



TESIS-TE142599

**KONTROL *TRACKING* UNTUK QUADROTOR
MENGUNAKAN METODE OPTIMAL *OUTPUT
FEEDBACK* DENGAN *COMMAND-GENERATOR
TRACKER***

FENI ISDARYANI
2214202006

DOSEN PEMBIMBING
Prof. Dr. Ir. Muhammad Nuh, DEA.
Dr. Trihastuti Agustinah, S.T., M.T.

PROGRAM MAGISTER
BIDANG KEAHLIAN SISTEM PENGATURAN
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016



THESIS-TE142599

TRACKING CONTROL FOR QUADROTOR USING OPTIMAL OUTPUT FEEDBACK METHOD WITH COMMAND-GENERATOR TRACKER

FENI ISDARYANI
2214202006

SUPERVISOR

Prof. Dr. Ir. Muhammad Nuh, DEA.
Dr. Trihastuti Agustinah, S.T., M.T.

MAGISTER PROGRAM
CONTROL SYSTEM ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING
FACULTY OF INDUSTRIAL TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Teknik (M.T)
di
Institut Teknologi Sepuluh Nopember

oleh:
Feni Isdaryani
NRP. 2214202006

Tanggal Ujian : 13 Juni 2016
Periode Wisuda : September 2016

Disetujui oleh:


1. **Prof. Dr. Ir. Muhammad Nuh, DEA.** (Pembimbing 1)
NIP. 195906171984031002


2. **Dr. Trihastuti Agustinah, S.T., M.T.** (Pembimbing 2)
NIP. 196808121994032001


3. **Prof. Ir. Abdullah Alkaff, M.Sc., Ph.D.** (Penguji 1)
NIP. 195501231980031002


4. **Prof. Dr. Ir. Achmad Jazidie, M.Eng.** (Penguji 2)
NIP. 195902191986101001


5. **Dr. Ir. Mochammad Rameli** (Penguji 3)
NIP. 195412271981031002

Direktur Program Pascasarjana,


Prof. Ir. Djauhar Manfaat, M.Sc., Ph.D.
NIP. 196012021987011001

KONTROL *TRACKING* UNTUK QUADROTOR MENGUNAKAN METODE *OPTIMAL OUTPUT FEEDBACK* DENGAN *COMMAND-GENERATOR TRACKER*

Nama mahasiswa : Feni Isdaryani
NRP : 2214202006
Pembimbing : Prof. Dr. Ir. Muhammad Nuh, DEA.
Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRAK

Quadrotor UAV merupakan sistem yang nonlinier, *underactuated* dan tidak stabil. Karakteristik tersebut menjadi fokus perhatian untuk perkembangan penelitian kontrol quadrotor. Dalam tesis ini metode kontrol *optimal output feedback* digunakan untuk kontrol *attitude* dan posisi quadrotor. Kontrol *optimal output feedback* yang memiliki performa H_∞ digunakan untuk menjaga kestabilan dan kekokohan quadrotor. Gain kontroler diperoleh menggunakan teknik numerik *iterative Linear Matrix Inequality* (LMI). Untuk kontrol *tracking* posisi pada sumbu X dan Y , digunakan struktur *Command-Generator Tracker* (CGT). Permasalahan kontrol *tracking* tersebut diselesaikan menggunakan metode LQ regulator dengan *output feedback*. Simulasi menunjukkan bahwa sistem kontrol dapat men-*track* sinyal referensi yang diberikan. Selain itu, sistem kontrol juga mampu mengatasi gangguan eksternal berupa *wind gust* dengan penyimpangan posisi maksimum mencapai 0.28 m.

Kata kunci: Quadrotor UAV, performa H_∞ , *Linear Matrix Inequality*, *Command-Generator Tracker*, LQ regulator.

TRACKING CONTROL OF QUADROTOR USING OPTIMAL OUTPUT FEEDBACK METHOD WITH COMMAND-GENERATOR TRACKER

By : Feni Isdaryani
Student Identity Number : 2214202006
Supervisor : Prof. Dr. Ir. Muhammad Nuh, DEA.
Co-Supervisor : Dr. Trihastuti Agustinah, S.T., M.T.

ABSTRACT

Quadrotor UAV is a nonlinear, underactuated and unstable systems. These systems characteristic, will be the main concern of the quadrotor control development. In this thesis, optimal output feedback control is used for quadrotor attitude and position control. The optimal output feedback control with H_∞ performance is used to maintain stability and robustness of the quadrotor. Gain controller is obtained using iterative numerical techniques, Linear Matrix Inequality (LMI). Command-Generator Tracker (CGT) structure is used for position *tracking* control along X-axis and Y-axis. The position tracking control problem is solved using the LQ regulator method with output feedback. Simulation shows that the control systems is able to track the reference signal. In addition, the control system has a robust performance in the presence of wind gust disturbance which quantified by with maximum deviation about 0.28 m.

Key words: Quadrotor UAV, H_∞ Performance, Linear Matrix Inequality, Command-Generator Tracker, LQ regulator.

DAFTAR ISI

| | |
|---|-------------|
| LEMBAR PENGESAHAN | iii |
| PERNYATAAN KEASLIAN TESIS | v |
| ABSTRAK | vii |
| ABSTRACT | ix |
| KATA PENGANTAR | xi |
| DAFTAR ISI | xiii |
| DAFTAR GAMBAR | xv |
| DAFTAR TABEL | xvi |
| BAB 1 PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 3 |
| 1.3 Tujuan | 3 |
| 1.4 Batasan Masalah | 3 |
| 1.5 Kontribusi | 3 |
| 1.6 Metodologi Penelitian | 3 |
| BAB 2 KAJIAN PUSTAKAN DAN TEORI DASAR | 5 |
| 2.1 Kajian Penelitian Terkait | 5 |
| 2.1.1 <i>Takagi-Sugeno Model for Quadrotor Modelling and Control Using Nonlinear State Feedback Controller</i> [2] | 5 |
| 2.1.2 <i>Optimal path tracking control of a Quadrotor</i> [3] | 7 |
| 2.1.3 <i>Simulation of the quadrotor controlled with LQR with Integral Effect</i> [4] | 9 |
| 2.1.4 <i>Robust H_2 Static Output feedback Tracking Controller Design of Longitudinal Dynamics of a Miniature Helicopter via LMI Technique</i> [6] | 10 |
| 2.1.5 <i>Design of H_∞ Command and Control Loops for Unmanned Aerial Vehicles using Static Output-Feedback</i> [7] | 12 |
| 2.2 Teori Dasar | 14 |
| 2.2.1 Konsep Dasar Quadrotor | 15 |
| 2.2.2 Kinematika Quadrotor | 16 |
| 2.2.3 Model Dinamika Translasi Quadrotor | 20 |
| 2.2.4 Model Dinamika Rotasi Quadrotor | 21 |
| 2.2.5 Analisa Kestabilan Sistem Linier | 24 |

| | |
|--|-----------|
| 2.2.6 Kontrol <i>Static Output Feedback</i> dengan performa H_∞ | 26 |
| 2.2.7 <i>Command-Generator Tracker</i> | 28 |
| 2.2.8 <i>Linear Matrix Inequality</i> (LMI) | 31 |
| 2.2.9 Blok Koreksi..... | 32 |
| BAB 3 PERANCANGAN SISTEM | 33 |
| 3.1 Linierisasi Dinamika Quadrotor | 33 |
| 3.2 Strategi Kontrol | 37 |
| 3.3 Prosedur Perancangan Sistem..... | 39 |
| 3.4 Perancangan Kontroler Rotasi | 40 |
| 3.5 Perancangan Kontroler Posisi..... | 44 |
| 3.5.1 Perancangan kontroler posisi pada sumbu X | 45 |
| 3.5.2 Perancangan kontroler posisi pada sumbu Y | 46 |
| 3.6 Blok Koreksi..... | 50 |
| BAB 4 HASIL DAN PEMBAHASAN | 53 |
| 4.1 Hasil Perancangan Sistem Kontrol | 53 |
| 4.2 Pengujian Respon Sistem Untuk Gerak Translasi | 55 |
| 4.3 Pengujian Respon Sistem Untuk Gerak Rotasi | 59 |
| 4.4 Pemodelan Efek Embusan Angin (<i>Wind gust</i>) | 60 |
| 4.5 Pengujian Respon Sistem Terhadap Efek <i>Wind gust</i> | 61 |
| 4.6 Pengujian Respon Sistem Terhadap Noise Pengukuran | 68 |
| 4.7 Simulasi dengan Sinyal Referensi yang Berbeda-beda | 71 |
| 4.8 Perbandingan Kontroler Hasil Desain dengan Kontroler Nonlinear PD | 73 |
| BAB 5 PENUTUP | 79 |
| 5.1 Kesimpulan | 79 |
| 5.2 Saran | 79 |
| DAFTAR PUSTAKA | 81 |
| LAMPIRAN | 82 |
| RIWAYAT PENULIS | 89 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 3.1 Parameter Sistem Quanser Qball-X4 [9] | 34 |
| Tabel 3.2. Tabel Algoritma untuk Kontroler Rotasi | 43 |
| Tabel 3.3 Algoritma Kontrol untuk Kontroler Posisi..... | 48 |
| Tabel 3.4 Nilai parameter yang digunakan untuk kontroler posisi X dan Y..... | 49 |
| Tabel 4.1 Penyimpangan posisi quadrotor pada sumbu X dan Y saat ada <i>wind gust</i> | 67 |
| Tabel 4.2 Nilai gain K pada inner loop dan norm- ∞ sistem untuk berbagai nilai γ | 67 |
| Tabel 4.3 Nilai Integral Square Error (ISE) untuk Berbagai Nilai Noise | 71 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2.1 Respon <i>attitude</i> quadrotor (ϕ, θ, ψ) [2]..... | 6 |
| Gambar 2.2 (a) Posisi 3D untuk LMI tanpa <i>constraint</i> , (b) Posisi 3D untuk LMI dengan <i>constraint</i> [2] | 6 |
| Gambar 2.3 Diagram Blok LQ <i>Tracking</i> waktu diskrit [3]..... | 7 |
| Gambar 2.4 Respon <i>Path Tracking</i> (x, y, z) tanpa gangguan [3]..... | 8 |
| Gambar 2.5 Respon <i>Path Tracking</i> (x, y, z) dengan gangguan [3] | 8 |
| Gambar 2.6 Struktur LQR dengan Integral [4] | 9 |
| Gambar 2.7 Respon <i>attitude</i> dan <i>altitude</i> dari quadrotor [4] | 10 |
| Gambar 2.8 Struktur sistem <i>tracking</i> pada [6] | 11 |
| Gambar 2.9 Respon sistem ketika tanpa gangguan [6] | 12 |
| Gambar 2.10 Respon sistem dengan gangguan [6] | 12 |
| Gambar 2.11 Struktur kontroler <i>inner</i> dan <i>outer loop</i> [7]..... | 13 |
| Gambar 2.12 Respon Posisi terhadap <i>command</i> yang diberikan [7] | 14 |
| Gambar 2.13 Quadrotor UAV dan Gaya Angkat Pada Setiap Rotor..... | 15 |
| Gambar 2.14 Diagram Cartesius Rotasi Terhadap Sumbu X | 18 |
| Gambar 2.15 Diagram Cartesius Rotasi Terhadap Sumbu Y | 19 |
| Gambar 2.0.16 Diagram Cartesius Rotasi Terhadap Sumbu Z..... | 19 |
| Gambar 3.1 Model dari sumbu <i>roll/pitch</i> [9] | 33 |
| Gambar 3.2 Model dari sumbu yaw dengan arah putaran baling-baling | 36 |
| Gambar 3.3 Diagram Blok Sistem Kontrol Quadrotor Hasil Perancangan | 38 |
| Gambar 3.4 Diagram blok kontroler <i>outer loop</i> | 39 |
| Gambar 3.5 Diagram blok kontroler <i>inner loop</i> | 40 |
| Gambar 3.6 Respon frekuensi dari $\ T_{zw}(s)\ _{\infty}$ | 44 |
| Gambar 3.7 Diagram Simulink Kontroler untuk Posisi X | 46 |
| Gambar 3.8 Diagram Simulink Kontroler untuk Posisi Y | 48 |
| Gambar 4.1 (a) Perbandingan respon θ dengan posisi X, (b) Perbandingan respon ϕ dengan posisi Y | 54 |
| Gambar 4.2 Sinyal Referensi Posisi pada Sumbu X dan Y | 55 |
| Gambar 4.3 Referensi Lintasan dalam bidang X, Y, Z..... | 55 |
| Gambar 4.4 Respon <i>Tracking</i> Posisi pada sumbu X dan Y | 56 |
| Gambar 4.5 Respon <i>Tracking</i> Posisi pada sumbu X..... | 57 |
| Gambar 4.6 Respon <i>Tracking</i> Posisi pada Sumbu Y | 57 |
| Gambar 4.7 Respon <i>Tracking</i> Posisi pada Sumbu Z..... | 58 |
| Gambar 4.8 <i>Screenshoot</i> dari <i>virtual world</i> quadrotor..... | 58 |
| Gambar 4.9 Respon Sudut <i>Roll</i> tanpa Gangguan..... | 59 |
| Gambar 4.10 Respon Sudut <i>Pitch</i> tanpa Gangguan | 59 |
| Gambar 4.11 Diagram blok untuk simulasi dengan gangguan | 60 |

| | |
|---|----|
| Gambar 4.12 (a) Kurva <i>wind gust</i> secara keseluruhan, (c) <i>wind gust</i> pada sistem. | 61 |
| Gambar 4.13 Respon <i>Tracking</i> dengan Gangguan dalam sumbu X,Y..... | 61 |
| Gambar 4.14 Respon sudut ϕ dan respon posisi Y saat terdapat <i>wind gust</i> | 62 |
| Gambar 4.15 Respon sudut θ dan posisi X saat terdapat <i>wind gust</i> | 63 |
| Gambar 4.16 <i>Screenshot virtual world</i> saat simulasi dengan efek gangguan | 64 |
| Gambar 4.17 <i>Virtual world</i> (tampak atas) saat diberi efek gangguan..... | 64 |
| Gambar 4.18 Respon posisi pada sumbu X dengan berbagai nilai γ | 65 |
| Gambar 4.19 Respon Kecepatan pada sumbu X dengan berbagai nilai γ | 65 |
| Gambar 4.20 Respon posisi pada sumbu Y dengan berbagai nilai γ | 66 |
| Gambar 4.21 Respon kecepatan pada sumbu Y dengan berbagai nilai γ | 66 |
| Gambar 4.22 Sinyal random Gaussian dengan $N(0, 0.01)$ | 68 |
| Gambar 4.23 Sudut <i>Pitch</i> dan posisi X saat diberi <i>noise</i> $N(0, 0.01)$ | 68 |
| Gambar 4.24 Sudut <i>Roll</i> dan posisi Y saat diberi <i>noise</i> $N(0, 0.01)$ | 69 |
| Gambar 4.25 Respon <i>tracking</i> lintasan dalam bidang XY saat terdapat <i>noise</i> | 70 |
| Gambar 4.26 <i>Virtual world</i> dari quadrotor saat simulasi diberi efek <i>noise</i> | 70 |
| Gambar 4.27 <i>Tracking</i> Posisi dengan Lintasan Berbentuk Spiral (Helix) | 72 |
| Gambar 4.28 <i>Tracking</i> Posisi dengan Lintasan Berbentuk Persegi..... | 72 |
| Gambar 4.29 Posisi X dan Y Pada Saat <i>Tracking</i> Lintasan Berbentuk Persegi..... | 72 |
| Gambar 4.30 Gangguan yang diberikan pada kedua metode kontrol..... | 73 |
| Gambar 4.31 Gangguan yang diberikan pada kedua metode kontrol..... | 74 |
| Gambar 4.32 Kurva respon <i>tracking</i> pada bidang X dan Y dengan gangguan | 75 |
| Gambar 4.33 Perbandingan respon posisi X dengan gangguan..... | 76 |
| Gambar 4.34 Perbandingan respon posisi Y dengan gangguan..... | 76 |
| Gambar 4.35 Perbandingan respon sudut <i>Pitch</i> dengan gangguan | 77 |
| Gambar 4.36 Perbandingan respon sudut <i>Roll</i> dengan gangguan | 77 |

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pesawat tanpa awak atau yang dikenal dengan istilah *Unmanned Aerial Vehicle* (UAV) jenis quadrotor memiliki empat buah rotor, yang biasanya digunakan dalam aplikasi untuk operasi penyelamatan, pengamatan, atau pengenalan objek. Untuk alasan tersebut, quadrotor perlu memiliki kemampuan untuk dapat *tracking*, bermanuver dan kokoh terhadap gangguan eksternal. Quadrotor UAV merupakan sistem yang nonlinier, *underactuated* dan tidak stabil. Karakteristik tersebut menjadi fokus perhatian untuk mengembangkan berbagai metode kontrol.

Berbagai metode kontrol diterapkan untuk mengatasi permasalahan kontrol pada quadrotor, seperti stabilisasi dan *tracking* posisi quadrotor. Kontrol *tracking* posisi quadrotor merupakan tantangan tersendiri karena keberadaan *coupling* antara dinamika rotasi dan translasi. Pada [1], teknik kontrol nonlinier digunakan untuk kontrol *trajectory tracking*. Metode nonlinear lebih bisa mendekati karakteristik sistem nonlinear, namun proses perhitungannya lebih rumit karena dibutuhkan pengetahuan yang utuh mengenai kompleksitas sistem itu sendiri. Strategi lain untuk menyelesaikan permasalahan kontrol yang kompleks tersebut adalah dengan membagi kompleksitas tersebut menjadi beberapa bagian yang sederhana. Bagian-bagian sederhana tersebut diselesaikan secara terpisah sehingga menghasilkan solusi bagi permasalahan yang kompleks tersebut. Strategi kontrol yang dipaparkan oleh [2]-[4] digunakan untuk menyelesaikan permasalahan kontrol quadrotor menggunakan pendekatan linier. Dimana model plant yang nonlinier didekati menjadi beberapa sistem linier melalui linierisasi.

Pada [2], sistem nonlinear didekati melalui linearisasi kemudian pendekatan *fuzzy* Takagi-Sugeno digunakan untuk menerapkan algoritma kontrol linear ke setiap subsistem linear. Pada [3] untuk menghasilkan *tracking* quadrotor yang optimal digunakan kontroler *Linear Quadratic Tracking* (LQT) dimana metode ini menggunakan pendekatan model linier. Untuk men-*track* sinyal step,

metode LQR dengan efek integral digunakan pada [4], dimana sistem dilinierisasi terlebih dahulu.

Penelitian [5]-[7] menggunakan *Static Output Feedback* (SOF) untuk mengatasi keterbatasan state yang terukur. Dalam kasus yang ideal semua vektor state akan tersedia, namun kadangkala, tidak semua vektor state dapat terukur menyebabkan kurangnya informasi untuk *feedback*. Keuntungan dari desain SOF adalah hanya memerlukan sinyal yang terukur dari plant untuk dikontrol. Pada [6], parameter ketidakpastian dari dinamika plant helikopter dipertimbangkan. Gain integrator dan proporsional ditambahkan dalam struktur kontrol yang diajukan. Kondisi perlu dari H_2 /LMI disajikan untuk menghasilkan gain kontrol. Hasilnya, kontroler dapat menstabilkan sistem dan *robust* terhadap gangguan angin, namun pada penelitian ini hanya dilakukan *tracking* pada longitudinal saja.

Kontroler SOF yang memiliki performa H_∞ digunakan untuk kontrol *tracking* posisi dan *attitude* dari helikopter [7]. Struktur kontrol *tracking* posisi didesain menggunakan teknik *loop shaping*. Gain kontroler dicari menggunakan solusi persamaan Ricatti. Algoritma kontrol yang disajikan menghasilkan kontrol yang *robust* terhadap gangguan.

Pada umumnya, struktur kontrol untuk *tracking* didesain dengan menambahkan gain integrator atau gain proporsional pada error *tracking*. Desain tersebut cocok untuk sinyal referensi berupa sinyal step ataupun ramp. Pada [8], disajikan desain struktur kontrol *tracking* untuk mengatasi permasalahan jika input referensi yang diinginkan bukan sinyal step. Desain struktur yang dimaksud adalah *Command-Generator Tracker* (CGT), dimana dinamikanya merepresentasikan input referensi.

Berdasarkan latar belakang tersebut, dalam Tesis ini diajukan desain metode kontrol optimal untuk mengontrol quadrotor agar dapat *tracking* lintasan yang ditentukan. Untuk kontroler *tracking* posisi, struktur *Command-Generator Tracker* (CGT) digunakan untuk dapat menghasilkan kesesuaian *tracking* sinyal referensi dengan *error tracking* sekecil mungkin. Untuk menjaga kestabilan *attitude*, digunakan kontroler *output feedback* yang memiliki performa H_∞ .

1.2 Rumusan Masalah

Rumusan masalah dari penelitian ini adalah bagaimana merancang struktur kontrol *tracking* untuk quadrotor yang dapat memperoleh kesesuaian *tracking* sambil menjaga stabilitas *attitude* walau terdapat gangguan eksternal.

1.3 Tujuan

Tujuan dari penelitian ini adalah memperoleh desain struktur kontrol *tracking* menggunakan struktur *Command-Generator Tracker* yang dikhususkan untuk quadrotor, sehingga dapat memperoleh kesesuaian *tracking* serta menjaga kestabilan *attitude* dengan kemampuan untuk mengatasi gangguan eksternal.

1.4 Batasan Masalah

Terdapat beberapa batasan masalah pada usulan tesis ini, yaitu:

- model quadrotor yang memiliki konfigurasi motor dalam bentuk ‘plus’,
- permasalahan kontrol *tracking* terjadi pada saat kondisi *hover* sehingga tidak membahas permasalahan kontrol pada saat *take off* dan *landing*,
- tidak membahas masalah pengendalian orientasi dari quadrotor dan
- gangguan yang diberikan bernilai terbatas.

1.5 Kontribusi

Menghasilkan desain kontrol menggunakan metode optimal *output feedback* dengan menambahkan struktur CGT khusus untuk quadrotor yang mana performa H_∞ dipertimbangkan untuk mengatasi gangguan eksternal.

1.6 Metodologi Penelitian

Metodologi yang digunakan dalam pengerjaan Tesis ini sebagai berikut:

1. Studi Literatur

Hal-hal yang akan dipelajari untuk mendukung pengerjaan Tesis adalah konsep Quadrotor, konsep kestabilan Lyapunov, kontrol *Linear Quadratic*

regulator berbasis *output-feedback*, konsep *Command-Generator Tracker*, Performansi H_∞ , *Linear Matrix Inequality* (LMI).

2. Pemodelan Sistem

Pemodelan dilakukan dengan melakukan identifikasi fisik dengan menggunakan pendekatan Newton-Euler agar diketahui hubungan antar gaya yang bekerja pada sistem.

3. Perancangan Kontroler dan Analisis Sistem

Tahap perancangan kontroler dilakukan untuk mendapatkan desain kontroler yang tepat untuk memenuhi performa yang diinginkan. Adapun kontroler yang akan dirancang adalah kontroler *optimal output feedback* yang memiliki performa H_∞ untuk kontrol stabilisasi dan *optimal output feedback* dengan struktur CGT untuk kontrol *tracking*.

4. Simulasi dan Analisis Performa Sistem

Kemampuan sistem untuk dapat melakukan *tracking* lintasan sesuai dengan referensi yang diinginkan akan dibuktikan pada simulasi. Hasil desain akan dianalisa apakah sistem sudah memenuhi kriteria kestabilan Lyapunov.

5. Penulisan dan Penyusunan Buku Tesis

Penyusunan buku Tesis yang terdiri dari pendahuluan, teori penunjang, perancangan kontroler, hasil simulasi, serta penutup.

BAB 2

KAJIAN PUSTAKA DAN TEORI DASAR

2.1 Kajian Penelitian Terkait

Pada penyusunan tesis ini, dilakukan kajian terhadap penelitian-penelitian terkait yang telah ada. Berbagai metode dan struktur kontrol yang digunakan pada setiap pustaka akan dipaparkan dalam sub-bab ini, sehingga diperoleh ide dasar untuk penyusunan tesis.

2.1.1 Takagi-Sugeno Model for Quadrotor Modelling and Control Using Nonlinear State Feedback Controller [2]

Paper ini menggunakan model fuzzy T-S untuk pemodelan quadrotor. Model T-S dikembangkan menggunakan pendekatan *multiple model*, terdiri dari tiga model lokal dalam daerah ruang kerja (*operating space*). Pendekatan *multiple model* merupakan pendekatan untuk pemodelan dengan interpolasi. Sistem nonlinier dibagi kedalam beberapa sub-sistem yang sederhana. Setiap sub-sistem sederhana tersebut adalah model lokal linier yang menggambarkan dinamika sistem nonlinier dalam region yang kecil dari ruang kerja.

Permasalahan kontrol yang diangkat yaitu permasalahan *tracking* altitude dan orientasi dari quadrotor. Solusi penyelesaian dilakukan menggunakan teknik numerik LMI. Terdapat dua metode kontrol yang digunakan untuk menentukan nilai gain state feedback, yaitu metode LMI tanpa *constraint* dan LMI dengan *constraint*. Dalam metode LMI tanpa *constraint*, gain K didesain sehingga kondisi berikut

$$G_{ii}^T P + P G_{ii} < 0, \quad \forall i \in I_r,$$

$$\left(\frac{G_{ij} + G_{ji}}{2} \right)^T P + P \left(\frac{G_{ij} + G_{ji}}{2} \right) < 0, \quad \forall (i, j) \in I_r^2, i < j$$

harus terpenuhi. dimana $G_{ii} = A_i - B_i K_j$.

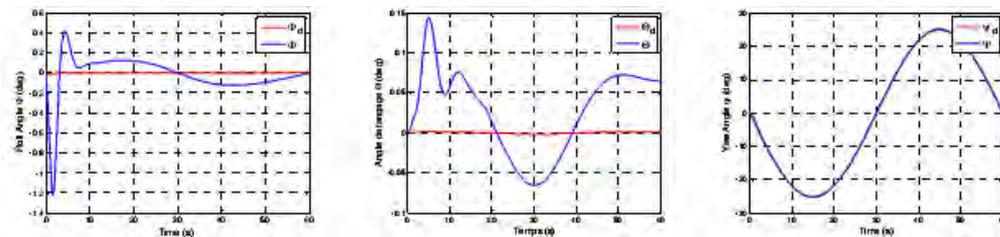
Metode kedua adalah metode LMI dengan *constraint*. Kestabilan sistem *closed loop* dicari menggunakan teknik *pole placement*. *Eigenvalue* dari model T-S ditempatkan dalam region (D) pada sebelah kiri bidang kompleks untuk mencegah dinamika kontroler yang terlalu cepat dan mencapai perilaku transien

yang diinginkan. Model *closed loop* dari T-S dikatakan stabil dalam region D untuk nilai state feedback K_i jika dan hanya jika terdapat matriks simetris positif X yang memenuhi pertidaksamaan berikut

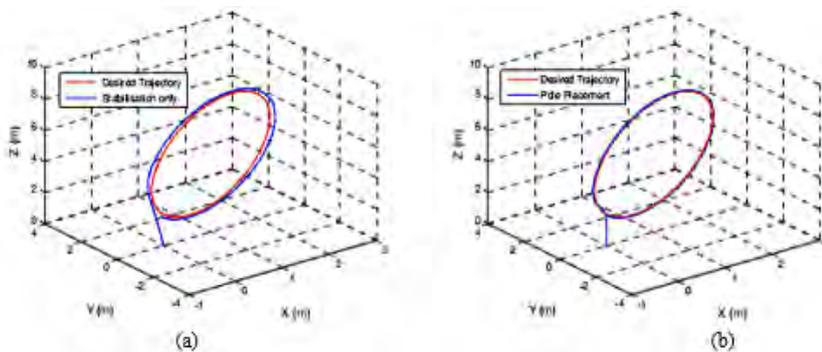
$$\begin{pmatrix} -\rho X & qX + XA_i^T + N_i^T B_i^T \\ qX + A_i X + B_i N_i & -\rho X \end{pmatrix} < 0, i = j.$$

Gambar 2.1, menunjukkan respon sudut roll, *pitch* dan yaw. Kurva respon disajikan hanya hingga detik ke-60 dimana respon roll dan *pitch* tidak dapat mengikuti referensi yang diberikan. Hal tersebut bukan masalah karena objektif kontrol pada paper ini adalah *tracking* orientasi quadrotor dan dapat dilihat bahwa sudut yaw dapat men-*track* referensi yang diberikan.

Seperti terlihat pada Gambar 2.2, kontroler LMI dengan *constraint* menunjukkan respon *tracking* lintasan secara tepat, sedangkan pada kontroler LMI tanpa *constraint* respon menunjukkan adanya penyimpangan posisi. Namun, dalam paper ini tidak dilakukan pengujian kekokohan sistem terhadap gangguan.



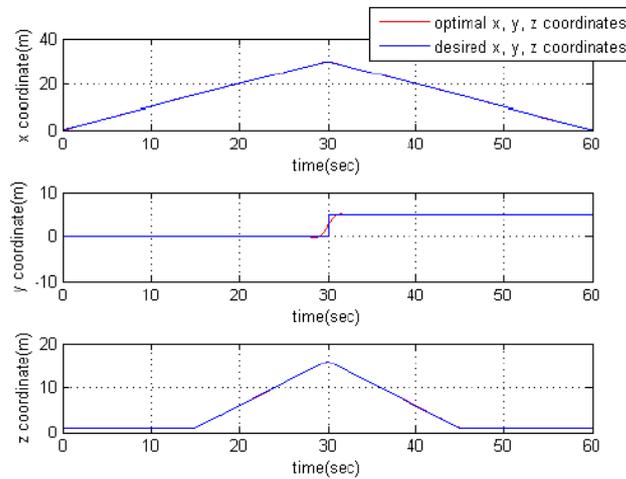
Gambar 2.1 Respon *attitude* quadrotor (ϕ, θ, ψ) [2]



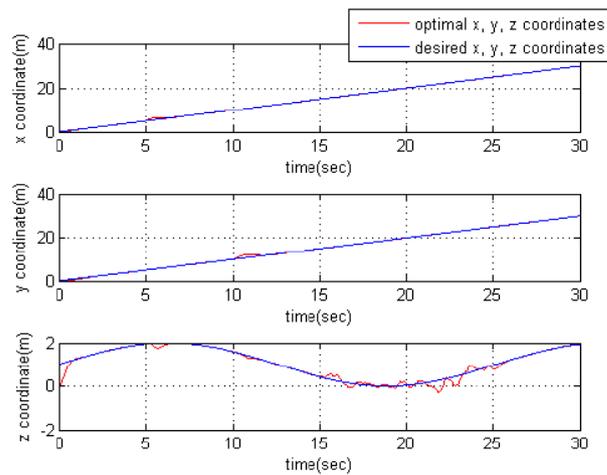
Gambar 2.2 (a) Posisi 3D untuk LMI tanpa *constraint*, (b) Posisi 3D untuk LMI dengan *constraint* [2]

Respon sistem dapat dilihat pada Gambar 2.4, sedangkan Gambar 2.5 menunjukkan respon posisi pada saat diberi gangguan. Sinyal referensi yang diberikan berupa sinyal ramp dan step. Gangguan yang diberikan diposisi sumbu z, mempengaruhi performa *tracking* dengan adanya deviasi posisi.

Kelebihan dari paper ini adalah penggunaan teknik linierisasi yang sederhana namun tetap dapat merepresentasikan model nonlinier dari quadrotor. Walau masih terdapat efek *coupling* dari sistem, namun kontroler dapat menghasilkan performa yang baik pada sistem.



Gambar 2.4 Respon *Path Tracking* (x, y, z) tanpa gangguan [3]



Gambar 2.5 Respon *Path Tracking* (x, y, z) dengan gangguan [3]

2.1.3 Simulation of the quadrotor controlled with LQR with Integral Effect [4]

Paper ini menjelaskan mengenai kontrol *attitude* pada quadrotor menggunakan teknik kontrol linier, yaitu LQR. Dinamika quadrotor yang nonlinier dan tidak stabil dilinierisasi pada kondisi *hover*. Permasalahan yang diangkat pada paper ini adalah permasalahan kontrol *tracking* untuk *altitude* dan *attitude*.

Sinyal referensi $z_d, \phi_d, \theta_d, \psi_d$ dibandingkan dengan output aktual z, ϕ, θ, ψ sehingga akan menghasilkan error *tracking*. Untuk meminimumkan error *tracking*, integrator ditambahkan dalam struktur kontrol ini, seperti yang terdapat pada Gambar 2.6.

Indeks performa yang digunakan adalah sebagai berikut:

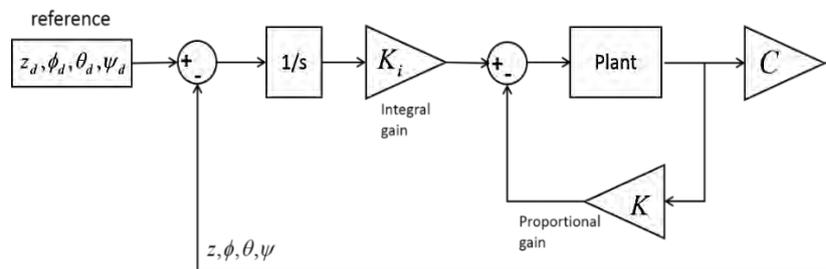
$$J = \int_0^{\infty} [\bar{x}^T(x) R_{zz} \bar{x}(x) + u^T(t) R_{uu} u(t)] dt$$

dimana \bar{x} adalah *state augmented* yang dituliskan dalam persamaan *state space* sebagai berikut:

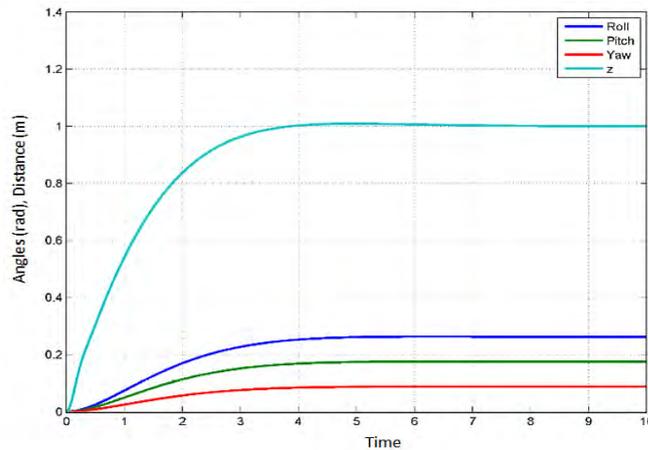
$$\begin{bmatrix} \dot{x}(t) \\ \dot{x}_1(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ I \end{bmatrix} r(t)$$

dapat dilihat bahwa, kontroler terdiri dari dinamika plant dan dinamika kompensator yang diperluas (augmented). Adapun sinyal kontrol dari sistem augmented adalah sebagai berikut

$$u(t) = -[K \quad K_1] \begin{bmatrix} x(t) \\ x_1(t) \end{bmatrix} = -\bar{K}\bar{x}(t)$$



Gambar 2.6 Struktur LQR dengan Integral [4]



Gambar 2.7 Respon *attitude* dan *altitude* dari quadrotor [4]

Hasil simulasi dapat dilihat pada Gambar 2.7, menggunakan kontrol LQR yang ditambah dengan integrator dapat membuat quadrotor *men-track* referensi sinyal step tanpa adanya *overshoot*. Namun, waktu untuk mencapai *steady state* cukup lama. Dalam paper ini tidak dilakukan uji coba terhadap gangguan dan juga sinyal referensi yang diberikan hanya berupa sinyal step.

2.1.4 Robust H_2 Static Output feedback Tracking Controller Design of Longitudinal Dynamics of a Miniature Helicopter via LMI Technique [6]

Paper ini memaparkan mengenai desain kontrol *robust static output feedback* (SOF) untuk plant helikopter. Dinamika longitudinal dari helikopter dilinierisasi pada kondisi nominalnya. Gain kontroler *output feedback* didapat dengan menggunakan teknik *iterative* LMI. Parameter ketidakpastian dari dinamika helikopter dipertimbangkan.

Tinjau dinamika ketidakpastian dari sistem adalah

$$\dot{x} = (A + \Delta A) + B_1 w + B_2 u$$

$$z = C_1 x + D_1 u$$

$$y = C_2 x$$

Objektif desain adalah membangun sinyal kontrol robust SOF $u = -Ky$ yang memiliki sistem *closed loop* yang stabil dan norm H_2 dari performa output z kurang dari nilai γ yang diberikan. Hasil linearisasi model longitudinal helikopter adalah sebagai berikut:

$$A = \begin{bmatrix} 0 & 9.46 & 0 & 9.788 & 0 \\ 0 & 0 & 0.9963 & 0 & 0 \\ 0 & 0 & 0 & 212.9649 & 0 \\ 0 & 0 & -1 & -8.35 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}; B = [0 \ 0 \ 0 \ 35.07 \ 0]^T$$

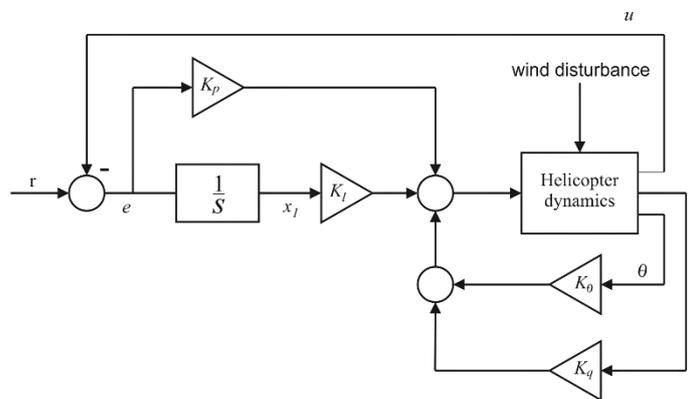
$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 57.2957 & 0 & 0 & 0 \\ 0 & 0 & 57.2957 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

dengan variable state $x = [e \ \theta \ q \ a_1 \ x_1]^T$ dan output terukur adalah

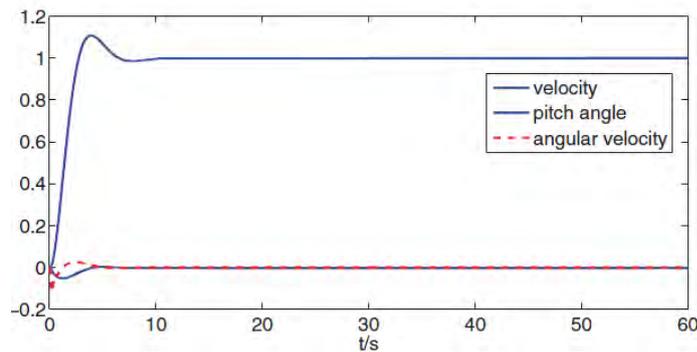
$$y = [e \ \theta \ q \ x_1]^T.$$

Kecepatan *vehicle* dinotasikan $U = [u \ v \ w]^T$ dan kecepatan sudut dinotasikan oleh $\omega = [p \ q \ r]^T$, sudut *flapping* longitudinal dinotasikan oleh a_1 dan error *tracking* dinotasikan oleh e sedangkan integral error dinotasikan oleh x_1 .

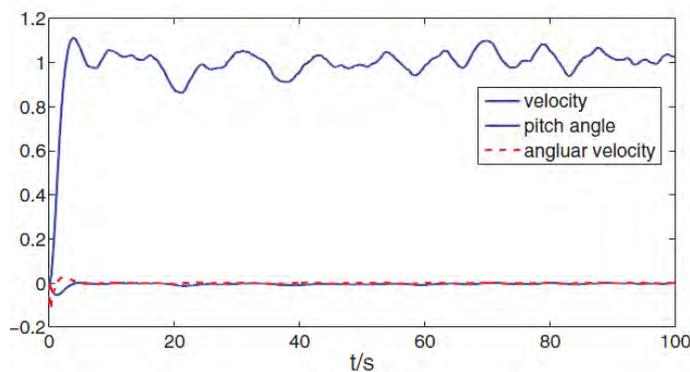
Dalam metode *output feedback* tidak semua variabel state dipakai untuk informasi *feedback*. Seperti pada paper ini, dari lima buah variable state sistem, hanya empat buah output terukur yang di-*feedback*-an. Struktur desain yang digunakan dapat dilihat pada Gambar 2.8. Perfoma output yang dibandingkan dengan referensi adalah u , yaitu kecepatan *vehicle* pada sumbu x . Output terukur yang memberi informasi *feedback* adalah sudut dan kecepatan sudut *pitch*. Dari diagram kontrol dapat dilihat bahwa kondisi helikopter pada ketinggian yang sama diharapkan menjaga kecepatan agar tetap pada kecepatan tertentu.



Gambar 2.8 Struktur sistem *tracking* pada [6]



Gambar 2.9 Respon sistem ketika tanpa gangguan [6]



Gambar 2.10 Respon sistem dengan gangguan [6]

Hasil simulasi dapat dilihat pada Gambar 2.9, dimana kontroler dapat membuat helikopter mencapai posisi yang diinginkan juga menjaga kecepatannya tetap berada pada 1 m/s. Gambar 2.10 menunjukkan respon ketika diberi gangguan. Gangguan yang diberikan bernilai -2.36 m/s hingga 1.54 m/s, hasilnya pada respon kecepatan helikopter terjadi penyimpangan yaitu sekitar 0.2 m/s. Oleh karena itu, struktur dan algoritma *iterative* LMI yang digunakan dapat menghasilkan performa *robust* yang baik terhadap gangguan.

2.1.5 Design of H_∞ Command and Control Loops for Unmanned Aerial Vehicles using Static Output-Feedback [7]

Tujuan dari paper ini adalah untuk mendesain kontrol posisi dari helikopter menggunakan kontrol SOF yang memiliki performa H_∞ . Permasalahan yang diangkat pada paper ini adalah permasalahan kontrol *tracking* untuk helikopter. Jika ditinjau persamaan sistem berikut:

$$\dot{x} = Ax + Bu + Dd$$

$$y = Cx$$

dan performa output memenuhi

$$\|z(t)\|^2 = x^T Qx + u^T Ru$$

dimana performa output merupakan norm Euclidean dari state dan input sistem.

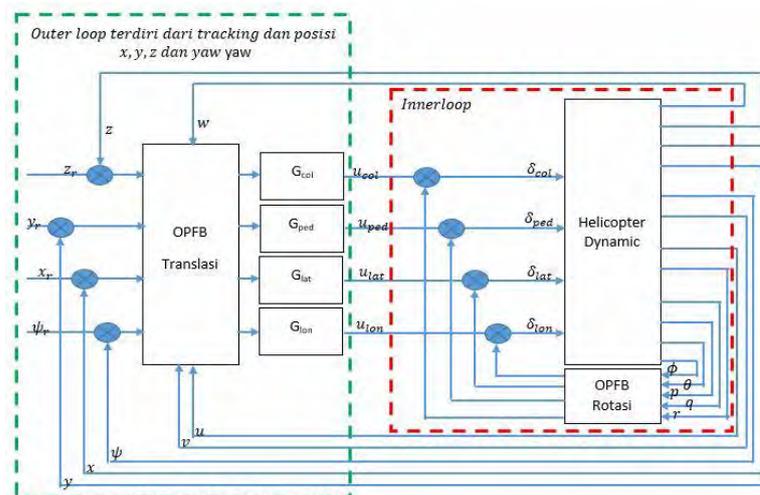
Sistem L₂ gain dikatakan dilemahkan oleh level pelemahan (γ) jika

$$\frac{\int_0^{\infty} \|z(t)\|^2 dt}{\int_0^{\infty} \|d(t)\|^2 dt} = \frac{\int_0^{\infty} (x^T Qx + u^T Ru) dt}{\int_0^{\infty} (d^T d) dt} \leq \gamma^2$$

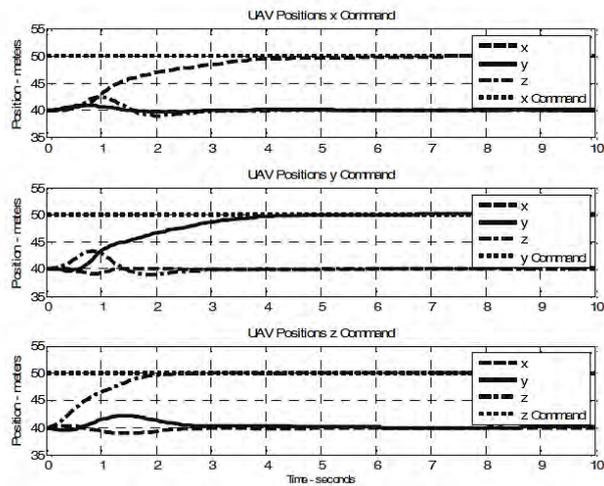
untuk setiap input gangguan d .

Untuk menghasilkan gain SOF sehingga $A_0 = (A - BKC)$ adalah stabil *asymptotic* dengan L₂ gain dilemahkan oleh γ , diasumsikan matriks (A, C) adalah *detectable* dan terdapat matriks L dan $P = P^T \geq 0$ sehingga gain *output feedback* adalah $KC = R^{-1}(B^T P + L)$.

Struktur kontrol yang digunakan terdiri dari *inner loop* dan *outer loop* seperti pada Gambar 2.11. Kontrol *inner loop* berfungsi untuk menjaga *attitude* helikopter dengan mengontrol state roll dan *pitch*. Kontroler *outer loop* berfungsi untuk menjaga posisi helikopter.



Gambar 2.11 Struktur kontroler *inner* dan *outer loop* [7]



Gambar 2.12 Respon Posisi terhadap *command* yang diberikan [7]

Algoritma kontrol optimal *output feedback* dengan performa H_∞ digunakan pada kedua kontroler (*inner* dan *outer loop*) namun, pada *outer loop* ditambahkan desain *loop shaping*. Hal tersebut bertujuan untuk mengatasi *trade off* antara performa dan kekakuan dalam *loop shaping*.

Hasil simulasi pada paper ini menunjukkan bahwa sistem kontrol yang dirancang dapat untuk membuat quadrotor melakukan *tracking* dengan memenuhi kriteria performa H_∞ . Hal yang menarik dari pustaka ini yaitu penentuan performa output yang terdiri dari persamaan kuadrik state dan input kontrol dari sistem.

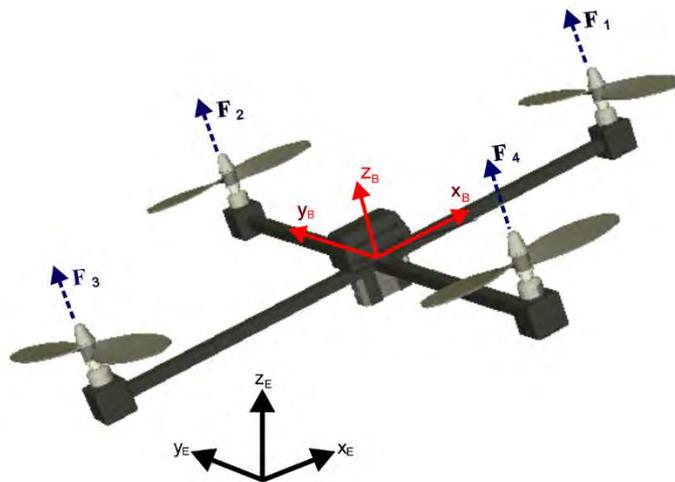
2.2 Teori Dasar

Teori dasar pada sub-bab ini meliputi teori-teori pendukung yang menunjang penyusunan tesis. Adapun teori-teori yang akan dipaparkan meliputi: konsep dasar quadrotor, pemodelan quadrotor, teori kestabilan sistem linier, teori optimal *output feedback* dengan performa H_∞ , teori dasar struktur *Command-Generator Tracker*, LMI, dan pembentukan blok koreksi sinyal kontrol untuk quadrotor.

2.2.1 Konsep Dasar Quadrotor

Quadrotor termasuk dalam jenis kendaraan udara tanpa awak (*unmanned aerial vehicles*), konsep quadrotor telah dikembangkan sejak lama. Quadrotor terdiri dari empat rotor yang polanya diletakkan secara menyilang. Setiap rotor terdiri dari baling-baling yang diterapkan pada motor DC. Secara umum quadrotor memiliki pengaturan motor depan dan motor belakang yang berputar searah jarum jam, sedangkan motor kiri dan motor kanan berputar berlawanan arah jarum jam. Dengan mengatur kecepatan dari setiap motor, maka pergerakan quadrotor dapat berubah.

Quadrotor memiliki 6 derajat kebebasan (*degree of freedom*), enam variabel tersebut digunakan untuk mengekspresikan posisi dan orientasi dalam bidang (x, y, z, ϕ, θ , dan ψ). Jarak pusat massa quadrotor sepanjang sumbu x, y dan z dari *frame* bumi dinotasikan oleh x, y dan z . Sudut Euler (ϕ, θ , dan ψ) merepresentasikan orientasi dari quadrotor. Sudut *roll* yang dinotasikan oleh ϕ merupakan sudut disekitar sumbu x . Sudut *pitch* dinotasikan oleh θ yang merupakan sudut pada sumbu y , dan sudut *yaw* dinotasikan oleh ψ yang merupakan sudut pada sumbu z .



Gambar 2.13 Quadrotor UAV dan Gaya Angkat Pada Setiap Rotor.

Gambar 2.13 merupakan ilustrasi quadrotor dengan acuan frame bumi dan frame *body*. Sudut *roll* dan *pitch* biasanya disebut dengan *attitude* untuk quadrotor, sedangkan sudut *yaw* ditunjukkan sebagai *heading* quadrotor. Untuk jarak quadrotor dari tanah dikenal sebagai *altitude* atau ketinggian, sedangkan posisi quadrotor dalam bidang X, Y biasa disebut dengan istilah posisi quadrotor.

Quadrotor dapat terbang dengan syarat gaya angkat pada quadrotor lebih besar daripada gaya gravitasi. Pada kondisi titik berat yang seimbang dan karakteristik motor yang sama, kondisi melayang (*hover*) tercapai saat semua motor memiliki kecepatan yang besarnya sama. Untuk mencapai posisi tertentu, quadrotor memiliki 4 pergerakan dasar yang digunakan berdasarkan pengaturan kecepatan dan arah putaran keempat motor, yaitu gerak *Thrust* (U_1), *Roll* (U_2), *Pitch* (U_3) dan *Yaw* (U_4).

Gerakan *thrust* yaitu gerakan menaikkan dan menurunkan posisi quadrotor pada sumbu vertikal. Pergerakan ini terjadi ketika kecepatan dari keempat motor dipercepat atau diperlambat dengan nilai penambahan yang sama. Semakin besar penambahan, maka semakin cepat pergerakan quadrotor ke atas. Jika pergerakan *thrust* diatur pada nilai tertentu, maka quadrotor dapat melakukan gerak *hover*.

Gerakan *roll* yaitu gerakan perputaran sudut quadrotor pada sumbu x . Pergerakan ini terjadi ketika kecepatan motor kiri, dipercepat atau diperlambat, dan kecepatan motor kanan diubah sebaliknya. Gerakan *pitch* yaitu gerakan perputaran sudut quadrotor pada sumbu y . Pergerakan ini terjadi ketika kecepatan motor depan, dipercepat atau diperlambat, dan kecepatan motor belakang diubah sebaliknya. Semakin besar percepatan sudut *pitch* maka semakin cepat juga quadrotor berputar.

Gerakan *yaw* merupakan gerakan perputaran sudut quadrotor pada sumbu z . Pergerakan ini terjadi dengan cara mempercepat atau memperlambat motor depan dan belakang sedangkan motor kiri dan kanan diubah sebaliknya.

2.2.2 Kinematika Quadrotor

Untuk dapat menganalisa kinematika maupun dinamika quadrotor, diperlukan pengetahuan mengenai matriks transformasi terlebih dahulu. Analisis digunakan menggunakan diagram cartesius tiga dimensi yaitu X, Y dan Z . Pada quadrotor, *frame* diagram cartesius terdiri dari dua yaitu *frame* bumi yang

dinotasikan oleh (X_E, Y_E, Z_E) dan *frame body* quadrotor (X_B, Y_B, Z_B) . *Frame* bumi merupakan rigid atau kaku sedangkan *frame body* bergerak baik rotasi maupun translasi.

Posisi linier quadrotor ditentukan dari koordinat vektor antara origin *frame body* serta origin *frame* bumi. Posisi linier dan posisi angular terdapat pada persamaan

$$v = (X, Y, Z)^T \quad (2.1)$$

$$\Theta = (\phi, \theta, \psi)^T \quad (2.2)$$

sedangkan enam derajat kebebasan quadrotor digambarkan oleh tiga buah translasi $V = (u, v, w)^T$ dan tiga buah rotasi $\Omega = (p, q, r)^T$.

Bila didefinisikan sumbu X menuju utara dan depan quadrotor, sumbu Z menjauhi pusat bumi dan menuju atas quadrotor, maka berdasarkan kaidah tangan kanan *cross vector* dapat ditentukan bahwa sumbu Y menuju barat atau kiri quadrotor seperti pada Gambar 2.13.

Sebuah matriks transformasi harus digunakan untuk mentransformasi vektor state dari *frame-E* ke *frame-B*. Matriks tersebut didapat berdasarkan kaidah dari pertama rotasi terhadap sumbu X , kemudian rotasi terhadap sumbu Y dan terakhir rotasi terhadap sumbu Z .

- Matriks rotasi terhadap sumbu X

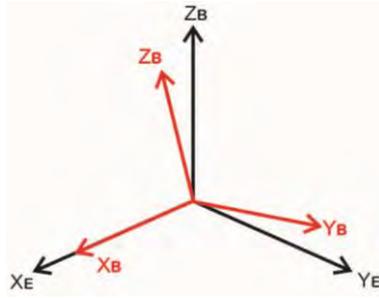
Seperti yang terlihat pada Gambar 2.14, nilai matriks rotasi pada sumbu X adalah sebagai berikut:

$$\begin{aligned} X_E &= X_B \cos 0 + Y_B \cos 90 + Z_B \cos 90 \\ Y_E &= X_B \cos 90 + Y_B \cos \phi + Z_B \cos(90 + \phi) \\ Z_E &= X_B \cos 90 + Y_B \cos(90 - \phi) + Z_B \cos \phi \end{aligned} \quad (2.3)$$

Jika dibentuk kedalam bentuk matriks menjadi

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix}$$

sehingga matriks rotasi terhadap sumbu X adalah



Gambar 2.14 Diagram Cartesius Rotasi Terhadap Sumbu X

$$R_{(x,\phi)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}. \quad (2.4)$$

- Matriks rotasi terhadap sumbu Y

Matriks rotasi pada sumbu Y ditunjukkan oleh Gambar 2.15, dimana hasil rotasi adalah sebagai berikut:

$$\begin{aligned} X_E &= X_B \cos \theta + Y_B \cos 90^\circ + Z_B \cos(90^\circ - \theta) \\ Y_E &= X_B \cos 90^\circ + Y_B \cos 0^\circ + Z_B \cos 90^\circ \\ Z_E &= X_B \cos(90^\circ + \theta) + Y_B \cos 90^\circ + Z_B \cos \theta \end{aligned} \quad (2.5)$$

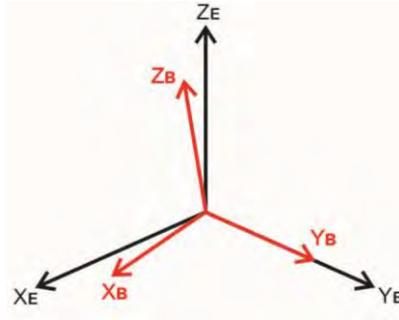
Bentuk *state space* dari (2.5) adalah sebagai berikut:

$$\begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix}$$

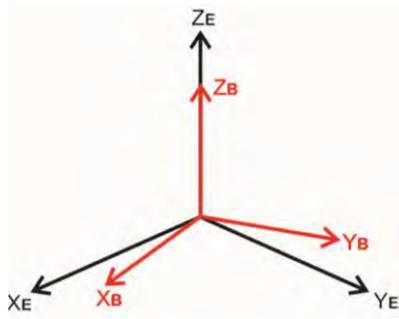
maka matriks rotasi terhadap sumbu Y terdapat pada (2.6).

$$R(y,\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}. \quad (2.6)$$

- Matriks rotasi terhadap sumbu Z dapat direpresentasikan oleh Gambar 2.16, dimana rotasi frame *body* dilakukan terhadap sumbu Z .



Gambar 2.15 Diagram Cartesius Rotasi Terhadap Sumbu Y



Gambar 2.0.16 Diagram Cartesius Rotasi Terhadap Sumbu Z

$$\begin{aligned}
 X_E &= X_B \cos \psi + Y_B \cos(90^\circ + \psi) + Z_B \cos 90^\circ \\
 Y_E &= X_B \cos(90^\circ - \psi) + Y_B \cos \psi + Z_B \cos 90^\circ \\
 Z_E &= X_B \cos 90^\circ + Y_B \cos 90^\circ + Z_B \cos 0^\circ
 \end{aligned} \tag{2.7}$$

Persamaan (2.7) dapat ditulis dalam bentuk (2.8).

$$R(z, \psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.8}$$

Menggunakan (2.4), (2.6) dan (2.8) akan diperoleh bentuk matriks transformasi dengan formula sebagai berikut:

$${}^B_E R = R_{(x,\phi)} + R_{(y,\theta)} + R_{(z,\psi)} = \begin{bmatrix} c\theta c\psi & s\psi c\theta & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \tag{2.9}$$

dengan $s(\cdot)$ menotasikan $\sin(\cdot)$ dan $c(\cdot)$ menotasikan $\cos(\cdot)$.

Adapun matriks transformasi dari *frame body* ke koordinat *frame* bumi ditunjukkan oleh persamaan berikut:

$${}^E_B R = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ s\psi c\theta & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix}. \quad (2.10)$$

Model kinematika dan dinamika Quadrotor diturunkan berdasarkan formula Newton-Euler dengan beberapa asumsi penyederhanaan. Percepatan gravitasi dianggap konstan dan tegak lurus terhadap permukaan bumi, posisi pusat massa tepat berada di tengah, efek momentum badan quadrotor pada gerakan translasi diabaikan ketika quadrotor bergerak perlahan. *Frame* yang digunakan pada quadrotor dianggap *rigid* dan simetris.

2.2.3 Model Dinamika Translasi Quadrotor

Gaya pada quadrotor dihasilkan oleh keempat motor penggerak yang dipasang baling-baling. Sebelum menganalisa model dinamika dari quadrotor, akan dijelaskan mengenai gaya-gaya yang bekerja pada quadrotor. Gaya dan momen aerodinamis pada quadrotor bernilai kecil sehingga dapat diabaikan dalam proses pemodelan. Persamaan berikut menunjukkan gaya-gaya yang terjadi [9]:

$$U_1 = F_1 + F_2 + F_3 + F_4 \quad (2.11)$$

$$U_2 = F_2 - F_4 \quad (2.12)$$

$$U_3 = F_1 - F_3 \quad (2.13)$$

$$U_4 = F_1 + F_3 - F_2 - F_4 \quad (2.14)$$

dengan d adalah konstanta gaya drag dan F_i adalah gaya angkat (*thrust*) yang dihasilkan oleh tiap motor,

$$F_i = K_T \frac{\omega}{s + \omega} u_i \quad (2.15)$$

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

Dinamika gerak translasi diperoleh berdasarkan persamaan hukum Newton II, yaitu

$$\begin{aligned}\sum F &= m\ddot{v} \\ U_1 + F_g z_E &= m\ddot{v}\end{aligned}\tag{2.16}$$

dengan m adalah massa quadrotor, posisi pusat massa yang berada pada *frame* bumi dinyatakan dengan $v = [x \ y \ z]^T$, total gaya angkat (*thrust*) yang dihasilkan oleh keempat motor dinyatakan dengan U_1 , sedangkan gaya gravitasi dinyatakan oleh $F_g = -mg$ dan $z_E = [0 \ 0 \ 1]^T$.

Pemodelan sumbu translasi terletak pada koordinat bumi, untuk transformasi dari koordinat *frame body* ke koordinat *frame* bumi maka diperlukan matriks rotasi ${}^E_B R$ seperti pada (2.10). Persamaan (2.16) dapat ditulis menjadi

$$U_1 {}^E_B R + F_g z_E = m\ddot{v}\tag{2.17}$$

$$\frac{U_1}{m} {}^E_B R - g z_E = \ddot{v}\tag{2.18}$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \frac{U_1}{m} \begin{bmatrix} C\theta C\psi & -C\phi S\psi + S\phi S\theta C\psi & S\phi S\psi + C\phi S\theta C\psi \\ C\theta S\psi & C\phi C\psi + S\phi S\theta S\psi & -S\phi C\psi + C\phi S\theta S\psi \\ -S\theta & S\phi C\theta & C\phi C\theta \end{bmatrix} \begin{bmatrix} X_B \\ Y_B \\ Z_B \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}\tag{2.19}$$

karena gaya pada quadrotor hanya terjadi pada sumbu Z_B (pada kondisi *hover*), dan total thrust dinotasikan oleh U_1 seperti pada (2.11), sehingga persamaan gerak translasi quadrotor dinyatakan sebagai berikut:

$$\ddot{x} = \frac{U_1}{m} (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi)\tag{2.20}$$

$$\ddot{y} = \frac{U_1}{m} (-\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi)\tag{2.21}$$

$$\ddot{z} = \frac{U_1}{m} \cos \phi \cos \theta - g\tag{2.22}$$

2.2.4 Model Dinamika Rotasi Quadrotor

Persamaan gerak rotasi diperoleh dalam *frame body* menggunakan metode Euler dengan formula berikut:

$$\sum {}^E \tau = J^E \ddot{\Theta}\tag{2.23}$$

$$\sum {}^E \tau = \frac{d}{dt} ({}^E R J {}^B \Omega) = J ({}^E \dot{R} \Omega + {}^E R \dot{\Omega}) \quad (2.24)$$

karena $\dot{R} = RS({}^B \Omega)$ dimana $S({}^B \Omega)$ adalah matriks *skew-symmetric* yang dinyatakan oleh

$$S({}^B \Omega) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}, \quad (2.25)$$

jika ruas kiri dikalikan dengan ${}^B R$ maka (2.24) menjadi

$$\begin{aligned} \sum {}^B R {}^E \tau &= J ({}^B \dot{R} {}^B \Omega + {}^B R {}^B \dot{\Omega}) \\ \sum {}^B R {}^E \tau &= J {}^B R S({}^B \Omega) {}^B \Omega + J {}^B R {}^B \dot{\Omega} \\ \sum {}^B \tau &= \Omega \times J \Omega + J \dot{\Omega} \end{aligned} \quad (2.26)$$

sehingga dapat pula ditulis menjadi (2.27)

$$\begin{aligned} J \dot{\Omega} &= -\Omega \times J \Omega + \tau_j \\ \dot{\Omega} &= J^{-1} (-\Omega \times J \Omega + \tau_f) \end{aligned} \quad (2.27)$$

dengan τ_j merupakan resultan torsi yang dihasilkan oleh masing-masing motor.

Torsi

Quadrotor memiliki batasan rotor yang rigid, sehingga dinamika setiap rotor disekitar sumbu rotasinya dapat diperlakukan sebagai sistem *decouple* dalam variabel kecepatan angular Ω secara umum. Torsi yang digunakan oleh setiap motor dinotasikan oleh τ_{f_i} . Dalam quadrotor, torsi motor berlawanan dengan gaya aerodinamik drag $\tau_{drag} = d\Omega^2$. Menggunakan Hukum II Newton,

$$I_M \dot{\Omega}_i = -\tau_{drag} + \tau_{f_i}. \quad (2.28)$$

dimana I_M adalah momen kecepatan sudut dari motor ke- i dan $d > 0$ merupakan konstanta manuver stationer dalam penerbangan bebas. Dalam keadaan *steady state*, yaitu ketika $\dot{\Omega} = 0$, torsi yaw adalah

$$\tau_{f_i} = d\Omega^2 \quad (2.29)$$

sehingga torsi dalam *frame body* secara umum dapat ditulis menjadi

$$\tau_f = \begin{bmatrix} \sum_{i=1}^4 \tau_{f_i} \\ l(F_2 - F_4) \\ l(F_1 - F_3) \end{bmatrix} = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix}$$

dengan l adalah jarak antara motor dan pusat gravitasi dan $\tau_\psi, \tau_\theta, \tau_\phi$ adalah momen *yawing*, *pitching* dan *rolling*, sehingga dapat ditulis menjadi persamaan berikut:

$$\begin{aligned} \tau_{f_x} &= l(F_2 - F_4) \\ \tau_{f_y} &= l(F_1 - F_3) \\ \tau_{f_z} &= d(F_1 + F_3 - F_2 - F_4) \end{aligned} \quad (2.30)$$

dengan l adalah jarak antara pusat massa quadrotor dengan sumbu rotasi baling-baling dan d adalah koefisien gaya drag.

Matriks $J \in R^{3 \times 3}$ adalah matriks inersia pada *frame body* yang didefinisikan oleh

$$J = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{yx} & J_{yy} & J_{yz} \\ J_{zx} & J_{zy} & J_{zz} \end{bmatrix} \quad (2.31)$$

nilai $J_{xy} = J_{xz} = J_{yx} = J_{yz} = J_{zx} = J_{zy} = 0$ karena quadrotor simetris, sehingga *inverse* dari J adalah

$$J^{-1} = \begin{bmatrix} \frac{1}{J_{xx}} & 0 & 0 \\ 0 & \frac{1}{J_{yy}} & 0 \\ 0 & 0 & \frac{1}{J_{zz}} \end{bmatrix}.$$

Kecepatan sudut dinotasikan oleh Ω yang didefinisikan sebagai

$$\Omega = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (3.32)$$

Nilai $-\Omega \times J\Omega$ dinyatakan dalam (2.33) yang dapat dihitung dengan mensubstitusi (2.31) dan (2.32).

$$-\Omega \times J\Omega = J\Omega \times \Omega$$

$$\begin{aligned}
&= \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \\
&= \dot{\theta}\dot{\phi}(J_{xx} - J_{yy})k + \dot{\phi}\dot{\psi}(J_{zz} - J_{xx})j + \dot{\theta}\dot{\psi}(J_{yy} - J_{zz})i
\end{aligned}$$

atau

$$-\Omega \times J\Omega = \begin{bmatrix} \dot{\theta}\dot{\psi}(J_{yy} - J_{zz}) \\ \dot{\phi}\dot{\psi}(J_{zz} - J_{xx}) \\ \dot{\theta}\dot{\phi}(J_{xx} - J_{yy}) \end{bmatrix} \quad (2.33)$$

Moment gaya τ_j pada (2.30) dan $-\Omega \times J\Omega$ pada (2.33) disubstitusi kedalam (2.27) maka dinamika gerak rotasi quadrotor menjadi seperti berikut:

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = J^{-1} \left(\begin{bmatrix} \dot{\theta}\dot{\psi}(J_{yy} - J_{zz}) \\ \dot{\phi}\dot{\psi}(J_{zz} - J_{xx}) \\ \dot{\theta}\dot{\phi}(J_{xx} - J_{yy}) \end{bmatrix} + \begin{bmatrix} l(F_{T2} - F_{T4}) \\ l(F_{T1} - F_{T3}) \\ d(F_{T1} + F_{T3} - F_{T2} - F_{T4}) \end{bmatrix} \right) \quad (2.34)$$

Jika (2.11)-(2.14) disubstitusi kedalam (2.34) dapat diperoleh persamaan lengkap dinamika gerak rotasi quadrotor adalah sebagai berikut:

$$\ddot{\phi} = \dot{p} = \frac{U_2 l}{J_{xx}} - \frac{qr}{J_{xx}} (J_{zz} - J_{yy}) \quad (2.35)$$

$$\ddot{\theta} = \dot{q} = \frac{U_3 l}{J_{yy}} - \frac{pr}{J_{yy}} (J_{xx} - J_{zz}) \quad (2.36)$$

$$\ddot{\psi} = \dot{r} = \frac{U_4 d}{J_{zz}} - \frac{pq}{J_{zz}} (J_{yy} - J_{xx}) \quad (2.37)$$

2.2.5 Analisa Kestabilan Sistem Linier

Dinamika sistem linier dapat direpresentasikan sebagai berikut:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.38)$$

$$y(t) = Cx(t)$$

dengan $x(t) \in R^n$ adalah vektor state dari sistem, $u(t) \in R^m$ adalah vektor input,

$A \in R^{n \times n}$ adalah matriks state dan $B(t) \in R^{n \times m}$ adalah matriks input.

Hal yang perlu diperhatikan dalam sistem linier adalah kontrolabilitas dari sistem tersebut. Jika sistem *controllable* maka akan mudah untuk membawa sistem dari nilai inisial ke nilai akhir. Sistem (2.38) dikatakan *controllable* jika dan hanya jika [10]:

$$\text{rank}([B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B]) = n \quad (2.39)$$

Kestabilan dan karakteristik natural respon dari sistem linier dapat dipelajari dengan melihat *pole-pole* dari sistem. Pada persamaan *state space*, *pole* sistem sama dengan *eigenvalue* dari matriks A.

Alternatif lain untuk melakukan analisa kestabilan sistem dapat menggunakan teori kestabilan Lyapunov. Berdasarkan metode Lyapunov, sistem dapat dikatakan stabil *asymptotic* dalam pengertian Lyapunov jika memungkinkan untuk menemukan sebuah fungsi $V(x) : R^n \rightarrow R$ yang mengikuti batasan:

Teorema 1 [11]: $V(x=0) = 0$; (2.40)

Teorema 2 [11]: $V(x) > 0, \forall x \neq 0$; (2.41)

Teorema 3 [11]: $\dot{V}(x) < 0, \forall x \neq 0$; (2.42)

Pada sistem linier, fungsi kandidat Lyapunov dinyatakan dengan

$$V(x) = x^T P x \quad (2.43)$$

dengan $P \in R^{n \times n}$ adalah matriks simetris. Kondisi untuk memenuhi batasan pada teorema (2.40)-(2.42) adalah:

- Batasan Teorema 1 selalu terpenuhi untuk $V(x)$ jika titik equilibrium sistem berada pada *origin* ($x = 0$), maka fungsi Lyapunov pada titik equilibrium adalah $0^T P 0 = 0, \forall P$
- Batasan Teorema 2, sistem global *asymptotic* apabila terdapat matriks P positif definit, atau dapat ditulis: $P > 0$,
- Pada teorema 3, persamaan $\dot{V}(x)$ dinyatakan sebagai:

$$\dot{V}(x) = \dot{x}^T P x + x^T P \dot{x} = (Ax + Bu)^T P x + x^T P (Ax + Bu)$$

dengan memilih $u = 0$, maka analisa kestabilan sistem *open-loop* menjadi:

$$\dot{V}(x) = x^T [A^T P + P A] x < 0$$

Maka teorema 3 terpenuhi jika dan hanya jika memenuhi pertidaksamaan:

$$A^T P + PA < 0$$

jika input $u = -Kx$, maka teorema 3 terpenuhi jika dan hanya jika sistem linier memenuhi pertidaksamaan berikut:

$$(A - BK)^T P + P(A - BK) < 0$$

2.2.6 Kontrol *Static Output Feedback* dengan performa H_∞

Static output feedback (SOF) merupakan salah satu dari banyak permasalahan dalam teori sistem kontrol. Penggunaan *output feedback* memungkinkan fleksibilitas dan kesederhanaan dalam implementasi. Terutama, dalam aplikasi yang pengukuran *full-state* biasanya tidak dapat dilakukan. Keuntungan dari desain SOF yaitu hanya memerlukan sinyal yang terukur dari plant untuk dikontrol.

Pada sub-bab ini akan dijelaskan mengenai metode untuk menemukan gain kontroler SOF yang memiliki performa H_∞ untuk permasalahan regulator. Gain kontroler dihitung dengan teknik numerik *iterative LMI*.

Pertimbangkan sistem *linear time-invariant*, dengan input kontrol $u(t)$, output $y(t)$, dan gangguan $w(t)$ diketahui sebagai berikut:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + Dw(t) \\ y(t) &= Cx(t) \end{aligned} \tag{2.44}$$

dengan performa output $z(t)$, sebagai berikut [7]:

$$\|z(t)\|_2 = (x^T Qx + u^T Ru)dt \tag{2.45}$$

dimana $Q > 0, R > 0$. Deskripsi sistem dapat dilihat pada Gambar 2.17.

Untuk permasalahan *output feedback*, solusi permasalahan kontrol LQ regulator pada (2.44) dan (2.45) adalah

$$u(t) = -Ky(t) = -KCx(t) \tag{2.46}$$

dengan definisi matriks (A, B) dikatakan *stabilizable* jika terdapat matriks K yang mana $A - BK$ adalah stabil *asymptotic*. Matriks (A, C) dikatakan *detectable* jika terdapat matriks L yang mana $A - LC$ adalah stabil.

Diasumsikan matriks (A, C) adalah *detectable* dan terdapat matriks L dan $P = P^T \geq 0$ sehingga gain *output feedback* adalah

$$KC = R^{-1}(B^T P + L). \quad (2.47)$$

Kontroler ini didesain untuk meminimalkan pengaruh terburuk dalam performa output ketika terdapat gangguan yang diberikan pada sistem. Pengaruh terburuk gangguan terhadap performa output harus dilemahkan pada level pelemahan (γ) tertentu. Pada [12] didefinisikan bahwa $norm-\infty$ dari fungsi alih $T_{zw}(s)$ dapat dihitung dari nilai maksimal L2 gain $z(t)$ terhadap $w(t)$. Sesuai definisi tersebut, maka $norm-\infty$ dapat dihitung seperti pada persamaan berikut:

$$\|T_{zw}\|_{\infty} = \sup_{\omega} |T_{zw}(j\omega)| = \sup_{\|w(t)\|_2 \neq 0} \frac{\|z(t)\|_2}{\|w(t)\|_2}. \quad (2.48)$$

Kestabilan L2 sangat memegang peran penting dalam analisis dari sebuah sistem. Dalam banyak permasalahan kontrol, sistem direpresentasikan sebagai pemetaan input-output, dari gangguan input w ke performa output z , yang harus bernilai kecil. Dengan sinyal input L2, sistem kontrol didesain untuk membuat gain L2 input-output stabil dan untuk meminimalkan L2 gain [12]. Oleh karena itu, sistem L2 gain dikatakan dilemahkan oleh γ jika

$$\frac{\int_0^{\infty} \|z(t)\|^2 dt}{\int_0^{\infty} \|w(t)\|^2 dt} = \frac{\int_0^{\infty} (x^T Qx + u^T Ru) dt}{\int_0^{\infty} (w^T w) dt} \leq \gamma^2 \quad (2.49)$$

untuk setiap input gangguan w yang *non-zero*.

Persamaan (2.46) akan membuat indeks performa mencapai nilai minimum $J = \frac{1}{2} x^T(0) P x(0)$, dimana P adalah solusi definit positif untuk persamaan Riccati. Karena *closed loop* dari sistem stabil, maka diasumsikan bentuk fungsi kuadrat Lyapunov $V(x) = x(t)^T P x(t)$, sehingga dapat diasumsikan $J \leq V(x(0))$. Sistem merupakan sistem linier, sehingga $J \leq V(x(0))$ jika dan hanya jika [13]:

$$\frac{dV(x(t))}{dt} \leq \gamma^2 w(t)^T w(t) - z(t)^T z(t). \quad (2.50)$$

dimana $w(t)$ adalah $w(t) = \frac{1}{\gamma^2} D^T P x(t)$.

2.2.7 Command-Generator Tracker

Command-Generator Tracker (CGT) merupakan struktur desain kontrol yang memberikan kompensator sesuai dengan kebutuhan sistem untuk mendapatkan *error tracking* yang kecil untuk input referensi $r(t)$ yang diinginkan. Pendekatan CGT ini akan menggabungkan model dinamika dari $r(t)$ kedalam sistem kontrol. Oleh karena itu, dinamika sistem kontrol *tracking* akan terdiri dari gabungan dinamika plant dan kompensator.

Persamaan *plant* dan output terukur seperti pada (2.44) dan performa output adalah $z(t) = Hx(t)$. (2.51)

Inisial kondisi dari sinyal referensi $r(t)$ memenuhi persamaan differensial berikut [8]:

$$r^{(d)} + a_1 r^{(d-1)} + \dots + a_d r = 0 \quad (2.52)$$

dengan d adalah derajat dan a_i adalah koefisien. Jika (2.52) ditulis dalam bentuk variabel state (kanonik observabel), untuk kasus $d = 3$, maka

$$\dot{\rho} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix} \rho \equiv G\rho. \quad (2.53)$$

Karakteristik polynomial dari *command generator* didefinisikan sebagai

$$\Delta(p) = p^d + a_1 p^{d-1} + \dots + a_d \quad (2.54)$$

dimana p menotasikan d/dt dalam domain waktu. Kemudian (2.52) dapat ditulis dalam domain p seperti berikut:

$$\Delta(p)r = 0. \quad (2.55)$$

Dalam teknik kontrol ini, permasalahan *tracking* akan diubah kedalam permasalahan regulator dimana error harus diatur menjadi nol. Untuk menghasilkan sistem *tracking* secara keseluruhan maka dilakukan modifikasi.

Didefinisikan vektor state modifikasi adalah

$$\xi = \Delta(p)x = x^d + a_1 x^{d-1} + \dots + a_d x \quad (2.56)$$

dan vektor error serta turunannya dinyatakan oleh

$$\varepsilon(t) = [e \quad \dot{e} \quad \dots \quad e^{(d-1)}]^T \quad (2.57)$$

maka error *tracking* dapat ditulis menjadi

$$\Delta(p)e = \Delta(p)r - \Delta(p)Hx = -H\xi. \quad (2.58)$$

Persamaan (2.58) dapat ditulis dalam bentuk kanonik seperti pada (2.59).

$$\dot{\varepsilon} = \begin{bmatrix} G & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \varepsilon \\ \xi \end{bmatrix} + \begin{bmatrix} 0 \\ -H \end{bmatrix} \begin{bmatrix} \varepsilon \\ \xi \end{bmatrix} \quad (2.59)$$

Persamaan dinamika dari ξ diperoleh dari (2.56) yang diopersikan dalam $\Delta(p)$

$$\dot{\xi} = A\xi + B\mu, \quad (2.60)$$

dimana input kontrol modifikasi adalah

$$\mu = \Delta(p)u = u^{(d)} + a_1 u^{(d-1)} + \dots + a_d u. \quad (2.61)$$

kemudian dinamika pada (2.59) dan (2.60) dibuat kedalam sebuah state *augmented* adalah sebagai berikut [8]:

$$\frac{d}{dt} \begin{bmatrix} \varepsilon \\ \xi \end{bmatrix} = \begin{bmatrix} G & 0 \\ 0 & A \end{bmatrix} \begin{bmatrix} \varepsilon \\ \xi \end{bmatrix} + \begin{bmatrix} 0 \\ B \end{bmatrix} \mu. \quad (2.62)$$

Desain regulator LQ diterapkan pada sistem ini, karena jika state pada sistem ini menuju nol, maka *error tracking* $e(t)$ akan hilang. Jika state output yang akan menjadi *feedback* bagi sistem, maka input kontrol sistem modifikasi menjadi

$$\mu = -[K_\varepsilon \quad K_\xi] \begin{bmatrix} \varepsilon \\ \xi \end{bmatrix} \quad (2.63)$$

Dalam permasalahan tesis ini, konsep CGT diterapkan dalam sistem kontrol *tracking* untuk quadrotor dan nilai gain K dicari menggunakan metode regulator LQ dengan *output feedback*. Jika sistem *closed loop* dari (2.44) dinyatakan oleh $\dot{x} = (A - BKC)x = A_c x$.

dengan indeks performa (PI) kuadratik untuk permasalahan ini adalah

$$J = \int_0^\infty (x^T Q x + u^T R u) dt \quad (2.64)$$

dengan $Q \geq 0$, $R > 0$ adalah matriks pembobot semidefinit positif. Diasumsikan sistem *closed-loop* adalah stabil *asymptotic* sehingga state $x(t)$ akan bernilai nol hingga waktu yang tak hingga, maka J menjadi

$$J = \frac{1}{2} (x^T(0)Px(0)) \quad (2.65)$$

Agar dapat menemukan matriks positif-semidefinit P , maka permasalahan optimisasi dapat dikonversi kedalam persamaan yang memenuhi

$$\frac{d}{dt}(x^T Px) = -J$$

$$x^T (A_c^T P + PA_c)x = -x^T (Q + C^T K^T RKC)x. \quad (2.66)$$

Kondisi ini berlaku untuk semua inisial kondisi oleh karena itu, untuk semua *state trajectory* $x(t)$, dapat ditulis menjadi

$$g \equiv A_c^T P + PA_c + C^T K^T RKC + Q = 0 \quad (2.67)$$

jika K dan Q diketahui dan P diperoleh, maka (2.67) disebut persamaan Lyapunov.

Inisial kondisi $x(0)$ saja yang diperlukan untuk menghitung biaya *closed-loop* dibawah pengaruh input kontrol $u = -Ky$. Untuk menghitung gain K yang meminimumkan indeks performa dapat menggunakan identitas *trace* dari matriks A dan B yaitu, $tr(AB) = tr(BA)$. sehingga PI pada (2.64) dapat ditulis menjadi

$$J = \frac{1}{2} tr(PX) \quad (2.68)$$

dimana $X \equiv \{x(0)x^T(0)\}$

namun, ketergantungan J pada inisial kondisi $x(0)$ tidak diharapkan karena membuat gain optimal tergantung pada nilai inisial kondisi. Permasalahan ini merupakan tipikal dari permasalahan desain *output-feedback*. Oleh karena itu, untuk mengatasi permasalahan ini, dilakukan dengan tidak meminimumkan PI tetapi meminimumkan nilai ekspektasi dari PI. Persamaan (2.68) dapat diganti menjadi

$$J = \frac{1}{2} E\{x^T(0)Px(0)\} = \frac{1}{2} tr(PX), \quad (2.69)$$

dengan $X \equiv E\{x(0)x^T(0)\}$ adalah nilai inisial autokorelasi dari state.

Fungsi Hamiltonian untuk permasalahan LQR *output feedback* ini adalah

$$H = tr(PX) + tr(gS) \quad (2.70)$$

dengan S adalah matriks simetris $n \times n$ dari perkalian Lagrange yang harus ditentukan. Kondisi perlu untuk solusi LQR dengan *output feedback* adalah [8]:

$$\begin{aligned}\frac{\partial H}{\partial S} &= A_c^T P + P A_c + C^T K^T R K C + Q = 0 \\ \frac{\partial H}{\partial P} &= A_c S + S A_c^T + X = 0 \\ \frac{1}{2} \frac{\partial H}{\partial K} &= R K C S C^T - B^T P S C^T = 0.\end{aligned}\tag{2.71}$$

Jika R adalah positif definit dan $C S C^T$ adalah non-singulir, maka gain K didapat sebagai berikut [8]

$$K = R^{-1} B^T P S C^T (C S C^T)^{-1}.\tag{2.72}$$

2.2.8 Linear Matrix Inequality (LMI)

Kestabilan sistem dapat diketahui jika terdapat fungsi Lyapunov yang memenuhi (2.41)-(2.42). Jika fungsi Lyapunov didefinisikan dalam $V(x) = x^T P x$, maka sistem dikatakan stabil jika memenuhi (2.41) dan $P > 0$. Untuk sistem yang sederhana, matriks P dapat dicari melalui perhitungan analitik. Namun untuk sistem orde tinggi, perhitungan analitik tidak mudah untuk dilakukan. Perhitungan numerik menjadi pilihan terbaik dengan menggunakan metode *Linear Matrix Inequalities* (LMI). LMI memiliki bentuk seperti berikut [14]:

$$F(x) = F_0 + \sum_{i=1}^m x_i F_i > 0\tag{2.73}$$

dengan $F_i = F_i^T \in R^{n \times n}$, $i = 0, 1, \dots, m$ adalah matriks yang diketahui dan $\mathbf{x} \in R^m$ adalah variabel. Terlihat dari (2.73), LMI merupakan fungsi kendala dari x . LMI adalah sebuah metode yang digunakan untuk mencari nilai x sedemikian hingga $F(x) > 0$. Jika terdapat lebih dari satu LMI, seperti $F^{(1)}(x) < 0, \dots, F^{(p)}(x) < 0$ maka beberapa LMI tersebut dapat disederhanakan menjadi satu LMI saja, seperti pertidaksamaan berikut ini:

$$\bar{F}(x) = \text{diag}([F^{(1)} \quad \dots \quad F^{(p)}(x)]) < 0 \rightarrow \begin{cases} F^{(1)}(x) < 0 \\ \vdots \\ F^{(p)}(x) < 0 \end{cases}\tag{2.74}$$

Oleh karena itu, penyelesaian beberapa LMI dapat disederhanakan menjadi satu $\bar{F}(x) < 0$. Kendala pertidaksamaan pada contoh diatas merupakan kombinasi linear dari variabel x . Jika kendala merupakan pertidaksamaan yang nonlinear, maka kendala tersebut dapat diubah menjadi LMI dengan menggunakan *Schur Complement*. Contoh jika terdapat LMI

$$L(x) = \begin{bmatrix} E(x) & F(x) \\ G(x) & H(x) \end{bmatrix} < 0 \quad (2.75)$$

dengan $E(x) = E(x)^T$, $H(x) = H(x)^T$ dan $G(x) = F(x)^T$ maka LMI (2.75) merupakan penyederhanaan dari fungsi kendala pertidaksamaan nonlinear pada (2.76).

$$L(x) < 0 \rightarrow \begin{cases} H(x) < 0 \\ E(x) - F(x)H(x)^{-1}G(x) < 0. \end{cases} \quad (2.76)$$

2.2.9 Blok Koreksi

Pada diagram blok sistem kontrol quadrotor yang terdapat dalam Gambar 3.3, terdapat sebuah blok koreksi. Blok koreksi berfungsi untuk memberikan hubungan antara input kontrol virtual (U_x, U_y) dengan sudut referensi (ϕ_d, θ_d). Dimana U_x dan U_y adalah output dari kontroler posisi pada sumbu x dan y berturut-turut, sedangkan ϕ_d dan θ_d adalah input referensi yang dibutuhkan oleh kontroler rotasi. Berdasarkan (2.16)-(2.18), pergerakan quadrotor dapat dinyatakan menjadi:

$$\dot{V} = -gz_E + \frac{F_T}{m} Rz_E \quad (2.77)$$

dimana F_T adalah total gaya angkat (*thrust*) yang dihasilkan oleh empat motor, $Z_E = (0,0,1)^T$ adalah unit vektor pada frame bumi, m adalah massa dan R adalah matriks rotasi.

Definisikan vektor dari input kontrol virtual sebagai $U_v = \ddot{v} = (U_x, U_y, U_z)^T$.

Pada (3.62) ganti \dot{V} dengan U_v sehingga menghasilkan [1]:

$$U_v = -gz_E + \frac{F_T}{m} Rz_E \quad (2.78)$$

BAB 3

PERANCANGAN SISTEM

3.1 Linierisasi Dinamika Quadrotor

Model dinamika quadrotor yang terdapat pada (2.20)-(2.22) dan (2.35)-(2.37) dilinierisasi dengan asumsi kondisi quadrotor dalam posisi melayang (*hover*) dimana sudut *yaw* bernilai 0 rad dan kecepatan sudut *roll*, *pitch*, dan *yaw* mendekati 0 rad/s. Gaya angkat (*thrust*) yang dibangkitkan oleh setiap baling-baling dimodelkan dalam bentuk sistem orde satu terdapat pada (2.15). Nilai parameter plant diperoleh dari [9], yang dituliskan dalam Tabel 3.1. Variabel state v , akan digunakan untuk merepresentasikan dinamika aktuator yang didefinisikan sebagai berikut

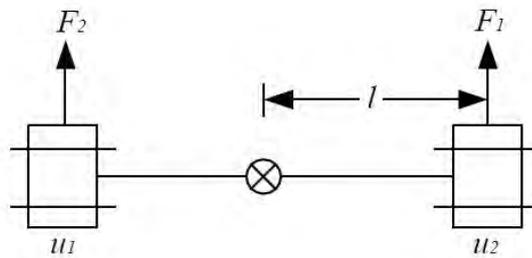
$$v = \frac{\omega}{s + \omega} u. \quad (3.1)$$

Model Roll/Pitch

Jika konstanta drag diabaikan dan quadrotor dianggap dalam kondisi melayang dimana percepatan quadrotor mendekati 0 rad/s maka (2.35)-(2.36) dapat dituliskan kembali menjadi

$$\ddot{\phi} = \dot{p} = \frac{U_2 l}{J_{xx}} \quad (3.2)$$

$$\ddot{\theta} = \dot{q} = \frac{U_3 l}{J_{yy}} \quad (3.3)$$



Gambar 3.1 Model dari sumbu *roll/pitch* [9]

Tabel 3.1 Parameter Sistem Quanser Qball-X4 [9]

| Parameter | Notasi | Nilai |
|-------------------------------------|-------------|------------------------|
| Gain penguat | K_T | 120 |
| Bandwidth | ω | 15 rad/detik |
| Moment Inersia Sekitar Sumbu X | J_{roll} | 0.03 Kg.m ² |
| Moment Inersia Sekitar Sumbu Y | J_{pitch} | 0.03 Kg.m ² |
| Moment Inersia Sekitar Sumbu Z | J_{yaw} | 0.04 Kg.m ² |
| Massa | m | 3.499 Kg |
| Jarak dari motor ke pusat quadrotor | l | 0.2 m |
| Gravitasi | g | 9.81 m/s ² |

dengan menerapkan asumsi rotasi disekitar sumbu x dan y adalah *decoupled*, pergerakan dalam sumbu *roll/pitch* dapat dimodelkan seperti pada Gambar 3.1.

Terlihat pada Gambar 3.1, kedua baling-baling berpengaruh terhadap gerakan dalam setiap sumbu. Gaya angkat yang dihasilkan oleh setiap motor dapat dikalkulasi dari (2.15) dan menggunakan input masing-masing yang berhubungan. Rotasi disekitar pusat gravitasi menghasilkan perbedaan dalam pembangkitan gaya angkat. Sudut *roll/pitch*, misal θ , dapat diformulasikan menggunakan dinamika berikut:

$$J\ddot{\theta} = \Delta Fl \quad (3.4)$$

dengan $J = J_{roll} = J_{pitch}$

l adalah jarak antara baling-baling dengan pusat gravitasi, dan

$$\Delta F = F_1 - F_2$$

menggambarkan perbedaan antara gaya yang dibangkitkan oleh kedua motor. Diketahui bahwa, perbedaan gaya dihasilkan oleh perbedaan input untuk kedua motor yaitu,

$$\Delta u = u_1 - u_2.$$

Persamaan (3.2) dan (3.3) dapat juga ditulis dalam model berikut:

$$\ddot{\phi} = \dot{p} = \frac{U_2 l}{J_{xx}} = \frac{l}{J} K_T \frac{\omega}{s + \omega} \Delta u_2 \quad (3.5)$$

$$\ddot{\theta} = \dot{q} = \frac{U_3 l}{J_{yy}} = \frac{l}{J} K_T \frac{\omega}{s + \omega} \Delta u_1 \quad (3.6)$$

dengan $\Delta u_1 = u_1 - u_3$ dan $\Delta u_2 = u_2 - u_4$

Jika (3.5)-(3.6) dan (3.1) dikombinasikan, maka bentuk persamaan *state space* untuk model linier dinamika *roll* dan *pitch* dapat ditulis menjadi:

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{lK_T}{J} \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \Delta u_2 \quad (3.7)$$

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{lK_T}{J} \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \Delta u_1. \quad (3.8)$$

Model Yaw

Gerakan pada sumbu *yaw* disebabkan oleh perbedaan torsi yang dihasilkan akibat perputaran dua baling-baling searah jarum jam dan dua baling-baling berlawanan arah jarum jam. Model *yaw* dapat dilihat dalam Gambar 3.2. Pergerakan dalam sumbu *yaw* dapat dimodelkan mengikuti persamaan berikut:

$$J_{zz} \ddot{\psi} = \Delta \tau \quad (3.9)$$

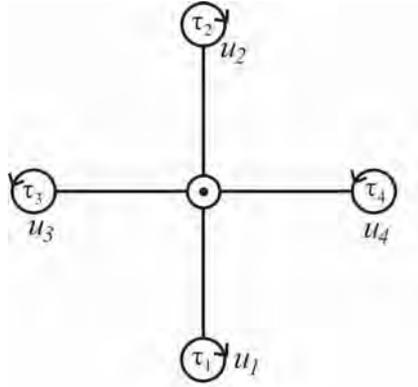
dengan τ adalah torsi yang dibangkitkan oleh setiap motor. Hubungan torsi dengan input motor didefinisikan oleh:

$$\tau = K_T u \quad (3.10)$$

K_T adalah gain positif dimana nilai parameter terdapat dalam Tabel 3.1. Oleh karena itu, dari kombinasi (2.37) dan (3.10) diperoleh model

$$\ddot{\psi} = \dot{r} = \frac{U_4}{J_{zz}} = \frac{K_T}{J_{zz}} \Delta u \quad (3.11)$$

dengan $\Delta u = u_1 + u_2 + u_3 + u_4$.



Gambar 3.2 Model dari sumbu yaw dengan arah putaran baling-baling

Model linier dari dinamika gerakan yaw dapat ditulis dalam bentuk *state space* yaitu

$$\begin{bmatrix} \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_T}{J_{zz}} \end{bmatrix} \Delta u. \quad (3.12)$$

Model Posisi X-Y

Pergerakan quadrotor sepanjang sumbu X dan Y disebabkan oleh gaya angkat total dan perubahan sudut *roll/pitch*. Asumsikan sudut yaw nol dan dinamika gerak dalam sumbu X dan Y dari (2.20) dan (2.21) dapat ditulis sebagai:

$$\ddot{x} = \frac{4F}{m} (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \quad (3.13)$$

$$\ddot{y} = \frac{4F}{m} (-\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi) \quad (3.14)$$

Jika diasumsikan percepatan sudut *roll* dan *pitch* mendekati nol, maka dengan mensubstitusi (3.1), model posisi sumbu X dan Y dapat ditulis menjadi:

$$\ddot{x} = \frac{4K_T}{m} v \sin \theta \approx \frac{4K_T}{m} v \theta$$

$$\ddot{y} = \frac{4K_T}{m} v (-\sin \phi) \approx -\frac{4K_T}{m} v \phi$$

Bentuk *state space* dari model linier dinamika posisi pada sumbu X dan Y adalah sebagai berikut:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{4K_T}{m}\theta \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} u \quad (3.15)$$

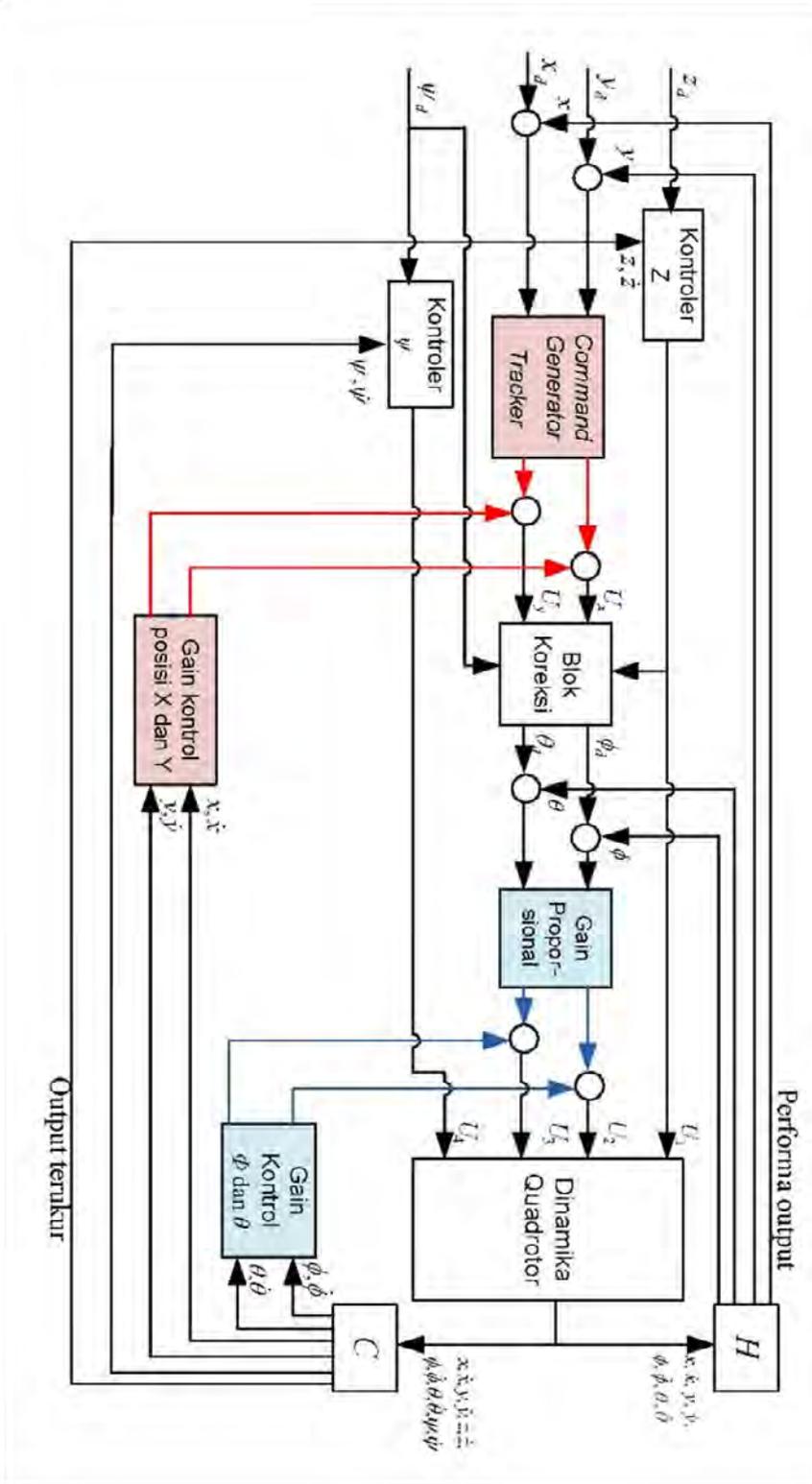
$$\begin{bmatrix} \dot{y} \\ \ddot{y} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -\frac{4K_T}{m}\phi \\ 0 & 0 & -\omega \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} u \quad (3.16)$$

3.2 Strategi Kontrol

Strategi kontrol Quadrotor yang telah dirancang dapat dilihat pada Gambar 3.3. Struktur kontrol terdiri *inner* dan *outer loop*. *Inner loop* (yang ditandai dengan blok berwarna biru) merupakan kontroler rotasi sudut (ϕ, θ, ψ), sedangkan *outer loop* (blok berwarna merah) merupakan kontroler *tracking* posisi (x, y dan z). Sinyal referensi untuk sistem kontrol quadrotor ini terdiri atas referensi untuk posisi x_d, y_d, z_d dan sudut ψ_d .

Sinyal referensi posisi x, y dibandingkan dengan sinyal aktualnya sehingga terdapat error *tracking*. Error *tracking* akan diproses pada blok *Command-Generator Tracker* dimana output dari CGT akan dibandingkan dengan output dari blok *gain output feedback* untuk posisi x dan y . Output dari kontroler posisi x dan y adalah input virtual yang dinotasikan oleh U_x dan U_y . Blok koreksi memberikan hubungan antara input virtual dengan referensi sudut yang diinginkan (ϕ_d, θ_d).

Pada kontroler rotasi, sinyal sudut referensi ϕ_d, θ_d, ψ_d dibandingkan dengan sinyal aktualnya sehingga menghasilkan error *tracking* rotasi. Error ini akan diolah, kemudian output dari kontroler rotasi adalah U_2, U_3 dan U_4 yang merupakan input kontrol untuk quadrotor. Sedangkan input kontrol U_1 dihasilkan dari output kontroler posisi z .

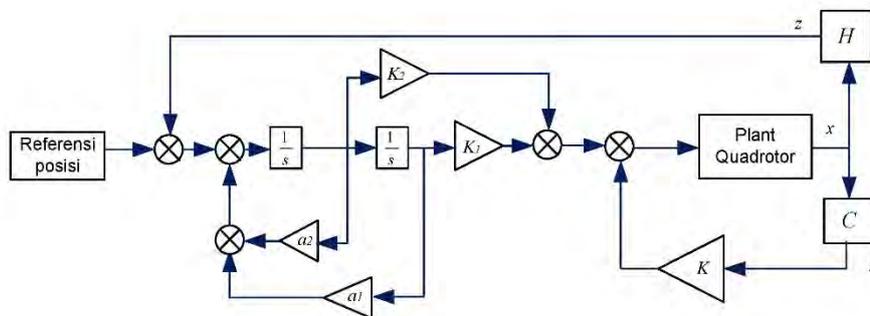


Gambar 3.3 Diagram Blok Sistem Kontrol Quadrotor Hasil Perancangan

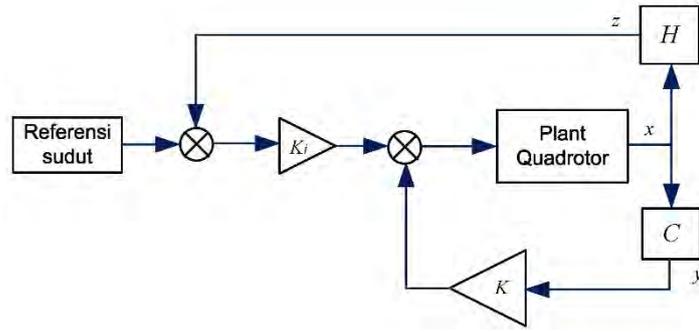
3.3 Prosedur Perancangan Sistem

Dalam merancang kontroler untuk quadrotor, akan ditentukan terlebih dahulu kriteria performa *tracking* yang ingin dicapai, sehingga karakteristik yang diinginkan dari *outer loop* telah diketahui. Kemudian, setelah menentukan karakteristik *outer loop*, maka dirancang kontroler *inner loop* sehingga dinamika *inner loop* harus lebih cepat dari dinamika *outer loop*. Adapun langkah-langkah desain kontroler untuk quadrotor pada tesis ini adalah:

1. Menentukan spesifikasi *outer loop* (kontroler posisi)
 - *Tracking* posisi pada sumbu X dan Y dilakukan ketika quadrotor sudah dalam kondisi melayang.
 - Karena solusi penyelesaian menggunakan metode output feedback, maka matriks (\sqrt{Q}, A) harus *detectable*.
 - Sistem *closed loop* dari dinamika posisi X dan Y harus memiliki *eigenvalue* bernilai negatif, penempatan pole *closed loop* lebih mendekati titik 0.
 - Kontroler *outer loop* dikhususkan untuk permasalahan *tracking* dengan sinyal referensi berbentuk lingkaran.
2. Menentukan spesifikasi *inner loop* (kontroler rotasi)
 - Penempatan pole-pole *closed loop* dari sistem rotasi harus berada minimal 2 kali lebih jauh dari penempatan pole dinamika translasi, atau *eigenvalue* dari *inner loop* $> (\text{minimal } 2 \text{ s/d } 4) * \text{outer loop}$.
3. Berikut adalah rancangan diagram blok untuk *outer loop* dan *inner loop*



Gambar 3.4 Diagram blok kontroler *outer loop*



Gambar 3.5 Diagram blok kontroler *inner loop*

3.4 Perancangan Kontroler Rotasi

Dinamika sistem quadrotor yang telah dilinierisasi dinyatakan oleh persamaan LTI pada (2.44)-(2.45). Sistem dikatakan memiliki performa H_∞ jika memenuhi pertidaksamaan (2.50). Pada perancangan ini, gain kontrol akan dicari menggunakan teknik numerik *iterative* LMI. Untuk merumuskan bentuk LMI, jika didefinisikan indeks performa dari (2.50) adalah sebagai berikut:

$$J = \int_0^{\infty} (\gamma^2 w(t)^T w(t) - z(t)^T z(t)) dt \quad (3.17)$$

dengan $\gamma > 0$. Dalam fungsi biaya ini, kuadrat norm dari input dan output didalam integral dikalikan dengan pembobot skalar yang berlawanan tanda.

Sistem (2.44) memiliki performa H_∞ jika dan hanya jika terdapat fungsi Lyapunov yang memenuhi (2.50). Selanjutnya, nilai $V(x)$ dari (2.50) diturunkan terhadap waktu,

$$\begin{aligned} \frac{dV(x(t))}{dt} &= ((A - BKC)x + Dw)^T Px + x^T P((A - BKC)x + Dw) \\ \frac{dV(x(t))}{dt} &= x^T ((A - BKC)^T P + P(A - BKC))x + w^T D^T Px + x^T PDw \end{aligned} \quad (3.19)$$

kemudian dilakukan manipulasi perhitungan sehingga diperoleh,

$$\begin{aligned} \frac{dV(x(t))}{dt} &= x^T ((A - BKC)^T P + P(A - BKC))x - \left(\frac{1}{\gamma} D^T Px - \gamma w\right)^T \left(\frac{1}{\gamma} D^T Px - \gamma w\right) \\ &\quad + \frac{1}{\gamma^2} x^T PDD^T Px + \gamma^2 w^T w \end{aligned} \quad \dots (3.20)$$

Jika (3.17) disubstitusi kedalam (2.50) maka,

$$\begin{aligned} \frac{dV(x)}{dt} &\leq x(t)^T \left(-\frac{1}{\gamma^2} PDD^T P - Q - C^T K^T RKC \right) x(t) \\ x(t)^T \left[(A - BKC)^T P + P(A - BKC) + \frac{1}{\gamma^2} PDD^T P + Q + C^T K^T RKC \right] x(t) &\leq 0 \\ x(t)^T \left[A^T P + PA + Q - PBKC - C^T K^T B^T P + C^T K^T RKC + \frac{1}{\gamma^2} PDD^T P \right] x(t) &\leq 0 \end{aligned} \quad \dots (3.21)$$

Substitusi gain K pada (2.47) kedalam (3.21), sehingga menghasilkan pertidaksamaan sebagai berikut:

$$A^T P + PA + Q + \frac{1}{\gamma^2} PDD^T P - PBR^{-1}B^T P + L^T R^{-1}L \leq 0 \quad (3.22)$$

Dengan menerapkan metode *Schur complement*, maka bentuk *iterative* LMI dari (3.22) adalah sebagai berikut:

$$\begin{bmatrix} P_n A + A^T P_n + Q + L_n^T R^{-1} L_n & P_n B & P_n D \\ B^T P_n & R & 0 \\ D^T P_n & 0 & -\gamma^2 I \end{bmatrix} \leq 0, \quad (3.23)$$

dimana n menotasikan iterasi ke- n . Untuk lebih jelas, proses mendapatkan bentuk LMI (3.23) dapat dilihat pada Lampiran A.

Kontroler output feedback dengan performa H_∞ dapat secara efektif diselesaikan dengan mengikuti permasalahan LMI optimisasi berikut:

$$\begin{aligned} &\text{minimize } \gamma^2 \\ &\text{subject to } P > 0, \gamma > 0, S(P, \gamma) < 0. \end{aligned} \quad (3.24)$$

Jika permasalahan optimisasi (3.24) dicapai, dengan mendapatkan nilai matriks P , maka gain kontroler *static output feedback* dinyatakan oleh

$$K = R^{-1}(B^T P + L)C^T (CC^T)^{-1} \quad (3.25)$$

Diagram blok yang didesain untuk kontrol stabilisasi ini dapat dilihat pada Gambar 3.3. State sistem hasil linierisasi untuk model *roll*, *pitch* dan *yaw* menjadi $x_{inner} = [\phi \ p \ v \ \theta \ q \ v \ \psi \ r]^T$.

Dalam perancangan kontroler rotasi, variabel utama yang dikontrol adalah sudut *roll* dan *pitch*, juga tiga kecepatan sudut yaitu kecepatan sudut *roll*, *pitch* dan *yaw*. Oleh karena itu, vektor output dari kontroler ini adalah

$$y_{inner} = [\phi \ \theta \ \psi \ p \ q]^T.$$

Simulasi dilakukan dengan nilai parameter matriks sistem sebagai berikut

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 800 \\ 0 & 0 & -15 \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 15 \end{bmatrix} \Delta u \quad (3.26)$$

$$y_{\phi} = C_{\phi} \begin{bmatrix} \phi \\ \dot{\phi} \\ v \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.1 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v \end{bmatrix}$$

dengan nilai performa output yang diinginkan adalah

$$z_{\phi} = Hx_{\phi} = \begin{bmatrix} 0.3 & 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ v \end{bmatrix} \quad (3.27)$$

sedangkan matriks sistem untuk sudut *pitch* adalah

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 800 \\ 0 & 0 & -15 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 15 \end{bmatrix} \Delta u \quad (3.28)$$

$$y_{\theta} = C_{\theta} \begin{bmatrix} \theta \\ \dot{\theta} \\ v \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ v \end{bmatrix}$$

$$z_{\theta} = Hx_{\theta} = \begin{bmatrix} 0.3 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ v \end{bmatrix} \quad (3.29)$$

Nilai matriks H dipilih berdasarkan performa output yang diinginkan, yaitu state rotasi disekitar titik equilibrium. Karena nilai $J_{xx} = J_{yy} = 0.03 \text{ Kg.m}^2$, maka dinamika sistem untuk gerak ϕ sama dengan dinamika gerak θ .

Sistem (3.26)–(3.29) akan memenuhi spesifikasi desain yaitu sistem stabil *asymptotic* dan sistem memenuhi performa H_{∞} dengan tingkat pelemahan gangguan $w(t)$ terhadap performa keluaran $z(t)$ kurang dari γ . Untuk memenuhi spesifikasi desain tersebut, maka harus terdapat matriks simetris P yang memenuhi LMI (3.23). Berikut adalah langkah–langkah yang dilakukan dalam simulasi adalah

Tabel 3.2. Tabel Algoritma untuk Kontroler Rotasi

| Algoritma Output feedback dengan performa H_∞ | |
|--|--|
| 1. Inisialisasi: | <p>Set nilai $n = 0$, definisikan $L_0 = 0$</p> <p>tentukan nilai γ, nilai toleransi α, matriks pembobot Q dan R</p> |
| 2. Penyelesaian Iterasi ke- n | <p>Mencari nilai P_n untuk iterasi ke-n pada Persamaan (3.23)</p> <p>Evaluasi nilai K dan update nilai L</p> $K_{n+1} = R^{-1}(B^T P_n + L_n)C^T (CC^T)^{-1}$ $L_{n+1} = RK_{n+1}C - B^T P_n$ <p>jika nilai K_n dan K_{n+1} cukup dekat satu sama lain, lanjut ke langkah 3, sebaliknya, set $n = n + 1$ dan kembali ke langkah 2 (mencari P_n yang <i>feasible</i>)</p> |
| 3. Stop: set $K = K_{n+1}$ | |

Untuk menyelesaikan LMI (3.23), dapat menggunakan LMI *toolbox* yang tersedia dalam *software* Matlab. Algoritma penyelesaiannya dapat dilihat pada Lampiran B.

Hasil terbaik diperoleh dengan nilai parameter sebagai berikut:

$$\gamma = 0.7, R = 10 \text{ dan } Q = \text{diag}\{755, 3, 0.1\}.$$

Adapun hasil matriks P , K , dan L setelah iterasi ke-12 adalah

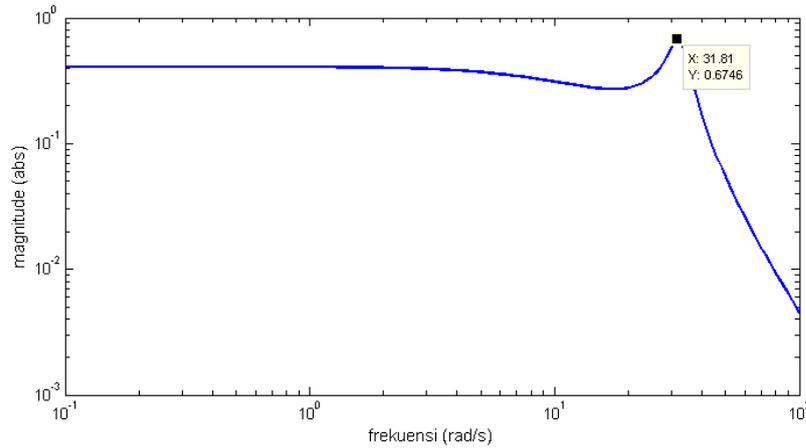
$$P = \begin{bmatrix} 29.8331 & 0.1572 & 0.4897 \\ 0.1572 & 0.0156 & 0.0545 \\ 0.4897 & 0.0545 & 0.3359 \end{bmatrix} \quad (3.30)$$

$$K = [0.7346 \quad 0.0817] \quad (3.31)$$

Fungsi alih lup tertutup dari $Z(s)$ terhadap $W(s)$ sebagai berikut

$$T_{zw}(s) = H(sI - A + BKC)^{-1}D \quad (3.32)$$

$$T_{zw}(s) = \frac{4050}{s^3 + 15s + 1102.7s + 9916.7}$$



Gambar 3.6 Respon frekuensi dari $\|T_{zw}(s)\|_{\infty}$

Selanjutnya dari (3.32) dicari nilai norm- ∞ dari $T_{zw}(s)$. Gambar 3.4 menunjukkan *magnitude* respon frekuensi $\|T_{zw}(s)\|_{\infty}$. Berdasarkan [15], nilai norm- ∞ dari $T_{zw}(s)$ adalah puncak dari nilai singular terbesar dari $T_{zw}(j\omega)$, sehingga pada Gambar 3.4 nilai $\|T_{zw}(s)\|_{\infty}$ yaitu:

$$\|T_{zw}(s)\|_{\infty} = \gamma^* = 0.67 \tag{3.33}$$

persamaan diatas menunjukkan bahwa ∞ -norm dari fungsi alih lup tertutup sistem memiliki tingkat pelemahan $\gamma^* = 0.67$, dimana tingkat plemahan ini kurang dari nilai γ yang dipilih yaitu $\gamma = 0.7$ atau $\|T_{zw}(s)\|_{\infty} < \gamma$. Nilai K akan menjamin pengaruh terburuk gangguan $w(t)$ terhadap performa output $z(t)$.

3.5 Perancangan Kontroler Posisi

Sistem kontrol posisi dirancang dengan tujuan agar respon sistem dapat mengikuti sinyal referensi yang diinginkan. Struktur kontrol yang digunakan pada kontroler posisi ini berbeda dengan struktur kontrol *tracking* yang telah dilakukan oleh [1]-[4], oleh karena itu struktur kontrol ini yang menjadi kontribusi dalam Tesis ini. Untuk menghasilkan *zero tracking error* antara sinyal referensi dan performa output, struktur kontrol *tracking* posisi diperluas (*augmented*) dengan menambahkan *Command-Generator Tracker* (CGT).

3.5.1 Perancangan kontroler posisi pada sumbu X

Objektif kontrol *tracking* posisi adalah membuat quadrotor dapat mengikuti lintasan yang diinginkan dalam kondisi quadrotor sudah melayang (*hover*). Oleh karena itu, sinyal referensi yang diberikan untuk men-*track* posisi X adalah sebuah sinyal sinusoidal dengan frekuensi 0.1047 rad/s atau sama dengan 0.0167 Hz dengan fasa sebesar -0.26 rad. Berikut adalah persamaan sinyal referensi:

$$x_d = -\sin(0.05\pi t) \quad (3.34)$$

jika diubah menggunakan transformasi Laplace, maka didapat fungsi transfernya adalah

$$\frac{-0.05\pi}{s^2 + (0.05\pi)^2} = \frac{-0.157}{s^2 + 0.02} \quad (3.35)$$

sehingga berdasarkan (2.52) karakteristik *polynomial* dari referensi x_d adalah

$$\Delta(p)x_d = p^2 + 0.02 = 0 \quad (3.36)$$

atau dapat ditulis juga dalam bentuk *state space* seperti berikut:

$$\dot{x}_d = \begin{bmatrix} 0 & 1 \\ -0.02 & 0 \end{bmatrix} x_d = Gx_d \quad (3.37)$$

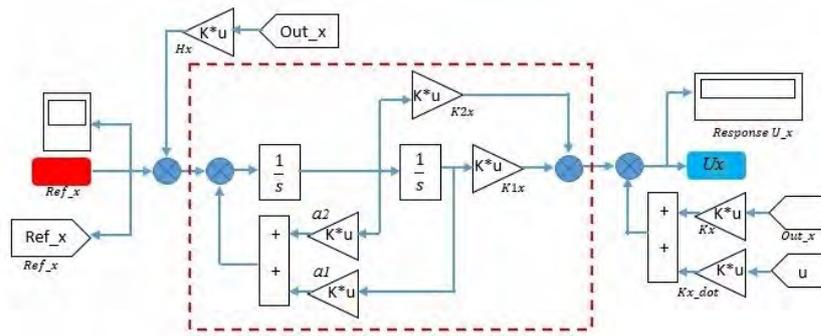
Persamaan state sistem quadrotor untuk posisi sumbu X sama dengan sistem untuk posisi Y, yaitu:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 136.2683 \\ 0 & 0 & -15 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 15 \end{bmatrix} U_1 \quad (3.38)$$

$$y = [1 \ 0 \ 0]x.$$

dengan performa output adalah

$$z = Hx = [1 \ 0 \ 0]x. \quad (3.39)$$



Gambar 3.7 Diagram Simulink Kontroler untuk Posisi X

Sesuai dengan (2.62), dimana dinamika sistem dimodifikasi dengan memperluas (*augmented*) dinamika pada posisi X dengan dinamika sinyal referensi. Berikut adalah persamaan state sistem yang telah dimodifikasi,

$$\dot{\tilde{x}}_x = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -0.02 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 136.2683 \\ 0 & 0 & 0 & 0 & -15 \end{bmatrix} \begin{bmatrix} \varepsilon \\ \dot{\varepsilon} \\ x \\ \dot{x} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 15 \end{bmatrix} U_1, \quad (3.40)$$

dan persamaan output menjadi

$$\tilde{y}_x = \tilde{C}\tilde{x} = \begin{bmatrix} I & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} \varepsilon \\ x \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \varepsilon \\ \dot{\varepsilon} \\ x \\ \dot{x} \\ v \end{bmatrix} \quad (3.41)$$

Diagram Simulink untuk kontrol posisi pada sumbu X dapat dilihat pada Gambar 3.5, dimana yang menjadi bagian dari CGT ditandai oleh garis putus-putus.

3.5.2 Perancangan kontroler posisi pada sumbu Y

Lintasan yang ingin diberikan sebagai sinyal referensi bagi quadrotor adalah berbentuk lingkaran (dalam bidang 3 dimensi). Oleh karena itu, sinyal referensi yang diberikan untuk posisi Y adalah sebagai berikut:

$$y_d = -1 - \cos(0.05\pi t) \quad (3.42)$$

jika diubah kedalam bentuk Laplace menjadi

$$y_d = -\frac{1}{s} - \frac{s^2}{s^2 + (0.05\pi)^2} = \frac{-(s^2 + 0.02) - s^3}{s(s^2 + 0.02)} = \frac{-s^3 - s^2 - 0.02}{s^3 + 0.02s}. \quad (3.43)$$

Persamaan karakteristik *polynomial* adalah

$$\Delta(p)y_d = p^3 + 0.02p = 0 \quad (3.44)$$

Persamaan (3.44) adalah persamaan karakteristik polynomial dari sinyal referensi dengan orde $d = 3$ jika berdasarkan (2.52)-(2.53), maka selanjutnya (3.51) ditulis dalam bentuk matriks state space dengan dimensi 3×3 . Dikarenakan struktur CGT merupakan representasi dari sinyal referensi, tidak semua derivatif dari sinyal referensi perlu digunakan dalam struktur sistem sehingga dalam perancangan ini dilakukan modifikasi, (3.44) dapat ditulis sebagai berikut:

$$\Delta(p)y_d = 0.02p = 0$$

$$\dot{y}_d = \begin{bmatrix} 0 & 1 \\ 0 & -0.02 \end{bmatrix} y_d = G y_d \quad (3.45)$$

dimana y_d menotasikan state referensi sistem pada posisi Y .

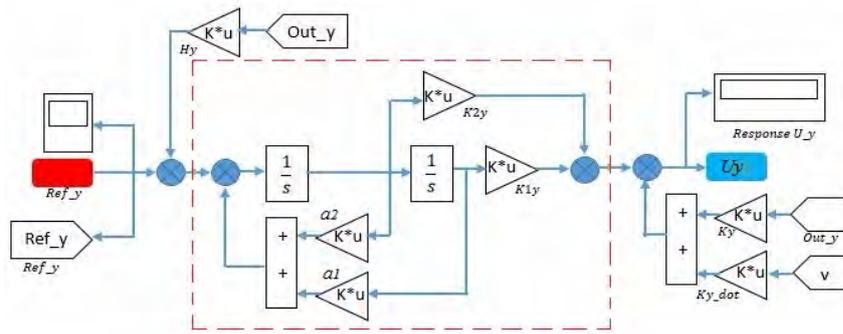
Bentuk dari modifikasi dari dinamika referensi dan dinamika sistem untuk posisi Y ditunjukkan oleh (3.15) dan (3.16) adalah sebagai berikut:

$$\tilde{x}_y = \begin{bmatrix} \dot{\varepsilon} \\ \ddot{\varepsilon} \\ \dot{y} \\ \ddot{y} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.02 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -136.2683 \\ 0 & 0 & 0 & 0 & -15 \end{bmatrix} \begin{bmatrix} \varepsilon \\ \dot{\varepsilon} \\ y \\ \dot{y} \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 15 \end{bmatrix} U_1, \quad (3.46)$$

dan persamaan output menjadi

$$\tilde{y}_y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \varepsilon \\ \dot{\varepsilon} \\ y \\ \dot{y} \\ v \end{bmatrix} \quad (3.47)$$

Diagram Simulink untuk kontrol posisi pada sumbu Y dapat dilihat pada Gambar 3.6, dimana yang menjadi bagian dari CGT ditandai oleh garis putus-putus berwarna merah.



Gambar 3.8 Diagram Simulink Kontroler untuk Posisi Y

Untuk kontroler pada setiap subsistem (subsistem posisi X dan Y) nilai gain kontrol dicari menggunakan metode *LQ regulator* dengan *output feedback*. Kondisi perlu untuk solusi permasalahan tersebut adalah terdapat pada (2.71), sedangkan algoritma kontrol yang digunakan terdapat dalam Tabel 3.3. Nilai-nilai dari parameter yang digunakan untuk mencari nilai gain K ditunjukkan dalam Tabel 3.4.

Tabel 3.3 Algoritma Kontrol untuk Kontroler Posisi

| Algoritma kontrol untuk Kontroler Posisi | |
|---|--|
| 1. | Inisialisasi nilai Q , R dan gain awal K_0 sehingga nilai $A - BK_0C$ adalah stabil <i>asymptotic</i> |
| 2. | Untuk setiap iterasi ke- n : Atur $\tilde{A}_n = \tilde{A} - \tilde{B}\tilde{K}_n\tilde{C}$ kemudian cari nilai P_n dan S_n dalam Persamaan (2.71) |
| 3. | Atur $J_n = tr(P_nX)$ evaluasi nilai gain \tilde{K} dengan $\Delta\tilde{K} = R^{-1}\tilde{B}^T P S \tilde{C}^T (\tilde{C} S \tilde{C}^T)^{-1} - \tilde{K}_n$ kemudian update nilai gain \tilde{K} dengan $\tilde{K}_{n+1} = \tilde{K}_n + \alpha \Delta\tilde{K}$ dimana α dipilih sehingga $\tilde{A}_n = \tilde{A} - \tilde{B}\tilde{K}_n\tilde{C}$ adalah stabil <i>asymptotic</i> dan jika $J_{n+1} \leq J_n$, konvergen (cukup dekat satu sama lain), maka lanjut ke tahap 4, jika selain itu set $n = n + 1$ dan kembali ke tahap 2. |
| 4. | Akhiri dengan set $\tilde{K} = \tilde{K}_{n+1}, J = J_{n+1}$ |

Tabel 3.4 Nilai parameter yang digunakan untuk kontroler posisi X dan Y

| Parameter | Kontroler posisi X | Kontroler posisi Y |
|-----------|----------------------------|-----------------------------------|
| K_0 | [-9.54 -23.52 23.25 10.52] | [31.62 64.76 66.09 -38.20 -10.39] |
| R | 1 | 1 |
| Q | diag{500, 1, 0.1, 500, 10} | diag{400, 5, 25, 40, 1} |
| a | 0.01 | 0.01 |

Untuk kontroler posisi X, matriks P , S dan K diperoleh setelah iterasi ke-13 dengan data sebagai berikut:

$$P_X = \begin{bmatrix} 996.5 & 1012.2 & -519.8 & -19.8 & 3.6 \\ 1012.2 & 1562.8 & -1068 & -42.5 & -7.3 \\ -519.8 & -1068 & 1077.5 & 41.8 & 7.9 \\ -19.8 & -42.5 & 41.8 & 23.2 & 3.8 \\ -3.6 & -7.3 & 7.9 & 3.8 & 7.3 \end{bmatrix} \quad (3.48)$$

$$S_X = \begin{bmatrix} 3.6800 & -0.5000 & 1.5178 & -0.5100 & -0.0574 \\ -0.5000 & 1.5914 & 0.5100 & 1.5643 & -0.0187 \\ 1.5178 & 0.5100 & 1.5340 & -0.5000 & -0.1325 \\ -0.5100 & 1.5643 & -0.5000 & 3.6122 & -0.0183 \\ -0.0574 & -0.0187 & -0.1325 & -0.0183 & 0.9462 \end{bmatrix} \quad (3.49)$$

$$\tilde{K}_X = [-4.4518 \quad -9.6604 \quad 9.4440 \quad 5.3433] \quad (3.50)$$

Untuk kontroler posisi Y, setelah iterasi ke-28 didapat nilai matriks P , S dan K sebagai berikut

$$P_Y = \begin{bmatrix} 575.5783 & 408.1571 & -141.8171 & -4.9340 & 2.6876 \\ 408.1571 & 441.7089 & -195.5831 & -7.2252 & 3.6818 \\ -141.8171 & -195.5831 & 137.8846 & 4.8847 & -2.8078 \\ -4.9340 & -7.2252 & 4.8847 & 1.8853 & -0.8695 \\ 2.6876 & 3.6818 & -2.8078 & -0.8695 & 7.9339 \end{bmatrix} \quad (3.51)$$

$$S_Y = \begin{bmatrix} 1.7843 & -0.50 & 1.2450 & -0.4753 & 0.0160 \\ -0.50 & 1.2350 & 0.4753 & 2.1803 & 0.0033 \\ 1.2450 & 0.4753 & 2.1708 & -0.50 & 0.0706 \\ -0.4770 & 2.1803 & -0.50 & 9.6152 & 0.0037 \\ 0.0160 & 0.0033 & 0.0706 & 0.0037 & 0.3017 \end{bmatrix} \quad (3.52)$$

$$\tilde{K}_Y = [15.9326 \quad 23.4931 \quad -15.8108 \quad -6.2181] \quad (3.53)$$

3.6 Blok Koreksi

Dari (2.78), akan dilakukan perhitungan matematika hingga mendapatkan persamaan input ϕ_d dan θ_d untuk *inner loop*. Pindahkan kecepatan gravitasi ke sebelah kanan persamaan dan kalikan kedua belah sisi dengan R^T , maka (2.78) dapat ditulis menjadi:

$$R^T (U_v + gz_E) = \frac{F_T}{m} z_E \quad (3.54)$$

$$\begin{pmatrix} C\theta C\psi & C\theta S\psi & -S\theta \\ -C\phi S\psi + S\phi S\theta C\psi & C\phi C\psi + S\phi S\theta S\psi & S\phi C\theta \\ S\phi S\psi + C\phi S\theta C\psi & -S\phi C\psi + C\phi S\theta S\psi & C\phi C\theta \end{pmatrix} \begin{pmatrix} U_x \\ U_y \\ U_z + g \end{pmatrix} = \frac{1}{m} \begin{pmatrix} 0 \\ 0 \\ F_T \end{pmatrix} \quad (3.55)$$

dimana $U_z = U_1$ adalah input kontrol yang dihasilkan oleh kontroler posisi Z.

Setelah melakukan perhitungan sederhana, maka didapat persamaan-persamaan berikut:

$$U_x C\theta C\psi + U_y C\theta S\psi - (U_z + g)S\theta = 0 \quad (3.56)$$

$$U_x (-C\phi S\psi + S\phi S\theta C\psi) + U_y (C\phi C\psi + S\phi S\theta S\psi) + (U_z + g)S\phi C\theta = 0 \quad (3.57)$$

$$U_x (S\phi S\psi + C\phi S\theta C\psi) + U_y (-S\phi C\psi + C\phi S\theta S\psi) + (U_z + g)C\phi C\theta = \frac{F_T}{m} \quad (3.58)$$

bagi kedua sisi dari (3.56) dengan $\cos \theta$, sehingga sudut *pitch* yang diinginkan θ_d adalah sebagai berikut:

$$U_x C\psi + U_y S\psi - (U_z + g) \frac{S\theta}{C\theta} = 0$$

$$\theta_d = \arctan\left(\frac{U_x C \psi + U_y S \psi}{(U_z + g)}\right) \quad (3.59)$$

Sudut *roll* yang diinginkan dicari dengan cara mengalikan (3.57) dengan $\cos \phi$

$$U_x(-C^2 \phi S \psi + C \phi S \phi S \theta C \psi) + U_y(C^2 \phi C \psi + C \phi S \phi S \theta S \psi) + (U_z + g)C \phi S \phi C \theta = 0 \quad \dots(3.60)$$

dan mengalikan (3.58) dengan $\sin \phi$ hasilnya adalah,

$$U_x(S^2 \phi S \psi + S \phi C \phi S \theta C \psi) + U_y(-S^2 \phi C \psi + S \phi C \phi S \theta S \psi) + (U_z + g)S \phi C \phi C \theta = \frac{F_T}{m} S \phi \quad \dots (3.61)$$

kemudian kurangi (3.61) dengan (3.60), sehingga menghasilkan

$$U_x \sin \psi - U_y \cos \psi = \frac{F_T}{m} \sin \phi. \quad (3.62)$$

Pindahkan kecepatan gravitasi pada (2.78) ke ruas sebelah kiri, kemudian mengalikan dengan transpose-nya maka akan didapat hubungan seperti dibawah ini

$$(U_v + g z_E)^T (U_v + g z_E) = \left(\frac{F_T}{m} z_E\right)^T \left(\frac{F_T}{m} z_E\right)$$

$$U_x^2 + U_y^2 + (U_z + g)^2 = \left(\frac{F_T}{m}\right)^2$$

$$\sqrt{U_x^2 + U_y^2 + (U_z + g)^2} = \frac{F_T}{m} \quad (3.63)$$

Substitusi (3.63) kedalam (3.62) maka akan dihasilkan sudut *roll* ϕ_d yang diinginkan adalah

$$\phi_d = \arcsin\left(\frac{U_x \sin \psi - U_y \cos \psi}{\sqrt{U_x^2 + U_y^2 + (U_z + g)^2}}\right). \quad (3.64)$$

Persamaan (3.59) dan (3.64) adalah input sudut referensi *attitude* (ϕ_d dan θ_d) untuk *inner loop* dari sistem.

Halaman ini sengaja dikosongkan

BAB 4

HASIL DAN PEMBAHASAN

4.1 Hasil Perancangan Sistem Kontrol

Setelah dilakukan perancangan kontroler, yang telah dijelaskan pada sub-bab 3.4 dan 3.5. Selanjutnya dilakukan analisa pada kontroler hasil desain tersebut. Hal ini ditujukan untuk membuktikan bahwa dinamika *inner loop* lebih cepat daripada dinamika *outer loop*.

1. Hasil Perancangan *Outer Loop* (Kontroler Posisi)

- Mencari *eigenvalue* dari sistem *closed loop* untuk dinamika Posisi X

$$\tilde{A}c_x = \tilde{A}_x - \tilde{B}_x \tilde{K}_x \tilde{C}_x = 0 \quad (4.1)$$

jika nilai matriks (3.40)-(3.41) dan (3.50) disubstitusi ke (4.1), maka diperoleh *eigenvalue* dari sistem *closed loop* ini adalah:

$$\lambda = \{-6.6157 \pm j45.992, -0.8303, -0.4732 \pm j0.9135\}. \quad (4.2)$$

- Mencari *eigenvalue* dari *closed loop* sistem untuk dinamika Posisi Y

$$\tilde{A}c_y = \tilde{A}_y - \tilde{B}_y \tilde{K}_y \tilde{C}_y = 0 \quad (4.3)$$

jika nilai matriks (3.46)-(3.47) dan (3.53) disubstitusi ke (4.3), maka *eigenvalue* dari sistem *closed loop* untuk posisi Y adalah

$$\lambda = \{-6.22 \pm j1.1241, -1.17, -0.7 \pm j1.31\}. \quad (4.4)$$

- Mengecek *detectability* dari matriks (\sqrt{Q}, A) , dengan cara melihat *rank* dari matrik tersebut,

$$O = \begin{bmatrix} \sqrt{Q} \\ \sqrt{Q}, \tilde{A} \\ \vdots \\ \sqrt{Q}, \tilde{A}^4 \end{bmatrix} \quad (4.5)$$

Jika nilai matriks Q pada Tabel 3.4 dan \tilde{A} pada (3.40) dan (3.46) disubstitusi kedalam (4.5), maka diperoleh bahwa sistem memiliki $rank = 5$, atau rank penuh, sehingga kondisi *detectability* terpenuhi.

2. Hasil Perancangan Inner Loop (Kontroler Rotasi)

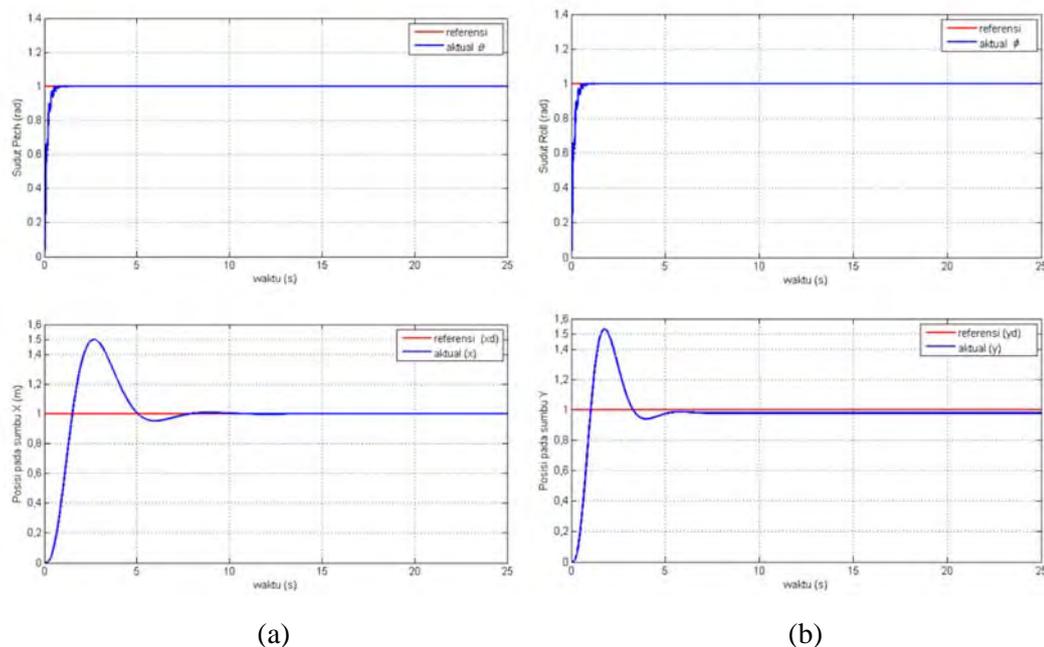
- Mencari *eigenvalue* dari sistem rotasi,

Dikarenakan dinamika rotasi ϕ sama dengan dinamika rotasi θ maka, *eigenvalue* dari sistem *closed loop* untuk dinamika ϕ dan θ adalah

$$\det(\lambda I - (A - BKC)) = 0$$

$$\lambda = \{-9.4423, -2.7789 \pm j32.2332\} \quad (4.6)$$

dapat dilihat pada (4.2), (4.4), dan (4.6) bahwa penempatan pole dinamika rotasi (4.6) lebih jauh dari penempatan pole dinamika translasi (4.2) dan (4.4). Oleh karena itu sistem menjamin bahwa kecepatan *inner loop* lebih cepat daripada *outer loop*. Gambar 4.1 menunjukkan berbanding waktu yang dibutuhkan respon posisi dan *attitude* untuk mencapai posisi *steady state* dengan referensi sinyal step.

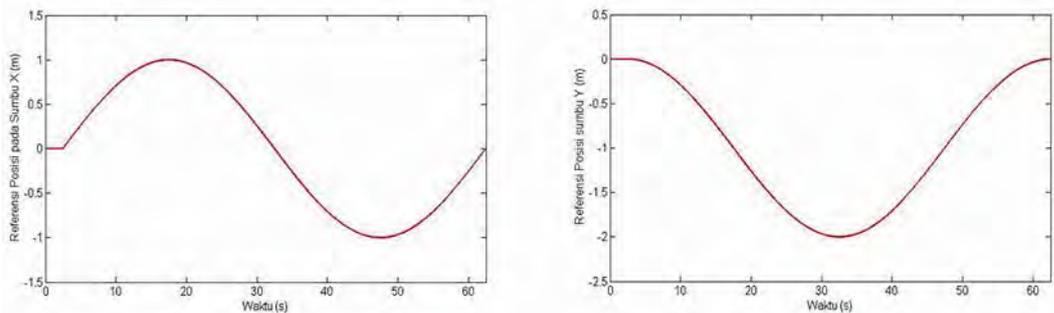


Gambar 4.1 (a) Perbandingan respon θ dengan posisi X , (b) Perbandingan respon ϕ dengan posisi Y

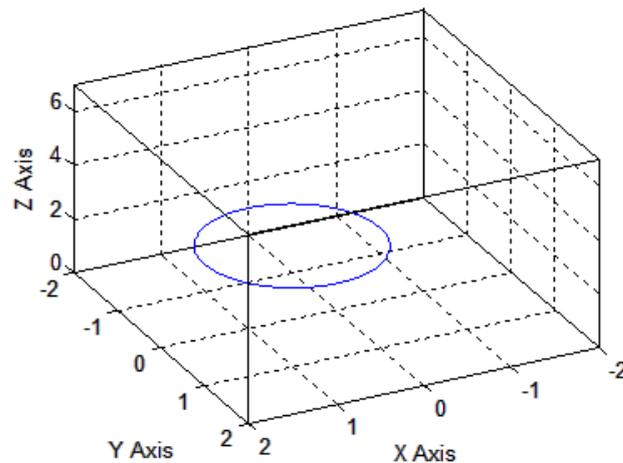
4.2 Pengujian Respon Sistem Untuk Gerak Translasi

Pengujian pertama yang dilakukan terhadap kontroler posisi yang telah didesain adalah dengan memberikan sinyal referensi lintasan yang harus diikuti oleh quadrotor. Gambar 4.2 menunjukkan sinyal referensi (x_d, y_d) yang diberikan kepada quadrotor dalam sumbu X dan Y , berturut-turut. Referensi pada sumbu Z hanya diberikan sinyal step dengan nilai $z_d = 1$ m. Jika dilihat dalam bidang X, Y, Z ilustrasi referensi lintasan dapat dilihat pada Gambar 4.3.

Dalam pengujian kontroler quadrotor, sinyal referensi diberikan pada saat quadrotor sudah berada pada kondisi melayang (*hover*). Seperti yang dapat dilihat pada Gambar 4.2, sinyal referensi lintasan diberikan pada detik ke-2.5, dimana pada saat waktu tersebut quadrotor telah berada dalam kondisi melayang. Pengujian pertama dilakukan dengan memilih nilai parameter γ yang terbaik, yaitu $\gamma = 0.7$.



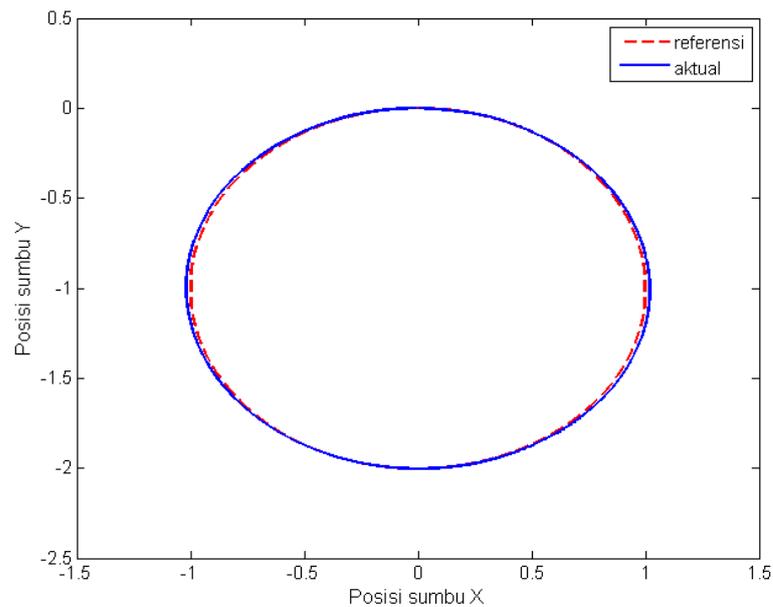
Gambar 4.2 Sinyal Referensi Posisi pada Sumbu X dan Y



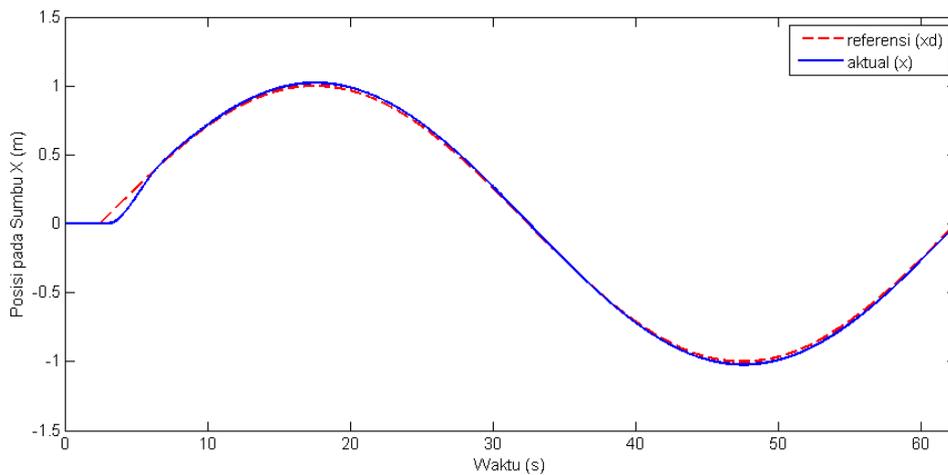
Gambar 4.3 Referensi Lintasan dalam bidang X, Y, Z

Hasil nilai gain *output feedback* (K) didapat melalui komputasi menggunakan *software* Matlab. Respon *tracking* sistem terdapat pada Gambar 4.4, menunjukkan respon jika dilihat pada bidang X dan Y dimana lintasan berbentuk lingkaran. Respon dapat mengikuti referensi yang diberikan dengan rata-rata penyimpangan ± 0.09 m.

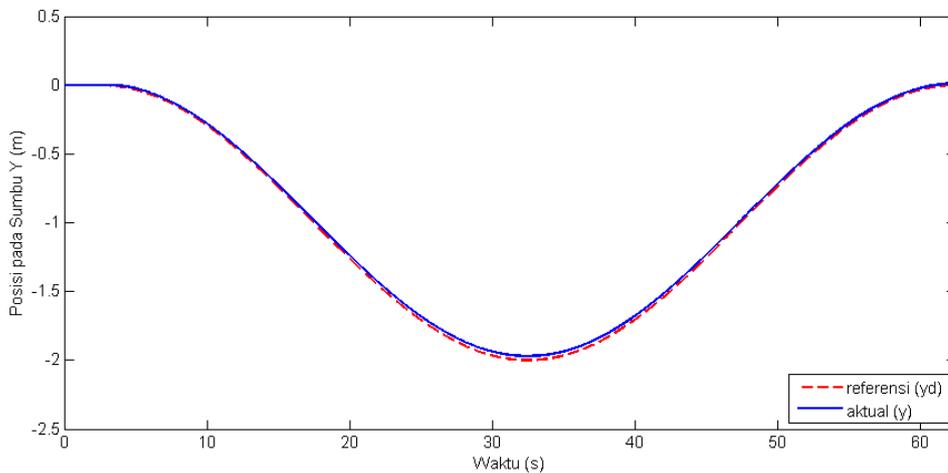
Pada Gambar 4.5 dan Gambar 4.6, adalah respon *tracking* dari quadrotor jika dideskripsikan pada sumbu X dan Y secara terpisah. Gambar 4.5 menunjukkan bahwa posisi quadrotor pada sumbu X dapat mengikuti referensi yang berbentuk sinyal sinusoidal. *Tracking* dimulai pada detik ke-2.5 (pada awal *tracking*), terjadi keterlambatan respon sekitar 0.5 detik yang menyebabkan penyimpangan respon sebesar 0.1 m. Pada detik ke-5.5 respon dapat kembali berada pada lintasan referensi. Nilai *Integral Square Error* (ISE) *tracking* pada posisi X sebesar 0.06446.



Gambar 4.4 Respon *Tracking* Posisi pada sumbu X dan Y

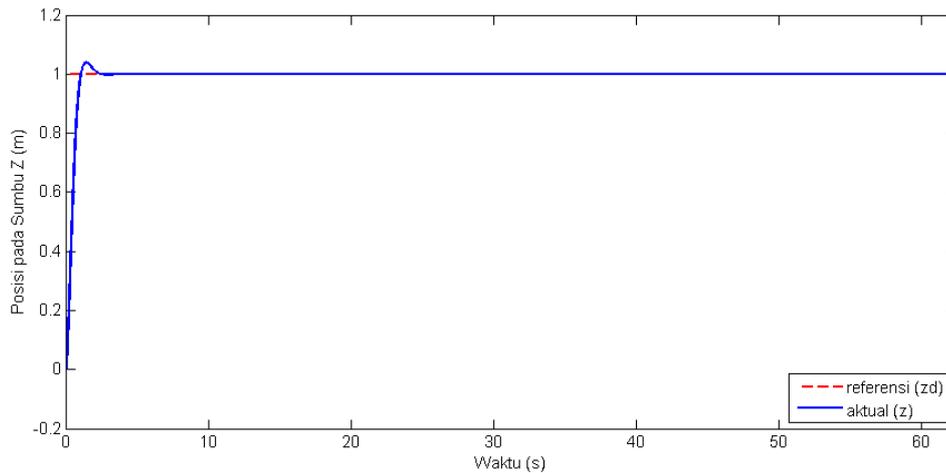


Gambar 4.5 Respon *Tracking* Posisi pada sumbu *X*

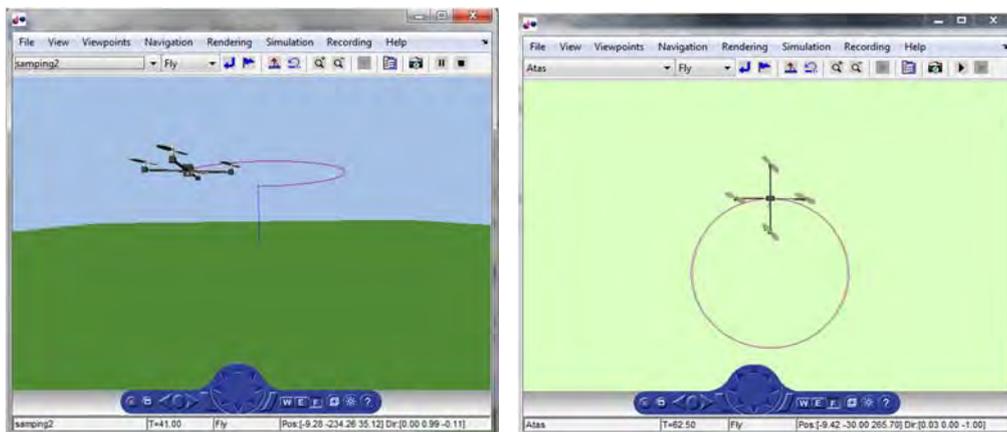


Gambar 4.6 Respon *Tracking* Posisi pada Sumbu *Y*

Respon posisi translasi pada sumbu *Y* ditunjukkan oleh Gambar 4.6. Respon dapat mengikuti referensi lintasan yang diberikan dengan tepat. Pergerakan pada sumbu *Y* memiliki nilai ISE sebesar 0.0415. Posisi pada sumbu *Z* ditunjukkan oleh Gambar 4.7. Sinyal referensi yang diberikan adalah sinyal step. Hasilnya, respon dapat mengikuti referensi dengan *overshoot* sekitar 0.2 m kemudian respon mencapai *steady state* pada detik ke-2.3. Secara keseluruhan, kontroler *tracking* hasil rancangan dapat menghasilkan kesesuaian *tracking*. Respon dapat mengikuti lintasan yang diinginkan dengan nilai ISE *tracking* bernilai kurang dari 0.1.



Gambar 4.7 Respon *Tracking* Posisi pada Sumbu Z

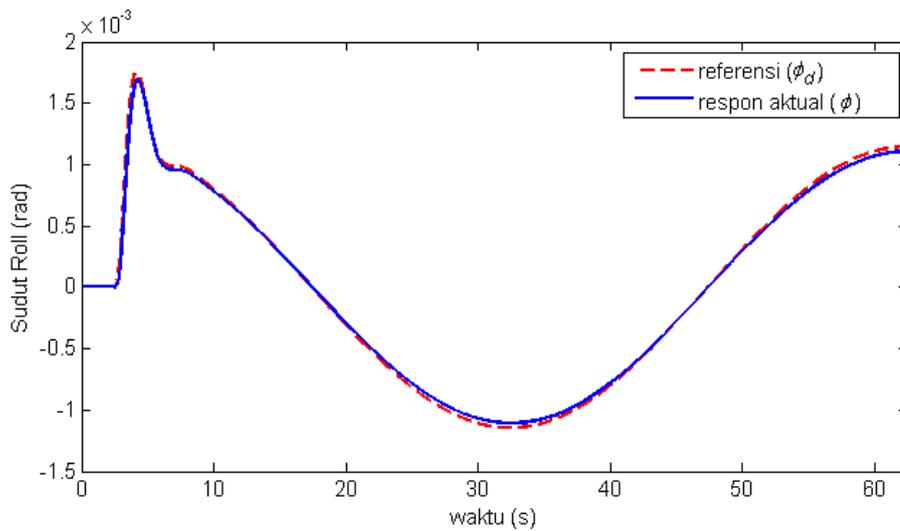


Gambar 4.8 *Screenshoot* dari *virtual world* quadrotor

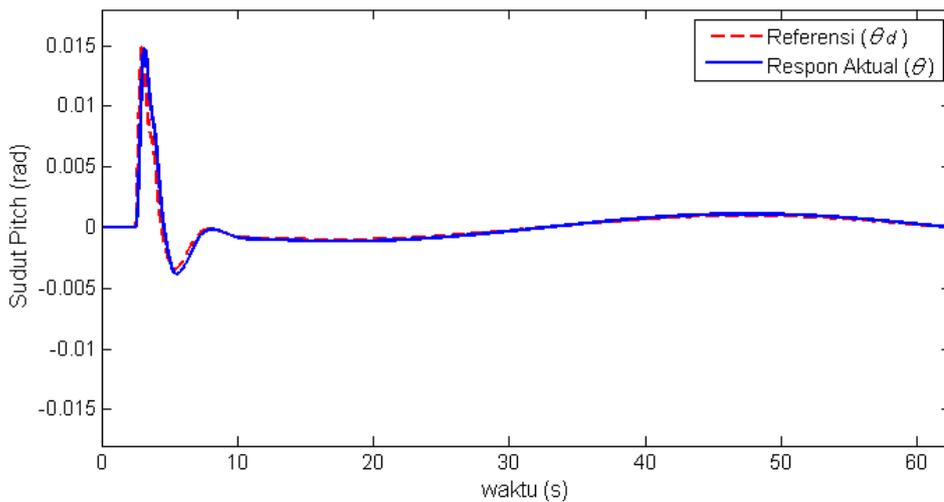
Untuk memperjelas hasil simulasi, pada dibuat pula sebuah *virtual world* yang dapat merepresentasikan dinamika gerak rotasi dan translasi dari quadrotor yang disajikan dalam bentuk tiga dimensi. Animasi 3D ini dibangun menggunakan *3D World Editor* yang terdapat pada *software* Matlab. Gambar 4.8 merupakan *screenshoot* dari *virtual world* quadrotor ketika simulasi sedang berlangsung, garis berwarna merah merupakan jejak lintasan yang dilalui oleh quadrotor. Dalam *virtual world* ini terdapat dua buah sudut pandang (*view point*) yaitu tampak atas dan tampak samping.

4.3 Pengujian Respon Sistem Untuk Gerak Rotasi

Seperti yang telah dibahas pada sub-bab 3.6, input referensi untuk gerak rotasi diperoleh dari output kontroler translasi yang dikonversi menjadi ϕ_d dan θ_d . Gambar 4.9 dan 4.10 menunjukkan sinyal referensi dan respon sistem untuk gerak rotasi. Dapat dilihat bahwa respon sudut *roll* dan *pitch* dapat mengikuti sinyal referensi yang diberikan dari output kontroler posisi (*outer loop*), dengan rata-rata error sebesar 0.001594 rad.



Gambar 4.9 Respon Sudut *Roll* tanpa Gangguan



Gambar 4.10 Respon Sudut *Pitch* tanpa Gangguan

4.4 Pemodelan Efek Embusan Angin (*Wind gust*)

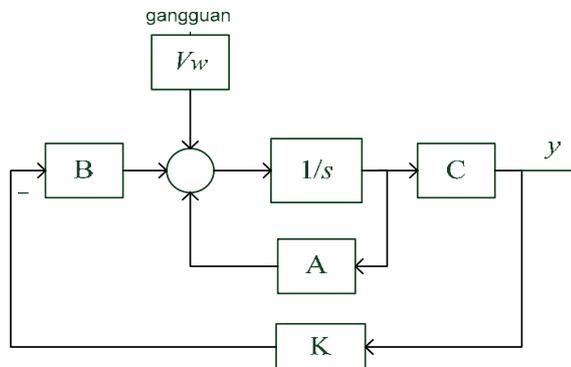
Pengaruh kondisi cuaca masuk kedalam salah satu permasalahan kontrol penerbangan quadrotor. Kondisi cuaca yang utama terdiri dari temperatur, tekanan, kerapatan (*density*) udara, dan pergerakan udara (angin). Angin memiliki pengaruh yang cukup besar terhadap dinamika gerak quadrotor, sedangkan perubahan temperatur, tekanan dan kerapatan udara tidak terlalu berpengaruh. Angin dispesifikasikan oleh kecepatan angin dan arah angin dalam koordinat bumi. Kecepatan dan arah dapat berupa konstan maupun variabel terhadap waktu.

Diasumsikan quadrotor hanya berskala laboratorium sehingga quadrotor hanya terbang didalam ruangan, model kecepatan angin yang diberikan bernilai kecil. Dalam pemodelan *wind gust* ini karakteristik yang dipertimbangkan, yaitu magnitude dari kecepatan angin. Magnitude dari kecepatan angin diformulasikan dalam persamaan berikut [16]:

$$|V_w| = \begin{cases} 0, & t < t_0, \\ \frac{V_m}{2} \left(1 - \cos\left(\frac{\pi(t-t_0)}{d_n-t_0}\right) \right), & t_0 < t < d_n, \\ V_m, & t < t_m \end{cases} \quad (4.7)$$

dengan V_m adalah *gust* magnitude, d_n adalah panjang interval dari *gust*, t_0 adalah waktu awal, dan t_m adalah waktu terbang maksimum.

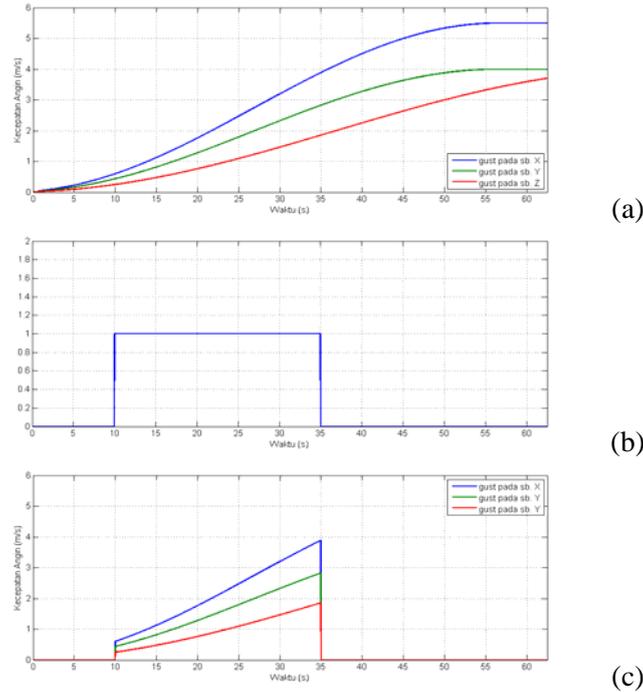
Gangguan diberikan kedalam dinamika sistem quadrotor, jika dideskripsikan seperti pada Gambar 4.11. Pada gangguan ini, parameter yang digunakan adalah $t_0 = 0$ s, *gust* magnitude untuk u , v , dan w berturut-turut adalah $V_m = [5.5; 4.0; 4.0]$ m/s dan panjang *gust* adalah $d_n = [60; 60; 80]$ m.



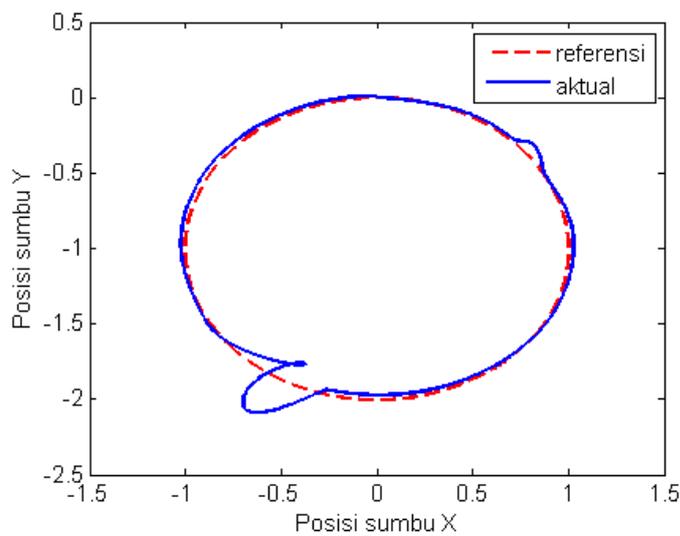
Gambar 4.11 Diagram blok untuk simulasi dengan gangguan

4.5 Pengujian Respon Sistem Terhadap Efek *Wind gust*

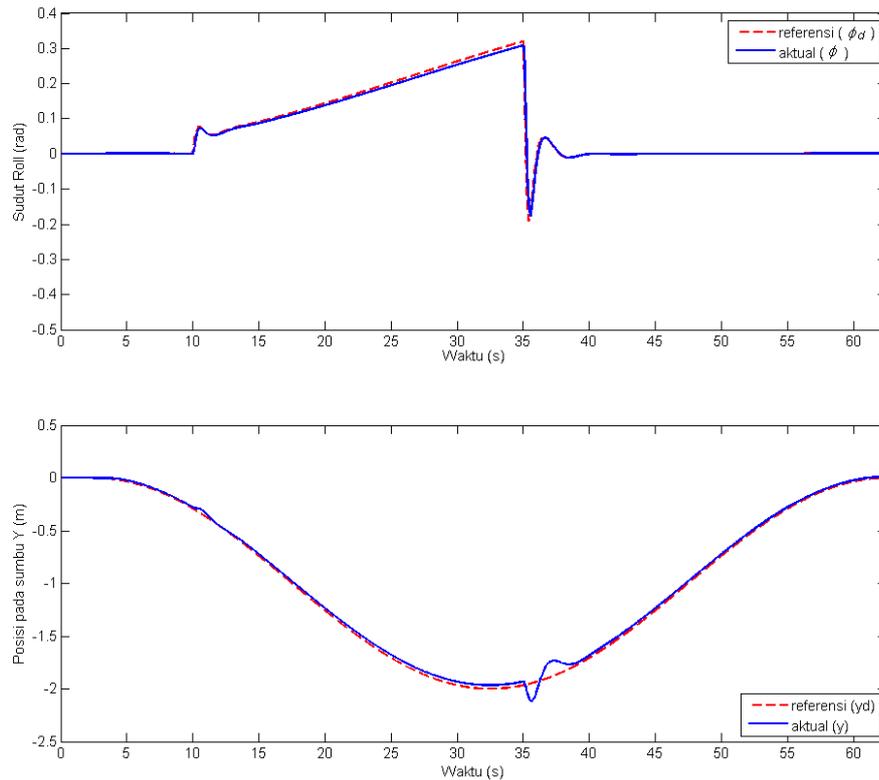
Simulasi selanjutnya adalah simulasi dengan efek *wind gust* yang diberikan pada dinamika plant. Kurva model *wind gust* dari (4.7), dapat dilihat pada Gambar 4.12 (a). Diasumsikan quadrotor berada dalam ruangan, kemudian gangguan diberikan pada detik ke-10 hingga detik ke-35 (Gambar 4,12(b)), sehingga gangguan yang masuk ke dalam sistem adalah pada Gambar 4.12 (c).



Gambar 4.12 (a) Kurva *wind gust* secara keseluruhan, (c) *wind gust* pada sistem



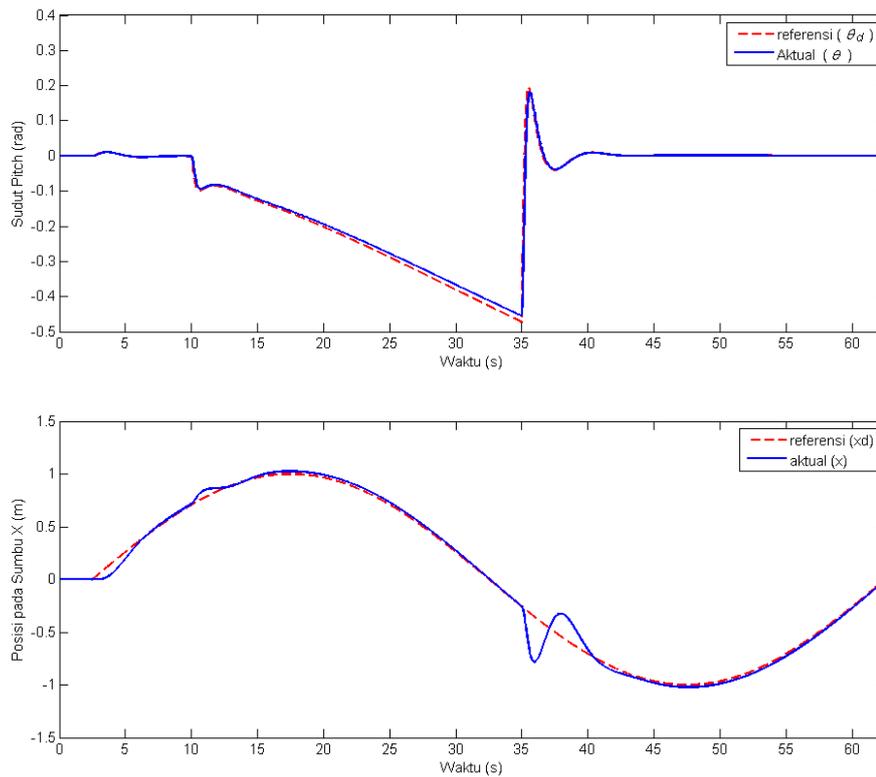
Gambar 4.13 Respon Tracking dengan Gangguan dalam sumbu X, Y



Gambar 4.14 Respon sudut ϕ dan respon posisi Y saat terdapat *wind gust*

Gambar 4.13 menunjukkan respon *tracking* quadrotor pada sumbu X dan Y. Keberadaan *wind gust* membuat quadrotor menyimpang dari lintasan referensi, pada saat yang bersamaan gerakan *roll* dan *pitch* dari quadrotor menyesuaikan agar pengaruh *wind gust* membuat quadrotor dapat kembali *tracking* sesuai dengan referensi yang diberikan.

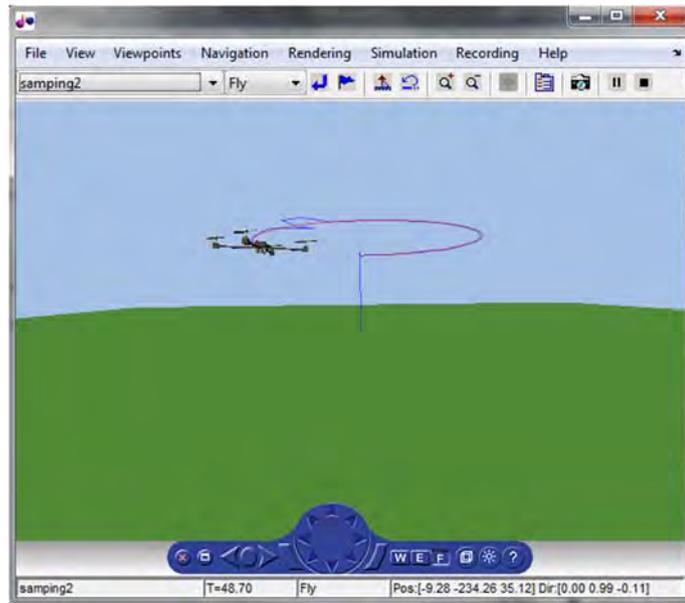
Seperti yang ditunjukkan pada Gambar 4.14, pengaruh *wind gust* terhadap sudut ϕ terjadi dari detik ke-10 s/d detik ke-40, pergerakan dari -0.2 rad s/d 0.3 rad atau sekitar -11° s/d 18° . Pada respon posisi pada sumbu Y, penyimpangan yang kecil terjadi pada detik ke-10, sedangkan pada detik ke-35 s/d detik ke-40 penyimpangan terjadi dengan nilai *overshoot* = 0.08 m dan *undershoot* = 0.1 m. Pergerakan rotasi mengkompensasi gangguan agar pada gerakan translasi tidak terjadi penyimpangan yang besar.



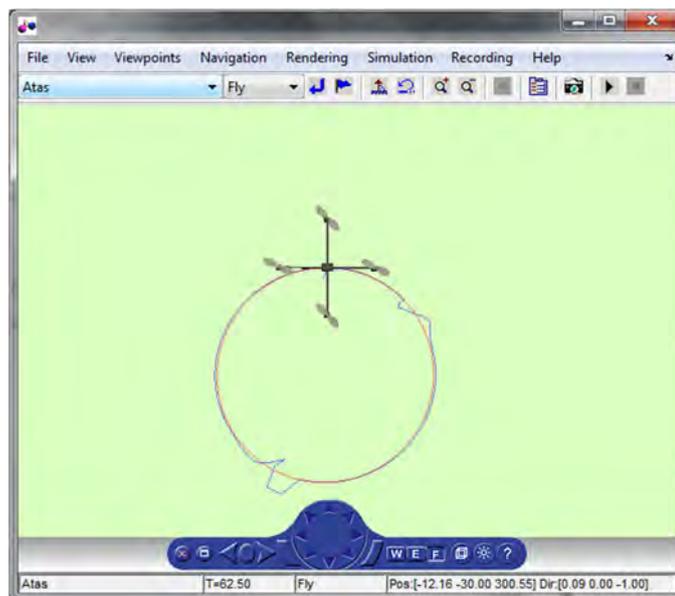
Gambar 4.15 Respon sudut θ dan posisi X saat terdapat *wind gust*

Gambar 4.15 menunjukkan respon sudut θ dan posisi pada sumbu X pada saat terdapat gangguan berupa *wind gust*. Ketika angin berhembus membuat quadrotor menyimpang dari posisi *tracking*-nya. Penyimpangan posisi pada sumbu X terjadi pada detik ke-10 dengan penyimpangan sebesar 0.05 m dan detik ke-35 dengan penyimpangan sebesar ± 0.175 m. Respon sudut θ dapat mengikuti referensinya, namun pada saat terdapat *wind gust*, respon sudut θ berosilasi dari -0.5 rad s/d 0.2 rad atau sekitar -30° s/d 11° . Hal tersebut terjadi untuk membuat quadrotor tetap stabil sehingga dapat kembali men-*track* sinyal referensi.

Visualisasi quadrotor pada saat tracking dengan gangguan dapat dilihat pada Gambar 4.16, yang mana quadrotor sedang terbang menuju titik akhir referensi. Gambar 4.17 menunjukkan quadrotor tampak atas ketika sudah selesai melakukan tracking, garis melingkar berwarna merah merupakan referensi dan garis berwarna biru merupakan sinyal aktual dari quadrotor.

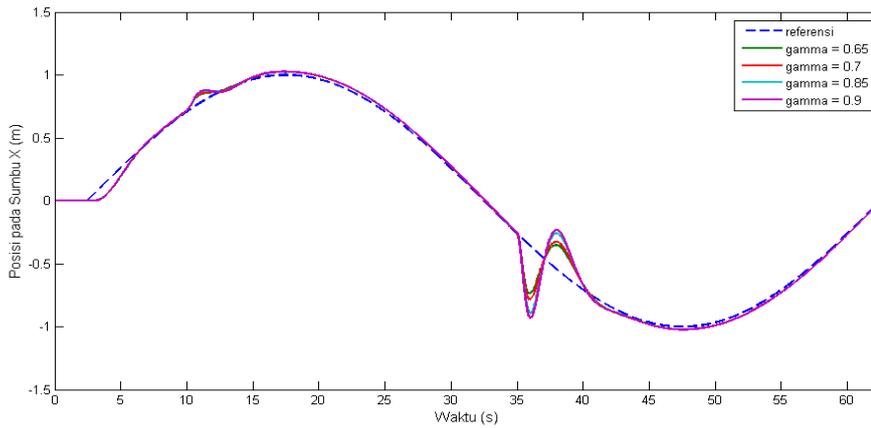


Gambar 4.16 *Screenshot virtual world* saat simulasi dengan efek gangguan

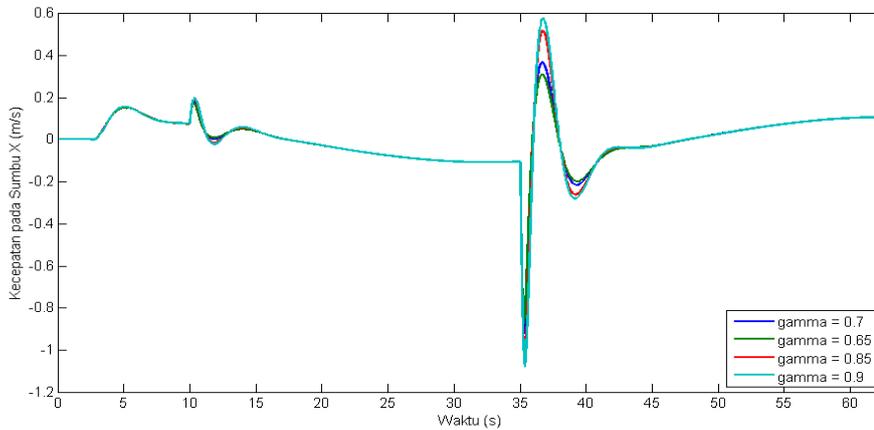


Gambar 4.17 *Virtual world* (tampak atas) saat diberi efek gangguan

Selanjutnya dilakukan simulasi dengan nilai γ yang berbeda-beda ($\gamma = 0.65, 0.7, 0.85, \text{ dan } 0.9$) untuk mengetahui sejauh mana pengaruh tingkat level pelemahan terhadap gangguan. Hasil simulasi disajikan dalam bentuk analisa posisi dan kecepatan quadrotor dengan berbagai nilai γ pada saat terdapat gangguan. Seperti yang terdapat pada Gambar 4.18 – 4.21.



Gambar 4.18 Respon posisi pada sumbu X dengan berbagai nilai γ



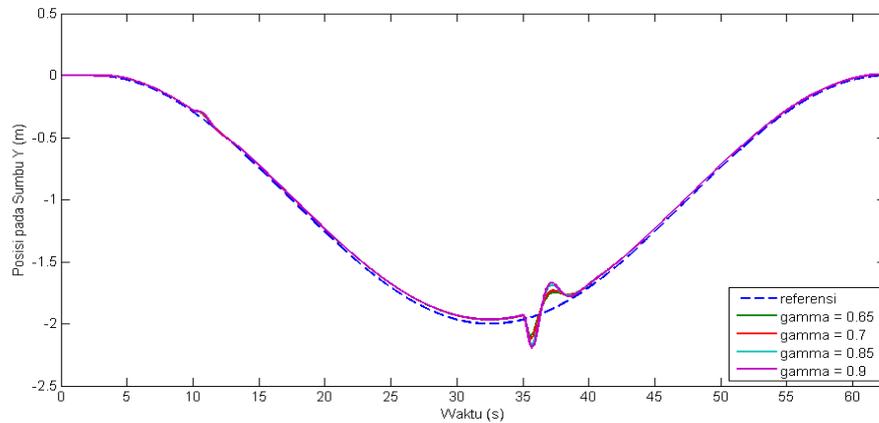
Gambar 4.19 Respon Kecepatan pada sumbu X dengan berbagai nilai γ

Respon posisi quadrotor pada sumbu X pada saat diberi gangguan *wind gust*, dapat dilihat pada Gambar 4.18. Nilai γ yang berbeda-beda berpengaruh terhadap seberapa besar maksimum *overshoot* pada setiap penyimpangan *tracking* yang terjadi. Penyimpangan pada detik ke-10 cukup kecil yaitu rata-rata untuk semua nilai γ adalah 0.05m. Pada saat $\gamma = 0.7$ maksimum *overshoot* adalah 0.175 m dan *undershoot* 0.28 m, sedangkan saat $\gamma = 0.65$ maksimum *overshoot* 0.17 m dan *undershoot* 0.285 m. Pada saat $\gamma = 0.85$ *overshoot* dan *undershoot* mencapai 0.195 m dan 0.38 m. Respon dengan penyimpangan terbesar terjadi ketika nilai $\gamma = 0.9$, dengan maksimum *overshoot* 0.2 m dan *undershoot* 0.45 m.

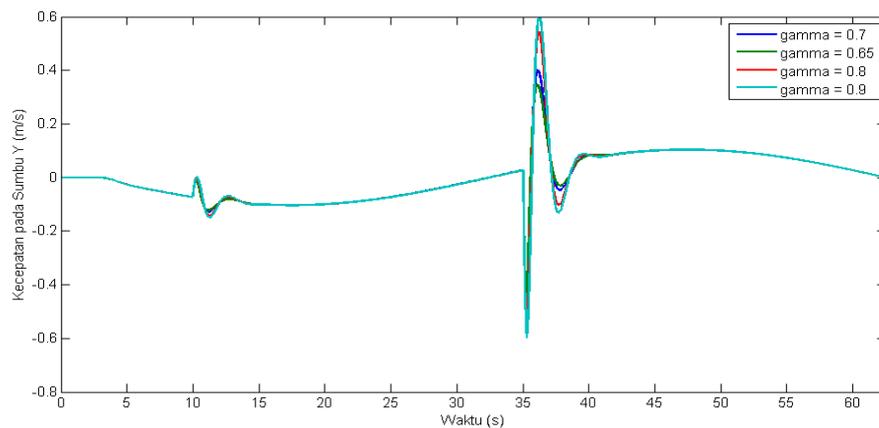
Gambar 4.19 menunjukkan kecepatan quadrotor pada posisi X. Terdapat dua kali peningkatan kecepatan quadrotor, yaitu pada detik ke-10 s/d 15 dan detik ke-

35 s/d 45. Pada saat $\gamma = 0.65$ kecepatan berkisar antara -0.86 m/s hingga 0.28 m/s. Pada $\gamma = 0.9$ maksimum *overshoot* dan *undershoot* adalah 0.6 m/s dan -1.1 m/s.

Gambar 4.20 adalah respon posisi pada sumbu Y dengan berbagai nilai γ . Penyimpangan pada detik ke-10 cukup kecil yaitu rata-rata 0.04 m. Untuk $\gamma = 0.65$, maksimum *overshoot* dan *undershoot* bernilai 0.09 m dan 0.13 m sedangkan $\gamma = 0.7$ *undershoot* = 0.12 dan *overshoot* = 0.1 m. Untuk $\gamma = 0.9$ maksimum *overshoot* = 0.2 m dan *undershoot* = 0.22 m. Kecepatan pada sumbu Y ditunjukkan oleh Gambar 4.21. Kecepatan maksimum untuk $\gamma = 0.9$ adalah ± 0.6 m/s, untuk $\gamma = 0.7$ adalah ± 0.3 m/s, dan untuk $\gamma = 0.65$ adalah ± 0.26 m/s



Gambar 4.20 Respon posisi pada sumbu Y dengan berbagai nilai γ



Gambar 4.21 Respon kecepatan pada sumbu Y dengan berbagai nilai γ

Nilai γ yang berbeda-beda, akan mempengaruhi besar nilai matriks P dan gain K . Oleh karena itu, respon yang dihasilkan akan berbeda untuk setiap nilai γ yang diberikan. Berdasarkan seluruh respon sistem saat simulasi dengan efek gangguan, menunjukkan bahwa perbedaan nilai γ akan mempengaruhi penyimpangan posisi quadrotor. Besarnya penyimpangan posisi yang terjadi pada quadrotor pada nilai γ yang berbeda-beda dapat dilihat pada Tabel. 4.1.

Penyimpangan terkecil terjadi ketika $\gamma = 0.7$, jika dilihat pada Tabel 4.2, bahwa nilai γ^* mendekati γ yaitu 0.67. Semakin nilai γ^* mendekati γ ($\gamma^* = \gamma$), maka pada kondisi ini menghasilkan ketahanan sistem terhadap gangguan dengan performa yang baik.

Tabel 4.1 Penyimpangan posisi quadrotor pada sumbu X dan Y saat ada *wind gust*

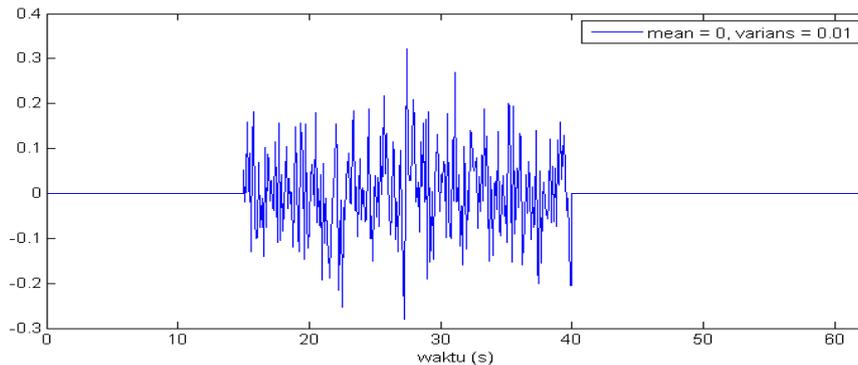
| γ | Penyimpangan | Penyimpangan | ISE | ISE |
|-------------|---------------|--------------|------------|------------|
| | Posisi X | Posisi Y | Posisi X | Posisi Y |
| 0.65 | ± 0.285 m | ± 0.12 m | 0.1928 | 0.05852 |
| 0.7 | ± 0.28 m | ± 0.11 m | 0.2000 | 0.05518 |
| 0.85 | ± 0.38 m | ± 0.2 m | 0.2777 | 0.06838 |
| 0.9 | ± 0.45 m | ± 0.22 m | 0.3779 | 0.08821 |

Tabel 4.2 Nilai gain K pada *inner loop* dan norm- ∞ sistem untuk berbagai nilai γ

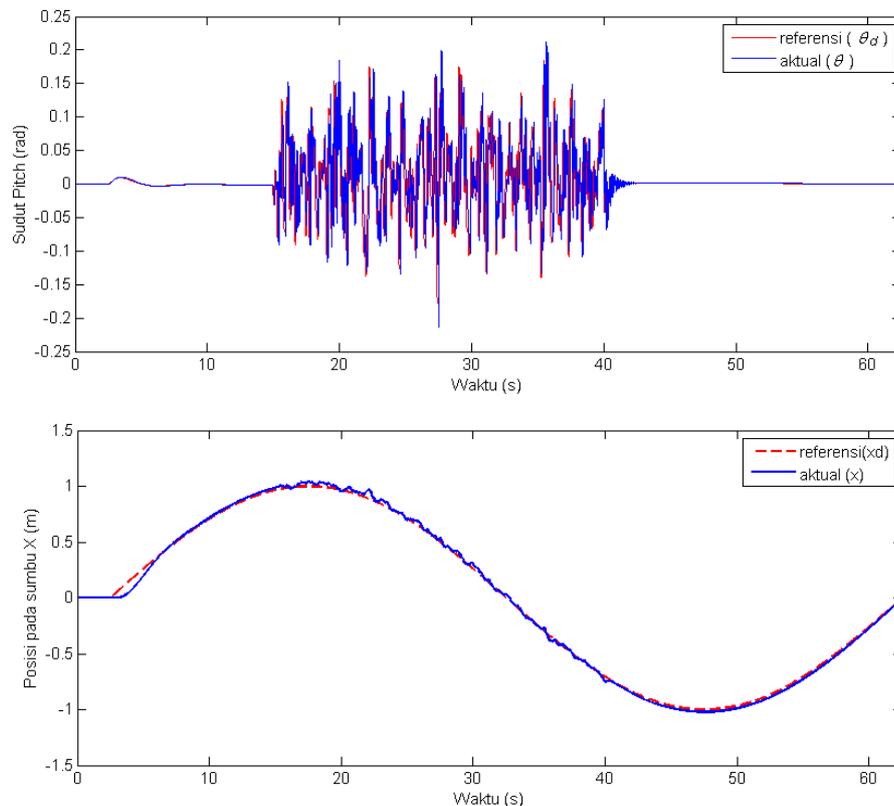
| γ | K | $\ T_{zw}\ _{\infty} = \gamma^*$ |
|-------------|-----------------|----------------------------------|
| 0.65 | [0.6570 0.0752] | 0.69 |
| 0.7 | [0.7346 0.0817] | 0.67 |
| 0.85 | [0.9040 0.1000] | 0.545 |
| 0.9 | [0.9609 0.1061] | 0.5081 |

4.6 Pengujian Respon Sistem Terhadap Noise Pengukuran

Noise pengukuran diberikan pada state pengukuran dari sistem untuk menguji kemampuan sistem dalam mengatasi noise pengukuran. Noise yang diberikan berupa sinyal random Gaussian dengan mean bernilai 0 sedangkan varian bernilai 0.01 atau $N(0, 0.01)$ yang diberikan pada detik ke-15 hingga 40. Sinyal noise random ditunjukkan pada Gambar 4.22.



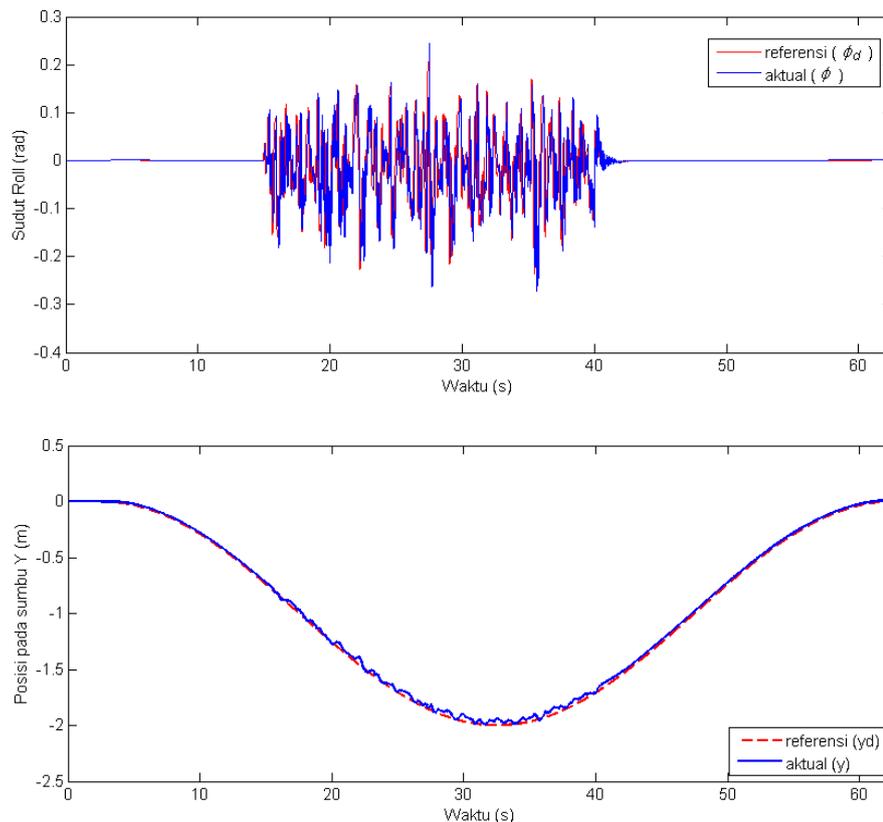
Gambar 4.22 Sinyal random Gaussian dengan $N(0, 0.01)$



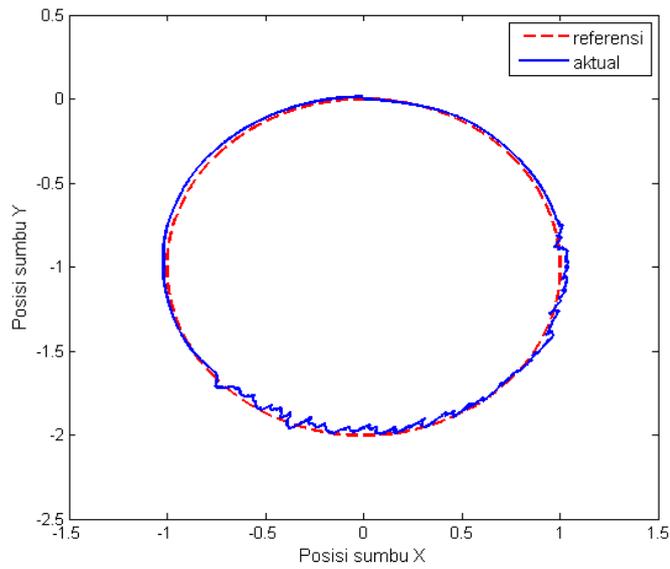
Gambar 4.23 Sudut *Pitch* dan posisi *X* saat diberi *noise* $N(0, 0.01)$

Gambar 4.23 menunjukkan perilaku sudut *pitch* dan posisi pada sumbu X pada saat diberi *noise*. Untuk mengatasi sinyal random yang diberikan, sudut *pitch* beresilasi dengan nilai *overshoot/undershoot* sebesar ± 0.2 rad atau 12° . Pengaruh *noise* terhadap *tracking* posisi X, terjadi ada detik ke-15 hingga detik ke-40 respon sistem tidak *smooth* melainkan terjadi riak kecil selama rentang waktu tersebut. Berdasarkan nilai ISE, rata-rata penyimpangan yang terjadi sebesar 0.04181 m.

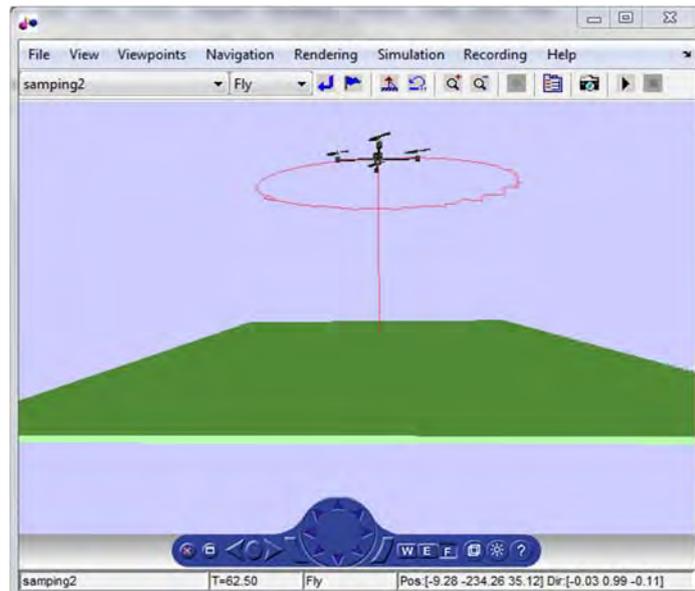
Gambar 4.24 menunjukkan sudut *roll* dan posisi Y saat diberi *noise*. Sama halnya dengan *tracking* pada posisi X, pergerakan pada sumbu Y mengalami penyimpangan selama 25 detik. Nilai ISE pada pergerakan ini adalah 0.03552 m. Respon sudut *roll* beresilasi dengan nilai *overshoot/undershoot* sebesar ± 0.25 rad atau 15° . Pada saat *noise* sudah tidak diberikan pada sistem, quadrotor dapat kembali melakukan *tracking* dengan stabil seperti yang dapat dilihat pada detik ke-43 hingga detik ke-62.5.



Gambar 4.24 Sudut *Roll* dan posisi Y saat diberi *noise* $N(0, 0.01)$



Gambar 4.25 Respon *tracking* lintasan dalam bidang XY saat terdapat noise



Gambar 4.26 *Virtual world* dari quadrotor saat simulasi diberi efek noise

Respon *tracking* pada saat diberi noise jika dilihat dalam bidang X,Y ditunjukkan oleh Gambar 4.25 sedangkan *virtual world* dari quadrotor terdapat pada Gambar 4.26. Saat diberi noise, pergerakan quadrotor menjadi tidak *smooth* dimana untuk mengatasi noise, pergerakan rotasi quadrotor menjadi berosilasi, namun hal tersebut membuat penyimpangan pada posisi quadrotor menjadi kecil.

Tabel 4.3 Nilai *Integral Square Error* (ISE) untuk Berbagai Nilai Noise

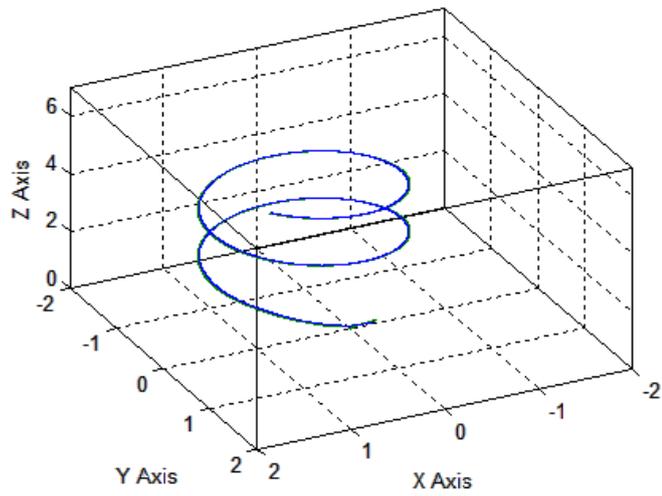
| ISE | Noise | | |
|--------------|-----------------------|----------------------|----------------------|
| | Mean = 0, Var = 0.001 | Mean = 0, Var = 0.01 | Mean = 0, Var = 0.05 |
| <i>Roll</i> | 0.01492 rad = 0.85° | 0.1498 rad = 8.58° | 0.7818 rad = 44,7° |
| <i>Pitch</i> | 0.01027 rad = 0.58° | 0.1031 rad = 6° | 0.5384 rad = 30.84° |
| Posisi X | 0.03496 m | 0.04181 m | 0.1071 m |
| Posisi Y | 0.02871 m | 0.03552 m | 0.1037 m |

Hasil simulasi dengan sinyal noise yang berbeda-beda telah disimpulkan pada Tabel 4.3. Sistem kontrol diuji dengan noise dengan nilai yang berbeda-beda, yaitu $N(0, 0.001)$, $N(0, 0.01)$ dan $N(0, 0.05)$. Berdasarkan nilai ISE dari setiap pergerakan quadrotor, diperoleh bahwa semakin besar nilai rata-rata amplitude dari noise yang diberikan, maka semakin besar pula nilai ISE dari pergerakan quadrotor. Hal ini berpengaruh pula kepada besarnya nilai sinyal kontrol yang masuk ke plant quadrotor. Maksimal noise pengukuran yang dapat diatasi oleh sistem kontrol adalah sebesar $N(0, 0.05)$ dengan standar deviasi 0.22.

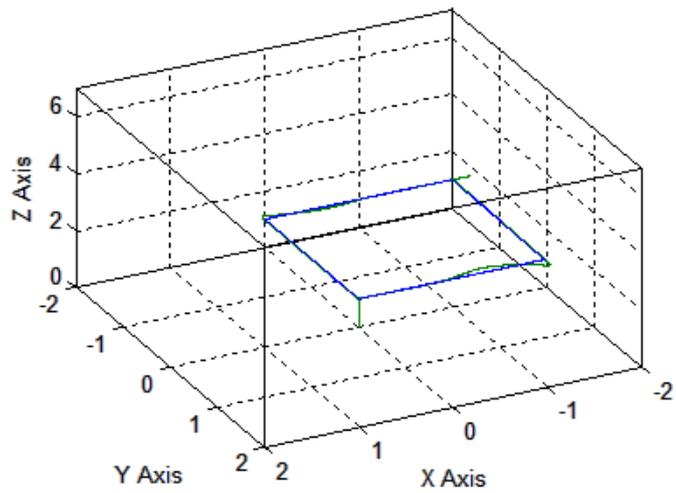
4.7 Simulasi dengan Sinyal Referensi yang Berbeda-beda

Pengujian dilakukan dengan memberikan sinyal referensi yang berbeda-beda terhadap sistem kontrol yang telah didesain tanpa mengubah parameter kontrol apapun. Seperti pada Gambar 4.27, referensi yang diberikan adalah sinyal berbentuk spiral (*helix*) dengan $x_d = \sin(0.03\pi t)$, $y_d = -1 + \cos(0.03\pi t)$, $z_d = t$. Hasilnya, sistem quadrotor dapat melakukan *tracking* mengikuti referensi yang diberikan.

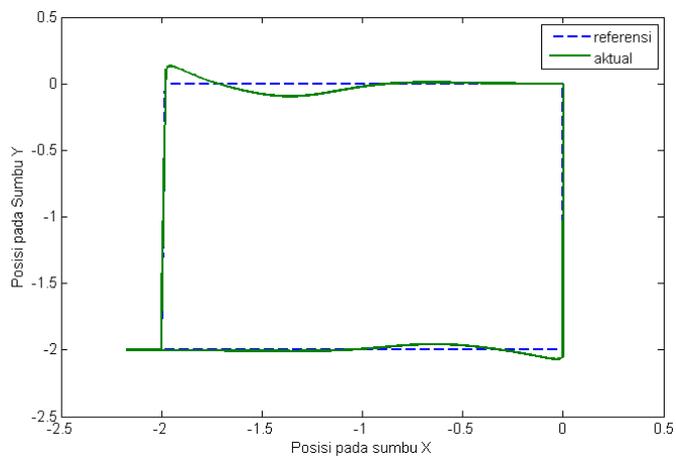
Selain itu, sinyal referensi dengan bentuk persegi juga diberikan kepada sistem, seperti yang ditunjukkan oleh Gambar 4.28. Hasilnya, sistem dapat melakukan *tracking* namun terdapat penyimpangan posisi sebesar 0.18 m pada posisi X dan 0.16 m pada posisi Y. Hal ini dikarenakan sistem tidak dapat berbelok dengan ketajaman sudut sebesar 90°, dapat dilihat pada Gambar 4.29 bahwa penyimpangan hanya terjadi pada saat quadrotor harus berbelok mengikuti referensi yang diberikan.



Gambar 4.27 *Tracking* Posisi dengan Lintasan Berbentuk Spiral (*Helix*)



Gambar 4.28 *Tracking* Posisi dengan Lintasan Berbentuk Persegi



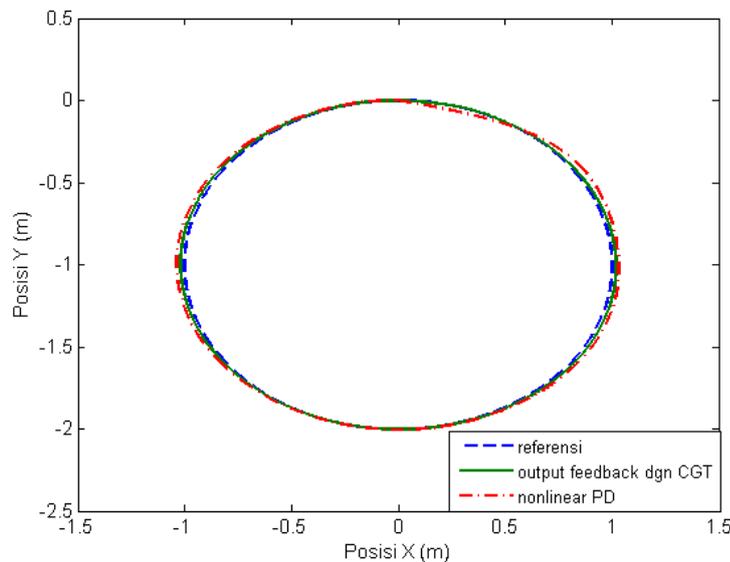
Gambar 4.29 Posisi X dan Y Pada Saat *Tracking* Lintasan Berbentuk Persegi

4.8 Perbandingan Kontroler Hasil Desain dengan Kontroler Nonlinear PD

Pada sub-bab ini, hasil simulasi dari kontroler *tracking* menggunakan struktur CGT yang didesain pada tesis ini akan dibandingkan dengan kontroler nonlinear PD [17]. Perbandingan hasil simulasi yang akan disajikan yaitu hasil *tracking* tanpa gangguan dan hasil *tracking* dengan adanya *wind gust*.

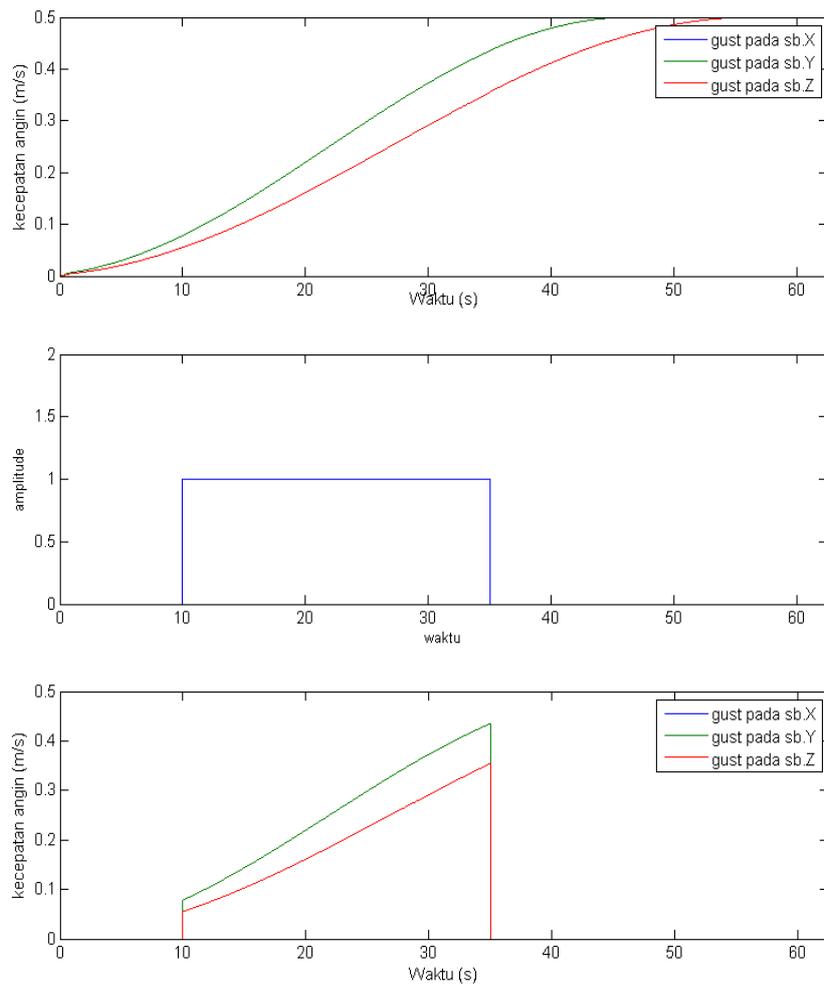
Sinyal referensi yang diberikan pada kedua kontroler adalah sama yaitu, lintasan berbentuk lingkaran. Referensi untuk posisi X dan Y diberikan setelah quadrotor berada pada posisi melayang, yaitu sekitar detik ke-2.5. Gambar 4.30 menunjukkan perbandingan respon *tracking* ketika tidak ada gangguan. Terlihat bahwa respon dengan kontroler nonlinear PD dapat melakukan *tracking* melingkar namun, terdapat sedikit pergeseran *tracking* dari referensinya di beberapa koordinat. Kontroler output feedback dengan CGT menghasilkan *tracking* lintasan yang tepat sesuai dengan referensi.

Selanjutnya disimulasikan untuk membandingkan ketahanan sistem terhadap gangguan *wind gust* yang diberikan. Untuk membandingkan kedua metode kontrol, perlakuan yang sama diterapkan baik pada dinamika plant maupun kontroler dari kedua metode kontrol.

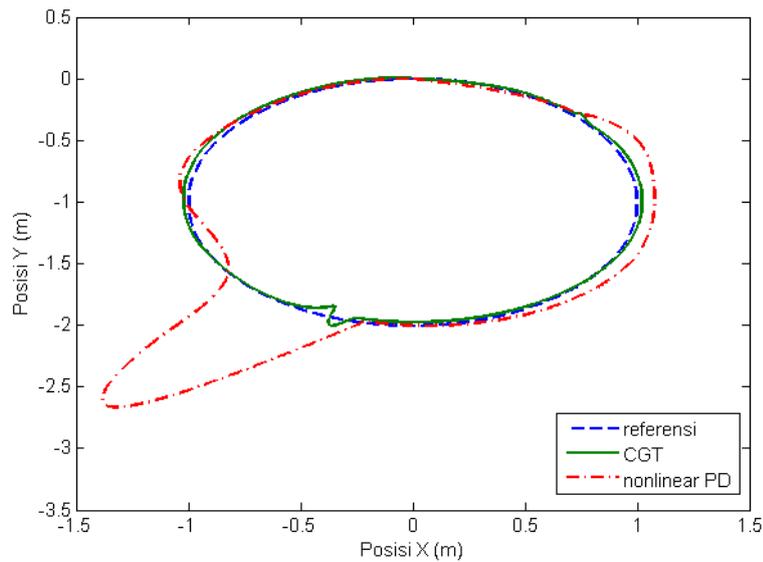


Gambar 4.30 Gangguan yang diberikan pada kedua metode kontrol

Pada [17], gangguan hanya dapat diberikan dengan maksimum amplitude dari *wind gust* sebesar 0.5 m/s, sedangkan pada desain kontroler yang diajukan dapat mencapai 4 m/s. Oleh karena itu, perbandingan dua buah kontroler ini dilakukan dengan besar amplitude dari gangguan seperti pada Gambar 4.31. Dimana *wind gust* hanya diberikan dalam rentang waktu detik ke-10 hingga ke-35 dengan amplitude 0.5 m/s.



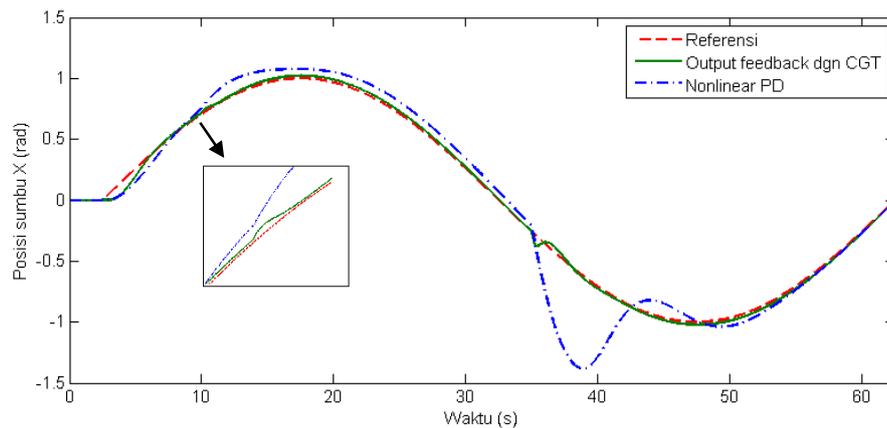
Gambar 4.31 Gangguan yang diberikan pada kedua metode kontrol



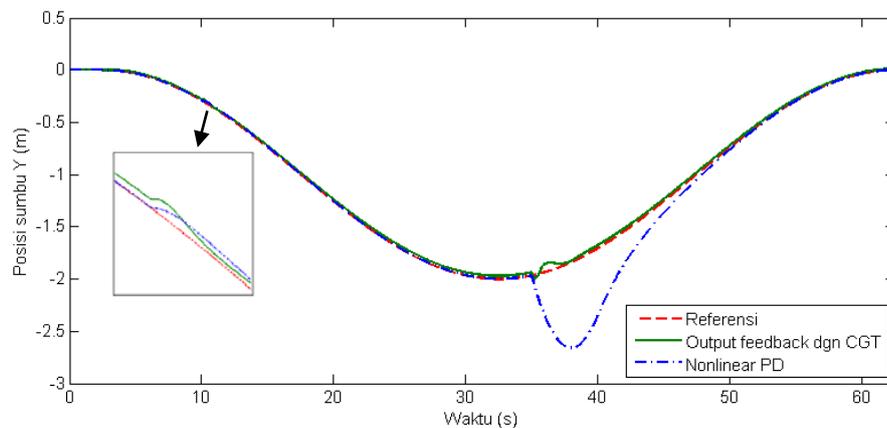
Gambar 4.32 Kurva respon *tracking* pada bidang X dan Y dengan gangguan

Gambar 4.32 menunjukkan respon *tracking* dengan nonlinear PD dan *output feedback* dengan CGT jika dilihat pada bidang X dan Y . Untuk respon dengan CGT menghasilkan *tracking* yang sesuai dengan referensi dengan penyimpangan sekitar 0.04 m pada saat diberi *wind gust*. Sistem kontrol dengan metode nonlinear PD tidak dapat mengatasi gangguan yang diberikan. Penyimpangan *tracking* yang terjadi sangat besar, yaitu maksimum penyimpangan mencapai 1.5 m. Untuk lebih jelas, hasil *tracking* disajikan dalam sumbu X dan Y secara terpisah seperti pada Gambar 4.33 dan 4.34

Gambar 4.33 menunjukkan perbandingan respon posisi pada sumbu X . Respon kontroler *output feedback* dengan CGT menyimpang dari referensi hanya pada detik ke-10 dan detik ke-35 dimana membutuhkan waktu kurang dari 5 detik untuk dapat kembali *tracking* sesuai referensinya. Penyimpangan terjadi sekitar 0.04 m. Penyimpangan respon untuk kontroler nonlinear PD sangat besar dan terjadi selama rentang detik ke-10 sampai 50. Maksimum penyimpangan posisi X mencapai 0.7 m.



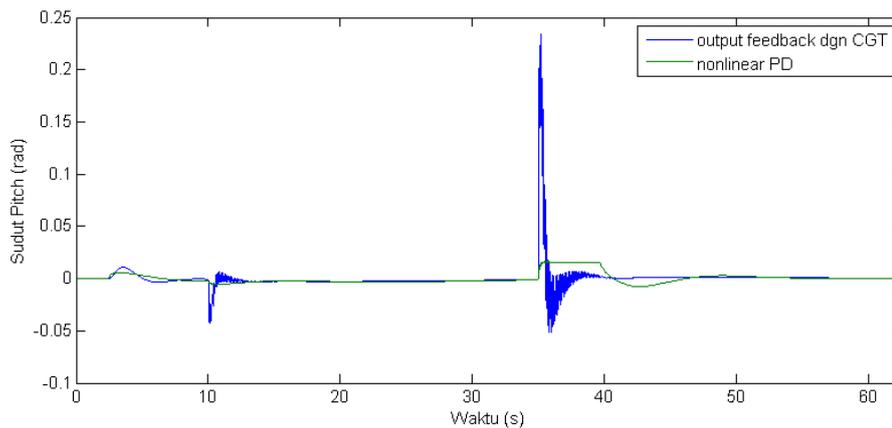
Gambar 4.33 Perbandingan respon posisi X dengan gangguan



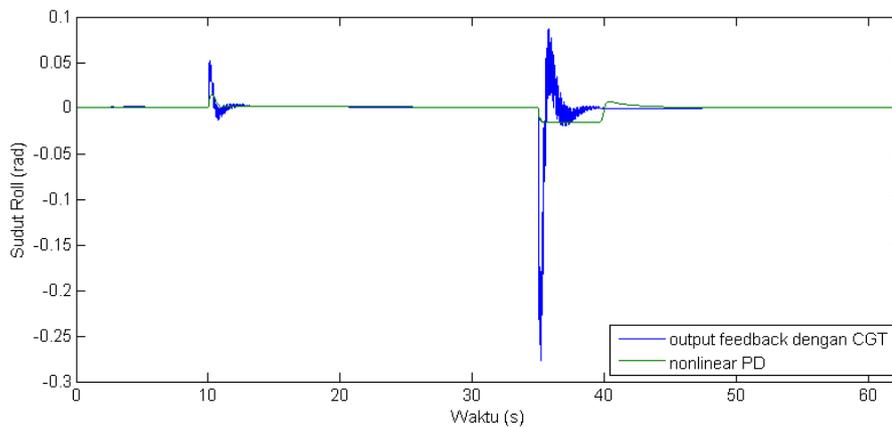
Gambar 4.34 Perbandingan respon posisi Y dengan gangguan

Gambar 4.34 menunjukkan respon posisi pada sumbu Y . Pengaruh gangguan yang diberikan pada detik ke-10 cukup kecil, yaitu membuat penyimpangan pada kedua kontroler sebesar ± 0.04 m. Namun, untuk gangguan pada detik ke-35 membuat respon dari kedua metode kontrol menyimpang. Untuk kontroler output feedback dengan CGT penyimpangan terjadi sebesar 0.09 m, sedangkan untuk kontroler nonlinear PD penyimpangan mencapai 0.7 m.

Kontroler output feedback dengan CGT dapat membawa quadrotor mengikuti referensi yang diberikan, dan menghasilkan error *tracking* yang kecil pada saat terdapat gangguan. Ketika terdapat gangguan, penyimpangan tetap ada, namun sistem kontrol yang diajukan pada tesis ini dapat dengan cepat membuat quadrotor kembali pada lintasan referensi.



Gambar 4.35 Perbandingan respon sudut *Pitch* dengan gangguan

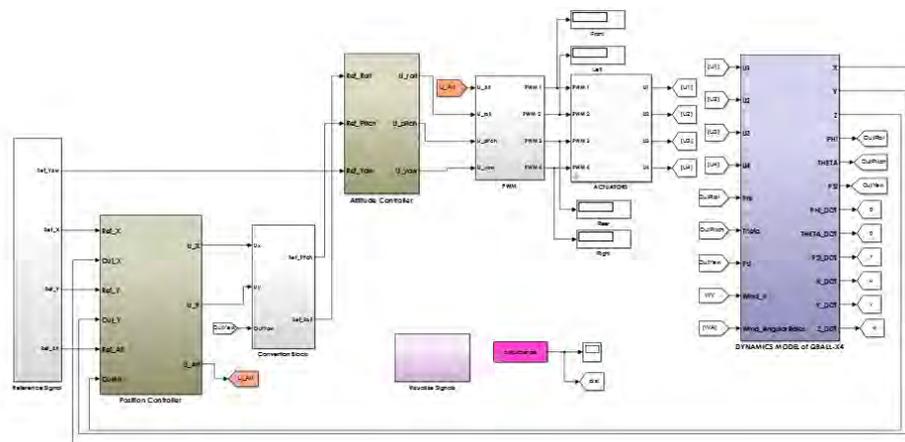


Gambar 4.36 Perbandingan respon sudut *Roll* dengan gangguan

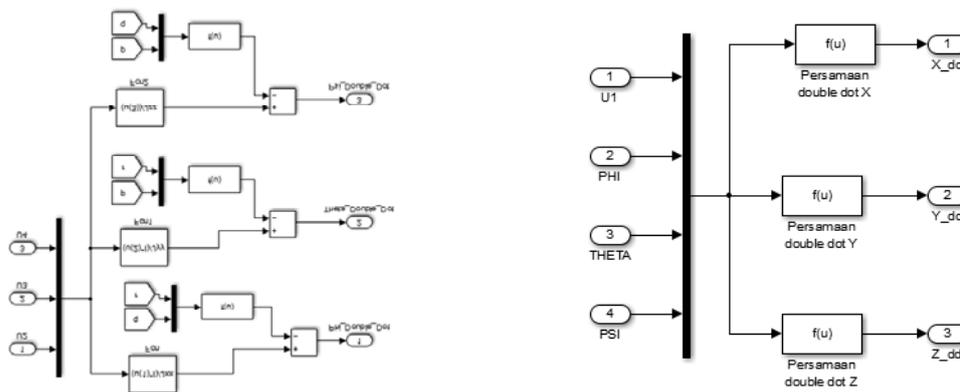
Gambar 4.35 dan Gambar 4.36 menunjukkan respon sudut *pitch* dan *roll* hasil perbandingan kedua metode kontrol. Respon sudut *roll* dan *pitch* dari kontroler output feedback dengan CGT terjadi *overshoot/undershoot* hingga ± 0.2 rad dan kembali pada posisi 0 rad dengan cepat yaitu kurang dari 5 detik. Perilaku sudut *roll* dan *pitch* dari kontroler nonlinear PD memiliki maksimum amplitud ± 0.02 rad dan terjadi dalam rentang waktu yang cukup lama yaitu sekitar 10 detik. Hal tersebut menyebabkan respon posisi quadrotor menyimpang dan cukup lama untuk dapat kembali pada lintasan referensinya.

Halaman ini sengaja dikosongkan

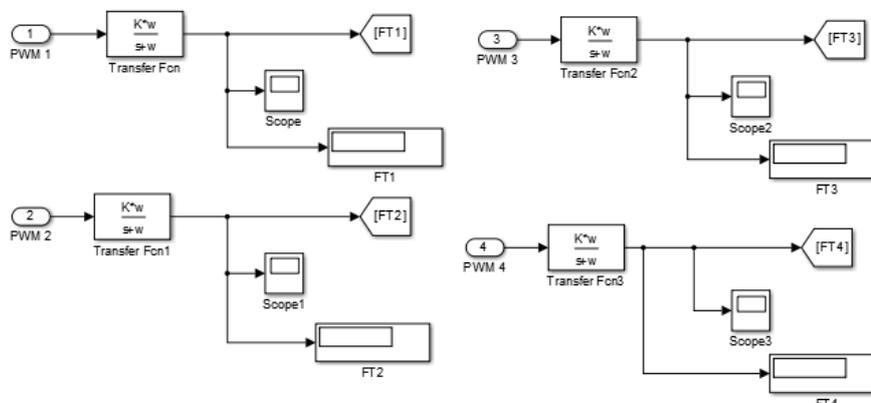
Desain Kontroler Secara Keseluruhan



Blok Model Dinamika Rotasi dan Translasi



Blok Model Aktuator



B.2 Pemograman

1. Program *Iterative* LMI dengan performa H_∞ untuk masalah stabilisasi

```
function K=Hitung_OPFB_Baru2(Am,Bm,Cm,Bwm,Qm,Rm,N,n,gam,toler,H)
%inisialisasi %program mencari H-inf OPFB tanpa inisial Gain
L0 = [0.1 0.1];
Ln = [0 0];
K = 0;
Kn = 0;
a=0;
toler = 0.01; %nilai epsilon(toleransi)
gam = 0.7;
Rr = inv(Rm);

while abs(Ln(1)-L0(1))>=toler || abs(Ln(2)-L0(2))>=toler
    a=a+1;;
    L0 = Ln;
    K=Kn;

    %Insialisasi LMI 1
    setlmis([]);
    Y=lmivar(1,[2 1]);

    lmiterm([1 1 1 Y],1,-Am,'s');
    lmiterm([1 1 1 0],-Qm);
    lmiterm([1 1 1 0],-L0'*Rr*L0);
    lmiterm([1 2 1 Y],Bm',1);
    lmiterm([1 2 2 0],Rm);
    lmiterm([1 3 1 Y],Bwm',1);
    lmiterm([1 3 3 0],-(gam^2));

    lmiterm([-2 1 1 Y],1,1); %nilai Y>0

    lmis=getlmis;
    [tm,xf]=feasp(lmis);

    %Hasil matriks P
    Y=dec2mat(lmis,xf,Y);
    P=Y;
    Kn = (inv(Rm))*(Bm'*P+L0)*Cm'*(inv(Cm*Cm')) %Gain Kn
    Ln = Rm*Kn*Cm-(Bm'*P); %Gain Ln
    L0;
end
K = Kn;
```

2. Program untuk Kontroler *Altitude* dan *Attitude*

```
% Menghitung Gain Sudut Roll Pitch Yaw
phi = 0.2; tetha = 0.2; psi = 0.2; %titik kerja
%parameter quadrotor
m = 3.499; l=0.225; g=9.81; Kmt=120; w=15;
d = 3.23*(10^-7);Ix = 0.03; Iy = 0.03; Iz=0.04;

% ===== Mencari Gain Roll dan Pitch =====
Am = [0 1 0;0 0 (Kmt*l)/Ix;0 0 -15];
```

```

Bm      = [0 0 15]';
H = [0.3 0 0]; %performansi output
Cm = [1 0 0;0 1 0];
Bwm = Bm;

%Inisialisasi
N = 100; K0=0; L0=0; gam = 0.7;
Rm = 10;
Qm =[755 0 0;
      0 3 0;
      0 0 .1];
toler = 0.1; %nilai epsilon(toleransi)

k = Hitung_OPFB_Coba(Am,Bm,Cm,Bwm,Qm,Rm,N,gam,toler,L0,H)
Ei=Am-Bm*k*Cm;
ei=eig(Ei); %eigenvalue closedloop

% condition for convergen
AQob = obsv(sqrt(Qm),Am);
AQunob = length(Am)-rank(AQob);
AQobservability = rank(AQob)

% ===== Menghitung Gain Yaw =====
Ky      = 4;
Amy     = [0 1;0 0];
Bmy     = [0 4*Ky/Iz]';
Cmy     = [1 0]; Dmy      = [0];

Qy      = diag([1 3]);
Ry      = 1;
[Kmy] = lqr(Amy,Bmy,Qy,Ry); %gain feedback
Nbar = rscale(Amy,Bmy,Cmy,Dmy,Kmy) ; %gain precompensator

%===== Menghitung Gain Z =====
Az = [0 1 0;0 0 0;1 0 0];
Bz = [0 4*Kmt/m 0]';
Cz = eye(3);
Qz = diag([0.5 1 100]);
Rz = 1;
[Kz] = lqr(Az,Bz,Qz,Rz);

%Matriks gain state-feedback K
Kzi = Kz(3);
Kzz=[Kz(1) Kz(2)];

```

3. Program Kontroler *Tracking* Posisi X

```

%Program untuk Tracking XY
m = 3.522462; l=0.225; g=9.81; Kmt=120; w=15;
d = 3.23*(10^-7); b=2.98*(10^5); Jr=2.84*(10^-5);
Ix = 0.03; Iy = 0.03; Iz=0.04;

%Matriks Sistem posisi X
Ax = [0 1 0;0 0 (4*Kmt/m)*0.2;0 0 -w];
Bx = [0 0 w]';
Hx = [1 0 0]; %performansi output

```

```

%matriks referensi
Arx = [0 1;-0.02 0];
%Augmented sistem
Brx = [zeros(1,3);-Hx];
Agx = [Arx Brx;zeros(3,2) Ax];
Bgx = [0;0;Bx];
Cgx = [1 0 0 0 0;0 1 0 0 0;0 0 1 0 0;0 0 0 1 0.0001];
%Matriks Pembobot
R0x = .1;
Q0x=diag([10 10 5 10 0.1]);

[kox]=lqr(Agx,Bgx,Q0x,R0x)
k0x = [kox(1) kox(2) kox(3) kox(4)] %Inisial Gain

% inisialisasi
v=1; tolerx=0.01;
alphax=.1; Nx=100; r0x =1;
Rx = 1;
Qx=diag([500 1 .1 500 10]);
% Gain OPFB
kx=optk2(Agx,Bgx,Cgx,k0x,Rx,Qx,Nx,tolerx,alphax)
Acx=Agx-Bgx*kx*Cgx; e=eig(Acx)

ob_posX = obsv(Agx,sqrt(Qx));
unob_posX = length(Agx)-rank(ob_posX);
observability_posX = rank(ob_posX)

=====%Matriks Sistem Y =====
Ay = [0 1 0;0 0 -4*Kmt/m;0 0 -w];
By = [0 0 w]';
Hy = [1 0 0]; %performansi output

%matriks referensi
Ary = [0 1;0 -0.02];

%Augmented sistem
Bry = [zeros(1,3);-Hy];
Agy = [Ary Bry;zeros(3,2) Ay];
Bgy = [0;0;By];
Cgy = [1 0 0 0 0;0 1 0 0 0;0 0 1 0 0;0 0 0 1 0.0001];

%Matriks Pembobot untuk inisial gain
R0y = 1;
Q0y =diag([100 10 5 100 10]);
%Inisial Gain K0
[koy]=lqr(Agy,Bgy,Q0y,R0y);
k0y=[koy(1) koy(2) koy(3) koy(4)]
% inisialisasi
v=1; tolery=0.001;
alphay=.1; Ny=100; r0y =1;
% matriks pembobot
Ry = 1
Qy=diag([400 5 25 40 1]);
% Gain Kontrol
ky=optk2_Y(Agy,Bgy,Cgy,k0y,Ry,Qy,Ny,tolery,alphay)*3
Acy=Agy-Bgy*ky*Cgy; e=eig(Acy)

```

DAFTAR PUSTAKA

- [1] Z. Zou, "Trajectory Tracking Control Design with Command-filtered Compensation for a Quadrotor," *IET Journal Control Theory Appl*, vol. 4, no. 11, pp. 2343-2355, 2010.
- [2] F. Yacef, O. Bouhali, H. Khebbache and F. Boudjema, "Takagi-Sugeno Model for Quadrotor Modelling and Control using Nonlinear State Feedback Controller," *International Journal of Control Theory and Computer Modelling (IJCTCM)*, vol. 2, 2012.
- [3] E. C. Suicmez and A. T. Kutay, "Optimal Path Tracking Control of a Quadrotor UAV," in *International Conference on Unmanned Aircraft System*, Orlando, 2014.
- [4] E. C. Vivas G, M. V. Pena G and C. I. Rodriguez F, "Simulation of The Quadrotor Controller with LQR with Integral Effect," *ABCM Symposium Series in Mechatronics*, vol. 5, pp. 390-399, 2012.
- [5] A. Fujimori, "Optimization of Static Output Feedback Using Substitutive LMI Formulation," *IEEE Transaction on Automatic Control*, vol. 49, 2004.
- [6] N. Wang, H. Pei, Y. He and Q. Zhang, "Robust H₂ Static Output Feedback Tracking Controller Design of Longitudinal Dynamics of a Miniature Helicopter via LMI Technique," in *24th Chinese Control and Decision Conference (CCDC)*, 2012.
- [7] J. Gadewadikar, F. L. Lewis, K. Subbarao and B. M. Chen, "Design of H_∞ Command and Control Loops for Unmanned Aerial Vehicles using Static Output-Feedback," in *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, 2007.
- [8] F. L. Lewis and V. L. Syrmos, *Optimal Control* 2nd edition, New York: Jhon Wiley & Sons, Inc., 1995.
- [9] Quanser, "Quanser Q-Ball X-4 User Manual," 2010, pp. i-46.
- [10] K. Ogata, *Modern Control Engineering*, 3rd Edition, New Jersey: Prentice Hall, 1997.
- [11] M. M. de Almeida Neto, "Control Strategies of a Tilt-rotor UAV for Load Transportation," Master Thesis, Universidade Federal de Minas Gerias, 2014.
- [12] K. Khalil, *Nonlinear System* 3rd Edition, New Jersey: Prentice Hall, 2000.

- [13] J. Li, H. O. Wang and D. Niemann, "Relations Between LMI (Linear Matrix Inequality) and ARE (Algebraic Riccati Equation) with their applications to Absolute Stability Criteria, Robustness Analysis and Optimal Control," 1998.
- [14] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, Philadelphia: Society for Industrial and Applied Mathematics, 1994.
- [15] K. Zhou, *Essentials of Robust Control*, New Jersey: Prentice-Hall, 1999.
- [16] V. Solovyev Viktor, I. Finaev Valery, A. Zargaryan Yuri, O. Shapovalov Igor and A. Beloglazov Denis, "Simulation of Wind Effect on A Quadrotor Flight," *ARN Journal Engineering and Applied Sciences*, vol. 10, pp. 1535-1538, 2015.
- [17] H. Arrosida, R. Effendi, T. Agustinah and J. Pramudijanto, "Design of Decoupling and Nonlinear PD Controller for Cruise Control of a Quadrotor," in *International Seminar on Intelligent Technology and Its Applications*, Surabaya, 2015.

RIWAYAT PENULIS



Feni Isdaryani dilahirkan pada tahun 1991 di Bandung, Indonesia. Pendidikan tingginya ditempuh dari jenjang Diploma III di Politeknik Negeri Bandung, Jurusan Teknik Elektronika. Penulis melanjutkan studi program Sarjana di Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember (ITS) dan mendapatkan gelar Sarjana Teknik pada tahun 2014. Dalam tahun yang sama, penulis mendapatkan beasiswa *fresh graduate* dari Departemen Pendidikan Tinggi (DIKTI) untuk program Magister di Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember dengan bidang keahlian yang sama yaitu, Teknik Sistem Pengaturan. Pada saat ini, penulis sedang menyelesaikan studi program magisternya pada bulan Juni 2016.