

Rancang Sistem Pengaturan Gerakan Vertikal Pada *Towercopter* Menggunakan Kontrol *Proportional Integral Derivatif* (PID)

PERANCANGAN SISTEM PENGATURAN GERAKAN VERTIKAL PADA *TOWERCOPTER* MENGGUNAKAN KONTROL *PROPORTIONAL INTEGRAL DERIVATIF* (PID) BERBASIS LABVIEW

Chintia Putri Unarya

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya, Ketintang 60231, Indonesia
e-mail : chintiaunarya@mhs.unesa.ac.id

Puput Wanarti Rusimamto

Dosen Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya, Ketintang 60231, Indonesia
e-mail : puputwanarti@unesa.ac.id

Abstrak

Salah satu alat bidang *aeromodeling* adalah pesawat tanpa awak atau *Unmanned Aerial Vehicle* (UAV). Pesawat tanpa awak atau *Unmanned Aerial Vehicle* (UAV) salah satunya *singlecopter*. Prinsip kerja dari pesawat tanpa awak yakni *Vertical Take-Off and Landing* (VTOL) yaitu menstabilkan posisi ketinggian dengan bergerak pada sumbu Y. *Towercopter* merupakan *prototype singlecopter* yang secara khusus bergerak secara vertikal. *Prototype* ini dirancang menggunakan 1 *motor brushless* dan 1 *propeller* yang dipasang pada *frame* diantara 2 tiang penyangga. Tujuan penelitian ini adalah untuk menghasilkan sistem pengendali posisi ketinggian pada *towercopter* menggunakan PID serta mengetahui hasil respon sistem dinamik pada *towercopter* dengan mengimplementasikan nilai K_p , K_i dan K_d . *Software* LabVIEW 2014 digunakan untuk menampilkan hasil respon secara *real-time*. Hasil penelitian menunjukkan pada *setpoint* 20 cm tanpa beban menghasilkan nilai $Ess = 0.11\%$ dan nilai $MO = 4.89\%$. Sedangkan pengujian *setpoint* 20 cm dengan beban sebesar 2.32 gr menghasilkan nilai $Ess = 1.67\%$ dan $MO = 16.89\%$.

Kata kunci : *Towercopter*, PID, LabVIEW.

Abstract

One of the tools for aeromodelling is Unmanned Aerial Vehicle (UAV). Unmanned Aerial Vehicle (UAV) is one of the helicopters. The working principle of unmanned aircraft is Vertical Take-Off and Landing (VTOL), which stabilizes the height position by moving on the Y-axis. Towercopter is a single copter prototype that specifically moves vertically. This prototype is designed using 1 brushless motor and 1 propeller mounted on the frame between 2 support poles. The purpose of this study is to produce a height position control system on tower copter using PID and find out the results of a dynamic system response on tower copter by implementing the values of K_p , K_i , and K_d . LabVIEW 2014 software is used to display response results in real-time. The results showed that at a 20 cm no-load setpoint the Ess value was 0.11% and the MO value was 4.89%. While testing a 20 cm setpoint with a load of 2.32 gr produces an Ess value = 1.67% and $MO = 16.89\%$.

Keywords: Towercopter, PID, LabVIEW.

PENDAHULUAN

Pesawat udara tanpa awak atau *Unmanned Aerial Vehicle* (UAV) merupakan teknologi yang sedang mengalami perkembangan yang pesat. UAV memiliki potensi yang besar, baik untuk keperluan sipil maupun militer. Contoh pengaplikasian pada UAV adalah untuk memantau adanya kebakaran, kebutuhan survey, patroli, riset dan kebutuhan lainnya. UAV dapat digunakan pada misi berbahaya tanpa membahayakan manusia. (Priambodo dkk, 2012). Pesawat tanpa awak memiliki beberapa jenis salah satunya pesawat *singlecopter*. *Singlecopter* ini bergerak dengan 1 buah motor. *Singlecopter* dapat melakukan *take off and landing* secara vertikal tetapi rentan akan gangguan dari luar seperti angin. (Jae-Sung, 2018).

Pada tahun 2012, David Sánchez Benítez, dkk melakukan penelitian yang berjudul "Vertical Rotor for The Implementation of Control Laws". Pada penelitian ini, *Towercopter* dapat bergerak ke atas menggunakan 1 buah motor brushless. *Towercopter* dikembangkan dengan kontroler PID untuk sistem pengaturan ketinggiannya. Hasil yang diperoleh adalah untuk kontroler PID, nilai parameter $K_p = 0.9471$, $K_i = 0.8094$, $K_d = 0.1001$ yang didapat dari metode *trial and error* dan menghasilkan respon sistem dapat mengikuti setpoint dengan rata-rata maksimal overshoot $<15\%$ dan *error steady state* $<20\%$.

Pada tahun 2019, Rifqi Nabila Zufar melakukan penelitian dengan judul "Rancang Bangun sistem kendali posisi ketinggian pada towercopter berbasis proportional integral menggunakan tuning artificial neural network".

Dalam penelitian ini tentang sistem kendali posisi ketinggian pada *towercopter*. *Towercopter* menggunakan kendali PI (Propotional Integral) dengan nilai parameter $K_p = 0.16$ dan $K_i = 0.51$. Hasil respon yang didapat adalah *plant* dapat mengikuti setpoint dengan maksimal *overshoot* 22.3%.

Penelitian menerapkan kendali PID (*Proportional Integral Derivatif*) untuk mengatur posisi ketinggian yang digerakkan keatas oleh motor brushless. Salah satu aspek yang dikembangkan adalah kecepatan putar baling - baling dan posisi dalam mencapai setpoint karena semakin tinggi kecepatan putarnya maka semakin mencapai posisi yang ditentukan sebaliknya jika semakin rendah kecepatan putarnya maka semakin sulit untuk mencapai setpoint. Rumusan masalah dari penelitian ini adalah bagaimana merancang sistem pengendali posisi ketinggian pada *towercopter* dengan kendali PID dan hasil respon sistem dinamik pengendali posisi ketinggian dengan PID. Dengan demikian dapat diperoleh nantinya tujuandari penelitian ini adalah untuk merancang sistem pengendali posisi ketinggian pada *towercopter* dengan PID dan untuk mengetahui hasil respon dinamikanya.

KAJIAN PUSTAKA

Towercopter

Towercopter terdiri dari 1 buah motor Brushless dengan propellernya yang dipasang pada 1 frame diantara ada 2 tiang penyangga. *Towercopter* merupakan sebuah prototype digunakan mengembangkan sistem kendali pada quadcopter. Prinsip kerja dari alat ini adalah motor akan bekerja meluncur secara vertikal. Prinsip kerja alat ini adalah gaya dorong yang dihasilkan berasal dari kecepatan putar propelernya. Semakin cepat putaran propeller, semakin besar pula gaya dorong yang dihasilkan. Untuk mempertahankan tinggi motor maka perlu adanya sebuah metode kontrol yang lebih baik (*Benítez- David Sánchez, 2012*).

Pemodelan Matematis Sistem

Gaya thrust adalah gaya yang mengakibatkan towercopter bergerak naik atau turun searah sumbu Y. Gaya dorong terjadi jika $F_{thrust} > F_{grafitasi}$ maka kerangka akan bergerak keatas. F_{gesek} berpengaruh pada F_{thrust} ataupun $F_{grafitasi}$. Pada saat kerangka terangkat terangkat ke atas F_{gesek} akan bernilai (-) terhadap F_{thrust} , sebaliknya jika kerangka menuju kebawah F_{gesek} bernilai (+) terhadap F_{thrust} (*Sanchez-Benitez, 2012*). Sehingga berdasarkan *Newton's second law* didapatkan persamaan sebagai berikut,

$$M \vec{a} = -M\vec{g} + \vec{F}_{gesek} + \vec{F}_{thrust} \tag{1}$$

Keterangan:

- M : Massa (gr)
- \vec{a} : Vector Percepatan
- \vec{g} : Vector Gravitasi
- \vec{F}_{gesek} : Gaya Gesek (N)
- \vec{F}_{thrust} : Gaya Angkat (N)

Massa kabel yang terdapat pada kerangka juga diperhitungkan karena dapat menambah masa dari kerangka. Sehingga didapatkan,

$$M = M_{fr} + M_c = (M_{fr} + \rho_c)X \tag{2}$$

Keterangan:

- M_{fr} : Massa Frame (gr)
- M_c : Massa Kabel (gr)
- ρ_c : Massa Jenis Kabel (gr/cm³)
- X : Panjang Kabel (m)

F_{gesek} *Towercopter* tergantung pada kecepatan. Semakin cepat frame bergerak pada sumbu vertikal, semakin sedikit resistensi yang yang dihadapi. Selain itu, karena gaya gravitasi diambil sebagai negatif dan dorong sebagai kekuatan positif, tanda kecepatan termasuk dalam ekspresi gesekan. Kemudian, F_{gesek} dapat dimodelkan sebagai berikut :

$$F_{gesek} = - \frac{\dot{x}\mu_s}{(1+\mu_s)} \tag{3}$$

Keterangan:

- \dot{x} : Turunan Pertama dari x
- μ_s : Koeffisien Gesek

Gaya yang dihasilkan oleh propeller sebanding dengan sinyal kontrol. Untuk menyederhanakan model, ini dinyatakan sebagai :

$$F_{thrust} = Ku \tag{4}$$

Keterangan:

- K : Konstanta Sistem
- u : Sinyal Kontrol

Berdasarkan persamaan 2 sampai dengan persamaan 4, dapat diringkas dan diperoleh persamaan dinamis dari *Towercopter* sebagai berikut:

$$M\ddot{x} = -Mg - \frac{\dot{x}\mu_s}{(1+\mu_s)} + F_{thrust} \tag{5}$$

Keterangan:

- \ddot{x} : Turunan Kedua dari x

Rancang Sistem Pengaturan Gerakan Vertikal Pada *Towercopter* Menggunakan Kontrol *Proportional Integral Derivatif* (PID)

Persamaan 6 dapat disederhanakan sehingga:

$$\dot{x} = -\frac{g(M_{fr} + \rho_c)}{M_{fr} + \rho_c} x - \frac{\mu_s}{M_{fr} + \rho_c} \dot{x} + \frac{K}{M_{fr} + \rho_c} u \quad (6)$$

Persamaan 6 kemudian dapat ditulis kedalam 2 dimensional *State Space*:

$$\dot{x}_1 = x_2 \quad (7)$$

$$\dot{x}_2 = -\frac{g(M_{fr} + \rho_c)}{M_{fr} + \rho_c} x_1 - \frac{\mu_s}{M_{fr} + \rho_c} x_2 + \frac{K}{M_{fr} + \rho_c} u \quad (8)$$

Keterangan:

\dot{x}_n : Dimensional *State Space*
 n : Nilai ke-n

Linearisasi

Perkembangan sistem nonlinear dapat didekati dengan sistem linear dengan teori linearisasi. Teori linearisasi digunakan untuk mengatasi permasalahan ketidaklinearan sistem, sehingga sistem yang nonlinear dapat didekati dengan sistem linear. Teknik yang digunakan untuk linearisasi adalah menggunakan Matriks *Jacobi* (Wicaksono, 2016).

$$y = (c) \quad (9)$$

Keterangan:

y : Keluaran Sistem
 c : Masukan Sistem

Sistem nonlinear merupakan permisalan dari nilai (x), sehingga:

$$f(c) = \dot{x} = \left. \frac{df(c)}{dx} \right|_{c=x} c \quad (10)$$

Keterangan:

f(c) : Keluaran Sistem
 df(c) : Turunan Fungsi
 dx : Differensial x

Sistem nonlinear loop terbuka dengan jumlah state sama. Persamaan 10 menunjukkan bahwa sistem nonlinear dapat didekati menjadi sistem linear di sekitar titik ekuilibriumnya dengan A merupakan Matriks *Jacobi* dari f(x).

$$\dot{x} = Ax + Bu \quad (11)$$

Keterangan:

A : Matriks *Jacobi* A
 B : Matriks *Jacobi* B
 x : Jumlah *State Space*

Analisis Karakteristik Sistem Orde Dua

$$G(s) = \frac{K \omega_n^2}{s^2 + 2\xi \omega_n s + \omega_n^2} \quad (12)$$

Keterangan :

K : Gain Overall
 ω_n : Frekuensi Alami
 ξ : Rasio Redaman
 s : Penunjuk persamaan Transformasi *Laplace*

Karakteristik respon transien sistem orde kedua terdiri dari :

Rise Time : Waktu untuk naik dari 5% ke 95% respon keadaan waktu tunak (s).
 Maksimal *Overshoot* : Nilai puncak output kurva respon diukur dari satuan (%).
Setting Time : Waktu mencapai setpoint (s).
Error Steady State : *Error* pada waktu tunak (%).

Kontroler PID

Kontroler proporsional integral dan derivatif adalah untuk mempercepat reaksi sebuah sistem dan memperkecil *error* atau meredam *overshoot*. Kontroler proporsional integral dan derivatif hubungannya adalah u(t) sebagai output dari kontroler dan e(t) adalah sinyal *error* (Ogata, 1985). Aksi kontrol proporsional integral derivatif tersebut didefinisikan sebagai berikut :

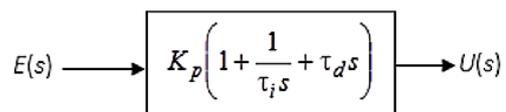
$$G(s) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (13)$$

$$G(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (14)$$

Keterangan :

G(s) : Fungsi Kontrol
 Kp : Konstanta *Proportional*
 Ti : Waktu *Integral*
 Td : Waktu *Derivatif*
 t : Waktu
 dt : Turunan Waktu
 $\int_0^t e(t) dt$: *Integral* sinyal *error* terhadap waktu

Diagram Blok Kontroler PID ditunjukkan pada Gambar 1.



Gambar 1. Diagram Blok Kontroler PID (Sumber : Ogata, 1985)

Motor Brushless DC

Motor brushless DC merupakan motor sinkron dengan magnet permanen yang menggunakan sensor posisi dan sebuah rangkaian elektronika (*inverter*) untuk

mengendalikan arus *armature* yang ada. Motor Brushless DC berbeda dengan motor DC biasa, motor brushless DC tidak menggunakan brush (sikat) sebagai media komutatifnya melainkan terkomunikasi secara elektris dan minim perawatan (Azzumar, 2012).

Electrical Speed Control (ESC)

ESC berfungsi untuk mengendalikan kecepatan motor brushless. Cara kerja secara cepat untuk menghidupkan atau mematikan pulsa ke motor, sehingga respon kendali motor ESC cepat. Fitur yang dimiliki oleh ESC adalah *brake* berfungsi untuk menghentikan motor secara spontan, *soft start* berfungsi sebagai pengatur lama waktu menyalakan sistem, *Microprocessor* berfungsi untuk mengatur agar dapat diprogram, dan *cut off* berfungsi sebagai pemotong arus jika baterai akan habis (Ghani Akbar Habibi, 2013)

Arduino UNO R3

Arduino Uno R3 merupakan *prototyping platform* sebuah paket berupa papan elektronik dan lingkungan pengembangan (*software*) yang memanfaatkan kemampuan mikrokontroler jenis Atmel seri Atmega 328 (Wardana, 2015). Mikrokontroler adalah suatu IC yang dapat diprogram dengan komputer. Program yang direkam digunakan agar rangkaian elektronik dapat membaca masukan, memproses dan menghasilkan keluaran sesuai yang diinginkan.

Sensor Ultrasonic HC-SR04

Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek tertentu di depannya, frekuensi kerjanya pada daerah di atas gelombang suara dari 20 KHz hingga 2 MHz. Sensor ultrasonik terdiri dari dua unit, yaitu unit pemancar dan unit penerima (Arief, 2011).

METODE PENELITIAN

Pendekatan Penelitian

Pendekatan penelitian pada penelitian ini adalah penelitian kuantitatif. Pada penelitian ini menggunakan LabVIEW 2014 sebagai menampilkan data secara *real-time*. Desain kontroler yang digunakan adalah kontroler *Proporsional-Integral-Derivatif* (PID) dengan pendekatan orde 2. Model kontroler ini kemudian akan diterapkan pada Instrumen kontrol menggunakan Arduino Uno R3 sebagai mengatur gerakan vertikal dari *towercopter* dan *HC-SR04* sebagai sensor ketinggian.

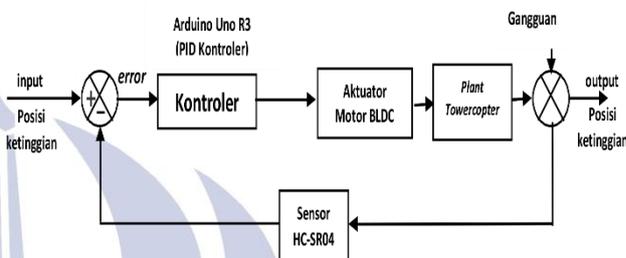
Instrumen Pengumpulan Data

Instrumen pengumpulan data yang digunakan dalam penelitian ini adalah berupa komputer (PC) ataupun

laptop serta perangkat lunak *LabVIEW* 2014 yang digunakan untuk menganalisis data keluaran sistem. Pengumpulan data kuantitatif diperoleh dari *LabVIEW* 2014 yang digunakan untuk menampilkan respon simulasi sistem pada kontrol ketinggian *Towercopter*.

Desain Sistem

Metode kontrol yang digunakan menggunakan PID dengan Berikut desain diagram blok sistem kendali ketinggian *Towercopter* yang ditunjukkan pada Gambar 2.



Gambar 2. Diagram Blok Sistem

Nilai masukan (*setpoint*) sistem berupa tegangan. Nilai dari input tersebut akan diproses oleh Arduino Uno R3 yang didalamnya sudah terdapat kontroler *Proporsional-integral-derivatif* yang nantinya akan membantu proses gerakan vertikal pada *towercopter*. Hasil dari keluaran kontroler berupa tegangan yang akan masuk ke ESC. ESC akan mengatur kecepatan putar pada motor BLDC. Kemudian sensor *ultrasonic* akan bekerja untuk mengukur nilai dari ketinggian *towercopter* setelah menjadikan nilai pembacaan sebagai nilai umpan balik. Ketika sistem memiliki selisih $\neq 0$, maka blok dari PID akan merevisi nilai *error* hingga mendekati 0 dan menghasilkan nilai K_p , K_i dan K_d .

Perancangan Kontroler Sistem

Untuk kontroler PID dilakukan dengan menggunakan pendekatan karakteristik respon sistem orde 2 yang memiliki parameter K , ζ dan ω_n . Metode ini dilakukan dengan cara memberikan *setpoint* ketinggian yang bervariasi.

Kemudian jika sudah mendapatkan nilai parameter K , ζ dan ω_n sehingga mendapat nilai K_p , K_i dan K_d untuk menggerakkan sistem ini maka sistem ini akan diuji dengan menginput nilai K_p , K_i dan K_d . Jika responnya baik dengan *overshoot* kecil maka kontroler PID sudah sesuai tetapi jika belum maka akan dilakukan pengujian ulang.

HASIL DAN PEMBAHASAN

Hasil Perancangan Plant

Untuk pemodelan pada *Towercopter* harus diketahui spesifikasi dari *hardware*.

Rancang Sistem Pengaturan Gerakan Vertikal Pada *Towercopter* Menggunakan Kontrol *Proportional Integral Derivatif* (PID)

Berikut adalah rancang Bangun *plant Towercopter* ditunjukkan pada Gambar 3.



Gambar 3. Rancang Bangun *Plant Towercopter*

Berikut adalah Tabel 1. Spesifikasi dari *Plant Towercopter* yang telah dirancang,

Tabel 1. Spesifikasi *Plant Towercopter*

No	Nama	Spesifikasi
1	Tinggi Jalur	±30 cm
2	Motor Brushles Emax cf2812	±1500Kv
3	Propeller	6x 5 inc
4	Sensor Jarak	HC-Sr04
5	LCD	16 x 2
6	Instrumen Kontrol	Raspberry Pi 3B
7	Driver Motor	ESC 30A
8	Power Supply	12 V, 10A

Perancangan Kontroler PID

Pada Penelitian ini, kontroler yang digunakan untuk mengendalikan posisi ketinggian *Towercopter* menggunakan Kontroler PID.

1. Pemodelan Matematis *Plant Towercopter*

Plant yang sudah dibuat, kemudian diukur untuk mengetahui konstanta-konstanta yang belum diketahui yakni, Massa Frame ($M_{fr} + \rho_c$) sebesar ± 190gr, Koeffisien Gesek sebesar 0.12, dan *Trust Force Koeffisien* (K) sebesar 0.8113. Konstanta yang telah didapat disubstitusikan pada model *State-Space* sebagai berikut,

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g(M_{fr} + \rho_c)}{M_{fr} + \rho_c} & -\frac{\mu_s}{M_{fr} + \rho_c} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K}{M_{fr} + \rho_c} \end{bmatrix} u \quad (15)$$

$$y = [1 \ 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u \quad (16)$$

Sehingga didapatkan Matriks A, B, C, D sebagai berikut,

$$A = \begin{bmatrix} 0 & 1 \\ -1.2791 & -2.2619 \end{bmatrix} \quad (17)$$

$$B = \begin{bmatrix} 0 \\ 0.8113 \end{bmatrix} \quad (18)$$

$$C = [1 \ 0] \quad (19)$$

$$D = 0 \quad (20)$$

Setelah didapatkan Matriks A, B, C, D, maka selanjutnya mengubahnya menjadi persamaan *Transfer Function* menggunakan persamaan 22.

$$G(s) = C(sI - A)^{-1}B + D \quad (21)$$

Keterangan:

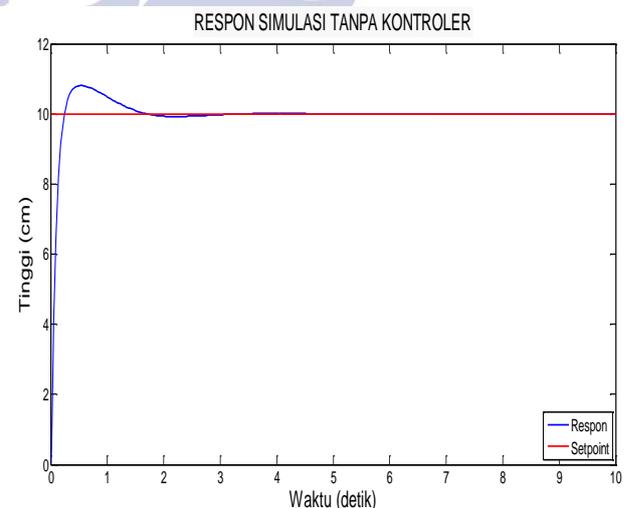
$G(s)$: Fungsi alih pemodelan sistem

I : Matriks Identitas

Sehingga didapatkan *Transfer Function* sebagai berikut,

$$G(s) = \frac{0.8113}{1.279s^2 + 1.382s + 0.7818}$$

Dari model *Transfer Function* tersebut, kemudian disimulasikan pada software MATLAB untuk diketahui respon sistem tanpa kontroler. Berikut adalah hasil dari respon simulasi tanpa kontroler ditunjukkan pada Gambar 4.

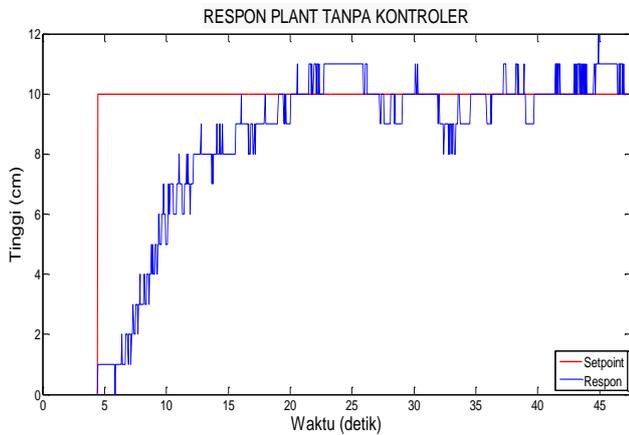


Gambar 4. Respon Simulasi Tanpa Kontroler

Pada Gambar 4. menunjukkan hasil dari respon simulasi tanpa kontroler untuk mendekati *setpoint* masih kurang bagus dikarenakan memiliki maksimal *overshoot* sebesar 45.45%. Sehingga perlu adanya kontroler PID agar memperbaiki respon lebih stabil dan mengurangi nilai *overshoot*.

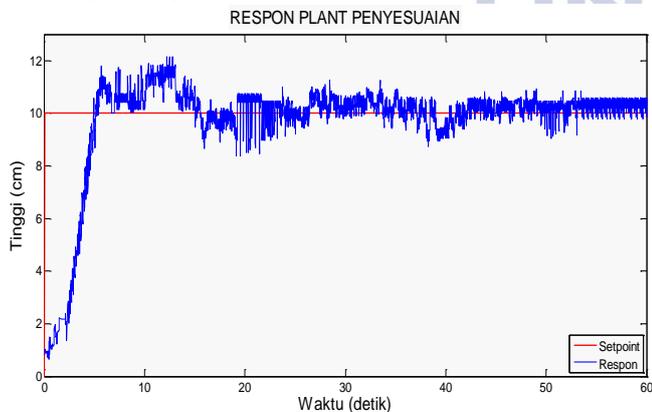
2. Pengujian Real Plant

Transfer Function yang telah didapat dari penurunan model matematis *plant*, kemudian dianalisa menggunakan pendekatan sistem orde 2 pada persamaan 21. Sehingga didapatkan $K_p = 1.6618$, $K_i = 1.202$ dan $K_d = 0.93$. Berikut merupakan respon *real plant Towercopter* yang ditunjukkan pada Gambar 5.



Gambar 5. Respon *Real Plant* Tanpa Kontroler

Hasil respon *real plant* menunjukkan bahwa respon dapat mengikuti nilai setpoint yang diberikan. Dengan *Error Steady-State* sebesar 1.86 % dan respon mulai bergerak naik pada 5.075 detik. Respon dinilai kurang baik karena masih mempunyai Maksimal *Overshoot* sebesar 25.06%. Oleh karena itu, parameter K_p , K_i dan K_d perlu untuk disesuaikan ulang dengan cara mengurangi atau menambah nilai K_p , K_i dan K_d dengan tujuan untuk memperoleh respon sistem dengan Maksimal *Overshoot* <20%. Sehingga didapatkan parameter K_p dan K_i akhir adalah $K_p = 1.055$, $K_i = 0.60$ dan $K_d = 0.025$. Berikut merupakan respon *real plant Towercopter* menggunakan K_p dan K_i yang baru yang ditunjukkan pada Gambar 6.



Gambar 6. Respon *Real Plant* Menggunakan K_p , K_i dan K_d Hasil Penyesuaian

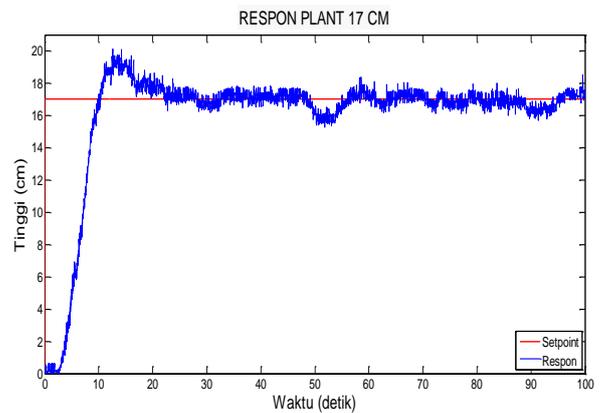
Pada Gambar 6 didapatkan hasil respon *real plant* dengan *Error Steady-State* sebesar 1.67%, respon mulai

bergerak naik pada 3.35 detik dengan Maksimal *Overshoot* sebesar 23.19 %. Respon yang dihasilkan sudah sesuai yang diinginkan sehingga parameter K_p dan K_i yang baru ini, kemudian diimplementasikan untuk pengujian *plant* dengan *setpoint* berbeda.

3. Hasil Pengujian Respon Dinamik

Setelah perancangan kontroler telah selesai selanjutnya melakukan pengujian sistem dari *Plant Towercopter*. Pengujian menggunakan kendali PID dengan *setpoint* yang berbeda – beda. Berikut adalah respon *real plant* dari beberapa kali percobaan :

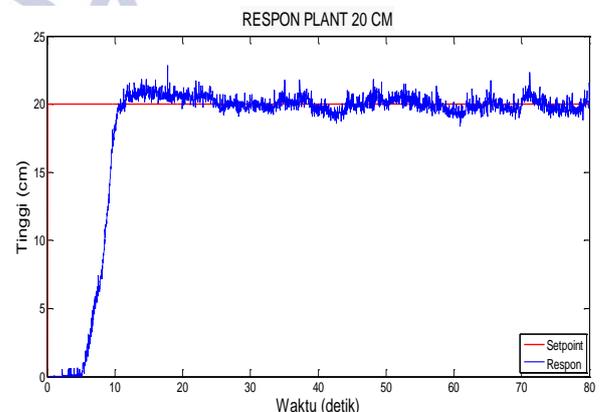
a. Setpoint 17 cm



Gambar 7. Respon *Real Plant* Setpoint 17 cm

Pada Gambar 7. didapatkan hasil respon *plant* menunjukkan bahwa *error steady state setpoint* 17 cm yaitu sebesar 0.28 % dan maksimal *overshoot* sebesar 11.50 %. Waktu naik untuk mencapai *setpoint* yaitu sebesar 10.75 s sedangkan waktu yang dibutuhkan untuk mencapai *steady state* sebesar 22.44 s.

b. Setpoint 20 cm



Gambar 8. Respon *Real Plant* Setpoint 20 cm

Pada Gambar 8 didapatkan hasil respon *plant* menunjukkan bahwa *error steady state setpoint* 20 cm yaitu sebesar 0.11 % dan maksimal *overshoot* sebesar

Rancang Sistem Pengaturan Gerakan Vertikal Pada *Towercopter* Menggunakan Kontrol *Proportional Integral Derivatif* (PID)

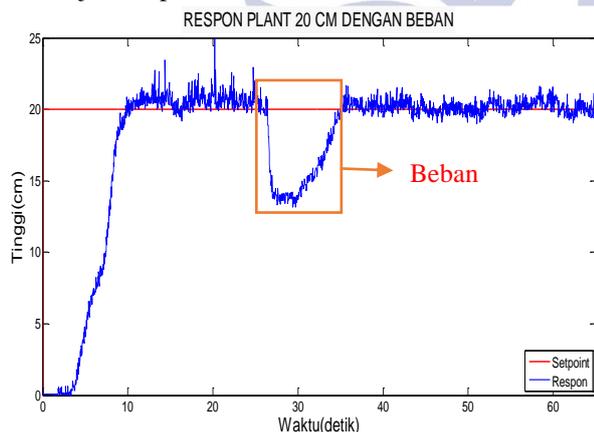
4.89 %. Waktu naik untuk mencapai *setpoint* yaitu sebesar 10.05 s sedangkan waktu yang dibutuhkan untuk mencapai *steady state* sebesar 18.45 s. Berikut merupakan dari hasil respon dinamik menggunakan pengendali PID dari beberapa kali pengujian dengan tanpa beban ditunjukkan pada Tabel 2.

Tabel 2. Hasil Respon Dinamik Menggunakan Pengendali PID dari Beberapa Kali Pengujian Tanpa Beban

Parameter	Setpoint 17 cm	Setpoint 20 cm
<i>Error Steady State</i>	0.28%	0.11%
<i>Rise Time</i>	10.75s	10.05s
<i>Time Delay</i>	7.25s	8.40s
<i>Setting Time</i>	22.44s	18.45s
Maksimal <i>Overshoot</i>	11.50%	0.11%

4. Pengujian Dengan Beban

Berikut ini adalah hasil respon real *Plant* *Setpoint* 20 cm dengan beban seberat 2.32 gr dimana yang ditandai dengan kotak merah adalah saat diberikan beban *towercopter* bergerak turun kemudian naik kembali sampai mencapai *steady state*, dapat ditunjukkan pada Gambar 9 berikut :



Gambar 9. Respon *Real Plant* menggunakan Kontroler PID Dengan Beban

Hasil respon *plant* menunjukkan pada *setpoint* 20 cm mengalami perubahan posisi setelah diberikan beban kemudian kembali mencapai *setpoint* dengan waktu naik yakni 33.89 s dan waktu yang dibutuhkan untuk mencapai *steady state* adalah 40.03 s. *Error Steady state* sebesar 1.62 % dan maksimal *overshoot* sebesar 16.89 %. Berikut merupakan hasil respon sistem dinamik dari *plant* dengan *setpoint* 20 cm menggunakan beban ditunjukkan pada Tabel 3.

Tabel 3. Hasil Respon Sistem Dinamik dari *Plant* Dengan *Setpoint* 20 cm Menggunakan Beban

Parameter	Setpoint 20 cm
<i>Error Steady State</i>	1.62%
<i>Rise Time</i>	33.89s
<i>Time Delay</i>	7.89s
<i>Setting Time</i>	40.03s
Maksimal <i>Overshoot</i>	16.89%

PENUTUP

Simpulan

Dalam merancang sistem pengendalian posisi ketinggian pada *towercopter* menggunakan *Proportional-Integral-Derivatif* dengan pendekatan orde 2 ada beberapa tahapan yaitu dengan pemodelan matematis *Plant* untuk mengetahui *Transfer Function* dari *towercopter*. Kemudian, merancang kontroler PID untuk mendapatkan parameter K_p , K_i dan K_d sebagai acuan. Parameter yang didapatkan adalah $K_p = 1.055$, $K_i = 0.60$ dan $K_d = 0.025$. Hasil respon sistem dinamik pengendali posisi ketinggian pada *towercopter* setelah melakukan pengujian 18 kali tanpa menggunakan beban, respon yang mendekati *setpoint* yakni *setpoint* 20 cm. Dengan nilai *Rise Time* = 10.05s, *Setting Time* = 18.45s, Maksimal *Overshoot* = 4.89%, dan *Error Steady State* = 0.11%. Sedangkan, *plant* diuji menggunakan beban seberat 2.32 gr menghasilkan nilai *Rise Time* = 33.89s, *Setting Time* = 40.03s, Maksimal *Overshoot* = 16.89%, dan *Error steady state* = 1.67%.

Saran

Dapat dikembangkan menggunakan sensor yang lebih presisi untuk mengurangi error pembacaan pada sensor *ultrasoni*. Dapat dikembangkan menggunakan kontroler PID dengan metode yang lain guna mengurangi *error steady state* dan *overshoot* yang ada. Dapat dikembangkan dengan jumlah motor yang berbeda yaitu 2, 3 atau 4 seperti halnya *quadcopter* ataupun *octacopter*. Serta dapat menggunakan *microcontroller* yang lainnya seperti *Raspberry*, *Arduino AtMega* dan sebagainya.

DAFTAR PUSTAKA

- Arief, U.M. 2011. Pengujian sensor ultrasonik ping untuk pengukuran level ketinggian dan volume air. *Jurnal Ilmiah Elektrical Engineering UNHAS*
- Azzumar, Muhammad. 2012. "Pemodelan dan Desain Kendali Sistem Aktuator Kendali Sirip Berbasis Brushless DC Motor". Depok: Universitas Indonesia.

- Benitez, David Sanchez, dkk.2012. "Vertical rotor for the implementation of control laws". *Department Of Computer Architecture And Automatics, Universidad Complutense De Madrid*.
- Habibie, Ghani Akbar. 2013. "Perancangan Dan Analisis Otomasi Sistem Kendali Quadcopter Melalui Koordinat Dengan Global Positioning System Tracker". Bandung : Institut Teknologi TELKOM.
- Moon, Jae-Sung, dkk. 2018. "A portable single-rotor-powered spherical unmanned aerial vehicle (UAV) with an easy to assemble and flexible structure". *Journal of Mechanical Science and Technology*. 2289-2298.
- Ogata, Katsuhiko. 1985. *Teknik Kontrol Automatik jilid 1*. Terjemahan Edi Laksono. Jakarta: Erlangga.
- Wardana, I Nyoman Kusuma, dkk. 2015. Penerapan mikrokontroler sebagai sistem kendali perangkat listrik berbasis arduino. Bali : STIKOM.
- Zufar, Rifqi Nabila. 2019. "Rancang Bangun Sistem Kendali Posisi Ketinggian Pada Towercopter Berbasis *Proportional Integral* Menggunakan *Tuning Artificial Neural Network*". (Dalam proses skripsi). Surabaya. Universitas Negeri Surabaya
- Priambodo, Ardy Seto, dkk.2012. "Perancangan dan Implementasi Sistem Kendali PID untuk Pengendalian Gerakan Hover pada UAV Quadcopter". *Jurnal Teknik POMITS*. Vol. 1 No. 1, 1-5.

