APPLICATION OF CONTROL SYSTEM TECHNIQUES FOR PARAMETER IDENTIFICATION AND ERROR ANALYSIS IN TIME-DELAY SYSTEMS

Nderim ZEQIRI[1], Ferit IDRIZI[2], Asan IDRIZI[3]

[1*]*University of Tetova, Faculty of Applied Sciences, p.n. 1200, Tetove, nderim.zeqiri@unite.edu.mk*
[2]*University of Tetova, Faculty of Applied Sciences, p.n. 1200, Tetove, ferit.idrizi@unite.edu.mk*
[3]*University of Tetova, Faculty of Applied Sciences, p.n. 1200, Tetove, asan.idrizi@unite.edu.mk*
[*]*Corresponding author e-mail: nderim.zeqiri@unite.edu.mk*

Abstract

This paper focuses on the application of control system techniques for the identification and determination of key parameters of integrating systems. Parameters such as gain, time constants, and disturbance characteristics play a critical role in controller design, influencing stability margins, transient response, and robustness. By employing systematic methods from control theory—such as system identification, modeling, and feedback analysis—engineers can obtain reliable parameter estimates that enable the design of effective control strategies. Time-delay systems are processes where the output reacts to the input after some delay. These systems appear in many engineering applications, like chemical reactors, heat exchangers, and transport processes. Time-delay systems are processes where the output reacts to the input after some delay. These systems appear in many engineering applications, like chemical reactors, heat exchangers, and transport processes. Knowing the main parameters, such as delay time, system gain, and dynamic behavior, is important for stable and efficient control. Modern software tools and programming languages, such as C++ and Matlab, can be used to simulate, analyze, and manage these time delays, allowing engineers to design and test controllers before applying them to real systems. Without accurate parameters, controllers may not work properly, causing instability or poor performance. The study highlights the importance of precise parameter determination in improving system performance and ensuring stable operation in practical integrating processes.

Keywords: Identification, error, parameter, control system techniques, time-delay systems.

Introduction

Time-delay systems are dynamic processes where the effect of an input on the output occurs after a certain period of time. These delays often arise due to the physical transport of materials, signal processing, measurement lags from sensors, or mechanical inertia in actuators. Time delays are common in industrial systems such as chemical reactors, heat exchangers, conveyor belts, and other transport processes. They can significantly affect the stability and performance of control systems, making the identification of system parameters essential for reliable operation.
Control system techniques are applied to accurately determine the key parameters of time-delay systems. These parameters typically include system gain**,** time constants**,** delay time, and sometimes higher-order dynamic coefficients depending on the system complexity. Identifying these parameters allows engineers to model the system accurately and design controllers that can compensate for delays, ensuring stable and predictable behavior. Modern control techniques can be implemented using algorithms such as PID (Proportional-Integral-Derivative) control**,** model predictive control (MPC)**,** or adaptive control, which can adjust to parameter variations and external disturbances.
These control techniques can be implemented in software using tools like Matlab for simulation and analysis, or in embedded systems using programming languages like C++ for microcontrollers or industrial computers. In this context, system parameters correspond not

only to physical characteristics of the process (such as fluid levels, motor positions, or temperatures), but also to digital representations suitable for computation, including sampling times, sensor sensitivities, actuator limits, and signal processing delays. Properly identifying these parameters ensures that the control system can generate stable outputs, compensate for delays, and operate safely within the physical limits of sensors and actuators.

By applying these techniques, time-delay systems can be managed reliably. Accurate parameter identification allows for the implementation of stable control loops, even in the presence of long or variable delays. It ensures that the system responds correctly to inputs, minimizes oscillations, and optimizes performance.

Determination of appropriate control parameters for managing delays in control systems

In modern industrial automation and control systems, the accurate management of command parameters and the determination of time delays between terminals constitute key elements for the efficient and stable operation of the process. Command parameters represent values that determine the behavior of a system, such as voltage, current, position, speed, pressure or response time.

A time-delay control system is a type of system where the output reacts to the input after a certain delay. This delay can happen for many reasons, like physical transport of materials, slow sensors, signal processing, or mechanical inertia of actuators. These delays are common in engineering applications such as chemical plants, heating systems, conveyor belts, or motor control.

This system is necessary because, without proper control, the delay can cause instability, oscillations, or poor performance. A time-delay control system helps to manage the process so that the output reaches the desired value at the right time, even when the system responds slowly.

The system works by measuring the output, comparing it to the desired value, and adjusting the input to correct any difference. To do this effectively, we need to know certain key parameters of the system, such as the delay time, system gain, time constants, and characteristics of sensors and actuators. These parameters allow engineers to design controllers that can respond properly and keep the system stable.

During the design of a time-delay control system, it is important to consider:

- [1] The length of the delay and its variation.
- [2] System gain and dynamics, to know how strongly the system reacts.
- [3] Sensors and actuators limits, so the control actions do not exceed their capacity.
- [4] Choice of control algorithm, such as PID, model predictive or adaptive control, to match the system behavior.
- [5] Simulation and testing, often done in software like Matlab or C++, to verify that the system will work safely before implementation.

By understanding and applying these principles, engineers can design a control system that handles delays effectively, keeps the process stable, and ensures good performance.

Determining values for a time-delay system

Assumption: the system has three coefficients K1, K2, and K3 for the components of the controller:

[1] K1 → proportional error (Proportional)

[2] K2 → accumulated error (Integral)

[3] K3 → change in error (Derivative)

The formula for the Control Action column is: =K1*F4 + K2*G4+ K3*H4, or through the cells of Microsoft Excel "=F$16*F4+F$17*G4+F$18*H4" where: F4 - current error, G4 - cumulative error, H4 - change in error.

The values of the coefficients are assigned in the Microsoft Excel cells as follows:

1. K1 = 0.8 (cell F16)
2. K2 = 0.2 (cell F17)
3. K3 = 0.1 (cell F18)

In continuation, the study presents the generated values according to the requirements of the problem of defining parameters based on the arguments presented in Table 1.

Table 1- General controller table for time-delay system

A	B	C	D	E	F	G	H	I
	Time (t)	Input (u)	Output (y)	Desired_output_y (y_desired)	Error (y_desired-y)	Cumulative Error (ΣError)	Change in Error (ΔError/Δt)	Control action (u_control)
1	1	0.01	0.02	1.02	0.01	0.01	0	0.01
2	2	0.02	0.04	1.03	0.02	0.03	1	0.122
3	3	0.03	0.06	1.04	0.03	0.06	1	0.136
4	4	0.04	0.08	1.05	0.04	0.1	1	0.152
5	5	0.05	0.1	1.06	0.05	0.15	1	0.17
6	6	0.06	0.12	1.07	0.06	0.21	1	0.19
7	7	0.07	0.14	1.08	0.07	0.28	1	0.212
8	8	0.08	0.16	1.09	0.08	0.36	1	0.236
9	9	0.09	0.18	1.1	0.09	0.45	1	0.262
	10	0.1	0.2	1.11	0.1	0.55	1	0.29
	…							..

1	K1	0.8
2	K2	0.2
3	K3	0.1

This table provides a structured way to simulate and analyze a time-delay control system. By generating values for the error, cumulative error, change in error, and control action, engineers can understand how the system responds to different inputs and delays. The importance of this process lies in several aspects:

- Parameter Tuning: The table allows engineers to test different values of the controller coefficients (K1, K2, K3) and observe how these changes affect the system response. This helps in finding suitable parameters for stable and efficient control.
- Prediction of System Behavior: By calculating the control action at each time step, the table predicts how the system will respond before actual implementation. This reduces the risk of instability or undesired oscillations in real applications.
- Practical Design Tool: In practice, such tables help in designing, testing, and optimizing controllers without needing to experiment directly on physical systems, which may be expensive or risky.
- Visualization and Analysis: Values generated in the table can be plotted to visualize the system behavior over time, making it easier to detect problems and refine the controller settings.
- Educational and Developmental Use: For students and engineers, this table provides a clear example of how theoretical control concepts are applied in practice, bridging the gap between mathematical formulas and real-world implementation.

The algorithm that represents a generalized closed-loop control process

In the following Figure 1, the algorithm and the signal flow of the system are presented. The algorithm represents a generalized closed-loop control process. It processes system errors and adjusts control parameters to improve system performance and stability over time. The role of the algorithm is presented through the text inside its blocks, describing the signal flow and the updating of data according to the defined steps.

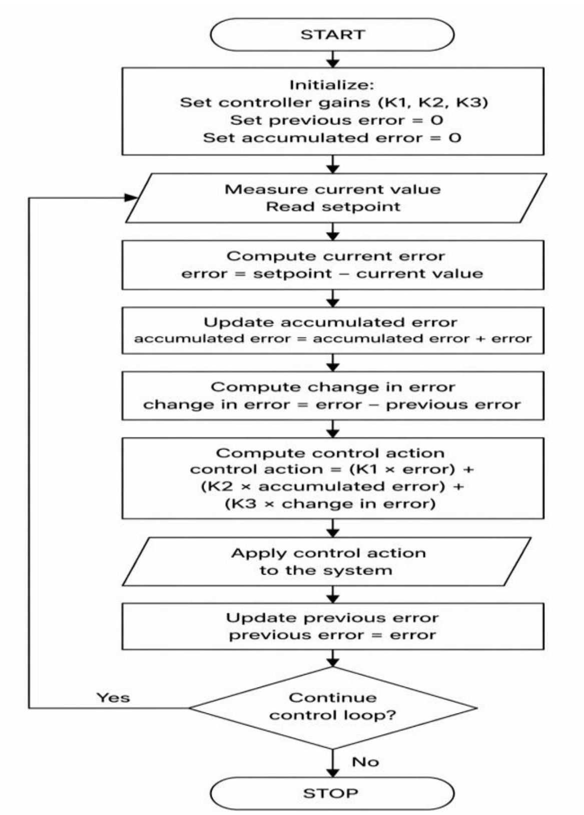

Figure 1- This algorithm represents a generalized closed-loop control process based on the principles of a PID controller

Table 2- C++ program code: the output changes according to the general controller and time parameters & Matlab code-this Matlab script simulates the same logic as the C++ code.

```
// C++ program code: The simulation is performed over 10 time steps. The output changes according to the general controller. The parameters K1, K2, K3, and dt can be modified to observe different system behavior.
#include <iostream>
#include <vector>
using namespace std;
int main() {
// Time simulation parameters
double dt = 1.0; // time step
int steps = 10; // number of time steps
// Controller coefficients
double K1 = 0.8;
double K2 = 0.2;
double K3 = 0.1;
// System vectors
vector<double> time(steps);
vector<double> output(steps, 0.0);
vector<double> desired(steps, 1.0); // desired output
vector<double> error(steps, 0.0);
vector<double> cumError(steps, 0.0);
vector<double> deltaError(steps, 0.0);
vector<double> control(steps, 0.0);
// Simulation loop
for (int i = 0; i < steps; i++) {
time[i] = i * dt;
// Compute error
error[i] = desired[i] - output[i];
// Compute cumulative error
cumError[i] = (i == 0) ? error[i] : cumError[i-1] + error[i];
// Compute change in error
deltaError[i] = (i == 0) ? 0 : (error[i] - error[i-1])/dt;
// Compute control action (general controller)
control[i] = K1*error[i] + K2*cumError[i] + K3*deltaError[i];
// Simple system response simulation
// output changes proportionally to control action (example)
output[i] += control[i]*dt;
// Print results
cout << "t=" << time[i]
<< " y=" << output[i]
<< " e=" << error[i]
<< " u=" << control[i] << endl;
}
return 0;
}
```

```
% Time parameters. Matlab code: This Matlab script simulates the same logic as the C++ code. It generates a plot showing the output and the desired output. The parameters K1, K2, and K3 can be modified to observe different system responses.
dt = 1; % time step
steps = 10; % number of steps
% Controller coefficients
K1 = 0.8;
K2 = 0.2;
K3 = 0.1;
% Initialize vectors
time = (0:steps-1)*dt;
output = zeros(1, steps);
desired = ones(1, steps); % desired output
error = zeros(1, steps);
cumError = zeros(1, steps);
deltaError = zeros(1, steps);
control = zeros(1, steps);
for i = 1:steps
% Compute error
error(i) = desired(i) - output(i);
% Compute cumulative error
if i == 1
cumError(i) = error(i);
deltaError(i) = 0;
else
cumError(i) = cumError(i-1) + error(i);
deltaError(i) = (error(i) - error(i-1))/dt;
end
% Compute control action
control(i) = K1*error(i) + K2*cumError(i) + K3*deltaError(i);
% Simple system response
output(i) = output(i) + control(i)*dt;
end
% Plot results
figure;
plot(time, output, '-o', 'DisplayName', 'Output');
hold on;
plot(time, desired, '--', 'DisplayName', 'Desired Output');
xlabel('Time (s)');
ylabel('Output');
legend;
grid on;
title('Simulation of General Controller');
```

Mathematical model of a closed-loop control system based on error computation and control signal generation

- Error: $e(k)=r(k)-x(k)e(k)$ → The error is the difference between the reference value and the system output.
- Proportional term: $P=Kp\cdot e(k)$ → The proportional part depends on the current error.
- Integral term: $I=Ki\cdot\sum e(k)$ → The integral part is based on the accumulated error over time.
- Derivative term: $D=Kd\cdot(e(k)-e(k-1)$ → The derivative part depends on the change in error.
- Control signal: $u(k)=P+I+D$ → The control signal is the sum of proportional, integral, and derivative terms.

This algorithm represents a generalized closed-loop control process based on the principles of a PID (Proportional–Integral–Derivative) controller. Its purpose is to continuously regulate a system so that the output follows a desired reference value (setpoint) with minimal error. The process begins with initialization, where the controller parameters (gains) are defined and internal variables, such as accumulated error and previous error, are set to zero. These parameters determine how strongly the controller reacts to current, past, and predicted future errors. At each iteration, the system measures the current output and compares it with the desired reference value.

The difference between these two values is defined as the current error. This error is then used in three ways: first, it is considered directly (proportional effect); second, it is accumulated over time (integral effect), which helps eliminate steady-state error; and third, its rate of change is calculated (derivative effect), which improves system stability and anticipates future behavior. The control action is computed as a combination of these three components, each weighted by its respective coefficient. This action is then applied to the system to adjust its behavior.

After applying the control signal, the algorithm updates the stored values (such as previous error) and repeats the cycle. This continuous loop enables real-time correction and adaptation, ensuring that the system remains stable, accurate, and responsive to changes or disturbances.

The following algorithm presents the possibility of accessing the terminal and finding the terminal. This is a very important step, because it deals with the process of identifying terminals, as a necessary condition for accessing external objects of different categories.

Figure 2 shows the algorithm for design and simulation of PID Control (Kp, Ki, Kd), which describes the structure and functioning of the proportional, integral, and derivative parameters used in the control system. This algorithm is essential in automatic control systems because it ensures stability, accuracy, and fast response to changes in the system. By continuously correcting the error between the desired value and the actual output, the PID controller improves performance and reduces oscillations, making it widely used in industrial processes, robotics, and many engineering applications.

This automatic control algorithm (such as a PID controller – Proportional, Integral, Derivative) plays a very important role in modern automation and control systems. Its main purpose is to keep a process or system at a desired state by continuously correcting the deviations between the reference value and the actual output of the system.

The algorithm works by calculating the error $e(k)=r(k)-x(k)$, where r(k)) is the desired value (setpoint) and x(k) is the actual system output. The proportional part reacts immediately to the current error, making the system respond quickly. The integral part considers the accumulated error over time, eliminating steady-state errors and improving long-term accuracy. Meanwhile, the derivative part predicts future changes of the error and helps stabilize the system by reducing oscillations.

Overall, this algorithm is widely used in fields such as industry, robotics, temperature control systems, electric motors, and chemical processes because it ensures stability, accuracy, and fast response to system changes.

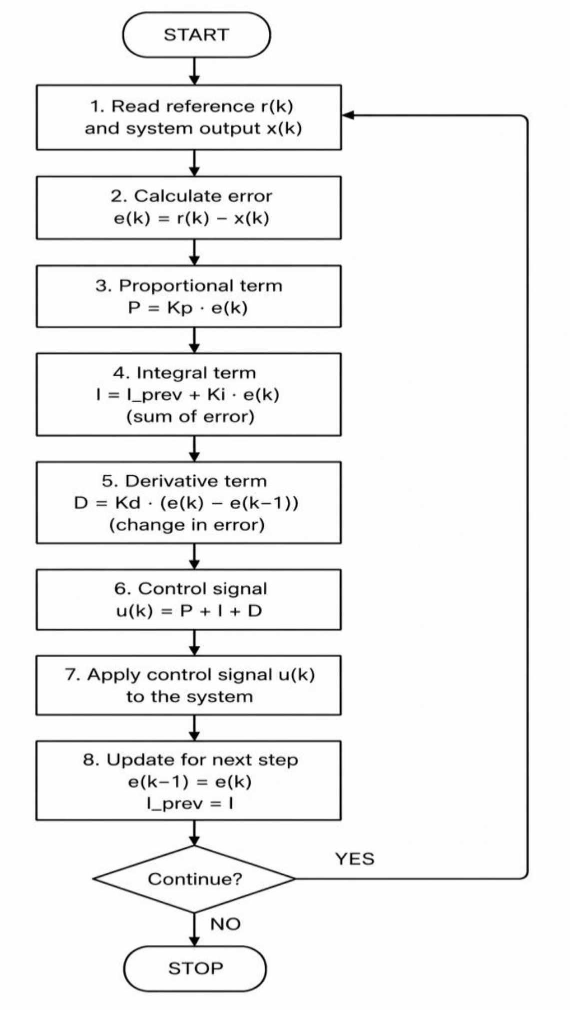

Figure 2- Algorithm for design and simulation of PID Control (Kp, Ki, Kd)

Conclusions

This paper examines the application of control system methods for the estimation and identification of key parameters in integrating processes. Fundamental factors such as system gain, time constants, and external disturbances have a strong impact on controller performance, especially regarding stability, transient response, and overall robustness. By applying structured approaches from control theory, including mathematical modeling, system identification, and feedback analysis, reliable parameter estimation can be achieved, supporting the design of effective control strategies.

Time-delay systems represent dynamic processes where the output response occurs after a certain time lag following the input signal. This type of behavior is widely encountered in engineering applications such as chemical reactors, heat exchangers, and transport systems.

In addition, errors play a critical role in system performance analysis, as they represent the deviation between the desired reference value and the actual system output. These errors may arise from measurement noise, modeling inaccuracies, or external disturbances, and they can significantly affect parameter identification and controller accuracy. Therefore, proper error detection, analysis, and minimization are essential for improving system reliability and achieving precise control behavior.

Moreover, modern simulation tools such as C++ and Matlab offer strong capabilities for modeling, analyzing, and compensating time-delay effects, allowing engineers to evaluate controller behavior before real-world implementation. Incorrect parameter estimation or unaccounted errors may result in reduced efficiency or even system instability. For this reason, the study highlights the importance of accurate parameter identification and systematic error analysis as essential elements for improving system reliability and performance in practical integrating and time-delay applications.

.

References

[1]. Ogata, K. Modern Control Engineering. 5th Edition, 2010. Prentice Hall. ISBN: 9780136156734

[2]. Franklin, G. F., Powell, J. D., Emami-Naeini, A. Feedback Control of Dynamic Systems. 7th Edition, 2015. Pearson. ISBN: 9780133496598

[3]. Nise, N. S. Control Systems Engineering. 7th Edition, 2015. Wiley. ISBN: 9781118170519

[4]. Dorf, R. C., Bishop, R. H. Modern Control Systems. 13th Edition, 2016. Pearson. ISBN: 9780134407623

[5]. Astrom, K. J., Murray, R. M. Feedback Systems: An Introduction for Scientists and Engineers. 2017 printing. Princeton University Press. ISBN: 9780691193984

[6]. Franklin, G. F., Powell, J. D., Emami-Naeini, A. Feedback Control of Dynamic Systems. 7th Edition, 2015. Pearson. ISBN: 9780133496598

[7]. N. Zeqiri. Computer Network Applications, Practical Implementations and Structural Control System Representations. 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).

[8]. N. Zeqiri, E. Rufati, B. Zeqiri. Functional Parameters and Performance Requirement in the Industrial System. JAS-SUT Journal of Applied Sciences – SUT, 5(9–10), 14–21, 2019.

[9]. N.Zeqiri, A. Luma. A/D Conversion, Synchronization and Control and Algorithm for Hardware Realization. Croatian Society for Information and Communication Technology, Electronics and Microelectronics, MIPRO, Croatia. ISBN 977-852-233-044. Pages197-201. 2009.

[10]. N.Zeqiri, F.Idrizi. Software management and algorithms for real-time control systems. Journal of Applied Sciences-JAS-SUT (ISSN: 1857-9930/print & ISSN: 2671-304.2018.

[11]. Seborg, D. E., Edgar, T. F., Mellichamp, D. A., Doyle, F. J. Process Dynamics and Control. 4th Edition, 2016. Wiley. ISBN: 9781119285915.

[12]. Richard, J.-P. Time-Delay Systems: An Introduction and Applications. 2003. Springer. ISBN: 9783540404535.