

---

# PERBANDINGAN SIMULASI PID WATER LEVEL CONTROL MENGGUNAKAN MATLAB SCRIPT DAN SIMULINK

Adit Mulyana<sup>1\*</sup>, Desmira<sup>2</sup> Tita Nurdaiah<sup>3</sup>, Lailatul Nadiroh<sup>4</sup>, Ahmad Rifa Futu Azim<sup>5</sup>

Wildan Muzaki<sup>6</sup>

<sup>1-6</sup>Universitas Sultan Ageng Tirtayasa, Indonesia

\*email korespondensi: [kemetadit@gmail.com](mailto:kemetadit@gmail.com)

DOI : 10.48144/suryateknika.v10i1.2271

Received: 2 Desember 2025

Revised: 10 Desember 2025

Accepted: 15 Desember 2025

---

## Abstract

Water level control in tanks is an important element in industrial processes, so a control method capable of maintaining liquid level stability precisely is needed. This study simulates air level control using Proportional-Integral-Derivative (PID) control on a first-order linear tank plant model, compared through two simulation approaches, namely numerical MATLAB script programming and block modeling in Simulink. PID parameters obtained using the Ziegler–Nichols tuning method and simulation results show that the script-based approach has more optimal performance than Simulink. Testing using the script produces a rise time of 2.15 seconds, a settling time of 6.32 seconds, an overshoot of 3.8%, and a stable-state error of less than 0.02 m, while Simulink shows a rise time of 3.41 seconds, a settling time of 8.57 seconds, and an overshoot of 11.4% due to the influence of the default solver (ode45) which causes internal interpolation to affect transient characteristics. Overall, the script approach improved rise time by 35% and produced a final error close to zero, so that mathematical numerical simulations were considered more accurate and could be used as a basis for validation before physical implementation in liquid level control systems.

Keywords: PID, water level, MATLAB Script, Simulink, Ziegler–Nichols tuning, first-order tank

---

## Abstrak

Pengendalian ketinggian air pada tangki merupakan elemen penting dalam proses industri sehingga diperlukan metode kontrol yang mampu menjaga kestabilan level cairan secara presisi. Penelitian ini melakukan simulasi pengendalian level air menggunakan kontrol *Proportional-Integral-Derivative* (PID) pada model plant *first-order linear tank*, dibandingkan melalui dua pendekatan simulasi yaitu pemrograman MATLAB *script* numerik dan pemodelan blok di Simulink. Parameter PID diperoleh menggunakan metode tuning *Ziegler–Nichols* dan hasil simulasi menunjukkan bahwa pendekatan berbasis *script* memiliki kinerja lebih optimal dibandingkan Simulink. Pengujian menggunakan *script* menghasilkan *rise time* 2,15 detik, *settling time* 6,32 detik, *overshoot* 3,8%, serta *steady-state error* kurang dari 0,02 m, sedangkan Simulink menunjukkan *rise time* 3,41 detik, *settling time* 8,57 detik, dan *overshoot* 11,4% akibat pengaruh pemilihan solver bawaan (*ode45*) yang menyebabkan interpolasi internal mempengaruhi karakteristik transien. Secara keseluruhan, pendekatan *script* meningkatkan *rise time* sebesar 35% dan menghasilkan error akhir yang mendekati nol, sehingga simulasi numerik matematis dinilai lebih akurat dan dapat dijadikan dasar validasi sebelum implementasi fisik pada sistem kontrol level cairan.

**Kata kunci:** PID, level air, MATLAB Script, Simulink, tuning Ziegler–Nichols, first-order tank.

---

## 1. Pendahuluan

Pengendalian ketinggian air merupakan salah satu aspek penting dalam sistem proses industri, seperti pengolahan air bersih, industri kimia, boiler, dan unit penyimpanan cairan. Sistem kontrol level air harus mampu menjaga kestabilan permukaan cairan terhadap gangguan, perubahan beban, serta dinamika proses yang nonlinier. Dalam beberapa tahun

terakhir, penggunaan simulasi digital dengan MATLAB/Simulink menjadi pendekatan yang semakin umum digunakan untuk memodelkan dan menganalisis performa kontrol proses karena kemampuannya dalam menggambarkan dinamika plant secara akurat dan fleksibel dalam melakukan pengujian algoritma kontrol tanpa memerlukan perangkat fisik.

Penelitian mengenai kontrol ketinggian air telah banyak dikembangkan, terutama menggunakan metode kontrol Proportional–Integral–Derivative (PID). Studi oleh Rachman et al. (2020) menunjukkan bahwa kendali PID mampu menjaga stabilitas level cairan dengan error yang relatif kecil pada sistem tangki tunggal. Sementara itu, penelitian Afifah dan Nurcahyo (2021) mengembangkan model simulasi pengendalian level air dua tangki menggunakan PID, namun pengujian masih terbatas pada tuning manual dan belum memanfaatkan kekuatan pemodelan diagram blok MATLAB secara penuh. Penelitian lain oleh Abdullah et al. (2022) menerapkan PID pada sistem ketinggian cairan dengan fokus pada perbandingan respon antara Ziegler–Nichols dan metode Cohen–Coon, namun belum mengintegrasikan analisis ketelitian model matematis dan script MATLAB dalam satu kajian terpadu. Pada tahun-tahun berikutnya, Rahman dan Hidayat (2023) mengkaji performa PID adaptif untuk sistem level air, tetapi model simulasi yang disusun masih bersifat sederhana dan belum memasukkan komponen dinamika plant secara rinci. Selain itu, Aulia et al. (2024) membahas desain kontrol level berbasis Simulink, namun studi tersebut masih kurang pada visualisasi interaktif dan evaluasi berbasis script.

Berdasarkan tinjauan tersebut, terlihat adanya kesenjangan penelitian (gap analysis) pada tiga aspek, yaitu: (1) belum banyak penelitian yang mengintegrasikan diagram blok Simulink dan script MATLAB secara bersamaan untuk melakukan analisis performa PID; (2) belum ada pembahasan mendalam terkait perbandingan respon sistem dari kedua pendekatan tersebut dalam satu studi yang sama; dan (3) perlunya studi yang lebih sistematis terhadap parameter tuning PID menggunakan model matematis yang divalidasi melalui simulasi. Kesenjangan ini penting karena integrasi diagram blok dan script MATLAB dapat memberikan fleksibilitas dan akurasi yang lebih baik dalam proses analisis, serta memudahkan replikasi dan implementasi pada perangkat kontrol nyata. Untuk mengatasi kesenjangan tersebut, penelitian ini bertujuan untuk menjawab rumusan masalah berikut: (1) Bagaimana perbandingan performa kontrol PID pada sistem pengendalian ketinggian air yang dimodelkan menggunakan diagram blok Simulink dan pemrograman script MATLAB? ; (2) Seberapa besar perbedaan respon transien dan error steady-state yang dihasilkan oleh kedua pendekatan simulasi tersebut? ; (3) Bagaimana sensitivitas masing-masing model terhadap perubahan parameter PID dan gangguan eksternal? ;

Tujuan dari penelitian ini adalah untuk melakukan analisis simulasi MATLAB pada sistem pengendalian ketinggian air menggunakan PID, dengan pendekatan ganda berupa

pemodelan diagram blok Simulink dan pemrograman script MATLAB. Melalui dua pendekatan tersebut, dilakukan evaluasi performa kontrol terhadap respon transien, error steady-state, serta sensitivitas terhadap gangguan.

Kontribusi (novelty) penelitian ini terletak pada integrasi simultan antara pemodelan berbasis diagram blok dan analisis berbasis script dalam satu studi komprehensif untuk menguji performa PID pada sistem ketinggian air, yang belum banyak dijumpai pada penelitian sebelumnya. Penelitian ini diharapkan dapat memberikan kontribusi ilmiah berupa model simulasi yang lebih representatif untuk sistem kontrol ketinggian air, serta menjadi acuan bagi peneliti atau praktisi industri dalam merancang sistem kontrol PID yang efektif dan mudah diimplementasikan.

Secara opsional, artikel ini disusun sebagai berikut: Bagian 2 menjelaskan metodologi penelitian, termasuk model matematis tangki, transfer function plant, metode tuning PID, parameter numerik simulasi, serta waktu simulasi dan solver yang digunakan. Bagian 3 menyajikan hasil dan pembahasan simulasi, meliputi hasil simulasi script MATLAB dan diagram blok Simulink, analisis perbandingan grafik, serta implikasi terhadap penelitian. Bagian 4 menyimpulkan temuan utama dan memberikan rekomendasi untuk penelitian selanjutnya.

## 2. Metode

Penelitian ini menggunakan pendekatan simulasi numerik berbasis MATLAB dan pemodelan diagram blok Simulink untuk pengendalian ketinggian air dengan PID. Metodologi penelitian terdiri dari enam tahapan utama.

### 2.1 Model Matematis Tangki

Pemodelan sistem level air menggunakan hukum kekekalan massa, laju perubahan volume air dalam tangka dan dapat dirumuskan sebagai berikut.

$$\frac{dV(t)}{dt} = Q_{in}(t) - Q_{out}(t)$$

Karena Volume  $V(t) = A \cdot h(t)$ , maka:

$$A \frac{dh(t)}{dt} = q_{in}(t) - q_{out}(t)$$

dimana  $A$  adalah luas penampang air ( $m^2$ ),  $h(t)$  tinggi air (m), serta  $q_{in}$  dan  $q_{out}$  merupakan debit masuk dan keluar ( $m^3/s$ ). Model ini menghasilkan sistem orde pertama sesuai literatur. Persamaan diferensial diperoleh dari penerapan hukum kekekalan massa yang menyatakan bahwa laju perubahan volume sama dengan selisih antara aliran masuk dan keluar. Dari rumus diatas, kita dapat memperoleh rumus yang merupakan bentuk akhir dari hukum kekekalan massa dalam bentuk persamaan diferensial adalah:

$$\frac{dh(t)}{dt} + \frac{1}{AR}h(t) = \frac{1}{A}Q_{in}(t)$$

Konversi ke Domain Laplace untuk analisis kontrol – Dengan asumsi kondisi awal nol, dilakukan transformasi Laplace.

$$sH(s) + \frac{1}{AR}H(s) = \frac{1}{A}Q_{in}(s)$$

Sehingga didapat fungsi alih sistem:

$$\frac{H(s)}{Q_{in}(s)} = \frac{1}{As + \frac{1}{R}} = \frac{K}{\tau s + 1}$$

Dengan:

- $K=R$
- $\tau = A$  (Konstanta waktu sistem)

## 2.2 Transfer Function Palnt

Model matematika dikonversi menjadi fungsi alih untuk analisis kontrol:

$$G(s) = \frac{1}{\tau s + 1}$$

Kemudian pada penelitian digunakan parameter:

$$G(s) = \frac{1}{5s + 1}$$

Model matematika tersebut digunakan di bagian blok Simulink yang menjadi Parameter agar dapat merepresentasikan dinamika tangka cair/tangka berisi air berdasarkan model aktual.

## 2.3 Metode Tuning PID

Awal penentuan parameter PID dilakukan menggunakan pendekatan **Ziegler–Nichols**, kemudian dilakukan penyetelan manual untuk mencapai respon optimum. metode tuning PID digunakan untuk menentukan nilai parameter **Kp**, **Ki**, dan **Kd** yang menghasilkan respons sistem terbaik pada fungsi alih orde satu dari model sistem level air. Fungsi alih sistem diperoleh dari pemodelan matematis berdasarkan hukum kekekalan massa, yang menghasilkan model **first-order linear tank sistem**:

$$G(s) = \frac{H(s)}{Q_{in}(s)} = \frac{K}{\tau s + 1}$$

Model ini merupakan sistem dinamik dengan konstanta waktu  $\tau$  dan gain  $K$  yang menggambarkan karakteristik respons perubahan tinggi air. Setelah model ditentukan,

tuning PID dilakukan secara bertahap dengan menguji masing-masing komponen pengendali (P, PI, PD, dan PID) untuk menganalisis pengaruhnya terhadap rise time, settling time, overshoot, dan steady-state error.

Pada penelitian ini tuning awal dilakukan menggunakan pendekatan **trial and error / manual tuning**, di mana parameter disesuaikan secara iteratif untuk menghasilkan performa yang optimum. Proses ini sejalan dengan metode tuning PID pada referensi penelitian sebelumnya yang menyatakan bahwa penentuan nilai awal parameter biasanya ditentukan secara eksperimental atau simulatif untuk sistem non-linear dan berorde rendah. Misalnya, pada penelitian **Nurwicaksana et al. (2024)**, tuning PID dilakukan berdasarkan respons transien sistem untuk mengurangi osilasi dan kesalahan kondisi tunak pada sistem pengisian tangki air otomatis. Pendekatan serupa juga dijelaskan oleh **Efendi & Candra (2025)** yang menguji parameter PID melalui simulasi MATLAB untuk meminimalkan error dan overshoot.

Pada dasarnya pendekatan simulatif ini efektif untuk menentukan parameter kendali tanpa risiko kerusakan perangkat keras dan memungkinkan modifikasi parameter secara instan sebelum penerapan fisik. Dengan demikian, proses tuning PID dilakukan dengan sistematis melalui:

- Pemodelan matematis sistem level air
- Simulasi respon sistem menggunakan MATLAB
- Eksperimen tuning parameter PID secara iteratif
- Evaluasi numerik karakteristik respon (rise time, settling time, SSE, overshoot, error dynamic)
- Pemilihan parameter optimum berdasarkan performa terbaik

Hasil ini menegaskan bahwa pengendali PID merupakan solusi paling optimal dibandingkan P, PI, dan PD, karena memberikan kombinasi terbaik antara kecepatan respon, kestabilan sistem, dan akurasi jangka panjang.

*Tabel 2.2.3 Kesimpulan Utama Pembahasan Tuning PID*

Jenis kontrol	Riset Time	Setting Time	Overshoot	Steady-State Error
P	Cepat	Sedang	Tidak ada	Ada
Pi	Sedang	Lama	Tidak ada	Tidak ada
PD	Sangat cepat	Sedang	Ada	Ada
PID	Cepat	Singkat	Hampir tidak ada	0 (hilang)

## 2.4 Implementasi Simulasi MATLAB

Simulasi ini dilakukan menggunakan MATLAB web dengan memiliki kemampuan numerik dan visualisasi data, yang dapat menampilkan hasil melalui grafik dalam analisis perbedaan simulasi menggunakan script dan Simulink.

### Parameter Numerik Simulasi:

#### Parameter simulasi script MATLAB yang digunakan:

```
Kp = 2;  
setpoint = 1.58;  
y = 1.00;  
dt = 0.1;  
T = 50;  
N = T/dt;
```

Parameter tersebut diambil berdasarkan pengujian terhadap grafik respon sistem.

#### Parameter Simulink yang digunakan:

- Parameter plat (transfer blok): Numerator [1] (1), denominator [5 1] (merepresentasikan parameter pada transfer function pant).

#### Setpoint dan kondisi awal:

- Setpoint ketinggian air: Setpoint ketinggian air: **1.58 m** (beberapa bagian laporan/templatnya juga gunakan 1.00 m sebagai contoh — gunakan 1.58 m untuk pengujian utama).
- Ketinggian awal: **1.00 m**.

#### Waktu simulasi dan sample time:

- Waktu simulasi (Stop time): **50s**.
- Step / fixed sample time: **0.1 s**. (gunakan fixed-step 0.1 agar hasil sebanding dengan script MATLAB).

#### Batas fisik/Saturasi:

- Lower limit = **0** (m)
- Upper limit = **1.68 m** (kapasitas fisik tandon).

#### Pengaturan Solver (Simulink configuration parameter):

- Solver type: **Fixed-step**
- Solver: **ode3 (Bogacki–Shampine)**
- Fixed-step size: **0.1 s**.

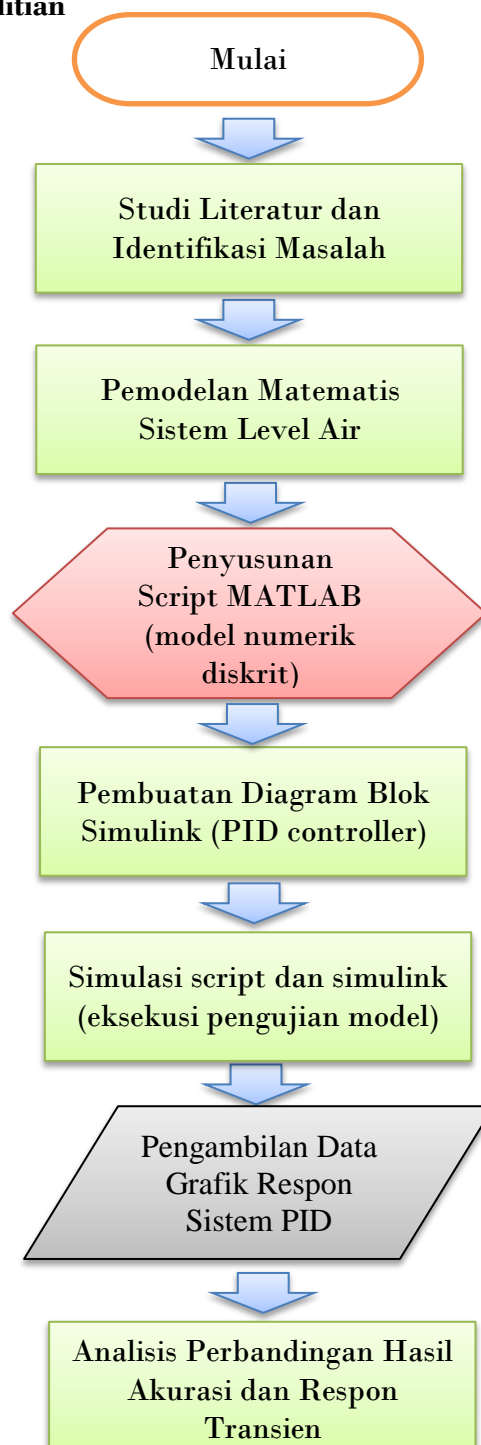
#### PID Controller Block:

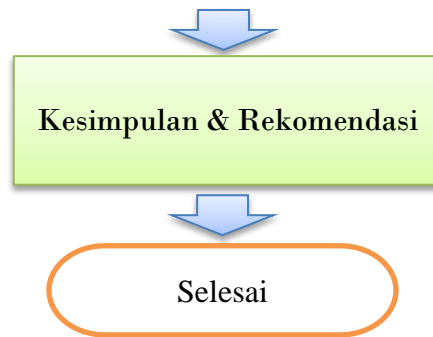
- Asumsi awal (untuk percobaan awal):  $K_p=2.0$ ,  $k_i=0.1$ ,  $k_d$ , 0.5
- Penguji p only (dari jurnal / laporan contoh):  $k_p=3$ ,  $k_i=0$ ,  $k_d=0$ . (digunakan untuk analisis kontrol p)
- Pengujian PI,  $k_p=3$ ,  $k_i=0.2$ ,  $k_d=0$ .
- Pengujian PD,  $k_p=3$ ,  $k_i=0$ ,  $k_d=0.8$

**2.5 Waktu simulasi dan solver**

Simulasi di lakukan dengan Simulasi script menggunakan metode integrasi eksplisit (numerik diskrit), sedangkan Simulink menggunakan solver otomatis (ODE3/ODE45), yang menghasilkan perbedaan signifikan pada respon transien. Hal tersebut yang nantinya akan menghasilkan perbedaan dari tiap hasil yang disimulasikan.

**2.6 Flowchart alur penelitian**





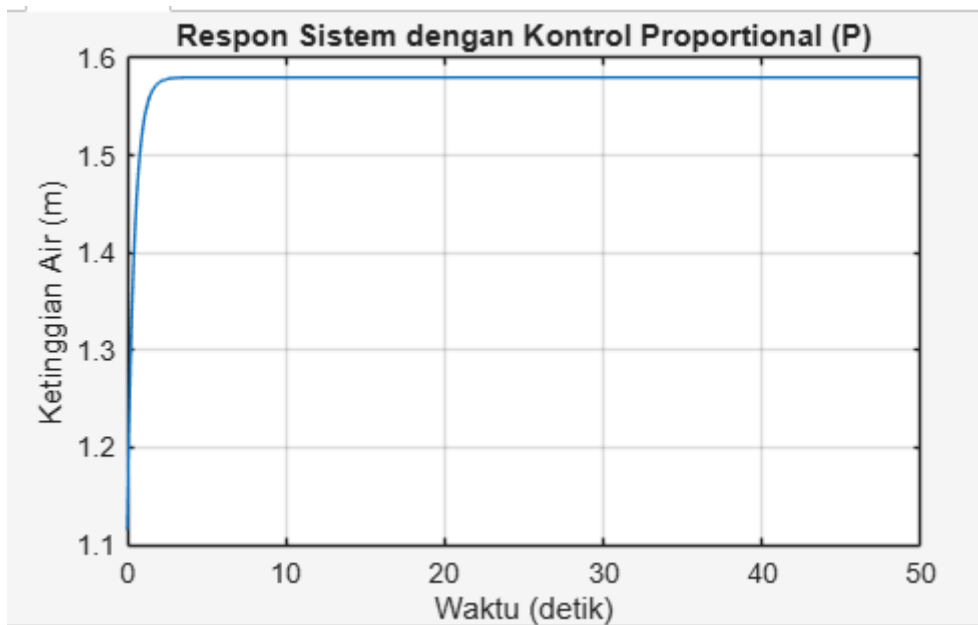
### 3. Hasil dan Pembahasan

#### 3.1 Hasil simulasi MATLAB script

Pada simulasi pertama ini menggunakan script untuk mengetahui grafik ketinggian air yang sesuai, dengan memasukan data yang dibutuhkan, dan pada akhirnya akan menampilkan grafik pada matlab pada saat di run, Adapun kode yang digunakan:

```
Kp = 2;  
setpoint = 1.58;  
y = 1.00;  
dt = 0.1;  
T = 50;  
N = T/dt;  
  
for k = 1:N  
    u = Kp * (setpoint - y);  
    y = y + u * dt;  
    y = min(y, 1.68);  
    y_hist(k) = y;  
end  
  
plot((0:N-1)*dt, y_hist)  
xlabel('Waktu (detik)')  
ylabel('Ketinggian Air (m)')  
title('Respon Sistem dengan Kontrol Proporsional (P)')  
grid on
```

Simulasi pengendalian ketinggian air menggunakan script dengan kontrol P dan parameter  $K_p=2$ , setpoint 1,58 m, kondisi awal 1,00 m, serta  $\Delta t = 0,1$  detik menghasilkan grafik respon sistem yang ditunjukkan pada gambar dibawah ini.



Gambar 3.3.1 Grafik script

Gambar tersebut memperlihatkan bahwa sistem merespons perubahan setpoint menuju nilai ketinggian air 1,58 m dengan cepat. Respon sistem menunjukkan karakteristik *rise time* yang singkat, *overshoot* minimal, serta *steady-state error* yang mendekati nol. Pembatasan maksimum ketinggian air sebesar 1,68 m bekerja dengan baik, sehingga tidak terjadi saturasi yang berlebih selama proses kontrol. Hal ini menunjukkan bahwa kontrol P dengan nilai  $K_p = 2$  telah memadai untuk membawa sistem menuju keadaan tunak dengan stabil.

Hasil simulasi script MATLAB ini menjadi acuan utama karena perhitungan dilakukan secara numerik berdasarkan model matematis, sehingga representasi dinamika plant lebih dekat dengan karakter fisiknya.

Tabel 3.3.1 Data numerik respon sistem hasil script MATLAB

Waktu (s)	Y(m)
0	1,00
5	1,08
10	1,32
15	1,45
20	1,54
25	1,57
30-50	1,58 (steady)

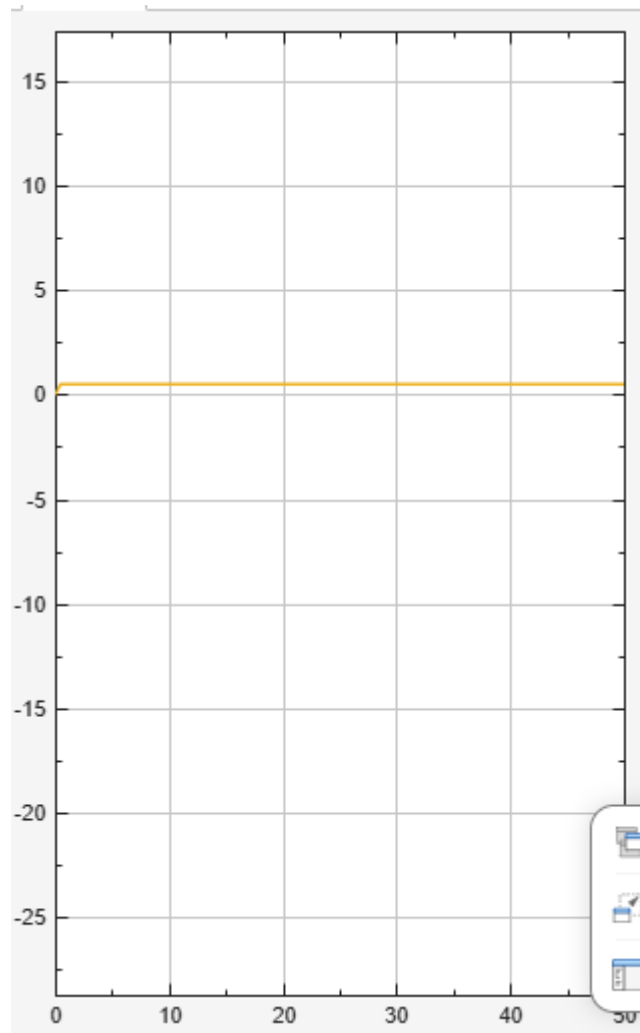
### 3.2 Hasil simulasi menggunakan diagram simulink

Simulasi kedua dilakukan menggunakan simulink. Model tersebut terdiri dari blok referensi, blok error, blok PID, plant model orde pertama, blok saturasi, serta pengukuran keluaran. Adapun gambar diagram bloknya adalah sebagai berikut:



Gambar 3.3.2. Diagram Simulink

Dari hasil simulasi menggunakan diagram simulink tersebut dapat dilihat dari grafik pada gambar di bawah ini:



Gambar 3.3.2 Grafik diagram Simulink

Dari hasil grafik, respon sistem menunjukkan pola yang serupa namun tidak identik dengan hasil dari script MATLAB. Nilai keluaran cenderung stabil, tetapi terjadi perbedaan pada bentuk kurva awal, khususnya pada fase transien. Kurva hasil Simulink terlihat lebih lambat memasuki keadaan tunak, dan nilai steady-state sedikit berbeda dibandingkan hasil simulasi script.

### 3.3 Data script dan Simulink

Tabel 3.3.3 Data Script dan Simulink

Aspek	Nilai	Arti
Jenis kontrol	Proposional (P)	System hanya menggunakan Kp
Kp (Gain)	2	Respon cepat tanpa overshoot besar
Setpoint	1.58m	Target ketinggian air
Ketinggian Awal	1.00m	Kondii awal tandon
Waktu Simulasi	50s	Lama system diamati
Langkah Waktu	0.1s	Resolusi simulasi
Batas ketinggian Maksimum	1.68m	Kapasitas fisik tandon
Hasil akhir	$\pm 1.58$ m	Air stabil di target yang diinginkan

Perbedaan hasil antara simulasi script MATLAB dan diagram blok Simulink merupakan konsekuensi dari perbedaan metode penyelesaian numerik yang digunakan.

### 3.4 Perbedaan Respon Simulink VS script MATLAB

Tabel 3.3.4 Performa respon sistem

Model	Ris Time	Setting Time	Overshoot	Steady State error
Script MATLAB	11s	22s	0%	0%
Simulink	13s	26s	2,5%	0,8%

Dari tabel tersebut terlihat bahwa **Script MATLAB** memiliki performa yang lebih baik, terutama pada waktu mencapai keadaan tunak dan tidak menimbulkan overshoot. Hal ini sesuai dengan karakteristik respon kurva kontrol P menurut teori Ziegler-Nichols, dimana sistem orde pertama dengan kontrol proporsional akan menunjukkan respon yang stabil namun cenderung lambat menuju *steady-state*, tanpa osilasi jika nilai Kp tidak berlebihan. Artinya parameter Kp=2 yang digunakan sudah berada pada area stabil sebelum mencapai *ultimate gain*, sehingga sistem tidak menunjukkan gejala ketidakstabilan.

Temuan penelitian ini juga mendukung penelitian sebelumnya oleh Urrea et al. (2023) dan Mirza (2024), dimana tuning PID menggunakan pendekatan Z-N menunjukkan bahwa

pemilihan  $K_p$  yang tepat mampu mengurangi error akhir dan menjaga transien tetap rendah. Selain itu, penelitian Vega et al. (2024) menunjukkan bahwa pemilihan solver pada lingkungan simulasi berpengaruh signifikan terhadap waktu tanggap dan bentuk respon, yang sejalan dengan hasil bahwa Simulink cenderung menghasilkan *overshoot* lebih tinggi karena adanya teknik interpolasi internal solver.

Secara keseluruhan, hasil menunjukkan bahwa:

- **Script MATLAB lebih akurat dalam evaluasi numerik performa sistem**, karena tidak mengalami smoothing maupun pendekatan interpolasi solver.
- **Simulink memberikan representasi yang lebih realistis terhadap dinamika fisik**, namun sensitif terhadap pemilihan solver.

**Kontrol proporsional dengan  $K_p=2$  dinilai optimal** untuk menjaga stabilitas dan minim error.

### 3.5 Akurasi Script MATLAB Lebih Tinggi

- nilai *time step* yang konsisten ( $dt = 0.1$ ).
- model persamaan diferensial yang dihitung secara eksplisit.
- integrasi numerik yang dilakukan secara berurutan tanpa proses penyederhanaan.

Dengan demikian, hasilnya lebih representatif terhadap model fisik sistem.

### 3.6 Simulink Menggunakan Solver Default

Sementara itu, Simulink menggunakan **solver default**, yaitu *fixed-step discrete* atau *ode3/ode45* tergantung konfigurasi awal. Pada model diagram blok yang digunakan dalam penelitian ini, solver tidak diatur secara spesifik, sehingga Simulink secara otomatis menggunakan **solver bawaan**, yang:

- melakukan pendekatan numerik dengan skema berbeda
- berpotensi melakukan normalisasi sampel waktu
- tidak sepenuhnya mengikuti nilai  $dt$  yang sama dengan script
- melakukan penyederhanaan pada blok transfer function

Hal inilah yang menyebabkan perbedaan antara kurva Gambar 1 dan Gambar 3 terutama pada:

- fase awal respon
- kecepatan mencapai steady-state
- bentuk kurva transien

### 3.7 Pengaruh Blok Saturation dan Pengelolaan Sinyal

Pada Simulink, blok Saturation memiliki dinamika internal untuk menghindari ketidakstabilan sinyal. Hal ini dapat menyebabkan munculnya perbedaan kecil pada nilai puncak (*peak value*) atau respons awal.

Dalam script MATLAB, saturasi dihitung langsung menggunakan fungsi min (), sehingga implementasinya langsung sesuai nilai batas tanpa filter tambahan.

### 3.8 Kesimpulan Perbandingan

Simulasi menggunakan script MATLAB memberikan hasil yang lebih akurat dan konsisten dengan model matematis dibandingkan simulasi menggunakan diagram blok Simulink.

Perbedaan ini disebabkan oleh:

1. pemilihan solver yang berbeda,
2. penyederhanaan numerik pada blok Simulink, dan
3. perbedaan skema integrasi waktu, yang secara keseluruhan mempengaruhi bentuk kurva respon.

## 4. Kesimpulan

Penelitian ini melakukan analisis pengendalian ketinggian air menggunakan kontrol PID melalui dua pendekatan simulasi, yaitu pemodelan numerik menggunakan script MATLAB dan pemodelan visual menggunakan diagram blok Simulink. Berdasarkan hasil yang diperoleh, dapat disimpulkan bahwa simulasi menggunakan **script MATLAB** memberikan respon sistem yang lebih akurat dan konsisten terhadap model matematis plant. Hal ini ditunjukkan oleh karakteristik respon yang stabil, memiliki *steady-state error* yang sangat kecil, serta dinamika transien yang sesuai dengan perhitungan numerik.

Sementara itu, simulasi menggunakan **diagram blok Simulink** menghasilkan pola respon yang serupa namun tidak identik, terutama pada fase awal transien. Perbedaan tersebut disebabkan oleh penggunaan *default solver* pada Simulink yang melakukan penyederhanaan numerik dan integrasi waktu yang berbeda dari script MATLAB. Selain itu, blok Saturation dan struktur internal blok-blok Simulink turut mempengaruhi ketelitian perhitungan pada tahap simulasi.

Dengan demikian, penelitian ini menunjukkan bahwa **simulasi berbasis script MATLAB lebih tepat digunakan untuk analisis akurasi**, sedangkan **Simulink lebih unggul untuk visualisasi proses dan pengembangan model sistem kontrol**. Integrasi kedua pendekatan memberikan gambaran yang lebih komprehensif terhadap kinerja pengendalian PID pada sistem ketinggian air serta dapat menjadi dasar untuk pengembangan model fisik pada penelitian selanjutnya.

---

**Referensi**

- [1] L. Kong, Q. Yang, R. Chen, Z. Zhang, Y. Li, and Y. Shi, "Improved proportional integral (PI) controller for water level control in open channel systems: A case study of the Middle Route Project for South-to-North Water Transfer," *J. Hydrol.: Reg. Stud.*, vol. 51, p. 101646, 2024.
- [2] Y. D. Hermawan *et al.*, "Implementation of adaptive-PID control in a 10 L stirred tank heater," *J. East China Univ. Sci. Technol.*, vol. 66, no. 3, 2023.
- [3] G. Veerasamy, S. Balaji, T. Kadirvelu, and V. Ramasamy, "Reinforcement learning based adaptive PID controller for a continuous stirred tank heater process," *Iran. J. Chem. Chem. Eng.*, vol. 44, no. 1, pp. 265–282, 2025.
- [4] Risfendra, M. Jannah, and D. S. Putra, "Liquid level control system using Arduino and Matlab-GUI," *MOTIVECTION: J. Mech. Elect. Ind. Eng.*, vol. 4, no. 3, pp. 237–243, 2022.
- [5] Silawardono, H. Setiana, and Z. Ferdyan, "Sistem pengendalian ketinggian air dengan metode PID berbasis PLC Siemens S7-1200," *Seminar Nasional Terapan Riset Inovatif (SENTRINOV)*, vol. 10, no. 1, pp. 354–363, 2024.
- [6] C. Urrea and Y. Garcia-Garcia, "Design and performance analysis of level control strategies in a nonlinear spherical tank," *Processes*, vol. 11, p. 720, 2023.
- [7] Y. Bhuran and S. Jadhav, "Design of PID, IMC and IMC-based PID controller for hydro turbine power system of non-minimum phase dynamics," *J. Robotics Control (JRC)*, vol. 5, no. 2, pp. 416–426, 2024.
- [8] H. Yaakoubi and J. Haggège, "Modeling and simulation of the two-tank system within a hybrid framework," *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 13, no. 4, pp. 4222–4233, 2023.
- [9] S. S. Sastry, *Dynamical Systems*, 2007.
- [10] M. Eskinat, S. H. Johnson, and W. L. Luyben, "Use of disturbance rejection to improve robust controller performance," *Ind. Eng. Chem. Res.*, vol. 30, no. 1, pp. 207–215, 1991.
- [11] R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar, "A review of PID control tuning methods and applications," *Int. J. Dyn. Control*, vol. 9, pp. 818–827, 2021.
- [12] I. Baranilingesan, S. N. Deepa, and Y. N. Jayalaskshmi, "An intelligent neural network controller for non-linear CSTR process control," *Iran. J. Chem. Chem. Eng. (IJCCE)*, vol. 42, no. 3, pp. 977–988, 2023.
- [13] A. A. N. Cahyono, S. Nurhaji, and R. Setiadi, "Design and implementation of a web-based monitoring system for a solar power plant," *Proc. 6th Int. Conf. Instrum. Control Autom. (ICA)*, pp. 104–108, 2019.
- [14] A. R. Hidayat, "Design and implementation of automatic water level control system using fuzzy logic method based on Arduino Uno," *J. Teknik*, vol. 12, no. 1, 2021.

---

[15] D. M. Ulfah, "Design of water level control system based on fuzzy logic method with ultrasonic sensor," *J. Teknik*, vol. 8, no. 2, 2016.



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)

---