

# Penyetelan Sistem Kontrol PID Berbasis IoT Dengan Metode Algoritma Ant Colony

Muhammad Bariq Athallah Hardianto

Informatika, Universitas Pembangunan Nasional “Veteran” Jawa Timur

[20081010119@student.upnjatim.ac.id](mailto:20081010119@student.upnjatim.ac.id)

**Abstrak**— Sistem kontrol PID banyak digunakan dalam aplikasi industri karena mampu memberikan pengendalian dengan respon cepat dan stabilitas yang baik. Namun, performa kontroler PID sangat dipengaruhi oleh ketepatan dalam penyetelan parameter sehingga diperlukan metode optimasi yang efektif. Pada penelitian ini, diusulkan penyetelan parameter PID menggunakan algoritma *Ant Colony Optimization* (ACO) yang diterapkan pada perangkat iTCLab sebagai objek pengendalian suhu. Selain itu, sistem kontrol dirancang berbasis *Internet of Things* (IoT) sehingga proses *monitoring* dan pengaturan *set point* dapat dilakukan secara real-time dan jarak jauh melalui aplikasi IoT MQTT Panel. Dalam proses optimasi, algoritma ACO mencari kombinasi parameter PID terbaik berdasarkan nilai kriteria performansi ITAE. Hasil pengujian menunjukkan bahwa sistem kontrol mampu menjaga suhu pada *set point* 40°C dengan *rise time* sekitar 250 detik, *overshoot* kecil  $\pm 2-3^{\circ}\text{C}$ , serta *settling time* 380–420 detik. Sistem juga berhasil menampilkan data suhu serta mengimplementasikan perubahan *set point* melalui komunikasi MQTT menggunakan broker EMQX dengan tingkat respons yang baik. Berdasarkan hasil tersebut, metode ACO terbukti mampu meningkatkan kinerja kontrol PID pada sistem termal dan integrasi IoT memungkinkan fleksibilitas kontrol secara *remote*.

**Kata Kunci**— PID, Ant Colony Optimization, IoT, iTCLab, MQTT, Kontrol Suhu

## I. PENDAHULUAN

Dalam perkembangan pada dunia industri, ilmu pengetahuan dan teknologi dituntut untuk berkembang sangat pesat untuk mempermudah proses pekerjaan dikarenakan meningkatnya dan beragam kebutuhan yang diperlukan oleh sebuah industri [1]. Dengan meningkatnya kebutuhan, industri akan memperbanyak produksi untuk memenuhi kebutuhan konsumen. Selain memperbanyak produksi, industri tidak boleh mengabaikan stabilitas dan kualitas sebuah produk. Untuk menjaga stabilitas dan kualitas sebuah produk, industri kerap mengimplementasikan sebuah sistem kontrol dalam proses produksinya. Sistem kontrol digunakan untuk mengatur variabel seperti suhu, tekanan, kecepatan, atau aliran, yang semuanya sangat mempengaruhi efisiensi dan keamanan proses industri. Urgensi penerapan sistem kontrol semakin meningkat seiring dengan kompleksitas sistem industri, terutama dalam menjaga stabilitas produksi di tengah permintaan pasar yang semakin ketat. Oleh karena itu, pengembangan sistem kontrol menjadi kebutuhan bagi industri.

Salah satu tugas yang banyak dilakukan di berbagai industri adalah pengukuran suhu karena penting dalam berbagai proses operasional, pengukuran ini dapat mencakup seperti

mengoperasikan pemanas atau memantau suhu menggunakan sensor [2]. Dalam pengukuran suhu sudah banyak dilakukan sebuah inovasi yang mencakup berbagai alat, mulai dari termometer tradisional hingga sensor modern yang terhubung dengan aplikasi IoT untuk memungkinkan pemantauan suhu yang lebih efisien [3]. Selain dalam pengukuran, banyak pula inovasi untuk metode kontrol sebuah pemanas. Tujuan digunakannya sebuah sistem kontrol yaitu untuk mengendalikan sebuah keluaran sesuai dengan masukan yang diberikan. Dengan dikembangkannya sebuah sistem kendali diharapkan meningkatkan efisiensi, kualitas, mengurangi kesalahan yang disebabkan oleh manusia serta meningkatkan konsistensi. Sistem kontrol yang umum digunakan yaitu *Proporsional-Integral-Derivatif* (PID). Sistem kontrol ini umum digunakan pada sistem kontrol pada industri ataupun pada skenario lainnya yang memerlukan pengaturan yang terus menerus serta dengan memanfaatkan PID dapat memberikan pengaturan yang stabil, tepat, dan relatif mudah diimplementasikan [4]. Sistem kontrol PID ini bekerja dengan memanfaatkan 3 aksi yaitu dengan dihitung dengan cara proporsional, integral, dan *derivative*. Tiap aksi ini memiliki fungsinya sendiri mulai dari dihitung dengan proporsional untuk mengurangi error saat proses kontrol, dihitung dengan integral membantu untuk menghilangkan *error steady-state*, dan dihitung dengan *derivative* memberikan prediksi terhadap perubahan yang akan datang berdasarkan laju perubahan nilai *error*. Namun, performa dari sistem kontrol PID sangat bergantung pada penyesuaian parameter masing-masing aksi, yaitu konstanta proporsional  $K_p$ , konstanta integral  $K_i$ , dan konstanta derivatif  $K_d$ . Penyetelan parameter yang tepat dapat memberikan performa kontrol yang optimal, seperti waktu respons yang cepat, stabilitas yang baik, dan minimal *overshoot*. Kinerja dari sistem kontrol PID sangat bergantung pada nilai dari parameter, agar mendapatkan respon yang cepat dan akurat dalam perubahan pada sistem, dilakukan penyetelan untuk nilai parameter PID [4]. Dalam perkembangan ilmu pengetahuan, banyak cara yang telah dilakukan untuk mendapatkan nilai parameter yang optimal untuk sistem kontrol PID. Mulai dari penyetelan nilai parameter secara manual sampai dengan penyetelan menggunakan pendekatan yang paling klasik yaitu menggunakan teknik Ziegler-Nichols yang melibatkan pengujian bertahap untuk menentukan perolehan kritis dan periode osilasi, yang kemudian digunakan untuk menentukan nilai perolehan proporsional, integral, dan *derivative* [5]. Oleh karena itu dilakukan penelitian untuk mencari alternatif pendekatan lainnya untuk menyetel parameter PID. Salah satu pendekatan alternatif yang diteliti yaitu dengan menerapkan algoritma heuristik. Salah satu algoritma heuristik yang dapat diterapkan pada penelitian ini sebagai metode pada penyetelan parameter PID yaitu algoritma *Ant Colony*. Algoritma ini terinspirasi dari perilaku semut dalam mencari

rute terpendek menuju sumber makanan mereka. Setiap semut meninggalkan jejak feromon saat mereka bergerak, dan semut-semut lainnya cenderung mengikuti jalur dengan jejak feromon yang lebih kuat, yang secara bertahap memandu koloni ke jalur optimal. Metode ini memiliki sifat yang baik dan banyak diterapkan pada banyak masalah di berbagai bidang dengan hasil yang menjanjikan [6]. Pada penelitian yang dilakukan oleh [6], penggunaan algoritma *Ant Colony* untuk menyetel parameter PID memberikan hasil yang layak dan lebih baik daripada metode penyetelan parameter PID secara tradisional yang sering diterapkan serta lebih efektif.

Dengan kemunculan teknologi *Internet of Things* (IoT), *Internet of Things* (IoT) banyak digunakan pada industri untuk memantau dan mengontrol sistem kontrol secara langsung bahkan dari jarak jauh. Dengan IoT, data dari sensor dapat dikirim secara langsung ke server atau *cloud*, dimana proses analisis dan pengendalian dapat dilakukan. Hal ini memberikan dampak positif karena dapat meningkatkan efisiensi sebuah pekerjaan terutama pada sebuah sistem kontrol. Pada penelitian yang dilakukan oleh [7] mengemukakan bahwa *monitoring* terkait suhu menggunakan PID dapat dilakukan secara jarak jauh dengan memanfaatkan teknologi *Internet of Things* (IoT) dan bekerja sangat baik.

Berdasarkan berbagai penelitian yang telah ada serta analisis yang dilakukan, penulis memutuskan untuk mengembangkan sebuah penelitian “Penyetelan Sistem Kontrol PID Berbasis IoT Dengan Metode Algoritma *Ant Colony*” yang bertujuan untuk mencoba alternatif lain dalam sistem kontrol terkait suhu yang lebih efisien dan optimal serta dapat dilakukan kontrol secara langsung maupun dari jarak jauh. Dengan memanfaatkan algoritma *Ant Colony* sebagai metode penyetelan parameter PID dan dibuat berbasis *Internet of Things* (IoT) diharapkan akan mengoptimalkan performa kendali PID dengan menemukan parameter yang lebih tepat dan memungkinkan untuk dipantau dan dikendalikan secara langsung dan dari jarak yang jauh. Dengan demikian, diharapkan dapat meningkatkan efisiensi proses dan mengurangi kesalahan dalam proses kontrol.

## II. KAJIAN PUSTAKA

### A. iTCLab

Kit iTCLab adalah perangkat pembelajaran untuk kontrol suhu yang memanfaatkan mikrokontroler ESP32, dua pemanas, LED, dan dua sensor temperatur. Sistem ini mengatur daya pemanas agar suhu dapat dipertahankan sesuai nilai set point yang diatur. Proses perpindahan panas menuju sensor berlangsung melalui konduksi, konveksi, dan radiasi, sementara sebagian panas lainnya hilang ke lingkungan [8]. Perangkat ini dikembangkan sebagai modifikasi dari TCLab milik Brigham Young University (BYU), dengan perubahan utama berupa penggunaan ESP32 menggantikan Arduino Uno, sehingga perangkat dapat mendukung komunikasi berbasis *Internet of Things* (IoT). iTCLab berfungsi sebagai media pembelajaran praktis untuk eksperimen kontrol, pemrograman IoT, dinamika sistem, hingga *Machine Learning* [4].

Rangkaian iTCLab menggunakan PCB yang terhubung dengan ESP32, dilengkapi dua sensor suhu LM35 dan dua transistor NPN TIP120 sebagai pemanas. ESP32 memiliki ADC 10-bit untuk membaca sinyal sensor serta PWM 8-bit untuk mengatur pemanas dan LED. Sensor LM35 memberikan keluaran yang sebanding dengan suhu tanpa perlu kalibrasi tambahan, dan

akurat hingga  $0,5^{\circ}\text{C}$  pada suhu  $25^{\circ}\text{C}$ . Untuk mencegah panas berlebih, sistem otomatis mematikan pemanas ketika suhu mencapai lebih dari  $60^{\circ}\text{C}$ . Daya perangkat berasal dari adaptor 12V 2A dan dilengkapi dengan *heat sink* agar transistor tidak mengalami *overheating* [5].

### B. Sistem Kontrol PID

Sistem kontrol PID merupakan sistem kontrol loop tertutup yang paling banyak diterapkan dalam industri karena sederhana, stabil, dan ekonomis. Prinsip kerja kontroler ini adalah meminimalkan error antara nilai keluaran proses dan set point melalui sinyal koreksi yang dihasilkan dari tiga aksi utama yaitu proporsional, integral, dan derivatif [9]. Persamaan sistem kontrol PID dapat ditulis sebagai berikut :

$$U(t) = \text{proporsional} + \text{integral} + \text{derivatif} \quad (1)$$

$$\text{proporsional} = K_p \cdot e(t) \quad (2)$$

$$\text{integral} = K_i \int e(t) \quad (3)$$

$$\text{derivatif} = K_d \frac{de(t)}{dt} = -K_d \frac{d(pv)}{dt} \quad (4)$$

Dimana untuk nilai  $K_p$ ,  $K_i$ ,  $K_d$  dan dapat dijabarkan sebagai berikut :

$$K_p = K_c \quad (5)$$

$$K_i = \frac{K_c}{\tau_i} \quad (6)$$

$$K_d = K_c \cdot \tau_d \quad (7)$$

Sehingga persamaan sistem kontrol PID sebagai berikut :

$$U(t) = K_c \cdot e(t) + \frac{K_c}{\tau_i} \int e(t) - K_c \cdot \tau_d \frac{d(pv)}{dt} \quad (8)$$

Aksi proporsional bertugas merespons error saat ini, namun nilai  $K_p$  yang terlalu besar dapat menyebabkan osilasi. Aksi integral berfungsi menghilangkan *error* jangka panjang, tetapi jika  $K_i$  terlalu tinggi, sistem bisa menjadi tidak stabil. Sedangkan aksi derivatif  $K_d$  memperkirakan perubahan *error* untuk memperbaiki stabilitas dan mempercepat respon sistem. Penyetelan parameter  $K_p$ ,  $K_i$ , dan  $K_d$  sangat penting agar sistem mencapai performa optimal. Banyak pendekatan yang dapat digunakan, mulai dari metode klasik seperti Ziegler-Nichols hingga metode optimasi cerdas seperti GA, PSO, dan ACO. Dalam penelitian ini, parameter PID disetel menggunakan bentuk dependen yang memanfaatkan penguatan kontrol  $K_c$ , waktu reset integral  $\tau_i$ , dan konstanta derivatif  $\tau_d$  agar kontrol lebih terintegrasi dan responsif.

### C. Algoritma Ant Colony

Algoritma *Ant Colony* merupakan algoritma heuristik yang terinspirasi dari perilaku koloni semut untuk mencari makanan. Dalam mencari makanan semut memanfaatkan suatu zat yang disebut feromon. Semut meninggalkan jejak feromon saat bergerak, jejak ini berfungsi sebagai alat komunikasi dengan

semut lain. Setiap semut bergerak dengan acak mencari rutenya sendiri, namun pada saat bertemu dengan feromon, semut dapat memutuskan untuk mengikuti jalur tersebut atau tidak. Keputusan ini didasarkan pada probabilitas yang dipengaruhi oleh kekuatan feromon. Semakin banyak semut yang mengikuti jalur tertentu, semakin kuat jejak feromonnya, sehingga meningkatkan kemungkinan semut lain untuk mengikuti jalur yang sama [10].

Pada dasarnya penerapan algoritma *Ant Colony* berdasarkan [6] sebagai berikut:

#### 1) Penentuan Node

Pada langkah awal penerapan algoritma *Ant Colony*, permasalahan nyata harus direpresentasikan dalam bentuk graf atau kumpulan *node*. Setiap *node* berfungsi sebagai titik yang dapat dilewati oleh semut selama proses pencarian, mulai dari titik awal hingga ke titik tujuan atau lokasi makanan. Pemilihan *node* yang sesuai menjadi aspek penting karena akan memengaruhi keberhasilan algoritma dalam menemukan solusi yang optimal.

#### 2) Ukuran Koloni dan Waktu Mencari Makan

Jumlah semut dalam koloni mencerminkan banyaknya agen yang bergerak menuju sumber makanan, sedangkan durasi pencarian digambarkan melalui jumlah iterasi. Kedua parameter ini memiliki peran penting dalam menentukan kinerja algoritma. Jika jumlah semut maupun iterasi terlalu sedikit, proses pencarian solusi dapat berhenti lebih awal sebelum mencapai konvergensi sehingga hasil yang diperoleh tidak optimal. Namun, apabila jumlahnya terlalu banyak, algoritma memang cenderung mencapai konvergensi, tetapi proses komputasinya menjadi tidak efisien karena iterasi yang dilakukan melewati kebutuhan sebenarnya. Akibatnya, waktu eksekusi meningkat secara signifikan dan keunggulan algoritma menjadi berkurang.

#### 3) Evaluasi Jalur

Ketika semut melakukan pencarian makanan, kualitas jalur yang ditemukan umumnya dapat dinilai dari seberapa pendek jalur tersebut. Namun, dalam penerapan algoritma *Ant Colony* untuk menyetel parameter PID, diperlukan ukuran penilaian yang lebih relevan. Kualitas solusi tidak lagi hanya digambarkan melalui panjang jalur, melainkan melalui suatu indeks yang mampu merepresentasikan performa solusi dalam konteks pengendalian, di mana solusi yang diperoleh dianalogikan sebagai jalur yang dipilih oleh semut.

#### 4) Probabilitas Pemilihan Jalur

Probabilitas pemilihan jalur merupakan prinsip utama dalam pergerakan semut dari satu *node* ke *node* lainnya. Ketika semut menentukan jalur yang akan dilalui, nilai probabilitas dihitung menggunakan persamaan yang ditunjukkan pada Persamaan 9.

$$P_{(i,j)}^k(t) = \frac{[\tau_{(i,j)}]^\alpha \cdot [\eta_{(i,j)}]^\beta}{\sum_{S \in J(i)} [\tau_{(i,j)}]^\alpha \cdot [\eta_{(i,j)}]^\beta}, j \in J_i \quad (9)$$

$$0, j \in J_i$$

Simbol  $P_{(i,j)}^k$  menunjukkan peluang bagi semut ke- $k$  untuk berpindah dari *node*  $i$  ke *node*  $j$  pada iterasi ke- $k$ . Sementara itu,  $\eta$  menggambarkan jumlah feromon yang terdapat pada jalur antara *node*  $i$  dan *node*  $j$ , dan  $\tau$  mewakili nilai heuristik dari *node* tersebut. Parameter  $\alpha$  digunakan sebagai pengatur

seberapa besar pengaruh feromon terhadap keputusan pemilihan jalur, sedangkan  $\beta$  mengatur seberapa kuat pengaruh nilai heuristik pada proses pemilihan. Dengan demikian, terlihat bahwa peluang suatu jalur dipilih akan semakin besar apabila memiliki feromon yang tinggi serta kualitas jalur yang baik.

#### 5) Pembaruan Feromon

Ketika semua semut selesai melakukan pencarian jalur, maka feromon pada rute yang telah dilewati akan diperbarui. Peningkatan jumlah feromon menandakan bahwa jalur tersebut memiliki kualitas lebih baik karena lebih sering dipilih oleh semut. Proses pembaruan feromon ini dijelaskan melalui persamaan 10.

$$\tau_{(i,j)} = \rho \cdot \tau_{(i,j)} + \Delta\tau_{(i,j)} \quad (10)$$

$$\Delta\tau_{(i,j)} = \frac{\text{cost terbaik}}{\text{cost semut } (A)} \quad (11)$$

Pada persamaan tersebut,  $\rho$  merupakan parameter penguatan atau koefisien penguapan feromon. Semakin besar nilai feromon yang tertinggal pada suatu jalur, semakin tinggi peluang jalur tersebut dipilih kembali oleh semut pada iterasi berikutnya. Nilai  $\Delta\tau_{(i,j)}$  menunjukkan tambahan feromon pada jalur tersebut dan dirumuskan lebih lanjut dalam Persamaan 11, yang memperhitungkan perbandingan antara solusi terbaik yang ditemukan dengan kualitas solusi dari semut yang melewati jalur tersebut.

#### D. EMQX MQTT

*Message Queuing Telemetry Transport* (MQTT) adalah protokol komunikasi IoT berbasis *publish-subscribe* yang ringan dan cocok digunakan pada jaringan dengan bandwidth terbatas serta koneksi tidak stabil. Protokol ini memerlukan broker sebagai perantara antara *publisher* dan *subscriber*, di mana EMQX menjadi salah satu broker open-source yang banyak digunakan karena mampu menangani jutaan koneksi dengan latensi rendah, *throughput* tinggi, dan *skalabilitas* yang baik. Dengan performanya yang andal, EMQX sangat sesuai diterapkan pada sistem inkubator untuk memastikan komunikasi *real-time* dan stabil antara sensor, *aktuator*, dan sistem kendali sehingga proses *monitoring* dan kontrol dapat dilakukan dari jarak jauh [11].

#### E. ITAE

*Integral of Time-weighted Absolute Error* (ITAE) atau merupakan indeks untuk mengevaluasi performa kinerja dari sistem kontrol. Indeks ini cukup populer dan diusulkan oleh Graham dan Lathrop pada tahun 1953. Mereka mengembangkan serangkaian koefisien fungsi transfer yang dinormalisasi untuk sistem dengan orde ke-2 hingga ke-8. Tujuannya adalah untuk meminimalkan kriteria ITAE saat sistem diberikan *step input*. Secara matematis, ITAE didefinisikan sebagai berikut [12]:

$$ITAE = \int t \cdot |e(t)| dt \quad (12)$$

#### F. IoT MQTT Panel

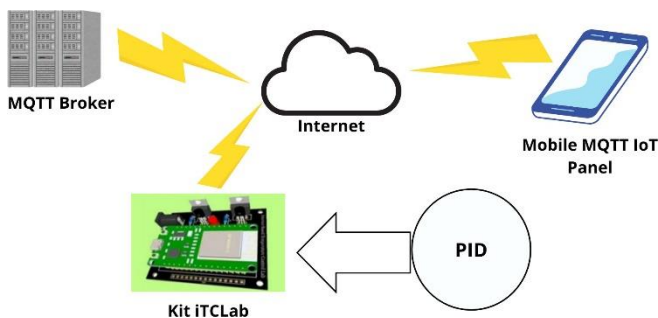
Dalam implementasi pemanfaatan konsep IoT, baik untuk pemantauan atau kontrol hasil dari pembacaan sensor pada *aktuator* secara jarak jauh dibutuhkan aplikasi IoT MQTT Panel yang berjalan pada perangkat *mobile* yang memiliki sistem

operasi Android maupun IOS [8]. Aplikasi ini memiliki kemudahan dalam kustomisasi *widget* yang diperlukan oleh pengguna dalam pemantauan atau memberikan tindakan pada perangkat IoT yang terhubung. Aplikasi ini memanfaatkan protokol MQTT dalam berkomunikasi dengan perangkat IoT. Dengan menggunakan aplikasi ini, pemantauan dan kontrol suhu yang menerapkan cara kerja konsep IoT akan mudah diimplementasikan.

### III. METODOLOGI

#### A. Desain Arsitektur Sistem

Pada penelitian ini dirancang arsitektur sistem yang terdiri dari empat komponen utama, yaitu: (1) *Plant*, (2) Internet, (3) *Cloud*, dan (4) *Interface*. *Plant* merupakan perangkat keras yang berfungsi melakukan pengukuran suhu melalui sensor dan mengatur pemanas sebagai *aktuator*, dimana pengendaliannya menggunakan PID yang parameternya telah disetel menggunakan algoritma *Ant Colony*. Pada penelitian ini, *plant* yang digunakan adalah kit iTCLab. Komponen berikutnya yaitu jaringan internet yang berperan sebagai media komunikasi antar komponen dengan protokol MQTT. Selanjutnya, *cloud* digunakan sebagai wadah penyimpanan dan distribusi data dari *plant* untuk diteruskan ke *interface*, dan dalam sistem ini EMQX digunakan sebagai broker. Terakhir, *interface* berfungsi untuk *monitoring* dan kontrol suhu secara langsung oleh pengguna, yang pada penelitian ini berupa aplikasi *mobile* menggunakan IoT MQTT Panel.



Gbr. 1 Desain Arsitektur Sistem

#### B. Desain Sistem Kontrol

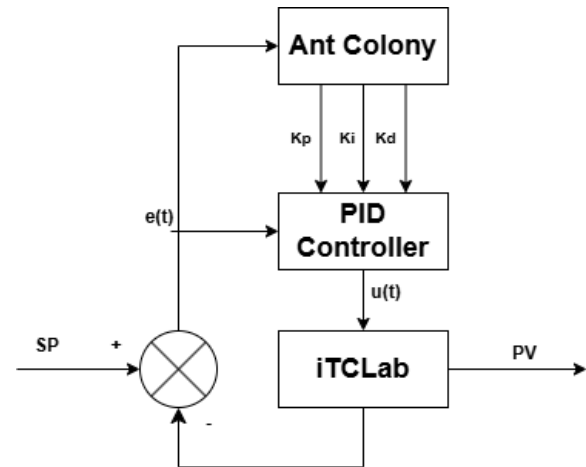
Penelitian ini mengembangkan sistem kontrol untuk mengatur pemanas pada perangkat iTCLab menggunakan metode PID. Tujuan utamanya adalah meningkatkan performa pemanas melalui penyesuaian parameter PID yang lebih tepat, sehingga respon suhu menjadi lebih cepat, *error* semakin kecil, dan kestabilan sistem lebih baik. Proses perancangan kontrol melibatkan penentuan parameter  $K_c$ ,  $\tau_i$ , dan  $\tau_d$  sebagai bentuk dependen dari  $K_p$ ,  $K_i$ , dan  $K_d$  guna memperoleh hasil kontrol yang optimal. Pada penelitian ini, proses *tuning* parameter PID dilakukan dengan algoritma *Ant Colony*. Desain sistem kontrol yang diusulkan dapat dilihat pada Gbr 2. Alur dari implementasi penyetelan parameter PID menggunakan algoritma *Ant Colony* dapat dilihat pada Gbr 3.

##### 1) Inisiasi Parameter

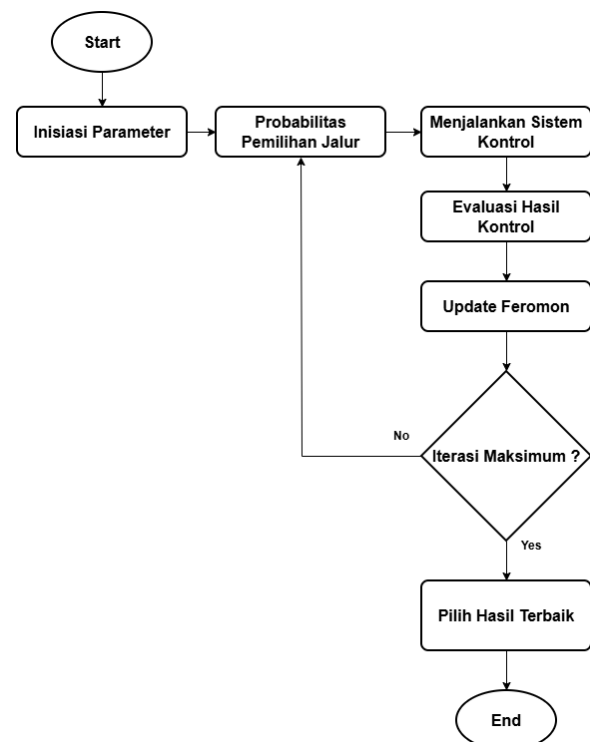
Tabel 1 Parameter Algoritma Ant Colony

Parameter	Parameter
Jumlah Iterasi	100

Jumlah Semut	10
$\alpha$	0.9
$\beta$	2
$\rho$	0.5
Banyak Node	1000
Batas Parameter PID	[0.1, 60]



Gbr. 2 Desain Sistem Kontrol



Gbr. 3 Alur Algoritma Ant Colony

##### 2) Probabilitas Pemilihan Jalur

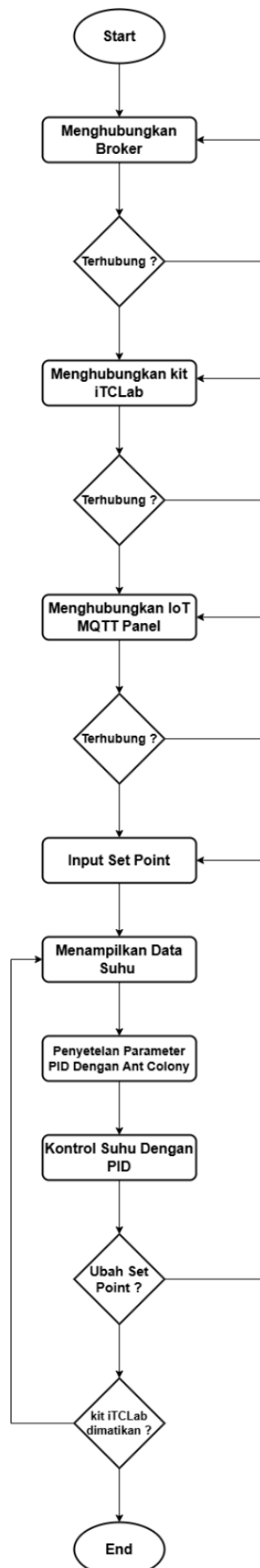
Probabilitas pemilihan jalur berdasarkan konteks penelitian ini yaitu menghitung kemungkinan parameter PID yang akan dipilih oleh tiap semut dalam menjalankan sistem kontrol. Rumus untuk perhitungan probabilitas dapat dilihat pada persamaan 9.

##### 3) Menjalankan Sistem Kontrol

Setelah semua semut memilih kombinasi parameter PID berdasarkan nilai probabilitasnya, kombinasi parameter tersebut digunakan untuk menghitung nilai kontrol yang akan dilakukan. Rumus untuk menghitung nilai kontrol dapat dilihat pada



persamaan 8.



Gbr. 4 Alur Sistem

#### 4) Evaluasi Sistem Kontrol

Tiap semut akan melakukan kontrol PID dengan parameter berdasarkan probabilitasnya. Untuk menentukan kualitas kontrol yang telah dilakukan tiap semut, dilakukan evaluasi menggunakan ITAE di mana persamaannya dapat dilihat pada persamaan 11. Hasil dari evaluasi ini akan dijadikan patokan dalam kualitas kombinasi parameter yang dipilih semut sehingga pada iterasi selanjutnya semut akan bisa mencari kombinasi parameter yang lebih baik dari sebelumnya.

#### 5) Perubahan Feromon

Setelah semut melakukan perjalanan dalam konteks penelitian ini semut memilih kombinasi parameter dan telah melakukan kontrol, jalur atau kombinasi yang dipilih oleh semut akan ditandai dengan mengisi feromon. Untuk menghitung nilai perubahan feromon yang dilewati para semut dapat dilihat pada persamaan 10. Dengan melakukan ini semut akan lebih mudah mencari kemungkinan kombinasi nilai parameter PID terbaik.

#### 6) Pilih Hasil Terbaik

Setelah semua iterasi dilakukan dalam mencari kombinasi nilai parameter PID berdasarkan algoritma *Ant Colony*, dilakukan pemilihan kombinasi nilai parameter terbaik. Pemilihan kombinasi nilai parameter terbaik didasarkan pada nilai performa kontrol terbaik yang didapatkan dari nilai ITAE yang dilakukan tiap semut pada tiap iterasinya.

#### C. Alur Sistem

Pada *flowchart* sistem yang dibuat, proses dimulai dengan menghubungkan PC sebagai pemroses kontrol suhu dengan MQTT broker untuk meneruskan data menuju pengguna. Setelah koneksi dengan broker berhasil, PC kemudian disambungkan ke kit iTCLab, yang ditandai dengan LED pada perangkat menyala. Selanjutnya, aplikasi IoT MQTT Panel dihubungkan dengan broker sehingga data suhu dari sensor pada iTCLab dapat ditampilkan pada *interface*. Pengguna kemudian dapat memasukkan *set point* suhu yang diinginkan, dan sistem akan menjalankan kontrol PID dengan parameter yang telah disetel menggunakan algoritma *Ant Colony*. Selama proses kontrol berlangsung, nilai suhu dikirim dan ditampilkan secara *real-time* kepada pengguna. Jika pengguna ingin mengubah *set point*, nilai tersebut dapat diperbarui kapan saja. Sistem akan bekerja secara berkelanjutan untuk menjaga suhu tetap pada *set point* yang ditargetkan selama perangkat masih aktif dan tidak dimatikan.

## IV. HASIL

Pada penelitian ini dilakukan pengujian performa sistem kontrol PID yang parameternya disetel menggunakan algoritma *Ant Colony* pada perangkat iTCLab. Evaluasi hasil dilakukan dengan melihat respons suhu terhadap nilai *set point* serta kemampuan sistem dalam melakukan *monitoring* dan kontrol berbasis IoT. Berdasarkan grafik hasil kontrol yang ditampilkan pada Gbr. 5, dapat diamati bahwa sistem mampu mengatur suhu menuju nilai *set point* sebesar 40°C. Nilai suhu mengalami kenaikan secara bertahap hingga mendekati *set point* dengan karakteristik performa sebagai berikut:

- *Rise Time* berada pada rentang sekitar 250 detik, yang menunjukkan bahwa respon pemanasan bekerja dengan cukup cepat dalam mencapai nilai mendekati *set point*.
- *Overshoot* terjadi kecil dengan lonjakan suhu pada kisaran

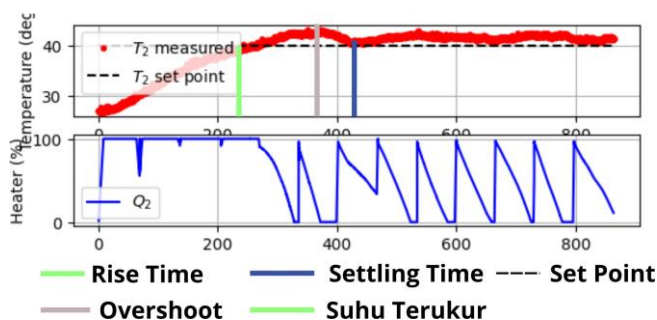
$\pm 2-3^{\circ}\text{C}$  di atas *set point*, namun masih dalam batas aman dan dapat diterima dalam pengendalian sistem termal.

- *Settling Time* berada pada sekitar 380–420 detik, menunjukkan bahwa sistem dapat kembali stabil secara cepat setelah mengalami sedikit osilasi pada fase awal.

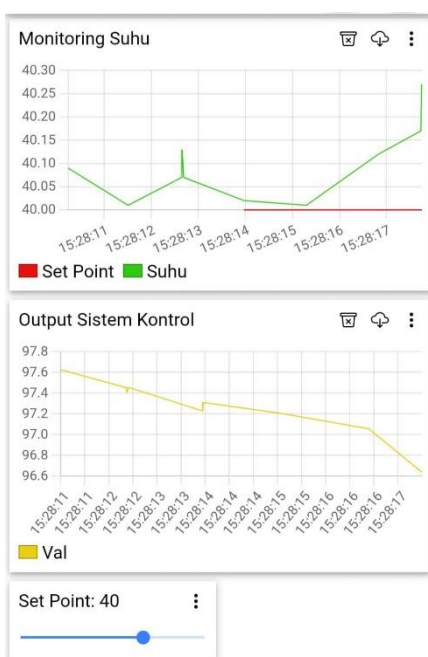
Pengujian berikutnya dilakukan untuk memastikan bahwa sistem dapat dikendalikan dan dipantau secara jarak jauh. Implementasi IoT menggunakan protokol MQTT memperbolehkan pengguna mengubah nilai *set point* melalui aplikasi IoT MQTT Panel pada *smartphone*. Hasil tampilan antarmuka aplikasi diperlihatkan pada Gbr. 6 dan Gbr. 7 yang menunjukkan bahwa:

- Nilai suhu dapat terbaca secara real-time dan ditampilkan dalam bentuk grafik.
- Pengguna dapat dengan langsung mengubah *set point* menggunakan *widget slider*.
- Nilai keluaran sistem kontrol (PWM) turut ditampilkan sehingga perubahan tindakan kendali dapat dipantau setiap waktu.

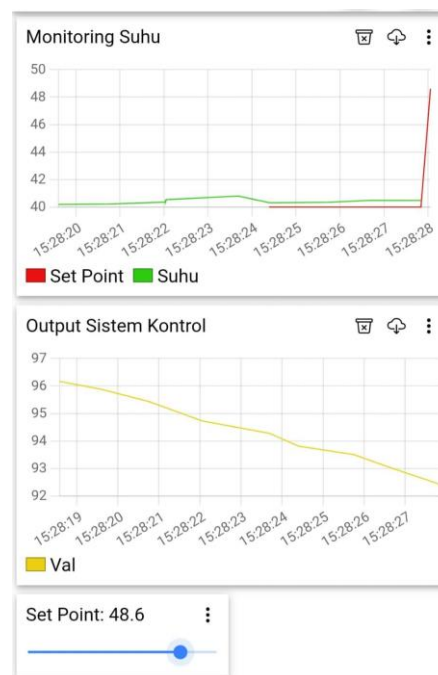
Pengamatan ini membuktikan bahwa integrasi teknologi IoT pada sistem kontrol berjalan dengan baik, sehingga pengendalian tidak hanya efektif secara lokal tetapi juga fleksibel dilakukan dari jarak jauh.



Gbr. 5 Hasil Kontrol iTCLab



Gbr. 6 Monitor dan Kontrol dari Ponsel



Gbr. 7 Mengubah Set Point dari Ponsel

## V. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, sistem kontrol PID yang telah disetel menggunakan algoritma *Ant Colony* berhasil mendapatkan performa pengendalian suhu pada perangkat iTCLab yang ditandai dengan *rise time* di sekitar 250 detik di mana relatif cepat dalam mencapai *set point*, *overshoot* yang dialami kurang lebih di antara  $2^{\circ}-3^{\circ}$  celcius, *settling time* yang dimulai di sekitar 380 - 420 detik di mana setelah waktu ini performa iTCLab menjadi stabil. Sistem *monitoring* dan kontrol berbasis *Internet of Things* (IoT) dapat berjalan dengan baik menggunakan aplikasi IoT MQTT Panel sehingga pengguna mampu melakukan pemantauan dan kontrol suhu secara *real-time* dan jarak jauh ditandai dengan berhasilnya pengguna dalam mengubah set point suhu yang diinginkan serta pemantauan nilai suhu dan nilai kontrol secara *real-time*. Integrasi metode optimasi ACO pada sistem kontrol PID dan teknologi IoT terbukti memberikan solusi pengendalian suhu yang lebih efisien, adaptif, dan fleksibel dalam implementasi.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada tim Santika yang telah memberikan kesempatan untuk membagikan penelitian yang telah penulis lakukan.

## REFERENSI

- [1] I. G. Widharma, "Kajian analisis sistem kendali berbasis PLC dalam dunia industri," 2021. <https://doi.org/10.13140/RG.2.2.28533.09448>
- [2] S. Hadi, R. P. M. D. Labib, and P. D. Widayaka, "Perbandingan akurasi pengukuran sensor LM35 dan sensor DHT11 untuk monitoring suhu berbasis Internet of Things," STRING (Satuan Tulisan Riset dan Inovasi Teknologi), vol. 6, no. 3, pp. 269–278, 2022.
- [3] M. Ikhwanshohfa, A. Nuramal, and N. I. Supardi, "Pemanfaatan Internet of Things untuk monitoring suhu di BPPT-MEPPPO," RekayasaMekanika, vol. 4, no. 1, pp. 19–24, 2020. <https://doi.org/10.33369/rekayasamekanika.v4i1.13274>
- [4] B. Rahmat, M. Waluyo, T. Rachmanto, M. Afandi, N. K. Sari, H.

- Widyantara, and H. Harianto, "iTCLab temperature monitoring and control system based on PID and Internet of Things (IoT)," in Handbook of Research on Emerging Designs and Applications for Microwave And Millimeter Wave Circuits, 2023. <https://doi.org/10.4018/978-1-6684-5629-3.ch012>
- [5] B. Rahmat et al., "iTCLab PID control tuning using deep learning," in Proc. 2023 IEEE 9th Information Technology International Seminar (ITIS), Batu Malang, Indonesia, 2023, pp. 1–4. <https://doi.org/10.1109/ITIS59651.2023.10420130>
- [6] L. Wang, Y. Luo, and H. Yan, "Ant Colony optimization-based adjusted PID parameters: a proposed method," PeerJ Computer Science, vol. 9, e1660, 2023.
- [7] B. Rahmat, M. Waluyo, and T. A. Rachmanto, "Temperature monitoring via the Internet of Things using PID-iTCLab," Nusantara Science and Technology Proceedings, vol. 2023, no. 33, pp. 197–203, 2023. <https://doi.org/10.11594/nstp.2023.3332>
- [8] Rahmat, Basuki & Muljono, Muljono. (2024). PEMROGRAMAN INTERNET OF THINGS (IoT) DENGAN ARDUINO DAN PYTHON JILID 1 Basuki Rahmat Muljono PENERBIT CV.EUREKA MEDIA AKSARA.
- [9] S. Prainetr, T. Phurahong, K. Janprom, and N. Prainetr, "Design tuning PID controller for temperature control using Ant Colony optimization," in Proc. 2019 IEEE 2nd Int. Conf. Power and Energy Applications (ICPEA), Singapore, 2019, pp. 124–127. <https://doi.org/10.1109/ICPEA.2019.8818517>
- [10] F. Z. Baghli, Y. Lakhali, and Y. Ait El Kadi, "The efficiency of an optimized PID controller based on Ant Colony Algorithm (ACO-PID) for the position control of a multi-articulated system," J. Robot. Control, vol. 4, no. 3, p. 289, 2023. <https://doi.org/10.18196/jrc.v4i3.17709>
- [11] M. Kashyap, A. K. Dev dan V. Sharma, "Implementation and analysis of EMQX broker for MQTT protocol in the Internet of Things," e-Prime - Advances in Electrical Engineering, Electronics and Energy, vol. 10, p. 100846, 2024.
- [12] A. E. A. Awouda dan R. B. Mamat, "New PID tuning rule using ITAE criteria," International Journal of Engineering (IJE), vol. 3, no. 6, p. 597–608, 20