

Analisis Kestabilan Terbang Quadrotor Dengan Pengendali PD Terhadap Gangguan

Ardy Seto Priambodo, Adha Imam Cahyadi, Samiadji Herdjunto

Departemen Teknik Elektro dan Teknologi Informasi Fakultas Teknik Universitas Gadjah Mada
Jl. Grafika No. 2, Kampus UGM, Kabupaten Sleman, Daerah Istimewa Yogyakarta 55284
e-mail : seto.sie14@ee.ugm.ac.id adha.imam@ugm.ac.id samiadji@ugm.ac.id

Abstrak—Tujuan dari penelitian ini adalah menghasilkan sistem kendali pada sebuah obyek quadrotor yang dapat mengkompensasi gangguan. Formula Euler-Newton digunakan dalam menentukan model matematika dari sebuah obyek quadrotor. Sistem kendali yang dirancang pada penelitian ini adalah Sistem Kendali PD (Proporsional-Derivatif). Sistem kendali PD memiliki parameter K_p dan K_d yang besarnya mempengaruhi respon dari keluaran sistem kendali. Model matematika dari quadrotor dan sistem kendali PD yang dirancang kemudian akan dijalankan secara simulasi pada perangkat lunak MATLAB. Terdapat 4 buah sistem kendali yang masing-masing bekerja secara sendiri-sendiri. Gangguan yang diberikan pada simulasi model quadrotor berupa perubahan sudut pada roll, pitch, dan yaw serta perubahan ketinggian. Hasil dari simulasi ditampilkan dalam 3 dimensi sehingga memudahkan dalam proses analisis. Dari hasil simulasi menunjukkan bahwa sistem kendali PD dapat bekerja sesuai yang diharapkan pada saat terdapat gangguan.

Kata Kunci : Kendali PD, Quadrotor, MATLAB, Gangguan

Abstract— *The main purpose of this paper is to provide a control system of an object quadrotor that can compensate the disturbance. The Euler-Newton formula is used in determining the mathematical model of a quadrotor. The control system designed in this paper is PD (Proportional-Derivative) Control System. The PD control system has 2 parameters (K_p and K_d) that affect the response of the control system output. The mathematical model of the quadrotor and the designed PD control system will then be run simulated using MATLAB. There are 4 pieces of control system that each work independent. The disturbance given in the simulation of quadrotor model is angle change on roll, pitch, and yaw and altitude change. The results of the simulation are shown in 3 dimensions making it easier in the analysis process. From the simulation results show that the PD control system can work as expected when there is disturbance.*

Keywords-component : PD Controller, Quadrotor, MATLAB, Disturbance

I. PENDAHULUAN

Ketertarikan pada bidang robotika akhir-akhir cukup meningkat pesat. Banyak aplikasi pada dunia industri yang menggunakan robot untuk menggantikan peran manusia dalam menjalankan tugasnya. Mulai dari industri otomotif, manufaktur, industri makanan, obat-obatan, dan masih banyak lainnya. Robot digunakan sebagai pengganti peran manusia pada tugas-tugas yang sulit, berbahaya, dan keadaan yang membosankan sehingga hasil dari sebuah produk selalu dapat terjaga [3].

UAV adalah salah satu cabang robotika yang cukup populer sekarang ini. Obyek penelitian yang banyak dilakukan salah satunya adalah pada salah satu jenis uav, yaitu quadrotor. Masih banyak jenis uav yang lain, seperti *fixed wing* atau helikopter konvensional namun dibandingkan dengan *fixed wing* atau helikopter konvensional, quadrotor memiliki beberapa keuntungan [1]. Quadrotor dipilih pada penelitian ini karena konstruksinya yang cukup sederhana, dan mampu terbang melayang serta dapat melakukan lepas landas dan mendarat secara vertikal (biasa disebut VTOL – *Vertical Take-Off and Landing*) sehingga banyak aplikasi yang dapat diimplementasikan pada quadrotor. Aplikasi-aplikasi seperti pengawasan udara, foto dan video udara, atau misi penyelamatan pada daerah konflik dan bencana adalah beberapa aplikasi yang dapat dilakukan menggunakan quadrotor [2].

Quadcopter atau biasa disebut juga Quadrotor, merupakan jenis *rotor-craft* yang memiliki 4 buah rotor pada ujung-ujung framenya, seperti terlihat pada gambar 1. Pada penelitian ini model dari quadrotor akan dituliskan dalam bentuk persamaan matematika yang kemudian disimulasikan melalui perangkat lunak pada komputer. Selain melakukan simulasi dari model quadrotor itu sendiri, juga dilakukan simulasi mengenai sistem kendali PD yang sudah dirancang untuk menstabilkan terbang dari mode quadrotor.



Gambar 1. Bentuk Fisik Quadrotor

II. MODEL QUADROTOR

A. Gerakan Dasar Quadrotor

Quadrotor yang menjadi obyek penelitian ini adalah sebuah armada terbang yang memiliki 4 buah rotor penggerak baling-baling yang berguna untuk menghasilkan gaya angkat [5]. Konfigurasi putaran rotor yang ada pada quadrotor adalah saling berlawanan, dengan kecepatan

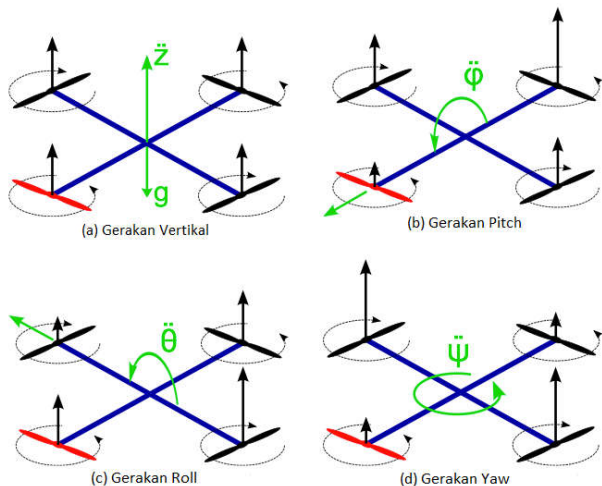
yang sama maka torsi rotasi yang terjadi pada putaran rotor dapat diabaikan. Gerakan dasar dari quadrotor seperti terlihat pada gambar 2.

Gerakan pada quadrotor dibagi menjadi 4 gerakan dasar [4]. Gerakan pertama adalah gerakan vertikal, merupakan gerakan bergerak naik dan turun. Gerakan ini terjadi karena perbedaan kecepatan tiap-tiap motor dengan kecepatan nominal. Apabila kecepatan motor 1-4 melebihi kecepatan nominal maka quadrotor akan bergerak naik dan sebaliknya apabila kecepatan motor 1-4 dibawah kecepatan nominal maka quadrotor akan bergerak turun.

Gerakan yang kedua adalah gerakan roll, yaitu perubahan sudut yang bernilai positif maupun negatif terhadap sumbu X. Gerakan ini dihasilkan dari perbedaan kecepatan antara motor 2 dan 4. Apabila kecepatan motor 2 melebihi kecepatan motor 4 maka yang terjadi adalah roll kiri dan begitupula sebaliknya apabila kecepatan motor 2 lebih rendah daripada kecepatan motor 4 maka yang terjadi adalah roll kanan.

Gerakan ketiga yang ada pada quadrotor adalah gerakan pitch. Gerakan ini sama pada gerakan roll namun perubahan sudutnya terhadap sumbu Y. Motor yang berpengaruh untuk menghasilkan gerakan ini adalah motor 1 dan motor 3. Apabila kecepatan motor 1 lebih cepat dibandingkan motor 3 maka gerakan yang terjadi adalah pitch mundur. Namun apabila kecepatan motor 3 lebih cepat dibandingkan kecepatan motor 1 maka yang terjadi adalah pitch maju.

Gerakan yang terakhir adalah gerakan yaw. Gerakan ini dihasilkan karena perbedaan putaran pasangan motor 1 & 3 serta motor 2 & 4. Ketika motor 1 & 3 putarannya lebih cepat dibandingkan dengan motor 2 & 4 maka yang terjadi adalah yaw kiri begitu pula sebaliknya ketika motor 2 & 4 lebih cepat putarannya dibandingkan putaran motor 1 & 3 maka yang terjadi adalah yaw kanan.

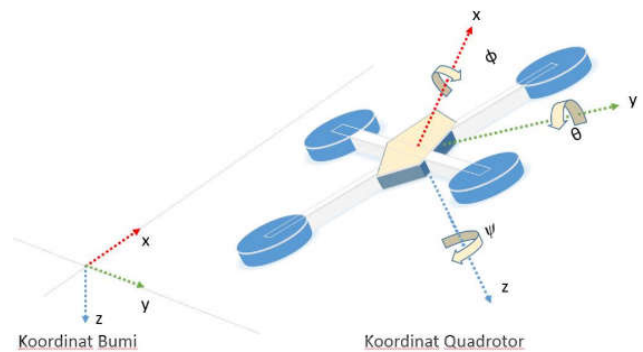


Gambar 2. Gerakan Dasar Quadrotor

B. Kinematika dan Dinamika Quadrotor

Ada 2 sistem koordinat yang digunakan pada quadrotor, *Inertial Frame* atau juga disebut *Frame Bumi* (E-Frame)

dan *Frame Badan* (B-Frame) [4]. Sistem koordinat ini digunakan untuk mengetahui posisi translasi dan posisi rotasi yang terjadi pada quadrotor. E-Frame kondisinya tetap pada suatu titik sedangkan B-Frame letaknya berubah-ubah sesuai kondisi dari quadrotor tersebut.



Gambar 3. Sistem Koordinat pada Quadrotor

Posisi quadrotor pada sistem koordinat E-Frame x, y, z didefinisikan dengan ξ sedangkan posisi sudut pada B-Frame didefinisikan dengan η . Sudut pitch θ merupakan rotasi quadrotor pada sumbu y. Sudut roll ϕ merupakan rotasi quadrotor pada sumbu x dan sudut yaw ψ pada sumbu z. Vektor q terdiri dari vektor posisi translasi ξ dan vektor euler angle η . Persamaan vektor posisi translasi ξ dan vektor euler angle η dapat dilihat pada persamaan 1.

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, q = \begin{bmatrix} \xi \\ \eta \end{bmatrix} \tag{1}$$

$$V_B = \begin{bmatrix} V_{x, B} \\ V_{y, B} \\ V_{z, B} \end{bmatrix}, v = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2}$$

Pada B-Frame, vektor kecepatan gerak translasi didefinisikan dengan V_B dan vektor kecepatan sudut dengan v . Hubungan antara vektor posisi translasi ξ dan vektor kecepatan V_B ditunjukkan pada persamaan 3.

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = R \begin{pmatrix} V_{x, B} \\ V_{y, B} \\ V_{z, B} \end{pmatrix} \tag{3}$$

dimana matriks rotasi R dari B-Frame ke E-Frame adalah

$$R = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \tag{4}$$

dimana $S_x = \text{Sin}(x)$ dan $C_x = \text{Cos}(x)$. Matriks rotasi R adalah matriks orthogonal sehingga $R^{-1} = R^T$ yang mana merupakan matriks rotasi dari E-Frame ke B-Frame. Matriks transformasi untuk kecepatan sudut dari E-Frame ke B-Frame dinotasikan dengan W_η dan dari B-Frame ke

E-Frame adalah W_{η}^{-1} seperti yang ditunjukkan pada persamaan 5.

$$\dot{\eta} = W_{\eta}^{-1}v, \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & S_{\phi} T_{\theta} & C_{\phi} T_{\theta} \\ 0 & C_{\phi} & -S_{\phi} \\ 0 & S_{\phi}/C_{\theta} & C_{\phi}/C_{\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (5)$$

$$v = W_{\eta} \dot{\eta}, \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_{\phi} \\ 0 & C_{\phi} & C_{\theta} S_{\phi} \\ 0 & -S_{\phi} & C_{\theta} C_{\phi} \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (6)$$

dimana $T_x = \tan(x)$.

Quadrotor diasumsikan memiliki struktur yang simetris dengan empat lengan yang sejajar dengan B-Frame pada sumbu x dan y. Dengan demikian, matriks inersia adalah matriks diagonal dimana $I_{xx}=I_{yy}$.

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (7)$$

Dinamika quadrotor dijabarkan menggunakan persamaan differensial yang diperoleh dengan metode Newton-Euler seperti pada persamaan 8-13 [8]. Dari persamaan ini kita membuat model simulasi dari quadrotor melalui software simulink matlab.

$$\ddot{x} = (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) U1/m \quad (8)$$

$$\ddot{y} = (\cos \phi \sin \theta \cos \psi - \sin \phi \sin \psi) U1/m \quad (9)$$

$$\ddot{z} = -g + (\cos \phi \cos \theta) U1/m \quad (10)$$

$$\ddot{\phi} = (I_{yy} - I_{zz}/I_{xx})qr + (Jr/I_{xx})q\Omega + U2/I_{xx} \quad (11)$$

$$\ddot{\theta} = (I_{zz} - I_{xx}/I_{yy})pr + (Jr/I_{yy})p\Omega + U3/I_{yy} \quad (12)$$

$$\ddot{\psi} = (I_{xx} - I_{yy}/I_{zz})pq + U4/I_{zz} \quad (13)$$

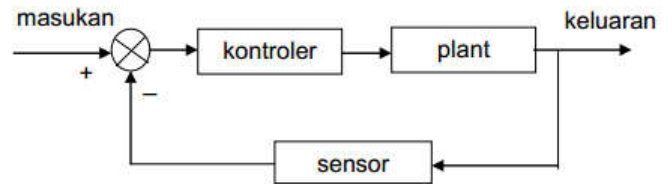
III. PERANCANGAN SISTEM KENDALI PD

Dasar dari sistem kendali berbasis PID adalah pembacaan sensor yang kemudian dibandingkan dengan suatu referensi (set poin) dan hasil keluarannya dihitung sesuai dengan parameter proporsional, integral dan derivatif-nya. Hasil total dari kendali PID merupakan gabungan atau penjumlahan dari tiap-tiap komponen P, I, dan D [9]. Pada kasus ini sistem kendali yang digunakan adalah PD yang terdiri atas kendali Proporsional dan Derivatif. Sebelum membahas tentang kendali PD dasar yang perlu dipahami adalah mengenai sistem kalang tertutup.

A. Sistem Kalang Tertutup

Dalam sebuah kendali sistem secara umum, ada variabel proses yang merupakan parameter dari sistem

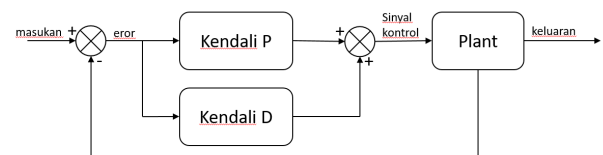
yang perlu di kendali seperti suhu (°C), tekanan (psi), atau kecepatan putar suatu motor. Untuk mengetahui variabel proses tersebut maka dibutuhkan sensor yang kemudian menjadi masukan sebuah sistem kendali [6]. Dari hasil pengukuran sensor tersebut dapat diketahui perbedaan antara hasil yang terukur pada kondisi riil dan hasil yang diharapkan (set poin), perbedaan tersebut dinamakan error atau kesalahan. Misal suatu tungku pemanas diharuskan bekerja pada suhu 120°C namun yang terbaca dari sensor adalah 100°C, maka erornya adalah 20°C. Sistem kendali akan mengolah suatu komputasi berdasarkan dari error yang ada sehingga mengeluarkan sinyal kendali agar pemanas tersebut menaikkan suhunya sampai yang diinginkan atau erornya mendekati 0°C. Diagram blok dari kalang terbuka system ditunjukkan pada gambar 4.



Gambar 4. Diagram Blok Sitem Kalang Tertutup

B. Sistem Kendali PD

Merupakan sistem kendali yang terdiri dari 2 kendali yaitu kendali proporsional dan derivatif [6]. Pada kendali proporsional, hasil sinyal output dari kendali ini adalah perkalian dari error dan parameter kendali p, sehingga apabila error nilainya besar maka sinyal kendali yang dihasilkan akan besar pula. Parameter kendali p disebut K_p dan nilainya sangat mempengaruhi hasil dari kendali ini, semakin besar nilai K_p maka respon akan semakin cepat namun apabila terlalu besar maka respon akan berosilasi dan ada pada suatu kondisi dimana apabila K_p sangat besar maka osilasi akan semakin membesar dan membuat sistem tidak terkendali.



Gambar 5. Diagram Blok Sistem Kendali PD

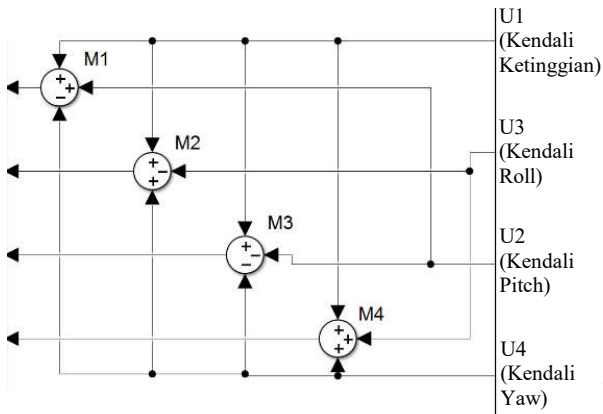
Kendali derivatif merupakan suatu kendali dengan respon output menurun jika variabel proses meningkat dengan cepat. Respon output pada kendali ini sebanding dengan perubahan dari variabel proses. Peningkatan dari parameter T_d (waktu derivatif) akan menyebabkan sistem kontrol bereaksi lebih responsif terhadap perubahan dari variabel proses dan akan meningkatkan kecepatan respon sistem kontrol secara keseluruhan. Sebagian besar sistem kontrol yang diaplikasikan menggunakan parameter T_d yang kecil karena respon dari kendali ini cukup sensitif dengan adanya noise pada variabel proses. Diagram blok dari kendali PD ditunjukkan pada gambar 5.

$$u(t) = K_p e(t) + K_d de(t) / dt \tag{14}$$

Persamaan dari sistem kendali PD ini ditunjukkan dalam persamaan 14 [7]. Dimana $u(t)$ merupakan hasil sinyal kendali, $e(t)$ adalah eror atau selisih antara hasil terukur dan set poin, K_p adalah parameter dari kendali proporsional, dan K_d adalah parameter dari kendali derivatif. Dalam bentuk diskrit persamaan dari sistem kontroler PD ditunjukkan pada persamaan 15. Pada penelitian ini sesuai dengan dasar teori gerakan dari quadrotor dan hasil dari model yang didapatkan, untuk menerbangkan quadrotor dalam keadaan melayang maka ada 4 variabel yang harus kita kendalikan. Variabel tersebut adalah sudut pitch θ , roll ϕ , yaw ψ dan ketinggian Z. Semua variabel-variabel tersebut dapat dikendalikan secara independen sehingga masing-masing memiliki sistem kontrol PD sendiri dengan parameter yang tentunya berbeda satu dengan lainnya.

$$u(k) = K_p e_k + (1/T)K_d(e_k - e_{k-1}) \tag{15}$$

Keluaran masing-masing dari tiap kontroler memiliki efek terhadap kecepatan putaran rotor M1, M2, M3, dan M4 [9]. Sesuai dengan dasar teori pergerakan dari quadrotor, misal kontroler sudut pitch θ , akan mempengaruhi kecepatan putar M1 dan M3 yang di representasikan dengan U2. Begitupula pada kontroler sudut roll ϕ , akan mempengaruhi kecepatan putar motor M2 dan M4 yang di representasikan dengan U3, dan untuk kontroler yaw ψ dan ketinggian Z memiliki pengaruh pada motor M1-M4 yang di representasikan dengan U4 dan U1. Diagram blok dari kombinasi beberapa kontroler ini terhadap motor M1-M4 dapat di lihat pada gambar 6.



Gambar 6. Kombinasi Keluaran Kendali Terhadap Motor Quadrotor

IV. SIMULASI MODEL DAN SISTEM KENDALI PD

Berdasarkan persamaan model matematika yang kita dapat sebelumnya, maka kita bisa membuat sebuah model simulasi quadrotor. Program yang digunakan untuk menjalankan simulasi ini adalah matlab simulink. Nilai parameter dari fisik quadrotor yang digunakan dalam simulasi ini dipresentasikan pada tabel 1. Ada 2 blok besar yang terdiri atas blok sistem kendali dan blok model quadrotor. Untuk mengatur kestabilan terbang melayang

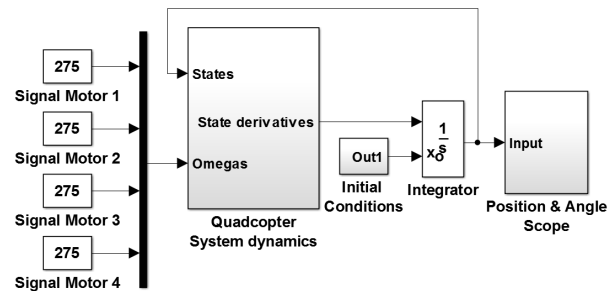
dari quadrotor terdapat 4 buah set poin yaitu roll, pitch, yaw, dan ketinggian. Keluaran dari kontroler PD merupakan sinyal kontrol U1-U4 yang menjadi masukan dari blok model quadrotor. Model quadrotor yang sudah dibuat memiliki 12 keluaran, dari keluaran tersebut dapat kita ambil data posisi translasi X,Y, Z dan posisi sudut pitch θ , roll ϕ , yaw ψ . Blok lainnya pada simulasi adalah blok *scope* yang berisi komponen *scope* dan komponen *workspace* guna menyimpan variabel ke workspace matlab agar mempermudah dalam penggambaran grafik. Blok diagram lengkap dari simulasi dapat dilihat pada gambar 7.

TABLE I. NILAI PARAMETER SIMULASI QUADROTOR

Nama	Parameter	Nilai	Satuan
gravitasi	g	9.81	m/s ²
massa	m	1.0	kg
Panjang lengan	l	0.25	m
koefisien thrust	b	3.25×10 ⁻⁵	N s ²
koefisien drag	d	7.5×10 ⁻⁷	Nm s ²
inersia sumbu X	I _{xx}	7.5×10 ⁻³	kg m ²
inersia sumbu Y	I _{yy}	7.5×10 ⁻³	kg m ²
inersia sumbu Z	I _{zz}	1.3×10 ⁻²	kg m ²
inersia motor	J _r	6×10 ⁻⁶	kg m ²

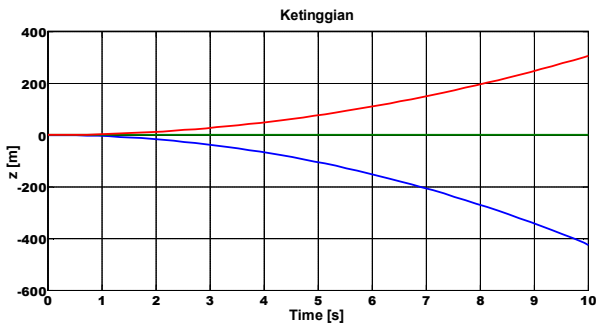
A. Simulasi Sistem Kalang Terbuka

Setelah model matematika dari obyek quadrotor sudah selesai dirancang, langkah pertama yang dilakukan adalah melakukan simulasi pada model yang sudah kita buat dengan memberikan masukan secara langsung atau disebut simulasi sistem kalang terbuka. Dengan memberikan beberapa nilai masukan pada kecepatan motor 1-4 kita mendapatkan beberapa respon pada sudut posisi Z.



Gambar 7. Diagram Blok Simulasi Sistem Kalang Terbuka Quadrotor

Masukan yang diberikan nilainya adalah 100, 275, dan 350. Untuk respon dari tiap-tiap masukan dapat dilihat pada gambar 8. Garis biru, garis hijau, dan garis merah menggambarkan respon dengan masukan 100, 275, dan 350 secara berurutan. Karena nilai tiap motor yang diberikan adalah sama besarnya maka tidak ada perubahan pada sudut roll, pitch, yaw dan perubahan hanya terjadi pada ketinggian dari quadrotor.

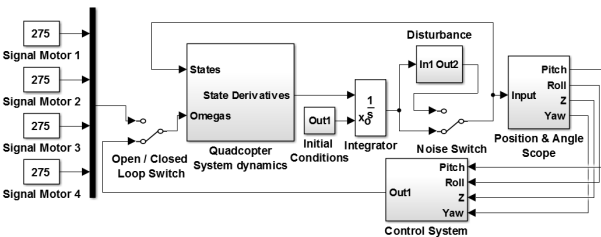


Gambar 8. Respon Ketinggian Quadrotor pada Simulasi Sistem Kalang Terbuka dengan Masukan = 100 (biru), 275 (hijau), dan 350 (merah)

Terlihat pada gambar 8 respon ketinggian dengan 3 nilai masukan berbeda, dari respon tersebut dapat diambil kesimpulan bahwa nilai 275 adalah nilai nominal agar quadrotor dapat terbang melayang. Nilai ini nantinya akan menjadi acuan dalam perancangan sistem kendali.

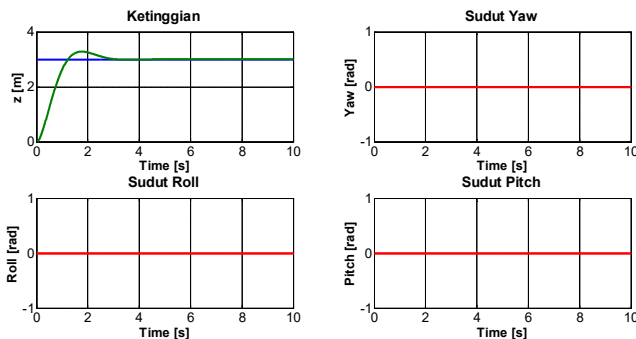
B. Simulasi Sistem Kalang Tertutup

Pada simulasi ini sistem kendali yang sudah dirancang kemudian difungsikan. Ada 4 buah set poin yang ada pada masing-masing sistem kendali. PD. Keluaran dari model quadrotor yaitu roll, pitch, yaw dan ketinggian menjadi umpan balik menuju kendali. Terlihat diagram simulasi pada simulink matlab, sistem kendali PD dan model dari quadrotor pada gambar 9.



Gambar 9. Diagram Blok Simulasi Sistem Kalang Tertutup Quadrotor

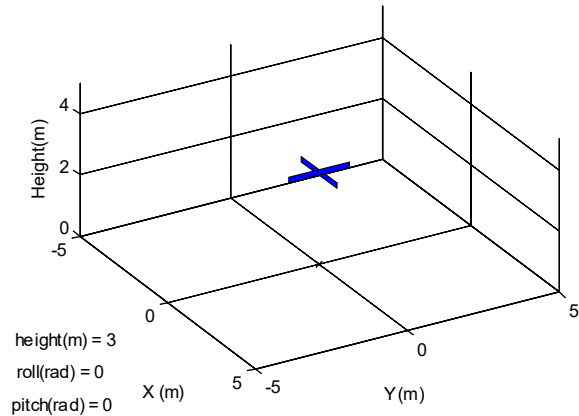
Kondisi awal dari quadrotor adalah ketinggian = 0, sudut roll, pitch, dan yaw = 0. Untuk mengatur kondisi awal pada simulink matlab bagian yang perlu diatur adalah pada bagian integrator. Untuk nilai set poin yang diatur adalah ketinggian = 3, sudut roll, pitch, dan yaw = 0. Respon yang terjadi ketika simulasi dijalankan dapat dilihat pada gambar 10.



Gambar 10. Respon Keluaran Z, Roll, Pitch, dan Yaw

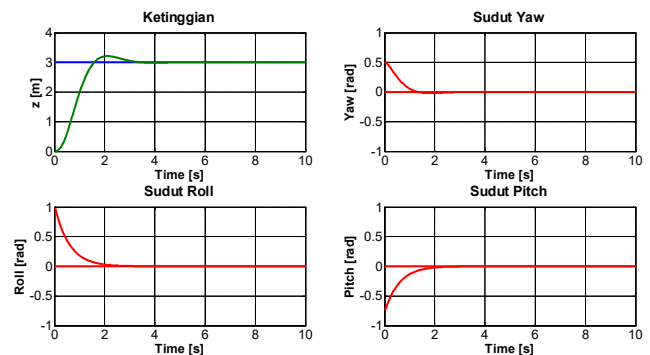
Pada gambar 10 terlihat quadrotor melakukan lepas landas dari ketinggian 0 sampai naik ke set poin 3m dengan baik walaupun ada sedikit overshoot. Rise time pada respon ketinggian adalah 1.32 detik, peak time pada 1.74 detik, dan settling time pada detik ke 3.22. Untuk sudut roll, pitch, dan yaw karena kondisi awal sudut quadrotor baik sudut roll, pitch, dan yaw adalah 0 dan set poin yang diberikan untuk sudut roll, pitch, yaw adalah 0 maka tidak ada perubahan yang terlihat pada bagian ini. Untuk membantu analisis dan melihat pergerakan quadrotor secara 3 dimensi maka ditampilkan simulasi pergerakan quadrotor secara 3d seperti terlihat pada gambar 11.

Quadcopter 3D Simulation



Gambar 11. Simulasi 3D Quadrotor

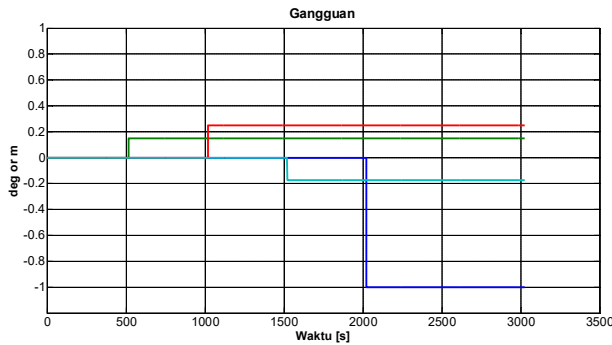
Pada simulasi selanjutnya diatur kondisi awal ketinggian = 0, sudut roll = 1, pitch = -0.75, dan yaw = 0.5. Respon keluaran ketinggian, sudut roll, pitch, dan yaw dapat dilihat pada gambar 12. Terlihat respon keluaran dari ketinggian dapat mengikuti set poin pada detik ke-4. Untuk sudut roll settling time adalah 3.57 detik, sudut pitch settling time adalah 3.07 detik, dan terakhir untuk sudut yaw settling time adalah 3.24 detik. Hal ini menunjukkan kontroler PD yang dirancang bekerja dengan baik sehingga respon mendekati nilai set poin.



Gambar 12. Respon keluaran Z, Roll, Pitch, Yaw

Simulasi selanjutnya dilakukan dengan memberikan gangguan pada sudut roll dengan gangguan berupa bentuk step yang nilai akhirnya adalah 0.15 rad. Gangguan jg diberikan pada sudut pitch dan yaw yang

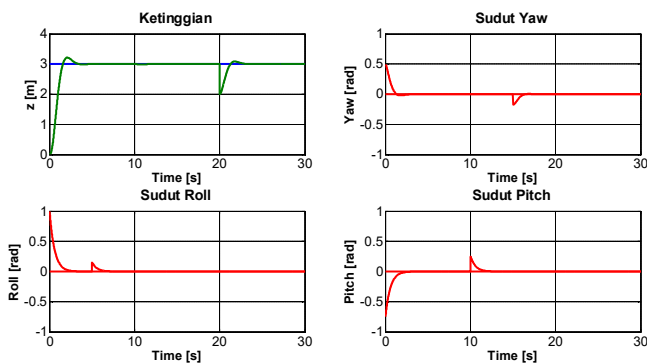
masing-masing juga merupakan bentuk step dengan nilai akhir 0.25 dan -0.175 rad. Sedangkan gangguan ketinggian diberikan pada detik ke 20 dengan bentuk yang sama juga yaitu step dengan nilai akhir -1. Bentuk sinyal dari gangguan yang diberikan dapat dilihat pada gambar 13.



Gambar 13. Sinyal gangguan sudut roll (hijau), pitch (merah), yaw (biru), dan ketinggian (ungu)

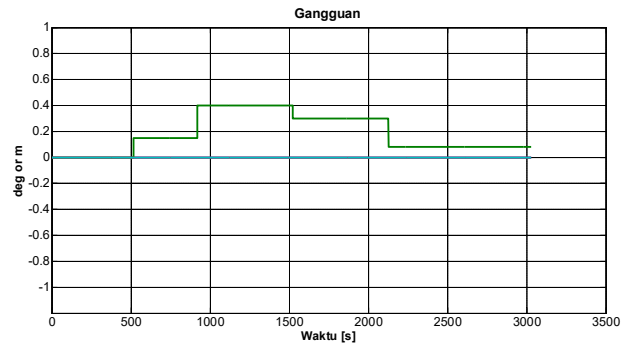
Hasil keluaran sudut roll, pitch, yaw dan ketinggian dari quadrotor dikatakan baik dan kontroler bekerja sesuai dengan yang diharapkan karena ketika mendapatkan gangguan segera dapat mengoreksi menuju ke titik set poin. Hal ini terlihat pada gambar 14 yang menunjukkan respon quadrotor ketika mendapatkan gangguan.

Terlihat di gambar 14 pada respon ketinggian untuk mengoreksi gangguan ketika diturunkan 1m di detik ke-20 dilakukan dengan baik dan dapat kembali ke set poin setelah 2-3 detik. Koreksi pada sudut roll, pitch, dan yaw saat diberikan gangguan dapat dikatakan baik dan sesuai yang diinginkan. Terbukti waktu *settling time* untuk ketiga respon ini kurang dari 3 detik.



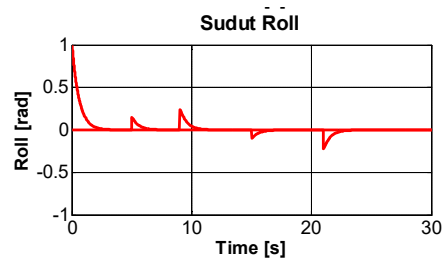
Gambar 14. Respon ketinggian, sudut roll, pitch, dan yaw ketika mendapatkan gangguan gangguan

Kita juga melakukan pengujian pada salah satu sudut, yaitu sudut roll dengan beberapa gangguan sekaligus dan pengamatan dilakukan pada hasil keluarannya. Untuk set poin yang digunakan adalah tetap yaitu 0 deg. Bentuk dari gangguan yang diberikan terlihat seperti pada gambar 15. Gangguan ini merupakan gabungan dari beberapa unit step yang dilakukan bertahap pada waktu-waktu tertentu.



Gambar 15. Sinyal gangguan sudut roll (hijau)

Hasil yang dapat diamati pada keluaran sudut roll terlihat pada gambar 16. Ada 4 gangguan yang diberikan pada waktu tertentu, besarnya bervariasi mulai dari nilai positif hingga negatif. Dari respon yang terjadi dapat dilihat kontroler bekerja dengan baik dan keluaran selalu mendekati dengan nilai set poin yang diberikan.



Gambar 16. Respon sudut roll pada beberapa gangguan

V. KESIMPULAN

Quadrotor adalah sistem MIMO (*Multiple Input Multiple Output*) yang terdiri dari 4 masukan (kecepatan tiap motor) dan 6 keluaran (posisi translasi x,y,z dan posisi sudut roll, pitch, dan yaw). Secara umum, untuk merancang sistem kendali pada sistem MIMO adalah kompleks, namun pada model quadrotor sistem kendali yang dirancang bisa dikatakan sederhana karena menjadi SISO (*Single Input Single Output*). Pada kendali ketinggian misalnya, kita cukup merancang sistem kendali PD untuk 1 masukan yaitu ketinggian, dan 1 keluaran. Begitu pula pada sudut roll, pitch, dan yaw yang mana tiap-tiap kendali memiliki masukan sendiri-sendiri dengan set poin sendiri-sendiri tanpa ada pengaruh dari kendali lain. Jadi ke-4 sistem kendali yang dirancang ini sifatnya independen. Hasil respon baik ketinggian, maupun sudut roll, pitch, dan yaw terhadap gangguan dapat dikatakan baik karena pada grafik hasil menunjukkan respon mendekati set poin sesaat ketika mendapat gangguan yang berupa bentuk step.

REFERENCES

[1] Tripathi, Vibhu Kumar, et al. "Design of Sliding Mode and Backstepping Controllers for a Quadrotor." 2015 39th National Systems Conference (NSC), IEEE, Dec. 2015.

- [2] Sam, Rosidah, et al. "Quad-Copter Using ATmega328 Microcontroller." 2013 International Conference on Electrical Machines and Systems (ICEMS), IEEE, Oct. 2013.
- [3] De Carli, Davide, et al. "Measuring Intent in Human-Robot Cooperative Manipulation." 2009 IEEE International Workshop on Haptic Audio Visual Environments and Games, IEEE, Nov. 2009.
- [4] Bresciani, Tommaso. "Modelling, Identification and Control of a Quadrotor Helicopter." Master Thesis, Lund University, Oct. 2008.
- [5] Gupte, Shweta, et al. "A Survey of Quadrotor Unmanned Aerial Vehicles." 2012 Proceedings of IEEE Southeastcon, IEEE, Mar. 2012.
- [6] Ogata, Katsuhiko. "Modern Control Engineering Fifth Edition," New Jersey: Prentice Hall, 2010.
- [7] F. A. Salem and A. A. Aly, "PD Controller Structures: Comparison and Selection for an Electromechanical System," International Journal of Intelligent Systems and Applications, vol. 7, no. 2. MECS Publisher, pp. 1–12, 08-Jan-2015.
- [8] H. C. T. E. Fernando, A. T. A. De Silva, M. D. C. De Zoysa, K. A. D. C. Dilshan, and S. R. Munasinghe, "Modelling, simulation and implementation of a quadrotor UAV," 2013 IEEE 8th International Conference on Industrial and Information Systems. IEEE, Dec-2013.