



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE 141599

**KONTROL *TRACKING FUZZY* BERBASIS PERFORMA
ROBUST UNTUK *QUADROTOR***

Dinang Sohendri
NRP 2212100097

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

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT -TE 141599

**FUZZY TRACKING CONTROL BASED ON ROBUST
PERFORMANCE OF A QUADROTOR**

Dinang Sohendri
NRP 2212 100 097

Supervisor
Dr. Trihastuti Agustinah, ST., MT.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2016

TUGAS AKHIR

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

Menyetujui :

Dosen Pembimbing



Dr. Trihastuti Agustinah, ST., MT.
NIP:196808121994032001



**LEMBAR PERNYATAAN
PERSETUJUAN PUBLIKASI KARYA ILMIAH
UNTUK KEPENTINGAN AKADEMIS**

Sebagai mahasiswa Institut Teknologi Sepuluh Nopember Surabaya, yang bertanda tangan di bawah ini saya :

Nama : Dinang Shendri
Nrp. : 2212100097
Jurusan / Fak. : Teknik Elektro / F.T. Industri
Alamat kontak : Rt 19 Rw 07 Ds Pardon Kec/kab. Trenggalek
a. Email : dinang_s61@gmail.com
b. Telp/HP : 085259053502

Menyatakan bahwa semua data yang saya *upload* di Digital Library ITS merupakan hasil final (revisi terakhir) dari karya ilmiah saya yang sudah disahkan oleh dosen penguji. Apabila dikemudian hari ditemukan ada ketidaksesuaian dengan kenyataan, maka saya bersedia menerima sanksi.

Demi perkembangan ilmu pengetahuan, saya menyetujui untuk memberikan **Hak Bebas Royalti Non-Eksklusif (Non-Exclusive Royalti-Free Right)** kepada Institut Teknologi Sepuluh Nopember Surabaya atas karya ilmiah saya yang berjudul :

KONTROL TRACKING FUZZY BERBASIS PERFORMA ROBUST
UNTUK QUADROTOR

Dengan Hak Bebas Royalti Non-Eksklusif ini, Institut Teknologi Sepuluh Nopember Surabaya berhak menyimpan, mengalih-media/format-kan, mengelolanya dalam bentuk pangkalan data (*database*), mendistribusikannya, dan menampilkan/mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta. Saya bersedia menanggung secara pribadi, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya Ilmiah saya ini tanpa melibatkan pihak Institut Teknologi Sepuluh Nopember Surabaya.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Surabaya
Pada tanggal : 20-06-2016
Yang menyatakan,

Dosen Pembimbing 1

Anhashtu
Dr. Tri Hastuti Agustindh, S.T.M.T.
NIP. 196808121994032001

Dinang Shendri
Nrp. 2212100097

KETERANGAN :

Tanda tangan pembimbing wajib dibubuhi stempel jurusan.

Form dicetak dan diserahkan di bagian Pengadaan saat mengumpulkan hard copy TA/Tesis/Disertasi.

KONTROL *TRACKING FUZZY* BERBASIS PERFORMA *ROBUST* UNTUK *QUADROTOR*

Nama : Dinang Sohendri
NRP : 2212100097
Pembimbing : Dr. Trihastuti Agustinah, ST., MT.
NIP : 1968 08 12 1994 03 2001

ABSTRAK

Quadrotor merupakan salah satu jenis *Unmanned Aerial Vehicle* (UAV) yang memiliki 4 buah baling-baling atau *propeller*. Desain kontrol *tracking fuzzy* Takagi-Sugeno digunakan untuk mengatur *tracking Quadrotor* mengikuti sinyal referensi dan kontrol *state-feedback* untuk mengatur kestabilan sudut *Quadrotor*. Metode kontrol *fuzzy* Takagi-Sugeno merepresentasikan nonlinearitas dari *Quadrotor* dalam beberapa model linear. Model linear ini diperoleh dari linearisasi di beberapa titik kerja *Quadrotor*. Berdasarkan model tersebut, aturan kontrol *fuzzy* T-S disusun dengan konsep *Parallel Distributed Compensation* (PDC). Performa *tracking H_∞* dirancang untuk mencari *gain* kontroler yang paling sesuai untuk mengatasi gangguan pada sistem. Selanjutnya, persoalan diselesaikan dengan pendekatan *Linear Matrix Inequality* (LMI) sehingga diperoleh *gain* kontrol berbasis performa H_∞ . Hasil simulasi menunjukkan bahwa sistem kontrol hasil desain dapat mengatur gerak *Quadrotor* sesuai lintasan yang diinginkan dengan *Integral Absolut Error* 0,1149 pada sumbu X dan 0,0617 pada sumbu Y . Selain itu, ∞ -norm dari performa keluaran memiliki tingkat pelemahan kurang dari γ ketika gangguan diberikan.

Kata Kunci : *Fuzzy* Takagi-Sugeno, *Linear Matrix Inequality*, Performa *Tracking H_∞* , *Quadrotor*

FUZZY TRACKING CONTROL BASED ON ROBUST PERFORMANCE OF A QUADROTOR

Name : Dinang Sohendri
Registration Number : 2212100097
Supervisor : Dr. Trihastuti Agustinah, ST., MT.
ID Number : 1968 08 12 1994 03 2001

ABSTRACT

Quadrotor is a type of Unmanned Aerial Vehicle (UAV) which has four pieces of propeller. The state-feedback control is used to regulate the angular stability of Quadrotor. Fuzzy tracking control based on Takagi-Sugeno model is designed such that the Quadrotor can follow the reference signal. Takagi-Sugeno (T-S) fuzzy model is used to represent nonlinearity of the Quadrotor into several linear models. These linear models are obtained by linearization at the Quadrotor operating points. Based on the models, fuzzy T-S control rules are arranged with the concept of Parallel Distributed Compensation (PDC). H_∞ tracking performance is designed to find the most appropriate controller to refuse the disturbance in the system. Furthermore, the problem is resolved by Linear Matrix Inequality (LMI) approach in order to obtain the gain control based on H_∞ performance. The simulation results show that the Quadrotor can follow the desired trajectory with Integral Absolute Error 0.1149 at the X axis and 0.0617 at the Y axis. In addition, ∞ -norm the performance output has a level of attenuation less than γ in the present of disturbance.

Keywords: H_∞ Tracking Performance, Linear Matrix Inequality, Takagi-Sugeno Fuzzy, Quadrotor

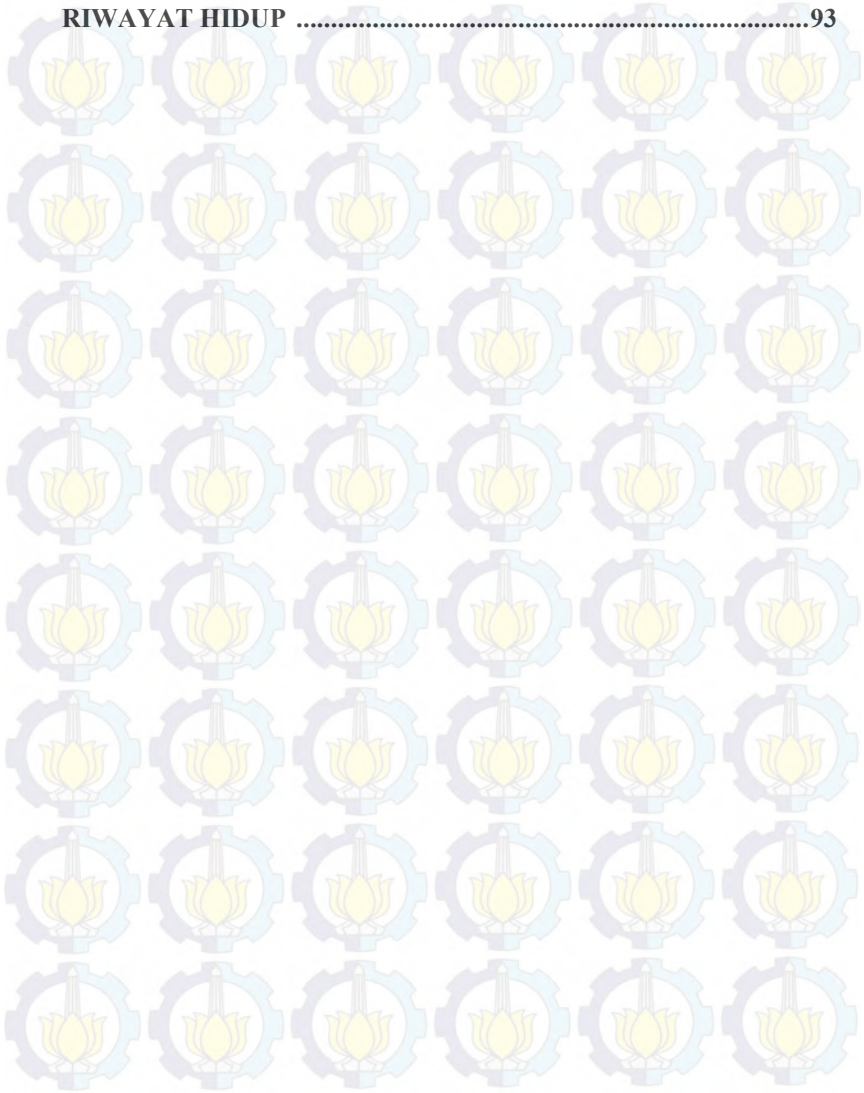
DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN TUGAS AKHIR	iii
HALAMAN PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Laporan	3
1.7 Relevansi	4
BAB II TEORI DASAR	5
2.1 Tinjauan Pustaka	5
2.2 <i>Quadrotor</i>	6
2.3 Pemodelan <i>Quadrotor</i>	8
2.4 Linearisasi	9
2.5 Kontroler <i>State Feedback</i>	13
2.6 Teori <i>Fuzzy</i>	14
2.6.1 Fungsi Keanggotaan	14
2.6.2 Mekanisme Inferensi <i>Fuzzy</i>	16
2.7 Model <i>Fuzzy</i> Takagi-Sugeno	17
2.8 Model Referensi	20
2.9 Analisis Kestabilan Lyapunov	21
2.10 Performansi H_∞	23
2.11 <i>Linear Matrix Inequality</i>	25
BAB III PERANCANGAN SISTEM	27
3.1 Model Dinamika <i>Quadrotor</i>	27
3.1.1 Persamaan <i>State</i> Dinamika <i>Quadrotor</i>	27

3.1.2	Parameter <i>Quadrotor</i>	29
3.2	Perancangan Kontroler.....	29
3.2.1	Stabilisasi <i>Quadrotor</i>	30
3.2.2	<i>Altitude Z</i>	32
3.2.3	<i>Tracking Quadrotor</i>	33
3.2.3.1	<i>Linearisasi Model Matematika Quadrotor</i> ...	33
3.2.3.2	<i>Pemodelan Fuzzy Takagi-Sugeno</i>	35
3.2.3.3	<i>Model Referensi</i>	38
3.2.3.4	<i>Augemented Fuzzy System (AFS)</i>	38
3.2.3.5	<i>Perumusan Fungsi Lyapunov</i>	39
3.2.3.6	<i>Performa Tracking H_{∞}</i>	40
3.2.4	Penggabungan Sinyal Kontrol	48
3.3	Perancangan Perangkat Simulasi.....	49
BAB IV HASIL SIMULASI.....		50
4.1	Simulasi Stabilisasi <i>Quadrotor</i>	50
4.2	Simulasi <i>Tracking Quadrotor</i>	53
4.2.1	Simulasi <i>Altitude Quadrotor</i>	30
4.2.2	Simulasi Gerak Sumbu X,Y <i>Quadrotor</i>	54
4.2.3	Simulasi Gerak <i>Quadrotor</i> dengan Gangguan	56
BAB V PENUTUP		67
5.1	Kesimpulan.....	67
5.2	Saran	67
DAFTAR PUSTAKA		69
LAMPIRAN A		71
A.1	Pemodelan <i>Quadrotor</i>	71
A.1.1	Koordinat <i>Frame</i>	71
A.1.2	Asumsi Pemodelan <i>Quadrotor</i>	72
A.1.3	Kinematika	72
A.1.4	Dinamika	77
A.1.4.1	<i>Penurunan Gerak Translasi</i>	78
A.1.4.2	<i>Penurunan Gerak Rotasi</i>	79
LAMPIRAN B		83
B.1	Program Kontrol Stabilisasi <i>Quadrotor</i>	83
B.2	Program Kontrol <i>Tracking Quadrotor</i>	85

B.3 Blok Diagram *Simulink* 90

RIWAYAT HIDUP **93**



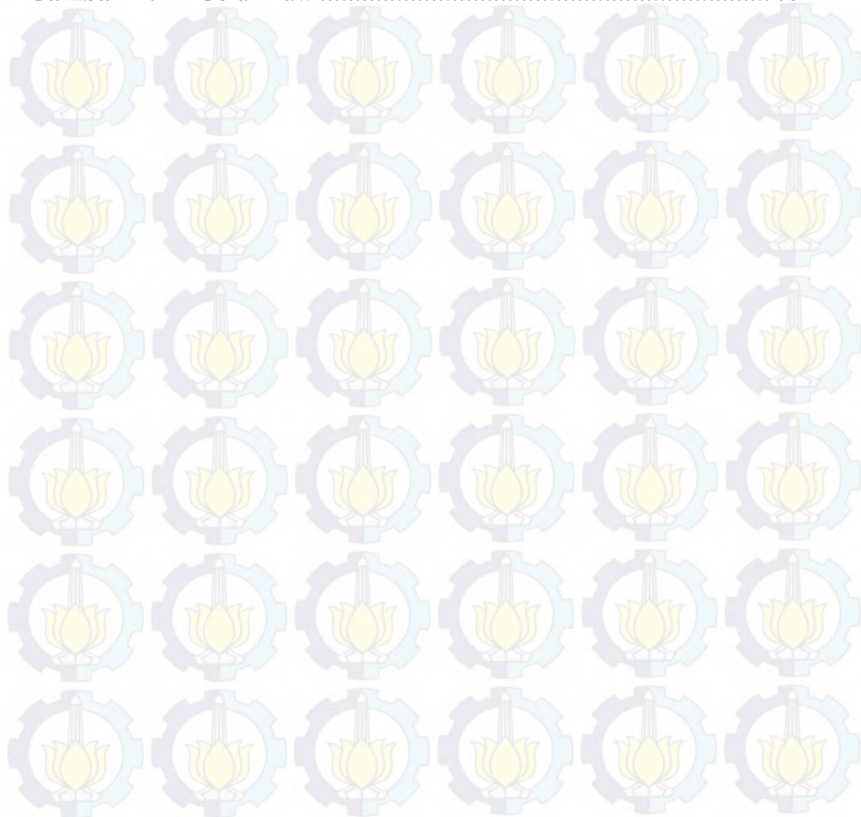
DAFTAR TABEL

Tabel 3.1	Nilai Parameter <i>Quadrotor</i>	29
Tabel 3.2	Daerah Linearisasi ϕ dan θ	34
Tabel 4.1	Nilai <i>Gain State-Feedback</i> dan ∞ -norm Sistem dengan γ yang Berbeda-beda	58
Tabel 4.2	Kesalahan Posisi <i>Quadrotor</i> pada Sumbu <i>X</i> dengan Berbagai Nilai γ	60
Tabel 4.3	Nilai <i>Gain State-Feedback</i> dan ∞ -norm Sistem dengan γ yang Berbeda-beda	60
Tabel 4.4	Kesalahan Posisi <i>Quadrotor</i> pada Sumbu <i>Y</i> dengan Berbagai Nilai γ	62
Tabel 4.5	Hasil Simulasi Gerak <i>Quadrotor</i> dengan Variasi <i>Noise</i>	65

DAFTAR GAMBAR

Gambar 2.1	<i>Quadrotor</i>	7
Gambar 2.2	<i>Thrust</i>	7
Gambar 2.3	<i>Roll</i>	7
Gambar 2.4	<i>Pitch</i>	7
Gambar 2.5	<i>Yaw</i>	7
Gambar 2.6	Struktur Kontrol <i>Fuzzy</i>	14
Gambar 2.7	Skema <i>Parallel Distributed Compensation</i> (PDC)	19
Gambar 2.8	Model Referensi	20
Gambar 2.9	Konfigurasi Sistem Kontrol <i>Robust</i>	24
Gambar 3.1	Diagram Blok Sistem Kontrol <i>Servo</i> Tipe 1	32
Gambar 3.2	Fungsi Keanggotaan untuk Aturan <i>Plant</i> dan Kontroler	37
Gambar 3.3	Plot <i>Magnitude Bode</i> $T_{z,w}^{(x1)}$	47
Gambar 3.4	Plot <i>Magnitude Bode</i> $T_{z,w}^{(x2)}$	47
Gambar 3.5	Plot <i>Magnitude Bode</i> $T_{z,w}^{(y1)}$	47
Gambar 3.6	Plot <i>Magnitude Bode</i> $T_{z,w}^{(y2)}$	47
Gambar 3.7	Arsitektur Sistem Kontrol untuk <i>Quadrotor</i>	49
Gambar 3.8	Diagram <i>Simulink</i> Sistem Kontrol <i>Tracking</i> <i>Quadrotor</i>	50
Gambar 4.1	Respon Sudut ϕ	52
Gambar 4.2	Respon Sudut θ	52
Gambar 4.3	Respon Sudut ψ	52
Gambar 4.4	Posisi <i>Z Quadrotor</i>	53
Gambar 4.5	Referensi <i>Tracking</i> untuk <i>Quadrotor</i>	54
Gambar 4.6	<i>Tracking</i> Posisi <i>Quadrotor</i>	55
Gambar 4.7	Sudut <i>Roll</i> dan <i>Pitch</i> untuk Gerak <i>Quadrotor</i>	55
Gambar 4.8	Plot 3D hasil simulasi	55
Gambar 4.9	Perbandingan Referensi <i>Tracking</i> dengan Model Referensi untuk Gerak <i>Quadrotor</i>	56
Gambar 4.10	Kesalahan <i>Tracking</i> Gerak <i>Quadrotor</i>	57
Gambar 4.11	Gangguan pada <i>Quadrotor</i>	57
Gambar 4.12	Perbandingan Posisi <i>Quadrotor</i> dengan Variasi γ	58
Gambar 4.13	Kesalahan Posisi <i>X</i>	59
Gambar 4.14	Sinyal Kontrol <i>U3</i> untuk Gerak Sumbu <i>X</i>	59
Gambar 4.15	Perbandingan Posisi <i>Quadrotor</i> dengan Variasi γ	60
Gambar 4.16	Kesalahan Posisi <i>Y</i>	61
Gambar 4.17	Sinyal Kontrol <i>U2</i> untuk Gerak Sumbu <i>Y</i>	61

Gambar 4.18	<i>Noise</i> pada Sensor Kecepatan X dan Kecepatan Y	63
Gambar 4.19	Kesalahan Posisi pada <i>Quadrotor</i> dengan Pengaruh <i>Noise</i> pada Sumbu X	63
Gambar 4.20	Kesalahan Posisi pada <i>Quadrotor</i> dengan Pengaruh <i>Noise</i> pada Sumbu Y	64
Gambar 4.21	Sinyal Kontrol U_2 untuk Gerak <i>Quadrotor</i>	64
Gambar 4.22	Sinyal Kontrol U_3 untuk Gerak <i>Quadrotor</i>	64
Gambar A.1	<i>Frame</i> Bumi (<i>E-frame</i>) dan <i>Frame</i> Badan (<i>B-frame</i>)..	71
Gambar A.2	Gerak <i>Roll</i>	73
Gambar A.3	Gerak <i>Pitch</i>	74
Gambar A.4	Gerak <i>Yaw</i>	75



BAB I

PENDAHULUAN

Pada Bab ini dijelaskan latar belakang mengenai Tugas Akhir ini. Kemudian disusun perumusan masalah dan tujuan penelitian. Selanjutnya dibahas juga metodologi dan sistematika yang digunakan pada penelitian serta relevansinya.

1.1 Latar belakang

Teknologi penerbangan sedang gencar-gencarnya mengembangkan teknologi pesawat tanpa awak *Quadrotor* atau *Unmanned Aerial Vehicle* (UAV). *Quadrotor* ini memiliki 4 buah baling-baling atau *propeller* dengan konfigurasi menyilang, *propeller* depan dan belakang berputar searah jarum jam (*clockwise*) dan *propeller* kiri dan kanan berputar berlawanan arah jarum jam (*counter clockwise*). *Quadrotor* memiliki dua permasalahan yang harus dijaga agar *Quadrotor* mampu terbang dengan stabil. Masalah pertama yaitu kestabilan sudut *roll*, *pitch* dan *yaw* pada saat terbang dan permasalahan kedua adalah *tracking Quadrotor* mengikuti sinyal referensi.

Beberapa penelitian terkait *Quadrotor* seperti pada [1], menjaga kestabilan sudut *roll*, *pitch*, dan *yaw* menggunakan kontroler *fuzzy* T-S sehingga mampu dipertahankan disekitar nol. Untuk mengatur gerak *Quadrotor*, [2] menggunakan *Continuous Sliding Mode Control* untuk bergerak pada lintasan lurus dengan hasil yang gerak yang baik, namun dengan metode tersebut muncul efek *chattering*. Metode kontrol Adaptif juga diterapkan untuk mengatur gerak *Quadrotor*, namun masih rentan terhadap gangguan seperti pada [3]. Oleh karena itu harus dipilih metode kontrol yang tepat untuk mengatur gerak *Quadrotor* seperti pada [4]. Metode kontrol *fuzzy* T-S digunakan untuk mengatur gerak *Quadrotor* dengan hasil yang baik dan mampu tahan terhadap gangguan.

Pada Tugas Akhir ini kontroler *tracking* akan didesain dengan metode *fuzzy* Takagi-Sugeno berbasis performa *tracking H_∞* . Digunakan teknik pemrograman *Linear Matrix Inequality* (LMI) untuk mencari solusi dari metode kontrol tersebut. Performa *tracking H_∞* didesain untuk mempertahankan *Quadrotor* agar mampu mengatasi gangguan dari luar.

1.2 Perumusan Masalah

Dalam suatu misi penerbangan, *Quadrotor* akan diatur bergerak menuju titik referensi tertentu. Untuk melakukan misi tersebut diperlukan suatu pengontrolan agar *Quadrotor* dapat terbang sesuai dengan sinyal referensi. Gerak manuver (*tracking*) merupakan gerak dimana *Quadrotor* dapat terbang pada sumbu X dan Y . Gerakan ini menuntut adanya kestabilan sudut *pitch* dan *roll Quadrotor* agar sesuai dengan lintasan gerak yang diinginkan sehingga gerak *tracking Quadrotor* bisa stabil. Oleh karena itu muncul permasalahan sebagai berikut:

1. Bagaimana merancang sistem kontroler yang mampu menjaga kestabilan sudut *Quadrotor*.
2. Bagaimana merancang sistem kontrol yang mampu mengatur gerak *Quadrotor* mengikuti sinyal referensi.

1.3 Batasan Masalah

Batasan masalah dalam perancangan serta pengujian Tugas Akhir ini adalah sebagai berikut:

1. Model dan parameter *Quadrotor* yang digunakan adalah model *Quadrotor* Quanser Q-Ball X-4.
2. Konfigurasi *propeller* dari *Quadrotor* berbentuk “plus”.
3. Simulasi penerbangan *Quadrotor* diasumsikan terbang didalam ruangan.
4. Titik origin *frame* inersia bumi dan *frame* badan *Quadrotor* berhimpit.
5. *Tracking* posisi dilakukan setelah *Quadrotor* terbang
6. Dalam Tugas Akhir ini tidak diatur orientasi *Quadrotor*, dimana orientasi *yaw Quadrotor* ditentukan pada arah utara

1.4 Tujuan Penelitian

Tujuan dari penyusunan Tugas Akhir ini adalah untuk memperoleh sistem kontrol *tracking fuzzy* berbasis performa *robust* untuk mengatur gerak *Quadrotor* sesuai lintasan yang diinginkan dengan meminimumkan *error tracking* serta tahan terhadap gangguan dari luar.

1.5 Metodologi Penelitian

Pada penelitian ini dilakukan beberapa proses secara bertahap dan berurutan agar tercapai tujuan akhir dari penelitian

1. Studi Literatur

Pada tahap ini akan dilakukan kegiatan pengumpulan dan pengkajian hal-hal terkait teori, informasi maupun hasil eksperimen serupa yang dapat dijadikan referensi dalam proses penyusunan tugas akhir ini..

2. Pemodelan dan Identifikasi Sistem

Model matematika dari *Quadrotor* diperoleh dari pemodelan fisis analisis kinematika dan dinamika sistem.

3. Desain Kontroler

Dari pemodelan sistem yang telah diperoleh, maka akan dilakukan perancangan kontroler *fuzzy* Takagi-Sugeno dengan pendekatan LMI untuk mengoptimalkan gerak *tracking Quadrotor*.

4. Pengujian dan Analisa

Setelah kontroler selesai dibuat selanjutnya yaitu dilakukan pengujian pada sistem dan dianalisa hasil penelitian. Simulasi dilakukan dengan menggunakan *software* Matlab.

5. Kesimpulan

Penarikan kesimpulan mengacu pada data hasil pengujian, analisis data, dan referensi terkait. Kesimpulan menunjukkan hasil penelitian yang telah dilakukan sesuai dengan rumusan masalah yang telah dibuat.

6. Penyusunan Buku Tugas Akhir

Pada tahap ini akan dilakukan penyusunan laporan terkait hasil proses Tugas Akhir yang telah dilakukan. Penyusunan buku tugas akhir dilakukan sebagai bentuk laporan tertulis.

1.6 Sistematika Laporan

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

BAB 1 : Pendahuluan

Bab ini meliputi latar belakang, permasalahan, tujuan, metodologi, sistematika dan relevansi.

BAB 2 : Teori Dasar

Menjelaskan tentang tinjauan pustaka, konsep dasar dari *Quadrotor*, teknik linearisasi, kontroler *state-feedback*, model *fuzzy* Takagi-Sugeno (T-S), model referensi, analisis kestabilan Lyapunov, performa H_∞ dan *Linear Matrix Inequality* (LMI).

BAB 3 : Perancangan Sistem

Bab ini membahas perancangan kontroler untuk stabilisasi sudut *roll*, *pitch* dan *yaw* dan perancangan kontroler *fuzzy* untuk *tracking Quadrotor*.

BAB 4 : Hasil Simulasi

Bab ini berisi hasil simulasi serta analisa hasil simulasi dari penelitian.

BAB 5 : Penutup

Berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.7 Relevansi

Hasil yang diperoleh pada Tugas Akhir ini diharapkan menjadi referensi pengembangan desain kontrol *tracking* pada *Quadrotor*. Selain itu, hasil dari Tugas Akhir ini diharapkan memberikan manfaat dalam ilmu sistem kontrol khususnya pada sistem *Quadrotor*.

BAB II TEORI DASAR

Pada bab ini terdapat teori yang menunjang dalam merumuskan dan menyelesaikan masalah yang dihadapi dalam pengerjaan Tugas Akhir. Bagian awal terdapat teori tentang *Quadrotor*, baik secara umum maupun konsep gerak dari *Quadrotor*. Bagian selanjutnya membahas tentang konsep linearisasi, kontroler *state-feedback*, model *fuzzy* Takagi-Sugeno (T-S), Model Referensi, analisis kestabilan Lyapunov, performa H_∞ dan *Linear Matrix Inequality* (LMI).

2.1 Tinjauan Pustaka

Pada *Quadrotor* terdapat dua permasalahan yang menjadi pokok bahasan pada sistem kontrol. Permasalahan tersebut yaitu kestabilan sudut gerak *Quadrotor* dan permasalahan *Quadrotor* bergerak sesuai referensi yang diinginkan (*tracking*). *Quadrotor* menggunakan 4 buah baling-baling untuk bisa terbang sehingga memiliki sifat nonlinearitas yang tinggi dan rentan terhadap gangguan.

Pada [1] kestabilan sudut *roll*, *pitch*, dan *yaw* dicapai dengan menggunakan kontroler *fuzzy* T-S sehingga mampu dipertahankan disekitar nol. Kontrol *tracking* diterapkan pada *Quadrotor* menggunakan *Continuous Sliding Mode Control* untuk *tracking* lintasan lurus [2]. Hasil yang dicapai *Quadrotor* mampu terbang stabil mengikuti sinyal referensi, namun dengan metode kontrol tersebut efek *chattering* yang muncul masih banyak. Metode kontrol *tracking* lain yaitu PID-LQR juga telah diterapkan sebagai metode kontrol untuk memaksa *Quadrotor* terbang mengikuti referensi. Metode ini memiliki keterlambatan respon sehingga *Quadrotor* memiliki *delay* waktu pada saat terbang. Metode kontrol yang lebih modern, Adaptif Kontrol, juga diterapkan pada *Quadrotor* seperti pada [3]. Metode ini memiliki hasil respon yang cukup bagus namun masih rentan terhadap gangguan, sehingga pada pengontrolan gerak *Quadrotor* harus dipilih metode yang tepat dan harus tahan terhadap gangguan seperti pada [4]. Metode *fuzzy* T-S dirancang untuk mengatur gerak *Quadrotor* mengikuti lintasan yang diinginkan. Performa *robust* juga diterapkan sehingga metode kontrol mampu mengatas gangguan dari luar. Metode *fuzzy* dirasa cukup mudah dan tidak memerlukan waktu yang lama dalam proses komputasi.

Pada Tugas Akhir ini kontroler *tracking* akan didesain dengan metode *fuzzy* Takagi-Sugeno berbasis performa *tracking* H_∞ . Model Referensi digunakan untuk memaksa *Quadrotor* mampu terbang mengikuti sinyal referensi. Selanjutnya digunakan teknik pemrograman *Linear Matrix Inequality* (LMI) untuk mencari solusi dari metode kontrol tersebut. Performa *tracking* H_∞ didesain untuk mempertahankan *Quadrotor* agar tetap terbang sesuai referensi yang diinginkan walaupun ada gangguan dari luar.

2.2 *Quadrotor* [5]

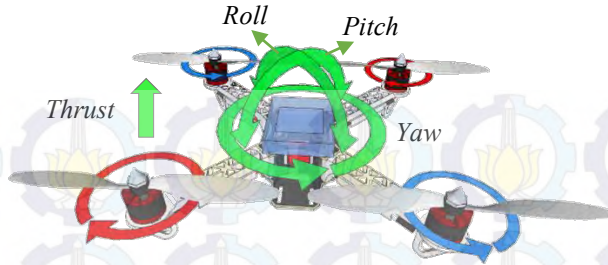
Quadrotor merupakan suatu robot terbang kecil memiliki model mekanik yang terdiri dari empat rotor yang dipasang pada sumbu silang simetris. Bentuk ini diharapkan tipis dan kaku, sehingga diperoleh friksi udara yang kecil dan komponen yang bergerak pada *Quadrotor* hanyalah putaran *propeller*. Setiap *propeller* pada *Quadrotor* ini diputar oleh satu motor elektrik, sehingga terdapat empat motor sebagai aktuatur untuk menghasilkan gaya angkat pada *Quadrotor*.

Dengan batasan menggunakan karakteristik motor dan *propeller* yang relatif sama, maka kondisi melayang yang stabil akan diperoleh kecepatan motor yang sama di tiap *propeller*. Namun pada tiap *propeller* yang berputar akan menghasilkan torsi yang bergerak bidang horizontal, maka konfigurasi *propeller* terdiri dari dua yang bergerak searah jarum jam dan dua yang berlawanan arah jarum jam dan dalam tiap satu sumbu, *propeller* berputar dengan arah yang sama. Pada Gambar 2.1, ditunjukkan bahwa terdapat empat gerakan yang terdapat pada struktur *Quadrotor*, yaitu *roll*, *pitch*, *yaw* dan *thrust*.

Dengan melakukan pengaturan kecepatan putaran *propeller* maka dihasilkan beberapa komando input diantaranya:

a. *Thrust* (U_1)

Dengan mempercepat (warna merah) atau memperlambat (warna hijau) kecepatan motor dari seluruh *propeller* secara bersamaan akan menghasilkan percepatan vertikal yang ditunjukkan pada Gambar 2.2



Gambar 2.1 *Quadrotor*



Gambar 2.2 *Thrust*

b. *Roll (U_2)*

Dengan mempercepat atau memperlambat *propeller* pada sisi kiri dan secara bersamaan memperlambat atau mempercepat *propeller* di sisi kanan akan diperoleh rotasi dengan kecepatan sudut yang tersaji pada Gambar 2.3.



Gambar 2.3 *Roll*

c. *Pitch (U_3)*

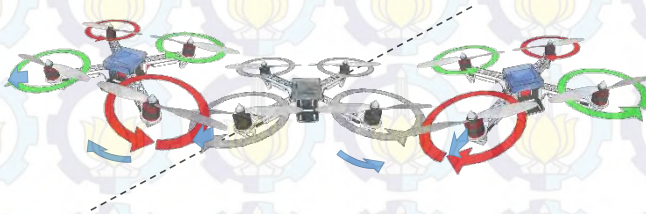
Pada Gambar 2.4 mekanisme yang sama seperti gerak *roll*, namun dilakukan pada *propeller* di sisi depan dan belakang akan diperoleh percepatan sudut *pitch*.



Gambar 2.4 *Pitch*

d. *Yaw* (U_4)

Dengan mempercepat atau memperlambat *propeller* yang berputar berlawanan arah jarum jam, dan memperlambat atau mempercepat *propeller* yang berputar searah jarum jam akan diperoleh gerak *yaw* dengan percepatan sudut yang ditampilkan pada Gambar 2.5.



Gambar 2.5 *Yaw*

2.3 **Pemodelan *Quadrotor* [5]**

Quadrotor merupakan sistem yang memiliki 6 derajat kebebasan (*degree of freedom*) dengan 12 keluaran, dimana keluaran dari sistem tersebut menentukan *attitude* (ϕ, θ, ψ), *altitude* (Z) dan gerak sumbu X, Y *Quadrotor*. Dengan analisis kinematika dan dinamika didapatkan persamaan model matematika dari *Quadrotor* seperti pada (2.1).

$$\left\{ \begin{array}{l} \ddot{X} = (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Y} = (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Z} = -g + (\cos \theta \cos \phi) \frac{U_1}{m} \\ \dot{p} = \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{U_2 l}{J_{xx}} \\ \dot{q} = \frac{J_{zz} - J_{xx}}{J_{yy}} pr + \frac{U_3 l}{J_{yy}} \\ \dot{r} = \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4 d}{J_{zz}} \end{array} \right. \quad (2.1)$$

dengan X , Y dan Z merupakan posisi *Quadrotor*. Massa *Quadrotor* dimisalkan m , gaya gravitasi bumi dinotasikan g dan J_{xx} , J_{yy} , J_{zz} merupakan momen inersia sepanjang sumbu x , y dan z . (ϕ, θ, ψ) merupakan sudut yang dibentuk akibat gerak *roll*, *pitch* dan *yaw*, sedangkan p , q dan r merupakan kecepatan sudut. Konstanta drag dan jarak pusat massa terhadap motor dimisalkan d dan l .

Gaya yang bekerja pada *Quadrotor* ini merupakan kombinasi gaya angkat (F) tiap motor sesuai dengan gerakan yang terjadi. U_1 , U_2 , U_3 dan U_4 akan menghasilkan gerak *thrust*, *roll*, *pitch* dan *yaw* sesuai dengan kombinasi gaya F dari 4 buah motor seperti pada (2.2).

$$\left\{ \begin{array}{l} U_1 = F_1 + F_2 + F_3 + F_4 \\ U_2 = F_2 - F_4 \\ U_3 = F_1 - F_3 \\ U_4 = F_1 - F_2 + F_3 - F_4 \end{array} \right. \quad (2.2)$$

2.4 Linearisasi [6]

Sistem nonlinear memiliki titik ekuilibrium lebih dari satu, tidak seperti sistem yang linear. Titik ekuilibrium sendiri merupakan titik dimana perubahan *state* dari sistem bernilai nol. Sifat nonlinear dibagi menjadi dua, yaitu *inherent (natural)* dan *intentional (artificial)*. Sifat nonlinear *inherent* adalah sifat nonlinear yang muncul akibat aktivitas

gerak dari sistem, seperti gaya gesek, histerisis, dan saturasi. Sedangkan sifat nonlinear *intentional* merupakan sifat nonlinear yang sengaja dimunculkan. Pada umumnya untuk menyelesaikan permasalahan sebuah sistem yang mengandung sifat nonlinear secara matematis sangat sulit dan memerlukan waktu yang cukup lama. Oleh karena itu linearisasi merupakan metode yang cukup efektif untuk menyelesaikan permasalahan nonlinearitas dengan melakukan pendekatan kedalam beberapa daerah kerja yang terbatas, sehingga analisa kestabilan pada sistem linear bisa diterapkan dan memudahkan dalam proses desain.

Sistem dapat direpresentasikan kedalam persamaan diferensial seperti pada (2.3).

$$\left. \begin{aligned} \dot{x}_1 &= f_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ \dot{x}_2 &= f_2(x_1, \dots, x_n, u_1, \dots, u_m) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, \dots, x_n, u_1, \dots, u_m) \end{aligned} \right\} \quad (2.3)$$

$$\left. \begin{aligned} y_1 &= h_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ y_2 &= h_2(x_1, \dots, x_n, u_1, \dots, u_m) \\ &\vdots \\ y_p &= h_p(x_1, \dots, x_n, u_1, \dots, u_m) \end{aligned} \right\} \quad (2.4)$$

Jika vektor kolom $x = [x_1, \dots, x_n]^T$ merupakan vektor state sistem, vektor kolom $u = [u_1, \dots, u_m]^T$ merupakan sinyal kontrol dan vektor kolom $y = [y_1, \dots, y_p]^T$ merupakan keluaran sistem, maka (2.3) dan (2.4) dapat disederhakan menjadi (2.5) dan (2.6).

$$f(x, u) = \begin{bmatrix} f_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ f_2(x_1, \dots, x_n, u_1, \dots, u_m) \\ \vdots \\ f_n(x_1, \dots, x_n, u_1, \dots, u_m) \end{bmatrix} \quad (2.5)$$

$$h(x, u) = \begin{bmatrix} h_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ h_2(x_1, \dots, x_n, u_1, \dots, u_m) \\ \vdots \\ h_p(x_1, \dots, x_n, u_1, \dots, u_m) \end{bmatrix} \quad (2.6)$$

sehingga dapat ditulis:

$$\begin{aligned} \dot{x} &= f(x, u) \\ y &= h(x, u) \end{aligned} \quad (2.7)$$

Jika (2.7) merupakan sistem linear, maka sistem tersebut dapat ditulis dalam representasi *state space*:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (2.8)$$

dengan $A \in \mathcal{R}^{n \times n}$, $B \in \mathcal{R}^{n \times m}$, $C \in \mathcal{R}^{p \times n}$, $D \in \mathcal{R}^{p \times m}$.

Suatu sistem nonlinear dinyatakan dalam bentuk sebagai berikut

$$\dot{x} = f(x) \quad (2.9)$$

dimana $y = f(x)$ merupakan suatu kurva nonlinear. Jika $f(x)$ direpresentasikan dalam pendekatan linear di sekitar titik x^* maka dapat ditulis

$$\delta y = f \delta x = \left. \frac{df(x)}{dx} \right|_{x=x^*} \delta x \quad (2.10)$$

dengan $\delta y = f(x) - f(x^*)$ dan $\delta x = x - x^*$. Apabila sistem bekerja pada titik origin (0,0), maka;

$$\begin{aligned} \delta y &= f(x) - f(x^*) \\ \delta y &= f(x) - 0 \\ \delta y &= f(x) \end{aligned}$$

$$\delta x = x - x^*$$

$$\delta x = x - 0$$

$$\delta x = x$$

$$y = \left. \frac{df(x)}{dx} \right|_{x=x^*} x \quad (2.11)$$

Jika f dan h pada (2.7) merupakan sistem nonlinear dari x dan u , kemudian dilakukan pendekatan linear disekitar titik $x^* = [x_1^*, \dots, x_n^*]^T$, dengan titik tersebut diperoleh pada saat $u^* = [u_1^*, \dots, u_m^*]^T$, maka hasil linearisasi lokal (2.7) berdasarkan (2.11) dapat dituliskan dengan (2.12-2.15) sebagai berikut:

$$A = \left. \frac{df(x,u)}{dx} \right|_{x^*, u^*} = \begin{bmatrix} \frac{df_1(x,u)}{dx_1} & \dots & \frac{df_1(x,u)}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n(x,u)}{dx_1} & \dots & \frac{df_n(x,u)}{dx_n} \end{bmatrix}_{x^*, u^*} \quad (2.12)$$

$$B = \left. \frac{df(x,u)}{du} \right|_{x^*, u^*} = \begin{bmatrix} \frac{df_1(x,u)}{du_1} & \dots & \frac{df_1(x,u)}{du_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n(x,u)}{du_1} & \dots & \frac{df_n(x,u)}{du_n} \end{bmatrix}_{x^*, u^*} \quad (2.13)$$

$$C = \left. \frac{dh(x,u)}{dx} \right|_{x^*, u^*} = \begin{bmatrix} \frac{dh_1(x,u)}{dx_1} & \dots & \frac{dh_1(x,u)}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{dh_n(x,u)}{dx_1} & \dots & \frac{dh_n(x,u)}{dx_n} \end{bmatrix}_{x^*, u^*} \quad (2.14)$$

$$D = \frac{dh(x,u)}{du} \Big|_{x^*,u^*} = \begin{bmatrix} \frac{dh_1(x,u)}{du_1} & \dots & \frac{dh_1(x,u)}{du_n} \\ \vdots & \ddots & \vdots \\ \frac{dh_n(x,u)}{du_1} & \dots & \frac{dh_n(x,u)}{du_n} \end{bmatrix} \Big|_{x^*,u^*} \quad (2.15)$$

2.5 Kontroler *State-Feedback* [6]

Sebuah *plant* diharapkan mampu kembali ke posisi ekuilibriumnya untuk menjaga kestabilan sistem. Permasalahan tersebut merupakan masalah *regulator* dimana kontroler yang disusun harus mampu mengembalikan *state plant* ke titik ekuilibriumnya atau 0. Salah satu metode kontrol untuk mengatasi masalah *regulator* adalah kontrol *state-feedback*. Jika sebuah sistem didefinisikan seperti berikut:

$$\dot{x} = Ax + Bu \quad (2.16)$$

dengan

- x = vektor *state plant*
- u = sinyal kontrol
- A = matriks sistem $n \times n$
- B = matriks sistem $n \times 1$

maka sinyal kontrol dapat didefinisikan sebagai (2.17).

$$u = -Kx \quad (2.17)$$

Dengan memasukkan (2.17) ke (2.16) maka didapatkan persamaan seluruh sistem seperti (2.18).

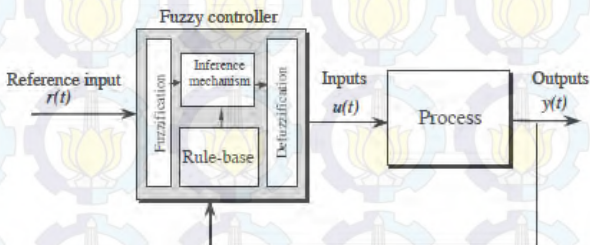
$$\dot{x} = (A - BK)x \quad (2.18)$$

dari (2.18) dapat dilihat bahwa sinyal kontrol ditentukan oleh *state plant* saat itu juga. Struktur kontrol tersebut disebut dengan *state-feedback*. Karakteristik dari sistem (2.16) akan ditentukan oleh *eigenvalue* dari (2.18).

2.6 Teori Fuzzy [7]

Fuzzy pertama kali dikenalkan oleh Lotfi Zadeh pada tahun 1965. Kontrol *fuzzy* merupakan metode untuk merepresentasikan, memanipulasi, dan mengimplementasi konsep pemikiran manusia kedalam sistem kontrol. Logika *fuzzy* adalah logika kekaburan (*fuzziness*) antara benar dan salah. Dalam logika *fuzzy* sebuah nilai dapat bernilai benar dan salah secara bersamaan dan nilai benar atau salah tersebut bergantung pada derajat keanggotaan untuk mengetahui besar nilainya.

Kontrol *fuzzy* memiliki blok diagram seperti pada Gambar 2.6, dengan kontrol *fuzzy* terletak didalam lup tertutup sistem kontrol. Keluaran dari *plant* dinotasikan sebagai $y(t)$, masukan *plant $u(t)$, dan masukan referensi pada kontrol *fuzzy* adalah $r(t)$.*



Gambar 2.6 Struktur Kontrol Fuzzy [7]

Kontrol *fuzzy* memiliki empat komponen utama: (1) “*rule-base*” komponen ini akan menentukan aturan keluaran dari kontrol *fuzzy*; (2) Mekanisme inferensi *fuzzy* untuk mengambil keputusan dari aturan (*rule-base*) yang telah dibuat; (3) Fuzzifikasi untuk mengubah informasi masukan tegas menjadi masukan *fuzzy*; dan (4) proses defuzzifikasi yaitu menghitung keluaran kontrol *fuzzy* menjadi keluaran tegas.

2.6.1 Fungsi Keanggotaan

Jika dimisalkan x adalah suatu variabel premis, M adalah himpunan *fuzzy*, dan $\mu_m(x)$ adalah fungsi kanggotaan. Fungsi keanggotaan adalah fungsi yang memetakan variabel premis x kedalam suatu nilai antara 0 dan 1 dari suatu himpunan *fuzzy* M . Nilai pemetaan tersebut merupakan nilai derajat keanggotaan dari variabel premis.

Fungsi keanggotaan segitiga memiliki tiga parameter yang menentukan daerah keanggotaanya. Fungsi keanggotaan segitiga memiliki Persamaan (2.15) dengan parameter $a < b < c$.

$$\mu_{\text{segitiga}}(x) = \begin{cases} 0 & , x \leq a \\ \frac{x-a}{b-a} & , a \leq x \leq b \\ \frac{c-x}{c-b} & , b \leq x \leq c \\ 0 & , c \leq x \end{cases} \quad (2.16)$$

Fungsi keanggotaan *Trapezoid* memiliki empat parameter yang menentukan daerah keanggotaannya dengan parameter $a < b < c < d$ seperti pada (2.17).

$$\mu_{\text{trapezoid}}(x) = \begin{cases} 0 & , x \leq a \\ \frac{x-a}{b-a} & , a \leq x \leq b \\ 1 & , b \leq x \leq c \\ \frac{d-x}{d-c} & , c \leq x \leq d \\ 0 & , d \leq x \end{cases} \quad (2.17)$$

Fungsi keanggotaan *Gaussian* yang ditentukan dengan dua parameter yang menentukan pusat dan lebar fungsi keanggotaan sesuai (2.18).

$$\mu_{\text{gaussian}}(x) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2} \quad (2.18)$$

Fungsi keanggotaan *Generalized Bell* memiliki tiga parameter dimana parameter b bernilai positif, parameter c menentukan pusat letak x yang memiliki derajat keanggotaan bernilai satu. Fungsi *Generalized Bell* seperti pada (2.19).

$$\mu_{\text{generalizedbell}}(x) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2.19)$$

2.6.2 Mekanisme Inferensi Fuzzy

Inferensi *fuzzy* merupakan proses pengambilan keputusan sesuai dengan aturan yang telah dibuat. Penyusunan aturan *fuzzy* didasarkan pada logika manusia seperti dicontohkan:

Aturan yang disusun	: <i>If</i> air tangki adalah kosong <i>Then</i> nyalakan pompa air
Fakta	: air ditanki sudah kosong
Konsekuensi	: pompa air harus dinyalakan

Secara umum inferensi fuzzy memiliki bentuk:

If premis *Then* konsekuensi
If x is A *Then* y is B

dengan A dan B adalah nilai linguistik yang didefinisikan dengan derajat keanggotaan 1 yang didefinisikan oleh himpunan *fuzzy* dari variabel linguistik x dan y . Mekanisme inferensi *fuzzy* dapat dijelaskan sesuai langkah berikut:

1. Fuzzifikasi masukan
Mengubah masukan tegas kedalam masukan *fuzzy* yang didapatkan dari pemetaan masukan kedalam fungsi keanggotaan yang bernilai 0 sampai 1 untuk semua himpunan *fuzzy*.
2. Aplikasi operator *fuzzy*
Penerapan operator *fuzzy* (*AND* atau *OR*) untuk setiap premis dari aturan *fuzzy*.
3. Metode implikasi
Proses mendapatkan konsekuensi dengan menerapkan *minimum* atau *algebraic product* untuk operasi *AND* dan menerapkan *maximum* atau *algebraic sum* untuk operasi *OR* pada tiap aturan.
4. Agregasi keluaran
Menggabungkan semua hasil implikasi *fuzzy* dari setiap aturan untuk menghasilkan keluaran *fuzzy*.
5. Defuzzifikasi
Perhitungan untuk mengubah keluaran *fuzzy* menjadi keluaran tegas.

Dalam melakukan defuzzifikasi dikenal beberapa metode diantaranya *center of gravity* dan *center (weighted) average*. *Center of gravity* adalah defuzzifikasi dengan mencari pusat massa dari gabungan seluruh hasil implikasi *fuzzy*. Keluaran tegas dari *center of gravity* dapat dirumuskan seperti pada (2.20).

$$y_{tegas} = \frac{\sum_{i=1}^r b_i \int \mu_i(y) dy}{\sum_{i=1}^r \int \mu_i(y) dy} \quad (2.20)$$

dengan r adalah banyaknya aturan, b_i adalah titik tengah fungsi keanggotaan dan $\int \mu_i(y) dy$ adalah luas daerah konsekuensi tiap-tiap aturan. Sedangkan defuzzifikasi dengan *center (weighted) average* yaitu mencari rata-rata nilai tengah dari hasil implikasi *fuzzy*. Keluaran dari *center (weighted) average* dapat dirumuskan seperti pada (2.21).

$$y_{tegas} = \frac{\sum_{i=1}^r b_i \mu_i}{\sum_{i=1}^r \mu_i} \quad (2.21)$$

dengan r adalah banyaknya aturan, b_i adalah titik tengah fungsi keanggotaan dan μ_i adalah nilai bobot tiap aturan.

2.7 Model Fuzzy Takagi-Sugeno [8]

Model *fuzzy* Takagi-Sugeno (T-S) merupakan salah satu pemodelan yang merepresentasikan sistem nonlinear ke dalam beberapa pendekatan sistem linear. Pendekatan sistem linear ini didasarkan pada daerah kerja dari sistem. Model *fuzzy* Takagi-Sugeno akan melakukan pendekatan sistem nonlinear di beberapa daerah kerja. Sistem bisa dikatakan linear pada daerah tersebut dan dibobotkan berdasarkan nilai derajat keanggotaannya, sehingga analisis kestabilan sistem kontrol dapat diterapkan. Model *fuzzy* T-S terdiri dari dua aturan *If-Then* yaitu aturan *plant* dan aturan kontroler yang dapat dinyatakan dengan (2.22).

Aturan *plant* ke- i

If z_1 is M_{i1} and z_2 is M_{i2} ... and z_j is M_{ij}

Then $\dot{x} = A_i x + B_i u$

$$y = C_i x + D_i u \quad ; i = 1, 2, \dots, r \quad (2.22)$$

dengan $x \in \mathcal{R}^n, u \in \mathcal{R}^m, y \in \mathcal{R}^l$, r adalah jumlah aturan *fuzzy* dan g adalah jumlah himpunan *fuzzy* dalam satu aturan, M_{ig} adalah himpunan *fuzzy*, dan z adalah variabel premis yang dapat berupa variabel *state* sistem. Matriks A, B, C dan D adalah matriks konstan yang dimensinya bersesuaian.

Pada Tugas Akhir ini digunakan operator AND (*algebraic product*) sebagai mekanisme inferensi *fuzzy* dan *center* (*weighted average*) sebagai metode defuzzifikasi. Bentuk keseluruhan model *fuzzy* T-S dapat dinyatakan ke dalam Persamaan (2.23).

$$\begin{aligned} \dot{x} &= \sum_{i=1}^r m_i(z) [A_i x + B_i u] \\ y &= \sum_{i=1}^r m_i(z) [C_i x + D_i u] \end{aligned} \quad (2.23)$$

dengan

$$\begin{aligned} m(z) &= \frac{\mu_i(z)}{\sum_{i=1}^r \mu_i(z)} \\ \mu_i(z) &= \prod_{j=1} M_{ij}(z_j) \end{aligned} \quad (2.24)$$

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

$$\begin{aligned} m_i &\geq 0 \\ \sum_{i=1}^r m_i(z) &= 1, \quad i = 1, 2, \dots, r \end{aligned} \quad (2.25)$$

$$\mu_i \geq 0$$

$$\sum_{i=1}^r \mu_i(z) > 0, \quad i = 1, 2, \dots, r \quad (2.26)$$

Dari aturan *plant* yang ada dapat disusun aturan kontroler *state-feedback* dengan skema *Parallel Distributed Compensation* (PDC). Skema PDC dapat dilihat pada Gambar 2.7.

Aturan kontroler ke-*i*

If z_1 is M_{i1} and z_2 is M_{i2} ... and z_r is M_{ir}

Then $u = -K_i[x(t) - x_r(t)]; \quad i = 1, 2, \dots, r$ (2.27)

Keluaran dari kontroler *fuzzy* secara keseluruhan dapat ditulis dengan:

$$u = -\sum_{i=1}^r m_i(z)K_i[x(t) - x_r(t)] \quad (2.28)$$

Aturan *Plant*



Himpunan fuzzy sama



Aturan Kontroler



Gambar 2.7 Skema *Parallel Distributed Compensation* (PDC) [9]

Dengan memasukkan persamaan (2.28) ke persamaan (2.23) maka didapatkan keseluruhan lup tertutup dari sistem seperti pada persamaan (2.29).

$$\begin{aligned}\dot{x} &= \sum_{i=1}^r \sum_{j=1}^r m_i(z) [(A_i - B_i K_j)x] \\ y &= \sum_{i=1}^r \sum_{j=1}^r m_i(z) [(C_i - D_i K_j)x]\end{aligned}\quad (2.29)$$

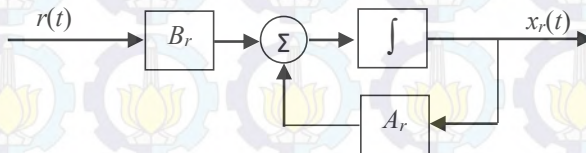
2.8 Model Referensi [6]

Pada sistem kontrol *tracking* sebuah sistem dirancang untuk bergerak mengikuti sinyal referensi yang diinginkan. Model Referensi merupakan salah satu metode untuk membantu *plant* agar mampu mengikuti trayektori referensi yang diinginkan. Secara umum Model Referensi memiliki persamaan sebagai berikut:

$$\dot{x}_r = A_r x_r + B_r r(t) \quad (2.30)$$

dengan $x_r(t)$ adalah *state* referensi, A_r adalah matriks stabil asimtotik, B_r adalah matriks masukan, dan $r(t)$ adalah masukan referensi yang nilainya terbatas. Oleh karena itu dapat diasumsikan nilai $x_r(t)$, untuk semua $t > 0$, merepresentasikan trayektori yang diinginkan untuk diikuti oleh *state plant* $x(t)$. Struktur Model Referensi dapat dilihat pada Gambar 2.8.

Model Referensi dapat ditentukan dari nilai *eigenvalue* sistem lup tertutupnya. Berdasarkan nilai *eigenvalue* tersebut dapat diketahui karakteristik dari sistem. Pada Tugas Akhir ini struktur sistem kontrol yang digunakan dimodifikasi sedemikian rupa sehingga persoalan kontrol *tracking* dapat diselesaikan menggunakan LMI. Oleh karena itu Model Referensi sulit dibentuk dari *eigenvalue* sistem lup tertutupnya, sehingga Model Referensi dibentuk dengan syarat kestabilan matriks Hurwitz.



Gambar 2.8 Model Referensi [6]

2.9 Analisis Kestabilan Lyapunov [6]

Dalam merancang sistem kontrol, kestabilan merupakan hal yang sangat penting untuk dipenuhi. Jika sebuah sistem memiliki sifat linear dan *time invariant*, banyak analisis kestabilan yang dapat diterapkan. Seperti analisis kestabilan Nyquist dan kestabilan Routh's. Namun jika sistem tersebut adalah nonlinear, atau linear tetapi bergantung pada waktu (*time-varying*), maka sulit untuk menerapkan analisis kestabilan.

Tahun 1892, A.M. Liapunov mengenalkan dua metode (metode pertama dan metode kedua) untuk menganalisa kestabilan sistem dinamik dengan menggunakan persamaan differensial. Secara eksplisit, metode pertama terdiri dari beberapa prosedur yang mengandung solusi persamaan differensial dari analisa kestabilan. Sedangkan metode kedua tidak membutuhkan prosedur-prosedur yang sulit karena tidak perlu menyelesaikan persamaan differensial dalam menganalisis kestabilan sistem, sehingga proses analisis lebih mudah. Representasi sistem dari persamaan Lyapunov yaitu:

$$\dot{x} = f(x, t) \quad (2.31)$$

Dengan $x \in \mathcal{R}^n$ adalah *state* sistem dan $f(x, t) \in \mathcal{R}^n$ adalah fungsi dari x dan t . Sistem pada (2.31) memiliki *state* ekuilibrium x_e untuk semua t seperti pada (2.32).

$$f(x_e, t) = 0 \quad (2.32)$$

Pada metode kedua Lyapunov, jika sistem dikatakan stabil asimtotik pada *state* ekuilibriumnya, maka energi yang tersimpan akan berkurang seiring berjalannya waktu hingga mencapai batas minimum dari *state* ekuilibriumnya. Oleh karena itu, Lyapunov merumuskan sebuah fungsi energi buatan yang nantinya digunakan dalam menganalisa kestabilan sistem.

Fungsi Lyapunov bergantung pada x_1, x_2, \dots, x_n dan t . Fungsi tersebut dapat didefinisikan sebagai $V(x_1, x_2, \dots, x_n, t)$ atau dapat disederhanakan $V(x, t)$. Jika secara eksplisit fungsi Lyapunov tidak mengandung t , maka fungsi dapat disederhanakan menjadi $V(x_1, x_2, \dots, x_n)$ atau $V(x)$. Dengan berdasar pada metode kedua Lyapunov, sifat dari $V(x, t)$ dan turunannya $\dot{V}(x, t) = dV(x, t)/dt$ memberikan informasi kestabilan,

kestabilan asimtotik, dan ketidakstabilan dari sebuah *state* ekuilibrium tanpa harus menyelesaikan solusi persamaan differensial (2.31).

Jika suatu sistem dinyatakan dengan (2.31) dengan

$$f(0, t) = 0, \text{ untuk semua nilai } t \quad (2.33)$$

Jika terdapat fungsi skalar $V(x, t)$ yang mempunyai turunan parsial pertama yang kontinu dan memenuhi kondisi:

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

maka *state* ekuilibriumnya stabil asimtotik pada titik origin.

Jika sebuah sistem linear time-invariant didefinisikan dengan (2.34).

$$\dot{x} = Ax \quad (2.34)$$

dengan $x \in \mathcal{R}^n$ adalah *state* sistem, $A \in \mathcal{R}^{n \times n}$ matriks konstan non-singular, dan *state* ekuilibrium pada origin ($x = 0$). Untuk mengetahui kestabilan sistem digunakan metode kedua Lyapunov pada *state* ekuilibriumnya, dengan fungsi kandidat sebagai berikut.

$$V(x) = x^T P x \quad (2.35)$$

dengan P adalah matriks simetris. Turunan pertama dari fungsi kandidat Lyapunov (2.35) adalah.

$$\begin{aligned} \dot{V}(x) &= \dot{x}^T P x + x^T P \dot{x} \\ \dot{V}(x) &= (Ax)^T P x + x^T P (Ax) \\ \dot{V}(x) &= x^T A^T P x + x^T P A x \\ \dot{V}(x) &= x^T (A^T P + P A) x \end{aligned} \quad (2.36)$$

Dari penurunan fungsi kandidat Lyapunov dan derivatifnya maka didapatkan syarat kestabilan Lyapunov, sehingga dapat dituliskan sebagai berikut:

1. $V(x,t)$ adalah definit positif
 $P > 0$ dan simetris
2. $\dot{V}(x,t)$ adalah definit negatif
 $A^T P + P A < 0$

2.10 Performa H_∞ [10]

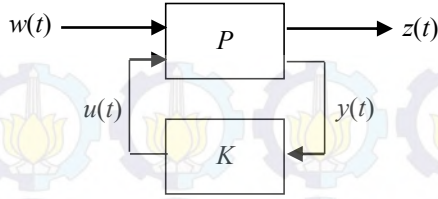
Dalam sistem nyata, ketidakpastian parameter dan gangguan dari luar sangat mempengaruhi kestabilan sistem. Ketidakpastian dan gangguan ini tidak dapat diprediksi dan diperhitungkan namun sangat berpengaruh pada kerja sistem. Oleh karena itu sistem kontrol yang dibangun harus mampu bertahan terhadap gangguan tersebut atau sering disebut *robust*.

Salah satu metode kontrol *robust* yang sering digunakan adalah kontrol *robust* H_∞ . Pada kontrol H_∞ , kontroler didesain untuk meminimalkan pengaruh terburuk pada keluaran performa sistem ketika terdapat gangguan yang mempengaruhi sistem. Pengaruh gangguan terhadap keluaran performa dapat diketahui dengan menghitung ∞ -norm dari fungsi alih sistem yang dapat didefinisikan sebagai berikut [10]:

$$\|G(s)\|_\infty = \sup_\omega \|G(j\omega)\| \quad (2.37)$$

dengan $G(s)$ merupakan fungsi alih keluaran performa terhadap gangguan. Nilai ∞ -norm pada (2.37) adalah nilai maksimal dari magnitude respon frekuensi sehingga dapat dikatakan bahwa gangguan akan teredam sesuai besar nilai ∞ -norm.

Konfigurasi sistem kontrol *robust* dapat dijelaskan seperti pada Gambar 2.9. Pada gambar tersebut dapat dilihat bahwa terdapat dua masukan dan dua keluaran pada plant. P menyatakan *plant* yang dikontrol sedangkan K adalah kontroler. Masukan yang mempengaruhi *plant* yaitu $w(t)$ sinyal gangguan yang diberikan pada sistem dan $u(t)$ sinyal kontrol. Sedangkan keluaran dari *plant* yaitu $z(t)$ keluaran performa sistem dan $y(t)$ keluaran terukur yang masuk ke kontroler.



Gambar 2.9 Konfigurasi Sistem Kontrol *Robust* [10]

Nilai penguat kontroler dirancang sedemikian rupa hingga ∞ -*norm* dari fungsi alih terhadap gangguan bernilai kurang dari level pelemahan (γ) yang diinginkan. Performa *tracking* H_∞ yang berkaitan dengan kesalahan *tracking* $x(t) - x_r(t)$ adalah sebagai berikut:

$$\frac{\|z(t)\|_2}{\|\tilde{w}(t)\|_2} = \frac{\sqrt{\int_0^{t_f} \{[x(t) - x_r(t)]^T Q[x(t) - x_r(t)]\} dt}}{\sqrt{\int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt}} \leq \gamma \quad (2.38)$$

dengan mengkuadratkan kedua ruas didapatkan

$$\frac{\int_0^{t_f} \{[x(t) - x_r(t)]^T Q[x(t) - x_r(t)]\} dt}{\int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt} \leq \gamma^2 \quad (2.39)$$

atau

$$\int_0^{t_f} \{[x(t) - x_r(t)]^T Q[x(t) - x_r(t)]\} dt \leq \gamma^2 \int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt \quad (2.40)$$

dengan $\tilde{w}(t)$ merupakan matriks augmentasi dari $w(t)$ atau gangguan, $r(t)$ adalah masukan referensi, t_f adalah waktu akhir, Q adalah matriks pembobot definit positif, dan γ adalah tingkat pelemahan. Arti secara fisik dari (2.40) adalah bahwa semua nilai $\tilde{w}(t)$ pada kesalahan *tracking*

$x(t) - x_r(t)$ akan dilemahkan sampai di bawah atau sama dengan tingkat pelemahan (γ) yang diinginkan.

2.11 Linear Matrix Inequality (LMI) [11]

Sesuai dengan teori kestabilan Lyapunov bahwa sistem dikatakan stabil jika memenuhi syarat berikut:

1. $P > 0$
2. $A^T P + P A < 0$

Dari syarat tersebut dapat dilihat bahwa matriks P harus bernilai definit positif dengan memilih matriks pembobot $Q = Q^T > 0$. Kemudian menyelesaikan syarat kedua dengan persamaan $A^T P + P A = -Q$. Untuk sistem yang sederhana, solusi matriks P dapat dicari dengan perhitungan analitik. Namun untuk sistem orde tinggi atau kompleks, perhitungan analitik sulit dilakukan, sehingga proses perhitungan dilakukan dengan numerik menggunakan metode optimasi konveks yang disebut LMI.

Linear Matrix Inequality memiliki bentuk

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i > 0 \quad (2.41)$$

dengan $x \in R^m$ adalah variabel dan $F_i = F_i^T \in R^m$, $i = 0, \dots, m$ adalah matriks simetris. Persamaan (2.41) merupakan fungsi kendala dari x dengan tujuan adalah mendapatkan matriks $F(x)$ simetris positif definit.

Beberapa LMI $F^{(1)}(x) > 0, \dots, F^{(p)}(x) > 0$ dapat dituliskan dalam satu LMI dengan peramaan diag $(F^{(1)}(x), \dots, F^{(p)}(x)) > 0$. Ketika matriks F_i adalah matriks diagonal, LMI $F(x) > 0$ menjadi kumpulan suatu pertidaksamaan linear. Jika terdapat pertidaksamaan nonlinear (2.42), dengan metode *Schur Complement* peridaksamaan tersebut dapat diubah kedalam bentuk LMI menjadi (2.43)

$$R(x) > 0, Q(x) - S(x)R^{-1}(x)S(x) > 0 \quad (2.42)$$

$$\begin{bmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{bmatrix} > 0 \quad (2.43)$$

dengan $Q(x) = Q(x)^T$, $R(x) = R(x)^T$. Jika persamaan kestabilan Lyapunov dibawa kedalam bentuk LMI maka dapat dinyatakan sebagai berikut:

$$\begin{cases} F^{(1)} = P > 0 \\ F^{(2)} = -A^T P - PA > 0 \end{cases} \quad (2.44)$$

Karena terdapat dua LMI, maka kedua LMI tersebut dapat dituliskan dalam satu LMI menjadi

$$\text{diag}(F^{(1)}(x), \dots, F^{(p)}(x)) > 0$$

$$\begin{bmatrix} F^{(1)}(x) & 0 \\ 0 & F^{(p)}(x) \end{bmatrix} > 0$$

$$\begin{bmatrix} P & 0 \\ 0 & -A^T P - PA \end{bmatrix} > 0 \quad (2.45)$$

BAB III PERANCANGAN SISTEM

Pada bab ini, perancangan sistem dibagi menjadi tiga tahap. Tahap pertama mendefinisikan model matematika *Quadrotor* kedalam bentuk persamaan *state*, sehingga bisa dilakukan analisa dan perancangan kontroler. Tahap kedua yaitu menyusun kontroler pada *Quadrotor* dimana kontroler ini akan mengatur kestabilan gerak rotasi dan gerak translasi *Quadrotor*. Pada tahap terakhir yaitu penyusunan simulasi sistem yang dibangun dengan menggunakan bantuan Matlab.

3.1 Model Dinamika *Quadrotor*

Pada bagian ini akan dijelaskan mengenai sistem dinamik *Quadrotor*, mulai persamaan dinamika hingga persamaan *state space*. Persamaan tersebut akan digunakan dalam perancangan kontroler dan simulasi pada Matlab-Simulink.

3.1.1 Persamaan *State* Dinamika *Quadrotor*

Dalam menganalisa dan desain kontroler, sebuah *plant* harus dibawa kedalam bentuk persamaan *state space*, sehingga dimungkinkan untuk melakukan linearisasi hingga penyusunan kontroler. Dengan mengacu pada lampiran A, didapatkan model dinamika *Quadrotor* seperti pada (3.1) [5].

$$\begin{cases} \ddot{X} = (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Y} = (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Z} = -g + (\cos \theta \cos \phi) \frac{U_1}{m} \\ \dot{p} = \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{U_2 l}{J_{xx}} \\ \dot{q} = \frac{J_{zz} - J_{xx}}{J_{yy}} pr + \frac{U_3 l}{J_{yy}} \\ \dot{r} = \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4 d}{J_{zz}} \end{cases} \quad (3.1)$$

dengan X , Y dan Z merupakan posisi *Quadrotor*. ϕ , θ dan ψ merupakan sudut yang dibentuk akibat gerak *roll*, *pitch* dan *yaw*, sedangkan p , q dan r merupakan kecepatan sudut. Dengan persamaan gaya yang bekerja pada *Quadrotor* adalah sebagai berikut:

$$\begin{aligned} U_1 &= F_1 + F_2 + F_3 + F_4 \\ U_2 &= F_2 - F_4 \\ U_3 &= F_1 - F_3 \\ U_4 &= F_1 - F_2 + F_3 - F_4 \end{aligned} \quad (3.2)$$

dengan F merupakan gaya angkat tiap motor.

Jika didefinisikan vektor *state* $x = [x_1 x_2 \dots x_{12}]^T \in \mathcal{R}^{12}$ dan dipilih elemen himpunan x adalah sebagai berikut:

$$\begin{aligned} x_1 &= X & x_7 &= \phi \\ x_2 &= \dot{X} & x_8 &= \dot{\phi} \\ x_3 &= Y & x_9 &= \theta \\ x_4 &= \dot{Y} & x_{10} &= \dot{\theta} \\ x_5 &= Z & x_{11} &= \psi \\ x_6 &= \dot{Z} & x_{12} &= \dot{\psi} \end{aligned} \quad (3.3)$$

Dengan menggunakan (3.3), (3.1) dapat diubah kedalam bentuk *state space* seperti pada (3.4).

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ \dot{Y} \\ \ddot{Y} \\ \dot{Z} \\ \ddot{Z} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} x_2 \\ (s_{x11}s_{x7} + c_{x11}s_{x9}c_{x7})U_1/m \\ x_4 \\ (-c_{x11}s_{x7} + s_{x11}s_{x9}c_{x7})U_1/m \\ x_6 \\ -g + (c_{x9}c_{x7})U_1/m \end{bmatrix}$$

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} x_8 \\ ((J_{yy} - J_{zz})x_{10}x_{12} + U_2l) / J_{xx} \\ x_{10} \\ ((J_{zz} - J_{xx})x_8x_{12} + U_3l) / J_{yy} \\ x_{12} \\ ((J_{xx} - J_{yy})x_8x_{10} + U_4d) / J_{zz} \end{bmatrix} \quad (3.4)$$

3.1.2 Parameter *Quadrotor*

Pada Tugas Akhir ini parameter yang digunakan adalah parameter Qball-X4. Parameter ini digunakan sebagai acuan penyusunan kontroler dan simulasi. Nilai parameter Qball-X4 dapat dinyatakan pada Tabel 3.1.

Tabel 3.1 Nilai Parameter *Quadrotor* [12]

No	Parameter	Simbol	Nilai
1	Massa	m	3,499 kg
2	Gravitasi	g	9,81 kg/m ²
3	Momen Inersia pada sumbu X	J_{xx}	0,03 kg.m ²
4	Momen Inersia pada sumbu Y	J_{yy}	0,03 kg.m ²
5	Momen Inersia pada sumbu Z	J_{zz}	0,04 kg.m ²
6	Jarak rotor dari pusat massa	l	0,2 m
7	Gaya <i>drag</i>	d	3,13x10 ⁻⁵
8	Gaya <i>thrust</i>	b	7,5x10 ⁻⁷
9	<i>Bandwith</i> aktuator	ω	15 rad/s
10	Konstanta gaya dorong	K	120 N

3.2 Perancangan Kontroler

Pada *Quadrotor* terdapat dua macam permasalahan yang harus diselesaikan agar *Quadrotor* mampu terbang dan berjalan sesuai lintasan trayektori yang diinginkan. Masalah pertama yaitu kestabilan sudut ϕ , θ dan ψ akibat gerak *roll*, *pitch*, dan *yaw*. Dan yang kedua yaitu tracking X , Y dan Z , atau memaksa *Quadrotor* agar mampu bergerak pada lintasan yang diinginkan. Pada Tugas akhir ini permasalahan kestabilan sudut ϕ , θ dan ψ akan diselesaikan dengan kontrol *state-feedback*, sedangkan permasalahan *tracking* diselesaikan dengan metode *fuzzy Takagi-Sugeno* berbasis *robust*.

3.2.1 Stabilisasi *Quadrotor*

Pada permasalahan stabilisasi, sudut ϕ , θ dan ψ harus dijaga disekitar titik ekuilibriumnya (0^0) sehingga *Quadrotor* dapat terbang dengan stabil. Dinamika gerak rotasi *Quadrotor* didekati dengan nilai sudut terkecil (0^0) sehingga didapatkan model linear didaerah ekuilibriumnya. Dengan menggunakan linearisasi jacobian didapatkan model linear dari persamaan gerak rotasi sebagai berikut.

Gerak *roll* (ϕ)

$$\dot{x}_7 = x_8$$

$$\dot{x}_8 = ((J_{yy} - J_{zz})x_{10}x_{12} + U_2l) / J_{xx} \quad (3.5)$$

Jika dilinearisasi disekitar sudut terkecil (0^0) didapatkan (3.6).

$$A_{roll} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{roll} = \begin{bmatrix} 0 \\ 6,6667 \end{bmatrix} \quad (3.6)$$

Gerak *pitch* (θ)

$$\dot{x}_9 = x_{10}$$

$$\dot{x}_{10} = ((J_{zz} - J_{xx})x_8x_{12} + U_3l) / J_{yy} \quad (3.7)$$

Jika dilinearisasi disekitar sudut terkecil (0^0) didapatkan (3.8).

$$A_{pitch} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{pitch} = \begin{bmatrix} 0 \\ 6,6667 \end{bmatrix} \quad (3.8)$$

Gerak *yaw* (ψ)

$$\dot{x}_{11} = x_{12}$$

$$\dot{x}_{12} = ((J_{xx} - J_{yy})x_8x_{10} + U_4d) / J_{zz} \quad (3.9)$$

Jika dilinearisasi disekitar sudut terkecil (0^0) didapatkan (3.10).

$$A_{yaw} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{yaw} = 10^{-3} \begin{bmatrix} 0 \\ 0,7825 \end{bmatrix} \quad (3.10)$$

Setelah dilakukan linearisasi disekitar 0, maka disusun kontrol *state-feedback*, untuk menjaga agar *state* tetap berada pada sekitar titik ekuilibriumnya. Dengan *state* berada pada disekitar titik ekuilibriumnya maka *Quadrotor* dapat terbang dengan stabil.

Pada kontrol *state-feedback* terdapat gain kontrol K yang dicari menggunakan teknik *pole placement*. Teknik *pole placement* ini mencari gain kontroler dengan melihat peletakan *pole* dominan. *Pole* dominan sendiri adalah *pole* dari *state* yang dikontrol. Sedangkan *pole state* yang lain minimal harus terletak tiga kali lebih jauh dari *pole* dominan. *Pole* dominan dapat dicari dengan pendekatan respon orde 1 seperti berikut.

$$\tau = \frac{T_s}{4}, \quad \text{dengan spesifikasi } T_s(2\%) = 4\tau$$

dengan T_s adalah waktu dimana respon mencapai nilai *steady state*-nya.

$$G = \frac{1}{\tau s + 1} = \frac{1}{\frac{T_s}{4}s + 1}, \quad \text{pendekatan orde 1}$$

sehingga *pole* desain adalah

$$s = -\frac{4}{T_s}, \quad (3.11)$$

Jika *state* diinginkan memiliki $T_s = 1$ detik, maka nilai *pole* desainnya adalah $s = -4$ sebagai *pole* dominan, sedangkan nilai *pole* yang lainnya terletak tiga kali lebih jauh dari *pole* dominan tersebut atau $p = [-4 \ -12]$.

Dengan menggunakan nilai *pole* yang telah dirancang maka dicari nilai *gain* kontroler untuk mengatasi masalah regulator, sehingga didapatkan *gain* K untuk permasalahan gerak *roll*, *pitch* dan *yaw* adalah sebagai berikut.

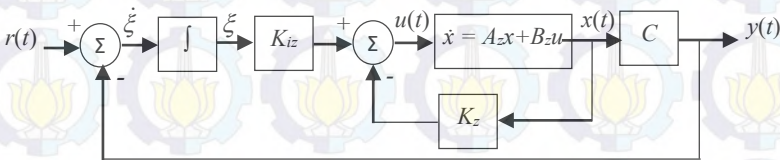
$$u = -Kx \quad (3.12)$$

dengan

$$\begin{aligned}
 K_{roll} &= [7,2 \quad 2,4] \\
 K_{pitch} &= [7,2 \quad 2,4] \\
 K_{yaw} &= 10^4 [6,1342 \quad 2,0447]
 \end{aligned}
 \tag{3.13}$$

3.2.2 Altitude Z

Posisi Z diatur pada ketinggian konstan yaitu 1 m diatas lantai. Metode yang digunakan untuk mengatur posisi Z adalah kontroler *Servo* tipe 1. Metode ini menambahkan *integrator* pada umpan balik antara *error comparator* dan *plant* seperti pada Gambar 3.1. Pada gambar tersebut dapat dilihat bahwa terdapat dua *gain* kontroler, yaitu K dan K_i . *Gain* K_z adalah *gain* umpan balik dan K_{iz} adalah *gain* kompensator. *Gain* kontroler dicari menggunakan metode *pole placement*.



Gambar 3.1. Diagram Blok Sistem Kontrol *Servo* Tipe 1 [6]

Dinamika Z memiliki persamaan seperti pada (3.14) dimana akan dilinearisasi menggunakan matriks jacobian dengan daerah kerja sekitar 0 meter. Didapatkan model linear dari dinamika posisi Z seperti pada (3.15).

$$\begin{aligned}
 \dot{x}_5 &= x_6 \\
 \dot{x}_6 &= -g + (c_{x9}c_{x7})U_1 / m
 \end{aligned}
 \tag{3.14}$$

$$A_z = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_z = \begin{bmatrix} 0 \\ 0,2858 \end{bmatrix}
 \tag{3.15}$$

Untuk mencari *gain* kontroler maka matriks sistem harus diaugmentasi seperti (3.16) sehingga gain K_{iz} dapat diperoleh.

$$A_{zj} = \begin{bmatrix} A_Z & 0 \\ -C & 0 \end{bmatrix}, \quad B_{zj} = \begin{bmatrix} B_Z \\ 0 \end{bmatrix}$$

$$A_{zj} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad B_{zj} = \begin{bmatrix} 0 \\ 0,2858 \\ 0 \end{bmatrix} \quad (3.16)$$

Jika *state* Z diinginkan memiliki $T_s = 1$ detik, maka nilai *pole* desainnya adalah $s = -4$ sebagai *pole* dominan, sedangkan nilai *pole* yang lainnya terletak tiga kali lebih jauh dari *pole* dominan tersebut atau $p = [-4 \ -12 \ -13]$. Dengan *pole* yang telah dirancang maka didapatkan *gain* kontroler (3.18), sehingga posisi Z dapat terbang sesuai referensi yang diinginkan.

$$u = -K_z x + K_{iz} \xi \quad (3.17)$$

dengan

$$K_{z_i} = 10^3 [0,8957 \quad 0,1015 \quad -2,1834]$$

$$K_z = 10^3 [0,8957 \quad 0,1015], \quad K_{iz} = 10^3 [-2,1834] \quad (3.18)$$

3.2.3 Tracking Quadrotor

Pada permasalahan *tracking Quadrotor* posisi X dan Y akan dipaksa untuk mengikuti sinyal referensi yang diinginkan. Pada Tugas Akhir ini digunakan model referensi sebagai referensi *tracking* agar respon posisi X dan Y bergerak mengikuti referensi. Kemudian kontroler disusun menggunakan metode *fuzzy* T-S berbasis performa *robust* yang diselesaikan dengan LMI.

3.2.3.1 Linearisasi Model Matematika Quadrotor

Persamaan (3.19) dan (3.21) merupakan model dinamika *Quadrotor* untuk gerak translasi pada sumbu X dan Y . Persamaan tersebut merupakan sistem nonlinear, sehingga perlu dilakukan linearisasi untuk dapat menerapkan kontrol linear yang akan dirancang. Linearisasi akan dilakukan menggunakan metode linearisasi jacobian, dengan daerah kerja disekitar titik ekuilibriumnya.

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= (s_{x11}s_{x7} + c_{x11}s_{x9}c_{x7})U_1 / m\end{aligned}\quad (3.19)$$

dengan linearisasi $A_{xi} = \frac{df(x,u)}{dx}$ dan $B_{xi} = \frac{df(x,u)}{d\theta}$, maka didapatkan

$$\begin{aligned}\dot{x} &= A_{xi} + B_{xi}\theta \\ A_{xi} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{xi} = \begin{bmatrix} 0 \\ (c_{x11}c_{x9}c_{x7})U_1 / m \end{bmatrix}\end{aligned}\quad (3.20)$$

Sedangkan linearisasi dinamika gerak sumbu Y didapatkan

$$\begin{aligned}\dot{x}_3 &= x_4 \\ \dot{x}_4 &= (-c_{x11}s_{x7} + s_{x11}s_{x9}c_{x7})U_1 / m\end{aligned}\quad (3.21)$$

dengan linearisasi $A_{yi} = \frac{df(x,u)}{dx}$ dan $B_{yi} = \frac{df(x,u)}{d\phi}$, maka didapatkan

$$\begin{aligned}\dot{x} &= A_{yi} + B_{yi}\phi \\ A_{yi} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{yi} = \begin{bmatrix} 0 \\ (-c_{x11}c_{x7})U_1 / m \end{bmatrix}\end{aligned}\quad (3.22)$$

Jika linearisasi lokal dilakukan didaerah kerja ϕ dan θ seperti pada Tabel 3.2, maka didapatkan matriks sistem hasil linearisasi seperti (3.23-3.26).

$$A_{x1} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{x1} = \begin{bmatrix} 0 \\ 9,81 \end{bmatrix}\quad (3.23)$$

$$A_{x2} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{x2} = \begin{bmatrix} 0 \\ 9,2184 \end{bmatrix}\quad (3.24)$$

$$A_{y1} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{y1} = \begin{bmatrix} 0 \\ -9,81 \end{bmatrix}\quad (3.25)$$

$$A_{y2} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{y2} = \begin{bmatrix} 0 \\ -9,2184 \end{bmatrix}\quad (3.26)$$

Tabel 3.2 Daerah Linearisasi ϕ dan θ

No	Parameter	State
1	$\phi = \theta = 0, \quad \dot{\phi} = \dot{\theta} = 0$	$[0 \ 0]$
2	$\phi = \theta = \pm 20^\circ, \quad \dot{\phi} = \dot{\theta} = 0$	$[\pm 20 \ 0]$

3.2.3.2 Pemodelan Fuzzy Takagi-Sugeno

Model *fuzzy* Takagi-Sugeno dibangun dari linearisasi persamaan gerak pada sumbu X dan Y. Linearisasi sistem dilakukan pada dua daerah kerja seperti pada Tabel 3.2, sehingga didapatkan variabel premis untuk model *fuzzy* T-S yaitu sudut ϕ untuk gerak sumbu Y dan sudut θ untuk gerak sumbu X.

Berdasarkan sistem hasil linearisasi (3.23-3.26) dapat dibentuk model *fuzzy* T-S dengan aturan *plant* sebagai berikut:

aturan *plant* untuk gerak X

Aturan *plant* ke-1:

If x_9 is M_1 (sekitar 0 rad)

Then $\dot{x} = A_{x1}x + B_{x1}\theta$ (3.27)

$y = C_{x1}x + D_{x1}\theta$

Aturan *plant* ke-2:

If x_9 is M_2 ($\pm\pi/9$ rad)

Then $\dot{x} = A_{x2}x + B_{x2}\theta$ (3.28)

$y = C_{x2}x + D_{x2}\theta$

aturan *plant* untuk gerak Y

Aturan *plant* ke-1:

If x_7 is M_1 (sekitar 0 rad)

Then $\dot{x} = A_{y1}x + B_{y1}\phi$ (3.29)

$y = C_{y1}x + D_{y1}\phi$

Aturan *plant* ke-2:

If x_7 is M_2 ($\pm\pi/9$ rad)

Then $\dot{x} = A_{y2}x + B_{y2}\phi$ (3.30)

$y = C_{y2}x + D_{y2}\phi$

dengan matriks keluaran (C_{xi} , C_{yi}) dan matriks transmisi langsung (D_{xi} , D_{yi}) adalah sebagai berikut:

$$\begin{aligned} C_{xi} = C_{yi} &= [1 \quad 0] \\ D_{xi} = D_{yi} &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned} \quad (3.31)$$

Berdasarkan skema PDC, dapat disusun aturan kontroler *fuzzy* yang bersesuaian dengan aturan *plant* (3.27-3.30) sebagai berikut:

aturan kontroler untuk gerak X

Aturan kontroler ke-1:

$$\begin{aligned} \text{If } x_9 \text{ is } M_1 \text{ (sekitar } 0 \text{ rad)} \\ \text{Then } \theta &= K_{x1}[x - x_r] \end{aligned} \quad (3.32)$$

Aturan kontroler ke-2:

$$\begin{aligned} \text{If } x_9 \text{ is } M_2 (\pm \pi / 9 \text{ rad)} \\ \text{Then } \theta &= K_{x2}[x - x_r] \end{aligned} \quad (3.33)$$

aturan kontroler untuk gerak Y

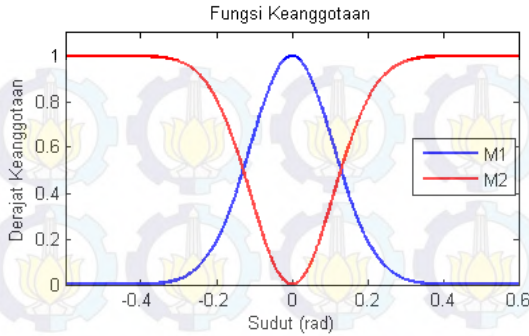
Aturan kontroler ke-1:

$$\begin{aligned} \text{If } x_7 \text{ is } M_1 \text{ (sekitar } 0 \text{ rad)} \\ \text{Then } \phi &= K_{x1}[x - x_r] \end{aligned} \quad (3.34)$$

Aturan kontroler ke-2:

$$\begin{aligned} \text{If } x_7 \text{ is } M_2 (\pm \pi / 9 \text{ rad)} \\ \text{Then } \phi &= K_{x2}[x - x_r] \end{aligned} \quad (3.35)$$

M_1 merupakan fungsi keanggotaan yang didefinisikan dengan fungsi *gaussian* seperti pada Gambar 3.2 dengan $c = 0$ dan $\sigma = 0.11$. Sedangkan M_2 merupakan fungsi keanggotaan yang nilainya komplemen dari M_1 .



Gambar 3.2 Fungsi Keanggotaan untuk Aturan *Plant* dan Kontroler

Tugas Akhir ini menggunakan metode implikasi *product* dan metode agregasi *sum*, serta metode defuzzifikasi *center (weighted) average*. Oleh karena itu keseluruhan sistem lup tertutup dapat dituliskan sebagai berikut:

$$\dot{x}(t) = \sum_{i=1}^2 h_i(x_z(t)) [A_i x(t) + B_i u(t)] + w(t) \quad (3.36)$$

$$u(t) = \sum_{j=1}^2 h_j(x_z(t)) K_j [x(t) - x_r(t)] \quad (3.37)$$

substitusi (3.37) ke (3.36)

$$\begin{aligned} \dot{x}(t) = & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) [(A_i + B_i K_j) x(t) \\ & - B_i K_j x_r(t)] + w(t) \end{aligned} \quad (3.38)$$

dengan x_z adalah *state* yang dikontrol.

$$h_j(x_z(t)) = \frac{\mu_j(x_z(t))}{\sum_{i=1}^2 \mu_i(x_z(t))}$$

$$\mu_i(x_z(t)) = M_i(x_z(t))$$

3.2.3.3 Model Referensi

Pada Subbab 2.8 sudah dijelaskan bahwa salah satu cara untuk memaksa respon *Quadrotor* bergerak mengikuti sinyal referensi adalah dengan menggunakan Model Referensi (2.30). Matriks A_r dibentuk sedemikian rupa sehingga memenuhi syarat kestabilan matriks Hurwitz, dengan *eigenvalue* negatif. A_r dan B_r dipilih dengan nilai selisih *error* terkecil antara sinyal referensi dan *state* referensi. Nilai matriks A_r dan B_r yang diperoleh seperti pada (3.39).

$$A_r = \begin{bmatrix} 0 & 1 \\ -8,5 & -7 \end{bmatrix}, \quad B_r = \begin{bmatrix} 0 \\ 8,5 \end{bmatrix} \quad (3.39)$$

3.2.3.4 Augmented Fuzzy System (AFS)

Untuk mempermudah penghitungan maka dibuat matriks perluasan (AFS) dengan menggabungkan (2.30) dengan (3.38), sehingga dapat diperoleh sebagai berikut:

$$\dot{\tilde{x}}(t) = \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) [\tilde{A}_{ij} \tilde{x}(t) + \tilde{E} \tilde{w}(t)] \quad (3.40)$$

dengan

$$\tilde{A}_{ij} = \begin{bmatrix} A_i + B_i K_j & -B_i K_j \\ 0 & A_r \end{bmatrix}, \quad \tilde{E} = \begin{bmatrix} I & 0 \\ 0 & B_r \end{bmatrix}$$

$$\tilde{x}(t) = \begin{bmatrix} x(t) \\ x_r(t) \end{bmatrix}, \quad \tilde{w}(t) = \begin{bmatrix} w(t) \\ r(t) \end{bmatrix}$$

Performa *tracking* H_∞ memiliki persamaan (2.40), namun jika kondisi awal dipertimbangkan maka dapat dimodifikasi menjadi seperti berikut:

$$\int_0^{t_f} \{ [x(t) - x_r(t)]^T Q [x(t) - x_r(t)] \} dt = \int_0^{t_f} \tilde{x}(t)^T \tilde{Q} \tilde{x}(t) dt$$

$$\leq \tilde{x}(0)^T \tilde{P}\tilde{x}(0) + \gamma^2 \int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt \quad (3.41)$$

dengan \tilde{P} adalah matriks pembobot definit positif dan

$$\tilde{Q} = \begin{bmatrix} Q & -Q \\ -Q & Q \end{bmatrix} \quad (3.42)$$

Perancangan dari sistem kontrol ini bertujuan untuk menentukan kontroler *fuzzy* dalam (3.37) untuk AFS (3.40) dengan performa *tracking* H_∞ dalam (3.41) terpenuhi untuk semua $\tilde{w}(t)$. Selain itu sistem lup tertutup (3.43) stabil kuadratik.

$$\dot{\tilde{x}} = \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \tilde{A}_{ij} \tilde{x}(t) \quad (3.43)$$

3.2.3.5 Perumusan Fungsi Lyapunov

Dipilih fungsi kandidat Lyapunov untuk AFS (3.40) adalah sebagai berikut:

$$V(\tilde{x}) = \tilde{x}^T \tilde{P}\tilde{x} \quad (3.44)$$

dengan matriks pembobot $\tilde{P} = \tilde{P}^T > 0$. Derivatif dari $V(\tilde{x})$ adalah

$$\dot{V}(\tilde{x}) = \tilde{x}^T \tilde{P}\dot{\tilde{x}} + \dot{\tilde{x}}^T \tilde{P}\tilde{x} \quad (3.45)$$

dengan mensubstitusikan AFS (3.40) ke (3.45) maka diperoleh

$$\begin{aligned} \dot{V}(\tilde{x}) = & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) [(\tilde{A}_{ij} \tilde{x}(t) + \tilde{E}\tilde{w}(t))^T \tilde{P}\tilde{x}(t) \\ & + \tilde{x}(t)^T \tilde{P}(\tilde{A}_{ij} \tilde{x}(t) + \tilde{E}\tilde{w}(t))] \end{aligned}$$

$$\begin{aligned} \dot{V}(\tilde{x}) = & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \{ \tilde{x}(t)^T (\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij}) \tilde{x}(t) \} \\ & + \tilde{w}(t)^T \tilde{E}^T \tilde{P} \tilde{x}(t) + \tilde{x}(t)^T \tilde{P} \tilde{E} \tilde{w}(t) \end{aligned} \quad (3.46)$$

3.2.3.6 Performa Tracking H_∞

Solusi umum (3.40) dengan $\tilde{P} = \tilde{P}^T > 0$ adalah petidaksamaan (3.47) untuk $i = j = 1, 2$ sehingga performa *tracking* H_∞ dalam (3.41) terjamin untuk nilai γ^2 .

$$\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij} + \frac{1}{\gamma^2} \tilde{P} \tilde{E} \tilde{E}^T \tilde{P} + \tilde{Q} < 0 \quad (3.47)$$

Bukti: Dari (3.45), maka dapat diperoleh

$$\begin{aligned} \dot{V}(\tilde{x}) = & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \{ \tilde{x}(t)^T (\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij}) \tilde{x}(t) \} \\ & + \left[\tilde{w}(t)^T \tilde{E}^T \tilde{P} \tilde{x}(t) + \tilde{x}(t)^T \tilde{P} \tilde{E} \tilde{w}(t) - \gamma^2 \tilde{w}(t)^T \tilde{w}(t) \right. \\ & \left. - \frac{1}{\gamma^2} \tilde{x}(t)^T \tilde{P} \tilde{E} \tilde{E}^T \tilde{P} \tilde{x}(t) \right] + \frac{1}{\gamma^2} \tilde{x}(t)^T \tilde{P} \tilde{E} \tilde{E}^T \tilde{P} \tilde{x}(t) \\ & + \gamma^2 \tilde{w}(t)^T \tilde{w}(t) \\ = & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \{ \tilde{x}(t)^T (\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij}) \tilde{x}(t) \} \\ & - \left(\frac{1}{\gamma} \tilde{E}^T \tilde{P} \tilde{x}(t) - \gamma \tilde{w}(t) \right)^T \left(\frac{1}{\gamma} \tilde{E}^T \tilde{P} \tilde{x}(t) - \gamma \tilde{w}(t) \right) \\ & + \frac{1}{\gamma^2} \tilde{x}(t)^T \tilde{P} \tilde{E} \tilde{E}^T \tilde{P} \tilde{x}(t) + \gamma^2 \tilde{w}(t)^T \tilde{w}(t) \\ \leq & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \left\{ \tilde{x}(t)^T (\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij}) \right. \end{aligned}$$

$$\left. + \frac{1}{\gamma^2} \tilde{P} \tilde{E} \tilde{E}^T \tilde{P} \tilde{x}(t) \right\} + \gamma^2 \tilde{w}(t)^T \tilde{w}(t) \quad (3.48)$$

Pertidaksamaan (3.47) dapat diubah menjadi

$$\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij} + \frac{1}{\gamma^2} \tilde{P} \tilde{E} \tilde{E}^T \tilde{P} < -\tilde{Q} \quad (3.49)$$

dengan memasukkan (3.49) ke (3.48) maka didapatkan

$$\begin{aligned} \dot{V}(\tilde{x}) \leq & \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \{ \tilde{x}(t)^T (-\tilde{Q}) \tilde{x}(t) \} \\ & + \gamma^2 \tilde{w}(t)^T \tilde{w}(t) \end{aligned} \quad (3.50)$$

persamaan (3.50) dapat dinyatakan sebagai (3.51) dengan didefinisikan sifat $h_i(x_z(t)) \geq 0$ dan $\sum_1^2 h_i(x_z(t)) = 1$.

$$\dot{V}(\tilde{x}) \leq -\tilde{x}(t)^T \tilde{Q} \tilde{x}(t) + \gamma^2 \tilde{w}(t)^T \tilde{w}(t) \quad (3.51)$$

dengan melakukan integral dari $t = 0$ sampai $t = t_f$ diperoleh

$$\int_0^{t_f} \dot{V}(\tilde{x}(t)) dt \leq - \int_0^{t_f} \tilde{x}(t)^T \tilde{Q} \tilde{x}(t) dt + \gamma^2 \int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt \quad (3.52)$$

$$V(\tilde{x}(t_f)) - V(\tilde{x}(0)) \leq - \int_0^{t_f} \tilde{x}(t)^T \tilde{Q} \tilde{x}(t) dt + \gamma^2 \int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt$$

(3.53)

substitusi (3.44) ke (3.53) sehingga didapatkan

$$\int_0^{t_f} \tilde{x}(t)^T \tilde{Q} \tilde{x}(t) dt \leq \tilde{x}(0)^T \tilde{P} \tilde{x}(0) + \gamma^2 \int_0^{t_f} \tilde{w}(t)^T \tilde{w}(t) dt \quad (3.54)$$

dari hasil penurunan didapatkan (3.54) sama dengan (3.41), sehingga terbukti bahwa performa *tracking* H_∞ terjamin untuk tingkat pelemahan γ . Performa *tracking* H_∞ terbaik dicari dengan formulasi minimisasi berikut:

$$\begin{aligned} & \min_{\tilde{P}} \gamma^2, \\ & \text{dengan syarat} \quad - \tilde{P} > 0 \\ & \quad \quad \quad - (3.47) \end{aligned} \quad (3.55)$$

Selanjutnya yaitu membuktikan bahwa sistem lup tertutup (3.43) adalah stabil kuadratik.

Bukti: Dari (3.51) didapatkan

$$\dot{V}(\tilde{x}(t)) < -\tilde{x}(t)^T \tilde{Q} \tilde{x}(t) < 0 \quad (3.56)$$

dan dari (3.46) didapatkan

$$\dot{V}(\tilde{x}(t)) = \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_z(t)) h_j(x_z(t)) \{ \tilde{x}(t)^T (\tilde{A}_{ij}^T \tilde{P} + \tilde{P} \tilde{A}_{ij}) \tilde{x}(t) \} \quad (3.57)$$

dari (3.56) dan (3.57) dapat diketahui memiliki bentuk yang sama, sehingga dapat dikatakan sistem lup tertutup (3.43) stabil kuadratik.

Untuk mendapatkan solusi umum $\tilde{P} > 0$ dari persoalan minimisasi (3.55) tidak mudah, sehingga perlu diubah kedalam bentuk *Linear Matrix Inequality*. Matriks pertidaksamaan (3.47) dibawa kedalam bentuk LMI dan diselesaikan menggunakan optimasi konveks.

Untuk kemudahan dalam perancangan, didefinisikan

$$\tilde{P} = \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} \quad (3.58)$$

dengan mensubstitusikan (3.58) ke dalam (3.47) didapatkan

$$\begin{aligned} & \begin{bmatrix} A_i + B_i K_j & -B_i K_j \\ 0 & A_r \end{bmatrix}^T \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} + \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} \begin{bmatrix} A_i + B_i K_j & -B_i K_j \\ 0 & A_r \end{bmatrix} \\ & + \frac{1}{\gamma^2} \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & B_r \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & B_r \end{bmatrix}^T \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} \\ & + \begin{bmatrix} Q & -Q \\ -Q & Q \end{bmatrix} < 0 \end{aligned} \quad (3.59)$$

Pertidaksamaan (3.59) dapat disederhanakan secara ringkas seperti berikut:

$$\begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} < 0 \quad (3.60)$$

dengan

$$S_{11} = (A_i + B_i K_j)^T P_{11} + P_{11} (A_i + B_i K_j) + \frac{1}{\gamma^2} P_{11} P_{11} + Q$$

$$S_{12} = S_{21}^T = -P_{11} B_i K_j - Q$$

$$S_{22} = A_r^T P_{22} + P_{22} A_r + \frac{1}{\gamma^2} P_{22} B_r B_r^T P_{22} + Q$$

Melalui metode *Schur Complement*, (3.60) ekuivalen dengan persamaan LMI berikut:

$$\begin{bmatrix} H_{11} & H_{12} & 0 \\ H_{21} & H_{22} & P_{22} B_r \\ 0 & B_r^T P_{22} & -\gamma^2 I \end{bmatrix} < 0 \quad (3.61)$$

dengan

$$H_{11} = (A_i + B_i K_j)^T P_{11} + P_{11} (A_i + B_i K_j) + \frac{1}{\gamma^2} P_{11} P_{11} + Q$$

$$H_{12} = H_{21}^T = -P_{11}B_iK_j - Q$$

$$H_{22} = A_r^T P_{22} + P_{22}A_r + Q$$

Solusi P_{11} , P_{22} dan K_j dari (3.61) dapat diselesaikan dengan dua tahap. Pertama, dicari P_{11} dan K_j dengan meminimumkan H_{11} , kemudian dimasukkan kedalam (3.61) untuk mendapatkan P_{22} .

Langkah pertama yaitu solusi dari $H_{11} < 0$

$$(A_i + B_iK_j)^T P_{11} + P_{11}(A_i + B_iK_j) + \frac{1}{\gamma^2} P_{11}P_{11} + Q < 0 \quad (3.62)$$

dengan $Y_{11} = P_{11}^{-1}$ dan $X_j = K_j Y_{11}$, (3.62) ekuivalen dengan

$$Y_{11}A_i^T + A_iY_{11} + B_iX_j + (B_iX_j)^T + \frac{1}{\gamma^2} I + Y_{11}QY_{11} < 0 \quad (3.63)$$

Melalui *Schur Complement*, (3.63) dapat diubah kedalam bentuk LMI sebagai berikut:

$$\begin{bmatrix} Y_{11}A_i^T + A_iY_{11} + B_iX_j + (B_iX_j)^T + \frac{1}{\gamma^2} I & Y_{11} \\ Y_{11} & -Q^{-1} \end{bmatrix} < 0 \quad (3.64)$$

parameter Y_{11} dan X_j dapat diperoleh dengan menyelesaikan persoalan LMI (3.64).

Langkah kedua yaitu melakukan substitusi P_{11} dan K_j ke dalam (3.61) sehingga berubah ke dalam bentuk LMI standar. Dengan cara yang sama (3.61) diselesaikan dan didapatkan solusi P_{22} . Performa *tracking* H_∞ terjamin untuk tingkat pelemahan γ jika didapatkan solusi P_{11} dan P_{22} memiliki nilai definit positif yang memenuhi (3.67).

$$\min_{P_{11}, P_{22}} \gamma^2$$

dengan syarat

- $P_{11} = P_{11}^T > 0$
- $P_{22} = P_{22}^T > 0$

(3.61)

Berdasarkan analisis yang telah dilakukan, kontrol *tracking fuzzy* berbasis performa *robust* dapat disusun berdasarkan langkah-langkah berikut:

Prosedur Desain:

1. Tentukan fungsi keanggotaan dan susun aturan *fuzzy* pada (2.22).
2. Berikan tingkat pemelemahan awal γ^2 .
3. Selesaikan (3.64) untuk mendapatkan Y_{11} dan X_j (P_{11} dan K_j juga didapatkan).
4. Substitusi P_{11} dan K_j ke dalam (3.61) kemudian selesaikan untuk mendapatkan P_{22} .
5. Turunkan tingkat pelemahan γ^2 dan ulangi langkah 3-5 sampai solusi P_{11} dan P_{22} tidak definit positif.
6. Susun kontroler *fuzzy* (2.27)

Pada Tugas Akhir ini, digunakan parameter berikut dalam perancangan sistem kontrol.

1. $Q = 10^{-1} \times \begin{bmatrix} 4,8 & 0 \\ 0 & 1,8 \end{bmatrix}$.
2. Tingkat pelemahan $\gamma = 0,85$.

Penyelesaian persoalan LMI dapat diselesaikan dengan menggunakan LMI *Toolbox* pada Matlab. Program yang telah dibangun dapat dilihat pada Lampiran B.1. Dari program tersebut didapatkan matriks K_{x1} , K_{x2} , K_{y1} , K_{y2} , P_{x1} , P_{x2} , P_{y1} , dan P_{y2} sebagai berikut:

$$K_{x1} = [-1,2448 \quad -0,7130]$$

$$K_{x2} = [-1,3247 \quad -0,7588]$$

$$K_{y1} = [1,2448 \quad 0,7130]$$

$$K_{y2} = [1,3247 \quad 0,7588]$$

$$P_{x1} = \begin{bmatrix} 1,5730 & 0,5745 \\ 0,5745 & 0,3371 \end{bmatrix}$$

$$P_{x2} = \begin{bmatrix} 0,7483 & 0,3977 \\ 0,3977 & 0,3537 \end{bmatrix}$$

$$P_{y1} = \begin{bmatrix} 1,5730 & 0,5745 \\ 0,5745 & 0,3371 \end{bmatrix}$$

$$P_{y2} = \begin{bmatrix} 0,7483 & 0,3977 \\ 0,3977 & 0,3537 \end{bmatrix}$$

Dengan parameter kontroler yang didapat kemudian disubstitusikan pada persamaan lup tertutup sistem (3.38) dan didapatkan *eigenvalue* sistem sebagai berikut:

$$\det(\lambda I - (A_i + B_i K_j \quad -B_i K_j ; \text{zeros}(2,2) \quad A_r_i)) = 0$$

$$\begin{aligned} \lambda_{x1} &= \{-3,3576, \quad -3,6371, \quad -1,5635, \quad -5,4365\} \\ \lambda_{x2} &= \{-3,3576, \quad -3,6371, \quad -1,5635, \quad -5,4365\} \\ \lambda_{y1} &= \{-3,3576, \quad -3,6371, \quad -1,5635, \quad -5,4365\} \\ \lambda_{y2} &= \{-3,3576, \quad -3,6371, \quad -1,5635, \quad -5,4365\} \end{aligned} \quad (3.65)$$

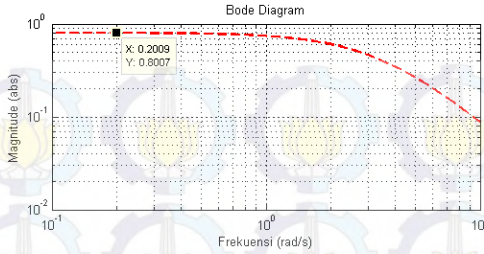
Eigenvalue dari (3.65) memiliki bilangan riil negatif yang berarti bahwa nilai kontroler K_{x1} , K_{x2} , K_{y1} , K_{y2} yang telah dirancang menjamin kestabilan sistem (3.38).

Untuk mengetahui level pelemahan pada sistem maka dicari dari *magnitute* maksimum respon frekuensi pada lup tertutup sistem seperti pada persamaan berikut:

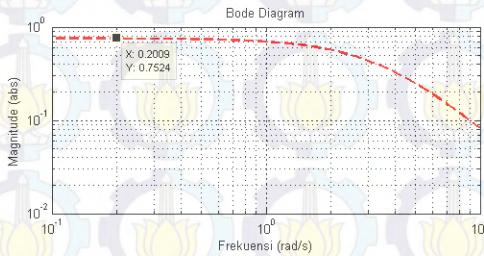
$$T_{z,w}^{(i)}(s) = C_i (sI - (A_i + B_i K_j \quad -B_i K_j ; \text{zeros}(2,2) \quad A_r_i)) B_w$$

$$\begin{aligned} T_{z,w}^{(x1)}(s) &= \frac{9,81s^2 + 68,67s + 83,39}{s^4 + 13,99s^3 + 69,97s^2 + 144,9s + 103,8} \\ T_{z,w}^{(x2)}(s) &= \frac{9,218s^2 + 64,53s + 78,36}{s^4 + 13,99s^3 + 69,97s^2 + 144,9s + 103,8} \end{aligned} \quad (3.66)$$

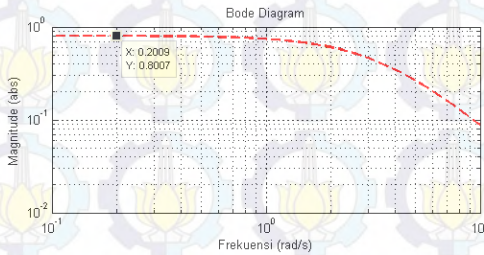
$$\begin{aligned} T_{z,w}^{(y1)}(s) &= \frac{-9,81s^2 - 68,67s - 83,39}{s^4 + 13,99s^3 + 69,97s^2 + 144,9s + 103,8} \\ T_{z,w}^{(y2)}(s) &= \frac{-9,218s^2 - 64,53s - 78,36}{s^4 + 13,99s^3 + 69,97s^2 + 144,9s + 103,8} \end{aligned} \quad (3.67)$$



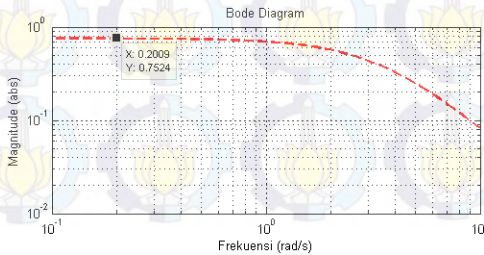
Gambar 3.3 Plot *Magnitude Bode* $T_{z,w}(x1)$



Gambar 3.4 Plot *Magnitude Bode* $T_{z,w}(x2)$



Gambar 3.5 Plot *Magnitude Bode* $T_{z,w}(y1)$



Gambar 3.6 Plot *Magnitude Bode* $T_{z,w}(y2)$

Gambar 3.3-3.6 menunjukkan plot *magnitude Bode* untuk (3.66) dan (3.67). ∞ -norm dari sistem (3.38) yaitu:

$$\begin{aligned} \|T_{z,w}^{(x1)}(s)\|_{\infty} &= \gamma_{x1}^* = 0,8007 \\ \|T_{z,w}^{(x2)}(s)\|_{\infty} &= \gamma_{x2}^* = 0,7524 \\ \|T_{z,w}^{(y1)}(s)\|_{\infty} &= \gamma_{y1}^* = 0,8007 \\ \|T_{z,w}^{(y2)}(s)\|_{\infty} &= \gamma_{y2}^* = 0,7524 \end{aligned} \quad (3.68)$$

Dari hasil (3.68) dapat diketahui bahwa ∞ -norm sistem pada *plant X* memiliki tingkat pelemahan maksimal sebesar $\gamma^* = 0,8007$ yang kurang dari $\gamma = 0,85$, sehingga nilai kontroler K_{x1} dan K_{x2} menjamin pengaruh terburuk dari gangguan $w(t)$ terhadap keluaran $z(t)$. Peristiwa ini terjadi juga pada *plant Y*, dengan tingkat pelemahan maksimal $\gamma^* = 0,8007$ yang kurang dari $\gamma = 0,85$ sehingga K_{y1} dan K_{y2} menjamin pengaruh terburuk dari gangguan $w(t)$ terhadap keluaran $z(t)$.

3.2.4 Penggabungan Sinyal Kontrol

Tujuan utama dalam Tugas Akhir ini adalah merancang sebuah kontroler yang mampu mengatur pergerakan *Quadrotor* sesuai referensi yang diberikan. Untuk dapat mengatur pergerakan *Quadrotor* maka sinyal kontrol U_1, U_2, U_3 dan U_4 akan diatur sesuai persamaan (3.65). Keluaran sinyal kontrol dari kontroler pada tiap sumbu harus dikombinasi dan ditambahkan gaya angkat minimal (u_{th}) seperti persamaan berikut:

$$\begin{aligned} u_1 &= u_{th} + v_{\theta} + v_{\psi} \\ u_2 &= u_{th} + v_{\phi} - v_{\psi} \\ u_3 &= u_{th} - v_{\theta} + v_{\psi} \\ u_4 &= u_{th} - v_{\phi} - v_{\psi} \end{aligned} \quad (3.69)$$

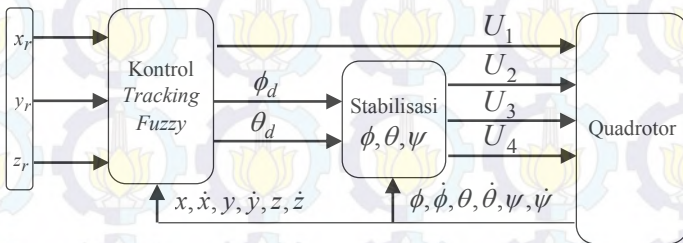
Pembuktian dari kombinasi sinyal kontrol tiap sumbu ini dapat dilihat pada (3.70) yang diturunkan berdasarkan pergerakan *Quadrotor*.

$$\begin{aligned} U_1 &= u_1 + u_2 + u_3 + u_4 \\ &= u_{th} + v_{\theta} + v_{\psi} + u_{th} + v_{\phi} - v_{\psi} + u_{th} - v_{\theta} + v_{\psi} \end{aligned}$$

$$\begin{aligned}
& +u_{th} - v_{\varphi} - v_{\psi} = 4u_{th} \\
U_2 &= u_2 - u_4 \\
&= u_{th} + v_{\varphi} - v_{\psi} - (u_{th} - v_{\varphi} - v_{\psi}) \\
&= 2v_{\varphi} \\
U_3 &= u_1 - u_3 \\
&= u_{th} + v_{\theta} + v_{\psi} - (u_{th} - v_{\theta} + v_{\psi}) \\
&= 2v_{\theta} \\
U_4 &= u_1 - u_2 + u_3 - u_4 \\
&= u_{th} + v_{\theta} + v_{\psi} - (u_{th} + v_{\varphi} - v_{\psi}) + u_{th} - v_{\theta} + v_{\psi} \\
&\quad - (u_{th} - v_{\varphi} - v_{\psi}) = 4v_{\psi}
\end{aligned} \tag{3.70}$$

3.3 Perancangan Perangkat Simulasi

Untuk menguji desain kontrol yang telah dirancang maka diuji dengan perangkat simulasi menggunakan *Simulink* Matlab. Simulasi bertujuan untuk mengetahui karakteristik kontroler yang telah dirancang dan kemampuan untuk mempertahankan respon terhadap gangguan. Dari hasil perancangan yang telah dibuat, diagram sistem kontrol yang dibangun untuk mengatur pergerakan *Quadrotor* dapat dilihat pada Gambar 3.7. Dengan menggunakan diagram kontrol tersebut dibuat diagram *Simulink* untuk mensimulasikan sistem kontrol seperti pada Gambar 3.8.



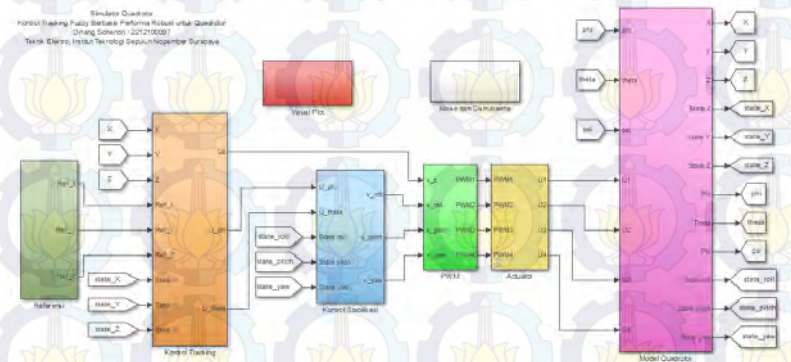
Gambar 3.7 Arsitektur Sistem Kontrol untuk *Quadrotor*

Pada Gambar 3.8, *plant* nyata *Quadrotor* direpresentasikan dengan persamaan nonlinear pada (3.1) dalam blok Model *Quadrotor*.

Pada blok tersebut terdapat masukan berupa U_1, U_2, U_3 dan U_4 serta 12 keluaran $state(x, \dot{x}, y, \dot{y}, z, \dot{z}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi})$. Keluaran $state$ muncul akibat gerak translasi dan gerak rotasi *Quadrotor*. Subsistem pendukung lain yaitu blok kontrol stabilisasi, kontrol *tracking*, referensi, *noise* dan *disturbance*, dan visual plot.

Blok referensi berisi trayektori *tracking* yang diinginkan. Referensi X, Y, Z akan masuk ke dalam kontrol *tracking*. Dengan menggunakan metode kontrol *tracking fuzzy* respon posisi X, Y, Z diatur sehingga *Quadrotor* mampu bergerak mengikuti sinyal referensi yang diinginkan. Kontrol stabilisasi merupakan blok yang mengontrol kestabilan sudut ϕ, θ, ψ agar tetap berada disekit nol, sehingga *Quadrotor* dapat terbang dengan stabil, tanpa berputar-putar dan bergerak sembarangan. Namun karena *Quadrotor* memiliki dua lup yang saling berkaitan yaitu lup dalam atau stabilisasi ϕ, θ, ψ dan lup luar *tracking* X, Y, Z , maka keluaran dari kontrol *tracking* digunakan sebagai masukan dilup dalam, sehingga kestabilan dan *tracking Quadrotor* bisa tercapai.

Untuk menguji performa *tracking* yang telah dirancang maka ditambahkan blok *noise* dan *disturbance* sebagai representasi gangguan dari luar. *Disturbance* diberikan pada *plant* posisi X dan Y sedangkan *noise* diberikan pada kecepatan X dan Y . Visual plot digunakan untuk melihat respon $state Quadrotor$ dan menggambaran hasil *tracking* dalam koordinat tiga dimensi.



Gambar 3.8 Diagram *Simulink* Sistem Kontrol *Tracking Quadrotor*

BAB IV HASIL SIMULASI

Pada bab ini akan dibahas tentang hasil yang didapatkan dari pengujian simulasi sistem kontrol yang telah dirancang pada Bab III. Stabilisasi sudut akibat gerak *Quadrotor* diatur menggunakan kontroler *state-feedback* dan *tracking* posisi X, Y, Z diatur menggunakan kontroler *fuzzy* Takagi-Sugeno berbasis performa *robust*.

4.1 Simulasi Stabilisasi *Quadrotor*

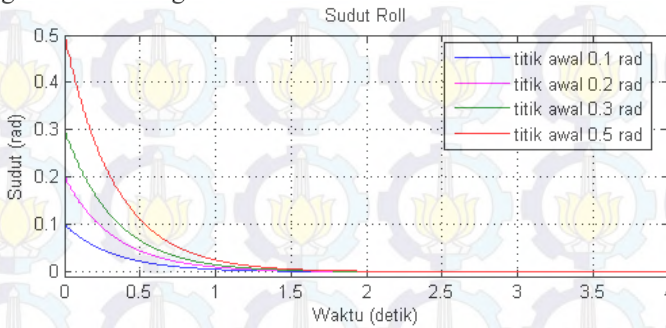
Pada sistem *Quadrotor*, stabilisasi sudut *roll*, *pitch* dan *yaw* harus dicapai terlebih dahulu sebelum kontrol *tracking* diterapkan. Karena *Quadrotor* memiliki *lup* dalam (stabilisasi) yang membutuhkan keluaran *lup* luar (*tracking*) sebagai referensi untuk bergerak. Pada subbab 3.31 dijelaskan bahwa stabilisasi sudut ϕ , θ dan ψ dicapai dengan kontroler *state-feedback* dengan parameter *gain* K seperti pada (4.1). Sudut ϕ , θ dan ψ dijaga agar kembali pada titik ekuilibriumnya (0^0), sehingga *Quadrotor* mampu terbang dengan stabil.

$$\begin{aligned} K_{roll} &= [7,2 \quad 2,4] \\ K_{pitch} &= [7,2 \quad 2,4] \\ K_{yaw} &= 10^4 [6,1342 \quad 2,0447] \end{aligned} \quad (4.1)$$

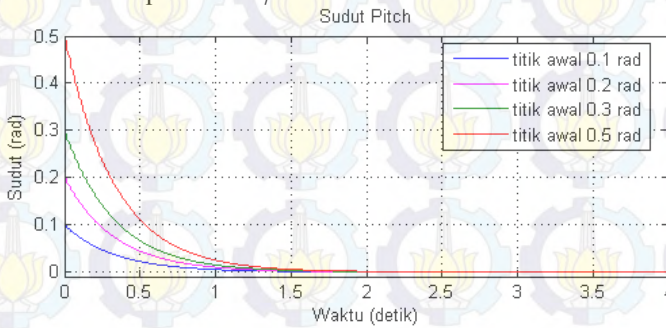
Simulasi pengujian kestabilan sudut ϕ , θ dan ψ dilakukan dengan berbagai kondisi awal, sehingga bisa dilihat bahwa kontroler yang dirancang mampu mengembalikan sudut ϕ , θ dan ψ kembali ke titik ekuilibriumnya atau stabil. Hasil dari simulasi pengujian sudut ϕ , θ dan ψ dapat dilihat pada Gambar 4.1-4.3.

Pada perancangan awal, kontroler dirancang untuk mengembalikan sudut ϕ , θ dan ψ kembali ke titik nol dengan waktu *steady state* 1 detik. Namun pada hasil simulasi, respon mencapai *steady state* pada waktu 1,75 detik dengan *error* (E_{ss}) 0% untuk sudut ϕ . Hasil yang sama juga didapatkan untuk sudut θ dan ψ . Hal ini menandakan bahwa respon sudut mengalami keterlambatan sekitar satu detik. Tetapi jika dilihat dari hasil simulasi dengan kondisi awal yang berbeda-beda *state* sudut ϕ , θ dan ψ dapat kembali ke titik ekuilibriumnya, sehingga dapat

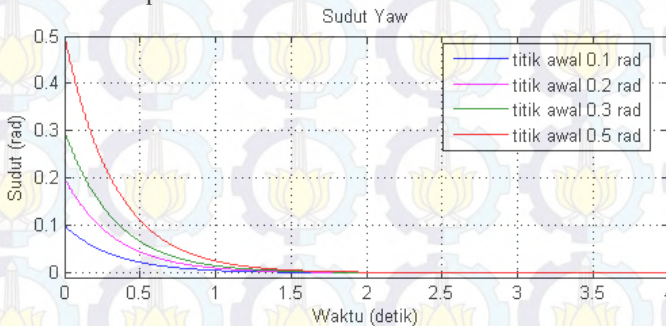
dikatakan bahwa kestabilan sudut ϕ , θ dan ψ tercapai dengan kontroler yang telah dirancang.



Gambar 4.1 Respon Sudut ϕ



Gambar 4.2 Respon Sudut θ



Gambar 4.3 Respon Sudut ψ

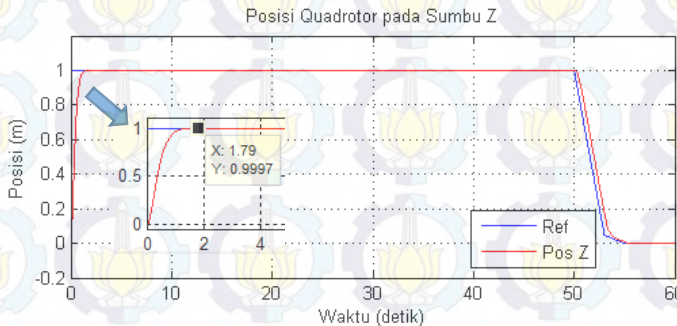
4.2 Simulasi Tracking Quadrotor

Setelah stabilisasi sudut ϕ , θ dan ψ tercapai, maka dapat diterapkan kontrol *tracking* pada *Quadrotor*. Pada subbab 3.2.2 dijelaskan bahwa *altitude Quadrotor* diatur menggunakan kontroler *Servo* tipe 1 dengan *plant* tidak memiliki *integrator*, sedangkan *tracking* posisi X dan Y diatur menggunakan kontrol *fuzzy* Takagi-Sugeno. Dengan penerapan kontrol tersebut *Quadrotor* diharapkan mampu terbang dan bergerak mengikuti lintasan *tracking* yang diinginkan.

4.2.1 Simulasi Altitude Quadrotor

Pada perancangan kontrol ketinggian, *Quadrotor* diatur agar terbang pada ketinggian 1 meter di atas permukaan dan mempertahankan ketinggian pada saat bergerak pada sumbu X maupun sumbu Y . Parameter (4.2) merupakan kontroler yang dirancang agar ketinggian Z memiliki respon dengan waktu *steady state* 1 detik, namun pada hasil simulasi posisi Z memiliki waktu *steady state* 1,8 detik dengan *error (Ess)* 0% seperti pada Gambar 4.4. Hal ini menunjukkan bahwa respon memiliki keterlambatan waktu untuk mencapai nilai referensi.

$$K_z = 10^3 [0,8957 \quad 0,1015], K_{iz} = 10^3 [-2,1834] \quad (4.2)$$



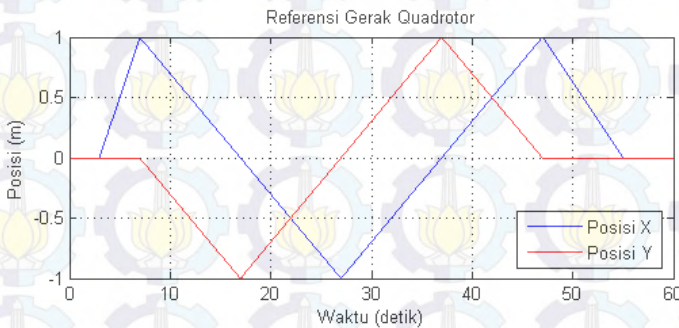
Gambar 4.4 Posisi Z *Quadrotor*

Dari hasil simulasi tersebut dapat dilihat bahwa posisi Z terbang dari ketinggian 0 meter menuju referensi 1 meter dengan waktu *steady state* 1,8 detik. Setelah ketinggian tercapai kontroler akan mempertahankan ketinggian sampai akhirnya *Quadrotor* mendarat.

Kontroler yang dirancang mampu mempertahankan posisi Z dengan *Quadrotor* bergerak pada sumbu X , Y , hal ini terlihat bahwa dari detik ke 2 sampai ke 50 posisi Z tetap berada pada ketinggian 1 meter.

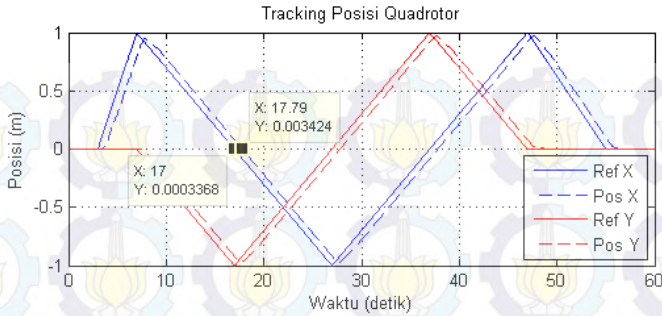
4.2.2 Simulasi Gerak Sumbu X, Y *Quadrotor*

Kontrol *tracking fuzzy* yang telah dirancang, akan digunakan untuk menerbangkan *Quadrotor* dengan referensi seperti pada Gambar 4.5. *Quadrotor* akan disimulasikan terbang pada sumbu X dan Y dengan masing-masing lintasan berupa gelombang segitiga yang mempunyai amplitudo 1 meter. Simulasi dilakukan dengan total waktu 60 detik. *Quadrotor* akan terbang dengan lintasan lurus menuju koordinat (1,0) pada detik ketuju kemudian menuju (0,-1) pada detik 17, (-1,0) pada detik ke 27, (0,1) pada detik ke 37, dan (1,0) pada detik ke 47, serta kembali ke posisi awal.

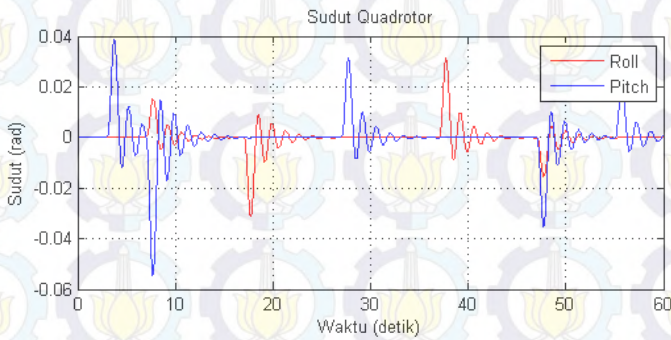


Gambar 4.5 Referensi *Tracking* untuk *Quadrotor*

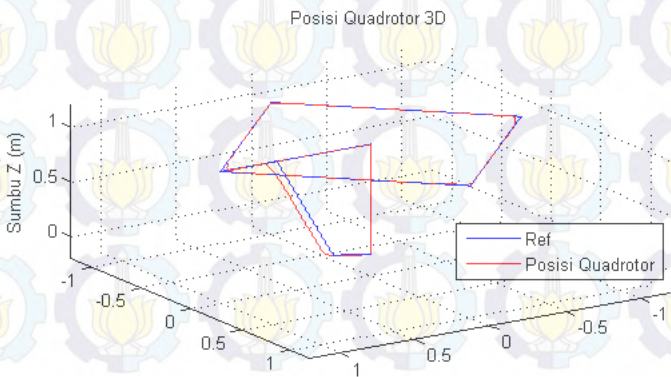
Fuzzy Takagi-Sugeno dirancang untuk mengendalikan gerak *Quadrotor* pada sumbu X , Y . Pada subbab 3.2.3.6 telah didapatkan parameter kontroler untuk mengatur gerak *Quadrotor*. Kontroler yang telah dirancang diharapkan mampu mengatur gerak *Quadrotor* mengikuti sinyal referensi. Kontroler juga diatur untuk mampu bertahan terhadap gangguan, sehingga hasil desain terbukti *robust*. Simulasi dilakukan dengan menggunakan arsitektur sistem kontrol seperti pada Gambar 3.8 untuk gerak *Quadrotor* pada sumbu X dan Y . Hasil dari simulasi dapat dilihat pada Gambar 4.6.



Gambar 4.6 Tracking Posisi *Quadrotor*



Gambar 4.7 Sudut Roll dan Pitch untuk Gerak *Quadrotor*



Gambar 4.8 Plot 3D Hasil Simulasi

Dari hasil simulasi dapat diketahui bahwa *Quadrotor* mampu bergerak mengikuti referensi *tracking* yang diinginkan dengan perubahan sudut *roll* dan *pitch* tidak terlalu besar. Hal ini menunjukkan bahwa *Quadrotor* dapat bergerak dengan stabil dengan *Integral Absolute Error* (IAE) 0,1149 pada sumbu *X* dan 0,06171 pada sumbu *Y*. Akan tetapi gerak tersebut memiliki osilasi di awal gerak dan muncul beda fasa terhadap referensi sebesar 0,79 detik. Beda fasa ini muncul akibat struktur kontrol yang digunakan, dimana keluaran aktual dari *plant* tidak dibandingkan langsung dengan referensi sebagai masukan *error* pada kontroler. Struktur kontrol tersebut secara alami akan menghasilkan beda fasa antara referensi dan aktual. Dapat dilihat pada Gambar 4.9, Model Referensi yang digunakan memiliki beda fasa sebesar 0,78 detik. Beda fasa pada Model Referensi ini diakibatkan oleh struktur kontrol yang digunakan.



Gambar 4.9 Perbandingan Referensi *Tracking* dengan Model Referensi untuk Gerak *Quadrotor*

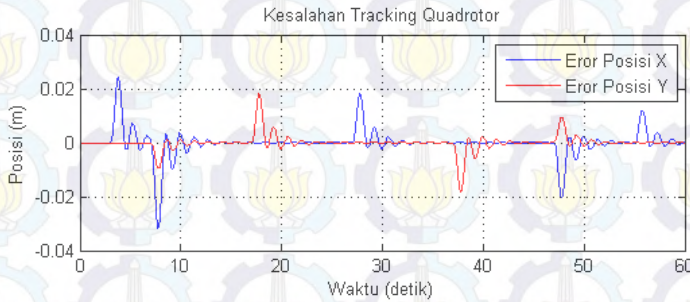
Dengan menggunakan kontroler yang telah dirancang *Quadrotor* bergerak dan memiliki kesalahan posisi sekitar 0,02 meter. Kesalahan ini bersangsur-angsur berkurang sampai nol, sehingga gerak *Quadrotor* tercapai. Kesalahan *tracking* pada *Quadrotor* dengan metode kontrol yang telah dirancang dapat dilihat pada Gambar 4.10.

4.2.3 Simulasi Gerak *Quadrotor* dengan Gangguan

Untuk menguji ketahanan sistem kontrol yang dirancang maka dilakukan simulasi sistem kontrol dengan pemberian gangguan. Gangguan yang diberikan memiliki dua jenis yaitu gangguan pada *plant* dan *noise* pada pengukuran. Gangguan hembusan angin diberikan pada

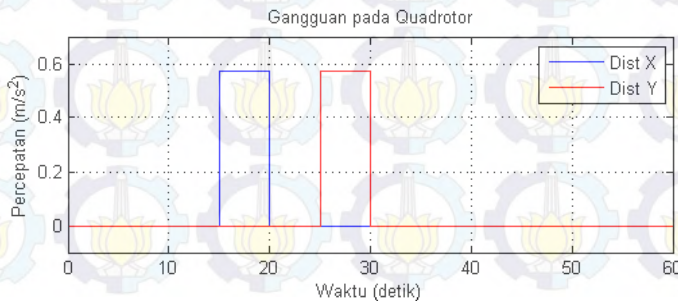
plant posisi X dan *plant* posisi Y seperti pada (4.3) yang diibaratkan dengan sinyal pulsa dengan besar $2N$ selama lima detik. Gangguan ini diberikan pada *plant* posisi X dan Y sehingga harus disesuaikan dengan *plant* posisi X dan Y dengan membagi gangguan dengan massa *Quadrotor* sehingga berubah menjadi percepatan. Amplitudo gangguan berubah menjadi $0,5716$.

$$\begin{aligned}\ddot{X} &= (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{U_1}{m} + \Delta Dx \\ \ddot{Y} &= (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{U_1}{m} + \Delta Dy\end{aligned}\quad (4.3)$$



Gambar 4.10 Kesalahan *Tracking* Gerak *Quadrotor*

Gangguan *plant* posisi X diberikan pada detik ke 15 sampai 20 sedangkan gangguan pada *plant* posisi Y diberikan pada detik 25 sampai 30 seperti pada Gambar 4.11.



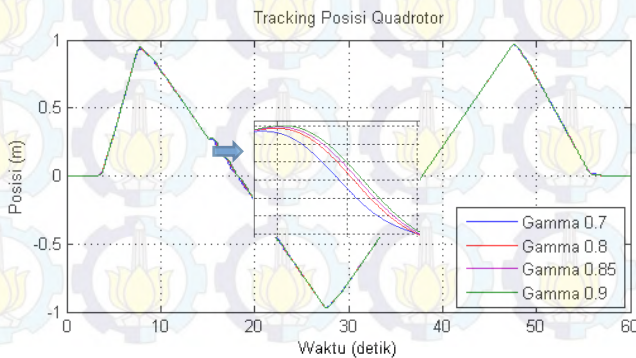
Gambar 4.11 Gangguan pada *Quadrotor*

Untuk melihat pengaruh γ , maka divariasikan dengan nilai 0,9, 0,85, 0,8, dan 0,7. Pemberian level pelemahan yang berbeda-beda akan mempengaruhi nilai dari matriks P_{xi} sehingga nilai K_{x1} , K_{x2} dan ∞ -norm sistem yang didapat juga berbeda-beda. Hasil yang diperoleh *gain state-feedback* K_{x1} , K_{x2} dan ∞ -norm untuk nilai γ yang berbeda-beda dapat dilihat pada Tabel 4.1.

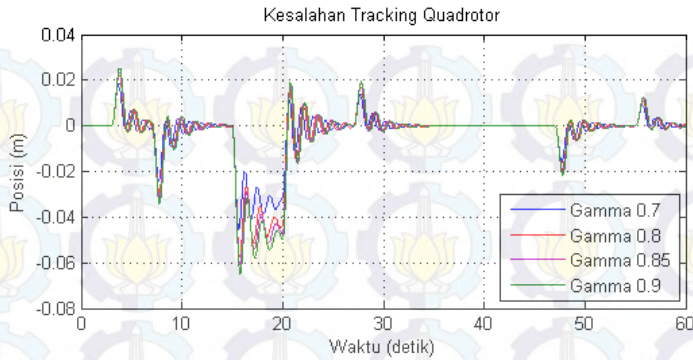
Tabel 4.1 Nilai *Gain State-Feedback* dan ∞ -norm sistem dengan γ yang berbeda-beda

γ	K_{x1}	K_{x2}	$\ T_{z,w}(s)\ = \gamma^*$
0,9	[-1,1678 -0,6751]	[-1,2428 -0,7184]	0,8535
0,85	[-1,2448 -0,7130]	[-1,3247 -0,7588]	0,8007
0,8	[-1,3371 -0,7572]	[-1,4229 -0,8058]	0,7454
0,7	[-1,6935 -0,9303]	[-1,8022 -0,9900]	0,5883

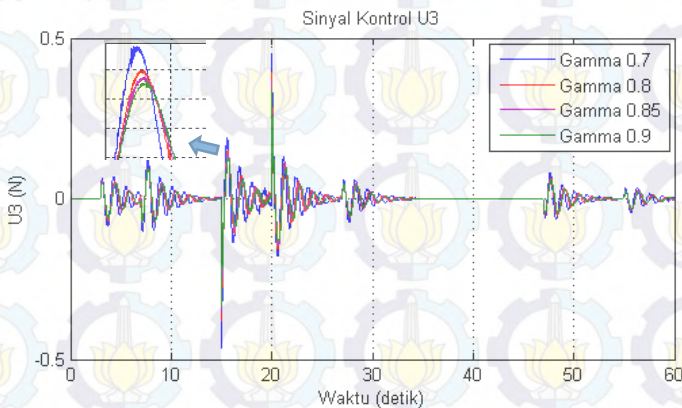
Pada sumbu X gerak *Quadrotor* mengalami sedikit pergeseran dari lintasan awal, pada detik ke 15 posisi *Quadrotor* bergeser sehingga muncul deviasi posisi sekitar 0,06 meter. Variasi γ berpengaruh pada besar deviasi kesalahan posisi *Quadrotor*, dengan deviasi terkecil ketika nilai $\gamma = 0,7$. Untuk melihat jumlah *error* posisi *Quadrotor* pada sumbu X dapat dilihat pada Tabel 4.2. Ketika gangguan diberikan pada detik ke 15, muncul kesalahan posisi *Quadrotor* sampai detik ke 20 ketika gangguan sudah hilang.



Gambar 4.12 Perbandingan Posisi *Quadrotor* dengan Variasi γ



Gambar 4.13 Kesalahan Posisi X



Gambar 4.14 Sinyal Kontrol U_3 untuk Gerak Sumbu X

Dapat dilihat pada Gambar 4.14, pada detik ke 15 gangguan diberikan pada *plant* X , sinyal kontrol U_3 berubah sebesar $-0,5$ N. Sinyal kontrol berkurang sebesar $0,5$ N untuk mengatasi pergeseran posisi X . Ketika gangguan diberikan, posisi gerak *Quadrotor* dijaga disekitar lintasan gerak dan ketika gangguan sudah hilang maka posisi kembali pada lintasan yang diinginkan.

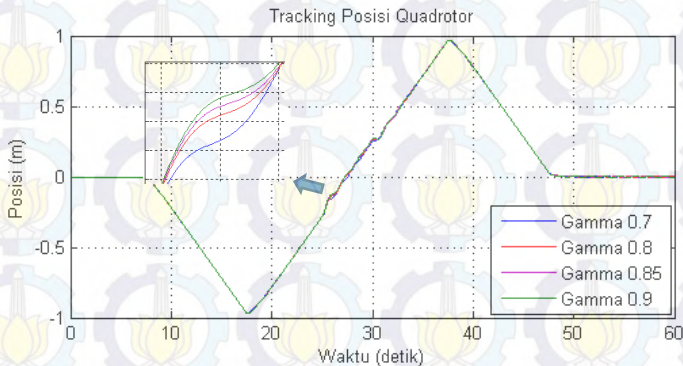
Tabel 4.2 Kesalahan Posisi *Quadrotor* pada Sumbu *X* dengan Berbagai Nilai γ

γ	Error	IAE
0,9	$\pm 0,065$	0,3976
0,85	$\pm 0,061$	0,3729
0,8	$\pm 0,057$	0,3459
0,7	$\pm 0,045$	0,2701

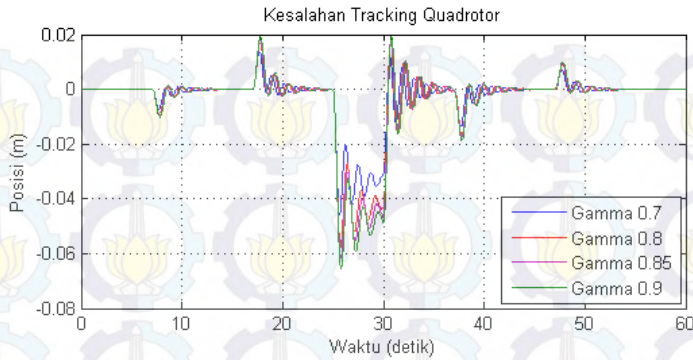
Pengaruh yang sama juga terjadi pada sumbu *Y*. *Plant* posisi *Y* diberi gangguan dengan nilai γ divariasikan 0,9, 0,85, 0,8, dan 0,7. Hasil yang diperoleh *gain state-feedback* K_{y1} , K_{y2} dan ∞ -norm untuk nilai γ yang berbeda-beda dapat dilihat pada Tabel 4.3.

Tabel 4.3 Nilai *Gain State-Feedback* dan ∞ -norm Sistem dengan γ yang Berbeda-beda

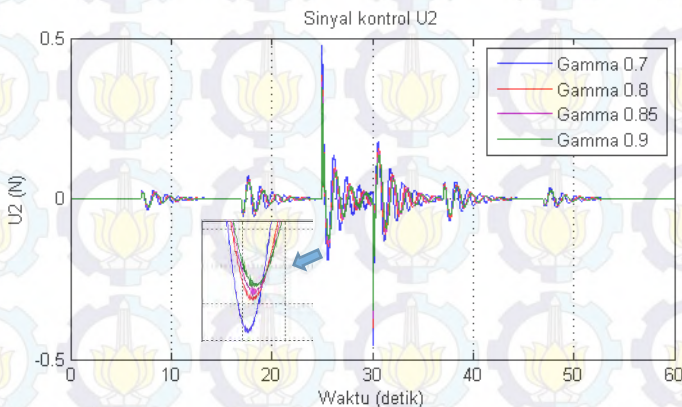
γ	K_{y1}	K_{y2}	$\ T_{z,w}(s)\ = \gamma^*$
0,9	$[-1,1678 \quad -0,6751]$	$[-1,2428 \quad -0,7184]$	0,8535
0,85	$[-1,2448 \quad -0,7130]$	$[-1,3247 \quad -0,7588]$	0,8007
0,8	$[-1,3371 \quad -0,7572]$	$[-1,4229 \quad -0,8058]$	0,7454
0,7	$[-1,6935 \quad -0,9303]$	$[-1,8022 \quad -0,9900]$	0,5883



Gambar 4.15 Perbandingan Posisi *Quadrotor* dengan Variasi γ



Gambar 4.16 Kesalahan Posisi Y



Gambar 4.17 Sinyal Kontrol U_2 untuk Gerak Sumbu Y

Pada detik ke 25, posisi Y mengalami pergeseran sehingga mempunyai deviasi posisi sekitar 0,06 meter seperti pada Gambar 4.16. Deviasi ini muncul selama gangguan tetap diberikan. Variasi γ yang diberikan mempengaruhi nilai *error* posisi Y dengan nilai deviasi terkecil pada $\gamma = 0,7$. Pada Tabel 4.4 dapat dilihat dengan $\gamma = 0,7$ memiliki nilai *error* dan IAE terkecil, sehingga $\gamma = 0,7$ merupakan level pelemahan optimal. Sinyal kontrol U_2 bertambah 0,5 N untuk mengatasi pergeseran posisi Y . Oleh karena itu posisi *Quadrotor* dijaga disekitar lintasan gerak dan dikembalikan pada lintasan tersebut.

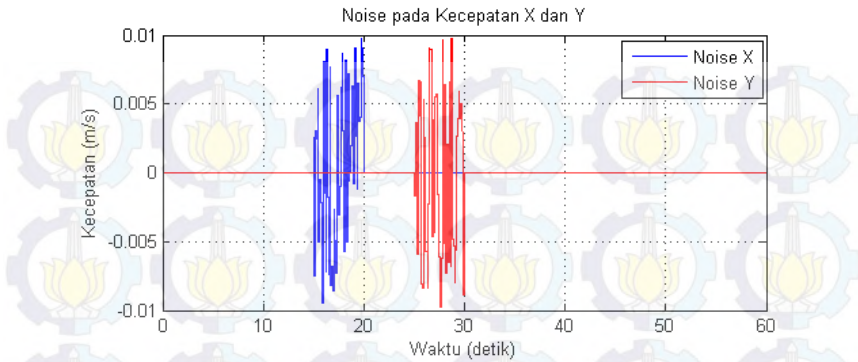
Tabel 4.4 Kesalahan Posisi *Quadrotor* pada Sumbu Y dengan Berbagai nilai γ

γ	<i>Error</i>	<i>IAE</i>
0,9	$\pm 0,065$	0,3976
0,85	$\pm 0,061$	0,3729
0,8	$\pm 0,057$	0,3459
0,7	$\pm 0,045$	0,2701

Dari hasil simulasi tersebut kesalahan posisi terkecil baik pada sumbu *X* atau sumbu *Y* terjadi ketika nilai $\gamma = 0,7$. Pada level pelemahan tersebut posisi *Quadrotor* dijaga dengan kokoh disekitar lintasan gerakannya, sehingga dapat disimpulkan bahwa sistem yang telah dirancang terbukti *robust* terhadap gangguan dari luar.

Setelah didapatkan nilai level pelemahan yang optimal maka dilakukan pengujian sistem terhadap *noise*. Pengujian *noise* pada *Quadrotor* diberikan pada sensor pengukur kecepatan *Quadrotor*. Diibaratkan sensor pengukur kecepatan *Quadrotor* mengalami gangguan sehingga tidak mampu memberikan informasi kecepatan secara akurat. Pada kecepatan *X*, *noise* diberikan pada detik ke 15-20 sedangkan kecepatan *Y* pada detik 25-30 seperti pada Gambar 4.18. *Noise* yang diberikan berupa sinyal acak yang dimodelkan dengan *Uniform Random Number* dengan variasi mulai 5% - 80% dari nilai nominalnya.

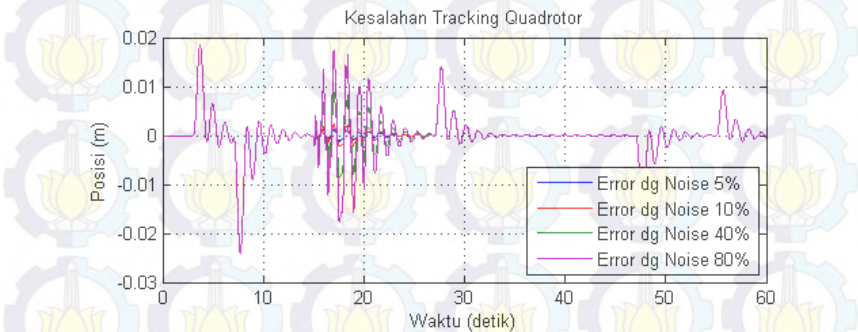
Pada Gambar 4.19-4.20 dapat dilihat bahwa semakin besar kesalahan pembacaan sensor maka kesalahan posisi *Quadrotor* juga semakin besar. Ketika besar *noise* yang diberikan 5% *Quadrotor* mengalami kesalahan posisi sebesar 0,001 m pada posisi *X* dan posisi *Y*. Sedangkan ketika *noise* yang diberikan 80 % kesalahan posisi sebesar 0,012 m pada posisi *X* dan *Y*.



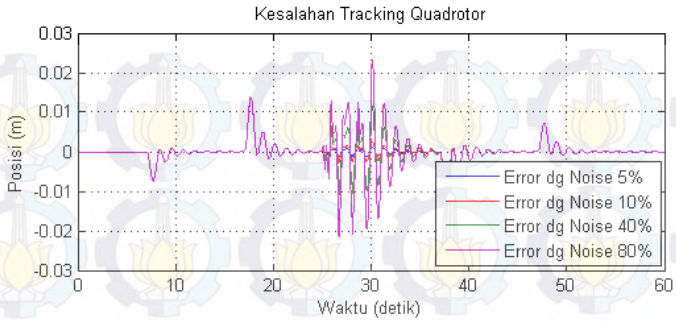
Gambar 4.18 Noise pada Sensor Kecepatan X dan Kecepatan Y

Gambar 4.21-4.22 merupakan sinyal kontrol U_2 dan U_3 yang bekerja untuk mengatasi *noise* yang diberikan. Ketika *noise* 5% sinyal kontrol bernilai 0,5 N sedangkan pada *noise* 80% sinyal kontrol bernilai cukup besar yaitu sekitar 8 N. Sinyal kontrol mengalami *chattering*, sehingga membutuhkan energi yang cukup besar. Besar sinyal kontrol untuk mengatasi *noise* sebanding dengan besar *noise* yang diberikan.

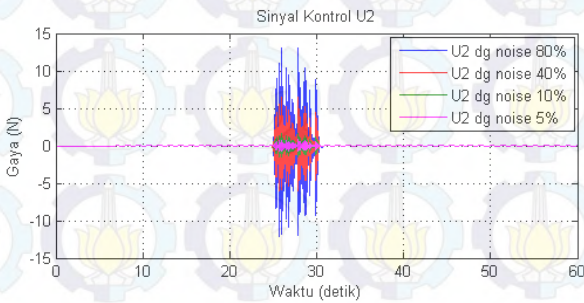
Hasil simulasi pengaruh *noise* dirangkum pada Tabel 4.5 dengan besar *noise* divariasikan mulai 5% sampai 80%. *Noise* divariasikan untuk mengetahui batas kemampuan kontroler untuk mentoleransi *noise* tersebut.



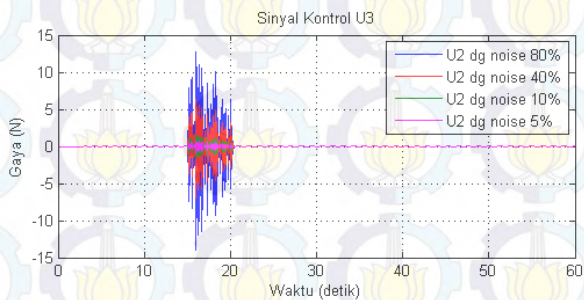
Gambar 4.19 Kesalahan Posisi pada *Quadrotor* dengan Pengaruh *Noise* pada Sumbu X



Gambar 4.20 Kesalahan Posisi pada *Quadrotor* dengan Pengaruh *Noise* pada Sumbu *Y*



Gambar 4.21 Sinyal Kontrol *U2* untuk Gerak *Quadrotor*



Gambar 4.22 Sinyal Kontrol *U3* untuk Gerak *Quadrotor*

Tabel 4.5 Hasil Simulasi Gerak *Quadrotor* dengan Variasi *Noise*

No	Besar <i>Noise</i>	Deviasi Posisi X (m)	Deviasi Posisi Y (m)	U2 (N)	U3 (N)	IAE Sb X	IAE Sb Y
1	5%	0,001	0,001	0,5	0,5	0,0866	0,0492
2	10%	0,002	0,002	1	1	0,0901	0,0536
3	20%	0,005	0,005	2	2	0,0972	0,0624
4	40%	0,01	0,01	4	4	0,1114	0,0801
5	80%	0,02	0,02	8	8	0,1398	0,1152

Pada Tabel 4.5, *noise* yang yang diberikan sangat berpengaruh signifikan pada sinyal kontrol U_2 dan U_3 . Untuk mengkompensasi *noise* 10%, sinyal kontrol yang bekerja cukup besar yaitu 1 N. Jika dibandingkan dengan sinyal kontrol tanpa *noise* yaitu 0.15 N, nilai tersebut memiliki selisih cukup besar. Ditambah U_2 dan U_3 mengalami *chattering*, sehingga membutuhkan energi yang besar pula. Hal ini menunjukkan bahwa untuk menjaga posisi *Quadrotor* karena pengaruh *noise* dibutuhkan energi yang cukup besar sehingga akan menguras energi baterai.



BAB V PENUTUP

5.1 Kesimpulan

Metode kontrol *fuzzy* Takagi-Sugeno bekerja dengan baik untuk mengendalikan gerak *Quadrotor* mengikuti lintasan yang diinginkan dengan nilai *Integral Absolute Error* (IAE) 0,1149 pada sumbu X dan 0,06171 pada sumbu Y . Pada saat bergerak *Quadrotor* memiliki beda fasa sebesar 0,79 detik. Kontrol *tracking fuzzy* yang dirancang memiliki performa *robust* dengan performa keluaran dibandingkan dengan tingkat pelemahan gangguan kurang dari sama dengan tingkat pelemahan γ yang ditentukan. Hal ini menunjukkan bahwa ketika gangguan dan *noise* diberikan pada sistem, kontroler mampu mereduksi pengaruh yang muncul sehingga performa keluaran dari sistem masih sesuai dengan yang diinginkan.

5.2 Saran

Pada saat pengujian *noise*, sinyal kontrol mengalami efek *chattering* dengan amplitudo yang besar. Hal ini menunjukkan bahwa *Quadrotor* membutuhkan energi yang cukup besar untuk mereduksi *noise*. Pengaruh *chattering* ini sangat tidak diinginkan, sehingga perlu diminimalkan atau bahkan dihilangkan.

DAFTAR PUSTAKA

- [1] Wang, Hanghao dan Wang, Mao, “Robust Fuzzy Variable Structure Control of T-S Model for a Quadrotor Unmanned Air Vehicle”, *International Journal of Multimedia and Ubiquitous Engineering*, China, 2015.
- [2] Fang, Zhuo dkk., “Feedback Linearization and Continuous Sliding Mode Control for a Quadrotor UAV”, *Proceedings of the 27th Chinese Control Conference*, China, 2008.
- [3] Razinkova, Anastasia dkk., “Adaptive Control over Quadcopter UAV under Disturbances”, *International Conference on Control, Automation and Systems (ICCAS 2014)*, Korea, 2014.
- [4] Lee, Hyeonbeom dan Kim, H. J., “Robust Control of a Quadrotor using Takagi-Sugeno Fuzzy Model and an LMI Approach”, *International Conference on Control, Automation and Systems (ICCAS 2014)*, Korea, 2014.
- [5] Tommaso Bresciani, “Modelling, Identification and Control of a Quadrotor Helicopter”. *Thesis*, Department of Automatic Control Lund University, 2008.
- [6] Ogata, Katsuhiko, “*Modern Control Engineering 3rd edition*”, Prentice-Hall, New Jersey, 1997.
- [7] Passino, K. M. dan Yurkovich, S., “*Fuzzy Control*”, Addison Wesley Longman, California, 1998.
- [8] Tanaka, Kazuo. dan Wang H. O., “*Fuzzy Control System Design and Analysis*”, John Wiley & Sons, Inc., 2001
- [9] Altway, M. dan Agustinah T., “Desain Kontrol Tracking Fuzzy dengan Batasan Input-Output pada Sistem Pendulum Kereta”, *Tugas Akhir*, Teknik Elektro ITS, Surabaya, 2014.
- [10] Zhou, Kemin, “*Essentials of Robust Control*”, Prentice-Hall, New Jersey, May 1999.
- [11] Boyd, Stephen, El Ghaoui, L., Feron, Eric, dan Balakrishnan, V., “*Linear Matrix Inequality in System and Control Theory*”, SIAM, Philadelphia, 1994
- [12] _____, “*Quanser Qball-X4*”, Document Number 888.Revision:2, Inovative Educate, Quanser, 2012.
- [13] Spong, M.W, Vidyasagar, M. “*Robot Dynamic and Control*”, Prentice Hall, New Jersey, 2004.

LAMPIRAN A

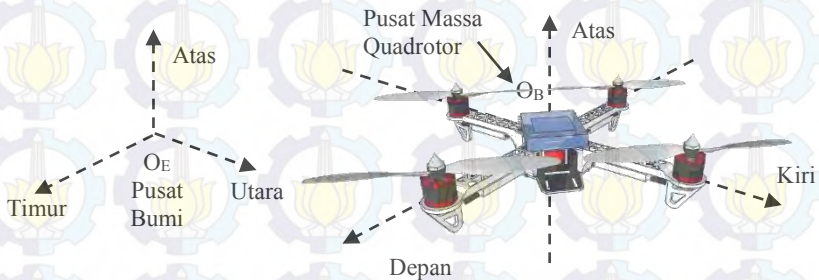
A.1 Pemodelan *Quadrotor* [5]

Pada bagian ini akan dijelaskan analisis tentang penurunan persamaan matematika dari *Quadrotor*, dimana analisis ini diperoleh dari analisis kinematika dan dinamika dari *Quadrotor*. Hasil penurunan persamaan model matematika ini akan menggambarkan semua pergerakan *Quadrotor* dan dari persamaan yang diperoleh digunakan sebagai acuan dalam menyusun kontroler sesuai dengan yang diinginkan.

A.1.1. Koordinat *Frame*

Quadrotor merupakan sistem yang memiliki 6 derajat kebebasan atau 6 *degree of freedom* (DOF). Sehingga digunakan dua sistem *frame* koordinat sebagai referensi pergerakan *Quadrotor*, yaitu *earth frame* (*E-frame*) dan *body frame* (*B-frame*).

Frame Bumi (*E-frame*) akan menentukan orientasi *Quadrotor* terhadap bumi, dimana *E-frame* ini akan menunjukkan arah utara-timur-atas terhadap titik origin O_E . Titik origin bumi pada dasarnya akan berubah-ubah sesuai dengan titik awal *Quadrotor* terbang. Pada Gambar A.1 dapat diilustrasikan koordinat *frame* bumi yang digunakan sebagai referensi *Quadrotor*, dengan sumbu *X* positif mengarah ke arah utara, sumbu *Y* positif mengarah ke arah timur, dan sumbu *Z* positif mengarah keatas atau tegak lurus terhadap bumi.



Gambar A.1 *Frame* Bumi (*E-frame*) dan *Frame* Badan (*B-frame*)

Frame Badan (*B-frame*) pada *Quadrotor* akan menentukan orientasi *Quadrotor* berdasarkan *body* dari *Quadrotor* itu sendiri. Titik

origin dari *B-frame* ini berada pada pusat massa dari *Quadrotor*, sehingga pergerakan dari *Quadrotor* dapat didefinisikan sebagai gerak *roll* (ϕ) atau mengguling kiri-kanan pada sumbu y , gerak *pitch* (θ) atau maju-mundur pada sumbu x dan gerak *yaw* (ψ) atau memutar pada sumbu z .

A.1.2. Asumsi Pemodelan *Quadrotor*

Untuk memudahkan dalam pemodelan *Quadrotor* maka dapat diterapkan beberapa asumsi berikut:

- a) Bumi dianggap datar, tidak berotasi dan sesuai dengan acuan inersia dari *frame*.
- b) Percepatan gravitasi dianggap konstan dan tegak lurus terhadap permukaan bumi.
- c) Desain pemodelan *Quadrotor* yang dibuat, dianggap simetris dengan asumsi sumbu yang ditentukan sebelumnya.
- d) Badan serta baling-baling *Quadrotor* dianggap sebagai benda tegar (*rigid*), sehingga untuk pemodelan dapat menggunakan pendekatan *Newton-Euler*.
- e) Qball-X4 merupakan jenis *Quadrotor* yang digunakan untuk eksperimen dalam ruangan. Oleh karena itu, kecepatan dianggap rendah, gesekan udara, efek *gyroscopic* dan torsi aerodinamis dapat diabaikan.

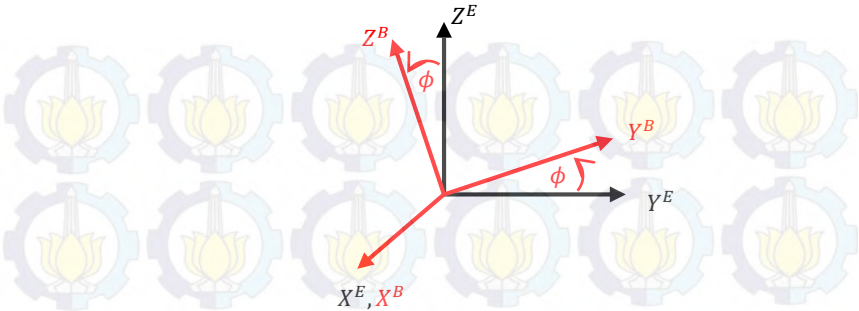
A.1.3. Kinematika

Kinematika merupakan cabang ilmu mekanika yang membahas gerak sebuah benda dengan mengabaikan gaya penyebab gerakan. Pada *Quadrotor* analisa gerak kinematika meliputi gerak *roll*, *pitch*, dan *yaw* yang mengacu pada koordinat *B-frame*, sehingga untuk menentukan posisi linear terhadap bumi diperlukan sebuah matriks transformasi dari koordinat *B-frame* ke koordinat *E-frame*. Sehingga posisi linear maupun posisi *angular* dapat didefinisikan sebagai berikut:

$$\xi^E = [X \quad Y \quad Z]^T$$

$$\Theta^E = [\phi \quad \theta \quad \psi]^T$$

- a) Gerak *Roll*
Gerak *roll* adalah gerak *Quadrotor* yang berotasi pada sumbu x sebesar ϕ atau dilambangkan $R(\phi, x)$. Gerak tersebut dapat dilihat pada Gambar A.2.



Gambar A.2 Gerak *Roll*

Berdasarkan gerak *roll* pada *Quadrotor* didapatkan persamaan kinematika seperti pada (A.1).

$$\begin{aligned} X^E &= X^B \\ Y^E &= \cos \phi Y^B - \sin \phi Z^B \\ Z^E &= \sin \phi Y^B + \cos \phi Z^B \end{aligned} \quad (\text{A.1})$$

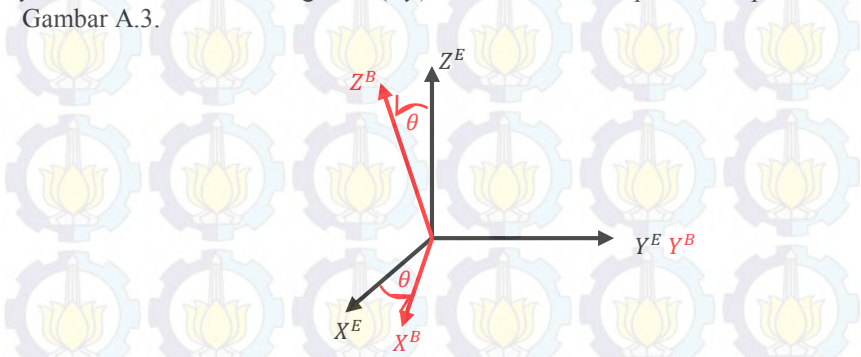
Persamaan (A.1) dapat diubah dalam bentuk matriks seperti (A.2) dan (A.3).

$$\begin{bmatrix} X^E \\ Y^E \\ Z^E \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} X^B \\ Y^B \\ Z^B \end{bmatrix} \quad (\text{A.2})$$

$$R^{BE}(\phi, x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}, \quad \phi = \left[-\frac{\pi}{2}, \frac{\pi}{2} \right] \quad (\text{A.3})$$

b) Gerak *Pitch*

Gerak *pitch* adalah gerak *Quadrotor* yang berotasi pada sumbu y sebesar θ atau dilambangkan $R(\theta, y)$. Gerak tersebut dapat dilihat pada Gambar A.3.



Gambar A.3 Gerak *Pitch*

Berdasarkan gerak *pitch* pada *Quadrotor* didapatkan persamaan kinematika seperti pada (A.4).

$$\begin{aligned} X^E &= \cos \theta X^B + \sin \theta Z^B \\ Y^E &= Y^B \\ Z^E &= -\sin \theta X^B + \cos \theta Z^B \end{aligned} \quad (\text{A.4})$$

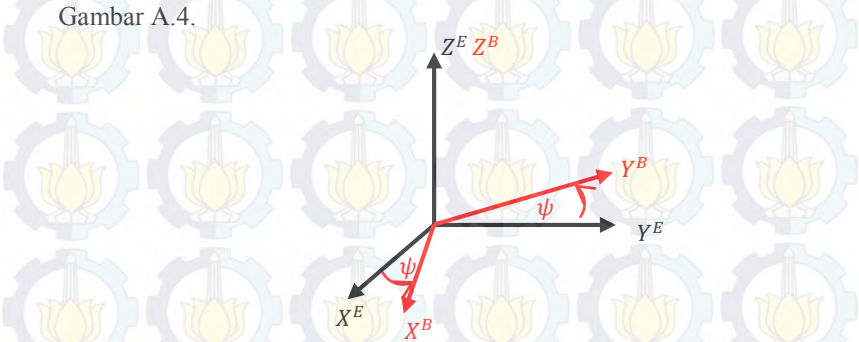
Persamaan (A.4) dapat diubah dalam bentuk matriks seperti persamaan (A.5) dan (A.6).

$$\begin{bmatrix} X^E \\ Y^E \\ Z^E \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} X^B \\ Y^B \\ Z^B \end{bmatrix} \quad (\text{A.5})$$

$$R^{BE}(\theta, y) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, \quad \theta = \left[-\frac{\pi}{2}, \frac{\pi}{2} \right] \quad (\text{A.6})$$

c) Gerak *Yaw*

Gerak *yaw* adalah gerak *Quadrotor* yang berotasi pada sumbu z sebesar ψ atau dilambangkan $R(\psi, z)$. Gerak tersebut dapat dilihat pada Gambar A.4.



Gambar A.4 Gerak *Yaw*

Berdasarkan gerak *yaw* pada *Quadrotor* didapatkan persamaan kinematika seperti pada (A.7).

$$\begin{aligned} X^E &= \cos \psi X^B - \sin \psi Y^B \\ Y^E &= \sin \psi X^B + \cos \psi Y^B \\ Z^E &= Z^B \end{aligned} \quad (\text{A.7})$$

Persamaan (A.5) dapat diubah dalam bentuk matriks seperti persamaan (A.8) dan (A.9).

$$\begin{bmatrix} X^E \\ Y^E \\ Z^E \end{bmatrix} = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X^B \\ Y^B \\ Z^B \end{bmatrix} \quad (\text{A.8})$$

$$R^{BE}(\psi, z) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \psi = \left[-\frac{\pi}{2}, \frac{\pi}{2} \right] \quad (\text{A.9})$$

Untuk mendapatkan matriks transformasi R_{ξ}^{BE} dari koordinat B -frame ke koordinat E -frame maka analisa kinematika dari gerak *roll*, *pitch*, *yaw* dapat digabung kedalam satu matriks. Matriks R_{ξ}^{BE} diperoleh dengan mengalikan matriks rotasi dari tiap gerak *Quadrotor* seperti pada persamaan berikut.

$$R_{\xi}^{BE} = R(\psi, z).R(\theta, y).R(\phi, x)$$

$$R_{\xi}^{BE} = \begin{bmatrix} c_{\psi}c_{\theta} & -s_{\psi}c_{\phi} + c_{\psi}s_{\theta}s_{\phi} & s_{\psi}s_{\phi} + c_{\psi}s_{\theta}c_{\phi} \\ s_{\psi}c_{\theta} & c_{\psi}c_{\phi} + s_{\psi}s_{\theta}s_{\phi} & -c_{\psi}s_{\phi} + s_{\psi}s_{\theta}c_{\phi} \\ -s_{\theta} & c_{\theta}s_{\phi} & c_{\theta}c_{\phi} \end{bmatrix} \quad (A.10)$$

Selain matriks tranformasi R_{ξ}^{BE} , ada pula matriks transformasi T_{Θ}^{BE} untuk mengubah besaran *angular* dari koordinat B -frame ke koordinat E -frame. Matriks transformasi T_{Θ}^{BE} mengacu kecepatan *Euler* dalam B -frame dengan membalik pola perputaran sudut *roll*, *pitch*, dan *yaw* peserti pada (A.11).

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = I \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R(\phi, x)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R(\phi, x)^{-1} R(\theta, y)^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (A.11)$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = T_{\Theta}^{BE-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (A.12)$$

Dengan menyelesaikan (A.12) maka didapatkan matriks transformasi T_{Θ}^{BE} .

$$T_{\Theta}^{BE-1} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix}$$

$$T_{\Theta}^{BE^{-1}} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix} \quad (\text{A.13})$$

A.1.4. Dinamika

Dinamika merupakan analisis gerak suatu benda dengan memperhatikan gaya yang berpengaruh pada benda tersebut. Analisis dinamika dilakukan dengan menggunakan hukum Newton *Euler* tentang gerak suatu benda. Gaya yang muncul akibat gerak dari *Quadrotor* dapat ditulis sebagai berikut:

$$\begin{bmatrix} F_{\text{trust}} \\ F_{\text{roll}} \\ F_{\text{pitch}} \\ F_{\text{yaw}} \end{bmatrix} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} F_1 + F_2 + F_3 + F_4 \\ F_2 - F_4 \\ F_1 - F_3 \\ F_1 - F_2 + F_3 - F_4 \end{bmatrix} \quad (\text{A.14})$$

dimana gaya F_i merupakan gaya angkat yang dihasilkan oleh tiap baling-baling motor yang didefinisikan sebagai berikut:

$$F_i = K \frac{\omega}{s + \omega} u_i \quad (\text{A.15})$$

dengan konstanta K merupakan gaya dorong, ω bandwidth motor dan u adalah sinyal kontrol dari kontroler. Untuk memperoleh konstanta K maka dicari dengan menerbangkan *Quadrotor* pada posisi hover. Nilai K akan sebanding dengan total gaya angkat dari keempat motor *Quadrotor* saat *hover*.

Pemodelan *Quadrotor* dilakukan dengan kombinasi koordinat *frame* atau disebut *Hibrid-Frame (H-frame)*. Penurunan model gerak translasi akan diturunkan terhadap koordinat bumi (*E-frame*) karena berkaitan dengan posisi dan kecepatan *Quadrotor* terhadap bumi. Sedangkan penurunan gerak rotasi diturunkan terhadap koordinat *body (B-frame)* karena gerak rotasi mempengaruhi gerakan *Quadrotor* itu sendiri.

A.1.4.1. Penurunan Gerak Translasi

Berdasarkan aksioma pertama *Euler* dari hukum II Newton didapatkan persamaan gerak translasi sebagai berikut:

$$\sum F^E = \ddot{\xi}^E m \quad (A.16)$$

dengan ΣF^E merupakan resultan gaya yang bekerja pada *Quadrotor*, m merupakan total massa *Quadrotor* dan ξ^E merupakan percepatan gerak *Quadrotor*. Sehingga (A.16) dapat dirubah menjadi (A.17)

$$F_f^E + F_g = \ddot{\xi} m \quad (A.17)$$

dengan $\xi = [x \ y \ z]^T$ merupakan posisi *Quadrotor* terhadap bumi, F_f^E merupakan resultan gaya pada tiap sumbu koordinat, dan $F_g = [0 \ 0 \ -mg]^T$ merupakan gaya gravitasi. Jika didefinisikan $F_f^B = [F_x \ F_y \ F_z]^T$ merupakan resultan gaya pada koordinat *B-frame*, maka harus ditransformasikan ke dalam koordinat bumi *E-frame*. Karena gaya yang bekerja hanya pada sumbu z maka resultan gaya yang muncul hanya F_z atau gaya thrust (U_1).

$$F_f^E = R_{\xi}^{BE} F_f^B$$

$$F_f^E = \begin{bmatrix} F_z (s_{\psi} s_{\phi} + c_{\psi} s_{\theta} c_{\phi}) \\ F_z (-c_{\psi} s_{\phi} + s_{\psi} s_{\theta} c_{\phi}) \\ F_z (c_{\theta} c_{\phi}) \end{bmatrix} = \begin{bmatrix} U_1 (s_{\psi} s_{\phi} + c_{\psi} s_{\theta} c_{\phi}) \\ U_1 (-c_{\psi} s_{\phi} + s_{\psi} s_{\theta} c_{\phi}) \\ U_1 (c_{\theta} c_{\phi}) \end{bmatrix}$$

(A.18)

Dengan memasukkan (A.18) ke (A.17) maka diperoleh

$$\begin{bmatrix} U_1 (s_{\psi} s_{\phi} + c_{\psi} s_{\theta} c_{\phi}) \\ U_1 (-c_{\psi} s_{\phi} + s_{\psi} s_{\theta} c_{\phi}) \\ U_1 (c_{\theta} c_{\phi}) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} = \begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} m \quad (A.19)$$

Setelah (A.19) diselesaikan maka diperoleh persamaan gerak translasi *Quadrotor* seperti pada (A.20).

$$\begin{cases} \ddot{X} = (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Y} = (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Z} = -g + (\cos \theta \cos \phi) \frac{U_1}{m} \end{cases} \quad (\text{A.20})$$

A.1.4.2. Penurunan Gerak Rotasi

Berdasarkan aksioma kedua *Euler* pada hukum II Newton didapatkan persamaan gerak rotasi sebagai berikut:

$$\tau^E = J\ddot{\Theta}^E \quad (\text{A.21})$$

Persamaan (A.21) merupakan persamaan di koordinat *E-frame*, sehingga perlu ditransformasi ke dalam koordinat *B-frame*. Jika persamaan tersebut dibawa ke dalam koordinat *B-frame* menjadi (A.22).

$$\begin{aligned} T_{\Theta} \tau^B &= J(T_{\Theta} \omega^B) \\ T_{\Theta} \tau^B &= J(\dot{T}_{\Theta} \omega^B + T_{\Theta} \dot{\omega}^B) \end{aligned} \quad (\text{A.22})$$

Jika turunan dari matriks transformasi T_{Θ} adalah $T_{\Theta} S(\omega^B)$, dengan $S(\omega^B)$ adalah matriks *skew-symmetric* [13], maka persamaan (A.22) dapat diubah menjadi (A.23).

$$\begin{aligned} T_{\Theta} \tau^B &= J(T_{\Theta} S(\omega^B) \omega^B + T_{\Theta} \dot{\omega}^B) \\ T_{\Theta} \tau^B &= T_{\Theta} (S(\omega^B) J \omega^B + J \dot{\omega}^B) \\ T_{\Theta} \tau^B &= T_{\Theta} (\omega^B \times J \omega^B + J \dot{\omega}^B) \end{aligned} \quad (\text{A.23})$$

Karena pada kedua ruas terdapat matriks transformasi T_{Θ} , maka dapat dihilangkan atau persamaan (A.24) dapat dianggap sudah ditransformasikan ke dalam koordinat *B-frame*.

$$\tau^B = \omega^B \times J \omega^B + J \dot{\omega}^B \quad (\text{A.24})$$

Persamaan (A.24) dapat disederhanakan menjadi (A.25).

$$J\dot{\omega}^B = -(\omega^B \times J\omega^B) + \tau^B \quad (\text{A.25})$$

dengan $J = \text{diag}[J_{xx} \ J_{yy} \ J_{zz}]$ merupakan matriks diagonal yang berisi momen inersia tiap sumbu, $\omega = [p \ q \ r]^T = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ merupakan kecepatan sudut *Quadrotor*, $\tau = [U_2l \ U_3l \ U_4d]^T$ torsi yang bekerja pada *Quadrotor*, l jarak motor terhadap pusat massa, dan d konstanta drag *Quadrotor*. Untuk mempermudah proses penghitungan maka diselesaikan dulu perkalian *cross product* didalam kurung.

$$\begin{aligned} -(\omega^B \times J\omega^B) &= J\omega^B \times \omega^B \\ &= \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} (J_{yy} - J_{zz})qr \\ (J_{zz} - J_{xx})pr \\ (J_{xx} - J_{yy})pq \end{bmatrix} \end{aligned} \quad (\text{A.26})$$

Dengan menyelesaikan (A.25) maka diperoleh persamaan gerak rotasi *Quadrotor* seperti pada (A.27)

$$\begin{aligned} J\dot{\omega}^B &= -(\omega^B \times J\omega^B) + \tau^B \\ \begin{bmatrix} J_{xx}\dot{p} \\ J_{yy}\dot{q} \\ J_{zz}\dot{r} \end{bmatrix} &= \begin{bmatrix} (J_{yy} - J_{zz})qr \\ (J_{zz} - J_{xx})pr \\ (J_{xx} - J_{yy})pq \end{bmatrix} + \begin{bmatrix} U_2l \\ U_3l \\ U_4d \end{bmatrix} \\ \begin{cases} \dot{p} = \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{U_2l}{J_{xx}} \\ \dot{q} = \frac{J_{zz} - J_{xx}}{J_{yy}} pr + \frac{U_3l}{J_{yy}} \\ \dot{r} = \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4d}{J_{zz}} \end{cases} & \end{aligned} \quad (\text{A.27})$$

Dengan mengambil persamaan gerak translasi (A.20) dan persamaan gerak rotasi (A.27), maka keseluruhan model dinamika *Quadrotor* dapat dinyatakan sebagai

$$\left\{ \begin{array}{l} \ddot{X} = (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Y} = (-\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) \frac{U_1}{m} \\ \ddot{Z} = -g + (\cos \theta \cos \phi) \frac{U_1}{m} \\ \dot{p} = \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{U_2 l}{J_{xx}} \\ \dot{q} = \frac{J_{zz} - J_{xx}}{J_{yy}} pr + \frac{U_3 l}{J_{yy}} \\ \dot{r} = \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4 d}{J_{zz}} \end{array} \right. \quad (\text{A.28})$$



LAMPIRAN B

B.1 Program Kontrol Stabilisasi *Quadrotor*

```
clear;clc;
Jxx=0.03; %kgm^2
Jyy=0.03; %kgm^2
Jzz=0.04; %kgm^2
m=3.499; %kg
g=9.81; %kgm^-2
l=0.2; %m
d=3.13e-5; %3.13e-5
b=7.5e-7;
Jr=6.5e-5;
w=15; %omegaha (rad/s)
k=120; %N

%-----Kontrol Stabilisasi-----%
%Linearisasi dititik 0
[A0,B0,C0,D0]=linmod2('model',[0 0 0 0 0 0 0 0 0
0 0 0],[0 0 0 0]);
%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem roll
ts1=1; %second
tao1=ts1/4; %menggunakan ts(2%)=4*tao
sp1=(-1/tao1);
sp2=3*sp1;
pole_roll=[sp1 sp2];

%Roll
Aroll=A0([7:8],[7:8]);
Broll=B0([7:8],2);
Kroll=place(Aroll,Broll,pole_roll);

%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem pitch
ts2=1; %second
```

```

tao2=ts2/4; %menggunakan ts(2%)=4*tao
sr1=(-1/tao2);
sr2=3*sr1;
pole_pitch=[sr1 sr2];

%Pitch
Apitch=A0([9:10],[9:10]);
Bpitch=B0([9:10],3);
Kpitch=place(Apitch,Bpitch,pole_pitch);

%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem yaw
ts3=1; %second
tao3=ts3/4; %menggunakan ts(2%)=4*tao
sy1=(-1/tao3);
sy2=3*sy1;
pole_yaw=[sy1 sy2];

%Yaw
Ayaw=A0([11:12],[11:12]);
Byaw=B0([11:12],4);
Kyaw=place(Ayaw,Byaw,pole_yaw);

%Stabilisasi Z
%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem Z
ts4=1; %second
tao4=ts4/4; %menggunakan ts(2%)=4*tao
sz1=(-1/tao4);
sz2=3*sz1;
pole_z=[sz1 sz2 sz2-1];

%Z
Az=[A0([5:6],[5:6]),[0 0]';[-1 0 0]];
Bz=[B0([5:6],1);0];
Kzz=place(Az,Bz,pole_z);
Kz=Kzz(1:2);Kiz=Kzz(3);

```

B.2 Program Kontrol *Tracking Quadrotor*

```
%-----Kontrol Fuzzy Tracking-----%
%linearisasi disudut roll dan pitch
%Di sumbu X
phi=0;theta=0;psi=0; %sekitar 0
A1=[0 1;0 0];
B1=[0;cos(phi)*cos(theta)*cos(psi)*9.81];

phi=0;theta=pi/9;psi=0; %theta sekitar 20°
A2=[0 1;0 0];
B2=[0;cos(phi)*cos(theta)*cos(psi)*9.81];

Arx=[0 1;-8.5 -7];
Brx=[0;8.5];
q=(10^-1)*diag([4.8 1.8]);
gamma=0.7;
Q=inv(q);
gam=1/(gamma^2);
gamk=gamma^2;

setlmis([]);
Y11=lmivar(1,[2 1]);
X1=lmivar(2,[1 2]);
X2=lmivar(2,[1 2]);

%lmiterm([term baris kolom var],operator)
lmiterm([1 1 1 Y11],1,A1','s')
%LMI term#1 : Y11*A11'+A11*Y11
lmiterm([1 1 1 X1],B1,1,'s')
%LMI term#1 : B11*X1+(B11*X1)'
lmiterm([1 1 1 0],gam)
%LMI term#1 : 1/gamma^2
lmiterm([1 2 1 Y11],1,1)
%LMI term#1 : Y11
lmiterm([1 2 2 0],-Q)
%LMI term#1 : -inv(Q)
```

```

lmiterm([2 1 1 Y11],1,A2','s')
%LMI term#2 : Y11*A22'+A22*Y11
lmiterm([2 1 1 X2],B2,1,'s')
%LMI term#2 : B22*X2+(B22*X2)'
lmiterm([2 1 1 0],gam)
%LMI term#2 : 1/gamma^2
lmiterm([2 2 1 Y11],1,1)
%LMI term#2 : Y11
lmiterm([2 2 2 0],-Q)
%LMI term#2 : -inv(Q)

lmiterm([-3 1 1 Y11],1,1)
%LMI term#3 : Syarat P11>0

lmis=getlmis;
[tmin,xfeas]=feasp(lmis);
Y11=dec2mat(lmis,xfeas,Y11);
X1=dec2mat(lmis,xfeas,X1);
X2=dec2mat(lmis,xfeas,X2);

evlmi = evallmi(lmis,xfeas);
[lhs1,rhs1]=showlmi(evlmi,1);

Px1=inv(Y11);
Kx1=X1*Px1
Kx2=X2*Px1

setlmis([]);
Px2=lmivar(1,[2 1]);

lmiterm([1 1 1 0],(A1+B1*Kx1) '*Px1
+Px1*(A1+B1*Kx1)+gam*Px1*Px1+q)';
%LMI term#1:
(A1+B1*K1) '*P11+P11*(A1+B1*K1)+g*P11*P11+Q
lmiterm([1 2 1 0],(Px1*B1*Kx1-q)');
%LMI term#1: (-P11*B1*K1-Q)'
lmiterm([1 2 2 Px2],Arx',1,'s');
%LMI term#1: Ar' *P22+P22*Ar

```



```

lmiterm([1 2 2 0],q);
lmiterm([1 3 2 Px2],Brx',1);
lmiterm([1 3 3 0],-gamk);

lmiterm([2 1 1 0],(A2+B2*Kx2) '*Px1
+Px1*(A2+B2*Kx2)+gam*Px1*Px1+q);
%LMI term#1:
(A1+B1*K1) '*P11+P11*(A1+B1*K1)+g*P11*P11+Q
lmiterm([2 2 1 0],(Px1*B2*Kx2-q) ');
%LMI term#1: (-P11*B1*K1-Q) '
lmiterm([2 2 2 Px2],Arx',1,'s');
%LMI term#1: Ar' *P22+P22*Ar
lmiterm([2 2 2 0],q);
lmiterm([2 3 2 Px2],Brx',1);
lmiterm([2 3 3 0],-gamk);

lmiterm([-3 1 1 Px2],1,1);
%LMI term#3 : Syarat P22>0

lmis=getlmis;
[tmin,xfes]=feasp(lmis);
Px2=dec2mat(lmis,xfes,Px2);

evlmi = evallmi(lmis,xfes);
[lhs1,rhs1]=showlmi(evlmi,1);

Asys1=[A1+B1*Kx1 -B1*Kx1;zeros(2,2) Arx];
Asys2=[A2+B2*Kx2 -B2*Kx2;zeros(2,2) Arx];
eigenvalue1=eig(Asys1)
eigenvalue2=eig(Asys2)

%-----Kontrol Fuzzy Tracking-----%
%linearisasi disudut roll dan pitch
%Di sumbu Y
phi=0;theta=0;psi=0; %sekitar 0
A3=[0 1;0 0]; B3=[0;-cos(phi)*cos(psi)*9.81];

```

```

phi=pi/9;theta=0;psi=0;           %theta sekitar 20°
A4=[0 1;0 0]; B4=[0;-cos(phi)*cos(psi)*9.81];

Ary=[0 1;-8.5 -7];
Bry=[0;8.5];
q=(10^-1)*diag([4.8 1.8]);
gamma=0.7;
Q=inv(q);
gam=1/(gamma^2);
gamk=gamma^2;

setlmis([]);
Y22=lmivar(1,[2 1]);
X11=lmivar(2,[1 2]);
X22=lmivar(2,[1 2]);

%lmiterm([term baris kolom var],operator)
lmiterm([1 1 1 Y22],1,A3','s');
%LMI term#1 : Y11*A11'+A11*Y11
lmiterm([1 1 1 X11],B3,1,'s');
%LMI term#1 : B11*X1+(B11*X1)'
lmiterm([1 1 1 0],gam);
%LMI term#1 : 1/gamma^2
lmiterm([1 2 1 Y22],1,1);
%LMI term#1 : Y11
lmiterm([1 2 2 0],-Q);
%LMI term#1 : -inv(Q)

lmiterm([2 1 1 Y22],1,A4','s');
%LMI term#1 : Y11*A22'+A22*Y11
lmiterm([2 1 1 X22],B4,1,'s');
%LMI term#1 : B22*X2+(B22*X2)'
lmiterm([2 1 1 0],gam);
%LMI term#1 : 1/gamma^2
lmiterm([2 2 1 Y22],1,1);
%LMI term#1 : Y11
lmiterm([2 2 2 0],-Q);
%LMI term#1 : -inv(Q)

```

```

lmiterm([-3 1 1 Y22],1,1);
%LMI term#3 : Syarat P11>0

lmis=getlmis;
[tmin,xffeas]=feasp(lmis);
Y22=dec2mat(lmis,xffeas,Y22);
X11=dec2mat(lmis,xffeas,X11);
X22=dec2mat(lmis,xffeas,X22);

evlmi = evallmi(lmis,xffeas);
[lhs1,rhs1]=showlmi(evlmi,1);

Py1=inv(Y22);
Ky1=X11*Py1
Ky2=X22*Py1

setlmis([]);
Py2=lmivar(1,[2 1]);

lmiterm([1 1 1 0],(A3+B3*Ky1)'*Py1
+Py1*(A3+B3*Ky1)+gam*Py1*Py1+q);
%LMI term#1:
(A1+B1*K1)'*P11+P11*(A1+B1*K1)+g*P11*P11+Q
lmiterm([1 2 1 0],(Py1*B3*Ky1-q)');
%LMI term#1: (-P11*B1*K1-Q)'
lmiterm([1 2 2 Py2],Ary',1,'s');
%LMI term#1: Ar'*P22+P22*Ar
lmiterm([1 2 2 0],q);
lmiterm([1 3 2 Py2],Bry',1);
lmiterm([1 3 3 0],-gamk);

lmiterm([2 1 1 0],(A4+B4*Ky2)'*Py1
+Py1*(A4+B4*Ky2)+gam*Py1*Py1+q);
%LMI term#2:
(A1+B1*K1)'*P11+P11*(A1+B1*K1)+g*P11*P11+Q
lmiterm([2 2 1 0],(Py1*B4*Ky2-q)');
%LMI term#2: (-P11*B1*K1-Q)'

```

```

lmiterm([2 2 2 Py2],Ary',1,'s');
%LMI term#2: Ar'*P22+P22*Ar
lmiterm([2 2 2 0],q);
lmiterm([2 3 2 Py2],Bry',1);
lmiterm([2 3 3 0],[-gamk);

lmiterm([-3 1 1 Py2],1,1);
%LMI term#3 : Syarat P11>0

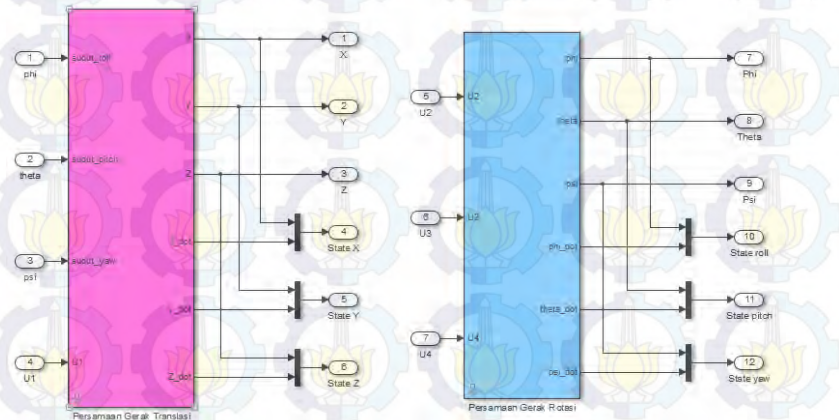
lmis=getlmis;
[tmin,xfeas]=feasp(lmis);
Py2=dec2mat(lmis,xfeas,Py2);

evlmi = evallmi(lmis,xfeas);
[lhs1,rhs1]=showlmi(evlmi,1);

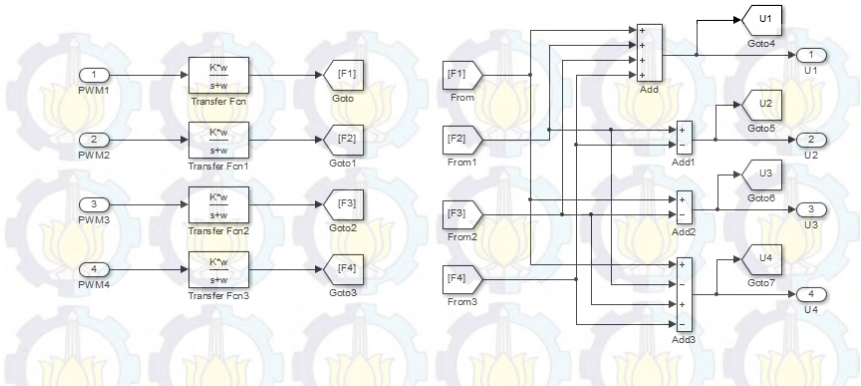
Asys3=[A3+B3*Ky1 -B3*Ky1;zeros(2,2) Ary];
Asys4=[A4+B4*Ky2 -B4*Ky2;zeros(2,2) Ary];
eigenvalue1=eig(Asys3)
eigenvalue2=eig(Asys4)

```

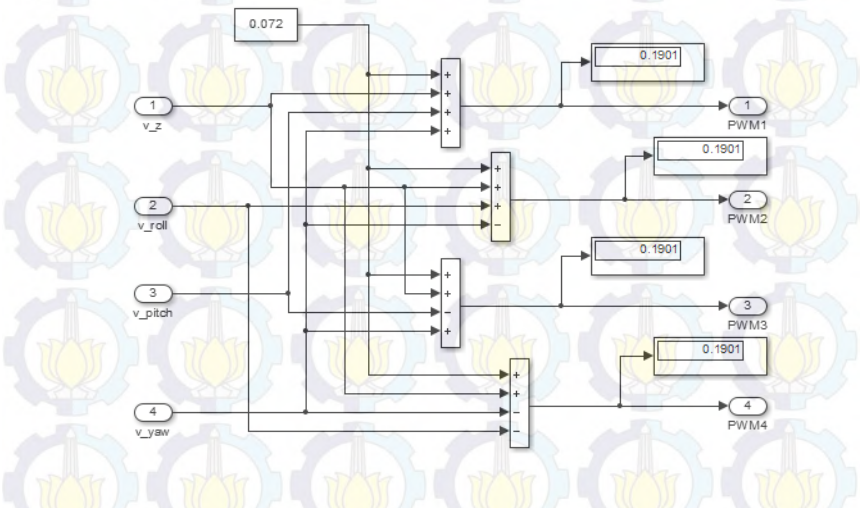
B.3 Blok Diagram *Simulink* Blok Dinamika Gerak *Quadrotor*



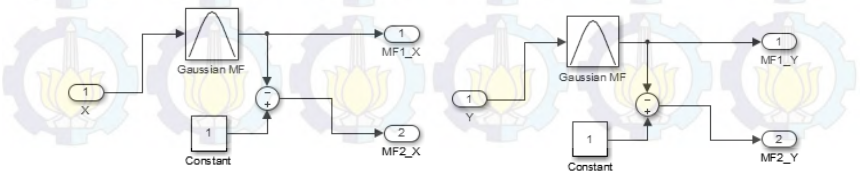
Blok Aktuator



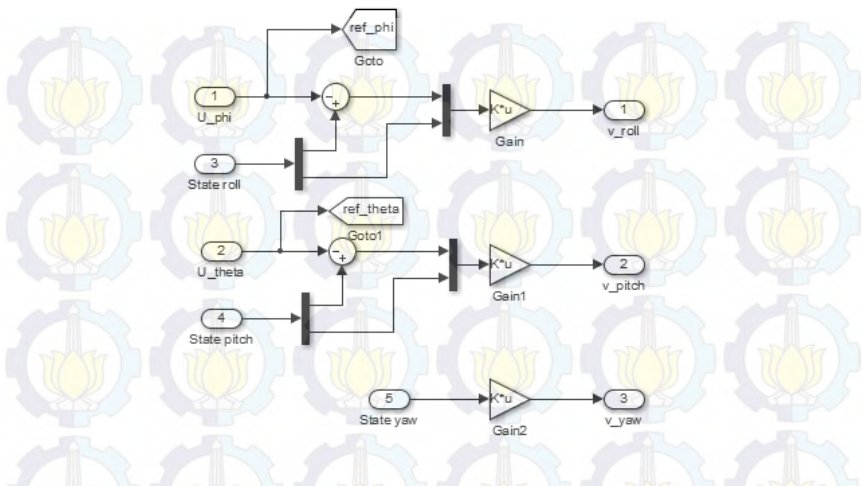
Blok PWM



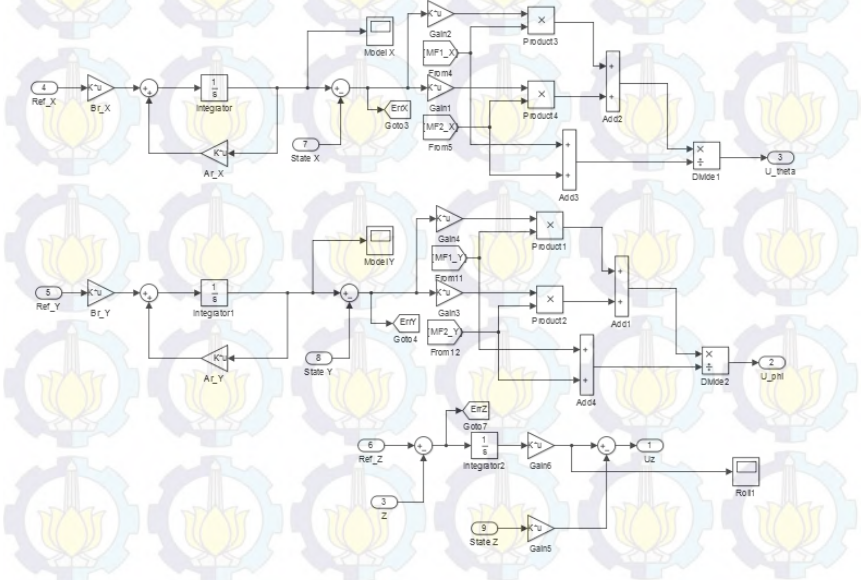
Blok Membership Function X dan Y



Blok Kontrol Stabilisasi



Blok Kontrol Tracking



RIWAYAT HIDUP



Dinang Sohendri lahir di Trenggalek pada tanggal 18 September 1993. Penulis telah menyelesaikan pendidikannya di TK Dharmawanita Parakan Trenggalek, SDN 2 Parakan Trenggalek, MTsN Model Trenggalek, dan SMAN 1 Trenggalek. Pada tahun 2012, penulis melanjutkan studi di Institut Teknologi Sepuluh Nopember (ITS), Jurusan Teknik Elektro melalui jalur Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN) jalur Ujian Tulis. Di Jurusan

Teknik Elektro ini, penulis mengambil Teknik Sistem Pengaturan sebagai bidang studi yang ditekuni. Pada bulan Juni 2016, penulis mengikuti seminar dan ujian Tugas Akhir sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro.

e-mail : dinangs61@gmail.com