



TUGAS AKHIR - TE 091599

**KONTROL *FUZZY* OPTIMAL UNTUK *TRAJECTORY*
TRACKING PADA *QUADROTOR***

**IMROATUL HUDATI
NRP 2215105083**

**Dosen Pembimbing
Dr. Trihastuti Agustinah S.T., M.T.**

**DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2017**



FINAL PROJECT - TE 091599

***OPTIMAL FUZZY CONTROL FOR TRAJECTORY
TRACKING OF QUADROTOR***

**IMROATUL HUDATI
NRP 2215105083**

Supervisor
Dr. Trihastuti Agustinah S.T., M.T.

***ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2017***

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul "**Kontrol Fuzzy Optimal untuk Trajectory Tracking pada Quadrotor**" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya mandiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi peraturan yang berlaku.

Surabaya, Juli 2017



Introatul Hudati
NRP 2215105083

**KONTROL FUZZY OPTIMAL UNTUK TRAJECTORY
TRACKING PADA QUADROTOR**

TUGAS AKHIR

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

Menyetujui :

Dosen Pembimbing I



Dr. Tri Hastuti Agustinah, S.T., M.T.

NIP. 196808121994032001



KONTROL FUZZY OPTIMAL UNTUK TRAJECTORY TRACKING QUADROTOR

Nama : Imroatul Hudati

Dosen Pembimbing : Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRAK

Quadrotor merupakan salah satu jenis *Unmanned Aerial Vehicle (UAV)* atau pesawat tanpa awak yang memiliki empat buah motor *propeller* sebagai penggerakannya. Dalam pemanfaatan *Quadrotor* digunakan untuk berbagai tujuan, kestabilan pada *Quadrotor* untuk dapat menuju titik referensi sangatlah penting dan harus dimiliki *Quadrotor* agar pemanfaatannya dapat maksimal.

Oleh sebab itu, masalah pertama pada *Quadrotor* yaitu kestabilan sudut *roll*, *pitch* dan *yaw* pada saat terbang dan permasalahan kedua adalah *tracking Quadrotor* mengikuti sinyal referensi. Pada Tugas Akhir ini digunakan *Fuzzy Takagi-Sugeno* untuk merepresentasikan nonlinearitas sistem dari *Quadrotor*. Struktur *tracking* menggunakan metode optimal. Metode optimal secara umum ditujukan untuk memilih *input plant u* dengan indeks performansi yang minimum.

Hasil pengujian kontroler menunjukkan bahwa sistem hasil desain dapat mengatur gerak *Quadrotor* sesuai dengan lintasannya. Nilai RMSE pada sumbu X adalah sebesar 0,1470 sedangkan pada sumbu Y sebesar 0,1225.

Kata Kunci: *Quadrotor*, *Telusur trajectory*, *Fuzzy Takagi Sugeno*, *Linear Quadratic Optimal*

Halaman ini sengaja dikosongkan

OPTIMAL FUZZY CONTROL FOR TRAJECTORY TRACKING OF QUADROTOR

Name : Imroatul Hudati

Supervisor : Dr. Trihastuti Agustinah ST.,MT.

ABSTRACT

Quadrotor is one type of Unmanned Aerial Vehicle (UAV) or unmanned aircraft that has four motor propellers as the propulsion. In the use of Quadrotor used for various purposes, the stability of the Quadrotor to get to the reference point is very important and must be owned Quadrotor for maximum utilization.

Therefore, the first problem of the Quadrotor is the stability of the roll, pitch and yaw angles during flight and the second problem is the Quadrotor tracking following the reference signal. In this Final Project used Fuzzy Takagi Sugeno to represent nonlinearitas system from Quadrotor. Tracking structure use optimal method. The optimum method is generally intended to select the plant input u with a minimum performance index.

The result of the controller test shows that the system of rural results can adjust the Quadrotor motion according to its path. The value of RMSE on the X axis is 0.1470 and on the Y axis is 0.1225.

Keywords: Quadrotor, Trajectory Tracking, Fuzzy Takagi Sugeno, Linear Quadratic Optimal

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji Syukur kehadirat Allah SWT yang senantiasa memberikan ridho dan rahmat-Nya. Tak lupa Shalawat serta Salam penulis haturkan kepada junjungan kita Rosulullah Muhammad SAW sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan tepat waktu. Tugas Akhir dengan judul “**Kontrol Fuzzy Optimal untuk Trajectory Tracking pada Quadrotor**”.

Banyak pihak telah berperan serta membantu selama proses studi dan penulisan buku Tugas Akhir ini. Oleh Karena itu, penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Keluarga penulis yang telah senantiasa menasehati, mendukung dan mendo'akan atas keberhasilan Tugas Akhir ini.
2. Ibu Dr. Trihastuti Agustinah, S.T., M.T. selaku pembimbing yang telah dengan arif dan sabar dalam mengarahkan dan membimbing selama proses pengerjaan Tugas Akhir ini.
3. Mb Farida, mb Chur, mas Angga, dan Dinang atas bantuannya yang tidak terhingga.
4. Mb Kiki dan mb Devina yang sudah menjadi rekan kerja dalam bimbingan dan pengerjaan tugas akhir.
5. Semua teman-teman laboratorium B105, AJ104 dan Lintas Jalur 2015 yang tidak dapat disebutkan satu persatu yang telah banyak memberi dukungan dan nasehat.
6. Semua pihak yang telah membantu kami dalam penyelesaian Tugas Akhir ini yang tidak dapat kami sebutkan satu persatu.

Penulis berharap Tugas Akhir ini dapat bermanfaat bagi pembaca pada umumnya dan rekan-rekan mahasiswa Jurusan Teknik Elektro ITS pada khususnya. Saran dan kritik membangun selalu penulis harapkan untuk kelengkapan pengetahuan penulis terkait Tugas Akhir ini.

Surabaya, Juni 2017

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

| | |
|--|------|
| HALAMAN JUDUL | i |
| PERNYATAAN KEASLIAN | iii |
| LEMBAR PENGESAHAN | v |
| ABSTRAK | vii |
| ABSTRACT | ix |
| KATA PENGANTAR | xi |
| DAFTAR ISI | xiii |
| DAFTAR GAMBAR | xv |
| DAFTAR TABEL | xvii |
| BAB 1 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 2 DASAR TEORI | 5 |
| 2.1 Tinjauan Pustaka | 5 |
| 2.2 <i>Quadrotor</i> | 5 |
| 2.3 Pemodelan <i>Quadrotor</i> | 9 |
| 2.4 Parameter <i>Quadrotor</i> | 10 |
| 2.5 Linearisasi | 10 |
| 2.6 Teori <i>Fuzzy</i> | 13 |
| 2.6.1 Himpunan <i>Fuzzy</i> | 13 |
| 2.6.2 Fungsi Keanggotaan | 13 |
| 2.6.3 Operasi Himpunan <i>Fuzzy</i> | 15 |
| 2.6.4 Sistem Interferensi <i>Fuzzy</i> | 16 |
| 2.5.5 <i>Fuzzy</i> Takagi-Sugeno | 18 |
| 2.7 <i>Linear Quadratic Tracking</i> (LQT) | 19 |
| BAB 3 PERANCANGAN SISTEM | 23 |
| 3.1 Model Dinamika <i>Quadrotor</i> | 23 |
| 3.1.1 Persamaan <i>State</i> Dinamika <i>Quadrotor</i> | 23 |
| 3.2 Perancangan Kontroler | 25 |
| 3.2.1 Stabilisasi <i>Quadrotor</i> | 26 |
| 3.2.2 <i>Altitude Z</i> | 29 |
| 3.2.3 <i>Tracking Quadrotor</i> | 30 |

| | |
|--|-----------|
| 3.2.4 Perancangan Sinyal Kontrol | 41 |
| 3.3 Perancangan Perangkat Simulasi | 42 |
| BAB 4 HASIL DAN ANALISA | 45 |
| 4.1 Simulasi Stabilisasi <i>Quadrotor</i> | 45 |
| 4.2 Pengujian <i>Tracking Quadrotor</i> | 47 |
| 4.2.1 Pengujian <i>Altitude Quadrotor</i> | 47 |
| 4.2.2 Uji Gerak Sumbu X,Y <i>Quadrotor</i> | 48 |
| 4.2.3 Simulasi Gerak <i>Quadrotor</i> dengan Gangguan | 54 |
| 4.2.4 Simulasi Gerak <i>Quadrotor</i> dengan Gangguan Pengukuran | 56 |
| 4.3 Simulasi Sinyal Referensi yang Berbeda-beda | 59 |
| BAB 5 PENUTUP..... | 61 |
| DAFTAR PUSTAKA | 63 |
| LAMPIRAN | 65 |
| A.1 Pemodelan <i>Quadrotor</i> | 65 |
| A.1.1 Koordinat <i>Frame</i> | 65 |
| A.1.2 Asumsi Pemodelan <i>Quadrotor</i> | 66 |
| A.1.3 Kinematika | 66 |
| A.1.3 Dinamika | 70 |
| A.2 Linearisasi | 74 |
| B.1 Program Kontrol Stabilisasi <i>Quadrotor</i> | 78 |
| B.2 Program Kontrol <i>Tracking Quadrotor</i> | 80 |
| B.3 Blok Diagram <i>Simulink</i> | 81 |
| BIOGRAFI PENULIS | 85 |

DAFTAR GAMBAR

| | | |
|---------------------|---|----|
| Gambar 2. 1 | Arah Putaran <i>Propeller</i> | 6 |
| Gambar 2. 2 | Gerak <i>Thrust</i> | 6 |
| Gambar 2. 3 | Gerak <i>Roll</i> pada <i>Quadrotor</i> | 7 |
| Gambar 2. 4 | Gerak <i>Pitch</i> | 8 |
| Gambar 2. 5 | Gerak <i>Yaw</i> | 8 |
| Gambar 2. 6 | <i>Union</i> | 15 |
| Gambar 2. 7 | <i>Intersection</i> | 16 |
| Gambar 2. 8 | Sistem Kontrol <i>Fuzzy</i> | 17 |
| Gambar 2. 9 | Aturan <i>Plant</i> dan Kontroler <i>Fuzzy T-S</i> | 19 |
| Gambar 2. 10 | Blok Diagram LQT | 21 |
| Gambar 3. 1 | Perancangan Blok Kontroler | 25 |
| Gambar 3. 2 | Blok Diagram Kontrol Rotasi | 27 |
| Gambar 3. 3 | Blok Diagram Kontrol <i>Tracking</i> | 30 |
| Gambar 3. 4 | Himpunan Keanggotaan <i>Fuzzy</i> | 33 |
| Gambar 3. 5 | Model <i>Following</i> | 34 |
| Gambar 3. 6 | Kontrol <i>Fuzzy Optimal</i> | 39 |
| Gambar 3. 7 | Diagram Simulink..... | 43 |
| Gambar 4. 1 | Respon Sudut ϕ | 46 |
| Gambar 4. 2 | Respon Sudut θ | 46 |
| Gambar 4. 3 | Respon Sudut ψ | 47 |
| Gambar 4. 4 | Posisi Z <i>Quadrotor</i> | 48 |
| Gambar 4. 5 | <i>Tracking</i> Sumbu X dan Y <i>Quadrotor</i> | 49 |
| Gambar 4. 6 | Referensi <i>Tracking</i> untuk <i>Quadrotor</i> | 49 |
| Gambar 4. 7 | Respon <i>Tracking</i> Posisi Sumbu X | 50 |
| Gambar 4. 8 | Respon <i>Tracking</i> Posisi Sumbu Y | 50 |
| Gambar 4. 9 | Respon <i>Tracking</i> Posisi Sumbu Z..... | 51 |
| Gambar 4. 10 | Sudut <i>Roll</i> dan <i>Pitch</i> untuk Gerak <i>Quadrotor</i> | 51 |
| Gambar 4. 11 | Plot 3D Hasil Simulasi | 52 |
| Gambar 4. 12 | <i>Screenshoot</i> dari <i>Virtual World Quadrotor</i> | 52 |
| Gambar 4. 13 | Respon Sudut <i>Roll</i> Tanpa Gangguan | 53 |
| Gambar 4. 14 | Respon Sudut <i>Pitch</i> Tanpa Gangguan | 53 |
| Gambar 4. 15 | <i>Disturbance</i> | 54 |
| Gambar 4. 16 | Respon Sumbu X dengan Gangguan Sinyal <i>Step</i> | 55 |
| Gambar 4. 17 | Respon Sumbu Y dengan Gangguan Sinyal <i>Step</i> | 56 |

| | |
|--|----|
| Gambar 4. 18 <i>Tracking</i> Posisi dengan Lintasan Berbentuk Persegi | 59 |
| Gambar 4. 19 Posisi X dan Y pada saat <i>Tracking</i> Lintasan Berbentuk Persegi..... | 60 |

DAFTAR TABEL

| | |
|--|----|
| Tabel 3. 1 Daerah Linearisasi ϕ dan θ | 30 |
| Tabel 4. 1 <i>Error</i> pada Sumbu X dan Y Tanpa Gangguan | 52 |
| Tabel 4. 2 <i>Error</i> Sumbu X dan Y dengan <i>Disturbance</i> | 56 |
| Tabel 4. 3 <i>Error</i> Sumbu X dan Y dengan <i>Noise</i> | 59 |

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Quadrotor merupakan salah satu jenis pesawat tanpa awak yang biasa disebut sebagai *Unmanned Aerial Vehicle* (UAV) yang memiliki empat buah *propeller* dengan konfigurasi *plus* (+). Biasanya digunakan untuk operasi penyelamatan, pengamatan atau pengenalan objek. Dimana dua *propeller* bergerak searah jarum jam dan dua lainnya bergerak berlawanan arah jarum jam. Hal tersebut digunakan agar *Quadrotor* memiliki keseimbangan yang baik pada saat terbang.

Quadrotor memiliki dua macam gerakan, yaitu gerak rotasi dan gerak translasi. Gerak rotasi merupakan gerakan yang terdiri dari gerak *roll*, *pitch*, dan *yaw*. Sedangkan gerak translasi merupakan gerakan yang dihasilkan akibat adanya gerak rotasi. Gerakan ini dibagi menjadi dua, yaitu gerak lateral dan gerak longitudinal. Gerak longitudinal meliputi tiga fase utama, yaitu *take-off* (tinggal landas), *hovering* (melayang), dan *landing* (pendaratan), sedangkan gerak lateral mencakup gerakan dari satu titik ke titik lain secara horisontal (*waypoint*).

Quadrotor memiliki dua permasalahan yang harus dijaga agar 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.

Terdapat beberapa penelitian mengenai *tracking Quadrotor* seperti pada [1] menggunakan kontroler *Fuzzy T-S* dengan menggunakan pendekatan *Linear Matrix Inequality* (LMI) agar *Quadrotor* dapat melakukan *tracking* sesuai dengan sinyal referensi, namun terdapat penyimpangan lintasan yang cukup besar. Penelitian Penilitian selanjutnya pada [2] digunakan kontroler PID untuk mengatur ketinggian dan sudut dari *Quadrotor*, selanjutnya *gain* PID akan dimasukkan pada *Fuzzy logic* agar *Quadrotor* dapat terbang sesuai dengan posisi yang diinginkan.

Pada Tugas Akhir ini digunakan kontroler *Fuzzy T-S* dimana setiap nilai *gain* yang digunakan dicari dengan menggunakan kontrol *Linear Quadratic Tracking* (LQT) untuk mengatur *tracking Quadrotor*.

Diharapkan dengan menggunakan metode tersebut, maka *Quadrotor* dapat terbang sesuai dengan sinyal yang diberikan.

1.2 Perumusan Masalah

Permasalahan yang diatur dalam tugas akhir ini adalah sebagai berikut:

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

1.3 Batasan Masalah

Adapun batasan masalah dari pembuatan tugas ini adalah sebagai berikut:

1. Model *Quadrotor* yang digunakan adalah model *Quadrotor* Quanser Q-Ball X-4
2. Simulasi penerbangan *Quadrotor* diasumsikan terbang didalam ruangan
3. Titik origin *frame* inersia bumi dan *frame* badan *Quadrotor* berhimpitan
4. Struktur kontrol *tracking* menggunakan model *following* berbasis *Fuzzy T-S*
5. Kontrol *tracking* diberikan untuk posisi *Quadrotor* pada keadaan terbang dengan spesifikasi sinyal referensi yang sudah ditentukan

1.4 Tujuan Penelitian

Tujuan penelitian adalah merancang sistem kontrol *tracking Fuzzy* untuk *tracking* sehingga hasilnya diharapkan *Quadrotor* mampu bergerak sesuai lintasan yang diinginkan dengan meminimumkan *error tracking*.

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, metode kontrol optimal *Linear Quadratic Tracking (LQT)*.

BAB 3 : Perancangan Sistem

Bab ini membahas perancangan kontroler untuk stabilisasi sudut *roll*, *pitch* dan *yaw* dan perancangan kontroler *Fuzzy optimal* 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 dari tugas akhir ini diharapkan dapat menjadi referensi dan pengembangan pada desain kontrol *tracking* pada *Quadrotor* dengan menggunakan *Fuzzy T-S*.

BAB 2 TEORI PENUNJANG

Pada bab ini akan dibahas mengenai teori penunjang yang berkaitan dengan topik yang diambil dalam merumuskan dan menyelesaikan masalah. Teori penunjang yang diambil terdiri dari pengetahuan mengenai *Quadrotor*. Teori selanjutnya yaitu membahas mengenai linearisasi, model *Fuzzy Takagi-Sugeno*, dan *Linear Quadratic Tracking*.

2.1. Tinjauan Pustaka

Penelitian mengenai *Quadrotor* telah banyak dilakukan seiring berkembangnya metode yang digunakan untuk menangani permasalahan kontrol pada *Quadrotor*. Beberapa permasalahan yang diambil adalah seperti pengaturan kestabilan *Quadrotor* pada saat *hover* hingga melakukan pendaratan, *tracking*, hingga misi *tracking target*.

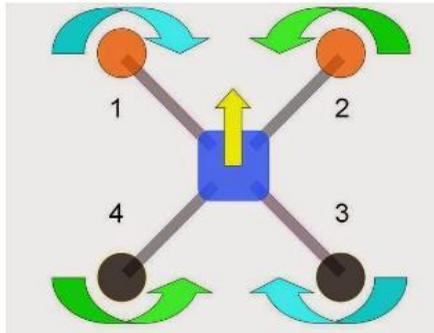
Terdapat beberapa penelitian mengenai *tracking Quadrotor* seperti pada [1] menggunakan kontroler *Fuzzy T-S* dengan menggunakan pendekatan *Linear Matrix Inequality (LMI)* agar *Quadrotor* dapat melakukan *tracking* sesuai dengan sinyal referensi, namun terdapat penyimpangan lintasan yang cukup besar. Penelitian Penilitian selanjutnya pada [2] digunakan kontroler *PID* untuk mengatur ketinggian dan sudut dari *Quadrotor*, selanjutnya *gain PID* akan dimasukkan pada *Fuzzy logic* agar *Quadrotor* dapat terbang sesuai dengan posisi yang diinginkan.

2.2. *Quadrotor*

Quadrotor adalah salah satu jenis dari *Unmanned Aerial Vehicle (UAV)* yang memiliki empat rotor yang dikonfigurasi dalam bentuk *frame plus (+)*. *Quadrotors* pada dasarnya terdiri dari perangkat pengontrol penerbangan, *casing*, sebuah kontrol kecepatan elektronik, baling-baling dan bingkai pelindung.

Komponen yang bergerak pada *Quadrotor* hanyalah *propeller*. Setiap *propeller* ini digerakkan dengan menggunakan sebuah motor elektrik sehingga terdapat empat motor untuk menggerakkan *Quadrotor*. Gerakan pada *propeller* yaitu terdapat dua buah *propeller* yang bergerak searah jarum jam dan dua *propeller* yang lain bergerak berlawanan arah jarum jam. Pada Gambar 2.1 dapat dilihat, *propeller* 1 dan 3 bergerak

searah jarum jam. Sedangkan *propeller* 2 dan 4 bergerak berlawanan arah jarum jam.

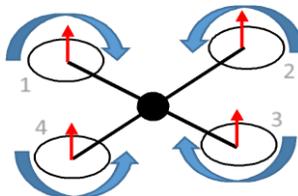


Gambar 2. 1 Arah Putaran *Propeller* [3]

Gerakan pada *propeller* tersebut akan menyebabkan quadrotor dapat bergerak *thrust*, *roll*, *pitch*, dan *yaw* dikarenakan terdapat perubahan torsi gaya pada masing-masing *propeller*. Berikut akan dijelaskan mengenai masing-masing gerakan pada *Quadrotor*.

a. *Thrust* (U_1)

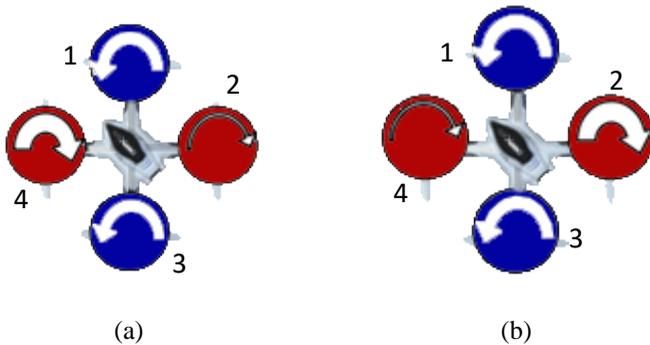
Thrust adalah gerak pada *Quadrotor* untuk melayang. Gaya *thrust* ini didapatkan apabila kecepatan motor pada keempat *propeller* sama. Untuk mengubah ketinggian dari *Quadrotor* yaitu dengan memperlambat atau mempercepat kecepatan keempat motor secara bersamaan. Apabila kecepatan keempat *propeller* dipercepat, maka *Quadrotor* akan bergerak naik. Sedangkan apabila kecepatan keempat *propeller* diperlambat, maka *Quadrotor* akan bergerak turun. Pergerakan *Quadrotor* pada keadaan hover arah panah warna biru menunjukkan jika setiap motor pada *Quadrotor* berputar dengan kecepatan yang sama.



Gambar 2. 2 Gerak *Thrust* [3]

b. *Roll* (U_2)

Roll adalah gerak rotasi *Quadrotor* terhadap sumbu X. Gerak *roll* dapat dilakukan dengan mempercepat atau memperlambat *propeller* 4 dan dalam waktu yang bersamaan, *propeller* 2 diperlambat atau dipercepat. Dan untuk *propeller* 1 dan 3 masih tetap memiliki kecepatan yang sama. Pada Gambar 2.3 (a) ditampilkan gerak *roll* dari *Quadrotor*. Dimana pada *propeller* warna biru menunjukkan kecepatan nominal yang sama antara *propeller* 1 dan 3. Sedangkan *propeller* 4 lebih cepat dibandingkan *propeller* 2 sehingga mengakibatkan *Quadrotor* berputar searah dengan jarum jam. Sedangkan pada Gambar 2.3 (b) menampilkan gerak *roll* pada *Quadrotor* yang berlawanan dengan arah jarum jam.

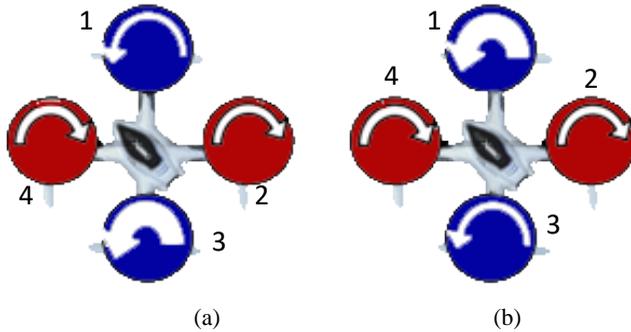


Gambar 2. 3 Gerak *Roll* Pada *Quadrotor* [3]

c. *Pitch* (U_3)

Pitch adalah gerak rotasi pada sumbu Y. Gerak *pitch* didapatkan dengan mempercepat atau memperlambat *propeller* 3 dan secara bersamaan memperlambat atau mempercepat *propeller* 1. Dan kecepatan pada *propeller* 2 dan 4 tetap sama. Pada Gambar 2.4 (a) menunjukkan *Quadrotor* bergerak ke arah *propeller* 3. Dimana *propeller* 4 dan 2 memiliki kecepatan nominal. Sedangkan *propeller* 1 lebih lambat dibandingkan *propeller* 2 dan 4. Dan untuk *propeller* 4 lebih cepat dari *propeller* 2 dan 4. Sehingga mengakibatkan *Quadrotor* bergerak menuju

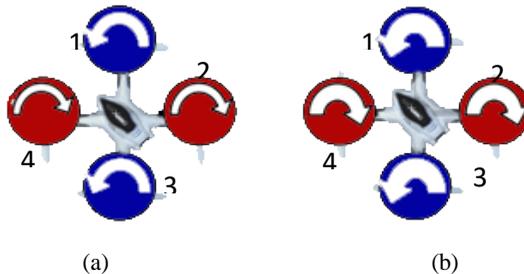
propeller 3. Gambar 2.4 (b) menampilkan *Quadrotor* bergerak menuju *propeller* 1.



Gambar 2. 4 Gerak *Pitch* [3]

d. *Yaw* (U_4)

Yaw adalah gerak *Quadrotor* terhadap sumbu Z. Gerak *yaw* dapat diperoleh dengan mempercepat *propeller* yang bergerak searah jarum jam atau mempercepat *propeller* yang bergerak berlawanan arah jarum jam. Pada Gambar 2.5 (a) ditampilkan *Quadrotor* yang bergerak berlawanan arah jarum jam. Dimana pada *propeller* 1 dan 3 kecepatannya diperbesar, sedangkan *propeller* 4 dan 2 kecepatannya diperlambat. Sehingga menyebabkan *Quadrotor* bergerak berlawanan arah jarum jam. Sedangkan untuk *Quadrotor* yang bergerak searah jarum jam terdapat pada Gambar 2.5 (b).



Gambar 2. 5 Gerak *Yaw* [3]

2.3. Pemodelan *Quadrotor* [4]

Pada *Quadrotor* terdapat 6 derajat kebebasan (*degree of freedom*) dengan 12 keluaran untuk mengatur gerak translasi (X,Y,Z) dan rotasi (ϕ, θ, Ψ) pada *Quadrotor*. Dengan menggunakan analisis kinematika dan dinamika didapatkan persamaan model matematika dari *Quadrotor* seperti pada Persamaan (2.1).

$$\begin{aligned}\ddot{x} &= (\cos \phi \sin \theta \cos \Psi + \sin \phi \sin \Psi) \frac{U_1}{m} \\ \ddot{y} &= (\cos \phi \sin \theta \sin \Psi - \sin \phi \cos \Psi) \frac{U_1}{m} \\ \ddot{z} &= (-g + \cos \phi \cos \theta) \frac{U_1}{m} \\ \dot{p} &= \frac{I_{yy} - I_{zz}}{I_{xx}} qr + \frac{U_2 l}{I_{xx}} \\ \dot{q} &= \frac{I_{zz} - I_{yy}}{I_{yy}} qr + \frac{U_3 l}{I_{yy}} \\ \dot{r} &= \frac{I_{xx} - I_{yy}}{I_{zz}} pq + \frac{U_4 d}{I_{zz}}\end{aligned}\quad (2.1)$$

dengan

- x, y, z = posisi *Quadrotor*
- g = gaya gravitasi
- m = massa *Quadrotor*
- \ddot{x} = Percepatan terhadap sumbu X
- \ddot{y} = Percepatan terhadap sumbu Y
- \ddot{z} = Percepatan terhadap sumbu Z
- Φ = Sudut yang dibentuk akibat gerak *roll*
- θ = Sudut yang dibentuk akibat gerak *pitch*
- Ψ = Sudut yang dibentuk akibat gerak *yaw*
- \dot{p} = Percepatan sudut terhadap sudut X
- \dot{q} = Percepatan sudut terhadap sudut Y
- \dot{r} = Percepatan sudut terhadap sudut Z
- p = Kecepatan sudut X
- q = Kecepatan sudut Y
- r = Kecepatan sudut Z
- I_{xx} = Momen inersia frame pada sumbu X
- I_{yy} = Momen inersia frame pada sumbu Y
- I_{zz} = Momen inersia frame pada sumbu Z

Gaya pada *Quadrotor* merupakan kombinasi gaya angkat (F) dari 4 motor sehingga akan menghasilkan gerak *thrust* (U_1), *roll* (U_2), *pitch* (U_3), dan *yaw* (U_4) sesuai dengan Persamaan (2.2).

$$\begin{aligned} U_1 &= F_1 + F_2 + F_3 + F_4 \\ U_2 &= F_2 - F_4 \\ U_3 &= F_1 - F_3 \\ U_4 &= F_1 + F_3 - F_2 - F_4 \end{aligned} \quad (2.2)$$

2.4 Parameter *Quadrotor* [4]

Pada Tugas Akhir ini parameter yang digunakan sebagai acuan penyusunan kontroler dan simulasi adalah parameter Qball-X4. Nilai parameter Qball-X4 dapat dinyatakan pada Tabel 2.1.

Tabel 2. 1 Nilai Parameter *Quadrotor*

| 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> actuator | ω | 15 rad/s |
| 10 | Konstanta gaya dorong | K | 120 N |

2.5 Linearisasi[1]

Berdasarkan sifat linearitasnya, sistem dibagi menjadi dua bagian, yaitu sistem linear dan sistem nonlinear. Sistem linear adalah sistem yang memenuhi hukum superposisi. Prinsip superposisi adalah respon sistem (keluaran) terhadap jumlah bobot sinyal akan sama dengan jumlah bobot yang sesuai dari respon (keluaran) sistem terhadap masing-masing sinyal masukan individual. Sedangkan sistem nonlinear adalah suatu sistem yang sifatnya tidak tetap, mudah berubah, sulit dikontrol, dan sulit diprediksi. Selain itu, sistem nonlinear memiliki perilaku kualitatif, yaitu mempunyai nilai ekuilibrium lebih dari satu.

Oleh karena itu, digunakan sistem linearisasi 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 dapan direpresentasikan seperti pada Persamaan (2.3).

$$\begin{aligned}
 \dot{x}_1 &= f_1(x_1, \dots, x_n, u_1, \dots, u_m) \\
 \dot{x}_2 &= f_2(x_1, \dots, x_n, u_1, \dots, u_m) \\
 &\vdots \\
 \dot{x}_n &= f_n(x_1, \dots, x_n, u_1, \dots, u_m) \\
 \dot{y}_1 &= h_1(x_1, \dots, x_n, u_1, \dots, u_m) \\
 \dot{y}_2 &= h_2(x_1, \dots, x_n, u_1, \dots, u_m) \\
 &\vdots \\
 \dot{y}_p &= h_p(x_1, \dots, x_n, u_1, \dots, u_m)
 \end{aligned} \tag{2.3}$$

$x = [x_1, \dots, x_n]^T$ merupakan vektor state sistem
 $u = [u_1, \dots, u_m]^T$ merupakan sinyal kontrol
 $x = [y_1, \dots, y_p]^T$ merupakan keluaran sistem

Persamaan (2.3) dapat disederhanakan menjadi Persamaan (2.5) dan (2.6).

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

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

sehingga dapat ditulis:

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

maka hasil linearisasi dari sistem:

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

dengan $A \in \mathcal{R}^{n \times n}$, $B \in \mathcal{R}^{n \times n}$, $C \in \mathcal{R}^{n \times n}$, $D \in \mathcal{R}^{n \times n}$. Jika $\dot{x} = f(x)$ dan $y = f(x)$ merupakan kurva nonlinier. Maka:

$$\delta y = f \delta x = \frac{df(x)}{dx}\tag{2.9}$$

dengan $\delta x \delta y = f(x) - f(x^*)$ dan $\delta y = x - x^*$

Apabila sistem bekerja pada titik $(0,0)$

$$\begin{aligned}\delta y &= f(x) - f(x^*) \\ \delta y &= f(x) - 0 \\ \delta y &= f(x) \\ \delta x &= x - x^* \\ \delta x &= x - 0 \\ \delta x &= x \\ y &= \frac{df(x)}{dx}\end{aligned}\tag{2.10}$$

Maka hasil linearisasi dapat ditulis seperti pada persamaan berikut:

$$A = \frac{df(x,u)}{dx} = \begin{bmatrix} \frac{df_1(x,u)}{dx_1} & \dots & \frac{df_1(x,u)}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n(x,u)}{dx_1} & \dots & \frac{df_n(x,u)}{dx_n} \end{bmatrix}\tag{2.11}$$

$$B = \frac{df(x,u)}{du} = \begin{bmatrix} \frac{df_1(x,u)}{du_1} & \dots & \frac{df_1(x,u)}{du_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n(x,u)}{du_1} & \dots & \frac{df_n(x,u)}{du_n} \end{bmatrix}\tag{2.12}$$

$$C = \frac{dh(x,u)}{dx} = \begin{bmatrix} \frac{dh_1(x,u)}{dx_1} & \dots & \frac{dh_1(x,u)}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{dh_n(x,u)}{dx_1} & \dots & \frac{dh_n(x,u)}{dx_n} \end{bmatrix}\tag{2.13}$$

$$D = \frac{dh(x,u)}{du} = \begin{bmatrix} \frac{dh_1(x,u)}{du_1} & \dots & \frac{dh_1(x,u)}{du_n} \\ \vdots & \ddots & \vdots \\ \frac{dh_n(x,u)}{du_1} & \dots & \frac{dh_n(x,u)}{du_n} \end{bmatrix} \quad (2.14)$$

2.6 Teori Fuzzy [1]

Logika *Fuzzy* pertama kali dikembangkan oleh Lotfi A. Zadeh melalui tulisannya pada tahun 1965 tentang teori himpunan *Fuzzy*. Pada tahun tersebut, Zadeh memodifikasi teori himpunan dimana setiap anggotanya memiliki derajat keanggotaan antara 0 sampai 1, himpunan ini disebut himpunan samar (*Fuzzy set*). Di mana logika klasik menyatakan bahwa segala hal dapat diekspresikan dalam istilah binary (0 atau 1, hitam atau putih, ya atau tidak), logika *Fuzzy* menggantikan kebenaran boolean dengan tingkat kebenaran. Sistem logika *Fuzzy* terdiri dari himpunan *Fuzzy* dan aturan *Fuzzy*.

2.6.1 Himpunan Fuzzy

Yaitu cara untuk pengelompokan objek berdasarkan variabel bahasa (linguistik) yang dinyatakan dalam fungsi keanggotaan. Himpunan tegas menyatakan bahwa setiap pembicaraan X terdiri dari semua elemen yang membentuk himpunan yang dimaksud. Jika terdapat suatu himpunan semesta A dan a adalah elemen dari A ($a \in A$), maka memiliki 2 kemungkinan yaitu:

$$\mu_J(a) = \begin{cases} 1; & x \in A \\ 0; & x \notin A \end{cases} \quad (2.15)$$

2.6.2 Fungsi Keanggotaan

Fungsi keanggotaan dinyatakan dengan derajat keanggotaan sebuah nilai terhadap nilai tegasnya yang berkisar antara 0 sampai 1. Fungsi keanggotaan memiliki berbagai bentuk antara lain segitiga, trapesium, *Gaussian*, *Generalized Bell*, dan beberapa bentuk lain. Ada beberapa fungsi yang bisa digunakan untuk memetakan titik-titik input data ke dalam nilai keanggotaannya (derajat keanggotaan):

1. Fungsi Segitiga

Kurva segitiga merupakan gabungan antara 2 garis linear. Fungsi keanggotaan segitiga ditentukan oleh tiga parameter $\{a, b, c\}$ dengan $a < b < c$ sesuai dengan Persamaan (2.14). Ketiga parameter tersebut menyatakan titik-titik *input* x untuk sudut segitiga.

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

dimana a dan c menyatakan nilai batas sedangkan b menyatakan puncak segitiga.

2. Fungsi *Gaussian*

Fungsi keanggotaan *Gaussian* ditentukan oleh dua parameter $\{c, \sigma\}$ sesuai dengan Persamaan (2.17) yang menentukan nilai domain pada pusat kurva dan lebar kurva.

$$f(x; \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \quad (2.17)$$

dimana c menyatakan titik pusat derajat keanggotaan di titik tersebut bernilai 1 dan σ menyatakan lebar fungsi *Gaussian*.

3. Fungsi *generalized bell*

Fungsi keanggotaan *generalized bell* ditentukan oleh tiga parameter $\{a, b, c\}$ sesuai dengan Persamaan (2.18).

$$f(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2.18)$$

dimana c menyatakan titik pusat dengan keanggotaan bernilai 1, b merupakan bilangan positif dan a menyatakan lebar fungsi *bell*.

4. Fungsi Trapezoidal

Fungsi keanggotaan trapezoidal pada dasarnya sama seperti bentuk segitiga. Hanya saja ada beberapa titik memiliki nilai keanggotaan 1. Dinyatakan dengan empat parameter $\{a, b, c, d\}$ dengan $a < b < c < d$.

$$f(x; a, b, c) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & b \leq x \leq c \\ \frac{d-x}{d-c} & c \leq x \leq d \\ 0 & d \leq x \end{cases} \quad (2.19)$$

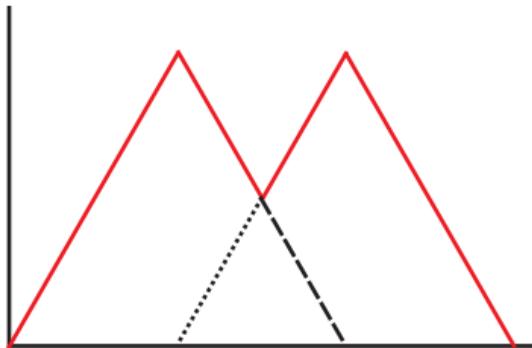
dimana a menyatakan nilai batas minimum, b dan c menyatakan titik puncak trapezium dan d menyatakan batas maksimum.

2.6.3. Operasi Himpunan *Fuzzy*

Operasi logika adalah operasi yang mengkombinasikan dan memodifikasi 2 atau lebih himpunan *Fuzzy*. Operasi ini dilakukan dalam proses inferensi *Fuzzy*. Macam – macam operator *Fuzzy* antara lain adalah operator *min* (minimum), *max* (maximum), dan *product*. Berikut ini akan dijelaskan mengenai penggunaan beberapa operasi pada himpunan *Fuzzy* tersebut.

- **OR (Union)**

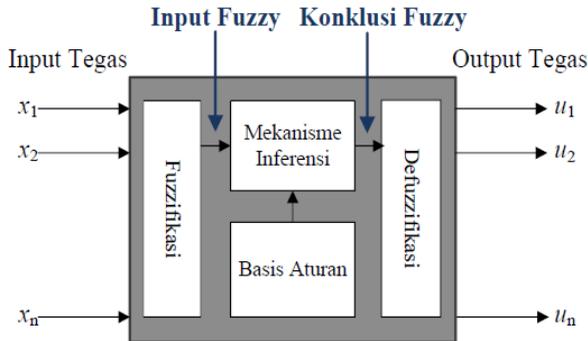
Union dari 2 himpunan adalah maksimum dari tiap pasang elemen element pada kedua himpunan



Gambar 2. 6 *Union*

- **AND (Intersection)**

Intersection adalah irisan dari 2 himpunan *Fuzzy* adalah minimum dari tiap pasang elemen pada kedua himpunan.



Gambar 2. 8 Sistem kontrol *Fuzzy*

Pada Gambar 2.8 memperlihatkan bahwa terdapat 4 komponen dalam kontroler *Fuzzy*, yaitu:

1. Basis aturan
Didalam basis aturan berisikan pemetaan *input* tegas ke *output* dalam sistem *Fuzzy* sesuai dengan variabel linguistik yang telah ditentukan untuk mencapai hasil kontrol yang baik.
2. Mekanisme inferensi
Interpretasi dan aplikasi pengetahuan untuk mengambil keputusan sesuai dengan basis aturan agar diperoleh cara pengontrolan *plant* yang terbaik.
3. Fuzifikasi
Mengubah *input* tegas menjadi informasi yang mempermudah mekanisme inferensi *Fuzzy* untuk mengaktifkan dan menggunakan aturan sesuai dengan variabel linguistik yang telah ditentukan. Terdapat tiga tipe strategi fuzzifikasi:
 - a. *Fuzzy singleton*
Input diinterpretasikan sebagai himpunan *Fuzzy* A yang memiliki fungsi keanggotaan $\mu_A(x)$ sama dengan nol kecuali pada titik $x_0 \rightarrow \mu_A(x) = 1$
 - b. *Fuzzy number*
Random varabel (*noisy measurement*)
 - c. *Hybrid Fuzzy/random number*

4. Defuzifikasi

Proses defuzzifikasi yaitu menghitung keluaran kontrol *Fuzzy* menjadi keluaran tegas. Terdapat 2 metode defuzzifikasi:

a. *Center of gravity* (COG)

Center of gravity adalah defuzzifikasi dengan mencari pusat *massa* dari gabungan seluruh hasil implikasi *Fuzzy*. Syarat dari COG adalah fungsi keanggotaan harus simetris.

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

r = Banyaknya aturan *Fuzzy*

b_i = Titik tengah fungsi keanggotaan

$\int \mu_i$ = Area di bawah fungsi keanggotaan μ_i

b. *Center-(weighted) average*:

Center (weighted) average yaitu mencari rata-rata nilai tengah dari hasil implikasi *Fuzzy*. Syarat dari *Center (weighted) average* adalah fungsi keanggotaan harus simetris.

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

b_i = Titik tengah fungsi keanggotaan

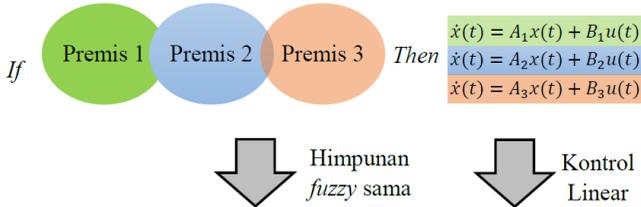
μ_i = nilai bobot tiap aturan (hasil aplikasi operator *Fuzzy*)

2.6.5 *Fuzzy Takagi-Sugeno*

Model *Fuzzy Takagi-Sugeno* (T-S) merupakan salah satu pemodelan yang merepresentasikan sistem nonlinear ke dalam beberapa pendekatan sistem linear. Model *Fuzzy Takagi-Sugeno* akan melakukan pendekatan sistem nonlinear di beberapa daerah kerja. Model *Fuzzy T-S* terdiri dari dua aturan *If-Then* yaitu aturan *plant* dan aturan kontroler yang dapat dinyatakan dengan:

- Model *Fuzzy Takagi Sugeno*

Aturan *Plant*



Aturan Kontroler



Gambar 2. 9 Aturan *Plant* dan Kontroler *Fuzzy T-S*

2.7 *Linear Quadratic Tracking (LQT)* [5]

Kontrol optimal memberikan solusi permasalahan *tracking* pada *Quadrotor* dengan pencarian nilai sinyal kontrol $u(t)$ yang optimal, sehingga indeks performansi dapat dioptimasi. Indeks performansi adalah ukuran kuantitatif performansi dari sistem yang dipilih, sehingga penekanan diberikan pada spesifikasi sistem yang dipentingkan, misalnya *error* dan sinyal kontrol.

Linear Quadratic Tracking merupakan salah satu metode kontrol optimal yang dikembangkan pada *plant* linear untuk mengatasi permasalahan *tracking*. Sebuah *plant* linear dinyatakan dalam bentuk sebagai berikut:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), x(t_0) = x_0 \\ y(t) &= Cx(t) \end{aligned} \quad (2.22)$$

dimana $x(t)$ adalah urutan ke n dari vektor *state*, $u(t)$ adalah urutan ke r dari vektor kontrol dan $y(t)$ adalah urutan ke m dari vektor keluaran. Dimisalkan $z(t)$ adalah urutan ke m dari output yang diinginkan (*set point*) dan matrix $A(t)$, $B(t)$, $C(t)$ memiliki dimensi matriks yang bersesuaian. Tujuan kontrol dari LQT adalah agar *output* sistem y dapat

mengikuti *output* model referensi sedekat mungkin dan meminimalkan indeks performansi yang diberikan. Adapun indeks performansi kuadratik secara matematis dinyatakan sebagai:

$$J(t_0) = \frac{1}{2} (C(T) - r(T))^T P (Cx(T) - r(T)) + \frac{1}{2} \int_{t_0}^T [(Cx - r)^T Q (Cx - r) + u^T R u] dt \quad (2.23)$$

dengan $P \geq 0, Q \geq 0, R > 0$

Untuk mendapatkan solusi optimal, maka terlebih dahulu dibentuk Persamaan Hamiltonian (2.20):

$$H(x, u, \lambda) = \frac{1}{2} [(Cx - r)^T Q (Cx - r) + u^T R u] + \lambda^T (Ax + Bu) \quad (2.24)$$

Agar diperoleh nilai minimum, maka syarat perlu keoptimalan adalah sebagai berikut:

| | | |
|----------------------------|---|--------|
| - Persamaan <i>state</i> | $\dot{x} = \frac{\partial H}{\partial \lambda} = Ax + Bu$ | (2.25) |
| - Persamaan <i>costate</i> | $-\dot{\lambda} = C^T Q (Cx - r) + A^T \lambda$ | |
| - Kondisi stasioner | $\frac{\partial H}{\partial u} = 0$ | |
| | $Ru + B^T \lambda = 0$ | |
| - Kontrol optimal | $u = -R^{-1} B^T \lambda$ | |

Maka akan didapatkan solusi trivial yaitu keadaan dimana sisi kanan=sisi kiri=0. Dari solusi trivial maka didapatkan persamaan *Riccati*.

$$-\dot{S} = -SA + SBR^{-1}B^T S + A^T S + C^T Q C \quad (2.26)$$

Untuk matriks Q dan R, diasumsikan sesuai dengan peforma yang diinginkan untuk sistem. Setelah didapatkan persamaan *riccati*, didapatkan persamaan differensial yaitu model *following*.

$$-\dot{v} = -SBR^{-1}B^T v + A^T v + C^T Q r \quad (2.27)$$

dengan didapakkannya matriks S yang merupakan matriks definit positif yang simetris dan v, hanya tersisa *gain* K(t) untuk dicari. K(t) dengan Persamaan (2.28).

$$K(t) = R^{-1} B^T S(t) \quad (2.28)$$

Model *following*

$$-\dot{v} = (A - BK)^T v + C^T Q r \quad (2.29)$$

Solusi *steady-state*

- Aljabar Riccati (ARE)

$$A^T S S A - S B R^{-1} B^T S + C^T Q C = 0 \quad (2.30)$$

- Gain Kalman

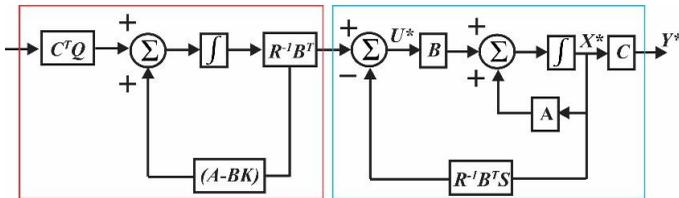
$$K(\infty) = R^{-1} B^T S(\infty) \quad (2.31)$$

- Model *following*

$$-\dot{v} = (A - BK(\infty))^T v + C^T Q r \quad (2.32)$$

dengan $x^*(t)$ adalah umpan balik dari sistem. Umpan balik nantinya akan dikalikan dengan *gain* kontrol $K(t)$ dan ditambahkan dengan $-R^{-1}B^T v$ untuk menghasilkan $u^*(t)$ yang akan digunakan sebagai masukan kepada sistem agar hasil yang diinginkan tercapai.

$$u = -K(\infty)x - R^{-1}B^T v \quad (2.33)$$



Gambar 2. 10 Blok Diagram LQT

dimana blok yang berwarna merah menunjukkan *model following* dari LQT dan untuk blok berwarna biru merupakan blok kontroler

Halaman ini sengaja dikosongkan

BAB 3 PERANCANGAN SISTEM

Pada bab ini, membahas mengenai perancangan sistem. Perancangan dibagi menjadi **tiga** tahap. Tahap pertama Linearisasi model matematika *Quadrotor* pada beberapa posisi yaitu pada sumbu X,Y,Z dan pada posisi *thrust, roll, pitch*, dan *yaw* 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 model dinamika *Quadrotor*, akan dijelaskan mengenai persamaan dinamika *Quadrotor*. Persamaan tersebut akan digunakan dalam pemodelan sistem *Quadrotor* pada Matlab-Simulink. Sehingga dapat ditentukan persamaan *state space* dari *Quadrotor*.

3.1.1 Persamaan *State Dinamika Quadrotor*

Dalam menganalisa dan desain kontroler, sebuah *plant* harus dibawa kedalam bentuk persamaan *state space*, sehingga dapat dilakukan linearisasi hingga penyusunan kontroler. Mengacu pada lampiran A, didapatkan model dinamika *Quadrotor* seperti pada Persamaan (3.1) [3].

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

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. Gaya angkat (*thrust*) yang dibangkitkan oleh setiap baling-baling dimodelkan dalam bentuk sistem orde satu terdapat pada Persamaan (3.3)[6].

$$F_i = K_T \frac{\omega}{s+\omega} u_i \quad (3.3)$$

Dengan K_T adalah konstanta gaya dorong, ω adalah lebar *bandwidth* motor dan u_i adalah sinyal kontrol dari *inner loop* kontroler ke 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:

$$\begin{aligned} x_1 &= X & x_7 &= \phi \\ x_2 &= \dot{X} & x_8 &= \dot{\phi} \\ x_3 &= Y & x_9 &= \theta \\ x_4 &= \dot{Y} & x_{10} &= \dot{\theta} \\ x_5 &= Z & x_{11} &= \psi \\ x_6 &= \dot{Z} & x_{12} &= \dot{\psi} \end{aligned} \quad (3.4)$$

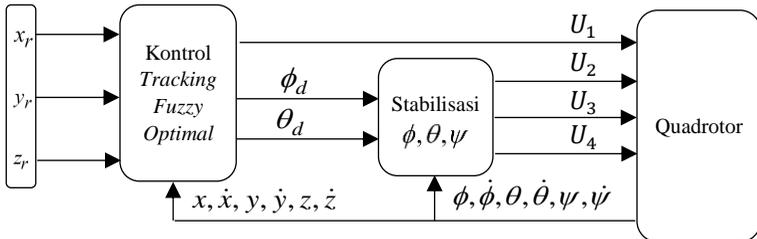
dengan menggunakan Persamaan (3.1) dapat diubah kedalam bentuk *state space* seperti pada Persamaan (3.5) dan Persamaan (3.6). Dimana terdapat enam *state space* yang digunakan untuk mengatur gerak translasi dan terdapat enam *state space* yang digunakan untuk mengatur gerak rotasi.

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ \dot{Y} \\ \ddot{Y} \\ \dot{Z} \\ \ddot{Z} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} x_2 \\ (s_{x11}s_{x7} + c_{x11}s_{x9}c_{x7})U_1 / m \\ x_4 \\ (-c_{x11}s_{x7} + s_{x11}s_{x9}c_{x7})U_1 / m \\ x_6 \\ -g + (c_{x9}c_{x7})U_1 / m \end{bmatrix} \quad (3.5)$$

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} x_8 \\ ((J_{yy} - J_{zz})x_{10}x_{12} + U_2l) / J_{xx} \\ x_{10} \\ ((J_{zz} - J_{xx})x_8x_{12} + U_3l) / J_{yy} \\ x_{12} \\ ((J_{xx} - J_{yy})x_8x_{10} + U_4d) / J_{zz} \end{bmatrix} \quad (3.6)$$

3.2 Perancangan Kontroler

Pada *Quadrotor* terdapat dua macam permasalahan yang harus diselesaikan agar *Quadrotor* mampu terbang dan benjalan 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* pada sumbu 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* dan metode optimal. Pada Gambar 3.1 ditampilkan blok diagram dari perancangan kontroler pada tugas akhir ini.



Gambar 3. 1 Perancangan Blok Kontroler

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 (0^0) sehingga didapatkan model *linear* didaerah ekuilibriumnya. Dengan menggunakan linearisasi Jacobian didapatkan model *linear* dari persamaan gerak rotasi sebagai berikut.

Gerak *roll* (ϕ)

$$\begin{aligned}\dot{x}_7 &= x_8 \\ \dot{x}_8 &= ((J_{yy} - J_{zz})x_{10}x_{12} + U_2l) / J_{xx}\end{aligned}\quad (3.7)$$

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

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

State space dari Persamaan (3.8) terdapat pada Persamaan (3.9).

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 6,6667 \end{bmatrix} U_\phi\quad (3.9)$$

Gerak *pitch* (θ)

$$\begin{aligned}\dot{x}_9 &= x_{10} \\ \dot{x}_{10} &= ((J_{zz} - J_{xx})x_8x_{12} + U_3l) / J_{yy}\end{aligned}\quad (3.10)$$

Jika dilinearisasi disekitar sudut terkecil (0^0) didapatkan Persamaan (3.11).

$$A_{pitch} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{pitch} = \begin{bmatrix} 0 \\ 6,6667 \end{bmatrix}\quad (3.11)$$

State space dari Persamaan (3.11) terdapat pada Persamaan (3.12).

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 6,6667 \end{bmatrix} U_\theta\quad (3.12)$$

Gerak *yaw* (ψ)

$$\begin{aligned}\dot{x}_{11} &= x_{12} \\ \dot{x}_{12} &= ((J_{xx} - J_{yy})x_8x_{10} + U_4d) / J_{zz}\end{aligned}\quad (3.13)$$

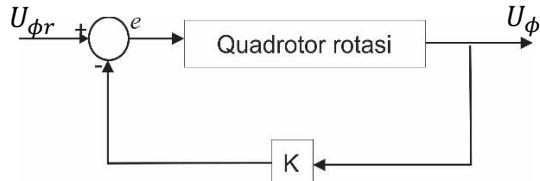
Jika dilinearisasi disekitar sudut terkecil (0^0) didapatkan Persamaan (3.14).

$$A_{yaw} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{yaw} = 10^{-3} \begin{bmatrix} 0 \\ 0,7825 \end{bmatrix} \quad (3.14)$$

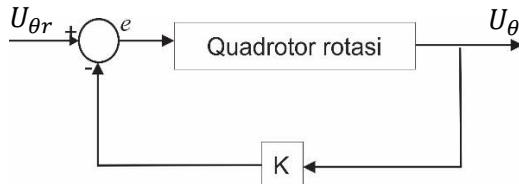
State space dari Persamaan (3.14) terdapat pada Persamaan (3.15).

$$\begin{bmatrix} \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 \\ 6,6667 \end{bmatrix} U_{\psi} \quad (3.16)$$

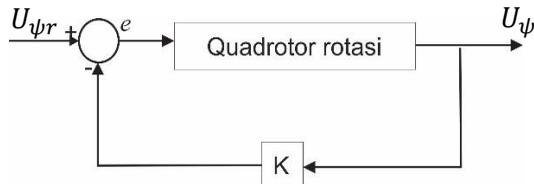
Setelah dilakukan linearisasi disekitar 0, maka disusun kontrol *state-feedback*, untuk menjaga agar *state* tetap berada pada sekitar titik ekuilibriumnya. Dengan *state* berada pada disekitar titik ekuilibriumnya maka *Quadrotor* dapat terbang dengan stabil.



(a) Kontrol sudut roll



(b) Kontrol sudut pitch



(c) Kontrol sudut yaw

Gambar 3. 2 Blok Diagram Kontrol Rotasi

Pada kontrol *state-feedback* terdapat *gain* kontrol K yang dicari menggunakan teknik *ackerman*. Teknik *ackerman* ini mencari *gain*

kontroler dengan melihat peletakan *pole* dominan. *Pole* dominan sendiri adalah *pole* dari *state* yang dikontrol.

Langkah pertama adalah menentukan *rise time* dari sistem. Nilai ω didapatkan dari Persamaan (3.17)[7].

$$\begin{aligned} t_r &= 0,3 \text{ detik} \\ \omega &= \frac{1,8}{t_r} \end{aligned} \quad (3.17)$$

dengan maksimum *overshoot* 5%

$$M_p = 5\%$$

nilai zeta bisa dicari dengan Persamaan (3.18).

$$\zeta = \frac{\sqrt{(\log M_p)^2}}{\sqrt{\log M_p + \pi^2}} \quad (3.18)$$

sehingga desain pole yang diinginkan dapat dicari dengan menggunakan pendekatan orde 2 seperti pada Persamaan (3.19).

$$G_h = \frac{\omega^2}{1 + 2\zeta\omega + \omega^2} \quad (3.19)$$

untuk mencari *gain* menggunakan *Ackerman's formula* pertama-tama harus diperiksa matriks *controllability* pada sistem. Matriks *controllability* ditunjukkan pada Persamaan (3.20). Nilai *rank* pada matriks harus sama dengan ordo matriks itu sendiri.

$$M = [A \quad AB] \quad (3.20)$$

$$M_{roll} = \begin{bmatrix} 0 & 6,6667 \\ 6,6667 & 0 \end{bmatrix} \quad rank = 2 \quad (3.21)$$

$$M_{pitch} = \begin{bmatrix} 0 & 6,6667 \\ 6,6667 & 0 \end{bmatrix} \quad rank = 2$$

$$M_{yaw} = \begin{bmatrix} 0 & 0,7825 \times 10^{-3} \\ 0,7825 \times 10^{-3} & 0 \end{bmatrix} \quad rank = 2$$

Pada persamaan tugas akhir ini didapatkan matriks *controllability* seperti pada Persamaan (3.21).

Jika menggunakan *Ackerman's formula*, untuk menentukan matriks *gain state feedback* harus ditentukan matriks karakteristik *polynomial* $\phi(A)$, dimana nilai $\phi(A)$ dapat ditentukan dari Persamaan (3.22).

$$\phi(A) = A^2 + \alpha A + \alpha_3 I \quad (3.22)$$

sehingga matriks *gain feedback* dapat ditentukan dengan Persamaan (3.23).

$$K = [0 \quad 1]M^{-1}\phi(A) \quad (3.23)$$

nilai *gain* yang didapatkan berdasarkan Persamaan (3.23) ditunjukkan pada Persamaan (3.24).

$$\begin{aligned} K_{roll} &= [5,4000 \quad 2,0567] \\ K_{pitch} &= [5,4000 \quad 2,0567] \\ K_{yaw} &= [4,6006 \times 10^4 \quad 1,7523 \times 10^4] \end{aligned} \quad (3.24)$$

3.2.2 Altitude Z

Posisi Z diatur pada ketinggian konstan yaitu 1 m diatas lantai. Metode yang digunakan untuk mengatur posisi Z adalah metode optimal LQT sesuai dengan Gambar 2.13. Metode ini menggunakan persamaan kuadrat yang dicari dengan menggunakan persamaan *linear*. Dipilih nilai Q dan R dengan menggunakan *trial error* sehingga error dapat sekecil mungkin. Dari metode optimal tersebut akan didapatkan nilai *gain* K untuk altitude dari *Quadrotor*.

Dinamika Z memiliki persamaan seperti pada Persamaan (3.25) dan (3.26) dimana akan dilinearisasi menggunakan matriks Jacobian dengan daerah kerja sekitar 0^0 . Didapatkan model *linear* dari dinamika Z seperti pada Persamaan (3.27).

$$\dot{x}_5 = x_6 \quad (3.25)$$

$$\dot{x}_6 = \frac{-g + (c_{x9}c_{x7})U_1}{m} \quad (3.26)$$

$$A_z = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, B_z = \begin{bmatrix} 0 \\ 0,2858 \end{bmatrix} \quad (3.27)$$

Untuk mencari *gain* kontroler maka digunakan metode optimal dengan nilai Q dan R pada Persamaan (3.28).

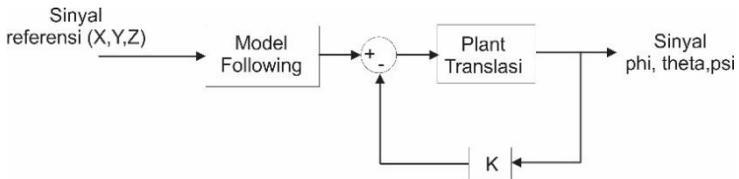
$$Q = 150; \quad R = 1 \quad (3.28)$$

sehingga didapatkan *gain* untuk *altitude* Z sebagai berikut.

$$K_z = [12,2474 \quad 9,2578] \quad (3.29)$$

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* dan *LQT*.



Gambar 3. 3 Blok Diagram Kontrol *Tracking*

3.2.3.1 Linearisasi Model Matematika *Quadrotor*

Persamaan (3.30) dan (3.31) 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 dilakukan menggunakan metode linearisasi Jacobian, dengan daerah kerja disekitar titik 0^0 , 10^0 dan -10^0 seperti yang ditampilkan pada Tabel 3.1.

Tabel 3. 1 Daerah Linearisasi ϕ dan θ

| No | Parameter | State |
|----|---|----------------------|
| 1 | $\phi = \theta = 0, \quad \dot{\phi} = \dot{\theta} = 0$ | $[0 \quad 0]$ |
| 2 | $\phi = \theta = \pm 10^0, \quad \dot{\phi} = \dot{\theta} = 0$ | $[\pm 10^0 \quad 0]$ |

Linearisasi dinamika gerak sumbu X didapatkan pada Persamaan (3.30).

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{(s_{x11}s_{x7} + c_{11}s_{x9}c_{x7})U_1}{m} \end{aligned} \quad (3.30)$$

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

$$\dot{x} = A_{xi} + B_{xi}\theta$$

$$A_{xi} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{xi} = \begin{bmatrix} 0 \\ \frac{(c_{x11}c_{x9}c_{x7})U_1}{m} \end{bmatrix} \quad (3.31)$$

Hasil linearisasi pada sumbu X ditampilkan pada Persamaan (3.32).

$$A_{x1} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{x1} = \begin{bmatrix} 0 \\ 9,81 \end{bmatrix} \quad (3.32)$$

$$A_{x2} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{x2} = \begin{bmatrix} 0 \\ 6,8107 \end{bmatrix}$$

Sedangkan linearisasi dinamika gerak sumbu Y didapatkan pada Persamaan (3.33).

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{(-c_{x11}s_{x7} + s_{x11}s_{x9}c_{x7})U_1}{m} \quad (3.33)$$

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

$$\dot{x} = A_{yi} + B_{yi}\theta$$

$$A_{yi} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{yi} = \begin{bmatrix} 0 \\ \frac{(-c_{x11}s_{x7} + s_{x11}s_{x9}c_{x7})U_1}{m} \end{bmatrix} \quad (3.34)$$

Jika linearisasi lokal dilakukan didaerah kerja ϕ dan θ seperti pada Tabel 3.1, maka didapatkan matriks sistem hasil linearisasi seperti pada Persamaan (3.34) dimana linearisasi dilakukan pada titik kerja 0^0 dan $\pm 10^0$.

$$A_{y1} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{y1} = \begin{bmatrix} 0 \\ -9,81 \end{bmatrix} \quad (3.35)$$

$$A_{y2} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B_{y2} = \begin{bmatrix} 0 \\ 8,1062 \end{bmatrix}$$

3.2.3.2 Pemodelan *Fuzzy* Takagi-Sugeno

Model *Fuzzy* Takagi-Sugeno dibentuk dari linearisasi persamaan gerak pada sumbu X dan Y. Linearisasi sistem dilakukan pada tiga daerah kerja seperti pada Tabel 3.1, sehingga didapatkan variabel premis untuk model *Fuzzy* T-S yaitu sudut ϕ untuk gerak sumbu Y dan sudut θ untuk gerak sumbu X.

Apabila sudah didapatkan matriks linearisasi pada sumbu X dan Y, maka dapat dibuat aturan *Fuzzy* T-S sebagai berikut:

Aturan *plant* untuk gerak X

Aturan *plant* ke-1:

If x_9 is M_1 (sekitar $\pm 0^0$)

Then $\dot{x} = A_{x1}x + B_{x1}\theta$ (3.36)

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

Aturan *plant* ke-2:

If x_9 is $M_2 \pm \frac{\pi}{18} \text{rad}$

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

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

Aturan *plant* untuk gerak Y

Aturan *plant* ke-1:

If x_7 is M_1 (sekitar $\pm 0^0$)

Then $\dot{x} = A_{y1}x + B_{y1}\phi$ (3.37)

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

Aturan *plant* ke-2:

If x_7 is $M_2 \pm \frac{\pi}{18} \text{rad}$

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

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

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

$$C_{xi} = C_{yi} = [1 \quad 0] \quad (3.38)$$

$$D_{xi} = D_{yi} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Berdasarkan skema PDC, dapat disusun aturan kontroler *Fuzzy* yang bersesuaian dengan aturan *plant* pada Persamaan (3.37-3.38) sebagai berikut:

Aturan kontroler untuk gerak X

Aturan kontroler ke-1:

$$\begin{aligned} & \text{If } x_9 \text{ is } M_2 \text{ } 0^0 \text{ (sekitar } \pm 0^0) \\ & \text{Then } \theta = K_{x2}[x - x_r] \end{aligned} \quad (3.39)$$

Aturan kontroler ke-2:

$$\begin{aligned} & \text{If } x_9 \text{ is } M_2 \frac{\pi}{18} \text{ } 0 \text{ (sekitar } \pm 10^0) \\ & \text{Then } \theta = K_{x2}[x - x_r] \end{aligned}$$

Aturan kontroler untuk gerak Y

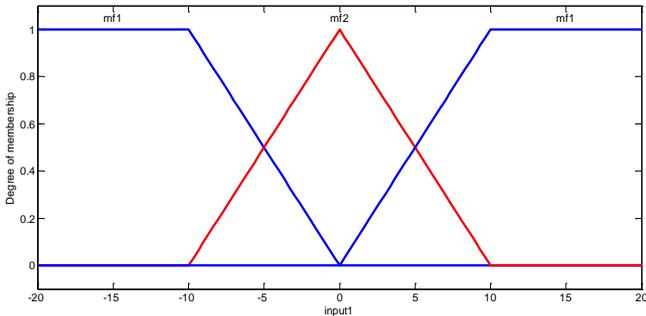
Aturan kontroler ke-1:

$$\begin{aligned} & \text{If } x_7 \text{ is } M_1 \text{ (sekitar } \pm 0^0) \\ & \text{Then } \phi = K_{x1}[x - x_r] \end{aligned} \quad (3.40)$$

Aturan kontroler ke-2:

$$\begin{aligned} & \text{If } x_7 \text{ is } M_2 \frac{\pi}{18} \text{ } 0 \text{ (sekitar } \pm 10 \text{ rad)} \\ & \text{Then } \phi = K_{x2}[x - x_r] \end{aligned}$$

M_1 merupakan fungsi keanggotaan yang didefinisikan dengan fungsi segitiga. Pada tugas akhir ini digunakan 3 derajat keanggotaan segitiga. Dengan nilai yang terdapat pada Gambar 3.4.



Gambar 3. 4 Himpunan Keanggotaan *Fuzzy*

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:

$$\dot{x}(t) = \sum_{i=1}^2 h_i(x_z(t)) [A_i x(t) + B_i u(t)] \quad (3.41)$$

$$u(t) = \sum_{j=1}^2 h_j(x_z(t)) K_j [x(t) - x_r(t)] \quad (3.42)$$

substitusi (3.41) ke (3.42)

$$\dot{x}(t) = \sum_{i=1}^2 \sum_{j=1}^2 h_j(x_z(t)) h_i(x_z(t)) [(A_i + B_i K_j)x(t) - B_i K_j x_r(t)] \quad (3.43)$$

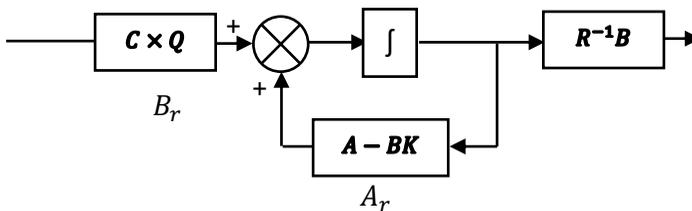
dengan x_z adalah *state* yang dikontrol.

$$h_j(x_z(t)) = \frac{\mu_j(x_z(t))}{\sum_{i=1}^2 \mu_i(x_z(t))} \quad (3.44)$$

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

3.2.1.3 Linear Quadratic Tracking

Pada Subbab 2.8 sudah dijelaskan bahwa salah satu cara untuk memaksa respon *Quadrotor* bergerak mengikuti sinyal referensi adalah dengan menggunakan model referensi. Model referensi disini ditentukan dengan menggunakan metode optimal. Model *following* diperoleh dengan menggunakan metode LQT. Yang nantinya model *following* tersebut akan dibandingkan dengan *gain* optimal. Nilai matriks pada *gain following* yang digunakan dapat dicari dengan menggunakan Persamaan (3.40).



Gambar 3. 5 Model *Following*

Dimana matriks A , B dan C didapatkan dari linearisasi *Quadrotor* pada sudut $0, \pm 10$ derajat. Sedangkan matriks K didapatkan dari Persamaan (3.45).

$$K = R^{-1}BS^T \quad (3.45)$$

dimana S merupakan matriks yang didapatkan dari persamaan Riccati. Nilai S harus definite positif agar sistem dapat stabil. Hasil persamaan *riccati* untuk sumbu X pada titik kerja 0^0 akan didapatkan matriks S seperti pada Persamaan (3.46).

$$A^T S + SA - SBR^{-1}B^T S + C^T QC = 0 \quad (3.46)$$

$$\begin{aligned} \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \\ - \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 \\ 9,81 \end{bmatrix} \begin{bmatrix} 0,3472 \\ 0,981 \end{bmatrix} \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \\ + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1,8 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} \begin{bmatrix} 0 & 0 \\ S_1 & S_2 \end{bmatrix} + \begin{bmatrix} 0 & S_1 \\ 0 & S_2 \end{bmatrix} - \begin{bmatrix} 33,4131S_2^2 & 33,4131S_2S_3 \\ 33,4131S_2S_3 & 33,4131S_3^2 \end{bmatrix} + \begin{bmatrix} 1,8 & 0 \\ 0 & 0 \end{bmatrix} \\ = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{aligned}$$

$$\begin{bmatrix} -33,4131S_2^2 + 1,8 & -33,4131S_2S_3 + S_1 \\ -33,4131S_2S_3 + S_1 & -33,4131S_3^2 + 2S_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

- $-33,4131S_2^2 + 1,8 = 0$
 $S_2 = \begin{bmatrix} 0,2321 \\ -0,2321 \end{bmatrix}$
- $-33,4131S_3^2 + 2S_2 = 0$
 $-33,4131S_3^2 + 2(0,2321) = 0$
 $S_3 = \begin{bmatrix} 0,1179 \\ -0,1179 \end{bmatrix}$
- $-33,4131S_2S_3 + S_1 = 0$
 $-33,4131(0,2321)(0,1179) + S_1 = 0$
 $S_1 = 0,9143$

$$\begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} = \begin{bmatrix} 0,9143 & 0,2321 \\ 0,2321 & 0,1179 \end{bmatrix}$$

Nilai *gain* K dapat dicari dengan menggunakan Persamaan (2.31), sehingga

$$\begin{aligned}
 K &= R^{-1}B^T S \\
 &= [0,3472][0 \quad 9,81] \begin{bmatrix} 0,9143 & 0,2321 \\ 0,2123 & 0,1179 \end{bmatrix} \\
 &= [0,7905 \quad 0,4015] \tag{3.47}
 \end{aligned}$$

persamaan *riccati* untuk sumbu X pada titik kerja 10^0 terdapat pada Persamaan (3.48).

$$\begin{aligned}
 &\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} - \\
 &\quad \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 \\ 6,8107 \end{bmatrix} [0,1667] [0 \quad 6,8107] \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \\
 &\quad \begin{bmatrix} 1 \\ 0 \end{bmatrix} [0,51] \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\
 &\begin{bmatrix} 0 & 0 \\ S_1 & S_2 \end{bmatrix} + \begin{bmatrix} 0 & S_1 \\ 0 & S_2 \end{bmatrix} - \begin{bmatrix} 7,7325S_2^2 & 7,7325S_2S_3 \\ 7,7325S_2S_3 & 7,7325S_3^2 \end{bmatrix} + \begin{bmatrix} 0,51 & 0 \\ 0 & 0 \end{bmatrix} = \\
 &\quad \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\
 &\begin{bmatrix} -7,7325S_2^2 + 0,51 & S_1 - 7,7325S_2S_3 \\ S_1 - 7,7325S_2S_3 & 2S_2 - 7,7325S_3^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\
 &\quad \bullet -7,7325S_2^2 + 0,51 = 0 \\
 &\quad S_2 = \begin{bmatrix} 0,2568 \\ -0,2568 \end{bmatrix} \\
 &\quad \bullet 2S_2 - 7,7325S_3^2 = 0 \\
 &\quad 2(0,2568) - 7,7325S_3^2 = 0 \\
 &\quad S_3 = \begin{bmatrix} 0,2577 \\ -0,2577 \end{bmatrix} \\
 &\quad \bullet S_1 - 7,7325S_2S_3 = 0 \\
 &\quad S_1 - 7,7325(0,2568)(0,2577) = 0 \\
 &\quad S_1 = 0,5117 \\
 &\begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} = \begin{bmatrix} 0,5117 & 0,2568 \\ 0,2568 & 0,2577 \end{bmatrix} \tag{3.48}
 \end{aligned}$$

Sehingga nilai *gain* K terdapat pada Persamaan (3.39).

$$\begin{aligned}
K &= [0,1667][0 \quad 6,8107] \begin{bmatrix} 0,5117 & 0,2568 \\ 0,2568 & 0,2577 \end{bmatrix} \\
&= [0,2916 \quad 0,2926] \tag{3.49}
\end{aligned}$$

Persamaan *riccati* pada sumbu Y pada titik kerja 0^0 terdapat pada Persamaan (3.50).

$$\begin{aligned}
&\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} - \\
&\begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 \\ -9,81 \end{bmatrix} [0,3472] [0 \quad -9,81] \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \\
&\begin{bmatrix} 1 \\ 0 \end{bmatrix} [1,8] [1 \quad 0] = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\
&\begin{bmatrix} 0 & 0 \\ S_1 & S_2 \end{bmatrix} + \begin{bmatrix} 0 & S_1 \\ 0 & S_2 \end{bmatrix} - \begin{bmatrix} 33,4131S_2^2 & 33,4131S_2S_3 \\ 33,4131S_2S_3 & 33,4131S_3^2 \end{bmatrix} + \begin{bmatrix} 1,8 & 0 \\ 0 & 0 \end{bmatrix} = \\
&\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\
&\begin{bmatrix} -33,4131S_2^2 + 1,8 & -33,4131S_2S_3 + S_1 \\ -33,4131S_2S_3 + S_1 & -33,4131S_3^2 + 2S_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\
&\bullet \quad -33,4131S_2^2 + 1,8 = 0 \\
&\quad S_2 = \begin{bmatrix} 0,2321 \\ -0,2321 \end{bmatrix} \\
&\bullet \quad -33,4131S_3^2 + 2S_2 = 0 \\
&\quad -33,4131S_3^2 + 2(0,2321) = 0 \\
&\quad S_3 = \begin{bmatrix} 0,1179 \\ -0,1179 \end{bmatrix} \\
&\bullet \quad -33,4131S_2S_3 + S_1 = 0 \\
&\quad -33,4131(0,2321)(0,1179) + S_1 = 0 \\
&\quad S_1 = 0,9143 \\
&\begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} = \begin{bmatrix} 0,9143 & 0,2321 \\ 0,2321 & 0,1179 \end{bmatrix} \tag{3.50}
\end{aligned}$$

Nilai *gain* K dapat dicari dengan menggunakan Persamaan (2.31), sehingga

$$K = R^{-1}B^T S$$

$$\begin{aligned}
&= [0,3472][0 \quad -9,81] \begin{bmatrix} 0,9143 & 0,2321 \\ 0,2123 & 0,1179 \end{bmatrix} \\
&= [-0,7905 \quad -0,4015] \tag{3.51}
\end{aligned}$$

Persamaan *riccati* untuk sumbu Y pada titik kerja 10^0 terdapat pada Persamaan (4.52).

$$\begin{aligned}
&\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} - \\
&\begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} \begin{bmatrix} 0 \\ 8,1062 \end{bmatrix} [0,1667] \begin{bmatrix} 0 & 8,1062 \end{bmatrix} \begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} + \\
&\begin{bmatrix} 1 \\ 0 \end{bmatrix} [0,51] \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
&\begin{bmatrix} 0 & 0 \\ S_1 & S_2 \end{bmatrix} + \begin{bmatrix} 0 & S_1 \\ 0 & S_2 \end{bmatrix} - \begin{bmatrix} 10,9539S_2^2 & 10,9539S_2S_3 \\ 10,9539S_2S_3 & 10,9539S_3^2 \end{bmatrix} + \\
&\begin{bmatrix} 0,51 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}
\end{aligned}$$

$$\begin{bmatrix} -10,9539S_2^2 + 0,51 & S_1 - 10,9539S_2S_3 \\ S_1 - 10,9539S_2S_3 & 2S_2 - 10,9539S_3^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

- $-10,9539S_2^2 + 0,51 = 0$
 $S_2 = \begin{bmatrix} 0,2158 \\ -0,2158 \end{bmatrix}$
- $2S_2 - 10,9539S_3^2 = 0$
 $2(0,2158) - 10,9539S_3^2 = 0$
 $S_3 = \begin{bmatrix} 0,1985 \\ -0,1985 \end{bmatrix}$
- $S_1 - 10,9539S_2S_3 = 0$
 $S_1 - 10,9539(0,2158)(0,1985) = 0$
 $S_1 = 0,4692$

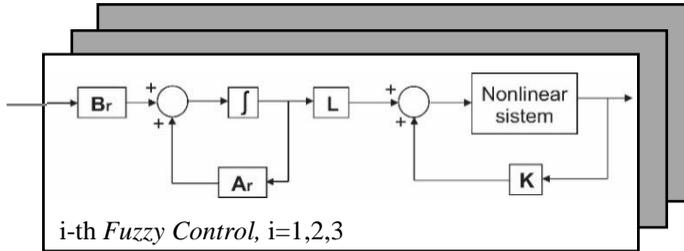
$$\begin{bmatrix} S_1 & S_2 \\ S_2 & S_3 \end{bmatrix} = \begin{bmatrix} 0,4692 & 0,2158 \\ 0,2158 & 0,1985 \end{bmatrix} \tag{3.52}$$

sedangkan untuk nilai K pada sumbu Y pada titik kerja 10^0 adalah

$$\begin{aligned}
K &= [0,1667][0 \quad 8,1062] \begin{bmatrix} 0,4692 & 0,2158 \\ 0,2158 & 0,1985 \end{bmatrix} \\
&= [0,2915 \quad 0,2682] \tag{3.53}
\end{aligned}$$

3.2.1.4 Perancangan Kontroler *Fuzzy Optimal* pada *Tracking Quadrotor*

Pada subbab ini akan dijelaskan desain kontroler untuk sistem loop tertutup sistem *Quadrotor*. Desain kontroler secara keseluruhan dapat dilihat pada Gambar 3.6. Dimana dari setiap model referensi yang digunakan akan dimasukkan pada aturan *Fuzzy Takagi-Sugeno*. Terdapat tiga model referensi yang digunakan.



Gambar 3. 6 Kontrol *Fuzzy Optimal*

sehingga nilai A_r pada model *following* sumbu X adalah sebagai berikut

$$A_{r1} = \begin{bmatrix} 0 & -7,7555 \\ 1 & -3,9384 \end{bmatrix} \quad (3.54)$$

$$A_{r2} = \begin{bmatrix} 0 & -1,983 \\ 1 & -1,9915 \end{bmatrix}$$

Nilai A_r pada model *following* sumbu Y adalah sebagai berikut

$$A_{r4} = \begin{bmatrix} 0 & -7,7555 \\ 1 & -3,9384 \end{bmatrix} \quad (3.55)$$

$$A_{r5} = \begin{bmatrix} 0 & -2,3634 \\ 1 & -2,1741 \end{bmatrix}$$

Nilai B_r dapat dicari dengan menggunakan Persamaan (3.56).

$$B_r = C^T Q \quad (3.56)$$

dimana nilai C ditentukan berdasarkan *state* matriks yang digunakan. Nilai Q didapatkan dari nilai dengan selisih *error* terkecil. Sehingga didapatkan matriks Br sebagai berikut

$$C = [1 \quad 0] \quad ; Q = [1,8] \quad (3.57)$$

$$B_{r1} = \begin{bmatrix} 1,8 \\ 0 \end{bmatrix} \quad (3.58)$$

$$B_{r2} = \begin{bmatrix} 0,51 \\ 0 \end{bmatrix}$$

$$B_{r4} = \begin{bmatrix} 1,8 \\ 0 \end{bmatrix}$$

$$B_{r5} = \begin{bmatrix} 0,51 \\ 0 \end{bmatrix}$$

sehingga didapatkan *gain* sistem dari titik 0^0 dan $\pm 10^0$ adalah sebagai berikut

Titik $x_1 = 0^0$

$$Q = [1,8], \quad R = [2,88] \quad (3.59)$$

$$S = \begin{bmatrix} 0,9141 & 0,2321 \\ 0,2321 & 0,1179 \end{bmatrix}$$

$$K_1 = [0,7906 \quad 0,4015]$$

Titik $x_2 = 10^0$

$$Q = [0,51], \quad R = [6] \quad (3.60)$$

$$S = \begin{bmatrix} 0,5117 & 0,2568 \\ 0,2568 & 0,2577 \end{bmatrix}$$

$$K_2 = [0,2916 \quad 0,2926]$$

Titik $y_1 = 0^0$

$$Q = [1,8], \quad R = [2,88] \quad (3.61)$$

$$S = \begin{bmatrix} 0,9141 & 0,2321 \\ 0,2321 & 0,1179 \end{bmatrix}$$

$$K_4 = [-0,7906 \quad -0,4015]$$

Titik $y_2 = 10^0$

$$Q = [0,51], \quad R = [6] \quad (3.63)$$

$$S = \begin{bmatrix} 0,4692 & 0,2158 \\ 0,2158 & 0,1985 \end{bmatrix}$$

$$K_5 = [0,2915 \quad 0,2682]$$

3.2.4 Perancangan Sinyal Kontrol

Dikarenakan nilai linearisasi dari setiap sudut berbeda, maka setiap *state* akan masuk kedalam sistem *Fuzzy Takagi Sugeno*. *Fuzzy* digunakan pada *tracking* sumbu X dan Y.

3.2.4.1 Perancangan 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 (3.2). Keluaran sinyal kontrol dari kontroler pada tiap sumbu harus dikombinasi dan ditambahkan gaya angkat minimal (u_{th}) seperti Persamaan berikut:

$$\begin{aligned}u_1 &= u_{th} + v_\theta + v_\psi \\u_2 &= u_{th} + v_\phi - v_\psi \\u_3 &= u_{th} - v_\theta + v_\psi \\u_4 &= u_{th} - v_\phi - v_\psi\end{aligned}\tag{3.64}$$

Pembuktian dari kombinasi sinyal kontrol tiap sumbu ini dapat dilihat pada (3.65) yang diturunkan berdasarkan pergerakan *Quadrotor*.

$$\begin{aligned}U_1 &= u_1 + u_2 + u_3 + u_4 \\&= u_{th} + v_\theta + v_\psi + u_{th} + v_\phi - v_\psi + u_{th} - v_\theta + v_\psi \\&\quad + u_{th} - v_\phi - v_\psi = 4u_{th} \\U_2 &= u_2 - u_4 \\&= u_{th} + v_\phi - v_\psi - (u_{th} - v_\phi - v_\psi) \\&= 2v_\phi \\U_3 &= u_1 - u_3 \\&= u_{th} + v_\theta + v_\psi - (u_{th} - v_\theta + v_\psi) \\&= 2v_\theta \\U_4 &= u_1 - u_2 + u_3 - u_4 \\&= u_{th} + v_\theta + v_\psi - (u_{th} + v_\phi - v_\psi) + u_{th} - v_\theta + v_\psi \\&\quad - (u_{th} - v_\phi - v_\psi) = 4v_\psi\end{aligned}\tag{3.65}$$

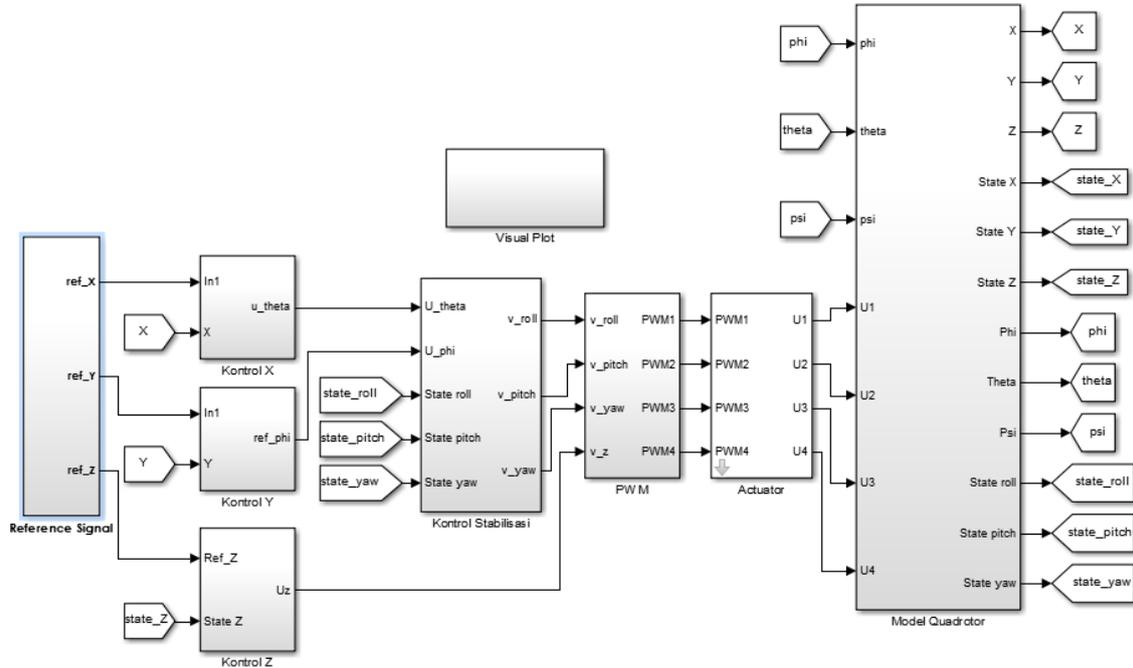
3.3 Perancangan Perangkat Simulasi

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. Dengan menggunakan diagram kontrol, dibuat diagram *Simulink* untuk mensimulasikan sistem kontrol seperti pada Gambar 3.8.

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. 7 Diagram Simulink

Halaman ini sengaja dikosongkan

BAB 4 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 Persamaan (4.1). Sudut ϕ , θ dan ψ dijaga agar kembali pada titik ekuilibriumnya (0^0), sehingga *Quadrotor* mampu terbang dengan stabil.

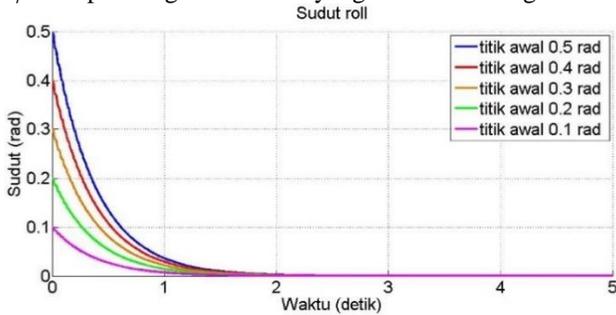
$$\begin{aligned} K_{roll} &= [5,4 \quad 2,0567] \\ K_{pitch} &= [5,4 \quad 1,7497] \\ K_{yaw} &= [0,00046 \quad 0,00014907] \end{aligned} \tag{4.1}$$

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.

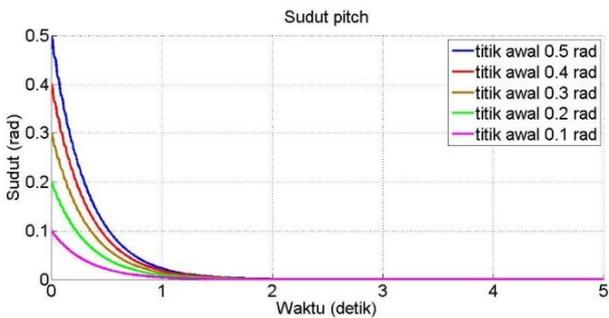
Pada perancangan awal, kontroler dirancang untuk mengembalikan sudut ϕ , θ dan ψ kembali ke titik nol dengan waktu naik 0,3 detik. Sehingga nilai ω dapat dicari dengan menggunakan Persamaan (4.2).

$$\omega = \frac{\pi - \beta}{t_r} \tag{4.2}$$

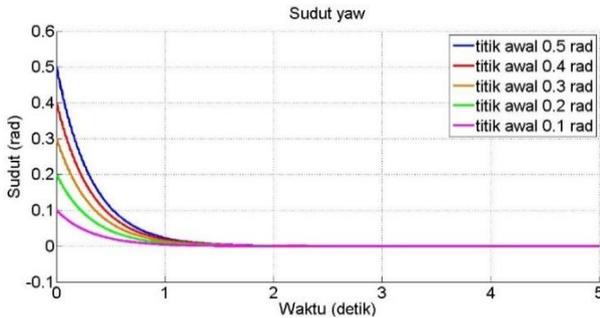
dimana $\pi - \beta = 1,8$ sedangkan nilai dari *overshoot* maksimum adalah 0,05 detik. Dari hasil simulasi, respon mencapai waktu naik pada waktu 2,3 detik dengan *error (Ess)* 0% untuk sudut ϕ . Hasil yang sama juga didapatkan untuk sudut θ dan ψ . Hal ini menandakan bahwa respon sudut mengalami keterlambatan sekitar dua 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

Tujuan dari pengujian ini adalah untuk mengetahui apakah kontroler sudut *roll*, *pitch*, dan *yaw* meregulasi ke titik awal. Jika kontroler tidak dapat meregulasi ketitik awal, maka *Quadrotor* akan melakukan gerak translasi secara terus menerus dan menyebabkan *Quadrotor* keluar dari lintasan *trajectory* yang telah ditentukan. Pada penelitian ini, pengendalian gerak rotasi *Quadrotor* menggunakan metode *ackerman's formula* untuk mendapatkan *gain state feedback* agar respon sistem dapat kembali ke titik awal.

Dari hasil yang ditampilkan pada Gambar 4.1 hingga 4.3 sudut *roll*, *pitch*, dan *yaw* dengan pemberian referensi dan nilai awal yang berbeda-beda akan kembali pada sudut 0^0 dengan pengujian titik awal yang berbeda-beda yaitu dari 0,5 rad hingga 0,1 rad.

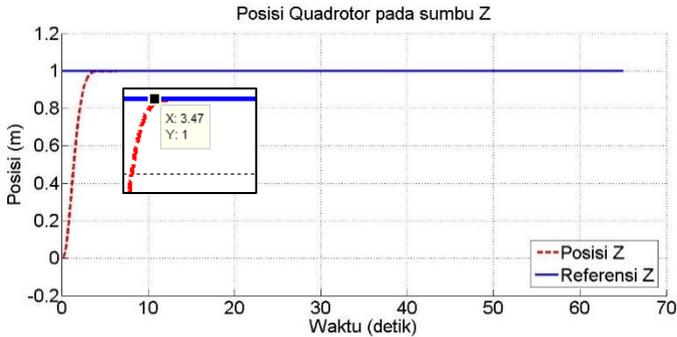
4.2 Pengujian *Tracking Quadrotor*

Setelah stabilisasi sudut ϕ , θ dan ψ tercapai, maka dapat diterapkan kontrol *tracking* pada *Quadrotor*. Model *Fuzzy Takagi-Sugeno* digunakan untuk merepresentasikan *tracking* posisi X dan Y. *Gain state feedback* pada *altitude*, *tracking* posisi X dan Y *Quadrotor* diperoleh menggunakan LQT. Dengan penerapan kontrol tersebut *Quadrotor* diharapkan mampu terbang dan bergerak mengikuti lintasan *tracking* yang diinginkan.

4.2.1 Pengujian *Altitude Quadrotor*

Pada perancangan kontrol ketinggian, *Quadrotor* diatur agar terbang pada ketinggian 1 meter diatas 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* (E_{ss}) 0% seperti pada Gambar 4.4. Hal ini menunjukkan bahwa respon memiliki keterlambatan waktu untuk mencapai nilai referensi.

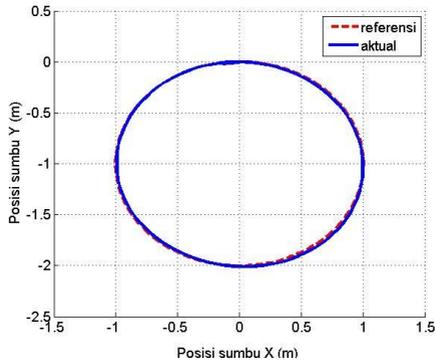


Gambar 4. 4 Posisi Z *Quadrotor*

Dari hasil simulasi tersebut dapat dilihat bahwa posisi Z naik dari ketinggian 0 meter menuju referensi 1 meter dengan waktu *steady state* 3,47 detik. Setelah ketinggian tercapai kontroler akan mempertahankan ketinggian *Quadrotor*. Kontroler yang dirancang mampu mempertahankan posisi Z dengan *Quadrotor* bergerak pada sumbu X, Y, hal ini terlihat bahwa dari detik ke 3,47 sampai ke 65 posisi Z tetap berada pada ketinggian 1 meter.

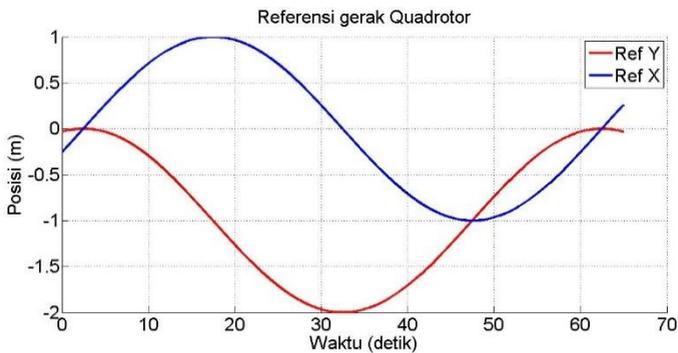
4.2.2 Uji Gerak Sumbu X,Y *Quadrotor*

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 *sinus* dan *cosinus* yang mempunyai amplitudo 1 meter yang terdapat pada Gambar 4.6. Simulasi dilakukan dengan total waktu 62 detik.



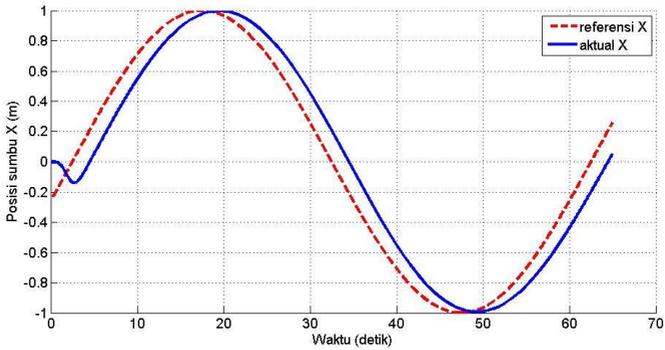
Gambar 4. 5 *Tracking Sumbu X dan Y Quadrotor*

Lintasan terbang *Quadrotor* yang diharapkan adalah berupa lingkaran dengan jari-jari sebesar 1 m.

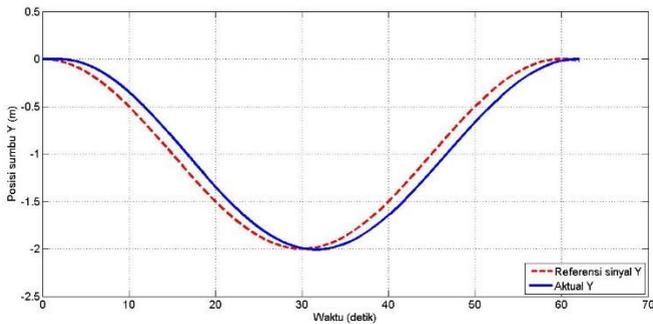


Gambar 4. 6 *Referensi Tracking untuk Quadrotor*

Gambar 4.6 menampilkan sinyal referensi yang diberikan pada sumbu X dan Y. Pada sumbu X diberikan sinyal *sinus* dengan amplitude sebesar 1 m, sedangkan pada sumbu Y diberikan sinyal *cosinus* dengan amplitude sebesar 1 m.

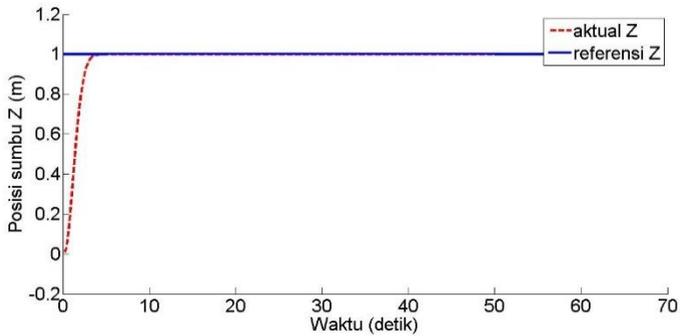


Gambar 4.7 Respon *Tracking* Posisi Sumbu X



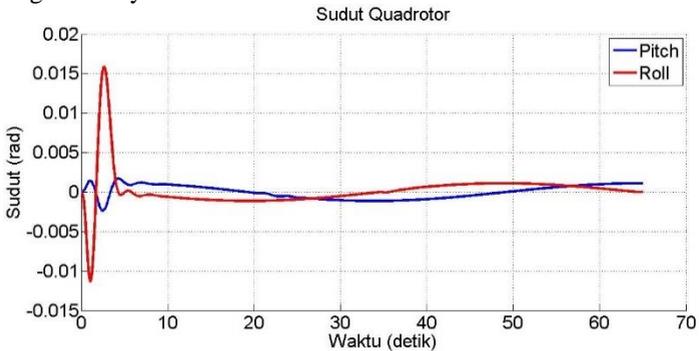
Gambar 4.8 Respon *Tracking* Posisi Sumbu Y

Respon posisi translasi pada sumbu X dan Y ditunjukkan pada Gambar 4.7 dan 4.8. Respon dapat mengikuti sinyal referensi lintasan yang diberikan dengan adanya keterlambatan respon sebesar 1,86 detik. Terdapat penyimpangan lintasan antara lintasan referensi dengan lintasan dari *Quadrotor* sebesar 0,024 m.

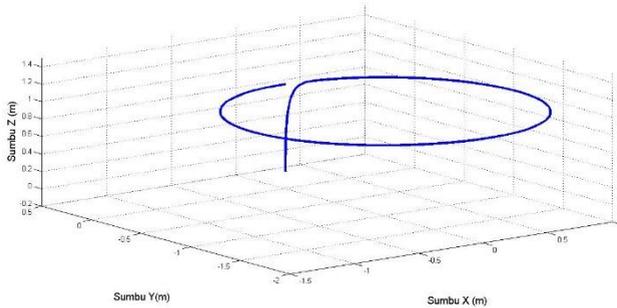


Gambar 4. 9 Respon *Tracking* Posisi Sumbu Z

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.



Gambar 4. 10 Sudut *Roll* dan *Pitch* untuk Gerak *Quadrotor*

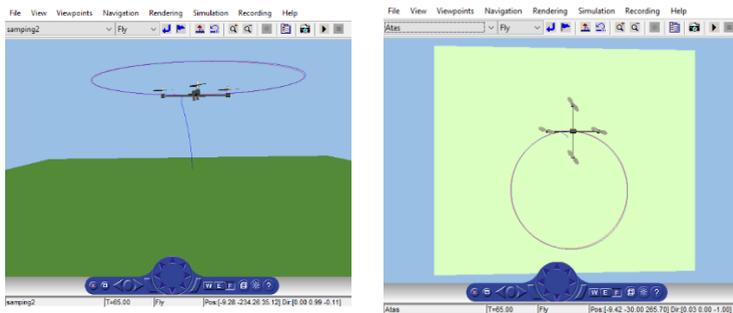


Gambar 4. 11 Plot 3D Hasil Simulasi

Dari hasil simulasi dapat diketahui bahwa *Quadrotor* mampu bergerak mengikuti referensi *tracking* yang diinginkan dengan perubahan. Hal ini menunjukkan bahwa *Quadrotor* dapat bergerak dengan stabil. Pada Tabel 4.1 merupakan hasil perhitungan RMSE dari sumbu X dan Y tanpa gangguan.

Tabel 4. 1 Error pada Sumbu X dan Y Tanpa Gangguan

| Sumbu | Error | RMSE |
|-------|----------|--------|
| X | 0,0072 m | 0,1470 |
| Y | 0,035 m | 0,1225 |



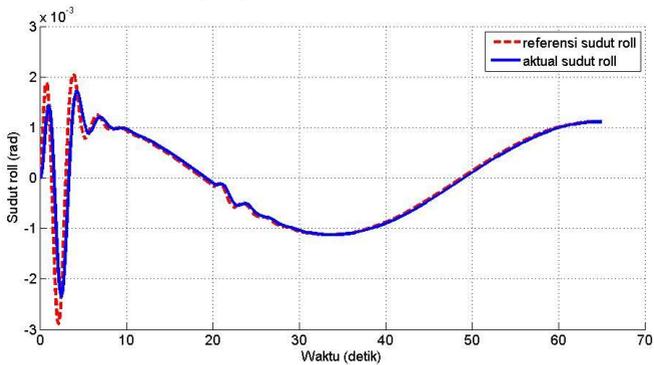
Gambar 4. 12 Screenshot dari *virtual world Quadrotor*

Untuk memperjelas hasil simulasi, dibuat pula sebuah *virtual world* yang dapat mempresentasikan dinamika gerak rotasi dan translasi dari *Quadrotor* yang disajikan dalam bentuk tiga dimensi.

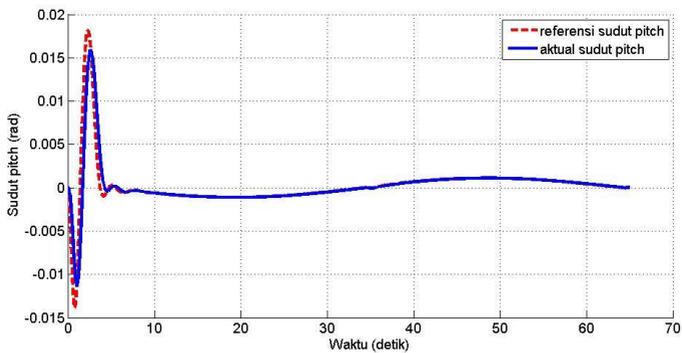
Gambar 4.8 merupakan *screenshot* dari *virtual world Quadrotor* ketika simulasi sedang berlangsung. Garis berwarna merah merupakan sinyal referensi yang diberikan. Sedangkan garis berwarna biru merupakan jejak lintasan yang diallui *Quadrotor*. Pada *virtual world* ini terdapat dua buah sudut pandang yaitu tampak atas dan tampak samping.

4.2.2 Pengujian Respon Sistem untuk Gerak Rotasi tanpa Gangguan

Hasil pengujian gerak rotasi pada sudut *roll* dan *pitch* saat dilakukan *tracking* terdapat pada Gambar 4.13 dan 4.14.



Gambar 4. 13 Respon Sudut *Roll* Tanpa Gangguan



Gambar 4. 14 Respon Sudut *Pitch* Tanpa Gangguan

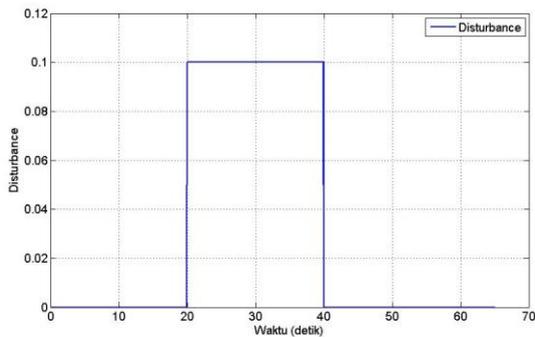
Dapat dilihat bahwa respon sudut *roll* dan *pitch* dapat mengikuti sinyal referensi yang diberikan dari output kontroler posisi. Dapat dilihat bahwa respon susur *roll* dan *pitch* dapat mengikuti sinyal referensi yang diberikan dari *output* kontroler posisi, dengan rata-rata *error* sebesar 0,000083772 rad.

4.2.3 Simulasi Gerak *Quadrotor* dengan Gangguan

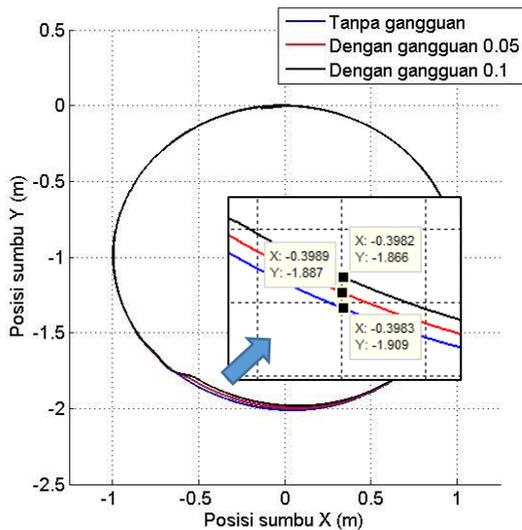
Untuk menguji ketahanan sistem kontrol yang dirancang maka dilakukan simulasi sistem kontrol dengan pemberian gangguan. Gangguan yang diberikan memiliki dua jenis yaitu gangguan pada *plant* dan pada pengukuran berupa *noise*.

4.2.3.1 Simulasi Gerak *Quadrotor* dengan Gangguan pada *Input Plant*

Gangguan pada *input plant* yang diberikan berupa sinyal *step* sebesar 0,05 dan 0,1. Sinyal tersebut diberikan pada masukan kecepatan *plant* dari *Quadrotor*. Sinyal *step* yang diberikan selama 20 detik yaitu dari detik ke-20 hingga detik ke-40. Bentuk sinyal *step* yang diberikan terdapat pada Gambar 4.15. Untuk respon *tracking* sumbu X dan Y terdapat pada Gambar 4.16-4.17.

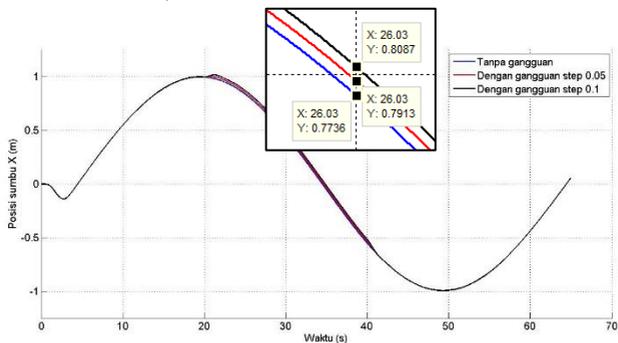


Gambar 4. 15 *Disturbance*



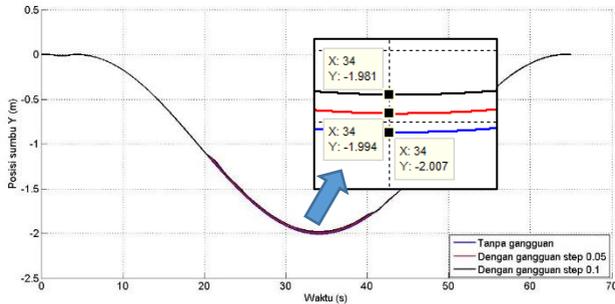
Gambar 4. 15 Tracking dengan Gangguan *Disturbance*

Pada Gambar 4.15 dapat dilihat bahwa *tracking Quadrotor* dengan adanya gangguan sebesar 0,05 terdapat perbedaan pada sumbu Y sebesar 0,022 m. Sedangkan apabila diberikan sinyal *step* sebesar 0,1, maka terdapat perbedaan dengan sinyal *tracking Quadrotor* sebesar 0,043 m.



Gambar 4. 16 Respon Sumbu X dengan Gangguan Sinyal *Step*

Gambar 4.16 menampilkan sumbu X yang diberikan gangguan berupa sinyal *step*. Terlihat bahwa terdapat perbedaan lintasan *tracking Quadrotor* saat diberikan gangguan sinyal *step* sebesar 0,05 sebesar 0,0177 m. Sedangkan perbedaan yang terjadi ketika diberikan sinyal *step* sebesar 0,1 adalah 0,0351 m.



Gambar 4. 17 Respon Sumbu Y dengan Gangguan Sinyal *Step*

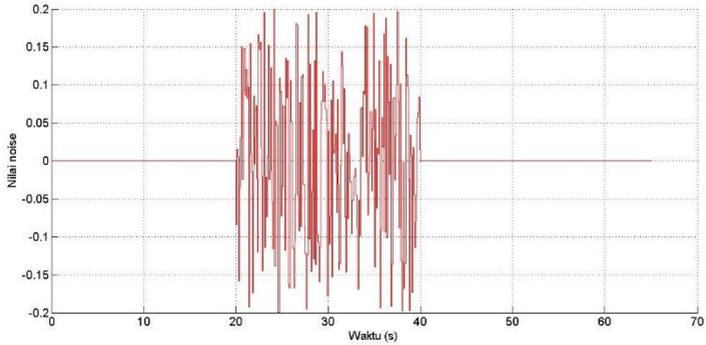
Pada Gambar 4.17 terlihat terdapat perbedaan lintasan *tracking Quadrotor* pada sumbu Y saat diberikan gangguan sinyal *step* sebesar 0,05 sebesar 0,013 m. Sedangkan perbedaan yang terjadi ketika diberikan sinyal *step* sebesar 0,1 adalah 0,026 m.

Tabel 4. 2 Error sumbu X dan Y dengan *disturbance*

| Sumbu | <i>Disturbance</i> | RMSE |
|-------|--------------------|--------|
| X | 0,05 | 0,1532 |
| | 0,1 | 0,1598 |
| Y | 0,05 | 0,1241 |
| | 0,1 | 0,1261 |

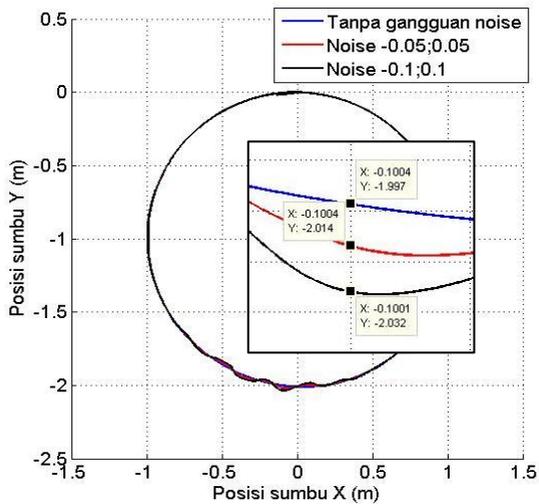
4.2.4 Simulasi Gerak *Quadrotor* dengan Gangguan Pengukuran

Gangguan pada pengukuran berupa nilai *random* dengan nilai maksimum 0,05 dan 0,1. Sedangkan nilai minimum yang diberikan adalah -0,05 dan -0,1. Gangguan diibaratkan sebagai kecepatan angin yang diberikan untuk mengganggu kecepatan *Quadrotor*.



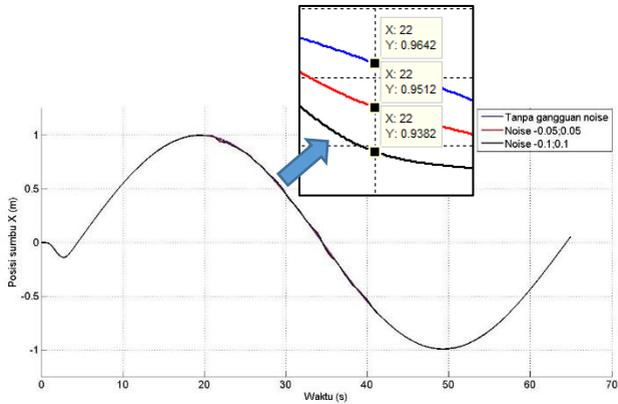
Gambar 4.16 *Noise yang Diberikan pada Quadrotor*

Hasil pengujian dengan gangguan ada trayektori gerak translasi sumbu X dan Y pada Gambar 4.7.



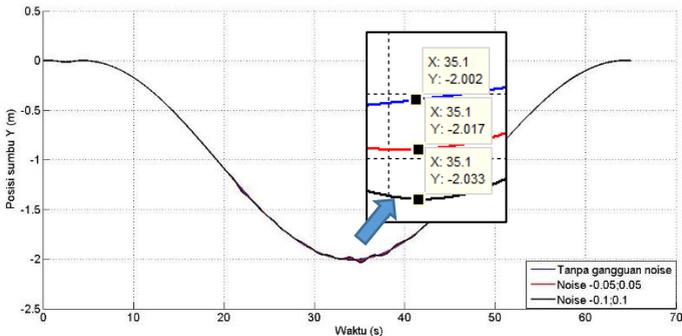
Gambar 4.17 *Kesalahan Tracking Gerak Quadrotor*

Gambar 4.21 menampilkan hasil respon dari sistem setelah diberikan sinyal *noise*. Terdapat perbedaan pada *tracking* antara sinyal referensi dan juga pergerakan dari *Quadrotor*.



Gambar 4. 18 Respon Posisi Sumbu X dengan Gangguan *Noise*

Gambar 4.18 menampilkan perbedaan lintasan dari *Quadrotor* setelah diberi gangguan berupa sinyal *random number*. Untuk nilai *random number* dengan nilai minimum; nilai maximum $(-0,05;0,05)$ terdapat perbedaan sebesar 0,013 pada detik ke-22. Sedangkan untuk nilai $-0,1;0,1$ terdapat perbedaan sebesar 0,026 pada detik ke-22.



Gambar 4. 19 Respon Posisi Sumbu Y dengan Gangguan *Noise*

Gambar 4.19 menampilkan perbedaan lintasan pada sumbu Y. Jika besar gangguan *random number* yang diberikan sebesar pada *output* kecepatan *Quadrotor* sebesar $-0,05;0,05$ maka terdapat

perbedaan lintasan sebesar 0,015 m. Sedangkan untuk gangguan sebesar $-0,1;0,1$, besar perbedaannya adalah 0,031 m.

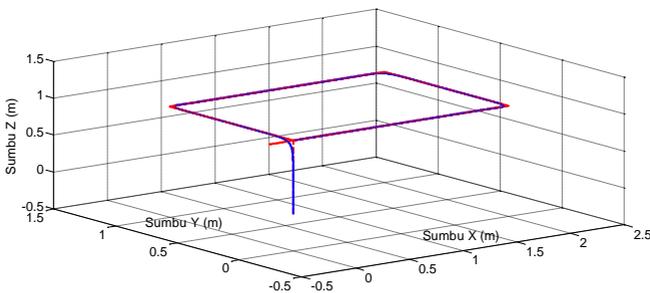
Pada Tabel 4.2 ditampilkan hasil perhitungan *error* dari pemberian *noise* yang berbeda.

Tabel 4.3 Error Sumbu X dan Y dengan Noise

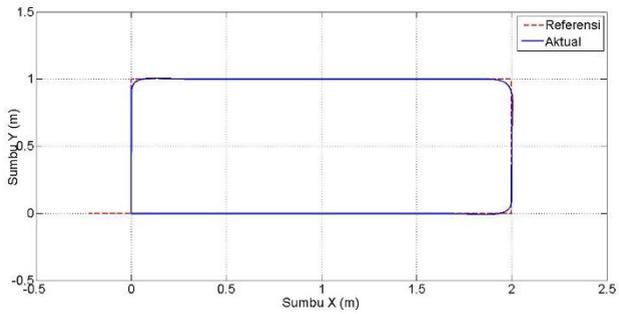
| Sumbu | Noise (max;min) | RMSE |
|-------|-----------------|--------|
| X | 0,05;0,05 | 0,1473 |
| | 0,1;0,1 | 0,1482 |
| Y | 0,05;0,05 | 0,1225 |
| | 0,1;0,1 | 0,1232 |

4.3 Simulasi Sinyal Referensi yang Berbeda-beda

Pengujian dilakukan dengan memberikan sinyal referensi yang berbeda-beda terhadap sistem kontrol yang telah didesain tanpa mengubah parameter kontrol apapun. Seperti pada Gambar 4.55, referensi yang diberikan berupa sinyal *step*. Hasilnya, sistem dapat melakukan *tracking* namun terdapat penyimpangan posisi pada sumbu X sebesar 0,025 m dan pada sumbu Y sebesar 0,029. Hal ini dikarenakan sistem tidak dapat berbelok dengan ketajaman sudut 90^0 , dapat dilihat pada Gambar 4.55 bahwa penyimpangan hanya terjadi pada saat *Quadrotor* harus berbelok mengikuti referensi yang diberikan.



Gambar 4. 18 Tracking Posisi dengan Lintasan Berbentuk Persegi



Gambar 4. 19 Posisi X dan Y pada saat *Tracking* Lintasan Berbentuk Persegi

BAB 5

PENUTUP

5.1 Kesimpulan

Metode kontrol *Fuzzy Takagi-Sugeno* bekerja dengan baik untuk mengendalikan gerak *Quadrotor* mengikuti lintasan yang diinginkan dengan adanya waktu tunda yang cukup besar, tetapi *Quadrotor* memiliki penyimpangan lintasan yang lebih kecil. Dengan nilai *Root Mean Square Error* (RMSE) sebesar 0,1470 pada sumbu X dan 0,1225 pada sumbu Y. Terdapat waktu tunda sebesar 1,86 detik antara lintasan referensi dengan gerak *Quadrotor*. Penyimpangan gerak *Quadrotor* terbesar adalah 0,02 m.

5.2 Saran

Pemodelan dan pemahan tentang *Quadrotor* akan membantu dalam perancangan sistem yang lebih baik dan mendapatkan model matematika yang lebih akurat.

Penggunaan metode LQT perlu adanya perbaikan untuk menghilangkan *delay* pada respon.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] D. Sohendri, “Kontrol Tracking Fuzzy Berbasis Peforma Robust untuk Quadrotor,” *Tugas Akhir. Teknik Elektro*. Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [2] B. Castillo-Toledo, S.D. Gennaro and F. Jurado, “Trajectory Tracking for a Quadrotor via Fuzzy Regulation,” in *World Automation Congress (WAC)*, Mexico, 2012.
- [3] M. Brown, “*How RC Quad Copters Work*”. [Online]. Available: www.hooked-on-rc-airplanes.com/rc-quad-copters.html, 2007.
- [4] Tommaso Bresciani, “Modelling, Identification and Control of a Quadrotor Helicopter”. *Thesis*, Department of Automatic Control Lund University, 2008.
- [5] F. L. Lewis and V. L. Syrmos, *Optimal Control Second Edition*, Toronto: A Wiley-Interscience Publication, 1995.
- [6] F. Isdaryani, “Kontrol Tracking untuk Quadrotor Menggunakan Metode Optimal Output Feedback dengan Command-Generator Tracker,” *Tesis. Teknik Elektro*. Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [7] K. Ogata, *Modern Control Engineering 3th Edition*, Minnesota: Prentice-Hall Intenational, 1997.

Halaman ini sengaja dikosongkan

LAMPIRAN A

A.1 Pemodelan *Quadrotor*

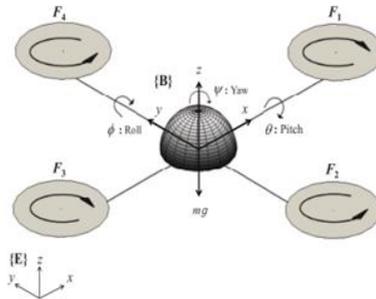
Pada bagian ini akan dijelaskan analisis tentang penurunan persamaan matematika dari *Quadrotor*, dimana analisis ini diperoleh dari analisis kinematika dan dinamika dari *Quadrotor*. Hasil penurunan persamaan model matematika ini akan menggambarkan semua pergerakan *Quadrotor* dan dari persamaan yang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

A.1.1 Koordinat *Frame*

Quadrotor memiliki 6 *degree of freedom* (DoF) dengan 12 keluaran, 6 keluaran dari 12 keluaran ini menentukan *altitude* dari *Quadrotor*. Model kinematika dan dinamika *Quadrotor* diturunkan berdasarkan Newton-Euler.

Untuk menentukan acuan-acuan dinamika pada *Quadrotor* perlu 2 sistem koordinat yang dibutuhkan untuk mendefinisikan keadaan *Quadrotor*. Pertama sumbu X_B, Y_B dan Z_B yang terletak pada titik gravitasi (COG) *Quadrotor*.

Kedua, sistem koordinat yang tetap yaitu pusat bumi (X_E, Y_E dan Z_E). Sehingga terdapat 2 *frame*, yaitu badan *frame* (*frame* badan *Quadrotor*) dan *earth frame* (*frame* bumi) dan sudut rotasi roll (ϕ), pitch (θ), yaw (ψ).



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

Posisi *linear Quadrotor* ditentukan dari koordinat vektor antara origin *frame body* serta origin *frame* bumi. Posisi linier dan posisi angular terdapat pada Persamaan

$$v = [X \ Y \ Z]^T$$

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

Sedangkan enam derajat kebebasan *Quadrotor* terdapat pada tiga buah translasi $V = [u \ v \ w]^T$ dan tiga buah rotasi $\Omega = [p \ q \ r]^T$.

A.1.2 Asumsi Pemodelan *Quadrotor*

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

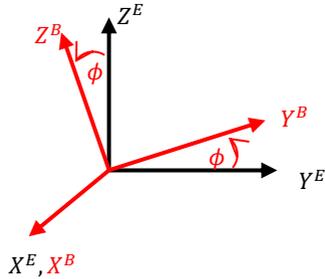
- a) Bumi dianggap datar, tidak berotasi dan sesuai dengan acuan inersia dari *frame*.
- b) Percepatan gravitasi dianggap konstan dan tegak lurus terhadap permukaan bumi.
- c) Desain pemodelan *Quadrotor* yang dibuat, dianggap simetris dengan asumsi sumbu yang ditentukan sebelumnya.
- d) Badan serta 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*.

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).

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

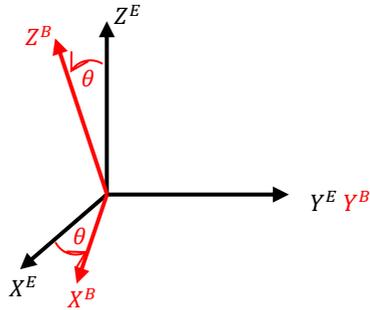
Persamaan (A.1) dapat diubah dalam bentuk matriks seperti (A.2) dan (A.3).

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

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

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).

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

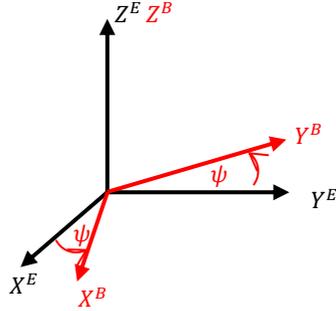
Persamaan (A.4) dapat diubah dalam bentuk matriks seperti Persamaan (A.5) dan (A.6).

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

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

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

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

$$R_v^{BE} = \begin{bmatrix} c\theta c\psi & s\psi c\theta & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta c\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta c\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \quad (\text{A.10})$$

Selain matriks tranformasi R_v^{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*.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = T_\Theta^{BE-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (\text{A.11})$$

Dengan menyelesaikan (A.11) maka didapatkan matriks transformasi T_Θ^{BE} .

$$T_\Theta^{BE} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \quad (\text{A.12})$$

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

A.1.3 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:

$$\begin{bmatrix} F \text{ trust} \\ F \text{ roll} \\ F \text{ pitch} \\ F \text{ yaw} \end{bmatrix} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} F_1 + F_2 + F_3 + F_4 \\ F_2 - F_4 \\ F_1 - F_3 \\ F_1 - F_2 + F_3 - F_4 \end{bmatrix} \quad (\text{A.13})$$

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

$$F_i = K \frac{\omega}{s + \omega} u_i \quad (\text{A.14})$$

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 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.1.3.1 Penurunan Gerak Translasi

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

$$\begin{aligned} \sum F_i = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} &= \frac{d}{dt}(mV) = m \frac{d}{dt} \vec{V} = m\ddot{x} = m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \\ \sum F^E &= m\ddot{v}^E \end{aligned} \quad (\text{A.15})$$

dengan ΣF^E merupakan resultan gaya yang bekerja pada *Quadrotor*, m merupakan total massa *Quadrotor* dan \ddot{v}^E merupakan percepatan gerak *Quadrotor*. Sehingga (A.15) dapat dirubah menjadi (A.16)

$$U_1 + F_g Z_E = m\ddot{v}^E \quad (\text{A.16})$$

dengan $v = [x \ y \ z]^T$ merupakan posisi *Quadrotor* terhadap bumi, U_1 merupakan resultan gaya angkat, sedangkan gaya gravitasi dinyatakan oleh $F_g = -mg$ dan $Z_B = [0 \ 0 \ 1]^T$. Persamaan (A.16) ditulis karena gaya pada *Quadrotor* hanya terjadi pada sumbu Z_B (pada kondisi hover). Pemodelan sumbu translasi dari koordinat *frame body* ke koordinat *frame* bumi diperlukan matriks rotasi seperti pada Persamaan (A.10). Sehingga Persamaan gerak translasi *Quadrotor* dinyatakan sebagai berikut.

$$\begin{aligned}
U_1 R^{EB} + F_g Z_E &= m \ddot{v}^E \\
\frac{U_1}{m} R^{EB} - g Z_E &= \ddot{v}^E
\end{aligned} \tag{A.17}$$

$$\begin{aligned}
\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} &= \frac{U_1}{m} \begin{bmatrix} c\theta c\psi & -c\phi s\psi - s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi s\theta & c\phi c\theta \end{bmatrix} \begin{bmatrix} X^B \\ Y^B \\ Z^B \end{bmatrix} \\
&\quad - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}
\end{aligned} \tag{A.18}$$

Karena resultan gaya yang bekerja pada sumbu Z maka resultan gaya yang terjadi hanyalah F_Z .

$$\begin{aligned}
\ddot{x} &= \frac{U_1}{m} (\sin\phi \sin\psi + \cos\phi \sin\theta \cos\psi) \\
\ddot{y} &= \frac{U_1}{m} (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) \\
\ddot{y} &= \frac{U_1}{m} (\cos\phi \cos\theta) - g
\end{aligned} \tag{A.19}$$

A.1.3.2 Penurunan Gerak Rotasi

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

$$\Sigma^E \tau = J^E \ddot{\Theta} \tag{A.20}$$

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 Persamaan (A.22).

$$\begin{aligned}
T_{\Theta} \tau^B &= J (T_{\Theta} \dot{\omega}^B) \\
T_{\Theta} \tau^B &= J (\dot{T}_{\Theta} \omega^B + T_{\Theta} \dot{\omega}^B)
\end{aligned} \tag{A.21}$$

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.21) dapat diubah menjadi Persamaan (A.22).

$$\begin{aligned}
T_{\Theta}\tau^B &= J(T_{\Theta}S(\omega^B)\omega^B + T_{\Theta}\dot{\omega}^B) \\
T_{\Theta}\tau^B &= T_{\Theta}(S(\omega^B)J\omega^B + J\dot{\omega}^B) \\
T_{\Theta}\tau^B &= T_{\Theta}(\omega^B \times J\omega^B + J\dot{\omega}^B) \tag{A.22}
\end{aligned}$$

Karena pada kedua ruas terdapat matriks transformasi T_{Θ} , maka dapat dihilangkan atau Persamaan (A.22) dapat dianggap sudah ditransformasikan ke dalam koordinat B -frame.

$$\tau^B = \omega^B \times J\omega^B + J\dot{\omega}^B \tag{A.23}$$

Persamaan (A.23) dapat disederhanakan menjadi (A.24).

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

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.

$$\begin{aligned}
-(\omega^B \times J\omega^B) &= J\omega^B \times \omega^B \\
&= \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} (J_{yy} - J_{zz})qr \\ (J_{zz} - J_{xx})pr \\ (J_{xx} - J_{yy})pq \end{bmatrix} \\
&\tag{A.25}
\end{aligned}$$

Dengan menyelesaikan (A.23) maka diperoleh Persamaan gerak rotasi *Quadrotor* seperti pada (A.26)

$$J\dot{\omega}^B = -(\omega^B \times J\omega^B) + \tau^B$$

$$\begin{aligned}
\begin{bmatrix} J_{xx}\dot{p} \\ J_{yy}\dot{q} \\ J_{zz}\dot{r} \end{bmatrix} &= \begin{bmatrix} (J_{yy} - J_{zz})qr \\ (J_{zz} - J_{xx})pr \\ (J_{xx} - J_{yy})pq \end{bmatrix} + \begin{bmatrix} U_2l \\ U_3l \\ U_4d \end{bmatrix} \\
\left\{ \begin{aligned} \dot{p} &= \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{U_2l}{J_{xx}} \\ \dot{q} &= \frac{J_{zz} - J_{xx}}{J_{yy}} pr + \frac{U_3l}{J_{yy}} \\ \dot{r} &= \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4d}{J_{zz}} \end{aligned} \right. & \quad (A.26)
\end{aligned}$$

Dengan mengambil Persamaan gerak translasi (A.19) dan Persamaan gerak rotasi (A.26), maka keseluruhan model dinamika *Quadrotor* dapat dinyatakan sebagai

$$\left\{ \begin{aligned} \ddot{X} &= (\sin\psi \sin\phi + \cos\psi \sin\theta \cos\phi) \frac{U_1}{m} \\ \ddot{Y} &= (-\cos\psi \sin\phi + \sin\psi \sin\theta \cos\phi) \frac{U_1}{m} \\ \ddot{Z} &= -g + (\cos\theta \cos\phi) \frac{U_1}{m} \\ \dot{p} &= \frac{J_{yy} - J_{zz}}{J_{xx}} qr + \frac{U_2l}{J_{xx}} \\ \dot{q} &= \frac{J_{zz} - J_{xx}}{J_{yy}} pr + \frac{U_3l}{J_{yy}} \\ \dot{r} &= \frac{J_{xx} - J_{yy}}{J_{zz}} pq + \frac{U_4d}{J_{zz}} \end{aligned} \right. \quad (A.27)$$

A.2 Linearisasi

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:

$$\begin{aligned}
x_1 &= X & x_7 &= \phi \\
x_2 &= \dot{X} & x_8 &= \dot{\phi} \\
x_3 &= Y & x_9 &= \theta \\
x_4 &= \dot{Y} & x_{10} &= \dot{\theta} \\
x_5 &= Z & x_{11} &= \psi \\
x_6 &= \dot{Z} & x_{12} &= \dot{\psi}
\end{aligned} \tag{A.28}$$

Persamaan (A.28) jika diubah menjadi bentuk *state-space* maka terdapat pada Persamaan (A.29).

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ \dot{Y} \\ \ddot{Y} \\ \dot{Z} \\ \ddot{Z} \\ \dot{\phi} \\ \ddot{\phi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} x_2 \\ (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \frac{U_1}{m} \\ x_4 \\ (\cos \phi \sin \theta \sin \psi + \sin \phi \cos \psi) \frac{U_1}{m} \\ x_6 \\ (-g + \cos \phi \cos \theta) \frac{U_1}{m} \\ x_8 \\ \frac{I_{yy} - I_{zz}}{I_{xx}} qr + \frac{U_2 l}{I_{xx}} \\ x_{10} \\ \frac{I_{zz} - I_{xx}}{I_{yy}} pr + \frac{U_3 l}{I_{yy}} \\ x_{12} \\ \frac{I_{xx} - I_{yy}}{I_{zz}} pq + \frac{U_4 d}{I_{zz}} \end{bmatrix} \tag{A.29}$$

Linearisasi pada matriks A dapat dilakukan dengan menggunakan matriks Jacobian seperti pada Persamaan (A.30).

$$A = \frac{df(x,u)}{dx} = \begin{bmatrix} \frac{df_1(x,u)}{dx_1} & \dots & \frac{df_1(x,u)}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n(x,u)}{dx_1} & \dots & \frac{df_n(x,u)}{dx_n} \end{bmatrix} \tag{A.30}$$

Sehingga linearisasi yang dilakukan pada *state space* adalah sebagai berikut

$$\begin{aligned}
\frac{df_1(x,u)}{dx} &= [0 \quad 1 \quad 0 \quad 0] \\
\frac{df_2(x,u)}{dx} &= [0 \quad 0 \quad 0]
\end{aligned} \tag{A.31}$$

$$\begin{aligned}
\frac{df_3(x,u)}{dx} &= [0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_4(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_5(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_6(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_7(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_8(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_9(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0] \\
\frac{df_{10}(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] \\
\frac{df_{11}(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1] \\
\frac{df_{12}(x,u)}{dx} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]
\end{aligned}$$

Linearisasi pada matriks B dilakukan pada titik kerja 0^0 untuk mengatur sudut *roll*, *pitch*, dan *yaw* serta *tracking* pada sumbu X dan Y . Linearisasi pada titik kerja 0^0 terdapat pada Persamaan (A.32).

$$B = \frac{df(x,u)}{du} = \begin{bmatrix} \frac{df_1(x,u)}{du_1} & \dots & \frac{df_1(x,u)}{du_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n(x,u)}{du_1} & \dots & \frac{df_n(x,u)}{du_n} \end{bmatrix}$$

$$\begin{aligned}
\frac{df_1(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0] \\
\frac{df_2(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0] \\
\frac{df_3(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0] \\
\frac{df_4(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0] \\
\frac{df_5(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0] \\
\frac{df_6(x,u)}{du_i} &= \begin{bmatrix} -gU_1 \\ m \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \\
\frac{df_7(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0] \\
\frac{df_8(x,u)}{du_i} &= \begin{bmatrix} 0 & \frac{U_2 l}{I_{xx}} & 0 & 0 \end{bmatrix} \\
\frac{df_9(x,u)}{du_i} &= [0 \ 0 \ 0 \ 0]
\end{aligned}
\tag{A.32}$$

$$\begin{aligned}\frac{df_{10}(x,u)}{du_i} &= \begin{bmatrix} 0 & 0 & \frac{U_3 l}{I_{yy}} & 0 \end{bmatrix} \\ \frac{df_{11}(x,u)}{du_i} &= \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \\ \frac{df_{12}(x,u)}{du_i} &= \begin{bmatrix} 0 & 0 & 0 & \frac{U_4 d}{I_{zz}} \end{bmatrix}\end{aligned}$$

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

%Linierisasi dititik 0
[A0,B0,C0,D0]=linmod('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
tr1=0.3; %second
omega1=1.8/tr1; %menggunakan ts(2%)=4*tao
Mp1=0.05;
zetal=sqrt((log(Mp1))^2/(log(Mp1)+pi^2));

%Roll
Aroll=A0([7:8],[7:8]);
Broll=B0([7:8],2);
num1=[omega1^2];
den1=[1 2*zetal*omega1 omega1^2];
Hs1=tf(num1,den1);
[z,p,k]=tf2zp(num1,den1);
alpha1=Aroll^2+den1(2)*Aroll+den1(3)*(eye(2,2)
);
M=[Broll Aroll*Broll];
J=[p(1) 0;0 p(2)];
JJ=poly(J);
```

```

Phi=polyvalm(JJ,Aroll);
Kroll=[0 1]*(inv(M))*Phi;

%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem pitch
tr2=0.3; %second
omega2=1.8/tr2; %menggunakan ts(2%)=4*tao
Mp2=0.05;
zeta2=sqrt((log(Mp2/pi))^2/(1+(log((Mp2)/pi))^2));

%Pitch
Apitch=A0([9:10],[9:10]);
Bpitch=B0([9:10],3);
num2=[omega2^2];
den2=[1 2*zeta2*omega2 omega2^2];
Hs2=tf(num2,den2);
[z,p,k]=tf2zp(num2,den2);
alpha2=Apitch^2+den2(2)*Apitch+den2(3)*(eye(2,2));
Kpitch=[0 1]*inv([Bpitch
Apitch*Bpitch])*alpha2;

%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem yaw
tr2=0.3; %second
omega3=1.8/tr2; %menggunakan ts(2%)=4*tao
Mp3=0.05;
zeta3=sqrt((log(Mp3/pi))^2/(1+(log((Mp3)/pi))^2));

%Yaw
Ayaw=A0([11:12],[11:12]);
Byaw=B0([11:12],4);
num3=[omega3^2];
den3=[1 2*zeta3*omega3 omega3^2];
Hs3=tf(num3,den3);
[z,p,k]=tf2zp(num3,den3);

```

```

alpha3=Ayaw^2+den3(2)*Ayaw+den3(3)*(eye(2,2))
;
Kyaw=[0 1]*inv([Byaw Ayaw*Byaw])*alpha3;

%Stabilisasi Z
%Masukan karakteristik yang diinginkan utk
menentukan pole subsistem Z
%Sumbu Z
Az=[0 1;0 0];
Bz=[0 0.2858]';
Cz=[1 0];
Qz=[150];
Rz=1;
[Sz,oz,mz,nz]=care(Az,Bz,Cz'*Qz*Cz,Rz);
Kz=inv(Rz)*Bz'*Sz;
ACLz=(Az-Bz*Kz)';
Lz=inv(Rz)*Bz';

```

B.2 Program Kontrol *Tracking Quadrotor*

```

%Sumbu X
phi1=0;theta1=0;psi1=0;           %sekitar 0
A1=[0 1;0 0];
B1=[0;cos(phi1)*cos(theta1)*cos(psi1)*9.81];
C1=[1 0];
Q1=[1.8];R1=[2.88];
W1=C1'*Q1;
[S,o,m,n]=care(A1,B1,(C1)'*Q1*C1,R1);
K1=inv(R1)*B1'*S;
ACL1=(A1-B1*K1)';
L1=inv(R1)*B1';

phi2=10;theta2=pi/18;psi2=10;     %theta
sekitar 10 derajat
A2=[0 1;0 0];
B2=[0;cos(phi2)*cos(theta2)*cos(psi2)*9.81];
C2=[1 0];
Q2=[0.51];R2=[6];
[S2,o2,m2,n2]=care(A2,B2,(C2)'*Q2*C2,R2);
K2=inv(R2)*B2'*S2;

```

```
ACL2=(A2-B2*K2)';
L2=inv(R2)*B2';
```

```
%Sumbu Y
```

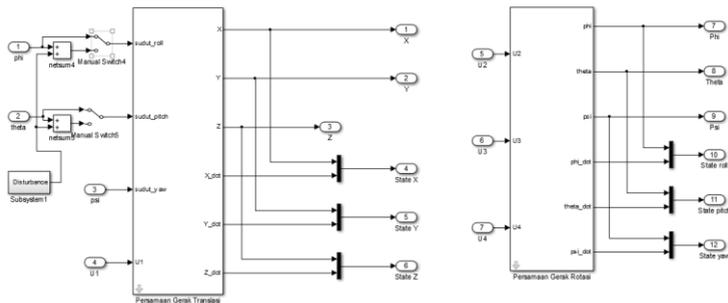
```
phi4=0;theta4=0;psi4=0; %sekitar 0
A4=[0 1;0 0]; B4=[0;-
cos(phi4)*cos(psi4)*9.81];
C4=[1 0];
Q4=[1.8];R4=[2.88];
[S4,o4,m4,n4]=care(A4,B4,(C4) '*Q4*C4,R4);
K4=inv(R4)*B4'*S4;
ACL4=(A4-B4*K4)';
L4=inv(R4)*B4';
```

```
phi5=pi/18;theta5=10;psi5=10; %theta
sekitar 10 derajat
```

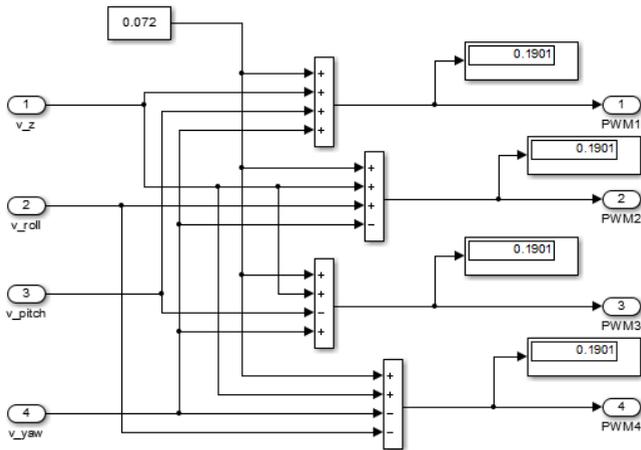
```
A5=[0 1;0 0]; B5=[0;-
cos(phi5)*cos(psi5)*9.81];
C5=[1 0];
Q5=[0.51];R5=[6];
[S5,o5,m5,n5]=care(A5,B5,(C5) '*Q5*C5,R5);
K5=inv(R5)*B5'*S5;
ACL5=(A5-B5*K5)';
L5=inv(R5)*B5';
```

B.3 Blok Diagram Simulink

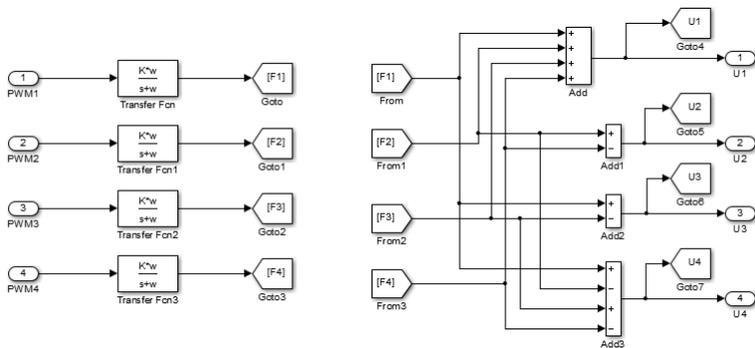
Blok Dinamika Gerak *Quadrotor*



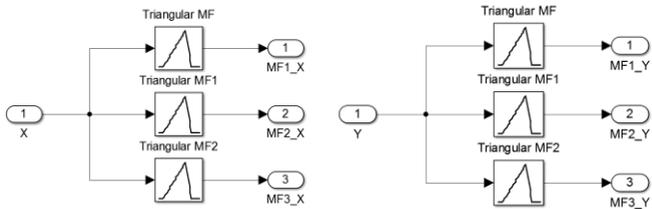
Blok Aktuator



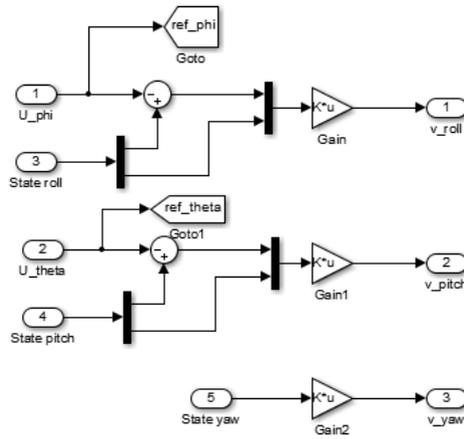
Blok PWM



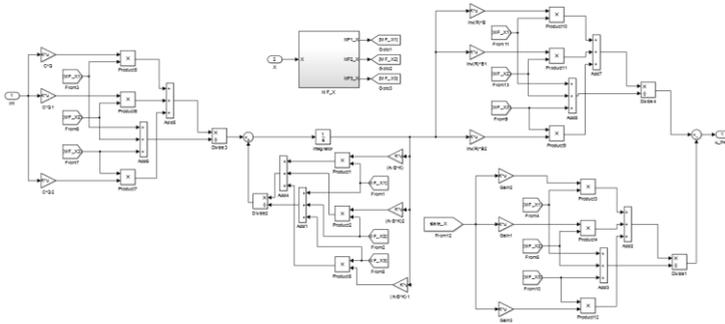
Blok Membership Function X dan Y

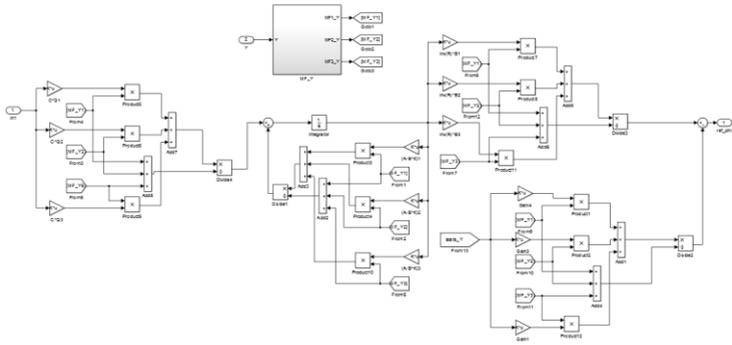


Blok Kontrol Stabilisasi



Blok Kontrol Tracking





BIOGRAFI PENULIS



Penulis bernama lengkap Imroatul Hudati, dilahirkan di Banjarnegara 15 Juni 1994. Penulis menempuh pendidikan di TK Pertiwi Karangobar pada tahun 1999, kemudian masuk SDN 1 Karangobar pada tahun 2000. Tahun 2006 penulis melanjutkan pendidikan di SMPN 1 Karangobar. Kemudian pada tahun 2009 masuk ke SMAN 1 Banjarnegara. Melanjtkan kuliah pada jenjang Diploma III di Diploma III Teknik Elektro Universitas Gadjah Mada angkatan 2012. Setelah lulus pada tahun 2015, penulis melanjutkan kuliah pada jenjang sarjana di Departemen Teknik Elektro Institut Teknologi Sepuluh Nopember pada bidang studi Teknik Sistem Pengaturan. Untuk menghubungi penulis dapat melalui email di imroatul.hudati15@gmail.com.