

TUGAS AKHIR - TE 141599

KONTROL TOLERANSI KESALAHAN SENSOR DAN AKTUATOR BERBASIS PROPORTIONAL-PROPORTIONAL INTEGRAL OBSERVER UNTUK SISTEM PENDULUM KERETA

Ardiansyah NRP 2211 100 096

Dosen Pembimbing Prof. Dr. Ir. Mohammad Nuh, DEA. Dr. Trihastuti Agustinah, S.T., M.T.

JURUSAN TEKNIK ELEKTRO Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya 2015





FINAL PROJECT-TE 141599

FAULT TOLERANT CONTROL FOR SENSOR AND ACTUATOR BASED ON PROPORTIONAL INTEGRAL OBSERVER FOR INVERTED PENDULUM CART SYSTEM

Ardiansyah NRP 2211 100 096

Advisors Prof. Dr. Ir. Muhammad Nuh, DEA Dr. Trihastuti Agustinah, ST., MT.

ELECTRICAL ENGINEERING DEPARTMENT Faculty of Industrial Technology Sepuluh Nopember Instituteof Technology Surabaya 2015





FINAL PROJECT-TE 141599

FAULT TOLERANT CONTROL FOR SENSOR AND ACTUATOR BASED ON PROPORTIONAL INTEGRAL OBSERVER FOR INVERTED PENDULUM CART SYSTEM

Ardiansyah NRP 2211 100 096

Advisors Prof. Dr. Ir. Muhammad Nuh, DEA Dr. Trihastuti Agustinah, ST., MT.

ELECTRICAL ENGINEERING DEPARTMENT Faculty of Industrial Technology Sepuluh Nopember Instituteof Technology Surabaya 2015

KONTROL TOLERANSI KESALAHAN SENSOR DAN AKTUATOR BERBASIS PROPORTIONAL-PROPORTIONAL INTEGRAL OBSERVER UNTUK SISTEM PENDULUM KERETA

TUGAS AKHIR

Diajukan untuk Memenuhi Sebagian Persyaratan Untuk Memperoleh Gelar Sarjana Teknik Pada Bidang Studi Teknik Sistem Pengaturan Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember

Menyetujui :

EXNOLOGI DE

JULL 2015 JULU 2015 JURUSAN TEKNIK ELEKTRO

Dosen Pembimbing I,

Dosen Pembimbing II,

Inhachti

Prof. Dr. Ir. Mohammad Nuh, DEA NIP, 1959 0617 1984 03 1002 Dr. Trihastuti Agustinah, S.T., M.T. NIP. 1968 0812 1994 03 2001

KONTROL TOLERANSI KESALAHAN SENSOR DAN AKTUATOR BERBASIS PROPORTIONAL-PROPORTIONAL INTEGRAL OBSERVER UNTUK SISTEM PENDULUM KERETA

Nama : Ardiansyah Pembimbing : 1. Prof. Dr. Ir. Mohammad Nuh, DEA 2. Dr. Trihastuti Agustinah, ST., MT.

ABSTRAK

Kontrol Toleransi Kesalahan atau Fault Tolerant Control (FTC) adalah suatu kontrol untuk meminimalisasi kesalahan yang terjadi pada sensor atau aktuator suatu sistem. Permasalahan pada FTC adalah bagaimana mendeteksi dan mengisolasi kesalahan pada sensor dan aktuator yang terjadi. Pada tugas akhir ini dirancang suatu FTC dengan merekontruksi dan mengompensasi kesalahan sensor dan aktuator yang teriadi secara simultan pada sistem nonlinear) menggunakan Proportional-Proportional Integral Observer (PPIO). Dengan menggunakan Linear Matrix Inequality (LMI), gain estimator kesalahan sensor dan aktuator ditentukan. Kontrol nominal sistem menggunakan Fuzzy Takagi-Sugeno (T-S) konsep Model Referensi PDC (Parallel Distributed Compensation) dengan state feedback gain ditentukan menggunakan LMI pole placement. Implementasi diterapkan pada sistem pendulum kereta, yaitu sistem nonlinear yang biasa berfungsi untuk menguji berbagai metode kontrol. Dari hasil simulasi dan implementasi, Kontrol Toleransi Kesalahan (FTC) yang dibangun dapat mengompensasi kesalahan sensor dan aktuator yang terjadi pada plant. Selain itu, FTC memenuhi performa H_o yang diinginkan.

Kata Kunci : FTC, Fuzzy T-S, LMI, LMI Pole Placement, Proportional-Proportional Integral Observer

FAULT TOLERANT CONTROL BASED ON PROPORTIONAL-PROPORTIONAL INTEGRAL OBSERVER WITH SENSOR AND ACTUATOR FAULTS FOR PENDULUM CART SYSTEM

Name Supervisors : Ardiansyah : 1. Prof. Dr. Ir. Muhammad Nuh, DEA 2. Dr. Trihastuti Agustinah, ST., MT.

ABSTRACT

FTC is a method for compensating or minimalizing faults that occurred mostly in sensor and actuator. There are two types of FTC. Active Fault Tolerant Control (AFTC) and Passive Fault Tolerant Control (PFTC). This undergraduate thesis concerns the designing AFTC to compensate simultaneous actuator and sensor faults. The AFTC is designed based on Fuzzy Takagi-Sugeno Proportional-Proportional Integral Observer (T-S PPIO). Observer gain for sensor and actuator faults are determined using Linear Matrix Inequality (LMI) with Lyapunov Stability and H_{∞} performance approach. The method is implemented on nonlinear system (pendulum cart system). The thesis uses Fuzzy Takagi-Sugeno with Model Reference Parallel Distribution Compensation (PDC) to ensure stability when no fault occurred in system (nominal control). State feedback gain of nominal control is determined using LMI Region. In addition, to satisfy implementation with real plant condition, the thesis uses Input-Output constraints. Simulation results show that designed FTC has the capability to obviate faults that occurred in sensor, actuator, or both sensor and actuator. The system also satisfies H_{∞} performance with L₂-gain less than determined attenuation. Implementation results shows FTC observer can estimate and compensate faults occurred in pendulum cart with certain condition.

Keywords : FTC, Fuzzy T-S, LMI, LMI Pole Placement, Proportional-Proportional Integral Observer

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa yang telah melimpahkan Rahmat dan Karunia-Nya sehingga dapat diselesaikan karya tulis ilmiah yang berjudul **"Kontrol Toleransi Kesalahan Sensor Dan Aktuator Berbasis** *Proportional-Proportional Integral Observer* **Untuk Sistem Pendulum Kereta"** dengan baik. Dalam kesempatan ini penulis mengucapkan terimakasih kepada :

- 1. Bapak Prof. Dr. Ir. Mohammad Nuh, DEA dan Ibu Trihastuti Agustinah, S.T, M.T, selaku dosen pembimbing utama yang memberikan saran, rekomendasi dan masukan dalam Tugas Akhir ini.
- 2. Dr. Tri Arief Sardjono, selaku ketua juruan Teknik Elektro yang memberikan rekomendasi dan motivasi guna mengembangkan Tugas Akhir ini.
- 3. Ayah dan Ibu tercinta yang memberikan doa dan motivasi dalam belajar dan menyelesaikan Tugas Akhir.
- 4. Ibu Katherin Indriawati, S.T. M.T. yang banyak memberikan saran terkait FTC guna pengembangan penelitian ini
- 5. Mujianto P. Saputro, Hendra Antomy, Aldhino Putra, Fandi Yunus Dedy Bachtiar sebagai tim utama dalam merancang dan membuat FTC dalam Tugas Akhir ini
- 6. Aisyah Mega Permata, yang menjadi motivasi hidupku, mudahmudahan ini menjadi awal untuk kita
- 7. Rekan-rekan Teknik Elektro ITS, yang memberikan dukungan dan saran guna kemajuan proses penelitian ini.

Penulis menyadari bahwa pengerjaan Tugas Akhir ini masih jauh dari sempurna. Untuk itu, penulis memohon maaf atas segala kekurangan dalam Buku Tugas Akhir ini. Semoga Tugas Akhir ini dapat memberikan manfaat dalam pengembangan ilmu pengetahuan dan teknologi.

Demikian kata-kata dari penulis, dan penulis sangat berharap adanya masukan guna penyempurnaan Tugas Akhir. Terimakasih

Surabaya, 2015

Penulis

DAFTAR ISI

HALAN	IAN JUDULi
PERNY	ATAAN KEASLIAN TUGAS AKHIRiii
ABSTR	AK
ABSTR	ACTix
KATA I	PENGANTARxi
DAFTA	R ISI
DAFTA	R N <mark>OTASI DAN SIM</mark> BOL xxv
KONVI	ENSI PENULISAN TUGAS AKHIR
BABIH	PENDAHULUAN
1.1.	Latar Belakang
1.2.	Perumusan Masalah
1.3.	Tujuan Penelitian
1.4.	Metodologi Penelitian
1.5.	Sistematika Laporan
1.6.	Relevansi
BAB II	TEORI PENUNIANG 7
2.1	Tinjauan Pustaka
2.2.	Sistem Pendulum Kereta
2.3.	Model Fisik Sistem Pendulum Kereta
2.4.	Model Matematika Sistem Pendulum Kereta
2.5.	Teori Fuzzy
2.6.	Model Fuzzy Takagi-Sugeno (T-S)
2.7.	Model Referensi Berbasis Kompensator
2.8.	Pengertian dan Klasifikasi Kesalahan (<i>fault</i>)
2.9.	Kontrol Toleransi Kesalahan (Fault Tolerant Control)
2.10.	Kontrol Toleransi Kesalahan Aktif (AFTC)
2.11.	Observer
2.12.	. Proportional-Proportional Integral Observer
2.13.	Estimasi dan Kompensasi
2.14.	Linear Matrix Inequality (LMI)
2.15.	. LMI Region dan Pole Placement LMI Region
2.16.	Batasan Input-Output
2.17.	Performa $H\infty$

BAB III PERANCANGAN SISTEM	37
4.1. Linearisasi Model Matematika Sistem Pendulum Kereta	37
4.2. Pemodelan Fuzzy Takagi-Sugeno (T-S)	
4.3. Perancangan Kontroler Fuzzy T-S Pole Placement	
dengan Batasan Input-Output	42
4.3.1. Perancangan Kontroler berbasis Model Referensi	43
4.3.2. Perancangan Fuzzy T-S Pole Placement	44
4.3.3. Batasan Input-Output	46
4.4.7 Perhitungan Gain Kontroler	48
4.5. Desain Observer Kesalahan Sensor dan Aktuator berbasis	
Fuzzy T-S PPIO	
(Proportional Proportional Integral Observer)	51
4.5.1. Desain Observer Kesalahan Sensor berbasis	
- Fuzzy T-S PPIO	51
4.5.2. Desain LMI (<i>Linear Matrix Equalities</i>) untuk	
Observer Kesalahan Sensor	
4.5.3. Desain <i>Observer</i> Kesalahan Aktuator Berbasis	
T-S PPIO	61
4.5.4. Desain LMI (<i>Linear Matrix Equalities</i>) untuk Observe	er
Kesalahan Aktuator	62
4.6. Perhitungan Observer gain Kesalahan Sensor dan Aktuator	66
4.7. Diagram Simulink Kontroler dan FTC untuk Simulasi dan	
Implementasi	68
BAB IV HASIL SIMULASI DAN IMPLEMENTASI	71
4.8. Hasil Simulasi	71
4.8.1. Hasil Simulasi Kontrol Nominal dengan Berbagai	
Kondisi Awal tanpa Kesalahan (Fault-Free Case)	71
4.8.2. Hasil Simulasi Kontrol Nominal dengan Berbagai	
Kondisi Kesalahan (Faulty Case)	77
4.8.3. Hasil Simulasi Sistem Kontrol Nominal dengan	
Kom pensasi Kesalahan (Sensor atau Aktuator)	97
4.8.4. Hasil Simulasi Sistem Kontrol Nominal dengan	
Kompensasi Kesalahan Secara Simultan (Sensor dan	
Aktuator)	110
4.9. Hasil Implementasi	117
4.9.1. Hasil Implementasi Kontrol Nominal tanpa Kesalaha	n
(Fault-Free Case)	119

	4.9.2.	Hasil Implementasi Kontrol Nominal dengan Berb	oagai
		Kondisi Kesalahan (Faulty Case)	132
	4.9.3.	Hasil Implementasi Sistem Kontrol Nominal deng	an
		Kompensasi Kesalahan	
		(Sensor, Aktuator dan Simultan)	144
BAB V	PENU	TUP	145
DAFT	AR PUS	STAKA	147
LAMP	IRAN.	AR DEVER DEVER	149
DAFT	AR RIV	VAYAT HIDUP	185



	DAFTAR TABEL
Tabel 2.1	Klasifikasi Kesalahan Sensor
Tabel 2.2	Klasifikasi Kesalahan Aktuator
Tabel 4.1	Perbandingan Respon Simulasi dengan Berbagai
	Kondisi Awal75
Tabel 4.2	Respon Pengaruh Pemberian Kondisi Awal pada Plant
	dengan Observer yang Terkoneksi Kontrol Nominal 76
Tabel 4.3	Respon Kontrol Nominal Terhadap Kesalahan Sensor
And A	Berbentuk Sinus dan Random
Tabel 4.4	Respon Kontrol Nominal Terhadap Kesalahan Sensor
DARE DI DA	Berbentuk Intermittent, Incipient dan Abrupt
Tabel 4.5	Respon Kontrol Nominal Terhadap Kesalahan Aktuator
	Berbentuk Sinus dan Random
Tabel 4.6	Respon Kontrol Nominal Terhadap Kesalahan Aktuator
	Berbentuk Intermittent, Incipient dan Abrupt
Tabel 4.7	Kriteria Kompensasi dan Performansi dari Observer
	Kesalahan Sensor
Tabel 4.8	Kriteria Kompensasi dan Performansi dari Observer
	Kesalahan Sensor
Tabel 4.9	Kriteria Kompensasi dan Performansi dari Observer
	Kesalahan Aktuator
Tabel 4.10	Kriteria Kompensasi dan Performansi dari Observer
T 1 1 4 1 1	Kesalahan Sensor
Tabel 4.11	Kriteria Kompensasi dan Performansi dari Observer
Tabal 412	Kesalahan Aktuator
Tabel 4.12	Kriteria Kompensasi Kesalahan Sensor dan Aktuator 114
1 abel 4.13	Kriteria Kompensasi Kesalanan Sensor dan Aktuator
Tabel 4 14	Kritaria Pospon Kontrol Nominal Implementasi 118
Tabel 4.14	Ritteria Respon Kontrol Nominal Tarbadan Kasalahan Sansor
1 abel 4.13	Berbentuk Sinus <i>Abrunt</i> dan Incipient Implementasi 122
Tabel 4 16	Respon Kontrol Nominal Terhadan Kesalahan Aktuator
1 abel 4.10	Berbentuk Sinus Abrunt dan Incipient Implementasi 126
Tabel 4 17	Respon Kontrol Nominal Terhadan Kesalahan Simultan
10001 4.17	Berbentuk Sinus, <i>Abrupt</i> dan <i>Incinient</i> Implementasi 131
Tabel 4.18	Kriteria Respon Kontrol Nominal Dengan Kompensasi
10001-110	dan Tanpa Kompensasi Kesalahan Sensor Hasil
Ser 7.	Implementasi



DAFTAR GAMBAR

Gambar 2.1	Perangkat Sistem Kontrol Sistem Pendulum Kereta9
Gambar 2.2	Diagram Fisik Sistem Pendulum Kereta
Gambar 2.3	Derajat Keanggotaan Himpunan Fuzzy
Gambar 2.4	Bentuk-bentuk Fungsi Keanggotaan14
Gambar 2.5	Model Referensi
Gambar 2.6	Jenis Kesalahan Berdasarkan Karakteristiknya
Gambar 2.7	Klasifikasi Kesalahan Berdasarkan Letaknya
Gambar 2.8	Skema dari Kontrol Toleransi Kesalahan (FTC)
Gambar 2.9	Hardware Redundancy pada Sistem
Gambar 2.10	Skema Model-Based Fault Diagnosis
Gambar 2.11	Struktur Lengkap AFTC
Gambar 2.12	Blok Diagram Observer
Gambar 2.13	Struktur Umum Proportional-Proportional Integral
	<i>Observer</i>
Gambar 2.14	Daerah pada LMI Region (Conic Sector+Half Plane). 30
Gambar 2.15	Diagram Blok Sistem
Gambar 2.16	Diagram Blok Sistem Persamaan (2.40)
Gambar 2.17	Penyederhanaan Sistem dalam Bentuk LFT35
Gambar 3.1	Fungsi keanggotaan <i>fuzzy</i> T-S41
Gambar 3.2	Desain Kontroler Nominal Keseluruhan
Gambar 3.3	Region dari pole-pole sistem lup tertutup (D)45
Gambar 3.4	Desain Observer PPIO Aktuator dan Sensor
Gambar 3.5	Desain Observer Kesalahan Sensor
Gambar 3.6	Low Pass Filter (LPF) pada Observer
Gambar 3.7	Desain observer kesalahan aktuator
Gambar 3.8	Diagram SIMULINK Matlab Hasil Perancangan69
Gambar 3.9	Diagram SIMULINK untuk Implementasi70
Gambar 4.1	Respon Posisi Kontrol Nominal Hasil Simulasi
Gambar 4.2	Respon Sudut Pendulum dengan Berbagai
	Kondisi Awal Sudut Pendulum Hasil Simulasi
Gambar 4.3	Sinyal Kontrol dengan Berbagai Kondisi
	Awal Sudut Pendulum Hasil Simulasi
Gambar 4.4	Kontrol Nominal dengan Berbagai Kondisi Awal Sudut
	Pendulum dan Kecepatan Sudut Pendulum Hasil
	Simulasi
Gambar 4.5	Respon Sudut Pendulum dengan Berbagai Kondisi Awal
	Sudut Pendulum dan Kecepatan Sudut Hasil Simulasi 74

Gambar 4.6	Sinyal Kontrol dengan Berbagai Kondisi Awal Sudut
	Pendulum dan Kecepatan Sudut Hasil Simulasi
Gambar 4.7	Respon Pengaruh Pemberian Kondisi Awal pada Plant
	dengan Observer yang Terkoneksi Kontrol Nominal
	Hasil Simulasi
Gambar 4.8	Faulty-case Sensor Kesalahan Bias Sinus
	Hasil Simulasi77
Gambar 4.9	Estimasi Kesalahan Sensor Berbentuk Sinus Hasil
	Simulasi
Gambar 4.10	Error Estimasi Kesalahan Sensor Sinus Simulasi
Gambar 4.11	Faulty-case Sensor dengan Kesalahan Bias Random
	Gaussian Hasil Simulasi
Gambar 4.12	Estimasi Kesalahan Sensor Random Gaussian Hasil
	Simulasi
Gambar 4.13	Error Estimasi Kesalahan Sensor Random Gaussian
	Hasil Simulasi
Gambar 4.14	Faulty Case Sensor dengan Kesalahan Bias Abrupt Hasil
	Simulasi
Gambar 4.15	Estimasi Kesalahan Sensor Bias Abrupt Simulasi82
Gambar 4.16	Faulty Case Sensor dengan Kesalahan Bias Intermittent
	Hasil Simulasi
Gambar 4.17	Estimasi Kesalahan Sensor Bias Intermittent Hasil
	Simulasi
Gambar 4.18	Faulty-Case Sensor dengan Kesalahan Bias Sensor
	Incipient Hasil Simulasi
Gambar 4.19	Estimasi Kesalahan Sensor Bias Incinient Simulasi85
Gambar 4.20	Faulty-Case Aktuator dengan Kesalahan Bias Sinus
	Hasil Simulasi
Gambar 4.21	Estimasi Kesalahan Aktuator Bias Sinus Simulasi87
Gambar 4.22	Error Estimasi Kesalahan Aktuator Hasil Simulasi87
Gambar 4.23	Faulty-Case Aktuator Kesalahan Bias Random Gaussian
	Hasil Simulasi
Gambar 4.24	Estimasi Kesalahan Aktuator Bias Random Gaussian
	Hasil Simulasi
Gambar 4.25	Error Estimasi Kesalahan Aktuator Bias Random
	Gaussian Hasil Simulasi
Gambar 4.26	Faulty Case Aktuator dengan Kesalahan Bias Abrupt
	Hasil Simulasi90

Gambar 4.27	Estimasi Kesalahan Aktuator Bias Abrupt Hasil
	Simulasi
Gambar 4.28	Faulty Case Aktuator dengan Kesalahan Bias
	Intermittent Hasil Simulasi
Gambar 4.29	Estimasi Kesalahan Aktuator Bias Intermittent Hasil
	Simulasi
Gambar 4.30	Faulty Case Aktuator dengan Kesalahan Bias Incipient
	Hasil Simulasi
Gambar 4.31	Estimasi Kesalahan Aktuator Aditif Incipient Hasil
	Simulasi
Gambar 4.32	Faulty Case Aktuator dengan Kesalahan Loss of
	Sensitivity Hasil Simulasi
Gambar 4.33	Faulty Case Sensor dengan Kesalahan Loss of
	Measurement Hasil Simulasi
Gambar 4.34	Faulty Case Aktuator dengan Kesalahan Loss of
	Effectiveness Hasil Simulasi
Gambar 4.35	Faulty Case Aktuator dengan Kesalahan Aktuator Loss
	of Measurement Hasil Simulasi
Gambar 4.36	Kompensasi Faulty Case Sensor dengan Berbagai
	Kesalahan Hasil Simulasi
Gambar 4.37	L ₂ -gain dari Observer Kesalahan Sensor berbentuk
	Sinus Hasil Simulasi
Gambar 4.38	Kompensasi Observer Sensor Terhadap Berbagai Jenis
	Kesalahan Berdasarkan Karakteristiknya Simulasi 100
Gambar 4.39	L2-Gain Masing-Masing untuk Kesalahan Berbagai
	Jenis Kesalahan Sensor Hasil Simulasi 102
Gambar 4.40	Kompensasi Faulty Case Aktuator dengan Berbagai
	Jenis Kesalahan Hasil Simulasi 103
Gambar 4.41	L ₂ -Observer gain Kesalahan Aktuator Kesalahan Sinus
	Hasil Simulasi 104
Gambar 4.42	Kompensasi Faulty Case Aktuator dengan Berbagai
	Jenis Kesalahan Berdasarkan Karakteristiknya Hasil
	Simulasi 105
Gambar 4.43	Kompensasi Faulty-Case Sensor Multiplikatif (Loss of
	Sensitivity 70%) Hasil Simulasi 107
Gambar 4.44	Estimasi Kesalahan Sensor Multiplikatif (Loss of
	Sensitivity) Hasil Simulasi107
Gambar 4.45	Kompensasi Faulty Case Sensor Kesalahan Multiplikatif
	(Loss of Measurement) Hasil Simulasi 108

Gambar 4.46	Kompensasi Faulty-Case Aktuator dengan Kesalahan
	Multiplikatif (Loss of Effectiveness 70%) Simulasi 109
Gambar 4.47	Estimasi Kesalahan Aktuator Multiplikatif (Loss of
	Effectiveness) Hasil Simulasi
Gambar 4.48	Kompensasi Kesalahan Aktuator Multiplikatif (Loss of
	Measurement) Hasil Simulasi
Gambar 4.49	Lo-Gain Masing-Masing untuk Berbagai Jenis
	Kesalahan Simultan Berdasarkan Karakteristiknya Hasil
	Simulasi DXG DXG DXG 11157
Gambar 4.50	Kompensasi Faulty Case Sensor dan Aktuator secara
Guilliour neo	Simultan dengan Berbagai Jenis Kesalahan Hasil
	Simulasi
Gambar 4.51	Estimasi Kesalahan Sensor dan Aktuator Sinus yang
	teriadi Secara Simultan Hasil Simulasi
Gambar 4.52	Estimasi Kesalahan Sensor dan Aktuator Random
Guillour nez	Gaussian vang terjadi Secara Simultan Simulasi 114
Gambar 4.53	Kompensasi <i>Faulty Case</i> Sensor dan Aktuator secara
	Simultan dengan kesalahan pada persamaan (4.23) Hasil
	Simulasi
Gambar 4.54	Estimasi Kesalahan Sensor dan Aktuator persamaan
	(4.23) vang terjadi Secara Simultan
Gambar 4.55	Respon Posisi Kereta Tanpa Kesalahan Hasil
	Implementasi 117
Gambar 4.56	Respon Sudut Pendulum Hasil Implementasi
Gambar 4.57	Sinval Kontrol Hasil Implementasi
Gambar 4.58	Faulty-case Sensor dengan Kesalahan Aditif Sinus119
Gambar 4.59	Sudut Pendulum saat Faulty-Case Implementasi 120
Gambar 4.60	Estimasi Kesalahan Sensor Aditif Sinus Hasil
	Implementasi
Gambar 4.61	Faulty-Case Sensor dengan Berbagai Jenis Kesalahan
	Berdasarkan Karakteristiknya Hasil Implementasi 121
Gambar 4.62	Respon Sudut Pendulum dengan Berbagai Jenis
	Kesalahan Hasil Implementasi
Gambar 4.63	Faulty-Case Aktuator dengan Kesalahan Aktuator
	Sinus Hasil Implementasi
Gambar 4.64	Sinyal Kontrol Faulty-Case Aktuator Sinus Hasil
Sel S	Implementasi
Gambar 4.65	Estimasi Kesalahan Aktuator Sinus Implementasi124

Gambar 4.66	Faulty-Case Aktuator dengan Berbagai Jenis Kesalahan
	Hasil Implementasi
Gambar 4.67	Sinyal Kontrol Berbagai Jenis Kesalahan Hasil
	Implementasi
Gambar 4.68	Estimasi Kesalahan Aktuator Hasil Implementasi 127
Gambar 4.69	Faulty-Case dengan Kesalahan Simultan berbentuk
	Sinus Hasil Implementasi
Gambar 4.70	Estimasi Kesalahan Sensor dan Aktuator Sinus Secara
	Simultan Hasil Implementasi
Gambar 4.71	Faulty Case Simultan Dengan Kesalahan Incipient
	Implementasi
Gambar 4.72	Estimasi Kesalahan Sensor dan Aktuator (Simultan)
	Hasil Implementasi
Gambar 4.73	<i>Faulty-Case</i> Sensor dan Aktuator (Simultan) Abrupt
	Hasil Implementasi
Gambar 4.74	Estimasi Kesalahan Sensor dan Aktuator (Simultan)
	Abrupt Hasil Implementasi
Gambar 4.75	Kompensasi Kesalahan Sensor Berbagai Jenis
Sels 7 S	Kesalahan Hasil Implementasi
Gambar 4.76	Sudut Pendulum Hasil Kompensasi Berbagai Jenis
	Kesalahan Hasil Implementasi 135
Gambar 4.77	Kompensasi Kesalahan Aktuator Berbagai Jenis
	Kesalahan Hasil Implementasi
Gambar 4.78	Sinyal Kontrol Kompensasi Berbagai Jenis Kesalahan
	Sinus Hasil Implementasi138
Gambar 4.79	Kompensasi Kesalahan Sensor dan Aktuator (Simultan)
	Berbagai Kesalahan
Gambar 4.80	Sudut Pendulum Hasil Kompensasi Sensor dan Aktuator
	(Simultan) Hasil Implementasi141
Gambar 4.81	Sinyal Kontrol Kompensasi Sensor dan Aktuator
	(Simultan) Hasil Implementasi142

DAFTAR NOTASI DAN SIMBOL

X	Vektor state x
Î	Estimasi dari <i>state</i> x
x	Augmentasi dari state x dengan state lain
x_n	State ke-n
\dot{x}_n	Turunan pertama state ke-n
	Anggota himpunan
¢	Bukan anggota himpunan
I	Matriks Identitas
\mathbf{A}^{T}	Transpos matriks A
\mathbf{A}^{-1}	Invers matriks A
A>0	Matriks A definit positif
A <0	Matriks A definit negatif
A	Matriks Augmentasi
	Transpos dari matriks diagonal
$\mathbf{diag}(f_1,, f_n)$	Matriks diagonal dengan f_n merupakan elemen
	diagonal ke-n
1	Bilangan imajiner
μ	Bilangan konstanta iterasi Fast Fault Estimation
γ	Lingkat pelemanan
	Juri juri hotos I MI Pagion
p	Jaii-jaii batas Livii Region
	Determinan matriks A
	m-norm
	Lanorm
sup	Supremum
sgn	Fungsi signum
$\Sigma(.)$	Penjumlahan dari nilai deret
ΠĆ	Perkalian dari nilai deret

RIWAYAT HIDUP PENULIS



Penulis bernama lengkap Ardiansyah, lahir di Kediri pada tanggal 25 Oktober 1992. Pada tahun 2005 penulis menyelesaikan masa belajar di SD Gemaharjo IV. Penulis melanjutkan pendidikan di **SMPN** 2 Tegalombo dan menyelesaikan studinva pada tahun 2008. Kemudian penulis melanjutkan pendidikan di SMAN 1 Pacitan yang menyelesaikan studi pada tahun 2011. Karena ketertarikan pada bidang studi, membuat penulis melanjutkan pendidikan

tinggi di Institut Teknologi Sepuluh Nopember jurusan Teknik Elektro konsentrasi Sistem Pengaturan. Semasa kuliah, penulis aktif di kegiatan organisasi yaitu BEM ITS di Kementerian Riset dan Teknologi dan CSSC (*Control System Service Center*). Pada bulan Juni 2015 penulis mengikuti ujian Tugas Akhir untuk memenuhi persyaratan gelar Sarjana Teknik Elektro.



BAB I PENDAHULUAN

1.1. Latar Belakang

Dengan semakin majunya teknologi, sistem teknologi akan semakin kompleks, sistem kontrol akan semakin dibutuhkan. Tantangan dalam keselamatan kerja, proteksi dan kenyamanan menjadi tujuan utama diciptakannya teknologi. Semua sistem kontrol, seberapa tangguh komponen yang digunakan, seberapa akurat proses manufaktur, akan mengalami penurunan performansi yang mengakibatkan sistem bekerja tidak sesuai dengan kriteria yang diinginkan. Salah satu cara untuk memastikan keandalan sistem dan mentoleransi kesalahan yang ada pada sistem adalah dengan Kontrol Toleransi Kesalahan (*Fault Tolerant Control*) [1].

Fault Tolerant Control (FTC) adalah sistem kontrol yang secara khusus didesain untuk mendeteksi sekaligus memperbaiki sistem saat terjadi kesalahan agar tetap memiliki respon sesuai dengan yang diinginkan [1] [2]. Desain dari FTC telah dikembangkan selama bertahun-tahun untuk sistem yang aman dan reliabel. Sensor dan aktuator adalah komponen yang penting dalam sistem kontrol. Ketika terjadi kesalahan pada kedua komponen ini, perubahan signifikan dapat terjadi pada karakteristik sistem, seperti kerusakan sistem *plant* atau steady state error [1].

Sistem kontrol otomasi secara luas digunakan dalam berbagai aplikasi. Sistem kontrol lup tertutup memastikan kestabilan sistem [3], namun semakin banyak sistem lup tertutup dalam suatu proses, kemungkinan untuk terjadinya kesalahan pada komponen-komponen sistem juga menjadi semakin tinggi. Kesalahan tersebut dapat terjadi pada sensor, aktuator atau komponen lain pada sistem. Kejadian ini dapat menyebabkan sistem menjadi tidak stabil. Pada sistem lup tertutup tidak semua perbaikan dapat dilakukan dengan cepat dan sesegera mungkin, karena itulah penting untuk mendesain suatu desain kontrol yang memastikan performansi sistem berada pada keadaan tetap pada kriteria yang diinginkan maskipun terjadi kesalahan sistem [4].

Sistem pendulum terbalik adalah sistem nonlinear yang pendekatan kontrolnya sulit dilakukan menggunakan kontrol klasik/linear. Salah satu metode kontrol linear yang mudah dilakukan adalah dengan

pendekatan *Parallel Distributed Compensation* (PDC) model Takagi Sugeno. Pendekatan model Takagi-Sugeno telah banyak diaplikasikan untuk sistem nonlinear, karena memiliki berbagai macam kelebihan [**5**]. Kesalahan dalam sistem pendulum dapat berupa kesalahan sensor, dimana kualitas sensor tidak lagi memiliki akurasi yang tinggi, atau kesalahan aktuator dimana putaran motor tidak sesuai dengan yang diharapkan. Karena itulah dibutuhkan suatu kontrol untuk meminimalisasi kesalahan yang terjadi pada sensor atau aktuator.

Pada tugas akhir ini dirancang Kontrol Toleransi Kesalahan Aktif (Active Fault Tolerant Control) untuk sistem pendulum terbalik beroda dua. Kontrol Toleransi Kesalahan ini digunakan untuk meminimalisasi kesalahan yang diakibatkan oleh sensor dan aktuator pada sistem. Fault Detection and Isolation (FDI) digunakan dalam sistem ini menggunakan Proportional-Proportional Integral Observer. Desain observer menggunakan teori kestabilan Lyapunov dan LMI, dimana akan didapatkan state feedback gain. Sehingga diharapkan, setelah terjadi kesalahan pada sensor dan aktuator pada rentang yang ditoleransi, sistem tetap memenuhi kriteria yang diinginkan.

1.2. Perumusan Masalah

Pada Tugas Akhir ini yang menjadi persoalan utama adalah kesalahan pada sensor dan aktuator dapat menyebabkan sistem pendulum kereta mengalami penurunan performa tracking yang telah Diperlukan mekanisme kompensasi sehingga sistem dirancang. pendulum kereta tetap mengikuti sinyal referensi yang telah ditentukan dengan tetap mempertahankan batang pendulum dalam kondisi pada posisi terbalik. Karena pada implementasi panjang rel dan sinyal kontrol terbatas, maka perlu adanya batasan pada sinyal kontrol dan keluaran. Pada sistem nyata, kesalahan sensor dan aktuator dapat terjadi secara bersamaan (simultan) sehingga diperlukan suatu metode untuk mengatasi hal tersebut.

Secara detail, persoalan dalam desain Kontrol Toleransi Kesalahan (FTC) berbasis *Proportional-Proportional Integral Observer* untuk sistem pendulum kereta adalah sebagai berikut:

1). Penentuan model *fuzzy* T-S dari sistem yang diperoleh dari linearisasi model nonlinear *plant* pada beberapa titik kerjanya.

- 2). Desain model *fuzzy* T-S untuk *plant* Sistem Pendulum-Kereta, pemilihan variabel premis, himpunan *fuzzy*, dan banyaknya aturan yang digunakan.
- 3). Desain kontrol *tracking* nominal yang memenuhi batasan *inputoutput* melalui skema PDC.
- 4). Desain observer untuk mengompensasi kesalahan pada sensor atau aktuator dan keduanya secara simultan menggunakan teori kestabilan Lyapunov dan *Linear Matrix Inequality* (LMI).
- 5). Uji sistem hasil desain melalui simulasi dan implementasi pada sistem pendulum kereta dan software Matlab.

1.3. Tujuan Penelitian

Tujuan dari tugas akhir ini adalah mendesain Kontrol Toleransi Kesalahan (FTC) untuk sistem nonlinear dalam hal ini pendulum kereta berbasis Proportional-Proportional Integral Observer. Kontrol nominal tracking dalam hal ini menggunakan fuzzy Takagi Sugeno dengan penentuan gain state feedback menggunakan LMI pole placement dan batasan input-output. Penentuan observer gain menggunakan pendekatan Linear Matrix Inequalities (LMI). Estimasi kesalahan dari observer digunakan untuk mengompensasi kesalahan aktual yang terjadi.

1.4. Metodologi Penelitian

Adapun metode penelitian dalam tugas akhir ini adalah :

1). Studi Literatur

Pada tahap ini dicari referensi-referensi yang relevan dengan tugas akhir dan digunakan sebagai acuan untuk mempelajari dasar dan pembahasan tugas akhir. Referensi yang berkaitan dengan Tugas Akhir ini antara lain, model fisik *plant "Digital Pendulum Mechanical Unit* 33-200", desain kontrol *pole placement, Linear Matrix Inequalities* (LMI), *observer* dan performansi H_{∞}

2). Perancangan Kontrol Nominal *Tracking*

Pada tahap ini, dirancang kontrol nominal yang membuat sistem pendulum kereta stabil. Kontrol nominal adalah keadaan sistem tanpa kesalahan sensor dan aktuator (*fault-free case*).

3). Perancangan *Observer* Kesalahan Sensor dan Aktuator (C) Setelah kontrol nominal mampu mengikuti trayektori yang diinginkan (*tracking*), dirancang *observer* yang mampu mengatasi kesalahan sensor dan aktuator (*faulty case*)

4). Simulasi

Simulasi menggunakan program Matlab 9.5.1. Simulasi sistem dalam keadaan *(faulty-case)* yaitu kesalahan sensor dan aktuator. Pada tahap ini dilihat seberapa besar kompensasi yang dapat ditoleransi oleh *observer* yang telah dirancang.

5). Implementasi

Implementasi dilakukan menggunakan *plant* sistem pendulum kereta (yang tersedia *Digital Pendulum Mechanical Unit* 33-200) di laboratorium Sistem Pengaturan b.105).

6). Penyusunan buku tugas akhir

Penyusunan buku terdiri atas pendahuluan, tinjauan pustaka, perancangan, simulasi dan implementasi, dan penutup. Penulisan buku memperhatikan kaidah EYD dan format tugas akhir yang telah disediakan.

1.5. Sistematika Laporan

Tugas Akhir ini dibagi dalam lima Bab dengan sistematika sebagai berikut:

Bab I – 🤃 Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, metodologi penelitian, sistematika laporan dan relevansi.

Bab II

: Teori Penunjang

Bab ini meliputi tinjauan pustaka, sistem pendulum kereta, model fisik pendulum kereta, model matematika sistem pendulum kereta, teori *fuzzy*, model *fuzzy* Takagi-Sugeno (T-S), model referensi berbasis kompensator, pengertian dan klasifikasi kesalahan, kontrol toleransi kesalahan, kontrol toleransi kesalahan aktif, *observer*, *proportional integral observer*, estimasi dan kompensasi, LMI, LMI region dan *pole placement* LMI region. : Perancangan Sistem

Bab III

Bab ini meliputi linearisasi model matematika sistem pendulum kereta, pemodelan *fuzzy* T-S, perancangan kontroler *fuzzy* T-S *pole placement* dengan batasan *input-output*, perancangan kontroler berbasis model referensi, desain *observer* kesalahan sensor dan aktuator berbasis T-S PPIO (*Proportional* Proportional Integral Observer), desain observer kesalahan sensor berbasis PPIO dan desain observer kesalahan aktuator berbasis PPIO.

Bab IV: Hasil Pengujian Simulasi dan Implementasi Bab ini meliputi hasil pengujian simulasi dan hasil implementasi pada sistem pendulum kereta beserta analisisnya.

Bab V : Penutup

Bab ini berisi tentang kesimpulan dan saran dari hasil pengujian yang telah diperoleh.

1.6. Relevansi

Hasil dari Tugas Akhir ini diharapkan menjadi acuan untuk penelitian atau Tugas Akhir tentang FTC selanjutnya, pengembangan dan perbandingan sehingga menghasilkan metode FTC yang lebih sederhana namun memiliki performa yang sama.

BAB II TEORI PENUNJANG

2.1. Tinjauan Pustaka

Kontrol Toleransi Kesalahan (FTC) adalah sistem kontrol yang mampu mendeteksi sekaligus memperbaiki sistem saat terjadi kesalahan agar tetap memiliki respon sesuai dengan yang diinginkan. Desain dari FTC bermula dari perancangan FDD (*Fault Detection and Diagnosis*). FDD adalah metode untuk mendeteksi adanya kesalahan pada sistem terutama sensor dan aktuator. FDD dibuat dengan adanya kebutuhan untuk mengatasi sistem yang mengalami kesalahan (*fault*), seperti pada kejadian kecelakaan pada sistem penerbangan tahun 1970, menginisiasi riset dan pengembangan FDD [**4**].

Disisi lain industri yang berkembang pesat dan kebutuhan manusia akan industri semakin bertambang, keandalan akan komponenindustri semakin dibutuhkan. Pengembangan komponen metode berdasarkan kebutuhan kompensasi dikembangkan ini. Metode kompensasi kesalahan dan FDD inilah dinamakan dengan FTC. Metode FDD dan kompensasi dalam FTC memiliki berbagai macam metode. Salah satu metode yang cukup mudah diimplementasikan adalah redundancy dan observer. Metode redundancy memerlukan komponen tambahan jika terjadi kerusakan, akan digantikan oleh komponen lain tersebut. Sistem ini cukup mahal untuk diimplementasikan.

Metode selanjutnya adalah *observer*. *Observer* berperan untuk mendeteksi kesalahan pada sistem terutama sensor dan aktuator, setelah mendeteksi adanya kesalahan, kemudian dikompensasi agar dapat kembali ke keadaan nominalnya (keadaan tanpa kesalahan). Dalam implementasinya metode ini banyak dikembangkan algoritmanya karena investasi yang dilakukan lebih sedikit. Dalam [4], metode FTC diterapkan untuk *plant* turbin angin. Penerapan FTC pada turbin angin dapat mengurangi biaya *maintenance* jika terjadi kesalahan yang menyebabkan daya yang dihasilkan tidak maksimal.

2.2. Sistem Pendulum Kereta

Sistem pendulum kereta terdiri dari sepasang pendulum yang terpasang pada sebuah kereta sehingga pendulum tersebut dapat berayun bebas pada bidang vertikal. Kereta digerakkan oleh motor DC yang dihubungkan dengan *belt*. Untuk mengayunkan dan menyeimbangkan

pendulum, kereta digerakkan ke kiri atau ke kanan pada rel yang panjangnya terbatas.

Posisi kereta pada lintasan dapat dipantau melalui sensor posisi yaitu *position encoder*. Sebagai pengaman, digunakan *limit switch* pada masing-masing ujung rel. ketika kereta berada di ujung lintasan, maka limit switch tertekan oleh kereta dan motor DC akan mati, sehingga kereta akan berhenti. Sedangkan posisi sudut pendulum terhadap sumbu vertikal dipantau oleh angle *encoder*.

Dalam implementasi skema kontrol yang dirancang, digunakan sistem pendulum kereta dari *Feedback Instruments* Ltd dengan tipe "Digital Pendulum Mechanical Unit 33-200". Penerapan sistem kontrol dilakukan pada komputer dengan bantuan software Simulink/MATLAB. Komputer dan sistem pendulum kereta terhubung melalui modul "Digital Pendulum Controller 33-201" sebagai kontroler antarmuka, serta board akuisisi data (DAQ) sebagai I/O komputer. Sinyal kontrol dari komputer keluar melalui Digital to Analog Computer (DAC) yang terdapat pada DAQ.

Power amplifier yang terhubung dengan port keluaran DAQ akan menerima sinyal kontrol yang kemudian dikirim ke motor DC untuk menggerakkan kereta. Sinyal respons dari kereta dan pendulum terbaca oleh encoder dan dikirim ke komputer melalui *Analog to Digital Converter (ADC)* pada DAQ. Bagan sistem beserta perangkat pendukungnya secara keseluruhan dapat dilihat pada Gambar 2.1.

2.3. Model Fisik Sistem Pendulum Kereta [3] [6]

Diagram fisik sistem pendulum kereta yang digunakan serta gayagaya yang terjadi pada sistem pendulum kereta secara berturut-turut ditunjukkan pada Gambar 2.2. Secara fisik, sistem pendulum kereta terdiri dari dua bagian utama, yaitu kereta dan pendulum. Kereta hanya dapat bergerak pada rel dalam bidang horizontal dan pendulum berotasi pada bidang vertikal yang bersumbu pada sisi kereta.

Gaya kontrol u yang sejajar dengan dengan rel dikenakan pada kereta. Gaya gesek kereta terhadap rel dinyatakan dengan T_c sedangkan V adalah gaya normal yang bekerja pada sistem pendulum kereta. Massa kereta dan massa pendulum dinyatakan dengan m_c dan m_p . Jarak antara sumbu rotasi pendulum ke pusat massa sistem dinyatakan l sedangkan Jadalah momen inersia sistem terhadap pusat massa sistem.





Gambar 2.2 Diagram Fisik Sistem Pendulum Kereta [3]

2.4. Model Matematika Sistem Pendulum Kereta [7]

Sistem pendulum kereta terdiri dari empat state, yaitu x_1 , x_2 , x_3 , dan x_4 dengan:

- x_1 : Posisi kereta diukur dari titik tengah rel
- x_2 : Sudut pendulum terhadap garis vertikal, diukur berlawanan
 - dengan arah jarum jam ($x_2 = 0$ menyatakan pendulum berada pada posisi terbalik)
- x_3 : Kecepatan kereta
- x_4 : Kecepatan sudut pendulum

Koordinat horizontal pusat massa adalah x_1 - $lsinx_2$, sedangkan koordinat vertikal pusat massa adalah $lcosx_2$. Berdasarkan hukum kedua Newton, sesuai dengan Gambar 2.2 persamaan gerak dapat ditulis,

$$u - T_{c} = (m_{c} + m_{p})(x_{1} - l\sin x_{2})^{"}$$
(2.1)

$$V - (m_{c} + m_{p})g = (m_{c} + m_{p})(l\cos x_{2})^{"}$$
(2.2)

$$(u - T_{c})l\cos x_{2} + Vl\sin x_{2} - D_{p} = Jx_{2}^{"}$$
(2.3)

Dengan D_p adalah momen gesek akibat gerak rotasi pendulum yang proporsional terhadap kecepatan sudut pendulum. D_p dapat dinyatakan dengan $D_p = f_p x_4$.

Persamaan (2.1) dan (2.2) menyatakan gerak translasi dari pusat massa sistem, sedangkan persamaan (2.3) menyatakan gerak rotasi sistem secara keseluruhan terhadap pusat massa sistem. Kecepatan kereta merupakan turunan pertama dari posisi kereta,

$$x_3 = \frac{dx_1}{dt}$$
 atau $x_1 = x_3$

sedangkan kecepatan sudut pendulum merupakan turunan pertama dari posisi sudut pendulum,

$$x_2 = \frac{dx_2}{dt}$$
 atau $x_2 = x_4$

Dengan mengeliminasi V pada persamaan (2.2) dan (2.3) serta beberapa manipulasi matematis dapat diperoleh persamaan state sistem pendulum kereta seperti pada persamaan (2.4).

$$\dot{x}_{1} = x_{3}$$

$$\dot{x}_{2} = x_{4}$$

$$\dot{x}_{3} = \frac{a(u - T_{c} - \mu x_{4}^{2} \sin x_{2}) + l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

$$\dot{x}_{4} = \frac{l \cos x_{2}(u - T_{c} - \mu x_{4}^{2} \sin x_{2}) + \mu g \sin x_{2} - f_{p} x_{4}}{J + \mu l \sin^{2} x_{2}}$$
(2.4)

dengan

$$\mu = (m_c + m_p)l, a = l^2 + \frac{J}{m_c + m_p}, \dot{x}_1 = x_1; i = 1, 2, 3, 4$$

Dari persamaan (2.4) dapat dilihat bahwa sistem pendulum kereta merupakan sistem nonlinear *Single Input Multiple Output (SIMO)*. Untuk mempermudah desain kontroler, maka dilakukan linearisasi pada persamaan (2.4) sehingga teori kontrol linear dapat digunakan.

2.5. Teori Fuzzy [8]

Sebelum munculnya teori logika *fuzzy*, dikenal sebuah logika Boolean yang memiliki nilai benar atau salah secara pasti. Sebaliknya, logika *fuzzy* merupakan sebuah logika yang memiliki nilai kekaburan (*fuzziness*) antara benar dan salah. Dalam teori logika *fuzzy*, sebuah nilai bisa bernilai benar dan salah secara bersamaan namun berapa besar kebenaran dan kesalahan suatu nilai tergantung kepada derajat keanggotaan yang dimilikinya.

Misalkan X merupakan semesta pembicaraan dan x adalah elemen X. Himpunan tegas A, $A \subseteq X$ didefinisikan sebagai kumpulan elemen atau objek $x \in X$, sehingga tiap-tiap x dikatakan sebagai anggota himpunan A atau tidak. Dalam teori himpunan *boolean*, dapat didefinisikan fungsi karakteristik tiap-tiap elemen x dalam X. Sehingga himpunan A dapat dinyatakan dengan (x, 0) atau (x, 1) yang masing-masing menyatakan x \notin A atau x \in A.

Dalam himpunan *fuzzy* (*fuzzy set*), masing-masing elemen memiliki derajat keanggotaannya masing-masing untuk menjadi anggota himpunan A. Sehingga fungsi karakteristik himpunan *fuzzy* bernilai antara 0 sampai 1, yang menyatakan derajat keanggotan masing-masing elemen dalam suatu himpunan. Secara matematis, himpunan *fuzzy* A dalam X dinyatakan:

$\mathbf{A} = \left\{ \left(x, \mu_A(x) \right) | x \in X \right\}$

dengan $\mu_A(x)$ adalah fungsi keanggotaan *(membership function)* untuk himpunan *fuzzy* A. Fungsi keanggotaan memetakan tiap-tiap elemen x ke derajat keanggotaan antara 0 sampai 1.

Sebagai contoh, pada Gambar 2.3 kecepatan dapat dikategorikan menjadi tiga himpunan, yaitu kecepatan rendah, sedang dan tinggi.

Kecepatan yang kurang dari 25 km/jam dikatakan rendah, kecepatan di sekitar 50 km/jam dikatakan sedang, sedangkan kecepatan yang lebih dari 75 km/jam dikatakan tinggi. Dalam himpunan *fuzzy*, kecepatan 60 km/jam bisa dikatakan kecepatan sedang ataupun kecepatan tinggi karena kecepatan 60 km/jam memotong dua fungsi keanggotaan, yaitu kecepatan sedang dan kecepatan tinggi. Namun, sesuai dengan Gambar 2.3, kecepatan 60 km/jam dikatakan lebih condong pada kecepatan sedang karena lebih dekat pada kecepatan 50 km/jam dibanding kecepatan 75 km/jam. Atau dengan kata lain, derajat keanggotaannya lebih besar untuk kecepatan sedang dibanding kecepatan tinggi.



Gambar 2.3 Derajat Keanggotaan Himpunan Fuzzy

Fungsi keanggotaan dari suatu himpunan *fuzzy* dinyatakan dengan derajat keanggotaan suatu nilai terhadap nilai tegasnya yang berkisar antara 0 sampai 1. Fungsi keanggotaan memiliki berbagai bentuk, antara lain segitiga, trapezium, *Gaussian, Generalized Bell*, dan bentuk-bentuk yang lain.

Fungsi keanggotaan segitiga ditentukan oleh tiga parameter (a,b,c) dengan a < b < c sesuai persamaan (2.5). Ketiga parameter tersebut menyatakan letak koordinat x untuk ketiga sudut segitiga.

$$\mu_{segitiga}(x) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{cases}$$
(2.5)

Fungsi keanggotaan trapezium ditentukan oleh empat parameter (a,b,c,d) dengan a < b < c < d sesuai dengan persamaan (2.6). Keempat

parameter tersebut menyatakan letak koordinat *x* untuk keempat sudut trapezium.

$$\mu_{trapesium}(x) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \frac{d-x}{d-c}, & c \le x \le d \\ 0, & d \le x \end{cases}$$
(2.6)

Fungsi Keanggotaan *Gaussian* ditentukan oleh dua parameter (c,σ) sesuai dengan persamaan (2.7) yang menentukan pusat dan lebar fungsi keanggotaan,

$$u_{Gaussian}(x) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma}\right)^2}$$
(2.7)

Fungsi keanggotaan Generalized Bell ditentukan oleh tiga parameter (a,b,c) sesuai dengan persamaan (2.8) dengan parameter b bernilai positif. Parameter c merupakan pusat yang menentukan letak x yang memiliki derajat keanggotaan bernilai satu.

$$\mu_{GeneralizedBell}(x) = \frac{1}{1 + \left|\frac{x - c}{a}\right|^{2b}}$$

Bentuk-bentuk fungsi keanggotaan dapat digambarkan seperti Gambar 2.4. Pada Gambar 2.4, (a) menunjukkan fungsi keanggotaan bentuk segitiga dengan parameter (10,25,40), (b) bentuk trapezium dengan parameter (10,15,30,40), (c) bentuk *Gaussian* dengan parameter (25,8), dan (d) bentuk *Generalized Bell* dengan parameter (10,5,25).

(2.8)

2.6. Model Fuzzy Takagi-Sugeno (T-S) [9]

Kontrol *Fuzzy* T-S banyak diterapkan untuk sistem nonlinear karena kemudahan implementasinya. Model *plant* pada persamaan (2.4) dalam bentuk aturan *Fuzzy* T-S adalah:

Rule j: IF p is M_{j1}



Gambar 2.4 Bentuk-bentuk Fungsi Keanggotaan

dengan A_j dan B_j adalah koefisien matriks yang telah diliniearisasi di beberapa titik daerah kerjanya, *r* adalah jumlah aturan *fuzzy* dan *p* adalah premis dari *fuzzy* T-S. Model *fuzzy* T-S merepresentasikan persamaan state ini yaitu:

$$\dot{\mathbf{x}}(t) = \frac{\sum_{j=1}^{r} \left[w_j(x) \left\{ \mathbf{A}_j \mathbf{x}(t) + \mathbf{B}_j \mathbf{u}(t) \right\} \right]}{\sum_{j=1}^{r} w_j(x)} = \sum_{j=1}^{r} h_j(x) \left\{ \mathbf{A}_j \mathbf{x}(t) + \mathbf{B}_j \mathbf{u}(t) \right\}$$
(2.9)

dengan

$$w_j(x) = \prod_{k=1}^p M_{jk}(x_k)$$

$$h_j(x) = \frac{w_j(x)}{\sum_{j=1}^r w_j(x)}$$

untuk mempermudah notasi penulisan selanjutnya dalam tugas akhir ini, persamaan *fuzzy* T-S disederhanakan menjadi:

$$\begin{split} \mathbf{A}(p) &= \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{A}_{j}, \\ \mathbf{B}(p) &= \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{B}_{j}, \\ \mathbf{C}(p) &= \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{C}_{j}, \end{split}$$

dengan:

$$0 \le \mathbf{h}_j(x) \le 1$$
 dan $\sum_{j=1}^r \mathbf{h}_j(x) = 1$

Persamaan (2.9) dapat ditulis kembali:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x} + \mathbf{B}(p)\mathbf{u}(t)$$

kontroler PDC yang dirancang adalah: Aturan kontroler j:

> **Rule j:** IF p is M_{j1} **THEN** $\mathbf{u}(t) = \mathbf{K}_{j}\mathbf{x}(t)$

dengan K_j adalah gain state feedback yang ditentukan melalui metode pole placement. Secara umum kontroler fuzzy T-S dinyatakan dalam:

$$\mathbf{u}(t) = \sum_{j=1}^{r} \mathbf{h}_{j}(x) \mathbf{K}_{j} \mathbf{x}(t)$$

(2.10)

2.7. Model Referensi Berbasis Kompensator [7]

Model referensi adalah model trayektori yang akan dijadikan masukan sistem lup tertutup. Metode ini adalah salah satu cara agar sistem mampu mengikuti trayektori yang diinginkan. Terdapat berbagai macam metode dalam model referensi, salah satunya adalah metode kompensator. Metode ini cukup sederhana dan mudah diimplementasikan. Kompensator dirancang sesuai sinyal referensi yang ingin didesain. Secara umum, model sinyal referensi adalah:

$$\dot{\mathbf{x}}_{r}(t) = \mathbf{A}_{r} \mathbf{x}_{r}(t) + \mathbf{B}_{r} r(t)$$
(2.11)

dengan $\mathbf{x}_{r}(t)$ adalah sinyal referensi, \mathbf{A}_{r} adalah matriks stabil asimtotik, \mathbf{B}_{r} adalah matrik masukan, dan r(t) adalah masukan referensi yang

nilainya terbatas. Diasumsikan nilai $\mathbf{x}_{r}(t)$, untuk semua t > 0, merepresentasikan trayektori yang diinginkan untuk diikuti oleh $\mathbf{x}(t)$. Bentuk umum dari model referensi adalah:



Suatu sistem dengan persamaan:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

jika dihubungkan dengan *plant* yang dikontrol dengan menganggap *tracking error* e(t) = r(t) - y(t) sebagai bentuk masukan dari model, maka akan diperoleh bentuk berikut:

$$\dot{\mathbf{x}}_{c}(t) = \mathbf{A}_{c}\mathbf{x}_{c}(t) + \mathbf{B}_{c}e(t)$$
$$\mathbf{y}_{r}(t) = \mathbf{x}_{c}(t)$$

(2.13)

(2.14)

(2.12)

dengan

$$\mathbf{A}_{c} = diag \left\{ \mathbf{A} \quad \mathbf{A} \quad \cdots \quad \mathbf{A} \right\}_{error}$$

$$\mathbf{B}_{c} = diag\{ \mathbf{B} \ \mathbf{B} \ \cdots \ \mathbf{B} \}$$

A adalah matriks polinomial karakteristik dari sinyal referensi. Model referensi ini dalam bentuk *augmented system*:

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_{c}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{B}_{c}\mathbf{C} & \mathbf{A}_{c} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_{c}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{c} \end{bmatrix} \mathbf{r}(t)$$

dan sinyal kontrolnya

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{K} & \mathbf{K}_c \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_c(t) \end{bmatrix}$$

dalam bentuk fuzzy T-S Parallel Distribution Compensation:

$$\mathbf{u}(t) = \sum_{j=1}^{r} \mathbf{h}_{j}(x) \mathbf{K}_{j}[\mathbf{x}(t) \quad \mathbf{x}_{c}(t)]$$
$$\mathbf{h}_{j}(x) = \frac{\mathbf{w}_{j}(x)}{\sum_{j=1}^{r} \mathbf{w}_{j}(x)}$$

2.8. Pengertian dan Klasifikasi Kesalahan (fault) [2] [4]

Kesalahan (fault) adalah keadaan yang mengakibatkan sistem lup tertutup mengalami penurunan performa atau degradasi. Blanke [2] mendefinisikan kesalahan sebagai suatu keadaan yang mengakibatkan parameter sistem menyimpang dari keadaan nominalnya. Keadaan nominal adalah keadaan sistem berada pada performa yang diinginkan. Konsep kesalahan, sangat berbeda dengan kegagalan (failure). Kegagalan adalah keadaan lebih menurun dibandingkan kesalahan. Suatu sistem mengalami kegagalan apabila komponen dalam sistem tersebut sama sekali tidak bekerja. Perbedaan keduanya juga tampak pada kontroler yang mampu diterapkan. Ketika sistem terjadi kesalahan dapat ditoleransi oleh kontroler lain sehingga performa sistem berada pada keadaan yang diinginkan, namun ketika terjadi kegagalan, komponen yang berada pada sistem tersebut harus diganti. Konsep ini mendorong suatu pengembangan suatu kontrol vang mampu mentoleransi kesalahan pada komponen yang mengalami kesalahan pada sistem, yang dinamakan Kontrol Toleransi Kesalahan (Fault Tolerant Control).

(2.15)

Berdasarkan karakteristiknya, kesalahan dapat diklasifikasikan menjadi 3 yaitu; *abrupt, incipient, dan intermittent*. Gambaran ketiganya dapat dilihat pada Gambar 2.5



a. Abrupt Fault (Kesalahan Tiba-tiba)

b.

c.

a.

Gambar 2.5 (a) menunjukkan grafik *abrupt fault* terhadap fungsi waktu. Kesalahan ini datang secara acak terhadap fungsi waktu dan besarnya juga secara acak, sehingga menyerupai fungsi step.

Incipient Fault (Kesalahan Perlahan)

Incipent fault lebih mudah diatasi dibandingkan kesalahan yang lain. Hal ini dikarenakan sifatnya yang datang perlahan dan terus membesar menyerupai fungsi ramp. Dalam beberapa kasus sistem yang mengalami incipent fault, terdapat fungsi alarm yang memberi peringatan untuk mencegah kesalahan yang terjadi terus membesar.

Intermittent Fault (Kesalahan Sesaat)

Kesalahan ini menyerupai fungsi impuls, yang datang secara tiba-tiba namun berlangsung sangat singkat, ditunjukkan oleh Gambar 2.5 (c). Kesalahan sesaat tidak mudah diatasi karena sifatnya yang acak dan periodenya yang sangat singkat, namun dapat diantisipasi.

Berdasarkan lokasi dari kesalahan, kesalahan dapat dibagi menjadi tiga; kesalahan sensor, kesalahan aktuator, dan kesalahan proses.



Gambar 2.7 Klasifikasi Kesalahan Berdasarkan Letaknya [4]

Kesalahan Aktuator (Actuator Fault)

Kesalahan ini terletak di aktuator dan berhubungan dengan sinyal kontrol yang dihasilkan oleh aktuator. Ketika terjadi kerusakan, sinyal kontrol yang diberikan tidak lagi berada pada keadaan yang diinginkan sehingga mengakibatkan performa sistem menurun atau bahkan menyebabkan kerusakan
plant. Kesalahan aktuator dapat disebabkan oleh kebakaran, keretakan atau kesalahan pemasangan kabel. Ketika aktuator sama sekali tidak bekerja maka disebut *complete actuator fault/actuator failure*.

Kesalahan Sensor (Sensor Fault)

b.

C.

Kesalahan ini terletak di bagian sensor, yaitu bagian yang melakukan pengukuran untuk diumbanbalikkan ke kontroler. Kesalahan sensor berarti terdapat kesalahan pengukuran yang dilakukan oleh sensor yang mengakibatkan kesalahan pembacaan oleh kontroler, sehingga mengakibatkan *setpoint* berada pada keadaan yang tidak diinginkan. Kesalahan sensor lebih mudah diatasi dibandingkan kesalahan aktuator dalam penerapannya.

Kesalahan Proses (Process Fault)

Kesalahan proses adalah kesalahan yang paling kompleks karena terjadi perubahan parameter di dalam *plant*. Kesalahan ini dapat terjadi di salah satu komponen *plant* yang mengakibatkan *plant* tidak dapat beroperasi dengan baik. Banyaknya komponen di dalam *plant* akan mengakibatkan kesalahan ini menjadi kompleks dan sulit diatasi.

Beberapa literatur mengklasifikasikan kesalahan lebih jauh, seperti kesalahan aditif dan kesalahan multiplikatif. Kesalahan aditif adalah kesalahan yang bersifat menambah nilai sebenarnya dari parameter sistem berdasarkan model matematisnya, sedangkan kesalahan multiplikatif adalah kesalahan yang bersifat mengalikan (faktor) parameter sistem berdasarkan model matematikanya. terhadap Klasifikasi kesalahan seperti ini didasarkan model matematika pada keadaan sebenarnya. Klasifikasi kesalahan sangat penting untuk menentukan metode apa yang mampu mengatasinya.

Kesalahan yang muncul dalam sistem dapat dimodelkan secara matematis. Model matematika dari kesalahan jauh lebih sederhana dibanding kesalahan yang terjadi sesungguhnya pada *real plant*. Model matematika untuk kesalahan aktuator:

$$u^{J} = \varepsilon^{a} u + \alpha$$

(2.16)

dengan ε^a adalah matriks diagonal dengan elemen $0 \le \varepsilon_i^a \le 1$; i = 1,2,...m. Setiap elemen dari matriks ε_i menentukan intentitas dari kesalahan aktuator dengan mengindikasikan kesalahan aktuator. Simbol

 α adalah kesalahan aditif. Ketika ε_i^a menunjukkan aktuator bekerja dengan sepenuhnya, ketika $\alpha = 0$ menunjukkan sensor bekerja sepenuhnya. Anggap suatu sistem dengan persamaan:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

model sistem ketika terjadi kesalahan aktuator:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(\varepsilon^a \mathbf{u}(t) + \alpha(t))$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

Model matematis kesalahan ini disebut kesalahan multiplikatif dan aditif yang digunakan untuk pemodelan *Closed Loop Tranfer Function* (CLTF). Dengan metode yang sama, model kesalahan sensor:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(\varepsilon^{a}\mathbf{u}(t) + \alpha(t))$$

$$\mathbf{y}(t) = \mathbf{C}(\varepsilon^{s}\mathbf{x}(t) + \alpha_{s}(t)) \qquad (2.18)$$

(2.17)

(2.19)

dengan ε^s adalah matriks diagonal dengan elemen $0 \le \varepsilon_i^s \le 1$; i = 1,2,...m. Setiap elemen dari matriks ε^s menentukan intensitas dari kesalahan sensor dan α_s menentukan seberapa besar kesalahan sensor. Ketika kesalahan sensor hanya terletak pada satu atau tidak semua elemen sensor, persamaan (2.19) menjadi:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(\varepsilon^a \mathbf{u}(t) + \alpha(t))$$
$$\mathbf{y}(t) = \mathbf{C}\varepsilon^s \mathbf{x}(t) + \mathbf{D}_f \alpha_s(t)$$

dengan matriks \mathbf{D}_{f} adalah matriks yang menentukan dimana kesalahan sensor terjadi.

Tabel 2.1 Riashikasi Resalahan Sensoi [1]		
	$\alpha_{\rm s}(t) = 0$	$\alpha_{\rm s}(t) \neq 0$
$\varepsilon^{s} = 1$	Fault Free	Bias
$0 < \varepsilon^{s} < 1$	Loss of Sensitivity	Loss of Sensitivity
$\varepsilon^{s} = 0$	Loss of Measurement	Sensor Freezing

Tabel 2.1 Klashikasi Kesalahali Selisul 1	.1 Klasifikasi Kesalahan Sensor [1]	Fabel 2.1
---	-------------------------------------	-----------

	$\alpha(t) = 0$	$\alpha(t) \neq 0$		
$\varepsilon^a = 1$	Fault Free	Bias		
$0 < \varepsilon^a < 1$	Loss of Effectiveness	Loss of Effectiv <mark>eness</mark>		
$\varepsilon^{a} = 0$	Loss of Measurement	Actuator Block		

Tabel 2.2 Klasifikasi Kesalahan Aktuator [1]

2.9. Kontrol Toleransi Kesalahan (*Fault Tolerant Control*) [2] [4]

Secara umum sistem yang bekerja pada kondisi yang diinginkan disebut dalam keadaan kontrol nominal (*nominal control*). Kontrol nominal menjamin sistem berada pada performa yang diinginkan. Kontrol nominal dalam keadaan lup tertutup (*Closed Loop Transfer Function*) dan diasumsikan tidak terjadi kesalahan pada komponenkomponennya (kondisi ideal). Namun pada sistem nyata, dapat terjadi degradasi pada sistem, terutama pada sensor dan aktuatornya. Dalam kontrol nominal, sensor dan aktuator diasumsikan dapat bekerja sepenuhnya. Ketika sensor atau aktuator mengalami penurunan performa, perlu adanya kontrol yang mampu menanggulangi keadaan tersebut. Kontrol ini dinamakan Kontrol Toleransi Kesalahan (FTC). FTC dibagi menjadi dua yaitu; Kontrol Toleransi Kesalahan Aktif (AFTC) dan Kontrol Toleransi Kesalahan Pasif (PFTC).

Pada PFTC, kontrol nominal dibuat *robust* terhadap kesalahan sensor atau aktuator, sehingga tidak membutuhkan sistem rekonfigurasi. Kesalahan sensor atau aktuator dianggap mirip *disturbance* sehingga kontroller mampu mengatasinya. Berbeda dengan PFTC, struktur AFTC lebih rumit. AFTC memiliki sistem rekonfigurasi dan estimasi kesalahan untuk melakukan perhitungan, karena itu disebut "aktif". AFTC mempunyai nama lain "*self-repairing*", "*reconfigurable*", "*restructurable*", atau "*self-designing*". Untuk membedakan kedua metode Kontrol Toleransi Kesalahan ini dengan mudah, dapat dilihat pada Gambar 2.8.



Gambar 2.8 Skema dari Kontrol Toleransi Kesalahan (FTC)

Langkah-langkah yang diperlukan untuk mentoleransi kesalahan secara umum adalah:

- 1). Identifikasi dan deteksi kesalahan yang ada pada sistem. Mekanisme ini dinamakan diagnosis, untuk menentukan aksi selanjutnya dari FTC.
- 2). Toleransi kesalahan menggunakan informasi yang telah didapatkan dari diagnosis sistem. Toleransi ini dinamakan rekonfigurasi. Rekonfigurasi bersifat *adaption*, yaitu menyesuaikan dengan keadaan dalam sistem. Rekonfigurasi nonaktif ketika tidak ada kesalahan dalam sistem dan memunculkan aksi ketika terjadi kesalahan.

2.10. Kontrol Toleransi Kesalahan Aktif (AFTC) [1] [2] [4] [10]

Dewasa ini, AFTC lebih banyak dikembangkan daripada PFTC, hal ini dikarenakan metode yang digunakan pada AFTC lebih luas dan lebih fleksibel. AFTC memastikan sistem beradaptasi dengan cepat terhadap kesalahan yang terjadi. Sistem AFTC lebih luas lagi dibagi menjadi empat subsistem yaitu:

- 1. Fault Detection and Diagnosis (FDD)
- 2. Reconfigurable Controller
- 3. Controller Reconfiguration Mechanism
- 4. Command/Reference Governor

FDD dan *reconfigurable controller* adalah subsistem yang membedakan AFTC dengan PFTC. FDD adalah awal mula untuk mendesain AFTC. Dalam FDD terdapat beberapa tahap:

1). Fault Detection

Pada tahap ini, kesalahan dideteksi dari bagian komponen yang mengalami degradasi fault, *fault detection* juga dinamakan residual.

2). Fault Isolation

Pada tahap ini ditentukan komponen mana yang mengalami kesalahan (sensor, aktuator atau proses).

3). Fault Identification

Pada tahap ini ditentukan jenis kesalahan apakah bersifat aditif atau multiplikatif, *abrupt fault* atau *incipent fault*.

FDD yang didesain juga harus mendeteksi adanya kesalahan dengan cepat yang ada dalam sistem, hal inilah yang membuat AFTC memiliki banyak kelebihan dibanding PFTC. FDD dalam AFTC juga dapat beradaptasi dengan cepat terhadap waktu, sedangkan PFTC lebih lambat, hal ini mengakibatkan sistem di era sekarang lebih banyak menerapkan AFTC untuk meningkatkan performa.

FDD yang paling sederhana adalah *redundancy* pada *hardware* sistem. *Redundancy* ini dapat berupa sensor atau komponen lain. FDD yang menjadi topik riset populer saat ini adalah FDD menggunakan model dari *plant*. Secara umum, berdasarkan model yang digunakan terdapat dua metode FDD; *model-based* dan *model-free*.

1). Model-free FDD

FDD ini tidak menggunakan model atau data dari *plant* untuk menentukan kesalahan sensor atau aktuator pada sistem, contoh nyata dari model ini penerapan beberapa sensor untuk mengatasi kesalahan sensor yang ada pada *plant*, dapat dilihat pada Gambar 2.9



Gambar 2.9 Hardware Redundancy pada Sistem [2] [4]

Beberapa sensor diletakkan *parallel* terhadap sensor utama, perbedaan kerja dari dua atau lebih sensor memunculkan peringatan/alarm untuk menunjukkan telah terjadi kesalahan sensor. Sistem ini memungkinkan penggantian sensor yang rusak dengan cepat. Beberapa metode lain dalam model-free adalah *limit checking* dan *frequency spectral analysis*.

Model-based FDD

2).

Model ini dikembangkan sekitar tahun 1980-an sebagai metode baru FTC. Inti dari konsep *model-based* FDD adalah membandingkan model real-*plant* dengan model tiruan yang ada di komputer/algoritma. Perbedaan dari kedua model disebut dengan *residual*. Dari *residual* ini ditentukan apakah terjadi kesalahan sensor/aktuator atau tidak pada sistem. Konsep ini memiliki banyak kelebihan dibandingkan *model-free*, salah satunya adalah penghematan komponen yang akan dibeli seperti halnya *hardware redundancy*. Skema model-based dapat dilihat pada Gambar 2.10



Gambar 2.10 Skema Model-Based Fault Diagnosis [2] [4]

Pada *model-based*, terdapat tiga metode untuk menentukan *residual generation*, yaitu:

a. Observer-based FDD

Metode *observer* menggunakan estimasi parameter dari keluaran. *Residual* didapatkan dengan mengestimasi error antara sinyal yang terukur dan sinyal yang terestimasi. Kemudahan struktur *observer* menjadi metode FDD yang paling mudah dikembangkan.

b. Parity relation based FDD

Pada pendekatan ini, sinyal residual didapatkan dari pengukuran konsisten masukan dan keluaran sistem pada rentang waktu tertentu.

c. Parameter estimation

Metode ini adalah pengembangan dari sistem *observer*. Kesalahan direpresentasikan sebagai parameter dalam sistem, kemudian parameter tersebut diestimasi. Hasil dari estimasi digunakan untuk mengkompensasi kesalahan tadi.

Langkah selanjutnya setelah FDD adalah reconfigurable controller. Metode reconfigurable controller digunakan untuk memberikan sinyal kontrol yang baru pada sistem, sehingga sinyal bebas kontrol tersebut telah dari kesalahan sensor/aktuator. *Reconfigurable controller* tidak harus ada pada sistem AFTC, metode ini dikembangkan pada sistem *plant* tiga tangki untuk mengatasi sensor dan aktuator. Namun pada pengembangan riset selanjutnya, metode kompensasi lebih sederhana.



Gambar 2.11 Struktur Lengkap AFTC [2]

2.11. Observer [3]

Observer adalah metode untuk mengestimasi state dari suatu sistem. Dalam sistem kontrol, khususnya sistem lup tertutup, tidak semua state sistem dapat diukur, sehingga diperlukan metode untuk mengestimasi state tersebut. Tinjaulah suatu sistem:

 $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$

 $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$

Estimasi menggunakan observer, memiliki persamaan:

$$\dot{\tilde{\mathbf{x}}}(t) = \mathbf{A}\tilde{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{K}_{e}(\mathbf{y}(t) - \mathbf{C}\tilde{\mathbf{x}}(t))$$

(2.20)

Persamaan (2.20) memiliki residual dari hasil pengurangan keluaran sebenarnya dikurangi dengan keluaran estimasi. *Residual* ini dikalikan dengan gain K_e yaitu observer gain. Struktur dari observer dapat dilihat pada Gambar 2.10.

2.12. Proportional-Proportional Integral Observer [4] [11] [12]

Metode estimasi state ini dapat meningkatkan efisiensi dalam suatu sistem, misalnya menggantikan fungsi sensor yang mahal untuk menggantikan kesalahan sensor pada plant. Terdapat beberapa jenis observer, berdasarkan derivatifnya yaitu proporsional, proporsional integral, dan proporsional integral derivatif. Perbedaan dari ketiga observer tersebut adalah penentuan observer gainnya. Observer proporsional mempunyai satu observer gain, proporsional integral mempunyai dua observer gain, dan proporsional integral derivatif observer mempunyai tiga observer. Penambahan integral disini dimaksudkan untuk menghilangkan error estimasi.



Gambar 2.12 Blok Diagram Observer [3]

Persamaan *Proportional-Proportional Integral Observer* (PPIO) dapat dilihat pada persamaan (2.21)

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \hat{\mathbf{p}}(t) + \mathbf{P}\mathbf{e}_{y}(t)$$
$$\dot{\hat{\mathbf{p}}} = \mathbf{L}(\mathbf{e}_{y}(t) + \dot{\mathbf{e}}_{y}(t))$$
$$\mathbf{e}_{y}(t) = \mathbf{y}(t) - \hat{\mathbf{y}}(t)$$
$$\hat{\mathbf{y}}(t) = \mathbf{C}\hat{\mathbf{x}}(t)$$

Dari persamaan (2.21) terlihat observer PPIO memiliki dua gain yaitu P dan L. Gain P adalah gain proportional, gain L adalah gain integral. Penambahan gain integral disini dimaksudkan untuk mempercepat estimasi yang didapatkan. Penambahan gain integral dalam PPIO juga dinamakan Fast Fault Estimation (FFE). Respon transion dari estimasi yang didapatkan sangat cepat dan error dinamika yang didapatkan selalu menuju nol. Struktur observer PPIO dapat dilihat pada Gambar 2.13.



(2.21)

2.13. Estimasi dan Kompensasi [1] [4]

Secara konvensional FTC memastikan kontrol nominal dengan baik dengan adanya FDD dan controller reconfiguration. Pada subbab 2.8 dijelaskan bahwa controller reconfiguration adalah mekanisme konvensional dalam kompensasi kesalahan sensor atau aktuator. Karena FDD harus bereaksi cepat terhadap kesalahan yang terjadi, terdapat konsep yang lebih sederhana dalam AFTC, yaitu estimasi dan estimasi kompensasi. Konsep dari dan kompensasi adalah menghilangkan efek dari kesalahan sensor dan aktuator dengan mengurangi sinyal hasil estimasi dengan sinyal sebenarnya yang telah terjadi kesalahan, sehingga saling meniadakan. Estimasi dan kompensasi telah dikembangkan bertahun-tahun untuk membuat struktur AFTC menjadi lebih sederhana. Secara umum konsep kompensasi yang dimaksud adalah:

 $u(t) = u_n(t) + u_{add}(t)$ $u_{add}(t) = K_f \hat{f}(t); f(t) \neq 0$ $u_{add}(t) = 0; f(t) = 0$

(2.22)

Sinyal u(t) adalah sinyal kontrol. $u_n(t)$ adalah sinyal kontrol ketika sistem tidak terjadi kesalahan. $u_{add}(t)$ adalah sinyal penambahan dari hasil estimasi kesalahan yang didapatkan dari obsever FTC. Ketika terjadi kesalahan $u_{add}(t)$ bekerja dengan memberikan tambahan sinyal kontrol yaitu $K_{ffs}(t)$. Ketika tidak terjadi kesalahan, $u_{add}(t)=0$. Penambahan ini mengakibatkan $u_n(t) + u_{add}(t) = 0$ kembali ke sinyal kontrol yang diinginkan. Estimasi adalah hal yang sangat penting dalam metode AFTC ini, karena besarnya estimasi akan ditambahkan ke sinyal kontrol sebenarnya, maka estimasi kesalahan sensor atau aktuator harus akurat.

2.14. Linear Matrix Inequality (LMI) [13]

LMI adalah penyederhanaan metode penyelesaian pertidaksamaan. Dalam sistem kontrol, LMI sangat berguna untuk menyelesaikan masalah kestabilan. Masalah kestabilan yang sering diselesaikan dalam LMI adalah kestabilan Lyapunoy. Bentuk umum dari LMI adalah:

$$\mathbf{F}(x) = \mathbf{F}_0 + \sum_{i=1}^m x_i \mathbf{F}_i > 0$$

(2.23)

dimana $x \in R^m$ adalah variabel dan \mathbf{F}_0 , \mathbf{F}_i adalah konstanta matriks simetris. LMI berikut adalah dasar dari analisa kestabilan Lyapunov:

 $\mathbf{P}\mathbf{A} + \mathbf{A}^T \mathbf{P} < 0 \tag{2.24}$

dengan variabel $\mathbf{P} \in R^{nxn}$, syarat $\mathbf{P} > 0$. Lyapunov menunjukkan secara analitik bahwa persamaan ini dapat diselesaikan dengan memilih $\mathbf{Q} = \mathbf{Q}^{T} > 0$ dan kemudian menyelesaikan persamaan linear $\mathbf{A}^{T}\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{Q}$ untuk mendapatkan matriks definit positif \mathbf{P} . Untuk perhitungan beberapa pertidaksamaan, mudah diselesaikan. Namun untuk orde tinggi dan syarat yang banyak perhitungan sangat sulit dilakukan. Karena itulah dalam LMI dikenal istilah Schur Complement. Persamaan Schur Complement mengubah bentuk pertidaksamaan matriks menjadi bentuk LMI yaitu:

dengan $\mathbf{Q}(\mathbf{x}) = \mathbf{Q}(\mathbf{x})^{\mathrm{T}}$ dan $\mathbf{R}(\mathbf{x}) = \mathbf{R}(\mathbf{x})^{\mathrm{T}}$ adalah ekuiyalen dan dapat dinyatakan dengan

$$\mathbf{R}(x) > 0$$

$$\mathbf{Q}(x) - \mathbf{S}(x)\mathbf{R}(x)^{-1}\mathbf{S}(x)^{T} > 0$$
(2.26)

beberapa pertidaksamaan dapat digabungkan menjadi satu. Misalkan terdapat persamaan:

$$\mathbf{F}_{0} = \begin{bmatrix} F_{01} & 0 & 0 & 0 \\ 0 & F_{02} & 0 & 0 \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & F_{0p} \end{bmatrix} = diag(F_{01}, \dots F_{0p})$$
$$\mathbf{G}_{i} = diag(G_{i1}, \dots G_{ip})$$
$$\mathbf{H}_{i} = diag(H_{i1}, \dots H_{ip})$$
$$\mathbf{V}_{i} = diag(V_{i1}, \dots V_{ip})$$

Persamaan \mathbf{F}_{i} , \mathbf{G}_{i} , \mathbf{H}_{i} dan \mathbf{V}_{i} dapat digabung dalam bentuk:

$$\mathbf{F}_{big}(V_1,\ldots,V_n) \coloneqq \mathbf{F}_0 + \sum_{i=1}^n \mathbf{G}_i \mathbf{V}_i \mathbf{H}_i > 0$$

2.15. LMI Region dan Pole Placement LMI Region [14]

Kestabilan adalah masalah yang paling banyak dibahas dalam sistem kontrol, hingga berbagai macam metode diterapkan memenuhi kriteria performa yang diinginkan. Pada sistem yang harus memiliki respon cepat seperti pendulum, overhoot harus seminimal mungkin dan respon transien secepat mungkin. Kriteria overshoot dan respon transien dipengaruhi oleh letak pole-pole pada lup tertutup. Terdapat beberapa cara untuk meletakkan pole-pole pada daerah yang diinginkan, salah satunya adalah pole placement. Pole placement menggunakan LMI adalah salah satu metode yang cukup sederhana untuk menentukan gain-gain matrik kestabilan. Hal ini dikarenakan pada LMI, dapat ditambahkan syarat kestabilan, seperti batasan input-output, atau kriteria lain. Daerah pole-pole pada LMI didefinisikan oleh Gambar 2.14.

(2.27)



Gambar 2.14 Daerah pada LMI Region (Conic Sector+Half Plane) [14]

D adalah daerah pada bidang kompleks yang dinamakan LMI region, θ adalah sudut batas dimana pole-pole harus terdapat didalamnya, x adalah batas sumbu minimal pole harus berada di dalamnya. Daerah ini ada jika ada matrik simetris L dan matrik M sehingga:

$$\mathbf{D} = \left\{ z \in C : f_d(z) < 0 \right\}$$
(2.28)

dengan $f_d(z) = L + zM + zM^T$ dan $f_d(z)$ dinamakan fungsi karakteristik dari daerah bidang kompleks **D**. Contoh dari LMI region adalah:

- 1. *Half Plane* (setengah bidang) $Re(z) = -\alpha, f_d(z) = z + z + 2\alpha < 0$ (2.29)
- 2. Disk (lingkaran) berpusat di (-q, 0) dengan radius r

$$f_d(z) = \begin{bmatrix} -r & q+z \\ z+\overline{z} & -r \end{bmatrix} < 0$$
(2.30)

3. *Conic Sector* (kerucut) 2θ

$$f_d(z) = \begin{bmatrix} \sin\theta(z+\overline{z}) & \cos\theta(z+\overline{z}) \\ \cos\theta(z+\overline{z})^T & \sin\theta(z+\overline{z}) \end{bmatrix} < 0$$
(2.31)

Daerah D sebagai daerah *pole-pole* yang diinginkan, memiliki karakteristik berbeda untuk setiap sistem. Gambar 2.14 menggambarkan daerah D dalam bentuk kerucut dan *half plane* yang digabung sehingga menghasilkan bentuk trapesium. Daerah pada masing-masing bidang kompleks dapat dikombinasikan menghasilkan D yang berbeda. Sistem lup tertutup dengan persamaan dinamika kestabilan A+BK, memiliki *pole-placement* dalam bentuk LMI sebagai berikut:

$\mathbf{P} > 0$

$$(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} + \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T} + 2\gamma\mathbf{P} < 0$$

 $\begin{bmatrix} \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} + \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}] \\ \cos\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} - \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}]^{T} \\ \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} + \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}] \end{bmatrix} < 0$ (2.32)

Sistem lup tertutup A+BK stabil jika dan hanya jika terdapat matrik **P** yang memenuhi LMI pertidaksamaan (2.32). Karena terdapat dua variabel yaitu **P** dan **K**, maka perlu membentuk LMI pertidaksamaan (2.32) menjadi bentuk linear dengan membuat variabel baru, yaitu **Y** = **KP**, sehingga persamaan (2.33) menjadi:

$\mathbf{P} > 0$

$$\mathbf{AP} + \mathbf{PA}^{T} + \mathbf{BY} + \mathbf{Y}^{T} \mathbf{B}^{T} + 2\gamma \mathbf{P} < 0$$

$$\sin \theta [\mathbf{AP} + \mathbf{PB}^{T} + \mathbf{BY} + \mathbf{Y}^{T} \mathbf{B}^{T}]$$

$$\cos \theta [\mathbf{AP} - \mathbf{PA}^{T} + \mathbf{BY} - \mathbf{Y}^{T} \mathbf{B}^{T}]^{T}$$

 $\frac{\cos \theta [\mathbf{A}\mathbf{P} - \mathbf{P}\mathbf{A}^{T} + \mathbf{B}\mathbf{Y} - \mathbf{Y}^{T}\mathbf{B}^{T}]}{\sin \theta [\mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^{T} + \mathbf{B}\mathbf{Y} + \mathbf{Y}^{T}\mathbf{B}^{T}]} < 0$ (2.33)

2.16. Batasan Input-Output [8]

Dalam implementasi nyata, ketidakpastian parameter serta gangguan dari luar dapat diberikan pada sistem. Selain itu, dalam *real plant, state* yang terukur bisa terbatas dan sinyal kontrol yang diberikan juga terbatas. Agar dapat diimplementasikan dalam sistem nyata, besaran-besaran penting seperti sinyal kontrol perlu dibatasi, agar tidak terjadi guncangan atau kerusakan. Suatu *plant* yang memiliki:

 $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$

 $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$

Untuk memberikan batasan input-output dimodifikasi menjadi

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{z}_{1}(t) = \mathbf{C}_{z1}\mathbf{x}(t)$$
$$\mathbf{z}_{2}(t) = \mathbf{C}_{z2}\mathbf{x}(t)$$
$$\mathbf{y}(t) = \mathbf{C}_{y}\mathbf{x}(t)$$

dengan $\mathbf{u}(\mathbf{t})$ adalah sinyal kontrol, $\mathbf{z}_1(t)$ adalah keluaran performa, $\mathbf{z}_2(t)$ adalah keluaran yang dibatasi, dan $\mathbf{y}(t)$ adalah pengukuran untuk kontroler. Jika $\mathbf{C}_{\mathbf{y}}$ merupakan matriks identitas, maka $\mathbf{y}(t) = \mathbf{x}(t)$. Karena sinyal kontrol yang digunakan adalah $\mathbf{u}(t) = -\mathbf{K}\mathbf{y}(t) = -\mathbf{K}\mathbf{x}(t)$, maka persamaan (2.20) menjadi:

$$\dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t)$$
$$\mathbf{z}_{1}(t) = \mathbf{C}_{z1}\mathbf{x}(t)$$
$$\mathbf{z}_{2}(t) = \mathbf{C}_{z2}\mathbf{x}(t)$$

(2.35)

(2.34)

Sehingga perumusan untuk batasan pada sinyal kontrol **u**(t) dapat dinyatakan sebagai berikut:

 $\left\|\mathbf{u}(t)\right\| \leq u_{\max}$

 $\left\| -\mathbf{K}\mathbf{x}(t) \right\| \le \left\| \mathbf{u}(t) \right\| \le u_{\max}$

Sedangkan perumusan untuk batasan pada keluaran yang dibatasi $\mathbf{z}_2(t)$ dapat dinyatakan sebagai berikut:

$$\begin{aligned} \left\| \mathbf{Z}(t) \right\| &\leq \mathbf{Z}_{2\max} \\ \left\| \mathbf{C}_{z2} \mathbf{x}(t) \right\| &\leq \left\| \mathbf{Z}_{2}(t) \right\| \leq z_{2\max} \end{aligned}$$

2.17. Performa H_{∞} [8]

Pengembangan metode kontrol *robust* telah menjadi fokus utama dalam dua dekade ini khususnya dalam komunitas kotnrol. Ketahanan suatu sistem kontrol terhadap ganggaun (*disturbance*) dan ketida kpastian (*uncertainty*) selalu menjadi topik utama dalam pembahasan masalah *feedback control*. Masalah *feedback* tidak akan terlalu menarik pada kebanyakan sistem kontrol jika didalamnya tidak terdapat gangguan dan ketidakpastian.

Pemberian kontrol pada suatu sistem bertujuan untuk membuat sistem tersebut robust terhadap gangguan sesuai dengan keinginan. Desain kontrol H_{∞} atau dengan H_2 menjamin ketahanan suatu sistem tetapi pada umumnya desain kontrol dengan menggunakan H_{∞} lebih banyak diminati dibanding desain kontrol dengan H_2 karena kontrol yang dihasilkan lebih robust. Pengaruh buruk gangguan terhadap keluaran performa dapat diketahui dengan menghitung ∞ -norm dari fungsi alih sistem. ∞ -norm dari suatu fungsi alih didefinisikan sebagai

$$G(s) = \sup_{\omega} G(j\omega)$$

(2.38)

Dari persamaan $(2.21) \infty$ -norm dari suatu fungsi alih adalah nilai maksimal dari magnitude respon frekuensi. Jika G(s) menyatakan fungsi alih keluaran performa terhadap gangguan, dan ∞ -norm dari G(s) bernilai kecil, maka dapat dikatakan bahwa gangguan yang diberikan teredam sesuai besarnya ∞ -norm.

Gambar 2.15 menunjukkan diagram blok sistem dengan dua masukan dan dua keluaran $\mathbf{w}(t)$ adalah gangguan dari luar, $\mathbf{u}(t)$ adalah sinyal kontrol, $\mathbf{z}(t)$ adalah keluaran performa, dan $\mathbf{y}(t)$ adalah pengukuran untuk kontroler. Dalam bentuk state space, sistem pada Gambar 2.15 dapat ditulis sebagai berikut:

(2.36)

(2.37)



Sistem pada Gambar 2.15 ditulis dalam bentuk persamaan:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_{w}\mathbf{w}(t) + \mathbf{B}_{u}\mathbf{u}(t)$$

$$\mathbf{z}(t) = \mathbf{C}_{z}\mathbf{x}(t)$$

$$\mathbf{y}(t) = \mathbf{C}_{x}\mathbf{x}(t)$$
(2.39)

Jika sinyal kontrol yang digunakan adalah $\mathbf{u}(t) = -\mathbf{K}\mathbf{y}(t)$, maka sistem dapat disederhanakan menjadi sistem dengan satu masukan yaitu $\mathbf{w}(t)$ dan satu keluaran yaitu $\mathbf{z}(t)$ dengan bentuk state space.

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{w}(t)$$

$$\mathbf{z}(t) = \overline{\mathbf{C}}\mathbf{x}(t)$$

(2.40)

dan dengan

$$\overline{\mathbf{A}} = (\mathbf{A} - \mathbf{B}_u \mathbf{K} \mathbf{C}_y),$$
$$\overline{\mathbf{B}} = \mathbf{B}_w,$$
$$\overline{\mathbf{C}} = \mathbf{C},$$

Gambar 2.16 menunjukkan diagram blok sistem sesuai dengan persamaan (2.40). Jika $T_{zw}(s)$ menunjukkan fungsi alih Z(s) terhadap W(s), maka $T_{zw}(s)$ dapat dihitung sebagai berikut:

 $T_{zw}(s) = \overline{\mathbf{C}}(s\mathbf{I} - \overline{\mathbf{A}})^{-1}\overline{\mathbf{B}}$ $T_{zw}(s) = \mathbf{C}_{z}(s\mathbf{I} - \mathbf{A} + \mathbf{B}_{u}\mathbf{K}\mathbf{C}_{y})^{-1}\mathbf{B}_{w}$

(2.41)



Gambar 2.16 Diagram Blok Sistem Persamaan (2.40) [8]

Secara sederhana sistem pada Gambar 2.16 dapat ditulis dengan mengubah kedalam bentuk transformasi Laplace seperti pada Gambar 2.17.

$$W(s) \longrightarrow T_{ZW}(s) \longrightarrow Z(s)$$

Gambar 2.17 Penyederhanaan Sistem dalam Bentuk LFT [8]

hubungan antara masukan W(s) dan keluaran Z(s) dapat dinyatakan dengan persamaan:

$$\frac{Z(s)}{W(s)} = T_{zw}(s) \tag{2.42}$$

Sistem dalam bentuk LFT pada Gambar 2.17 didesain agar masukan yang berupa gangguan kurang dari pelemahan/atenuasi tertentu. Performa H_{∞} dicari menggunakan bentuk L_2 -gain yang dituliskan dalam persamaan:

$$\left\|\mathbf{w}(t)\right\|_{2} = \sqrt{\int_{0}^{\infty} \mathbf{w}(t)^{T} \mathbf{w}(t) dt}$$

(2.43)

maka L_2 -gain atau perbandingan L_2 -norm $\mathbf{z}(t)$ terhadap L_2 -norm $\mathbf{w}(t)$ adalah:

$$\frac{\left\|\mathbf{z}(t)\right\|_{2}}{\left\|\mathbf{w}(t)\right\|_{2}} = \frac{\sqrt{\int_{0}^{\infty} \mathbf{z}(t)^{T} \mathbf{z}(t) dt}}{\sqrt{\int_{0}^{\infty} \mathbf{w}(t)^{T} \mathbf{w}(t) dt}}$$
(2.44)

 ∞ -norm dari fungsi alih $T_{zw}(s)$ dapat dihitung dari maksimal L_2 -gain $\mathbf{z}(t)$ terhadap $\mathbf{w}(t)$. Sesuai definisi tersebut, maka ∞ -norm dari fungsi alih $T_{zw}(s)$ dapat dihitung dengan

$$\left\|T_{zw}(s)\right\|_{\infty} = \sup_{\boldsymbol{\omega}} \left|T_{zw}(j\boldsymbol{\omega})\right| = \sup_{\left\|\boldsymbol{w}(t)\right\|_{2} \neq 0} \frac{\left\|\mathbf{z}(t)\right\|_{2}}{\left\|\mathbf{w}(t)\right\|_{2}}$$
(2.45)

Jika tingkat pelemahan maksimal yang diinginkan adalah kurang dari γ maka performa desain yang digunakan dalam kontrol *robust* dapat dinyatakan dalam persamaan (2.46)

$$\|T_{zw}(s)\|_{\infty} = \gamma^* < \gamma$$

$$\|w(t)\|_{2} = \gamma^* < \gamma$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.46)$$

$$(2.4$$

BAB III PERANCANGAN SISTEM

Pada Bab ini akan dibahas perancangan kontrol nominal (kontrol sistem pendulum kereta tanpa kesalahan sensor dan aktuator). Kontrol nominal dirancang menggunakan metode LMI *pole placement* dan batasan *input output*. Pencarian *gain* kontrol nominal menggunakan metode LMI *pole placement* yang menjamin *pole-pole* sistem lup tertutup berada pada daerah kestabilan yang diinginkan. *Observer* kesalahan sensor dan aktuator dirancang menggunakan metode PPIO (*Proportional Proportional Integral Observer*) dengan pencarian *gain* berbasis LMI yang menjamin kestabilan Lyapunov.

4.1. Linearisasi Model Matematika Sistem Pendulum Kereta

Untuk mendesain kontrol fuzzy T-S, model matematika pada persamaan (2.4) dilinerisasi di titik daerah kerjanya. Titik ekuilibrium sistem harus diketahui terlebih dahulu. Persamaan (2.4) ditulis kembali:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + h(\mathbf{x}(t), \mathbf{u}(t))$$
(3.1)

dengan

$$f(\mathbf{x}(t)) = \begin{bmatrix} f_1(\mathbf{x}(t)) \\ f_2(\mathbf{x}(t)) \\ f_3(\mathbf{x}(t)) \\ f_4(\mathbf{x}(t)) \end{bmatrix} = \begin{bmatrix} u \\ a(u - T_c - \mu x_4^2 \sin x_2) + l \cos x_2(\mu g \sin x_2 - f_p x_4) \\ J + \mu l \sin^2 x_2 \\ l \cos x_2(u - T_c - \mu x_4^2 \sin x_2) + \mu g \sin x_2 - f_p x_4 \\ J + \mu l \sin^2 x_2 \end{bmatrix}$$

$$h(\mathbf{x}(t)) = \begin{bmatrix} h_1(\mathbf{x}(t), \mathbf{u}(t)) \\ h_2(\mathbf{x}(t), \mathbf{u}(t)) \\ h_3(\mathbf{x}(t), \mathbf{u}(t)) \\ h_4(\mathbf{x}(t), \mathbf{u}(t)) \\ h_4(\mathbf{x}(t), \mathbf{u}(t)) \end{bmatrix} = \begin{bmatrix} 0 & 0 & \frac{au}{J + \mu l \sin^2 x_2} & \frac{u l \cos x_2}{J + \mu l \sin^2 x_2} \end{bmatrix}$$

Saat sistem berada pada titik ekuilibrium, laju perubahan *state* sistem bernilai nol. Jika sistem diberi kondisi awal yang berada pada titik ekuilibrium, maka *state* sistem akan selalu berada pada titik tersebut. Dari persamaan (3.1), titik ekuilibrium sistem pendulum kereta dinyatakan:

$$\mathbf{x}_{e}(t) = \begin{bmatrix} x_{1e} & x_{2e} & x_{3e} & x_{4e} \end{bmatrix}^{T} \operatorname{dan} \mathbf{u}_{e}(t)$$
(3.2)

nilai titik ekuilibrium persamaan (3.2) disubtitusikan persamaan (3.1) sehingga dinyatakan:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}_e(t)) + h(\mathbf{x}_e(t), \mathbf{u}_e(t))$$

$$0 = f(\mathbf{x}_e(t)) + h(\mathbf{x}_e(t), \mathbf{u}_e(t))$$

solusi persamaan (3.3) adalah:

$$\mathbf{x}_e(t) = \begin{bmatrix} c & n\pi & 0 & 0 \end{bmatrix}^I \, \mathrm{dan} \, \mathbf{u}_e(t) = 0 \tag{3.4}$$

dengan *n* bilangan bulat dan *c* adalah real.

Kontrol *tracking* dirancang agar sistem mampu mengikuti trayektori yang diinginkan, namun tetap mempertahankan sudut pendulum pada 0 radian. Sudut 0 radian memenuhi persamaan (3.4) sehingga sistem pendulum kereta dapat dilinearisasi pada titik ini.

Persamaan (3.4) menunjukkan linearisasi lokal hanya dapat divariasi pada x_1 dan x_2 . Pada tugas akhir ini nilai x_1 dipilih nol, sedangkan nilai x_2 divariasi pada 3 titik yaitu 0, ±0,2616 dan ±0,5232 radian. Model sistem linear pada $x^* = [0 \ 0 \ 0 \ 0]^T$ dan $u^* = 0$, $x^* = [0 \pm 0,2616 \ 0 \ 0]^T$ dan $u^* = 0$, serta $x^* = [0 \pm 0,5232 \ 0 \ 0]^T$ dan $u^* = 0$ digambarkan dalam 3 subsistem sesuai persamaan (3.5), persamaan (3.6) dan persamaan (3.7). Penurunan perhitungan masing-masing elemen matriks dapat dilihat pada lampiran A.2

$$\dot{\mathbf{x}}(t) = \mathbf{A}_1 \mathbf{x}(t) + \mathbf{B}_1 \mathbf{u}(t)$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}_2 \mathbf{x}(t) + \mathbf{B}_2 \mathbf{u}(t)$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}_3 \mathbf{x}(t) + \mathbf{B}_3 \mathbf{u}(t)$$
(3.5)
(3.6)
(3.7)

dengan

$$\mathbf{A}_{1} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2526 & 0 & -0,0001 \\ 0 & 15,0421 & 0 & -0,0079 \end{bmatrix}, \mathbf{B}_{1} = \begin{bmatrix} 0 \\ 0 \\ 0,8272 \\ 1,2370 \end{bmatrix}$$
$$\mathbf{A}_{2} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2176 & 0 & -0,0001 \\ 0 & 14,4555 & 0 & -0,0079 \end{bmatrix}, \mathbf{B}_{2} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0,8258 \\ 1,1929 \end{bmatrix}$$
$$\mathbf{A}_{3} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0,1232 & 0 & -0,0001 \\ 0 & 0,1232 & 0 & -0,0001 \\ 0 & 12,7812 & 0 & -0,0079 \end{bmatrix}, \mathbf{B}_{3} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0,8219 \\ 1,0647 \end{bmatrix}$$

Model sistem pada persamaan (3.5), (3.6) dan (3.7) adalah subsistem untuk masing-masing titik kerja $x_2 = 0, \pm 0,2616$ dan $\pm 0,5232$ rad.

4.2. Pemodelan Fuzzy Takagi-Sugeno (T-S)

Fuzzy T-S dibangun dari hasil linearisasi lokal sistem pendulum kereta pada Subbab 3.1. Sistem pendulum kereta dilinearisasi pada tiga titik kerja, sehingga model *fuzzy* T-S yang digunakan memiliki tiga aturan dan satu premis yaitu sudut pendulum. Dari subsistem persamaan (3.5), (3.6) dan (3.7), model *fuzzy* T-S dapat dibentuk dengan aturan *plant* sebagai berikut:

Aturan model *plant* ke-1 If $x_2(t)$ is M_1 (sekitar 0 rad) Then $\dot{\mathbf{x}}(t) = \mathbf{A}_1 \mathbf{x}(t) + \mathbf{B}_1 \mathbf{u}(t)$ $\mathbf{y}(t) = \mathbf{C}_1 \mathbf{x}(t)$ $\mathbf{z}(t) = \mathbf{C}_{-1} \mathbf{x}(t)$

Aturan model *plant* ke-2 If $x_2(t)$ is M₂ (sekitar ±0,2616 rad) Then $\dot{\mathbf{x}}(t) = \mathbf{A}_2 \mathbf{x}(t) + \mathbf{B}_2 \mathbf{u}(t)$ $\mathbf{y}(t) = \mathbf{C}_2 \mathbf{x}(t)$ $\mathbf{z}(t) = \mathbf{C}_{z,2} \mathbf{x}(t)$

Aturan model *plant* ke-3 If $x_2(t)$ is M₃ (sekitar ±0,5232 rad) Then $\dot{\mathbf{x}}(t) = \mathbf{A}_3 \mathbf{x}(t) + \mathbf{B}_3 \mathbf{u}(t)$

 $\mathbf{y}(t) = \mathbf{C}_3 \mathbf{x}(t)$ $\mathbf{z}(t) = \mathbf{C}_3 \mathbf{x}(t)$

dengan matriks keluaran, matriks transmisi langsung, $C_{z,1}$ dan $C_{z,2}$ sebagai berikut:

$$\mathbf{C}_{1} = \mathbf{C}_{2} = \mathbf{C}_{3} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$
$$\mathbf{C}_{z,1} = \mathbf{C}_{z,2} = \mathbf{C}_{z,3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Sesuai dengan konsep *Parallel Distributed Compensation* (PDC) aturan kontroler disusun sebagai berikut:

Aturan kontroler ke-1 If $x_2(t)$ is M_1 (sekitar 0 rad) Then $\mathbf{u}(t) = \mathbf{K}_{n1}\mathbf{x}(t)$

Aturan kontroler ke-2 If $x_2(t)$ is M₂ (sekitar ±0,2616 rad) Then $\mathbf{u}(t) = \mathbf{K}_{n2}\mathbf{x}(t)$

Aturan kontroler ke-3 (()) If $x_2(t)$ is M₃ (sekitar ±0,5232 rad) Then $\mathbf{u}(t) = \mathbf{K}_{n3}\mathbf{x}(t)$

Fungsi keanggotaan M_1 dipilih fungsi segitiga seperti pada Gambar 3.1. Begitu juga fungsi keanggotaan M_2 dan M_3 . Fungsi keanggotaan M_1 , M_2 dan M_3 adalah fungsi dari sudut pendulum. Persamaan matematis untuk masing-masing fungsi keanggotaan ditunjukkan persamaan (3.8), (3.9) dan (3.10). Inferensi *fuzzy* menggunakan penghubung AND dan metode defuzzifikasi yang digunakan adalah rata-rata bobot. Bentuk keseluruhan model *fuzzy* T-S ditunjukkan oleh persamaan (3.11).



$$\mathbf{y}(t) = \frac{\sum_{j=1}^{r} \left[w_{j}(x) \{ \mathbf{C}_{j} \mathbf{x}(t) \} \right]}{\sum_{j=1}^{r} w_{j}(x)} = \sum_{j=1}^{r} h_{j}(x) \{ \mathbf{C}_{j} \mathbf{x}(t) \}$$
$$\mathbf{z}(t) = \frac{\sum_{j=1}^{r} \left[w_{j}(x) \{ \mathbf{C}_{z,j} \mathbf{x}(t) \} \right]}{\sum_{j=1}^{r} w_{j}(x)} = \sum_{j=1}^{r} h_{j}(x) \{ \mathbf{C}_{z,j} \mathbf{x}(t) \}$$

dengan,

$$w_{j}(x) = \prod_{k=1}^{p} M_{jk}(x_{k})$$
$$h_{j}(x) = \frac{w_{j}(x)}{\sum_{j=1}^{r} w_{j}(x)}$$
$$0 \le h_{j}(x) \le 1$$
$$\sum_{j=1}^{r} h_{j}(x) = 1$$

Sedangkan kontroler Fuzzy menurut aturan PDC dapat ditulis dengan:

$$\mathbf{u}(t) = \sum_{j=1}^{r} h_j(\mathbf{x}) \mathbf{K}_{n,j} \mathbf{x}(t)$$

Penjabaran sinyal kontrol menurut fungsi keanggotaan setiap subsistem akan menghasilkan persamaan (3.11)

$$\mathbf{u}(t) = M_1(x_2(t))\mathbf{K}_{n,1}\mathbf{x}(t) + M_2(x_2(t))\mathbf{K}_{n,2}\mathbf{x}(t) + M_1(x_2(t))\mathbf{K}_{n,1}\mathbf{x}(t)$$
(3.11)

dengan $\mathbf{K}_{n,j}$ (j = 1,2,3) adalah kontroler gain state feedback yang akan dicari menggunakan LMI pole placement.

4.3. Perancangan Kontroler Fuzzy Pole Placement dengan Batasan Input-Output

Pada subbab ini akan dijelaskan desain kontroler untuk sistem lup tertutup sistem pendulum kereta. Desain kontroler ini disebut juga kontrol nominal (kontrol dimana kesalahan sensor dan aktuator tidak terjadi). Kontroler memastikan *pole-pole* sistem lup tertutup dari sistem pendulum kereta berada di sebelah kiri sumbu imajiner dan daerah tertentu (*region*). Untuk menempatkan *pole-pole* pada daerah tertentu digunakan LMI *region*. Untuk menjamin sistem lup tertutup menghasilkan sinyal kontrol yang tidak terlalu besar (untuk keperluan implementasi) digunakan batasan *input-output*. Kombinasi *constraint* dari LMI region dan batasan *input-output* menghasilkan desain LMI dari kontroler secara keseluruhan. Desain kontroler secara keseluruhan dapat dilihat pada Gambar 3.2.





4.3.1. Perancangan Kontroler berbasis Model Referensi

Kontroler dirancang mengikuti trayektori yang diinginkan. Pada tugas akhir ini dirancang kontroler berbasis kompensator. Kompensator dirancang berdasarkan sinyal referensi dan gangguan. Sesuai dengan persamaan (2.14) dan penjelasan Subbab (2.7) persamaan augmentasi ini dimodifikasi dalam bentuk *fuzzy* T-S untuk tiga subsistem:

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_{c}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{j} & 0 \\ -\mathbf{B}_{c}\mathbf{C}_{j} & \mathbf{A}_{c} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_{c}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{j} \\ 0 \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} 0 \\ \mathbf{B}_{c} \end{bmatrix} \mathbf{r}(t)$$

$$j = 1, 2, 3$$
(3.12)

sehingga sinyal kontrol yang dihasilkan adalah hasil augmentasi pencarian gain kompensator dan gain state feedback.

$$\mathbf{u}(t) = \sum_{j=1}^{3} h_j(x) \{ \begin{bmatrix} \mathbf{K}_{n,j} & \mathbf{K}_{c,j} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_c(t) \end{bmatrix} \}$$
(3.1)

3)

3.15)

dengan \mathbf{K}_n adalah *gain state feedback* dan \mathbf{K}_c adalah *gain* kompensator. Bentuk sinyal kontrol persamaan (3.13) disederhanakan menjadi:

$$\mathbf{u}(t) = \sum_{j=1}^{3} h_j(x) \mathbf{K}_j \mathbf{x}(t)$$
(3.14)

dengan K adalah gain kontroler yang merupakan augmentasi gain state feedback dan gain kompensator. Pada tugas akhir ini, sinyal referensi yang digunakan adalah sinyal sinus dengan frekuensi 0,1 Hz dan amplitudo 0,2 m. Dalam domain s, sinyal referensi tersebut dapat dirumuskan:

$$R(s) = \frac{0.0628}{s^2 + 0.3948} \tag{0}$$

Penurunan persamaan (3.14) dapat dilihat pada lampiran A.3. Persamaan *state* kompensator dapat dibangun berdasarkan sinyal referensi tersebut dengan elemen matriks kompensator:

$$\mathbf{A}_{c} = \begin{bmatrix} 0 & 1 \\ -0.3948 & 0 \end{bmatrix}; \mathbf{B}_{c} = \begin{bmatrix} 0 \\ 0.0628 \end{bmatrix}$$
(3.16)

Dari bentuk sistem augmentasi persamaan (3.12) dapat dibentuk sistem lup tertutup dari ketiga subsistem pendulum kereta sebagai berikut:

$$\begin{bmatrix} \mathbf{A}_{j} & \mathbf{0} \\ -\mathbf{B}_{c}\mathbf{C}_{j} & \mathbf{A}_{j} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{j} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{K}_{n,j} & \mathbf{K}_{c,j} \end{bmatrix}$$
(3.17)
 $j = 1,2,3$

4.3.2. Perancangan Fuzzy T-S Pole Placement

Sistem lup tertutup menjamin kestabilan sistem dengan meletakkan *pole-pole* pada sebelah kiri sumbu imajiner. Metode meletakkan *pole-pole* sistem ini disebut *pole placement*. Terdapat beragam cara dalam *pole placement*, pada tugas akhir ini *pole placement* dirancang menggunakan LMI. Daerah *pole* sistem lup tertutup yang

diinginkan (*region*) dapat dilihat pada Gambar 3.3. Daerah ini membentuk trapesium. *Region* dari *pole* merupakan irisan dari *conic sector* (kerucut) dan *half-plane* (setengah bidang), konsep keduanya telah dijelaskan pada subbab (2.15). Pertidaksamaan untuk mencari *state feedback gain* pada LMI *pole-placement*:

 $P^{-1} > 0$

 $(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} + \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^T + 2\gamma\mathbf{P}^{-1} < 0$

 $\begin{bmatrix} \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} + \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^T] & \cos\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} - \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^T] \\ \cos\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} - \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^T]^T & \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} + \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^T] \end{bmatrix} < 0$ (3.18)

dengan persamaan sistem lup tertutup,

 $\dot{\mathbf{x}}(t) = (\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{x}(t)$



(3.19)

Gambar 3.3 Region dari pole-pole sistem lup tertutup (D)

Dengan memanipulasi secara matematis persamaan (3.18), didapatkan: $\mathbf{P}^{-1} > 0$

 $\mathbf{A}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T} + 2\gamma\mathbf{P}^{-1} < 0$

 $\begin{bmatrix} \sin\theta [\mathbf{A}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{B}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}] & \cos\theta [\mathbf{A}\mathbf{P}^{-1} - \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} - \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}] \\ \cos\theta [\mathbf{A}\mathbf{P}^{-1} - \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} - \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}]^{T} & \sin\theta [\mathbf{A}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}] \end{bmatrix} < 0$ (3.20)

persamaan (3.20) masih mengandung dua variabel yang berdekatan yaitu **K** dan \mathbf{P}^{-1} . Dengan subtitusi variabel berikut:

$$\mathbf{Y} = \mathbf{K}\mathbf{P}^{-1}$$
$$\mathbf{Q} = \mathbf{P}^{-1}$$

didapatkan LMI yang menjamin sistem lup tertutup pendulum kereta stabil berada pada *region* tertentu. LMI ini ditunjukkan oleh persamaan (3.21), dengan tiga subsistem yang masing-masing mempunyai *gain* kontroler (**K**).

$$\mathbf{Q}_j > 0$$

$$\mathbf{A}_{j}\mathbf{Q}_{j} + \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} + \mathbf{Y}_{j}^{T}\mathbf{B}_{j} + 2\gamma\mathbf{P}_{j} < 0$$

$$\left[\frac{\sin\theta \left[\mathbf{A}_{j}\mathbf{Q}_{j} + \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} + \mathbf{Y}_{j}^{T}\mathbf{B}_{j}\right]}{*} \cos\theta \left[\mathbf{A}_{j}\mathbf{Q}_{j} - \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} - \mathbf{Y}_{j}^{T}\mathbf{B}_{j}\right] \\ \sin\theta \left[(\mathbf{A}_{j}\mathbf{Q}_{j} + \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} + \mathbf{Y}_{j}^{T}\mathbf{B}_{j})^{T} \right] \right] < 0$$

j = 1, 2, 3

dengan θ adalah sudut antara sumbu real dan sumbu daerah kerucut yang diingkan, γ adalah sumbu batas half plane yang diinginkan berada di sebelah kiri sumbu tersebut.

4.3.3. Batasan Input-Output

Selain memenuhi kestabilan, sistem lup tertutup juga dirancang agar masukan dan keluarannya terbatas. Batasan tersebut dinyatakan dengan batasan sinyal kontrol $\mathbf{u}(t)$ dan keluaran yang dibatasi $\mathbf{z}(t)$ sesuai dengan pertidaksamaan (2.36) dan (2.37). Pertidaksamaan tersebut dapat dinyatakan dengan,

$$\left\|\mathbf{K}_{j}\mathbf{x}(t)\right\| \leq \left\|\mathbf{u}(t)\right\| \leq u_{\max}$$
$$\left\|\mathbf{C}_{z,j}\mathbf{x}(t)\right\| \leq \left\|\mathbf{z}(t)\right\| \leq z_{\max}$$
$$j = 1, 2, 3$$

Untuk mempermudah penerapan batasan pertidaksamaan (3.22) maka ditinjau terlebih dahulu tinjau persamaan Lyapunov sebagai berikut:

$$V(\mathbf{x}(t)) = \mathbf{x}(t)^T \mathbf{P} \mathbf{x}(t)$$

(3.23)

(3.22)

(3.21)

Sinyal kontrol dan keluaran yang dibatasi akan memenuhi batasan yang dirumuskan

$$V(\mathbf{x}(T)) \leq \beta$$
Subtitusi persamaan (3.23) ke (3.24)
$$\mathbf{x}(t)^{T} \mathbf{P} \mathbf{x}(t) \leq \beta$$

$$\frac{1}{\beta} \mathbf{x}(t)^{T} \mathbf{P} \mathbf{x}(t) \leq 1$$
(3.25)

Penurunan LMI untuk batasan pada sinyal kontrol $\mathbf{u}(t)$ dapat diperoleh dari persamaan (3.22) sebagai berikut:

$$\begin{aligned} \left\| \mathbf{K}_{j} \mathbf{x}(t) \right\| &\leq u_{\max} \\ \mathbf{x}(t)^{T} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \mathbf{x}(t) \leq u_{\max}^{2} \\ \frac{1}{u_{\max}^{2}} \mathbf{x}(t)^{T} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \mathbf{x}(t) \leq 1 \end{aligned}$$
(3.26)

Subtitusi persamaan (3.26) ke persamaan (3.25) akan diperoleh

$$\frac{1}{u_{\max}^{2}} \mathbf{x}(t)^{T} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \mathbf{x}(t) - \frac{1}{\beta} \mathbf{x}(t)^{T} \mathbf{P}_{j} \mathbf{x}(t) \le 0$$

$$\mathbf{x}(t)^{T} \left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{u_{\max}^{2}} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \right] \mathbf{x}(t) \ge 0$$

$$\left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{u_{\max}^{2}} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \right] \ge 0$$
(3.27)

Sedangkan penurunan LMI untuk batasan keluaran yang dibatasi $\mathbf{z}(t)$ dapat dilakukan dari (3.22) sebagai berikut:

$$\begin{aligned} \left\| \mathbf{C}_{z,j} \mathbf{x}(t) \right\| &\leq z_{\max} \\ \mathbf{x}(t)^T \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \mathbf{x}(t) &\leq z_{\max}^2 \\ \frac{1}{z_{\max}^2} \mathbf{x}(t)^T \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \mathbf{x}(t) &\leq 1 \end{aligned}$$
(3.28)

Subtitusi persamaan (3.25 ke persamaan (3.28) diperoleh

$$\frac{1}{z_{\max}^{2}} \mathbf{x}(t)^{T} \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \mathbf{x}(t) - \frac{1}{\beta} \mathbf{x}(t)^{T} \mathbf{P}_{j} \mathbf{x}(t) \leq 0$$

$$\mathbf{x}(t)^{T} \left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{z_{\max}^{2}} \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \right] \mathbf{x}(t) \geq 0$$

$$\left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{z_{\max}^{2}} \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \right] \geq 0$$
(3.29)

Penerapan Schur Complement dan pre-multiplying serta postmultiplying LMI (3.27) dan (3.29) dengan matriks **P**⁻¹ akan menghasilkan:

$$\begin{bmatrix} -\mathbf{Q}_{j} & -\mathbf{Y}_{j}^{T} \\ * & -\frac{u_{\max}^{2}}{\beta} \end{bmatrix} < 0$$
$$\begin{bmatrix} -\mathbf{Q}_{j} & -\mathbf{Q}_{j}\mathbf{C}_{z,j}^{T} \\ * & -\frac{z_{\max}^{2}}{\beta} \end{bmatrix} < 0$$

(3.30)

(3.31)

dengan

$$\mathbf{Q}_j = \mathbf{P}_j^{-1}$$

$$\mathbf{Y}_{j} = \mathbf{K}_{j} \mathbf{P}_{j}^{-}$$

Penurunan LMI (3.30) dan (3.31) dapat dilihat pada Lampiran A.4

4.4. Perhitungan Gain Kontroler

Subbab (3.3) menjelaskan bagaimana LMI untuk kontroler diperoleh yaitu LMI pole placement dan batasan *input-output*. Kedua LMI ini akan digunakan untuk menentukan gain kontroler. Spesifikasi desain yang dirancang dalam tugas akhir ini adalah:

- 1). Sistem stabil asimtotik
- 2). Pole-pole dari sistem lup tertutup berada pada region **D** seperti tampak pada Gambar 3.3.
- Sinyal kontrol maksimal kurang dari atau sama dengan u_{max} seperti pada pertidaksamaan (3.22)

- 4). Keluaran yang dibatasi, yaitu $\mathbf{z}(t)$ kurang dari atau sama dengan z_{max} seperti pada pertidaksamaan (3.22)
- 5). Sinyal kontrol dan keluaran yang dibatasi memenuhi batasan seperti pada pertidaksamaan (3.24)

(3.32)

Untuk memenuhi spesifikasi desain tersebut, harus terdapat matriks simetris Q yang memenuhi LMI:

 $A_{j} > 0$ $B_{j} < 0$ $X_{j} < 0$ $E_{j} > 0$ $H_{i} > 0$

dengan masing-masing constraint pada LMI (3.32),

$$A_{j} = Q_{j}$$

$$B_{j} = A_{j}Q_{j} + Q_{j}A_{j}^{T} - B_{j}Y_{j} - Y_{j}^{T}B_{j} + 2\gamma P$$

$$X_{j} = \begin{bmatrix} \sin\theta[A_{j}Q_{j} + Q_{j}A_{j}^{T} - B_{j}Y_{j} - Y_{j}^{T}B_{j}] & \cos\theta[A_{j}Q_{j} - Q_{j}A_{j}^{T} - B_{j}Y_{j} + Y_{j}^{T}B_{j}] \\ & \sin\theta[(A_{j}Q_{j} + Q_{j}A_{j}^{T} - B_{j}Y_{j} - Y_{j}^{T}B_{j})^{T}] \end{bmatrix}$$

$$E_{j} = \begin{bmatrix} Q_{j} & Y_{j}^{T} \\ * & \frac{u_{max}^{2}}{\beta} \end{bmatrix}$$

$$H_{j} = \begin{bmatrix} -Q_{j} & -Q_{j}C_{z,j}^{T} \\ * & -\frac{z_{max}^{2}}{\beta} \end{bmatrix}$$

$$Q_{j} = P_{j}^{-1}$$

$$Y_{j} = K_{j}P_{j}^{-1}$$
dengan gain kontroler yang dicari adalah:

$$K_{j} = Y_{j}P_{j}$$

$$(3.33)$$

Dalam Tugas Akhir ini digunakan parameter-parameter sebagai berikut:

- 1). **A**₁ dan **B**₁ adalah matriks hasil linearisasi sistem pendulum kereta di sekitar $x_2(t) = 0$ radian.
- 2). A_2 dan B_2 adalah matriks hasil linearisasi sistem pendulum kereta di sekitar $x_2(t) = \pm 0,2616$ radian.
- 3). **A**₃ dan **B**₃ adalah matriks hasil linearisasi sistem pendulum kereta di sekitar $x_2(t) = \pm 0,5232$ radian.

Parameter region pole-pole sistem pendulum kereta

- 1). $\theta = \pi/12$, yaitu sudut batas pole-pole diletakkan pada daerah conic sector (sudut antara sumbu batas dan sumbu real)
- 2). $\gamma = 2$, yaitu sumbu batas half plane pada kiri sumbu imajiner.

Parameter batasan input-output sistem pendulum kereta

- 1). $\mathbf{C}_{z,1} = \mathbf{C}_{z,2} = \mathbf{C}_{z,3} = [1 \ 0 \ 0 \ 0 \ 0]$, menyatakan keluaran yang dibatasi hanya $x_1(t)$, yaitu posisi kereta.
- 2). $\beta = 5$, fungsi energi Lyapunov yang dibatasi.
- 3). $u_{\text{max}} = 17,5$ N, yang menyatakan bahwa sinyal kontrol maksimal adalah $\pm 17,5$ N
- 4). $z_{\text{max}} = 0,4$ yang menyatakan bahwa posisi kereta maksimal adalah $\pm 0,4$

Untuk menyelesaikan LMI (3.32), digunakan MATLAB. Parameter dari penyelesaian LMI ini yaitu gain kontrol \mathbf{K}_1 , \mathbf{K}_2 , \mathbf{K}_3 dan nilai pole-pole yang didapatkan adalah:

						(3.34)
<i>K</i> ₃	$=10^{4}[0,1975]$	-0,2289	0,0772	-0,0643	- 2,1547	-4,6524]
K_2	$=10^{4}[0,1504]$	-0,1664	0,0580	-0,0442	-1,5903	-3,5215]
K_1	$=10^{4}[0,1415]$	-0,1537	0,0542	-0,0401	-1,4 <mark>842</mark>	-3,310 <mark>0</mark>]

Gain kontroler ketiga subsistem pada persamaan (3.34) adalah *gain* untuk masing-masing sistem pada ketiga titik kerja. Selanjutnya bentuk *Fuzzy* T-S akan dibentuk kontroler. Keempat elemen matriks *gain* pertama adalah *gain state feedback* dan kedua elemen selanjutnya adalah *gain* kompensator. Sedangkan *pole-pole* sistem lup tertutupnya,

Subsistem 1	Subsistem 2	Subsistem 3
$x_2(t) = \pm 0$ radian	$x_2(t) = \pm 0,26$ radian	$x_2(t) = \pm 0,52$ radian
-26,5603+0.0000i	-27,0589+0.0000i	-28,9491+0.0000i
- 8,1007 + 0.0000i	- 8,1184 + 0.0000i	-8,7111+0.0000i
- 2,1765 + 0.3681i	- 2,2029 + 0.3568i	- 4,8649 + 0.0000i
- 2,1765 - 0.3681i	- 2,2029 - 0.3568i	- 2,2217 + 0.3682i
- 4,5090 + 0.0000i	- 4,5227 + 0.0000i	- 2,2217 - 0.3682i
- 3,8251 + 0.0000i	- 3,7589 + 0.0000i	- 3,5560 + 0.0000i

(3.35)

Pole-pole sistem lup tertutup (3.35) terletak di sebelah kiri dari desain yang diinginkan yaitu $\gamma = 2 \text{ dan } \theta = \pi/12$ radian.

4.5. Desain Observer Kesalahan Sensor dan Aktuator berbasis Fuzzy T-S PPIO (Proportional Proportional Integral Observer)

Pada subbab ini dijelaskan desain *observer* yang berfungsi untuk mengestimasi kesalahan sensor dan aktuator berbasis PPIO. Perbedaan *observer* PPIO dengan *observer* pada umumnya adalah *observer gain* pada *observer* PPIO terdapat dua, yaitu gain proportional dan gain proportional integral. Penambahan gain proportional integral disini dimaksudnya untuk mempercepat proses estimasi dari *observer*. Struktur *observer* secara umum dapat dilihat pada Gambar 3.4.



Gambar 3.4 Desain Observer PPIO Kesalahan Aktuator dan Sensor

4.5.1. Desain *Observer* **Kesalahan Sensor berbasis** *Fuzzy* **T-S PPIO** *Observer* kesalahan sensor digunakan untuk mengestimasi dan mengompensasi kesalahan sensor. *Observer* yang didesain adalah tipe

model-based (menggunakan model *plant*). Struktur observer dapat dilihat pada Gambar 3.5. Terdapat dua gain pada desain, yaitu gain proportional integral ($\overline{\mathbf{F}}$) dan gain proportional ($\overline{\mathbf{P}}$). Masukan dari observer adalah sinyal kontrol ($\mathbf{u}(t)$), estimasi kesalahan aktuator (\hat{f}_a) dan keluaran yang belum terkompensasi ($\mathbf{y}(t)$).





Untuk menghindari perkalian langsung kesalahan sensor dan *observer gain*, digunakan LPF (*Low Pass Filter*). Desain LPF dapat dilihat pada Gambar 3.6.



Observer didesain berbasis *Fuzzy* T-S. Untuk mempermudah perhitungan dan penulisan, sistem *Fuzzy* T-S pada persamaan (2.9) diubah menggunakan variabel pada persamaan (3.36)

$$\mathbf{A}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{A}_{j},$$

$$\mathbf{B}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{B}_{j},$$

$$\mathbf{C}_{c}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{C}_{j},$$
(3.36)

dengan C_c adalah keluaran untuk keperluan desain *observer*, dengan adanya kesalahan sensor, sistem *Fuzzy* T-S dapat ditulis:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x} + \mathbf{B}(p)(\mathbf{u}(t) + f_a(t))$$

$$\mathbf{y}(t) = \mathbf{C}_c \mathbf{x}(t) + \mathbf{D}_f f_s(t)$$
(3.37)

dengan $\dot{\mathbf{x}}(t) \in \mathbb{R}^n$ adalah persamaan *state*, $\mathbf{u}(t) \in \mathbb{R}^m$ adalah sinyal kontrol, $\mathbf{y}(t) \in \mathbb{R}^l$ adalah keluaran yang terukur, $f_s(t) \in \mathbb{R}^g$ adalah vektor kesalahan sensor, $f_a(t) \in \mathbb{R}^m$ adalah vektor kesalahan aktuator dan D_f adalah vektor yang mempresentasikan kesalahan sensor. $\mathbf{C}_c(p)$ adalah keluaran untuk keperluan desain *observer*, yang dituliskan:

$$\mathbf{C}_{c,1} = \mathbf{C}_{c,2} = \mathbf{C}_{c,3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$
(3.38)

Keluaran untuk mendesain *observer* berbeda dengan keluaran untuk mendesain kontroler, hal ini karena desain *observer* harus memenuhi kondisi observabilitas yang didefinisikan:

$$rank\begin{bmatrix} s\mathbf{I} - \mathbf{A} & 0\\ \mathbf{A}_s \mathbf{C} & s\mathbf{I} + \mathbf{A}_s\\ \mathbf{0} & \mathbf{I} \end{bmatrix} = n + l \tag{3.39}$$

Sistem pada persamaan (3.37) diaugmentasi dengan sistem LPF persamaan (3.40) sehingga menghasilkan sistem pada persamaan (3.41).

$$\dot{\mathbf{x}}_{s}(t) = -\mathbf{A}_{s}\mathbf{x}_{s}(t) + \mathbf{A}_{s}\mathbf{C}_{c}\mathbf{x}(t) + \mathbf{A}_{s}\mathbf{D}_{f}f_{s}(t)$$

dengan matriks A_s adalah matriks full rank. Sehingga sistem augmentasinya:

$$\dot{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}(p)\overline{\mathbf{x}}(t) + \overline{\mathbf{B}}(p)(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f f_s(t)$$

$$\overline{\mathbf{A}}(p) = \begin{bmatrix} \mathbf{A}(p) & 0 \\ \mathbf{A}_s \mathbf{C}_c & -\mathbf{A}_s \end{bmatrix}, \overline{\mathbf{B}}(p) = \begin{bmatrix} \mathbf{B}(p) \\ 0 \end{bmatrix}, \overline{\mathbf{D}}_f = \begin{bmatrix} 0 \\ \mathbf{A}_s \mathbf{D}_f \end{bmatrix}$$
$$\overline{\mathbf{x}}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_s(t) \end{bmatrix}, \overline{\mathbf{C}}_c = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

Dari sistem augmentasi ini dibangun *observer* untuk kesalahan sensor. *Observer* didesain berbasis *Fuzzy* T-S dengan premis dari sudut pendulum $(x_2(t))$. Terdapat tiga aturan untuk membangun *Fuzzy* T-S *Observer* ditunjukkan pada persamaan (3.42). Tiga aturan tersebut adalah $x_2(t) = 0$ rad, $\pm 0,2616$ rad, dan $\pm 0,5232$ rad. Aturan *Fuzzy* ini disederhanakan menggunakan persamaan (3.36) menjadi persamaan (3.43).

(3.41)

Aturan *observer* ke-1 If $x_2(t)$ is M₁ (sekitar 0 rad)

Then $\hat{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}_1 \hat{\overline{\mathbf{x}}}(t) + \overline{\mathbf{B}}_1(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f \hat{f}_s(t) + \overline{\mathbf{L}}_1 \overline{\mathbf{C}}_c e_x(t)),$

$$\hat{f}_{s}(t) = \overline{\mathbf{F}}_{1}\overline{\mathbf{C}}_{c}(\dot{e}_{x}(t) + e_{x}(t)),$$

Aturan *observer* ke-2 If $x_0(t)$ is M₂ (sekitar 0.2616 rad)

Then
$$\overline{\mathbf{x}}(t) = \mathbf{A}_2 \overline{\mathbf{x}}(t) + \mathbf{B}_2 (\mathbf{u}(t) + f_a(t)) + \mathbf{D}_f f_s(t) + \mathbf{L}_2 \mathbf{C}_c e_x(t)),$$

$$\hat{f}_s(t) = \overline{\mathbf{F}}_2 \overline{\mathbf{C}}_c(\dot{e}_x(t) + e_x(t)),$$

Aturan observer ke-3 If $x_2(t)$ is M₃ (sekitar 0.5232 rad)

Then $\dot{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}_3 \dot{\overline{\mathbf{x}}}(t) + \overline{\mathbf{B}}_3(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f \hat{f}_s(t) + \overline{\mathbf{L}}_3 \overline{\mathbf{C}}_c e_x(t)),$
$$\hat{f}_s(t) = \overline{\mathbf{F}}_3 \overline{\mathbf{C}}_c(\dot{e}_x(t) + e_x(t)),$$

penyederhaan aturan Fuzzy T-S:

$$\dot{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}(p)\widehat{\overline{\mathbf{x}}}(t) + \overline{\mathbf{B}}(p)(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f \hat{f}_s(t) + \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_c e_x(t)),$$
$$\dot{\overline{f}}_s(t) = \overline{\mathbf{F}}(p)\overline{\mathbf{C}}_c (\dot{e}_x(t) + e_x(t)),$$
(3.43)

dengan $\hat{\overline{x}}(t) \in \mathbb{R}^{n+l}$ adalah estimasi *state*, \hat{f}_s adalah estimasi kesalahan aktuator yang dihasilkan *observer* lain, $\overline{L}(p) \in \mathbb{R}^{(n+l)xl}$ dan $\overline{F}(p) \in \mathbb{R}^{gxl}$ adalah *observer gain* yang akan didesain berbasis *Fuzzy* T-S, dan $e_x(t)$ adalah error estimasi. Dengan *gain proportional* dan *observer gain proportional* dan *observer gain* (3.45) dan (3.46):

$$e_x = \overline{\mathbf{x}}(t) - \hat{\overline{\mathbf{x}}}(t), \qquad (3.44)$$

$$\overline{\mathbf{L}}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \overline{\mathbf{L}}_{j},$$

$$\overline{\mathbf{F}}(p) = \sum_{i=1}^{r} \mathbf{h}_{j} * \overline{\mathbf{F}}_{j},$$

4.5.2. Desain LMI (*Linear Matrix Equalities*) untuk Observer Kesalahan Sensor

Desain observer memastikan performa H_{∞} sehingga error estimasi dibuat sekecil mungkin. LMI digunakan untuk menentukan observer gain. Desain LMI menggunakan persamaan matriks augmentasi estimasi state (3.43) dan persamaan state (3.41). Untuk mendesain LMI, langkah pertama adalah menentukan error estimasi state. Dari persamaan (3.43) dan (3.41) didapatkan error estimasi state:

$$\dot{e}_{x}(t) = \mathbf{\bar{x}} - \mathbf{\bar{x}}$$
$$\dot{e}_{x}(t) = (\mathbf{\overline{A}}(p) - \mathbf{\overline{L}}(p)\mathbf{\overline{C}}_{c})e_{x}(t) + \mathbf{\overline{D}}_{f}e_{fs}(t) + \mathbf{\overline{B}}(p)e_{fs}(t)$$

(3.47)

dengan,

$$e_{fs}(t) = f_s(t) - f_s(t)$$

(3.45)

(3.42)

(3.46)

 $e_{fa}(t) = f_a(t) - \hat{f}_a(t)$

Persamaan (3.46) disebut error estimasi *state*, yaitu selisih nilai *state* yang sebenarnya (matriks augmentasi) dikurangi nilai *state* hasil estimasi *observer*. Selanjutnya dari error estimasi *state* dicari error estimasi kesalahan sensor yang dirumuskan:

$$e_{fs}(t) = f_s(t) - \hat{f}_s(t)$$
 (3.48)

Error estimasi kesalahan sensor adalah selisih kesalahan sensor yang sebenarnya dikurangi dengan estimasi kesalahan sensor. Dari hasil subtitusi persamaan (3.43) dan (3.46) ke persamaan (3.47) didapatkan:

$$\frac{\dot{e}_{fs}(t) = \dot{f}_{s}(t) - \overline{\mathbf{F}}(p)\overline{\mathbf{C}}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}} + \mathbf{I})e_{x}(t) - \overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{D}}_{f}e_{fs}(t) - \overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{B}}(p)e_{fa}(t)$$
(3.49)

Persamaan error estimasi *state* (3.46) dan error estimasi kesalahan sensor (3.48) digunakan untuk menentukan matrik kandidat fungsi Lyapunov. Kandidat fungsi Lyapunov digunakan untuk menjamin sistem *observer* (3.43) stabil dan memenuhi performa H_{∞} seperti berikut:

$$\frac{\left\| \tilde{\boldsymbol{e}}_{p}(t) \right\|_{2}}{\left\| \tilde{\boldsymbol{z}}(t) \right\|_{2}} <$$

dengan,

$$\widetilde{e}_{p}(t) = \mathbf{C}_{p}\widetilde{e}_{as}(t)$$
$$\widetilde{e}_{as}(t) = \begin{bmatrix} e_{x}(t) \\ e_{fs}(t) \end{bmatrix}$$
$$\widetilde{\mathbf{z}}(t) = \begin{bmatrix} e_{fa}(t) \\ \dot{f}_{s}(t) \end{bmatrix}$$

(3.50)

(3.52)

Persamaan (3.50) adalah L_2 -gain untuk observer kesalahan sensor. L_2 -gain dirancang untuk kurang dari atenuasi/tingkat pelemahan tertentu. Masukan performa ditunjukkan persamaan (3.52), yang terdiri atas error estimasi *state* dan error estimasi kesalahan sensor. Keluaran performa ditunjukkan persamaan (3.53) yang terdiri atas error estimasi kesalahan aktuator dan derivatif turunan kesalahan sensor. Tingkat pelemahan/atenuasi dicari agar estimasi memiliki akurasi yang tinggi dan cepat. Dari persamaan (3.46) dan (3.48) didapatkan matriks augmentasi error estimasi *state* dan error estimasi kesalahan sensor:

$$\begin{bmatrix} \dot{e}_{x}(t) \\ \dot{e}_{fs}(t) \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} & \overline{\mathbf{D}}_{f} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} + \mathbf{I}) & -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{D}}_{f} \end{bmatrix} \begin{bmatrix} e_{x}(t) \\ e_{fs}(t) \end{bmatrix} \\ + \begin{bmatrix} \overline{\mathbf{B}}(p) & 0 \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{B}}(p) & \mathbf{I} \end{bmatrix} \begin{bmatrix} e_{fa} \\ \dot{f}_{s} \end{bmatrix}$$

(3.53)

(3.56)

Dari matriks augmentasi kandidat fungsi Lyapunov (3.53) definisikan variabel berikut untuk penyederhanaan dalam desain LMI:

$$\begin{split} \widetilde{\boldsymbol{e}}_{as}(t) &= \begin{bmatrix} \boldsymbol{e}_{x}(t) \\ \boldsymbol{e}_{fs}(t) \end{bmatrix}, \widetilde{\boldsymbol{z}} \begin{bmatrix} \boldsymbol{e}_{fa}(t) \\ \dot{f}_{s}(t) \end{bmatrix} \\ \widetilde{\boldsymbol{A}}_{s}(p,p) &= \begin{bmatrix} \overline{\boldsymbol{A}}(p) - \overline{\boldsymbol{L}}(p)\overline{\boldsymbol{C}}_{c} & \overline{\boldsymbol{D}}_{f} \\ -\overline{\boldsymbol{F}}(p)\overline{\boldsymbol{C}}\overline{\boldsymbol{C}}\overline{\boldsymbol{A}}(p) - \overline{\boldsymbol{L}}(p)\overline{\boldsymbol{C}}_{c} + \mathbf{I}) & -\overline{\boldsymbol{F}}(p)\overline{\boldsymbol{C}}\overline{\boldsymbol{D}}_{f} \\ \widetilde{\boldsymbol{N}}(p,p) &= \begin{bmatrix} \overline{\boldsymbol{B}}(p) & 0 \\ -\overline{\boldsymbol{F}}(p)\overline{\boldsymbol{C}}\overline{\boldsymbol{B}}(p) & \mathbf{I} \end{bmatrix} \end{split}$$

(3.54) Dari variabel yang didefinisikan pada persamaan (3.54), disubtitusikan ke persamaan (3.53) menghasilkan persamaan (3.55)

$$\dot{\tilde{e}}_{as}(t) = \tilde{\mathbf{A}}_{s}(p,p)\tilde{e}_{as}(t) + \tilde{\mathbf{N}}(p,p)\tilde{\mathbf{z}}(t)$$
(3.55)

Persamaan (3.55) adalah kandidat fungsi Lyapunov. Kandidat fungsi Lyapunov ini digunakan untuk menentukan LMI *Observer*, menjamin kestabilan dan keluaran L_2 -gain didesain kurang dari γ tertentu.

Definisikan fungsi Lyapunov,

$$v(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)\overline{\mathbf{P}}\tilde{e}_{as}(t)$$
$$\overline{\mathbf{P}} = \sum_{j=1}^{3} h_{j}(x)\mathbf{P}_{j}, \ j = 1,2,3$$

Derivatif fungsi Lyapunov,

$$\dot{v}(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)(\tilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\tilde{\mathbf{A}}_{s}(p,p))\tilde{e}_{as}(t)$$

$$+ \tilde{e}_{as}^{T}\overline{\mathbf{P}}\tilde{\mathbf{N}}(p,p)\tilde{\mathbf{z}} + \tilde{\mathbf{z}}^{T}\tilde{\mathbf{N}}^{T}(p,p)\overline{\mathbf{P}}\tilde{e}_{as}(t)$$
(3.57)

Persamaan L_2 -gain pada (3.49) dapat dilakukan manipulasi matematis sehingga menghasilkan persamaan (3.58)

$$\frac{1}{\gamma} \int_{0}^{\infty} \tilde{e}_{as}^{T} \overline{\mathbf{C}}_{p} \overline{\mathbf{C}}_{p} \tilde{\overline{\mathbf{c}}}_{as} dt - \gamma \int_{0}^{\infty} \tilde{\mathbf{z}}^{T} \tilde{\mathbf{z}} dt < 0$$
(3.58)

Persamaan (3.59) dalam bentuk fungsi Lyapunov,

$$\dot{\mathbf{v}}(\tilde{\boldsymbol{e}}_{as}) + \frac{1}{\gamma} \tilde{\boldsymbol{e}}_{as}^T \overline{\mathbf{C}}_p^T \overline{\mathbf{C}}_p \tilde{\boldsymbol{e}}_{as} - \gamma \tilde{\mathbf{z}}^T \tilde{\mathbf{z}} < 0$$
(3.59)

Subtitusi persamaan fungsi Lyapunov (3.57) ke persamaan (3.59)

$$\widetilde{\widetilde{e}}_{as}^{T}(t)(\widetilde{\mathbf{A}}_{s}^{T}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{P}}+\widetilde{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(\boldsymbol{p},\boldsymbol{p}))\widetilde{e}_{as}(t)+\widetilde{e}_{as}^{T}\widetilde{\mathbf{P}}\widetilde{\mathbf{N}}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{z}}$$

$$+\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{N}}^{T}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{P}}\widetilde{e}_{as}(t)+\frac{1}{\gamma}\widetilde{e}_{as}^{T}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p}\widetilde{e}_{as}-\gamma\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{z}}<0$$
(3.60)

dengan metode Schur Complement dan metode *pre-multiplying* dan *post-multipling* didapatkan pertidaksamaan (3.61). Penurunan pertidaksamaan ini dapat dilihat pada Lampiran A.5

$$\begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}}+\overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) & \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) & \overline{\mathbf{C}}_{p1}^{T} & \mathbf{0} \\ & & & -\gamma \mathbf{I} & \mathbf{0} & \overline{\mathbf{C}}_{p2}^{T} \\ & & & * & -\gamma \mathbf{I} & \mathbf{0} \\ & & & & * & * & -\gamma \mathbf{I} \end{bmatrix} <$$

(3.61)

dengan subtitusi variabel berikut ke pertidaksamaan (3.61),

$$\widetilde{\mathbf{A}}_{s}(p,p) = \begin{bmatrix} \overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} & \overline{\mathbf{D}}_{f} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} + \mathbf{I}) & -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{D}}_{f} \end{bmatrix},$$

$$\widetilde{\mathbf{N}}(p,p) = \begin{bmatrix} \overline{\mathbf{B}}(p) & 0 \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{B}}(p) & \mathbf{I} \end{bmatrix},$$

$$\overline{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{1} & 0 \\ 0 & \mathbf{I} \end{bmatrix},$$

$$\overline{\mathbf{H}}(p) = \mathbf{P}_{1}\overline{\mathbf{L}}(p),$$

$$\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c} = \overline{\mathbf{D}}_{f}\mathbf{P}_{1}$$
(3.62)

didapatkan LMI pada (3.63),

-

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & 0 \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma \mathbf{I} & 0 & 0 \\ * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & -\gamma \mathbf{I} \end{bmatrix} < 0$$

$$w_{11} = \mathbf{P}_{\mathbf{I}} \overline{\mathbf{A}}(p) + (\mathbf{P}_{\mathbf{I}} \overline{\mathbf{A}}(p))^{T} - \overline{\mathbf{H}}(p) \overline{\mathbf{C}}_{c} - (\overline{\mathbf{H}}(p) \mathbf{C}_{c})^{T},$$

$$w_{12} = -(\overline{\mathbf{A}}^{T}(p) \mathbf{P}_{\mathbf{I}} \mathbf{D}_{f} - \overline{\mathbf{C}}_{c}^{T} \overline{\mathbf{H}}^{T}(p) \overline{\mathbf{D}}_{f}),$$

$$w_{13} = \mathbf{P}_{\mathbf{I}} \overline{\mathbf{B}}(p),$$

$$w_{22} = -2 \overline{\mathbf{D}}_{f}^{T} \mathbf{P}_{\mathbf{I}} \overline{\mathbf{D}}_{f},$$

$$w_{23} = -2 \overline{\mathbf{D}}_{f}^{T} \mathbf{P}_{\mathbf{I}} \overline{\mathbf{B}}(p),$$

untuk meningkatkan performa *observer* digunakan LMI yang menjamin *Fast Fault Estimation* (3.64).

$$\begin{bmatrix} \mu \mathbf{I} & \overline{\mathbf{D}}_{f} \mathbf{P}_{1} - \overline{\mathbf{F}}(p) \overline{\mathbf{C}}_{c} \\ * & \mu \mathbf{I} \end{bmatrix} > 0$$
(3.64)

(3.63)

Sedangkan untuk meletakkan *pole-pole observer* pada daerah tertentu yang memiliki kriteria performa yang lebih baik, seperti *overshoot*, digunakan LMI Region (3.65).

(3.65)

$$\begin{split} & \sum_{i} + \sum_{i}^{T} + 2\rho \overline{\mathbf{P}} < 0 \\ & \left[\sin(\theta) [\sum_{i} + \sum_{i}^{T}] \cos(\theta) [\sum_{i} + \sum_{i}^{T}] \\ & * \sin(\theta) [\sum_{i} + \sum_{i}^{T}] \right] < 0 \end{split}$$

dengan,

$$\begin{split} \boldsymbol{\Sigma}_{i} &= \overline{\mathbf{P}} \mathbf{A}_{s}(p, p) = \\ \begin{bmatrix} \mathbf{P}_{1} \overline{\mathbf{A}}(p) - \overline{\mathbf{H}}(p) \overline{\mathbf{C}}_{c} & \mathbf{P}_{1} \overline{\mathbf{D}}_{f} \\ - (\overline{\mathbf{A}}^{T}(p) \mathbf{P}_{1} \overline{\mathbf{D}}_{f} - \overline{\mathbf{C}}_{c}^{T} \overline{\mathbf{H}}^{T} \overline{\mathbf{D}}_{f} + \mathbf{P}_{1} \overline{\mathbf{D}}_{f} \end{bmatrix} - \overline{\mathbf{D}}^{T}_{f} \mathbf{P}_{1} \overline{\mathbf{D}}_{f} \end{split}$$

Sehingga constraint LMI pada (3.63), (3.64) dan (3.65) dapat digabung:

$$\begin{split} & \min(\gamma + \mu) \\ \begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & 0 \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma \mathbf{I} & 0 & 0 \\ * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & * & -\gamma \mathbf{I} \end{bmatrix} < 0 \\ \begin{bmatrix} \mu \mathbf{I} & \overline{\mathbf{D}}_{f} \mathbf{P}_{1} - \overline{\mathbf{F}}(p) \overline{\mathbf{C}}_{c} \\ * & \mu \mathbf{I} \end{bmatrix} > 0 \\ \sum_{i} + \sum_{i}^{T} + 2\rho \overline{\mathbf{P}} < 0 \\ \begin{bmatrix} \sin(\theta) [\sum_{i} + \sum_{i}^{T}] & \cos(\theta) [\sum_{i} + \sum_{i}^{T}] \\ * & \sin(\theta) [\sum_{i} + \sum_{i}^{T}] \end{bmatrix} < 0 \\ \text{dengan,} \\ w_{13} = \mathbf{P}_{I} \overline{\mathbf{B}}(p), \\ w_{22} = -2 \overline{\mathbf{D}}_{f}^{T} \mathbf{P}_{I} \overline{\mathbf{D}}_{f}, \end{split}$$
(3.66)

$$\begin{split} w_{23} &= -2\overline{\mathbf{D}}_{f}^{T}\mathbf{P}_{I}\overline{\mathbf{B}}(p), \\ w_{11} &= \mathbf{P}_{I}\overline{\mathbf{A}}(p) + (\mathbf{P}_{I}\overline{\mathbf{A}}(p))^{T} - \overline{\mathbf{H}}(p)\overline{\mathbf{C}}_{c} - (\overline{\mathbf{H}}(p)\mathbf{C}_{c})^{T}, \\ w_{12} &= -(\overline{\mathbf{A}}^{T}(p)\mathbf{P}_{I}\mathbf{D}_{f} - \overline{\mathbf{C}}_{c}^{T}\overline{\mathbf{H}}^{T}(p)\overline{\mathbf{D}}_{f}), \\ \Sigma_{i} &= \overline{\mathbf{P}}\mathbf{A}_{s}(p, p) = \\ \begin{bmatrix} \mathbf{P}_{I}\overline{\mathbf{A}}(p) - \overline{\mathbf{H}}(p)\overline{\mathbf{C}}_{c} & \mathbf{P}_{I}\overline{\mathbf{D}}_{f} \\ -(\overline{\mathbf{A}}^{T}(p)\mathbf{P}_{I}\overline{\mathbf{D}}_{f} - \overline{\mathbf{C}}_{c}^{T}\overline{\mathbf{H}}^{T}\overline{\mathbf{D}}_{f} + \mathbf{P}_{I}\overline{\mathbf{D}}_{f}) & -\overline{\mathbf{D}}^{T}_{f}\mathbf{P}_{I}\overline{\mathbf{D}}_{f} \end{bmatrix} \end{split}$$

Observer gain proportional dan *proportional integral* pada persamaan (3.66) dapat dicari dengan persamaan berikut:

$$\overline{\mathbf{L}}(p) = \mathbf{P}_{1}^{-1} \overline{\mathbf{H}}(p)$$

$$\overline{\mathbf{F}}(p)$$
(3.67)

(3.68)

dengan estimasi kesalahan sensornya,

$$\hat{f}_s(t) = \overline{\mathbf{F}}(p)\overline{\mathbf{C}}_c \int (\dot{e}_x(t) + e_x(t))dt$$

4.5.3. Desain Observer Kesalahan Aktuator Berbasis T-S PPIO

Desain observer kesalahan aktuator hampir sama dengan observer kesalahan sensor, hanya saja desain observer kesalahan aktuator tidak menggunakan augmentasi dengan LPF. Desain untuk Observer kesalahan aktuator dapat dilihat pada Gambar 3.7.



Gambar 3.7 Desain observer kesalahan aktuator

terdapat dua gain pada desain, yaitu proportional integral gain (\mathbf{F}_{a}) dan proportional gain (\mathbf{P}_{a}). Masukan dari observer adalah sinyal kontrol ($\mathbf{u}(t)$), dan keluaran yang sudah terkompensasi ($\mathbf{y}(t)$). Tinjau sistem dengan kesalahan aktuator dan sensor seperti persamaan (3.27) berikut:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x}(t) + \mathbf{B}(p)(\mathbf{u}(t) + f_a(t))$$

$$\mathbf{y}(t) = \mathbf{C}_c \mathbf{x}(t) + \mathbf{D}_f f_s(t)$$

 $\mathbf{F}_{a}(p) = \sum_{i=1}^{n} \mathbf{h}_{j} * \mathbf{F}_{a,j}$

Observer kesalahan aktuator direpresentasikan dengan persamaan (3.69) sebagai berikut:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x}(t) + \mathbf{B}(p)(\mathbf{u}(t) + \hat{f}_a) + \mathbf{L}_a(p)(\mathbf{C}\mathbf{x}(t) + \mathbf{D}_f e_{f\bar{\mathbf{x}}}(t) - \mathbf{C}\hat{\mathbf{x}}(t)),$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}_f e_{f\bar{\mathbf{x}}}(t)$$

(3.69)

dengan $\mathbf{\hat{x}}(t) \in \mathbb{R}^{n+l}$ adalah estimasi *state*, $\mathbf{\hat{f}}_a$ adalah estimasi kesalahan aktuator, $\mathbf{L}_{\mathbf{a}}(\mathbf{p}) \in \mathbb{R}^{n\times l}$ dan $\mathbf{L}_{\mathbf{a}}(\mathbf{p}) \in \mathbb{R}^{m\times l}$ adalah observer gain yang akan didesain berbasis Fuzzy T-S, dan $e_x(t)$ adalah error estimasi. Dengan gain proportional dan gain proportional integral observer ditunjukkan:

$$e_{x}(t) = \overline{\mathbf{x}}(t) - \overline{\mathbf{x}}(t), \qquad (3.70)$$

$$\mathbf{L}_{a}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{L}_{a,j} \qquad (3.71)$$

$$\mathbf{F}_{a}(p) = \sum_{i=1}^{r} \mathbf{h}_{i} * \mathbf{F}_{a,i} \qquad (3.72)$$

4.5.4. Desain LMI (*Linear Matrix Equalities*) untuk Observer Kesalahan Aktuator

Seperti pada observer kesalahan sensor, LMI untuk kesalahan aktuator memiliki desain yang hampir sama. Desain observer menjamin kestabilan observer dan performa H_{∞} . Dengan prosedur yang hampir sama, persamaan error estimasi *state* dan error estimasi kesalahan sensornya:

$$\dot{\boldsymbol{e}}_{x}(t) = (\mathbf{A}(\boldsymbol{p}) - \mathbf{L}_{a}(\boldsymbol{p})\mathbf{C}_{c})\boldsymbol{e}_{x}(t) + \mathbf{B}(\boldsymbol{p})\boldsymbol{e}_{fa}(t) - \mathbf{L}_{a}(\boldsymbol{p})\mathbf{D}_{f}\boldsymbol{e}_{fs}(t)$$

(3.73)

$$\dot{e}_{fa}(t) = f_a(t) - \mathbf{F}_a(p)\mathbf{C}_c(\mathbf{A}(p) - \mathbf{L}_a(p)\mathbf{C}_c + \mathbf{I})e_x(t) - \mathbf{F}_a(p)\mathbf{C}_c\mathbf{D}_f e_{fa}(t) + \mathbf{F}_a(p)\mathbf{C}_c\mathbf{L}_a(p)\mathbf{D}_f e_{fs}(t)$$

Dari persamaan error estimasi *state* dan error estimasi kesalahan aktuator dibentuk matriks augmentasi untuk kandidat fungsi Lyapunov sebagai berikut:

(3.74)

(3.75)

$$\begin{split} \dot{\tilde{e}}_{a}(t) &= \tilde{\mathbf{A}}(p, p)\tilde{e}_{a}(t) + \tilde{\mathbf{N}}(p, p)\tilde{\mathbf{z}}(t) \\ \tilde{\mathbf{A}}(p, p) &= \begin{bmatrix} \mathbf{A}(p) - \mathbf{L}_{a}(p)\mathbf{C}_{c} & \mathbf{B}(p) \\ -\mathbf{F}_{a}(p)\mathbf{C}_{c}(\mathbf{A}(p) - \mathbf{L}_{a}(p)\mathbf{C}_{c} + \mathbf{I}) & -\mathbf{F}_{a}(p)\mathbf{C}_{c}\mathbf{B}(p) \end{bmatrix}, \\ \tilde{\mathbf{N}}(p, p) &= \begin{bmatrix} -\mathbf{L}_{a}(p)\mathbf{D}_{f} & 0 \\ -\mathbf{F}_{a}(p)\mathbf{C}\mathbf{L}_{a}(p)\mathbf{D}_{f} & I \end{bmatrix}, \end{split}$$

Kandidat fungsi Lyapunov pada persamaan (3.75) digunakan untuk memenuhi performa H_{∞} dengan L_2 -gain sebagai berikut:

$$\frac{\left\| \tilde{\boldsymbol{e}}_{p}(t) \right\|_{2}}{\left\| \tilde{\boldsymbol{z}}(t) \right\|_{2}} < \gamma$$
(3.76)

mgan,
$$\tilde{\boldsymbol{e}}_{p}(t) = \mathbf{C}_{p} \tilde{\boldsymbol{e}}_{as}(t)$$

$$\tilde{\boldsymbol{e}}_{a}(t) = \begin{bmatrix} \boldsymbol{e}_{x}(t) \\ \boldsymbol{e}_{fa}(t) \end{bmatrix}$$
(3.77)
$$\tilde{\boldsymbol{z}}(t) = \begin{bmatrix} \boldsymbol{e}_{fs}(t) \\ \boldsymbol{\dot{f}}_{a}(t) \end{bmatrix}$$
(3.78)

Persamaan (3.76) adalah L_2 -gain untuk observer kesalahan aktuator. L_2 -gain dirancang untuk kurang dari atenuasi/tingkat pelemahan tertentu. Masukan performa ditunjukkan persamaan (3.77), yang terdiri atas error estimasi state dan error estimasi kesalahan aktuator. Keluaran performa ditunjukkan persamaan (3.78) yang terdiri atas error estimasi kesalahan sensor dan derivatif turunan kesalahan aktuator.

definisikan fungsi Lyapunov,

$$v(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)\mathbf{P}_{a}\tilde{e}_{as}(t)$$
$$\mathbf{P}_{a} = \sum_{j=1}^{3} h_{j}(x)\mathbf{P}_{a,j}$$

derivatif fungsi Lyapunov,

$$\dot{\psi}(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)(\tilde{\mathbf{A}}_{s}^{T}(p,p)\mathbf{P}_{a} + \mathbf{P}_{a}\tilde{\mathbf{A}}_{s}(p,p))\tilde{e}_{as}(t)$$

$$+ \tilde{e}_{as}^{T}\mathbf{P}_{a}\tilde{\mathbf{N}}(p,p)\tilde{\mathbf{z}} + \tilde{\mathbf{z}}^{T}\tilde{\mathbf{N}}^{T}(p,p)\mathbf{P}_{a}\tilde{e}_{as}(t)$$

$$(3.80)$$

(3.79)

(3.81)

dengan metode yang hampir sama untuk menentukan matriks kestabilan Lyapunov sesuai persamaan (3.57), (3.58), (3.59) dan (3.60) didapatkan LMI (3.81).

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & 0 \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma_{a}\mathbf{I} & 0 & 0 \\ * & * & * & -\gamma_{a}\mathbf{I} & 0 \\ * & * & * & * & -\gamma_{a}\mathbf{I} \\ w_{13} = -\mathbf{H}(p)\mathbf{D}_{f}, \\ w_{22} = -(\mathbf{B}(p)^{T}\mathbf{P}_{a}\mathbf{B}(p) + \mathbf{B}(p)\mathbf{P}_{a}\mathbf{B}(p)^{T}), \end{bmatrix} < 0$$

$$w_{22} = -\mathbf{B}(p)^{T} \mathbf{H}(p) \mathbf{D}_{f},$$

$$w_{11} = \mathbf{P}_{a} \mathbf{A}(p) + (\mathbf{P}_{a} \mathbf{A}(p))^{T} - \mathbf{H}(p) \mathbf{C}_{c} - (\mathbf{H}(p) \mathbf{C}_{c})^{T},$$

$$w_{12} = -(\mathbf{A}^{T}(p) \mathbf{P}_{a} \mathbf{B}(p) - \mathbf{C}_{c}^{T} \mathbf{H}^{T}(p) \mathbf{B}(p)),$$

untuk meningkatkan performa digunakan LMI fast fault estimation:

$$\begin{bmatrix} \mu_a \mathbf{I} & \mathbf{B}(p)^T \mathbf{P}_a - \mathbf{F}_a(p) \mathbf{C}_c \\ * & \mu_a \mathbf{I} \end{bmatrix} > 0$$
(3.82)

untuk meletakkan pole-pole *observer* pada kriteria yang diinginkan memiliki kriteria performa yang lebih baik, seperti *overshoot*, digunakan LMI Region (3.83)

$$\sum_{ai} + \sum_{ai}^{T} + 2\rho \mathbf{P}_{a} < 0$$

$$\sum_{ai} + \sum_{ai}^{T} \frac{1}{\cos(\theta)} \sum_{ai} + \sum_{ai}^{T} \frac{1}{\sin(\theta)} \sum_{ai} + \sum_{ai}^{T} \frac{1}{\sin(\theta)}] < 0$$
(3.83)

Sehingga penggabungan constraint untuk LMI *observer* kesalahan aktuator dari pertidaksamaan (3.81), (3.82) dan (3.83) untuk mencari *observer gain*:

$$\min(\gamma_{a} + \mu_{a}) = \begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & \mathbf{0} \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma_{a}\mathbf{I} & 0 & 0 \\ * & * & * & -\gamma_{a}\mathbf{I} & \mathbf{0} \\ * & * & * & * & -\gamma_{a}\mathbf{I} \end{bmatrix} < 0$$

$$\begin{array}{c|c} \mu_{a}\mathbf{I} & \mathbf{B}(p)^{T} \mathbf{P}_{a} - \mathbf{F}_{a}(p) \mathbf{C}_{c} \\ \ast & \mu_{a}\mathbf{I} \end{array} > 0$$

 $\sum_{ai} + \sum_{ai}^{T} + 2\rho \mathbf{P}_{a} < 0$ $\left[\frac{\sin(\theta) [\sum_{ai} + \sum_{ai}^{T}]}{*} \frac{\cos(\theta) [\sum_{ai} + \sum_{ai}^{T}]}{\sin(\theta) [\sum_{ai} + \sum_{ai}^{T}]} \right] < 0$

(3.84)

dengan,

$$w_{11} = \mathbf{P}_{a}\mathbf{A}(p) + (\mathbf{P}_{a}\mathbf{A}(p))^{T} - \mathbf{H}(p)\mathbf{C}_{c} - (\mathbf{H}(p)\mathbf{C}_{c})^{T},$$

$$w_{12} = -(\mathbf{A}^{T}(p)\mathbf{P}_{a}\mathbf{B}(p) - \mathbf{C}_{c}^{T}\mathbf{H}^{T}(p)\mathbf{B}(p)),$$

$$w_{13} = -\mathbf{H}(p)\mathbf{D}_{f},$$

$$w_{22} = -(\mathbf{B}(p)^{T}\mathbf{P}_{a}\mathbf{B}(p) + \mathbf{B}(p)\mathbf{P}_{a}\mathbf{B}(p)^{T}),$$

$$w_{33} = -\mathbf{B}(p)^{T}\mathbf{H}(p)\mathbf{D}_{f},$$

$$\Sigma_{ai} = \mathbf{P}_{a1}\mathbf{A}_{s}(p, p) = \begin{bmatrix} \mathbf{P}_{a1}\mathbf{A}(p) - \mathbf{H}(p)\mathbf{C}_{c} & \mathbf{P}_{a1}\mathbf{B}(p) \\ -(\mathbf{A}^{T}(p)\mathbf{P}_{a1}\mathbf{B}(p) - \mathbf{C}_{c}^{T}\mathbf{H}^{T}\mathbf{B}(p) + \mathbf{B}(p)^{T}\mathbf{P}_{a1})^{T} & -\mathbf{B}(p)^{T}\mathbf{P}_{a1}\mathbf{B}(p) \end{bmatrix}$$

observer gain didapatkan dengan persamaan berikut:

$$\mathbf{L}_{a}(p) = \mathbf{P}_{a}^{-1}\mathbf{H}(p)$$

$$\mathbf{F}_{a}(p)$$
(3.85)

dengan estimasi kesalahan aktuator dituliskan:

$$\hat{f}_a(t) = \mathbf{F}_a(p)\mathbf{C}_c \int (\dot{e}_x(t) + e_x(t))dt$$
(3.86)

4.6. Perhitungan Observer gain Kesalahan Sensor dan Aktuator

Dari Subbab 3.5 telah dijelaskan cara menurunkan LMI untuk kesalahan sensor dan aktuator, spesifikasi desain yang digunakan untuk keduanya adalah:

- 1). Observer berbasis Fuzzy T-S dengan premis dari sudut pendulum dan tiga aturan $(x_2(t) = 0 \text{ rad}, \pm 0.2616 \text{ rad}, \text{ dan} \pm 0.5232 \text{ rad})$
- 2). Sistem memenuhi performa H_{∞} dengan tingkat atenuasi/pelemahan tertentu
- 3). Jumlah kesalahan aktuator = 1, jumlah kesalahan sensor = 1.
- 4). Matriks kesalahan sensor didefinisikan, $\mathbf{D}_{f} = [1 \ 1 \ 1]^{T}$
- 5). Matriks performa fungsi Lyapunov (C_p) didefinisikan

$$\mathbf{C}_{p,1} = 0.4 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\mathbf{C}_{p,2} = 0.4[1]$$

$$C_{pa,1} = 2 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C_{pa,2} = 2[1]$$

6). Spesifikasi LMI Region untuk *observer* kesalahan sensor $\rho = 0.1$

$$\theta = \frac{\pi}{4}$$

7). Spesifikasi LMI Region untuk *observer* kesalahan aktuator $\rho = 0.5$

$$\theta = \frac{\pi}{4}$$

Dari LMI untuk kesalahan sensor (3.66) dan LMI untuk kesalahan aktuator (3.86) didapatkan *observer gain* untuk masing-masing subsistem sebagai berikut:

	-106,0	42,80	66,60	474	-104,2	42,60	65,00	
TO J	- 408,1	344,7	<mark>89,00</mark>	61	<mark>- 39</mark> 3,0	332,6	85,20	
	- 575,1	- 296,2	857,8		- 549,9	-267,5	805,0	
$L_1 =$	- 2087	909,5	1254	$\mathbf{L}_2 =$	-1961	890,9	1144	
TAT	-11,00	13,10	0,600		-11,50	13,10	0,600	
SE	- 48,80	56,90	-2,500		-47,70	57,70	- 2,400	K
	- 61,90	-0,500	64,00		- 60,30	-0,400	61,80	R.
			-118,3	60,	50 61	,70]		
			-434,5	5 390	0,1 69	,70		
			- 53,62	2 -25	57,5 78	1,5		
		$L_3 =$	-1953	104	44 98	1,0		
			-13,80) 15,	30 0,7	700		
			- 55,90) 63,	80 - 2.	,300	3 N	
			- 63,60) -0,:	540 60	,20		

$$\mathbf{L}_{a,1} = \begin{bmatrix} -11,85 & 14,51 & -2,308 \\ -33,05 & 41,89 & -7,857 \\ -5,742 & 6,588 & -0,684 \\ -114,6 & 147,1 & -29,35 \end{bmatrix} \mathbf{L}_{a,2} = \begin{bmatrix} -11,93 & 14,94 & -2,608 \\ -32,74 & 42,39 & -8,560 \\ -5,603 & 6,551 & -0,772 \\ -111,5 & 145,9 & -31,01 \end{bmatrix}$$
$$\mathbf{L}_{a,3} = \begin{bmatrix} -12,04 & 16,44 & -3,845 \\ -31,27 & 44,01 & -11,34 \\ -5,142 & 6,473 & -1,112 \\ -101,2 & 142,8 & -37,49 \end{bmatrix}$$

Gain integralnya,

 $\overline{\mathbf{F}}_{1} = \begin{bmatrix} 513,33 & -110,72 & -10,078 \end{bmatrix}$ $\overline{\mathbf{F}}_{2} = \begin{bmatrix} 427,20 & -94,193 & -8,7106 \end{bmatrix}$ $\overline{\mathbf{F}}_{3} = \begin{bmatrix} 418,11 & -94,128 & -9,1363 \end{bmatrix}$ $\mathbf{F}_{a,1} = \begin{bmatrix} -67,084 & -39,403 & 385,70 \end{bmatrix}$ $\mathbf{F}_{a,2} = \begin{bmatrix} -69,309 & -37,032 & 386,10 \end{bmatrix}$ $\mathbf{F}_{a,3} = \begin{bmatrix} -60,042 & -27,020 & 328,43 \end{bmatrix}$

Sedangkan atenuasi yang didapatkan dari hasil minimisasi ($\gamma + \mu$) dan minimisasi ($\gamma_a + \mu_a$) adalah:

 $\gamma = 0,0884$ $\gamma_a = 0,5058$ $\mu = 0,0534$ $\mu_a = 0,4878$

4.7. Diagram Simulink Kontroler dan FTC untuk Simulasi dan Implementasi

Hasil perancangan kontroler pada Subbab 3.3 dan perancangan FTC pada Subbab 3.5 terlebuh dahulu disimulasikan menggunakan MATLAB Simulink. Model Simulink hasil rancangan dapat dilihat pada Gambar 3.8. Warna hijau pada model Simulink menunjukkan diagram untuk kontrol nominal, warna kuning untuk *observer* dan kompensasi kesalahan aktuator, sedangkan warna biru untuk *observer* dan

kompensasi kesalahan sensor. Semua variabel $x_1(t)$, $x_2(t)$, $x_3(t)$ dan $x_4(t)$, sinyal kontrol, derajat keanggotaan, L_2 -gain ditampilkan pada scope masing-masing. Selain itu untuk mengetahui Integral Absolute Error (IAE) dari posisi kereta dapat dilihat pada scope IAE.

Setelah melakukan simulasi dan didapatkan hasil yang cukup baik, maka sistem kontrol hasil desain diuji pada *plant* nyata, "*Digital Pendulum Mechanical Unit* 33-200" [6]. Diagram Simulink yang digunakan dapat dilihat pada Gambar 3.9. *State* yang terukur, yaitu posisi kereta $(x_1(t))$ dan posisi sudut pendulum $(x_2(t))$ dibaca melalui blok sensor posisi kereta dan sensor posisi sudut pendulum. Sedangkan *state* yang tidak terukur, yaitu kecepatan kereta $(x_3(t))$ dan kecepatan sudut pendulum $(x_4(t))$ didapat dengan mengambil turunan pertama dari posisi kereta $(x_1(t))$ dan posisi sudut pendulum $(x_2(t))$.

Gambar 3.9 menunjukkan diagram Simulink untuk implementasi. Sinyal kontrol yang didapat dari kontroler masuk ke dalam blok DAC. Pada blok DAC, sinyal kontrol yang berupa gaya dalam satuan Newton dikonversikan ke dalam tegangan dalam satuan Volt. Hubungan antara gaya dan tegangan diasumsikan linear, yaitu V = 1/9 u. Karena DAC yang digunakan bekerja pada ±2,5 Volt, maka tegangan yang digunakan dibatasi pada ±1,9 Volt. Isi dari masing-masing blok pada Gambar 3.8 secara detil dapat dilihat pada Lampiran B.3.





BAB IV HASIL PENGUJIAN SIMULASI DAN IMPLEMENTASI

Pada Bab ini akan dijelaskan hasil simulasi dan implementasi rancangan kontrol nominal dan *observer* pada sistem pendulum kereta. Simulasi menggunakan Matlab 9.5.1 dan implementasi menggunakan "*Digital Pendulum Mechanical Unit* 33-200" dari *Feedback Instrument* Ltd. Simulasi dilakukan dengan berbagai kondisi kesalahan, yaitu bias berdasarkan dinamikanya (sinus dan random Gaussian) karakteristiknya (*abrupt fault, incipient fault* dan *intermittent fault*)) dan multiplikatif.

4.8. Hasil Simulasi

Pada subbab ini akan dijelaskan simulasi kondisi tanpa kesalahan (*fault-free case*) dan kondisi dengan kesalahan sensor dan aktuator (*faulty case*). Simulasi menggunakan Matlab 9.5.1.

4.8.1. Hasil Simulasi Kontrol Nominal dengan Berbagai Kondisi Awal tanpa Kesalahan (*Fault-Free Case*)

Pada bagian ini simulasi dilakukan dengan dua kondisi awal yaitu pada sudut 0,2 radian dan sudut 0,4 radian. *State feedback gain* telah ditentukan pada perancangan di Bab III Perancangan. Kondisi awal untuk posisi kereta, dan kecepatan kereta adalah nol, sedangkan kondisi awal kecepatan sudut kereta ditentukan agar sinyal kontrol memenuhi batasan yang diinginkan. Gambar 4.1 menunjukkan grafik respon posisi kontrol nominal tanpa kesalahan dengan berbagai kondisi awal.



Gambar 4.1 Respon Posisi Kontrol Nominal Hasil Simulasi

Pada Gambar 4.1 terlihat untuk kondisi awal sudut pendulum 0,2 radian memiliki memiliki respon yang lebih baik dibandingkan kondisi awal sudut pendulum 0,4 radian. Besar selisih amplitudo absolut masing-masing untuk kondisi awal 0,2 radian dan 0,4 radian adalah 0,00002 dan 0,000021. Beda fasa untuk kedua kondisi awal sangat kecil sehingga respon posisi dikatakan memenuhi trayektori yang diinginkan pada detik ke 3,5 dan seterusnya. Pada waktu sekitar 0,2 detik, kereta bergerak ke menuju posisi -0,07 m dan -0,09 m untuk masing-masing sudut awal 0,2 dan 0,4 radian. Kondisi ini terjadi karena kereta berusaha untuk memaksa posisi batang pendulum menuju sudut nol radian dari sudut awal yang diberikan. Nilai *Integral Absolute Error* (IAE) adalah 0,1256 dan 0,1286 untuk masing-masing sudut awal 0,2 dan 0,4 radian. Terlihat nilai IAE sangat kecil sehingga posisi kereta dapat mengikuti sinyal referensi dengan sangat baik.

Gambar 4.2 menunjukkan respon sudut pendulum dengan sudut awal yang bervariasi. Untuk dua kondisi awal yang berbeda yaitu sudut awal 0,2 dan 0,4 radian, mengalami penurunan nilai dan melonjak ke nilai untuk masing-masing sudut awal 0,04 dan 0,07 radian. Pada saat kondisi *steady state*, terdapat osilasi sekitar 0,004 radian. Osilasi ini diakibatkan oleh *tracking* yang dilakukan oleh kereta. Secara umum respon sudut pendulum memiliki performa yang baik.

Gambar 4.3 menunjukkan sinyal kontrol untuk dua kondisi awal yang berbeda. Terlihat untuk sudut awal pendulum 0,4 radian memiliki sinyal kontrol maksimal yang lebih besar dibandingkan sudut awal 0,2 radian yaitu sebesar -800 N, sedangkan untuk sudut awal 0,2 radian memiliki sinyal kontrol -310 N.



Gambar 4.2 Respon Sudut Pendulum dengan Berbagai Kondisi Awal Sudut Pendulum Hasil Simulasi

Besar *overshoot* sinyal kontrol untuk sudut awal 0,2 radian dan 0,4 radian hampir sama yaitu 20 N. Sinyal kontrol ini tidak memenuhi batasan yang telah dirancang, yaitu 17,5 N. Untuk memenuhi batasan *input-output* diberikan kondisi awal pada kecepatan sudut. Pada simulasi ini, kondisi awal yang diberikan adalah [0 0,2 0 -0,77] dan [0 0,4 0 - 1,53].



Gambar 4.3 Sinyal Kontrol dengan Berbagai Kondisi Awal Sudut Pendulum Hasil Simulasi

Gambar 4.4 menunjukkan grafik kontrol nominal *tracking* dengan kondisi awal pada sudut pendulum dan kecepatan pendulum. Besar selisih maksimum amplitudo untuk kedua kondisi awal hampir sama saat tidak diberi kecepatan sudut pendulum, 0,00002 dan 0,000021.



Gambar 4.4 Kontrol Nominal dengan Berbagai Kondisi Awal Sudut Pendulum dan Kecepatan Sudut Pendulum Hasil Simulasi

Besar IAE untuk kondisi awal sudut pendulum 0,2 radian dan kecepatan sudut -0,77 rad/s adalah 0,1194, dan untuk kondisi awal sudut pendulum 0,4 radian dan kecepatan sudut -1,43 rad/s adalah 0,1239. Nilai ini lebih kecil dibanding nilai IAE grafik pada Gambar 4.1, sehingga disimpulkan respon posisi ini memiliki performa yang lebih baik dibandingkan tidak memiliki kondisi awal pada kecepatan sudut pendulum.

Gambar 4.5 menunjukkan respon sudut pendulum untuk berbagai kondisi awal. Untuk kedua kondisi awal pendulum dapat distabilkan sekitar 2 detik. Untuk sudut awal 0,2 radian memiliki *undershoot* pada nilai sekitar 0.008 radian, sedangkan untuk kondisi awal 0,4 radian tidak memiliki *undershoot*. Pada detik ke-2 dan seterusnya terjadi osilasi sudut pendulum di sekitar 0,004 radian.



Gambar 4.5 Respon Sudut Pendulum dengan Berbagai Kondisi Awal Sudut Pendulum dan Kecepatan Sudut Hasil Simulasi

Gambar 4.6 menunjukkan sinyal kontrol untuk kondisi awal sudut pendulum 0,2 radian, kecepatan sudut -0,77 rad/s dan sudut pendulum 0,4 radian, kecepatan sudut -1,43 rad/s. Untuk sudut awal 0,2 radian memiliki *overshoot* 0,9 N sedangkan untuk sudut awal 0,4 radian memiliki *overshoot* 7 N. Sinyal kontrol berosilasi pada detik ke-6 dan seterusnya dengan nilai osilasi di sekitar 1,5 N, hal ini dikarenakan *tracking* sinyal trayektori yang digunakan sinus, osilasi sinyal kontrol membentuk sinus.

Sinyal kontrol maksimal untuk kedua kondisi adalah 7 N dan -14 N. Besarnya sinyal kontrol ini sudah sesuai dengan batasan *input-output* yang ditetapkan yaitu maksimal sinyal kontrol yang dihasilkan adalah sebesar 17,5 N. Sinyal kontrol yang terlalu besar mengakibatkan sistem tidak dapat diimplementasikan. Pemberian nilai kondisi awal pada

kecepatan sudut mengakibatkan sinyal kontrol yang dihasilkan lebih kecil dan sesuai dengan kriteria perancangan yang telah diterapkan.





Tabel 4.1 menunjukkan nilai-nilai keseluruhan dari perbandingan hasil respon untuk berbagai kondisi awal. Terdapat empat kriteria respon yaitu undershoot posisi kereta, undershoot posisi sudut pendulum, sinyal kontrol minimal dan sinyal kontrol maksimal.

Kriteria	Kondisi Awal					
Kincha	[0 0,2 0 0]	[0 0,4 0 0]	[0 0,2 0 -0,77]	[0 0,4 0 -1,53]		
Undershoot Posisi Kereta (m)	0,07	0,09	0,01	0,06		
Undershoot Posisi Sudut Pendulum (rad)	0	0	0,008	0		
Sinyal Kontrol Minimal (N)	-310	-800	-2	-14		
Sinyal Kontrol Maksimal (N)	20	20	6	6		
IAE	0,1256	0,1286	0,1194	0,1239		

Tabel 4.1 Perbandingan Respon Simulasi Berbagai Kondisi Awal

Kriteria respon dari Tabel 4.1 adalah respon kontrol nominal tanpa *observer*. Ketika *observer* terkoneksi dengan kontrol nominal terdapat perbedaan respon transien. Kondisi awal yang diberikan pada *plant* mempengaruhi respon pada kontrol nominal. Perbedaan ini dapat dilihat pada Gambar 4.7. Grafik pada Gambar 4.7 menunjukkan pengaruh kondisi awal pada *observer*.





Ketika observer telah terkoneksi dengan kontrol nominal. pemberian kondisi awal pada sudut 0 radian dan kecepatan sudut 0 rad/s memiliki respon dengan undershoot paling kecil, hal ini dikarenakan kondisi awal *observer* sama dengan kondisi awal pada *plant*, yaitu [0 0 0 0]. Sedangkan ketika *plant* diberi kondisi awal pada sudut 0,2 radian dan kecepatan sudut -0,77 rad/s terdapat undershoot pada -0,1 radian. Respon paling buruk terjadi ketika pemberian kondisi awal paling besar vaitu sudut 0,4 radian dan kecepatan sudut -1,43 rad/s, terdapat undershoot pada -0,17 radian. Hal ini terjadi karena observer memerlukan waktu untuk menyesuaikan dengan kondisi awal dari plant, waktu penyesuaian ini mengakibatkan respon transien berubah. Untuk kriteria Integral Absolute Error (IAE), plant dengan kondisi awal sudut 0 radian memiliki IAE 0,1187, kondisi awal sudut 0,2 radian memiliki IAE 0,1969 dan kondisi awal sudut 0,4 radian memiliki IAE 0,2940, semakin besar kondisi awal yang diberikan semakin besar nilai IAE, yang berarti respon yang didapatkan semakin buruk.

Tabel 4.2	Respon	Pengaruh	Pemberian	Kond1s1	Awal	pada	Plant	dengan
Observer ya	ng Terko	oneksi Kon	trol Nomina					

Vritorio	Kondisi Awal				
Kinella	[0 0 0 0]	[0 0,2 0 -0,77]	[0 0,4 0 -1,53]		
Undershoot Posisi Kereta (m)	0,01	0,1	0,17		
IAE	0,1187	0,1969	0,2940		

4.8.2. Hasil Simulasi Kontrol Nominal dengan Berbagai Kondisi Kesalahan (*Faulty Case*)

Pada sub-subbab ini dibahas kontrol nominal dengan berbagai kondisi kesalahan yang terdapat pada sensor dan aktuator, secara umum kesalahan yang diuji adalah kesalahan aditif dan multiplikatif. Kesalahan aditif terdiri atas sinyal random dan sinus. Selain itu terdapat kesalahan menurut proses waktunya yaitu, *intermittent fault, incipient fault dan abrupt fault.* Untuk kesalahan multiplikatif terdiri atas *loss of measurement* dan *lost of effectiveness.* Semua kesalahan disimulasikan pada kondisi awal [0 0,2 0 -0,77], karena memiliki respon dengan *observer* yang cukup baik.

4.8.2.1. Simulasi Faulty-Case Sensor dengan Kesalahan Bias (Aditif) Kesalahan bias adalah kesalahan yang bersifat aditif (pertambahan) terhadap keluaran, seperti yang telah dijelaskan pada subbab 2.8. Pengujian pertama terhadap sinyal sinus, dengan kesalahan sinus direpresentasikan dalam persamaan matematis berikut:

$$f_s(t) = 0.05k * \sin(0.5\pi)$$

$$k = 1.2.3$$
(4.1)

Gambar 4.8 memperlihatkan grafik kontrol nominal dengan kesalahan aditif. Semakin besar kesalahan sinus, semakin buruk respon yang didapatkan. Dari grafik terlihat bahwa pemberian kesalahan sinus menyebabkan penyimpanan pada sinyal referensi. Besarnya penyimpangan dapat dilihat pada nilai IAE. Untuk kesalahan sensor $0,05\sin(0,5\pi)$ nilai IAE sebesar 0,1995, untuk kesalahan $0,10\sin(0,5\pi)$ nilai IAE sebesar 0,2018, untuk kesalahan $0,15\sin(0,5\pi)$ nilai IAE sebesar 0,2048.



Gambar 4.8 Faulty-case Sensor Kesalahan Bias Sinus Hasil Simulasi

Sedangkan estimasi kesalahan dapat dilihat pada Gambar 4.9. Pada grafik terlihat *observer* mampu mengestimasi kesalahan yang terjadi, kesalahan yang diestimasi adalah 0,15sin(0,5 π). Waktu untuk menyesuaikan dengan kesalahan yang terjadi adalah 7 s. Setelah detik ke-7 *observer* mengestimasi kesalahan dengan baik, hal ini ditunjukkan dengan nilai IAE yang stabil pada detik ke-7 dan seterusnya, yaitu 0,2048. *Observer* perlu menyesuaikan dengan kondisi awal dari *plant*.



Gambar 4.9 Estimasi Kesalahan Sensor Berbentuk Sinus Hasil Simulasi

Sedangkan Gambar 4.10 memperlihatkan error estimasi kesalahan, sesuai perancangan pada Subbab 3.4 error estimasi kesalahan adalah kesalahan sebenarnya dikurangi dengan estimasi kesalahan. Error estimasi terbesar adalah pada detik ke-1 dengan *absolute error* sekitar 0,07m. Pada detik ke-7 dan seterusnya error kesalahan berosilasi pada nilai 0,015m.



Gambar 4.10 Error Estimasi Kesalahan Sensor Sinus Hasil Simulasi

Simulasi selanjutnya adalah kesalahan sensor berbentuk sinyal random. Sinyal random dalam hal ini berbentuk *Gaussian*. Kesalahan dibuat dalam varian yang berbeda-beda. Secara umum bentuk kesalahannya adalah:

$$f_{s}(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x-\mu)^{2}}{2\sigma^{2}}\right]$$

$$\sigma = 0,001 + 0,001k, \mu = 0,00$$

$$k = 0,1,2$$
(4.2)

Gambar 4.11 memperlihatkan respon kontrol nominal setelah diberi kesalahan berbentuk random Gaussian. Dari grafik terlihat untuk kesalahan yang paling besar memperlihatkan respon yang paling dengan penyimpangan yang paling besar. Nilai IAE untuk kesalahan dengan varian 0,001 adalah 0,3806, sedangkan nilai IAE untuk kesalahan dengan varian 0,002 dan 0,003 berturut-turut adalah 0,7379 dan 1,406. Etror yang terjadi pada kesalahan random Gaussian terpengaruh pada besarnya varian, hal ini dikarenakan nilai IAE adalah *integral* luasan dibawah kurva, sehingga dengan varian yang besar luasan error dibawah kurva lebih besar atau lebih luas.



Gambar 4.11 Faulty-case Sensor dengan Kesalahan Bias Random Gaussian Hasil Simulasi

Estimasi kesalahan sensor diperlihatkan pada Gambar 4.12. Pada grafik ini terlihat *observer* mampu mengestimasi kesalahan sensor dengan baik pada detik ke-7 dan seterusnya. Hal ini terlihat dari respon yang hampir sama kesalahan yang terjadi dengan estimasi kesalahan. Untuk kesalahan bentuk random Gaussian terdapan respon transien pada setiap transisi besarnya kesalahan yang terjadi, hal ini terlihat pada zoom Gambar 4.12. *Observer* mengikuti kesalahan mirip dengan respon orde kesatu.



Gambar 4.12 Estimasi Kesalahan Sensor Random Gaussian Hasil Simulasi

Error estimasi kesalahan diperlihatkan pada Gambar 4.13. Error estimasi memperlihatkan amplitudo yang berbeda-beda dengan periode yang singkat. Hal ini menunjukkan penyesuain *observer* terhadap kesalahan yang terjadi, yaitu berbentuk sinyal random. Kesalahan berbentuk sinyal random Gaussian mirip dengan noise yang terjadi pada *plant*. Hal ini dikarenakan sifatnya acak dan berlangsung dengan periode tertentu secara acak.



Gambar 4.13 Error Estimasi Kesalahan Sensor Random Gaussian Hasil Simulasi

Tabel 4.3 Respon Kontrol Nominal Terhadap Kesalahan Sensor

 Berbentuk Sinus dan Random

	Kriteria			
Jenis Kesalahan	Error Estimasi Maksimum (m)	IAE		
$f_s(t) = 0.05 \sin(0.5\pi t)$	0,07	0,1995		
$f_s(t) = 0.10 \sin(0.5\pi t)$	0,07	0,2018		
$f_s(t) = 0.15 \sin(0.5\pi t)$	0,07	0,2048		
7				
Random Gaussian, mean = 0, varian = 0,001	0,45	0,3806		
Random Gaussian, mean = 0, varian = 0,002	0,46	0,7379		
Random Gaussian, mean $= 0$, varian $= 0,003$	0,47	1,4060		

Kesalahan sensor selanjutnya adalah aditif dengan jenis kesalahan *intermittent*, *intermittent* dan *abrupt*. Kesalahan *abrupt* mirip dengan kesalahan fungsi step yang berlangsung kontinu dimulai pada waktu tertentu. Secara umum fungsi kesalahan *abrupt* yang diuji adalah:

$$f_s(t) = 0.05k * u(t - 10)$$

k = 1.2.3

Respon dari kontrol nominal setelah diberi kesalahan ini dapat dilihat pada Gambar 4.14. Dari grafik terlihat keluaran sistem bergeser dari travektori yang diinginkan.

(4.3)



Gambar 4.14 Faulty Case Sensor dengan Kesalahan Bias Abrupt Hasil Simulasi

Semakin besar amplitudo kesalahan, semakin besar pergeseran yang terjadi. Nilai IAE berturut-turut untuk kesalahan dengan amplitudo 0,05 0,10 dan 0,15m adalah 0,2561, 0,3960 dan 0,5380. Pada detik dimana kesalahan *abrupt* terjadi, yaitu detik ke-10, terjadi *overshoot*. *Overshoot* tertinggi terjadi pada kesalahan dengan amplitudo paling besar, yaitu 0,24m. Sedangkan *overshoot* untuk kesalahan amplitudo 0,05 dan 0,10m adalah 0,16m dan 0,09m.

Estimasi kesalahan dapat dilihat pada Gambar 4.15. Kesalahan *abrupt* terjadi tiba-tiba pada detik ke-10 yang berlangsung secara kontinu. Pada detik ke-1 sampai ke-10 *observer* menyesuaikan diri dengan kondisi awal dari *plant* sehingga tampak terjadi *overshoot*. Respon transien juga terjadi pada detik ke-10 dimana *observer* memerlukan waktu untuk mengestimasi kesalahan sampai *steady-state*.





Kesalahan berdasarkan karakteristiknya, selanjutnya adalah kesalahan *intermittent*. Kesalahan *intermittent* hampir sama dengan kesalahan inpuls, dimana kesalahan muncul secara tiba-tiba dan berhenti secara tiba-tiba dengan periode yang sangat singkat. Fungsi kesalahan ini dirumuskan sebagai berikut:

$$f_s(t) = 0.05k * u((t-10) - (t-10,1))$$

$$k = 1,2,3$$
(4.4)

Periode kesalahan *intermittent* adalah 0,1s. Respon kontrol nominal terhadap kesalahan ini dapat dilihat pada Gambar 4.16. Kesalahan *intermittent* berakibat penyimpangan secara tiba-tiba respon sistem lup tertutup terhadap keadaan *steady state*-nya. Respon paling

DAFTAR PUSTAKA

- [1] Hassan Noura, Fault Tolerant Control Systm : Design and Its Application, United Arab Emrates: United Arab Emrates University, 2009.
- [2] M. Blanke, M. Kinneart, J. Lunze, and M. Staroswiecki, *Diagnosis* and Fault-Tolerant Control, 2nd ed. Berlin/Heidelberg, German: Springer-Verlag, 2006.
- [3] K. Ogata, Modern Control Engineering. New Jersey: Prentice-Hall, 1997.
- [4] Montadher Sami, "Active Fault-Tolerant Control of Nonlinear Systems with Wind Turbine Application," University of Hull, Hull, PhD Thesis 2012.
- [5] A. Benzaouia and A. El Hajjaji, *Advanced Takagi-Sugeno Fuzzy* Systems. Switzerland: Springer, 2014.
- [6] "Control in a MATLAB Environment," *Feddback Instruments Ltd*, 2004.
- [7] Andir Ashfahani and Trihastuti Agustinah, "Desain Fuzzy Tracking Controller pada Pendulum Terbalik dengan Memperhitungkan Model Friksi," in *SITIA*, Surabaya, 2008.
- [8] Trihastuti Agustinah, Achmad Jazidie, and Mohammad Nuh, "Fuzzy Tracking Control Based on H-infinity Performance for Nonlinear Systems," WSEAS Transactions on Systems and Control, pp. 393-403, 2011.
- [9] M. Sami and R. J. Patton, "Active Fault Tolerant Control for Nonlinear Systems with Simultaneous Actuator and Sensor Faults," *International Journal Control, Automation, and Systems*, vol. 11(6), pp. 1149-1161, 2013.
- [10] Marcin Witczak, Fault Diagnosis and Fault-Tolerant Control Strategies for Non-Linear Systems. New York: Springer, 2013.
- [11] Krishna K. Busawon and Pousga Kabore, "On the design of integral and proportional integral observer," *Proceedings of the American Control Conference*, 2000.
- [12] Atef Kheder and Kamel Benothman, "Active Fault Tolerant Control (FTC) Design for Takagi-Sugeno Fuzzy Systems with Weighting Functions Depending on the FTC," *International Journal of*

Computer Science Issues, vol. 8, no. 3, pp. 1694-0814, May 2011.

- [13] S. Boyd, L. El Ghouni, and Balakrishnan, "Linear Matrix Inequality in System and Control Theory," *SIAM Philadelphia*, 1994.
- [14] Usha Mahato, "Low Gain Controller Design with Regional Pole Placement Constraint," National Institute of Technology Rourkela, Odisha, Master Thesis 2013.
- [15] R. Isermann, Fault Diagnosis Systems (An Introduction from Fault Detection to Fault Tolerant). New York: Kluwer Academic Publisher, 2006.
- [16] Halim Alwi, Christoper Edwards, and Chee Pin Tan, "Fault Detection and Fault Tolerant Control Using Sliding Modes," pp. 7-28, 2011.
- [17] Kemin Zhou, *Essentials of Robust Control*. New Jersey: Prentice-Hall, 1999.
- [18] S. Boyd, L. El Ghaouni, E. Feron, and V. Balakrishnan, *Linear Matrix Inequality in System and Control Theory*. Philadelphia: SIAM, 1994.
- [19] Ke Zhang, Bin Jiang, and Vincent C., "Adaptive Observer-based Fast Fault Estimation," in *International Journal of Control*, *Automation, and Systems*, 2008, pp. vol. 6, no. 3, pp. 320-326.



BAB V PENUTUP

Dari hasil pengujian simulasi dan implementasi, kontrol nominal yang dirancang dapat membuat Sistem Pendulum-Kereta mengikuti sinyal referensi yang diinginkan, yaitu sinyal berbentuk sinus dengan tetap membertahankan batang pendulum di sekitar 0 radian. Selain itu kontrol nominal memenuhi spesifikasi batasan input-output dan LMI *region* yang telah dirancang. Dari hasil simulasi dan implementasi, Kontrol toleransi kesalahan (FTC) yang dirancang mampu mengompensasi kesalahan sensor, aktuator, atau sensor dan aktuator secara simultan dengan baik. Hal ini terlihat dari error estimasi kesalahan sensor dan aktuator yang kecil dan nilai IAE dengan kompensasi FTC lebih kecil dibandingkan nilai IAE tanpa kompensasi FTC. Observer untuk kesalahan sensor dan aktuator juga memenuhi performa H_{α} dengan nilai pelemahan dibawah dari γ yang dirancang saat teriadi kesalahan.

LAMPIRAN A

A.1 Penurunan Persamaan State Sistem Pendulum-Kereta

Persamaan *state* Sistem Pendulum-Kereta dapat diperoleh dari analisis gerak sesuai hukum kedua Newton. Persamaan gerak translasi pada bidang horizontal adalah

$$\sum F_x = ma_x$$

(A.1)

(A.2)

(A.5)

Dapat dilihat bahwa gaya-gaya yang bekerja pada bidang horizontal adalah gaya kontrol u dan gaya gesek T_c . Massa keseluruhan sistem (m) adalah jumlah dari massa kereta (m_c) dan massa pendulum (m_p) . Percepatan pusat massa pada bidang horizontal adalah

$$a_x = (x_1 - l \sin x_2)''$$

Dari (A.1) dan (A.2), diperoleh(2.1) yang dapat diuraikan sebagai berikut:

$$u - T_c = (m_c + m_p)(x_1 - l\sin x_2)"$$

$$u - T_c = (m_c + m_p)(x_1' - x_2' l\cos x_2)'$$

$$u - T_c = (m_c + m_p)(x_1'' - x_2'' l\cos x_2 + (x_2')^2 l\sin x_2)$$
(A.3)

Persamaan gerak translasi pada bidang vertikal dapat ditulis sebagai berikut:

$$\sum F_y = ma_y \qquad (A.4)$$

Dari Gambar 2.4 dan Gambar 2.5, dapat dilihat bahwa gaya-gaya yang bekerja pada bidang vertikal adalah gaya normal V dan berat sistem, yaitu $(m_c + m_p)g$. Percepatan pusat massa pada bidang vertikal adalah

$$a_v = (l \cos x_2)^{t}$$

Dari (A.4) dan (A.5), diperoleh (2.2) yang dapat diuraikan sebagai berikut:

$$V = (m_c + m_p)g = (m_c + m_p)(l\cos x_2)''$$

$$V = (m_c + m_p)g = (m_c + m_p)(-lx_2'\sin x_2)'$$

$$V = (m_c + m_p)g + (m_c + m_p)(-lx_2''\sin x_2 - l(x_2')^2\cos x_2)$$

$$V = (m_c + m_p)(g - lx_2''\sin x_2 - l(x_2')^2\cos x_2)$$
(A.6)

Selanjutnya, persamaan gerak rotasi dapat ditulis sebagai berikut:

 $\sum \tau = J\alpha$

(A.7)

(A.9)

(A.11)

dengan α adalah percepatan sudut dan *t*adalah torsi yang bekerja pada sistem. (A.7) dapat dijabarkan sehingga diperoleh persamaan gerak rotasi sesuai dengan (2.3), yaitu

$$\frac{(u - T_c)l\cos x_2 + Vl\sin x_2 - D_p = Jx_2"}{(u - T_c)l\cos x_2 + Vl\sin x_2 - f_p x_4 = Jx_2"}$$
(A.8)

dengan D_p adalah momen gesek akibat gerak putar pendulum.

Dinamika *state* x_4 dapat diperoleh dengan melakukan substitusi (A.6) ke (A.8) sebagai berikut:

$$Jx_2'' = (u - T_c)l\cos x_2 - f_p x_4$$

$$+(m_c + m_p)(g - lx_2"\sin x_2 - l(x_2')^2 \cos x_2)l\sin x_2$$

Dengan mendefinisikan $\mu = (m_c + m_p)l$, maka diperoleh

$$Jx_{4}' = (u - T_{c})l\cos x_{2} - f_{p}x_{4} + (m_{c} + m_{p})l(g\sin x_{2})$$

$$-lx_{4}'\sin^{2} x_{2} - l(x_{2}')^{2}\cos x_{2}\sin x_{2})$$

$$Jx_{4}' = (u - T_{c} - \mu x_{4}^{2}\sin x_{2})l\cos x_{2} + \mu g\sin x_{2}$$

$$-\mu lx_{4}'\sin^{2} x_{2} - f_{p}x_{4}$$

$$x_{4}'(J + \mu l\sin^{2} x_{2}) = (u - T_{c} - \mu x_{4}^{2}\sin x_{2})l\cos x_{2} + \mu g\sin x_{2} - f_{p}x_{4}$$

$$x_{4}' = \frac{(u - T_{c} - \mu x_{4}^{2}\sin x_{2})l\cos x_{2} + \mu g\sin x_{2} - f_{p}x_{4}}{J + \mu l\sin^{2} x_{2}}$$
(A.10)

Dinamika *state* x_3 dapat diperoleh dari (A.3) sebagai berikut:

$$u - T_c = (m_c + m_p)(x_3' - x_4' l \cos x_2 + (x_4)^2 l \sin x_2)$$

 $(m_c + m_p)x_3' = u - T_c - \mu x_4^2 \sin x_2 + \mu x_4' \cos x_2$

Dengan mendefinisikan $a = l^2 + J/(m_c + m_p)$ danmelakukan substitusi (A.10) ke (A.11), maka akan diperoleh

$$(m_{c} + m_{p})x_{3}' = u - T_{c} - \mu x_{4}^{2} \sin x_{2}$$

$$+ \frac{\mu d \cos^{2} x_{2}(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$+ \frac{\mu \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d \sin^{2} x_{2})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$+ \frac{\mu d \cos^{2} x_{2}(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}' = \frac{(J + \mu d)(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu d \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})(J + \mu d \sin^{2} x_{2})$$

$$x_{3}' = \left(l^{2} + \frac{J}{(m_{c} + m_{p})}\right) \frac{(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu l \sin^{2} x_{2}}$$

+
$$\frac{l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

$$x_{3}' = \frac{a(u - T_{c} - \mu x_{4}^{2} \sin x_{2}) + l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

Dari (A.10) dan (A.12), dapat diperoleh bentuk persamaan *state* Sistem Pendulum-Kereta sepertipada (2.4).

(A.12)

Momen inersia dari Sistem Pendulum-Kereta diperoleh dari hasil analisis momen inersia yang diilustrasikan pada Gambar A.1. Dari gambar tersebut, m_{pw} adalah massa beban pendulum (kg), m_{ps} adalah massa batang pendulum (kg), l_p adalah panjang batang pendulum (m), l_{po} adalah jarak antara titik pusat massa batang pendulum dengan sumbu putar batang pendulum (m), l_c adalah panjang beban pendulum (m), l_{co} adalah jarak antara titik pusat massa beban pendulum dengan sumbu putar batang pendulum (m), r_p adalah jari-jari batang pendulum (m), dan r_c adalah jari-jari beban pendulum (m). Momen inersia pendulum dari sumbu rotasi pendulum ditunjukkan oleh (A.13).



Gambar A.2 Ilustrasi Momen Inersia Sistem Pendulum-Kereta

$$J_{p} = m_{pw} \left(\frac{1}{12} l_{c}^{2} + \frac{1}{4} r_{c}^{2} + l_{co}^{2} \right) + m_{ps} \left(\frac{1}{12} l_{p}^{2} + \frac{1}{4} r_{p}^{2} + l_{po}^{2} \right)$$

(A.13)

(A.14)

(A.15)

(A.16)

Untuk kesederhanaan perhitungan, pendulum diasumsikan sebagai silinder dengan ketebalan yang seragam dan memiliki panjang Lserta massa m_p . Posisi sumbu putar berada pada pusat massa kereta dan ujung batang pendulum sebagai kereta dapat dipandang sebagai titik massa M_c sedangkan pendulum sebagai titik massa M_p . Titik pusat massa sistem berada di dekat titik pusat massa kereta seperti diilustrasikan pada Gambar A.2. Jika momen inersia dari sumbu tertentu terhadap titik pusat massa dinyatakan dengan Jdan momen inersia dari sumbu yang paralel dengan sumbu sebelumnyadan dipisahkan oleh jarak sebesar d dinyatakan dengan J_p , maka hubungan antara J dan J_p adalah

$$J_p = J + Md$$

Dari Gambar A.2, titik pusat massa sistem dapat dirumuskan sebagai berikut:

$$M_{n}a = M_{c}b$$

a+b=L/2

Oleh karena itu, dari (A.14) dan (A.15), diperoleh momen inersia sistem terhadap pusat massa, yaitu

$$J = M_{p}a^{2} + M_{p}b^{2} + \frac{M_{p}}{12}L^{2}$$

dengan

$$a = \frac{LM_{c}}{2(M_{p} + M_{c})} \operatorname{dan} b = \frac{LM_{P}}{2(M_{p} + M_{c})}$$

Jika (A.16) dijabarkan akan didapat bentuk akhir persamaan momen inersia sistem seperti ditunjukkan pada (A.17).

$$J = \frac{L^2 m_p (4m_c + m_p)}{12(m_c + m_p)}$$
(A.17)

Sedangkan jarak pusat massa sistem terhadap sumbu putar adalah

$$l = \frac{m_p L}{2(m_c + m_p)} \tag{A.18}$$
Parameter	Simbol	Nilai
Masa kereta (kg)	m_c	1,12
Massa beban pendulum (kg)	m _{pw}	0,095
Massa batang pendulum (kg)	m_{ps}	0,025
Panjang rel (m)	R_l	1
Panjang batang pendulum (m)	l_p	0,402
Jarakpusat massa batang ke sumbu putar(m)	l_{po}	0,146
Panjang beban (m)	l_c	0,041
Jarak pusat massa beban ke sumbu putar (m)	l _{co}	0,347
Jari-jari beban pendulum (m)	r _c	0,02
Jari-jari batang pendulum (m)	r_p	0,006
Gaya gesek statis(N)	F_s	2,5316
Gaya gesek dinamis (Coulomb)(N)	F_c	2,2813
Gaya control maksimal(N)	<i>u_{max}</i>	17,5
Gaya kontrol minimal (N)	DZ_u	1,3792
Kecepatan minimal kereta(m/detik)	DZ	0,008
Kecepatan ayun minimal pendulum (rad/detik)	DZ_{pv}	0,034
Gaya gesek pendulum (kg.m ² /detik)	f_p	0,0001
Momen inersia pendulum (kg.m ²)	J_p	0,0139
Momen inersia sistem(kg.m ²)	J	0,0136
Jarak sumbu rotasi ke pusat massa sistem (m)		0,0168
Kec. kereta awal daerah 5 gaya gesek (m/detik)	x_c	0,3955
Gaya gesek saat kecepatan kereta $x_c(N)$	Y_c	2,3815
Tegangan kontrol maksimal (Volt)	V_m	2,5

 Tabel A.1
 Parameter-parameter Sistem Pendulum-Kereta

A.2. Elemen-elemen Matriks Linearisasi Model

Dari (3.1), dapat diperoleh

$$f_{1}(\mathbf{x}) = x_{3}$$

$$f_{2}(\mathbf{x}) = x_{4}$$

$$f_{3}(\mathbf{x}) = \frac{a(-T_{c} - \mu x_{4}^{2} \sin x_{2}) + l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

$$f_{4}(\mathbf{x}) = \frac{l \cos x_{2}(-T_{c} - \mu x_{4}^{2} \sin x_{2}) + \mu g \sin x_{2} - f_{p} x_{4}}{J + \mu l \sin^{2} x_{2}}$$
(A.19)

dan

$$h_{1}(\mathbf{x}, u) = 0$$

$$h_{2}(\mathbf{x}, u) = 0$$

$$h_{3}(\mathbf{x}, u) = \frac{au}{J + \mu l \sin^{2} x_{2}}$$
(A.20)
$$h_{4}(\mathbf{x}, u) = \frac{\mu l \cos x_{2}}{J + \mu l \sin^{2} x_{2}}$$
sehingga elemen matriks A diperoleh
$$\frac{\partial f_{1}(\mathbf{x})}{\partial x_{1}} = 0, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{2}} = 0, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{3}} = 1, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{4}} = 0$$

$$\frac{\partial f_{2}(\mathbf{x})}{\partial x_{1}} = 0, \frac{\partial f_{2}(\mathbf{x})}{\partial x_{2}} = 0, \frac{\partial f_{2}(\mathbf{x})}{\partial x_{3}} = 0, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{4}} = 1$$

$$\frac{\partial f_{3}(\mathbf{x})}{\partial x_{1}} = 0$$

$$\frac{\partial f_{3}(\mathbf{x})}{\partial x_{2}}$$

$$= \frac{g\mu l \cos^{2} x_{2} - a\mu x_{4}^{2} \cos x_{2} - l \sin x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

$$- \frac{2\mu l \cos x_{2} \sin x_{2}(l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4}))}{(J + \mu l \sin^{2} x_{2})^{2}}$$

$$\frac{\partial f_3(\mathbf{x})}{\partial x_3} = \frac{a}{J + \mu l \sin^2 x_2} \frac{\partial T_c}{\partial x_3}$$

$$\frac{\partial f_3(\mathbf{x})}{\partial x_4} = \frac{-2a\mu x_4 \sin x_2 - f_p l \cos x_2}{J + \mu l \sin^2 x_2}$$

$$\frac{\partial f_4(\mathbf{x})}{\partial x_1} = 0$$

$$\frac{\partial f_4(\mathbf{x})}{\partial x_2}$$

$$= \frac{\mu g \cos x_2 - \mu l x_4^2 \cos^2 x_2 - l \sin x_2 (-T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2}$$

$$-\frac{-2\mu l \cos x_2 \sin x_2 (l \cos x_2 (-T_c - \mu x_4^2 \sin x_2))}{(J + \mu l \sin^2 x_2)^2}$$

$$-\frac{-2\mu l \cos x_2 \sin x_2 (\mu g \sin x_2 - f_p x_4)}{(J + \mu l \sin^2 x_2)^2}$$

$$\frac{\partial f_4(\mathbf{x})}{\partial x_4} = -\frac{l \cos x_2}{J + \mu l \sin^2 x_2}$$
Sedangkan elemen matriks B adalah
$$\frac{\partial h_1(\mathbf{x}, u)}{\partial u} = 0$$

$$\frac{\partial h_3(\mathbf{x}, u)}{\partial u} = \frac{a}{J + \mu l \sin^2 x_2}$$

~

A.3. Penurunan Persamaan State Kompensator

Penurunan persamaan state kompensator didasari pada transformasi fungsi sinus dengan persamaan berikut:

A sin
$$(2\pi f)$$
 (A.21)
Dalam bentuk transformasi Laplace persamaan (A.21) dapat diubah
menjadi:

$$\frac{A2\pi f}{s^2 + (2\pi f)^2}$$
(A.22)

Dalam model kompensator, sistem lup tertutup mengikuti model referensi yang dituliskan dalam persamaan:

$$\dot{x}_{c}(t) = A_{c}x_{c}(t) + B_{c}e(t)$$
(A.23)

$$y_c(t) = x_c(t)$$

Untuk membentuk model referensi (A.23) persamaan (A.22) terlebih dahulu diubah menjadi bentuk:

$$\frac{Y_c(s)}{E_f(s)} = \frac{A2\pi f}{s^2 + (2\pi f)^2}$$
(A.24)

Dengan $Y_{\rm c}({\rm s}) = X_{\rm c}({\rm s})$ maka

$$S^{2}X_{c}(s) + (2\pi f)^{2}X_{c}(s) = A2\pi f E_{f}$$
(A.25)

$$\ddot{x}_{c}(t) + (2\pi f)^{2} x_{c}(t) = A2\pi f e_{f}(t)$$

Misalkan:

$$x_{c1}(t) = x_{c}(t) \Rightarrow \dot{x}_{c1}(t) = \dot{x}_{c}(t) = x_{c2}(t)$$

$$x_{c2}(t) = \dot{x}_{c}(t) \Rightarrow \dot{x}_{c2}(t) = \ddot{x}_{c}(t)$$
(A.26)

Persamaan (A.25) subtitusi ke persamaan (A.26) menjadi:

$$\dot{x}_{c2}(t) = -(2\pi f)^2 x_{c1}(t) + A2\pi f e_f(t)$$
(A.27)

Dengan membentuk Matriks Augmentasi persamaan (A.26) dan (A.27) menjadi:

$$\begin{bmatrix} \dot{x}_{c1}(t) \\ \dot{x}_{c2}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -(2\pi f)^2 & 0 \end{bmatrix} \begin{bmatrix} x_{c1}(t) \\ x_{c2}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ A2\pi f \end{bmatrix} e_f(t)$$
(A.28)

dengan

$$\mathbf{A}_{c} = \begin{bmatrix} \mathbf{0} & \mathbf{1} \\ -(2\pi f)^{2} & \mathbf{0} \end{bmatrix}, B_{c} = \begin{bmatrix} \mathbf{0} \\ A2\pi f \end{bmatrix}$$

A.4. Penurunan LMI Batasan Input-Output

Pre-multiplying dan *post-multiplying* LMI dengan matriks \mathbf{P}^{-1} akan didapat

$$\mathbf{P}^{-1} \begin{bmatrix} \mathbf{P} - \frac{\beta}{u_{\max}^{2}} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \end{bmatrix} \mathbf{P}^{-1} \ge \mathbf{P}^{-1} \mathbf{O} \mathbf{P}^{-1}$$

$$\mathbf{P}^{-1} - \mathbf{P}^{-1} \mathbf{K}_{j}^{T} \begin{bmatrix} \frac{u_{\max}^{2}}{\beta} \end{bmatrix} \mathbf{K}_{j} \mathbf{P}^{-1} \ge 0$$

$$\mathbf{E} - \mathbf{F}(\mathbf{H})^{-1} \mathbf{G} \ge 0$$
lengan
$$\mathbf{E} = \mathbf{P}^{-1}, \mathbf{F} = \mathbf{G}^{T} = \mathbf{P}^{-1} \mathbf{K}_{j}^{T}, \mathbf{H} = \frac{u_{\max}^{2}}{\beta}$$
(A.29)

(A.30)

dengan Schur Complement, LMI (A.29) ekuilaven dengan

$$\mathbf{L} \ge 0 \Leftrightarrow \begin{cases} \mathbf{H} \ge 0 \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} \ge 0 \end{cases}$$
$$\mathbf{L} = \begin{bmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{bmatrix} \ge 0$$
$$\mathbf{L} = \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{P}^{-1} \mathbf{K}_{j}^{T} \\ \mathbf{K}_{j} \mathbf{P}^{-1} & \frac{u_{\max}^{2}}{\beta} \end{bmatrix}$$

Subtitusi $\mathbf{Q} = \mathbf{P}^{-1}$ dan $\mathbf{Y}_j = \mathbf{K}_j \mathbf{P}^{-1}$ pada LMI (A.30) akan didapat

 $\begin{bmatrix} \mathbf{Q} & \mathbf{Y}_{j}^{T} \\ * & \frac{u_{\max}^{2}}{\beta} \end{bmatrix} \ge 0$ $\begin{bmatrix} -\mathbf{Q} & -\mathbf{Y}_{j}^{T} \\ * & -\frac{u_{\max}^{2}}{\beta} \end{bmatrix} < 0$

Pre-multiplying dan *post-multiplying* LMI dengan matriks \mathbf{P}^{-1} akan didapat

$$\mathbf{P}^{-1} \left[\mathbf{P} - \frac{\beta}{z_{\max}^{2}} \mathbf{C}_{z}^{T} \mathbf{C}_{z} \right] \mathbf{P}^{-1} \ge \mathbf{P}^{-1} \mathbf{O} \mathbf{P}^{-1}$$

$$\mathbf{P}^{-1} - \mathbf{P}^{-1} \mathbf{C}_{z}^{T} \left(\frac{z_{\max}^{2}}{\beta} \right) \mathbf{C}_{z}^{2} \mathbf{P}^{-1} \ge \mathbf{0}$$
(A.31)
$$\mathbf{E} - \mathbf{F}(\mathbf{H})^{-1} \mathbf{G} \ge \mathbf{0}$$
dengan
$$\mathbf{E} = \mathbf{P}^{-1}, \mathbf{F} = \mathbf{G}^{T} = \mathbf{P}^{-1} \mathbf{C}_{z}^{T}, \mathbf{H} = \frac{z_{\max}^{2}}{\beta}$$
dengan Schur Complement, LMI (A.31) ekuilaven dengan
$$\mathbf{L} \ge \mathbf{0} \Leftrightarrow \left\{ \mathbf{H} \ge \mathbf{0} \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} \ge \mathbf{0} \right\}$$

$$\mathbf{L} = \begin{bmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{bmatrix} \ge \mathbf{0}$$

$$\mathbf{L} = \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{P}^{-1} \mathbf{C}_{z}^{T} \\ \mathbf{C}_{z} \mathbf{P}^{-1} & \frac{z_{\max}^{2}}{\beta} \end{bmatrix}$$
Subtitusi $\mathbf{Q} = \mathbf{P}^{-1}$ dan $\mathbf{Y}_{j} = \mathbf{K}_{j} \mathbf{P}^{-1}$ pada LMI (A.32) akan didapat
$$\begin{bmatrix} \mathbf{Q} & \mathbf{Q} \mathbf{C}_{z}^{T} \\ \mathbf{x} & \frac{z_{\max}^{2}}{\beta} \end{bmatrix} \ge \mathbf{0}$$

$$\begin{bmatrix} -\mathbf{Q} & -\mathbf{Q} \mathbf{C}_{z}^{T} \\ \mathbf{x} & \frac{z_{\max}^{2}}{\beta} \end{bmatrix} < \mathbf{0}$$

A.5. Penurunan LMI Pencarian Observer Gain Kesalahan Sensor

Sesuai dengan persamaan (3.60) pertidaksamaan berikut adalah penurunan performansi H_{∞} dapat dituliskan kembali sebagai berikut:

$$\widetilde{\boldsymbol{e}}_{as}^{T}(t)(\widetilde{\mathbf{A}}_{s}^{T}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{P}}+\widetilde{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(\boldsymbol{p},\boldsymbol{p}))\widetilde{\boldsymbol{e}}_{as}(t)+\widetilde{\boldsymbol{e}}_{as}^{T}\widetilde{\mathbf{P}}\widetilde{\mathbf{N}}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{z}}$$

$$+\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{N}}^{T}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{P}}\widetilde{\boldsymbol{e}}_{as}(t)+\frac{1}{\gamma}\widetilde{\boldsymbol{e}}_{as}^{T}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p}\widetilde{\boldsymbol{e}}_{as}-\gamma\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{z}}<0$$
(A.33)

Pertidaksamaan (A.33) dapat diubah menjadi bentuk matriks yaitu dapat dituliskan sebagai berikut:

$$\begin{bmatrix} \widetilde{\boldsymbol{e}}_{as}^{T} & \widetilde{\boldsymbol{z}}^{T} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(\boldsymbol{p},\boldsymbol{p}) \overline{\mathbf{P}} + \overline{\mathbf{P}} \widetilde{\mathbf{A}}_{s}(\boldsymbol{p},\boldsymbol{p}) + \frac{1}{\gamma} \overline{\mathbf{C}}_{\boldsymbol{p}}^{T} \overline{\mathbf{C}}_{\boldsymbol{p}} & \overline{\mathbf{P}} \widetilde{\mathbf{N}}(\boldsymbol{p},\boldsymbol{p}) \\ & * & -\gamma \mathbf{I} \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{e}}_{as} \\ \widetilde{\boldsymbol{z}} \end{bmatrix} < 0$$
(A.34)

Secara implisit persamaan (A.34) dapat ditulis sebagai berikut:

$$\widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) + \frac{1}{\gamma}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p} \quad \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p)$$

$$* \qquad -\gamma \mathbf{I}$$

$$(A.35)$$

Persamaan (A.35) dapat dipisah menjadi dua bagian menjadi berikut:

$$\begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) & \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) \\ * & -\gamma \mathbf{I} \end{bmatrix} + \begin{bmatrix} \frac{1}{\gamma}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p} & \mathbf{0} \\ \mathbf{0} \end{bmatrix} < \mathbf{0} \quad (A.36)$$

Bagian kedua pada (A.36) dipisah menjadi

Dengan menggunakan teori Schur Complement dengan variabel berikut:

$$\mathbf{E} = \begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) & \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) \\ * & -\gamma \mathbf{I} \end{bmatrix}$$
$$\mathbf{G} = \mathbf{F}^{T} = \begin{bmatrix} \overline{\mathbf{C}}_{p}^{T} & \mathbf{0} \end{bmatrix}$$
$$\mathbf{H} = \begin{bmatrix} \gamma & \mathbf{0} \\ \mathbf{0} & \gamma \end{bmatrix}, \mathbf{H}^{-1} = \begin{bmatrix} \gamma^{-1} & \mathbf{0} \\ \mathbf{0} & \gamma^{-1} \end{bmatrix}$$

Sehingga dalam bentuk *Schur Complement* (A.38) dituliskan sebagai berikut:

(A.38)

 $\mathbf{E} - \mathbf{F}\mathbf{H}^{-1}\mathbf{G} < 0$

LMI (A.38) memiliki bentuk yang sama dengan LMI pada (A.39) sehingga LMI (A.38) ekuivalen dengan:

$$\mathbf{L} < \mathbf{0} \Leftrightarrow \begin{cases} \mathbf{H} < \mathbf{0} \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} < \mathbf{0} \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} < \mathbf{0} \\ \mathbf{L} = \begin{bmatrix} \mathbf{E} & \vdots & \mathbf{F} \\ \cdots & \cdots & \cdots \\ \mathbf{G} & \vdots & \mathbf{H} \end{bmatrix}$$

Sehingga bentuk pertidaksamaan (A.38) menjadi:

Pertidaksamaan (A.40) adalah pertidaksamaan bentuk Schur Complement dari pertidaksamaan (A.35). Dalam pertidaksamaan (A.40) masih terdapat dua variabel yang belum tersubtitusi yaitu:

 $\widetilde{\mathbf{A}}_{s}^{T}(p,p), \overline{\mathbf{P}}, \widetilde{\mathbf{N}}(p,p)$

dengan mensubtitusikan ketiga variabel ini ke pertidaksamaan (A.40) yaitu:

$$\begin{split} \widetilde{\mathbf{A}}_{s}(p,p) &= \begin{bmatrix} \overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} & \overline{\mathbf{D}}_{f} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} + \mathbf{I}) & -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{D}}_{f} \end{bmatrix} \\ \widetilde{\mathbf{N}}(p,p) &= \begin{bmatrix} \overline{\mathbf{B}}(p) & 0 \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{B}}(p) & \mathbf{I} \end{bmatrix}, \\ \overline{\mathbf{P}} &= \begin{bmatrix} \mathbf{P}_{1} & 0 \\ 0 & \mathbf{I} \end{bmatrix}, \end{split}$$

Dan variabel *premultipling*, untuk menghindari dua variabel dalam satu pertidaksamaan yaitu:

 $\overline{\mathbf{H}}(p) = \mathbf{P}_{1}\overline{\mathbf{L}}(p),$ $\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c} = \overline{\mathbf{D}}_{f}\mathbf{P}_{1}$

Didapatkan pertidaksamaan dalam bentuk LMI (A.41) berikut:

 \mathbf{C}_{p1}^{T} 0 0 W11 W13 w_{12} \mathbf{C}_{p2}^{T} 0 I W23 W22 0 - 7I 0 0 <0 0 0 * - 1 -11 0 * * - **A** $\mathbf{w}_{11} = \mathbf{P}_1 \overline{\mathbf{A}}(p) + (\mathbf{P}_1 \overline{\mathbf{A}}(p))^T - \overline{\mathbf{H}}(p) \overline{\mathbf{C}}_c - (\overline{\mathbf{H}}(p) \mathbf{C}_c)^T,$ $w_{12} = -(\overline{\mathbf{A}}^T(p)\mathbf{P}_1\mathbf{D}_f - \overline{\mathbf{C}}_c^T\overline{\mathbf{H}}^T(p)\overline{\mathbf{D}}_f),$ $w_{13} = \mathbf{P}_1 \overline{\mathbf{B}}(p),$ $w_{22} = -2\overline{\mathbf{D}}_f^T \mathbf{P}_1 \overline{\mathbf{D}}_f$ $w_{23} = -2\overline{\mathbf{D}}_f^T \mathbf{P}_1 \overline{\mathbf{B}}(p),$

(A.41)

Matriks kestabilan Lyapunov pada LMI (A.41) menjamin kestabilan observer untuk kesalahan sensor dan memenuhi performa H_{∞} pada pertidaksamaan (A.33). Untuk mencari *observer gain* sensor dapat dilanjutkan dengan pertidaksamaan (3.64) pada Subbab 4.5.



LAMPIRAN B

```
B.1 Program untuk Penurunan Gain State Feeback
%% PLANT INVERTED PENDULUM LINIERISASI
%Model plant : x3dot = (a*(u(5)-
\min(u(4)^2) + \sin(u(2)) + 1 + \cos(u(2)) + (\min(g) + \sin(u(2)))
fp*u(4)))/(J+miu*l*((sin(u(2)))^2))
[A1,B1,Cp1,D1]=linmod('pend',[0 0 0 0]',0) %Linearisasi pada
titik 0 rad
[A2,B2,Cp2,D2]=linmod('pend',[0 0.25 0 0]',0) %Linearisasi
pada titik 0.2612 rad
[A3,B3,Cp3,D3]=linmod('pend',[0 0.5232 0 0]',0) %Linearisasi
pada titik 0.5232 rad
C1 = [1 0 0 0;0 1 0 0;0 0 1 0]; %Keluaran untuk Observer
C2 = C1;
C3 = C1;
C_1 = [1 0 0 0]; %Keluaran untuk Kontroler
C_2 = C_1;
C_{3} = C_{2};
D1 = 0; %Disturbance
D2 = D1;
D3 = D1;
SPENCARIAN GAIN KONTROLER NOMINAL
%% Kasus Tracking
%Matriks Kompensator
Ac = [0 \ 1; \ -0.3948 \ 0];
BC = [0; 0.0628];
%Mencari matriks augmentasi tracking
Albar = [A1 \operatorname{zeros}(4,2); -Bc*C 1 Ac];
A2bar = [A2 \ zeros(4,2); -Bc*C \ 2 \ Ac];
A3bar = [A3 zeros(4,2); -Bc*C_3 Ac];
%Mencari matriks B augmentasi tracking
B1bar = [B1;0;0];
B2bar = [B2;0;0];
B3bar = [B3;0;0];
%%PENCARIAN PDC POLE PLACEMENT DENGAN BATASAN INPUT-OUTPUT
sint = sin((1/12)*3.14); %Theta
cost = cos((1/12)*3.14); %Theta
Cz1 = [1 0 0 0 0 0]; %Batasan Output
bet = 5; %Beta
```

```
umax = 17; %u maksimal
zmax = 0.4i
%Pole Placement LMI1
setlmis([])
Q1bar = lmivar(1,[6 1]);
Y1bar = 1mivar(2,[1 6]);
gamli = 2;
lmiterm([1 1 1 Qlbar],Albar,1,'s');
lmiterm([1 1 1 Ylbar],Blbar,1,'s');
lmiterm([1 1 1 0lbar], 2*gamli, 1);
lmiterm([2 1 1 Qlbar],sint*Albar,1,'s');
lmiterm([2 1 1 Ylbar],sint*Blbar,1,'s');
lmiterm([2 1 2 Qlbar],cost*Albar,1);
lmiterm([2 1 2 Qlbar],-cost,Albar');
lmiterm([2 1 2 Y1bar], cost*B1bar, 1);
lmiterm([2 1 2 -Y1bar],-cost,B1bar');
lmiterm([2 2 2 Qlbar],sint*Albar,1,'s');
lmiterm([2 2 2 Ylbar],sint*Blbar,1,'s');
lmiterm([-3 1 1 Qlbar],1,1);
lmiterm([-4 1 1 Qlbar],1,1);
lmiterm([-4 2 1 Ylbar],1,1);
lmiterm([-4 2 2 0],(umax<sup>2</sup>)/bet);
lmiterm([-5 1 1 Qlbar],1,1);
lmiterm([-5 1 2 Qlbar],1,Cz1');
lmiterm([-5 2 2 0],(zmax<sup>2</sup>)/bet);
lmisa=getlmis;
[tmin,xfeas]=feasp(lmisa);
Q_1 = dec2mat(lmisa,xfeas,Qlbar);
Y_1 = dec2mat(lmisa,xfeas,Ylbar);
Klbar = Y_1 * inv(Q_1)
%Pole Placement LMI2
setlmis([])
Q2bar = lmivar(1,[6 1]);
Y2bar = lmivar(2, [1 6]);
gam2i = 2
lmiterm([1 1 1 Q2bar],A2bar,1,'s');
lmiterm([1 1 1 Y2bar],B2bar,1,'s');
lmiterm([1 1 1 Q2bar], 2*gam2i, 1);
lmiterm([2 1 1 Q2bar],sint*A2bar,1,'s');
lmiterm([2 1 1 Y2bar],sint*B2bar,1,'s');
```

```
lmiterm([2 1 2 Q2bar],cost*A2bar,1);
lmiterm([2 1 2 Q2bar],-cost,A2bar');
lmiterm([2 1 2 Y2bar],cost*B2bar,1);
lmiterm([2 1 2 -Y2bar],-cost,B2bar');
lmiterm([2 2 2 Q2bar],sint*A2bar,1,'s');
lmiterm([2 2 2 Y2bar],sint*B2bar,1,'s');
```

```
lmiterm([-3 1 1 Q2bar],1,1);
```

```
lmiterm([-4 1 1 Q2bar],1,1);
lmiterm([-4,2,1 Y2bar],1,1);
lmiterm([-4,2,2 0],(umax<sup>2</sup>)/bet);
```

```
lmiterm([-5 1 1 Q2bar],1,1);
lmiterm([-5 1 2 Q2bar],1,Czl');
lmiterm([-5 2 2 0],(zmax<sup>2</sup>)/bet);
```

```
lmisb=getlmis;
```

```
[tmin,xfeas]=feasp(lmisb);
Q_2 = dec2mat(lmisb,xfeas,Q2bar);
Y_2 = dec2mat(lmisb,xfeas,Y2bar);
K2bar = Y_2*inv(Q_2)
```

```
%Pole Placement LMI3
```

```
setlmis([])
Q3bar = lmivar(1,[6 1]);
Y3bar = lmivar(2,[1 6]);
gam3i = 2
```

```
lmiterm([1 1 1 Q3bar],A3bar,1, s');
lmiterm([1 1 1 Y3bar],B3bar,1, 's');
lmiterm([1 1 1 Q3bar],2*gam3i,1);
```

```
Imiterm([2 1 1 Q3bar],sint*A3bar,1,'s');
Imiterm([2 1 1 Y3bar],sint*B3bar,1,'s');
Imiterm([2 1 2 Q3bar],cost*A3bar,1);
Imiterm([2 1 2 Q3bar],-cost,A3bar');
Imiterm([2 1 2 Y3bar],cost*B3bar,1);
Imiterm([2 1 2 -Y3bar],-cost,B3bar');
Imiterm([2 2 Q3bar],sint*A3bar,1,'s');
Imiterm([2 2 Y3bar],sint*B3bar,1,'s');
```

lmiterm([-3 1 1 Q3bar],1,1);

```
lmiterm([-4,1 1 Q3bar],1,1);
lmiterm([-4,2 1 Y3bar],1,1);
lmiterm([-4,2 2 0],(umax<sup>2</sup>)/bet);
```

```
lmiterm([-5 1 1 Q3bar],1,1);
lmiterm([-5 1 2 Q3bar],1,Cz1');
```

lmiterm([-5 2 2 0],(zmax²)/bet);

```
lmisc=getlmis;
[tmin,xfeas]=feasp(lmisc);
Q_3 = dec2mat(lmisc,xfeas,Q3bar);
Y_3 = dec2mat(lmisc,xfeas,Y3bar);
K3bar = Y_3*inv(Q_3)
```

B.2 Program untuk Penurunan Observer gain

```
%PENCARIAN OBSERVER GAIN ESTIMASI SENSOR
%%Matriks pendulum saat teta atau x1 =0, x2
= 0, dan x4 = 0
Df1 = [1;1;1]; %Matriks kesalahan sensor
```

0 derajat, x3

%Matrik Augmentasi

```
nl = size(A1,1); %Jumlah state %x
ml = size(B1,2); %Jumlah input %u
l1 = size(C1,1); %Jumlah output %state As
h_1 = 1; %Aktuator fault
g1 = 1; %Sensor Fault
```

%Filter LPF

```
Asl = eye(11);
rho = 0.1;
thes = sin((1/4)*3.14);
thec = cos((1/4)*3.14);
```

%Matriks Augmentasi

```
Abar1 = [A1 zeros(n1,l1); As1*C1 -As1];
Bbar1 = [B1; zeros(l1,m1)];
Dfbar1 = [zeros(n1,size(Df1,2));As1*Df1];
Cbar1 = [zeros(l1,n1) eye(l1)];
```

%LMI Fuzzy_1

```
nbar1=size(Abar1,1);
Cpl_1=0.1*eye(nbar1);
Cp2_1=0.1*eye(g1);
CP = 0.1*eye(nbar1+g1);
```

```
setlmis([])
gam1 = lmivar(1,[1 1]);
miu1 = lmivar(1,[1 1]);
P1 = lmivar(1,[nbar1 1]);
F1 = lmivar(2,[g1 l1]);
H1 = lmivar(2,[n1+11 11]);
```

```
lmiterm([1 1 1 P1],1,Abar1,'s'); %P1*Abar
lmiterm([1 1 1 H1],-1,Cbar1,'s'); %-H*Cbar1
lmiterm([1 1 2 P1],-Abar1',Dfbar1); %-Abar1*P1*Dfbar1
```

```
lmiterm([1 1 2 -H1],Cbar1',Dfbar1); %Cbar1'*H*Dfbar1
lmiterm([1 1 3 P1],1,Bbar1); %P1*Bbar1
lmiterm([1 1 5 0],Cp1_1'); %Cp1'
lmiterm([1 2 2 P1],-2*Dfbar1',Dfbar1); %-2*Dfbar1*P1*Dfbar1
lmiterm([1 2 3 P1],-Dfbar1',Bbar1); %-Dfbar1*P1*Bbar1
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cp2_1') %Cp2'
lmiterm([1 3 3 gam1],-1,1) %-gam*I
lmiterm([1 4 4 gam1],-1,1) %-gam*I
lmiterm([1 5 5 gam1],-1,1) %-gam*I
```

```
lmiterm([-2 1 1 miul],1,1) %miu*I
lmiterm([-2 1 2 P1],Dfbar1',1) %Dfbar1'*P1
lmiterm([-2 1 2 F1],-1,Cbar1) %F*Cbar1
lmiterm([-2 2 2 miul],1,1) %miu*I
```

```
lmiterm([-3 1 1 P1],1,1); %P1 > 0
lmiterm([-3 2 2 0],1);
```

```
lmiterm([4 1 1 P1],1,Abar1,'s');
lmiterm([4 1 1 H1],-1,Cbar1,'s');
lmiterm([4 1 1 P1],2*rho,1);
lmiterm([4 1 2 P1],-Abar1',Dfbar1);
lmiterm([4 1 2 -H1],Cbar1',Dfbar1);
lmiterm([4 2 2 P1],-Dfbar1',Dfbar1,'s');
lmiterm([4 2 2 0],2*rho);
```

```
lmiterm([5 1 1 P1],thes,Abar1,'s');
lmiterm([5 1 1 H1], -thes, Cbar1, 's');
lmiterm([5 1 2 P1], -thes*Abar1', Dfbar1);
lmiterm([5 1 2 -H1], thes*Cbar1', Dfbar1);
lmiterm([5 2 2 P1],-thes*Dfbar1',Dfbar1,'s');
lmiterm([5 1 3 P1],thec,Abar1,'s');
lmiterm([5 1 3 H1], -thec, Cbar1, 's');
lmiterm([5 1 4 P1],-thec*Abar1',Dfbar1);
lmiterm([5 1 4 -H1],thec*Cbarl',Dfbarl);
lmiterm([5 2 3 -P1],thec*Dfbar1',Abar1);
lmiterm([5 2 3 H1],thec*Dfbar1',Cbar1);
lmiterm([5 2 4 P1],-thec*Dfbar1',Dfbar1,'s');
lmiterm([5 3 3 P1], thes, Abar1, 's');
lmiterm([5 3 3 H1], -thes, Cbar1, 's');
lmiterm([5 3 4 P1], -thes*Abar1', Dfbar1);
lmiterm([5 3 4 -H1],thes*Cbar1',Dfbar1);
lmiterm([5 4 4 P1], -thes*Dfbar1', Dfbar1, 's');
```

%Solusi LMI Optimal_1
lmis1=getlmis;
[tmin,xfeas]=feasp(lmis1);

```
%Fungsi minimize (theta+gama)
```

q_l=decnbr(lmisl); c_l=zeros(q_l,l); for j=1:q_l Gamlj=defcx(lmisl,j,gaml); Miulj=defcx(lmisl,j,miul); c_l(j)=trace(Gamlj)+trace(Miulj); end

```
%Observer gain
```

[opt,xopt]=mincx(lmis1,c_1,[100 0 0 0 0]); GamlO = dec2mat(lmis1,xopt,gaml) %Gama Optimal 1 MiulO=dec2mat(lmis1,xopt,miul); plO = dec2mat(lmis1,xopt,Pl); hlO = dec2mat(lmis1,xopt,Hl); LbarlO = inv(plO)*hlO %Lbar Optimal flO = dec2mat(lmis1,xopt,Fl); %F Optimal

%% matriks pendulum saat teta atau x1 =0, x2 = 15 derajat, x3 = 0, dan x4 = 0 Df2 = Df1;

```
%Matrik Augmentasi
n2 = size(A2,1); %Jumlah state %x
m2 = size(B2,2); %Jumlah input %u
l2 = size(C2,1); %Jumlah output %state As
h_2 = 1; %Aktuator fault
g2 = 1; %Sensor Fault
```

%Filter LPF
As2 = eye(12);

```
%Matriks Augmentasi
Abar2 = [A2 zeros(n2,12); As2*C2 -As2];
Bbar2 = [B2; zeros(12,m2)];
Dfbar2 = [zeros(n2,size(Df2,2));As2*Df2];
Cbar2 = [zeros(12,n2) eye(12)];
```

```
%LMI Fuzzy_2
```

nbar2=size(Abar2,1); Cp1_2=0.1*eye(nbar2); Cp2_2=0.1*eye(g2);

```
setlmis([])
gam2 = lmivar(1,[1 1]);
miu2 = lmivar(1,[1 1]);
P2 = lmivar(1,[nbar2 1]);
F2 = lmivar(2,[g2 12]);
H2 = lmivar(2,[n2+12 12]);
```

```
lmiterm([1 1 1 P2],1,Abar2,'s'); %P2*Abar
lmiterm([1 1 1 H2],-1,Cbar2,'s'); %-H*Cbar2
lmiterm([1 1 2 P2],-Abar2',Dfbar2); %-Abar2*P2*Dfbar2
lmiterm([1 2 1 H2],Dfbar2',Cbar2); %Cbar2'*H*Dfbar2
lmiterm([1 1 3 P2],1,Bbar2); %P2*Bbar2
lmiterm([1 1 5 0],Cp1_2'); %Cp2'
lmiterm([1 2 2 P2],-2*Dfbar2',Dfbar2); %-2*Dfbar2*P2*Dfbar2
lmiterm([1 2 3 P2],-Dfbar2',Bbar2); %-Dfbar2*P2*Bbar2
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cp2 2') %Cp2'
lmiterm([1 3 3 gam2],-1,1) %-gam*I
lmiterm([1 4 4 gam2], -1,1) %-gam*I
lmiterm([1 5 5 gam2],-1,1) %-gam*I
lmiterm([1 6 6 gam2],-1,1) %-gam*I
lmiterm([-2 1 1 miu2],1,1) %miu*I
lmiterm([-2 1 2 P2],Dfbar2',1) %Dfbar2'*P2
lmiterm([-2 1 2 F2], -1, Cbar2) %F*Cbar2
lmiterm([-2 2 2 miu2],1,1) %miu*I
lmiterm([-3 1 1 P2],1,1);
lmiterm([-3 2 2 0],1);
lmiterm([4 1 1 P2],1,Abar2,'s');
lmiterm([4 1 1 H2],-1,Cbar2,'s');
lmiterm([4 1 1 P2],2*rho,1);
lmiterm([4 1 2 P2], -Abar2', Dfbar2);
lmiterm([4 1 2 -H2], Cbar2', Dfbar2);
lmiterm([4 2 2 P2], -Dfbar2', Dfbar2, 's');
lmiterm([4 2 2 0],2*rho);
lmiterm([5 1 1 P2],thes,Abar2,'s');
lmiterm([5 1 1 H2],-thes,Cbar2,'s');
lmiterm([5 1 2 P2], -thes*Abar2', Dfbar2);
lmiterm([5 1 2 -H2],thes*Cbar2',Dfbar2);
lmiterm([5 2 2 P2],-thes*Dfbar2',Dfbar2,'s');
lmiterm([5 1 3 P2],thec,Abar2,'s');
lmiterm([5 1 3 H2], -thec, Cbar2, 's');
lmiterm([5 1 4 P2],-thec*Abar2',Dfbar2);
lmiterm([5 1 4 -H2],thec*Cbar2',Dfbar2);
lmiterm([5 2 3 -P2],thec*Dfbar2',Abar2);
lmiterm([5 2 3 H2],thec*Dfbar2',Cbar2);
lmiterm([5 2 4 P2],-thec*Dfbar2',Dfbar2,'s');
lmiterm([5 3 3 P2],thes,Abar2,'s');
lmiterm([5 3 3 H2],-thes,Cbar2,'s');
lmiterm([5 3 4 P2],-thes*Abar2',Dfbar2);
lmiterm([5 3 4 -H2],thes*Cbar2',Dfbar2);
```

```
lmiterm([5 4 4 P2],-thes*Dfbar2',Dfbar2,'s');
```

```
%Solusi LMI Optimal_2
```

```
lmis2=getlmis;
[tmin,xfeas]=feasp(lmis2);
```

```
%Fungsi minimize (theta + gama)
q_2=decnbr(lmis2);
c_2=zeros(q_2,1);
for j=1:q_2
Gam2j=defcx(lmis2,j,gam2);
Miu2j=defcx(lmis2,j,miu2);
c_2(j)=trace(Gam2j)+trace(Miu2j);
end
```

%Observer gain

[opt,xopt]=mincx(lmis2,c_2,[100 0 0 0 0]); Gam2O = dec2mat(lmis2,xopt,gam2) %Gama Optimal 1 Miu2O=dec2mat(lmis2,xopt,miu2); p2O = dec2mat(lmis2,xopt,P2); h2O = dec2mat(lmis2,xopt,H2); Lbar2O=inv(p2O)*h2O %Lbar Optimal f2O=dec2mat(lmis2,xopt,F2) %F Optimal

%% matriks pendulum saat teta atau x1 =0, x2 = 30 derajat,x3 = 0, dan x4 = 0 Df3 = Df2;

%Matrik Augmentasi

n3 = size(A3,1); %Jumlah state %x
m3 = size(B3,2); %Jumlah input %u
l3 = size(C3,1); %Jumlah output %state As
h_3 = 1; %Aktuator fault
g3 = 1; %Sensor Fault

%Filter

As3 = eye(13);

```
%Matriks Augmentasi
Abar3 = [A3 zeros(n3,l3); As3*C3 -As3];
Bbar3 = [B3; zeros(l3,m3)];
Dfbar3 = [zeros(n3,size(Df3,2));As3*Df3];
```

```
Cbar3 = [zeros(13,n3) eye(13)];
```

```
%LMI Fuzzy_3
```

nbar3=size(Abar3,1); Cp1_3=0.1*eye(nbar3); Cp2_3=0.1*eye(g3);

```
setlmis([])
gam3 = lmivar(1,[1 1]);
miu3 = lmivar(1,[1 1]);
P3 = lmivar(1,[nbar3 1]);
```

```
F3 = lmivar(2, [q3 13]);
H3 = lmivar(2,[n3+13 13]);
lmiterm([1 1 1 P3],1,Abar3,'s'); %P3*Abar
lmiterm([1 1 1 H3],-1,Cbar3,'s'); %-H*Cbar3
lmiterm([1 1 2 P3], -Abar3', Dfbar3); %-Abar3*P3*Dfbar3
lmiterm([1 2 1 H3],Dfbar3',Cbar3); %Cbar3'*H*Dfbar3
lmiterm([1 1 3 P3],1,Bbar3); %P3*Bbar3
lmiterm([1 1 5 0],Cp1 3'); %Cp3'
lmiterm([1 2 2 P3],-2*Dfbar3',Dfbar3); %-2*Dfbar3*P3*Dfbar3
lmiterm([1 2 3 P3], -Dfbar3', Bbar3); %-Dfbar3*P3*Bbar3
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cp2_3') %Cp3'
lmiterm([1 3 3 gam3],-1,1) %-gam*I
lmiterm([1 4 4 gam3],-1,1) %-gam*I
lmiterm([1 5 5 gam3],-1,1) %-gam*I
lmiterm([1 6 6 gam3],-1,1) %-gam*I
lmiterm([-2 1 1 miu3],1,1) %miu*I
lmiterm([-2 1 2 P3],Dfbar3',1) %Dfbar3'*P3
lmiterm([-2 1 2 F3],-1,Cbar3) %F*Cbar3
lmiterm([-2 2 2 miu3],1,1) %miu*I
lmiterm([-3 1 1 P3],1,1);
lmiterm([-3 2 2 0],1);
lmiterm([4 1 1 P3],1,Abar3,'s');
lmiterm([4 1 1 H3], -1, Cbar3, 's');
lmiterm([4 1 1 P3], 2*rho, 1);
lmiterm([4 1 2 P3],-Abar3',Dfbar3);
lmiterm([4 1 2 -H3],Cbar3',Dfbar3);
lmiterm([4 2 2 P3], -Dfbar3', Dfbar3, 's');
lmiterm([4 2 2 0],2*rho);
lmiterm([5 1 1 P3],thes,Abar3,'s');
lmiterm([5 1 1 H3], -thes, Cbar3, 's');
lmiterm([5 1 2 P3],-thes*Abar3',Dfbar3);
lmiterm([5 1 2 -H3], thes*Cbar3', Dfbar3);
lmiterm([5 2 2 P3],-thes*Dfbar3',Dfbar3,
                                            );
lmiterm([5 1 3 P3],thec,Abar3,'s');
lmiterm([5 1 3 H3],-thec,Cbar3,'s');
lmiterm([5 1 4 P3],-thec*Abar3',Dfbar3);
lmiterm([5 1 4 -H3],thec*Cbar3',Dfbar3);
lmiterm([5 2 3 -P3],thec*Dfbar3',Abar3);
lmiterm([5 2 3 H3],thec*Dfbar3',Cbar3);
lmiterm([5 2 4 P3],-thec*Dfbar3',Dfbar3,'
lmiterm([5 3 3 P3], thes, Abar3, 's');
lmiterm([5 3 3 H3],-thes,Cbar3,'s');
lmiterm([5 3 4 P3],-thes*Abar3',Dfbar3);
lmiterm([5 3 4 -H3],thes*Cbar3',Dfbar3);
```

lmiterm([5 4 4 P3],-thes*Dfbar3',Dfbar3,'s');

%Solusi LMI Optimal_3
lmis3=getlmis;
[tmin,xfeas]=feasp(lmis3);

```
%Fungsi minimize (theta+gama)
q_3=decnbr(lmis3);
c_3=zeros(q_3,1);
for j=1:q_3
Gam3j=defcx(lmis3,j,gam3);
Miu3j=defcx(lmis3,j,miu3);
c_3(j)=trace(Gam3j)+trace(Miu3j);
end
```

```
%Observer gain
```

```
[opt,xopt]=mincx(lmis3,c_3,[100 0 0 0 0]);
Gam30 = dec2mat(lmis3,xopt,gam3); %Gama Optimal
Miu30=dec2mat(lmis3,xopt,miu3);
p30 = dec2mat(lmis3,xopt,P3);
h30 = dec2mat(lmis3,xopt,H3);
Lbar30=inv(p30)*h30; %L Optimal
f30=dec2mat(lmis3,xopt,F3); %F Optimal)
```

```
%% PENCARIAN OBSERVER GAIN AKTUATOR
%LMI Aktuator 1
Cpal_1=0.5*eye(n1);
Cpal_2=0.5*eye(g1);
CPA = 0.5*eye(n1+g1);
ro = 0.5
tes = sin((1/4)*3.14);
tec = cos((1/4)*3.14);
```

```
setlmis([])
gama1 = lmivar(1,[1 1]);
miu_a1 = lmivar(1,[1 1]);
Pla = lmivar(1,[n1 1]);
Fla = lmivar(2,[h_1 11]);
Hla = lmivar(2,[n1 11]);
```

```
lmiterm([1 1 1 Pla],1,A1,'s'); %Pla*A1
lmiterm([1 1 1 Hla],-1,C1,'s'); %-H*C21
lmiterm([1 1 2 Pla],-A1',B1); %-A1'*Pla*B1
lmiterm([1 2 1 Hla],B1',C1); %C1'*H1'*B1
lmiterm([1 3 Hla],-1,Df1); %-H1*Df1
lmiterm([1 1 5 0],Cpa1_1'); %Cpa1'
lmiterm([1 2 2 Pla],-2*B1',B1); %-B1'*Pla*B1
lmiterm([1 2 3 Hla],-B1',Df1); %-B1'*H1*Df1
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cpa1_2') %Cpa2'
```

```
lmiterm([1 3 3 gama1],-1,1) %-gama*I
lmiterm([1 4 4 gama1],-1,1) %-gama*I
lmiterm([1 5 5 gama1],-1,1) %-gama*I
lmiterm([1 6 6 gama1],-1,1) %-gama*I
```

```
lmiterm([-2 1 1 miu_al],1,1) %miu*I
lmiterm([-2 1 2 Pla],B1',1) %B1'*Pla
lmiterm([-2 1 2 Fla],-1,C1) %Fla*C1
lmiterm([-2 2 2 miu_al],1,1) %miu*I
```

lmiterm([-3 1 1 P1a],1,1); %P1a>0

```
lmiterm([4 1 1 Pla],1,A1,'s');
lmiterm([4 1 1 Hla],-1,C1,'s');
lmiterm([4 1 1 Pla],2*ro,1);
lmiterm([4 1 2 Pla],-A1',B1);
lmiterm([4 1 2 -Hla],C1',B1);
lmiterm([4 2 2 Pla],-B1',B1,'s');
lmiterm([4 2 2 0],2*ro);
```

```
lmiterm([5 1 1 P1a],tes,A1,'s');
lmiterm([5 1 1 H1a], -tes, C1, 's');
lmiterm([5 1 2 P1a],-tes*A1',B1);
lmiterm([5 1 2 -H1a],tes*C1',B1);
lmiterm([5 2 2 Pla],-tes*Bl',Bl,'s');
lmiterm([5 1 3 Pla],tec,A1,'s');
lmiterm([5 1 3 H1a],-tec,C1,'s');
lmiterm([5 1 4 Pla], -tec*A1', B1);
lmiterm([5 1 4 -H1a],tec*C1',B1);
lmiterm([5 2 3 -P1a],tec*B1',A1);
lmiterm([5 2 3 H1a],tec*B1',C1);
lmiterm([5 2 4 P1a],-tec*B1',B1,'s');
lmiterm([5 3 3 P1a],tes,A1,'s');
lmiterm([5 3 3 H1a],-tes,C1,'s');
lmiterm([5 3 4 P1a], -tes*A1',B1);
lmiterm([5 3 4 -H1a],tes*C1',B1);
lmiterm([5 4 4 Pla],-tes*B1',B1,'s');
```

```
%Solusi LMI
```

lmisla=getlmis; [tmin,xfeas]=feasp(lmisla);

```
%Fungsi minimize(theta+gama)
qla=decnbr(lmisla);
cla=zeros(qla,1);
for j=1:qla
Gamlja=defcx(lmisla,j,gamal);
Miulja=defcx(lmisla,j,miu_al);
cla(j)=trace(Gamlja)+trace(Miulja);
end
```

%Observer gain

```
[opt,xopt]=mincx(lmisla,cla,[100 10 0 0 0]);
GamlOa = dec2mat(lmisla,xopt,gama1) %Gama Optimal
MiulOa=dec2mat(lmisla,xopt,miu_a1);
plOa = dec2mat(lmisla,xopt,Pla);
hlOa = dec2mat(lmisla,xopt,Hla);
LbarlOa=inv(plOa)*hlOa %Lbar Optimal
flOa=dec2mat(lmisla,xopt,Fla) %Lbar Optimal
```

%LMI_Aktuator_2 Cpal_1=0.5*eye(n2);

Cpa1_2=0.5*eye(g2);

setlmis([])
gama2 = lmivar(1,[1 1]);
miu_a2 = lmivar(1,[1 1]);
P2a = lmivar(1,[n2 1]);
F2a = lmivar(2,[h_2 12]);
H2a = lmivar(2,[n2 12]);

```
lmiterm([1 1 1 P2a],1,A2,'s'); %P2a*A2
lmiterm([1 1 1 H2a],-1,C2,'s'); %-H*C2
lmiterm([1 1 2 P2a],-A2',B2); %-A2'*P2a*B2
lmiterm([1 2 1 H2a],B2',C2); %-C2'*H2'*B2
lmiterm([1 1 3 H2a],-1,Df2); %-H2*Df2
lmiterm([1 1 5 0],Cpa1_1'); %Cpa1'
lmiterm([1 2 2 P2a],-2*B2',B2); %-B2'*P2a*B2
lmiterm([1 2 3 H2a],-B2',Df2); %-B2'*P2a*B2
lmiterm([1 2 3 H2a],-B2',Df2); %-B2'*P2a*B2
lmiterm([1 2 6 0],Cpa1_2') %Cpa2'
lmiterm([1 3 3 gama2],-1,1) %-gama*I
lmiterm([1 4 4 gama2],-1,1) %-gama*I
lmiterm([1 5 5 gama2],-1,1) %-gama*I
lmiterm([1 6 6 gama2],-1,1) %-gama*I
```

```
lmiterm([-2 1 1 miu_a2],1,1) %miu*I
lmiterm([-2 1 2 P2a],B2',1) %B2'*P2a
lmiterm([-2 1 2 F2a],-1,C2) %F2a*C2
lmiterm([-2 2 2 miu_a2],1,1) %miu*I
```

lmiterm([-3 1 1 P2a],1,1); %P2a>0

```
lmiterm([4 1 1 P2a],1,A2,'s');
lmiterm([4 1 1 H2a],-1,C2,'s');
lmiterm([4 1 1 P2a],2*ro,1);
lmiterm([4 1 2 P2a],-A2',B2);
lmiterm([4 1 2 -H2a],C2',B2);
lmiterm([4 2 2 P2a],-B2',B2,'s');
lmiterm([4 2 2 0],2*ro);
```

```
lmiterm([5 1 1 P2a],tes,A2,'s');
lmiterm([5 1 1 H2a],-tes,C2,'s');
lmiterm([5 1 2 P2a], -tes*A2', B2);
lmiterm([5 1 2 -H2a],tes*C2',B2);
lmiterm([5 2 2 P2a],-tes*B2',B2,'s');
lmiterm([5 1 3 P2a],tec,A2,'s');
lmiterm([5 1 3 H2a],-tec,C2,'s');
lmiterm([5 1 4 P2a],-tec*A2',B2);
lmiterm([5 1 4 -H2a],tec*C2',B2);
lmiterm([5 2 3 -P2a],tec*B2',A2);
lmiterm([5 2 3 H2a],tec*B2',C2);
lmiterm([5 2 4 P2a],-tec*B2',B2,'s');
lmiterm([5 3 3 P2a],tes,A2,'s');
lmiterm([5 3 3 H2a],-tes,C2,'s');
lmiterm([5 3 4 P2a], -tes*A2', B1);
lmiterm([5 3 4 -H2a],tes*C2',B1);
lmiterm([5 4 4 P2a], -tes*B2',B1,'s');
```

```
%Solusi LMI
```

lmis2a=getlmis;
[tmin,xfeas]=feasp(lmis2a);

```
%Fungsi minimize (theta+gama)
q2a=decnbr(lmis2a);
```

```
c2a=zeros(q2a,1);
for j=1:q2a
Gam2ja=defcx(lmis2a,j,gama2);
Miu2ja=defcx(lmis2a,j,miu_a2);
c2a(j)=trace(Gam2ja)+trace(Miu2ja);
end
```

```
[opt,xopt]=mincx(lmis2a,c2a,[100 10 0 0 0]);
Gam2Oa = dec2mat(lmis2a,xopt,gama2) %Gama Optimal
Miu2Oa=dec2mat(lmis2a,xopt,miu_a2);
p2Oa = dec2mat(lmis2a,xopt,P2a);
h2Oa = dec2mat(lmis2a,xopt,H2a);
Lbar2Oa=inv(p2Oa)*h2Oa %Lbar Optimal
f2Oa=dec2mat(lmis2a,xopt,F2a) %F Optimal
```

```
%LMI Aktuator 3
Cpa1_1=0.5*eye(n2);
Cpa1_2=0.5*eye(g2);
```

```
setlmis([])
gama3 = lmivar(1,[1 1]);
miu_a3 = lmivar(1,[1 1]);
P3a = lmivar(1,[n3 1]);
F3a = lmivar(2,[h_3 13]);
H3a = lmivar(2,[n3 13]);
```

```
lmiterm([1 1 1 P3a],1,A3,'s'); %P3a*A3
lmiterm([1 1 1 H3a],-1,C3,'s'); %-H*C3
lmiterm([1 1 2 P3a],-A3',B3); %-A3'*P3a*B3
lmiterm([1 2 1 H3a],B3',C3); %C3'*H3'*B3
lmiterm([1 1 3 H3a],-1,Df3); %-H3*Df3
lmiterm([1 1 5 0],Cpa1 1'); %Cpa3'
lmiterm([1 2 2 P3a],-2*B3',B3); %-B3'*P3a*B3
lmiterm([1 2 3 H3a],-B3',Df3); %-B2'*H3*Df3
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cpa1_2') %Cpa2'
lmiterm([1 3 3 gama3],-1,1) %-gama*I
lmiterm([1 4 4 gama3],-1,1) %-gama*I
lmiterm([1 5 5 gama3],-1,1) %-gama*I
lmiterm([1 6 6 gama3],-1,1) %-gama*I
lmiterm([-2 1 1 miu_a3],1,1) %miu*I
lmiterm([-2 1 2 P3a],B3',1) %B3'*P3a
lmiterm([-2 1 2 F3a],-1,C3) %F3a*C3
lmiterm([-2 2 2 miu a3],1,1) %miu*I
lmiterm([-3 1 1 P3a],1,1); %P3a>0
lmiterm([4 1 1 P3a],1,A3,'s');
lmiterm([4 1 1 H3a],-1,C3,'s');
lmiterm([4 1 1 P3a],2*ro,1);
lmiterm([4 1 2 P3a],-A3',B3);
lmiterm([4 1 2 -H3a],C3',B3);
lmiterm([4 2 2 P3a],-B3',B3,'s');
lmiterm([4 2 2 0],2*ro);
lmiterm([5 1 1 P3a],tes,A3,'s');
lmiterm([5 1 1 H3a],-tes,C3,'s');
lmiterm([5 1 2 P3a],-tes*A3',B3);
lmiterm([5 1 2 -H3a],tes*C3',B3);
lmiterm([5 2 2 P3a],-tes*B3',B3,'s');
lmiterm([5 1 3 P3a],tec,A3,'s');
lmiterm([5 1 3 H3a],-tec,C3,'s');
lmiterm([5 1 4 P3a],-tec*A3',B3);
lmiterm([5 1 4 -H3a],tec*C3',B3);
lmiterm([5 2 3 -P3a],tec*B3',A3);
lmiterm([5 2 3 H3a],tec*B3',C3);
lmiterm([5 2 4 P3a],-tec*B3',B3,'s');
lmiterm([5 3 3 P3a],tes,A3,'s');
lmiterm([5 3 3 H3a],-tes,C3,'s');
lmiterm([5 3 4 P3a],-tes*A3',B3);
lmiterm([5 3 4 -H3a],tes*C3',B3);
lmiterm([5 4 4 P3a],-tes*B3',B3,'s');
```

%Solusi LMI

```
lmis3a=getlmis;
[tmin,xfeas]=feasp(lmis3a);
```

```
%Fungsi minimize (theta+gama)
q3a=decnbr(lmis3a);
c3a=zeros(q3a,1);
for j=1:q3a
Gam3ja=defcx(lmis3a,j,gama3);
Miu3ja=defcx(lmis3a,j,miu_a3);
c3a(j)=trace(Gam3ja)+trace(Miu3ja);
end
```

[opt,xopt]=mincx(lmis3a,c3a,[100 10 0 0 0]); Gam3Oa = dec2mat(lmis3a,xopt,gama3) %Gama Optimal Miu3Oa=dec2mat(lmis3a,xopt,miu_a3); p3Oa = dec2mat(lmis3a,xopt,P3a); h3Oa = dec2mat(lmis3a,xopt,H3a); Lbar3Oa=inv(p3Oa)*h3Oa %Lbar Optimal f3Oa=dec2mat(lmis3a,xopt,F3a) %F Optimal

%%END OF THIS PROGRAM :)

B.3 Blok Diagram Simulink

Blok inverted pendulum



Blok Numerator x3d



Blok Diagram Fungsi Bobot





Blok Diagram Fuzzy T-S PPIO Sensor Observer



Blok Diagram Fuzzy T-S Actuator Observer

_____2 u_kompensasi 1 u_error 2 Gain 1 fault 3 Kbar Blok Diagram Kompensasi Kesalahan Sensor Low Pass Filter 2 y_filter 2 1 f_topi Gain Gain 4 x_esti 3 y_kom Blok Diagram LPF <u>1</u> s Integrator 1 Add1

Blok Diagram Kompensasi Kesalahan Aktuator

KONTROL TOLERANSI KESALAHAN SENSOR DAN AKTUATOR BERBASIS *PROPORTIONAL-PROPORTIONAL* INTEGRAL OBSERVER UNTUK SISTEM PENDULUM KERETA

Nama : Ardiansyah Pembimbing : 1. Prof. Dr. Ir. Mohammad Nuh, DEA 2. Dr. Trihastuti Agustinah, ST., MT.

ABSTRAK

Kontrol Toleransi Kesalahan atau Fault Tolerant Control (FTC) adalah suatu kontrol untuk meminimalisasi kesalahan yang terjadi pada sensor atau aktuator suatu sistem. Permasalahan pada FTC adalah bagaimana mendeteksi dan mengisolasi kesalahan pada sensor dan aktuator yang terjadi. Pada tugas akhir ini dirancang suatu FTC dengan merekontruksi dan mengompensasi kesalahan sensor dan aktuator yang teriadi secara simultan pada sistem nonlinear menggunakan Proportional-Proportional Integral Observer (PPIO). Dengan menggunakan Linear Matrix Inequality (LMI), gain estimator kesalahan sensor dan aktuator ditentukan. Kontrol nominal sistem menggunakan Fuzzy Takagi-Sugeno (T-S) konsep Model Referensi PDC (Parallel Distributed Compensation) dengan state feedback gain ditentukan menggunakan LMI pole placement. Implementasi diterapkan pada sistem pendulum kereta, yaitu sistem nonlinear yang biasa berfungsi untuk menguji berbagai metode kontrol. Dari hasil simulasi dan implementasi, Kontrol Toleransi Kesalahan (FTC) yang dibangun dapat mengompensasi kesalahan sensor dan aktuator yang terjadi pada plant. Selain itu, FTC memenuhi performa H_w yang diinginkan.

Kata Kunci : FTC, Fuzzy T-S, LMI, LMI Pole Placement, Proportional-Proportional Integral Observer

FAULT TOLERANT CONTROL BASED ON PROPORTIONAL-PROPORTIONAL INTEGRAL OBSERVER WITH SENSOR AND ACTUATOR FAULTS FOR PENDULUM CART SYSTEM

Name Supervisors Ardiansyah
1. Prof. Dr. Ir. Muhammad Nuh, DEA
2. Dr. Trihastuti Agustinah, ST., MT.

ABSTRACT

FTC is a method for compensating or minimalizing faults that occurred mostly in sensor and actuator. There are two types of FTC. Active Fault Tolerant Control (AFTC) and Passive Fault Tolerant Control (PFTC). This undergraduate thesis concerns the designing AFTC to compensate simultaneous actuator and sensor faults. The AFTC-is designed based on Fuzzy Takagi-Sugeno Proportional-Proportional Integral Observer (T-S PPIO). Observer gain for sensor and actuator faults are determined using Linear Matrix Inequality (LMI) with Lyapunov Stability and H_{∞} performance approach. The method is implemented on nonlinear system (pendulum cart system). The thesis uses Fuzzy Takagi-Sugeno with Model Reference Parallel Distribution Compensation (PDC) to ensure stability when no fault occurred in system (nominal control). State feedback gain of nominal control is determined using LMI Region. In addition, to satisfy implementation with real plant condition, the thesis uses Input-Output constraints. Simulation results show that designed FTC has the capability to obviate faults that occurred in sensor, actuator, or both sensor and actuator. The system also satisfies H_{∞} performance with L₂-gain less than determined attenuation. Implementation results shows FTC observer can estimate and compensate faults occurred in pendulum cart with certain condition.

Keywords : FTC, Fuzzy T-S, LMI, LMI Pole Placement, Proportional-Proportional Integral Observer

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa yang telah melimpahkan Rahmat dan Karunia-Nya sehingga dapat diselesaikan karya tulis ilmiah yang berjudul **"Kontrol Toleransi Kesalahan Sensor Dan Aktuator Berbasis** *Proportional-Proportional Integral Observer* **Untuk Sistem Pendulum Kereta**" dengan baik. Dalam kesempatan ini penulis mengucapkan terimakasih kepada :

- 1. Bapak Prof. Dr. Ir. Mohammad Nuh, DEA dan Ibu Trihastuti Agustinah, S.T, M.T, selaku dosen pembimbing utama yang memberikan saran, rekomendasi dan masukan dalam Tugas Akhir ini.
- 2. Dr. Tri Arief Sardjono, selaku ketua juruan Teknik Elektro yang memberikan rekomendasi dan motivasi guna mengembangkan Tugas Akhir ini.
- 3. Ayah dan Ibu tercinta yang memberikan doa dan motivasi dalam belajar dan menyelesaikan Tugas Akhir.
- 4. Ibu Katherin Indriawati, S.T., M.T., yang banyak memberikan saran terkait FTC guna pengembangan penelitian ini
- 5. Mujianto P. Saputro, Hendra Antomy, Aldhino Putra, Fandi Yunus Dedy Bachtiar sebagai tim utama dalam merancang dan membuat FTC dalam Tugas Akhir ini
- 6. Aisyah Mega Permata, yang menjadi motivasi hidupku, mudahmudahan ini menjadi awal untuk kita
- 7. Rekan-rekan Teknik Elektro ITS, yang memberikan dukungan dan saran guna kemajuan proses penelitian ini.

Penulis menyadari bahwa pengerjaan Tugas Akhir ini masih jauh dari sempurna. Untuk itu, penulis memohon maaf atas segala kekurangan dalam Buku Tugas Akhir ini. Semoga Tugas Akhir ini dapat memberikan manfaat dalam pengembangan ilmu pengetahuan dan teknologi.

Demikian kata-kata dari penulis, dan penulis sangat berharap adanya masukan guna penyempurnaan Tugas Akhir. Terimakasih

Surabaya, 2015

Penulis

DAFTAR ISI

PERNYATAAN KEASLIAN TUGAS AKHIR iii ABSTRAK vii ABSTRACT ix KATA PENGANTAR xi DAFTAR ISI xiii
ABSTRAK ABSTRACT KATA PENGANTAR DAFTAR ISI xiii
ABSTRACT ix KATA PENGANTAR xi DAFTAR ISI xiii
KATA PENGANTAR xiii DAFTAR ISI
DAFTAR ISI
DAFTAR NOTASI DAN SIMBOL xxv
KONVENSI PENULISAN TUGAS AKHIR
BAB I PENDAHULUAN
1.1. Latar Belakang
1.2. Perumusan Masalah
1.3. Tujuan Penelitian
1.4. Metodologi Penelitian
1.5. Sistematika Laporan
1.6. Relevansı
DAD II TEODI DENUNIANO
BAB II TEORI PENUNJANG
2.1. Tinjauan Pustaka
2.2. Sistem Pendulum Kereta
2.5. Model Fisik Sistem Pendulum Kereta
2.4. Model Matematika Sistem Pendulum Kereta
2.5. Teori Puzzy Talagi Sugara (T.S)
2.0. Model Puzzy Takagi-Sugelio (1-5)
2.7. Model Referensi Derodsis Kompensator
2.6. Feligertial dal Klasifikasi Kesalahan (Juur)
2.9. Kontrol Toleransi Kesalahan Aktif (AETC)
2.10. Konuor Toleransi Kesalahan Akun (AFTC)
2.11. Observer
2.12. 1 roportional-1 roportional integral Observer
2.15. Estimasi dan Kompensasi
2.17. Entern Maria Inequality (EMI)
2.15. Entrategion dan Pole Pracement Even Region
2.10. Datasan Input Output 22

BAB III PERANCANGAN SISTEM	.37
4.1. Linearisasi Model Matematika Sistem Pendulum Kereta	.37
4.2. Pemodelan Fuzzy Takagi-Sugeno (T-S)	.39
4.3. Perancangan Kontroler Fuzzy T-S Pole Placement	
dengan Batasan Input-Output	.42
4.3.1. Perancangan Kontroler berbasis Model Referensi	.43
4.3.2. Perancangan Fuzzy T-S Pole Placement	.44
4.3.3. Batasan Input-Output.	.46
4.4. Perhitungan Gain Kontroler	.48
4.5. Desain Observer Kesalahan Sensor dan Aktuator berbasis	SRA
Fuzzy T-S PPIO	
(Proportional Proportional Integral Observer)	.51
4.5.1. Desain Observer Kesalahan Sensor berbasis	
Fuzzy T-S PPIO	.51
4.5.2. Desain LMI (<i>Linear Matrix Equalities</i>) untuk	35.
Observer Kesalahan Sensor	.55
4.5.3. Desain <i>Observer</i> Kesalahan Aktuator Berbasis	
T-S PPIO. DATA DATA	.61
4.5.4. Desain LMI (<i>Linear Matrix Equalities</i>) untuk Observe	r
Kesalahan Aktuator	.62
4.6. Perhitungan Observer gain Kesalahan Sensor dan Aktuator	.66
4.7. Diagram Simulink Kontroler dan FTC untuk Simulasi dan	
Implementasi	.68
BAB IV HASIL SIMULASI DAN IMPLEMENTASI	.71
4.8. Hasil Simulasi	.71
4.8.1. Hasil Simulasi Kontrol Nominal dengan Berbagai	7(1)
Kondisi Awal tanpa Kesalahan (Fault-Free Case)	.71
4.8.2. Hasil Simulasi Kontrol Nominal dengan Berbagai	
Kondisi Kesalahan (Faulty Case)	.77
4.8.3. Hasil Simulasi Sistem Kontrol Nominal dengan	THE Y
Kompensasi Kesalahan (Sensor atau Aktuator)	.97
4.8.4. Hasil Simulasi Sistem Kontrol Nominal dengan	
Kompensasi Kesalahan Secara Simultan (Sensor dan	
Aktuator)	110
4.9. Hasil Implementasi	117
4.9.1. Hasil Implementasi Kontrol Nominal tanpa Kesalahar	n S
(Fault-Free Case)	119

4.9.2. 4.9.3.	Hasil Implementasi Kontrol Nominal dengan Be Kondisi Kesalahan (<i>Faulty Case</i>) Hasil Implementasi Sistem Kontrol Nominal de Kompençasi Kesalahan	erbagai 132 ngan
	(Sensor, Aktuator dan Simultan)	
BAB V PENU DAFTAR PUS	TUP STAKA	145
DAFTAR RIV	VAYAT HIDUP	

		DAFTAR TABEL
DAMA.)) I DY T	
	Tabel 2.1	Klasifikasi Kesalahan Sensor
	Fabel 2.2	Klasifikasi Kesalahan Aktuator
100	Tabel 4.1	Perbandingan Respon Simulasi dengan Berbagai
		Kondisi Awal75
17 TH	Tabel 4.2	Respon Pengaruh Pemberian Kondisi Awal pada Plant
		dengan Observer yang Terkoneksi Kontrol Nominal 76
	Fabel 4.3	Respon Kontrol Nominal Terhadap Kesalahan Sensor
10h		Berbentuk Sinus dan Random
	Tabel 4.4	Respon Kontrol Nominal Terhadap Kesalahan Sensor
DATA	DI DITT	Berbentuk Intermittent, Incipient dan Abrupt
	Fabel 4.5	Respon Kontrol Nominal Terhadap Kesalahan Aktuator
		Berbentuk Sinus dan Random
	Fabel 4.6	Respon Kontrol Nominal Terhadap Kesalahan Aktuator
		Berbentuk Intermittent, Incipient dan Abrupt
DYCH	Fabel 4.7	Kriteria Kompensasi dan Performansi dari Observer
		Kesalahan Sensor
	Fabel 4.8	Kriteria Kompensasi dan Performansi dari Observer
		Kesalahan Sensor
	Fabel 4.9	Kriteria Kompensasi dan Performansi dari Observer
NYCH'	I'M WAR	Kesalahan Aktuator
	Fabel 4.10	Kriteria Kompensasi dan Performansi dari Observer
		Kesalahan Sensor
	Fabel 4.11	Kriteria Kompensasi dan Performansi dari Observer
		Kesalahan Aktuator 110
White I	Fabel 4.12	Kriteria Kompensasi Kesalahan Sensor dan Aktuator 114
	Fabel 4.13	Kriteria Kompensasi Kesalahan Sensor dan Aktuator
CARE .		secara Simultan
]	Fabel 4.14	Kriteria Respon Kontrol Nominal Implementasi 118
	Fabel 4.15	Respon Kontrol Nominal Terhadap Kesalahan Sensor
NA	D D D T	Berbentuk Sinus, Abrupt dan Incipient Implementasi 122
	Fabel 4.16	Respon Kontrol Nominal Terhadap Kesalahan Aktuator
		Berbentuk Sinus, Abrupt dan Incipient Implementasi 126
	l'abel 4.17	Respon Kontrol Nominal Terhadap Kesalahan Simultan
		Berbentuk Sinus, Abrupt dan Incipient Implementasi 131
1 Mail	l'abel 4.18	Kriteria Respon Kontrol Nominal Dengan Kompensasi
	// – ((()))	dan Tanpa Kompensasi Kesalahan Sensor Hasil
X		Implementasi


xxiv

DAFTAR GAMBAR

Gambar 2.1	Perangkat Sistem Kontrol Sistem Pendulum Kereta	.9
Gambar 2.2	Diagram Fisik Sistem Pendulum Kereta	
Gambar 2.3	Derajat Keanggotaan Himpunan Fuzzy	
Gambar 2.4	Bentuk-bentuk Fungsi Keanggotaan	
Gambar 2.5	Model Referensi	
Gambar 2.6	Jenis Kesalahan Berdasarkan Karakteristiknya	
Gambar 2.7	Klasifikasi Kesalahan Berdasarkan Letaknya	18
Gambar 2.8	Skema dari Kontrol Toleransi Kesalahan (FTC)	
Gambar 2.9	Hardware Redundancy pada Sistem	23
Gambar 2.10	Skema Model-Based Fault Diagnosis	24
Gambar 2.11	Struktur Lengkap AFTC	25
Gambar 2.12	Blok Diagram Observer	26
Gambar 2.13	Struktur Umum Proportional-Proportional Integral	
	Observer	27
Gambar 2.14	Daerah pada LMI Region (Conic Sector+Half Plane).	30
Gambar 2.15	Diagram Blok Sistem	33
Gambar 2.16	Diagram Blok Sistem Persamaan (2.40)	34
Gambar 2.17	Penyederhanaan Sistem dalam Bentuk LFT	35
Gambar 3.1	Fungsi keanggotaan fuzzy T-S	41
Gambar 3.2	Desain Kontroler Nominal Keseluruhan	43
Gambar 3.3	Region dari pole-pole sistem lup tertutup (D)	45
Gambar 3.4	Desain Observer PPIO Aktuator dan Sensor	51
Gambar 3.5	Desain Observer Kesalahan Sensor	52
Gambar 3.6	Low Pass Filter (LPF) pada Observer	52
Gambar 3.7	Desain observer kesalahan aktuator	61
Gambar 3.8	Diagram SIMULINK Matlab Hasil Perancangan	69
Gambar 3.9	Diagram SIMULINK untuk Implementasi	70
Gambar 4.1	Respon Posisi Kontrol Nominal Hasil Simulasi	71
Gambar 4.2	Respon Sudut Pendulum dengan Berbagai	7)
Call S T.C.	Kondisi Awal Sudut Pendulum Hasil Simulasi	
Gambar 4.3	Sinyal Kontrol dengan Berbagai Kondisi	
	Awal Sudut Pendulum Hasil Simulasi	73
Gambar 4.4	Kontrol Nominal dengan Berbagai Kondisi Awal Sudut	
	Pendulum dan Kecepatan Sudut Pendulum Hasil	
	Simulasi	73
Gambar 4.5	Respon Sudut Pendulum dengan Berbagai Kondisi Aw	al
	Sudut Pendulum dan Kecepatan Sudut Hasil Simulasi	74

Gambar 4.6	Sinyal Kontrol dengan Berbagai Kondisi Awal Sudut Pendulum dan Kecenatan Sudut Hasil Simulasi 75	
Gambar 47	Respon Pengaruh Pemberian Kondisi Awal pada <i>Plant</i>	
Gambai 4.7	dongon Observer vong Terkonski Kontrol Nominel	
	Hasil Simulasi	
Cambar 48	Faulty-case Sensor Kesalahan Bias Sinus	
Gambai 4.0	Hasil Simulasi 77	
Cambar 49	Estimasi Kesalahan Sensor Berbentuk Sinus Hasil	
Gambai 4.7	Simulasi	
Cambar 4 10	Error Estimasi Kesalahan Sensor Sinus Simulasi	
Cambar 4.10	Egylty case Sensor dengan Kesalahan Bias Random	
Gallibal 4.11	Gaussian Hasil Simulasi	
Cambar 112	Ectimaci Kecalahan Sensor Random Gaussian Hasil	
	Simulasi	
Combor 113	Error Estimosi Vesalahan Sensor Pandom Gaussian	
Galilbar 4.15	Hasil Simulasi	
Combor 111	Egulty Case Sensor dengen Keselehen Bigs Abrunt Hesil	
Galiibai 4.14	Simulagi	
Combor 415	Estimaci Vacalahan Sancar Piac Abrunt Simulaci	
Gambar 4.15 Combor 4.16	Estimation Resolution School Blas Advantation Simulasi	
Gallibar 4.10	Pauly Case Sensor deligan Resaranan Dias Internitient	
Cambon 417	Fatimori Kosalahan Sangar Diag Internet Hagil	
Gambar 4.17	Simulagi	
Combon 4 19	Simulasi	
Galilbar 4.10	Incinient Hegil Simulaci	
Combor 1 10	Estimagi Vasalahan Sangar Dias Inginiant Simulagi 85	
Gambar 4.19	Equity Case Althotor dongon Koselehon Dies Sinus	
Gallibar 4.20	Hagil Simulagi	
Combor 4 21	Estimasi Kasalahan Aktuatar Bias Sinus Simulasi 87	
Gambar 4.21	Error Estimaci Kasalahan Aktuator Hasil Simulasi	
Cambar 4.22	Early Case Altustor Kesalahan Bias Pandom Gaussian	
Gambai 4.23	Hasil Simulasi	
Gambar 4.24	Estimasi Kesalahan Aktuator Bias Random Gaussian	
	Hasil Simulasi	
Gambar 4.25	Error Estimasi Kesalahan Aktuator Bias Random	
	Gaussian Hasil Simulasi	
Gambar 4.26	Faulty Case Aktuator dengan Kesalahan Bias Abrupt	
	Hasil Simulasi	

Gambar 4.27	Estimasi Kesalahan Aktuator Bias <i>Abrupt</i> Hasil Simulasi	.91
Gambar 4.28	Faulty Case Aktuator dengan Kesalahan Bias	
	Intermittent Hasil Simulasi	.91
Gambar 4.29	Estimasi Kesalahan Aktuator Bias Intermittent Hasil	
	Simulasi	.92
Gambar 4.30	Faulty Case Aktuator dengan Kesalahan Bias Incipier	nt
	Hasil Simulasi	.92
Gambar 4.31	Estimasi Kesalahan Aktuator Aditif <i>Incipient</i> Hasil	93
Gambar 4.32	Faulty Case Aktuator dengan Kesalahan Loss of	7
and and	Sensitivity Hasil Simulasi	.94
Gambar 4.33	Faulty Case Sensor dengan Kesalahan Loss of	
	Measurement Hasil Simulasi	.95
Gambar 4.34	Faulty Case Aktuator dengan Kesalahan Loss of	
	Effectiveness Hasil Simulasi	.96
Gambar 4.35	Faulty Case Aktuator dengan Kesalahan Aktuator Los	SS
(17 (17)) (17)	of Measurement Hasil Simulasi	.96
Gambar 4.36	Kompensasi Faulty Case Sensor dengan Berbagai	
	Kesalahan Hasil Simulasi	.98
Gambar 4.37	L2-gain dari Observer Kesalahan Sensor berbentuk	
ATTE TO AT	Sinus Hasil Simulasi	. 99
Gambar 4.38	Kompensasi Observer Sensor Terhadap Berbagai Jen	is
	Kesalahan Berdasarkan Karakteristiknya Simulasi	100
Gambar 4.39	L2-Gain Masing-Masing untuk Kesalahan Berbagai	
	Jenis Kesalahan Sensor Hasil Simulasi	102
Gambar 4.40	Kompensasi Faulty Case Aktuator dengan Berbagai	
	Jenis Kesalahan Hasil Simulasi	103
Gambar 4.41	L2-Observer gain Kesalahan Aktuator Kesalahan Sinu	15
	Hasil Simulasi	104
Gambar 4.42	Kompensasi Faulty Case Aktuator dengan Berbagai	\overline{n}
Jes Ja	Jenis Kesalahan Berdasarkan Karakteristiknya Hasil	
	Simulasi	105
Gambar 4.43	Kompensasi Faulty-Case Sensor Multiplikatif (Loss of	of
and and	Sensitivity 70%) Hasil Simulasi	107
Gambar 4.44	Estimasi Kesalahan Sensor Multiplikatif (Loss of	W/c
	Sensitivity) Hasil Simulasi	107
Gambar 4.45	Kompensasi Faulty Case Sensor Kesalahan Multiplik	atif
	(Loss of Measurement) Hasil Simulasi	108

Gambar 4.46	Kompensasi Faulty-Case Aktuator dengan Kesalahan	
	Multiplikatif (<i>Loss of Effectiveness</i> 70%) Simulasi 109	
Gambar 4.47	Estimasi Kesalahan Aktuator Multiplikatif (Loss of	
	Effectiveness) Hasil Simulasi	
Gambar 4.48	Kompensasi Kesalahan Aktuator Multiplikatif (Loss of	
	Measurement) Hasil Simulasi	
Gambar 4.49	Lo-Gain Masing-Masing untuk Berbagai Jenis	
Guillour 1112	Kesalahan Simultan Berdasarkan Karakteristiknya Hasil	
	Simulasi DATA DATA 1117	
Gambar 4 50	Kompensasi <i>Equity Case</i> Sensor dan Aktuator secara	
Gambai 4.50	Simultan dengan Berbagai Jenis Kesalahan Hasil	
	Simulasi	
Gambar 4 51	Estimasi Kesalahan Sensor dan Aktuator Sinus yang	
Gainbai 4.31	teriadi Secara Simultan Hasil Simulasi	
Combor 4 52	Estimagi Kesalahan Sensor dan Aktuator Pandom	
Gainbal 4.52	Caussian yang terjadi Secara Simultan Simulasi 114	
Combon 153	Vomponsosi Egulty Case Sonsor den Altuster soner	
Gallibar 4.55	Simultan dangan kasalahan nada parsamaan (4.22) Hasil	
	Simulaali dengan kesalahan pada persamaan (4.25) Hash	
Combon 454	Simulasi	
Gambar 4.54	Estimasi Kesalahan Sensor dan Aktualor persamaan	
Combo A SE	(4.25) yang terjadi Secara Simulan	
Gambar 4.55	Respon Posisi Kereta Tanpa Kesalanan Hasil	
0	Implementasi	
Gambar 4.56	Respon Sudut Pendulum Hasil Implementasi	
Gambar 4.57	Sinyal Kontrol Hasil Implementasi.	
Gambar 4.58	Faulty-case Sensor dengan Kesalahan Aditif Sinus 119	
Gambar 4.59	Sudut Pendulum saat Faulty-Case Implementasi120	
Gambar 4.60	Estimasi Kesalahan Sensor Aditif Sinus Hasil	
	Implementasi	
Gambar 4.61	Faulty-Case Sensor dengan Berbagai Jenis Kesalahan	
	Berdasarkan Karakteristiknya Hasil Implementasi 121	
Gambar 4.62	Respon Sudut Pendulum dengan Berbagai Jenis	
	Kesalahan Hasil Implementasi	
Gambar 4.63	Faulty-Case Aktuator dengan Kesalahan Aktuator	
7	Sinus Hasil Implementasi	
Gambar 4.64	Sinyal Kontrol Faulty-Case Aktuator Sinus Hasil	
	Implementasi	
Gambar 4.65	Estimasi Kesalahan Aktuator Sinus Implementasi124	

Gambar 4.66	Faulty-Case Aktuator dengan Berbagai Jenis Kesalahan	
	Hasil Implementasi 1	25
Gambar 4.67	Sinyal Kontrol Berbagai Jenis Kesalahan Hasil	
	Implementasi	.26
Gambar 4.68	Estimasi Kesalahan Aktuator Hasil Implementasi 1	27
Gambar 4.69	Faulty-Case dengan Kesalahan Simultan berbentuk	
	Sinus Hasil Implementasi 1	28
Gambar 4.70	Estimasi Kesalahan Sensor dan Aktuator Sinus Secara	1
THE NO	Simultan Hasil Implementasi	29
Gambar 4.71	Faulty Case Simultan Dengan Kesalahan Incipient	
	Implementasi	.29
Gambar 4.72	Estimasi Kesalahan Sensor dan Aktuator (Simultan)	
	Hasil Implementasi 1	.30
Gambar 4.73	<i>Faulty-Case</i> Sensor dan Aktuator (Simultan) <i>Abrupt</i>	
	Hasil Implementasi	.31
Gambar 4.74	Estimasi Kesalahan Sensor dan Aktuator (Simultan)	
	Abrupt Hasil Implementasi 1	.32
Gambar 4.75	Kompensasi Kesalahan Sensor Berbagai Jenis	
Ser 1	Kesalahan Hasil Implementasi 1	.34
Gambar 4.76	Sudut Pendulum Hasil Kompensasi Berbagai Jenis	
	Kesalahan Hasil Implementasi 1	35
Gambar 4.77	Kompensasi Kesalahan Aktuator Berbagai Jenis	
	Kesalahan Hasil Implementasi	37
Gambar 4.78	Sinyal Kontrol Kompensasi Berbagai Jenis Kesalahan	2.
An al	Sinus Hasil Implementasi 1	.38
Gambar 4.79	Kompensasi Kesalahan Sensor dan Aktuator (Simultan	n)
	Berbagai Kesalahan 1	40
Gambar 4.80	Sudut Pendulum Hasil Kompensasi Sensor dan Aktua	tor
	(Simultan) Hasil Implementasi 1	.41
Gambar 4.81	Sinyal Kontrol Kompensasi Sensor dan Aktuator	1
THE THE	(Simultan) Hasil Implementasi1	42

DAFTAR NOTASI DAN SIMBOL

	in all all all all all all
X	Vektor state x
X	Estimasi dari <i>state</i> x
x	Augmentasi dari state x dengan state lain
x_n	State ke-n
\dot{x}_n	Turunan pertama state ke-n
	Anggota himpunan
€	Bukan anggota himpunan
I	Matriks Identitas
A	Transpos matriks A
	Invers matriks A
A>0	Matriks A definit positif
A <0	Matriks A definit negatif
A	Matriks Augmentasi
*	Transpos dari matriks diagonal
$\mathbf{diag}(f_1,\ldots,f_n)$	Matriks diagonal dengan f_n merupakan elemen
	diagonal ke-n
Jan An	Bilangan imajiner
μ	Bilangan konstanta iterasi Fast Fault Estimation
γ	Tingkat pelemahan
	Sudut batas LMI Region
ρ	Jari-jari batas LMI Region
	Injinity
	Determinan matriks A
	w-norm
. ₂	L ₂ -norm
sup	Supremum
Sgn S()	Penjumlahan dari nilai deret
	Perkalian dari nilai deret
	Terkanan darrinnar deret

RIWAYAT HIDUP PENULIS



Penulis bernama lengkap Ardiansvah, lahir di Kediri pada tanggal 25 Oktober 1992. Pada tahun 2005 penulis menyelesaikan masa belajar di SD Gemaharjo IV. Penulis melanjutkan pendidikan di SMPN 2 Tegalombo dan menyelesaikan studinva pada tahun 2008. Kemudian penulis melanjutkan pendidikan di SMAN 1 Pacitan yang menyelesaikan studi pada tahun 2011. Karena ketertarikan pada bidang studi, membuat penulis melanjutkan pendidikan

tinggi di Institut Teknologi Sepuluh Nopember jurusan Teknik Elektro konsentrasi Sistem Pengaturan. Semasa kuliah, penulis aktif di kegiatan organisasi yaitu BEM ITS di Kementerian Riset dan Teknologi dan CSSC (*Control System Service Center*). Pada bulan Juni 2015 penulis mengikuti ujian Tugas Akhir untuk memenuhi persyaratan gelar Sarjana Teknik Elektro.



BAB I PENDAHULUAN

1.1. Latar Belakang

Dengan semakin majunya teknologi, sistem teknologi akan semakin kompleks, sistem kontrol akan semakin dibutuhkan. Tantangan dalam keselamatan kerja, proteksi dan kenyamanan menjadi tujuan utama diciptakannya teknologi. Semua sistem kontrol, seberapa tangguh komponen yang digunakan, seberapa akurat proses manufaktur, akan mengalami penurunan performansi yang mengakibatkan sistem bekerja tidak sesuai dengan kriteria yang diinginkan. Salah satu cara untuk memastikan keandalan sistem dan mentoleransi kesalahan yang ada pada sistem adalah dengan Kontrol Toleransi Kesalahan (*Fault Tolerant Control*) [1].

Fault Tolerant Control (FTC) adalah sistem kontrol yang secara khusus didesain untuk mendeteksi sekaligus memperbaiki sistem saat terjadi kesalahan agar tetap memiliki respon sesuai dengan yang diinginkan [1] [2]. Desain dari FTC telah dikembangkan selama bertahun-tahun untuk sistem yang aman dan reliabel. Sensor dan aktuator adalah komponen yang penting dalam sistem kontrol. Ketika terjadi kesalahan pada kedua komponen ini, perubahan signifikan dapat terjadi pada karakteristik sistem, seperti kerusakan sistem *plant* atau steady state error [1].

Sistem kontrol otomasi secara luas digunakan dalam berbagai aplikasi. Sistem kontrol lup tertutup memastikan kestabilan sistem [3], namun semakin banyak sistem lup tertutup dalam suatu proses, kemungkinan untuk terjadinya kesalahan pada komponen-komponen sistem juga menjadi semakin tinggi. Kesalahan tersebut dapat terjadi pada sensor, aktuator atau komponen lain pada sistem. Kejadian ini dapat menyebabkan sistem menjadi tidak stabil. Pada sistem lup tertutup tidak semua perbaikan dapat dilakukan dengan cepat dan sesegera mungkin, karena itulah penting untuk mendesain suatu desain kontrol yang memastikan performansi sistem berada pada keadaan tetap pada kriteria yang diinginkan maskipun terjadi kesalahan sistem [4].

Sistem pendulum terbalik adalah sistem nonlinear yang pendekatan kontrolnya sulit dilakukan menggunakan kontrol klasik/linear. Salah satu metode kontrol linear yang mudah dilakukan adalah dengan

pendekatan *Parallel Distributed Compensation* (PDC) model Takagi Sugeno. Pendekatan model Takagi-Sugeno telah banyak diaplikasikan untuk sistem nonlinear, karena memiliki berbagai macam kelebihan [**5**]. Kesalahan dalam sistem pendulum dapat berupa kesalahan sensor, dimana kualitas sensor tidak lagi memiliki akurasi yang tinggi, atau kesalahan aktuator dimana putaran motor tidak sesuai dengan yang diharapkan. Karena itulah dibutuhkan suatu kontrol untuk meminimalisasi kesalahan yang terjadi pada sensor atau aktuator.

Pada tugas akhir ini dirancang Kontrol Toleransi Kesalahan Aktif (Active Fault Tolerant Control) untuk sistem pendulum terbalik beroda dua. Kontrol Toleransi Kesalahan ini digunakan untuk meminimalisasi kesalahan yang diakibatkan oleh sensor dan aktuator pada sistem. Fault Detection and Isolation (FDI) digunakan dalam sistem ini menggunakan Proportional-Proportional Integral Observer. Desain observer menggunakan teori kestabilan Lyapunov dan LMI, dimana akan didapatkan state feedback gain. Sehingga diharapkan, setelah terjadi kesalahan pada sensor dan aktuator pada rentang yang ditoleransi, sistem tetap memenuhi kriteria yang diinginkan.

1.2. Perumusan Masalah

Pada Tugas Akhir ini yang menjadi persoalan utama adalah kesalahan pada sensor dan aktuator dapat menyebabkan sistem pendulum kereta mengalami penurunan performa tracking yang telah dirancang. Diperlukan mekanisme kompensasi sehingga sistem pendulum kereta tetap mengikuti sinyal referensi yang telah ditentukan dengan tetap mempertahankan batang pendulum dalam kondisi pada posisi terbalik. Karena pada implementasi panjang rel dan sinyal kontrol terbatas, maka perlu adanya batasan pada sinyal kontrol dan keluaran. Pada sistem nyata, kesalahan sensor dan aktuator dapat terjadi secara bersamaan (simultan) sehingga diperlukan suatu metode untuk mengatasi hal tersebut.

Secara detail, persoalan dalam desain Kontrol Toleransi Kesalahan (FTC) berbasis *Proportional-Proportional Integral Observer* untuk sistem pendulum kereta adalah sebagai berikut:

1). Penentuan model *fuzzy* T-S dari sistem yang diperoleh dari linearisasi model nonlinear *plant* pada beberapa titik kerjanya.

- 2). Desain model *fuzzy* T-S untuk *plant* Sistem Pendulum-Kereta, pemilihan variabel premis, himpunan *fuzzy*, dan banyaknya aturan yang digunakan.
- 3). Desain kontrol *tracking* nominal yang memenuhi batasan *inputoutput* melalui skema PDC.
- Desain observer untuk mengompensasi kesalahan pada sensor atau aktuator dan keduanya secara simultan menggunakan teori kestabilan Lyapunov dan *Linear Matrix Inequality* (LMI).
- 5). Uji sistem hasil desain melalui simulasi dan implementasi pada sistem pendulum kereta dan software Matlab.

1.3. Tujuan Penelitian

Tujuan dari tugas akhir ini adalah mendesain Kontrol Toleransi Kesalahan (FTC) untuk sistem nonlinear dalam hal ini pendulum kereta berbasis Proportional-Proportional Integral Observer. Kontrol nominal tracking dalam hal ini menggunakan fuzzy Takagi Sugeno dengan penentuan gain state feedback menggunakan LMI pole placement dan batasan input-output. Penentuan observer gain menggunakan pendekatan Linear Matrix Inequalities (LMI). Estimasi kesalahan dari observer digunakan untuk mengompensasi kesalahan aktual yang terjadi.

1.4. Metodologi Penelitian

Adapun metode penelitian dalam tugas akhir ini adalah :

1). Studi Literatur

Pada tahap ini dicari referensi-referensi yang relevan dengan tugas akhir dan digunakan sebagai acuan untuk mempelajari dasar dan pembahasan tugas akhir. Referensi yang berkaitan dengan Tugas Akhir ini antara lain, model fisik *plant "Digital Pendulum Mechanical Unit* 33-200", desain kontrol *pole placement, Linear Matrix Inequalities* (LMI), *observer* dan performansi H_{∞}

2). Perancangan Kontrol Nominal *Tracking*

Pada tahap ini, dirancang kontrol nominal yang membuat sistem pendulum kereta stabil. Kontrol nominal adalah keadaan sistem tanpa kesalahan sensor dan aktuator (*fault-free case*).

3). Perancangan *Observer* Kesalahan Sensor dan Aktuator (C) Setelah kontrol nominal mampu mengikuti trayektori yang diinginkan (*tracking*), dirancang *observer* yang mampu mengatasi kesalahan sensor dan aktuator (*faulty case*) 4). Simulasi

Simulasi menggunakan program Matlab 9.5.1. Simulasi sistem dalam keadaan *(faulty-case)* yaitu kesalahan sensor dan aktuator. Pada tahap ini dilihat seberapa besar kompensasi yang dapat ditoleransi oleh *observer* yang telah dirancang.

5). Implementasi

Implementasi dilakukan menggunakan *plant* sistem pendulum kereta (yang tersedia *Digital Pendulum Mechanical Unit* 33-200) di laboratorium Sistem Pengaturan b.105).

6). Penyusunan buku tugas akhir

Penyusunan buku terdiri atas pendahuluan, tinjauan pustaka, perancangan, simulasi dan implementasi, dan penutup. Penulisan buku memperhatikan kaidah EYD dan format tugas akhir yang telah disediakan.

1.5. Sistematika Laporan

Tugas Akhir ini dibagi dalam lima Bab dengan sistematika sebagai berikut:

Bab I : Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, metodologi penelitian, sistematika laporan dan relevansi.

Bab II

Teori Penunjang

Bab ini meliputi tinjauan pustaka, sistem pendulum kereta, model fisik pendulum kereta, model matematika sistem pendulum kereta, teori *fuzzy*, model *fuzzy* Takagi-Sugeno (T-S), model referensi berbasis kompensator, pengertian dan klasifikasi kesalahan, kontrol toleransi kesalahan, kontrol toleransi kesalahan aktif, *observer*, *proportional integral observer*, estimasi dan kompensasi, LMI, LMI region dan *pole placement* LMI region. Perancangan Sistem

Bab III 💦 : Pe

Bab ini meliputi linearisasi model matematika sistem pendulum kereta, pemodelan *fuzzy* T-S, perancangan kontroler *fuzzy* T-S *pole placement* dengan batasan *input-output*, perancangan kontroler berbasis model referensi, desain *observer* kesalahan sensor dan aktuator berbasis T-S PPIO (*Proportional* Proportional Integral Observer), desain observer kesalahan sensor berbasis PPIO dan desain observer kesalahan aktuator berbasis PPIO.

Bab IV: Hasil Pengujian Simulasi dan Implementasi Bab ini meliputi hasil pengujian simulasi dan hasil implementasi pada sistem pendulum kereta beserta analisisnya.

Bab V : Penutup

Bab ini berisi tentang kesimpulan dan saran dari hasil pengujian yang telah diperoleh.

1.6. Relevansi

Hasil dari Tugas Akhir ini diharapkan menjadi acuan untuk penelitian atau Tugas Akhir tentang FTC selanjutnya, pengembangan dan perbandingan sehingga menghasilkan metode FTC yang lebih sederhana namun memiliki performa yang sama.

BAB II TEORI PENUNJANG

2.1. Tinjauan Pustaka

Kontrol Toleransi Kesalahan (FTC) adalah sistem kontrol yang mampu mendeteksi sekaligus memperbaiki sistem saat terjadi kesalahan agar tetap memiliki respon sesuai dengan yang diinginkan. Desain dari FTC bermula dari perancangan FDD (*Fault Detection and Diagnosis*). FDD adalah metode untuk mendeteksi adanya kesalahan pada sistem terutama sensor dan aktuator. FDD dibuat dengan adanya kebutuhan untuk mengatasi sistem yang mengalami kesalahan (*fault*), seperti pada kejadian kecelakaan pada sistem penerbangan tahun 1970, menginisiasi riset dan pengembangan FDD [**4**].

Disisi lain industri yang berkembang pesat dan kebutuhan manusia akan industri semakin bertambang, keandalan akan komponenindustri semakin dibutuhkan. Pengembangan komponen metode dikembangkan berdasarkan kebutuhan Metode kompensasi ini. kompensasi kesalahan dan FDD inilah dinamakan dengan FTC. Metode FDD dan kompensasi dalam FTC memiliki berbagai macam metode. Salah satu metode yang cukup mudah diimplementasikan adalah redundancy dan observer. Metode redundancy memerlukan komponen tambahan jika terjadi kerusakan, akan digantikan oleh komponen lain tersebut. Sistem ini cukup mahal untuk diimplementasikan.

Metode selanjutnya adalah *observer*. *Observer* berperan untuk mendeteksi kesalahan pada sistem terutama sensor dan aktuator, setelah mendeteksi adanya kesalahan, kemudian dikompensasi agar dapat kembali ke keadaan nominalnya (keadaan tanpa kesalahan). Dalam implementasinya metode ini banyak dikembangkan algoritmanya karena investasi yang dilakukan lebih sedikit. Dalam [4], metode FTC diterapkan untuk *plant* turbin angin. Penerapan FTC pada turbin angin dapat mengurangi biaya *maintenance* jika terjadi kesalahan yang menyebabkan daya yang dihasilkan tidak maksimal.

2.2. Sistem Pendulum Kereta

Sistem pendulum kereta terdiri dari sepasang pendulum yang terpasang pada sebuah kereta sehingga pendulum tersebut dapat berayun bebas pada bidang vertikal. Kereta digerakkan oleh motor DC yang dihubungkan dengan *belt*. Untuk mengayunkan dan menyeimbangkan

pendulum, kereta digerakkan ke kiri atau ke kanan pada rel yang panjangnya terbatas.

Posisi kereta pada lintasan dapat dipantau melalui sensor posisi yaitu *position encoder*. Sebagai pengaman, digunakan *limit switch* pada masing-masing ujung rel. ketika kereta berada di ujung lintasan, maka limit switch tertekan oleh kereta dan motor DC akan mati, sehingga kereta akan berhenti. Sedangkan posisi sudut pendulum terhadap sumbu vertikal dipantau oleh angle *encoder*.

Dalam implementasi skema kontrol yang dirancang, digunakan sistem pendulum kereta dari *Feedback Instruments* Ltd dengan tipe "Digital Pendulum Mechanical Unit 33-200". Penerapan sistem kontrol dilakukan pada komputer dengan bantuan software Simulink/MATLAB. Komputer dan sistem pendulum kereta terhubung melalui modul "Digital Pendulum Controller 33-201" sebagai kontroler antarmuka, serta board akuisisi data (DAQ) sebagai I/O komputer. Sinyal kontrol dari komputer keluar melalui Digital to Analog Computer (DAC) yang terdapat pada DAQ.

Power amplifier yang terhubung dengan port keluaran DAQ akan menerima sinyal kontrol yang kemudian dikirim ke motor DC untuk menggerakkan kereta. Sinyal respons dari kereta dan pendulum terbaca oleh encoder dan dikirim ke komputer melalui *Analog to Digital Converter (ADC)* pada DAQ. Bagan sistem beserta perangkat pendukungnya secara keseluruhan dapat dilihat pada Gambar 2.1.

2.3. Model Fisik Sistem Pendulum Kereta [3] [6]

Diagram fisik sistem pendulum kereta yang digunakan serta gayagaya yang terjadi pada sistem pendulum kereta secara berturut-turut ditunjukkan pada Gambar 2.2. Secara fisik, sistem pendulum kereta terdiri dari dua bagian utama, yaitu kereta dan pendulum. Kereta hanya dapat bergerak pada rel dalam bidang horizontal dan pendulum berotasi pada bidang vertikal yang bersumbu pada sisi kereta.

Gaya kontrol u yang sejajar dengan dengan rel dikenakan pada kereta. Gaya gesek kereta terhadap rel dinyatakan dengan T_c sedangkan V adalah gaya normal yang bekerja pada sistem pendulum kereta. Massa kereta dan massa pendulum dinyatakan dengan m_c dan m_p . Jarak antara sumbu rotasi pendulum ke pusat massa sistem dinyatakan l sedangkan Jadalah momen inersia sistem terhadap pusat massa sistem.







Gambar 2.2 Diagram Fisik Sistem Pendulum Kereta [3]

2.4. Model Matematika Sistem Pendulum Kereta [7]

Sistem pendulum kereta terdiri dari empat state, yaitu x_1 , x_2 , x_3 , dan x_4 dengan:

- x_1 : Posisi kereta diukur dari titik tengah rel
- x_2 : Sudut pendulum terhadap garis vertikal, diukur berlawanan dengan arah jarum jam ($x_2 = 0$ menyatakan pendulum berada pada posisi terbalik)
- x_3 : Kecepatan kereta
- x_4 : Kecepatan sudut pendulum

Koordinat horizontal pusat massa adalah x_1 - $lsinx_2$, sedangkan koordinat vertikal pusat massa adalah $lcosx_2$. Berdasarkan hukum kedua Newton, sesuai dengan Gambar 2.2 persamaan gerak dapat ditulis,

$$u - T_{c} = (m_{c} + m_{p})(x_{1} - l\sin x_{2})^{"}$$
(2.1)

$$V - (m_{c} + m_{p})g = (m_{c} + m_{p})(l\cos x_{2})^{"}$$
(2.2)

$$(u - T_{c})l\cos x_{2} + Vl\sin x_{2} - D_{p} = Jx_{2}^{"}$$
(2.3)

Dengan D_p adalah momen gesek akibat gerak rotasi pendulum yang proporsional terhadap kecepatan sudut pendulum. D_p dapat dinyatakan dengan $D_p = f_p x_4$.

Persamaan (2,1) dan (2,2) menyatakan gerak translasi dari pusat massa sistem, sedangkan persamaan (2,3) menyatakan gerak rotasi sistem secara keseluruhan terhadap pusat massa sistem. Kecepatan kereta merupakan turunan pertama dari posisi kereta,

$$x_3 = \frac{dx_1}{dt}$$
 atau $x_1 = x_3$

sedangkan kecepatan sudut pendulum merupakan turunan pertama dari posisi sudut pendulum,

$$x_2 = \frac{dx_2}{dt}$$
 atau $\dot{x_2} = x_4$

Dengan mengeliminasi V pada persamaan (2.2) dan (2.3) serta beberapa manipulasi matematis dapat diperoleh persamaan state sistem pendulum kereta seperti pada persamaan (2.4).

$$\dot{x}_{1} = x_{3}$$

$$\dot{x}_{2} = x_{4}$$

$$\dot{x}_{3} = \frac{a(u - T_{c} - \mu x_{4}^{2} \sin x_{2}) + l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

$$\dot{x}_{4} = \frac{l \cos x_{2}(u - T_{c} - \mu x_{4}^{2} \sin x_{2}) + \mu g \sin x_{2} - f_{p} x_{4}}{J + \mu l \sin^{2} x_{2}}$$
(2.4)

dengan

$$\mu = (m_c + m_p)l, a = l^2 + \frac{J}{m_c + m_p}, \dot{x}_1 = x_1; i = 1, 2, 3, 4$$

Dari persamaan (2.4) dapat dilihat bahwa sistem pendulum kereta merupakan sistem nonlinear *Single Input Multiple Output (SIMO)*. Untuk mempermudah desain kontroler, maka dilakukan linearisasi pada persamaan (2.4) sehingga teori kontrol linear dapat digunakan.

2.5. Teori Fuzzy [8]

Sebelum munculnya teori logika *fuzzy*, dikenal sebuah logika Boolean yang memiliki nilai benar atau salah secara pasti. Sebaliknya, logika *fuzzy* merupakan sebuah logika yang memiliki nilai kekaburan *(fuzziness)* antara benar dan salah. Dalam teori logika *fuzzy*, sebuah nilai bisa bernilai benar dan salah secara bersamaan namun berapa besar kebenaran dan kesalahan suatu nilai tergantung kepada derajat keanggotaan yang dimilikinya.

Misalkan X merupakan semesta pembicaraan dan x adalah elemen X. Himpunan tegas A, $A \subseteq X$ didefinisikan sebagai kumpulan elemen atau objek $x \in X$, sehingga tiap-tiap x dikatakan sebagai anggota himpunan A atau tidak. Dalam teori himpunan *boolean*, dapat didefinisikan fungsi karakteristik tiap-tiap elemen x dalam X. Sehingga himpunan A dapat dinyatakan dengan (x, 0) atau (x, 1) yang masing-masing menyatakan x \notin A atau x \in A.

Dalam himpunan *fuzzy* (*fuzzy set*), masing-masing elemen memiliki derajat keanggotaannya masing-masing untuk menjadi anggota himpunan A. Sehingga fungsi karakteristik himpunan *fuzzy* bernilai antara 0 sampai 1, yang menyatakan derajat keanggotan masing-masing elemen dalam suatu himpunan. Secara matematis, himpunan *fuzzy* A dalam X dinyatakan:

 $\mathbf{A} = \left\{ \left(x, \mu_A(x) \right) | x \in X \right\}$

dengan $\mu_A(x)$ adalah fungsi keanggotaan *(membership function)* untuk himpunan *fuzzy* A. Fungsi keanggotaan memetakan tiap-tiap elemen x ke derajat keanggotaan antara 0 sampai 1.

Sebagai contoh, pada Gambar 2.3 kecepatan dapat dikategorikan menjadi tiga himpunan, yaitu kecepatan rendah, sedang dan tinggi.

Kecepatan yang kurang dari 25 km/jam dikatakan rendah, kecepatan di sekitar 50 km/jam dikatakan sedang, sedangkan kecepatan yang lebih dari 75 km/jam dikatakan tinggi. Dalam himpunan *fuzzy*, kecepatan 60 km/jam bisa dikatakan kecepatan sedang ataupun kecepatan tinggi karena kecepatan 60 km/jam memotong dua fungsi keanggotaan, yaitu kecepatan sedang dan kecepatan tinggi. Namun, sesuai dengan Gambar 2.3, kecepatan 60 km/jam dikatakan lebih condong pada kecepatan sedang karena lebih dekat pada kecepatan 50 km/jam dibanding kecepatan 75 km/jam. Atau dengan kata lain, derajat keanggotaannya lebih besar untuk kecepatan sedang dibanding kecepatan tinggi.



Gambar 2.3 Derajat Keanggotaan Himpunan Fuzzy

Fungsi keanggotaan dari suatu himpunan *fuzzy* dinyatakan dengan derajat keanggotaan suatu nilai terhadap nilai tegasnya yang berkisar antara 0 sampai 1. Fungsi keanggotaan memiliki berbagai bentuk, antara lain segitiga, trapezium, *Gaussian, Generalized Bell*, dan bentuk-bentuk yang lain.

Fungsi keanggotaan segitiga ditentukan oleh tiga parameter (a,b,c) dengan a < b < c sesuai persamaan (2.5). Ketiga parameter tersebut menyatakan letak koordinat x untuk ketiga sudut segitiga.

$$\mu_{segitiga}(x) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ \frac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{cases}$$

Fungsi keanggotaan trapezium ditentukan oleh empat parameter (a,b,c,d) dengan a < b < c < d sesuai dengan persamaan (2.6). Keempat

(2.5)

parameter tersebut menyatakan letak koordinat *x* untuk keempat sudut trapezium.

$$\mu_{rapesium}(x) = \begin{cases} 0, & x \le a \\ \frac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \frac{d-x}{d-c}, & c \le x \le d \\ 0, & d \le x \end{cases}$$
(2.6)

Fungsi Keanggotaan *Gaussian* ditentukan oleh dua parameter (c,σ) sesuai dengan persamaan (2.7) yang menentukan pusat dan lebar fungsi keanggotaan,

$$u_{Gaussian}(x) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma}\right)^2}$$
(2.7)

Fungsi keanggotaan *Generalized Bell* ditentukan oleh tiga parameter (a,b,c) sesuai dengan persamaan (2.8) dengan parameter b bernilai positif. Parameter c merupakan pusat yang menentukan letak x yang memiliki derajat keanggotaan bernilai satu.

$$\mu_{GeneralizedBell}(x) = \frac{1}{1 + \left|\frac{x - c}{a}\right|^{2b}}$$
(2)

8)

Bentuk-bentuk fungsi keanggotaan dapat digambarkan seperti Gambar 2.4. Pada Gambar 2.4, (a) menunjukkan fungsi keanggotaan bentuk segitiga dengan parameter (10,25,40), (b) bentuk trapezium dengan parameter (10,15,30,40), (c) bentuk *Gaussian* dengan parameter (25,8), dan (d) bentuk *Generalized Bell* dengan parameter (10,5,25).

2.6. Model Fuzzy Takagi-Sugeno (T-S) [9]

Kontrol *Fuzzy* T-S banyak diterapkan untuk sistem nonlinear karena kemudahan implementasinya. Model *plant* pada persamaan (2.4) dalam bentuk aturan *Fuzzy* T-S adalah:

Rule j: IF p is M_{j1}



Gambar 2.4 Bentuk-bentuk Fungsi Keanggotaan

dengan A_j dan B_j adalah koefisien matriks yang telah diliniearisasi di beberapa titik daerah kerjanya, *r* adalah jumlah aturan *fuzzy* dan *p* adalah premis dari *fuzzy* T-S. Model *fuzzy* T-S merepresentasikan persamaan state ini yaitu:

$$\dot{\mathbf{x}}(t) = \frac{\sum_{j=1}^{r} \left[w_j(x) \left\{ \mathbf{A}_j \mathbf{x}(t) + \mathbf{B}_j \mathbf{u}(t) \right\} \right]}{\sum_{j=1}^{r} w_j(x)} = \sum_{j=1}^{r} h_j(x) \left\{ \mathbf{A}_j \mathbf{x}(t) + \mathbf{B}_j \mathbf{u}(t) \right\}$$
(2.9)

dengan

$$w_j(x) = \prod_{k=1}^p M_{jk}(x_k)$$

$$h_j(x) = \frac{w_j(x)}{\sum_{j=1}^r w_j(x)}$$

untuk mempermudah notasi penulisan selanjutnya dalam tugas akhir ini, persamaan *fuzzy* T-S disederhanakan menjadi:

$$\begin{split} \mathbf{A}(p) &= \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{A}_{j}, \\ \mathbf{B}(p) &= \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{B}_{j}, \\ \mathbf{C}(p) &= \sum_{r} \mathbf{h}_{j} * \mathbf{C}_{j}, \end{split}$$

dengan:

$$0 \le \mathbf{h}_j(x) \le 1$$
 dan $\sum_{j=1}^r \mathbf{h}_j(x) = 1$

Persamaan (2.9) dapat ditulis kembali:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x} + \mathbf{B}(p)\mathbf{u}(t)$$

kontroler PDC yang dirancang adalah: Aturan kontroler j:

> Rule j: IF p is M_{j1} THEN $\mathbf{u}(t) = \mathbf{K}_{j} \mathbf{x}(t)$

dengan K_j adalah gain state feedback yang ditentukan melalui metode pole placement. Secara umum kontroler fuzzy T-S dinyatakan dalam:

$$\mathbf{u}(t) = \sum_{j=1}^{r} \mathbf{h}_{j}(x) \mathbf{K}_{j} \mathbf{x}(t)$$

(2.10)

2.7. Model Referensi Berbasis Kompensator [7]

Model referensi adalah model trayektori yang akan dijadikan masukan sistem lup tertutup. Metode ini adalah salah satu cara agar sistem mampu mengikuti trayektori yang diinginkan. Terdapat berbagai macam metode dalam model referensi, salah satunya adalah metode kompensator. Metode ini cukup sederhana dan mudah diimplementasikan. Kompensator dirancang sesuai sinyal referensi yang ingin didesain. Secara umum, model sinyal referensi adalah:

$$\dot{\mathbf{x}}_r(t) = \mathbf{A}_r \mathbf{x}_r(t) + \mathbf{B}_r r(t)$$
(2.11)

dengan $\mathbf{x}_{r}(t)$ adalah sinyal referensi, \mathbf{A}_{r} adalah matriks stabil asimtotik, \mathbf{B}_{r} adalah matrik masukan, dan r(t) adalah masukan referensi yang

nilainya terbatas. Diasumsikan nilai $\mathbf{x}_{r}(t)$, untuk semua t > 0, merepresentasikan trayektori yang diinginkan untuk diikuti oleh $\mathbf{x}(t)$. Bentuk umum dari model referensi adalah:



Suatu sistem dengan persamaan:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

jika dihubungkan dengan *plant* yang dikontrol dengan menganggap *tracking error* e(t) = r(t) - y(t) sebagai bentuk masukan dari model, maka akan diperoleh bentuk berikut:

$$\dot{\mathbf{x}}_{c}(t) = \mathbf{A}_{c} \mathbf{x}_{c}(t) + \mathbf{B}_{c} e(t)$$
$$\mathbf{y}_{r}(t) = \mathbf{x}_{c}(t)$$

(2.13)

(2.12)

dengan

$$\mathbf{A}_{c} = diag \left\{ \mathbf{A} \quad \mathbf{A} \quad \cdots \quad \mathbf{A} \right\}$$

$$\mathbf{B}_c = diag\{\mathbf{B} \ \mathbf{B} \ \cdots \ \mathbf{B}\}$$

A adalah matriks polinomial karakteristik dari sinyal referensi. Model referensi ini dalam bentuk *augmented system*:

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_{c}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{B}_{c}\mathbf{C} & \mathbf{A}_{c} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_{c}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{c} \end{bmatrix} \mathbf{r}(t)$$

(2.14)

dan sinyal kontrolnya

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{K} & \mathbf{K}_c \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_c(t) \end{bmatrix}$$

dalam bentuk fuzzy T-S Parallel Distribution Compensation:

$$\mathbf{u}(t) = \sum_{j=1}^{r} \mathbf{h}_{j}(x) \mathbf{K}_{j}[\mathbf{x}(t) \quad \mathbf{x}_{c}(t)]$$
$$\mathbf{h}_{j}(x) = \frac{\mathbf{w}_{j}(x)}{\sum_{i=1}^{r} \mathbf{w}_{i}(x)}$$

2.8. Pengertian dan Klasifikasi Kesalahan (fault) [2] [4]

Kesalahan (*fault*) adalah keadaan yang mengakibatkan sistem lup tertutup mengalami penurunan performa atau degradasi. Blanke [2] mendefinisikan kesalahan sebagai suatu keadaan yang mengakibatkan parameter sistem menyimpang dari keadaan nominalnya. Keadaan nominal adalah keadaan sistem berada pada performa yang diinginkan. Konsep kesalahan, sangat berbeda dengan kegagalan (failure). Kegagalan adalah keadaan lebih menurun dibandingkan kesalahan. Suatu sistem mengalami kegagalan apabila komponen dalam sistem tersebut sama sekali tidak bekerja. Perbedaan keduanya juga tampak pada kontroler yang mampu diterapkan. Ketika sistem terjadi kesalahan dapat ditoleransi oleh kontroler lain sehingga performa sistem berada pada keadaan yang diinginkan, namun ketika terjadi kegagalan, komponen yang berada pada sistem tersebut harus diganti. Konsep ini mendorong suatu pengembangan suatu kontrol vang mampu mentoleransi kesalahan pada komponen yang mengalami kesalahan pada sistem, yang dinamakan Kontrol Toleransi Kesalahan (Fault Tolerant Control).

(2.15)

Berdasarkan karakteristiknya, kesalahan dapat diklasifikasikan menjadi 3 yaitu; *abrupt, incipient, dan intermittent*. Gambaran ketiganya dapat dilihat pada Gambar 2.5



a. Abrupt Fault (Kesalahan Tiba-tiba)

b.

c.

a.

Gambar 2.5 (a) menunjukkan grafik *abrupt fault* terhadap fungsi waktu. Kesalahan ini datang secara acak terhadap fungsi waktu dan besarnya juga secara acak, sehingga menyerupai fungsi step.

Incipient Fault (Kesalahan Perlahan)

Incipent fault lebih mudah diatasi dibandingkan kesalahan yang lain. Hal ini dikarenakan sifatnya yang datang perlahan dan terus membesar menyerupai fungsi ramp. Dalam beberapa kasus sistem yang mengalami incipent fault, terdapat fungsi alarm yang memberi peringatan untuk mencegah kesalahan yang terjadi terus membesar.

Intermittent Fault (Kesalahan Sesaat)

Kesalahan ini menyerupai fungsi impuls, yang datang secara tiba-tiba namun berlangsung sangat singkat, ditunjukkan oleh Gambar 2.5 (c). Kesalahan sesaat tidak mudah diatasi karena sifatnya yang acak dan periodenya yang sangat singkat, namun dapat diantisipasi.

Berdasarkan lokasi dari kesalahan, kesalahan dapat dibagi menjadi tiga; kesalahan sensor, kesalahan aktuator, dan kesalahan proses.



Gambar 2.7 Klasifikasi Kesalahan Berdasarkan Letaknya [4]

Kesalahan Aktuator (Actuator Fault)

Kesalahan ini terletak di aktuator dan berhubungan dengan sinyal kontrol yang dihasilkan oleh aktuator. Ketika terjadi kerusakan, sinyal kontrol yang diberikan tidak lagi berada pada keadaan yang diinginkan sehingga mengakibatkan performa sistem menurun atau bahkan menyebabkan kerusakan *plant.* Kesalahan aktuator dapat disebabkan oleh kebakaran, keretakan atau kesalahan pemasangan kabel. Ketika aktuator sama sekali tidak bekerja maka disebut *complete actuator fault/actuator failure.*

Kesalahan Sensor (Sensor Fault)

b.

Kesalahan ini terletak di bagian sensor, yaitu bagian yang melakukan pengukuran untuk diumbanbalikkan ke kontroler. Kesalahan sensor berarti terdapat kesalahan pengukuran yang dilakukan oleh sensor yang mengakibatkan kesalahan pembacaan oleh kontroler, sehingga mengakibatkan *setpoint* berada pada keadaan yang tidak diinginkan. Kesalahan sensor lebih mudah diatasi dibandingkan kesalahan aktuator dalam penerapannya.

Kesalahan Proses (Process Fault)

Kesalahan proses adalah kesalahan yang paling kompleks karena terjadi perubahan parameter di dalam *plant*. Kesalahan ini dapat terjadi di salah satu komponen *plant* yang mengakibatkan *plant* tidak dapat beroperasi dengan baik. Banyaknya komponen di dalam *plant* akan mengakibatkan kesalahan ini menjadi kompleks dan sulit diatasi.

Beberapa literatur mengklasifikasikan kesalahan lebih jauh, seperti kesalahan aditif dan kesalahan multiplikatif. Kesalahan aditif adalah kesalahan yang bersifat menambah nilai sebenarnya dari parameter sistem berdasarkan model matematisnya. sedangkan kesalahan multiplikatif adalah kesalahan yang bersifat mengalikan (faktor) terhadap parameter sistem berdasarkan model matematikanya. Klasifikasi kesalahan seperti ini didasarkan model matematika pada keadaan sebenarnya. Klasifikasi kesalahan sangat penting untuk menentukan metode apa yang mampu mengatasinya.

Kesalahan yang muncul dalam sistem dapat dimodelkan secara matematis. Model matematika dari kesalahan jauh lebih sederhana dibanding kesalahan yang terjadi sesungguhnya pada *real plant*. Model matematika untuk kesalahan aktuator:

$$u^J = \varepsilon^a u + \alpha$$

(2.16)

dengan ε^a adalah matriks diagonal dengan elemen $0 \le \varepsilon_i^a \le 1$; i = 1,2,...m. Setiap elemen dari matriks ε_i menentukan intentitas dari kesalahan aktuator dengan mengindikasikan kesalahan aktuator. Simbol

 α adalah kesalahan aditif. Ketika ε_i^a menunjukkan aktuator bekerja dengan sepenuhnya, ketika $\alpha = 0$ menunjukkan sensor bekerja sepenuhnya. Anggap suatu sistem dengan persamaan:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

model sistem ketika terjadi kesalahan aktuator:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(\varepsilon^a \mathbf{u}(t) + \alpha(t))$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$$

Model matematis kesalahan ini disebut kesalahan multiplikatif dan aditif yang digunakan untuk pemodelan *Closed Loop Tranfer Function* (CLTF). Dengan metode yang sama, model kesalahan sensor:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(\varepsilon^a \mathbf{u}(t) + \alpha(t))$$

$$\mathbf{y}(t) = \mathbf{C}(\varepsilon^s \mathbf{x}(t) + \alpha_s(t)) \qquad (2.18)$$

(2.17)

dengan ε^s adalah matriks diagonal dengan elemen $0 \le \varepsilon_i^s \le 1$; i = 1,2,...m. Setiap elemen dari matriks ε^s menentukan intensitas dari kesalahan sensor dan α_s menentukan seberapa besar kesalahan sensor. Ketika kesalahan sensor hanya terletak pada satu atau tidak semua elemen sensor, persamaan (2.19) menjadi:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(\varepsilon^{a}\mathbf{u}(t) + \alpha(t))$$

$$\mathbf{y}(t) = \mathbf{C}\varepsilon^{s}\mathbf{x}(t) + \mathbf{D}_{f}\alpha_{s}(t)$$

(2.19)

dengan matriks $\mathbf{D}_{\rm f}$ adalah matriks yang menentukan dimana kesalahan sensor terjadi.

Tabel 2.1 Itlasiii	tusi itesululluli Selisoi [1]	
	$\alpha_{\rm s}(t) = 0$	$\alpha_{\rm s}(t) \neq 0$
$\varepsilon^{s} = 1$	Fault Free	Bias
$0 < \varepsilon^{s} < 1$	Loss of Sensitivity	Loss of Sensitivity
$\varepsilon^{s} = 0$	Loss of Measurement	Sensor Freezing

Tabel 2.1 Klasifikasi Kesalahan Sensor [1]

$\alpha(t) = 0$	$\alpha(t) \neq 0$
Fault Free	Bias
Loss of Effectiveness	Loss of Effectiveness
Loss of Measurement	Actuator Block
	$\alpha(t) = 0$ Fault Free Loss of Effectiveness Loss of Measurement

Tabel 2.2 Klasifikasi Kesalahan Aktuator [1]

2.9. Kontrol Toleransi Kesalahan (*Fault Tolerant Control*) [2] [4]

Secara umum sistem yang bekerja pada kondisi yang diinginkan disebut dalam keadaan kontrol nominal (*nominal control*). Kontrol nominal menjamin sistem berada pada performa yang diinginkan. Kontrol nominal dalam keadaan lup tertutup (*Closed Loop Transfer Function*) dan diasumsikan tidak terjadi kesalahan pada komponen-komponennya (kondisi ideal). Namun pada sistem nyata, dapat terjadi degradasi pada sistem, terutama pada sensor dan aktuatornya. Dalam kontrol nominal, sensor dan aktuator diasumsikan dapat bekerja sepenuhnya. Ketika sensor atau aktuator mengalami penurunan performa, perlu adanya kontrol yang mampu menanggulangi keadaan tersebut. Kontrol ini dinamakan Kontrol Toleransi Kesalahan (FTC). FTC dibagi menjadi dua yaitu; Kontrol Toleransi Kesalahan Aktif (AFTC) dan Kontrol Toleransi Kesalahan Pasif (PFTC).

Pada PFTC, kontrol nominal dibuat *robust* terhadap kesalahan sensor atau aktuator, sehingga tidak membutuhkan sistem rekonfigurasi. Kesalahan sensor atau aktuator dianggap mirip *disturbance* sehingga kontroller mampu mengatasinya. Berbeda dengan PFTC, struktur AFTC lebih rumit. AFTC memiliki sistem rekonfigurasi dan estimasi kesalahan untuk melakukan perhitungan, karena itu disebut "aktif". AFTC mempunyai nama lain "*self-repairing*", "*reconfigurable*", "*restructurable*", atau "*self-designing*". Untuk membedakan kedua metode Kontrol Toleransi Kesalahan ini dengan mudah, dapat dilihat pada Gambar 2.8.



Gambar 2.8 Skema dari Kontrol Toleransi Kesalahan (FTC)

Langkah-langkah yang diperlukan untuk mentoleransi kesalahan secara umum adalah:

- 1). Identifikasi dan deteksi kesalahan yang ada pada sistem. Mekanisme ini dinamakan diagnosis, untuk menentukan aksi selanjutnya dari FTC.
- 2). Toleransi kesalahan menggunakan informasi yang telah didapatkan dari diagnosis sistem. Toleransi ini dinamakan rekonfigurasi. Rekonfigurasi bersifat *adaption*, yaitu menyesuaikan dengan keadaan dalam sistem. Rekonfigurasi nonaktif ketika tidak ada kesalahan dalam sistem dan memunculkan aksi ketika terjadi kesalahan.

2.10. Kontrol Toleransi Kesalahan Aktif (AFTC) [1] [2] [4] [10]

Dewasa ini, AFTC lebih banyak dikembangkan daripada PFTC, hal ini dikarenakan metode yang digunakan pada AFTC lebih luas dan lebih fleksibel. AFTC memastikan sistem beradaptasi dengan cepat terhadap kesalahan yang terjadi. Sistem AFTC lebih luas lagi dibagi menjadi empat subsistem yaitu:

- 1. Fault Detection and Diagnosis (FDD)
- 2. Reconfigurable Controller
- 3. Controller Reconfiguration Mechanism
- 4. Command/Reference Governor

FDD dan *reconfigurable controller* adalah subsistem yang membedakan AFTC dengan PFTC. FDD adalah awal mula untuk mendesain AFTC. Dalam FDD terdapat beberapa tahap:

1). Fault Detection

Pada tahap ini, kesalahan dideteksi dari bagian komponen yang mengalami degradasi fault, *fault detection* juga dinamakan residual.

2). Fault Isolation

Pada tahap ini ditentukan komponen mana yang mengalami kesalahan (sensor, aktuator atau proses).

3). Fault Identification

Pada tahap ini ditentukan jenis kesalahan apakah bersifat aditif atau multiplikatif, *abrupt fault* atau *incipent fault*.

FDD yang didesain juga harus mendeteksi adanya kesalahan dengan cepat yang ada dalam sistem, hal inilah yang membuat AFTC memiliki banyak kelebihan dibanding PFTC. FDD dalam AFTC juga dapat beradaptasi dengan cepat terhadap waktu, sedangkan PFTC lebih lambat, hal ini mengakibatkan sistem di era sekarang lebih banyak menerapkan AFTC untuk meningkatkan performa.

FDD yang paling sederhana adalah *redundancy* pada *hardware* sistem. *Redundancy* ini dapat berupa sensor atau komponen lain. FDD yang menjadi topik riset populer saat ini adalah FDD menggunakan model dari *plant*. Secara umum, berdasarkan model yang digunakan terdapat dua metode FDD; *model-based* dan *model-free*.

1). Model-free FDD

FDD ini tidak menggunakan model atau data dari *plant* untuk menentukan kesalahan sensor atau aktuator pada sistem, contoh nyata dari model ini penerapan beberapa sensor untuk mengatasi kesalahan sensor yang ada pada *plant*, dapat dilihat pada Gambar 2.9



Gambar 2.9 Hardware Redundancy pada Sistem [2] [4]

Beberapa sensor diletakkan *parallel* terhadap sensor utama, perbedaan kerja dari dua atau lebih sensor memunculkan peringatan/alarm untuk menunjukkan telah terjadi kesalahan sensor. Sistem ini memungkinkan penggantian sensor yang rusak dengan cepat. Beberapa metode lain dalam *model-free* adalah *limit checking* dan *frequency spectral analysis*.

Model-based FDD

2).

Model ini dikembangkan sekitar tahun 1980-an sebagai metode baru FTC. Inti dari konsep model-based FDD adalah membandingkan model real-*plant* dengan model tiruan yang ada di komputer/algoritma. Perbedaan dari kedua model disebut dengan *residual*. Dari *residual* ini ditentukan apakah terjadi kesalahan sensor/aktuator atau tidak pada sistem. Konsep ini memiliki banyak kelebihan dibandingkan model-*free*, salah satunya adalah penghematan komponen yang akan dibeli seperti halnya *hardware redundancy*. Skema model-based dapat dilihat pada Gambar 2.10



Gambar 2.10 Skema Model-Based Fault Diagnosis [2] [4]

Pada *model-based*, terdapat tiga metode untuk menentukan *residual generation*, yaitu:

a. Observer-based FDD

Metode *observer* menggunakan estimasi parameter dari keluaran. *Residual* didapatkan dengan mengestimasi error antara sinyal yang terukur dan sinyal yang terestimasi. Kemudahan struktur *observer* menjadi metode FDD yang paling mudah dikembangkan.

b. Parity relation based FDD

Pada pendekatan ini, sinyal residual didapatkan dari pengukuran konsisten masukan dan keluaran sistem pada rentang waktu tertentu.

c. Parameter estimation

Metode ini adalah pengembangan dari sistem *observer*. Kesalahan direpresentasikan sebagai parameter dalam sistem, kemudian parameter tersebut diestimasi. Hasil dari estimasi digunakan untuk mengkompensasi kesalahan tadi.

Langkah selanjutnya setelah FDD adalah reconfigurable controller. Metode reconfigurable controller digunakan untuk memberikan sinyal kontrol yang baru pada sistem, sehingga sinyal kontrol tersebut bebas kesalahan telah dari sensor/aktuator Reconfigurable controller tidak harus ada pada sistem AFTC, metode ini dikembangkan pada sistem *plant* tiga tangki untuk mengatasi sensor dan aktuator. Namun pada pengembangan riset selanjutnya, metode kompensasi lebih sederhana.



Gambar 2.11 Struktur Lengkap AFTC [2]

2.11. Observer [3]

Observer adalah metode untuk mengestimasi state dari suatu sistem. Dalam sistem kontrol, khususnya sistem lup tertutup, tidak semua state sistem dapat diukur, sehingga diperlukan metode untuk mengestimasi state tersebut. Tinjaulah suatu sistem:

 $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$

 $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$

Estimasi menggunakan observer, memiliki persamaan:

$$\dot{\widetilde{\mathbf{x}}}(t) = \mathbf{A}\widetilde{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{K}_{e}(\mathbf{y}(t) - \mathbf{C}\widetilde{\mathbf{x}}(t))$$

(2.20)

Persamaan (2.20) memiliki residual dari hasil pengurangan keluaran sebenarnya dikurangi dengan keluaran estimasi. *Residual* ini dikalikan dengan gain K_e yaitu observer gain. Struktur dari observer dapat dilihat pada Gambar 2.10.

2.12. Proportional-Proportional Integral Observer [4] [11] [12]

Metode estimasi state ini dapat meningkatkan efisiensi dalam suatu sistem, misalnya menggantikan fungsi sensor yang mahal untuk menggantikan kesalahan sensor pada plant. Terdapat beberapa jenis observer, berdasarkan derivatifnya yaitu proporsional, proporsional integral, dan proporsional integral derivatif. Perbedaan dari ketiga observer tersebut adalah penentuan observer gainnya. Observer proporsional mempunyai satu observer gain, proporsional integral mempunyai dua observer gain, dan proporsional integral derivatif observer mempunyai tiga observer. Penambahan integral disini dimaksudkan untuk menghilangkan error estimasi.



Gambar 2.12 Blok Diagram Observer [3]

Persamaan *Proportional-Proportional Integral Observer* (PPIO) dapat dilihat pada persamaan (2.21)

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \hat{\mathbf{p}}(t) + \mathbf{P}\mathbf{e}_{y}(t)$$
$$\dot{\hat{\mathbf{p}}} = \mathbf{L}(\mathbf{e}_{y}(t) + \dot{\mathbf{e}}_{y}(t))$$
$$\mathbf{e}_{y}(t) = \mathbf{y}(t) - \hat{\mathbf{y}}(t)$$
$$\hat{\mathbf{y}}(t) = \mathbf{C}\hat{\mathbf{x}}(t)$$

Dari persamaan (2.21) terlihat observer PPIO memiliki dua gain yaitu P dan L. Gain P adalah gain proportional, gain L adalah gain integral. Penambahan gain integral disini dimaksudkan untuk mempercepat estimasi yang didapatkan. Penambahan gain integral dalam PPIO juga dinamakan Fast Fault Estimation (FFE). Respon transion dari estimasi yang didapatkan sangat cepat dan error dinamika yang didapatkan selalu menuju nol. Struktur observer PPIO dapat dilihat pada Gambar 2.13.

(2.21)



2.13. Estimasi dan Kompensasi [1] [4]

Secara konvensional FTC memastikan kontrol nominal dengan baik dengan adanya FDD dan controller reconfiguration. Pada subbab 2.8 dijelaskan bahwa controller reconfiguration adalah mekanisme konvensional dalam kompensasi kesalahan sensor atau aktuator. Karena FDD harus bereaksi cepat terhadap kesalahan yang terjadi, terdapat konsep yang lebih sederhana dalam AFTC, yaitu estimasi dan Konsep dari estimasi dan kompensasi kompensasi. adalah menghilangkan efek dari kesalahan sensor dan aktuator dengan mengurangi sinyal hasil estimasi dengan sinyal sebenarnya yang telah terjadi kesalahan, sehingga saling menjadakan. Estimasi dan kompensasi telah dikembangkan bertahun-tahun untuk membuat struktur AFTC menjadi lebih sederhana. Secara umum konsep kompensasi yang dimaksud adalah:

 $u(t) = u_n(t) + u_{add}(t)$ $u_{add}(t) = K_f \hat{f}(t); f(t) \neq 0$ $u_{add}(t) = 0; f(t) = 0$

(2.22)

Sinyal u(t) adalah sinyal kontrol. $u_n(t)$ adalah sinyal kontrol ketika sistem tidak terjadi kesalahan. $u_{add}(t)$ adalah sinyal penambahan dari hasil estimasi kesalahan yang didapatkan dari obsever FTC. Ketika terjadi kesalahan $u_{add}(t)$ bekerja dengan memberikan tambahan sinyal kontrol yaitu $K_{ffs}(t)$. Ketika tidak terjadi kesalahan, $u_{add}(t)=0$. Penambahan ini mengakibatkan $u_n(t) + u_{add}(t) = 0$ kembali ke sinyal kontrol yang diinginkan. Estimasi adalah hal yang sangat penting dalam metode AFTC ini, karena besarnya estimasi akan ditambahkan ke sinyal kontrol sebenarnya, maka estimasi kesalahan sensor atau aktuator harus akurat.

2.14. Linear Matrix Inequality (LMI) [13]

LMI adalah penyederhanaan metode penyelesaian pertidaksamaan. Dalam sistem kontrol, LMI sangat berguna untuk menyelesaikan masalah kestabilan. Masalah kestabilan yang sering diselesaikan dalam LMI adalah kestabilan Lyapunoy. Bentuk umum dari LMI adalah:

$$\mathbf{F}(x) = \mathbf{F}_0 + \sum_{i=1}^m x_i \mathbf{F}_i > 0$$

(2.23)

dimana $x \in R^m$ adalah variabel dan \mathbf{F}_0 , \mathbf{F}_i adalah konstanta matriks simetris. LMI berikut adalah dasar dari analisa kestabilan Lyapunov:

 $\mathbf{P}\mathbf{A} + \mathbf{A}^T \mathbf{P} < 0 \tag{2.24}$

dengan variabel $\mathbf{P} \in \mathbb{R}^{n\times n}$, syarat $\mathbf{P} > 0$. Lyapunov menunjukkan secara analitik bahwa persamaan ini dapat diselesaikan dengan memilih $\mathbf{Q} = \mathbf{Q}^{T} > 0$ dan kemudian menyelesaikan persamaan linear $\mathbf{A}^{T}\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{Q}$ untuk mendapatkan matriks definit positif \mathbf{P} . Untuk perhitungan beberapa pertidaksamaan, mudah diselesaikan. Namun untuk orde tinggi dan syarat yang banyak perhitungan sangat sulit dilakukan. Karena itulah dalam LMI dikenal istilah Schur Complement. Persamaan Schur Complement mengubah bentuk pertidaksamaan matriks menjadi bentuk LMI yaitu:

$$\frac{\mathbf{Q}(x)}{\mathbf{S}(x)^{T}} \frac{\mathbf{S}(x)}{\mathbf{R}(x)} < 0$$
(2.25)

dengan $\mathbf{Q}(\mathbf{x}) = \mathbf{Q}(\mathbf{x})^{T}$ dan $\mathbf{R}(\mathbf{x}) = \mathbf{R}(\mathbf{x})^{T}$ adalah ekuiyalen dan dapat dinyatakan dengan

$$\mathbf{R}(x) > 0$$

$$\mathbf{Q}(x) - \mathbf{S}(x)\mathbf{R}(x)^{-1}\mathbf{S}(x)^{T} > 0$$
(2.26)

beberapa pertidaksamaan dapat digabungkan menjadi satu. Misalkan terdapat persamaan:

$$\mathbf{F}_{0} = \begin{bmatrix} F_{01} & 0 & 0 & 0 \\ 0 & F_{02} & 0 & 0 \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & F_{0p} \end{bmatrix} = diag(F_{01}, \dots F_{0p})$$
$$\mathbf{G}_{i} = diag(G_{i1}, \dots G_{ip})$$
$$\mathbf{H}_{i} = diag(H_{i1}, \dots H_{ip})$$
$$\mathbf{V}_{i} = diag(V_{i1}, \dots V_{ip})$$

Persamaan \mathbf{F}_{i} , \mathbf{G}_{i} , \mathbf{H}_{i} dan \mathbf{V}_{i} dapat digabung dalam bentuk:
$$\mathbf{F}_{big}(V_1,\ldots,V_n) := \mathbf{F}_0 + \sum_{i=1}^n \mathbf{G}_i \mathbf{V}_i \mathbf{H}_i > 0$$

2.15. LMI Region dan Pole Placement LMI Region [14]

Kestabilan adalah masalah yang paling banyak dibahas dalam sistem kontrol, hingga berbagai macam metode diterapkan memenuhi kriteria performa yang diinginkan. Pada sistem yang harus memiliki respon cepat seperti pendulum, *overhoot* harus seminimal mungkin dan respon transien secepat mungkin. Kriteria *overshoot* dan respon transien dipengaruhi oleh letak *pole-pole* pada lup tertutup. Terdapat beberapa cara untuk meletakkan *pole-pole* pada daerah yang diinginkan, salah satunya adalah *pole placement*. *Pole placement* menggunakan LMI adalah salah satu metode yang cukup sederhana untuk menentukan *gaingain* matrik kestabilan. Hal ini dikarenakan pada LMI, dapat ditambahkan syarat kestabilan, seperti batasan *input-output*, atau kriteria lain. Daerah *pole-pole* pada LMI didefinisikan oleh Gambar 2.14.

(2.27)



Gambar 2.14 Daerah pada LMI Region (Conic Sector+Half Plane) [14]

D adalah daerah pada bidang kompleks yang dinamakan LMI region, θ adalah sudut batas dimana pole-pole harus terdapat didalamnya, x adalah batas sumbu minimal pole harus berada di dalamnya. Daerah ini ada jika ada matrik simetris L dan matrik M sehingga:

$$\mathbf{D} = \left\{ z \in C : f_d(z) < 0 \right\}$$
(2.28)

dengan $f_d(z) = L + zM + zM^T \operatorname{dan} f_d(z)$ dinamakan fungsi karakteristik dari daerah bidang kompleks **D**. Contoh dari LMI region adalah:

- 1. Half Plane (setengah bidang) $Re(z) = -\alpha, f_1(z) = z + z + 2\alpha < 0$ (2.29)
- 2. Disk (lingkaran) berpusat di (-q, 0) dengan radius r

$$f_d(z) = \begin{bmatrix} -r & q+z \\ z+\overline{z} & -r \end{bmatrix} < 0$$
(2.30)

3. Conic Sector (kerucut) 2θ

$$f_d(z) = \begin{bmatrix} \sin\theta(z+\overline{z}) & \cos\theta(z+\overline{z}) \\ \cos\theta(z+\overline{z})^T & \sin\theta(z+\overline{z}) \end{bmatrix} < 0$$
(2.31)

Daerah D sebagai daerah *pole-pole* yang diinginkan, memiliki karakteristik berbeda untuk setiap sistem. Gambar 2.14 menggambarkan daerah D dalam bentuk kerucut dan *half plane* yang digabung sehingga menghasilkan bentuk trapesium. Daerah pada masing-masing bidang kompleks dapat dikombinasikan menghasilkan D yang berbeda. Sistem lup tertutup dengan persamaan dinamika kestabilan A+BK, memiliki *pole-placement* dalam bentuk LMI sebagai berikut:

 $\mathbf{P} > 0$

$$(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} + \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T} + 2\gamma\mathbf{P} < 0$$

 $\begin{bmatrix} \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} + \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^T] \\ \cos\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} - \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^T]^T \\ \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P} + \mathbf{P}(\mathbf{A} + \mathbf{B}\mathbf{K})^T] \end{bmatrix} < 0$ (2.32)

Sistem lup tertutup A+BK stabil jika dan hanya jika terdapat matrik **P** yang memenuhi LMI pertidaksamaan (2.32). Karena terdapat dua variabel yaitu **P** dan **K**, maka perlu membentuk LMI pertidaksamaan (2.32) menjadi bentuk linear dengan membuat variabel baru, yaitu **Y** = **KP**, sehingga persamaan (2.33) menjadi:

P > 0

$$\mathbf{AP} + \mathbf{PA}^{T} + \mathbf{BY} + \mathbf{Y}^{T}\mathbf{B}^{T} + 2\gamma\mathbf{P} < 0$$

$$\sin\theta[\mathbf{AP} + \mathbf{PB}^{T} + \mathbf{BY} + \mathbf{Y}^{T}\mathbf{B}^{T}]$$

$$\cos\theta[\mathbf{AP} - \mathbf{PA}^{T} + \mathbf{BY} - \mathbf{Y}^{T}\mathbf{B}^{T}]^{T}$$

 $\cos \theta [\mathbf{A}\mathbf{P} - \mathbf{P}\mathbf{A}^{T} + \mathbf{B}\mathbf{Y} - \mathbf{Y}^{T}\mathbf{B}^{T}]] < 0$ $\sin \theta [\mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^{T} + \mathbf{B}\mathbf{Y} + \mathbf{Y}^{T}\mathbf{B}^{T}]] < 0$ (2.33)

2.16. Batasan Input-Output [8]

Dalam implementasi nyata, ketidakpastian parameter serta gangguan dari luar dapat diberikan pada sistem. Selain itu, dalam *real plant, state* yang terukur bisa terbatas dan sinyal kontrol yang diberikan juga terbatas. Agar dapat diimplementasikan dalam sistem nyata, besaran-besaran penting seperti sinyal kontrol perlu dibatasi, agar tidak terjadi guncangan atau kerusakan. Suatu *plant* yang memiliki:

 $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$

 $\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t)$

Untuk memberikan batasan input-output dimodifikasi menjadi

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{z}_{1}(t) = \mathbf{C}_{z1}\mathbf{x}(t)$$
$$\mathbf{z}_{2}(t) = \mathbf{C}_{z2}\mathbf{x}(t)$$
$$\mathbf{y}(t) = \mathbf{C}_{y}\mathbf{x}(t)$$

dengan $\mathbf{u}(t)$ adalah sinyal kontrol, $\mathbf{z}_1(t)$ adalah keluaran performa, $\mathbf{z}_2(t)$ adalah keluaran yang dibatasi, dan $\mathbf{y}(t)$ adalah pengukuran untuk kontroler. Jika $\mathbf{C}_{\mathbf{y}}$ merupakan matriks identitas, maka $\mathbf{y}(t) = \mathbf{x}(t)$. Karena sinyal kontrol yang digunakan adalah $\mathbf{u}(t) = -\mathbf{K}\mathbf{y}(t) = -\mathbf{K}\mathbf{x}(t)$, maka persamaan (2.20) menjadi:

$$\dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t)$$
$$\mathbf{z}_{1}(t) = \mathbf{C}_{z1}\mathbf{x}(t)$$
$$\mathbf{z}_{2}(t) = \mathbf{C}_{z2}\mathbf{x}(t)$$

(2.35)

(2.34)

Sehingga perumusan untuk batasan pada sinyal kontrol u(t) dapat dinyatakan sebagai berikut:

 $\|\mathbf{u}(t)\| \leq u_{\max}$

 $\left\| -\mathbf{K}\mathbf{x}(t) \right\| \le \left\| \mathbf{u}(t) \right\| \le u_{\max}$

Sedangkan perumusan untuk batasan pada keluaran yang dibatasi $\frac{z_2(t)}{z_2(t)}$ dapat dinyatakan sebagai berikut:

$$\begin{aligned} \|\mathbf{Z}(t)\| &\leq \mathbf{Z}_{2\max} \\ \|\mathbf{C}_{z2}\mathbf{x}(t)\| &\leq \|\mathbf{Z}_{2}(t)\| \leq z_{2\max} \end{aligned}$$

2.17. Performa H_{∞} [8]

Pengembangan metode kontrol *robust* telah menjadi fokus utama dalam dua dekade ini khususnya dalam komunitas kotnrol. Ketahanan suatu sistem kontrol terhadap ganggaun *(disturbance)* dan ketida kpastian *(uncertainty)* selalu menjadi topik utama dalam pembahasan masalah *feedback control*. Masalah *feedback* tidak akan terlalu menarik pada kebanyakan sistem kontrol jika didalamnya tidak terdapat gangguan dan ketidakpastian.

Pemberian kontrol pada suatu sistem bertujuan untuk membuat sistem tersebut *robust* terhadap gangguan sesuai dengan keinginan. Desain kontrol H_{∞} atau dengan H_2 menjamin ketahanan suatu sistem tetapi pada umumnya desain kontrol dengan menggunakan H_{∞} lebih banyak diminati dibanding desain kontrol dengan H_2 karena kontrol yang dihasilkan lebih *robust*. Pengaruh buruk gangguan terhadap keluaran performa dapat diketahui dengan menghitung ∞ -norm dari fungsi alih sistem. ∞ -norm dari suatu fungsi alih didefinisikan sebagai

$$G(s) = \sup_{\alpha} G(j\omega)$$

(2.38)

Dari persamaan $(2.21) \infty$ -norm dari suatu fungsi alih adalah nilai maksimal dari magnitude respon frekuensi. Jika G(s) menyatakan fungsi alih keluaran performa terhadap gangguan, dan ∞ -norm dari G(s) bernilai kecil, maka dapat dikatakan bahwa gangguan yang diberikan teredam sesuai besarnya ∞ -norm.

Gambar 2.15 menunjukkan diagram blok sistem dengan dua masukan dan dua keluaran w(t) adalah gangguan dari luar, u(t) adalah sinyal kontrol, z(t) adalah keluaran performa, dan y(t) adalah pengukuran untuk kontroler. Dalam bentuk state space, sistem pada Gambar 2.15 dapat ditulis sebagai berikut:

(2.36)

(2.37)



Sistem pada Gambar 2.15 ditulis dalam bentuk persamaan:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_{w}\mathbf{w}(t) + \mathbf{B}_{u}\mathbf{u}(t)$$

$$\mathbf{z}(t) = \mathbf{C}_{z}\mathbf{x}(t)$$

$$\mathbf{y}(t) = \mathbf{C}_{v}\mathbf{x}(t)$$
(2.39)

Jika sinyal kontrol yang digunakan adalah $\mathbf{u}(t) = -\mathbf{K}\mathbf{y}(t)$, maka sistem dapat disederhanakan menjadi sistem dengan satu masukan yaitu $\mathbf{w}(t)$ dan satu keluaran yaitu $\mathbf{z}(t)$ dengan bentuk state space.

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{w}(t)$$

$$\mathbf{z}(t) = \mathbf{C}\mathbf{x}(t)$$

(2.40)

dan dengan

$$\overline{\mathbf{A}} = (\mathbf{A} - \mathbf{B}_u \mathbf{K} \mathbf{C}_y),$$
$$\overline{\mathbf{B}} = \mathbf{B}_w,$$
$$\overline{\mathbf{C}} = \mathbf{C},$$

Gambar 2.16 menunjukkan diagram blok sistem sesuai dengan persamaan (2.40). Jika $T_{zw}(s)$ menunjukkan fungsi alih Z(s) terhadap W(s), maka $T_{zw}(s)$ dapat dihitung sebagai berikut:

 $T_{zw}(s) = \overline{\mathbf{C}}(s\mathbf{I} - \overline{\mathbf{A}})^{-1}\overline{\mathbf{B}}$ $T_{zw}(s) = \overline{\mathbf{C}}_{z}(s\mathbf{I} - \mathbf{A} + \mathbf{B}_{u}\mathbf{K}\mathbf{C}_{y})^{-1}\mathbf{B}_{w}$

(2.41)



Gambar 2.16 Diagram Blok Sistem Persamaan (2.40) [8]

Secara sederhana sistem pada Gambar 2.16 dapat ditulis dengan mengubah kedalam bentuk transformasi Laplace seperti pada Gambar 2.17.

$$W(s) \longrightarrow T_{ZW}(s) \longrightarrow Z(s)$$

Gambar 2.17 Penyederhanaan Sistem dalam Bentuk LFT [8]

hubungan antara masukan W(s) dan keluaran Z(s) dapat dinyatakan dengan persamaan:

$$\frac{Z(s)}{W(s)} = T_{zw}(s) \tag{2.42}$$

Sistem dalam bentuk LFT pada Gambar 2.17 didesain agar masukan yang berupa gangguan kurang dari pelemahan/atenuasi tertentu. Performa H_{∞} dicari menggunakan bentuk L_2 -gain yang dituliskan dalam persamaan:

$$\left\|\mathbf{w}(t)\right\|_{2} = \sqrt{\int_{0}^{\infty} \mathbf{w}(t)^{T} \mathbf{w}(t) dt}$$

(2.43)

maka L_2 -gain atau perbandingan L_2 -norm $\mathbf{z}(t)$ terhadap L_2 -norm $\mathbf{w}(t)$ adalah:

$$\left\| \mathbf{z}(t) \right\|_{2} = \sqrt[\infty]{0} \mathbf{z}(t)^{T} \mathbf{z}(t) dt$$

$$\left\| \mathbf{w}(t) \right\|_{2} = \sqrt[\infty]{0} \mathbf{w}(t)^{T} \mathbf{w}(t) dt$$
(2.44)

 ∞ -norm dari fungsi alih $T_{zw}(s)$ dapat dihitung dari maksimal L_2 -gain $\mathbf{z}(t)$ terhadap $\mathbf{w}(t)$. Sesuai definisi tersebut, maka ∞ -norm dari fungsi alih $T_{zw}(s)$ dapat dihitung dengan

$$\|T_{zw}(s)\|_{\infty} = \sup_{\boldsymbol{\omega}} |T_{zw}(j\boldsymbol{\omega})| = \sup_{\|\boldsymbol{w}(t)\|_{2} \neq 0} \frac{\|\mathbf{z}(t)\|_{2}}{\|\mathbf{w}(t)\|_{2}}$$
(2.45)

Jika tingkat pelemahan maksimal yang diinginkan adalah kurang dari γ maka performa desain yang digunakan dalam kontrol *robust* dapat dinyatakan dalam persamaan (2.46)

$$\|T_{zw}(s)\|_{\infty} = \gamma^* < \gamma$$

$$\|w(t)\|_{2} = \gamma^* < \gamma$$
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)
(2.46)

BAB III PERANCANGAN SISTEM

Pada Bab ini akan dibahas perancangan kontrol nominal (kontrol sistem pendulum kereta tanpa kesalahan sensor dan aktuator). Kontrol nominal dirancang menggunakan metode LMI *pole placement* dan batasan *input output*. Pencarian *gain* kontrol nominal menggunakan metode LMI *pole placement* yang menjamin *pole-pole* sistem lup tertutup berada pada daerah kestabilan yang diinginkan. *Observer* kesalahan sensor dan aktuator dirancang menggunakan metode PPIO (*Proportional Proportional Integral Observer*) dengan pencarian *gain* berbasis LMI yang menjamin kestabilan Lyapunov.

4.1. Linearisasi Model Matematika Sistem Pendulum Kereta

Untuk mendesain kontrol *fuzzy* T-S, model matematika pada persamaan (2.4) dilinerisasi di titik daerah kerjanya. Titik ekuilibrium sistem harus diketahui terlebih dahulu. Persamaan (2.4) ditulis kembali:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + h(\mathbf{x}(t), \mathbf{u}(t))$$
(3.1)

dengan

$$f(\mathbf{x}(t)) = \begin{bmatrix} f_1(\mathbf{x}(t)) \\ f_2(\mathbf{x}(t)) \\ f_3(\mathbf{x}(t)) \\ f_4(\mathbf{x}(t)) \end{bmatrix} = \begin{bmatrix} u & x_3 \\ x_4 \\ \frac{a(u - T_c - \mu x_4^2 \sin x_2) + l \cos x_2(\mu g \sin x_2 - f_p x_4)}{J + \mu' \sin^2 x_2} \\ \frac{J + \mu' \sin^2 x_2}{J + \mu' \sin^2 x_2} \end{bmatrix}$$

$$h(\mathbf{x}(t)) = \begin{bmatrix} h_1(\mathbf{x}(t), \mathbf{u}(t)) \\ h_2(\mathbf{x}(t), \mathbf{u}(t)) \\ h_3(\mathbf{x}(t), \mathbf{u}(t)) \\ h_4(\mathbf{x}(t), \mathbf{u}(t)) \\ h_4(\mathbf{x}(t), \mathbf{u}(t)) \end{bmatrix} = \begin{bmatrix} 0 & 0 & \frac{au}{J + \mu' \sin^2 x_2} & \frac{ul \cos x_2}{J + \mu' \sin^2 x_2} \end{bmatrix}$$

Saat sistem berada pada titik ekuilibrium, laju perubahan state sistem bernilai nol. Jika sistem diberi kondisi awal yang berada pada titik ekuilibrium, maka state sistem akan selalu berada pada titik tersebut. Dari persamaan (3.1), titik ekuilibrium sistem pendulum kereta dinvatakan:

$$\mathbf{x}_{e}(t) = \begin{bmatrix} x_{1e} & x_{2e} & x_{3e} & x_{4e} \end{bmatrix}^{T} \operatorname{dan} \mathbf{u}_{e}(t)$$
(3.2)

nilai titik ekuilibrium persamaan (3.2) disubtitusikan persamaan (3.1) sehingga dinyatakan:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}_e(t)) + h(\mathbf{x}_e(t), \mathbf{u}_e(t))$$

 $0 = f(\mathbf{x}_{e}(t)) + h(\mathbf{x}_{e}(t), \mathbf{u}_{e}(t))$

solusi persamaan (3.3) adalah:

$$\mathbf{x}_e(t) = \begin{bmatrix} c & n\pi & 0 & 0 \end{bmatrix}^I \, \mathrm{dan} \, \mathbf{u}_e(t) = 0 \tag{3.4}$$

dengan *n* bilangan bulat dan *c* adalah real.

Kontrol *tracking* dirancang agar sistem mampu mengikuti trayektori yang diinginkan, namun tetap mempertahankan sudut pendulum pada 0 radian. Sudut 0 radian memenuhi persamaan (3.4) sehingga sistem pendulum kereta dapat dilinearisasi pada titik ini.

Persamaan (3.4) menunjukkan linearisasi lokal hanya dapat divariasi pada x_1 dan x_2 . Pada tugas akhir ini nilai x_1 dipilih nol, sedangkan nilai x_2 divariasi pada 3 titik yaitu 0, ±0,2616 dan ±0,5232 radian. Model sistem linear pada $x^* = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T dan u^* = 0$, $x^* = \begin{bmatrix} 0 \\ 0 & 0 \end{bmatrix}^T$ $\pm 0,2616 \ 0 \ 0]^{\mathrm{T}} \operatorname{dan} u^* = 0$, serta $x^* = \begin{bmatrix} 0 \ \pm 0,5232 \ 0 \ 0 \end{bmatrix}^{\mathrm{T}} \operatorname{dan} u^* = 0$ digambarkan dalam 3 subsistem sesuai persamaan (3.5), persamaan (3.6) dan persamaan (3.7). Penurunan perhitungan masing-masing elemen matriks dapat dilihat pada lampiran A.2

$$\dot{\mathbf{x}}(t) = \mathbf{A}_1 \mathbf{x}(t) + \mathbf{B}_1 \mathbf{u}(t)$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}_2 \mathbf{x}(t) + \mathbf{B}_2 \mathbf{u}(t)$$

$$\dot{\mathbf{x}}(t) = \mathbf{A}_3 \mathbf{x}(t) + \mathbf{B}_3 \mathbf{u}(t)$$
(3.5)
(3.6)
(3.7)

dengan

$$\mathbf{A}_{1} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2526 & 0 & -0,0001 \\ 0 & 15,0421 & 0 & -0,0079 \end{bmatrix}, \mathbf{B}_{1} = \begin{bmatrix} 0 \\ 0 \\ 0,8272 \\ 1,2370 \end{bmatrix}$$
$$\mathbf{A}_{2} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0,2176 & 0 & -0,0001 \\ 0 & 14,4555 & 0 & -0,0079 \end{bmatrix}, \mathbf{B}_{2} = \begin{bmatrix} 0 \\ 0 \\ 0,8258 \\ 1,1929 \end{bmatrix}$$
$$\mathbf{A}_{3} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0,1232 & 0 & -0,0001 \\ 0 & 12,7812 & 0 & -0,0079 \end{bmatrix}, \mathbf{B}_{3} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0,8219 \\ 1,0647 \end{bmatrix}$$

Model sistem pada persamaan (3.5), (3.6) dan (3.7) adalah subsistem untuk masing-masing titik kerja $x_2 = 0, \pm 0, 2616$ dan $\pm 0,5232$ rad.

4.2. Pemodelan Fuzzy Takagi-Sugeno (T-S)

Fuzzy T-S dibangun dari hasil linearisasi lokal sistem pendulum kereta pada Subbab 3.1. Sistem pendulum kereta dilinearisasi pada tiga titik kerja, sehingga model *fuzzy* T-S yang digunakan memiliki tiga aturan dan satu premis yaitu sudut pendulum. Dari subsistem persamaan (3.5), (3.6) dan (3.7), model *fuzzy* T-S dapat dibentuk dengan aturan *plant* sebagai berikut:

Aturan model *plant* ke-1 If $x_2(t)$ is M_1 (sekitar 0 rad) Then $\dot{\mathbf{x}}(t) = \mathbf{A}_1 \mathbf{x}(t) + \mathbf{B}_1 \mathbf{u}(t)$ $\mathbf{y}(t) = \mathbf{C}_1 \mathbf{x}(t)$ $\mathbf{z}(t) = \mathbf{C}_{-1} \mathbf{x}(t)$

Aturan model *plant* ke-2 If $x_2(t)$ is M₂ (sekitar ±0,2616 rad) Then $\dot{\mathbf{x}}(t) = \mathbf{A}_2 \mathbf{x}(t) + \mathbf{B}_2 \mathbf{u}(t)$ $\mathbf{y}(t) = \mathbf{C}_2 \mathbf{x}(t)$ $\mathbf{z}(t) = \mathbf{C}_{z,2} \mathbf{x}(t)$

Aturan model *plant* ke-3 If $x_2(t)$ is M₃ (sekitar ±0,5232 rad) Then $\dot{\mathbf{x}}(t) = \mathbf{A}_3 \mathbf{x}(t) + \mathbf{B}_3 \mathbf{u}(t)$

 $\mathbf{y}(t) = \mathbf{C}_3 \mathbf{x}(t)$ $\mathbf{z}(t) = \mathbf{C}_{2,2} \mathbf{x}(t)$

dengan matriks keluaran, matriks transmisi langsung, $\mathbf{C}_{z,1}$ dan $\mathbf{C}_{z,2}$ sebagai berikut:

$$\mathbf{C}_{1} = \mathbf{C}_{2} = \mathbf{C}_{3} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$
$$\mathbf{C}_{z1} = \mathbf{C}_{z2} = \mathbf{C}_{z3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Sesuai dengan konsep *Parallel Distributed Compensation* (PDC) aturan kontroler disusun sebagai berikut:

Aturan kontroler ke-1 If $x_2(t)$ is M₁ (sekitar 0 rad) Then $\mathbf{u}(t) = \mathbf{K}_{n1}\mathbf{x}(t)$

Aturan kontroler ke-2 If $x_2(t)$ is M₂ (sekitar ±0,2616 rad) Then $\mathbf{u}(t) = \mathbf{K}_{n2}\mathbf{x}(t)$

Aturan kontroler ke-3 (()) If $x_2(t)$ is M₃ (sekitar ±0,5232 rad) Then $\mathbf{u}(t) = \mathbf{K}_{n3}\mathbf{x}(t)$

Fungsi keanggotaan M_1 dipilih fungsi segitiga seperti pada Gambar 3.1. Begitu juga fungsi keanggotaan M_2 dan M_3 . Fungsi keanggotaan M_1 , M_2 dan M_3 adalah fungsi dari sudut pendulum. Persamaan matematis untuk masing-masing fungsi keanggotaan ditunjukkan persamaan (3.8), (3.9) dan (3.10). Inferensi *fuzzy* menggunakan penghubung AND dan metode defuzzifikasi yang digunakan adalah rata-rata bobot. Bentuk keseluruhan model *fuzzy* T-S ditunjukkan oleh persamaan (3.11).



$$\mathbf{y}(t) = \frac{\sum_{j=1}^{r} \left[w_{j}(x) \{ \mathbf{C}_{j} \mathbf{x}(t) \} \right]}{\sum_{j=1}^{r} w_{j}(x)} = \sum_{j=1}^{r} h_{j}(x) \{ \mathbf{C}_{j} \mathbf{x}(t) \}$$
$$\mathbf{z}(t) = \frac{\sum_{j=1}^{r} \left[w_{j}(x) \{ \mathbf{C}_{z,j} \mathbf{x}(t) \} \right]}{\sum_{j=1}^{r} w_{j}(x)} = \sum_{j=1}^{r} h_{j}(x) \{ \mathbf{C}_{z,j} \mathbf{x}(t) \}$$

dengan,

$$w_{j}(x) = \prod_{k=1}^{p} M_{jk}(x_{k})$$
$$h_{j}(x) = \frac{w_{j}(x)}{\sum_{j=1}^{r} w_{j}(x)}$$
$$0 \le h_{j}(x) \le 1$$
$$\sum_{j=1}^{r} h_{j}(x) = 1$$

Sedangkan kontroler Fuzzy menurut aturan PDC dapat ditulis dengan:

$$\mathbf{u}(t) = \sum_{j=1}^{r} h_j(\mathbf{x}) \mathbf{K}_{n,j} \mathbf{x}(t)$$

Penjabaran sinyal kontrol menurut fungsi keanggotaan setiap subsistem akan menghasilkan persamaan (3.11)

$$\mathbf{u}(t) = M_1(x_2(t))\mathbf{K}_{n,1}\mathbf{x}(t) + M_2(x_2(t))\mathbf{K}_{n,2}\mathbf{x}(t) + M_1(x_2(t))\mathbf{K}_{n,1}\mathbf{x}(t)$$
(3.11)

dengan $\mathbf{K}_{n,j}$ (j = 1,2,3) adalah kontroler gain state feedback yang akan dicari menggunakan LMI pole placement.

4.3. Perancangan Kontroler *Fuzzy Pole Placement* dengan Batasan *Input-Output*

Pada subbab ini akan dijelaskan desain kontroler untuk sistem lup tertutup sistem pendulum kereta. Desain kontroler ini disebut juga kontrol nominal (kontrol dimana kesalahan sensor dan aktuator tidak terjadi). Kontroler memastikan *pole-pole* sistem lup tertutup dari sistem pendulum kereta berada di sebelah kiri sumbu imajiner dan daerah tertentu (*region*). Untuk menempatkan *pole-pole* pada daerah tertentu digunakan LMI *region*. Untuk menjamin sistem lup tertutup menghasilkan sinyal kontrol yang tidak terlalu besar (untuk keperluan implementasi) digunakan batasan *input-output*. Kombinasi *constraint* dari LMI region dan batasan *input-output* menghasilkan desain LMI dari kontroler secara keseluruhan. Desain kontroler secara keseluruhan dapat dilihat pada Gambar 3.2.



Gambar 3.2 Desain Kontroler Nominal Keseluruhan

4.3.1. Perancangan Kontroler berbasis Model Referensi

Kontroler dirancang mengikuti trayektori yang diinginkan. Pada tugas akhir ini dirancang kontroler berbasis kompensator. Kompensator dirancang berdasarkan sinyal referensi dan gangguan. Sesuai dengan persamaan (2.14) dan penjelasan Subbab (2.7) persamaan augmentasi ini dimodifikasi dalam bentuk *fuzzy* T-S untuk tiga subsistem:

$$\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}_{c}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{j} & \mathbf{0} \\ -\mathbf{B}_{c}\mathbf{C}_{j} & \mathbf{A}_{c} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_{c}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{j} \\ \mathbf{0} \end{bmatrix} \mathbf{u}(t) + \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_{c} \end{bmatrix} \mathbf{r}(t)$$

$$j = 1, 2, 3$$
(3.12)

sehingga sinyal kontrol yang dihasilkan adalah hasil augmentasi pencarian gain kompensator dan gain state feedback.

$$\mathbf{u}(t) = \sum_{j=1}^{3} h_j(\mathbf{x}) \{ \begin{bmatrix} \mathbf{K}_{n,j} & \mathbf{K}_{c,j} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{x}_c(t) \end{bmatrix} \}$$
(3.13)

dengan K_n adalah *gain state feedback* dan K_c adalah *gain* kompensator. Bentuk sinyal kontrol persamaan (3.13) disederhanakan menjadi:

$$\mathbf{u}(t) = \sum_{j=1}^{3} h_j(x) \mathbf{K}_j \mathbf{x}(t)$$
(3.14)

dengan K adalah *gain* kontroler yang merupakan augmentasi *gain state feedback* dan *gain* kompensator. Pada tugas akhir ini, sinyal referensi yang digunakan adalah sinyal sinus dengan frekuensi 0,1 Hz dan amplitudo 0,2 m. Dalam domain s, sinyal referensi tersebut dapat dirumuskan:

$$R(s) = \frac{0.0628}{s^2 + 0.3948} \tag{3.1}$$

5)

Penurunan persamaan (3.14) dapat dilihat pada lampiran A.3. Persamaan *state* kompensator dapat dibangun berdasarkan sinyal referensi tersebut dengan elemen matriks kompensator:

$$\mathbf{A}_{c} = \begin{bmatrix} 0 & 1 \\ -0.3948 & 0 \end{bmatrix}; \mathbf{B}_{c} = \begin{bmatrix} 0 \\ 0.0628 \end{bmatrix}$$
(3.16)

Dari bentuk sistem augmentasi persamaan (3.12) dapat dibentuk sistem lup tertutup dari ketiga subsistem pendulum kereta sebagai berikut:

$$\begin{bmatrix} \mathbf{A}_{j} & \mathbf{0} \\ -\mathbf{B}_{e}\mathbf{C}_{j} & \mathbf{A}_{j} \end{bmatrix} + \begin{bmatrix} \mathbf{B}_{j} \\ \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{K}_{n,j} & \mathbf{K}_{c,j} \end{bmatrix}$$

$$j = 1, 2, 3$$
(3.17)

4.3.2. Perancangan Fuzzy T-S Pole Placement

Sistem lup tertutup menjamin kestabilan sistem dengan meletakkan *pole-pole* pada sebelah kiri sumbu imajiner. Metode meletakkan *pole-pole* sistem ini disebut *pole placement*. Terdapat beragam cara dalam *pole placement*, pada tugas akhir ini *pole placement* dirancang menggunakan LMI. Daerah *pole* sistem lup tertutup yang

diinginkan (*region*) dapat dilihat pada Gambar 3.3. Daerah ini membentuk trapesium. *Region* dari *pole* merupakan irisan dari *conic sector* (kerucut) dan *half-plane* (setengah bidang), konsep keduanya telah dijelaskan pada subbab (2.15). Pertidaksamaan untuk mencari *state feedback gain* pada LMI *pole-placement*:

 $P^{-1} > 0$

 $(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} + \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T} + 2\gamma\mathbf{P}^{-1} < 0$ $\begin{bmatrix} \sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} + \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}] \\ \cos\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} - \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}]^{T} \end{bmatrix}^{T} \quad \frac{\cos\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} - \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}]}{\sin\theta[(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{P}^{-1} + \mathbf{P}^{-1}(\mathbf{A} + \mathbf{B}\mathbf{K})^{T}]} \\ < 0$

dengan persamaan sistem lup tertutup,

 $\dot{\mathbf{x}}(t) = (\mathbf{A} + \mathbf{B}\mathbf{K})\mathbf{x}(t)$



(3.18)

(3.19)

Gambar 3.3 Region dari pole-pole sistem lup tertutup (D)

Dengan memanipulasi secara matematis persamaan (3.18), didapatkan: $P^{-1} > 0$

 $\mathbf{A}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T} + 2\gamma\mathbf{P}^{-1} < 0$

 $\begin{bmatrix} \sin\theta [\mathbf{A}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{B}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}] \\ \cos\theta [\mathbf{A}\mathbf{P}^{-1} - \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} - \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}]^{T} \\ \sin\theta [\mathbf{A}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{A}^{T} + \mathbf{B}\mathbf{K}\mathbf{P}^{-1} + \mathbf{P}^{-1}\mathbf{K}^{T}\mathbf{B}^{T}] \end{bmatrix} < 0$ (3.20)

persamaan (3.20) masih mengandung dua variabel yang berdekatan yaitu **K** dan P^{-1} . Dengan subtitusi variabel berikut:

$$\mathbf{Y} = \mathbf{K}\mathbf{P}^{-1}$$
$$\mathbf{Q} = \mathbf{P}^{-1}$$

didapatkan LMI yang menjamin sistem lup tertutup pendulum kereta stabil berada pada *region* tertentu. LMI ini ditunjukkan oleh persamaan (3.21), dengan tiga subsistem yang masing-masing mempunyai *gain* kontroler (**K**).

$$\mathbf{Q}_i > 0$$

$$\mathbf{A}_{j}\mathbf{Q}_{j} + \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} + \mathbf{Y}_{j}^{T}\mathbf{B}_{j} + 2\gamma\mathbf{P}_{j} < 0$$

$$\left[\sin\theta \left[\mathbf{A}_{j}\mathbf{Q}_{j} + \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} + \mathbf{Y}_{j}^{T}\mathbf{B}_{j} \right] \cos\theta \left[\mathbf{A}_{j}\mathbf{Q}_{j} - \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} - \mathbf{Y}_{j}^{T}\mathbf{B}_{j} \right] \\ \sin\theta \left[(\mathbf{A}_{j}\mathbf{Q}_{j} + \mathbf{Q}_{j}\mathbf{A}_{j}^{T} + \mathbf{B}_{j}\mathbf{Y}_{j} + \mathbf{Y}_{j}^{T}\mathbf{B}_{j})^{T} \right]] < 0$$

(3.21)

(3.22)

j = 1, 2, 3

dengan θ adalah sudut antara sumbu real dan sumbu daerah kerucut yang diingkan, γ adalah sumbu batas half plane yang diinginkan berada di sebelah kiri sumbu tersebut.

4.3.3. Batasan Input-Output

Selain memenuhi kestabilan, sistem lup tertutup juga dirancang agar masukan dan keluarannya terbatas. Batasan tersebut dinyatakan dengan batasan sinyal kontrol $\mathbf{u}(t)$ dan keluaran yang dibatasi $\mathbf{z}(t)$ sesuai dengan pertidaksamaan (2.36) dan (2.37). Pertidaksamaan tersebut dapat dinyatakan dengan,

$$\|\mathbf{K}_{j}\mathbf{x}(t)\| \leq \|\mathbf{u}(t)\| \leq u_{\max}$$
$$\|\mathbf{C}_{z,j}\mathbf{x}(t)\| \leq \|\mathbf{z}(t)\| \leq z_{\max}$$
$$j = 1,2,3$$

Untuk mempermudah penerapan batasan pertidaksamaan (3.22) maka ditinjau terlebih dahulu tinjau persamaan Lyapunov sebagai berikut:

$$V(\mathbf{x}(t)) = \mathbf{x}(t)^T \mathbf{P} \mathbf{x}(t)$$
(3.23)

Sinyal kontrol dan keluaran yang dibatasi akan memenuhi batasan yang dirumuskan

$$V(\mathbf{x}(T)) \le \beta$$
Subtitusi persamaan (3.23) ke (3.24)
$$\mathbf{x}(t)^{T} \mathbf{P} \mathbf{x}(t) \le \beta$$

$$\frac{1}{\beta} \mathbf{x}(t)^{T} \mathbf{P} \mathbf{x}(t) \le 1$$
(3.25)

Penurunan LMI untuk batasan pada sinyal kontrol $\mathbf{u}(t)$ dapat diperoleh dari persamaan (3.22) sebagai berikut:

$$\begin{aligned} \left\| \mathbf{K}_{j} \mathbf{x}(t) \right\| &\leq u_{\max} \\ \mathbf{x}(t)^{T} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \mathbf{x}(t) &\leq u_{\max}^{2} \\ \frac{1}{u_{\max}^{2}} \mathbf{x}(t)^{T} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \mathbf{x}(t) &\leq 1 \end{aligned}$$
(3.26)

Subtitusi persamaan (3.26) ke persamaan (3.25) akan diperoleh

$$\frac{1}{u_{\max}^{2}} \mathbf{x}(t)^{T} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \mathbf{x}(t) - \frac{1}{\beta} \mathbf{x}(t)^{T} \mathbf{P}_{j} \mathbf{x}(t) \le 0$$

$$\mathbf{x}(t)^{T} \left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{u_{\max}^{2}} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \right] \mathbf{x}(t) \ge 0$$

$$\left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{u_{\max}^{2}} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \right] \ge 0$$
(3.27)

Sedangkan penurunan LMI untuk batasan keluaran yang dibatasi z(t) dapat dilakukan dari (3.22) sebagai berikut:

$$\begin{aligned} \left\| \mathbf{C}_{z,j} \mathbf{x}(t) \right\| &\leq z_{\max} \\ \mathbf{x}(t)^T \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \mathbf{x}(t) &\leq z_{\max}^2 \\ \frac{1}{z_{\max}^2} \mathbf{x}(t)^T \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \mathbf{x}(t) &\leq 1 \end{aligned}$$
(3.28)

Subtitusi persamaan (3.25 ke persamaan (3.28) diperoleh

$$\frac{1}{z_{\max}^{2}} \mathbf{x}(t)^{T} \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \mathbf{x}(t) - \frac{1}{\beta} \mathbf{x}(t)^{T} \mathbf{P}_{j} \mathbf{x}(t) \leq 0$$
$$\mathbf{x}(t)^{T} \left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{z_{\max}^{2}} \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \right] \mathbf{x}(t) \geq 0$$
$$\left[\frac{\mathbf{P}_{j}}{\beta} - \frac{1}{z_{\max}^{2}} \mathbf{C}_{z,j}^{T} \mathbf{C}_{z,j} \right] \geq 0$$
(3.29)

Penerapan Schur Complement dan pre-multiplying serta postmultiplying LMI (3.27) dan (3.29) dengan matriks **P**⁻¹ akan menghasilkan:

$$\begin{bmatrix} -\mathbf{Q}_{j} & -\mathbf{Y}_{j}^{T} \\ * & -\frac{u_{\max}^{2}}{\beta} \end{bmatrix} < 0$$
$$\begin{bmatrix} -\mathbf{Q}_{j} & -\mathbf{Q}_{j}\mathbf{C}_{z,j}^{T} \\ * & -\frac{z_{\max}^{2}}{\beta} \end{bmatrix} < 0$$

(3.30)

(3.31)

dengan

$$\mathbf{Q}_j = \mathbf{P}_j^{-1}$$

$$\mathbf{Y}_{j} = \mathbf{K}_{j} \mathbf{P}_{j}^{\top}$$

Penurunan LMI (3.30) dan (3.31) dapat dilihat pada Lampiran A.4

4.4. Perhitungan Gain Kontroler

Subbab (3.3) menjelaskan bagaimana LMI untuk kontroler diperoleh yaitu LMI pole placement dan batasan *input-output*. Kedua LMI ini akan digunakan untuk menentukan gain kontroler. Spesifikasi desain yang dirancang dalam tugas akhir ini adalah:

- 1). Sistem stabil asimtotik
- 2). Pole-pole dari sistem lup tertutup berada pada region **D** seperti tampak pada Gambar 3.3.
- Sinyal kontrol maksimal kurang dari atau sama dengan u_{max} seperti pada pertidaksamaan (3.22)

- 4). Keluaran yang dibatasi, yaitu $\mathbf{z}(t)$ kurang dari atau sama dengan z_{max} seperti pada pertidaksamaan (3.22)
- 5). Sinyal kontrol dan keluaran yang dibatasi memenuhi batasan seperti pada pertidaksamaan (3.24)

(3.32)

Untuk memenuhi spesifikasi desain tersebut, harus terdapat matriks simetris Q yang memenuhi LMI:

 $A_{j} > 0$ $B_{j} < 0$ $X_{j} < 0$ $E_{j} > 0$ $H_{j} > 0$

dengan masing-masing constraint pada LMI (3.32),

$$A_{j} = Q_{j}$$

$$B_{j} = A_{j}Q_{j} + Q_{j}A_{j}^{T} - B_{j}Y_{j} - Y_{j}^{T}B_{j} + 2\gamma P$$

$$X_{j} = \begin{bmatrix} \sin \theta [A_{j}Q_{j} + Q_{j}A_{j}^{T} - B_{j}Y_{j} - Y_{j}^{T}B_{j}] & \cos \theta [A_{j}Q_{j} - Q_{j}A_{j}^{T} - B_{j}Y_{j} + Y_{j}^{T}B_{j}] \\ \sin \theta [(A_{j}Q_{j} + Q_{j}A_{j}^{T} - B_{j}Y_{j} - Y_{j}^{T}B_{j})^{T}] \end{bmatrix}$$

$$E_{j} = \begin{bmatrix} Q_{j} & Y_{j}^{T} \\ * & \frac{u_{max}^{2}}{\beta} \end{bmatrix}$$

$$H_{j} = \begin{bmatrix} -Q_{j} & -Q_{j}C_{z,j}^{T} \\ * & -\frac{z_{max}^{2}}{\beta} \end{bmatrix}$$

$$Q_{j} = P_{j}^{-1}$$

$$Y_{j} = K_{j}P_{j}^{-1}$$
dengan gain kontroler yang dicari adalah:

$$K_{j} = Y_{j}P_{j}$$

$$j = 1,2,3$$
(3.3)

Dalam Tugas Akhir ini digunakan parameter-parameter sebagai berikut:

- 1). A_1 dan B_1 adalah matriks hasil linearisasi sistem pendulum kereta di sekitar $x_2(t) = 0$ radian.
- 2). A₂ dan B₂ adalah matriks hasil linearisasi sistem pendulum kereta di sekitar $x_2(t) = \pm 0,2616$ radian.
- 3). A₃ dan B₃ adalah matriks hasil linearisasi sistem pendulum kereta di sekitar $x_2(t) = \pm 0,5232$ radian.

Parameter region pole-pole sistem pendulum kereta

- 1). $\theta = \pi/12$, yaitu sudut batas pole-pole diletakkan pada daerah conic sector (sudut antara sumbu batas dan sumbu real)
- 2). $\gamma = 2$, yaitu sumbu batas half plane pada kiri sumbu imajiner.

Parameter batasan input-output sistem pendulum kereta

- 1). $\mathbf{C}_{z,1} = \mathbf{C}_{z,2} = \mathbf{C}_{z,3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix}$, menyatakan keluaran yang dibatasi hanya $x_1(t)$, yaitu posisi kereta.
- 2). $\beta = 5$, fungsi energi Lyapunov yang dibatasi.
- 3). $u_{\text{max}} = 17,5$ N, yang menyatakan bahwa sinyal kontrol maksimal adalah $\pm 17,5$ N
- 4). $z_{\text{max}} = 0.4$ yang menyatakan bahwa posisi kereta maksimal adalah ± 0.4

Untuk menyelesaikan LMI (3.32), digunakan MATLAB. Parameter dari penyelesaian LMI ini yaitu gain kontrol \mathbf{K}_1 , \mathbf{K}_2 , \mathbf{K}_3 dan nilai pole-pole yang didapatkan adalah:

7777					(3.34)
$K_3 = 10^4 [0,1975]$	-0,2289	0,0772	-0,0643	- 2,1547	-4,6524]
$K_2 = 10^4 [0,1504]$	- 0,1664	0,0580	-0,0442	-1,5903	-3,5215]
$K_1 = 10^4 [0, 1415]$	-0,1537	0,0542	-0,0401	-1,4842	-3,3100]

Gain kontroler ketiga subsistem pada persamaan (3.34) adalah *gain* untuk masing-masing sistem pada ketiga titik kerja. Selanjutnya bentuk *Fuzzy* T-S akan dibentuk kontroler. Keempat elemen matriks *gain* pertama adalah *gain state feedback* dan kedua elemen selanjutnya adalah *gain* kompensator. Sedangkan *pole-pole* sistem lup tertutupnya,

Subsistem 1	Subsistem 2	Subsistem 3
$x_2(t) = \pm 0$ radian	$x_2(t) = \pm 0,26$ radian	$x_2(t) = \pm 0,52$ radian
-26,5603+0.0000i	-27,0589+0.0000i	-28,9491+0.0000i
-8,1007+0.0000i	-8,1184+0.0000i	-8,7111+0.0000i
- 2,1765 + 0.3681i	- 2,2029 + 0.3568i	- 4,8649+0.0000i
- 2,1765 - 0.3681i	- 2,2029 - 0.3568i	- 2,2217 + 0.3682i
- 4,5090 + 0.0000i	- 4,5227 + 0.0000i	- 2,2217 - 0.3682i
-3,8251+0.0000i	-3,7589+0.0000i	-3,5560 + 0.0000i

(3.35)

Pole-pole sistem lup tertutup (3.35) terletak di sebelah kiri dari desain yang diinginkan yaitu $\gamma = 2 \text{ dan } \theta = \pi/12 \text{ radian.}$

4.5. Desain Observer Kesalahan Sensor dan Aktuator berbasis Fuzzy T-S PPIO (Proportional Proportional Integral Observer)

Pada subbab ini dijelaskan desain observer yang berfungsi untuk mengestimasi kesalahan sensor dan aktuator berbasis PPIO. Perbedaan observer PPIO dengan observer pada umumnya adalah observer gain pada observer PPIO terdapat dua, yaitu gain proportional dan gain proportional integral. Penambahan gain proportional integral disini dimaksudnya untuk mempercepat proses estimasi dari observer. Struktur observer secara umum dapat dilihat pada Gambar 3.4.



Gambar 3.4 Desain Observer PPIO Kesalahan Aktuator dan Sensor

4.5.1. Desain *Observer* Kesalahan Sensor berbasis *Fuzzy* T-S PPIO *Observer* kesalahan sensor digunakan untuk mengestimasi dan mengompensasi kesalahan sensor. *Observer* yang didesain adalah tipe model-based (menggunakan model *plant*). Struktur observer dapat dilihat pada Gambar 3.5. Terdapat dua gain pada desain, yaitu gain proportional integral ($\mathbf{\bar{F}}$) dan gain proportional ($\mathbf{\bar{P}}$). Masukan dari observer adalah sinyal kontrol ($\mathbf{u}(t)$), estimasi kesalahan aktuator (\hat{f}_a) dan keluaran yang belum terkompensasi ($\mathbf{y}(t)$).





Untuk menghindari perkalian langsung kesalahan sensor dan observer gain, digunakan LPF (Low Pass Filter). Desain LPF dapat dilihat pada Gambar 3.6.



Observer didesain berbasis *Fuzzy* T-S. Untuk mempermudah perhitungan dan penulisan, sistem *Fuzzy* T-S pada persamaan (2.9) diubah menggunakan variabel pada persamaan (3.36)

$$\mathbf{A}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{A}_{j},$$

$$\mathbf{B}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{B}_{j},$$

$$\mathbf{C}_{c}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{C}_{j},$$
(3.36)

dengan C_c adalah keluaran untuk keperluan desain *observer*, dengan adanya kesalahan sensor, sistem *Fuzzy* T-S dapat ditulis:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x} + \mathbf{B}(p)(\mathbf{u}(t) + f_a(t))$$

$$\mathbf{y}(t) = \mathbf{C}_c \mathbf{x}(t) + \mathbf{D}_f f_s(t)$$

(3.37)

dengan $\dot{\mathbf{x}}(t) \in \mathbb{R}^n$ adalah persamaan *state*, $\mathbf{u}(t) \in \mathbb{R}^m$ adalah sinyal kontrol, $\mathbf{y}(t) \in \mathbb{R}^l$ adalah keluaran yang terukur, $f_s(t) \in \mathbb{R}^g$ adalah vektor kesalahan sensor, $f_a(t) \in \mathbb{R}^m$ adalah vektor kesalahan aktuator dan D_f adalah vektor yang mempresentasikan kesalahan sensor. $\mathbf{C}_c(p)$ adalah keluaran untuk keperluan desain observer, yang dituliskan:

$$\mathbf{C}_{c,1} = \mathbf{C}_{c,2} = \mathbf{C}_{c,3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$
(3.38)

Keluaran untuk mendesain *observer* berbeda dengan keluaran untuk mendesain kontroler, hal ini karena desain *observer* harus memenuhi kondisi observabilitas yang didefinisikan:

$$rank\begin{bmatrix} s\mathbf{I} - \mathbf{A} & 0\\ \mathbf{A}_s \mathbf{C} & s\mathbf{I} + \mathbf{A}_s\\ \mathbf{0} & \mathbf{I} \end{bmatrix} = n + l \tag{3.39}$$

Sistem pada persamaan (3.37) diaugmentasi dengan sistem LPF persamaan (3.40) sehingga menghasilkan sistem pada persamaan (3.41).

$$\dot{\mathbf{x}}_{s}(t) = -\mathbf{A}_{s}\mathbf{x}_{s}(t) + \mathbf{A}_{s}\mathbf{C}_{c}\mathbf{x}(t) + \mathbf{A}_{s}\mathbf{D}_{f}f_{s}(t)$$

dengan matriks A_s adalah matriks full rank. Sehingga sistem augmentasinya:

$$\dot{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}(p)\overline{\mathbf{x}}(t) + \overline{\mathbf{B}}(p)(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f f_s(t)$$

$$\mathbf{\overline{A}}(p) = \begin{bmatrix} \mathbf{A}(p) & 0\\ \mathbf{A}_s \mathbf{C}_c & -\mathbf{A}_s \end{bmatrix}, \mathbf{\overline{B}}(p) = \begin{bmatrix} \mathbf{B}(p)\\ 0 \end{bmatrix}, \mathbf{\overline{D}}_f = \begin{bmatrix} 0\\ \mathbf{A}_s \mathbf{D}_f \end{bmatrix}$$
$$\mathbf{\overline{x}}(t) = \begin{bmatrix} \mathbf{x}(t)\\ \mathbf{x}_s(t) \end{bmatrix}, \mathbf{\overline{C}}_c = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

Dari sistem augmentasi ini dibangun *observer* untuk kesalahan sensor. *Observer* didesain berbasis *Fuzzy* T-S dengan premis dari sudut pendulum $(x_2(t))$. Terdapat tiga aturan untuk membangun *Fuzzy* T-S *Observer* ditunjukkan pada persamaan (3.42). Tiga aturan tersebut adalah $x_2(t) = 0$ rad, $\pm 0,2616$ rad, dan $\pm 0,5232$ rad. Aturan *Fuzzy* ini disederhanakan menggunakan persamaan (3.36) menjadi persamaan (3.43).

(3.41)

(t)),

Aturan *observer* ke-1 If $x_2(t)$ is M₁ (sekitar 0 rad)

-(-)

Then $\hat{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}_1 \hat{\overline{\mathbf{x}}}(t) + \overline{\mathbf{B}}_1(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f \hat{f}_s(t) + \overline{\mathbf{L}}_1 \overline{\mathbf{C}}_c e_x(t)),$

$$\hat{f}_{s}(t) = \overline{\mathbf{F}}_{1}\overline{\mathbf{C}}_{c}(\dot{e}_{x}(t) + e_{x}(t)),$$

Aturan *observer* ke-2 If $x_2(t)$ is M₂ (sekitar 0.2616 rad)

Then
$$\dot{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}_2 \hat{\overline{\mathbf{x}}}(t) + \overline{\mathbf{B}}_2 (\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_c \hat{f}_a(t) + \overline{\mathbf{L}}_2 \overline{\mathbf{C}}_a e_a$$

$$\hat{f}_s(t) = \overline{\mathbf{F}}_2 \overline{\mathbf{C}}_c (\dot{e}_x(t) + e_x(t)),$$

Aturan *observer* ke-3 If $x_2(t)$ is M₃ (sekitar 0.5232 rad)

Then $\hat{\overline{\mathbf{x}}}(t) = \overline{\mathbf{A}}_3 \hat{\overline{\mathbf{x}}}(t) + \overline{\mathbf{B}}_3(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f \hat{f}_s(t) + \overline{\mathbf{L}}_3 \overline{\mathbf{C}}_c e_x(t)),$

$$\hat{f}_s(t) = \overline{\mathbf{F}}_3 \overline{\mathbf{C}}_c(\dot{e}_x(t) + e_x(t)),$$

penyederhaan aturan Fuzzy T-S:

$$\dot{\bar{\mathbf{x}}}(t) = \overline{\mathbf{A}}(p)\hat{\bar{\mathbf{x}}}(t) + \overline{\mathbf{B}}(p)(\mathbf{u}(t) + \hat{f}_a(t)) + \overline{\mathbf{D}}_f \hat{f}_s(t) + \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_c e_x(t)),$$
$$\dot{\bar{f}}_s(t) = \overline{\mathbf{F}}(p)\overline{\mathbf{C}}_c(\dot{e}_x(t) + e_x(t)),$$
(3.43)

dengan $\hat{\mathbf{x}}(t) \in \mathbf{R}^{n+1}$ adalah estimasi state, \hat{f}_s adalah estimasi kesalahan aktuator yang dihasilkan observer lain, $\overline{\mathbf{L}}(p) \in \mathbf{R}^{(n+1)xl}$ dan $\overline{\mathbf{F}}(p) \in \mathbf{R}^{gxl}$ adalah observer gain yang akan didesain berbasis Fuzzy T-S, dan $e_x(t)$ adalah error estimasi. Dengan gain proportional dan observer gain proportional integral ditunjukkan persamaan (3.45) dan (3.46):

$$e_x = \overline{\mathbf{x}}(t) - \hat{\overline{\mathbf{x}}}(t), \tag{3.44}$$

$$\overline{\mathbf{L}}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \overline{\mathbf{L}}_{j},$$
(3.45)

$$\overline{\mathbf{F}}(p) = \sum_{j=1}^{n} \mathbf{h}_{j} * \overline{\mathbf{F}}_{j},$$

4.5.2. Desain LMI (*Linear Matrix Equalities*) untuk Observer Kesalahan Sensor

Desain observer memastikan performa H_{∞} sehingga error estimasi dibuat sekecil mungkin. LMI digunakan untuk menentukan observer gain. Desain LMI menggunakan persamaan matriks augmentasi estimasi state (3.43) dan persamaan state (3.41). Untuk mendesain LMI, langkah pertama adalah menentukan error estimasi state. Dari persamaan (3.43) dan (3.41) didapatkan error estimasi state:

$$\dot{e}_{x}(t) = \dot{\overline{\mathbf{x}}} - \dot{\overline{\mathbf{x}}}$$
$$\dot{e}_{x}(t) = (\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c})e_{x}(t) + \overline{\mathbf{D}}_{f}e_{fs}(t) + \overline{\mathbf{B}}(p)e_{fs}(t)$$

(3.47)

(3.46)

(3.42)

dengan,

$$e_{fs}(t) = f_s(t) - f_s(t)$$

 $e_{fa}(t) = f_a(t) - \hat{f}_a(t)$

Persamaan (3.46) disebut error estimasi *state*, yaitu selisih nilai *state* yang sebenarnya (matriks augmentasi) dikurangi nilai *state* hasil estimasi *observer*. Selanjutnya dari error estimasi *state* dicari error estimasi kesalahan sensor yang dirumuskan:

$$e_{fs}(t) = f_s(t) - \hat{f}_s(t)$$
 (3.48)

Error estimasi kesalahan sensor adalah selisih kesalahan sensor yang sebenarnya dikurangi dengan estimasi kesalahan sensor. Dari hasil subtitusi persamaan (3.43) dan (3.46) ke persamaan (3.47) didapatkan:

$$\dot{\boldsymbol{e}}_{fs}(t) = \dot{f}_s(t) - \overline{\mathbf{F}}(p)\overline{\mathbf{C}}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}} + \mathbf{I})\boldsymbol{e}_x(t)$$

$$-\overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{D}}_f \boldsymbol{e}_{fs}(t) - \overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{B}}(p)\boldsymbol{e}_{fa}(t)$$
(3.49)

Persamaan error estimasi *state* (3.46) dan error estimasi kesalahan sensor (3.48) digunakan untuk menentukan matrik kandidat fungsi Lyapunov. Kandidat fungsi Lyapunov digunakan untuk menjamin sistem *observer* (3.43) stabil dan memenuhi performa H_{∞} seperti berikut:

(3.50)

(3.51)

(3.52)

$$\frac{\left\|\widetilde{e}_{p}(t)\right\|_{2}}{\left\|\widetilde{\mathbf{z}}(t)\right\|_{2}} < \gamma$$

dengan,

$$\widetilde{e}_p(t) = \overline{C}_p \widetilde{e}_{as}(t)$$

$$\widetilde{\mathbf{e}}_{as}(t) = \begin{bmatrix} e_x(t) \\ e_{fs}(t) \end{bmatrix}$$
$$\widetilde{\mathbf{z}}(t) = \begin{bmatrix} e_{fa}(t) \\ \dot{f}_s(t) \end{bmatrix}$$

Persamaan (3.50) adalah L_2 -gain untuk observer kesalahan sensor. L_2 -gain dirancang untuk kurang dari atenuasi/tingkat pelemahan tertentu. Masukan performa ditunjukkan persamaan (3.52), yang terdiri atas error estimasi *state* dan error estimasi kesalahan sensor. Keluaran

performa ditunjukkan persamaan (3.53) yang terdiri atas error estimasi kesalahan aktuator dan derivatif turunan kesalahan sensor. Tingkat pelemahan/atenuasi dicari agar estimasi memiliki akurasi yang tinggi dan cepat. Dari persamaan (3.46) dan (3.48) didapatkan matriks augmentasi error estimasi *state* dan error estimasi kesalahan sensor:

$$\begin{bmatrix} \dot{e}_{x}(t) \\ \dot{e}_{fs}(t) \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} & \overline{\mathbf{D}}_{f} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} + \mathbf{I}) & -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{D}}_{f} \end{bmatrix} \begin{bmatrix} e_{x}(t) \\ e_{fs}(t) \end{bmatrix} \\ + \begin{bmatrix} \overline{\mathbf{B}}(p) & \mathbf{0} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}\overline{\mathbf{B}}(p) & \mathbf{I} \end{bmatrix} \begin{bmatrix} e_{fa} \\ \dot{f}_{s} \end{bmatrix}$$

Dari matriks augmentasi kandidat fungsi Lyapunov (3.53) definisikan variabel berikut untuk penyederhanaan dalam desain LMI:

$$\widetilde{e}_{as}(t) = \begin{bmatrix} e_x(t) \\ e_{fs}(t) \end{bmatrix}, \widetilde{z} \begin{bmatrix} e_{fa}(t) \\ f_s(t) \end{bmatrix}$$

$$\widetilde{A}_s(p, p) = \begin{bmatrix} \overline{A}(p) - \overline{L}(p)\overline{C}_c & \overline{D}_f \\ -\overline{F}(p)\overline{C}(\overline{A}(p) - \overline{L}(p)\overline{C}_c + \mathbf{I}) & -\overline{F}(p)\overline{C}\overline{D}_f \end{bmatrix}$$

$$\widetilde{N}(p, p) = \begin{bmatrix} \overline{B}(p) & 0 \\ -\overline{F}(p)\overline{C}\overline{B}(p) & \mathbf{I} \end{bmatrix}$$

(3.54)

(3.53)

Dari variabel yang didefinisikan pada persamaan (3.54), disubtitusikan ke persamaan (3.53) menghasilkan persamaan (3.55)

$$\widetilde{\widetilde{e}}_{as}(t) = \widetilde{\mathbf{A}}_{s}(p,p)\widetilde{e}_{as}(t) + \widetilde{\mathbf{N}}(p,p)\widetilde{\mathbf{z}}(t)$$
(3.55)

Persamaan (3.55) adalah kandidat fungsi Lyapunov. Kandidat fungsi Lyapunov ini digunakan untuk menentukan LMI Observer, menjamin kestabilan dan keluaran L_2 -gain didesain kurang dari γ tertentu.

Definisikan fungsi Lyapunov,

$$v(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)\overline{\mathbf{P}}\tilde{e}_{as}(t)$$

$$\overline{\mathbf{P}} = \sum_{j=1}^{3} h_{j}(x)\mathbf{P}_{j}, j = 1,2,3$$
(3.56)

Derivatif fungsi Lyapunov,

$$\dot{v}(\widetilde{e}_{as}(t)) = \widetilde{e}_{as}^{T}(t)(\widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p))\widetilde{e}_{as}(t)$$

$$+ \widetilde{e}_{as}^{T}\overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p)\widetilde{\mathbf{z}} + \widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{N}}^{T}(p,p)\overline{\mathbf{P}}\widetilde{e}_{as}(t)$$
(3.57)

Persamaan L_2 -gain pada (3.49) dapat dilakukan manipulasi matematis sehingga menghasilkan persamaan (3.58)

$$\frac{1}{\gamma} \int_0^\infty \tilde{\boldsymbol{e}}_{as}^T \overline{\mathbf{C}}_p^T \overline{\mathbf{C}}_p \tilde{\boldsymbol{e}}_{as} dt - \gamma \int_0^\infty \tilde{\mathbf{z}}^T \tilde{\mathbf{z}} dt < 0$$
(3.58)

Persamaan (3.59) dalam bentuk fungsi Lyapunov,

$$\dot{v}(\tilde{e}_{as}) + \frac{1}{\gamma} \tilde{e}_{as}^{T} \overline{\mathbf{C}}_{p}^{T} \overline{\mathbf{C}}_{p} \tilde{e}_{as} - \gamma \tilde{\mathbf{z}}^{T} \tilde{\mathbf{z}} < 0$$
(3.59)

Subtitusi persamaan fungsi Lyapunov (3.57) ke persamaan (3.59)

$$\widetilde{e}_{as}^{T}(t)(\widetilde{A}_{s}^{T}(p,p)\overline{\mathbf{P}}+\overline{\mathbf{P}}\widetilde{A}_{s}(p,p))\widetilde{e}_{as}(t)+\widetilde{e}_{as}^{T}\overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p)\widetilde{\mathbf{z}}$$

$$+\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{N}}^{T}(p,p)\overline{\mathbf{P}}\widetilde{e}_{as}(t)+\frac{1}{\gamma}\widetilde{e}_{as}^{T}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p}\widetilde{e}_{as}-\gamma\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{z}}<0$$
(3.60)

dengan metode Schur Complement dan metode *pre-multiplying* dan *post-multipling* didapatkan pertidaksamaan (3.61). Penurunan pertidaksamaan ini dapat dilihat pada Lampiran A.5

$$\begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}}+\overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) & \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) & \overline{\mathbf{C}}_{p1}^{T} & \mathbf{0} \\ & & & -\gamma \mathbf{I} & \mathbf{0} & \overline{\mathbf{C}}_{p2}^{T} \\ & & & * & -\gamma \mathbf{I} & \mathbf{0} \\ & & & & * & * & -\gamma \mathbf{I} \end{bmatrix} <$$

(3.61)

dengan subtitusi variabel berikut ke pertidaksamaan (3.61),

$$\widetilde{\mathbf{A}}_{s}(p,p) = \begin{bmatrix} \overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} & \overline{\mathbf{D}}_{f} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} + \mathbf{I}) & -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{D}}_{f} \end{bmatrix},$$

$$\widetilde{\mathbf{N}}(p,p) = \begin{bmatrix} \overline{\mathbf{B}}(p) & 0 \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{B}}(p) & \mathbf{I} \end{bmatrix},$$

$$\overline{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{1} & 0 \\ 0 & \mathbf{I} \end{bmatrix},$$

$$\overline{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{1} & 0 \\ 0 & \mathbf{I} \end{bmatrix},$$

$$\overline{\mathbf{H}}(p) = \mathbf{P}_{1}\overline{\mathbf{L}}(p),$$

$$\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c} = \overline{\mathbf{D}}_{f}\mathbf{P}_{1}$$
(3.62)
idapatkan LMI pada (3.63),
$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & 0 \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma\overline{\mathbf{A}} & 0 & 0 \\ * & * & * & -\gamma\overline{\mathbf{A}} & 0 & 0 \\ * & * & * & -\gamma\overline{\mathbf{A}} & 0 & 0 \\ * & * & * & * & -\gamma\overline{\mathbf{A}} & 0 \\ * & * & * & -\gamma\overline{\mathbf{A}} & 0 \\ * & * & * & -\gamma\overline{\mathbf{A}} &$$

$$w_{13} = \mathbf{P}_1 \overline{\mathbf{B}}(p),$$

$$w_{22} = -2\overline{\mathbf{D}}_f^T \mathbf{P}_1 \overline{\mathbf{D}}_f,$$

$$w_{23} = -2\overline{\mathbf{D}}_f^T \mathbf{P}_1 \overline{\mathbf{B}}(p),$$

untuk meningkatkan performa *observer* digunakan LMI yang menjamin *Fast Fault Estimation* (3.64).

$$\begin{bmatrix} \mu \mathbf{I} & \overline{\mathbf{D}}_{f} \mathbf{P}_{1} - \overline{\mathbf{F}}(p) \overline{\mathbf{C}}_{c} \\ * & \mu \mathbf{I} \end{bmatrix} > 0$$
(3.64)

(3.63)

Sedangkan untuk meletakkan *pole-pole observer* pada daerah tertentu yang memiliki kriteria performa yang lebih baik, seperti *overshoot*, digunakan LMI Region (3.65).

(3.65)

$$\begin{split} \sum_{i}^{T} + \sum_{i}^{T} + 2\rho \overline{\mathbf{P}} < 0 \\ \left[\sin(\theta) [\sum_{i}^{T} + \sum_{i}^{T}] \cos(\theta) [\sum_{i}^{T} + \sum_{i}^{T}] \\ * \sin(\theta) [\sum_{i}^{T} + \sum_{i}^{T}] \right] < 0 \end{split}$$

dengan,

$$\sum_{i} = \overline{\mathbf{P}} \mathbf{A}_{s}(p, p) = \begin{bmatrix} \mathbf{P}_{1} \overline{\mathbf{A}}(p) - \overline{\mathbf{H}}(p) \overline{\mathbf{C}}_{c} & \mathbf{P}_{1} \overline{\mathbf{D}}_{f} \\ - (\overline{\mathbf{A}}^{T}(p) \mathbf{P}_{1} \overline{\mathbf{D}}_{f} - \overline{\mathbf{C}}_{c}^{T} \overline{\mathbf{H}}^{T} \overline{\mathbf{D}}_{f} + \mathbf{P}_{1} \overline{\mathbf{D}}_{f} \end{bmatrix} - \overline{\mathbf{D}}^{T}_{f} \mathbf{P}_{1} \overline{\mathbf{D}}_{f} \end{bmatrix}$$

Sehingga constraint LMI pada (3.63), (3.64) dan (3.65) dapat digabung:

$$\begin{split} & \min(\gamma + \mu) \\ & \begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & 0 \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma \mathbf{I} & 0 & 0 \\ * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & * & -\gamma \mathbf{I} & 0 \\ * & * & * & * & -\gamma \mathbf{I} \end{bmatrix} < \mathbf{0} \\ & \begin{bmatrix} \mu \mathbf{I} & \overline{\mathbf{D}}_{f} \mathbf{P}_{1} - \overline{\mathbf{F}}(p) \overline{\mathbf{C}}_{c} \\ * & \mu \mathbf{I} \end{bmatrix} > \mathbf{0} \\ & \sum_{i} + \sum_{i}^{T} + 2\rho \overline{\mathbf{P}} < \mathbf{0} \\ & \begin{bmatrix} \sin(\theta) [\sum_{i} + \sum_{i}^{T}] & \cos(\theta) [\sum_{i} + \sum_{i}^{T}] \\ * & \sin(\theta) [\sum_{i} + \sum_{i}^{T}] \end{bmatrix} < \mathbf{0} \\ & \text{dengan,} \\ & w_{13} = \mathbf{P}_{1} \overline{\mathbf{B}}(p), \\ & w_{22} = -2 \overline{\mathbf{D}}_{f}^{T} \mathbf{P}_{1} \overline{\mathbf{D}}_{f}, \end{split}$$

$$\begin{split} w_{23} &= -2\overline{\mathbf{D}}_{f}^{T}\mathbf{P}_{\mathbf{l}}\overline{\mathbf{B}}(p), \\ w_{11} &= \mathbf{P}_{\mathbf{l}}\overline{\mathbf{A}}(p) + (\mathbf{P}_{\mathbf{l}}\overline{\mathbf{A}}(p))^{T} - \overline{\mathbf{H}}(p)\overline{\mathbf{C}}_{c} - (\overline{\mathbf{H}}(p)\mathbf{C}_{c})^{T}, \\ w_{12} &= -(\overline{\mathbf{A}}^{T}(p)\mathbf{P}_{\mathbf{l}}\mathbf{D}_{f} - \overline{\mathbf{C}}_{c}^{T}\overline{\mathbf{H}}^{T}(p)\overline{\mathbf{D}}_{f}), \\ \Sigma_{i} &= \overline{\mathbf{P}}\mathbf{A}_{s}(p, p) = \\ \begin{bmatrix} \mathbf{P}_{\mathbf{l}}\overline{\mathbf{A}}(p) - \overline{\mathbf{H}}(p)\overline{\mathbf{C}}_{c} & \mathbf{P}_{\mathbf{l}}\overline{\mathbf{D}}_{f} \\ -(\overline{\mathbf{A}}^{T}(p)\mathbf{P}_{\mathbf{l}}\overline{\mathbf{D}}_{f} - \overline{\mathbf{C}}_{c}^{T}\overline{\mathbf{H}}^{T}\overline{\mathbf{D}}_{f} + \mathbf{P}_{\mathbf{l}}\overline{\mathbf{D}}_{f}) &= \overline{\mathbf{D}}^{T}_{f}\mathbf{P}_{\mathbf{l}}\overline{\mathbf{D}}_{f} \end{bmatrix} \end{split}$$

Observer gain proportional dan *proportional integral* pada persamaan (3.66) dapat dicari dengan persamaan berikut:

$$\overline{\mathbf{L}}(p) = \mathbf{P}_1^{-1} \overline{\mathbf{H}}(p)$$
(3.67)
$$\overline{\mathbf{F}}(p)$$

(3.68)

dengan estimasi kesalahan sensornya,

$$\hat{f}_s(t) = \overline{\mathbf{F}}(p)\overline{\mathbf{C}}_c \int (\dot{e}_x(t) + e_x(t))dt$$

4.5.3. Desain Observer Kesalahan Aktuator Berbasis T-S PPIO

Desain *observer* kesalahan aktuator hampir sama dengan *observer* kesalahan sensor, hanya saja desain *observer* kesalahan aktuator tidak menggunakan augmentasi dengan LPF. Desain untuk *Observer* kesalahan aktuator dapat dilihat pada Gambar 3.7.



Gambar 3.7 Desain observer kesalahan aktuator

terdapat dua gain pada desain, yaitu proportional integral gain (\mathbf{F}_{a}) dan proportional gain (\mathbf{P}_{a}). Masukan dari observer adalah sinyal kontrol ($\mathbf{u}(t)$), dan keluaran yang sudah terkompensasi ($\mathbf{y}(t)$). Tinjau sistem dengan kesalahan aktuator dan sensor seperti persamaan (3.27) berikut:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x}(t) + \mathbf{B}(p)(\mathbf{u}(t) + f_a(t))$$

$$\mathbf{y}(t) = \mathbf{C}_c \mathbf{x}(t) + \mathbf{D}_f f_s(t)$$

 $\mathbf{F}_{a}(p) = \sum_{j=1}^{n} \mathbf{h}_{j} * \mathbf{F}_{a,j}$

Observer kesalahan aktuator direpresentasikan dengan persamaan (3.69) sebagai berikut:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(p)\mathbf{x}(t) + \mathbf{B}(p)(\mathbf{u}(t) + \hat{f}_a) + \mathbf{L}_a(p)(\mathbf{C}\mathbf{x}(t) + \mathbf{D}_f e_{f\hat{s}}(t) - \mathbf{C}\hat{\mathbf{x}}(t)),$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}_f e_{f\hat{s}}(t)$$

(3.69)

dengan $\mathbf{\hat{x}}(t) \in \mathbb{R}^{n+l}$ adalah estimasi *state*, $\mathbf{\hat{f}}_a$ adalah estimasi kesalahan aktuator, $\mathbf{L}_{\mathbf{a}}(\mathbf{p}) \in \mathbb{R}^{nxl}$ dan $\mathbf{L}_{\mathbf{a}}(\mathbf{p}) \in \mathbb{R}^{mxl}$ adalah observer gain yang akan didesain berbasis Fuzzy T-S, dan $e_x(t)$ adalah error estimasi. Dengan gain proportional dan gain proportional integral observer ditunjukkan:

$$e_{x}(t) = \overline{\mathbf{x}}(t) - \hat{\overline{\mathbf{x}}}(t), \qquad (3.70)$$

$$\mathbf{L}_{a}(p) = \sum_{j=1}^{r} \mathbf{h}_{j} * \mathbf{L}_{a,j} \qquad (3.71)$$

4.5.4. Desain LMI (*Linear Matrix Equalities*) untuk Observer Kesalahan Aktuator

Seperti pada *observer* kesalahan sensor, LMI untuk kesalahan aktuator memiliki desain yang hampir sama. Desain *observer* menjamin kestabilan *observer* dan performa H_{∞} . Dengan prosedur yang hampir sama, persamaan error estimasi *state* dan error estimasi kesalahan sensornya:

$$\dot{\mathbf{e}}_{x}(t) = (\mathbf{A}(p) - \mathbf{L}_{a}(p)\mathbf{C}_{c})\mathbf{e}_{x}(t) + \mathbf{B}(p)\mathbf{e}_{fa}(t) - \mathbf{L}_{a}(p)\mathbf{D}_{f}\mathbf{e}_{fs}(t)$$

(3.73)

(3.72)

$$\dot{e}_{fa}(t) = f_a(t) - \mathbf{F}_a(p)\mathbf{C}_c(\mathbf{A}(p) - \mathbf{L}_a(p)\mathbf{C}_c + \mathbf{I})e_x(t) - \mathbf{F}_a(p)\mathbf{C}_c\mathbf{D}_f e_{fa}(t) + \mathbf{F}_a(p)\mathbf{C}_c\mathbf{L}_a(p)\mathbf{D}_f e_{fs}(t)$$

Dari persamaan error estimasi *state* dan error estimasi kesalahan aktuator dibentuk matriks augmentasi untuk kandidat fungsi Lyapunov sebagai berikut:

(3.74)

(3.75)

$$\begin{split} \dot{\tilde{e}}_{a}(t) &= \widetilde{\mathbf{A}}(p, p) \widetilde{e}_{a}(t) + \widetilde{\mathbf{N}}(p, p) \widetilde{\mathbf{z}}(t) \\ \widetilde{\mathbf{A}}(p, p) &= \begin{bmatrix} \mathbf{A}(p) - \mathbf{L}_{a}(p) \mathbf{C}_{c} & \mathbf{B}(p) \\ - \mathbf{F}_{a}(p) \mathbf{C}_{c}(\mathbf{A}(p) - \mathbf{L}_{a}(p) \mathbf{C}_{c} + \mathbf{I}) & - \mathbf{F}_{a}(p) \mathbf{C}_{c} \mathbf{B}(p) \end{bmatrix}, \\ \widetilde{\mathbf{N}}(p, p) &= \begin{bmatrix} -\mathbf{L}_{a}(p) \mathbf{D}_{f} & 0 \\ - \mathbf{F}_{a}(p) \mathbf{C} \mathbf{L}_{a}(p) \mathbf{D}_{f} & I \end{bmatrix}, \end{split}$$

Kandidat fungsi Lyapunov pada persamaan (3.75) digunakan untuk memenuhi performa H_{∞} dengan L_2 -gain sebagai berikut:

$$\frac{\left\|\widetilde{e}_{p}(t)\right\|_{2}}{\left\|\widetilde{z}(t)\right\|_{2}} < \gamma$$
(3.76)

mgan,
$$\widetilde{e}_{p}(t) = \mathbf{C}_{p}\widetilde{e}_{as}(t)$$

$$\widetilde{e}_{a}(t) = \begin{bmatrix} e_{x}(t) \\ e_{fa}(t) \end{bmatrix}$$
(3.77)
$$\widetilde{z}(t) = \begin{bmatrix} e_{fs}(t) \\ \dot{f}_{a}(t) \end{bmatrix}$$
(3.78)

Persamaan (3.76) adalah L_2 -gain untuk observer kesalahan aktuator. L_2 -gain dirancang untuk kurang dari atenuasi/tingkat pelemahan tertentu. Masukan performa ditunjukkan persamaan (3.77), yang terdiri atas error estimasi state dan error estimasi kesalahan aktuator. Keluaran performa ditunjukkan persamaan (3.78) yang terdiri atas error estimasi kesalahan sensor dan derivatif turunan kesalahan aktuator.

definisikan fungsi Lyapunov,

$$v(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)\mathbf{P}_{a}\tilde{e}_{as}(t)$$
$$\mathbf{P}_{a} = \sum_{j=1}^{3} h_{j}(x)\mathbf{P}_{a,j}$$

derivatif fungsi Lyapunov,

$$\dot{v}(\tilde{e}_{as}(t)) = \tilde{e}_{as}^{T}(t)(\tilde{\mathbf{A}}_{s}^{T}(p,p)\mathbf{P}_{a} + \mathbf{P}_{a}\tilde{\mathbf{A}}_{s}(p,p))\tilde{e}_{as}(t)$$

$$+ \tilde{e}_{as}^{T}\mathbf{P}_{a}\tilde{\mathbf{N}}(p,p)\tilde{\mathbf{z}} + \tilde{\mathbf{z}}^{T}\tilde{\mathbf{N}}^{T}(p,p)\mathbf{P}_{a}\tilde{e}_{as}(t)$$
(3.80)

(3.79)

(3.81)

dengan metode yang hampir sama untuk menentukan matriks kestabilan Lyapunov sesuai persamaan (3.57), (3.58), (3.59) dan (3.60) didapatkan LMI (3.81).

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & 0 \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma_{a}\mathbf{I} & 0 & 0 \\ * & * & * & -\gamma_{a}\mathbf{I} & 0 \\ * & * & * & * & -\gamma_{a}\mathbf{I} \\ w_{13} = -\mathbf{H}(p)\mathbf{D}_{f}, \\ w_{22} = -(\mathbf{B}(p)^{T}\mathbf{P}_{a}\mathbf{B}(p) + \mathbf{B}(p)\mathbf{P}_{a}\mathbf{B}(p)^{T}), \\ w_{23} = -\mathbf{B}(p)^{T}\mathbf{H}(p)\mathbf{D}_{f}, \end{bmatrix} \leq 0$$

$$w_{11} = \mathbf{P}_a \mathbf{A}(p) + (\mathbf{P}_a \mathbf{A}(p))^T - \mathbf{H}(p) \mathbf{C}_c - (\mathbf{H}(p) \mathbf{C}_c)^T$$

$$w_{12} = -(\mathbf{A}^T(p)\mathbf{P}_a\mathbf{B}(p) - \mathbf{C}_c^T\mathbf{H}^T(p)\mathbf{B}(p)),$$

untuk meningkatkan performa digunakan LMI fast fault estimation:

$$\begin{bmatrix} \mu_{q}\mathbf{I} & \mathbf{B}(p)^{T}\mathbf{P}_{a} - \mathbf{F}_{a}(p)\mathbf{C}_{c} \\ * & \mu_{q}\mathbf{I} \end{bmatrix} > 0$$
(3.82)

untuk meletakkan pole-pole *observer* pada kriteria yang diinginkan memiliki kriteria performa yang lebih baik, seperti *overshoot*, digunakan LMI Region (3.83)

$$\sum_{ai} + \sum_{ai}^{T} + 2\rho \mathbf{P}_{a} < 0$$

$$\left[\sin(\theta) [\sum_{ai} + \sum_{ai}^{T}] \cos(\theta) [\sum_{ai} + \sum_{ai}^{T}] \right] < 0$$

$$* \sin(\theta) [\sum_{ai} + \sum_{ai}^{T}] < 0$$
(3.83)

Sehingga penggabungan constraint untuk LMI *observer* kesalahan aktuator dari pertidaksamaan (3.81), (3.82) dan (3.83) untuk mencari *observer gain*:

$$\min(\gamma_{a} + \mu_{a})$$

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 & \mathbf{C}_{p1}^{T} & \mathbf{0} \\ * & w_{22} & w_{23} & \mathbf{I} & 0 & \mathbf{C}_{p2}^{T} \\ * & * & -\gamma_{a}\mathbf{I} & 0 & 0 \\ * & * & * & -\gamma_{a}\mathbf{I} & 0 \\ * & * & * & * & -\gamma_{a}\mathbf{I} & 0 \\ * & * & * & * & * & -\gamma_{a}\mathbf{I} \end{bmatrix} < 0$$

$$\begin{array}{c|c} \mu_{a}\mathbf{I} & \mathbf{B}(p)^{T} \mathbf{P}_{a} - \mathbf{F}_{a}(p)\mathbf{C}_{c} \\ * & \mu_{a}\mathbf{I} \end{array} > 0$$

 $\begin{bmatrix} \sum_{ai} + \sum_{ai}^{T} + 2\rho \mathbf{P}_{a} < 0 \\ \begin{bmatrix} \sin(\theta) [\sum_{ai} + \sum_{ai}^{T}] & \cos(\theta) [\sum_{ai} + \sum_{ai}^{T}] \\ * & \sin(\theta) [\sum_{ai} + \sum_{ai}^{T}] \end{bmatrix} < 0$

(3.84)

dengan,

$$w_{11} = \mathbf{P}_{a}\mathbf{A}(p) + (\mathbf{P}_{a}\mathbf{A}(p))^{T} - \mathbf{H}(p)\mathbf{C}_{c} - (\mathbf{H}(p)\mathbf{C}_{c})^{T},$$

$$w_{12} = -(\mathbf{A}^{T}(p)\mathbf{P}_{a}\mathbf{B}(p) - \mathbf{C}_{c}^{T}\mathbf{H}^{T}(p)\mathbf{B}(p)),$$

$$w_{13} = -\mathbf{H}(p)\mathbf{D}_{f},$$

$$w_{22} = -(\mathbf{B}(p)^{T}\mathbf{P}_{a}\mathbf{B}(p) + \mathbf{B}(p)\mathbf{P}_{a}\mathbf{B}(p)^{T}),$$

$$w_{23} = -\mathbf{B}(p)^{T}\mathbf{H}(p)\mathbf{D}_{f},$$
$$\Sigma_{ai} = \mathbf{P}_{a1}\mathbf{A}_{s}(p, p) = \begin{bmatrix} \mathbf{P}_{a1}\mathbf{A}(p) - \mathbf{H}(p)\mathbf{C}_{c} & \mathbf{P}_{a1}\mathbf{B}(p) \\ -(\mathbf{A}^{T}(p)\mathbf{P}_{a1}\mathbf{B}(p) - \mathbf{C}_{c}^{T}\mathbf{H}^{T}\mathbf{B}(p) + \mathbf{B}(p)^{T}\mathbf{P}_{a1})^{T} & -\mathbf{B}(p)^{T}\mathbf{P}_{a1}\mathbf{B}(p) \end{bmatrix}$$

observer gain didapatkan dengan persamaan berikut:

$$\mathbf{L}_{a}(p) = \mathbf{P}_{a}^{-1}\mathbf{H}(p)$$

$$\mathbf{F}_{a}(p)$$
(3.85)

dengan estimasi kesalahan aktuator dituliskan:

$$\hat{f}_a(t) = \mathbf{F}_a(p)\mathbf{C}_c \int (\dot{e}_x(t) + e_x(t))dt$$
(3.86)

4.6. Perhitungan Observer gain Kesalahan Sensor dan Aktuator

Dari Subbab 3.5 telah dijelaskan cara menurunkan LMI untuk kesalahan sensor dan aktuator, spesifikasi desain yang digunakan untuk keduanya adalah:

- 1). Observer berbasis Fuzzy T-S dengan premis dari sudut pendulum dan tiga aturan $(x_2(t) = 0 \text{ rad}, \pm 0.2616 \text{ rad}, \text{ dan} \pm 0.5232 \text{ rad})$
- 2). Sistem memenuhi performa H_{∞} dengan tingkat atenuasi/pelemahan tertentu
- 3). Jumlah kesalahan aktuator = 1, jumlah kesalahan sensor = 1.
- 4). Matriks kesalahan sensor didefinisikan, $\mathbf{D}_{f} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^{T}$
- 5). Matriks performa fungsi Lyapunov (C_p) didefinisikan

	1	U	V	U	U	U	V	
	0	1	0	0	0	0	0	
	0	0	1	0	0	0	0	
$C_{p,1} = 0.4$	0	0	0	1	0	0	0	
	0	0	0	0	1	0	0	
	0	0	0	0	0	1	0	
	0	0	0	0	0	0	1	
$C_{n,2} = 0.4$	[1]		12			1	2	K

$$C_{pa,1} = 2 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C_{pa,2} = 2[1]$$

6). Spesifikasi LMI Region untuk *observer* kesalahan sensor $\rho = 0.1$

$$\theta = \frac{\pi}{4}$$

7). Spesifikasi LMI Region untuk *observer* kesalahan aktuator $\rho = 0.5$

 $\theta = \frac{\pi}{4}$

Dari LMI untuk kesalahan sensor (3.66) dan LMI untuk kesalahan aktuator (3.86) didapatkan *observer gain* untuk masing-masing subsistem sebagai berikut:

	-106,0	42,80	66,60	44	-104,2	42,60	65,00	
	- 408,1	344,7	89,00		<mark>- 39</mark> 3,0	332,6	85,20	
ABC	- 575,1	- 296,2	857,8		- 549,9	- 267,5	805,0	
L ₁ =	- 2087	909,5	1254	L ₂ =	-1961	890,9	1144	
17 Th	-11,00	13,10	0,600		-11,50	13,10	0,600	
SE	- 48,80	56,90	-2,500		-47,70	57,70	- 2,400	K
- OL	- 61,90	- 0,500	64,00		- 60,30	-0,400	61,80	1
			[-118,3	60,	50 61	,70]		
		WA-	-434,5	5 39	0,1 69	,70		
			- 53,62	2 -25	57,5 78	1,5		
		L ₃ =	-1953	10-	44 98	1,0		
17/17		Tr bd	-13,80) 15,	30 0,7	700		
	223	25	- 55,90) 63,	80 - 2	,300		K
			- 63,60) -0,	540 60	,20		

$$\mathbf{L}_{a,1} = \begin{bmatrix} -11,85 & 14,51 & -2,308 \\ -33,05 & 41,89 & -7,857 \\ -5,742 & 6,588 & -0,684 \\ -114,6 & 147,1 & -29,35 \end{bmatrix} \mathbf{L}_{a,2} = \begin{bmatrix} -11,93 & 14,94 & -2,608 \\ -32,74 & 42,39 & -8,560 \\ -5,603 & 6,551 & -0,772 \\ -111,5 & 145,9 & -31,01 \end{bmatrix}$$
$$\mathbf{L}_{a,3} = \begin{bmatrix} -12,04 & 16,44 & -3,845 \\ -31,27 & 44,01 & -11,34 \\ -5,142 & 6,473 & -1,112 \\ -101,2 & 142,8 & -37,49 \end{bmatrix}$$

Gain integralnya,

 $\overline{\mathbf{F}}_{1} = \begin{bmatrix} 513,33 & -110,72 & -10,078 \end{bmatrix}$ $\overline{\mathbf{F}}_{2} = \begin{bmatrix} 427,20 & -94,193 & -8,7106 \end{bmatrix}$ $\overline{\mathbf{F}}_{3} = \begin{bmatrix} 418,11 & -94,128 & -9,1363 \end{bmatrix}$ $\mathbf{F}_{a,1} = \begin{bmatrix} -67,084 & -39,403 & 385,70 \end{bmatrix}$ $\mathbf{F}_{a,2} = \begin{bmatrix} -69,309 & -37,032 & 386,10 \end{bmatrix}$ $\mathbf{F}_{a,3} = \begin{bmatrix} -60,042 & -27,020 & 328,43 \end{bmatrix}$

Sedangkan atenuasi yang didapatkan dari hasil minimisasi ($\gamma + \mu$) dan minimisasi ($\gamma_a + \mu_a$) adalah:

 $\gamma = 0,0884$ $\gamma_a = 0,5058$ $\mu = 0,0534$ $\mu_a = 0,4878$

4.7. Diagram Simulink Kontroler dan FTC untuk Simulasi dan Implementasi

Hasil perancangan kontroler pada Subbab 3.3 dan perancangan FTC pada Subbab 3.5 terlebuh dahulu disimulasikan menggunakan MATLAB Simulink. Model Simulink hasil rancangan dapat dilihat pada Gambar 3.8. Warna hijau pada model Simulink menunjukkan diagram untuk kontrol nominal, warna kuning untuk *observer* dan kompensasi kesalahan aktuator, sedangkan warna biru untuk *observer* dan

kompensasi kesalahan sensor. Semua variabel $x_1(t)$, $x_2(t)$, $x_3(t)$ dan $x_4(t)$, sinyal kontrol, derajat keanggotaan, L_2 -gain ditampilkan pada scope masing-masing. Selain itu untuk mengetahui Integral Absolute Error (IAE) dari posisi kereta dapat dilihat pada scope IAE.

Setelah melakukan simulasi dan didapatkan hasil yang cukup baik, maka sistem kontrol hasil desain diuji pada *plant* nyata, "*Digital Pendulum Mechanical Unit* 33-200" [6]. Diagram Simulink yang digunakan dapat dilihat pada Gambar 3.9. *State* yang terukur, yaitu posisi kereta $(x_1(t))$ dan posisi sudut pendulum $(x_2(t))$ dibaca melalui blok sensor posisi kereta dan sensor posisi sudut pendulum. Sedangkan *state* yang tidak terukur, yaitu kecepatan kereta $(x_3(t))$ dan kecepatan sudut pendulum $(x_4(t))$ didapat dengan mengambil turunan pertama dari posisi kereta $(x_1(t))$ dan posisi sudut pendulum $(x_2(t))$.

Gambar 3.9 menunjukkan diagram Simulink untuk implementasi. Sinyal kontrol yang didapat dari kontroler masuk ke dalam blok DAC. Pada blok DAC, sinyal kontrol yang berupa gaya dalam satuan Newton dikonversikan ke dalam tegangan dalam satuan Volt. Hubungan antara gaya dan tegangan diasumsikan linear, yaitu V = 1/9 u. Karena DAC yang digunakan bekerja pada $\pm 2,5$ Volt, maka tegangan yang digunakan dibatasi pada $\pm 1,9$ Volt. Isi dari masing-masing blok pada Gambar 3.8 secara detil dapat dilihat pada Lampiran B.3.





BAB IV HASIL PENGUJIAN SIMULASI DAN IMPLEMENTASI

Pada Bab ini akan dijelaskan hasil simulasi dan implementasi rancangan kontrol nominal dan *observer* pada sistem pendulum kereta. Simulasi menggunakan Matlab 9.5.1 dan implementasi menggunakan "*Digital Pendulum Mechanical Unit* 33-200" dari *Feedback Instrument* Ltd. Simulasi dilakukan dengan berbagai kondisi kesalahan, yaitu bias berdasarkan dinamikanya (sinus dan random Gaussian) karakteristiknya (*abrupt fault, incipient fault* dan *intermittent fault*) dan multiplikatif.

4.8. Hasil Simulasi

Pada subbab ini akan dijelaskan simulasi kondisi tanpa kesalahan (*fault-free case*) dan kondisi dengan kesalahan sensor dan aktuator (*faulty case*). Simulasi menggunakan Matlab 9.5.1.

4.8.1. Hasil Simulasi Kontrol Nominal dengan Berbagai Kondisi Awal tanpa Kesalahan (*Fault-Free Case*)

Pada bagian ini simulasi dilakukan dengan dua kondisi awal yaitu pada sudut 0,2 radian dan sudut 0,4 radian. *State feedback gain* telah ditentukan pada perancangan di Bab III Perancangan. Kondisi awal untuk posisi kereta, dan kecepatan kereta adalah nol, sedangkan kondisi awal kecepatan sudut kereta ditentukan agar sinyal kontrol memenuhi batasan yang diinginkan. Gambar 4.1 menunjukkan grafik respon posisi kontrol nominal tanpa kesalahan dengan berbagai kondisi awal.



Gambar 4.1 Respon Posisi Kontrol Nominal Hasil Simulasi

Pada Gambar 4.1 terlihat untuk kondisi awal sudut pendulum 0,2 radian memiliki memiliki respon yang lebih baik dibandingkan kondisi awal sudut pendulum 0,4 radian. Besar selisih amplitudo absolut masing-masing untuk kondisi awal 0,2 radian dan 0,4 radian adalah 0,00002 dan 0,000021. Beda fasa untuk kedua kondisi awal sangat kecil sehingga respon posisi dikatakan memenuhi trayektori yang diinginkan pada detik ke 3,5 dan seterusnya. Pada waktu sekitar 0,2 detik, kereta bergerak ke menuju posisi -0,07 m dan -0,09 m untuk masing-masing sudut awal 0,2 dan 0,4 radian. Kondisi ini terjadi karena kereta berusaha untuk memaksa posisi batang pendulum menuju sudut nol radian dari sudut awal yang diberikan. Nilai *Integral Absolute Error* (IAE) adalah 0,1256 dan 0,1286 untuk masing-masing sudut awal 0,2 dan 0,4 radian. Terlihat nilai IAE sangat kecil sehingga posisi kereta dapat mengikuti sinyal referensi dengan sangat baik.

Gambar 4.2 menunjukkan respon sudut pendulum dengan sudut awal yang bervariasi. Untuk dua kondisi awal yang berbeda yaitu sudut awal 0,2 dan 0,4 radian, mengalami penurunan nilai dan melonjak ke nilai untuk masing-masing sudut awal 0,04 dan 0,07 radian. Pada saat kondisi *steady state*, terdapat osilasi sekitar 0,004 radian. Osilasi ini diakibatkan oleh *tracking* yang dilakukan oleh kereta. Secara umum respon sudut pendulum memiliki performa yang baik.

Gambar 4.3 menunjukkan sinyal kontrol untuk dua kondisi awal yang berbeda. Terlihat untuk sudut awal pendulum 0,4 radian memiliki sinyal kontrol maksimal yang lebih besar dibandingkan sudut awal 0,2 radian yaitu sebesar -800 N, sedangkan untuk sudut awal 0,2 radian memiliki sinyal kontrol -310 N.



Gambar 4.2 Respon Sudut Pendulum dengan Berbagai Kondisi Awal Sudut Pendulum Hasil Simulasi

Besar *overshoot* sinyal kontrol untuk sudut awal 0,2 radian dan 0,4 radian hampir sama yaitu 20 N. Sinyal kontrol ini tidak memenuhi batasan yang telah dirancang, yaitu 17,5 N. Untuk memenuhi batasan *input-output* diberikan kondisi awal pada kecepatan sudut. Pada simulasi ini, kondisi awal yang diberikan adalah [0 0,2 0 -0,77] dan [0 0,4 0 - 1,53].



Gambar 4.3 Sinyal Kontrol dengan Berbagai Kondisi Awal Sudut Pendulum Hasil Simulasi

Gambar 4.4 menunjukkan grafik kontrol nominal *tracking* dengan kondisi awal pada sudut pendulum dan kecepatan pendulum. Besar selisih maksimum amplitudo untuk kedua kondisi awal hampir sama saat tidak diberi kecepatan sudut pendulum, 0,00002 dan 0,000021.



Gambar 4.4 Kontrol Nominal dengan Berbagai Kondisi Awal Sudut Pendulum dan Kecepatan Sudut Pendulum Hasil Simulasi

Besar IAE untuk kondisi awal sudut pendulum 0,2 radian dan kecepatan sudut -0,77 rad/s adalah 0,1194, dan untuk kondisi awal sudut pendulum 0,4 radian dan kecepatan sudut -1,43 rad/s adalah 0,1239. Nilai ini lebih kecil dibanding nilai IAE grafik pada Gambar 4.1, sehingga disimpulkan respon posisi ini memiliki performa yang lebih baik dibandingkan tidak memiliki kondisi awal pada kecepatan sudut pendulum.

Gambar 4.5 menunjukkan respon sudut pendulum untuk berbagai kondisi awal. Untuk kedua kondisi awal pendulum dapat distabilkan sekitar 2 detik. Untuk sudut awal 0,2 radian memiliki *undershoot* pada nilai sekitar 0.008 radian, sedangkan untuk kondisi awal 0,4 radian tidak memiliki *undershoot*. Pada detik ke-2 dan seterusnya terjadi osilasi sudut pendulum di sekitar 0,004 radian.



Gambar 4.5 Respon Sudut Pendulum dengan Berbagai Kondisi Awal Sudut Pendulum dan Kecepatan Sudut Hasil Simulasi

Gambar 4.6 menunjukkan sinyal kontrol untuk kondisi awal sudut pendulum 0,2 radian, kecepatan sudut -0,77 rad/s dan sudut pendulum 0,4 radian, kecepatan sudut -1,43 rad/s. Untuk sudut awal 0,2 radian memiliki *overshoot* 0,9 N sedangkan untuk sudut awal 0,4 radian memiliki *overshoot* 7 N. Sinyal kontrol berosilasi pada detik ke-6 dan seterusnya dengan nilai osilasi di sekitar 1,5 N, hal ini dikarenakan *tracking* sinyal trayektori yang digunakan sinus, osilasi sinyal kontrol membentuk sinus.

Sinyal kontrol maksimal untuk kedua kondisi adalah 7 N dan -14 N. Besarnya sinyal kontrol ini sudah sesuai dengan batasan *input-output* yang ditetapkan yaitu maksimal sinyal kontrol yang dihasilkan adalah sebesar 17,5 N. Sinyal kontrol yang terlalu besar mengakibatkan sistem tidak dapat diimplementasikan. Pemberian nilai kondisi awal pada

kecepatan sudut mengakibatkan sinyal kontrol yang dihasilkan lebih kecil dan sesuai dengan kriteria perancangan yang telah diterapkan.





Tabel 4.1 menunjukkan nilai-nilai keseluruhan dari perbandingan hasil respon untuk berbagai kondisi awal. Terdapat empat kriteria respon yaitu undershoot posisi kereta, undershoot posisi sudut pendulum, sinyal kontrol minimal dan sinyal kontrol maksimal.

Kriteria	Kondisi Awal					
Kincina	[0 0,2 0 0]	[0 0,4 0 0]	[0 0,2 0 -0,77]	[0 0,4 0 -1,53]		
<i>Undershoot</i> Posisi Kereta (m)	0,07	0,09	0,01	0,06		
Undershoot Posisi Sudut Pendulum (rad)	0		0,008			
Sinyal Kontrol Minimal (N)	-310	-800	-2	-14		
Sinyal Kontrol Maksimal (N)	20	20	6	6		
IAE	0,1256	0,1286	0,1194	0,1239		

Tabel 4.1 Perbandingan Respon Simulasi Berbagai Kondisi Awal

Kriteria respon dari Tabel 4.1 adalah respon kontrol nominal tanpa *observer*. Ketika *observer* terkoneksi dengan kontrol nominal terdapat perbedaan respon transien. Kondisi awal yang diberikan pada *plant* mempengaruhi respon pada kontrol nominal. Perbedaan ini dapat dilihat pada Gambar 4.7. Grafik pada Gambar 4.7 menunjukkan pengaruh kondisi awal pada *observer*.





Ketika observer telah terkoneksi dengan kontrol nominal, pemberian kondisi awal pada sudut 0 radian dan kecepatan sudut 0 rad/s memiliki respon dengan undershoot paling kecil, hal ini dikarenakan kondisi awal *observer* sama dengan kondisi awal pada *plant*, vaitu [0 0 0 0]. Sedangkan ketika plant diberi kondisi awal pada sudut 0,2 radian dan kecepatan sudut -0,77 rad/s terdapat undershoot pada -0,1 radian. Respon paling buruk terjadi ketika pemberian kondisi awal paling besar vaitu sudut 0,4 radian dan kecepatan sudut -1,43 rad/s, terdapat undershoot pada -0,17 radian. Hal ini terjadi karena observer memerlukan waktu untuk menyesuaikan dengan kondisi awal dari plant, waktu penyesuaian ini mengakibatkan respon transien berubah. Untuk kriteria Integral Absolute Error (IAE), plant dengan kondisi awal sudut 0 radian memiliki IAE 0,1187, kondisi awal sudut 0,2 radian memiliki IAE 0,1969 dan kondisi awal sudut 0,4 radian memiliki IAE 0,2940, semakin besar kondisi awal yang diberikan semakin besar nilai IAE, yang berarti respon yang didapatkan semakin buruk.

Tabel 4.2	Respon	Pengaruh	Pemberian	Kondisi	Awal	pada P	lant	dengan
Observer ya	ang Terke	oneksi Kon	trol Nomina		\sim		¥.	

Vritorio		Kondisi Awal	
KITTEITä	[0 0 0 0]	[0 0,2 0 -0,77]	[0 0,4 0 -1,53]
<i>Undershoot</i> Posisi Kereta (m)	0,01	0,1	0,17
IAE	0,1187	0,1969	0,2940

4.8.2. Hasil Simulasi Kontrol Nominal dengan Berbagai Kondisi Kesalahan (*Faulty Case*)

Pada sub-subbab ini dibahas kontrol nominal dengan berbagai kondisi kesalahan yang terdapat pada sensor dan aktuator, secara umum kesalahan yang diuji adalah kesalahan aditif dan multiplikatif. Kesalahan aditif terdiri atas sinyal random dan sinus. Selain itu terdapat kesalahan menurut proses waktunya yaitu, *intermittent fault, incipient fault dan abrupt fault.* Untuk kesalahan multiplikatif terdiri atas *loss of measurement* dan *lost of effectiveness.* Semua kesalahan disimulasikan pada kondisi awal [0 0,2 0 -0,77], karena memiliki respon dengan *observer* yang cukup baik.

4.8.2.1. Simulasi Faulty-Case Sensor dengan Kesalahan Bias (Aditif) Kesalahan bias adalah kesalahan yang bersifat aditif (pertambahan) terhadap keluaran, seperti yang telah dijelaskan pada subbab 2.8. Pengujian pertama terhadap sinyal sinus, dengan kesalahan sinus direpresentasikan dalam persamaan matematis berikut:

$$f_s(t) = 0.05k * \sin(0.5\pi)$$

k = 1.2.3 (4.1)

Gambar 4.8 memperlihatkan grafik kontrol nominal dengan kesalahan aditif. Semakin besar kesalahan sinus, semakin buruk respon yang didapatkan. Dari grafik terlihat bahwa pemberian kesalahan sinus menyebabkan penyimpanan pada sinyal referensi. Besarnya penyimpangan dapat dilihat pada nilai IAE. Untuk kesalahan sensor $0,05\sin(0,5\pi)$ nilai IAE sebesar 0,1995, untuk kesalahan $0,10\sin(0,5\pi)$ nilai IAE sebesar 0,2018, untuk kesalahan $0,15\sin(0,5\pi)$ nilai IAE sebesar 0,2048.



Gambar 4.8 Faulty-case Sensor Kesalahan Bias Sinus Hasil Simulasi

Sedangkan estimasi kesalahan dapat dilihat pada Gambar 4.9. Pada grafik terlihat *observer* mampu mengestimasi kesalahan yang terjadi, kesalahan yang diestimasi adalah 0,15sin(0,5 π). Waktu untuk menyesuaikan dengan kesalahan yang terjadi adalah 7 s. Setelah detik ke-7 *observer* mengestimasi kesalahan dengan baik, hal ini ditunjukkan dengan nilai IAE yang stabil pada detik ke-7 dan seterusnya, yaitu 0,2048. *Observer* perlu menyesuaikan dengan kondisi awal dari *plant*.



Gambar 4.9 Estimasi Kesalahan Sensor Berbentuk Sinus Hasil Simulasi

Sedangkan Gambar 4.10 memperlihatkan error estimasi kesalahan, sesuai perancangan pada Subbab 3.4 error estimasi kesalahan adalah kesalahan sebenarnya dikurangi dengan estimasi kesalahan. Error estimasi terbesar adalah pada detik ke-1 dengan *absolute error* sekitar 0,07m. Pada detik ke-7 dan seterusnya error kesalahan berosilasi pada nilai 0,015m.



Gambar 4.10 Error Estimasi Kesalahan Sensor Sinus Hasil Simulasi

Simulasi selanjutnya adalah kesalahan sensor berbentuk sinyal random. Sinyal random dalam hal ini berbentuk *Gaussian*. Kesalahan dibuat dalam varian yang berbeda-beda. Secara umum bentuk kesalahannya adalah:

$$f_{s}(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x-\mu)^{2}}{2\sigma^{2}}\right]$$

$$\sigma = 0,001 + 0,001k, \mu = 0,00$$

$$k = 0,1,2$$
(4.2)

Gambar 4.11 memperlihatkan respon kontrol nominal setelah diberi kesalahan berbentuk random Gaussian. Dari grafik terlihat untuk kesalahan yang paling besar memperlihatkan respon yang paling dengan penyimpangan yang paling besar. Nilai IAE untuk kesalahan dengan varian 0,001 adalah 0,3806, sedangkan nilai IAE untuk kesalahan dengan varian 0,002 dan 0,003 berturut-turut adalah 0,7379 dan 1,406. Error yang terjadi pada kesalahan random Gaussian terpengaruh pada besarnya varian, hal ini dikarenakan nilai IAE adalah *integral* luasan dibawah kurva, sehingga dengan varian yang besar luasan error dibawah kurva lebih besar atau lebih luas.





Estimasi kesalahan sensor diperlihatkan pada Gambar 4,12. Pada grafik ini terlihat *observer* mampu mengestimasi kesalahan sensor dengan baik pada detik ke-7 dan seterusnya. Hal ini terlihat dari respon yang hampir sama kesalahan yang terjadi dengan estimasi kesalahan. Untuk kesalahan bentuk random Gaussian terdapan respon transien pada setiap transisi besarnya kesalahan yang terjadi, hal ini terlihat pada zoom Gambar 4.12. *Observer* mengikuti kesalahan mirip dengan respon orde kesatu.



Gambar 4.12 Estimasi Kesalahan Sensor Random Gaussian Hasil Simulasi

Error estimasi kesalahan diperlihatkan pada Gambar 4.13. Error estimasi memperlihatkan amplitudo yang berbeda-beda dengan periode yang singkat. Hal ini menunjukkan penyesuain *observer* terhadap kesalahan yang terjadi, yaitu berbentuk sinyal random. Kesalahan berbentuk sinyal random Gaussian mirip dengan noise yang terjadi pada *plant*. Hal ini dikarenakan sifatnya acak dan berlangsung dengan periode tertentu secara acak.



Gambar 4.13 Error Estimasi Kesalahan Sensor Random Gaussian Hasil Simulasi

	Kriteria					
Jenis Kesalahan	Error Estimasi Maksimum (m)	IAE				
$f_{s}(t) = 0.05 \sin(0.5\pi t)$	0,07	0,1995				
$f_s(t) = 0,10sin(0,5\pi t)$	0,07	0,2018				
$f_s(t) = 0.15 \sin(0.5\pi t)$	0,07	0,2048				
Random Gaussian, mean $= 0$, varian $= 0,001$	0,45	0,3806				
Random Gaussian, mean = 0 , varian = $0,002$	0,46	0,7379				
Random Gaussian, mean $= 0$, varian $= 0,003$	0,47	1,4060				

Tabel 4.3 Respon Kontrol Nominal Terhadap Kesalahan Sensor

 Berbentuk Sinus dan Random

Kesalahan sensor selanjutnya adalah aditif dengan jenis kesalahan *intermittent*, *intermittent* dan *abrupt*. Kesalahan *abrupt* mirip dengan kesalahan fungsi step yang berlangsung kontinu dimulai pada waktu tertentu. Secara umum fungsi kesalahan *abrupt* yang diuji adalah:

$$f_s(t) = 0.05k * u(t-1)$$

k = 1, 2, 3

(4.3)

Respon dari kontrol nominal setelah diberi kesalahan ini dapat dilihat pada Gambar 4.14. Dari grafik terlihat keluaran sistem bergeser dari travektori yang diinginkan.

(0)



Gambar 4.14 Faulty Case Sensor dengan Kesalahan Bias Abrupt Hasil Simulasi

Semakin besar amplitudo kesalahan, semakin besar pergeseran yang terjadi. Nilai IAE berturut-turut untuk kesalahan dengan amplitudo 0,05 0,10 dan 0,15m adalah 0,2561, 0,3960 dan 0,5380. Pada detik dimana kesalahan *abrupt* terjadi, yaitu detik ke-10, terjadi *overshoot*. *Overshoot* tertinggi terjadi pada kesalahan dengan amplitudo paling besar, yaitu 0,24m. Sedangkan *overshoot* untuk kesalahan amplitudo 0,05 dan 0,10m adalah 0,16m dan 0,09m.

Estimasi kesalahan dapat dilihat pada Gambar 4.15. Kesalahan *abrupt* terjadi tiba-tiba pada detik ke-10 yang berlangsung secara kontinu. Pada detik ke-1 sampai ke-10 *observer* menyesuaikan diri dengan kondisi awal dari *plant* sehingga tampak terjadi *overshoot*. Respon transien juga terjadi pada detik ke-10 dimana *observer* memerlukan waktu untuk mengestimasi kesalahan sampai *steady-state*.





Kesalahan berdasarkan karakteristiknya, selanjutnya adalah kesalahan *intermittent*. Kesalahan *intermittent* hampir sama dengan kesalahan inpuls, dimana kesalahan muncul secara tiba-tiba dan berhenti secara tiba-tiba dengan periode yang sangat singkat. Fungsi kesalahan ini dirumuskan sebagai berikut:

$$f_s(t) = 0.05k * u((t-10) - (t-10,1))$$

k = 1,2,3 (4.4)

Periode kesalahan *intermittent* adalah 0,1s. Respon kontrol nominal terhadap kesalahan ini dapat dilihat pada Gambar 4.16. Kesalahan *intermittent* berakibat penyimpangan secara tiba-tiba respon sistem lup tertutup terhadap keadaan *steady state*-nya. Respon paling

DAFTAR PUSTAKA

- [1] Hassan Noura, Fault Tolerant Control Systm : Design and Its Application, United Arab Emrates: United Arab Emrates University, 2009.
- [2] M. Blanke, M. Kinneart, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control, 2nd ed.* Berlin/Heidelberg, German: Springer-Verlag, 2006.
- [3] K. Ogata, Modern Control Engineering. New Jersey: Prentice-Hall, 1997.
- [4] Montadher Sami, "Active Fault-Tolerant Control of Nonlinear Systems with Wind Turbine Application," University of Hull, Hull, PhD Thesis 2012.
- [5] A. Benzaouia and A. El Hajjaji, Advanced Takagi-Sugeno Fuzzy Systems. Switzerland: Springer, 2014.
- [6] "Control in a MATLAB Environment," *Feddback Instruments Ltd*, 2004.
- [7] Andir Ashfahani and Trihastuti Agustinah, "Desain Fuzzy Tracking Controller pada Pendulum Terbalik dengan Memperhitungkan Model Friksi," in *SITIA*, Surabaya, 2008.
- [8] Trihastuti Agustinah, Achmad Jazidie, and Mohammad Nuh, "Fuzzy Tracking Control Based on H-infinity Performance for Nonlinear Systems," WSEAS Transactions on Systems and Control, pp. 393-403, 2011.
- [9] M. Sami and R. J. Patton, "Active Fault Tolerant Control for Nonlinear Systems with Simultaneous Actuator and Sensor Faults," *International Journal Control, Automation, and Systems*, vol. 11(6), pp. 1149-1161, 2013.
- [10] Marcin Witczak, Fault Diagnosis and Fault-Tolerant Control Strategies for Non-Linear Systems. New York: Springer, 2013.
- [11] Krishna K. Busawon and Pousga Kabore, "On the design of integral and proportional integral observer," *Proceedings of the American Control Conference*, 2000.
- [12] Atef Kheder and Kamel Benothman, "Active Fault Tolerant Control (FTC) Design for Takagi-Sugeno Fuzzy Systems with Weighting Functions Depending on the FTC," *International Journal of*

Computer Science Issues, vol. 8, no. 3, pp. 1694-0814, May 2011.

- [13] S. Boyd, L. El Ghouni, and Balakrishnan, "Linear Matrix Inequality in System and Control Theory," *SIAM Philadelphia*, 1994.
- [14] Usha Mahato, "Low Gain Controller Design with Regional Pole Placement Constraint," National Institute of Technology Rourkela, Odisha, Master Thesis 2013.
- [15] R. Isermann, Fault Diagnosis Systems (An Introduction from Fault Detection to Fault Tolerant). New York: Kluwer Academic Publisher, 2006.
- [16] Halim Alwi, Christoper Edwards, and Chee Pin Tan, "Fault Detection and Fault Tolerant Control Using Sliding Modes," pp. 7-28, 2011.
- [17] Kemin Zhou, *Essentials of Robust Control*. New Jersey: Prentice-Hall, 1999.
- [18] S. Boyd, L. El Ghaouni, E. Feron, and V. Balakrishnan, *Linear Matrix Inequality in System and Control Theory*. Philadelphia: SIAM, 1994.
- [19] Ke Zhang, Bin Jiang, and Vincent C., "Adaptive Observer-based Fast Fault Estimation," in *International Journal of Control*, *Automation, and Systems*, 2008, pp. vol. 6, no. 3, pp. 320-326.



BAB V PENUTUP

Dari hasil pengujian simulasi dan implementasi, kontrol nominal yang dirancang dapat membuat Sistem Pendulum-Kereta mengikuti sinyal referensi yang diinginkan, yaitu sinyal berbentuk sinus dengan tetap membertahankan batang pendulum di sekitar 0 radian. Selain itu kontrol nominal memenuhi spesifikasi batasan input-output dan LMI *region* yang telah dirancang. Dari hasil simulasi dan implementasi, Kontrol toleransi kesalahan (FTC) vang dirancang mampu mengompensasi kesalahan sensor, aktuator, atau sensor dan aktuator secara simultan dengan baik. Hal ini terlihat dari error estimasi kesalahan sensor dan aktuator yang kecil dan nilai IAE dengan kompensasi FTC lebih kecil dibandingkan nilai IAE tanpa kompensasi FTC. Observer untuk kesalahan sensor dan aktuator juga memenuhi performa H_{α} dengan nilai pelemahan dibawah dari y yang dirancang saat teriadi kesalahan.

LAMPIRAN A

A.1 Penurunan Persamaan State Sistem Pendulum-Kereta

Persamaan *state* Sistem Pendulum-Kereta dapat diperoleh dari analisis gerak sesuai hukum kedua Newton. Persamaan gerak translasi pada bidang horizontal adalah

$$\sum F_x = ma_x$$

(A.1)

(A.2)

(A.5)

Dapat dilihat bahwa gaya-gaya yang bekerja pada bidang horizontal adalah gaya kontrol u dan gaya gesek T_c . Massa keseluruhan sistem (m) adalah jumlah dari massa kereta (m_c) dan massa pendulum (m_p) . Percepatan pusat massa pada bidang horizontal adalah

$$a_r = (x_1 - l \sin x_2)^{"}$$

Dari (A.1) dan (A.2), diperoleh(2.1) yang dapat diuraikan sebagai berikut:

$$u - T_c = (m_c + m_p)(x_1 - l\sin x_2)"$$

$$u - T_c = (m_c + m_p)(x_1' - x_2' l\cos x_2)'$$

$$u - T_c = (m_c + m_p)(x_1'' - x_2'' l\cos x_2 + (x_2')^2 l\sin x_2)$$
(A.3)

Persamaan gerak translasi pada bidang vertikal dapat ditulis sebagai berikut:

$$\sum F_y = ma_y \qquad (A.4)$$

Dari Gambar 2.4 dan Gambar 2.5, dapat dilihat bahwa gaya-gaya yang bekerja pada bidang vertikal adalah gaya normal V dan berat sistem, yaitu $(m_c + m_p)g$. Percepatan pusat massa pada bidang vertikal adalah

$$a_v = (l \cos x_2)'$$

Dari (A.4) dan (A.5), diperoleh (2.2) yang dapat diuraikan sebagai berikut:

$$V = (m_c + m_p)g = (m_c + m_p)(l\cos x_2)''$$

$$V = (m_c + m_p)g = (m_c + m_p)(-lx_2'\sin x_2)'$$

$$V = (m_c + m_p)g + (m_c + m_p)(-lx_2''\sin x_2 - l(x_2')^2\cos x_2)$$

$$V = (m_c + m_p)(g - lx_2''\sin x_2 - l(x_2')^2\cos x_2)$$
(A.6)

Selanjutnya, persamaan gerak rotasi dapat ditulis sebagai berikut:

(A.7)

(A.9)

dengan α adalah percepatan sudut dan τ adalah torsi yang bekerja pada sistem. (A.7) dapat dijabarkan sehingga diperoleh persamaan gerak rotasi sesuai dengan (2.3), yaitu

$$\frac{(u - T_c)l\cos x_2 + Vl\sin x_2 - D_p = Jx_2"}{(u - T_c)l\cos x_2 + Vl\sin x_2 - f_p x_4 = Jx_2"}$$
(A.8)

dengan D_p adalah momen gesek akibat gerak putar pendulum.

Dinamika *state* x_4 dapat diperoleh dengan melakukan substitusi (A.6) ke (A.8) sebagai berikut:

$$Jx_2'' = (u - T_c)l\cos x_2 - f_p x_4$$

 $\sum \tau = J\alpha$

$$+(m_c + m_p)(g - lx_2"\sin x_2 - l(x_2')^2 \cos x_2)l\sin x_2$$

Dengan mendefinisikan $\mu = (m_c + m_p)l$, maka diperoleh $lr_{c'} = (\mu - T)l\cos r_c = f_c r_c + (m_c + m_c)l(g \sin r_c)$

$$Jx_{4} = (u - T_{c})t \cos x_{2} - J_{p}x_{4} + (m_{c} + m_{p})t(g \sin x_{2})$$

$$-lx_{4}'\sin^{2}x_{2} - l(x_{2}')^{2}\cos x_{2}\sin x_{2})$$

$$Jx_{4}' = (u - T_{c} - \mu x_{4}^{2}\sin x_{2})l \cos x_{2} + \mu g \sin x_{2}$$

$$-\mu lx_{4}'\sin^{2}x_{2} - f_{p}x_{4}$$

$$x_{4}'(J + \mu l \sin^{2}x_{2}) = (u - T_{c} - \mu x_{4}^{2}\sin x_{2})l \cos x_{2} + \mu g \sin x_{2} - f_{p}x_{4}$$

$$x_{4}' = \frac{(u - T_{c} - \mu x_{4}^{2}\sin x_{2})l \cos x_{2} + \mu g \sin x_{2} - f_{p}x_{4}}{J + \mu l \sin^{2}x_{2}}$$
(A.10)

Dinamika state x3 dapat diperoleh dari (A.3) sebagai berikut:

$$u - T_c = (m_c + m_p)(x_3' - x_4' l \cos x_2 + (x_4)^2 l \sin x_2)$$

 $(m_c + m_p)x_3' = u - T_c - \mu r_4^2 \sin x_2 + \mu r_4' \cos x_2$

(A.11)

Dengan mendefinisikan $a = l^2 + J/(m_c + m_p)$ danmelakukan substitusi (A.10) ke (A.11), maka akan diperoleh

$$(m_{c} + m_{p})x_{3}^{'} = u - T_{c} - \mu x_{4}^{2} \sin x_{2}$$

$$+ \frac{\mu' \cos^{2} x_{2}(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$+ \frac{\mu \cos x_{2}(\mu z \sin x_{2} - f_{p} x_{4})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu' \sin^{2} x_{2})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$+ \frac{\mu' \cos^{2} x_{2}(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu')(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu')(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu')(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu')(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu')(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})x_{3}^{'} = \frac{(J + \mu')(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

$$(m_{c} + m_{p})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})$$

$$(m_{c} + m_{p})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})$$

$$(m_{c} + m_{p})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})$$

$$(m_{c} + m_{p})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})$$

$$(m_{c} + m_{p})(u - T_{c} - \mu x_{4}^{2} \sin x_{2})$$

$$x_{3}' = \left(l^{2} + \frac{J}{(m_{c} + m_{p})}\right) \frac{(u - T_{c} - \mu x_{4}^{2} \sin x_{2})}{J + \mu' \sin^{2} x_{2}}$$

+
$$\frac{l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu' \sin^{2} x_{2}}$$

$$x_{3}' = \frac{a(u - T_{c} - \mu x_{4}^{2} \sin x_{2}) + l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu' \sin^{2} x_{2}}$$

Dari (A.10) dan (A.12), dapat diperoleh bentuk persamaan *state* Sistem Pendulum-Kereta sepertipada (2.4).

(A.12)

Momen inersia dari Sistem Pendulum-Kereta diperoleh dari hasil analisis momen inersia yang diilustrasikan pada Gambar A.1. Dari gambar tersebut, m_{pw} adalah massa beban pendulum (kg), m_{ps} adalah massa batang pendulum (kg), l_p adalah panjang batang pendulum (m), l_{po} adalah jarak antara titik pusat massa batang pendulum dengan sumbu putar batang pendulum (m), l_c adalah panjang beban pendulum (m), l_{co} adalah jarak antara titik pusat massa beban pendulum dengan sumbu putar batang pendulum (m), r_p adalah jari-jari batang pendulum (m), dan r_c adalah jari-jari beban pendulum (m). Momen inersia pendulum dari sumbu rotasi pendulum ditunjukkan oleh (A.13).



Gambar A.2 Ilustrasi Momen Inersia Sistem Pendulum-Kereta

$$J_{p} = m_{pw} \left(\frac{1}{12} l_{c}^{2} + \frac{1}{4} r_{c}^{2} + l_{co}^{2} \right) + m_{ps} \left(\frac{1}{12} l_{p}^{2} + \frac{1}{4} r_{p}^{2} + l_{po}^{2} \right)$$

(A.13)

(A.14)

(A.15)

Untuk kesederhanaan perhitungan, pendulum diasumsikan sebagai silinder dengan ketebalan yang seragam dan memiliki panjang Lserta massa m_p . Posisi sumbu putar berada pada pusat massa kereta dan ujung batang pendulum sebagai kereta dapat dipandang sebagai titik massa M_c sedangkan pendulum sebagai titik massa M_p . Titik pusat massa sistem berada di dekat titik pusat massa kereta seperti diilustrasikan pada Gambar A.2. Jika momen inersia dari sumbu tertentu terhadap titik pusat massa dinyatakan dengan Jdan momen inersia dari sumbu yang paralel dengan sumbu sebelumnyadan dipisahkan oleh jarak sebesar d dinyatakan dengan J_p , maka hubungan antara J dan J_p adalah

$$J_p = J + Md$$

Dari Gambar A.2, titik pusat massa sistem dapat dirumuskan sebagai berikut:

$$M_n a = M_c b$$

a+b=L/2

Oleh karena itu, dari (A.14) dan (A.15), diperoleh momen inersia sistem terhadap pusat massa, yaitu

$$J = M_p a^2 + M_p b^2 + \frac{M_p}{12} L^2$$
(A.16)

dengan

$$a = \frac{LM_c}{2(M_p + M_c)} \operatorname{dan} b = \frac{LM_p}{2(M_p + M_c)}$$

Jika (A.16) dijabarkan akan didapat bentuk akhir persamaan momen inersia sistem seperti ditunjukkan pada (A.17).

$$J = \frac{L^2 m_p (4m_c + m_p)}{12(m_c + m_p)}$$
(A.17)

Sedangkan jarak pusat massa sistem terhadap sumbu putar adalah

$$l = \frac{m_p L}{2(m_c + m_p)} \tag{A.18}$$

Parameter	Simbol	Nilai
Masa kereta (kg)	m _c	1,12
Massa beban pendulum (kg)	m _{pw}	0,095
Massa batang pendulum (kg)	m _{ps}	0,025
Panjang rel (m)	R_l	1
Panjang batang pendulum (m)	l_p	0,402
Jarakpusat massa batang ke sumbu putar(m)	l_{po}	0,146
Panjang beban (m)	l_c	0,041
Jarak pusat massa beban ke sumbu putar (m)	l _{co}	0,347
Jari-jari beban pendulum (m)	r _c	0,02
Jari-jari batang pendulum (m)	r_p	0,006
Gaya gesek statis(N)	F_s	2,5316
Gaya gesek dinamis (Coulomb)(N)	F_c	2,2813
Gaya control maksimal(N)	<i>u_{max}</i>	17,5
Gaya kontrol minimal (N)	DZ_u	1,3792
Kecepatan minimal kereta(m/detik)	DZ _{cv}	0,008
Kecepatan ayun minimal pendulum (rad/detik)	DZ_{pv}	0,034
Gaya gesek pendulum (kg.m ² /detik)	f_p	0,0001
Momen inersia pendulum (kg.m ²)	J_p	0,0139
Momen inersia sistem(kg.m ²)	J	0,0136
Jarak sumbu rotasi ke pusat massa sistem (m)		0,0168
Kec. kereta awal daerah 5 gaya gesek (m/detik)	x_c	0,3955
Gaya gesek saat kecepatan kereta $x_c(N)$	Y _c	2,3815
Tegangan kontrol maksimal (Volt)	V_m	2,5

 Tabel A.1
 Parameter-parameter Sistem Pendulum-Kereta

A.2. Elemen-elemen Matriks Linearisasi Model Dari (3.1), dapat diperoleh

$$f_{1}(\mathbf{x}) = x_{3}$$

$$f_{2}(\mathbf{x}) = x_{4}$$

$$f_{3}(\mathbf{x}) = \frac{a(-T_{c} - \mu x_{4}^{2} \sin x_{2}) + l \cos x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu l \sin^{2} x_{2}}$$

$$f_{4}(\mathbf{x}) = \frac{l \cos x_{2}(-T_{c} - \mu x_{4}^{2} \sin x_{2}) + \mu g \sin x_{2} - f_{p} x_{4}}{J + \mu l \sin^{2} x_{2}}$$
(A.1)

dan

$$h_{1}(\mathbf{x}, u) = 0$$

$$h_{2}(\mathbf{x}, u) = 0$$

$$h_{3}(\mathbf{x}, u) = \frac{au}{J + \mu' \sin^{2} x_{2}}$$
(A.20)
$$h_{4}(\mathbf{x}, u) = \frac{\mu' \cos x_{2}}{J + \mu' \sin^{2} x_{2}}$$
schingga elemen matriks A diperoleh
$$\frac{\partial f_{1}(\mathbf{x})}{\partial x_{1}} = 0, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{2}} = 0, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{3}} = 1, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{4}} = 0$$

$$\frac{\partial f_{2}(\mathbf{x})}{\partial x_{1}} = 0, \frac{\partial f_{2}(\mathbf{x})}{\partial x_{2}} = 0, \frac{\partial f_{2}(\mathbf{x})}{\partial x_{3}} = 0, \frac{\partial f_{1}(\mathbf{x})}{\partial x_{4}} = 1$$

$$\frac{\partial f_{3}(\mathbf{x})}{\partial x_{2}}$$

$$= \frac{g\mu' \cos^{2} x_{2} - a\mu x_{4}^{2} \cos x_{2} - l \sin x_{2}(\mu g \sin x_{2} - f_{p} x_{4})}{J + \mu' \sin^{2} x_{2}}$$

$$- \frac{2\mu' \cos x_{2} \sin x_{2}(a(-T_{c} - \mu x_{4}^{2} \sin x_{2}))}{(J + \mu' \sin^{2} x_{2})^{2}}$$

$$\frac{\partial f_3(\mathbf{x})}{\partial x_3} = \frac{a}{J + \mu l \sin^2 x_2} \frac{\partial T_c}{\partial x_3}$$

$$\frac{\partial f_3(\mathbf{x})}{\partial x_4} = \frac{-2a\mu v_4 \sin x_2 - f_p l \cos x_2}{J + \mu l \sin^2 x_2}$$

$$\frac{\partial f_4(\mathbf{x})}{\partial x_1} = 0$$

$$\frac{\partial f_4(\mathbf{x})}{\partial x_2}$$

$$= \frac{\mu g \cos x_2 - \mu l x_4^2 \cos^2 x_2 - l \sin x_2 (-T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2}$$

$$= \frac{-2\mu l \cos x_2 \sin x_2 (l \cos x_2 (-T_c - \mu x_4^2 \sin x_2))}{(J + \mu l \sin^2 x_2)^2}$$

$$= \frac{-2\mu l \cos x_2 \sin x_2 (\mu g \sin x_2 - f_p x_4)}{(J + \mu l \sin^2 x_2)^2}$$

$$= \frac{\partial f_4(\mathbf{x})}{\partial x_4} = -\frac{2\mu l x_4 \cos x_2 \sin x_2 - f_p x_4}{J + \mu l \sin^2 x_2}$$
Sedangkan elemen matriks B adalah
$$\frac{\partial h_1(\mathbf{x}, u)}{\partial u} = 0$$

$$\frac{\partial h_3(\mathbf{x}, u)}{\partial u} = \frac{a}{J + \mu l \sin^2 x_2}$$

A.3. Penurunan Persamaan State Kompensator

Penurunan persamaan state kompensator didasari pada transformasi fungsi sinus dengan persamaan berikut:

A sin
$$(2\pi f)$$
 (A.21)
Dalam bentuk transformasi Laplace persamaan (A.21) dapat diubah
menjadi:

$$\frac{A2\pi t}{s^2 + (2\pi t)^2}$$
 (A.22)

Dalam model kompensator, sistem lup tertutup mengikuti model referensi yang dituliskan dalam persamaan:

$$\dot{x}_c(t) = A_c x_c(t) + B_c e(t) \tag{A.23}$$

$$y_c(t) = x_c(t)$$

Untuk membentuk model referensi (A.23) persamaan (A.22) terlebih dahulu diubah menjadi bentuk:

$$\frac{Y_c(s)}{E_f(s)} = \frac{A2\pi^f}{s^2 + (2\pi^f)^2}$$
(A.24)

Dengan $Y_{c}(s) = X_{c}(s)$ maka

$$s^{2}X_{c}(s) + (2\pi f)^{2}X_{c}(s) = A2\pi E_{f}$$
(A.25)

$$\ddot{x}_{c}(t) + (2\pi f)^{2} x_{c}(t) = A2\pi f e_{f}(t)$$

Misalkan:

$$x_{c1}(t) = x_{c}(t) \Rightarrow \dot{x}_{c1}(t) = \dot{x}_{c}(t) = x_{c2}(t)$$

$$x_{c2}(t) = \dot{x}_{c}(t) \Rightarrow \dot{x}_{c2}(t) = \ddot{x}_{c}(t)$$
(A.26)

Persamaan (A.25) subtitusi ke persamaan (A.26) menjadi:

$$\dot{x}_{c2}(t) = -(2\pi f)^2 x_{c1}(t) + A2\pi f e_f(t)$$
(A.27)

Dengan membentuk Matriks Augmentasi persamaan (A.26) dan (A.27) menjadi:

$$\begin{bmatrix} \dot{x}_{c1}(t) \\ \dot{x}_{c2}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -(2\pi^{f})^{2} & 0 \end{bmatrix} \begin{bmatrix} x_{c1}(t) \\ x_{c2}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ A2\pi^{f} \end{bmatrix} e_{f}(t)$$
(A.28)

dengan

$$\mathbf{A}_{c} = \begin{bmatrix} \mathbf{0} & \mathbf{1} \\ -(2\pi f)^{2} & \mathbf{0} \end{bmatrix}, B_{c} = \begin{bmatrix} \mathbf{0} \\ A2\pi f \end{bmatrix}$$

A.4. Penurunan LMI Batasan Input-Output

Pre-multiplying dan *post-multiplying* LMI dengan matriks \mathbf{P}^{-1} akan didapat

$$\mathbf{P}^{-1} \begin{bmatrix} \mathbf{P} - \frac{\beta}{u_{\max}^{2}} \mathbf{K}_{j}^{T} \mathbf{K}_{j} \end{bmatrix} \mathbf{P}^{-1} \ge \mathbf{P}^{-1} \mathbf{0} \mathbf{P}^{-1}$$

$$\mathbf{P}^{-1} - \mathbf{P}^{-1} \mathbf{K}_{j}^{T} \begin{bmatrix} \frac{u_{\max}^{2}}{\beta} \end{bmatrix} \mathbf{K}_{j} \mathbf{P}^{-1} \ge \mathbf{0}$$

$$\mathbf{E} - \mathbf{F}(\mathbf{H})^{-1} \mathbf{G} \ge \mathbf{0}$$
dengan
$$\mathbf{E} = \mathbf{P}^{-1}, \mathbf{F} = \mathbf{G}^{T} = \mathbf{P}^{-1} \mathbf{K}_{j}^{T}, \mathbf{H} = \frac{u_{\max}^{2}}{\beta}$$
(A.29)

(A.30)

dengan Schur Complement, LMI (A.29) ekuilaven dengan

$$\mathbf{L} \ge 0 \Leftrightarrow \begin{cases} \mathbf{H} \ge 0 \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} \ge 0 \end{cases}$$
$$\mathbf{L} = \begin{bmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{bmatrix} \ge 0$$
$$\mathbf{L} = \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{P}^{-1} \mathbf{K}_{j}^{T} \\ \mathbf{K}_{j} \mathbf{P}^{-1} & \frac{u_{\text{max}}^{2}}{\beta} \end{bmatrix}$$

Subtitusi $\mathbf{Q} = \mathbf{P}^{-1} \operatorname{dan} \mathbf{Y}_{j} = \mathbf{K}_{j} \mathbf{P}^{-1} \operatorname{pada} \operatorname{LMI} (\mathbf{A}.30)$ akan didapat

 $\begin{bmatrix} \mathbf{Q} & \mathbf{Y}_{j}^{T} \\ * & \frac{u_{\max}^{2}}{\beta} \end{bmatrix} \ge 0$ $\begin{bmatrix} -\mathbf{Q} & -\mathbf{Y}_{j}^{T} \\ * & -\frac{u_{\max}^{2}}{\beta} \end{bmatrix} < 0$

Pre-multiplying dan *post-multiplying* LMI dengan matriks P^{-1} akan didapat

$$\mathbf{P}^{-1} \left[\mathbf{P} - \frac{\beta}{z_{max}^{2}} \mathbf{C}_{z}^{T} \mathbf{C}_{z} \right] \mathbf{P}^{-1} \ge \mathbf{P}^{-1} \mathbf{O} \mathbf{P}^{-1}$$

$$\mathbf{P}^{-1} - \mathbf{P}^{-1} \mathbf{C}_{z}^{T} \left(\frac{z_{max}^{2}}{\beta} \right) \mathbf{C}_{z} \mathbf{P}^{-1} \ge \mathbf{0}$$
(A.31)
$$\mathbf{E} - \mathbf{F}(\mathbf{H})^{-1} \mathbf{G} \ge \mathbf{0}$$
dengan
$$\mathbf{E} = \mathbf{P}^{-1}, \mathbf{F} = \mathbf{G}^{T} = \mathbf{P}^{-1} \mathbf{C}_{z}^{T}, \mathbf{H} = \frac{z_{max}^{2}}{\beta}$$
dengan Schur Complement, LMI (A.31) ekuilaven dengan
$$\mathbf{L} \ge \mathbf{0} \Leftrightarrow \left\{ \mathbf{H} \ge \mathbf{0} \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} \ge \mathbf{0} \\ \mathbf{L} = \begin{bmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & \mathbf{H} \end{bmatrix} \ge \mathbf{0}$$

$$\mathbf{L} = \begin{bmatrix} \mathbf{P}^{-1} & \mathbf{P}^{-1} \mathbf{C}_{z}^{T} \\ \mathbf{C}_{z} \mathbf{P}^{-1} & \frac{z_{max}^{2}}{\beta} \end{bmatrix}$$
Subtitusi $\mathbf{Q} = \mathbf{P}^{-1} \operatorname{dan} \mathbf{Y}_{z} \mathbf{K}_{z} \mathbf{P}^{-1} \operatorname{pada} \mathrm{LMI} (A.32) \operatorname{akan} \operatorname{didapat}$

$$\begin{bmatrix} \mathbf{Q} & \mathbf{Q} \mathbf{C}_{z}^{T} \\ * & \frac{z_{max}^{2}}{\beta} \end{bmatrix} \ge \mathbf{0}$$

$$\begin{bmatrix} -\mathbf{Q} & -\mathbf{Q} \mathbf{C}_{z}^{T} \\ * & \frac{z_{max}^{2}}{\beta} \end{bmatrix} \le \mathbf{0}$$

A.5. Penurunan LMI Pencarian *Observer Gain* Kesalahan Sensor

Sesuai dengan persamaan (3.60) pertidaksamaan berikut adalah penurunan performansi H_{∞} dapat dituliskan kembali sebagai berikut:

$$\widetilde{\boldsymbol{e}}_{as}^{T}(t)(\widetilde{\mathbf{A}}_{s}^{T}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{P}}+\widetilde{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(\boldsymbol{p},\boldsymbol{p}))\widetilde{\boldsymbol{e}}_{as}(t)+\widetilde{\boldsymbol{e}}_{as}^{T}\widetilde{\mathbf{P}}\widetilde{\mathbf{N}}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{z}}$$

$$+\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{N}}^{T}(\boldsymbol{p},\boldsymbol{p})\widetilde{\mathbf{P}}\widetilde{\boldsymbol{e}}_{as}(t)+\frac{1}{\gamma}\widetilde{\boldsymbol{e}}_{as}^{T}\overline{\mathbf{C}}_{\boldsymbol{p}}^{T}\overline{\mathbf{C}}_{\boldsymbol{p}}\widetilde{\boldsymbol{e}}_{as}-\gamma\widetilde{\mathbf{z}}^{T}\widetilde{\mathbf{z}}<0$$
(A.33)

Pertidaksamaan (A.33) dapat diubah menjadi bentuk matriks yaitu dapat dituliskan sebagai berikut:

$$\begin{bmatrix} \widetilde{\boldsymbol{e}}_{as}^{T} & \widetilde{\boldsymbol{z}}^{T} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(\boldsymbol{p},\boldsymbol{p}) \overline{\mathbf{P}} + \overline{\mathbf{P}} \widetilde{\mathbf{A}}_{s}(\boldsymbol{p},\boldsymbol{p}) + \frac{1}{\gamma} \overline{\mathbf{C}}_{p}^{T} \overline{\mathbf{C}}_{p} & \overline{\mathbf{P}} \widetilde{\mathbf{N}}(\boldsymbol{p},\boldsymbol{p}) \begin{bmatrix} \widetilde{\boldsymbol{e}}_{as} \\ \widetilde{\boldsymbol{z}} \end{bmatrix} < 0 \\ * & -\gamma I \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{e}}_{as} \end{bmatrix} < 0$$
(A.34)

Secara implisit persamaan (A.34) dapat ditulis sebagai berikut:

$$\left. \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) + \frac{1}{\gamma}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p} - \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) \right| < 0$$
(A.35)

Persamaan (A.35) dapat dipisah menjadi dua bagian menjadi berikut:

$$\begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) & \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) \\ * & -\gamma \mathbf{I} \end{bmatrix} + \begin{bmatrix} \frac{1}{\gamma}\overline{\mathbf{C}}_{p}^{T}\overline{\mathbf{C}}_{p} & 0 \\ 0 & 0 \end{bmatrix} < 0 \quad (A.36)$$

Bagian kedua pada (A.36) dipisah menjadi

Dengan menggunakan teori Schur Complement dengan variabel berikut:

$$\mathbf{E} = \begin{bmatrix} \widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}} + \overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) & \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) \\ * & -\gamma\mathbf{I} \end{bmatrix}$$
$$\mathbf{G} = \mathbf{F}^{T} = \begin{bmatrix} \overline{\mathbf{C}}_{p}^{T} & \mathbf{0} \end{bmatrix}$$
$$\mathbf{H} = \begin{bmatrix} \gamma & \mathbf{0} \\ \mathbf{0} & \gamma \end{bmatrix}, \mathbf{H}^{-1} = \begin{bmatrix} \gamma^{-1} & \mathbf{0} \\ \mathbf{0} & \gamma^{-1} \end{bmatrix}$$

Sehingga dalam bentuk Schur Complement (A.38) dituliskan sebagai berikut:

(A.38)

 $\mathbf{E} - \mathbf{F}\mathbf{H}^{-1}\mathbf{G} < \mathbf{0}$

LMI (A.38) memiliki bentuk yang sama dengan LMI pada (A.39) sehingga LMI (A.38) ekuivalen dengan:

$$\mathbf{L} < \mathbf{0} \Leftrightarrow \begin{cases} \mathbf{H} < \mathbf{0} \\ \mathbf{E} - \mathbf{F} \mathbf{H}^{-1} \mathbf{G} < \mathbf{0} \\ \mathbf{L} = \begin{bmatrix} \mathbf{E} & \vdots & \mathbf{F} \\ \cdots & \cdots & \cdots \\ \mathbf{G} & \vdots & \mathbf{H} \end{bmatrix}$$

Sehingga bentuk pertidaksamaan (A.38) menjadi:

$$\widetilde{\mathbf{A}}_{s}^{T}(p,p)\overline{\mathbf{P}}+\overline{\mathbf{P}}\widetilde{\mathbf{A}}_{s}(p,p) \quad \overline{\mathbf{P}}\widetilde{\mathbf{N}}(p,p) \quad \overline{\mathbf{C}}_{p1}^{T} \quad 0$$

$$= \gamma \mathbf{I} \quad 0 \quad \overline{\mathbf{C}}_{p2}^{T}$$

$$= \gamma \mathbf{I} \quad 0 \quad \mathbf{I} \quad$$

Pertidaksamaan (A.40) adalah pertidaksamaan bentuk Schur Complement dari pertidaksamaan (A.35). Dalam pertidaksamaan (A.40) masih terdapat dua variabel yang belum tersubtitusi yaitu:

 $\widetilde{\mathbf{A}}_{s}^{T}(p,p), \overline{\mathbf{P}}, \widetilde{\mathbf{N}}(p,p)$

dengan mensubtitusikan ketiga variabel ini ke pertidaksamaan (A.40) yaitu:

$$\widetilde{\mathbf{A}}_{s}(p,p) = \begin{bmatrix} \overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} & \overline{\mathbf{D}}_{f} \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}(\overline{\mathbf{A}}(p) - \overline{\mathbf{L}}(p)\overline{\mathbf{C}}_{c} + \mathbf{I}) & -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{D}}_{f} \end{bmatrix}$$
$$\widetilde{\mathbf{N}}(p,p) = \begin{bmatrix} \overline{\mathbf{B}}(p) & 0 \\ -\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c}\overline{\mathbf{B}}(p) & \mathbf{I} \end{bmatrix},$$
$$\overline{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{1} & 0 \\ 0 & \mathbf{I} \end{bmatrix},$$

Dan variabel *premultipling*, untuk menghindari dua variabel dalam satu pertidaksamaan yaitu:

 $\overline{\mathbf{H}}(p) = \mathbf{P}_{1}\overline{\mathbf{L}}(p),$ $\overline{\mathbf{F}}(p)\overline{\mathbf{C}}_{c} = \overline{\mathbf{D}}_{f}\mathbf{P}_{1}$

Didapatkan pertidaksamaan dalam bentuk LMI (A.41) berikut:

 \mathbf{C}_{p1}^{T} 0 0 W13 W11 W_{12} \mathbf{C}_{p2}^{T} 0 W23 I W22 0 -1 0 0 0 < 0 0 -11 -11 0 - 1⁄I $\mathbf{w}_{11} = \mathbf{P}_1 \overline{\mathbf{A}}(p) + (\mathbf{P}_1 \overline{\mathbf{A}}(p))^T - \overline{\mathbf{H}}(p) \overline{\mathbf{C}}_c - (\overline{\mathbf{H}}(p) \mathbf{C}_c)^T,$ $w_{12} = -(\overline{\mathbf{A}}^T(p)\mathbf{P}_1\mathbf{D}_f - \overline{\mathbf{C}}_c^T\overline{\mathbf{H}}^T(p)\overline{\mathbf{D}}_f),$ $w_{13} = \mathbf{P}_1 \overline{\mathbf{B}}(p),$ $w_{22} = -2\overline{\mathbf{D}}_f^T \mathbf{P}_1 \overline{\mathbf{D}}_f$ $w_{23} = -2\overline{\mathbf{D}}_f^T \mathbf{P}_1 \overline{\mathbf{B}}(p),$

(A.41)

Matriks kestabilan Lyapunov pada LMI (A.41) menjamin kestabilan observer untuk kesalahan sensor dan memenuhi performa H_{∞} pada pertidaksamaan (A.33). Untuk mencari *observer gain* sensor dapat dilanjutkan dengan pertidaksamaan (3.64) pada Subbab 4.5.



LAMPIRAN B

```
B.1 Program untuk Penurunan Gain State Feeback
%% PLANT INVERTED PENDULUM LINIERISASI
%Model plant : x3dot = (a*(u(5) -
miu*(u(4)^2)*sin(u(2)))+1*cos(u(2))*(miu*g*sin(u(2))
fp*u(4)))/(J+miu*l*((sin(u(2)))^2))
[A1,B1,Cp1,D1]=linmod('pend',[0 0 0 0]',0) %Linearisasi pada
titik 0 rad
[A2,B2,Cp2,D2]=linmod('pend',[0 0.25 0 0]',0) %Linearisasi
pada titik 0.2612 rad
[A3,B3,Cp3,D3]=linmod('pend',[0 0.5232 0 0]',0) %Linearisasi
pada titik 0.5232 rad
C1 = [1 0 0 0;0 1 0 0;0 0 1 0]; %Keluaran untuk Observer
C2 = C1;
C3 = C1;
C 1 = [1 0 0 0]; %Keluaran untuk Kontroler
C 2 = C 1;
C_{3} = C_{2};
D1 = 0; %Disturbance
D2 = D1;
D3 = D1;
SPENCARIAN GAIN KONTROLER NOMINAL
%% Kasus Tracking
%Matriks Kompensator
Ac = [0 1; -0.3948 0];
Bc = [0; 0.0628];
%Mencari matriks augmentasi tracking
Albar = [A1 \operatorname{zeros}(4,2); -Bc*C 1 Ac];
A2bar = [A2 zeros(4,2); -Bc*C 2 Ac];
A3bar = [A3 zeros(4, 2); -Bc*C 3 Ac];
%Mencari matriks B augmentasi tracking
B1bar = [B1;0;0];
B2bar = [B2;0;0];
B3bar = [B3;0;0];
%%PENCARIAN PDC POLE PLACEMENT DENGAN BATASAN INPUT-OUTPUT
sint = sin((1/12)*3.14); %Theta
cost = cos((1/12)*3.14); %Theta
Cz1 = [1 0 0 0 0 0]; %Batasan Output
bet = 5; %Beta
```

```
umax = 17; %u maksimal
zmax = 0.4;
%Pole Placement LMI1
setlmis([])
Q1bar = lmivar(1, [6 1]);
Y1bar = 1mivar(2, [1 6]);
gamli = 2;
lmiterm([1 1 1 Qlbar], Albar, 1, 's');
lmiterm([1 1 1 Ylbar],Blbar,1,'s');
lmiterm([1 1 1 Qlbar], 2*gamli, 1);
lmiterm([2 1 1 Qlbar], sint*Albar, 1, 's');
lmiterm([2 1 1 Y1bar], sint*B1bar, 1, 's');
lmiterm([2 1 2 Qlbar],cost*Albar,1);
lmiterm([2 1 2 Qlbar], -cost, Albar');
lmiterm([2 1 2 Y1bar], cost*B1bar, 1);
lmiterm([2 1 2 -Y1bar],-cost,B1bar');
lmiterm([2 2 2 Qlbar], sint*Albar, 1, 's');
lmiterm([2 2 2 Ylbar], sint*Blbar, 1, 's');
lmiterm([-3 1 1 Qlbar],1,1);
lmiterm([-4 1 1 Qlbar],1,1);
lmiterm([-4 2 1 Y1bar],1,1);
lmiterm([-4 2 2 0], (umax^2)/bet);
lmiterm([-5 1 1 Qlbar],1,1);
lmiterm([-5 1 2 Qlbar],1,Cz1');
lmiterm([-5 2 2 0], (zmax^2)/bet);
lmisa=getlmis;
[tmin, xfeas] = feasp(lmisa);
Q 1 = dec2mat(lmisa, xfeas, Qlbar);
Y 1 = dec2mat(lmisa, xfeas, Ylbar);
K1bar = Y 1 * inv(Q 1)
%Pole Placement LMI2
setlmis([])
Q2bar = lmivar(1, [6 1]);
Y2bar = lmivar(2,[1 6]);
qam2i = 2
lmiterm([1 1 1 Q2bar], A2bar, 1, 's');
lmiterm([1 1 1 Y2bar], B2bar, 1, 's');
lmiterm([1 1 1 Q2bar], 2*gam2i, 1);
lmiterm([2 1 1 Q2bar], sint*A2bar, 1, 's');
```

lmiterm([2 1 1 Y2bar], sint*B2bar, 1, 's');
```
lmiterm([2 1 2 Q2bar],cost*A2bar,1);
lmiterm([2 1 2 Q2bar],-cost,A2bar');
lmiterm([2 1 2 Y2bar],cost*B2bar,1);
lmiterm([2 1 2 -Y2bar],-cost,B2bar');
lmiterm([2 2 2 Q2bar],sint*A2bar,1,'s');
lmiterm([2 2 2 Y2bar],sint*B2bar,1,'s');
```

```
lmiterm([-3 1 1 Q2bar],1,1);
```

```
lmiterm([-4 1 1 Q2bar],1,1);
lmiterm([-4 2 1 Y2bar],1,1);
lmiterm([-4 2 2 0],(umax<sup>2</sup>)/bet);
```

```
lmiterm([-5 1 1 Q2bar],1,1);
lmiterm([-5 1 2 Q2bar],1,Cz1');
lmiterm([-5 2 2 0],(zmax^2)/bet);
```

```
lmisb=getlmis;
```

```
[tmin,xfeas]=feasp(lmisb);
Q_2 = dec2mat(lmisb,xfeas,Q2bar);
Y_2 = dec2mat(lmisb,xfeas,Y2bar);
K2bar = Y_2*inv(Q_2)
```

```
%Pole Placement LMI3
```

```
setlmis([])
Q3bar = lmivar(1,[6 1]);
Y3bar = lmivar(2,[1 6]);
gam3i = 2
```

```
lmiterm([1 1 1 Q3bar],A3bar,1,'s');
lmiterm([1 1 1 Y3bar],B3bar,1,'s');
lmiterm([1 1 1 Q3bar],2*gam3i,1);
```

```
Imiterm([2 1 1 Q3bar],sint*A3bar,1,'s');
Imiterm([2 1 1 Y3bar],sint*B3bar,1,'s');
Imiterm([2 1 2 Q3bar],cost*A3bar,1);
Imiterm([2 1 2 Q3bar],-cost,A3bar');
Imiterm([2 1 2 Y3bar],-cost,A3bar');
Imiterm([2 1 2 -Y3bar],-cost,B3bar');
Imiterm([2 2 2 Q3bar],sint*A3bar,1,'s');
Imiterm([2 2 2 Y3bar],sint*B3bar,1,'s');
```

lmiterm([-3 1 1 Q3bar],1,1);

```
lmiterm([-4/1 1 Q3bar],1,1);
lmiterm([-4/2 1 Y3bar],1,1);
lmiterm([-4/2 2 0],(umax<sup>2</sup>)/bet);
```

```
lmiterm([-5 1 1 Q3bar],1,1);
lmiterm([-5 1 2 Q3bar],1,Cz1');
```

lmiterm([-5 2 2 0], (zmax^2)/bet);

```
lmisc=getlmis;
[tmin,xfeas]=feasp(lmisc);
Q_3 = dec2mat(lmisc,xfeas,Q3bar);
Y_3 = dec2mat(lmisc,xfeas,Y3bar);
K3bar = Y_3*inv(Q_3)
```

B.2 Program untuk Penurunan Observer gain

```
%PENCARIAN OBSERVER GAIN ESTIMASI SENSOR
%%Matriks pendulum saat teta atau x1 =0, x2 = 0 derajat, x3
= 0, dan x4 = 0
Df1 = [1;1;1]; %Matriks kesalahan sensor
```

%Matrik Augmentasi

```
n1 = size(A1,1); %Jumlah state %x
m1 = size(B1,2); %Jumlah input %u
l1 = size(C1,1); %Jumlah output %state As
h_1 = 1; %Aktuator fault
q1 = 1; %Sensor Fault
```

%Filter LPF

As1 = eye(11); rho = 0.1; thes = sin((1/4)*3.14); thec = cos((1/4)*3.14);

%Matriks Augmentasi

```
Abar1 = [A1 zeros(n1,l1); As1*C1 -As1];
Bbar1 = [B1; zeros(l1,m1)];
Dfbar1 = [zeros(n1,size(Df1,2));As1*Df1];
Cbar1 = [zeros(l1,n1) eye(l1)];
```

%LMI Fuzzy 1

```
nbar1=size(Abar1,1);
Cpl_1=0.1*eye(nbar1);
Cp2_1=0.1*eye(g1);
CP = 0.1*eye(nbar1+g1);
```

```
setlmis([])
gam1 = lmivar(1,[1 1]);
miu1 = lmivar(1,[1 1]);
P1 = lmivar(1,[nbar1 1]);
F1 = lmivar(2,[g1 l1]);
H1 = lmivar(2,[n1+11 11]);
```

```
lmiterm([1 1 1 P1],1,Abar1,'s'); %P1*Abar
lmiterm([1 1 1 H1],-1,Cbar1,'s'); %-H*Cbar1
lmiterm([1 1 2 P1],-Abar1',Dfbar1); %-Abar1*P1*Dfbar1
```

```
lmiterm([1 1 2 -H1],Cbar1',Dfbar1); %Cbar1'*H*Dfbar1
lmiterm([1 1 3 P1],1,Bbar1); %P1*Bbar1
lmiterm([1 1 5 0],Cp1 1'); %Cp1'
lmiterm([1 2 2 P1],-2*Dfbar1',Dfbar1); %-2*Dfbar1*P1*Dfbar1
lmiterm([1 2 3 P1],-Dfbar1',Bbar1); %-Dfbar1*P1*Bbar1
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cp2 1') %Cp2'
lmiterm([1 3 3 gam1,-1,1) %-gam*I
lmiterm([1 4 4 gam1],-1,1) %-gam*I
lmiterm([1 5 5 gam1],-1,1) %-gam*I
```

```
lmiterm([-2 1 1 miu1],1,1) %miu*I
lmiterm([-2 1 2 P1],Dfbar1',1) %Dfbar1'*P1
lmiterm([-2 1 2 F1],-1,Cbar1) %F*Cbar1
lmiterm([-2 2 2 miu1],1,1) %miu*I
```

```
lmiterm([-3 1 1 P1],1,1); %P1 > 0
lmiterm([-3 2 2 0],1);
```

```
lmiterm([4 1 1 P1],1,Abar1,'s');
lmiterm([4 1 1 H1],-1,Cbar1,'s');
lmiterm([4 1 1 P1],2*rho,1);
lmiterm([4 1 2 P1],-Abar1',Dfbar1);
lmiterm([4 1 2 -H1],Cbar1',Dfbar1);
lmiterm([4 2 2 P1],-Dfbar1',Dfbar1,'s');
lmiterm([4 2 2 0],2*rho);
```

```
lmiterm([5 1 1 P1],thes,Abar1,'s');
lmiterm([5 1 1 H1], -thes, Cbar1, 's');
lmiterm([5 1 2 P1], -thes*Abar1', Dfbar1);
lmiterm([5 1 2 -H1], thes*Cbar1', Dfbar1);
lmiterm([5 2 2 P1],-thes*Dfbar1',Dfbar1,'s');
lmiterm([5 1 3 P1], thec, Abar1, 's');
lmiterm([5 1 3 H1],-thec,Cbar1,'s');
lmiterm([5 1 4 P1], -thec*Abar1', Dfbar1);
lmiterm([5 1 4 -H1], thec*Cbar1', Dfbar1);
lmiterm([5 2 3 -P1], thec*Dfbar1', Abar1);
lmiterm([5 2 3 H1],thec*Dfbar1',Cbar1);
lmiterm([5 2 4 P1], -thec*Dfbar1', Dfbar1, 's');
lmiterm([5 3 3 P1], thes, Abar1, 's');
lmiterm([5 3 3 H1], -thes, Cbar1, 's');
lmiterm([5 3 4 P1], -thes*Abar1', Dfbar1);
lmiterm([5 3 4 -H1], thes*Cbar1', Dfbar1);
lmiterm([5 4 4 P1], -thes*Dfbar1', Dfbar1, 's');
```

Solusi LMI Optimal 1

```
lmis1=getlmis;
[tmin,xfeas]=feasp(lmis1);
```

```
%Fungsi minimize (theta+gama)
q_1=decnbr(lmis1);
c_1=zeros(q_1,1);
for j=1:q_1
Gamlj=defcx(lmis1,j,gam1);
Miulj=defcx(lmis1,j,miul);
c_1(j)=trace(Gamlj)+trace(Miulj);
end
```

```
%Observer gain
```

```
[opt,xopt]=mincx(lmis1,c_1,[100 0 0 0 0]);
Gam10)= dec2mat(lmis1,xopt,gam1) %Gama Optimal
Miu10=dec2mat(lmis1,xopt,miu1);
p10 = dec2mat(lmis1,xopt,P1);
h10 = dec2mat(lmis1,xopt,H1);
Lbar10 = inv(p10)*h10 %Lbar Optimal
f10 = dec2mat(lmis1,xopt,F1); %F Optimal
```

%% matriks pendulum saat teta atau x1 =0, x2 = 15 derajat, x3 = 0, dan x4 = 0 Df2 = Df1;

```
%Matrik Augmentasi
n2 = size(A2,1); %Jumlah state %x
m2 = size(B2,2); %Jumlah input %u
l2 = size(C2,1); %Jumlah output %state As
h_2 = 1; %Aktuator fault
q2 = 1; %Sensor Fault
```

```
%Filter LPF
As2 = eye(12);
```

```
%Matriks Augmentasi
Abar2 = [A2 zeros(n2,12); As2*C2 -As2];
Bbar2 = [B2; zeros(12,m2)];
Dfbar2 = [zeros(n2,size(Df2,2));As2*Df2];
Cbar2 = [zeros(12,n2) eye(12)];
```

```
%LMI Fuzzy_2
```

```
nbar2=size(Abar2,1);
Cp1_2=0.1*eye(nbar2);
Cp2_2=0.1*eye(g2);
```

```
setlmis([])
gam2 = lmivar(1,[1 1]);
miu2 = lmivar(1,[1 1]);
P2 = lmivar(1,[nbar2 1]);
F2 = lmivar(2,[g2 12]);
H2 = lmivar(2,[n2+12 12]);
```

```
lmiterm([1 1 1 P2],1,Abar2,'s'); %P2*Abar
lmiterm([1 1 1 H2],-1,Cbar2,'s'); %-H*Cbar2
lmiterm([1 1 2 P2], -Abar2', Dfbar2); %-Abar2*P2*Dfbar2
lmiterm([1 2 1 H2], Dfbar2', Cbar2); %Cbar2'*H*Dfbar2
lmiterm([1 1 3 P2],1,Bbar2); %P2*Bbar2
lmiterm([1 1 5 0],Cp1 2'); %Cp2'
lmiterm([1 2 2 P2],-2*Dfbar2',Dfbar2); %-2*Dfbar2*P2*Dfbar2
lmiterm([1 2 3 P2],-Dfbar2',Bbar2); %-Dfbar2*P2*Bbar2
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cp2 2') %Cp2'
lmiterm([1 3 3 gam2],-1,1) %-gam*I
lmiterm([1 4 4 gam2],-1,1) %-gam*I
lmiterm([1 5 5 gam2],-1,1) %-gam*I
lmiterm([1 6 6 gam2],-1,1) %-gam*I
lmiterm([-2 1 1 miu2],1,1) %miu*I
lmiterm([-2 1 2 P2], Dfbar2', 1) %Dfbar2'*P2
lmiterm([-2 1 2 F2], -1, Cbar2) %F*Cbar2
lmiterm([-2 2 2 miu2],1,1) %miu*I
lmiterm([-3 1 1 P2],1,1);
lmiterm([-3 2 2 0],1);
lmiterm([4 1 1 P2],1,Abar2,'s');
lmiterm([4 1 1 H2], -1, Cbar2, 's');
lmiterm([4 1 1 P2], 2*rho, 1);
lmiterm([4 1 2 P2], -Abar2', Dfbar2);
lmiterm([4 1 2 -H2], Cbar2', Dfbar2);
lmiterm([4 2 2 P2], -Dfbar2', Dfbar2, 's');
lmiterm([4 2 2 0],2*rho);
lmiterm([5 1 1 P2], thes, Abar2, 's');
lmiterm([5 1 1 H2], -thes, Cbar2, 's');
lmiterm([5 1 2 P2], -thes*Abar2', Dfbar2);
lmiterm([5 1 2 -H2],thes*Cbar2',Dfbar2);
lmiterm([5 2 2 P2],-thes*Dfbar2',Dfbar2,'s');
lmiterm([5 1 3 P2],thec,Abar2,'s');
lmiterm([5 1 3 H2], -thec, Cbar2, 's');
lmiterm([5 1 4 P2], -thec*Abar2', Dfbar2);
lmiterm([5 1 4 -H2], thec*Cbar2', Dfbar2);
lmiterm([5 2 3 -P2], thec*Dfbar2', Abar2);
lmiterm([5 2 3 H2],thec*Dfbar2',Cbar2);
lmiterm([5 2 4 P2],-thec*Dfbar2',Dfbar2,'s');
lmiterm([5 3 3 P2],thes,Abar2,'s');
lmiterm([5 3 3 H2], -thes, Cbar2, 's');
lmiterm([5 3 4 P2], -thes*Abar2', Dfbar2);
lmiterm([5 3 4 -H2], thes*Cbar2', Dfbar2);
lmiterm([5 4 4 P2], -thes*Dfbar2', Dfbar2, 's');
```

```
%Solusi LMI Optimal 2
```

```
lmis2=getlmis;
[tmin,xfeas]=feasp(lmis2);
```

```
%Fungsi minimize (theta + gama)
q_2=decnbr(lmis2);
c_2=zeros(q_2,1);
for j=1:q_2
Gam2j=defcx(lmis2,j,gam2);
Miu2j=defcx(lmis2,j,miu2);
c_2(j)=trace(Gam2j)+trace(Miu2j);
end
```

%Observer gain

[opt,xopt]=mincx(lmis2,c_2,[100 0 0 0 0]); Gam2O = dec2mat(lmis2,xopt,gam2) %Gama Optimal 1 Miu2O=dec2mat(lmis2,xopt,miu2); p2O = dec2mat(lmis2,xopt,P2); h2O = dec2mat(lmis2,xopt,P2); Lbar2O=inv(p2O)*h2O %Lbar Optimal f2O=dec2mat(lmis2,xopt,F2) %F Optimal

%% matriks pendulum saat teta atau x1 =0, x2 = 30 derajat,x3 = 0, dan x4 = 0 Df3 = Df2;

%Matrik Augmentasi

n3 = size(A3,1); %Jumlah state %x
m3 = size(B3,2); %Jumlah input %u
13 = size(C3,1); %Jumlah output %state As
h_3 = 1; %Aktuator fault
g3 = 1; %Sensor Fault

%Filter

As3 = eye(13);

%Matriks Augmentasi Abar3 = [A3 zeros(n3,13); As3*C3 -As3];

```
Bbar3 = [B3; zeros(13,m3)];
Dfbar3 = [zeros(n3,size(Df3,2));As3*Df3];
Cbar3 = [zeros(13,n3) eye(13)];
```

```
%LMI Fuzzy 3
```

nbar3=size(Abar3,1); Cp1_3=0.1*eye(nbar3); Cp2_3=0.1*eye(g3);

```
setlmis([])
gam3 = lmivar(1,[1 1]);
miu3 = lmivar(1,[1 1]);
P3 = lmivar(1,[nbar3 1]);
```

```
F3 = lmivar(2, [q3 13]);
H3 = lmivar(2, [n3+13 13]);
lmiterm([1 1 1 P3],1,Abar3,'s'); %P3*Abar
lmiterm([1 1 1 H3],-1,Cbar3,'s'); %-H*Cbar3
lmiterm([1 1 2 P3],-Abar3',Dfbar3); %-Abar3*P3*Dfbar3
lmiterm([1 2 1 H3], Dfbar3', Cbar3); %Cbar3'*H*Dfbar3
lmiterm([1 1 3 P3],1,Bbar3); %P3*Bbar3
lmiterm([1 1 5 0],Cp1 3'); %Cp3'
lmiterm([1 2 2 P3],-2*Dfbar3',Dfbar3); %-2*Dfbar3*P3*Dfbar3
lmiterm([1 2 3 P3], -Dfbar3', Bbar3); %-Dfbar3*P3*Bbar3
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cp2 3') %Cp3'
lmiterm([1 3 3 gam3],-1,1) %-gam*I
lmiterm([1 4 4 gam3],-1,1) %-gam*I
lmiterm([1 5 5 gam3],-1,1) %-gam*I
lmiterm([1 6 6 gam3],-1,1) %-gam*I
lmiterm([-2 1 1 miu3],1,1) %miu*I
lmiterm([-2 1 2 P3], Dfbar3', 1) %Dfbar3'*P3
lmiterm([-2 1 2 F3], -1, Cbar3) %F*Cbar3
lmiterm([-2 2 2 miu3],1,1) %miu*I
lmiterm([-3 1 1 P3],1,1);
lmiterm([-3 2 2 0],1);
lmiterm([4 1 1 P3],1,Abar3,'s');
lmiterm([4 1 1 H3], -1, Cbar3, 's');
lmiterm([4 1 1 P3], 2*rho, 1);
lmiterm([4 1 2 P3], -Abar3', Dfbar3);
lmiterm([4 1 2 -H3], Cbar3', Dfbar3);
lmiterm([4 2 2 P3], -Dfbar3', Dfbar3, 's');
lmiterm([4 2 2 0],2*rho);
lmiterm([5 1 1 P3],thes,Abar3,'s');
lmiterm([5 1 1 H3], -thes, Cbar3, 's');
lmiterm([5 1 2 P3],-thes*Abar3',Dfbar3);
lmiterm([5 1 2 -H3], thes*Cbar3', Dfbar3);
lmiterm([5 2 2 P3], -thes*Dfbar3', Dfbar3,
lmiterm([5 1 3 P3],thec,Abar3,'s');
lmiterm([5 1 3 H3], -thec, Cbar3, 's');
lmiterm([5 1 4 P3], -thec*Abar3', Dfbar3);
lmiterm([5 1 4 -H3], thec*Cbar3', Dfbar3);
lmiterm([5 2 3 -P3], thec*Dfbar3', Abar3);
lmiterm([5 2 3 H3],thec*Dfbar3',Cbar3);
lmiterm([5 2 4 P3], -thec*Dfbar3', Dfbar3, '
lmiterm([5 3 3 P3],thes,Abar3,'s');
lmiterm([5 3 3 H3], -thes, Cbar3, 's');
lmiterm([5 3 4 P3], -thes*Abar3', Dfbar3);
lmiterm([5 3 4 -H3], thes*Cbar3', Dfbar3);
```

```
lmiterm([5 4 4 P3],-thes*Dfbar3',Dfbar3,'s');
```

```
%Solusi LMI Optimal_3
lmis3=getlmis;
[tmin,xfeas]=feasp(lmis3);
```

```
%Fungsi minimize (theta+gama)
q_3=decnbr(lmis3);
c_3=zeros(q_3,1);
for j=1:q_3
Gam3j=defcx(lmis3,j,gam3);
Miu3j=defcx(lmis3,j,miu3);
c_3(j)=trace(Gam3j)+trace(Miu3j);
end
```

```
%Observer gain
```

```
[opt,xopt]=mincx(lmis3,c_3,[100 0 0 0 0]);
Gam30 = dec2mat(lmis3,xopt,gam3); %Gama Optimal
Miu30=dec2mat(lmis3,xopt,miu3);
p30 = dec2mat(lmis3,xopt,P3);
h30 = dec2mat(lmis3,xopt,H3);
Lbar30=inv(p30)*h30; %L Optimal
f30=dec2mat(lmis3,xopt,F3); %F Optimal
```

```
%% PENCARIAN OBSERVER GAIN AKTUATOR
%LMI Aktuator 1
Cpal 1=0.5*eye(n1);
Cpal 2=0.5*eye(g1);
```

```
CPA = 0.5*eye(n1+g1);
ro = 0.5
tes = sin((1/4)*3.14);
tec = cos((1/4)*3.14);
```

```
setlmis([])
gama1 = lmivar(1,[1 1]);
miu_a1 = lmivar(1,[1 1]);
Pla = lmivar(1,[n1 1]);
F1a = lmivar(2,[h_1 11]);
H1a = lmivar(2,[n1 11]);
```

```
lmiterm([1 1 1 P1a],1,A1,'s'); %P1a*A1
lmiterm([1 1 1 H1a],-1,C1,'s'); %-H*C21
lmiterm([1 1 2 P1a],-A1',B1); %-A1'*P1a*B1
lmiterm([1 2 1 H1a],B1',C1); %C1'*H1'*B1
lmiterm([1 1 3 H1a],-1,Df1); %-H1*Df1
lmiterm([1 1 5 0],Cpa1_1'); %Cpa1'
lmiterm([1 2 2 P1a],-2*B1',B1); %-B1'*P1a*B1
lmiterm([1 2 3 H1a],-B1',Df1); %-B1'*H1*Df1
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cpa1 2') %Cpa2'
```

```
lmiterm([1 3 3 gama1],-1,1) %-gama*I
lmiterm([1 4 4 gama1],-1,1) %-gama*I
lmiterm([1 5 5 gama1],-1,1) %-gama*I
lmiterm([1 6 6 gama1],-1,1) %-gama*I
```

Imiterm([-2 1 1 miu_al],1,1) %miu*I Imiterm([-2 1 2 P1a],B1',1) %B1'*P1a Imiterm([-2 1 2 F1a],-1,C1) %F1a*C1 Imiterm([-2 2 2 miu_al],1,1) %miu*I

lmiterm([-3 1 1 P1a],1,1); %P1a>0

lmiterm([4 1 1 P1a],1,A1,'s'); lmiterm([4 1 1 H1a],-1,C1,'s'); lmiterm([4 1 1 P1a],2*ro,1); lmiterm([4 1 2 P1a],-A1',B1); lmiterm([4 1 2 -H1a],C1',B1); lmiterm([4 2 2 P1a],-B1',B1,'s'); lmiterm([4 2 2 0],2*ro);

```
lmiterm([5 1 1 P1a],tes,A1,'s');
lmiterm([5 1 1 H1a], -tes, C1, 's');
lmiterm([5 1 2 P1a], -tes*A1', B1);
lmiterm([5 1 2 -H1a],tes*C1',B1);
lmiterm([5 2 2 Pla], -tes*B1', B1, 's');
lmiterm([5 1 3 P1a],tec,A1,'s');
lmiterm([5 1 3 H1a], -tec, C1, 's');
lmiterm([5 1 4 P1a], -tec*A1', B1);
lmiterm([5 1 4 -H1a], tec*C1', B1);
lmiterm([5 2 3 -P1a],tec*B1',A1);
lmiterm([5 2 3 H1a],tec*B1',C1);
lmiterm([5 2 4 P1a], -tec*B1', B1, 's');
lmiterm([5 3 3 P1a], tes, A1, 's');
lmiterm([5 3 3 H1a],-tes,C1,'s');
lmiterm([5 3 4 P1a], -tes*A1', B1);
lmiterm([5 3 4 -H1a],tes*C1',B1);
lmiterm([5 4 4 P1a], -tes*B1', B1, 's');
```

```
%Solusi LMI
lmisla=getlmis;
```

[tmin,xfeas]=feasp(lmis1a);

```
%Fungsi minimize(theta+gama)
qla=decnbr(lmisla);
cla=zeros(qla,1);
for j=1:qla
Gam1ja=defcx(lmisla,j,gama1);
Miulja=defcx(lmisla,j,miu_a1);
cla(j)=trace(Gam1ja)+trace(Miulja);
end
```

%Observer gain

```
[opt,xopt]=mincx(lmisla,cla,[100 10 0 0 0]);
GamlOa = dec2mat(lmisla,xopt,gamal) %Gama Optimal
MiulOa=dec2mat(lmisla,xopt,miu_al);
plOa = dec2mat(lmisla,xopt,Pla);
hlOa = dec2mat(lmisla,xopt,Hla);
LbarlOa=inv(plOa)*hlOa %Lbar Optimal
flOa=dec2mat(lmisla,xopt,Fla) %Lbar Optimal
```

%LMI_Aktuator_2 Cpa1_1=0.5*eye(n2); Cpa1_2=0.5*eye(q2);

setlmis([])
gama2 = lmivar(1,[1 1]);
miu_a2 = lmivar(1,[1 1]);
P2a = lmivar(1,[n2 1]);
F2a = lmivar(2,[h_2 12]);
H2a = lmivar(2,[n2 12]);

```
lmiterm([1 1 1 P2a],1,A2,'s'); %P2a*A2
lmiterm([1 1 1 H2a],-1,C2,'s'); %-H*C2
lmiterm([1 1 2 P2a],-A2',B2); %-A2'*P2a*B2
lmiterm([1 2 1 H2a],B2',C2); %C2'*H2'*B2
lmiterm([1 1 3 H2a],-1,Df2); %-H2*Df2
lmiterm([1 1 5 0],Cpa1_1'); %Cpa1'
lmiterm([1 2 2 P2a],-2*B2',B2); %-B2'*P2a*B2
lmiterm([1 2 3 H2a],-B2',Df2); %-B2'*P2a*B2
lmiterm([1 2 3 H2a],-B2',Df2); %-B2'*H2*Df2
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cpa1_2') %Cpa2'
lmiterm([1 3 3 gama2],-1,1) %-gama*I
lmiterm([1 5 5 gama2],-1,1) %-gama*I
lmiterm([1 6 6 gama2],-1,1) %-gama*I
```

```
lmiterm([-2 1 1 miu_a2],1,1) %miu*I
lmiterm([-2 1 2 P2a],B2',1) %B2'*P2a
lmiterm([-2 1 2 F2a],-1,C2) %F2a*C2
lmiterm([-2 2 2 miu a2],1,1) %miu*I
```

lmiterm([-3 1 1 P2a],1,1); %P2a>0

lmiterm([4 1 1 P2a],1,A2,'s'); lmiterm([4 1 1 H2a],-1,C2,'s'); lmiterm([4 1 1 P2a],2*ro,1); lmiterm([4 1 2 P2a],-A2',B2); lmiterm([4 1 2 -H2a],C2',B2); lmiterm([4 2 2 P2a],-B2',B2,'s'); lmiterm([4 2 2 0],2*ro);

```
lmiterm([5 1 1 P2a],tes,A2,'s');
lmiterm([5 1 1 H2a],-tes,C2,'s');
lmiterm([5 1 2 P2a], -tes*A2', B2);
lmiterm([5 1 2 -H2a],tes*C2',B2);
lmiterm([5 2 2 P2a],-tes*B2',B2,'s');
lmiterm([5 1 3 P2a],tec,A2,'s');
lmiterm([5 1 3 H2a],-tec,C2,'s');
lmiterm([5 1 4 P2a], -tec*A2', B2);
lmiterm([5 1 4 -H2a],tec*C2',B2);
lmiterm([5 2 3 -P2a], tec*B2', A2);
lmiterm([5 2 3 H2a],tec*B2',C2);
lmiterm([5 2 4 P2a], -tec*B2', B2, 's');
lmiterm([5 3 3 P2a],tes,A2,'s');
lmiterm([5 3 3 H2a],-tes,C2,'s');
lmiterm([5 3 4 P2a], -tes*A2', B1);
lmiterm([5 3 4 -H2a], tes*C2', B1);
lmiterm([5 4 4 P2a], -tes*B2', B1, 's');
```

```
%Solusi LMI
lmis2a=getlmis;
[tmin,xfeas]=feasp(lmis2a);
```

```
%Fungsi minimize (theta+gama)
q2a=decnbr(lmis2a);
c2a=zeros(q2a,1);
for j=1:q2a
Gam2ja=defcx(lmis2a,j,gama2);
Miu2ja=defcx(lmis2a,j,miu_a2);
c2a(j)=trace(Gam2ja)+trace(Miu2ja);
end
```

```
[opt,xopt]=mincx(lmis2a,c2a,[100 10 0 0 0]);
Gam2Oa = dec2mat(lmis2a,xopt,gama2) %Gama Optimal
Miu2Oa=dec2mat(lmis2a,xopt,miu_a2);
p2Oa = dec2mat(lmis2a,xopt,P2a);
h2Oa = dec2mat(lmis2a,xopt,H2a);
Lbar2Oa=inv(p2Oa)*h2Oa %Lbar Optimal
f2Oa=dec2mat(lmis2a,xopt,F2a) %F Optimal
```

```
%LMI Aktuator 3
Cpa1_1=0.5*eye(n2);
Cpa1 2=0.5*eye(q2);
```

```
setlmis([])
gama3 = lmivar(1,[1 1]);
miu_a3 = lmivar(1,[1 1]);
P3a = lmivar(1,[n3 1]);
F3a = lmivar(2,[h_3 13]);
H3a = lmivar(2,[n3 13]);
```

```
lmiterm([1 1 1 P3a],1,A3,'s'); %P3a*A3
lmiterm([1 1 1 H3a],-1,C3,'s'); %-H*C3
lmiterm([1 1 2 P3a],-A3',B3); %-A3'*P3a*B3
lmiterm([1 2 1 H3a],B3',C3); %C3'*H3'*B3
lmiterm([1 1 3 H3a],-1,Df3); %-H3*Df3
lmiterm([1 1 5 0],Cpa1 1'); %Cpa3'
lmiterm([1 2 2 P3a],-2*B3',B3); %-B3'*P3a*B3
lmiterm([1 2 3 H3a],-B3',Df3); %-B2'*H3*Df3
lmiterm([1 2 4 0],1); %I
lmiterm([1 2 6 0],Cpa1 2') %Cpa2'
lmiterm([1 3 3 gama3],-1,1) %-gama*I
lmiterm([1 4 4 gama3],-1,1) %-gama*I
lmiterm([1 5 5 gama3],-1,1) %-gama*I
lmiterm([1 6 6 gama3],-1,1) %-gama*I
lmiterm([-2 1 1 miu a3],1,1) %miu*I
lmiterm([-2 1 2 P3a],B3',1) %B3'*P3a
lmiterm([-2 1 2 F3a],-1,C3) %F3a*C3
lmiterm([-2 2 2 miu a3],1,1) %miu*I
lmiterm([-3 1 1 P3a],1,1); %P3a>0
lmiterm([4 1 1 P3a],1,A3,'s');
lmiterm([4 1 1 H3a],-1,C3,'s');
lmiterm([4 1 1 P3a],2*ro,1);
lmiterm([4 1 2 P3a],-A3',B3);
lmiterm([4 1 2 -H3a],C3',B3);
lmiterm([4 2 2 P3a], -B3', B3, 's');
lmiterm([4 2 2 0],2*ro);
lmiterm([5 1 1 P3a],tes,A3,'s');
lmiterm([5 1 1 H3a],-tes,C3,'s');
lmiterm([5 1 2 P3a], -tes*A3', B3);
lmiterm([5 1 2 -H3a],tes*C3',B3);
lmiterm([5 2 2 P3a],-tes*B3',B3,'s');
lmiterm([5 1 3 P3a],tec,A3,'s');
lmiterm([5 1 3 H3a],-tec,C3,'s');
lmiterm([5 1 4 P3a],-tec*A3',B3);
lmiterm([5 1 4 -H3a],tec*C3',B3);
lmiterm([5 2 3 -P3a],tec*B3',A3);
lmiterm([5 2 3 H3a],tec*B3',C3);
lmiterm([5 2 4 P3a],-tec*B3',B3,'s');
lmiterm([5 3 3 P3a],tes,A3,'s');
lmiterm([5 3 3 H3a], -tes, C3, 's');
lmiterm([5 3 4 P3a],-tes*A3',B3);
lmiterm([5 3 4 -H3a],tes*C3',B3);
lmiterm([5 4 4 P3a],-tes*B3',B3,'s');
```

%Solusi LMI

```
lmis3a=getlmis;
[tmin,xfeas]=feasp(lmis3a);
```

```
%Fungsi minimize (theta+gama)
g3a=decnbr(lmis3a);
c3a=zeros(q3a,1);
for j=1:q3a
Gam3ja=defcx(lmis3a,j,gama3);
Miu3ja=defcx(lmis3a,j,miu_a3);
c3a(j)=trace(Gam3ja)+trace(Miu3ja);
end
```

[opt,xopt]=mincx(lmis3a,c3a,[100 10 0 0 0]); Gam30a = dec2mat(lmis3a,xopt,gama3) %Gama Optimal Miu30a=dec2mat(lmis3a,xopt,miu_a3); p30a = dec2mat(lmis3a,xopt,P3a); h30a = dec2mat(lmis3a,xopt,H3a); Lbar30a=inv(p30a)*h30a %Lbar Optimal f30a=dec2mat(lmis3a,xopt,F3a) %F Optimal

%%END OF THIS PROGRAM :)

B.3 Blok Diagram Simulink

Blok inverted pendulum



Blok *Numerator x3d*



Blok Diagram Fungsi Bobot





Blok Diagram Fuzzy T-S PPIO Sensor Observer



Blok Diagram Fuzzy T-S Actuator Observer

_____2 u_kompensasi 1 u_error 2 Gain 1 fault 3 Kber Blok Diagram Kompensasi Kesalahan Sensor 2 y_filter Low Pass Filter 2 1 f_topi Gain Gain 4 x_esti 3 y_kom Blok Diagram LPF <u>1</u> s Integrator 1 Add1

Blok Diagram Kompensasi Kesalahan Aktuator