

OTOMASI PID TUNING UNTUK OPTIMASI KONTROL QUADCOPTER MENGGUNAKAN METODE HARMONY SEARCH

Wimmy Hartawan

Prodi SI Ilmu Komputer, UPP, Rejang-Lebong

*e-mail: *wimmy_asvera@yahoo.com*

Abstract

The search system intelligent tuning value on the Quadcopter control method the topics the one of research that continues to be developed, starting from conventional control systems to intelligent control systems. The methods of control commonly today used include PID control. The automatic control system is handle to able several disturbances from internal and external factors. Internal factors can be caused from vibration than perfect less and actuator devices. External factors form the changes in the wind direction and speed, waves and wind current speed resulting is changes in the orientation on direction from the quadcopter. External factors that are disturbances of difficult to predict and relatively in change non-linear.

The HS (Harmony Search) algorithm is one of the optimization algorithms can be used decision maker system, which a metaheuristic method inspired by compositions of jazz. The Harmony Search algorithm has initialization step a harmony memory. This stage a advantage of being to able simultaneously find several combinations of points from the initial route that be obtained from the given initial route. In the Harmony Search method, it by analogy can be explained, when is improvising a musical instrument player, there are can be done three things that, namely: playing notes that are composition in the song already available, playing notes that have in harmony accordance with the song composition (pitch adjustment) and playing notes randomly.

The results of the PID tuning simulation research using Harmony Search found constants K_p , K_i and K_d with the smallest rise time and overshoot for all angles. At the corner it is found that K_p is worth 0.176, K_i is worth 0.008 and K_d is 0.098.

Keywords : PID, Quadcopter, Adaptive Control, Harmony Search, Metaheuristic Control.

Abstract

Sistem pencarian nilai tuning cerdas pada metode kendali Quadcopter menjadi salah satu topik penelitian yang terus dikembangkan, berawal dari sistem kendali konvensional hingga sistem kendali cerdas. Metode kendali yang umum digunakan saat ini antara lain kendali PID. Sistem kendali otomatis mampu untuk menangani beberapa gangguan dari faktor internal maupun eksternal. Faktor internal dapat disebabkan karena getaran dan perangkat aktuator yang kurang sempurna. Faktor eksternal berupa perubahan arah dan kecepatan angin, gelombang dan kecepatan arus angin sehingga mengakibatkan perubahan orientasi arah quadcopter. Faktor eksternal merupakan gangguan yang sulit untuk diprediksi dan relatif tidak linear perubahannya.

Algoritma HS (Harmony Search) adalah salah satu algoritma optimasi yang dapat digunakan untuk pengambilan keputusan yang merupakan sebuah metode metaheuristik yang terinspirasi dari komposisi musik jazz. Algoritma Harmony Search memiliki langkah inisialisasi memori harmoni. Tahap ini memiliki kelebihan yaitu mampu secara serempak menemukan beberapa susunan kombinasi titik dari rute awal yang mungkin didapatkan dari rute awal yang diberikan. Dalam metode Harmony Search dapat diperjelas dengan analogi, ketika seorang pemain instrumen musik sedang berimprovisasi ada tiga hal yang mungkin dilakukan yaitu: memainkan note yang telah tersedia dalam komposisi lagu, memainkan note yang mempunyai harmony yang sesuai dengan komposisi lagu (pitch adjustment) serta memainkan note secara acak (random).

Hasil penelitian simulasi penalaan PID menggunakan Harmony Search menemukan konstanta K_p , K_i dan K_d dengan rise time dan overshoot terkecil untuk semua sudut. Pada sudut ditemukan K_p bernilai 0.176, K_i bernilai 0.008 dan K_d bernilai 0,098.

Kata kunci : PID, Quadcopter, Adaptive Control, Harmony Search, Metaheuristic Control.

PENDAHULUAN

Quadrotor atau sering disebut juga quadcopter merupakan jenis Unmanned Aerial Vehicle (UAV) yang terbang dengan menggunakan empat motor dan empat buah propeller pada masing-masing lengannya[1]. Gerakan quadrotor dihasilkan dengan menggunakan gaya dorong dari perputaran tiap motor. Pergerakan tersebut diatur oleh sistem kontrol attitude. Untuk menghasilkan attitude yang baik, diperlukan aturan pengendali yang benar dan pemodelan sistem. Dua metode yang berbeda telah diketahui untuk mencapai dinamika quadcopter yakni persamaan Lagrange dan hukum II Newton[2]. Hasil pemodelan dinamika quadcopter secara langsung dipengaruhi oleh sinyal masukan, sehingga dibutuhkan aturan sistem kendali lebih lanjut agar dapat menghasilkan kondisi output yang sesuai dengan kondisi plant. Salah satu sistem kendali yang dapat dirancang dan diimplementasikan pada quadcopter adalah kendali PID[3].

Sistem pencarian nilai tala cerdas pada metode kendali quadcopter menjadi salah satu topik penelitian yang terus dikembangkan, berawal dari sistem kendali klasik hingga sistem kendali cerdas. Metode kendali yang umum digunakan saat ini antara lain kendali PID. Sistem kendali PID banyak digunakan untuk mengendalikan berbagai aktuator dalam skala industri maupun skala penelitian, karena karakteristik pengendalian yang dapat diatur sesuai kondisi yang diinginkan dengan mengatur ketiga konstanta pengendaliannya, dapat menghasilkan ramp keluaran yang cocok untuk pengendalian aktuator dan cocok untuk pengendalian dengan sistem kalang tertutup[4]. Kendali PID memenuhi spesifikasi perancangan seperti respon transisi, stabilitas dan reliabilitas. Kendali PID juga mempunyai karakteristik kokoh dan sederhana dalam implementasi[5].

Optimal secara gambaran umum dengan mencari nilai kesalahan seminim mungkin[6], pada quadcopter perilaku kesalahan dapat disebabkan oleh besarnya selisih nilai antara sudut kemiringan yang terbaca oleh sensor terhadap nilai inputan dari remote terhadap derajat yang sama,

misalkan derajat kemiringan sudut pitch yang terbaca oleh sensor dengan nilai input pada channel pitch dari remote. optimal juga dapat dinyatakan terhadap respon kesalahan. Sifat adaptif juga dapat dikatakan dengan penyesuaian pencarian nilai penala[7], yang bertujuan untuk mendukung proses optimasi kendali yang menyerupai pola pikir manusia yang dilibatkan dalam penalaan PID quadcopter.

Kondisi beban saat quadcopter berada diatas jelas beda jika dibandingkan dengan saat quadcopter berada dibawah, dimana kondisi beban yang berubah-ubah akan menyebabkan parameter plant berubah[8]. Perubahan beban tersebut menyebabkan adanya ketidak pastian parameter plant yang dapat menyebabkan perubahan spesifikasi respon yang cukup signifikan. Kendali PID klasik umumnya tidak mampu mengatasi masalah diatas karena dinamika sistem yang bervariasi, kecuali dengan cara dilakukan penalaan berkala[9], sehingga dibutuhkan suatu pengendali yang kokoh dan dapat diandalkan untuk menyelesaikan permasalahan diatas.

Algoritma HS (Harmony Search) adalah salah satu algoritma optimasi yang dapat digunakan untuk pengambilan keputusan yang merupakan sebuah metode metaheuristik . yang terinspirasi dari komposisi musik jazz. Algoritma Harmony Search memiliki langkah inialisasi memori harmoni. Tahap ini memiliki kelebihan yaitu mampu secara serempak menemukan beberapa susunan kombinasi titik dari rute awal yang mungkin didapatkan dari rute awal yang diberikan. Penelitian yang telah dilakukan tentang aplikasi Harmony Search untuk VRP [10]. Serta belum menjelaskan secara mendetail tentang model dan bentuk penyelesaian algoritma yang dibangun. Pada perkembangannya metode ini diadaptasi unuk menyelesaika permasalahan kontinyu. Analogi improvisasi dalam musik jazz, yang dipakai sebagai analogi.

METODOLOGI PENELITIAN

Analisis Pemodelan



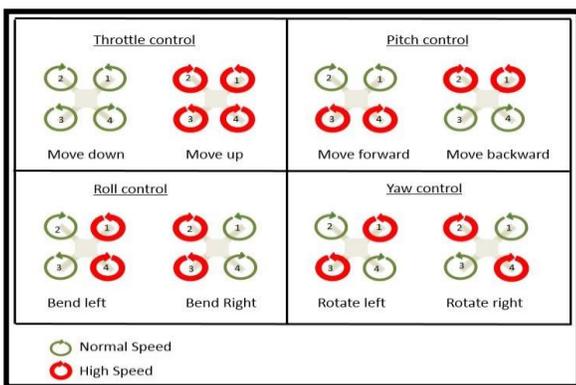
UAV (Unmanned Aerial Vehicle) adalah pesawat terbang tanpa awak yang dikendalikan secara jarak jauh melalui sebuah media. Salah satu jenis UAV adalah multicopter. Multicopter yang memiliki 4 buah baling dengan konfigurasi menyilang disebut quadcopter. Quadcopter juga sering disebut sebagai Quadrotor. Pada Gambar 1 dua pasang baling-baling (1, 3) dengan (2, 4) berputar secara berlawanan arah. Dengan memvariasikan kecepatan rotor, dapat mengubah gaya angkat dan menciptakan gerak. Dengan demikian, meningkatkan atau menurunkan kecepatan keempat baling-baling bersama-sama menghasilkan gerakan vertikal. Variasi kecepatan pada propeller (1, 2) dengan (3,4) menyebabkan perubahan sudut pitch quadcopter, dimana akan menghasilkan gerak maju dan mundur. Perubahan sudut roll terjadi dengan memvariasikan kecepatan baling-baling (1,4) dan (2,3) sehingga mengakibatkan quadcopter bergerak ke kanan dan kiri. Sudut yaw dipengaruhi oleh keseimbangan torsi aerodinamik antara kedua pasang baling-baling putaran searah jarum jam dan putaran berlawanan arah jarum jam sehingga jika salah satu dari 2 pasang baling-baling tersebut berputar lebih kencang maka akan membuat quadcopter berputar ke satu arah.

$$u_1 = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)$$

$$u_2 = b(\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2)$$

$$u_3 = b(-\omega_1^2 - \omega_2^2 + \omega_3^2 + \omega_4^2)$$

$$u_4 = d(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)$$

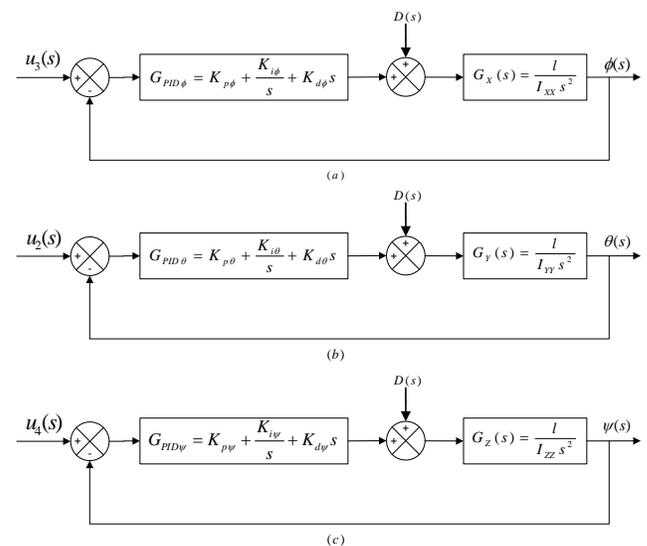


Gambar 1 Dinamika Gerak Quadcopter

Dengan u_1 pada persamaan(1), u_2 pada persamaan(2), u_3 pada persamaan(3) dan u_4 pada persamaan(4), masing-masing adalah input thrust, roll, pitch, yaw. Sedangkan ω_1 adalah kecepatan sudut motor 1, ω_2 adalah kecepatan sudut motor 2, ω_3 adalah kecepatan sudut motor 3, dan ω_4 adalah kecepatan sudut motor 4, b adalah thrust factor, dan d adalah drag factor.

Penentuan Fugsi Transfer

Diagram blok fungsi transfer domain Laplace sudut pitch, sudut roll, dan sudut yaw pada quadcopter ditunjukkan pada Gambar 2.



Gambar 2 Diagram blok kontrol pada simulasi PID Controller : (a) sudut roll, (b) sudut pitch, (c) sudut yaw

$G_{PID}(s)$ merupakan PID Controller domain Laplace, $G_x(s)$ merupakan fungsi transfer pada sudut roll, $G_y(z)$ merupakan fungsi transfer pada sudut pitch, dan $G_z(s)$ merupakan fungsi transfer pada sudut yaw. Ketiga blok diagram di atas dapat disederhanakan menggunakan aturan *unityfeedback*. Untuk sumbu x disederhanakan menjadi $G_{xuf}(s)$ yang dijabarkan dalam persamaan (5).



$$G_{xuf}(s) = \frac{K_{D\phi}ls^2 + K_{P\phi}ls + K_{i\phi}l}{I_{xx}s^2 + K_{D\phi}ls^2 + K_{P\phi}ls + K_{i\phi}l} \quad (5)$$

Untuk sumbu y disederhanakan menjadi $G_{yuf}(s)$ yang dijabarkan dalam persamaan (6).

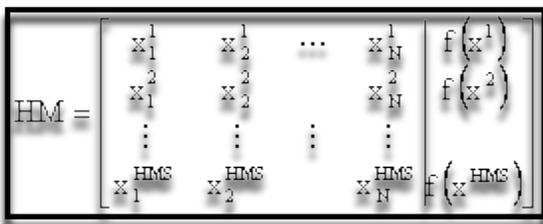
$$G_{yuf}(s) = \frac{K_{D\theta}ls^2 + K_{P\theta}ls + K_{i\theta}l}{I_{yy}s^2 + K_{D\theta}ls^2 + K_{P\theta}ls + K_{i\theta}l} \quad (6)$$

Untuk sumbu z disederhanakan menjadi $G_{zuf}(s)$ yang dijabarkan dalam persamaan (7).

$$G_{zuf}(s) = \frac{K_{D\psi}ls^2 + K_{P\psi}ls + K_{i\psi}l}{I_{zz}s^2 + K_{D\psi}ls^2 + K_{P\psi}ls + K_{i\psi}l} \quad (7)$$

Dari ketiga fungsi transfer unityfeedback di atas diharapkan kombinasi yang tepat antara Kp, Ki, Kd dapat meminimalkan nilai peak dan settling time.

Sesuai dengan namanya, parameter didalam HS menggunakan istilah-istilah yang terdapat didalam musik. Sebuah kandidat disebut dengan ‘harmony’ (bandingkan individu yang dipakai oleh Algoritma Genetik). Kumpulan dari kandidat penyelesaian disebut dengan harmony memory (seperti populasi dalam algoritma genetik). Didalam HS, nilai dari elemen kandidat penyelesaian yang baru dapat diambil dari salah satu harmoni memori (yang dipilih secara acak) dengan probabilitas tertentu. HS tersusun dari lima langkah yaitu: inialisasi parameter, inialisasi harmoni memori, improvisasi harmoni, memperbaharui harmoni memori dan menentukan kriteria menghentikan proses pencarian. Untuk memudahkan, asumsikan sebuah fungsi minimal dari sebuah permasalahan diskrit.



Gambar 3 Peta Harmony Memory.

Langkah terpenting dan sekaligus yang membedakan HS dengan metode lainya terletak pada langkah ketiga yaitu improvisasi harmoni.

$$x_j = \begin{cases} x_j \in \{x_j^1, x_j^2, \dots, x_j^s\} & \text{dengan prob HMCR} \\ x_j \in X_j & (1 - HMCR) \end{cases} \quad (8)$$

Setelah sebuah kandidat penyelesaian terbentuk, kandidat penyelesaian tersebut memiliki probabilitas untuk ‘diperbaiki’ lebih lanjut (pitch adjustment). Dengan probabilitas (7)

$HMCR \times PAR$, ($0 \leq PAR \leq 1$) sebuah nilai akan digeser ke nilai sebelahnya.

$$x_j = \begin{cases} x_j(k \pm l) & \text{dengan prob HMCR} \times PAR \\ x_j & HMCR(1 - HMCR) \end{cases} \quad (9)$$

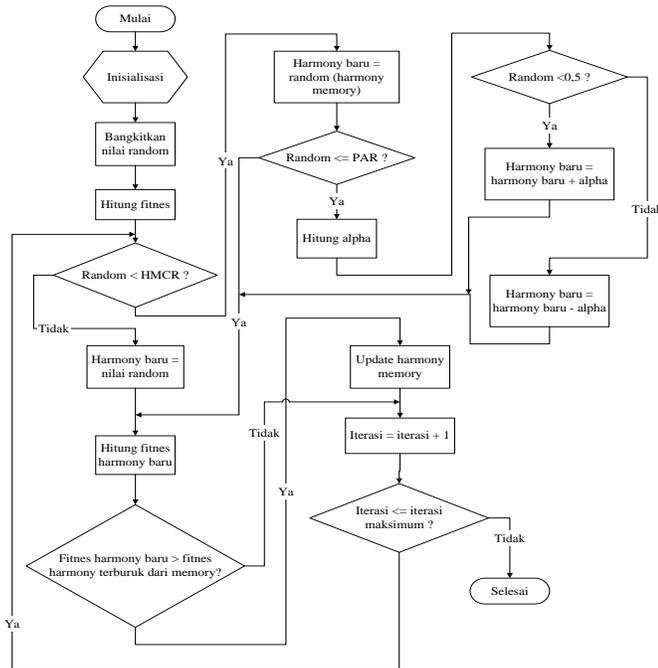
Dimana k adalah elemen ke k dari $[X]_j$ dan l menyatakan besarnya pergeseran (dalam variable diskrit menyatakan indek untuk nilai sebelahnya $l \in \{\dots, -2, -1, 1, 2, \dots\}$). Setelah proses pitch adjustment, kandidat penyelesaian ini kemudian dicek untuk menentukan bahwa tidak ada constraint yang dilanggar.

Proses penyelesaian akan dihentikan jika kriteria untuk berhenti terpenuhi. Kriteria berhenti ini bisa berupa jumlah maksimum literasi, nilai objektif mencapai nilai tertentu atau jika pencarian sudah mencapai titik stagnan (setelah sejumlah proses pencarian, nilai objektif dari kandidat terbaik tidak berubah).

Pada tahapan perancangan perangkat lunak yang perlu diketahui dengan menentukan nilai-nilai masukan untuk perancangan simulasi. Dimana nilai masukan dapat diperoleh dari nilai seperti fungsi transfer serta nilai minimum dan maksimum penalan setiap komponen kendali PID. Selain itu perlu juga ditentukan nilai objek yang dicari, dimana nilai objek berbeda-beda tiap komponen kendali PID. Misalnya



K_p untuk mencari nilai risetime, sedangkan nilai K_i mencari nilai settlingtime dan K_d mencari nilai overshoot terendah urutan proses pencarian di gambarkan dengan diagram alir pada Gambar 4.



Gambar 4 Diagram Alir Penalaan PID dengan Harmony Search.

PEMBAHASAN HASIL PENELITIAN

Implementasi

Nilai kesalahan dan delta error yang akan diakses adalah nilai kesalahan dari sudut yaw, pitch, roll yang didapatkan dari sensor IMU GY-521 pada program yang dapat memberikan nilai yaw, pitch, roll yang akan digunakan pada kendali PID yang diterapkan pada quadcopter, dengan nilai akhir yang akan diterapkan pada aktuator adalah nilai yang terdapat pada variabel PID. Variabel ini merupakan variabel yang memiliki nilai dari K_p , K_i , dan K_d yang merupakan konstanta-konstanta untuk kendali PID.



Pada implementasi penala PID dengan metode Harmony Search di eksekusi dengan cara simulasi Matlab urutan prosesnya, contoh program yang ditampilkan diambil proses pada sumbu pitch karena pada sumbu roll prosesnya sama, pada Gambar 5 ditunjukkan proses eksekusi pembaharuan harmony memori dan pada Gambar 6. kode program Matlab dalam melakukan perbandingan setiap strategi kontrol hasil pencarian harmony search.

```

1. harmony_terbaik_terjelek=max(nilai
   i_objektif);
2. if
   nilai_objektif_harmony_baru<harmony_
   terbaik_terjelek
3. index=find(harmony_terbaik_terjelek==nilai_objektif);
4. harmony_memory(index,:)=harmony_baru;
5. nilai_objektif(index,:)=nilai_objektif_harmony_baru;
6. end

```

Gambar 5 Kode Program Pembaharuan Harmony Memori

```

1. time=0:0.01:10;
2. GC1=pid(KP);
3. H1=series(GC1,GP);
4. T1=feedback(H1,1);
5. y1=step(T1,time);
6. GC2=pid(KP,0,KD);
7. H2=series(GC2,GP);
8. T2=feedback(H2,1);
9. y2=step(T2,time);
10. GC3=pid(KP,KI,KD);
11. H3=series(GC3,GP);
12. T3=feedback(H3,1);
13. y3=step(T3,time);
14. plot(time,y1,time,y2,time,y3)
15. legend('P','PD','PID')
16. title('grafik step respon','FontWeight','bold');

```

Gambar 6 Kode Program Matlab Metode Perbandingan Hasil

Pengujian Gangguan

Dari hasil yang didapatkan uji simulasi menggunakan harmony search, maka dilakukan perbandingan uji statis dengan menerapkan gangguan. Uji statis dilakukan

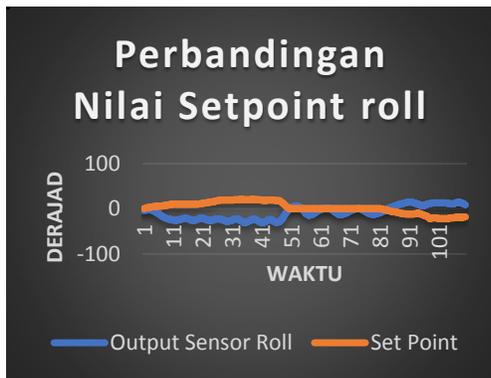
dengan cara menggantung quadcopter dan diberikan gangguan, Risetime adalah waktu yang dibutuhkan untuk mencapai 90% kondisi yang diinginkan dari kondisi 10% yang diinginkan. Overshoot adalah persentase kesalahan respon dari nilai yang diinginkan. Untuk menghitung nilai overshoot dapat menggunakan persamaan(10).

$$Overshoot(\%) = \frac{M_{pt} - fv}{fv} \times 100\% \quad (10)$$

Pada persamaan (10) M_{pt} adalah *peak value*, fv adalah nilai akhir yang diinginkan sehingga dapat ditunjukkan pada Gambar 7.



(a)



(b)

Gambar 7 Hasil Pengujian Alami Sudut Pitch (a), Sudut Roll

(b)

Pada Gambar 7(a) sudut pitch yang diinginkan yaitu 0°. Sistem diharapkan mampu kembali ke sudut pitch 0° dari gangguan sudut pitch sebesar 25°. 10% dari kondisi

yang diinginkan yaitu -22,5° dan 90% dari kondisi yang diinginkan yaitu -2,5o. Risetime pada respon sudut pitch sekitar 0,500 detik. Overshoot dari respon sudut pitch dapat dihitung dengan persamaan(10).

$$Overshoot(\%) = \frac{34 - 25}{25} \times 100\% = 36\%$$

Pada uji statis sudut pitch didapatkan risetime lebih dari 0,500 detik dan overshoot 36%. Nilai tersebut berbeda dengan yang didapatkan simulasi, yaitu risetime 1.2693x10-1 detik dan overshoot sebesar 8.9332%.

Pada Gambar 7(b) sudut roll yang diinginkan yaitu 0°. Sistem diharapkan mampu kembali ke sudut roll 0° dari gangguan sudut roll sebesar 30°. 10% dari kondisi yang diinginkan yaitu -27° dan 90% dari kondisi yang diinginkan yaitu -2,7°. Risetime pada respon sudut roll sekitar 0,4 detik. Overshoot dari respon sudut roll dapat dihitung dengan persamaan(10).

$$Overshoot(\%) = \frac{36.3 - 30}{30} \times 100\% = 21\%$$

Pada uji statis sudut roll didapatkan risetime 0,4 detik dan overshoot 21%. Nilai tersebut berbeda dengan yang didapatkan simulasi, yaitu risetime 1.2693x10-1 detik dan overshoot sebesar 8.9332%. risetime dan overshoot yang didapatkan dari hasil uji statis terhadap simulasi mengalami perbedaan nilai. Nilai tersebut masih masuk batas toleransi karena jika ada kesalahan sudut, quadrotor akan kembali pada sudut setpoint untuk melayang mendekati keadaan stasioner. Sehingga untuk mencari kemungkinan nilai rentang PID terbesar dapat dihitung dengan $PID=(0,2*Errorsekarang) + (0,01*Errorsum) + (0,1*Errordif)$ dapat dilihat pada tabel 3.1, dengan selisih satu sampling.

Tabel 3.1 Rentang nilai Kp, Ki, Kd maksimum

Error lalu	Error sekarang	Error Sum	Error Dif	P	I	D	P+I+D
90	-90	0	18	-15.84	0	1,764	-14,094
-90	90	0	-18	15.84	0	-1,764	14,094
-90	-90	-18	0	-15.84	-0,144	0	-15,984
90	90	18	0	15.84	0,144	0	15,984

Error lalu	Error sekarang	Error Sum	Error Dif	P	I	D	P+I+D
90	-90	0	18	-15.84	0	1,764	-14,094
-90	90	0	-18	15.84	0	-1,764	14,094
-90	-90	-18	0	-15.84	-0,8	0	-16,64
90	90	18	0	15.84	0,8	0	16,64

Tabel 3.2 Rentang nilai Kp, Ki, Kd hasil penalaan simulasi

Tabel 3.3 Rentang nilai Kp, Ki, Kd 6 kali Sampling

Karakteristik dari kendali "I" memperlihatkan bahwa dengan posisi kemiringan paling maksimum maupun paling minimum pada posisi yang tetap dengan waktu tertentu. Pembahasan selanjutnya dengan perbandingan nilai rentang PID menggunakan hasil penalaan dari simulasi dengan nilai $K_p=0.176$, $K_i=0.008$, $K_d=0.098$, ditunjukkan pada tabel 3.2. Selanjutnya ErrorSum dapat mencapai titik nilai konstrain yaitu -100 dan 100 dengan jumlah sampling sebanyak: Sampling = 100/18 yaitu 6 kali sampling dengan hasil pada tabel 3.3.

KESIMPULAN

Error lalu	Error sekarang	Error Sum	Error Dif	P	I	D	P+I+D
90	-90	0	18	-18	0	1,8	-16,2
-90	90	0	-18	18	0	-1,8	16,2
-90	-90	-18	0	-18	-0,18	0	-18,18
90	90	18	0	18	0,18	0	18,18

1. Pencarian nilai penala menggunakan metode Harmony Search berhasil di implementasikan pada quadcopter untuk menguji kestabilan quadcopter.

2. Model pencarian nilai penala otomatis menggunakan metode Harmony Search jauh lebih responsif secara waktu nyata

3. Tingkat ketahanan terhadap gangguan didapat hasil yang baik dengan respon waktu perubahan pada sumbu pitch dengan risetime 1.2693×10^{-1} detik dan overshoot sebesar 8.9332%, sedangkan pada sumbu roll dengan risetime 1.2693×10^{-1} detik dan overshoot sebesar 8.9332%..

DAFTAR PUSTAKA

- Dharmawan, A. dan Putera, C.A.L., 2012. Purwarupa Sistem Integrasi Quadcopter dan Mobile Robot. *IJEIS - Indonesian Journal of Electronics and Instrumentation Systems*, 2: 97–108.
- Dharmawan, A., Simanungkalit, Y.Y., dan Megawati, N.Y., 2014. Pemodelan Sistem Kendali PID pada Quadcopter dengan Metode Euler Lagrange. *IJEIS - Indonesian Journal of Electronics and Instrumentation Systems*, 4: 13–24.
- Bolandi, H., Rezaei, M., Mohsenipour, R., Nemati, H., dan Smailzadeh, S.M., 2013. Attitude Control of a Quadrotor with Optimized PID Controller. *Intelligent Control and Automation*, 04: 335.
- Cavalcante, S.R., Araujo, D.C., Varela, A.T., dan Barreto, A.G., 2013. 'Construction and PID Control for Stability of an Unmanned Aerial Vehicle of the Type Quadrotor'. Dipresentasikan pada Robotics Symposium and Competition (LARS/LARC), 2013 Latin American, IEEE, hal. 95–99.

Fan., dan Yongkun., 2007.

Application of PID Controller using MRAC Techniques for Control of the DC Electromotor Drive. China: Institute of Optics and Electronics, Chinese Academy of Sciences.

Xiao, S.L., Li, Y.M., dan Liu, J. G., 2012.

'A model reference adaptive PID control for electromagnetic actuated micro-positioning stage', in Proceedings of the IEEE International Conference, pp:97-102.

Li, X., Qu, S., dan Cai, L., 2014.

'Studing of PID Parameter Optimization Method for Attitude Controller of Quadrotor'. Journal of Computational Information Systems, 10: 7805–7812.

Matthews, J.S., 2006.

'Adaptive Control of Micro Air Vehicles' Thesis, Electrical and Computer Engineering Brigham Young University 2006.

Sadeghzadeh, I., Mehta, A., Zhang, Y., dan Rabath, C., 2011.

'Fault-Tolerant Trajectory Tracking Control of a Quadrotor Helicopter Using Gain Scheduled PID and Model Reference Adaptive Control', Annu. Conf. Progn. Heal. Manag. Soc., pp. 1-10, 2011.

Swarnkar, P. S., Jain, R.K., dan Nema., 2010

'Application of Model Reference Adaptive Control Scheme To Second Order System Using MIT Rule', International Conference on Electrical Power and Energy Systems (ICEPES-2010), MANIT, Bhopal, India.