilai dari *gain observer* maka diperlukan ekspansi matriks melalui persamaan ruang keadaan baru yang ditunjukkan pada persamaan (2.63),

$$\dot{\hat{x}}_a = A_a \hat{x} + B_a u + E_a \begin{bmatrix} \hat{f}_a \end{bmatrix} + L_x (y - \hat{y}) \quad (2.63)$$

$$\begin{bmatrix} \hat{f}_a \end{bmatrix} = L_f (y - \hat{y}) \quad (2.64)$$

$$\hat{y} = C_a \hat{x} \quad (2.65)$$

Apabila kedua persamaan (2.63) dan (2.64) dinyatakan dalam bentuk persamaan ruang keadaan sebagai berikut,

$$\begin{bmatrix} \dot{\hat{x}}_a \\ \dot{\hat{f}}_a \end{bmatrix} = \begin{bmatrix} A_a & E_a \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_a \\ \hat{f}_a \end{bmatrix} + \begin{bmatrix} B_a \\ 0 \end{bmatrix} u + \begin{bmatrix} L_x \\ L_f \end{bmatrix} [C_a \quad 0](y - \hat{y}) \quad (2.66)$$

$$\hat{y} = [C_a \quad 0] \begin{bmatrix} \hat{x} \\ \hat{f} \end{bmatrix} \quad (2.67)$$

2.7 Optimisasi dengan LMI

LMI (*Linier Matrikx Inequalities*) merupakan perhitungan numerik menggunakan metode optimasi konveks untuk menyelesaikan petidaksamaan matriks. Untuk kasus penyelesaian persamaan dengan LMI dapat menggunakan penurunan persamaan fungsi Lyapunov. Dengan syarat fungsi Lyapunov $P > 0$ dan $A^T P + PA < 0$ sehingga persamaan LMI dapat dituliskan sebagai berikut,

$$\begin{bmatrix} -P \\ A^T P + PA \end{bmatrix} < 0 \quad (2.68)$$

$$\begin{bmatrix} -P & 0 \\ 0 & A^T P + PA \end{bmatrix} < 0 \quad (2.69)$$

Contoh kasus LMI

$$\dot{x} = Ax + Bu \quad (2.70)$$

dengan

$$u = Kx$$

Sehingga persamaan (2.70) dapat dituliskan sebagai berikut:

$$\dot{x} = Ax + BKx \quad (2.71)$$

$$\dot{x} = (A + BK)x \quad (2.72)$$

$$\dot{x} = A_C x \quad (2.73)$$

Dengan menggunakan syarat fungsi Lyapunov $P > 0$ dan $A_C^T P + PA_C < 0$ sehingga penyelesaian LMI pada persamaan (2.70) dapat dituliskan sebagai berikut,

$$(A + BK)^T P + P(A + BK) \quad (2.74)$$

$$A^T P + (BK)^T P + PA + PBK \quad (2.75)$$

$$A^T P + K^T B^T P + PA + PBK \quad (2.76)$$

$$A^T P + PA + K^T B^T P + PBK \quad (2.77)$$

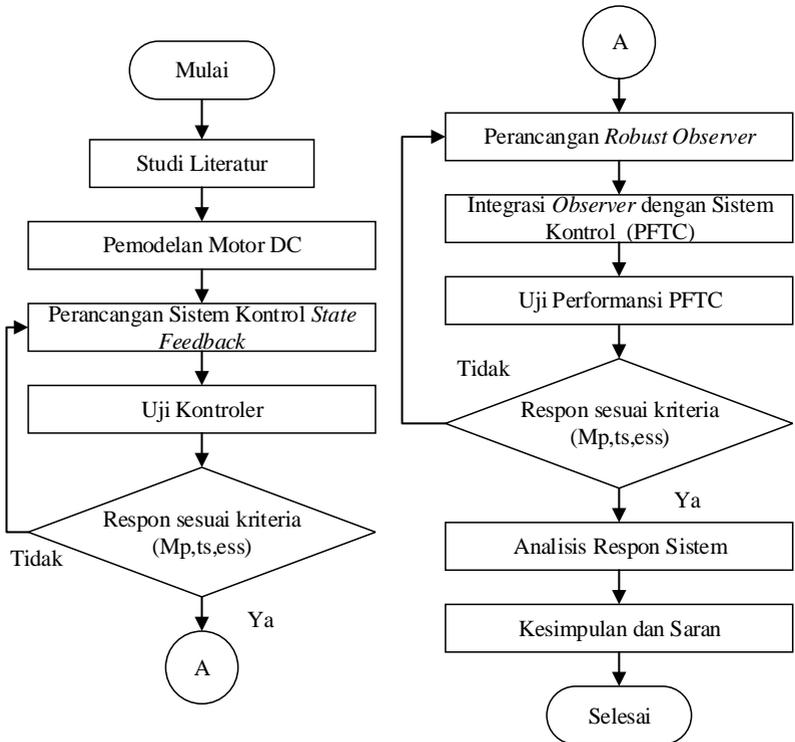
Dari persamaan (2.78) dapat dituliskan ke dalam bentuk persamaan ruang keadaan sebagai berikut,

$$\begin{bmatrix} -P & 0 \\ 0 & A_C^T P + PA_C \end{bmatrix} < 0 \quad (2.79)$$

$$\begin{bmatrix} -P & 0 \\ 0 & A^T P + PA + K^T B^T P + PBK \end{bmatrix} < 0 \quad (2.80)$$

BAB III METODOLOGI PENELITIAN

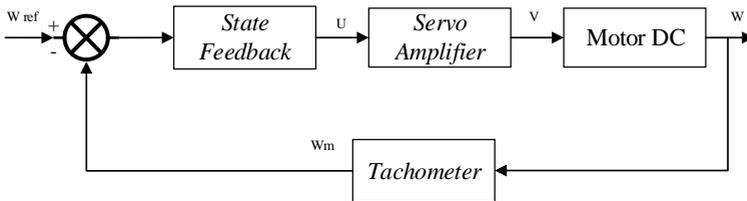
Pada bab ini berisi tahapan yang dilakukan dalam penelitian. Tahap penelitian meliputi studi literatur mengenai teori-teori yang dibutuhkan pada penelitian, pemodelan motor DC, perancangan sistem kontrol *state feedback*, perancangan *robust observer*, integrasi *observer* dengan sistem kontrol (PFTC) dan uji performansi pada PFTC. Tahapan yang dilakukan dalam Tugas Akhir ini ditampilkan dengan sebuah diagram alir (*flow chart*) pada Gambar 3.1.



Gambar 3.1 Diagram alir penelitian tugas akhir

3.1 Pemodelan Motor DC

Pada sistem servo motor DC terdiri atas beberapa komponen yaitu *plant* motor DC, sensor yang digunakan berupa *tachometer* dan aktuator yang digunakan berupa *servoamplifier*, secara sistem *closed loop* dapat ditunjukkan dalam diagram blok pada Gambar 3.2.



Gambar 3.2 Diagram blok sistem pengendalian

- Motor DC

Dari persamaan (2.13) diperoleh model matematik dari motor DC dengan nilai parameter motor DC didapatkan dari jurnal, dimana nilai parameter motor yang digunakan (Indriawati, 2015) :

$$J = 0.01$$

$$b = 0.1$$

$$K = 0.01$$

$$R = 1$$

$$L = 0.5$$

dengan

J = Momen inersia ekuivalen dari motor, (Kg.m^2)

b = Koefisien gesekan viskos ekuivalen dari motor dan beban pada poros motor (N.m/rad.dt)

K = Konstanta torsi motor

R = Tahanan kumparan jangkar (ohm)

L = Induktansi kumparan jangkar (henry)

Dari nilai parameter yang didapatkan dari jurnal dapat dicari fungsi transfer motor DC dengan menggunakan persamaan (2.13) sebagai berikut:

$$\frac{\omega(s)}{V(s)} = \frac{K}{RJs + Rb + K.Kb} \quad (3.1)$$

$$\frac{\omega(s)}{V(s)} = \frac{0,01}{1s + 1(0,1) + (0,01)(0,01)} \quad (3.2)$$

$$\frac{\omega(s)}{V(s)} = \frac{0,01}{s + 0,1 + (0,0001)} \quad (3.3)$$

Dari *plant* motor DC didapatkan nilai *time constant* sebesar 0,1 sekon. Dari fungsi transfer yang telah diperoleh dapat disusun dalam bentuk persamaan ruang keadaan untuk motor DC sesuai dengan persamaan (2.12) dan (2.13) sebagai berikut,

$$\begin{bmatrix} \dot{\omega} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} -10 & 1 \\ -0.02 & -2 \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix} V \quad (3.4)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} \quad (3.5)$$

- *Sensor Tachometer*

Sensor yang digunakan adalah tachometer dengan tipe BD2510B. Dari data spesifikasi *tachometer* tipe BD2510B. Dari data spesifikasi *tachometer* tipe BD2510B didapatkan nilai *time constant* sebesar $5,45 \times 10^{-5} s$. Nilai *time constant* sensor jauh lebih cepat daripada *time constant plant* maka *time constant* sensor dapat diabaikan. Sehingga nilai kecepatan yang diukur sama dengan kecepatan *plant*, memiliki dinamika yang sama dengan kecepatan *plant* atau $\omega = \omega_m$.

- *Servo Amplifier*

Aktuator yang digunakan untuk mengendalikan kecepatan motor DC adalah *servo amplifier*. *Servo amplifier* merupakan rangkaian elektronik yang memiliki *time constant* yang lebih cepat dibandingkan dengan *time constant plant* sehingga *time constant servo amplifier* dapat diabaikan (H. Kohlmann, 2015). Sehingga nilai sinyal kontrol yang dihasilkan sama dengan nilai tegangan yang dihasilkan oleh *servo amplifier* atau $U = V$.

$$U = aU_0 \quad (3.6)$$

dengan

U_0 = Sinyal kontrol tanpa kesalahan

U = Sinyal kontrol

Jika $a = 1$, tidak ada kesalahan maka $U = 0$. Jika $0 < a < 1$, ada kesalahan maka $U = aU_0$

3.2 Perancangan Sistem Kontrol *State Feedback*

Pada tahap perancangan sistem kontrol *state feedback* dengan *integrator* menggunakan persamaan *state space* yaitu,

$$\dot{x} = Ax + Bu \quad (3.7)$$

$$y = Cx + Du \quad (3.8)$$

Dengan $x = \begin{bmatrix} \omega \\ i \end{bmatrix}$ adalah kecepatan (ω) dan arus (i), sedangkan *output* yang diinginkan mengikuti *setpoint* adalah kecepatan sehingga,

$$y_r = C_r \cdot x = [1 \quad 0] \begin{bmatrix} \omega \\ i \end{bmatrix} = \omega \quad (3.9)$$

Maka didapatkan nilai matriks ruang keadaan untuk perancangan *state feedback* dengan *integrator* dari pemodelan motor DC sebagai berikut:

$$A = \begin{bmatrix} -10 & 1 \\ -0.02 & -2 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

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

$$D = 0$$

Dari persamaan (3.8) didapatkan persamaan *error* yaitu,

$$\dot{e} = \omega_{ref} - \omega \quad (3.10)$$

$$\dot{e} = \omega_{ref} - C_r \cdot x \quad (3.11)$$

$$\dot{e} = -\omega_{ref} + C_r \cdot x \quad (3.12)$$

Dengan menggabungkan persamaan *state space* (3.4) dan (3.12) maka dapat diperoleh persamaan ruang keadaan baru sebagai berikut,

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A & 0 \\ C_r & 0 \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \omega_{ref} \quad (3.13)$$

$$\dot{x}_c = A_c x_c + B_c u + D_c \omega_{ref} \quad (3.14)$$

$$\dot{x}_c = (A_c + B_c K_c) x_c + D_c \omega_{ref} \quad (3.15)$$

Dari persamaan \dot{x}_c maka didapatkan nilai matriks sebagai berikut,

$$A_c = \begin{bmatrix} -10 & 1 & 0 \\ -0.02 & -2 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$B_c = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix}$$

$$D_c = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Sehingga agar respon *maximum overshoot*, *settling time*, dan *error steady state* dapat mencapai nilai *steady* maka diperoleh *pole* sebagai berikut,

$$\text{Pole ke-1} = -15 + 15i$$

$$\text{Pole ke-2} = -15 - 5i$$

$$\text{Pole ke-3} = -20$$

Dengan menggunakan metode *pole placement* sehingga diperoleh nilai *gain* pengendali *state feedback* sebagai berikut,

$$\begin{aligned} K_c &= [K_p \quad K_i] \\ K_c &= [224,99 \quad 19 \quad 2,5] \end{aligned} \quad (3.16)$$

Sehingga nilai :

$$K_p = [224,99 \quad 19]$$

$$K_i = [2,5]$$

3.3 Perancangan *Robust Observer*

Perancangan *observer* digunakan untuk mengestimasi kesalahan yang ada, dengan cara mengembangkan persamaan ruang keadaan sistem. Persamaan ruang keadaan yang mengandung kesalahan aktuator dapat dituliskan sebagai berikut,

$$\dot{x} = Ax + Bu + F_a f_a \quad (3.17)$$

$$y = Cu + Du \quad (3.18)$$

Tipe kesalahan aktuator dimodelkan sebagai

$$s_f(t) = \alpha s(t) + \beta \quad (3.19)$$

$$f(t) = s_f(t) - s(t) \quad (3.20)$$

Dengan $s_f(t)$ adalah sinyal kesalahan dan $s(t)$ adalah sinyal actual, α dan β adalah konstanta.

Selanjutnya *state* baru, ditambahkan untuk memindahkan kesalahan aktuator sebagai berikut,

$$\dot{x}_z(t) = A_z(y - x_z) \quad (3.21)$$

dari persamaan (3.17) dapat disubstitusi ke dalam persamaan (3.20) sehingga diperoleh persamaan:

$$\dot{x}_z = A_z(Cx + Du - x_z) \quad (3.22)$$

Untuk A_z merupakan matriks identitas (I) untuk $A_z \in I^q$, sedangkan matriks dari kesalahan aktuator didefinisikan sebagai $f = [f_a]$.

Dari persamaan (3.22) dapat didefinisikan matriks yang akan digunakan untuk mendapatkan matriks yang sudah diekspansi yang ditunjukkan pada persamaan berikut

$$\dot{x}_a = A_a x_a + B_a u + E_a f \quad (3.23)$$

$$y_a = C_a x_a \quad (3.24)$$

Sehingga persamaan (3.23) dan (3.24) dapat diubah ke dalam bentuk persamaan ruang keadaan menjadi,

$$\begin{bmatrix} \dot{x} \\ \dot{x}_z \end{bmatrix} = \begin{bmatrix} A & 0 \\ A_z C & -A_z \end{bmatrix} \begin{bmatrix} x \\ x_z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} F_a \\ 0 \end{bmatrix} [f_a] \quad (3.25)$$

$$y = \dot{x}_z = \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} x \\ x_z \end{bmatrix} \quad (3.26)$$

dengan

$$x_a = \begin{bmatrix} x \\ x_z \end{bmatrix}, \quad Aa = \begin{bmatrix} A & 0 \\ AzC & -Az \end{bmatrix}, \quad Ba = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad Ca = \begin{bmatrix} 0 & I \end{bmatrix}$$

$$Ea = \begin{bmatrix} F_a \\ 0 \end{bmatrix}$$

Untuk memperoleh struktur *observer* maka diperlukan ekspansi matriks melalui persamaan ruang keadaan baru yang ditunjukkan pada persamaan (3.26),

$$\dot{\hat{x}}_a = A_a \hat{x} + B_a u + E_a \begin{bmatrix} \hat{f}_a \end{bmatrix} + L_x (y - \hat{y}) \quad (3.27)$$

$$\begin{bmatrix} \hat{f}_a \end{bmatrix} = L_f (y - \hat{y}) \quad (3.28)$$

$$\begin{bmatrix} \hat{f}_a \end{bmatrix} = L_f e_L \quad (3.29)$$

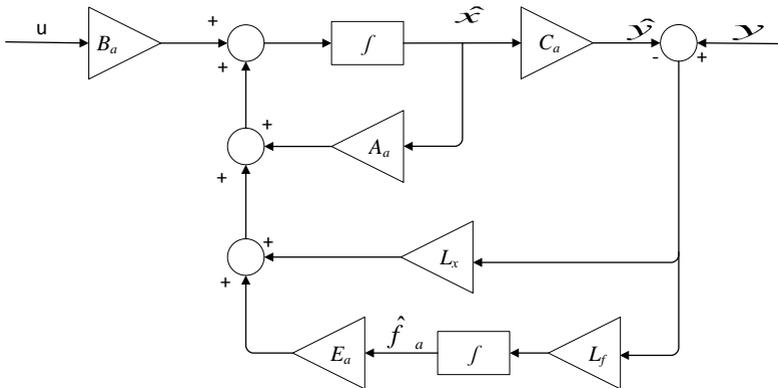
$$\hat{y} = C_a \hat{x} \quad (3.30)$$

Apabila kedua persamaan (3.27) dan (3.30) dinyatakan dalam bentuk persamaan ruang keadaan sebagai berikut,

$$\begin{bmatrix} \dot{\hat{x}}_a \\ \dot{\hat{f}}_a \end{bmatrix} = \begin{bmatrix} A_a & E_a \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_a \\ \hat{f}_a \end{bmatrix} + \begin{bmatrix} B_a \\ 0 \end{bmatrix} u + \begin{bmatrix} L_x \\ L_f \end{bmatrix} [C_a \quad 0] (y - \hat{y}) \quad (3.31)$$

$$\hat{y} = [C_a \quad 0] \begin{bmatrix} \hat{x}_a \\ \hat{f}_a \end{bmatrix} \quad (3.32)$$

Dengan \hat{x}_a , \hat{f}_a , \hat{y} berturut turut adalah hasil estimasi *augmented state*, kesalahan dan output. L_x adalah gain proporsional dari *observer* dan L_f adalah gain integral dari *observer*. Gain *observer* ini digunakan untuk menjamin kestabilan dinamik dari *error* estimasi.



Gambar 3.3 Struktur *extended state observer*

3.3.1 Penentuan *Generalized Plant*

Setelah menentukan struktur perancangan *observer* maka tahapan selanjutnya membuat *generalized plant* dengan persamaan sistem *generalized plant* sebagai berikut,

$$\dot{x} = Ax + B_0 w_0 + B_1 w_1 + B_2 u \quad (3.33)$$

$$z_0 = C_0 x + D_{01} w_1 \quad (3.34)$$

$$z_1 = C_1 x + D_{10} w_0 + D_{11} w_1 \quad (3.35)$$

Dengan w_0 adalah *set point*, w_1 *disturbance* dan z adalah *performansi*.

Pada perancangan sistem *generalized plant* digunakan metode dilatasi untuk menghasilkan persamaan ruang keadaan baru sebagai berikut,

$$\begin{bmatrix} I & O \\ O & O \end{bmatrix} \tilde{x} = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} \tilde{x} + \begin{bmatrix} \bar{B}_1 \\ \bar{B}_2 \end{bmatrix} u \quad (3.36)$$

$$y = [C \quad O] \tilde{x} + Du \quad (3.37)$$

Dengan $\tilde{E} =$ blok diagonal $\{I, O\}$, $\tilde{x} = [x^T \quad \xi^T]^T$ dan ξ merupakan *state* keadaan.

Jika dituliskan dalam bentuk persamaan *state space* sebagai berikut

$$\tilde{E}\dot{\tilde{x}} = \tilde{A}\tilde{x} + \tilde{B}\tilde{w} \quad (3.38)$$

$$\tilde{y} = \tilde{C}\tilde{x} + \tilde{D}\tilde{w} \quad (3.39)$$

Persamaan dalam bentuk matriks dapat dituliskan sebagai berikut:

$$\begin{bmatrix} \tilde{A} & \tilde{B} \\ \tilde{C} & \tilde{D} \end{bmatrix} = \begin{bmatrix} \tilde{A} & \tilde{B}_0 & \tilde{B}_1 \\ \tilde{C}_0 & \tilde{D}_{00} & \tilde{D}_{01} \\ \tilde{C}_1 & \tilde{D}_{10} & \tilde{D}_{11} \end{bmatrix} = \begin{bmatrix} A & B_L & B_0 & B_1 \\ O & -I & O & O \\ C_0 & O & O & D_{01} \\ C_1 & O & D_{10} & D_{11} \end{bmatrix} + \left\{ \begin{bmatrix} B_2 \\ O \\ O \\ O \end{bmatrix} \hat{K} + \begin{bmatrix} O \\ B_R \\ O \\ O \end{bmatrix} \right\} (K - \hat{K})x [C_2 \quad O \quad O \quad O] \quad (3.40)$$

Dari persamaan (3.40) dapat dijabarkan perhitungan sebagai berikut,

$$\begin{aligned} & \begin{bmatrix} A & B_L & B_0 & B_1 \\ O & -I & O & O \\ C_0 & O & O & D_{01} \\ C_1 & O & D_{10} & D_{11} \end{bmatrix} + \begin{bmatrix} B_2 \hat{K} C_2 & O & O & O \\ B_R (K - \hat{K}) C_2 & O & O & O \\ O & O & O & O \\ O & O & O & O \end{bmatrix} \\ & = \begin{bmatrix} A + B_2 \hat{K} C_2 & B_L & B_0 & B_1 \\ B_R (K - \hat{K}) C_2 & -I & O & O \\ C_0 & O & O & D_{01} \\ C_1 & O & D_{10} & D_{11} \end{bmatrix} \quad (3.41) \end{aligned}$$

dengan $\tilde{A} = \begin{bmatrix} A & B_L \\ O & -I \end{bmatrix}$, $B = \begin{bmatrix} \tilde{B}_0 & \tilde{B}_1 \end{bmatrix}$, $\tilde{C} = \begin{bmatrix} \tilde{C}_0 \\ \tilde{C}_1 \end{bmatrix} = \begin{bmatrix} C_0 & O \\ C_1 & O \end{bmatrix}$

$$\tilde{D} = \begin{bmatrix} \tilde{D}_{00} & \tilde{D}_{01} \\ \tilde{D}_{10} & \tilde{D}_{11} \end{bmatrix}, \quad B_2 = B_L B_R, \quad \xi = B_R U, \quad \tilde{x} = \begin{bmatrix} x \\ \xi \end{bmatrix},$$

$$\tilde{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}, \quad \text{dan } \tilde{y} = \begin{bmatrix} y_0 \\ y_1 \end{bmatrix}.$$

Setelah didapatkan persamaan sistem *generalized plant*, selanjutnya membuat persamaan *generalized* dari sistem servo motor DC. Pada penurunan persamaan kontroler didapatkan nilai output u_C dengan inputan y_{ref} dan \hat{x}

$$u_C = \begin{bmatrix} y_{ref} \\ \hat{x} \end{bmatrix} \quad (3.42)$$

Jika didefinisikan bentuk persamaan *state space* kontroler sebagai berikut,

$$\dot{x}_c = A_c x_c + B_c u_c \quad (3.43)$$

$$y_c = u \quad (3.44)$$

$$y_c = C_c x_c + D_c u_c \quad (3.45)$$

Dari persamaan (3.43) dan (3.45) dapat didefinisikan jika \dot{e}_c sama dengan *state* kontroler \dot{x}_c , dimana \dot{e}_c adalah error *setpoint* sehingga persamaan menjadi,

$$\dot{e}_c = \dot{x}_c = y_{ref} - C_r \hat{x} \quad (3.46)$$

$$y_c = -K_i e_c - K_p \hat{x} \quad (3.47)$$

Jika nilai $C_c = -K_i$, $D_c = [0 \quad -K_p]$, dan $u_c = \begin{bmatrix} y_{ref} \\ \hat{x} \end{bmatrix}$ sehingga persamaan (3.47) jika ditulis dalam bentuk persamaan (3.44) sebagai berikut,

$$y_c = C_c x_c + D_c u_c \quad (3.48)$$

$$y_c = -K_i e_c + [0 \quad K_p] \begin{bmatrix} y_{ref} \\ \hat{x} \end{bmatrix} \quad (3.49)$$

Setelah dilakukan perancangan tahap kontroler, maka selanjutnya dilakukan tahap perancangan aktuator dengan input u (sebagai output kontroler) dan output V_a ,

$$V_a = u.Fg_a \quad (3.50)$$

Dari persamaan (3.50) pada aktuator output V_a digunakan pada tahapan selanjutnya yaitu tahapan perancangan pada *plant* dengan input V_a dan output x sebagai berikut,

$$\dot{x} = Ax + BV_a \quad (3.51)$$

dimana nilai $V_a = u$ sehingga \dot{x} dapat dituliskan sebagai berikut,

$$\dot{x} = Ax + B(-K_i e_C - K_p \hat{x}) \quad (3.52)$$

Untuk output dari *plant* yaitu x dijadikan *input* pada tahapan perancangan selanjutnya yaitu *measurement* dengan *input* dari *plant* yaitu x dan output *measurement* Y_m .

Output measurement y_m sebagai *input* pada tahapan perancangan *observer*. Selain itu *input observer* berasal dari kontroler u . Sehingga mengacu pada persamaan struktur *observer* terdapat tiga *state* pada yaitu x_z , \hat{f} dan \hat{x} dengan persamaan sebagai berikut,

$$\dot{x}_z = A_z(Cx + Du - x_z) \quad (3.53)$$

$$\begin{bmatrix} \hat{f}_a \end{bmatrix} = L_f e_L \quad (3.54)$$

$$\dot{\hat{x}} = A\hat{x} + Bu + L_x e_L \quad (3.55)$$

Didefinisikan error residual dari *observer* dengan persamaan sebagai berikut,

$$e_{xz} = x_z - \dot{\hat{x}}_z \quad (3.56)$$

dengan

$$\dot{\hat{x}}_z = C_a \hat{x}_a \quad (3.57)$$

Output performansi adalah *output* yang dijadikan acuan dalam mewakili kinerja sistem. Output performansi terdiri dari *output*

error tracking dan *output error residual*. Sehingga *output* performansi e_z dapat dituliskan sebagai berikut,

$$e_z = y_r - C_r \cdot x \quad (3.58)$$

Sehingga

$$\dot{e}_z = \dot{y}_r - C_r \cdot \dot{x} \quad (3.59)$$

Pada *output* performansi terdapat *state* \dot{z} , dimana *state* \dot{z} mengandung fungsi pembobot (*weight function*) yang dapat dituliskan sebagai berikut

$$W(s) = \frac{z(s)}{e_z(s)} = \frac{A_1 s + A_2}{s + A_3} \quad (3.60)$$

Pada persamaan (3.60) jika dijabarkan menjadi,

$$\dot{z} + A_3 z = A_1 \dot{e}_z + A_2 e_z \quad (3.61)$$

Selanjutnya persamaan (3.59) disubstitusikan ke persamaan (3.61)

$$\dot{z} = -A_3 z + A_1 (\dot{y}_r - C_r \cdot \dot{x}) + A_2 e_z \quad (3.62)$$

$$\dot{z} = -A_3 z + A_1 \dot{y}_r - A_1 C_r \cdot \dot{x} + A_2 e_z \quad (3.63)$$

Persamaan ruang keadaan untuk *generalized plant* diperoleh enam *state* yaitu : $\hat{x}_z, \hat{f}, \hat{x}, x, e_c$, dan z sehingga dapat dituliskan dengan persamaan sebagai berikut,

$$\begin{aligned}
\begin{bmatrix} \dot{\hat{x}} \\ \dot{\hat{x}}_z \\ \dot{\hat{f}}_a \\ \dot{x} \\ \dot{e}_c \\ \dot{z} \end{bmatrix} &= \begin{bmatrix} 0 & 0 & -BK_p + A_a & 0 & -BK_i & 0 \\ -I & 0 & 0 & A_z C & 0 & 0 \\ 0 & 0 & F_a & 0 & 0 & 0 \\ 0 & 0 & -B.K_p & A & -BK_i & 0 \\ 0 & 0 & -C_r & 0 & 0 & 0 \\ 0 & 0 & -A_1 C_r & 0 & A_2 & -A_3 \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{x}_z \\ \hat{f}_a \\ x \\ e_c \\ z \end{bmatrix} \\
+ \begin{bmatrix} 0 & 0 \\ 0 & L_{xx} \\ 0 & L_{xxz} \\ 0 & 0 \\ I & 0 \\ A_1 & 0 \end{bmatrix} \begin{bmatrix} y_r \\ eL \end{bmatrix} & \quad (3.64)
\end{aligned}$$

3.3.2 Penentuan Gain Observer

Setelah memperoleh persamaan *generalized plant* maka dilakukan penentuan *gain observer* dengan formula LMI. LMI digunakan untuk meminimumkan γ dengan syarat sebagai berikut,

$$\gamma_2^2 \geq \text{trace}Q \quad (3.65)$$

$$\begin{bmatrix} H_e \{ \tilde{P}_2 \tilde{A} \} & \tilde{P}_2 \tilde{B}_0 \\ * & -I \end{bmatrix} < 0 \quad (3.66)$$

$$\begin{bmatrix} Q & C_0 + D_{02} K C_2 \\ * & P_2 \end{bmatrix} > 0 \quad (3.67)$$

$$\begin{bmatrix} H_e \{ \tilde{P}_\infty \tilde{A} \} & \tilde{P}_\infty \tilde{B}_1 & * \\ * & -\gamma_\infty I & * \\ \tilde{C}_1 & \tilde{D}_{11} & -\gamma_\infty I \end{bmatrix} < 0 \quad (3.68)$$

$$P_\infty > 0 \quad (3.69)$$

$$\tilde{P}_2 = \begin{bmatrix} P_2 & G_{21} \\ O & G_{22} \end{bmatrix} \quad (3.70)$$

$$\tilde{P}_2 = \begin{bmatrix} P_\infty & G_{\infty 1} \\ O & G_{\infty 2} \end{bmatrix} \quad (3.71)$$

Sehingga hasil dari meminimumkan γ didapatkan nilai *gain observer* secara iterasi yaitu

$$gg = 0.9362$$

3.4 Uji Performansi PFTC

Pada sistem PFTC terdapat penggabungan *gain kontroler* dengan *gain observer* yang disebut integrasi. Pada sistem PFTC besarnya kesalahan sudah ditentukan terlebih dahulu saat tahap perancangan sistem. Hal ini yang membedakan sistem PFTC dengan AFTC. Besaran ketetapan kesalahan disimbolkan dengan F_a . Selanjutnya dilakukan uji performansi berupa uji kesalahan *bias*, kesalahan *loss of effectiveness* dan uji ketidakpastian. Besaran kesalahan diberikan pada setiap uji performansi. Sehingga dapat dicari nilai F_a yang mampu menangani kesalahan dengan kesalahan terbesar dan menghasilkan respon yang baik.

BAB IV ANALISIS DATA DAN PEMBAHASAN

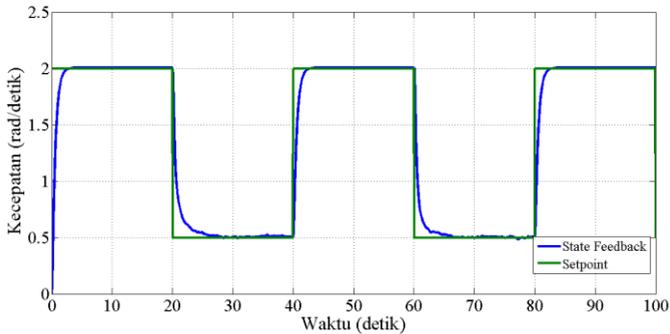
4.1 Uji *Tracking* Kontroler *State Feedback*

Untuk mengetahui performa kontroler *state feedback* maka dilakukan pengujian dengan cara uji *tracking* dimana terjadi perubahan *setpoint* yaitu 0,5 rad/detik dan 2 rad/detik. Dengan menggunakan metode *pole placement* maka diperoleh nilai *gain* kontroler sebagai berikut,

$$K_p = [224,99 \quad 19]$$

$$K_i = [2,5]$$

Grafik respon sistem kontrol *state feedback* dengan *integrator* pada motor DC secara simulasi ditunjukkan pada Gambar 4.1.

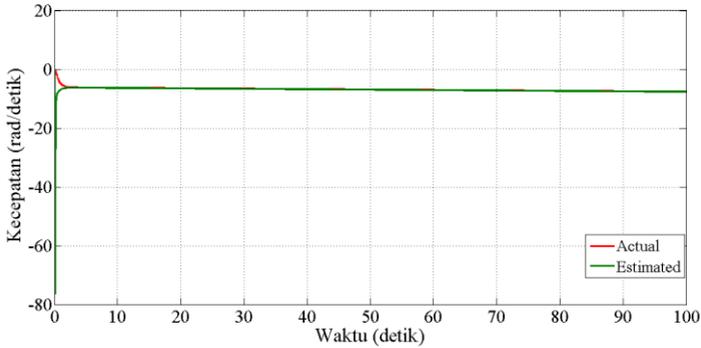


Gambar 4.1 Respon sistem perancangan sistem kontrol *state feedback* dengan *integrator* secara simulasi

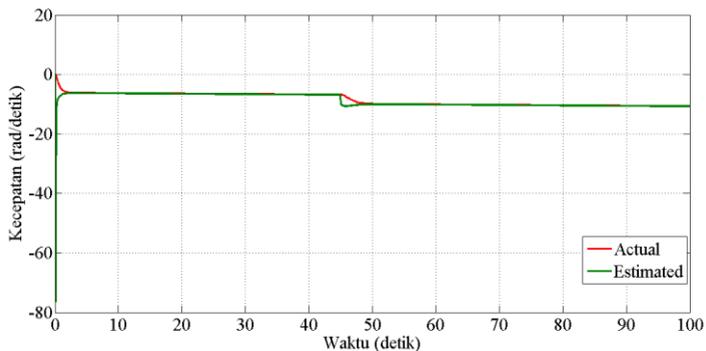
Gambar 4.1 menunjukkan dua grafik, yaitu grafik berwarna hijau yang menunjukkan nilai *setpoint* dan grafik berwarna biru yang menunjukkan hasil kendali *state feedback* dengan *integrator*, nilai *setpoint* sebesar 0,5 rad/detik dan 2 rad/detik diberikan pada waktu ke 0 detik, sehingga dari grafik tersebut dapat diketahui bahwa sistem dengan kontroler *state feedback* dengan *integrator* yang

dirancang dapat bekerja dengan baik karena mampu mencapai nilai *setpoint*. Dari respon grafik berwarna biru diperoleh nilai kriteria kontroler yaitu *settling time* sebesar 0,89 detik, tidak terdapat *maximum overshoot/undershoot* dan *error steady state*.

4.2 Hasil Estimasi *Observer*



Gambar 4.2 Hasil estimasi *observer* tanpa kesalahan



Gambar 4.3 Hasil estimasi *observer* dengan kesalahan

Pada tahap ini dilakukan simulasi untuk mengetahui unjuk kerja *observer*. Simulasi pertama dilakukan pada *plant* saat tidak terjadi kesalahan. Simulasi kedua dilakukan pada *plant* saat terjadi

kesalahan. Pada tahap ini digunakan nilai *gain observer* sebesar $gg = 0.9362$.

Hasil estimasi dari *observer* tanpa kesalahan ditunjukkan pada Gambar 4.2. Pada grafik tersebut respon dari *plant* dan hasil estimasi sudah memiliki hasil yang sama. Sedangkan untuk hasil estimasi dari *observer* dengan kesalahan *loss of effectiveness* sebesar 50% ditunjukkan pada Gambar 4.3 dengan memberikan kesalahan aktuator pada detik ke 45, respon sudah mampu mengestimasi kesalahan pada aktuator.

4.3 Uji Performansi PFTC

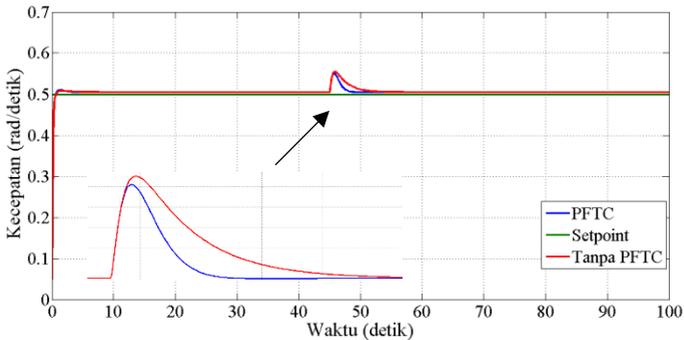
Pada tahap ini dilakukan uji performansi untuk mengetahui performansi *passive fault tolerant control* yang telah dirancang. Pengujian dilakukan dengan cara memberikan kesalahan pada aktuator berupa kesalahan *bias* dan kesalahan *loss of effectiveness*. Kesalahan bias merupakan kesalahan penyimpangan nilai dari nilai tetapan yang telah ditentukan. Kesalahan *bias* pada aktuator dapat berupa gangguan yang menghambat tegangan atau arus dari *servo amplifier* untuk masuk kedalam motor DC (Kresna, 2017). Kesalahan *loss of effectiveness* pada aktuator dapat berupa penurunan gain aktuator akibat penyumbatan atau valve berkarat. Dengan besaran ketetapan kesalahan (F_a) yang sudah ditentukan pada awal perancangan sebesar 0,5 (50%). Penggunaan nilai (F_a) sebesar 0,5 (50%) dikarenakan pada besaran ketetapan kesalahan tersebut menghasilkan respon closed loop yang baik, maka diberikan kesalahan *bias* pada aktuator sebesar 100% dan kesalahan *loss effectiveness* diberikan sebesar 95%.

4.3.1 Uji Kesalahan *Bias* Pada Aktuator

- Hasil respon sistem dengan *setpoint* 0,5 rad/detik dan kesalahan *bias* sebesar 100%.

Pada Gambar 4.4 respon sistem dengan PFTC yang ditunjukkan oleh grafik berwarna biru dan sistem tanpa PFTC yang ditunjukkan grafik berwarna merah, dengan nilai *setpoint* sebesar

0,5 rad/detik waktu ke 0 detik. Ketika kesalahan *bias* 100 % pada aktuator terjadi di detik ke 45, sistem mengalami *maximum overshoot* sebesar 10,98% untuk sistem tanpa PFTC dan 10,16% untuk sistem PFTC, sistem tanpa PFTC dan dengan PFTC tidak ada *error steady state*. Sedangkan untuk *settling time* sebesar 14,1 detik untuk sistem tanpa PFTC dan 4,4 detik untuk sistem PFTC.



Gambar 4.4 Respon sistem dengan *setpoint* 0,5 rad/detik dan kesalahan *bias* 100% secara simulasi

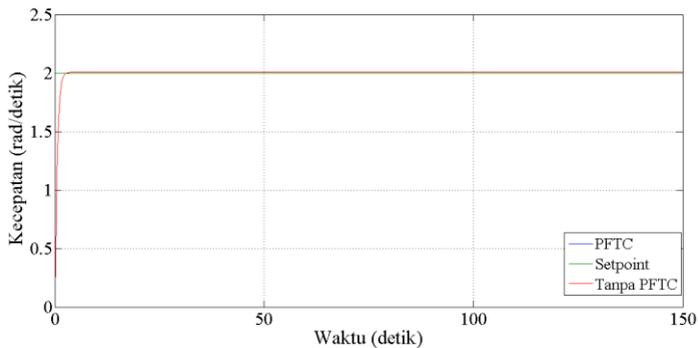
Tabel 4. 1 Performansi sistem dengan *setpoint* 0,5 rad/detik dan kesalahan *bias* 100% secara simulasi

Parameter	Kesalahan Aktuator	
	Dengan PFTC	Tanpa PFTC
<i>Maximum Overshoot</i>	10,16%	10,98%
<i>Settling Time</i>	4,4 detik	14,1 detik

- Hasil respon sistem dengan *setpoint* 2 rad/detik dan kesalahan *bias* sebesar 100%.

Pada Gambar 4.5 respon sistem dengan PFTC yang ditunjukkan oleh grafik berwarna biru dan sistem tanpa PFTC yang

ditunjukkan grafik berwarna merah, dengan nilai *setpoint* sebesar 2 rad/detik waktu ke 0 detik. Ketika kesalahan *bias* 100 % pada aktuator terjadi di detik ke 45, sistem tanpa PFTC dan dengan PFTC tidak mengalami *maximum overshoot/undershoot* dan *error steady state*. Sedangkan untuk *settling time* sebesar 0,001 detik untuk sistem tanpa PFTC dan 0,001 detik untuk sistem PFTC.

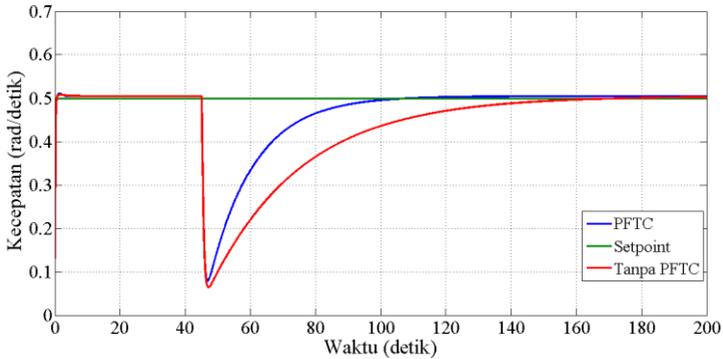


Gambar 4. 5 Respon sistem dengan *setpoint* 2 rad/detik dan kesalahan *bias* 100% secara simulasi

Secara keseluruhan hasil respon sistem dengan PFTC, ketika diberikan kesalahan *bias* sebesar 100% terjadi di waktu 45 detik, sistem dengan PFTC maupun tanpa PFTC tidak terdapat *error steady state*. Sistem dengan PFTC memiliki nilai *settling time* dan *maximum undershoot* yang lebih kecil dibandingkan sistem tanpa PFTC karena terdapat *observer* yang berfungsi untuk mengkompensasi atau memperbaiki kesalahan pada aktuator sehingga menyebabkan respon dari sistem dapat kembali ke nilai *setpoint*. Sehingga sistem PFTC mampu menangani kesalahan *bias* sebesar 100%.

4.3.2 Uji Kesalahan *Loss of Effectiveness* Pada Aktuator

- Hasil respon sistem dengan *setpoint* 0,5 rad/detik dan kesalahan *loss of effectiveness* sebesar 95%.



Gambar 4. 6 Respon sistem dengan *setpoint* 0,5 rad/detik dan kesalahan *loss of effectiveness* 95% secara simulasi

Tabel 4. 2 Performansi sistem dengan *setpoint* 0,5 rad/detik dan kesalahan *loss of effectiveness* 95% secara simulasi

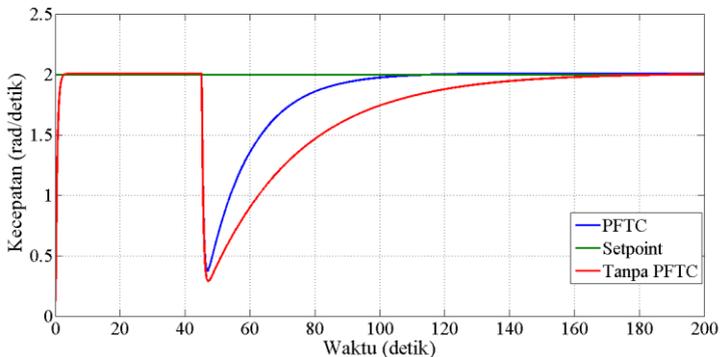
Parameter	Kesalahan Aktuator	
	Dengan PFTC	Tanpa PFTC
<i>Maximum Undershoot</i>	84,4%	87,1%
<i>Settling Time</i>	64,3 detik	130 detik

Pada Gambar 4.6 respon sistem dengan PFTC yang ditunjukkan oleh grafik berwarna biru dan sistem tanpa PFTC yang ditunjukkan grafik berwarna merah, dengan nilai *setpoint* sebesar 0,5 rad/detik waktu ke 0 detik. Ketika kesalahan *loss of effectiveness* 95% pada aktuator terjadi di 45 detik, sistem tanpa PFTC mengalami *maximum undershoot* sebesar 87,1% dan *settling time* sebesar 130 detik. Sedangkan sistem dengan PFTC menunjukkan nilai *maximum undershoot* sebesar 84,4% dan *settling time* sebesar 64,3 detik. Sistem dengan PFTC maupun tanpa PFTC tidak memiliki nilai *error steady state*. Sistem dengan

PFTC menunjukkan respon yang lebih baik karena nilai *maximum undershoot* yang kecil dan waktu yang diperlukan respon untuk kembali ke nilai *setpoint* lebih cepat dibandingkan dengan sistem tanpa PFTC.

- Hasil respon sistem dengan *setpoint* 2 rad/detik dan kesalahan *loss of effectiveness* sebesar 95%.

Pada Gambar 4.7 respon sistem dengan PFTC yang ditunjukkan oleh grafik berwarna biru dan sistem tanpa PFTC yang ditunjukkan grafik berwarna merah, dengan nilai *setpoint* sebesar 0,5 rad/detik pada waktu ke 0 detik. Ketika kesalahan *loss of effectiveness* 95% pada aktuator terjadi di 45 detik, sistem tanpa PFTC mengalami *maximum undershoot* sebesar 85,6% dan *settling time* sebesar 149,7 detik. Sedangkan sistem dengan PFTC menunjukkan nilai *maximum undershoot* sebesar 81,3% dan *settling time* sebesar 72 detik. Sistem dengan PFTC maupun tanpa PFTC tidak memiliki nilai *error steady state*. Sistem dengan PFTC menunjukkan respon yang lebih baik karena nilai *maximum undershoot* yang kecil dan waktu yang diperlukan respon untuk kembali ke nilai *setpoint* lebih cepat dibandingkan dengan sistem tanpa PFTC.



Gambar 4.7 Respon sistem dengan *setpoint* 2 rad/detik dan kesalahan *loss of effectiveness* 95% secara simulasi

Tabel 4. 3 Peformansi sistem dengan setpoint 2 rad/detik dan kesalahan loss of effectiveness 95% secara simulasi

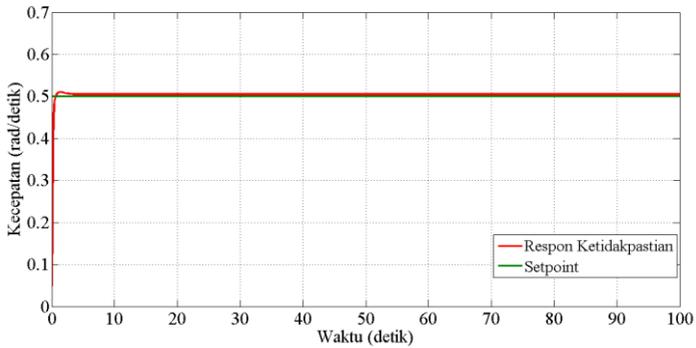
Parameter	Kesalahan Aktuator	
	Dengan PFTC	Tanpa PFTC
<i>Maximum Undershoot</i>	81,3%	85,6%
<i>Settling Time</i>	72 detik	149,7 detik

Secara keseluruhan hasil respon sistem dengan PFTC, ketika diberikan kesalahan *loss of effectiveness* sebesar 95% terjadi di waktu 45 detik, sistem dengan PFTC maupun tanpa PFTC tidak terdapat *error steady state*. Sistem dengan PFTC memiliki nilai *settling time* dan *maximum undershoot* yang lebih kecil dibandingkan sistem tanpa PFTC karena terdapat *observer* yang berfungsi untuk mengkompensasi atau memperbaiki kesalahan pada aktuator sehingga menyebabkan respon dari sistem dapat kembali ke nilai *setpoint*. Sehingga sistem PFTC mampu menangani kesalahan *loss of effectiveness* sebesar 95%.

4.3.3 Uji Ketidakpastian

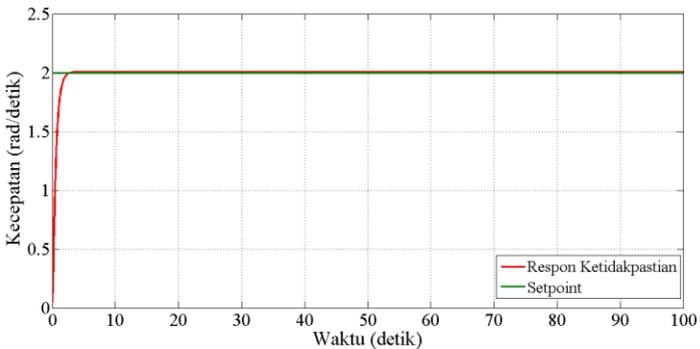
Uji ketidakpastian ini dilakukan untuk mengetahui sistem dengan PFTC mampu mengatasi besar ketidakpastian yang diberikan pada tahanan kumparan jangkar (R). Ketidakpastian pada motor DC dapat disebabkan karena perubahan tahanan jangkar (R) karena kenaikan suhu dan rugi-rugi. Rugi-rugi tersebut adalah: rugi-rugi tembaga, mekanis, dan arus, diberi ketidakpastian sebesar 10% dari nilai kumparan jangkar sebesar 1 ohm.

- Hasil respon sistem dengan *setpoint* 0,5 rad/detik dan ketidakpastian sebesar 10%.



Gambar 4. 8 Respon sistem dengan PFTC dan *setpoint* 0,5 rad/detik dan ketidakpastian 10% secara simulasi

- Hasil respon sistem dengan *setpoint* 2 rad/detik dan ketidakpastian sebesar 10%.



Gambar 4. 9 Respon sistem dengan PFTC dan *setpoint* 2 rad/detik dan ketidakpastian 10% secara simulasi

Pada Gambar 4.8 dan Gambar 4.9 respon sistem PFTC dengan ketidakpastian ditunjukkan oleh grafik berwarna merah dengan nilai *setpoint* 0,5 rad/detik dan 2 rad/detik diberikan pada waktu ke 0 detik, sehingga dari grafik tersebut dapat diketahui bahwa sistem PFTC dengan ketidakpastian yang diberikan sebesar 10% masih mampu bekerja dengan baik karena mampu mengikuti nilai *setpoint* yang diberikan, dengan karakteristik respon tidak terdapat nilai *maximum overshoot/undershoot* dan *error steady state*. Sedangkan *settling time* sebesar 0,13 detik untuk *setpoint* 0,5 rad/detik dan 0,922 detik untuk *setpoint* 2 rad/detik.

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis data yang telah dilakukan, didapatkan kesimpulan dari tugas akhir mengenai perancangan sistem *Passive Fault Tolerant Control* pada pengendalian kecepatan sistem servo motor DC dengan kesalahan pada aktuator adalah sebagai berikut:

- Telah dilakukan perancangan algoritma PFTC berbasis *observer*. Parameter ketetapan kesalahan yang digunakan sebesar 0,5. Penggunaan besaran ketetapan kesalahan 0,5 dikarenakan pada nilai tersebut menghasilkan respon closed loop yang baik.
- Sistem kontrol *state feedback* dengan *integrator* yang dirancang mampu tracking *setpoint* dengan nilai K_p sebesar 224,99 dan 19 sedangkan K_i sebesar 2,5.
- Sistem dengan PFTC lebih mampu menjaga performansi ketika terjadi kesalahan pada aktuator dibandingkan dengan sistem tanpa PFTC. Hal ini ditunjukkan oleh kriteria respon *maximum undershoot* dan *error settling time* yang lebih kecil dengan menggunakan sistem PFTC yaitu 10,16% dan 4,4 detik.
- Sistem dengan PFTC maupun tanpa PFTC tidak terdapat *error steady state* untuk kesalahan aktuator.
- Kesalahan *bias* terbesar yang masih mampu diakomodasi oleh sistem PFTC sebesar 100%. Sedangkan kesalahan *loss of effectiveness* terbesar yang masih mampu diakomodasi oleh sistem PFTC adalah sebesar 95%.
- Sistem PFTC mampu menangani ketidakpastian pada tahanan kumparan jangkar (R). Ketidakpastian yang masih mampu diakomodasi oleh sistem PFTC sebesar 10%.

5.2 Saran

Dari tugas akhir ini terdapat beberapa saran yang dapat digunakan untuk penelitian selanjutnya adalah:

- Mengimplementasikan sistem PFTC yang telah dibuat pada *plant riil*.
- Penelitian selanjutnya dapat menggunakan sistem PFTC dengan menggabungkan kesalahan aktuator dan sensor pada *plant* motor DC.

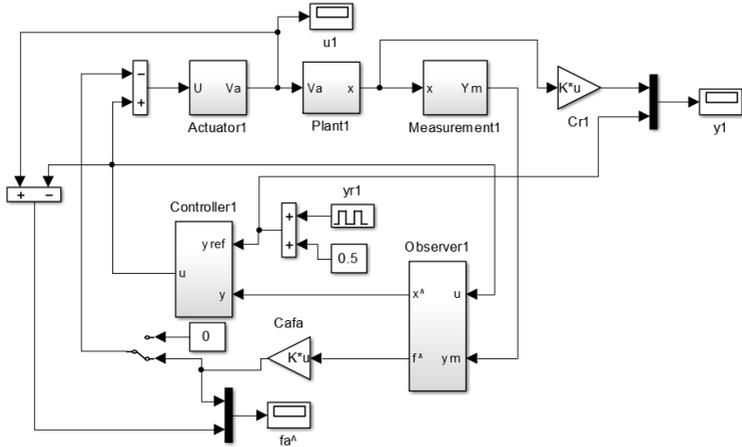
DAFTAR PUSTAKA

- Ardhiantama, A. (2016). Perancangan Active Fault Tolerant Control Pada Sistem Pengendalian Temperatur Fuel Gas Superheat Burner PT PETROKIMIA GRESIK Dengan Kesalahan Pada Sensor Temperatur. *Surabaya: Institut Teknologi Sepuluh Nopember*.
- Burn, R. S. (2001). *Advanced Control Engineering*. UK: Universty of Plymouth.
- G.Rigatos, G. (2007). *Adaptive fuzzy control of DC motors using state and output feedback*. IEEE.
- H. Kohlmann, W. S. (2015). *Compensation of Integrator Time Constants for Electric Field Measurement. Symposium on Lightning Protection (XIII SIPDA) : Balneario Camboriu, Brazil*.
- Hudaya, C. (2013). *Motor DC*. Jakarta: Universitas Indonesia.
- Indriawati, K. e. (2015). *Robust Observer-Based Fault Tolerant Tracking Control for Linear Systems with Simultaneous Actuator and Sensor Faults: Application to a DC Motor System. Praise Worthy Praise*, 8.
- Koksal, M. (2007). *Position Control of a Permanent Magnet DC motor by Model Reference Adaptive Control. IEEE*, 112-117.
- Kresna, V. (2017). Perancangan Sistem Active Fault Tolerant Control Pada Pengendalian Kecepatan Sistem Servo Modular MS150 DC Dengan Kesalahan Pada Aktuator dan Sensor. *Surabaya: Institut Teknologi Sepuluh Nopember*.
- Lee, J., & Lyou, J. (2001). *Fault Diagnosis and Fault Tolerant Control of DC Motor Driving System*.
- Noboru Sebe. (2012). *Fault-Tolerant Servo Systems against Actuator Failures. Preprints of the 7th IFAC Symposium on Robust Control Design*, 499-504.
- Ogata, K. (1997). *Modern Control Engineering 3rd ed*. London: Prentice Hall International.

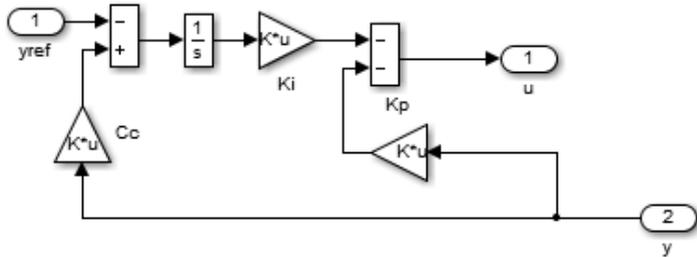
- Ogata, K. (2010). *Modern Control Engineering*. United States of America: Person Prentice Hall.
- Ogata, K. (2004). *System Dynamics*. New Jersey: Prentice Hall.
- Risfendra. (2009). Disain dan Implementasi Kontroler Kaskade Robust Pada Sistem Pressure Control Trainer Feedback 38-714.
- Sarah, A. (2015). Perancangan Active Fault Tolerant Control Pada Steam Drum Pada PLTU Paiton, JAWA TIMUR Dengan Kesalahan Pada Sensor Flow. *Surabaya: Institut Teknologi Sepuluh Nopember*.
- Sari, R. (2009). Penalaan Parameter Kontrol PID dengan Metode Heuristic , Aplikasi: Sistem Pengendalian Kecepatan Motor DC. *Institut Teknologi Sepuluh Nopember: Surabaya*.
- Scherer, C. (1997). *Multiobjective Output-Feedback Control via LMI Optimization*. *IEEE*, 896-910.
- Sebe, N. (2007). A New Dilated LMI Characterization And Iterative Control System Synthesis.
- Setyaningrum, D. (2012). Desain dan Implementasi Model Reference Adaptive Control untuk Pengaturan Tracking Optimasi Posisi Motor DC.
- Sugiono. (2015). Model Matematik Motor DC.
- Susanto, E., & Wibawa, P. (2015). Pengantar Kontrol Maju. *Telkom University*, 101-125.
- Waluyo, F. A. (2013). Analisis Penalaan Kontrol PID pada Simulasi Kendali Kecepatan Putaran Motor DC Berbeban Menggunakan Metode Heuristik. *Bandung: Institut Teknologi Itenas*.
- Zhang, Y & Jiang, J. (2008). *Bibliographical review on reconfigurable fault-tolerant control systems*. *Science Direct*, 229-252.
- Zhang, Y., & Jiang, J. (2003). *Fault Tolerant Control System Design With Explicit Consideration of Performance Degradation*. *IEEE*, 838-848.

LAMPIRAN A

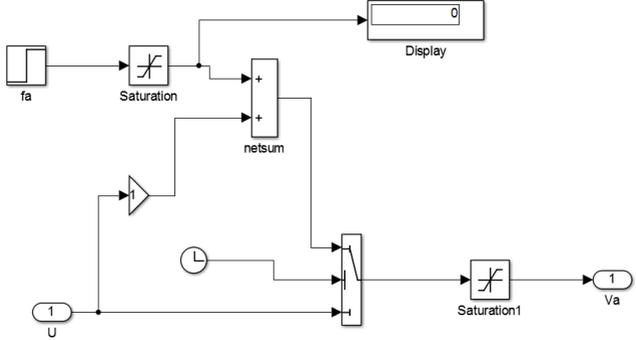
SIMULINK MATLAB R2013a KESELURUHAN SISTEM



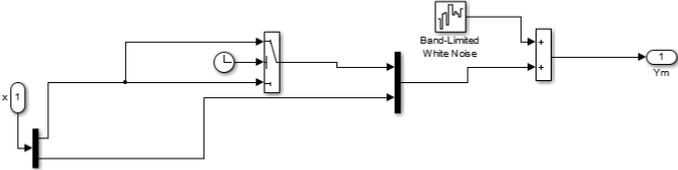
SIMULINK MATLAB R2013a KONTROLER



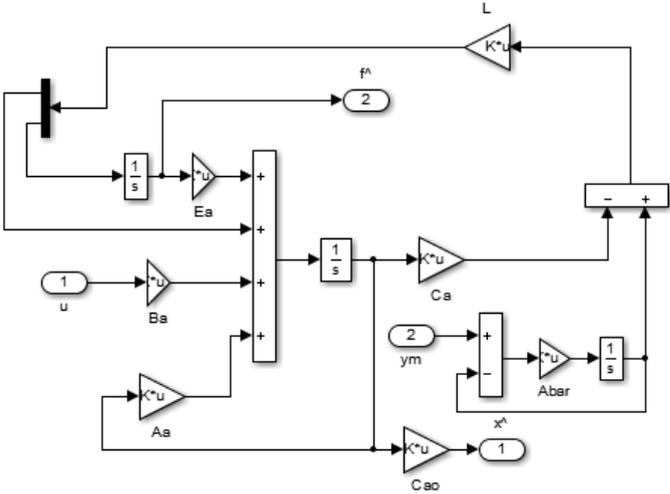
SIMULINK MATLAB R2013a FAULT AKTUATOR



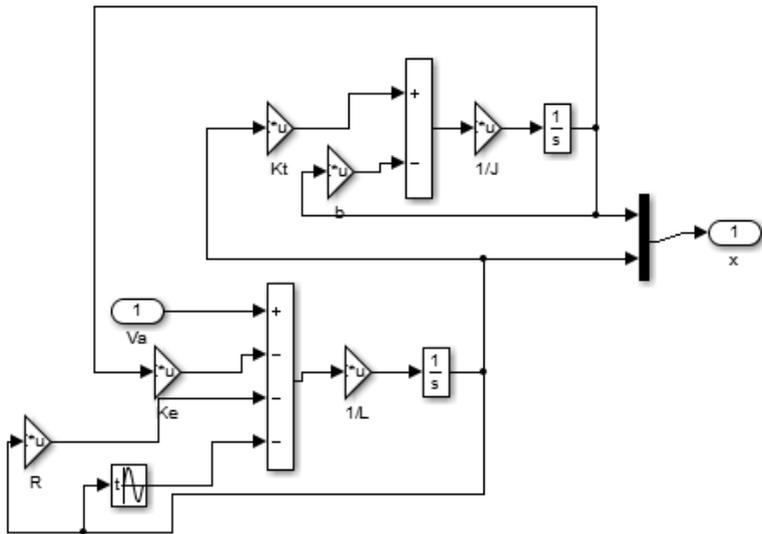
SIMULINK MATLAB R2013a SENSOR



SIMULINK MATLAB R2013a OBSERVER



SIMULINK MATLAB R2013a OBSERVER



LAMPIRAN B
LISTING PROGRAM MATLAB R2013a m.file SISTEM
MOTOR DC

```
J = 0.01;
b = 0.1;
Ke = 0.01;
R = 1;
delR = 0.1*1;
Lin = 0.5;

A = [-b/J   Ke/J
     -Ke/Lin -R/Lin];
B = [0
     1/Lin];
C = [1   0;0   1];
Cr = [1   0];
D = 0;

nx=size(A,1);
ny=size(C,1);
nu=size(B,2);
nr=1;          % number of reference inputs

% design PI controller
Ac = [A, zeros(nx,nr);C(1:nr,:) , zeros(nr) ];
Bc = [B;0];
p1 = -15 + 5i;
p2 = -15 - 5i;
p3 = -20;
Kc = place(Ac,Bc,[p1 p2 p3]);
Kp = Kc(1:2);
Ki = Kc(3);

Dd = [-1 0;0 -0.2];
Dw = [1; 0];
F = [1; 0];
E = B;

nw = size(Dw,2);
```

```

nd = size(Dd,2);
nfs = size(F,2);
nfa = size(E,2);

Aa = [A zeros(nx,ny); C -eye(ny)];
Ba = [B; zeros(ny,nu)];
Ea = [E zeros(nx,nfs); zeros(ny,nfa) F];
%Fa = [zeros(nx,nw); Dw];
%Fa = [Dd;zeros(ny,nd)];
Fa = [Dd zeros(nx,nw);zeros(ny,nd) Dw];
Ca = [zeros(ny,nx) eye(ny)];
Cao = [C zeros(ny)];
Cafa = [ones(1,nfa) zeros(1,nfs)];
%Bo = [zeros(n+p,fa+fs); eye(fa+fs)];
Bo = [Fa zeros(nx+ny,nfa+nfs);
zeros(nfa+nfs,nd+nw) eye(nfa+nfs)];

%% FTC against actuator and/or sensor failure
Aas=[Aa Ea; zeros(nfa+nfs,nx+ny)
zeros(nfa+nfs)];
Cas=[Ca zeros(ny,nfa+nfs)];

R4=eye(ny);
Q4=eye(nx+ny+nr*2)*1e2;
L4=lqr(Aas',Cas',Q4,R4);

L=L4';

Lqr=L4';

addpath inputs101

ws1=tf([0.72,0.3],[1,1e-2]);
ws2=tf([0.8,1],[1,1e-2]); % weight function for
tracking performance
nn=2; % number of generalized
plants
ps=struct('sys',cell(1,nn),'spec','hinf','Fopt',
1,'bound',0);

```

```

% generalized plant (normal case)
Fga = 1; Fgs = 1; Fga1 = 0; Fgr = 1;
[ta,tb,tc,td]=linmod2('clsys');
ps(1).sys=ss(ta,tb,tc,td);

% generalized plant (actuator failure case)
Fga = 0.5; Fgs = 1; Fga1 = 0; Fgr = 1;
[ta,tb,tc,td]=linmod2('clsys');
ps(2).sys=ss(ta,tb,tc,td);

lcmax=100; % maxum iteration
L_init=ss([],[],[],Lqr);
[gg,ga,Lopt]=multiobjective101(ps,L_init,[],100)
;

gg
Lopt

L=Lopt.d;

```

BIODATA PENULIS



Nama penulis Diyajeng Luluk Karlina dilahirkan di Tembagapura, tanggal 03 Nopember 1995. Saat ini penulis tinggal di Jalan Rawa Bakti No 5 Madiun, Provinsi Jawa Timur. Penulis telah menyelesaikan pendidikan di SDN 01 Klegen pada tahun 2008, pendidikan di SMPN 03 Madiun pada tahun 2011, pendidikan di SMAN 2 Madiun pada tahun 2014 dan sedang menempuh pendidikan S1 Teknik Fisika FTI di Institut Teknologi Sepuluh Nopember Surabaya hingga sekarang.

Pada bulan Januari 2018 penulis telah menyelesaikan Tugas Akhir dengan judul **Perancangan *Passive Fault Tolerant Control (PFTC)* Pada Pengendalian Kecepatan Sistem Motor DC Dengan Kesalahan Pada Aktuator**. Bagi pembaca yang memiliki kritik, saran atau ingin berdiskusi lebih lanjut mengenai tugas akhir ini, maka dapat menghubungi penulis melalui *email*: diyajengluluk@gmail.com.