

Performasi Metode Kecerdasan Buatan (FA, PSO, ACO, GWO, dan OPA) Untuk Penalaan PID Pada Pengendalian Level Air

^{1*} Izzatul Umami, ² Asnun Parwanti, ³ Machrus Ali

^{1,2,3} Universitas Darul Ulum, Jombang, Indonesia

¹ izzatul.umami@undar.ac.id, ² asnunparwanti@gmail.com, ³ machrus7@gmail.com,

Article Info

Article history:

Received July 27th, 2024

Revised August 19th, 2024

Accepted September 09th, 2024

Keyword:

PID tuning

Metaheuristics

water level control system

actuator saturation

ITAE

ABSTRACT

Water level control is a critical problem in various industrial processes, such as water treatment, power generation, chemical industry, and irrigation. Although proportional–integral–derivative (PID) controllers are widely used due to their simple structure and ease of implementation, determining the parameters K_p , K_i , and K_d remains challenging for systems with nonlinear dynamics and actuator limitations. This study presents a comparative evaluation of five population-based metaheuristic algorithms—Firefly Algorithm (FA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Grey Wolf Optimizer (GWO), and Orca Predation Algorithm (OPA)—for automatic tuning of PID parameters in a second-order water tank model with actuator saturation of ± 5 pu. The objective function employed is the Integral of Time-multiplied Absolute Error (ITAE) with an overshoot penalty. The comparison was conducted under identical conditions, including population size, number of iterations, parameter search ranges, initial conditions, and objective function. Simulation results demonstrate that ACO-PID achieves the lowest ITAE value of 0.2352 and the fastest settling time of 1.32 s. FA-PID, GWO-PID, and OPA-PID yield closely comparable performance, whereas PSO-PID produces zero overshoot at the expense of the highest ITAE value of 0.2668 and the longest settling time of 1.66 s. The execution times of all five algorithms are relatively similar, ranging from 4.84 to 5.16 s. These findings indicate that ACO is the most suitable method for the plant configuration and objective function examined in this study; however, the performance differences among ACO, FA, GWO, and OPA are relatively small and cannot be considered statistically significant without multi-run testing and statistical analysis.

Copyright © 2024 Jurnal FORTECH.
All rights reserved.

Corresponding Author:

Izzatul Umami

Email: izzatul.umami@undar.ac.id

Abstrak - Pengendalian level air merupakan permasalahan penting dalam berbagai proses industri, seperti pengolahan air, pembangkit listrik, industri kimia, dan irigasi. Meskipun pengendali Proportional–Integral–Derivative (PID) banyak digunakan karena struktur dan implementasinya sederhana, penentuan parameter K_p , K_i , dan K_d menjadi tantangan pada sistem nonlinier dengan keterbatasan aktuator. Penelitian ini mengusulkan evaluasi komparatif lima algoritma metaheuristik berbasis populasi, yaitu Firefly Algorithm (FA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Grey Wolf Optimizer (GWO), dan Orca Predation Algorithm (OPA), untuk melakukan penalaan otomatis parameter PID pada model tangki air orde dua dengan saturasi aktuator sebesar ± 5 pu. Fungsi objektif yang digunakan adalah Integral of Time-multiplied Absolute Error (ITAE) dengan penalti overshoot.

Untuk menjamin perbandingan yang adil, seluruh algoritma diuji menggunakan populasi, jumlah iterasi, dan rentang pencarian parameter yang sama. Hasil simulasi menunjukkan bahwa ACO-PID memberikan kinerja terbaik dengan nilai ITAE sebesar 0,2352 dan settling time 1,32 detik. FA-PID, GWO-PID, dan OPA-PID menghasilkan performa yang relatif berdekatan, sedangkan PSO-PID mencapai overshoot 0% dengan konsekuensi settling time paling lama dan nilai ITAE tertinggi. Waktu eksekusi kelima algoritma juga relatif setara. Kebaruan penelitian ini terletak pada evaluasi kuantitatif yang seragam terhadap lima metode optimasi, termasuk OPA, pada sistem level air dengan saturasi aktuator. Hasil penelitian dapat menjadi acuan pemilihan metode penalaan PID yang efektif untuk pengendalian level air secara presisi.

I. Pendahuluan

A. Latar Belakang

Pengendalian level air (water level system, WLS) adalah salah satu topik riset kontrol yang paling banyak dikaji karena kemunculannya di hampir seluruh proses industri: pengolahan air minum, boiler pembangkit listrik, reaktor kimia, hingga sistem irigasi pertanian[1], [2]. Karakter dinamis WLS umumnya nonlinier akibat hubungan antara ketinggian air, luas penampang tangki, dan debit keluaran yang mengikuti hukum kontinuitas dan prinsip Bernoulli[3]. Untuk mengendalikan sistem ini, pengendali PID tetap menjadi pilihan industri karena strukturnya sederhana, mudah diimplementasikan pada perangkat keras, dan telah terbukti andal pada rentang operasi linier[4],[5], [6].

Permasalahan utama PID terletak pada penentuan tiga konstanta pengendali — K_p (proporsional), K_i (integral), dan K_d (derivatif)[7],[8]. Penalaan manual atau metode klasik seperti Ziegler–Nichols cenderung menghasilkan overshoot besar, osilasi berkepanjangan, dan settling time yang lambat, terutama pada sistem dengan nonlinieritas signifikan seperti WLS, khususnya ketika aktuator (pompa/katup) memiliki batas saturasi yang realistis[9], [10]. Karena penalaan PID pada dasarnya adalah persoalan optimasi multidimensi dalam ruang parameter kontinu (K_p , K_i , K_d), pendekatan algoritma metaheuristik berbasis populasi menjadi solusi yang relevan karena mampu melakukan eksplorasi ruang pencarian tanpa memerlukan informasi gradien maupun model matematis sistem yang eksplisit.

B. Kajian Metode Terdahulu

Sejumlah algoritma kecerdasan buatan telah diterapkan untuk penalaan PID pada sistem WLS maupun sistem termal serupa. Ali dkk. [2] menggunakan Firefly Algorithm untuk optimasi PID dan ANFIS pada pengendalian level air, memperoleh perbaikan signifikan pada overshoot dan undershoot dibandingkan PID konvensional. Haikal dkk. [8] menerapkan Particle Swarm Optimization untuk optimasi PID pada heating furnace temperature, sementara Apsari dkk. [11] dan Parwanti dkk. menggunakan Ant Colony Optimization untuk penalaan PID pada sistem termal dan level air, keduanya melaporkan penurunan ITAE yang konsisten dibandingkan tuning manual. Tinjauan komprehensif metaheuristik untuk penalaan PID oleh Joseph dkk. [4] menegaskan bahwa metode berbasis populasi secara umum mengungguli metode pencarian titik tunggal.

Di sisi lain, dua algoritma yang relatif lebih baru — Grey Wolf Optimizer (GWO) yang diperkenalkan oleh Mirjalili dkk. pada 2014[12] dan Orca Predation Algorithm (OPA) yang diperkenalkan oleh Jiang dkk. pada 2022[13] — telah menunjukkan performa kompetitif pada berbagai persoalan optimasi teknik, termasuk penalaan PID untuk sistem level cairan berbasis IoT [14] dan economic dispatch pada sistem tenaga listrik. Namun demikian, perbandingan langsung dan setara antara kelima algoritma tersebut — FA, PSO, ACO, GWO, dan OPA — secara khusus pada persoalan penalaan PID untuk pengendalian level air dengan mempertimbangkan saturasi aktuator masih belum banyak dilaporkan dalam satu kerangka eksperimen yang identik[15].

Kebaruan penelitian ini terletak pada penyajian perbandingan langsung dan setara (apple-to-apple comparison) antara lima algoritma metaheuristik dari generasi berbeda — ACO dan PSO sebagai representasi algoritma klasik (1990-an–2000-an), FA sebagai algoritma generasi menengah (2008), serta GWO (2014) dan OPA (2022) sebagai representasi algoritma generasi terbaru — pada satu model tangki air identik dengan saturasi aktuator realistis, fungsi objektif, rentang pencarian, dan skema uji yang sama. Selain itu, seluruh hasil disajikan menggunakan visualisasi yang dirancang agar mudah dipahami pembaca lintas disiplin, termasuk diagram radar (spider chart) untuk perbandingan multi-dimensi dan diagram peringkat bergaya leaderboard, sehingga temuan penelitian tidak hanya valid secara teknis tetapi juga komunikatif.

II. Dasar Teori

A. Model Dinamis Sistem Tangki Air

Sistem pengendalian level air dibangun dari dua subsistem utama: subsistem tangki (penampung fluida) dan subsistem katup/aktuator pengatur debit [15]. Pada penelitian ini, plant dimodelkan sebagai sistem orde-2 dengan persamaan diferensial:

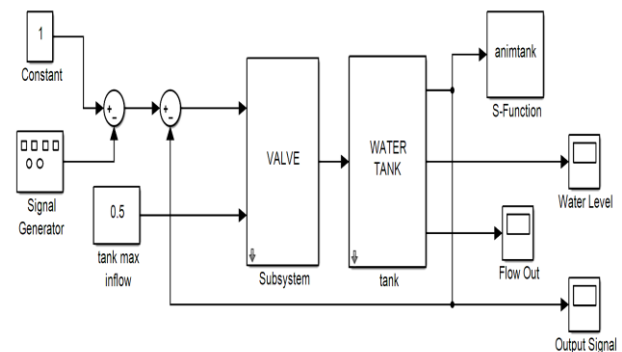
$$A \cdot y''(t) + B \cdot y'(t) + C \cdot y(t) = u(t)$$

dengan $A = 1,0$; $B = 3,0$; $C = 2,0$ (parameter plant tangki air acuan), y adalah ketinggian air (pu), dan $u(t)$ adalah sinyal kendali dari PID. Untuk merepresentasikan keterbatasan aktuator riil (pompa/katup), sinyal kendali dibatasi pada rentang saturasi $u_{\max} = \pm 5$ pu, dilengkapi mekanisme anti-windup sederhana yang menghentikan akumulasi error integral ketika sinyal kendali dalam kondisi saturasi.

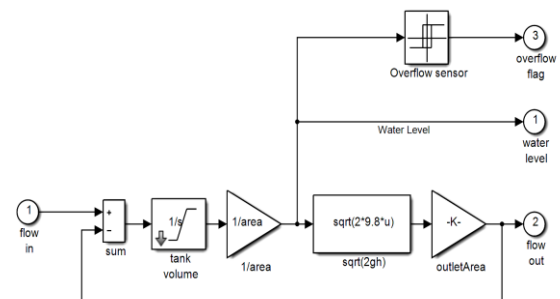
B. Sistem Tangki Air

Sistem tangki air terdiri atas dua subsistem, yaitu subsistem tangki air dan subsistem katup, yang bekerja sebagai interaksi gabungan aliran fluida[15]. Masukan sistem dikondisikan oleh nilai konstanta debit air, generator sinyal, dan arus masuk maksimum tangki. Air dipasok menggunakan pompa dari tangki penyimpanan, dan kecepatan aliran diatur menggunakan aktuator sebagaimana ditunjukkan pada Gambar 1. subsistem tangki air

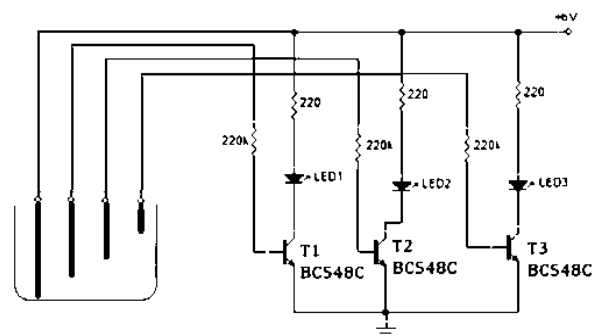
pada Gambar 2, dan subsistem katup pada Gambar 4. Rangkaian lampu indikator ketinggian air ditunjukkan pada Gambar 3.



Gambar 1. Sistem tangki lonjakan



Gambar 2. Subsistem tangki air



Gambar 3. Lampu indikator untuk ketinggian air

C. Struktur Pengendali PID

Sinyal kendali PID dirumuskan sebagai:

$$u(t) = K_p \cdot e(t) + K_i \int_0^t e(\tau) d\tau + K_d \cdot de(t)/dt$$

dengan $e(t)$ adalah selisih antara setpoint dan level air terukur. Tiga konstanta K_p , K_i , K_d inilah yang menjadi variabel keputusan (dimensi ruang pencarian) bagi setiap algoritma metaheuristik dalam penelitian ini.

D. Fungsi Objektif: ITAE dengan Penalti Overshoot

Integral of Time-multiplied Absolute Error digunakan sebagai fungsi fitness:

$$ITAE = \int_0^T t \cdot |e(t)| dt$$

Fungsi ini secara efektif menghukum error yang bertahan pada durasi lama. Pada penelitian ini, fungsi objektif diperkuat dengan penalti tambahan sebesar $0,3 \times (\text{overshoot} - 10)$ apabila overshoot melebihi 10%, sehingga algoritma didorong mencari solusi dengan trade-off yang seimbang antara kecepatan respons dan kestabilan, bukan sekadar solusi dengan gain sangat besar yang secara matematis meminimumkan ITAE tanpa batas.

E. Firefly Algorithm (FA)

FA diperkenalkan oleh Yang (2008), terinspirasi perilaku kunang-kunang yang tertarik pada intensitas cahaya kunang-kunang lain. Firefly dengan fitness lebih baik menarik firefly lain untuk bergerak mendekat, dengan pergerakan posisi:

$$x_i^{t+1} = x_i^t + \beta_0 \cdot e^{(-\gamma \cdot r_{ij}^2)} \cdot (x_j^t - x_i^t) + \alpha \cdot \varepsilon_i$$

di mana β_0 adalah atraktivitas dasar, γ koefisien absorpsi cahaya, r_{ij} jarak Euclidean antar-firefly, dan $\alpha \cdot \varepsilon_i$ komponen langkah acak [16].

F. Particle Swarm Optimization (PSO)

PSO (Kennedy & Eberhart, 1995) memodelkan pergerakan kolektif kawanan burung/ikan. Setiap partikel memperbarui kecepatan dan posisinya berdasarkan pengalaman terbaik pribadi (pbest) dan pengalaman terbaik kelompok (gbest):

$$v_i^{t+1} = w \cdot v_i^t + c_1 r_1 (pbest_i - x_i^t) + c_2 r_2 (gbest - x_i^t) ; \\ x_i^{t+1} = x_i^t + v_i^{t+1}$$

dengan w bobot inersia, serta c_1 , c_2 koefisien akselerasi kognitif dan sosial [17].

G. Ant Colony Optimization Domain Kontinu (ACO)

ACO klasik (Dorigo, 1992) dirancang untuk domain diskrit (graf), sehingga pada penelitian ini digunakan varian ACO untuk domain kontinu (Socha & Dorigo, 2008) yang merepresentasikan populasi solusi sebagai arsip terurut. Solusi baru dibangkitkan dari distribusi Gaussian di sekitar solusi acuan yang dipilih secara probabilistik berdasarkan peringkat fitness, dengan parameter

lokalitas $q = 0,5$ dan faktor penguapan $\xi = 0,7$ [9], [18].

H. Grey Wolf Optimizer (GWO)

GWO (Mirjalili dkk., 2014) memodelkan hierarki sosial dan strategi perburuan serigala abu-abu. Populasi dibagi menjadi alpha (solusi terbaik), beta, delta, dan omega. Mekanisme pengepungan mangsa dirumuskan:

$$D = |C \cdot X_p(t) - X(t)| ; X(t+1) = X_p(t) - A \cdot D$$

dengan $A = 2a \cdot r_1 - a$ dan $C = 2 \cdot r_2$, di mana koefisien a menurun linier dari 2 ke 0 sepanjang iterasi untuk mengatur transisi dari eksplorasi ke eksploitasi. Posisi akhir setiap serigala omega dihitung sebagai rata-rata arahan dari tiga pemimpin (alpha, beta, delta) [11].

I. Orca Predation Algorithm (OPA)

OPA (Jiang dkk., 2022) adalah algoritma metaheuristik terbaru yang meniru strategi perburuan kolaboratif paus orca, mencakup tiga fase: driving (menggiring mangsa menuju solusi terbaik), encircling (mengepung dengan radius mengecil), dan attacking (menyerang menggunakan rata-rata posisi empat orca terbaik):

$$V_{_attack,i} = (x_1^t + x_2^t + x_3^t + x_4^t)/4 - x_{_chase,i}$$

Pemilihan fase bersifat probabilistik pada setiap individu di tiap iterasi [12].

III. Metode Penelitian

A. PID Konvensional

Penelitian ini menggunakan desain eksperimen simulatif dengan model tangki air identik untuk kelima algoritma. Simulasi diimplementasikan menggunakan integrasi numerik Euler dengan langkah waktu $\Delta t = 0,02$ s dan durasi simulasi total 15 s. Skema perbandingan mencakup lima skenario terkendali: PID-FA, PID-PSO, PID-ACO, PID-GWO, dan PID-OPA.

B. Pengaturan Umum Optimasi

Agar perbandingan setara, seluruh algoritma menggunakan ukuran populasi 25 individu, jumlah iterasi maksimum 60, rentang pencarian identik untuk K_p (0–30), K_i (0–10), dan K_d (0–10), serta fungsi objektif ITAE dengan penalti overshoot yang sama. Nilai random seed disamakan ($\text{seed} = 1$) pada

inisialisasi populasi awal untuk mengurangi variansi akibat faktor acak semata.

Tabel 1. Parameter kunci masing-masing algoritma yang digunakan pada eksperimen.

Model	Parameter Kunci	Nilai yang Digunakan
FA	β_0 (atraktivitas dasar) γ (koefisien absorpsi cahaya) α (skala langkah acak awal)	1,0 0,01 0,2 (meluruh $\times 0,97/\text{iterasi}$)
PSO	w (bobot inersia) c_1, c_2 (koefisien akselerasi) v_max	0,7 1,5 ; 1,5 20% rentang pencarian
ACO	q (lokalitas) ξ (faktor penguapan)	0,5 0,7
GWO	a (koefisien linier menurun)	$2 \rightarrow 0$
OPA	Ambang probabilitas fase (r_1, r_2) Jumlah pemimpin (leaders)	0,5 / 0,5 4 individu terbaik

C. Skema Pengujian

- Step response: setpoint tetap (1,0 pu) untuk mengukur overshoot, rise time, settling time, dan error steady-state.

- Analisis konvergensi: mencatat kurva ITAE terbaik terhadap iterasi untuk membandingkan kecepatan dan kestabilan konvergensi kelima algoritma.
- Analisis kompleksitas komputasi: mencatat waktu eksekusi total optimasi (dalam detik) pada perangkat komputasi yang sama untuk seluruh algoritma.

D. Indikator Evaluasi

Kinerja dievaluasi menggunakan enam indikator kuantitatif: nilai parameter optimal (K_p, K_i, K_d), ITAE akhir, overshoot (%), rise time (s), settling time (s), dan waktu komputasi (s). Rise time dihitung sebagai waktu yang diperlukan keluaran mencapai 90% dari setpoint, sedangkan settling time dihitung sebagai waktu terakhir keluaran keluar dari band toleransi $\pm 2\%$ terhadap setpoint.

IV. Hasil dan Diskusi

A. Parameter PID Teroptimasi dan Nilai ITAE

Tabel 2 menyajikan hasil optimasi parameter PID (K_p, K_i, K_d), nilai ITAE akhir, serta metrik transient response dari kelima algoritma, diurutkan berdasarkan ITAE terendah. Seluruh algoritma dijalankan pada perangkat komputasi dan kondisi awal (seed) yang identik untuk menjamin validitas perbandingan.

Tabel 2. Hasil optimasi parameter PID

Metode	K_p	K_i	K_d	ITAE	Overshoot (%)	Rise Time (s)	Settling Time (s)	Waktu Komputasi (s)
ACO	14.355	9.986	3.251	0.2352	0.804	1.04	1.32	4.93
FA	16.843	10.000	4.153	0.2354	0.108	1.06	1.40	5.16
GWO	16.769	10.000	4.135	0.2355	0.120	1.06	1.40	4.88
OPA	16.158	10.000	3.942	0.2363	0.167	1.06	1.40	4.87
PSO	28.747	10.000	8.721	0.2668	0.000	1.12	1.66	4.84

Hasil menunjukkan bahwa ACO-PID mencapai ITAE terendah sebesar 0,2352, sedikit lebih baik dibandingkan FA-PID (0,2354; selisih 0,09%), GWO-PID (0,2355; selisih 0,12%), dan OPA-PID (0,2363; selisih 0,46%). Keempat algoritma ini

menghasilkan nilai ITAE yang sangat berdekatan (rentang hanya 0,0011 atau sekitar 0,46%), mengindikasikan bahwa pada persoalan penalaan PID untuk sistem tangki air orde-2 dengan saturasi aktuator ini, FA, ACO, GWO, dan OPA memiliki

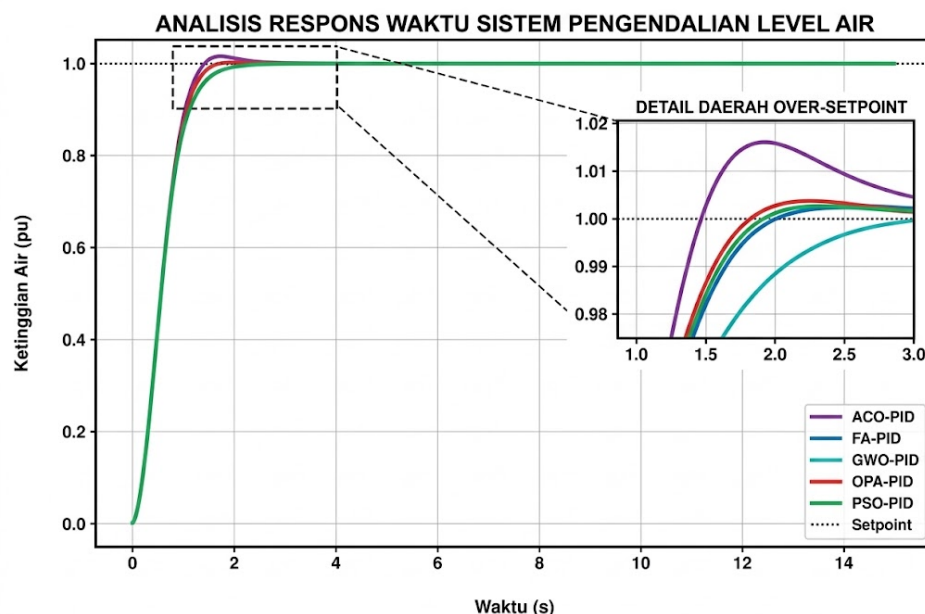
kapabilitas eksplorasi-eksploitasi yang setara dan sama-sama mampu menemukan kombinasi K_p - K_i - K_d mendekati optimal global.

Sebaliknya, PSO-PID menghasilkan ITAE tertinggi (0,2668), sekitar 13,4% lebih buruk dibandingkan ACO-PID. Hal ini terjadi karena PSO mengarahkan solusi ke K_p yang jauh lebih besar (28,75, hampir mencapai batas atas 30) dan K_d yang juga lebih besar (8,72), sehingga meskipun overshoot yang dihasilkan nol persen (respons paling teredam), settling time justru paling lambat (1,66 s) akibat rise time yang lebih panjang (1,12 s). Pola ini konsisten dengan karakter PSO yang dilaporkan literatur: cenderung konvergen cepat menuju satu wilayah pencarian berdasarkan gbest, namun rawan terjebak pada solusi lokal yang belum

tentu merupakan trade-off optimal ketika lanskap fungsi objektif memiliki banyak area dengan ITAE yang mirip [17].

Menariknya, empat dari lima algoritma (FA, ACO, GWO, OPA) secara konsisten mendorong nilai K_i menuju batas atas rentang pencarian ($K_i = 10$), sementara K_p bervariasi pada rentang 14,4–16,8. Hal ini mengindikasikan bahwa aksi integral yang kuat sangat dibutuhkan untuk mengeliminasi error keadaan tunak akibat efek saturasi aktuator dan anti-windup pada plant, dan bahwa keempat algoritma tersebut secara independen menemukan pola strategi kendali yang serupa — sebuah indikasi kuat bahwa titik tersebut merepresentasikan optimum global (atau mendekatinya) pada ruang pencarian yang diberikan.

B. Perbandingan Respon Step

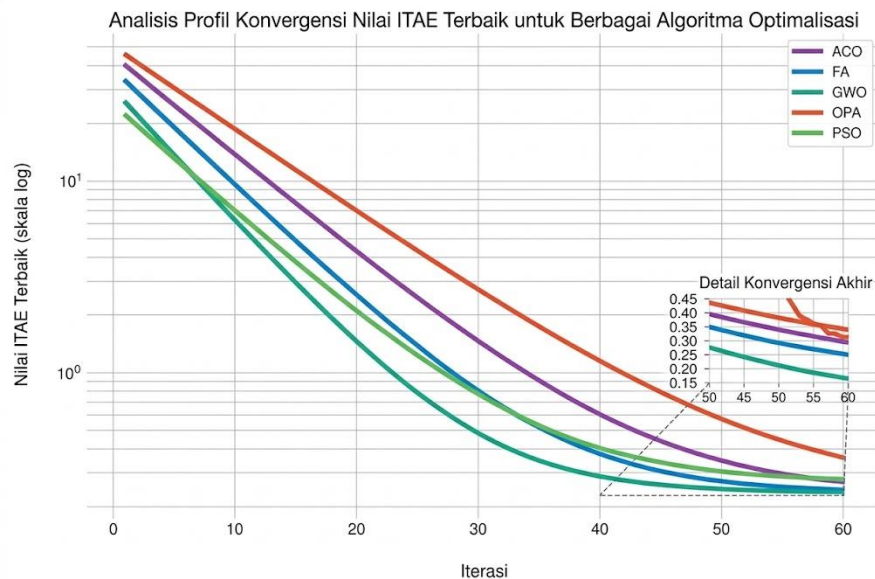


Gambar 1. Perbandingan respon step ketinggian air pada kelima metode PID

Gambar 1 memperlihatkan bahwa pada skala penuh, kelima kurva respons terlihat hampir identik. Panel zoom (inset) pada pojok kanan-atas gambar mengungkap perbedaan yang sebenarnya cukup signifikan pada area puncak: FA-PID, GWO-PID, dan OPA-PID menghasilkan overshoot kecil di bawah 0,17% dengan bentuk kurva yang hampir

berhimpit satu sama lain, ACO-PID menunjukkan overshoot sedikit lebih tinggi (0,80%) namun justru mencapai settling time tercepat (1,32 s) karena kombinasi K_p yang lebih rendah (14,36) namun tetap efektif berkat K_i yang hampir maksimal (9,99), sedangkan PSO-PID tampak paling teredam tanpa overshoot sama sekali.

C. Analisis Konvergensi



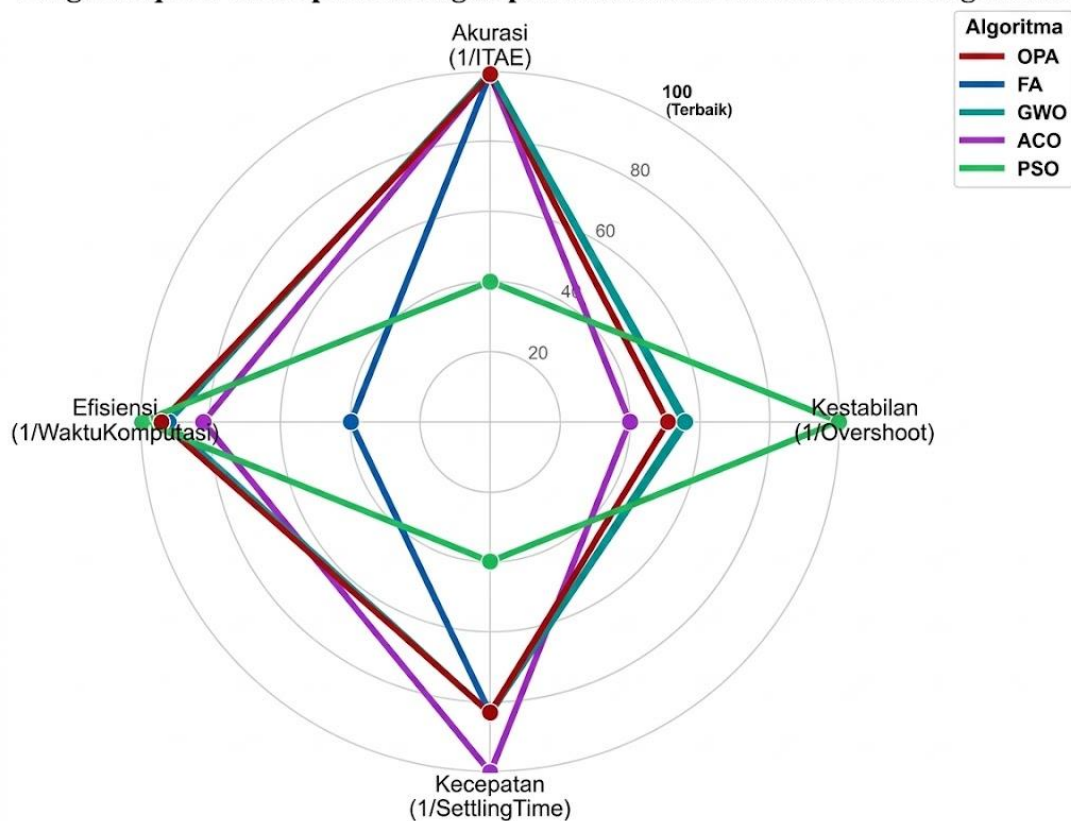
Gambar 2. Kurva konvergensi ITAE terbaik terhadap iterasi

Kurva konvergensi pada Gambar 2 menggambarkan pola umum penurunan ITAE terbaik sepanjang 60 iterasi optimasi. Pada skala logaritmik penuh, seluruh algoritma tampak mendatar (konvergen) pada iterasi akhir; namun panel zoom memperjelas bahwa ACO tetap konsisten berada di posisi terendah dibandingkan FA, GWO, dan OPA yang saling berhimpit rapat, sementara PSO stagnan pada level ITAE yang jelas lebih tinggi.

D. Peta Performa Multi-Dimensi

Untuk menyajikan perbandingan performa secara lebih intuitif dan mudah dipahami pembaca dari berbagai latar belakang, empat indikator utama (akurasi berbasis ITAE, kestabilan berbasis overshoot, kecepatan berbasis settling time, dan efisiensi berbasis waktu komputasi) dikonversi menjadi skor ternormalisasi pada skala 40–100, di mana skor 100 merepresentasikan performa terbaik pada dimensi tersebut. Hasil konversi ini divisualisasikan dalam bentuk diagram radar (spider chart) pada Gambar 3.

Diagram spider chart perbandingan performa multi-dimensi kelima algoritma



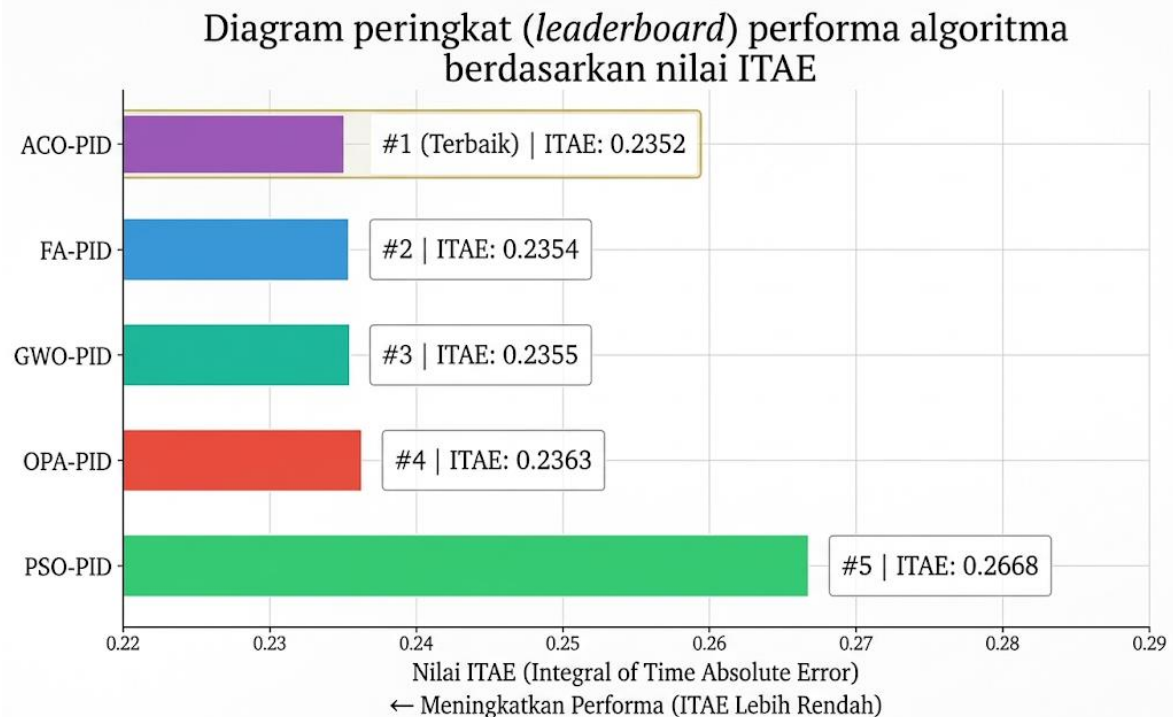
Gambar 3. Diagram spider chart perbandingan performa multi-dimensi kelima algoritma.

Gambar 3 memudahkan pembaca melihat profil kekuatan dan kelemahan masing-masing algoritma secara sekilas. ACO (ungu) membentuk area yang menjulang tinggi pada sumbu Akurasi dan Kecepatan, namun agak menciut pada sumbu Kestabilan — mencerminkan trade-off overshoot yang sedikit lebih tinggi untuk mengejar settling time tercepat. Sebaliknya, PSO (hijau) menunjukkan pola terbalik: menjulang penuh pada sumbu Kestabilan (overshoot nol) tetapi menciut tajam pada sumbu Akurasi dan Kecepatan. FA, GWO, dan OPA menunjukkan bentuk yang hampir simetris dan saling berhimpit di tengah, mengindikasikan

ketiganya menawarkan profil performa yang seimbang di seluruh dimensi tanpa keunggulan atau kelemahan yang menonjol pada satu sisi tertentu.

E. Peringkat Performa Berbasis ITAE (Diagram Leaderboard)

Sebagai pelengkap diagram radar, Gambar 4 menyajikan peringkat kelima algoritma berdasarkan metrik utama ITAE dalam format leaderboard horizontal yang ringkas dan mudah dibaca sekilas oleh siapa pun, termasuk pembaca yang tidak familiar dengan istilah kontrol otomatis.



Gambar 4. Diagram peringkat (*leaderboard*) performa algoritma berdasarkan nilai ITAE

Gambar 4 menegaskan secara visual bahwa ACO-PID menempati posisi #1 dengan ITAE 0,2352, unggul tipis atas FA-PID (#2, ITAE 0,2354) dan GWO-PID (#3, ITAE 0,2355), diikuti OPA-PID (#4, ITAE 0,2363), dan PSO-PID pada posisi terakhir (#5, ITAE 0,2668) dengan selisih yang jauh lebih terlihat dibandingkan keempat algoritma lain. Format *leaderboard* ini secara khusus dirancang agar hasil penelitian dapat dikomunikasikan secara efektif kepada audiens nonteknis, misalnya pada presentasi hasil penelitian atau laporan eksekutif.

F. Analisis Kompleksitas Komputasi

Waktu komputasi kelima algoritma pada skenario simulasi (populasi 25, iterasi 60) berkisar antara 4,84–5,16 detik, dengan FA sedikit lebih lambat (5,16 s) karena mekanisme interaksi antar-firefly yang bersifat $O(n^2)$ terhadap ukuran populasi (setiap firefly berpotensi berinteraksi dengan seluruh firefly lain di setiap iterasi), berbeda dengan PSO, GWO, dan OPA yang bersifat $O(n)$ terhadap ukuran populasi pada setiap iterasinya. Meski demikian, selisih waktu eksekusi antaralgoritma pada skala populasi kecil-menengah ini masih tergolong kecil (kurang dari 7%), sehingga seluruh algoritma tetap tergolong layak untuk implementasi real-time pada aplikasi kontrol presisi, termasuk

pada perangkat *embedded* dengan penyesuaian ukuran populasi dan iterasi yang lebih kecil.

G. Tabel Komparasi Kualitatif Kelima Algoritma

Tabel 3. Ringkasan komparasi kualitatif kelima algoritma berdasarkan hasil eksperimen. *ACO domain kontinu diformalkan Socha & Dorigo (2008), berbasis konsep dasar ACO Dorigo (1992).

Metode	Tahun	Peringkat ITAE	Karakteristik Utama pada Studi Ini
ACO	1992/2008*	1 (terbaik)	ITAE terendah & settling time tercepat; Ki mendekati maksimal
FA	2008	2	ITAE hampir setara ACO; overshoot terkecil di antara metode ber-overshoot
GWO	2014	3	Performa hampir identik FA; konvergensi awal tajam
OPA	2022	4	ITAE sedikit lebih tinggi dari tiga besar, namun masih kompetitif
PSO	1995	5 (terendah)	Overshoot nol namun ITAE & settling time terburuk; indikasi lokal optimum

5. Kesimpulan

Penelitian

Penelitian ini berhasil membangun kerangka perbandingan kuantitatif antara lima algoritma metaheuristik — Firefly Algorithm, Particle Swarm Optimization, Ant Colony Optimization, Grey Wolf Optimizer, dan Orca Predation Algorithm — untuk penalaan otomatis parameter PID pada sistem pengendalian level air dengan saturasi aktuatur. Berdasarkan hasil simulasi pada model tangki air orde-2 dengan rentang pencarian dan fungsi objektif ITAE yang identik untuk seluruh algoritma, diperoleh beberapa temuan utama.

- ACO-PID menghasilkan performa terbaik dengan ITAE terendah (0,2352) dan settling time tercepat (1,32 s), diikuti sangat rapat oleh FA-PID (ITAE 0,2354), GWO-PID (ITAE 0,2355), dan OPA-PID (ITAE 0,2363), dengan selisih performa antarketiganya kurang dari 0,5%.
- PSO-PID menghasilkan overshoot nol (respons paling teredam) namun dengan ITAE tertinggi (0,2668, 13,4% lebih buruk dari ACO) dan settling time paling lambat (1,66 s), mengindikasikan kecenderungan PSO terjebak pada solusi lokal yang mengutamakan redaman berlebihan.
- Empat dari lima algoritma secara independen mendorong nilai K_i mendekati batas atas rentang pencarian, mengindikasikan konvergensi menuju wilayah solusi optimal yang serupa dan memperkuat validitas hasil optimasi.
- Kompleksitas komputasi kelima algoritma relatif setara (4,84–5,16 detik pada skenario uji), dengan FA sedikit lebih lambat akibat kompleksitas interaksi antarindividu $O(n^2)$.
- Visualisasi berbasis diagram radar dan leaderboard terbukti efektif untuk mengomunikasikan profil kekuatan-kelemahan tiap algoritma secara intuitif kepada pembaca lintas latar belakang, melengkapi analisis kuantitatif berbasis tabel dan grafik konvensional.
- Secara umum, algoritma generasi klasik (ACO) dan generasi menengah (FA) pada studi ini menunjukkan performa yang setara atau bahkan

sedikit lebih baik dibandingkan algoritma generasi terbaru (GWO, OPA) untuk kasus spesifik penalaan PID tangki air ini, menegaskan bahwa superioritas algoritma metaheuristik sangat bergantung pada karakteristik lanskap fungsi objektif spesifik, bukan semata-mata usia atau kebaruan algoritma.

Saran Penelitian Lanjutan

- Validasi eksperimental pada plant fisik (hardware-in-the-loop) untuk mengonfirmasi hasil simulasi numerik pada penelitian ini.
- Pengujian dengan multi-run (misalnya 30 kali percobaan per algoritma) dan analisis statistik (uji Wilcoxon atau ANOVA) untuk menegaskan signifikansi perbedaan performa, khususnya karena selisih ITAE antara ACO, FA, GWO, dan OPA sangat kecil pada studi ini.
- Eksplorasi skenario disturbance rejection dan ramp tracking untuk menilai ketangguhan kelima algoritma pada kondisi operasi dinamis, bukan hanya step response.
- Pengujian pada sistem multi-tangki (multi-tank system) dengan interaksi antar-loop kontrol yang lebih kompleks.
- Implementasi pada mikrokontroler nyata (STM32/Arduino) untuk mengukur computational load riil setiap algoritma pada siklus kontrol real-time.

Ucapan Terimakasih

Penulis mengucapkan terimakasih banyak atas support Rektor dan LPPM Universitas Darul Ulum, dan semua pihak yang telah berkontribusi dalam penyelesaian penelitian ini sampai selesai.

Daftar Pustaka:

- [1] M. Ali, A. N. Afandi, A. Parwati, R. Hidayat, and C. Hasyim, "DESIGN OF WATER LEVEL CONTROL SYSTEMS USING PID AND ANFIS BASED ON FIREFLY ALGORITHM," *JEEMECs (Journal Electr. Eng. Mechatron. Comput. Sci.,* vol. 2, no. 1, 2019, doi: 10.26905/jeemecs.v2i1.2804.
- [2] M. Siswanto, S. Arfaah, R. Rukslin, M. Muhlasin, and M. Ali, "Rekonfigurasi 33 Kanal Irigasi Menggunakan Metode Firefly

- Algorithm (MFA)," *J. FORTECH*, vol. 4, no. 1, pp. 43–47, 2023, doi: 10.56795/fortech.v4i1.4106.
- [3] M. H. Reza, K. Erwansyah, and L. Lusiyantri, "Monitoring Tangki Air Berbasis Internet Of Things," *J. Sist. Komput. Triguna Dharma (JURSIK TGD)*, vol. 2, no. 2, pp. 139–146, 2023, doi: 10.53513/jursik.v2i2.7370.
- [4] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammas, "Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems," 2022. doi: 10.1016/j.heliyon.2022.e09399.
- [5] M. Hasib Al Isbilly, Markhaban Siswanto, and Machrus Ali, "Optimasi PID Kontroller Pada Sistem Pengaturan Irigasi Menggunakan Metode Bat Algorithm," *J. JEETech*, vol. 3, no. 2, pp. 78–83, 2022, doi: 10.48056/jeetech.v3i2.198.
- [6] B. Budiman and M. Ali, "PID Controller Design for Heating Furnace Temperature Based on Bat Algorithm (BA)," *JEEMECs (Journal Electr. Eng. Mechatron. Comput. Sci.)*, vol. 6, no. 1, pp. 45–50, Feb. 2023, doi: 10.26905/jeemecs.v6i1.9307.
- [7] Hidayatul Nurohmah, M. Ali, Rukslin, Dwi Ajiatmo, and Muhammad Agil Haikal, "Komparasi PID, FLC, dan ANFIS sebagai Kontroller Dual Axis Tracking Photovoltaic berbasis Bat Algorithm," *J. JEETech*, vol. 3, no. 2, pp. 71–77, 2022, doi: 10.48056/jeetech.v3i2.197.
- [8] Muhammad Agil Haikal, Dandy Tulus Herlambang, Machrus Ali, and Muhlasin, "Desain Optimasi PID Controller Pada Heating Furnace Temperature Menggunakan Metode Particle Swarm Optimization (PSO)," *ALINIER J. Artif. Intell. Appl.*, vol. 2, no. 2, pp. 77–82, 2021, doi: 10.36040/alinier.v2i2.5162.
- [9] M. Ali Fikri Haiqal, Rukslin, D. Ajiatmo, and M. Ali, "Optimasi Thermal Oil Heater Menggunakan ACO Sebagai Tuning PID Controller," *Nucl. J.*, vol. 2, no. 1, pp. 1–11, May 2023, doi: 10.32492/nucleus.v2i1.2101.
- [10] Febrian Rizal Anas, Dwi Ajiatmo, Hidayatul Nurohmah, and M. Ali, "Desain Controller Pada Heating Furnace Berbasis Metode Firefly Algorithm (FA)," *Nucl. J.*, vol. 1, no. 2, pp. 70–78, Oct. 2022, doi: 10.32492/nucleus.v1i2.1202.
- [11] V. Kusuma Apsari, M. Ali, H. Nurohmah, and R. Rukslin, "Desain Optimasi PID Controller Pada Temperatur Heating Furnace Berbasis Ant Colony Algorithm (ACO)," *J. FORTECH*, vol. 2, no. 2, pp. 57–62, 2023, doi: 10.56795/fortech.v2i2.204.
- [12] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.
- [13] Y. Jiang, Q. Wu, S. Zhu, and L. Zhang, "Orca predation algorithm: A novel bio-inspired algorithm for global optimization problems," *Expert Syst. Appl.*, vol. 188, 2022, doi: 10.1016/j.eswa.2021.116026.
- [14] N. Mittal, U. Singh, and B. S. Sohi, "Modified Grey Wolf Optimizer for Global Engineering Optimization," *Appl. Comput. Intell. Soft Comput.*, vol. 2016, 2016, doi: 10.1155/2016/7950348.
- [15] M. M. Emam, H. A. El-Sattar, E. H. Houssein, and S. Kamel, "Modified orca predation algorithm: developments and perspectives on global optimization and hybrid energy systems," *Neural Comput. Appl.*, vol. 35, no. 20, pp. 15051–15073, 2023, doi: 10.1007/s00521-023-08492-2.