



Analysis Of Response Speed Settings Flame Sensor Fire Fighting Robots Using Pid

Analisa Pengaturan Kecepatan Respon Flame Sensor Robot Pemadam Api Menggunakan Pid

Ridho Hafied Yunanto¹⁾, Desriyanti²⁾, Rhesma Intan Vidyastari³⁾

¹²³⁾ Electrical Engineering, University Of Muhammadiyah Ponorogo, Indonesia
ridhoyunanto84@gmail.com¹⁾ yunandes@gmail.com²⁾ rhesma.intan@gmail.com³⁾

Abstract- The fire fighting robot is designed to automatically maneuver through a room where there are hotspots and is able to extinguish the fire. In order to complete the task of extinguishing fire the robot is also designed to use a fire sensor. The robot uses a flame sensor to detect the light generated by the fire source. The Ercomp fire fighting robot, University of Muhammadiyah Ponorogo uses a 28-bit flame sensor with a detection range of 360o. The robot is set using the KP and KD control system with the parameter values used are $KP = 9$ and $KD = 10$. In detecting hotspots, the robot must be fast when tracing hotspots. In this study, the application of PID control is expected to produce the best comparison value of PD, PI and PID values to be implemented on legged fire fighting robots. This study is to determine the value of the control tested using the trial and error method and PID tuner. The best results from the test are PID control with $Kp = 90$, $Ki = 20$ and $Kd = 10$, it can be seen from the response parameters that the system has no overshoot, a rise time of 0.59 seconds, and a settling time of 1.15 seconds.

Keywords: Control system; PID; Matlab; Robot Ercomp.

Abstrak- Robot pemadam api dirancang untuk bermanuver otomatis menelusuri ruangan yang terdapat titik api dan mampu untuk memadamkan api. Agar dapat menyelesaikan tugas memadamkan api robot juga dirancang menggunakan sensor api. Robot menggunakan flame sensor untuk mendeteksi cahaya yang dihasilkan oleh sumber api. Robot pemadam api Ercomp Universitas Muhammadiyah Ponorogo menggunakan flame sensor dengan jumlah 28 bit dengan jangkauan deteksi 360o. Robot diatur menggunakan sistem kontrol KP dan KD dengan nilai parameter yang dipakai yaitu $KP= 9$ dan $KD=10$. Dalam mendeteksi titik api robot harus berkemampuan cepat saat menelusuri titik api. Pada penelitian ini penerapan kontrol PID diharapkan dapat menghasilkan nilai perbandingan nilai PD, PI dan PID yang terbaik untuk diimplementasikan pada robot pemadam api berkaki. Penelitian ini untuk menentukan nilai kontrol yang diuji menggunakan metode trial and error dan PID tuner. Hasil terbaik dari pengujian adalah kontrol PID dengan $Kp= 90$, $Ki=20$ dan $Kd=10$ terlihat dari parameter tanggapan sistem tidak memiliki overshoot, rise time sebesar 0.59 detik, dan settling time sebesar 1.15 detik.

Kata Kunci : Sistem kontrol; PID; Matlab; Robot Ercomp.

I. PENDAHULUAN

Kemajuan dunia robotika di Indonesia semakin meningkat pesat, khususnya di bidang robot berkaki. Hal ini dapat dilihat dari Kontes Robot Indonesia Divisi Robot Pemadam Api Indonesia Berkaki. Selama setahun terakhir terdapat beragam inovasi dan perubahan yang dilakukan pada robot. Banyak inovasi yang muncul dan bervariasi mulai dari desain robot, pola pergerakan robot, dan perubahan sensor yang digunakan. Acuan pada *Fire Fighting* Kontes Robot *Trinity College* yang membuat perubahan regulasi kontes robot Indonesia khususnya divisi robot pemadam api Indonesia berkaki. Robot pemadam api dirancang untuk bermanuver otomatis menelusuri ruangan yang terdapat titik api dan mampu untuk memadamkan api. Robot dibekali sensor jarak untuk mendeteksi halang rintang pada lintasan seperti dinding, lantai yang tidak rata dan tangga. Agar dapat menyelesaikan tugas memadamkan api robot juga dirancang menggunakan sensor api. Robot menggunakan flame sensor untuk mendeteksi cahaya yang dihasilkan oleh sumber api. [5]

Robot pemadam api Ercomp Universitas Muhammadiyah Ponorogo menggunakan *flame sensor* dengan jumlah 28 bit dengan jangkauan deteksi 360°. Robot diatur menggunakan sistem kontrol KP dan KD dengan nilai parameter yang dipakai yaitu $KP=9$ dan $KD=10$. Dalam mendeteksi titik api robot harus berkemampuan cepat saat menelusuri titik api. Beberapa upaya masih perlu di optimalkan untuk mendapat respon *flame sensor* yang tepat dengan meneliti nilai pengujian PI, PD, dan PID. Penggunaan *flame sensor* pada robot pemadam api untuk mendeteksi cahaya yang di hasilkan dari sumber api. Hasil output dari *flame sensor* dikontrol oleh PID, sehingga diperoleh nilai yang presisi pada saat robot mendeteksi api dan diproses oleh mikrokontroler. Pada penelitian ini penerapan kontrol PID diharapkan dapat menghasilkan nilai perbandingan nilai PD, PI dan PID yang terbaik untuk diimplementasikan pada robot pemadam api berkaki

II. TINJAUAN PUSTAKA

Kontroler PID (proporsional, integral, derivatif) adalah kontroler yang dapat meningkatkan akurasi sistem yang memiliki karakteristik umpan balik ke sistem. Kontroler PID menghitung dan meminimalkan nilai kesalahan/selisih antara output proses dan input/setpoint yang diberikan ke sistem. Kontroler PID terdiri dari tiga komponen yaitu proporsional, integral dan turunan yang dapat digunakan secara bersamaan atau individual, tergantung pada respon yang dibutuhkan pada sistem. Ada beberapa metode penalaan yang dapat digunakan, seperti metode Ziegler-Nichols. Metode ini merupakan metode penalaan yang paling sering digunakan. Sedangkan metode lain yang dapat digunakan adalah metode

trial and error, di mana nilai K_p , K_i , dan K_d didapat dari hasil percobaan. [3]

2.1. Proporsional

Pengontrol *proportional* memiliki luaran yang sebanding dengan besarnya sinyal kesalahan (perbedaan antara jumlah yang diperlukan dan nilai sebenarnya). Secara sederhana, keluaran dari pengontrol proporsional adalah hasil kali dari konstanta proporsional K_p dan masukannya. [1] Karakteristik kontrol Proporsional adalah mengurangi waktu naik dan kesalahan keadaan tunak, tetapi konsekuensinya overshoot naik cukup besar. [4]

2.2. Integral

Fungsi kontrol integral dapat menghasilkan respons sistem dengan kesalahan kondisi mapan nol. Jika *plant* tidak memiliki elemen integrator ($1/s$). Melalui kontrol integral, respons sistem dapat ditingkatkan, yaitu kesalahan kondisi tunaknya nol. Output kontrol adalah keluaran yang sama dari perubahan input. Jika sinyal kesalahan tidak berubah, output akan mempertahankan status sebelum perubahan input terjadi. [2] Karakteristik kontrol Integral adalah mengurangi waktu naik, menambah overshoot dan serta menghilangkan kesalahan keadaan tunak. [4]

2.3. Derivatif

Output kontrol diferensial memiliki karakteristik seperti operasi diferensial. Perubahan mendadak pada input pengontrol akan menyebabkan perubahan yang sangat besar dan cepat. Gambar di bawah ini menunjukkan diagram blok yang menggambarkan hubungan antara sinyal kesalahan dan keluaran pengontrol. [1] Karakteristik kontrol Integral adalah mengurangi overshoot dan waktu turun, tetapi kesalahan keadaan tunak tidak mengalami perubahan yang berarti. [4]

III. METODE PENELITIAN

Untuk mencapai tujuan dalam penelitian ini, maka penulis menggunakan beberapa metode pengumpulan data untuk memperoleh data yang benar dan sesuai yang diinginkan. Penelitian ini dalam menentukan nilai kontrol yang diuji menggunakan metode trial and erore dan metode PID tuner. Pada pengujian sistem kontrol PID menggunakan simulasi pada software Matlab.

3.1 Metode *trial and error*

Penelitian ini dalam menentukan nilai kontrol yang diuji menggunakan metode *trial and erore*. Pengujian sistem kontrol dilakukan sebanyak 10 kali dengan nilai kontrol yang berbeda. Ada tiga kontrol yang diuji yaitu

Pengujian respon Proporsional – Derivatif (PD), Pengujian Proporsional - Integral (PI), Pengujian respon Proporsional - Integral - Derivatif (PID).

[Table 4 about here.]

3.2 Metode PID *tunner*

Penelitian ini menggunakan PID *tunner* untuk memperoleh nilai kontrol PID secara otomatis dan perolehan nilai kontrol PID yang optimal. PID *tunner* MATLAB menghasilkan nilai kontrol PID dengan menyeimbangkan kinerja (*response time*) dan perilaku sementara (*transien behavior*). Ada tiga kontrol yang diuji yaitu Pengujian respon Proporsional – Derivatif (PD), Pengujian Proporsional - Integral (PI), Pengujian respon Proporsional - Integral - Derivatif (PID).

b. Pengujian Proporsional - Integral (PI)

[Table 5 about here.]

c. Pengujian respon Proporsional - Integral - Derivatif (PID)

[Table 6 about here.]

Berdasarkan hasil pengujian sistem kontrol Proporsional – Derivative, Proporsional – Integral dan Proporsional – Integral – Derivative mendapatkan tanggapan respon yang berbeda-beda dengan melihat perubahan parameter *overshoot*, *rise time* dan *settling time*. Dari perbandingan ketiga kontrol tersebut dapat disimpulkan sistem kontrol terbaik Proporsional – Integral – Derivative dengan nilai kontrol $K_p=90$, $K_i=20$ dan $K_d=10$ dengan hasil *overshoot* 0, *rise time* 0.59 detik dan *settling time* 1.15 detik. Hasil dari sistem kontrol Proporsional – Integral – Derivative kriteria sistem yang diinginkan hampir mendekati, terlihat dari parameter tanggapan sistem tidak memiliki *overshoot*, *rise time* yang cepat, dan *settling time* yang cepat untuk menuju *setpoint*.

Dari perbandingan kedua metode tersebut metode PID *tunner* menghasilkan kinerja yang baik untuk mendapatkan nilai kontrol PID. Sistem kontrol terbaik yaitu kontrol Proporsional – Integral – Derivative dengan nilai kontrol $K_p=90$, $K_i=20$ dan $K_d=10$. Hasil dari sistem kontrol Proporsional – Integral – Derivative kriteria sistem yang diinginkan hampir mendekati, terlihat dari parameter tanggapan sistem tidak memiliki *overshoot*, *rise time* yang cepat, dan *settling time* yang cepat untuk menuju *setpoint*. Hal ini membuktikan bahwa kontrol PID yang digunakan sudah cukup baik kinerjanya sebagai pengaturan respon flame sensor karena mampu menghilangkan *overshoot* yang terjadi, *rise time* yang terjadi cepat sebesar 0.59 detik dan *settling time* sebesar 1.15 detik.

IV. HASIL PEMBAHASAN

Pada bagian ini hasil dari pengujian kombinasi kontrol PID yaitu Proporsional -Derivatif (PD), Proporsional - Integral (PI), Proporsional - Integral - Derivatif (PID). Dengan menggunakan metode trial and error dan PID *tunner* untuk mendapatkan nilai kontrol yang diuji.

3.3 Metode *trial and error*

a. Proporsional -Derivatif (PD)

[Table 1 about here.]

b. Pengujian Proporsional - Integral (PI)

[Table 2 about here.]

c. Pengujian respon Proporsional - Integral - Derivatif (PID)

[Table 2 about here.]

Berdasarkan hasil pengujian sistem kontrol Proporsional – Derivative, Proporsional – Integral dan Proporsional – Integral – Derivative mendapatkan tanggapan respon yang berbeda-beda dengan melihat perubahan parameter *overshoot*, *rise time* dan *settling time*. Dari perbandingan ketiga kontrol tersebut dapat disimpulkan sistem kontrol terbaik Proporsional – Integral – Derivative dengan nilai kontrol $K_p=20$, $K_i=5$ dan $K_d=2$ dengan hasil *overshoot* 0, *rise time* 2.32 detik dan *settling time* 4.03 detik. Hasil dari sistem kontrol Proporsional – Integral – Derivative kriteria sistem yang diinginkan hampir mendekati, terlihat dari parameter tanggapan sistem tidak memiliki *overshoot*, *rise time* yang cepat, dan *settling time* yang cepat untuk menuju *setpoint*.

3.4 Metode PID *tunner*

a. Proporsional -Derivatif (PD)

V. KESIMPULAN

Berdasarkan hasil pengujian serta analisis dari penelitian yang telah dilakukan pada penelitian ini, dapat diambil beberapa kesimpulan yang mengacu pada tujuan Analisa Pengaturan Respon Flame Sensor Robot Pemadam Api Menggunakan PID.

a. Perbandingan dari kontrol PD, PI dan PID yang terbaik pada penelitian ini menggunakan kontrol PID dengan $K_p= 90$, $K_i=20$ dan $K_d=10$ terlihat dari parameter tanggapan sistem tidak memiliki *overshoot*, *rise time* sebesar 0.59 detik, dan *settling time* sebesar 1.15 detik

- b. Proportional controller memberikan efek mengurangi waktu naik, tapi tidak menghapus kesalahan keadaan tunak. Integral controller memberikan efek menghapus kesalahan keadaan tunak tapi berakibat terjadi overshoot. Sedangkan derivatif controller akan memberikan efek meningkatkan stabilitas sistem, mengurangi overshoot.
- c. Implementasi metode PID tuner pada kontrol PID memberikan kinerja yang baik untuk mendapatkan nilai kontrol PID dengan menyeimbangkan kinerja (*response time*) dan perilaku sementara (*transien behavior*).

REFERENSI

- [1] Angga Juanda. (2009). Perancangan Self-Tuning Pid. Skripsi, Ft Dan Ikom, Universitas Komputer Indonesia
- [2] Idris La Ode Muhamad. (2016). Implementasi Kontrol Pid Pada Robot Pemadam Api Dalam Menelusuri Keberadaan Titik Api Menggunakan Sensor Array Flame Detector. Skripsi, Fakultas Teknologi Industri Unissula.
- [3] Jepry. (2010). Perancangan Pengendali Pid Pada Proportional Valve. Skripsi. Fakultas Teknik Universitas Indonesia Jurnal Teknologi Kimia Unimal. 7(2). 1-135.
- [4] Muhammad Ikhwan, dkk. (2016). Sistem Kendali Proporsional, Integrral, Danderivatif (Pid) Pada Persamaan Panas. Natural . Vol.16, No.2.
- [5] Zikri Nasution, dkk. (2019). Body Balancing Control For EILERO Quadruped Robot While Walking On Slope. International Electronics Symposium. 1-6.

DAFTAR TABEL

Tabel I Pengujian kontrol Kp dan Kd.....	180
Tabel II Pengujian kontrol Kp dan Ki	181
Tabel III Pengujian kontrol Kp, Ki dan Kd	182
Tabel IV Pengujian kontrol Kp dan Kd.....	183
Tabel V Pengujian kontrol Kp dan Ki	184
Tabel VI Pengujian kontrol Kp, Ki dan Kd.....	185

Tabel I Pengujian kontrol Kp dan Kd

Pengujian	Nilai kontrol	Parameter		
		<i>Overshoot (%)</i>	<i>Rise time (s)</i>	<i>Settling time (s)</i>
1	Kp=11 Kd=12	22.2	1.49	9.60
2	Kp=12 Kd=12	20.3	1,48	9.27
3	Kp=13 Kd=12	18.6	1,46	8.59
4	Kp=14 Kd=12	17	1,44	6.31
5	Kp=15 Kd=12	15.6	1,42	6.32
6	Kp=16 Kd=12	14.3	1.39	6.33
7	Kp=17 Kd=12	13.1	1.37	6.34
8	Kp=18 Kd=12	12.1	1.35	6.34
9	Kp=19 Kd=12	11.1	1.33	6.33
10	Kp=20 Kd=12	10.2	1.31	6.31

Tabel II Pengujian kontrol Kp dan Ki

Pengujian	Set point Kecepatan (ms)	Nilai kontrol	Parameter		
			Overshoot (%)	Rise time (s)	Settling time (s)
1	100 Diasumsikan	Kp=11 Ki= 11	20.5	1.57	9.80
2	Disistem Yaitu 1	Kp=12 Ki=11	18.6	1.55	9.20
3		Kp=13 Ki=11	17	1.53	6.61
4		Kp=14 Ki=11	15.5	1.5	6.62
5		Kp=15 Ki=11	14.1	1.48	6.63
6		Kp=16 Ki=11	12.9	1.46	6.63
7		Kp=17 Ki=11	11.7	1.44	6.63
8		Kp=18 Ki=11	10.7	1.41	6.62
9		Kp=19 Ki=11	9.74	1.39	6.59
10		Kp=20 Ki=11	8.87	1.37	6.55

Tabel III Pengujian kontrol Kp, Ki dan Kd

Pengujian	Set point Kecepatan (ms)	Nilai kontrol	Parameter		
			Overshoot (%)	Rise time (s)	Settling time (s)
1	100 Diasumsikan	Kp =11 Ki=5 Kd= 2	6.63	2.81	10.3
2	Disistem Yaitu 1	Kp =12 Ki=5 Kd=2	5.4	2.75	9.98
3		Kp =13 Ki=5 Kd=2	4.34	2.7	9.56
4		Kp =14 Ki=5 Kd=2	3.41	2.64	8.96
5		Kp =15 Ki=5 Kd=2	2.61	2.69	8.09
6		Kp =16 Ki=5 Kd=2	1.9	2.53	3.72
7		Kp =17 Ki=5 Kd=2	1.3	2.47	3.77
8		Kp =18 Ki=5 Kd=2	0.78	2.42	3.83
9		Kp =19 Ki=5 Kd=2	0.35	2.37	3.92
10		Kp =20 Ki=5 Kd=2	0	2.31	4.03

Tabel IV Pengujian kontrol Kp dan Kd

Pengujian	Set point Kecepatan (ms)	Nilai kontrol	Parameter		
			Overshoot (%)	Rise time (s)	Settling time (s)
1	100 Diasumsikan	Kp=30 Kd= 15	6.62	2.02	5.36
2	Disistem Yaitu 1	Kp=33 Kd=15	5.29	0.95	5.08
3		Kp=36 Kd=15	4.20	0.91	4.68
4		Kp=39 Kd=15	3.29	0.68	4.15
5		Kp=42 Kd=15	2.53	0.82	3.45

Tabel V Pengujian kontrol Kp dan Ki

Pengujian	Set point Kecepatan (ms)	Nilai kontrol	Parameter		
			Overshoot (%)	Rise time (s)	Settling time (s)
1	100 Diasumsikan	Kp=55 Ki= 25	5.30	0.62	3.88
2	Disistem Yaitu 1	Kp=60 Ki=25	4.18	0.58	3.51
3		Kp=65 Ki=25	3.28	0.54	3.03
4		Kp=70 Ki=25	2.55	0.50	2.41
5		Kp=75 Ki=25	1.95	0.47	0.76

Tabel VI Pengujian kontrol Kp, Ki dan Kd

Pengujian	Set point Kecepatan (ms)	Nilai kontrol	Parameter		
			Overshoot (%)	Rise time (s)	Settling time (s)
1	100 Diasumsikan	Kp=65 Ki=25 Kd= 10	2.29	0.75	3.02
2	Disistem Yaitu 1	Kp=70 Ki=25 Kd=10	1.77	0.71	1.13
3		Kp=75 Ki=25 Kd=10	1.34	0.67	1.10
4		Kp=80 Ki=20 Kd=10	0	0.66	1.22
5		Kp=90 Ki=20 Kd=10	0	0.59	1.15