



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR -TE 141599

**KOMPENSASI KESALAHAN AKTUATOR MENGGUNAKAN
PROPORTIONAL MULTIPLE INTEGRAL OBSERVER PADA
STABILISASI SISTEM PENDULUM KERETA**

Mujiyanto Purnomo Saputro
NRP 2211 100 024

Dosen Pembimbing
Dr. Trihastuti Agustinah, ST., MT.

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



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - TE 141599

**ACTUATOR FAULT COMPENSATION USING
PROPORTIONAL MULTIPLE INTEGRAL OBSERVER FOR
STABILIZATION OF THE PENDULUM-CART SYSTEM**

Mujianto Purnomo Saputro
NRP 2211 100 024

Supervisor
Dr. Trihastuti Agustinah, ST., MT.

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

**KOMPENSASI KESALAHAN AKTUATOR
MENGUNAKAN *PROPORTIONAL MULTIPLE
INTEGRAL OBSERVER* PADA STABILISASI
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 :

Dosen Pembimbing



**SURABAYA
JULI, 2015**

KOMPENSASI KESALAHAN AKTUATOR MENGUNAKAN *PROPORTIONAL MULTIPLE INTEGRAL OBSERVER* PADA STABILISASI SISTEM PENDULUM KERETA

Nama : Mujianto Purnomo Saputro
Pembimbing : Dr. Trihastuti Agustinah, ST., MT.

ABSTRAK

Sistem pendulum kereta merupakan *plant* nonlinear yang terdiri dari batang pendulum yang menggantung dan bersumbu pada kedua sisi kereta yang dapat bergerak secara horizontal. Pada penelitian ini, persoalan kontrol yang dibahas adalah stabilisasi sistem pendulum kereta apabila terjadi kesalahan aktuator. *Fault Tolerant Control* (FTC) adalah suatu metode kontrol yang dapat menoleransi kesalahan atau malafungsi komponen saat menjaga stabilitas sistem, sehingga performa sistem dapat terjaga. Model nonlinear sistem pendulum kereta direpresentasikan dengan model *fuzzy* Takagi-Sugeno (T-S) untuk dua titik kerja, kemudian kontroler *state feedback* disusun dengan konsep *Parallel Distributed Compensation* (PDC) menggunakan *Linear Quadratic Regulator* (LQR). Untuk menjaga stabilisasi saat terjadi kesalahan pada aktuator, maka digunakan *Proportional Multiple Integral Observer* untuk mengestimasi kesalahan, kemudian hasil estimasi tersebut digunakan untuk mengkompensasi kesalahan pada aktuator. Hasil simulasi menunjukkan bahwa sistem kontrol pada sistem pendulum kereta mampu melakukan stabilisasi meskipun terjadi kesalahan aktuator dengan menggunakan FTC. Sedangkan hasil implementasi menunjukkan adanya FTC dapat mengurangi penurunan performa respon posisi kereta sebesar 81,25% untuk kesalahan *step* dan 87,5% untuk kesalahan *ramp*.

Kata Kunci : *Fault Tolerant Control*, *fuzzy* Takagi-Sugeno, *Linear Quadratic Regulator*, *Proportional Multiple Integral Observer* Sistem Pendulum Kereta, Stabilisasi.



ACTUATOR FAULT COMPENSATION USING PROPORTIONAL MULTIPLE INTEGRAL OBSERVER FOR STABILIZATION OF THE PENDULUM CART SYSTEM

Name : Mujianto Purnomo Saputro
Supervisor : Dr. Trihastuti Agustinah, ST., MT.

ABSTRACT

Pendulum-cart system is a nonlinear plant consisting of a hanging pendulum rod and the wheelbase on both sides of the train that can move horizontally. In this research, the control problem is stabilization of pendulum-cart system when an actuator fault is occurred. Fault Tolerant Control (FTC) is a control method that can tolerate faults or components malfunction while maintaining the stability of the system until the performance of the system can be maintained. Pendulum-Cart System is represented by Takagi-Sugeno (T-S) fuzzy model for two operating points, and state feedback controller is designed by Parallel Distributed Compensation (PDC) concept using Linear Quadratic Regulator (LQR). Proportional Multiple Integral Observer is used to estimate the fault. The result of that estimation is used to compensate actuator fault. The simulation results show that the control system of pendulum-cart system capable to stabilize despite the actuator fault using FTC. The implementation results show that FTC can decrease until 81.25% for step signal fault and 87.5% for ramp signal fault of the performance degradation of cart position response.

Keywords: *Fault Tolerant Control, Linear Quadratic Regulator, Pendulum-Cart System, , Proportional Multiple Integral Observer, Stabilization, Takagi-Sugeno (T-S) fuzzy.*



KATA PENGANTAR

Segala puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik dan lancar. Tugas Akhir ini disusun untuk memenuhi persyaratan kelulusan pendidikan strata-1 pada Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya yang berjudul:

Kompensasi Kesalahan Aktuator Menggunakan *Proportional Multiple Integral Observer* pada Stabilisasi Sistem Pendulum Kereta

Dalam menyelesaikan pengerjaan Tugas Akhir ini penulis mendapatkan bantuan dari beberapa pihak. Oleh karena itu penulis mengucapkan terima kasih kepada kedua orang tua yang telah memberikan motivasi, dukungan, dan doa yang selalu mengalir setiap hari. Penulis juga mengucapkan terima kasih kepada Ibu Dr. Trihastuti Agustinah, ST., MT., yang memberikan ilmu pengetahuan serta motivasi untuk menyelesaikan Tugas Akhir ini. Selain itu penulis berterima kasih kepada teman-teman seperjuangan Tim FTC yaitu Kendro, Dhino, Dedi, dan Lung beserta teman-teman Bidang Studi Teknik Sistem Pengaturan dan keluarga e-51, serta seluruh warga Jurusan Teknik Elektro yang selalu memberi motivasi penulis.

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.

Surabaya, Juli 2015

Penulis



DAFTAR ISI

HALAMAN JUDUL.....	i
PERNYATAAN KEASLIAN TUGAS AKHIR.....	iii
HALAMAN PENGESAHAN.....	v
ABSTRAK.....	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xix
DAFTAR NOTASI DAN SIMBOL.....	xxi
KONVENSI PENULISAN TUGAS AKHIR.....	xxiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Permasalahan.....	2
1.3 Tujuan Penelitian.....	2
1.4 Metodologi Penelitian.....	2
1.5 Sistematika Laporan.....	3
1.6 Relevansi.....	3
BAB 2 TEORI PENUNJANG.....	5
2.1 Tinjauan Pustaka.....	5
2.2 <i>Fault Tolerant Control (FTC)</i>	5
2.2.1 Definisi, Klasifikasi, dan Pemodelan Fault.....	6
2.2.2 Kesalahan Aktuator.....	8
2.2.3 Struktur FTC.....	9
2.2.4 <i>Fault Detection and Diagnosis</i>	10
2.3 Sistem Pendulum Kereta.....	11
2.4 Linearisasi.....	15
2.5 Logika <i>Fuzzy</i>	18
2.5.1 Mekanisme Inferensi.....	22
2.5.2 Defuzzifikasi.....	23
2.6 Model <i>Fuzzy</i> Takagi Sugeno (T-S).....	24
2.7 Linear Quadratic Regulator (LQR).....	26
2.8 <i>Proportional Multiple Integral Observer (PMIO)</i>	27
2.9 Analisis Kestabilan Lyapunov.....	29
2.10 <i>Linear Matrix Inequalities (LMI)</i>	30

2.10.1	<i>Schur Complement</i>	31
2.10.2	<i>LMI Region</i>	31
2.11	Performa H_{∞}	32
BAB 3	PERANCANGAN SISTEM KONTROL	35
3.1	Linearisasi Model <i>Plant</i>	35
3.2	Perancangan Kontrol <i>Linear Quadratic Regulator</i> (LQR) untuk Stabilisasi	37
3.3	Perancangan <i>Proportional Multiple Integral Observer</i> untuk Estimasi Kesalahan Aktuator	41
3.4	Struktur Sistem Kontrol Stabilisasi dan <i>Proportional Multiple Integral Observer</i>	46
3.5	Perancangan Simulasi <i>Plant</i> Sistem Pendulum Kereta	48
3.6	Perancangan Implementasi Sistem Pendulum Kereta	49
BAB 4	HASIL PENGUJIAN DAN ANALISIS	51
4.1	Hasil Simulasi	51
4.1.1	Hasil Simulasi dalam Kondisi <i>Fault Free Case</i>	51
4.1.2	Hasil Simulasi dalam Kondisi <i>Faulty Case</i>	63
4.2	Hasil Implementasi.....	68
4.1.1	Hasil Implementasi Kondisi <i>Fault Free Case</i>	68
4.1.2	Hasil Implementasi Kondisi <i>Faulty Case</i>	73
BAB 5	PENUTUP	77
5.1	Kesimpulan	77
5.2	Saran.....	77
	DAFTAR PUSTAKA	79
	LAMPIRAN A	81
A.1	Penurunan Model Matematika Sistem Pendulum Kereta	81
	LAMPIRAN B	90
B.1	Program untuk Mencari Nilai <i>State Feedback Gain Controller</i>	90
B.2	Program untuk Mencari Nilai <i>Observer Gain</i>	91
B.3	Diagram Blok <i>Simulink</i>	93
	RIWAYAT HIDUP	99

DAFTAR TABEL

Tabel 2.1	Jenis-Jenis <i>Actuator Fault</i>	8
Tabel 4.1	Kombinasi Matriks Q dan R	52
Tabel 4.2	<i>State Feedback Gain Controller</i> dari beberapa Kombinasi Matriks Q dan R	53
Tabel 4.3	Perbandingan Spesifikasi Respon Posisi Pendulum dengan Berbagai Kombinasi Matriks Q dan R	54
Tabel 4.4	Perbandingan Spesifikasi Transien Respon Posisi Kereta dengan Berbagai Kombinasi Matriks Q dan R	55
Tabel 4.5	Perbandingan Spesifikasi Transien Sinyal Kontrol dengan Berbagai Kombinasi Matriks Q dan R	56
Tabel 4.6	Perbandingan Spesifikasi Transien Respon Posisi Pendulum dengan Berbagai Kondisi Awal	57
Tabel 4.7	Perbandingan Spesifikasi Transien Respon Posisi Kereta dengan Berbagai Kondisi Awal	58
Tabel 4.8	Perbandingan Spesifikasi Transien Sinyal Kontrol dengan Berbagai Kondisi Awal	59
Tabel 4.9	Perbandingan Spesifikasi Transien Respon Posisi Pendulum Dengan Beberapa Kondisi Awal Kecepatan Pendulum.....	61
Tabel 4.10	Perbandingan Spesifikasi Transien Respon Posisi Kereta Dengan Beberapa Kondisi Awal Kecepatan Pendulum...	62
Tabel 4.11	Perbandingan Spesifikasi Transien Sinyal Kontrol Dengan Beberapa Kondisi Awal Kecepatan Pendulum.....	63
Tabel 4.12	Perbandingan Spesifikasi Transien Respon Posisi Kereta saat Implementasi <i>Fault Free Case</i>	68
Tabel 4.13	Perbandingan Spesifikasi Transien Respon Posisi Pendulum saat Implementasi <i>Fault Free Case</i>	69
Tabel 4.14	Spesifikasi Respon Posisi Kereta Tanpa Kesalahan saat Tanpa dan Dengan Diberi Kompensasi FTC.....	71
Tabel 4.15	Perbandingan Respon Posisi Kereta dengan Kesalahan Step	74



DAFTAR GAMBAR

Gambar 2.1	Klasifikasi <i>Fault</i> berdasarkan Dependensinya terhadap Waktu	6
Gambar 2.2	Klasifikasi Kesalahan berdasarkan Letak Terjadinya	6
Gambar 2.3	Klasifikasi Kesalahan berdasarkan Dampaknya terhadap Sistem	8
Gambar 2.4	Skema <i>Fault Tolerant Control</i>	10
Gambar 2.5	Skema <i>Fault Diagnosis</i>	11
Gambar 2.6	Bentuk Fisik <i>Digital Pendulum Mechanical Unit 33-200</i>	11
Gambar 2.7	Diagram Fisik Sistem Pendulum Kereta.....	12
Gambar 2.8	Perangkat Sistem Pendulum	13
Gambar 2.9	Gaya-Gaya yang Bekerja pada Sistem Pendulum Kereta	13
Gambar 2.10	Linearisasi $f(x)$ di x^*	16
Gambar 2.11	Sistem Kontrol <i>Fuzzy</i>	18
Gambar 2.12	Fungsi Keanggotaan Segitiga dengan $\{a, b, c\}=\{2, 5, 8\}$	20
Gambar 2.13	Fungsi Keanggotaan Trapesium dengan $\{a, b, c, d\}=\{2, 4, 6, 8\}$	20
Gambar 2.14	Fungsi Keanggotaan <i>Generalized Bell</i> dengan $\{a, b, c\}=\{3, 4, 5\}$	21
Gambar 2.15	Fungsi Keanggotaan Gaussian dengan $\{c, \sigma\}=\{1.2, 5\}$	21
Gambar 2.16	Blok Diagram Estimasi Kesalahan dengan PMIO....	28
Gambar 2.17	Skema Kontrol untuk Mendapatkan Performa H_∞ ..	33
Gambar 3.1	<i>Membership Function</i> Posisi Sudut Pendulum untuk Aturan <i>Plant</i> dan Aturan Kontroler	40
Gambar 3.2	Struktur Sistem Kontrol LQR.....	46
Gambar 3.3	Diagram Sistem Kontrol LQR.....	46
Gambar 3.4	Struktur Sistem Kontrol dengan <i>Proportional Multiple Integral Observer</i>	47
Gambar 3.5	Diagram Sistem Kontrol dengan <i>Proportional Multiple Integral Observer</i>	47
Gambar 3.6	Diagram Simulink untuk Simulasi <i>Plant</i> Sistem Pendulum Kereta	48
Gambar 3.7	Diagram <i>Simulink</i> untuk <i>Observer</i>	49

Gambar 3.8	Diagram <i>Simulink</i> untuk <i>State Feedback Gain Controller</i>	49
Gambar 4.1	Respon Posisi Pendulum dengan Berbagai Kombinasi Matriks Q dan R	53
Gambar 4.2	Respon Posisi Kereta dengan Berbagai Kombinasi Matriks Q dan R	55
Gambar 4.3	Sinyal Kontrol dengan Berbagai Kombinasi Matriks Q dan R.....	56
Gambar 4.4	Respon Posisi Pendulum dengan Berbagai Kondisi Awal.....	57
Gambar 4.5	Respon Posisi Kereta dengan Berbagai Kondisi Awal	58
Gambar 4.6	Sinyal Kontrol dengan Berbagai Kondisi Awal	59
Gambar 4.7	Respon Posisi Pendulum Dengan Beberapa Kondisi Awal Kecepatan Pendulum	60
Gambar 4.8	Respon Posisi Kereta Dengan Beberapa Kondisi Awal Kecepatan Pendulum.....	61
Gambar 4.9	Sinyal Kontrol Dengan Beberapa Kondisi Awal Kecepatan Pendulum.....	62
Gambar 4.10	Sinyal Kesalahan <i>Step</i> dan Estimasinya.....	64
Gambar 4.11	Respon Posisi Kereta Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan <i>Step</i>	64
Gambar 4.12	Respon Posisi Pendulum Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan <i>Step</i>	65
Gambar 4.13	<i>L2-Gain</i> untuk Kesalahan Berupa Sinyal <i>Step</i>	65
Gambar 4.14	Sinyal Kesalahan <i>Ramp</i> dan Estimasinya	66
Gambar 4.15	Respon Posisi Kereta Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan <i>Ramp</i>	66
Gambar 4.16	Respon Posisi Pendulum Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan <i>Ramp</i>	67
Gambar 4.17	<i>L2-Gain</i> untuk Kesalahan Berupa Sinyal <i>Ramp</i>	67
Gambar 4.18	Respon Posisi Kereta saat Implementasi <i>Fault Free Case</i>	68
Gambar 4.19	Respon Posisi Pendulum saat Implementasi <i>Fault Free Case</i>	69
Gambar 4.20	Sinyal Kontrol saat Implementasi <i>Fault Free Case</i> ..	70
Gambar 4.21	Tanpa Kesalahan dan Estimasinya	70
Gambar 4.22	Respon Posisi Kereta saat Implementasi Kondisi <i>Fault Free Case</i> Dengan dan Tanpa Kompensasi FTC	71

Gambar 4.23	Respon Posisi Pendulum saat Implementasi Kondisi <i>Fault Free Case</i> Dengan dan Tanpa Kompensasi FTC	72
Gambar 4.24	Sinyal Kontrol saat Implementasi Kondisi <i>Fault Free Case</i> Dengan dan Tanpa Kompensasi FTC	72
Gambar 4.25	Sinyal Kesalahan <i>Step</i> saat Implementasi.....	73
Gambar 4.26	Respon Posisi Kereta dengan Kesalahan <i>Step</i> saat Implementasi	73
Gambar 4.27	Respon Posisi Pendulum Implementasi dengan Kesalahan <i>Step</i>	74
Gambar 4.28	Sinyal Kesalahan <i>Ramp</i> saat Implementasi	75
Gambar 4.29	Respon Posisi Kereta dengan Kesalahan <i>Ramp</i> saat Implementasi	75
Gambar 4.30	Respon Posisi Pendulum Implementasi dengan Kesalahan <i>Ramp</i>	76





DAFTAR NOTASI DAN SIMBOL

\mathbf{x}	: Vektor <i>state</i> x
x_n	: <i>State</i> ke- n
\dot{x}_n	: Turunan pertama <i>state</i> ke- n
f_a	: Kesalahan aktuator
$f_a^{(s)}$: Turunan ke- s dari kesalahan aktuator
$\bar{\mathbf{x}}_a$: Vektor <i>state</i> estimasi <i>observer</i>
\in	: Anggota himpunan
\notin	: Bukan anggota himpunan
\mathbf{I}	: Matriks identitas
\mathbf{A}^T	: Transpos matriks \mathbf{A}
\mathbf{A}^{-1}	: Invers matriks \mathbf{A}
$\mathbf{A} > 0$: Matriks \mathbf{A} definit positif
$\mathbf{A} < 0$: Matriks \mathbf{A} definit negatif
γ_{pmi}	: Tingkat pelemahan
α	: Batas <i>pole</i> terbesar <i>observer</i>
θ	: Batas sudut <i>pole</i> imajiner <i>observer</i>
∞	: <i>Infinity</i>
$ \mathbf{A} $: Determinan matriks \mathbf{A}
$\ \cdot\ _\infty$: ∞ -norm
$\ \cdot\ _2$: L_2 -norm
sup	: Supremum
$\sum(\cdot)$: Penjumlahan dari nilai deret
$\prod(\cdot)$: Perkalian dari nilai deret
$\frac{d}{dt}(\cdot)$: Turunan pertama dari t



-Halaman ini sengaja dikosongkan-

BAB 1

PENDAHULUAN

Bab ini membahas tentang hal-hal yang mendahului pelaksanaan Tugas Akhir. Hal-hal tersebut meliputi alasan-alasan yang menjadi latar belakang Tugas Akhir, perumusan masalah, tujuan pengerjaan Tugas Akhir, sistematika penulisan, dan relevansi.

1.1 Latar Belakang

Penelitian mengenai sistem pendulum kereta untuk representasi model *plant* nonlinear sudah banyak dikembangkan pada bidang teknik pengaturan. Permasalahan kontrol yang sering digunakan pada sistem pendulum kereta adalah *swing-up*, stabilisasi dan *tracking*. *Swing-up* adalah usaha untuk mengayunkan pendulum dari posisi menggantung ke bawah menuju posisi terbalik ke atas. Stabilisasi merupakan usaha yang dilakukan untuk menjaga posisi batang pendulum tetap berada dalam posisi terbalik serta menjaga pergerakan kereta sekecil mungkin. Sedangkan *tracking* adalah usaha yang dilakukan agar kereta dapat bergerak mengikuti sinyal referensi yang diberikan dengan tetap mempertahankan pendulum pada posisi terbalik.

Model *plant* nonlinear dapat didekati dengan model linear dengan menggunakan teori linearisasi. Pemodelan *fuzzy* Takagi-Sugeno (T-S) digunakan untuk menerapkan algoritma kontrol linear ke setiap sub-sistem linear [1]. Dengan menggunakan model *fuzzy* T-S, sistem nonlinear global digambarkan sebagai model linear yang mewakili untuk setiap daerah operasi yang berbeda. Model keseluruhan sistem merupakan gabungan dari model-model linear tersebut.

Pada Tugas Akhir ini, skema kontrol secara keseluruhan akan mengikuti aturan *Parallel Distributed Compensation* (PDC), dimana dari masing-masing subsistem dapat dihitung *state feedback gain* yang akan mengompensasi model *fuzzy* T-S yang bersesuaian. *State feedback gain* dapat diperoleh dengan menggunakan metode *Linear Quadratic Regulator* (LQR) [2].

Dalam kenyataannya kerusakan sering terjadi pada komponen-komponen dalam sistem, termasuk aktuator. Dengan adanya kerusakan atau kesalahan dalam aktuator, maka dapat terjadi penurunan performa sistem atau bahkan terjadi ketidakstabilan. Oleh karena itu *Fault*

Tolerant Control (FTC) digunakan untuk menjamin kestabilan dan performa sistem apabila terjadi kesalahan pada aktuator.

FTC dapat mengestimasi kesalahan yang terjadi pada sistem dengan menggunakan *Proportional Multiple Integral Observer*. Hasil estimasi tersebut digunakan untuk mengkompensasi kesalahan aktuator yang terjadi sehingga dapat menjamin kestabilan dari sistem dan performa sistem dapat terjaga [3].

1.2 Perumusan Masalah

Sistem pendulum kereta mempunyai kemungkinan terjadinya kesalahan pada aktuator. Akibat kesalahan aktuator tersebut dapat mempengaruhi performa dari respon posisi pergerakan kereta dari sistem pendulum kereta. Sehingga perlu adanya metode kontrol FTC untuk mengkompensasi kesalahan aktuator tersebut. Pada Tugas Akhir ini didesain suatu *observer* yang digunakan untuk mengestimasi kesalahan aktuator yaitu dengan menggunakan *Proportional Multiple Integral Observer* (PMIO). Dengan adanya *observer* tersebut, diharapkan sistem kontrol mampu menjaga kestabilan dan performa sistem pendulum kereta. Pada Tugas Akhir ini, kesalahan yang diberikan pada aktuator adalah bersifat *bias* yang berupa kesalahan *step* dan *ramp*.

1.3 Tujuan Penelitian

Tugas Akhir ini bertujuan untuk merancang sistem kontrol *state feedback* menggunakan metode LQR berbasis *fuzzy T-S* yang dapat digunakan untuk stabilisasi sistem pendulum kereta. Dengan menambahkan metode FTC, sistem kontrol dapat menjaga stabilitas dan performa sistem pendulum kereta saat terjadi kesalahan aktuator. Hasil sistem kontrol yang telah dirancang kemudian diujikan pada *plant* nyata “*Digital Pendulum Mechanical Unit 33-200*”.

1.4 Metodologi Penelitian

Metodologi yang dilakukan pada Tugas Akhir ini adalah:

1. Studi literatur

Pada tahap ini dilakukan studi literatur mengenai referensi yang diperlukan dalam pengerjaan tugas akhir, seperti identifikasi fisik *plant* “*Digital Pendulum Mechanical Unit 33-200*”, desain *Linear Quadratic Regulator* berbasis *fuzzy T-S*, *Fault Tolerant Control*, *Proportional Multiple Integral Observer*, dan LMI

2. Pemodelan sistem

Membuat persamaan *state space* yang merepresentasikan *plant* pendulum dari hasil identifikasi fisik dan hukum-hukum fisika.

3. Perancangan kontroler

Pada tahap ini dilakukan perhitungan nilai *gain* kontroler menggunakan *listing program* pada *software* MATLAB, serta nilai *observer gain* dihitung menggunakan teknik pemrograman LMI.

4. Simulasi

Hasil pemodelan sistem dan perancangan kontroler disimulasikan terlebih dahulu sebelum diimplementasikan dengan menggunakan bantuan perangkat lunak *Simulink*/MATLAB.

5. Implementasi

Hasil simulasi kemudian diimplementasikan pada *plant* nyata "*Digital Pendulum Mechanical Unit 33-200*". Data-data dari hasil implementasi dianalisis dan dilaporkan, kemudian dapat diambil suatu kesimpulan.

6. Penulisan buku Tugas Akhir

Penulisan buku merupakan bagian terakhir dari pengerjaan Tugas Akhir. Penulisan buku ini berupa laporan ilmiah yang mencakup semua proses pengerjaan Tugas Akhir, mulai dari pendahuluan, teori penunjang, perancangan sistem, hasil simulasi dan implementasi, hingga penutup yang berisi kesimpulan dan saran yang diperoleh dari penelitian.

1.5 Sistematika Laporan

Pembahasan Tugas Akhir ini akan dibagi menjadi lima bab dengan sistematika sebagai berikut:

Bab I : Pendahuluan

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

Bab II : Teori Penunjang

Bab ini menjelaskan tentang tinjauan pustaka, konsep *Fault Tolerant Control*, konsep dasar dari Sistem Pendulum-Kereta, linearisasi, teori logika *fuzzy*, model *fuzzy* Takagi-Sugeno, *Linear Quadratic Regulator*, *Proportional Multiple Integral Observer*, analisis kestabilan Lyapunov, *Linear Matrix Inequalities*, dan performa H_{∞} .

Bab III : Perancangan Sistem Kontrol

Bab ini membahas desain dan perancangan algoritma kontrol untuk metode kontrol stabilisasi berdasarkan teori penunjang pada Bab II.

Bab IV : Hasil Simulasi dan Implementasi

Bab ini memuat hasil simulasi dan hasil implementasi serta analisis dari hasil tersebut.

Bab V : Penutup

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

1.6 Relevansi

Hasil yang diperoleh dari pengerjaan Tugas Akhir ini diharapkan dapat menjadi referensi perancangan *Fault Tolerant Control* pada sistem pendulum kereta saat terjadi kesalahan aktuator.

BAB 2

TEORI PENUNJANG

Pada bab ini akan dibahas mengenai teori-teori dasar yang digunakan dalam menyelesaikan Tugas Akhir. Teori-teori tersebut diperlukan sebagai bahan referensi atau menjadi bahan landasan dalam merancang sistem kontrol.

2.1 Tinjauan Pustaka

Sistem pendulum kereta merupakan sistem yang sering digunakan untuk menguji beberapa metode kontrol. Sistem pendulum kereta terdiri dari batang pendulum yang terpasang pada sebuah kereta. Batang pendulum dapat bergerak secara bebas rotasi pada bidang vertikal, sedangkan pergerakan kereta adalah di bidang horizontal dengan panjang lintasan tertentu.

Sistem pendulum kereta merupakan sistem nonlinear. Agar teori kontrol linear dapat digunakan maka proses linearisasi dibutuhkan agar beberapa sistem linear dapat diperoleh dan digunakan untuk merepresentasikan sistem nonlinear. Pendekatan model *fuzzy* Takagi-Sugeno digunakan untuk menerapkan algoritma kontrol linear ke setiap subsistem linear. Seperti pada [2], *state feedback gain* diperoleh dengan pendekatan kontrol optimal *Linear Quadratic Regulator* (LQR). Dalam memperoleh *state feedback gain* menggunakan metode LQR, dilakukan dengan mencari matriks **Q** dan **R** yang sesuai dengan indeks performa.

Sistem kontrol ini akan bekerja dalam menjaga stabilitas sistem pendulum kereta. Akan tetapi, apabila terjadi kesalahan pada komponen dalam sistem (pada tugas akhir ini, aktuator) maka akan terjadi penurunan performa dan stabilitas sistem tidak terjamin. Oleh karena itu, perlu adanya metode kontrol tambahan yang digunakan apabila terjadi kesalahan pada aktuator, yaitu *Fault Tolerant Control* (FTC).

Pada tugas akhir ini, akan dilakukan estimasi kesalahan pada aktuator dengan pendekatan metode *Proportional Multiple Integral Observer* [3]. Sedangkan pada kondisi nominal, sistem kontrol yang bekerja adalah *fuzzy* LQR untuk stabilisasi pendulum.

2.2 Fault Tolerant Control (FTC) [3]

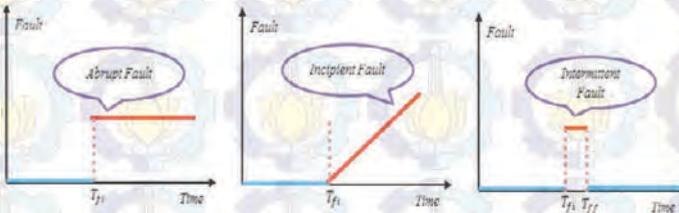
Suatu metode kontrol yang bernama kontrol toleransi kesalahan atau sering disebut *fault tolerant control* (FTC) dapat memenuhi kondisi

meningkatnya permintaan dalam mempertahankan kinerja sistem sesuai dengan yang diinginkan meskipun terjadi kerusakan. FTC dapat mengkompensasi dampak penurunan performa yang diakibatkan adanya kesalahan.

2.2.1 Definisi, Klasifikasi, dan Pemodelan *Fault*

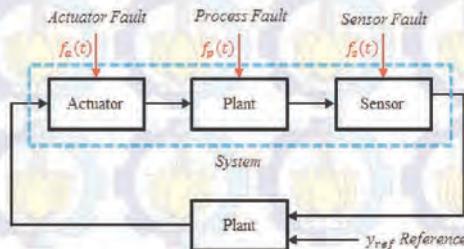
Keadaan cacat yang tidak terkendali pada struktur sistem atau parameter yang pada akhirnya mengarah pada penurunan kinerja sistem disebut kesalahan (*fault*). Sedangkan kesalahan pada kondisi yang sudah sangat parah disebut kegagalan (*failure*). Oleh karena itu, kesalahan adalah state yang dapat menyebabkan kerusakan atau kegagalan sistem.

Pada Gambar 2.1 kesalahan dapat diklasifikasikan berdasarkan karakteristik terhadap waktu yaitu *abrupt fault*, *incipient fault*, dan *intermittent fault*.



Gambar 2.1 Klasifikasi Fault berdasarkan Dependensinya terhadap Waktu

Abrupt fault (sinyal kesalahan berupa *step*) adalah kesalahan yang terjadi secara tiba-tiba dan nilainya konstan terhadap waktu. *Incipient fault* (sinyal kesalahan berupa *ramp*) adalah kesalahan yang terjadi secara bertahap besarnya berdasarkan waktu. Sedangkan *intermittent fault* (sinyal kesalahan berupa *impulse*) adalah kesalahan yang terjadi hanya sesaat saja atau pada selang waktu tertentu.



Gambar 2.2 Klasifikasi Kesalahan berdasarkan Letak Terjadinya

Pada Gambar 2.2 dapat dilihat bahwa *actuator fault* mempengaruhi besarnya sinyal masukan pada *plant* baik sepenuhnya ataupun sebagian. Kesalahan aktuator secara penuh mengakibatkan aktuator tidak dapat memberikan aktuasi meskipun diberikan sinyal masukan. Sedangkan kesalahan aktuator sebagian mengakibatkan respon yang diberikan aktuator menjadi lebih lambat. *Sensor fault* adalah kesalahan yang terjadi pada sensor, sehingga mengakibatkan pembacaan atau pengukuran dari keluaran suatu sistem menjadi tidak sesuai dengan kenyataannya. Sedangkan *process fault* adalah kesalahan yang mengakibatkan berubahnya parameter dalam sistem. *Process fault* lebih dikenal dengan kerusakan komponen.

Kesalahan juga dapat diklasifikasikan berdasarkan dampaknya sistem yaitu *additive fault* dan *multiplicative fault*. Secara struktur dapat dilihat pada Gambar 2.3. Sedangkan secara matematis, *actuator fault* dan *sensor fault* yang mengakibatkan *multiplicative fault* dapat dituliskan seperti pada (2.1) dan (2.2).

$$\mathbf{u}^f = \varepsilon^a \mathbf{u} \quad (2.1)$$

$$\mathbf{y}^f = \varepsilon^s \mathbf{y} \quad (2.2)$$

dengan ε^a dan ε^s adalah matriks diagonal yang elemennya $0 < \varepsilon_i^a < 1$ dan $0 < \varepsilon_i^s < 1, i = 1, 2, \dots, m$. Pada model *state space* sistem nominal

$$\left. \begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx} \end{aligned} \right\} \quad (2.3)$$

jika terjadi *multiplicative fault* maka model *state space* sistem menjadi

$$\left. \begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{B}\varepsilon^a \mathbf{u} \\ \mathbf{y}^f &= \varepsilon^s \mathbf{Cx} \end{aligned} \right\} \quad (2.4)$$

Sedangkan dampak dari *actuator fault* dan *sensor fault* pada model *state space* secara *additive* dapat dituliskan

$$\left. \begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} + \mathbf{F}_x \mathbf{f}_a \\ \mathbf{y} &= \mathbf{Cx} + \mathbf{F}_y \mathbf{f}_s \end{aligned} \right\} \quad (2.5)$$

dengan \mathbf{f}_a dan \mathbf{f}_s adalah sinyal kesalahan aktuator dan sensor.

Sedangkan F_x dan F_y adalah matriks yang bersesuaian dengan sinyal kesalahan aktuator dan sensor.



Gambar 2.3 Klasifikasi Kesalahan berdasarkan Dampaknya terhadap Sistem

2.2.2 Kesalahan Aktuator [4]

Kesalahan yang terjadi pada aktuator akan mengakibatkan tidak sesuainya sinyal kontrol yang menjadi masukan dari *plant*. Oleh karena itu, sinyal kontrol yang terkena *fault* ini akan mengakibatkan ketidakstabilan pada *plant* tersebut. Terdapat beberapa kondisi kesalahan aktuator yang memberi dampak pada performa sistem.

Sinyal kontrol yang terkena *fault* dapat dimodelkan seperti pada (2.6).

$$U_f = \Gamma U + U_{f0} \quad (2.6)$$

dengan U adalah sinyal kontrol keluaran kontroler, U_f adalah sinyal kontrol yang sudah terkena *actuator fault* dan menjadi masukan dari *plant*, Γ adalah sinyal yang memberi pengaruh *multiplicative actuator fault*, dan U_{f0} sinyal yang memberi pengaruh *additive actuator fault*. Dengan penjelasan tersebut, maka kesalahan aktuator dapat diklasifikasikan menjadi beberapa tipe berdasarkan pada dampak yang dihasilkan pada sistem seperti pada Tabel 2.1.

Tabel 2.1 Jenis-Jenis Actuator Fault

Kondisi	$U_{f0} = 0$	$U_{f0} \neq 0$
$\Gamma = 1$	<i>Fault free case</i>	<i>Bias</i>
$0 < \Gamma < 1$	<i>Loss of effectiveness</i>	<i>Loss of effectiveness</i>
$\Gamma = 0$	<i>Out of order</i>	<i>Actuator blocked</i>

Kondisi *fault free case* adalah kondisi saat tidak terdapat kesalahan pada aktuator, yang artinya sistem bekerja sesuai kontrol nominal. Kondisi *bias* adalah kondisi saat hanya terdapat kesalahan aktuator yang bersifat *additive*. Hal ini mengakibatkan sinyal kontrol yang menjadi

masuk dari *plant* mempunyai nilai menyimpang dari sinyal kontrol yang dihasilkan oleh kontroler. Pada kondisi *loss of effectiveness*, aktuator mengalami kesalahan yang sifatnya *multiplicative* terhadap sistem. Oleh karena nilai pengalinya adalah antara 0 sampai 1, maka besarnya sinyal kontrol yang masuk ke *plant* akan menjadi lebih kecil sesuai dengan skala tertentu jika dibandingkan dengan sinyal kontrol yang dihasilkan oleh kontroler. Atau dengan kata lain, aktuator tersebut telah kehilangan keefektifannya.

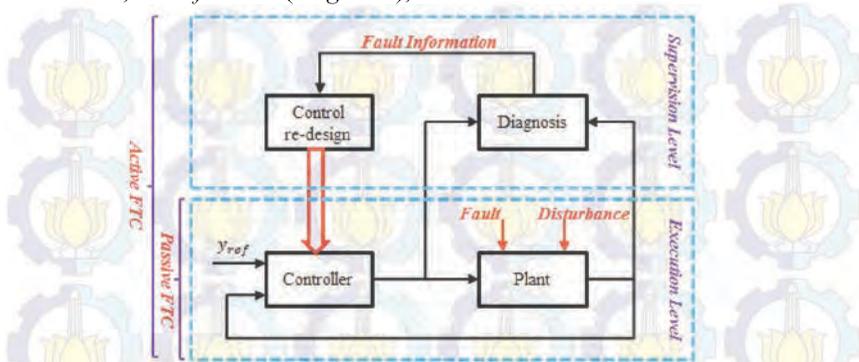
Aktuator pada kondisi *out of order* akan mengakibatkan sinyal kontrol yang masuk ke *plant* selalu bernilai 0. Akibatnya, berapapun besarnya sinyal kontrol yang dihasilkan oleh kontroler tidak akan mempengaruhi dari sistem. Sistem pada kondisi ini tidak akan bekerja lagi. Sedangkan kondisi *actuator blocked* adalah kondisi saat kontroler tidak mampu mengendalikan *plant* pada sistem, karena sinyal kontrol yang masuk ke *plant* murni dari sinyal kesalahan bukan dari sinyal yang dihasilkan oleh kontroler. Oleh karena itu, saat terjadi kondisi seperti ini maka aktuator harus diganti agar tidak terjadi kesalahan pada sistem.

2.2.3 Struktur FTC

Pada umumnya kontroler nominal dapat memberikan kestabilan dan performa *closed-loop* yang diinginkan pada kondisi operasi nominal (tanpa adanya kesalahan). Akan tetapi kondisi tidak dapat berjalan seperti itu disaat terjadi kesalahan pada sensor, aktuator, atau pada sistemnya sendiri. Agar kontroler mempunyai kemampuan dalam menoleransi kesalahan tersebut, maka dibutuhkan beberapa blok tambahan dalam struktur sistem *closed-loop* seperti pada Gambar 2.4.

Pada Gambar 2.4 dapat dilihat bahwa berdasarkan skema blok diagramnya, *fault tolerant control* (FTC) dibedakan dalam dua pendekatan yaitu *active* FTC (AFTC) dan *passive* FTC (PFTC). Pada pendekatan secara PFTC, kontroler didesain secara tetap dan menjadi *robust* terhadap perkiraan-perkiraan kesalahan. Jika dilihat secara sekilas tidak ada perbedaan dengan sistem kontrol nominal dalam hal skema blok diagramnya. Hal ini dikarenakan pada pendekatan ini tidak diperlukan skema *fault detection and diagnosis* (FDD) ataupun rekonfigurasi kontroler. Berbeda dengan PFTC, pada pendekatan AFTC memberikan aksi kontrol secara aktif terhadap kesalahan yang terjadi pada sistem dengan cara rekonfigurasi aksi kontrol. Hasilnya sistem kontrol menjadi lebih stabil dan performa dapat diterima secara penuh.

Pada beberapa penelitian, AFTC juga sering disebut sebagai skema *fault detection, identification (diagnosis), and accommodation*.



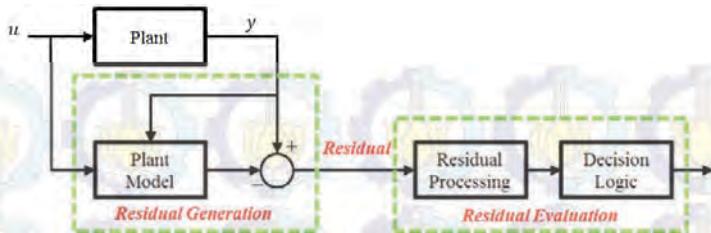
Gambar 2.4 Skema Fault Tolerant Control

2.2.4 Fault Detection and Diagnosis

Fault detection and diagnosis (FDD) merupakan prosedur dalam menentukan ada tidaknya kesalahan yang terjadi (*fault detection*) dan juga mempertimbangkan seberapa besar tingkat pengaruh kesalahan pada sistem (*fault diagnosis*) berdasarkan sinyal masukan dan keluaran. Pada proses FDD terdapat beberapa langkah sampai didapatkan diagnosis yang akurat. Langkah-langkah tersebut adalah:

- *Fault Detection*: Langkah ini menentukan terjadinya kesalahan dengan cara membangkitkan sinyal yang diakibatkan oleh sinyal kesalahan saja. Sinyal yang dibangkitkan ini disebut *residual* seperti pada Gambar 2.5. Pada beberapa literatur yang lain, langkah ini disebut *residual generation*.
- *Fault Isolation*: Langkah ini menentukan komponen mana yang terjadi kesalahan. Kesalahan tersebut bisa terjadi pada sensor, aktuator, ataupun pada sistem itu sendiri.
- *Fault Identification (Analysis)*: Langkah ini menentukan tipe atau jenis kesalahan apa yang terjadi dan dampak apa yang diakibatkan oleh kesalahan tersebut.

Pada kenyataannya *fault isolation and identification* didasarkan pada pengolahan sinyal *residual* untuk mendapatkan informasi kesalahan. Oleh karena itu pada beberapa literatur, proses tersebut dinamakan *decision making* atau *residual evaluation*.



Gambar 2.5 Skema *Fault Diagnosis* [5]

2.3 Sistem Pendulum Kereta [5]

Sistem pendulum kereta merupakan sistem nonlinear yang secara fisik terdiri atas batang pendulum yang terpasang pada sebuah kereta. Batang pendulum dapat bergerak secara bebas pada bidang vertikal, sedangkan pergerakan kereta adalah di bidang horizontal dengan panjang lintasan tertentu. Sistem pendulum kereta yang digunakan untuk implementasi adalah dari Feedback Instruments Ltd. dengan tipe “*Digital Pendulum Mechanical Unit 33-200*” seperti pada Gambar 2.6.



Gambar 2.6 Bentuk Fisik *Digital Pendulum Mechanical Unit 33-200*

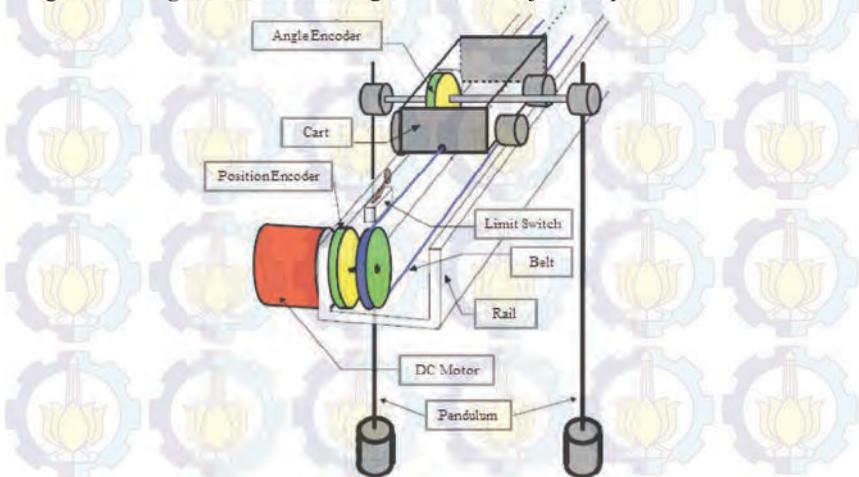
Penerapan sistem kontrol dilakukan pada komputer menggunakan Simulink pada *software* MATLAB. Modul “*Digital Pendulum Controller 33-201*” sebagai kontroler antarmuka digunakan untuk menghubungkan komputer dan sistem pendulum kereta, sedangkan *board* akuisisi data (DAQ) digunakan sebagai I/O pada

komputer. Sinyal kontrol dari komputer keluar melalui *Digital to Analog Converter* (DAC) yang terdapat pada DAQ.

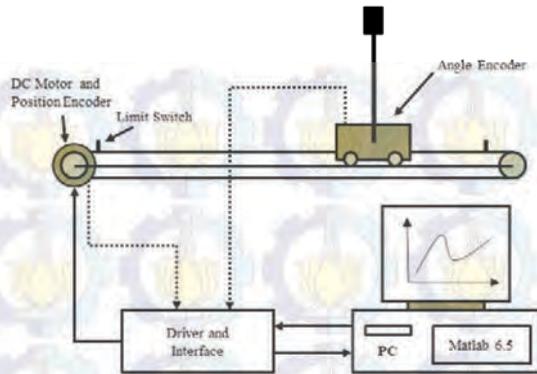
Power amplifier yang terhubung dengan port keluaran DAQ akan menerima sinyal kontrol yang kemudian dikirim ke motor DC untuk menggerakkan kereta. Sensor akan membaca sinyal respon motor DC kemudian diteruskan menuju encoder. Sinyal dari encoder akan diteruskan menuju komputer melalui ADC pada board DAQ.

Susunan rangkaian dari Sistem Pendulum Kereta secara keseluruhan dapat dilihat pada Gambar 2.7. Kereta dapat bergerak secara horizontal dengan menggunakan motor DC yang dihubungkan dengan *belt*. Sedangkan untuk mengayunkan dan menyeimbangkan batang pendulum, kereta didorong maju dan mundur pada rel.

Untuk mengontrol pergerakan kereta maupun batang pendulum, terdapat beberapa sensor yang digunakan pada sistem pendulum kereta. Karena panjang lintasan yang terbatas, maka pergerakan kereta harus diamankan dengan sensor *limit switch* agar posisi kereta tidak keluar dari lintasan tersebut. Sehingga saat kereta sampai di penghujung lintasan, kereta akan menyentuh *limit switch* yang secara otomatis mematikan motor DC sebagai penggerak kereta. Untuk mengukur posisi kereta pada lintasan, digunakan sensor posisi yaitu *position encoder*. Sedangkan untuk mengukur posisi sudut dari batang pendulum digunakan *angle encoder*. Konfigurasi ini ditunjukkan pada Gambar 2.8.

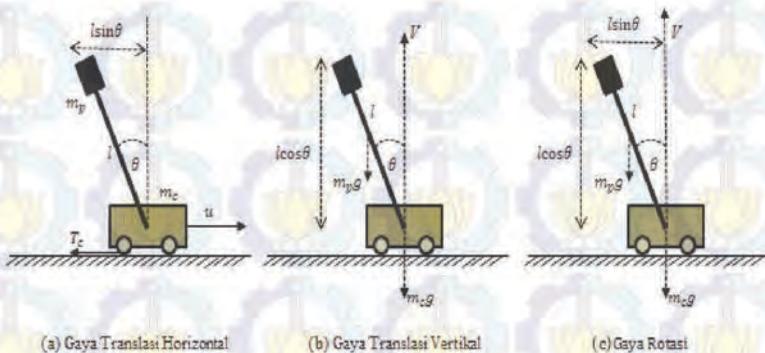


Gambar 2.7 Diagram Fisik Sistem Pendulum Kereta



Gambar 2.8 Perangkat Sistem Pendulum

Model matematika sistem pendulum kereta didapatkan melalui hasil identifikasi fisik atau penurunan rumus fisika. Penurunan rumus fisika yang digunakan adalah Hukum Newton dengan mengetahui gaya apa saja yang bekerja pada sistem tersebut. Pada Hukum Newton II, total gaya yang bekerja pada sistem akan berbanding lurus dengan percepatan gerak dari sistem tersebut. Sistem pendulum kereta memiliki dinamika pergerakan translasi dan rotasi. Oleh karena itu pada sistem pendulum kereta, gaya juga harus diidentifikasi secara translasi maupun rotasi. Gaya-gaya yang bekerja pada sistem pendulum kereta dapat dilihat pada Gambar 2.9.



Gambar 2.9 Gaya-Gaya yang Bekerja pada Sistem Pendulum Kereta

Pada gerak translasi bidang horizontal, gaya yang bekerja adalah F dan T_c . F adalah gaya yang timbul akibat aksi kontrol (u) yang dikenakan sejajar dengan arah pergerakan kereta. Sedangkan T_c adalah gaya gesek yang timbul akibat pergerakan dari kereta terhadap lintasan. Arah dari gaya gesek selalu berlawanan dengan arah pergerakan kereta. Hubungan antara F dan T_c dapat dilihat pada (2.7).

$$\left. \begin{aligned} \sum F_x &= ma_x \\ u - T_c &= (m_c + m_p)(\ddot{x} - l \sin \theta) \end{aligned} \right\} \quad (2.7)$$

Para gerak translasi bidang vertikal, gaya yang bekerja adalah V dan W seperti pada (2.8). V adalah gaya normal yang arahnya selalu tegak lurus terhadap bidang. Sedangkan W adalah gaya yang ditimbulkan oleh massa batang pendulum dan kereta, serta adanya percepatan gravitasi. Arah dari gaya W selalu menuju ke pusat bumi.

$$\left. \begin{aligned} \sum F_y &= ma_y \\ V - (m_c + m_p)g &= (m_c + m_p)(l \cos \theta) \ddot{\theta} \end{aligned} \right\} \quad (2.8)$$

Persamaan (2.9) menyatakan hubungan gaya-gaya yang bekerja secara keseluruhan pada gerak rotasi. D_p adalah momen gesek akibat pergerakan rotasi batang pendulum yang besarnya sebanding dengan kecepatan sudut pendulum. Sehingga D_p dapat dinyatakan sebagai $D_p = f_p x_4$. J adalah momen inersia dari batang pendulum.

$$\left. \begin{aligned} \sum \tau &= I \alpha \\ (u - T_c)l \cos \theta + V l \sin \theta - D_p &= J \ddot{\theta} \end{aligned} \right\} \quad (2.9)$$

Pada sistem pendulum kereta mempunyai vektor *state* yaitu $x = [x_1 \ x_2 \ x_3 \ x_4]^T$. Keempat elemen vektor *state* tersebut adalah:

- x_1 : Posisi kereta (dari titik tengah lintasan), x
- x_2 : Posisi pendulum terhadap garis vertikal, θ
- x_3 : Kecepatan kereta, $v = \dot{x}$
- x_4 : Kecepatan pendulum, $\omega = \dot{\theta}$

Dengan menguraikan persamaan gaya translasi dan rotasi, didapatkan persamaan *state* sistem pendulum kereta seperti pada (2.10).

$$\left. \begin{aligned}
 \mathfrak{X}_1 &= x_3 \\
 \mathfrak{X}_2 &= x_4 \\
 \mathfrak{X}_3 &= \frac{a(u - T_c - \mu x_4^2 \sin x_2) + l \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2} \\
 \mathfrak{X}_4 &= \frac{l \cos x_2 (u - T_c - \mu x_4^2 \sin x_2) + (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2}
 \end{aligned} \right\} \quad (2.10)$$

dengan $\mu = (m_c + m_p)l$ dan $a = l^2 + \frac{J}{m_c + m_p}$.

2.4 Linearisasi [6]

Salah satu cara untuk memahami karakteristik sistem adalah melalui pemahaman tentang bagaimana suatu sistem berinteraksi dengan sinyal masukan. Berdasarkan sifat linearitasnya, sistem dibagi menjadi dua yaitu sistem linear dan sistem nonlinear. Untuk menyelesaikan permasalahan sistem nonlinear, dapat dilakukan dengan cara linearisasi. Linearisasi adalah pendekatan linear di sekitar titik kerja yang ditentukan dari sistem nonlinear.

Dari hasil proses linearisasi, didapatkan suatu matriks konstanta yang bersesuaian dengan sistem kontrol linear di setiap daerah kerjanya, sehingga mempermudah pengontrolan. Sistem nonlinear dinyatakan dengan:

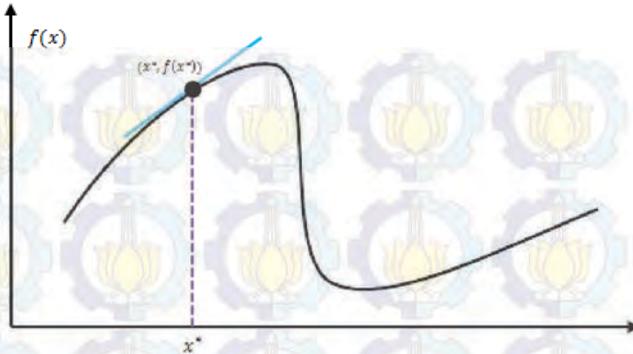
$$\mathfrak{X} = f(x) \quad (2.11)$$

Gambar 2.10 adalah kurva nonlinear dengan fungsi $y = f(x)$. Karena kurva tersebut nonlinear dan ingin didapatkan pendekatannya, maka perlu dilakukan linearisasi di titik kerja. Terlihat bahwa pada titik $(x^*, f(x^*))$, garis singgung merepresentasikan pendekatan linear $f(x)$ di sekitar titik $x = x^*$.

Hasil linearisasi dari sistem nonlinear pada (2.11) dapat dituliskan sebagai berikut:

$$y = f(x) \quad (2.12)$$

$$\delta y = (f(x) - f(x^*)), \delta x = x - x^* \quad (2.13)$$



Gambar 2.10 Linearisasi $f(x)$ di x^*

$$\delta y = f(\delta x) = \left. \frac{df(x)}{dx} \right|_{x=x^*} \delta x \quad (2.14)$$

Persamaan (2.14) menunjukkan hasil linearisasi di titik kerja sistem nonlinear. Titik kerja x^* merupakan titik pusat dari hasil linearisasi. Apabila titik kerja tersebut berada di titik origin, maka $f(x^*) = 0$ dan hasil linearisasi di titik tersebut:

$$f(x) = \mathfrak{X} = \left. \frac{df(x)}{dx} \right|_{x=x^*} x \quad (2.15)$$

Jika pada sistem nonlinear lup terbuka terdapat n buah *state*, maka dapat dilakukan linearisasi di titik kerja $(x_1^*, x_2^*, x_3^*, x_4^*, \dots, x_n^*)$. Hasil linearisasi tersebut menjadi:

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1(x_1, \dots, x_n)}{\partial x_1} & \dots & \frac{\partial f_1(x_1, \dots, x_n)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(x_1, \dots, x_n)}{\partial x_1} & \dots & \frac{\partial f_n(x_1, \dots, x_n)}{\partial x_n} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \Bigg|_{\substack{x_1=x_1^* \\ \vdots \\ x_n=x_n^*}}$$

atau dapat ditulis menjadi:

$$\mathfrak{X} = \mathbf{A} \mathbf{x} \quad (2.16)$$

dengan:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \mathbf{M} \\ x_n \end{bmatrix}$$

$$\mathbf{A} = \left[\begin{array}{cc} \frac{\partial f_1(x_1, \dots, x_n)}{\partial x_1} & \frac{\partial f_1(x_1, \dots, x_n)}{\partial x_n} \\ \mathbf{M} & \mathbf{O} \\ \frac{\partial f_n(x_1, \dots, x_n)}{\partial x_1} & \frac{\partial f_n(x_1, \dots, x_n)}{\partial x_n} \end{array} \right]_{x_1=x_1^*}$$

Dengan demikian, matriks A merupakan matriks Jacobi dari $f(x)$. Sistem nonlinear dapat didekati menjadi sistem linear berdasarkan hasil linearisasi di sekitar titik kerjanya dan dapat direpresentasikan dalam persamaan berikut:

$$\mathbf{x} = f(\mathbf{x}) + h(\mathbf{x}, \mathbf{u}) \quad (2.17)$$

Jika *state* dalam sistem berjumlah n , maka linearisasi sistem pada titik kerja $(x_1^*, x_2^*, x_3^*, x_4^*, \dots, x_n^*)$ adalah sebagai berikut:

$$\mathbf{x} = \mathbf{Ax} + \mathbf{Bu} \quad (2.18)$$

dengan:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \mathbf{M} \\ x_n \end{bmatrix}$$

$$\mathbf{A} = \left[\begin{array}{cc} \frac{\partial f_1(x_1, \dots, x_n)}{\partial x_1} & \frac{\partial f_1(x_1, \dots, x_n)}{\partial x_n} \\ \mathbf{M} & \mathbf{O} \\ \frac{\partial f_n(x_1, \dots, x_n)}{\partial x_1} & \frac{\partial f_n(x_1, \dots, x_n)}{\partial x_n} \end{array} \right]_{x_1=x_1^*}$$

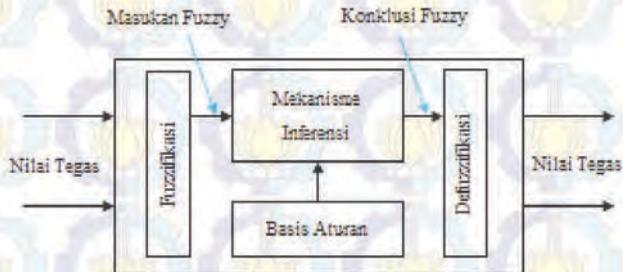
$$B = \left[\begin{array}{c} \frac{\partial h_1(x_1, \dots, x_n)}{\partial u} \\ M \\ \frac{\partial h_n(x_1, \dots, x_n)}{\partial u} \end{array} \right]_{\substack{x=x^* \\ u=u^*}}$$

2.5 Logika Fuzzy [7]

Sebelum munculnya teori logika *fuzzy* (*fuzzy logic*), dikenal sebuah logika tegas (*crisp logic*) yang memiliki nilai benar atau salah. Sebaliknya logika *fuzzy* merupakan sebuah logika yang memiliki nilai kekaburan atau kesamaran (*fuzzyness*) antara benar dan salah. Dalam teori logika *fuzzy* sebuah nilai bisa bernilai benar dan salah secara bersamaan, namun seberapa besar nilai kebenaran atau kesalahan tergantung pada bobot keanggotaan yang dimilikinya.

Dalam teori logika *fuzzy* dikenal himpunan *fuzzy* (*fuzzy set*) yang merupakan pengelompokan sesuatu berdasarkan variabel linguistic yang dinyatakan dalam fungsi keanggotaan. Di dalam semesta pembicaraan, fungsi keanggotaan dari suatu himpunan *fuzzy* tersebut bernilai antara 0 sampai dengan 1.

Beberapa komponen dan kaidah yang harus dipenuhi dalam mendesain sistem kontrol *fuzzy* agar respon sistem sesuai dengan yang diinginkan adalah fuzzifikasi *input*, basis aturan, mekanisme inferensi, dan defuzzifikasi seperti pada Gambar 2.11.



Gambar 2.11 Sistem Kontrol Fuzzy

Fuzzifikasi merupakan proses pengubahan nilai tegas yang ada ke dalam fungsi keanggotaan. Basis aturan berisi kuantifikasi logika *fuzzy* (aturan *If-Then*) tentang cara untuk mencapai hasil kontrol yang baik. Mekanisme inferensi adalah proses implikasi dalam menalar nilai masukan guna penentuan nilai keluaran sebagai bentuk pengambil keputusan. Sedangkan defuzzifikasi adalah proses pemetaan himpunan *fuzzy* ke himpunan tegas. Proses defuzzifikasi ini adalah kebalikan dari proses fuzzifikasi.

Fungsi keanggotaan dari suatu himpunan *fuzzy* dinyatakan dengan derajat keanggotaan suatu nilai terhadap nilai tegasnya yang berkisar antara 0 sampai dengan 1. Jika A adalah himpunan *fuzzy*, μ adalah fungsi keanggotaan, dan X adalah semesta, maka fungsi keanggotaan dalam suatu himpunan *fuzzy* dapat dinyatakan dengan:

$$A = \{(x, \mu(x)) \mid x \in X\} \quad (2.19)$$

Fungsi keanggotaan suatu himpunan *fuzzy* dapat ditentukan dengan beberapa fungsi, seperti fungsi segitiga (*triangle*), trapesium (*trapezoidal*), *generalized bell*, atau Fungsi Gauss (*Gaussian*).

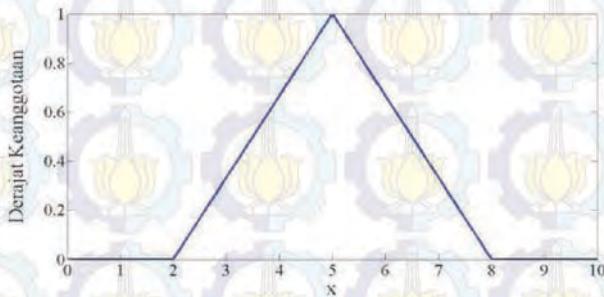
Fungsi keanggotaan yang paling sederhana dibentuk dengan menggunakan garis lurus. Dan yang paling sederhana adalah fungsi keanggotaan segitiga. Fungsi keanggotaan segitiga ditentukan oleh tiga parameter $\{a, b, c\}$ dengan $a < b < c$ seperti yang ditunjukkan pada (2.20). Gambar 2.12 menunjukkan fungsi keanggotaan segitiga dengan parameter $\{2, 5, 8\}$.

$$\mu_{\text{segitiga}}(x) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x < b \\ \frac{c-x}{c-b}, & b \leq x < c \\ 0, & x \geq c \end{cases} \quad (2.20)$$

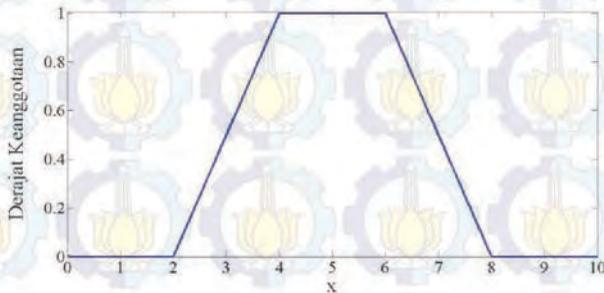
Fungsi keanggotaan trapesium juga dibentuk dengan menggunakan garis lurus. Akan tetapi fungsi keanggotaan trapesium ditentukan oleh empat parameter $\{a, b, c, d\}$ dengan $a < b < c < d$ seperti yang ditunjukkan pada (2.21). Parameter a dan d menyatakan nilai batas, sedangkan b dan c adalah nilai puncak. Gambar 2.13

menunjukkan fungsi keanggotaan trapesium dengan parameter $\{2, 4, 6, 8\}$.

$$\mu_{\text{trapesium}}(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x < b \\ 1 & b \leq x < c \\ \frac{d-x}{d-c} & c \leq x < d \\ 0 & x \geq d \end{cases} \quad (2.21)$$



Gambar 2.12 Fungsi Keanggotaan Segitiga dengan $\{a, b, c\}=\{2, 5, 8\}$



Gambar 2.13 Fungsi Keanggotaan Trapesium dengan $\{a, b, c, d\}=\{2, 4, 6, 8\}$

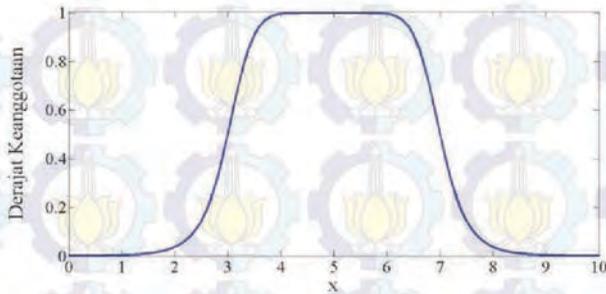
Persamaan fungsi keanggotaan *generalized bell* dapat dilihat pada (2.22) dan grafiknya pada Gambar 2.14. Fungsi keanggotaan *Generalized Bell* ditentukan oleh tiga parameter $\{a, b, c\}$. Parameter c menyatakan titik pusat dengan derajat keanggotaan bernilai 1, parameter

b menyatakan bilangan positif, dan parameter a menyatakan lebar fungsi *bell*.

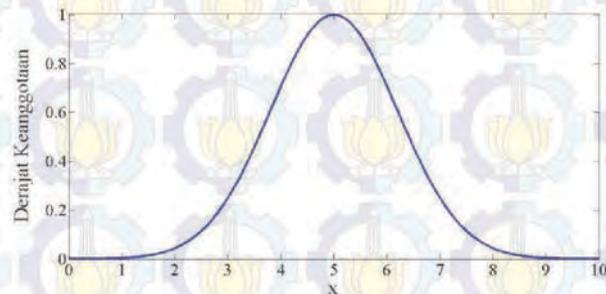
$$\mu_{\text{gbell}}(x) = \frac{1}{1 + \left| \frac{x-a}{c} \right|^{2b}} \quad (2.22)$$

Sedangkan persamaan fungsi keanggotaan *gaussian* dapat dilihat pada (2.23) dan grafiknya pada Gambar 2.15. Fungsi keanggotaan *Gaussian* ditentukan oleh dua parameter $\{c, \sigma\}$. Parameter c menyatakan titik pusat dengan derajat keanggotaan bernilai 1 dan parameter σ menyatakan lebar fungsi *gaussian*.

$$\mu_{\text{gaussian}}(x) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2} \quad (2.23)$$



Gambar 2.14 Fungsi Keanggotaan *Generalized Bell* dengan $\{a, b, c\} = \{3, 4, 5\}$



Gambar 2.15 Fungsi Keanggotaan *Gaussian* dengan $\{c, \sigma\} = \{1.2, 5\}$

2.5.1 Mekanisme Inferensi

Mekanisme inferensi adalah proses implikasi dalam menalar nilai masukan untuk menentukan nilai keluaran sebagai bentuk pengambil keputusan. Dibutuhkan beberapa basis aturan dalam menyelesaikan permasalahan kontrol *fuzzy* ini. Aturan *fuzzy* memetakan masukan *fuzzy* yang diperoleh dari fuzzifikasi terhadap keluaran *fuzzy* yang berupa pernyataan premis *If-Then* seperti berikut.

If premis *Then* konsekuen

If $x_1 = A$ AND $x_2 = B$ *Then* $u = C$

Mekanisme inferensi ini sendiri terdiri dari operasi himpunan *fuzzy*, aplikasi metode implikasi, dan agregasi semua *output*. Aplikasi operator *fuzzy* dilakukan karena pernyataan premis dapat melibatkan lebih dari satu variabel premis. Terdapat tiga operasi dasar pada himpunan *fuzzy*, yaitu irisan (*intersection*), gabungan (*union*), dan komplemen. Masing-masing operasi dasar pada himpunan *fuzzy* didefinisikan sebagai berikut:

1. *Intersection* (AND)

Misalkan A dan B adalah dua himpunan *fuzzy* yang berbeda dengan fungsi keanggotaan $\mu_A(x)$ dan $\mu_B(x)$. Irisan (*intersection*) dari himpunan *fuzzy* A dan B adalah himpunan *fuzzy* yang dinyatakan $A \cap B$ dengan fungsi keanggotaan yang dapat didefinisikan oleh dua metode berikut:

a. *Minimum*

Operasi *minimum* dapat dinyatakan sebagai berikut:

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x) : x \in X\} \quad (2.24)$$

b. *Algebraic Product*

Operasi *product* dapat dinyatakan sebagai berikut:

$$\mu_{A \cap B}(x) = \{\mu_A(x)\mu_B(x) : x \in X\} \quad (2.25)$$

2. *Union* (OR)

Misalkan A dan B adalah dua himpunan *fuzzy* yang berbeda dengan fungsi keanggotaan $\mu_A(x)$ dan $\mu_B(x)$. Gabungan (*union*) dari himpunan *fuzzy* A dan B adalah himpunan *fuzzy* yang dinyatakan $A \cup B$ dengan fungsi keanggotaan yang dapat didefinisikan oleh dua metode berikut:

a. *Maximum*

Operasi *maximum* dinyatakan sebagai berikut:

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x) : x \in X\} \quad (2.26)$$

b. *Algebraic Sum*

Operasi *algebraic sum* dinyatakan sebagai berikut:

$$\mu_{A \cup B}(x) = \{\mu_A(x) + \mu_B(x) - \mu_A(x)\mu_B(x) : x \in X\} \quad (2.27)$$

3. Komplemen

Komplemen dari himpunan *fuzzy* A dengan fungsi keanggotaan $\mu_A(x)$ memiliki fungsi keanggotaan sebagai berikut:

$$\mu_{A'}(x) = \{1 - \mu_A(x) : x \in X\} \quad (2.28)$$

Aplikasi metode implikasi memetakan pernyataan premis yang memenuhi sehingga didapatkan pernyataan konsekuen. Nilai dari hasil operasi himpunan *fuzzy* di pernyataan premis menyatakan seberapa besar pernyataan konsekuen tersebut dapat bekerja. Beberapa macam implikasi pernyataan konsekuen adalah Model Mamdani dan Model Sugeno.

Proses agregasi semua *output* adalah proses penggabungan semua hasil dari aturan-aturan tersebut. Karena nilai keluaran tiap aturan sudah merupakan nilai tegas, maka metode agregasi hanya menjumlahkan semua keluaran tersebut.

2.5.2 Defuzzifikasi

Defuzzifikasi adalah proses pemetaan himpunan *fuzzy* ke himpunan tegas. Terdapat beberapa metode defuzzifikasi, antara lain *center of gravity* (COG), *center-weighted average*, dan *center of area* (COA). Pada metode COG, *output* tegas dipilih dengan mempertimbangkan besarnya nilai titik tengah fungsi keanggotaan dan area dibawah fungsi keanggotaan. *Output* tegas metode COG dapat dilihat pada (2.29)

$$u_{\text{tegas}} = \frac{\sum_i b_i \int \mu_i(y) dy}{\sum_i \int \mu_i(y) dy} \quad (2.29)$$

dengan b_i adalah titik tengah fungsi keanggotaan dan $\int \mu_i$ adalah area dibawah fungsi keanggotaan. Pada metode *center-weighted average*, *output* tegas dipilih dengan mempertimbangkan besarnya nilai titik tengah fungsi keanggotaan dan nilai bobot setiap aturan. *Output* tegas metode ini dapat dilihat pada (2.30).

$$u_{\text{tegas}} = \frac{\sum_i b_i \mu_i}{\sum_i \mu_i} \quad (2.30)$$

dengan b_i adalah titik tengah fungsi keanggotaan dan μ_i adalah nilai bobot setiap aturan. Sedangkan pada metode COA, perhitungan *output* tegas dapat dilihat pada (2.31).

$$u_{\text{tegas}} = \frac{\int u \mu du}{\int \mu du} \quad (2.31)$$

dengan u adalah keluaran dan μ adalah derajat keanggotaan yang bersesuaian dengan nilai u .

2.6 Model Fuzzy Takagi Sugeno (T-S)

Dinamika dari model *fuzzy* Takagi-Sugeno (T-S) dapat merepresentasikan hubungan linear masukan-keluaran dari sistem yang nonlinear. Model *fuzzy* T-S merupakan kombinasi antara fungsi kualitatif pada bagian premis dan fungsi kuantitatif pada bagian konsekuen. Model kuantitatif yang digunakan dalam bagian konsekuen berupa model linear dinamik. Kombinasi dengan aturan logika pada bagian premis membuat model ini dapat dilihat sebagai model multi linear lokal.

Premis dari model *fuzzy* T-S merupakan model *plant* yang ditulis dalam bentuk aturan *If-Then* sebagai berikut:

Aturan *plant* ke- i :

If z_1 is M_{i1} and z_2 is $M_{i2} \dots z_j$ is M_{ij}

Then $\dot{\mathbf{x}} = \mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{u}$

$$\mathbf{y} = \mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{u} \quad (2.32)$$

dengan $i = 1, 2, \dots, q$ adalah banyaknya aturan *fuzzy*, j adalah banyaknya himpunan *fuzzy* dalam satu aturan, M_{ij} adalah himpunan *fuzzy*, z adalah variabel pada bagian premis, dan u adalah masukan sistem. Vektor *state*

$x \in R^n$, vektor keluaran sistem $y \in R^q$, sedangkan A, B, C , dan D adalah matriks dengan ukuran yang sesuai.

Pada tahap inferensi model *fuzzy* T-S menggunakan logika penghubung AND (operator *product*), sedangkan pada tahap defuzzifikasi menggunakan metode *center average*. Sehingga secara keseluruhan, model *fuzzy* T-S dapat dituliskan sebagai berikut:

$$\left. \begin{aligned} \mathbf{x} &= \sum_{i=1}^q m_i(z) (\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{u}) \\ \mathbf{y} &= \sum_{i=1}^q m_i(z) (\mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{u}) \end{aligned} \right\} \quad (2.33)$$

dengan

$$\left. \begin{aligned} \mu_i(z) &= \prod_{j=1}^p M_{ij}(z_j) \\ m_i(z) &= \frac{\mu_i(z)}{\sum_{i=1}^r \mu_i(z)} \end{aligned} \right\} \quad (2.34)$$

Pembobot $m_i[z]$ dan derajat keanggotaan $\mu_i(z)$ memiliki sifat sebagai berikut:

$$m_i(z) \geq 0; \sum_{i=1}^q m_i(z) = 1; \mu_i(z) \geq 0; \sum_{i=1}^q \mu_i(z) > 0 \quad (2.35)$$

Untuk kontroler *fuzzy*, pada proses pembobotan serta defuzzifikasinya sama. Nilai *gain* kontrol K dapat dicari dengan menggunakan metode LQR. Sehingga bentuk aturannya menjadi:

Aturan kontroler ke- i :

If z_1 is M_{i1} and z_2 is $M_{i2} \dots z_j$ is M_{ij}

$$\text{Then } \mathbf{u} = -\mathbf{K}_i \mathbf{x} \quad (2.36)$$

dengan $i = 1, 2, \dots, q$. Sehingga keluaran kontroler *fuzzy* menjadi:

$$\mathbf{u} = \sum_{i=1}^r m_i(z) (-\mathbf{K}_i \mathbf{x}) \quad (2.37)$$

2.7 Linear Quadratic Regulator (LQR) [2]

Salah satu metode untuk mendapatkan *state feedback gain* dalam pengendalian kestabilan sistem pendulum kereta yaitu *Linear Quadratic Regulator*. LQR disebut linear karena model dan kontrolernya berupa linear. Sedangkan disebut kuadratik karena *cost function* dari sistem merupakan kuadratik dan karena referensinya bukan berupa fungsi waktu maka disebut regulator. Sistem kontrol dikatakan baik, apabila sistem kontrol tersebut memiliki daya tanggap yang cepat dan stabil, tetapi tidak membutuhkan energi yang berlebihan.

Sistem kontrol dengan karakteristik tersebut dapat dicapai dengan menggunakan indeks performa yang tepat. Sistem kontrol yang dirancang berdasarkan optimasi indeks performansi disebut sistem kontrol optimal. Indeks performa dalam interval $[t_0, \infty)$ diupayakan dapat meminimalisasi energi (*cost function*).

Pada kasus *infinite interval* LQR - *time invariant* terdapat model *plant* sebagai berikut:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (2.38)$$

dengan indeks performa

$$J = \frac{1}{2} \int_0^{\infty} (\mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t))dt \quad (2.39)$$

Matriks $\mathbf{Q} \in \mathcal{R}^{n \times n}$ yang merupakan matriks semidefinit positif. Matriks $\mathbf{R} \in \mathcal{R}^{r \times r}$ adalah matriks definit positif. Sedangkan matriks \mathbf{A} dan \mathbf{B} dengan ukuran yang sesuai.

Terdapat beberapa langkah untuk menentukan solusi optimal dengan prosedur Pontryagin sebagai berikut:

- Menyelesaikan matriks *Algebraic Riccati Equation* (ARE)

$$-\mathbf{P}\mathbf{A} - \mathbf{A}^T\mathbf{P} - \mathbf{Q} + \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = 0 \quad (2.40)$$

- Menyelesaikan *state optimal* $\mathbf{x}^*(t)$

$$\dot{\mathbf{x}}^*(t) = (\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P})\mathbf{x}^*(t) \quad (2.41)$$

dengan kondisi awal $\mathbf{x}(t_0) = \mathbf{x}_0$

- Mendapatkan kontrol optimal $\mathbf{u}^*(t)$

$$\mathbf{u}^*(t) = -\mathbf{K}\mathbf{x}^*(t) \quad (2.42)$$

dengan $\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$

- Mendapatkan indeks performa optimal

$$J^*(t) = \frac{1}{2}\mathbf{x}^{*T}(t)\mathbf{P}\mathbf{x}^*(t) \quad (2.43)$$

2.8 Proportional Multiple Integral Observer (PMIO) [3]

Observer digunakan untuk mendeteksi adanya kesalahan. Untuk kesalahan yang terjadi pada aktuator, dapat menggunakan metode PMIO untuk mengestimasi kesalahan tersebut. Dengan model sistem *fuzzy* T-S seperti pada (2.33), kemudian dikembangkan menjadi sistem augmentasi dengan melibatkan turunan sinyal kesalahan ke- s , f_a^s .

$$\left. \begin{aligned} \dot{\mathbf{x}}_a &= \sum_{i=1}^r m_i(z) (\bar{\mathbf{A}}_i \mathbf{x}_a + \bar{\mathbf{B}}_i \mathbf{u}) + \bar{\mathbf{G}} f_a^{(s)} \\ \mathbf{y}_a &= \bar{\mathbf{C}} \mathbf{x}_a \end{aligned} \right\} \quad (2.44)$$

dengan:

$$\varphi_i = f_a^{(s-i)}, (i = 1, 2, \dots, s), \varphi_s = f_a^{(s)}, \varphi_1 = \varphi_1, \mathbf{K}, \varphi_s = \varphi_{s-1}$$

$$\mathbf{x}_a = [\mathbf{x}^T \quad \varphi_1 \quad \varphi_2 \quad \mathbf{K} \quad \varphi_s]^T \in \mathcal{R}^{\bar{n}}$$

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_{n \times n} & \mathbf{0}_{n \times (s-1)} & \mathbf{B}_{n \times 1} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times (s-1)} & \mathbf{0}_{1 \times 1} \\ \mathbf{0}_{(s-1) \times n} & \mathbf{I}_{(s-1) \times (s-1)} & \mathbf{0}_{(s-1) \times 1} \end{bmatrix}$$

$$\bar{\mathbf{B}} = [\mathbf{B}_{n \times m}^T \quad \mathbf{0}_{s \times m}]^T$$

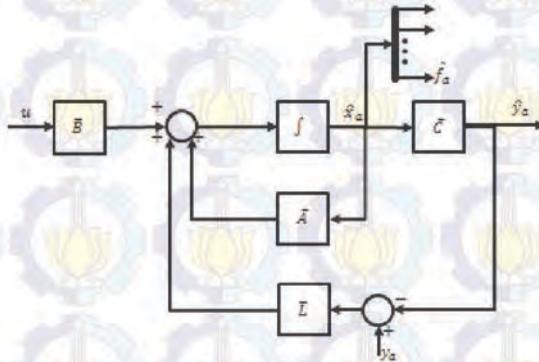
$$\bar{\mathbf{G}} = [\mathbf{0}_{n \times 1} \quad \mathbf{I}_{1 \times 1} \quad \mathbf{0}_{(s-1) \times 1}]^T$$

$$\bar{\mathbf{C}} = [\mathbf{C}_{q \times n} \quad \mathbf{0}_{q \times s}]$$

Dibawah ini adalah model *fuzzy* T-S PMIO yang digunakan untuk mengestimasi *state* sistem dan kesalahan aktuator secara simultan, serta secara grafis blok diagram dapat dilihat pada (2.45).

$$\left. \begin{aligned} \dot{\hat{x}}_a &= \sum_{i=1}^q m_i(z) (\bar{A}_i \hat{x}_a + \bar{B}_i u + \bar{L}_i (y_a - \hat{y}_a)) \\ \hat{y}_a &= \bar{C} \hat{x}_a \end{aligned} \right\} \quad (2.45)$$

dengan \bar{L}_i adalah *observer gain* hasil desain.



Gambar 2.16 Blok Diagram Estimasi Kesalahan dengan PMIO

Akan tetapi dalam desain *Fuzzy* T-S PMIO harus memenuhi syarat berikut ini:

$$\text{rank} \begin{bmatrix} A_i & B_i \\ C & 0 \end{bmatrix} = n + k \quad (2.46)$$

dengan k adalah ukuran vektor kesalahan aktuator. Selain itu, PMIO akan melemahkan efek kesalahan aktuator yang dibatasi turunan estimasi kesalahan jika terdapat matriks $\bar{P} = \bar{P}^T > 0$ dan matriks \bar{H}_i yang meminimalisasi nilai γ_{pmi} dengan konstrain LMI yang ditunjukkan pada (2.47).

$$\begin{bmatrix} \bar{\mathbf{P}}\bar{\mathbf{A}}_i + (\bar{\mathbf{P}}\bar{\mathbf{A}}_i)^T - \bar{\mathbf{H}}_i\bar{\mathbf{C}} - (\bar{\mathbf{H}}_i\bar{\mathbf{C}})^T & \bar{\mathbf{P}}\bar{\mathbf{G}} & \mathbf{C}_p^T \\ \bar{\mathbf{G}}^T\bar{\mathbf{P}} & -\gamma_{pmi}\mathbf{I} & 0 \\ \mathbf{C}_p & 0 & -\gamma_{pmi}\mathbf{I} \end{bmatrix} < 0 \quad (2.47)$$

dengan *observer gain* didapatkan dengan: $\bar{\mathbf{L}}_i = \bar{\mathbf{P}}^{-1}\bar{\mathbf{H}}_i$.

2.9 Analisis Kestabilan Lyapunov [6]

Kestabilan merupakan salah satu hal yang paling penting dalam suatu sistem kontrol. Pada Tugas Akhir ini menggunakan metode kedua Lyapunov untuk analisis kestabilan. Hal ini dikarenakan metode kedua Lyapunov dapat menganalisis kestabilan dari suatu sistem dapat ditentukan tanpa harus menyelesaikan persamaan *state* dari sistem tersebut.

Fungsi Lyapunov bergantung pada x_1, x_2, \dots, x_n , dan t yang dapat dinyatakan dengan $V(x_1, x_2, \dots, x_n, t)$ atau $V(\mathbf{x}, t)$. Akan tetapi, jika fungsi Lyapunov tidak bergantung pada t secara eksplisit, maka fungsi Lyapunov dapat dinyatakan dengan $V(x_1, x_2, \dots, x_n)$ atau $V(\mathbf{x})$. Dalam metode kedua Lyapunov, perilaku $V(\mathbf{x}, t)$ dan turunan pertamanya terhadap waktu $\dot{V}(\mathbf{x}, t) = dV(\mathbf{x}, t)/dt$ memberikan informasi mengenai kestabilan, kestabilan asimtotik, atau ketidakstabilan dari suatu *state* ekuilibrium tanpa harus mendapatkan solusi persamaan diferensial dari sistem. Suatu sistem dinyatakan dengan persamaan:

$$\dot{\mathbf{x}} = f(\mathbf{x}, t) \quad (2.48)$$

dengan $f(0, t) = 0$ untuk semua t . Jika terdapat fungsi skalar $V(\mathbf{x}, t)$ yang mempunyai turunan parsial pertama yang kontinu dan memenuhi kondisi:

1. $V(\mathbf{x}, t)$ adalah definit positif
2. $\dot{V}(\mathbf{x}, t)$ adalah definit negatif

maka *state* ekuilibrium pada *origin* adalah stabil asimtotik.

Jika terdapat sistem linear *time-invariant* dengan persamaan:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} \quad (2.49)$$

dengan \mathbf{x} adalah vektor *state* $n \times 1$, \mathbf{A} adalah matriks konstan *nonsingular* $n \times n$, dan *state* ekuilibrium berada pada *origin* ($x = 0$).

Kestabilan dari *state* ekuilibrium sistem (2.49) dapat diketahui dengan mudah menggunakan metode kedua Lyapunov. Untuk sistem (2.49), kemungkinan fungsi Lyapunov dipilih sebagai berikut:

$$V(x) = x^T \bar{P} x \quad (2.50)$$

dengan \bar{P} adalah matriks simetris definit positif. Turunan pertama terhadap waktu dari $V(x)$ adalah sebagai berikut:

$$\begin{aligned} \dot{V}(x) &= \dot{x}^T \bar{P} x + x^T \bar{P} \dot{x} \\ &= (Ax)^T \bar{P} x + x^T \bar{P} (Ax) \\ &= x^T A^T \bar{P} x + x^T \bar{P} A x \\ &= x^T (A^T \bar{P} + \bar{P} A) x \end{aligned} \quad (2.51)$$

Karena $V(x)$ dipilih definit positif, untuk kestabilan asimtotik, maka $\dot{V}(x)$ harus definit negatif. Oleh karena itu,

$$\dot{V}(x) = -x^T Q x \quad (2.52)$$

dengan

$$Q = -(A^T \bar{P} + \bar{P} A) \quad (2.53)$$

dan Q = definit positif. Sehingga untuk menjamin kestabilan sistem (2.49) harus terdapat matriks simetris \bar{P} yang memenuhi:

$$\left. \begin{aligned} \bar{P} &> 0 \\ A^T \bar{P} + \bar{P} A &< 0 \end{aligned} \right\} \quad (2.54)$$

2.10 Linear Matrix Inequalities (LMI) [8]

Sesuai dengan analisis Lyapunov bahwa sistem akan stabil saat terdapat matriks \bar{P} yang memenuhi (2.54). Matriks \bar{P} dapat dicari nilainya dengan perhitungan analitik untuk sistem yang sederhana. Pertidaksamaan Lyapunov tersebut dapat diselesaikan dengan memilih $Q = Q^T > 0$ yang digunakan untuk menyelesaikan permasalahan $A^T \bar{P} + \bar{P} A = -Q$ untuk mendapatkan matriks definit positif \bar{P} jika sistem (2.49) stabil. Akan tetapi, jika sistem tersebut adalah mempunyai tingkat kompleksitas yang tinggi, maka penyelesaian dengan cara LMI

merupakan pilihan yang paling baik. LMI memiliki bentuk sebagai berikut:

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}_0 + \sum_{i=1}^m \mathbf{x}_i \mathbf{F}_i > 0 \quad (2.55)$$

dengan $\mathbf{x} \in \mathbf{R}^m$ adalah variabel dan $\mathbf{F}_i = \mathbf{F}_i^T \in \mathbf{R}^{n \times n}$, $i = 0, \dots, m$, adalah matriks simetris yang diketahui. Tanda pertidaksamaan pada (2.55) menunjukkan bahwa $\mathbf{F}(\mathbf{x})$ adalah definit positif. LMI (2.55) merupakan fungsi kendala dari \mathbf{x} sehingga tujuan optimisasi dari LMI (2.55) adalah mencari nilai \mathbf{x} sedemikian hingga $\mathbf{F}(\mathbf{x}) > 0$.

Beberapa LMI, $\mathbf{F}^{(1)}(\mathbf{x}) > 0, \dots, \mathbf{F}^{(p)}(\mathbf{x}) > 0$, dapat dinyatakan sebagai LMI tunggal $\text{diag}(\mathbf{F}^{(1)}(\mathbf{x}), \dots, \mathbf{F}^{(p)}(\mathbf{x})) > 0$. Sehingga jika matriks \mathbf{F}_i adalah diagonal, maka LMI $\mathbf{F}(\mathbf{x}) > 0$ dapat dikatakan sebagai kumpulan beberapa pertidaksamaan linear.

2.10.1 Schur Complement

Pertidaksamaan nonlinear dapat diubah ke dalam bentuk LMI dengan menggunakan *Schur Complement*. Misalkan terdapat pertidaksamaan seperti pada (2.56)

$$\begin{aligned} \mathbf{R}(\mathbf{x}) &> 0 \\ \mathbf{Q}(\mathbf{x}) - \mathbf{S}(\mathbf{x})\mathbf{R}(\mathbf{x})^{-1}\mathbf{S}(\mathbf{x})^T &> 0 \end{aligned} \quad (2.56)$$

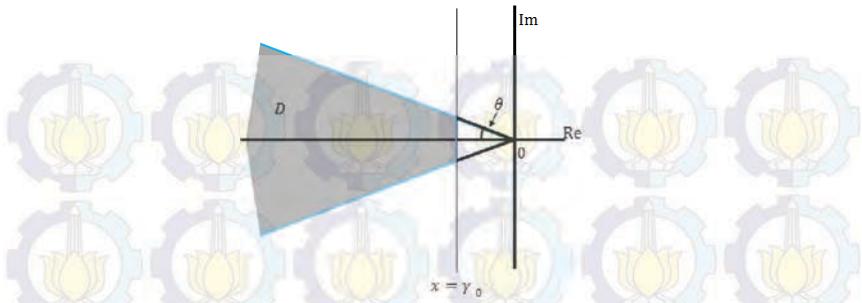
maka dapat diubah ke dalam bentuk LMI menjadi seperti pada (2.57)

$$\begin{bmatrix} \mathbf{Q}(\mathbf{x}) & \mathbf{S}(\mathbf{x}) \\ \mathbf{S}(\mathbf{x})^T & \mathbf{R}(\mathbf{x}) \end{bmatrix} > 0 \quad (2.57)$$

dengan $\mathbf{Q}(\mathbf{x}) = \mathbf{Q}(\mathbf{x})^T$ dan $\mathbf{R}(\mathbf{x}) = \mathbf{R}(\mathbf{x})^T$.

2.10.2 LMI Region [9]

Stabilitas adalah persyaratan yang harus dipenuhi dalam sistem kontrol apapun. Akan tetapi, kontroler yang baik mampu memberikan respon yang cepat yang dapat mudah dicapai dengan menempatkan *closed loop poles* pada daerah D yang diinginkan, seperti yang ditunjukkan pada Gambar 2.17. *Time settling* dan *overshoot* bergantung pada pemilihan besarnya nilai *damping ratio*, yaitu $\cos\theta$. Sedangkan kecepatan respon sistem bergantung pada besarnya nilai γ_0 .



Gambar 2.17 Region D pada Bidang Kompleks

Daerah D pada bidang kompleks disebut LMI *region* jika terdapat matriks simetris \mathbf{L} dan matriks \mathbf{M} sehingga:

$$D = z \in \mathbb{C}: f_d(z) < 0 \quad (2.58)$$

dengan $f_d(z) = \mathbf{L} + z\mathbf{M} + \bar{z}\mathbf{M}^T$ dan $f_d(z)$ disebut fungsi karakteristik dari daerah D . Beberapa contoh LMI *region* adalah:

- Setengah bidang dengan $Re(z) = -\alpha$

$$f_d(z) = z + \bar{z} + 2\alpha < 0 \quad (2.59)$$

- Lingkaran dengan pusat $(-q, 0)$ dan jari jari r

$$f_d(z) = \begin{bmatrix} -r & q+z \\ z+\bar{z} & -r \end{bmatrix} < 0 \quad (2.60)$$

- *Conic* 2θ

$$f_d(z) = \begin{bmatrix} \sin\theta(z+\bar{z}) & \cos\theta(z-\bar{z}) \\ \cos\theta(z-\bar{z}) & \sin\theta(z+\bar{z}) \end{bmatrix} < 0 \quad (2.61)$$

2.11 Performa H_∞

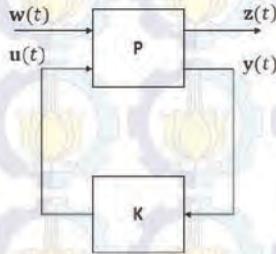
Observer didesain selain untuk menjamin kestabilan juga untuk melemahkan efek kesalahan aktuator. Pada metode ini, pengaruh gangguan terburuk dari luar (kesalahan aktuator) pada performa sistem akan dilemahkan hingga mencapai nilai pelemahan yang telah ditentukan. Pengaruh terburuk suatu gangguan pada sistem dengan

fungsi alih $G(s)$ dapat diketahui dengan menghitung ∞ -norm atau sama dengan menghitung nilai maksimal dari magnitude respon frekuensinya. Sehingga secara matematika dapat dirumuskan sebagai (2.62).

$$\|G(s)\|_{\infty} = \sup_{\omega} |G(j\omega)| \quad (2.62)$$

Secara umum, konfigurasi sistem kontrol untuk mendapatkan performa H_{∞} dijelaskan seperti pada Gambar 2.18. Matriks \mathbf{P} dan \mathbf{K} merupakan sistem dan kontroler. Sinyal $\mathbf{u}(t)$ dan $\mathbf{w}(t)$ merupakan sinyal kontrol dan gangguan. Sedangkan $\mathbf{y}(t)$ dan $\mathbf{z}(t)$ masing-masing adalah keluaran terukur dan keluaran performa dari sistem.

Dalam tugas akhir ini, performa H_{∞} akan diterapkan pada estimator kesalahan. oleh karena itu, diharapkan observer dapat melemahkan efek dari adanya kesalahan aktuator.



Gambar 2.17 Skema Kontrol untuk Mendapatkan Performa H_{∞}

Untuk memperjelas prinsip dari performa H_{∞} diasumsikan terdapat vektor *error* estimasi $\mathbf{e}_x(t) = \mathbf{x}_a(t) - \hat{\mathbf{x}}_a(t)$ yang mengalami gangguan kesalahan aktuator $f_a^{(s)}(t)$. Keluaran performa yang diharapkan adalah

$$\mathbf{z}(t) = \mathbf{C}_p \mathbf{e}_x(t) \quad (2.63)$$

dengan \mathbf{C}_p adalah matriks performa. Perhitungan tingkat pelemahan (γ) dapat dilakukan menggunakan pertidaksamaan (2.64)

$$\frac{\|\mathbf{z}(t)\|_2}{\|\mathbf{w}(t)\|_2} = \frac{\sqrt{\int_0^{t_f} \mathbf{e}_x(t)^T \bar{\mathbf{Q}} \mathbf{e}_x(t) dt}}{\sqrt{\int_0^{t_f} \mathbf{w}(t)^T \mathbf{w}(t) dt}} \leq \gamma_{pmi}, \quad \bar{\mathbf{Q}} = \mathbf{C}_p^T \mathbf{C}_p \quad (2.64)$$

Dengan mengintegalkan kedua ruas dari pertidaksamaan (2.64) akan diperoleh :

$$\frac{\int_0^{t_f} \mathbf{e}_x(t)^T \bar{\mathbf{Q}} \mathbf{e}_x(t) dt}{\int_0^{t_f} \mathbf{w}(t)^T \mathbf{w}(t) dt} \leq \gamma^2 \quad (2.65)$$

atau

$$\frac{1}{\gamma} \int_0^{t_f} \mathbf{e}_x(t)^T \bar{\mathbf{Q}} \mathbf{e}_x(t) dt - \gamma \int_0^{t_f} \mathbf{w}(t)^T \mathbf{w}(t) dt \leq 0 \quad (2.66)$$

dengan $\mathbf{w}(t) = f_a^{(s)}(t)$

Arti fisik dari (2.66) adalah efek terburuk dari $\mathbf{w}(t)$ pada *error* estimasi $\mathbf{e}_x(t) = \mathbf{x}_a(t) - \bar{\mathbf{x}}_a(t)$ harus dilemahkan sampai di bawah tingkat pelemahan (γ) yang ditentukan. Adapun ∞ -norm dari fungsi alih $G(s)$ merupakan nilai maksimum dari (2.64). Sesuai dengan definisi ini maka ∞ -norm dari fungsi alih sistem dinyatakan sebagai

$$\frac{\|\mathbf{Z}(s)\|_\infty}{\|\mathbf{W}(s)\|_\infty} = \sup_\omega \left| \frac{\mathbf{Z}(j\omega)}{\mathbf{W}(j\omega)} \right| = \sup_{\|\mathbf{w}(t)\|_2 \neq 0} \frac{\|\mathbf{z}(t)\|_2}{\|\mathbf{w}(t)\|_2} \quad (2.67)$$

Jika tingkat pelemahan maksimal yang diinginkan adalah kurang dari γ^* , maka performa dari sistem dinyatakan dalam pertidaksamaan (2.68).

$$\frac{\|\mathbf{Z}(s)\|_\infty}{\|\mathbf{W}(s)\|_\infty} = \sup_{\|\mathbf{w}(t)\|_2 \neq 0} \frac{\|\mathbf{z}(t)\|_2}{\|\mathbf{w}(t)\|_2} = \gamma^* < \gamma \quad (2.68)$$

BAB 3

PERANCANGAN SISTEM KONTROL

Pada bab ini akan dijelaskan mengenai linearisasi model matematika nonlinear sistem pendulum kereta pada titik kerjanya. Setelah itu dirancang kontroler nominal saat kondisi sistem tanpa adanya kesalahan pada aktuator. *State feedback gain* didapatkan dari perhitungan menggunakan LQR. Untuk menjaga stabilisasi sistem pendulum kereta saat terjadi kesalahan pada aktuator, maka dirancang *observer* dengan pendekatan *Proportional Multiple Integral Observer* yang mampu mengestimasi kesalahan yang terjadi pada aktuator. Pada akhir Bab ini, perancangan sistem menggunakan *Simulink/MATLAB* akan ditunjukkan untuk simulasi dan implementasi pada sistem pendulum kereta.

3.1 Linearisasi Model *Plant*

Sistem pendulum kereta adalah sistem nonlinear. Agar sistem kontrol linear dapat digunakan, maka sistem tersebut dilinearisasi terlebih dahulu. Untuk mendapatkan model matematika yang linear, maka dari Persamaan (2.17) didapatkan:

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

dengan

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ f_3(\mathbf{x}) \\ f_4(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} x_3 \\ x_4 \\ \frac{\alpha(-T_c - \mu x_4^2 \sin x_2) + l \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2} \\ \frac{l \cos x_2 (-T_c - \mu x_4^2 \sin x_2) + (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2} \end{bmatrix}$$

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

Berdasarkan teori linearisasi yang telah dijelaskan pada Bab II, maka didapatkan model linear sistem pendulum kereta menjadi seperti berikut ini:

$$\mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

dengan

$$\mathbf{A} = \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \frac{\partial f_1(\mathbf{x})}{\partial x_2} & \frac{\partial f_1(\mathbf{x})}{\partial x_3} & \frac{\partial f_1(\mathbf{x})}{\partial x_4} \\ \frac{\partial f_2(\mathbf{x})}{\partial x_1} & \frac{\partial f_2(\mathbf{x})}{\partial x_2} & \frac{\partial f_2(\mathbf{x})}{\partial x_3} & \frac{\partial f_2(\mathbf{x})}{\partial x_4} \\ \frac{\partial f_3(\mathbf{x})}{\partial x_1} & \frac{\partial f_3(\mathbf{x})}{\partial x_2} & \frac{\partial f_3(\mathbf{x})}{\partial x_3} & \frac{\partial f_3(\mathbf{x})}{\partial x_4} \\ \frac{\partial f_4(\mathbf{x})}{\partial x_1} & \frac{\partial f_4(\mathbf{x})}{\partial x_2} & \frac{\partial f_4(\mathbf{x})}{\partial x_3} & \frac{\partial f_4(\mathbf{x})}{\partial x_4} \end{bmatrix}_{x_1=x_1^*}$$

$$\mathbf{B} = \begin{bmatrix} \frac{\partial h_1(\mathbf{x}, u)}{\partial u} \\ \frac{\partial h_2(\mathbf{x}, u)}{\partial u} \\ \frac{\partial h_3(\mathbf{x}, u)}{\partial u} \\ \frac{\partial h_4(\mathbf{x}, u)}{\partial u} \end{bmatrix}_{\substack{x=x^* \\ u=u^*}}$$

untuk masing-masing elemen matriks \mathbf{A} dan \mathbf{B} dapat dilihat pada Lampiran A.1.2.

Dari penurunan rumus pada Persamaan di Lampiran tersebut dapat dilihat bahwa nilai dari x_1 tidak mempengaruhi dari model matematika sistem pendulum kereta, oleh karena itu linearisasi *plant* dapat dilakukan pada variasi nilai x_2 saja. Variasi nilai x_2 yang dipilih menjadi titik kerja *plant* pada perancangan sistem ini adalah di 0 radian dan $\pm 0,2$ radian.

Untuk titik kerja pertama yaitu $\mathbf{x}^* = [0 \ 0 \ 0 \ 0]^T$ dan $u^* = 0$, didapatkan:

$$\mathbf{x} = \mathbf{A}_1\mathbf{x} + \mathbf{B}_1u \quad (3.1)$$

dengan

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2522 & 0 & -0,0001 \\ 0 & 15,0113 & 0 & -0,0074 \end{bmatrix}, \mathbf{B}_1 = \begin{bmatrix} 0 \\ 0 \\ 0,8272 \\ 1,2353 \end{bmatrix}$$

Untuk titik kerja kedua yaitu $\mathbf{x}^* = [0 \quad \pm 0,2 \quad 0 \quad 0]^T$ dan $u^* = 0$, didapatkan :

$$\dot{\mathbf{x}} = \mathbf{A}_2 \mathbf{x} + \mathbf{B}_2 u \quad (3.2)$$

dengan

$$\mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2316 & 0 & -0,0001 \\ 0 & 14,6673 & 0 & -0,0073 \end{bmatrix}, \mathbf{B}_2 = \begin{bmatrix} 0 \\ 0 \\ 0,8264 \\ 1,2094 \end{bmatrix}$$

Sedangkan untuk matriks yang digunakan sebagai keluaran dan matriks transmisi masukan langsung ke keluaran untuk kedua titik kerja tersebut adalah:

$$\mathbf{C}_1 = \mathbf{C}_2 = [1 \quad 0 \quad 0 \quad 0], \mathbf{D}_1 = \mathbf{D}_2 = 0$$

Persamaan (3.1) dan (3.2) adalah model linear untuk sistem pendulum kereta. Algoritma yang digunakan untuk mendapatkan model tersebut dapat dilihat pada Lampiran B.1.

3.2 Perancangan Kontrol *Linear Quadratic Regulator* (LQR) untuk Stabilisasi

Stabilisasi pada sistem pendulum kereta adalah proses mempertahankan posisi sudut pendulum pada kondisi terbalik. Sistem kontrol yang digunakan untuk stabilisasi pada Tugas Akhir ini adalah *Linear Quadratic Regulator* (LQR). Sistem kontrol LQR digunakan untuk mengoptimalkan *cost function* berdasarkan indeks performa yang diinginkan. Untuk kasus *infinite time invariant Linear Quadratic Regulator* mempunyai indeks performa:

$$J = \frac{1}{2} \int_0^{\infty} (\mathbf{x}^T(t) \mathbf{Q} \mathbf{x}(t) + u^T(t) \mathbf{R} u(t)) dt \quad (3.3)$$

Agar mendapatkan nilai optimal dari *cost function* sesuai dengan indeks performa tersebut, maka harus memenuhi persamaan aljabar riccati seperti pada (3.3) dengan mencari kombinasi matriks \mathbf{Q} dan \mathbf{R} .

$$-\mathbf{P} \mathbf{A} - \mathbf{A}^T \mathbf{P} - \mathbf{Q} + \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} = 0 \quad (3.4)$$

Dengan menyelesaikan (3.4), didapatkan matriks \mathbf{P} yang digunakan untuk mencari nilai *state feedback gain controller* seperti pada (3.5) agar sistem pendulum kereta tercapai kestabilan.

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (3.5)$$

Pada perancangan kontrol nominal yaitu menggunakan *state feedback gain controller* menggunakan LQR berbasis *fuzzy T-S* dipengaruhi oleh pemilihan matriks \mathbf{Q} dan \mathbf{R} . Untuk Tugas Akhir ini dipilih nilai matriks \mathbf{Q} dan \mathbf{R} yaitu:

$$\mathbf{Q} = \begin{bmatrix} 5000 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 \\ 0 & 0 & 500 & 0 \\ 0 & 0 & 0 & 500 \end{bmatrix}$$

$$\mathbf{R} = 0,5$$

yang menggunakan dua titik kerja yang sudah dijelaskan pada Subbab 3.1 yaitu pada $x_2^* = 0$ radian dan $x_2^* = \pm 0,2$ radian. Dengan menyelesaikan persamaan aljabar *riccati* seperti pada (3.4), maka didapatkan nilai *state feedback gain* \mathbf{K} yang didapatkan dari hasil linearisasi dari dua titik kerja.

Untuk titik kerja yang pertama yaitu pada $x_2^* = 0$ radian:

$$\mathbf{x}^* = [0 \ 0 \ 0 \ 0]^T \text{ dan } u^* = 0$$

sehingga diperoleh:

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

dengan

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2522 & 0 & -0,0001 \\ 0 & 15,0113 & 0 & -0,0074 \end{bmatrix}, \mathbf{B}_1 = \begin{bmatrix} 0 \\ 0 \\ 0,8272 \\ 1,2353 \end{bmatrix}$$

Didapatkan matriks \mathbf{K}_1 yaitu:

$$\mathbf{K}_1 = [-100 \ 395,95 \ -93,98 \ 107,43]$$

Sedangkan untuk titik kerja yang kedua yaitu pada $x_2^* = \pm 0,2$ radian:

$\mathbf{x}^* = [0 \ 0 \ 0 \ 0]^T$ dan $u^* = 0$
sehingga diperoleh:

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

dengan

$$\mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0,2316 & 0 & -0,0001 \\ 0 & 14,6673 & 0 & -0,0073 \end{bmatrix}, \mathbf{B}_2 = \begin{bmatrix} 0 \\ 0 \\ 0,8264 \\ 1,2094 \end{bmatrix}$$

Didapatkan matriks \mathbf{K}_2 yaitu:

$$\mathbf{K}_2 = [-100 \ 399,48 \ -94,41 \ 109,37]$$

Setelah didapatkan kedua nilai \mathbf{K} masing-masing dari setiap titik kerjanya, maka disusunlah sistem kontrol LQR berbasis *fuzzy* T-S. Hal ini dimaksudkan untuk menentukan nilai \mathbf{K} mana yang bekerja saat proses stabilisasi sistem pendulum kereta. Penentuan besarnya proporsi nilai \mathbf{K}_1 dan \mathbf{K}_2 berdasarkan nilai sudut batang pendulum yang terukur, sehingga yang digunakan dalam pembuatan aturan *fuzzy* T-S juga hanya membership untuk posisi sudut pendulum.

Membership function yang digunakan adalah fungsi segitiga untuk titik kerja pertama yaitu saat $x_2^* = 0$ radian. Parameter fungsi segitiga adalah a, b , dan c sesuai dengan penjelasan pada (2.20). Akan tetapi pada *membership function* ini dipilih nilai a, b , dan c masing-masing adalah $-0,2; 0$; dan $0,2$.

Sedangkan *membership function* yang digunakan untuk titik kerja yang kedua yaitu saat $x_2^* = \pm 0,2$ radian adalah fungsi trapesium. Parameter fungsi trapesium adalah a, b, c , dan d sesuai dengan penjelasan pada (2.21). Akan tetapi pada *membership function* ini dipilih nilai a, b, c , dan d masing-masing adalah $-\infty; -\infty; -0,2$; dan 0 untuk membership sebelah kiri dan $0; 0,2; \infty$; dan ∞ untuk membership sebelah kanan dengan menganggap ∞ adalah nilai yang sangat besar (tak hingga). Dari kedua *membership function* tersebut dapat dilihat pada Gambar 3.1.

Aturan yang digunakan pada perancangan sistem kontrol LQR berbasis *fuzzy* T-S ini terdapat dua macam, yaitu aturan untuk *plant* dan

aturan untuk kontroler. Aturan untuk *plant* sistem pendulum kereta adalah:

Aturan 1 untuk *plant*:

IF x_2 adalah M_1 (sekitar 0 radian)

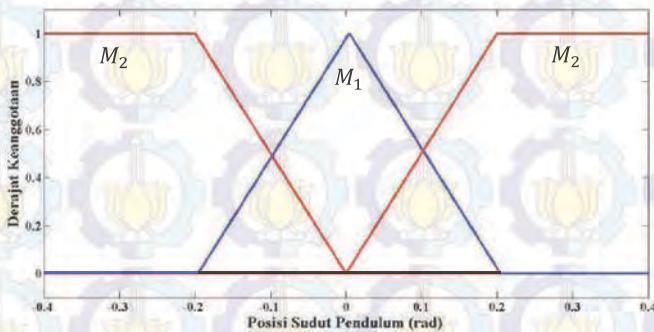
$$\text{THEN } \dot{\mathbf{x}} = \mathbf{A}_1 \mathbf{x} + \mathbf{B}_1 u \quad (3.6)$$

dan

Aturan 2 untuk *plant*:

IF x_2 adalah M_2 (sekitar $\pm 0,2$ radian)

$$\text{THEN } \dot{\mathbf{x}} = \mathbf{A}_2 \mathbf{x} + \mathbf{B}_2 u \quad (3.7)$$



Gambar 3.1 *Membership Function* Posisi Sudut Pendulum untuk Aturan *Plant* dan Aturan Kontroler

Selanjutnya, untuk menentukan parameter kontroler digunakan metode *Parallel Distributed Compensation* (PDC). Dengan menggunakan metode PDC ini, tiap-tiap aturan kontroler dirancang berdasarkan aturan *plant* yang bersesuaian. Sehingga dari aturan *plant* pada (3.6) dan (3.7) didapatkan aturan kontroler sebagai berikut:

Aturan 1 untuk kontroler:

IF x_2 adalah M_1 (sekitar 0 radian)

$$\text{THEN } u = -\mathbf{K}_1 \mathbf{x} \quad (3.8)$$

dan

Aturan 2 untuk kontroler:

IF x_2 adalah M_2 (sekitar $\pm 0,2$ radian)

$$\text{THEN } u = -\mathbf{K}_2 \mathbf{x} \quad (3.9)$$

Berdasarkan dua aturan kontroler tersebut, maka didapatkan sinyal kontrol secara keseluruhan adalah:

$$u = -(m_1(z)\mathbf{K}_1 + m_2(z)\mathbf{K}_2)\mathbf{x} \quad (3.10)$$

dengan $m_1(z)$ adalah nilai pembobot pada aturan 1 dan $m_2(z)$ adalah nilai pembobot pada aturan 2.

3.3 Perancangan *Proportional Multiple Integral Observer* untuk Estimasi Kesalahan Aktuator

Sistem kontrol LQR berbasis *fuzzy* T-S dapat digunakan untuk mendapatkan kestabilan sistem pendulum kereta pada kondisi nominal. Kondisi nominal adalah kondisi pada saat komponen dalam sistem dapat bekerja secara optimal, atau dalam kata lain tidak terdapat kerusakan atau kesalahan pada komponen-komponen tersebut.

Akan tetapi apabila terdapat kesalahan (pada Tugas Akhir ini kesalahan terjadi pada aktuator), maka akan terjadi penurunan performa sistem atau bahkan terjadi ketidakstabilan. Untuk mengatasi akibat dari kesalahan yang terjadi pada aktuator tersebut, maka dirancanglah sebuah *observer* yang mampu mengestimasi kesalahan tersebut. Hasil estimasi kesalahan tersebut dapat digunakan untuk mengompensasi dampak kesalahan tersebut pada sistem pendulum kereta.

Observer yang dapat mengestimasi *state* dan kesalahan aktuator secara simultan dan digunakan pada Tugas Akhir ini adalah *Proportional Multiple Integral Observer* (PMIO). Agar *observer* tersebut dapat menjalankan fungsinya, maka dibentuk terlebih dahulu *augmented system* yang didalamnya terdapat *state* dan sinyal kesalahan (f_a) beserta turunannya hingga turunan ke s ($f_a^{(s)}$).

Sistem augmentasi dirancang dengan melibatkan sinyal kesalahan dan turunannya yang kedua. Jumlah aturan *fuzzy* yang dirancang adalah dua, sehingga model sistem *fuzzy* T-S yang sudah diaugmentasi menjadi:

$$\left. \begin{aligned} \dot{\mathbf{x}}_a &= \sum_{i=1}^2 m_i(z) (\bar{\mathbf{A}}_i \mathbf{x}_a + \bar{\mathbf{B}}_i \mathbf{u}) + \bar{\mathbf{G}}_i f_a^{(s)} \\ \mathbf{y}_a &= \bar{\mathbf{C}} \mathbf{x}_a \end{aligned} \right\} \quad (3.11)$$

dengan

$$\mathbf{x}_a = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & \dot{f}_a & \ddot{f}_a & f_a \end{bmatrix}^T \quad (3.12)$$

$$\bar{\mathbf{A}}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0,2522 & 0 & -0,0001 & 0 & 0 & 0,8272 \\ 0 & 15,0133 & 0 & -0,0074 & 0 & 0 & 1,2353 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\bar{\mathbf{A}}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0,2316 & 0 & -0,0001 & 0 & 0 & 0,8264 \\ 0 & 14,6673 & 0 & -0,0073 & 0 & 0 & 1,2094 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\bar{\mathbf{B}}_1 = [0 \ 0 \ 0,8272 \ 1,2353 \ 0 \ 0 \ 0]^T$$

$$\bar{\mathbf{B}}_2 = [0 \ 0 \ 0,8264 \ 1,2094 \ 0 \ 0 \ 0]^T$$

$$\bar{\mathbf{C}} = [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$$

$$\bar{\mathbf{G}} = [0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0]^T$$

Setelah sistem augmentasi dirancang, maka *fuzzy* T-S PMIO yang mampu mengestimasi *state* sistem dan sinyal kesalahan secara simultan dapat diperoleh seperti pada (3.13).

$$\left. \begin{aligned} \dot{\hat{\mathbf{x}}}_a &= \sum_{i=1}^2 m_i(z) (\bar{\mathbf{A}}_i \hat{\mathbf{x}}_a + \bar{\mathbf{B}}_i \mathbf{u} + \bar{\mathbf{L}}_i (\mathbf{y}_a - \hat{\mathbf{y}}_a)) \\ \hat{\mathbf{y}}_a &= \bar{\mathbf{C}} \hat{\mathbf{x}}_a \end{aligned} \right\} \quad (3.13)$$

Akan tetapi untuk memastikan apakah *fuzzy* T-S tersebut dapat dirancang, maka harus dibuktikan seperti pada (2.46).

$$\text{rank} \begin{bmatrix} \mathbf{A}_i & \mathbf{E}_i \\ \mathbf{C} & \mathbf{0} \end{bmatrix} = 5$$

Dikarenakan jumlah *state* pada sistem pendulum kereta adalah 4 ($n = 4$) dan ukuran vektor kesalahan aktuator adalah 1 ($k = 1$). Oleh karena itu *observer* dapat dirancang dan digunakan untuk mengestimasi *state* dan kesalahan sktuator.

Dinamika *error* estimasi *state* pada (3.14) dapat diperoleh dengan mengurangi (3.13) dengan (3.11).

$$\dot{\mathbf{e}}_{\mathbf{x}} = \sum_{i=1}^2 m_i(z) (\bar{\mathbf{A}}_i - \bar{\mathbf{L}}_i \bar{\mathbf{C}}) \mathbf{e}_{\mathbf{x}} + \bar{\mathbf{G}} f_a^{(s)} \quad (3.14)$$

dengan

$$\mathbf{e}_{\mathbf{x}} = \hat{\mathbf{x}}_a - \mathbf{x}_a$$

dan $\mathbf{e}_{\mathbf{x}}$ adalah *error* estimasi *state*.

Untuk melemahkan dampak simultan dari adanya $f_a^{(s)}$ pada *error* estimasi dan juga untuk memastikan kestabilan sistem, maka harus memenuhi Pertidaksamaan (3.15).

$$\dot{V}(\mathbf{e}_{\mathbf{x}}) + \frac{1}{\gamma_{\text{pmi}}} \mathbf{e}_{\mathbf{x}}^T \mathbf{C}_{\mathbf{p}}^T \mathbf{C}_{\mathbf{p}} \mathbf{e}_{\mathbf{x}} - \gamma_{\text{pmi}} f_a^{(s)T} f_a^{(s)} < 0 \quad (3.15)$$

dengan $\dot{V}(\mathbf{e}_{\mathbf{x}})$ adalah turunan pertama terhadap waktu dari fungsi kandidat Lyapunov ($\dot{V}(\mathbf{e}_{\mathbf{x}}) = \mathbf{e}_{\mathbf{x}}^T \bar{\mathbf{P}} \dot{\mathbf{e}}_{\mathbf{x}}$), $\bar{\mathbf{P}} \in \mathcal{R}^{\bar{n} \times \bar{n}}$ adalah matriks simetris semi definit positif, dan $\mathbf{C}_{\mathbf{p}} \in \mathcal{R}^{\bar{n} \times \bar{n}}$ adalah matriks performa keluaran. Dari (3.21) dapat diperoleh:

$$\begin{aligned} \dot{V}(\mathbf{e}_{\mathbf{x}}) &= \dot{\mathbf{e}}_{\mathbf{x}}^T \bar{\mathbf{P}} \mathbf{e}_{\mathbf{x}} + \mathbf{e}_{\mathbf{x}}^T \bar{\mathbf{P}} \dot{\mathbf{e}}_{\mathbf{x}} \\ &= \sum_{i=1}^2 m_i(z) (((\bar{\mathbf{A}}_i - \bar{\mathbf{L}}_i \bar{\mathbf{C}}) \mathbf{e}_{\mathbf{x}})^T + (\bar{\mathbf{G}} f_a^{(s)})^T) \bar{\mathbf{P}} \mathbf{e}_{\mathbf{x}} \\ &\quad + \sum_{i=1}^2 m_i(z) \mathbf{e}_{\mathbf{x}}^T (((\bar{\mathbf{A}}_i - \bar{\mathbf{L}}_i \bar{\mathbf{C}}) \mathbf{e}_{\mathbf{x}}) + (\bar{\mathbf{G}} f_a^{(s)})) \\ &= \sum_{i=1}^2 m_i(z) \mathbf{e}_{\mathbf{x}}^T (\bar{\mathbf{A}}_i^T \bar{\mathbf{P}} + \mathbf{P} \bar{\mathbf{A}}_i - \bar{\mathbf{P}} \bar{\mathbf{L}}_i \bar{\mathbf{C}} \\ &\quad - (\bar{\mathbf{P}} \bar{\mathbf{L}}_i \bar{\mathbf{C}})^T) \mathbf{e}_{\mathbf{x}} + \mathbf{e}_{\mathbf{x}}^T \bar{\mathbf{P}} \bar{\mathbf{G}} f_a^{(s)} + \bar{\mathbf{G}}^T f_a^{(s)T} \bar{\mathbf{P}} \mathbf{e}_{\mathbf{x}} \end{aligned} \quad (3.16)$$

dengan memilih $\bar{\mathbf{H}}_i = \bar{\mathbf{P}} \bar{\mathbf{L}}_i$ maka (3.16) menjadi:

$$\dot{V}(\mathbf{e}_x) = \sum_{i=1}^2 m_i(z) \mathbf{e}_x^T (\bar{\mathbf{A}}_i^T \bar{\mathbf{P}} + \bar{\mathbf{P}} \bar{\mathbf{A}}_i - \bar{\mathbf{H}}_i \bar{\mathbf{C}} - (\bar{\mathbf{H}}_i \bar{\mathbf{C}})^T) \mathbf{e}_x + \mathbf{e}_x^T \bar{\mathbf{P}} \bar{\mathbf{G}} f_a^{(s)} + \bar{\mathbf{G}}^T f_a^{(s)T} \bar{\mathbf{P}} \mathbf{e}_x \quad (3.17)$$

Dari (3.15) dan (3.17) diperoleh:

$$\sum_{i=1}^2 m_i(z) \mathbf{e}_x^T (\bar{\mathbf{A}}_i^T \bar{\mathbf{P}} + \bar{\mathbf{P}} \bar{\mathbf{A}}_i - \bar{\mathbf{H}}_i \bar{\mathbf{C}} - (\bar{\mathbf{H}}_i \bar{\mathbf{C}})^T + \frac{1}{\gamma_{\text{pmi}}} \mathbf{C}_p^T \mathbf{C}_p) \mathbf{e}_x + \mathbf{e}_x^T \bar{\mathbf{P}} \bar{\mathbf{G}} f_a^{(s)} + \bar{\mathbf{G}}^T f_a^{(s)T} \bar{\mathbf{P}} \mathbf{e}_x - \gamma_{\text{pmi}} f_a^{(s)T} f_a^{(s)} < 0$$

Dengan mengubahnya menjadi bentuk matriks, sehingga diperoleh pertidaksamaan:

$$\sum_{i=1}^2 m_i(z) \begin{bmatrix} \mathbf{e}_x^T \\ f_a^{(s)T} \end{bmatrix}^T \begin{bmatrix} \varphi + \frac{1}{\gamma_{\text{pmi}}} \mathbf{C}_p^T \mathbf{C}_p & \bar{\mathbf{P}} \bar{\mathbf{G}} \\ \bar{\mathbf{G}}^T \bar{\mathbf{P}} & -\gamma_{\text{pmi}} \end{bmatrix} \begin{bmatrix} \mathbf{e}_x \\ f_a^{(s)} \end{bmatrix} < 0$$

Dengan menggunakan teori *Schur Complement*, maka didapatkan:

$$\sum_{i=1}^2 m_i[z] \begin{bmatrix} \mathbf{e}_x^T \\ f_a^{(s)T} \end{bmatrix}^T \begin{bmatrix} \varphi & \bar{\mathbf{P}} \bar{\mathbf{G}} & \mathbf{C}_p^T \\ \bar{\mathbf{G}}^T \bar{\mathbf{P}} & -\gamma_{\text{pmi}} & 0 \\ \mathbf{C}_p & 0 & -\gamma_{\text{pmi}} \end{bmatrix} \begin{bmatrix} \mathbf{e}_x \\ f_a^{(s)} \end{bmatrix} < 0 \quad (3.18)$$

Agar Pertidaksamaan (3.18) definit negatif, maka (3.18) menjadi (3.19).

$$\begin{bmatrix} \varphi & \bar{\mathbf{P}} \bar{\mathbf{G}} & \mathbf{C}_p^T \\ \bar{\mathbf{G}}^T \bar{\mathbf{P}} & -\gamma_{\text{pmi}} & 0 \\ \mathbf{C}_p & 0 & -\gamma_{\text{pmi}} \end{bmatrix} < 0 \quad (3.19)$$

dengan $\varphi = \bar{\mathbf{A}}_i^T \bar{\mathbf{P}} + \bar{\mathbf{P}} \bar{\mathbf{A}}_i - \bar{\mathbf{H}}_i \bar{\mathbf{C}} - (\bar{\mathbf{H}}_i \bar{\mathbf{C}})^T$ dan $\bar{\mathbf{L}}_i = \bar{\mathbf{P}}^{-1} \bar{\mathbf{H}}_i$.

Agar *pole observer* dapat diletakkan sesuai dengan desain yang diinginkan, maka harus menyertakan *constraint LMI region*. Dengan memilih daerah yang akan menjadi letak *pole observer* yaitu irisan antara setengah bidang dan kerucut sesuai dengan teori pada (2.59) dan (2.61), maka dalam perancangan *observer* juga harus memenuhi (3.20) dan (3.21).

$$\bar{\mathbf{A}}_i^T \bar{\mathbf{P}} + \bar{\mathbf{P}} \bar{\mathbf{A}}_i - \bar{\mathbf{H}}_i \bar{\mathbf{C}} - (\bar{\mathbf{H}}_i \bar{\mathbf{C}})^T + 2\alpha \bar{\mathbf{P}} < 0 \quad (3.20)$$

$$\begin{bmatrix} \sin\theta(z + \bar{z}) & \cos\theta(z - \bar{z}) \\ \cos\theta(z - \bar{z}) & \sin\theta(z + \bar{z}) \end{bmatrix} < 0 \quad (3.21)$$

dengan $z = \bar{\mathbf{P}} \bar{\mathbf{A}}_i - \bar{\mathbf{H}}_i \bar{\mathbf{C}}$ dan $\bar{z} = \bar{\mathbf{A}}_i^T \bar{\mathbf{P}} - (\bar{\mathbf{H}}_i \bar{\mathbf{C}})^T$.

Dengan memilih nilai parameter-parameter yang akan dirancang dalam mencari *observer gain* sebagai berikut:

$$\mathbf{C}_p = \begin{bmatrix} 0,01 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,01 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,01 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,01 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0,01 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0,01 \end{bmatrix}$$

serta

$$\alpha = 2 \text{ dan } \theta = 0,685 \text{ radian.}$$

Oleh karena itu dengan menyelesaikan pertidaksamaan pada (3.29)-(3.31) dengan LMI, diperoleh nilai *observer gain* dan level pelemahan:

$$\bar{\mathbf{L}}_1 = 10^6 \times \begin{bmatrix} 0,0006 \\ 1,3483 \\ 0,0173 \\ 5,1757 \\ -0,1328 \\ -0,2429 \\ -0,1906 \end{bmatrix}$$

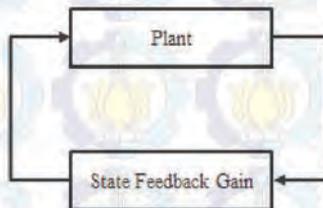
$$\bar{\mathbf{L}}_2 = 10^6 \times \begin{bmatrix} 0,0004 \\ 0,9461 \\ 0,0116 \\ 3,5924 \\ -0,0826 \\ -0,1529 \\ -0,1215 \end{bmatrix}$$

$$\gamma_{\text{pmi}} = 0.135$$

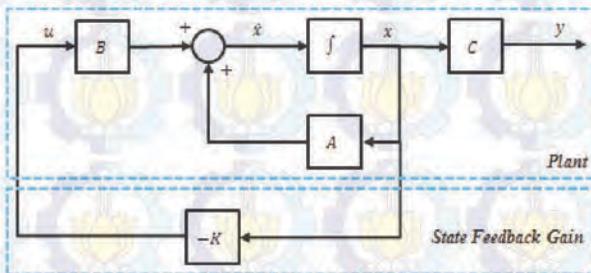
3.4 Struktur Sistem Kontrol Stabilisasi dan *Proportional Multiple Integral Observer*

Dalam kondisi nominal, sistem kontrol yang digunakan adalah kontrol LQR berbasis *fuzzy* T-S. Struktur dan diagram sistem kontrolnya dapat dilihat pada Gambar 3.2 dan Gambar 3.3. Pada Gambar 3.2 dapat dilihat bahwa struktur kontrol LQR terdiri dari dua blok yaitu blok *Plant* dan blok Nominal Controller.

Pada blok Nominal Controller berisikan nilai *state feedback gain* yang digunakan untuk stabilisasi sistem pendulum kereta.

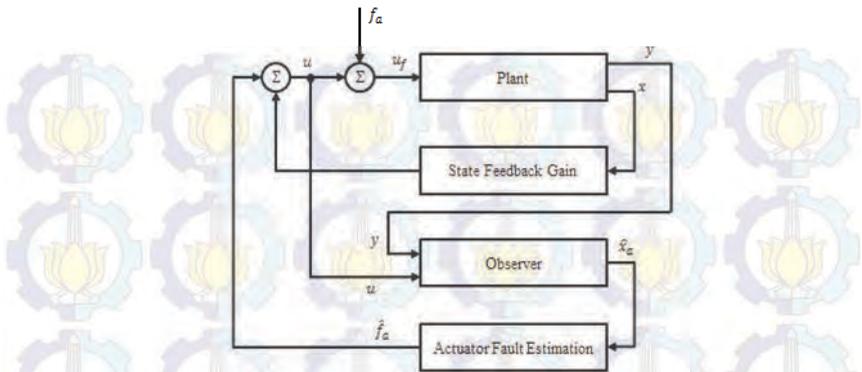


Gambar 3.2 Struktur Sistem Kontrol LQR

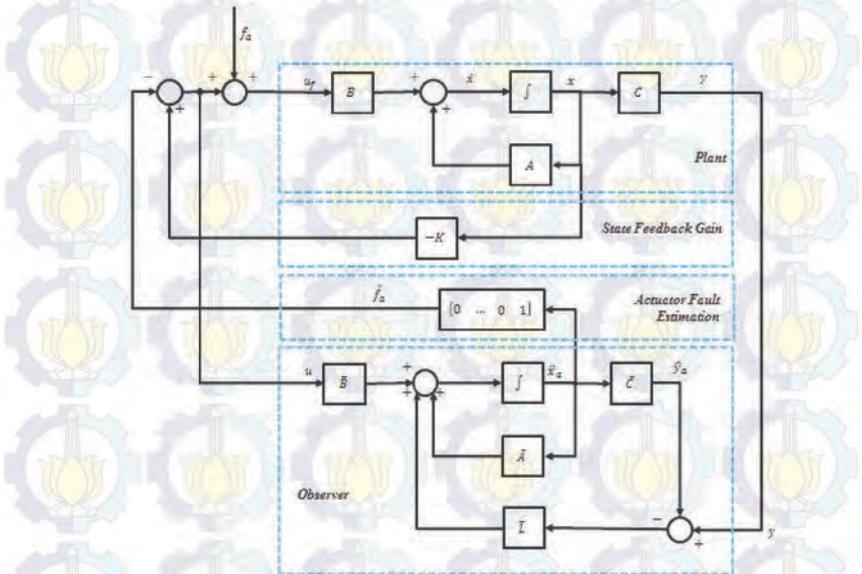


Gambar 3.3 Diagram Sistem Kontrol LQR

Pada saat terjadi kesalahan aktuator, maka dibutuhkan *Proportional Multiple Integral Observer* yang mampu mengestimasi *state* dan kesalahan secara simultan. Struktur dan diagram *observer* dapat dilihat pada Gambar 3.4 dan Gambar 3.5.



Gambar 3.4 Struktur Sistem Kontrol dengan *Proportional Multiple Integral Observer*



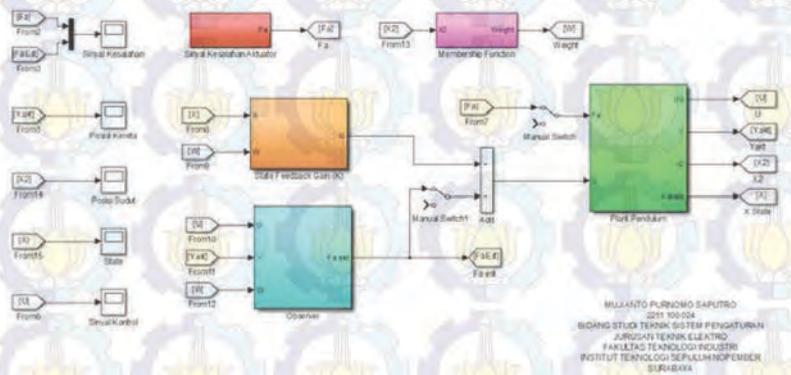
Gambar 3.5 Diagram Sistem Kontrol dengan *Proportional Multiple Integral Observer*

3.5 Perancangan Simulasi Sistem Pendulum Kereta

Simulasi sistem pendulum kereta dapat dilihat pada Gambar 3.6. Pada blok *Plant* Pendulum berisikan model matematika dari sistem pendulum kereta yang terdiri dari hubungan antara sinyal kontrol dan keempat *state* sistem. Pada blok *State Feedback Gain* (K) berisikan nilai blok *state feedback gain* berbasis *fuzzy* Takagi-Sugeno. Nilai *gain* tersebut sudah diberi bobot sesuai dengan aturan titik kerja. Blok tersebut merupakan blok yang menghubungkan antara nilai keempat *state* dengan nilai sinyal kontrol nominal.

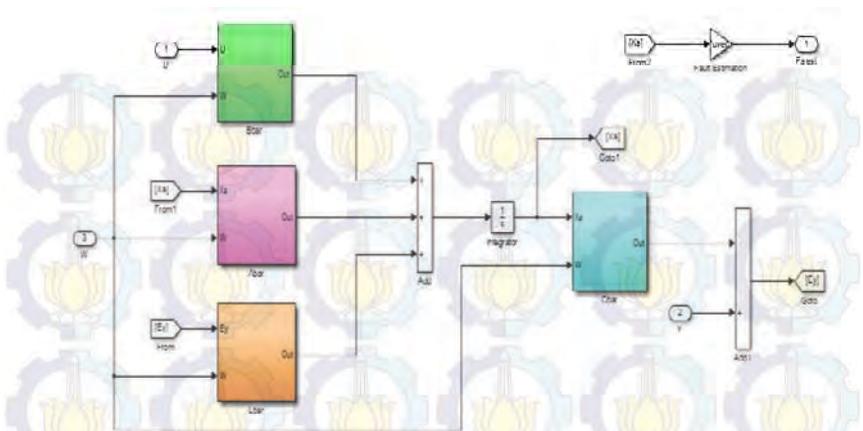
Pada blok *Observer* terdiri dari masukan berupa sinyal kontrol dan keluaran sistem, sedangkan keluaran dari blok tersebut adalah estimasi kesalahan aktuator. Pada blok *Observer* berisikan nilai blok *augmented system* berbasis *fuzzy* Takagi-Sugeno.

Blok Pembobotan terdiri dari masukan berupa *state* posisi sudut pendulum dan keluaran berupa vektor pembobot sesuai dengan aturan titik kerja. Blok Sinyal Kesalahan Aktuator berisikan sinyal step dan ramp yang digunakan sebagai sinyal kesalahan aditif pada sinyal kontrol nominal.

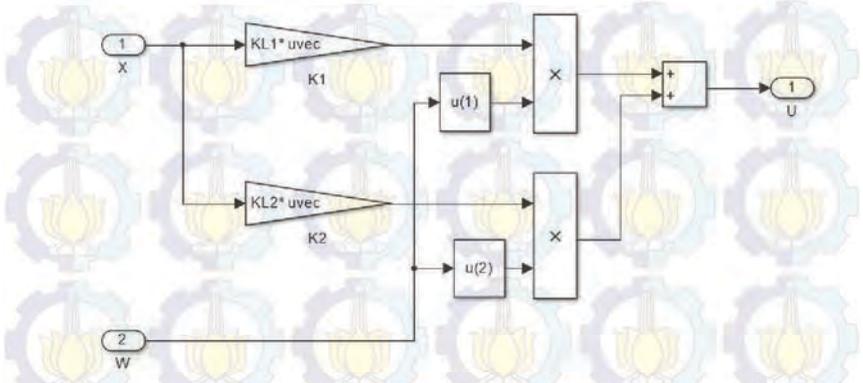


Gambar 3.6 Diagram Simulink untuk Simulasi Sistem Pendulum Kereta

Di dalam blok subsistem *Observer* dapat dilihat pada Gambar 3.7 dan blok subsistem *State Feedback Gain Controller* pada Gambar 3.8.



Gambar 3.7 Diagram Simulink untuk *Observer*



Gambar 3.8 Diagram Simulink untuk *State Feedback Gain Controller*

3.6 Perancangan Implementasi Sistem Pendulum Kereta

Algoritma kontrol yang telah dirancang dan diuji atau disimulasikan dengan *Simulink* MATLAB dapat diimplementasikan dengan membutuhkan beberapa peralatan (*hardware*) dan *software*. *Software* yang dibutuhkan adalah *Real Time Workshop-Real Time Windows* yang telah tersedia di dalam MATLAB, sedangkan peralatan (*hardware*) yang dibutuhkan adalah modul *input-output* yang menghubungkan antara komputer dengan *plant Pendulum Cart System* dan komputer sebagai kontroler yang dihubungkan secara *interface* 44

BAB 4

HASIL PENGUJIAN DAN ANALISIS

Pada Bab ini dilakukan pengujian dan analisis hasil simulasi dan implementasi sistem kontrol pada sistem pendulum kereta seperti yang telah dijelaskan pada Bab III. Tahap simulasi dilakukan dengan menggunakan *Simulink* pada MATLAB. Tahap implementasi akan diterapkan pada *plant* nyata, “*Digital Pendulum Mechanical Unit 33-200*” dari *Feedback Instruments Ltd*. Pada pembahasan simulasi dan implementasi akan ditampilkan kondisi nominal (*fault free case*) dan kondisi dengan kesalahan pada aktuator (*faulty case*).

4.1 Hasil Simulasi

Tahap simulasi dilakukan dengan dua kondisi yaitu *fault free case* dan kondisi *faulty case*. Pada kondisi *fault free case*, tidak terdapat kesalahan pada aktuator. Sedangkan pada kondisi *faulty case*, terdapat kesalahan pada aktuator. Kesalahan tersebut bersifat aditif, atau jenis kesalahan tersebut dinamakan *bias* yang berupa sinyal *step* dan *ramp*, seperti yang dijelaskan pada Subbab 2.2.

4.1.1 Hasil Simulasi *Fault Free Case*

Pada tahap simulasi kondisi *fault free case*, akan diujikan dengan beberapa kombinasi matriks **Q** dan **R** untuk menghasilkan respon sistem yang paling baik dari segi maksimum *overshoot*, maksimum *undershoot*, dan *settling time*.

Pengujian yang kedua adalah dengan memberikan berbagai kondisi awal *state* posisi pendulum untuk mengetahui respon sistem. Untuk pengujian terakhir yaitu memberikan kondisi awal pada *state* kecepatan pendulum disamping juga memberikan kondisi awal posisi pendulum. Hal tersebut dilakukan karena pada saat implementasi nantinya, batang pendulum diayunkan secara manual dengan tangan (*hand-stand*) sehingga mengakibatkan adanya kecepatan awal pendulum.

Pada perancangan kontroler LQR berbasis *fuzzy* T-S sangat dipengaruhi pada pemilihan kombinasi matriks **Q** dan **R** sesuai dengan indeks performa pada (2.39). Oleh karena itu dibutuhkan pemilihan kombinasi yang tepat agar respon sistem dapat memenuhi spesifikasi desain yang diinginkan.

Pemilihan kombinasi matriks **Q** dan **R** dilakukan secara *trial and error* dan pada pembahasan ini akan ditampilkan tiga kombinasi yang akan dibandingkan hasil respon sistemnya. Respon posisi kereta, posisi pendulum, dan sinyal kontrol dari ketiga kombinasi matriks **Q** dan **R** tersebut akan dibandingkan dan dipilih yang sesuai dengan spesifikasi yang diinginkan.

Ketiga kombinasi matriks **Q** dan **R** dapat dilihat pada Tabel 4.1 dan *state feedback gain controller* dari ketiga kombinasi matriks **Q** dan **R** tersebut dapat dilihat pada Tabel 4.2.

Tabel 4.1 Kombinasi Matriks **Q** dan **R**

Kombinasi ke-	Matriks Q	Matriks R
1	$\begin{bmatrix} 5000 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 \\ 0 & 0 & 500 & 0 \\ 0 & 0 & 0 & 500 \end{bmatrix}$	[0,5]
2	$\begin{bmatrix} 5000 & 0 & 0 & 0 \\ 0 & 500 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}$	[0,5]
3	$\begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 500 & 0 & 0 \\ 0 & 0 & 500 & 0 \\ 0 & 0 & 0 & 500 \end{bmatrix}$	[0,5]

Dari ketiga kombinasi matriks **Q** dan **R** tersebut kemudian diperoleh nilai *state feedback gain controller*. Dari tiap-tiap kombinasi matriks **Q** dan **R** akan didapatkan respon posisi kereta, posisi pendulum, dan sinyal kontrol yang akan dibandingkan nilai dari maksimum *undershoot*, maksimum *overshoot*, dan *settling time* dari tiap-tiap respon.

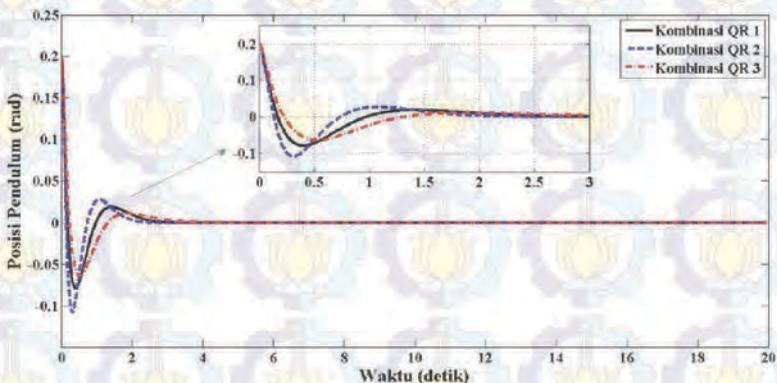
Maksimum *overshoot* adalah nilai simpangan terbesar dari respon yang bernilai positif terhadap nilai *steady state*. Maksimum *undershoot* adalah nilai simpangan terbesar dari respon yang bernilai negatif terhadap nilai *steady state*. *Settling time* adalah waktu saat nilai respon mendekati nilai referensi (nilai referensi untuk kasus stabilisasi

adalah 0) dengan nilai $error < 0,001$ untuk respon posisi kereta dan posisi pendulum dan nilai $error < 0,01$ untuk respon sinyal kontrol.

Tabel 4.2 *State Feedback Gain Controller* dari beberapa Kombinasi Matriks **Q** dan **R**

Kombinasi ke-	<i>State Feedback Gain Controller (K)</i>
1	$K_1 = [-100 \quad 395,95 \quad -93,98 \quad 107,43]$ $K_2 = [-100 \quad 399,48 \quad -94,41 \quad 109,37]$
2	$K_1 = [-100 \quad 292,01 \quad -76,89 \quad 77,04]$ $K_2 = [-100 \quad 295,31 \quad -77,36 \quad 78,70]$
3	$K_1 = [-44,72 \quad 298,43 \quad -60,11 \quad 83,63]$ $K_2 = [-44,72 \quad 300,56 \quad 60,29 \quad 84,93]$

Gambar 4.1 merupakan respon dari posisi pendulum dengan berbagai kombinasi matriks **Q** dan **R** pada kondisi awal *state* berada di $x = [0 \quad 0,2 \quad 0 \quad 0]$. Nilai maksimum *undershoot*, maksimum *overshoot*, dan *settling time* dari ketiga respon pada Gambar 4.1 dapat dilihat pada Tabel 4.3.



Gambar 4.1 Respon Posisi Pendulum dengan Berbagai Kombinasi Matriks **Q** dan **R**

Dari Tabel 4.3 dapat dilihat bahwa maksimum *overshoot* dan maksimum *undershoot* yang paling kecil diantara ketiga respon posisi pendulum adalah respon pada percobaan kombinasi ketiga yaitu sebesar 0,01144 rad dan 0,06373 rad. Nilai tersebut lebih kecil dari maksimum *overshoot* dan maksimum *undershoot* respon pada percobaan kombinasi pertama yaitu berturut-turut sebesar 0,01905 rad dan 0,07902 rad, serta respon pada kombinasi kedua berturut-turut sebesar 0,02732 rad dan 0,10840 rad.

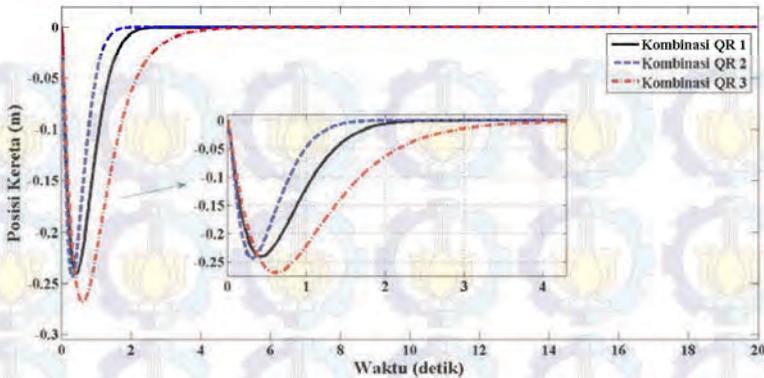
Akan tetapi respon pada percobaan kombinasi ketiga mempunyai nilai *settling time* yang lebih besar dari yang lainnya yaitu sebesar 3,891 detik. Respon yang memiliki nilai *settling time* paling kecil adalah respon pada percobaan kombinasi kedua yaitu sebesar 2,370 detik. Sedangkan respon dengan percobaan kombinasi pertama memiliki nilai *settling time* sebesar 2,858 detik.

Tabel 4.3 Perbandingan Spesifikasi Transien Respon Posisi Pendulum dengan Berbagai Kombinasi Matriks **Q** dan **R**

Kombinasi ke -	<i>Overshoot</i> (rad)	<i>Undershoot</i> (rad)	<i>Settling Time</i> (detik)
1	0,01905	0,07902	2,858
2	0,02732	0,10840	2,370
3	0,01144	0,06373	3,891

Gambar 4.2 merupakan respon posisi kereta dengan berbagai kombinasi matriks **Q** dan **R** pada kondisi awal *state* berada pada $\mathbf{x} = [0 \ 0,2 \ 0 \ 0]$. Dari Tabel 4.4 dapat dilihat bahwa nilai maksimum *undershoot* yang paling kecil adalah respon pada percobaan kombinasi pertama yaitu sebesar 0,2405 meter. Nilai tersebut lebih kecil dibandingkan dengan percobaan kombinasi yang kedua yaitu sebesar 0,2439 meter. Sedangkan nilai maksimum *undershoot* yang paling besar adalah pada respon pada percobaan kombinasi yang ketiga yaitu sebesar 0,269 meter.

Nilai *settling time* dari ketiga respon tersebut yang paling kecil adalah respon pada percobaan kombinasi kedua yaitu sebesar 1,798 detik dibandingkan dengan respon pada percobaan kombinasi pertama dan ketiga yaitu berturut-turut sebesar 2,446 detik dan 4,897 detik.



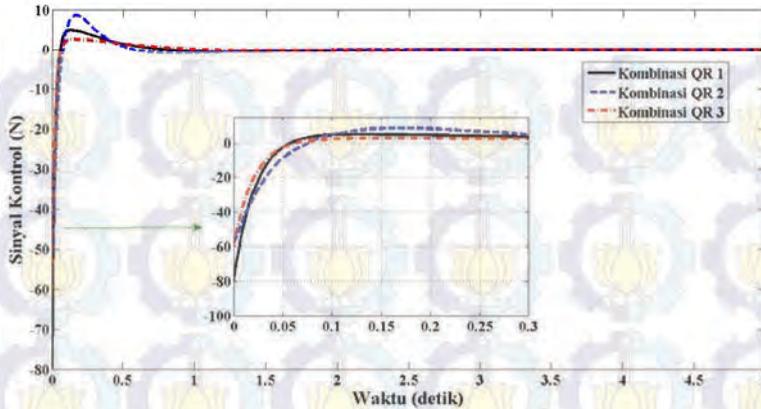
Gambar 4.2 Respon Posisi Kereta dengan Berbagai Kombinasi Matriks **Q** dan **R**

Tabel 4.4 Perbandingan Spesifikasi Transien Respon Posisi Kereta dengan Berbagai Kombinasi Matriks **Q** dan **R**

Kombinasi ke -	<i>Undershoot</i> (m)	<i>Settling Time</i> (detik)
1	0,2405	2,446
2	0,2439	1,798
3	0,269	4,897

Respon sinyal kontrol dengan berbagai kombinasi matriks **Q** dan **R** pada kondisi awal *state* berada pada $\mathbf{x} = [0 \ 0,2 \ 0 \ 0]$ dapat dilihat pada Gambar 4.3. Tabel 4.5 menunjukkan perbandingan nilai maksimum *undershoot*, maksimum *overshoot*, dan *settling time* dari ketiga respon. Nilai maksimum *overshoot* yang paling kecil adalah respon pada percobaan kombinasi ketiga yaitu sebesar 2,699 Newton. Respon pada percobaan kombinasi kedua mempunyai nilai maksimum *undershoot* paling kecil dibandingkan dengan respon yang lain yaitu sebesar 58,84 Newton.

Respon pada percobaan kombinasi kedua juga memiliki *settling time* yang paling kecil yaitu 2,07 detik dibandingkan dengan respon pada percobaan kombinasi pertama dan ketiga yang berturut-turut mempunyai nilai 2,672 detik dan 3,901 detik.



Gambar 4.3 Sinyal Kontrol dengan Berbagai Kombinasi Matriks **Q** dan **R**

Tabel 4.5 Perbandingan Spesifikasi Transien Sinyal Kontrol dengan Berbagai Kombinasi Matriks **Q** dan **R**

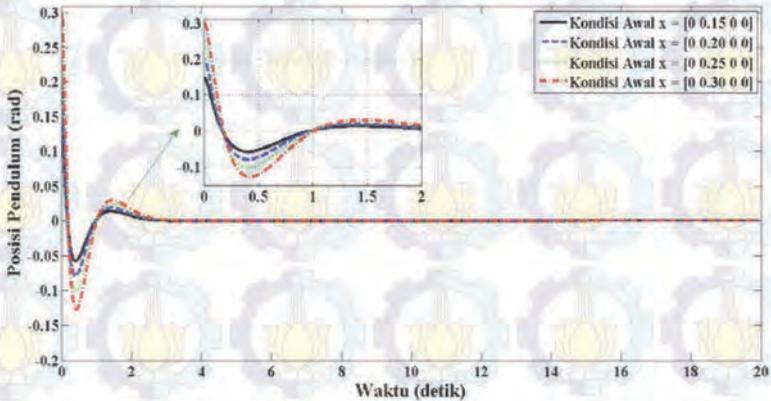
Kombinasi ke -	<i>Overshoot</i> (N)	<i>Undershoot</i> (N)	<i>Settling Time</i> (detik)
1	4,939	79,66	2,672
2	8,688	58,84	2,070
3	2,699	59,97	3,901

Berdasarkan pada Tabel 4.3-5, maka dipilih kombinasi matriks **Q** dan **R** yang pertama yaitu sesuai dengan pembahasan di subbab 3.2. Pemilihan tersebut dengan pertimbangan nilai maksimum *overshoot* dan maksimum *undershoot* yang tidak terlalu besar akan tetapi juga *settling time* yang tidak terlalu lama. Dibandingkan dengan percobaan kombinasi matriks yang kedua memiliki nilai *settling time* paling kecil pada respon posisi pendulum, tetapi memiliki *overshoot* dan *undershoot* yang sangat besar. Sedangkan pada percobaan kombinasi matriks yang ketiga memiliki *overshoot* dan *undershoot* yang paling kecil, akan tetapi *settling time* yang terlalu besar.

Respon sistem pendulum kereta juga dipengaruhi oleh kondisi awal dari *state*. Jika pada pembahasan sebelumnya dipilih nilai kondisi awal *state* pada posisi pendulum 0,2 rad, maka pada pembahasan ini akan diuji dengan kondisi awal posisi pendulum yang bervariasi. Variasi

nilai kondisi awal posisi pendulum yaitu 0,15 rad; 0,2 rad; 0,25 rad; dan 0,3 rad.

Dengan memilih kombinasi matriks **Q** dan **R** yang pertama yaitu sesuai dengan pembahasan di subbab 3.2, maka diperoleh respon posisi pendulum, posisi kereta, dan sinyal kontrol dengan berbagai kondisi awal seperti pada Gambar 4.4-6.



Gambar 4.4 Respon Posisi Pendulum dengan Berbagai Kondisi Awal

Tabel 4.6 Perbandingan Spesifikasi Transien Respon Posisi Pendulum dengan Berbagai Kondisi Awal

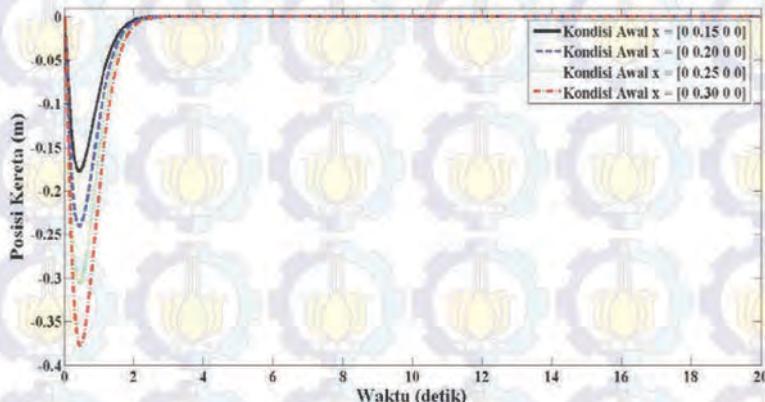
Kondisi Awal Posisi Pendulum (rad)	<i>Overshoot</i> (rad)	<i>Undershoot</i> (rad)	<i>Settling Time</i> (detik)
0,15	0,01406	0,05780	2,573
0,2	0,01905	0,07897	2,862
0,25	0,02430	0,10200	2,951
0,3	0,02994	0,12720	3,051

Pada Tabel 4.6 dapat dilihat bahwa nilai maksimum *overshoot* dan maksimum *undershoot* yang paling kecil adalah respon dengan kondisi awal 0,15 rad yaitu sebesar 0,01406 rad dan 0,05780 rad. Jika kondisi awal posisi semakin besar, maka nilai maksimum *overshoot* dan maksimum *undershoot* juga akan semakin besar. Sampai pada kondisi

awal posisi pendulum di 0,3 rad memiliki nilai maksimum *overshoot* dan maksimum *undershoot* yang paling besar yaitu 0,02994 rad dan 0,1272 rad.

Begitu juga dengan *settling time*, respon yang memiliki nilai *settling time* paling kecil adalah 2,573 detik saat kondisi awal 0,15 rad. Sedangkan respon yang memiliki *settling time* paling besar adalah 3,051 detik saat kondisi awal 0,3 rad.

Gambar 4.5 merupakan respon posisi kereta dengan berbagai kondisi awal posisi pendulum yang diberikan. Perbandingan respon dapat dilihat pada Tabel 4.7. Nilai maksimum *undershoot* terkecil adalah 0,1775 meter saat kondisi awal 0,15 rad, dan yang terbesar adalah 0,3780 meter saat kondisi awal 0,3 rad.

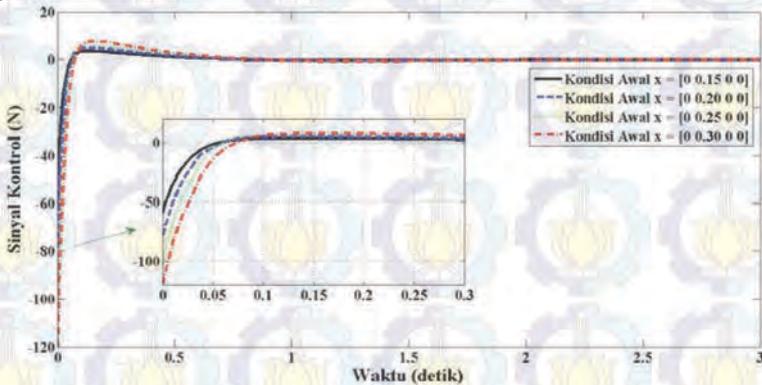


Gambar 4.5 Respon Posisi Kereta dengan Berbagai Kondisi Awal

Tabel 4.7 Perbandingan Spesifikasi Transien Respon Posisi Kereta dengan Berbagai Kondisi Awal

Kondisi Awal Posisi Pendulum (rad)	<i>Undershoot</i> (m)	<i>Settling Time</i> (detik)
0,15	0,1775	2,376
0,2	0,2406	2,475
0,25	0,3070	2,525
0,3	0,3780	2,608

Perbandingan spesifikasi transien sinyal kontrol dengan berbagai kondisi awal posisi pendulum dapat dilihat pada Tabel 4.8. Nilai maksimum *overshoot* dan maksimum *undershoot* yang paling kecil adalah sebesar 3,690 Newton dan 59,70 Newton saat kondisi awal 0,15 rad. Sedangkan saat kondisi awal 0,3 rad nilai maksimum *overshoot* dan maksimum *undershoot* merupakan yang paling besar yaitu 7,388 Newton dan 119,80 Newton. Pada saat kondisi awal 0,15 rad, respon memiliki *settling time* yang paling kecil yaitu 2,602 detik. Sedangkan respon yang memiliki *settling time* yang paling besar adalah 2,820 detik pada saat kondisi awal 0,3 rad.



Gambar 4.6 Sinyal Kontrol dengan Berbagai Kondisi Awal

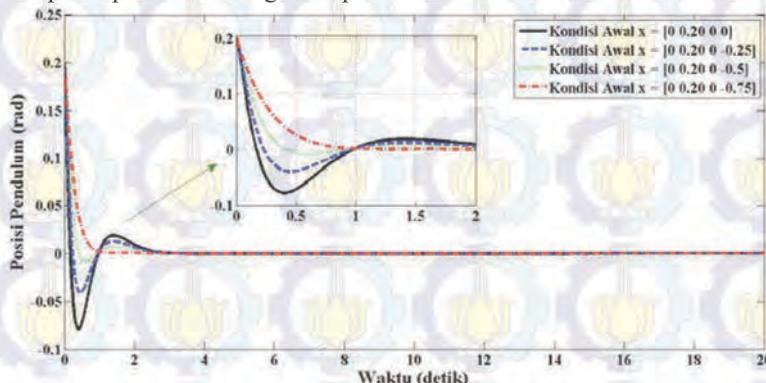
Tabel 4.8 Perbandingan Spesifikasi Transien Sinyal Kontrol dengan Berbagai Kondisi Awal

Kondisi Awal Posisi Pendulum (rad)	<i>Overshoot</i> (N)	<i>Undershoot</i> (N)	<i>Settling Time</i> (detik)
0,15	3,690	59,70	2,602
0,2	4,958	79,73	2,689
0,25	6,292	99,83	2,758
0,3	7,388	119,80	2,820

Dari pengujian berbagai kondisi awal posisi pendulum dapat ditarik sebuah analisis. Semakin besar kondisi awal posisi pendulum

maka semakin jauh dari posisi referensi 0 rad, yang mengakibatkan sinyal kontrol memberikan aksi yang semakin besar agar posisi pendulum cepat kembali pada posisi 0 rad. Hal ini juga mengakibatkan pada posisi kereta yang menyimpang semakin jauh untuk mengembalikan posisi pendulum.

Pada tahap implementasi, untuk menentukan kondisi awal posisi pendulum adalah dengan cara mengayunkan batang pendulum keatas secara manual. Hal tersebut akan mengakibatkan adanya kondisi awal kecepatan pendulum yang dipengaruhi oleh gaya dorong tangan. Oleh karena itu pada pembahasan simulasi ini akan dibandingkan respon sistem pendulum kereta antara pemberian nilai kondisi awal pada *state* kecepatan pendulum dengan tanpa diberi nilai kondisi awal.



Gambar 4.7 Respon Posisi Pendulum Dengan Beberapa Kondisi Awal Kecepatan Pendulum

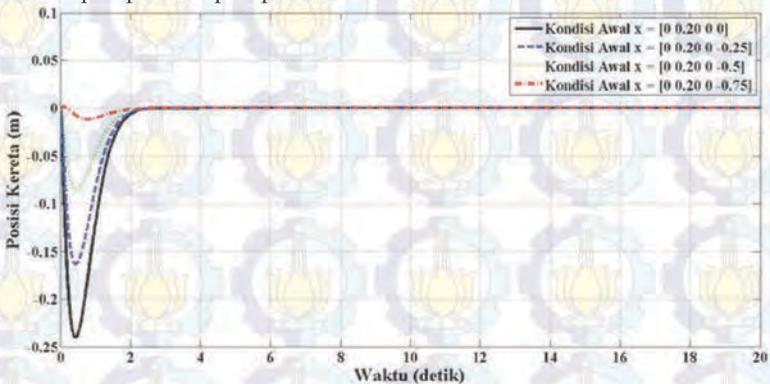
Didapatkan respon posisi pendulum dengan berbagai kondisi awal kecepatan pendulum seperti pada Gambar 4.7. Dari Tabel 4.9 dapat dilihat bahwa untuk kondisi awal kecepatan pendulum 0 rad/detik terdapat nilai maksimum *overshoot* dan maksimum *undershoot* pada respon posisi pendulum yang paling besar yaitu 0,01905 rad dan 0,07897 rad. Dengan pemberian kondisi awal kecepatan pendulum yang sesuai ternyata dapat mengurangi nilai maksimum *overshoot* dan maksimum *undershoot*. Dapat dilihat bahwa pada saat pemberian kondisi awal kecepatan pendulum 0,25 rad/detik; 0,5 rad/detik; dan 0,75 rad/detik secara berurutan dapat mengurangi nilai *overshoot* dan *undershoot* pada respon posisi pendulum yaitu menjadi 0,01304 rad dan 0,04058 rad (0,25 rad/detik), 0,00703 rad dan 0,00845 rad (0,5

rad/detik), serta tidak terdapat *overshoot* dan *undershoot* saat diberi kondisi awal kecepatan pendulum sebesar 0,75 rad/detik. Selain itu, dengan pemberian kondisi awal kecepatan pendulum yang tepat juga dapat mempercepat respon posisi pendulum.

Tabel 4.9 Perbandingan Spesifikasi Transien Respon Posisi Pendulum Dengan Beberapa Kondisi Awal Kecepatan Pendulum

Kondisi Awal Kecepatan Pendulum (rad/detik)	<i>Overshoot</i> (rad)	<i>Undershoot</i> (rad)	<i>Settling Time</i> (detik)
0	0,01905	0,07897	2,862
-0,25	0,01304	0,04058	2,757
-0,50	0,00703	0,00845	2,578
-0,75	-	-	1,852

Respon posisi kereta dengan pemberian kondisi awal kecepatan pendulum dapat dilihat pada Gambar 4.8. Dengan melihat Tabel 4.10 dapat dilihat bahwa pemberian kondisi awal kecepatan pendulum juga berdampak pada respon posisi kereta.



Gambar 4.8 Respon Posisi Kereta Dengan Beberapa Kondisi Awal Kecepatan Pendulum

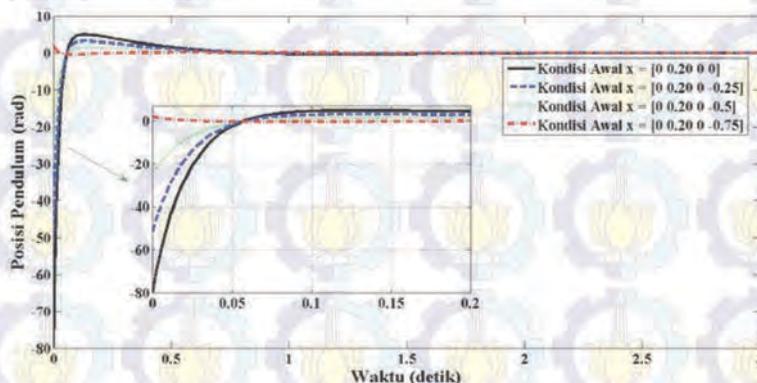
Sama seperti pada respon posisi pendulum, pada respon posisi kereta pemberian kondisi awal kecepatan dapat mengurangi nilai

maksimum *undershoot*. Terlihat bahwa pada Gambar 4.8 respon posisi kereta tanpa kondisi awal kecepatan pendulum memiliki *undershoot* yang sangat besar (0,2406 meter) dan respon yang lama (2,475 detik). Sedangkan dengan pemberian kondisi awal kecepatan pendulum yang tepat dapat mengurangi *undershoot* dan mempercepat respon.

Tabel 4.10 Perbandingan Spesifikasi Transien Respon Posisi Kereta Dengan Beberapa Kondisi Awal Kecepatan Pendulum

Kondisi Awal Kecepatan Pendulum (rad/detik)	<i>Undershoot</i> (m)	<i>Settling Time</i> (detik)
0	0,2406	2,475
-0,25	0,1630	2,388
-0,50	0,0854	2,316
-0,75	0,0118	2,181

Gambar 4.9 merupakan grafik sinyal kontrol dengan berbagai kondisi awal kecepatan pendulum. Pada saat tanpa diberi kondisi awal kecepatan pendulum, respon sinyal kontrol mempunyai nilai maksimum *overshoot* dan maksimum *undershoot* yang paling besar yaitu 4,958 Newton dan 79,73 Newton.



Gambar 4.9 Sinyal Kontrol Dengan Beberapa Kondisi Awal Kecepatan Pendulum

Dengan pemberian kondisi awal kecepatan pendulum cenderung mengurangi *overshoot* dan *undershoot* pada respon sinyal kontrol serta mempercepat respon. Pada kondisi awal kecepatan pendulum $-0,25$ rad/detik, respon sinyal kontrol mempunyai nilai maksimum *overshoot* dan maksimum *undershoot* sebesar $3,179$ Newton dan $52,5$ Newton serta memiliki *settling time* $2,618$ detik. Sedangkan pada pemberian kondisi awal kecepatan pendulum sebesar $-0,75$ rad/detik, respon sinyal kontrol memiliki *overshoot* dan *undershoot* yang jauh lebih kecil yaitu $0,49$ Newton dan $0,49$ Newton.

Tabel 4.11 Perbandingan Spesifikasi Transien Sinyal Kontrol Dengan Beberapa Kondisi Awal Kecepatan Pendulum

Kondisi Awal Kecepatan Pendulum (rad/detik)	<i>Overshoot</i> (N)	<i>Undershoot</i> (N)	<i>Settling Time</i> (detik)
0	4,958	79,73	2,689
-0,25	3,179	52,5	2,618
-0,50	1,431	25,27	2,488
-0,75	1,957	0,49	1,143

4.1.2 Hasil Simulasi *Faulty Case*

Jenis kesalahan aktuator berupa *bias* merupakan kesalahan yang bersifat aditif. Pada simulasi ini dilakukan pengujian dengan sinyal kesalahan berupa step dan ramp yang ditambahkan pada sinyal kontrol. Sehingga besarnya sinyal kontrol yang masuk ke *plant* tanpa adanya kompensasi dari FTC adalah:

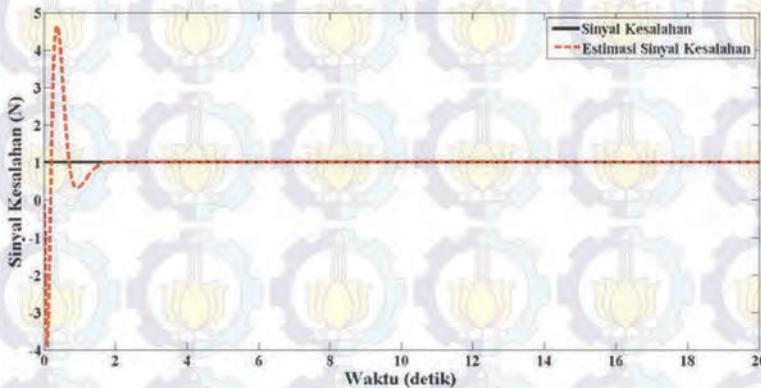
$$u_f = u + u_{f0}$$

dengan u_f adalah sinyal kontrol *faulty* yang masuk ke *plant*, u adalah sinyal kontrol yang merupakan keluaran dari kontroler, dan u_{f0} adalah sinyal kesalahan aktuator yang biasanya sering disebut f_a .

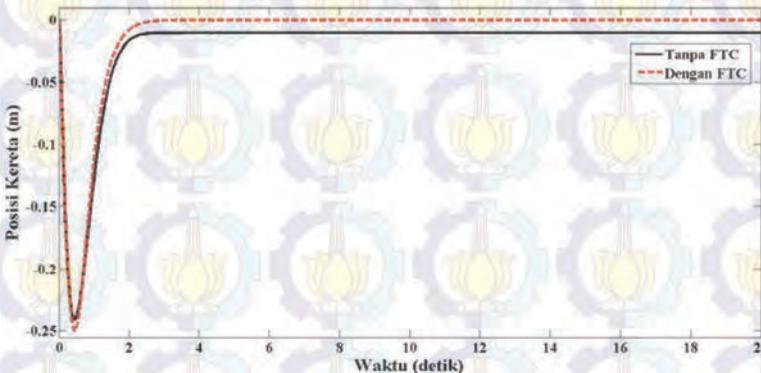
Pada pengujian pertama, sinyal kesalahan f_a yang diberikan adalah berupa sinyal *step* dengan magnitudo sebesar 1. Gambar 4.10 menunjukkan sinyal kesalahan aktuator beserta estimasinya. Dari gambar tersebut, observer dapat mengestimasi sinyal kesalahan dengan

tidak adanya *error* pada *steady state*. Akan tetapi pada kondisi transien, estimasi yang dihasilkan terdapat *undershoot* sebesar 3,86 Newton dan *overshoot* sebesar 4,40 Newton.

Gambar 4.11 menunjukkan respon posisi kereta dengan dan tanpa kompensasi kesalahan aktuator. Pada respon dengan kompensasi dapat menghasilkan sinyal posisi kereta kembali ke 0 meter, sedangkan respon tanpa kompensasi tidak dapat mengembalikan posisi kereta ke kondisi 0 meter, justru mengembalikannya ke posisi 0,015 meter.



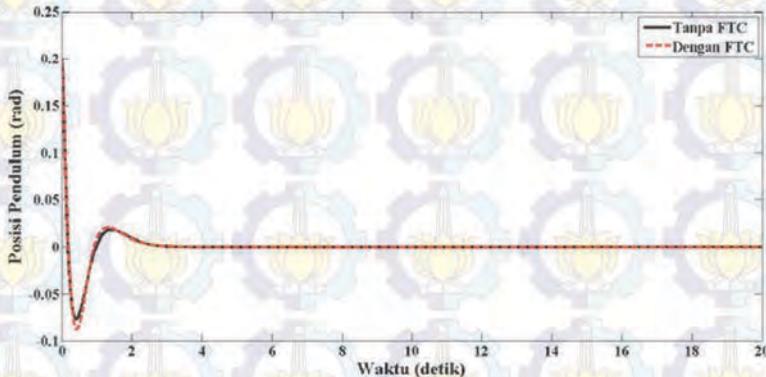
Gambar 4.10 Sinyal Kesalahan Step dan Estimasinya



Gambar 4.11 Respon Posisi Kereta Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan Step

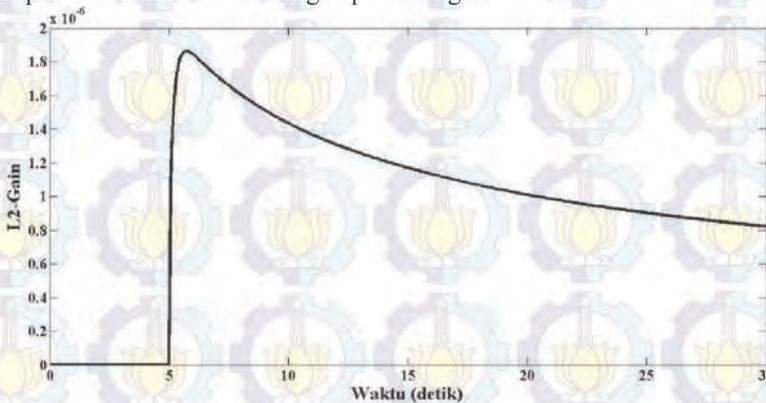
Perbandingan respon posisi pendulum dengan dan tanpa kompensasi kesalahan dapat dilihat pada Gambar 4.12. Kondisi *steady*

state dan settling time dari kedua respon mempunyai nilai yang sama. Akan tetapi pada kondisi transien saat ditambahkan kompensasi terdapat *undershoot* dan *overshoot* yang lebih besar yaitu 0,081 rad dan 0,02 rad sedangkan saat tanpa kompensasi mempunyai nilai yaitu 0,079 rad dan 0,019 rad.



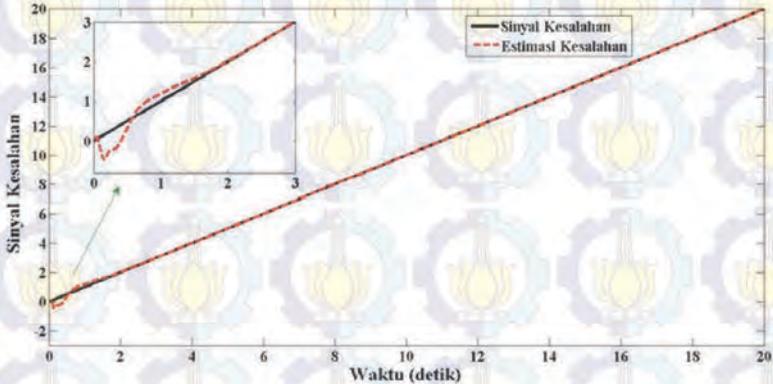
Gambar 4.12 Respon Posisi Pendulum Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan *Step*

Berdasarkan Gambar 4.13 dapat dilihat bahwa L_2 -gain memiliki nilai yang jauh lebih kecil daripada level pelemahan yang sebesar 0,135. Hal ini dapat disimpulkan bahwa efek dari sinyal kesalahan turunan ke-s dapat dilemahkan sesuai dengan perancangan.



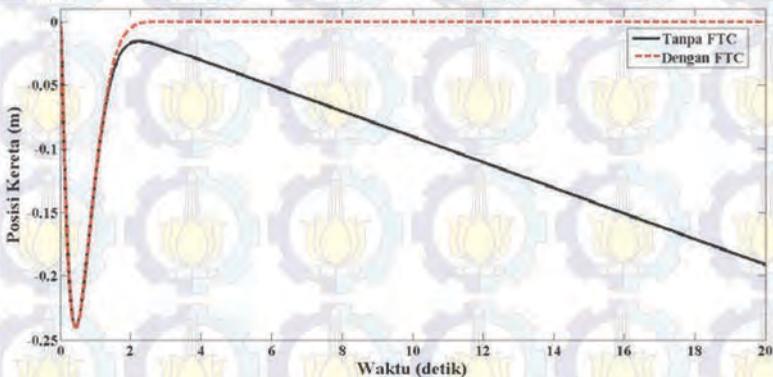
Gambar 4.13 L_2 -Gain untuk Kesalahan Berupa Sinyal *Step*

Pada pengujian kedua, sinyal kesalahan f_a pada Gambar 4.15 yang diberikan adalah sinyal *ramp* dengan kemiringan (*slope*) sebesar 1. Berdasarkan Gambar 4.14 terlihat bahwa estimasi kesalahan dapat mengikuti sinyal kesalahan setelah 2,15 detik.



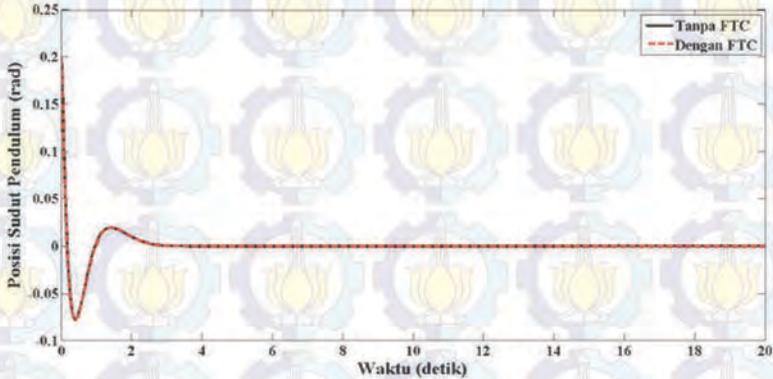
Gambar 4.14 Sinyal Kesalahan *Ramp* dan Estimasinya

Pada Gambar 4.15 menunjukkan respon posisi kereta dengan dan tanpa kompensasi. Pada kondisi transien kedua respon memiliki dinamika yang sama, akan tetapi pada saat memasuki daerah stabil (pada detik ke-2,05) respon tanpa kompensasi terlihat tidak bisa mengembalikan posisi kereta menuju 0 meter.



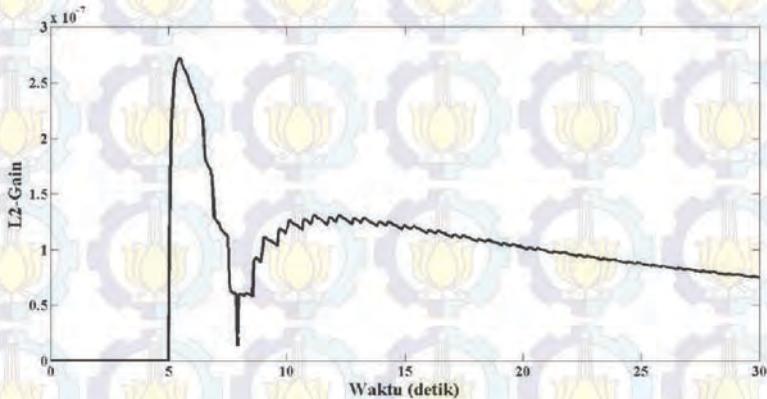
Gambar 4.15 Respon Posisi Kereta Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan *Ramp*

Respon tanpa kompensasi terlihat nilainya semakin menurun seperti mengikuti garis lurus. Sedangkan pada respon dengan kompensasi, posisi kereta pada daerah stabil dapat dikembalikan pada posisi 0 meter.



Gambar 4.16 Respon Posisi Pendulum Dengan dan Tanpa Kompensasi untuk Sinyal Kesalahan *Ramp*

Gambar 4.16 adalah respon posisi pendulum pada kondisi dengan dan tanpa kompensasi. Jika diperhatikan tidak terdapat perbedaan baik dari nilai *overshoot*, *undershoot*, maupun nilai *settling time* nya.



Gambar 4.17 L_2 -Gain untuk Kesalahan Berupa Sinyal *Ramp*

Berdasarkan Gmbar 4.17 dapat dilihat bahwa L_2 -gain memiliki nilai yang jauh lebih kecil daripada level pelemahan yang sebesar 0,135.

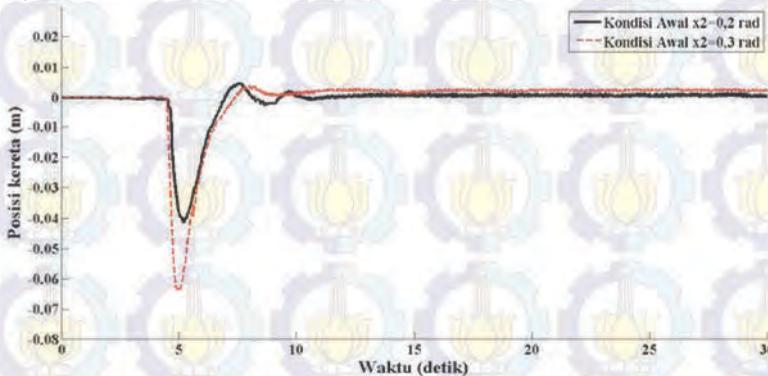
Hal ini dapat disimpulkan bahwa efek dari sinyal kesalahan turunan ke-s dapat dilemahkan sesuai dengan perancangan.

4.2 Hasil Implementasi

Pada tahap implementasi, batang pendulum diangkat manual dengan tangan (*hand-stand*) kemudian baru sistem kontrol berjalan. Seperti pada tahap simulasi, tahap implementasi ini akan diuji pada kondisi *fault free case* dan *faulty case*.

4.2.1 Hasil Implementasi *Fault Free Case*

Pada *fault free case*, sistem pendulum kereta akan diuji dengan beberapa nilai kondisi awal posisi pendulum. Dua kondisi awal yang diberikan adalah 0,2 radian dan 0,3 radian. Pada saat mengangkat batang pendulum ke atas menggunakan tangan secara manual (*hand-stand*), sehingga mempengaruhi adanya kecepatan awal pendulum yaitu sekitar -0,23 radian/detik.



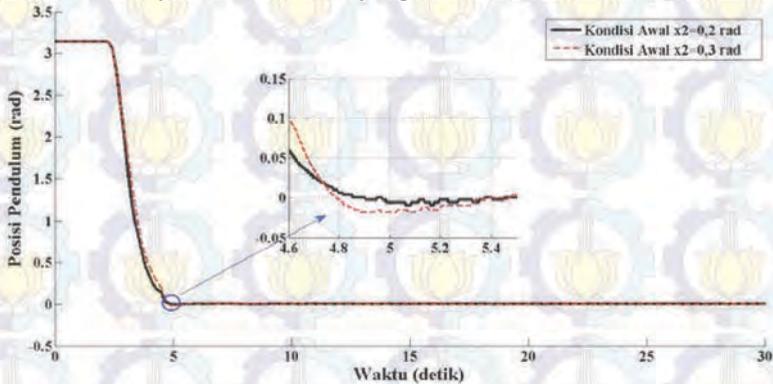
Gambar 4.18 Respon Posisi Kereta saat Implementasi *Fault Free Case*

Tabel 4.12 Perbandingan Spesifikasi Transien Respon Posisi Kereta saat Implementasi *Fault Free Case*

Kondisi Awal x_2 (rad)	<i>Undershoot</i> (m)	<i>Overshoot</i> (m)	<i>Settling Time</i> (det)
0,2	0,041	0,009	12,46
0,3	0,063	0,007	13,78

Perbandingan nilai *undershoot*, *settling time*, dan *steady state error* dari respon posisi kereta dengan kondisi awal posisi pendulum sebesar 0,2 radian dan 0,3 radian dapat dilihat pada Tabel 4.12.

Pada Tabel 4.12, nilai *undershoot* dan *overshoot* dari respon posisi kereta adalah sebesar 0,041 meter dan 0,009 meter saat posisi awal pendulum 0,2 radian serta 0,063 meter dan 0,007 meter. Hal tersebut diakibatkan semakin besar posisi awal pendulum, maka semakin besar pula *overshoot* dan *undershoot* dari posisi kereta karena dibutuhkan sinyal kontrol diawal yang semakin besar.



Gambar 4.19 Respon Posisi Pendulum saat Implementasi *Fault Free Case*

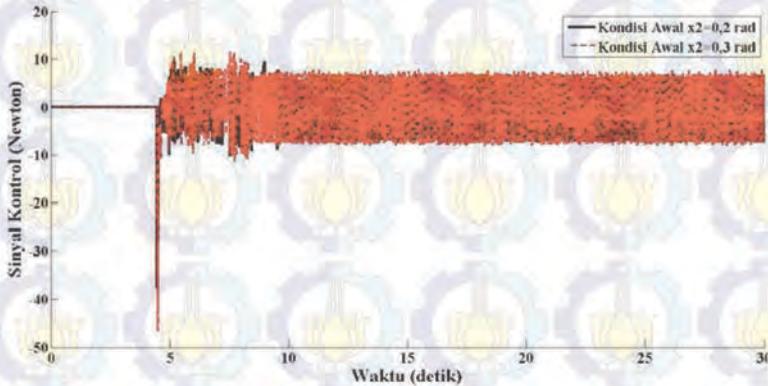
Perbandingan nilai *undershoot* dan *settling time* respon posisi pendulum Gambar 4.19 dapat dilihat pada Tabel 4.13. Pada posisi awal pendulum sebesar 0,2 radian, nilai *undershoot* dan *settling time* berturut-turut sebesar 0,005 radian dan 8,40 detik. Sedangkan pada saat posisi awal pendulum 0,3 radian mempunyai nilai *undershoot* dan *settling time* berturut-turut sebesar 0,008 radian dan 9,23 detik.

Tabel 4.13 Perbandingan Spesifikasi Transien Respon Posisi Pendulum saat Implementasi *Fault Free Case*

Kondisi Awal x_2 (rad)	<i>Undershoot</i> (rad)	<i>Settling Time</i> (det)
0,2	0,005	8,40
0,3	0,008	9,23

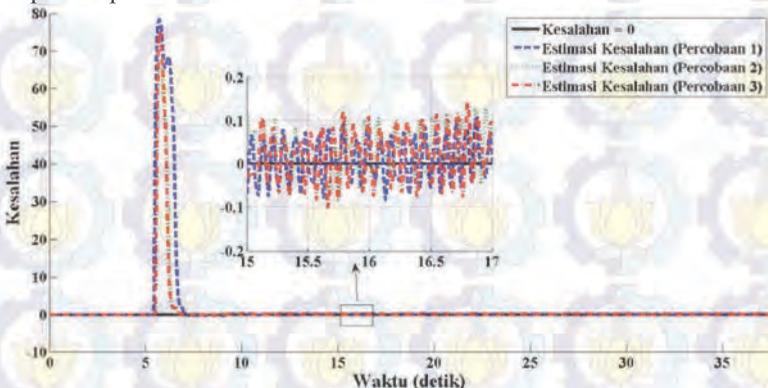
Gambar 4.20 adalah grafik sinyal kontrol saat posisi awal pendulum sebesar 0,2 radian dan 0,3 radian. Semakin besar posisi awal

pendulum mengakibatkan sinyal kontrol yang semakin besar. hal ini dikarenakan untuk mengembalikan posisi pendulum ke 0 radian membutuhkan sinyal kontrol yang lebih besar untuk posisi pendulum semakin besar. Pada kondisi awal posisi sudut pendulum 0,2 radian terdapat *undershoot* sinyal kontrol sebesar 37,92 Newton, sedangkan pada posisi awal pendulum di 0,3 radian terdapat *undershoot* sebesar 45,81 Newton.



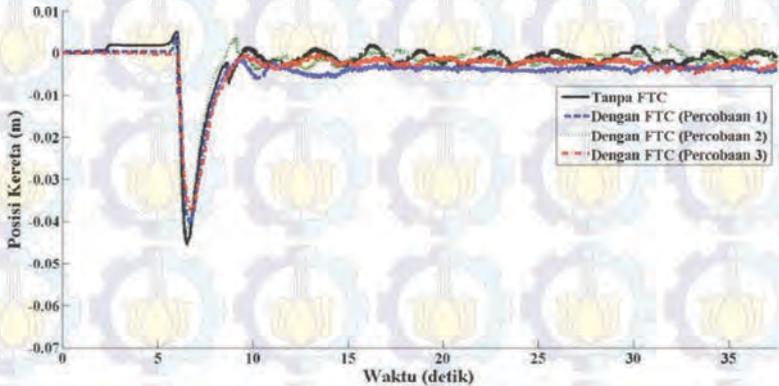
Gambar 4.20 Grafik Sinyal Kontrol saat Implementasi *Fault Free Case*

Kemudian pada pengujian selanjutnya akan dibandingkan apabila saat tidak terdapat kesalahan namun terdapat kompensasi dari FTC dan tanpa kompensasi FTC.



Gambar 4.21 Tanpa Kesalahan dan Estimasinya

Pada Gambar 4.21 dapat dilihat bahwa saat tidak terdapat kesalahan, atau dapat dikatakan terdapat kesalahan sebesar 0, maka FTC dapat mengestimasi kesalahan tersebut dengan *ripple* sebesar 0,1.

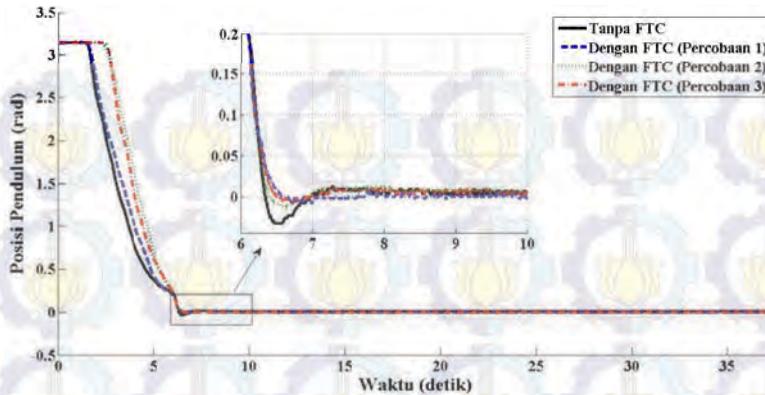


Gambar 4.22 Respon Posisi Kereta saat Implementasi Kondisi *Fault Free Case* Dengan dan Tanpa Kompensasi FTC

Gambar 4.22 menunjukkan perbandingan respon posisi kereta saat tidak dikompensasi FTC dan dengan dikompensasi FTC. Tabel 4.14 menunjukkan perbandingan spesifikasi respon posisi kereta. Pada saat diberikan kompensasi pada detik ke-15, terlihat bahwa tidak terdapat perbedaan pada kondisi *steady state*. Hal ini dikarenakan estimasi dari *observer* sudah menyerupai kesalahan yang diberikan (pada kasus ini kesalahannya adalah 0, atau tanpa kesalahan).

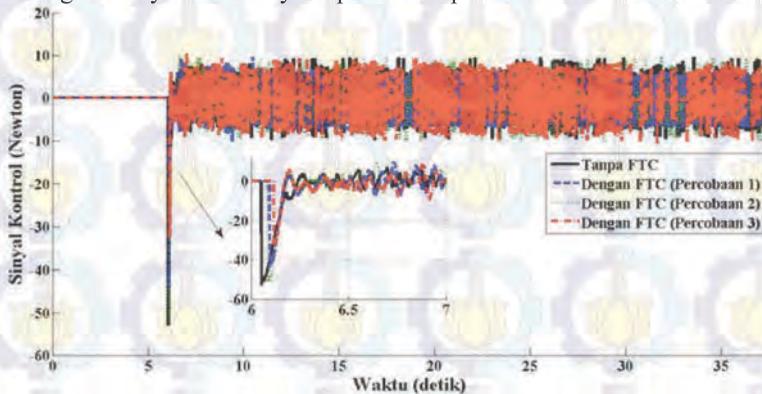
Tabel 4.14 Spesifikasi Respon Posisi Kereta Tanpa Kesalahan saat Tanpa dan Dengan Diberi Kompensasi FTC

Kondisi	<i>Undershoot</i> (rad)	<i>Settling Time</i> (det)
Tanpa FTC	0,045	12,40
Dengan FTC (Percobaan 1)	0,040	11,10
Dengan FTC (Percobaan 2)	0,041	10,92
Dengan FTC (Percobaan 3)	0,037	11,58



Gambar 4.23 Respon Posisi Pendulum saat Implementasi Kondisi *Fault Free Case* Dengan dan Tanpa Kompensasi FTC

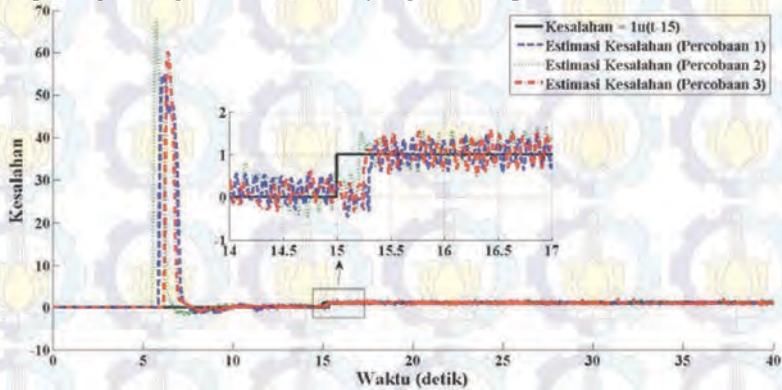
Gambar 4.23 merupakan respon posisi pendulum saat tidak terdapat kesalahan dengan dan tanpa kompensasi dari FTC. Pada kondisi *steady state* tidak terdapat perbedaan, sedangkan pada kondisi transien terdapat perbedaan sebesar 0.04 rad. Hal ini diakibatkan perbedaan kecepatan awal yang diberikan (karena pengaruh *hand-stand*). Sedangkan sinyal kontrolnya dapat dilihat pada Gambar 4.24.



Gambar 4.24 Sinyal Kontrol saat Implementasi Kondisi *Fault Free Case* Dengan dan Tanpa Kompensasi FTC

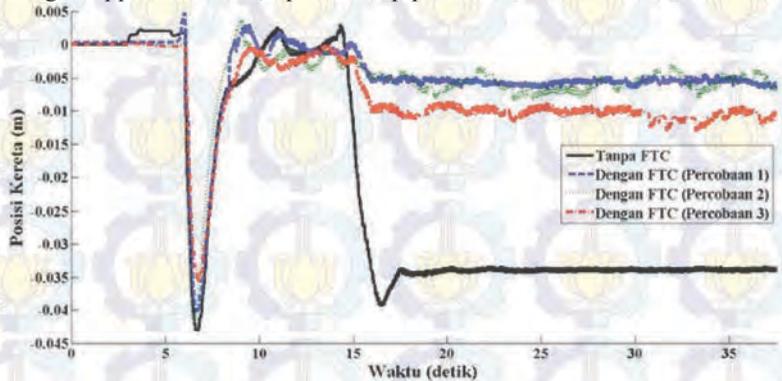
4.2.2 Hasil Implementasi *Faulty Case*

Pada implementasi saat *faulty case*, sinyal kesalahan yang diberikan adalah sinyal *step* dan *ramp*. Pada pengujian pertama, sinyal *step* dengan magnitudo sebesar 1 yang dimulai pada detik ke-15..



Gambar 4.25 Sinyal Kesalahan *Step* saat Implementasi

Gambar 4.25 adalah grafik sinyal kesalahan dan estimasinya. Estimasi kesalahan dapat mengikuti sinyal kesalahan setelah 0,3 detik; 0,20 detik; dan 0,26 detik berturut-turut pada percobaan 1, 2, dan 3 dengan *ripple* sebesar 0,1 pada setiap percobaan.



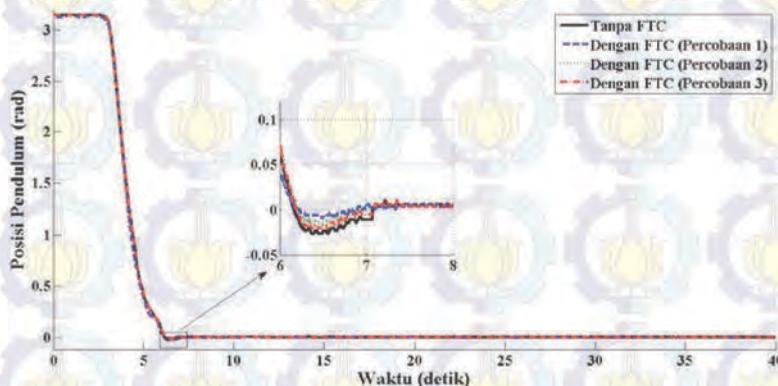
Gambar 4.26 Respon Posisi Kereta dengan Kesalahan *Step* saat Implementasi

Gambar 4.26 adalah respon posisi kereta dengan kesalahan *step* saat implemetasi tanpa adanya FTC dan dengan adanya FTC. Tabel 4.15 menunjukkan bahwa pada saat tanpa adanya skenario kontrol FTC,

posisi kereta tidak dapat kembali ke posisi 0 meter yaitu terdapat *steady state error* sebesar 0.032 meter. Hal ini berbeda jika dibandingkan dengan adanya FTC, *steady state error* hanya sebesar berturut-turut 0,006 meter; 0,008 meter; dan 0,010 meter pada percobaan 1, 2, dan 3.

Tabel 4.15 Perbandingan Respon Posisi Kereta dengan Kesalahan Step

Respon Posisi Kereta	<i>Undershoot</i> (m)	<i>Steady State Error</i> (m)	<i>Settling Time</i> (det)
Tanpa FTC	0,039	0,032	20,59
Dengan FTC (Percobaan 1)	0,008	0,006	22,61
Dengan FTC (Percobaan 2)	-	0,008	23,50
Dengan FTC (Percobaan 3)	-	0,010	22,73

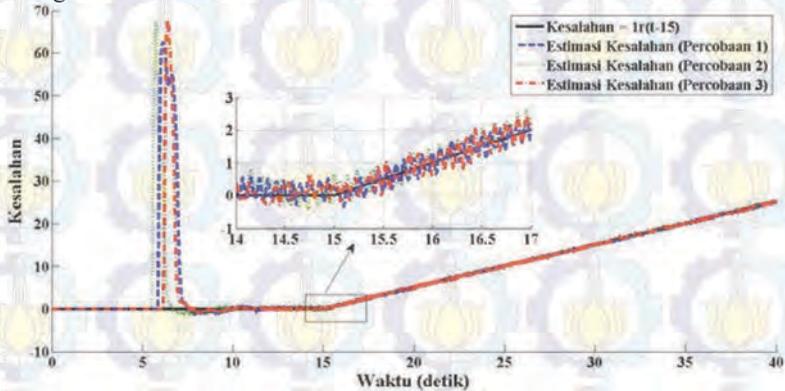


Gambar 4.27 Respon Posisi Pendulum Implementasi dengan Kesalahan *Step*

Sedangkan pada Gambar 4.27 adalah respon posisi pendulum dengan dan tanpa adanya skenario FTC. Tidak terdapat perbedaan yang signifikan. Hanya pada saat kondisi transien menuju *steady state* terdapat perbedaan sebesar 0.002 radian. Pada saat terjadi kesalahan (detik ke-15) posisi pendulum tetap berada di posisi 0 radian.

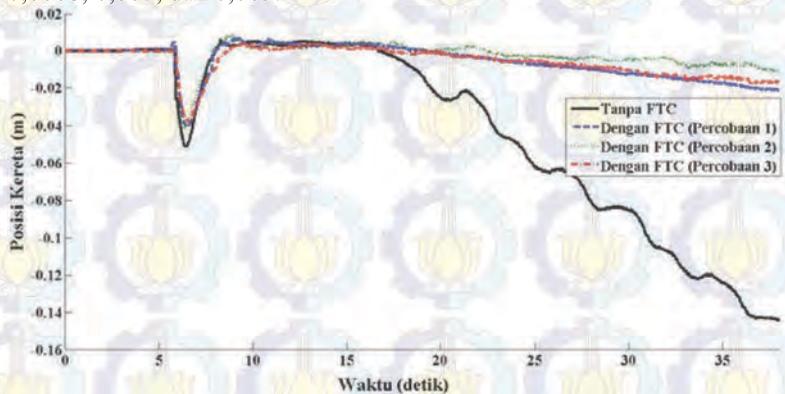
Pengujian yang kedua adalah kesalahan dengan sinyal *ramp*. Sinyal *ramp* yang diberikan mempunyai *slope* sebesar 1 dan dimulai

pada detik ke-15. Gambar 4.28 adalah grafik sinyal kesalahan dan estimasinya. Terlihat bahwa pada ketiga percobaan dengan FTC dapat mengestimasi kesalahan tersebut.



Gambar 4.28 Sinyal Kesalahan *Ramp* saat Implementasi

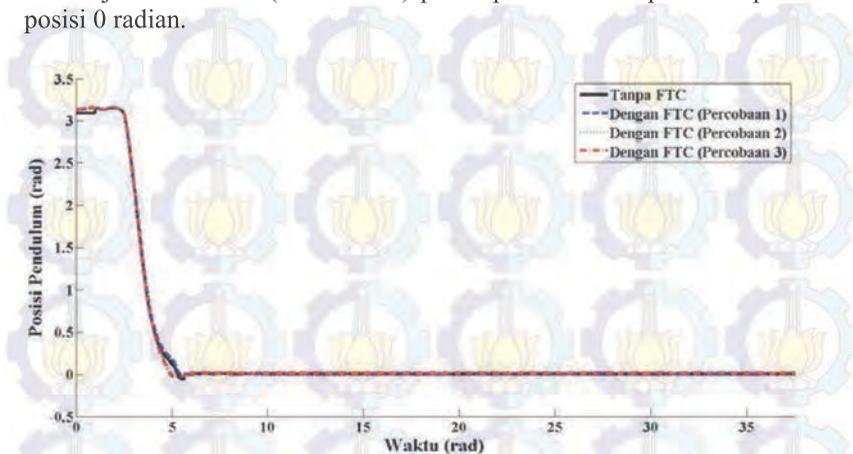
Gambar 4.29 adalah perbandingan respon posisi kereta dengan adanya kesalahan berupa sinyal ramp. Pada saat tanpa adanya FTC, posisi kereta semakin menjauh dari posisi 0 dengan perubahan *steady state error* sebesar 0,0064. Sedangkan pada saat diberi kompensasi dengan FTC perubahan *steady state error* hanya sebesar berturut-turut 0,0008; 0,007; dan 0,005.



Gambar 4.29 Respon Posisi Kereta dengan Kesalahan *Ramp* saat Implementasi

Respon posisi pendulum dapat dilihat pada Gambar 4.30. Sama seperti pada respon posisi pendulum saat terjadi kesalahan *step*, disini

pada saat terjadi kesalahan *ramp* juga tidak terdapat perbedaan. Pada saat terjadi kesalahan (detik ke-15) posisi pendulum tetap berada pada posisi 0 radian.



Gambar 4.30 Respon Posisi Pendulum Implementasi dengan Kesalahan *Ramp*

Dengan adanya pengujian kesalahan yang terjadi pada aktuator berupa sinyal *step* dan sinyal *ramp*, akibat yang ditimbulkan mempengaruhi performa dari respon posisi kereta. Hal ini dikarenakan sinyal kontrol adalah sinyal yang menggerakkan kereta dengan arah horizontal. Hal ini mengakibatkan jika terjadi kesalahan aktuator maka terjadi penurunan performa dari respon posisi kereta. Pada implementasi ini dengan adanya skenario kontrol FTC memang tidak dapat secara sempurna mengkompensasi dampak kesalahan yang terjadi. Dengan adanya FTC ini dapat mengurangi dampak penurunan performa pada respon posisi kereta.

BAB 5 PENUTUP

Bab ini berisi kesimpulan atas hasil pengujian yang telah dilakukan baik dari simulasi maupun implementasi dan saran untuk penelitian selanjutnya.

5.1 Kesimpulan

Dari hasil pengujian sistem kontrol yang telah dirancang pada simulasi, dapat diambil beberapa kesimpulan yaitu:

- Pada kondisi nominal, sistem kontrol LQR berbasis fuzzy T-S dengan PDC mampu melakukan stabilisasi pendulum pada posisi terbalik dan mempertahankan kereta pada titik tengah rel.
- Saat terdapat kesalahan aktuator berupa sinyal *step* dan *ramp*, observer mampu mengestimasi sinyal kesalahan tersebut.
- Pada saat kondisi terdapat kesalahan aktuator, hasil simulasi dengan kesalahan *step* maupun *ramp* menunjukkan bahwa adanya FTC dapat mengembalikan performa respon posisi kereta sesuai kondisi nominal. Respon posisi kereta saat tidak terdapat FTC tidak dapat mengembalikan posisi kereta ke posisi 0 meter. Sedangkan pada saat terdapat FTC, posisi kereta dapat kembali ke 0 meter.
- Pada implementasi, adanya FTC dapat mengurangi efek penurunan performa respon posisi kereta yaitu mengurangi *steady state error* sebesar hingga 81,25% untuk kesalahan berupa sinyal *step* dan mengurangi penurunan terhadap waktu dari *steady state error* sebesar 87,5% untuk kesalahan berupa sinyal *ramp*.

5.2 Saran

Saran untuk penelitian selanjutnya adalah model gaya gesek tidak diabaikan sehingga didapatkan hasil simulasi yang menyerupai sistem nyata.



-Halaman ini sengaja dikosongkan-

LAMPIRAN A

A.1 Penurunan Model Matematika Sistem Pendulum Kereta

Model matematika sistem pendulum kereta dapat diturunkan menggunakan rumus fisika yaitu tentang Hukum Newton. Gaya-gaya yang bekerja pada sistem pendulum kereta adalah gaya pada gerak translasi dan gaya pada gerak rotasi.

A.1.1 Penurunan Dinamika *State* Sistem Pendulum Kereta

Persamaan gerak translasi pada bidang horizontal adalah:

$$\sum F_x = ma_x \quad (\text{A.1})$$

Pada Gambar 2.9 (a) menunjukkan bahwa gaya translasi yang bekerja pada sistem pendulum kereta pada bidang horizontal adalah gaya dorong atau sinyal kontrol u dan gaya gesek T_c . Gaya dorong u memiliki arah yang searah dengan gerak benda, sedangkan gaya gesek T_c memiliki arah yang berlawanan dengan arah gerak benda. Sehingga tanda atau notasi pada u adalah positif, sedangkan T_c adalah negatif. Nilai dari m adalah massa total yang berpengaruh pada sistem gaya translasi tersebut, sehingga m adalah jumlah antara massa pendulum m_p ditambahkan dengan massa kereta m_c . Sedangkan a_x adalah percepatan pada bidang horizontal akibat adanya gaya translasi yang bekerja pada bidang tersebut. Nilai a_x dapat ditentukan dengan (A.2).

$$a_x = (x_1 - l \sin x_2)'' \quad (\text{A.2})$$

Dari (A.1) dan (A.2) dapat dijabarkan menjadi:

$$\begin{aligned} u - T_c &= (m_c + m_p)(x_1 - l \sin x_2)'' \\ u - T_c &= (m_c + m_p)(x_1' - x_2' l \cos x_2)' \\ u - T_c &= (m_c + m_p)(x_1'' - x_2'' l \cos x_2 + (x_2')^2 l \sin x_2) \end{aligned} \quad (\text{A.3})$$

Persamaan gerak translasi pada bidang vertikal adalah:

$$\sum F_y = ma_y \quad (\text{A.4})$$

Pada Gambar 2.9 (b) menunjukkan bahwa gaya translasi pada bidang vertikal yang bekerja pada sistem pendulum kereta adalah gaya normal V dan gaya berat. Gaya berat memiliki nilai $(m_c + m_p)g$, sedangkan nilai a_y ditentukan oleh (A.5).

$$a_y = (l \cos x_2)'' \quad (\text{A.5})$$

Dengan menjabarkan (A.4) dan (A.5) maka didapatkan:

$$\begin{aligned} V - (m_c + m_p)g &= (m_c + m_p)(l \cos x_2)'' \\ V &= (m_c + m_p)g + (m_c + m_p)(-lx_2' \sin x_2)' \\ V &= (m_c + m_p)g + (m_c + m_p)(-lx_2'' \sin x_2 - l(x_2')^2 \cos x_2) \\ V &= (m_c + m_p)(g - lx_2'' \sin x_2 - l(x_2')^2 \cos x_2) \end{aligned} \quad (\text{A.6})$$

Persamaan gerak rotasi adalah:

$$\sum \tau = J\alpha \quad (\text{A.7})$$

Gambar 2.9 (c) menunjukkan gaya-gaya yang bekerja secara keseluruhan pada gerak rotasi. D_p adalah momen gesek akibat pergerakan rotasi batang pendulum yang besarnya sebanding dengan kecepatan sudut pendulum. Sehingga D_p dapat dinyatakan sebagai $D_p = f_p x_4$. J adalah momen inersia dari batang pendulum.

$$\begin{aligned} (u - T_c)l \cos x_2 + Vl \sin x_2 - D_p &= Jx_2'' \\ (u - T_c)l \cos x_2 + Vl \sin x_2 - f_p x_4 &= Jx_2'' \end{aligned} \quad (\text{A.8})$$

Setelah mendapatkan hubungan gaya-gaya yang bekerja pada sistem pendulum kereta, kemudian dapat dicari dinamika dari *state* x_4 dengan melakukan substitusi (A.6) ke (A.8).

$$\begin{aligned} Jx_2'' &= (u - T_c)l \cos x_2 - f_p x_4 \\ &+ (m_c + m_p)(g - lx_2'' \sin x_2 - l(x_2')^2 \cos x_2)l \sin x_2 \end{aligned} \quad (\text{A.9})$$

$$\begin{aligned} Jx_4' &= (u - T_c)l \cos x_2 - f_p x_4 + (m_c + m_p)l \\ &(g \sin x_2 - lx_4' \sin^2 x_2 - l(x_2')^2 \cos x_2 \sin x_2) \\ Jx_4' &= (u - T_c - \mu x_4'^2 \sin x_2)l \cos x_2 + \mu g \sin x_2 \\ &- \mu lx_4' \sin^2 x_2 - f_p x_4 \end{aligned}$$

$$x_4'(J + \mu l \sin^2 x_2) = (u - T_c - \mu x_4^2 \sin x_2)l \cos x_2 + \mu g \sin x_2 - f_p x_4$$

$$x_4' = \frac{(u - T_c - \mu x_4^2 \sin x_2)l \cos x_2 + \mu g \sin x_2 - f_p x_4}{J + \mu l \sin^2 x_2} \quad (\text{A.10})$$

dengan $\mu = (m_c + m_p)l$.

Dinamika dari *state* x_3 dapat diturunkan dari (A.3) seperti persamaan dibawah ini:

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

$$(m_c + m_p)x_3' = u - T_c - \mu x_4^2 \sin x_2 + \mu x_4' \cos x_2 \quad (\text{A.11})$$

Kemudian dilakukan substitusi (A.10) ke (A.11) dan mendefinisikan nilai $a = l^2 + J/(m_c + m_p)$, sehingga:

$$(m_c + m_p)x_3' = u - T_c - \mu x_4^2 \sin x_2 + \frac{\mu l \cos^2 x_2 (u - T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2} + \frac{\mu \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2}$$

$$(m_c + m_p)x_3' = \frac{(J + \mu l \sin^2 x_2)(u - T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2} + \frac{\mu l \cos^2 x_2 (u - T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2} + \frac{\mu \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2}$$

$$(m_c + m_p)x_3' = \frac{(J + \mu l)(u - T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2} + \frac{\mu \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2}$$

$$x_3' = \frac{(J + \mu l)l}{(m_c + m_p)l} \frac{(u - T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2} + \frac{\mu l \cos x_2 (\mu g \sin x_2 - f_p x_4)}{(m_c + m_p)l(J + \mu l \sin^2 x_2)}$$

$$x_3' = \left(l^2 + \frac{J}{(m_c + m_p)} \right) \frac{(u - T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2} + \frac{l \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2}$$

$$x_3' = \frac{a(u - T_c - \mu x_4^2 \sin x_2) + l \cos x_2 (\mu g \sin x_2 - f_p x_4)}{J + \mu l \sin^2 x_2} \quad (\text{A.12})$$

Model matematika sistem pendulum kereta secara keseluruhan dapat dilihat pada (2.10)

A.1.2 Elemen Matriks A dan B pada Sistem Pendulum Kereta

Untuk mendapatkan model linear dari sistem pendulum kereta, maka dapat didekati dengan teori linearisasi pada dua titik kerja. Persamaan (2.10) dapat dituliskan menjadi:

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

dan untuk persamaan *state space* dapat dituliskan:

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

sehingga diperoleh matriks **A** dan **B** adalah:

$$\mathbf{A} = \begin{bmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \frac{\partial f_1(x)}{\partial x_3} & \frac{\partial f_1(x)}{\partial x_4} \\ \frac{\partial f_2(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_2} & \frac{\partial f_2(x)}{\partial x_3} & \frac{\partial f_2(x)}{\partial x_4} \\ \frac{\partial f_3(x)}{\partial x_1} & \frac{\partial f_3(x)}{\partial x_2} & \frac{\partial f_3(x)}{\partial x_3} & \frac{\partial f_3(x)}{\partial x_4} \\ \frac{\partial f_4(x)}{\partial x_1} & \frac{\partial f_4(x)}{\partial x_2} & \frac{\partial f_4(x)}{\partial x_3} & \frac{\partial f_4(x)}{\partial x_4} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \frac{\partial h_1(x, u)}{\partial u} \\ \frac{\partial h_2(x, u)}{\partial u} \\ \frac{\partial h_3(x, u)}{\partial u} \\ \frac{\partial h_4(x, u)}{\partial u} \end{bmatrix}$$

Elemen dari matriks **A** dan **B** adalah:

$$\frac{\partial f_1(x)}{\partial x_1} = 0, \quad \frac{\partial f_1(x)}{\partial x_2} = 0, \quad \frac{\partial f_1(x)}{\partial x_3} = 1, \quad \frac{\partial f_1(x)}{\partial x_4} = 0,$$

$$\frac{\partial f_2(x)}{\partial x_1} = 0, \quad \frac{\partial f_2(x)}{\partial x_2} = 0, \quad \frac{\partial f_2(x)}{\partial x_3} = 0, \quad \frac{\partial f_2(x)}{\partial x_4} = 1,$$

$$\frac{\partial f_3(x)}{\partial x_1} = 0,$$

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

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

$$\frac{\partial f_3(x)}{\partial x_3} = \frac{a}{J + \mu l \sin^2 x_2} \frac{\partial T_c}{\partial x_3}, \quad \frac{\partial f_3}{\partial x_4} = \frac{-2a\mu x_4 \sin x_2 - f_p l \cos x_2}{J + \mu l \sin^2 x_2}$$

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

$$\frac{\partial f_4}{\partial x_2} = \frac{g\mu \cos x_2 - \mu l x_4^2 \cos^2 x_2 - l \sin x_2 (-T_c - \mu x_4^2 \sin x_2)}{J + \mu l \sin^2 x_2}$$

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

$$\frac{\partial f_4(x)}{\partial x_3} = \frac{l \cos x_2}{J + \mu l \sin^2 x_2} \frac{\partial T_c}{\partial x_3}, \quad \frac{\partial f_4}{\partial x_4} = \frac{-2\mu l x_4 \cos x_2 \sin x_2 - f_p}{J + \mu l \sin^2 x_2}$$

$$\frac{\partial h_1}{\partial u} = 0 \quad \frac{\partial h_2}{\partial u} = 0 \quad \frac{\partial h_3}{\partial u} = \frac{a}{J + \mu l \sin^2 x_2} \quad \frac{\partial h_4}{\partial u} = \frac{l \cos x_2}{J + \mu l \sin^2 x_2}$$

A.1.3 Penurunan Rumus Gaya Gesek pada Sistem Pendulum Kereta

Kereta yang diam tidak memiliki gaya gesek dengan rel. Namun, kereta yang mulai bergerak memiliki gaya gesek statis (*static friction*) dengan rel. Setelah kereta bergerak, kereta memiliki gaya gesek dinamis (*coulomb friction*) yang bersifat tidak linear terhadap kecepatan kereta.

Dari Gambar A.3, F_s adalah gaya gesek statis (*static friction*), F_c adalah gaya gesek dinamis (*coulomb friction*), x_c adalah kecepatan kereta saat gaya gesek memasuki daerah linear, Y_c adalah gaya gesek pada titik x_c , dan DZ_{cv} adalah *dead-zone* untuk kecepatan kereta. Selain

itu, dari Gambar A.3, tampak bahwa gaya gesek dapat dibagi dalam lima daerah. Daerah 1 untuk $x_3 < -x_c$, daerah 2 untuk $-x_c \leq x_3 < -DZ_{cv}$, daerah 3 untuk $-DZ_{cv} \leq x_3 \leq DZ_{cv}$, daerah 4 untuk $DZ_{cv} < x_3 \leq x_c$, dan daerah 5 untuk $x_c < x_3$.

Karena daerah 1 dan 2 merupakan pencerminan dari daerah 4 dan 5, maka hanya dibahas tiga daerah saja, yaitu daerah 3, 4, dan 5. Pada daerah 3, gaya gesek merupakan fungsi linear terhadap kecepatan kereta yang memiliki persamaan sebagai berikut

$$T_c(x_3) = \frac{F_s}{DZ_{cv}} x_3 \quad (\text{A.13})$$

Untuk daerah 4, gaya gesek merupakan fungsi kuadratik terhadap kecepatan kereta sehingga persamaan gaya gesek pada daerah 4 adalah ditunjukkan pada (A.14).

$$T_c(x_3) = \frac{(F_s - Y_c)}{x_c^2} x_3^2 - 2 \frac{(F_s - Y_c)}{x_c} x_3 + F_s \quad (\text{A.14})$$

Sedangkan untuk daerah 5, gaya gesek merupakan fungsi linear terhadap kecepatan kereta dengan persamaan seperti pada (A.15)

$$T_c(x_3) = \frac{(Y_c - F_c)}{x_c} x_3 + F_c \quad (\text{A.15})$$

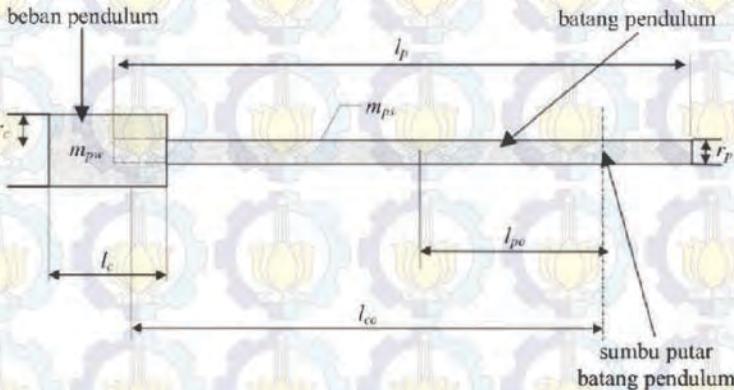
Dari (A.13-15) didapatkan persamaan gaya gesek secara keseluruhan seperti pada (A.16).

$$T_c(x_3) = \begin{cases} \frac{(Y_c - F_c)}{x_c} x_3 - F_c, & x_3 < -x_c \\ \frac{(Y_c - F_c)}{x_c^2} x_3^2 + 2 \frac{(Y_c - F_c)}{x_c} x_3 - F_s, & -x_c \leq x_3 < -DZ_{cv} \\ \frac{F_s}{DZ_{cv}} x_3, & -DZ_{cv} \leq x_3 \leq DZ_{cv} \\ \frac{(F_s - Y_c)}{x_c^2} x_3^2 - 2 \frac{(F_s - Y_c)}{x_c} x_3 + F_s, & DZ_{cv} < x_3 \leq x_c \\ \frac{(Y_c - F_c)}{x_c} x_3 + F_c, & x_c < x_3 \end{cases} \quad (\text{A.16})$$

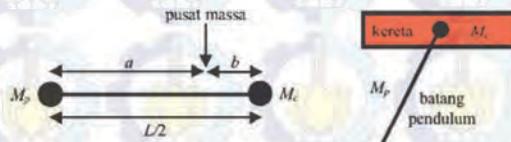
A.1.4 Penurunan Persamaan Momen Inersia pada Sistem Pendulum Kereta

Momen inersia (J) dari sistem pendulum kereta diperoleh dari hasil analisis momen inersia yang diilustrasikan pada Gambar A.1. Dari Gambar A.1, m_{pw} adalah massa beban pendulum (kg) sedangkan 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.1 Ilustrasi Pendulum



Gambar A.2 Ilustrasi Momen Inersia Sistem Pendulum Kereta

$$J_p = m_{pw} \left(\frac{1}{12} l_c^2 + \frac{1}{4} r_c^2 + l_{co}^2 \right) + m_{ps} \left(\frac{1}{12} l_p^2 + \frac{1}{4} r_p^2 + l_{po}^2 \right) \quad (\text{A.13})$$

Jika momen inersia dari sumbu tertentu terhadap titik pusat massa dinyatakan dengan J dan momen inersia dari sumbu yang paralel dengan sumbu sebelumnya dan dipisahkan oleh jarak sebesar d dinyatakan dengan J_p , maka hubungan antara J dan J_p adalah:

$$\begin{aligned} J_p &= J + Md^2 \\ M_p a &= M_c b \\ a + b &= L/2 \\ J &= M_p a^2 + M_p b^2 + \frac{M_p}{12} L^2 \end{aligned} \quad (\text{A.14})$$

dengan

$$a = \frac{LM_c}{2(M_p + M_c)} \text{ dan } b = \frac{LM_p}{2(M_p + M_c)}$$

Jika (A.14) dijabarkan lebih lanjut, maka akan didapatkan persamaan bentuk terakhir dari momen inersia seperti yang ditunjukkan pada (A.15).

$$J = \frac{L^2 m_p (4m_c + m_p)}{12(m_c + m_p)} \quad (\text{A.15})$$

Sedangkan jarak pusat massa terhadap sumbu putar sistem pendulum kereta adalah:

$$l = \frac{m_p L}{2(m_c + m_p)} \quad (\text{A.16})$$

A.1.5 Parameter Sistem Pendulum Kereta

Parameter-parameter yang digunakan untuk pemodelan sistem pendulum kereta dapat dilihat pada Tabel A.1.

Tabel A.1 Parameter-Parameter Sistem Pendulum Kereta

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
Jarak pusat 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 (<i>Coulomb</i>) (N)	F_c	2,2813
Gaya kontrol maksimal (N)	u_{max}	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)	l	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

LAMPIRAN B

B.1 Program untuk Mencari Nilai *State Feedback Gain Controller*

```
clear all
clc

%% Parameter Plant
Mc = 1.12; %massa kereta
mpw = 0.095; %massa batang pendulum
mps = 0.025; %massa beban pendulum
Rl = 1; %Panjang rel
L = 0.402; %panjang batang pendulum
fp = 0.0001; %gaya gesek pendulum
g = 9.8; %percepatan gravitasi
J = 0.0136; %momen inersia
l = 0.0168; %jarak pusat massa ke sumbu rotasi
Mp = mpw+mps;
miu = (Mc+Mp)*l;
a = l^2+J/(Mc+Mp);

%% Linierisasi
[A1,B1,C1,D1] = linmod('Model',[0 0 0 0]',0);
[A2,B2,C2,D2] = linmod('Model',[0 0.2 0 0]',0);

%% State Feedback Gain Controller dengan LQR
Q=[5000 0 0 0;0 1000 0 0;0 0 500 0;0 0 0 500];
R=0.5;
[KL1,S1,E1]=lqr(A1,B1,Q,R);
[KL2,S2,E2]=lqr(A2,B2,Q,R);

%% Menampilkan Nilai Gain Controller
disp('Didapatkan K1 = ')
disp(KL1)
disp('Didapatkan K2 = ')
disp(KL2)
```

B.2 Program untuk Mencari Nilai *Observer Gain*

```
%% Parameter LMI
Albar = [A1 zeros(4,2) B1; zeros(1,7);
         zeros(1,4) 1 0 0; zeros(1,4) 0 1 0];
A2bar = [A2 zeros(4,2) B2; zeros(1,7);
         zeros(1,4) 1 0 0; zeros(1,4) 0 1 0];

B1bar = [B1; 0; 0; 0];
B2bar = [B2; 0; 0; 0];

C1bar = [C1 0 0 0];
C2bar = [C2 0 0 0];

Gbar = [zeros(4,1); 1; 0; 0];

Cp = 0.01*eye(7);
gammao = 5;
sintet = 0.4;
costet = sqrt(1-(sintet^2));

%% LMI Observer 1
setlmis([]);

p1=lmivar(1,[7 1]);
h1=lmivar(2,[7 1]);
gamma1=lmivar(2,[1 1]);

lmiterm([1 1 1 p1],1,Albar,'s');
lmiterm([1 1 1 h1],1,-C1bar,'s');
lmiterm([1 1 2 p1],1,Gbar);
lmiterm([1 1 3 0],Cp);
lmiterm([1 2 2 gamma1],1,-1);
lmiterm([1 2 3 0],0);
lmiterm([1 3 3 gamma1],1,-1);
lmiterm([2 1 1 p1],1,-1);
lmiterm([3 1 1 p1],Albar',1,'s');
lmiterm([3 1 1 h1],1,-C1bar,'s');
lmiterm([3 1 1 p1],2*gammao,1);
lmiterm([4 1 1 p1],sintet*Albar',1,'s');
```

```

lmiterm([4 1 1 h1],sintet,-C1bar,'s');
lmiterm([4 1 2 p1],-costet*A1bar',1);
lmiterm([4 1 2 p1],costet,A1bar);
lmiterm([4 1 2 h1],costet,-C1bar);
lmiterm([4 1 2 -h1],costet*C1bar',1);
lmiterm([4 2 2 p1],sintet*A1bar',1,'s');
lmiterm([4 2 2 h1],sintet,-C1bar,'s');

PMIOobs1=getlmis;
[tmin, xfeas] = feasp(PMIOobs1);

P1 = dec2mat(PMIOobs1, xfeas, p1);
H1 = dec2mat(PMIOobs1, xfeas, h1);
Gamma1 = dec2mat(PMIOobs1, xfeas, gamma1);

L1 = inv(P1)*H1;

%% LMI Observer 2
setlmis([]);

p2=lmivar(1,[7 1]);
h2=lmivar(2,[7 1]);
gamma2=lmivar(2,[1 1]);

lmiterm([1 1 1 p2],1,A2bar,'s');
lmiterm([1 1 1 h2],1,-C2bar,'s');
lmiterm([1 1 2 p2],1,Gbar);
lmiterm([1 1 3 0],Cp);
lmiterm([1 2 2 gamma2],1,-1);
lmiterm([1 2 3 0],0);
lmiterm([1 3 3 gamma2],1,-1);
lmiterm([2 1 1 p2],1,-1);
lmiterm([3 1 1 p2],A2bar',1,'s');
lmiterm([3 1 1 h2],1,-C2bar,'s');
lmiterm([3 1 1 p2],2*gammao,1);
lmiterm([4 1 1 p2],sintet*A2bar',1,'s');
lmiterm([4 1 1 h2],sintet,-C2bar,'s');
lmiterm([4 1 2 p2],-costet*A2bar',1);
lmiterm([4 1 2 p2],costet,A2bar);

```

```

lmiterm([4 1 2 h2],costet,-C2bar);
lmiterm([4 1 2 -h2],costet*C2bar',1);
lmiterm([4 2 2 p2],sintet*A2bar',1,'s');
lmiterm([4 2 2 h2],sintet,-C2bar,'s');

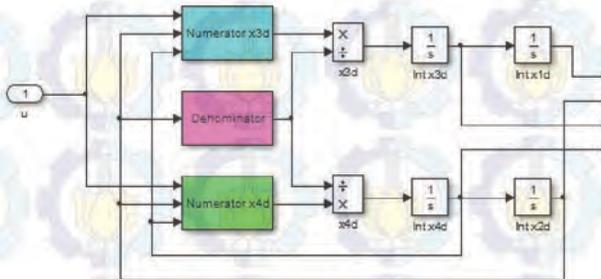
PMIOobs2=getlmis;
[tmin, xfeas] = feasp(PMIOobs2);

P2 = dec2mat(PMIOobs2, xfeas, p2);
H2 = dec2mat(PMIOobs2, xfeas, h2);
Gamma2 = dec2mat(PMIOobs2, xfeas, gamma2);
L2 = inv(P2)*H2;

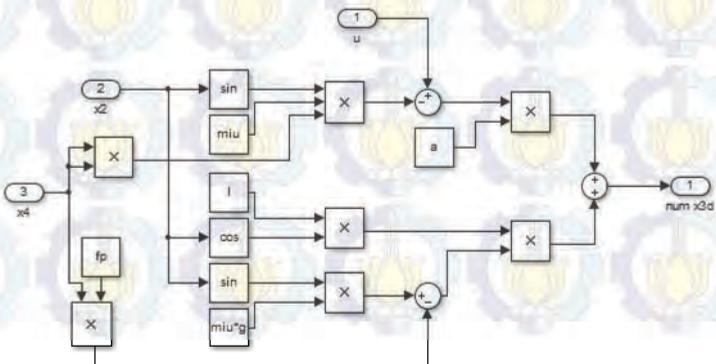
```

B.3 Blok Diagram Simulink

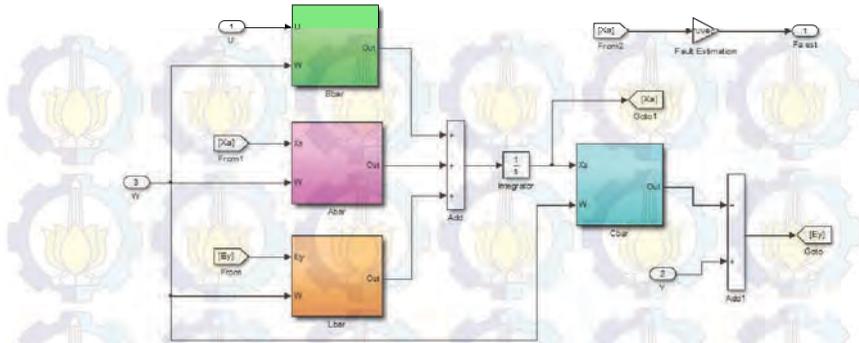
1. Blok Sistem Pendulum Kereta



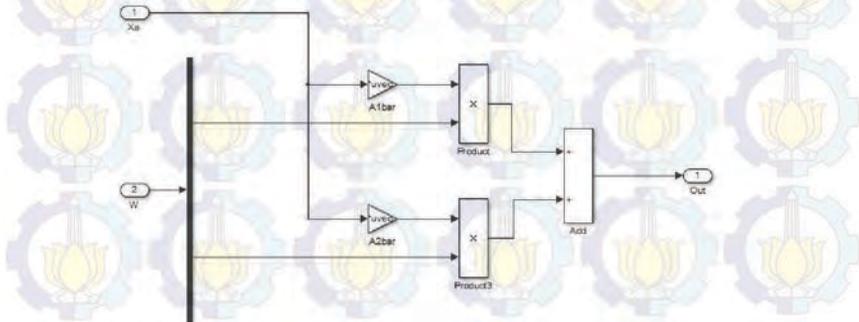
2. Sub-blok Numerator x3d (\dot{x}_3)



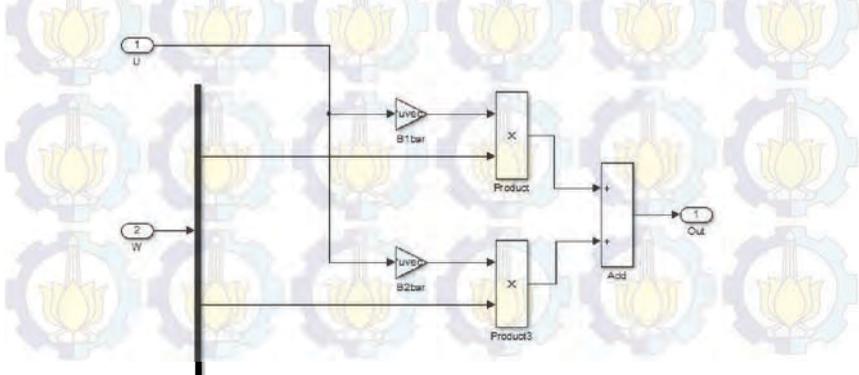
6. Blok Observer



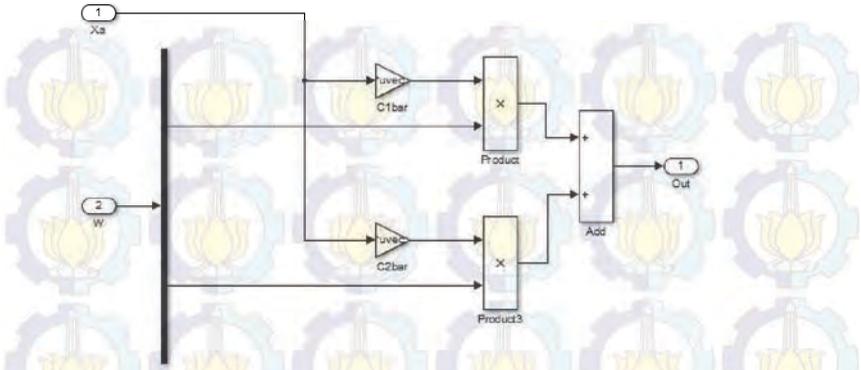
7. Sub-blok Abar



8. Sub-blok Bbar



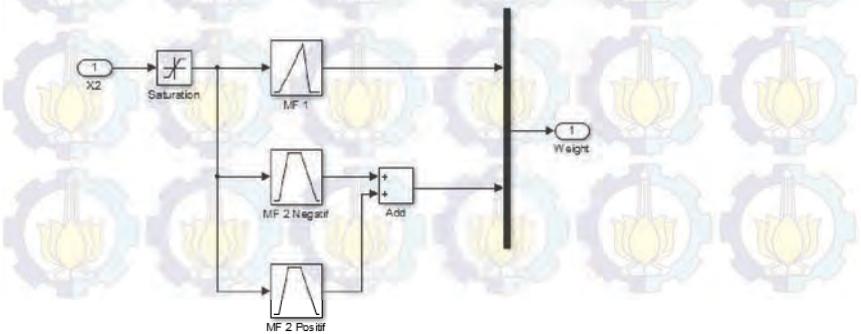
9. Sub-blok Cbar



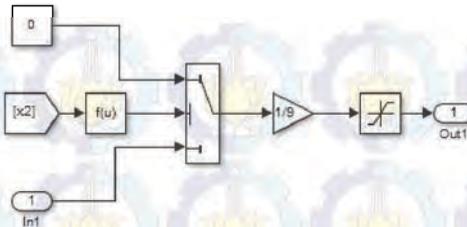
10. Sub-blok Lbar



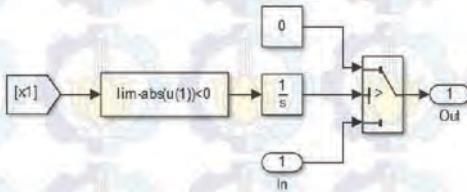
11. Blok Membership Function



12. Blok *Scaling and Safety*



13. Blok *Rail Limiter*





-Halaman ini sengaja dikosongkan-

DAFTAR PUSTAKA

- [1] H.O. Wang, K. Tanaka, dan M. Griffin, "Parallel Distributed Compensation of Nonlinear Systems by Takagi-Sugeno Fuzzy Model," *IEEE Fuzzy System*, pp. 531-538, 1995.
- [2] Naidu, D. S., "*Optimal Control System*", Idaho State University, Pocatello, Idaho, USA, 2002.
- [3] Shaker, M. S., "*Active Fault-Tolerant Control of Nonlinear Systems with Wind Turbine Application*", MSc Electronics, University of Hull, Baghdad, 2012.
- [4] Noura, H., Sauter, D., dan Hamelin, F., "Fault-tolerant control in dynamic systems: Application to a winding machine". *IEEE Control Systems Magazines* 20, pp. 33-49, 2000.
- [5] _____. "*Digital Pendulum: Control in a MATLAB Environment*", Feedback Instrument Ltd., England, 2004.
- [6] Ogata, K., "*Modern Control Engineering 3rd ed*", Prentice-Hall, New Jersey, 1997.
- [7] Passino, K. M., dan Yurkovich, S., "*Fuzzy Control*", Addison Wesley Longman, California, 1998.
- [8] Boyd, S., Ghaoui, L. E., Feron, E., dan Balakrishnan, V., "*Linear Matrix Inequalities in System and Control Theory*", Society for Industrial and Applied Mathematics, Philadelphia, 1994.
- [9] Ghosh, S., "*Low Gain Controller Design with Regional Pole Placement Constraints*", Odisha, 2013.



-Halaman ini sengaja dikosongkan-

RIWAYAT HIDUP



Mujianto Purnomo Saputro lahir di Madiun pada 23 Mei 1992. Penulis telah menyelesaikan pendidikannya di TK Dharmawanita 2 Jiwan, SDN Jiwan 1 Kab Madiun, SMPN 2 Madiun, dan SMAN 3 Madiun. Pada tahun 2011, penulis melanjutkan studi di Institut Teknologi Sepuluh Nopember (ITS), Jurusan Teknik Elektro melalui Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN) jalur undangan. Di Jurusan Teknik Elektro ini, penulis memilih Teknik Sistem Pengaturan sebagai bidang studi yang ditekuni. Pada bulan Juni 2015, penulis mengikuti seminar dan ujian Tugas Akhir sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro.

