IntechOpen

PID Control

New Design Methods and Applications

Edited by Constantin Voloşencu

PID Control - New Design Methods and Applications

Edited by Constantin Voloşencu

Published in London, United Kingdom

PID Control - New Design Methods and Applications
http://dx.doi.org/10.5772/intechopen.1003415
Edited by Constantin Voloşencu

Contributors
Constantin Voloșencu, Davarapalli Kishore, Ming-Yen Wei, Sabbarapu Ganesh, Vaska Lokesh, Yuri V. Kim

First published in London, United Kingdom, 2025 by IntechOpen
IntechOpen is the global imprint of INTECHOPEN LIMITED, registered in England and Wales, registration number: 11086078, 167-169 Great Portland Street, London, W1W 5PF, United Kingdom

For EU product safety concerns: IN TECH d.o.o., Prolaz Marije Krucifikse Kozulić 3, 51000 Rijeka, Croatia, info@intechopen.com or visit our website at intechopen.com.

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

PID Control - New Design Methods and Applications
Edited by Constantin Voloşencu
p. cm.
Print ISBN 978-0-85466-476-4
Online ISBN 978-0-85466-475-7
eBook (PDF) ISBN 978-0-85466-477-1

Meet the editor

Prof. Dr. Constantin Voloşencu graduated as an engineer from Politehnica University Timișoara in Romania, where he also obtained a doctorate. He is currently a full professor in the Department of Automation and Applied Informatics at the same university. Prof. Voloşencu is the author of 10 books, 7 book chapters, and more than 170 papers published in journals and conference proceedings. He has also edited 15 books and has 27 patents to his name. He is a manager of research grants, editor-in-chief, and member of international journal editorial boards. He is also a former keynote and plenary speaker, a member of scientific committees, and chair at international conferences. Prof. Volosencu is a science ambassador of LiveDNA and a member of the Asian Council of Science Editors (ACSE). His research areas include control systems, control of electric drives, fuzzy control systems, neural network applications, fault detection and diagnosis, sensor network applications, monitoring of distributed parameter systems, and power ultrasound applications. He has developed automation equipment for machine tools, spooling machines, high-power ultrasound processes, and more.

Contents

Preface

Proportional-Integral-Derivative(PID) control is a basic valuable tool for process control applications due to its simplicity and effectiveness demonstrated in practice. The proportional-integral-derivative control scheme is a popular approach used in programmable logic controllers and distributed control systems to control complex processes and manage dynamic industrial systems. Its use involves tuning the regulators, identifying the parameters of the regulated process, filtering signals, cascade regulation, feedforward control, and gain scheduling. The design aims to reduce control errors, minimize process fluctuations, reduce response time, overshoot, and more. The development of new mathematical concepts, such as artificial intelligence methods, brings new perspectives in a new digitalization context, and it renewed the role of the PID controller in a new computing environment and current challenges in the industry. The book aims to bring to the readers' attention the recent research results of specialists in topics such as tuning methodologies, multivariable control, identification methods, digital tools for controller design, fuzzy PI control, and industrial applications. The book covers some aspects of control theory, including new methodologies, techniques and applications. The book promotes the control theory in practical applications of engineering domains. It is a way to disseminate researchers' contributions in the field. This project presents applications that improve PID control systems' properties and performances in analysis and design by a higher technical level of scientific attainments. Applications which emphasize methodologies used with implementation and commissioning issues are presented. The authors have published work examples and case studies from their field research. The readers get new solutions and answers to questions about the emerging theoretical control theory in engineering applications and their implementation.

In a brief description, the book has five chapters in the domains such as basic theoretical issues in the digital implementation of PID controllers, implementation of PID controllers on digital equipment with application in servo-drive control systems, identification and regulation of complex processes such as chemical processes, design of feedback control systems based on Kalman Busy filter or tuning of fuzzy PI controllers based on equivalence with conventional linear controllers. The introductory chapter briefly introduces the PID control domain and presents some basic considerations regarding the modelling, synthesis, and analysis of digital control systems for the purpose of efficient implementation. The second chapter presents a technical report with the results of designing and developing a servo drive control system for a three-degree-of-freedom platform, with immersive full-view 3D stereoscopic images and joystick commands implemented on a digital signal processor. The third chapter presents a theoretical study on tuning of PID controllers for the control system of slow processes, with modelling, synthesis and analysis in the time domain, thus: an example of using relay feedback, an example of system identification with dedicated software with application at a reactor level control and a procedure for optimization for several tuning methods, such as the Ziegler-Nichols method and others. The fourth chapter presents a theoretical study on the design of a closed control system based on a suboptimal Kalman-Busy filter, with a calculation example, modelling, synthesis, and analysis in the time domain. The fifth chapter presents a theoretical study on tuning fuzzy PI controllers with integration at the output based on equivalence with linear PI controllers designed with three design methods: two specific to fast processes, namely Kessler's criteria and one specific to slow processes, namely the criterion of Ziegler-Nichols, with modelling, synthesis and analysis in the time domain through simulations.

The editor thanks the authors for their excellent contributions to the field and understanding during the editing process. Also, the editor thanks all the editorial personnel involved in this book publication.

Constantin Voloșencu
Department of Automation and Applied Informatics,
"Politehnica" University Timișoara,
Timișoara, Romania

Chapter 1

Introductory Chapter: Basic Considerations for the Digital Implementation of PI Controllers

Constantin Voloşencu

1. General aspects and applications

Linear PI regulators represent the most frequently used regulation solution in practical applications, due to their advantages and the simplicity of their design. These regulators have proven their effectiveness in practice for many years, and they remain in use despite the appearance of other more sophisticated regulation algorithms.

Due to the development of computing techniques, the appearance of high-performance digital equipment and a low cost price, PI regulations are currently implemented digitally, using DSPs or microcontrollers. So, in their design, methods of the regulation systems theory are used both in continuous time and in discrete time. The design phases of a numerical control system are analysis, modeling, stability analysis, synthesis, controller tuning and discretization.

The most common applications of PI regulators are as follows: speed and position regulation, in the control systems of electric drives, in cars, locomotives, ships, machine tools and others, flow and level regulation in hydraulic installations, temperature regulation in thermal installations, regulations in chemical coatings and many others.

Although there are design methods, developed for years and verified in practice, specialists in the field continue to research various aspects of these methods. Below is a brief presentation of some recent works that have as their theme the analysis of PI regulators, their design methods and their effects on the regulated processes.

In the article [1] is presented an application of control in power electronics, for single-inductor MIMO DC-DC converter in electric vehicles based on multi-sampling advanced PWM technique. In the article [2] is presented an application of PID++ control for voltage control at a day/night power generator station, for Lunar and Martian space exploration. In the article [3], the use of a PID algorithm ensures precise grip force control, thereby protecting the brown mushrooms from damage in an electric gripper for picking brown mushrooms with flexible force and *in situ* measurement. The book [4] focuses on the theory and application of PID control in industrial processes, presenting recent developments with PID control technology in industrial practice from simple to optimal solutions for controller tuning, open-loop plant tests and closed-loop experiments. The book [5] proposes some advances studies in PID controller designs for numerous modern technology applications, as tuning controllers for integrating processes with both inverse response and dead time, design and development of PID controller for active suspension system for automobiles, stabilizing PID controllers for time delay systems and other. The chapter [6] presents a study on saturated PID controllers for industrial robots, with a case study on robot arm. Techniques of artificial intelligence are used to enhance the performances of control systems, as neural networks, fuzzy logic and genetic algorithms. As a development of the possibilities of PID regulators, fuzzy logic is used in the structure of these regulators [7, 8], to increase the efficiency of control systems. The chapter [9] presents a study on genetic PID controllers for enhancing control systems response. The chapter [10] presents a conceptual model development for a knowledge base of PID controllers tuning in open loop.

2. Methods

The block diagram of a digital control system is presented in **Figure 1a**.

The meanings of the blocks are: C-digital controller from digital equipment, CAN—analog-digital converter, CAN—digital-analog convertor, EE—actuator, P—process, EM—element of measure. The converters perform the sampling and hold operations, with the sampling period h. The shapes of the sampling and hold signals are shown in **Figure 2**.

Where t_{k-1}, t_k and t_{k+1} are the sampling moments.

The process can be discretized exactly according to the scheme in **Figure 3**, using relations (1) for a state-space model and relations (2) for a transfer function.

$$\begin{aligned}
&\dot{x}\,(t) = Ax(t) + Bu(t) && \\
&y(t) = Cx(t) && A_d(h) = e^{Ah} = e^{At}\big|_{t=h} = L^{-1}\left\{(sI - A)^{-1}\right\}\Big|_{t=h} \\
&x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \dots \\ x_n(t) \end{bmatrix} && B_d(h) = \int_0^h e^{A\tau} d\tau B \\
& && C_d = C \\
&A_{nxn}, B_{nxm},\ C_{rxn} &&
\end{aligned} \tag{1}$$

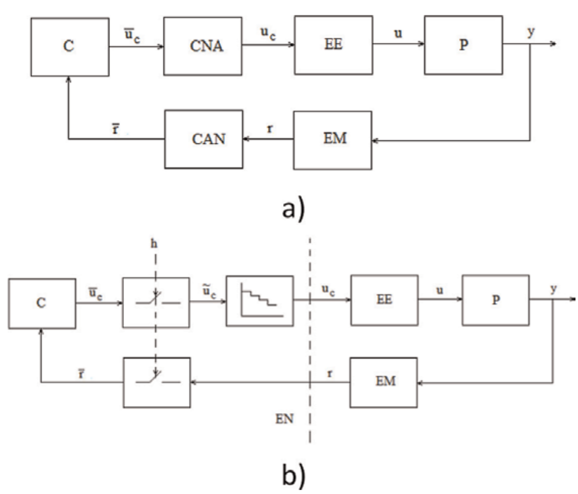

Figure 1.
(a) Block diagram of a numerical control system, (b) digital processing.

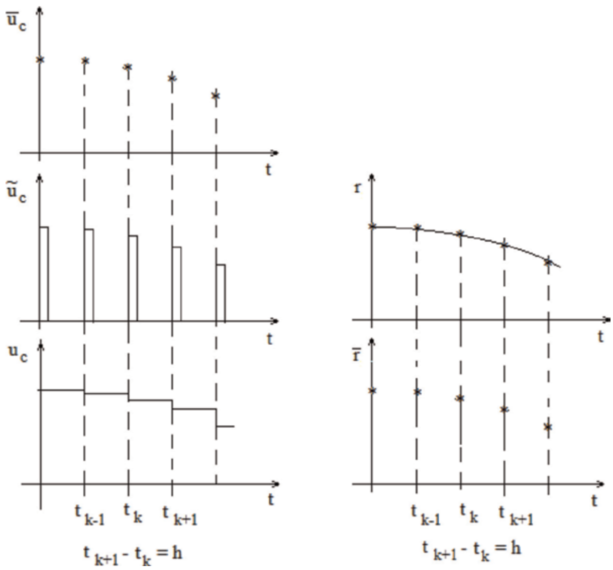

Figure 2.
Signals of sampling operations.

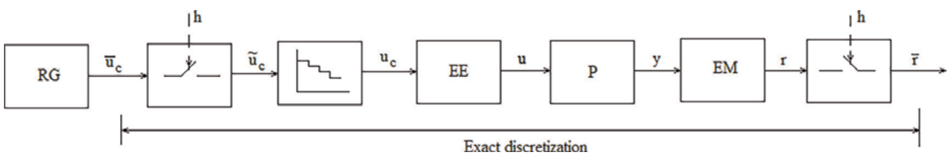

Figure 3.
The block diagram of exact discretization.

$$H(s) = \frac{y(s)}{u(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0}$$

$$H(z) = \frac{y(z)}{u(z)} = \frac{z-1}{z} Z\left\{ L^{-1}\left\{ H(s)\frac{1}{s} \right\}\Big|_{t=kh} \right\} \quad (2)$$

The operations that take place during a discretization period are presented in **Figure 4**.

In **Figure 5** shows the block diagram of the regulation system with transfer functions.

The blocks of the process are embedded in a link block L, as in **Figure 6**.

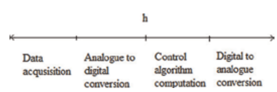

Figure 4.
Internal operations in the equipment during a discretization period.

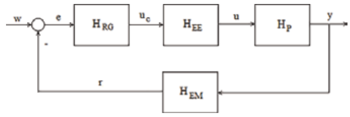

Figure 5.
The block diagram of the regulation system with transfer functions.

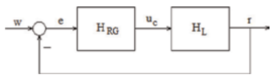

Figure 6.
The simplified block diagram.

The discrete transfer function $H_{\mathrm{RG}}(z)$ of the regulator is obtained from the transfer function of the regulator $H_{\mathrm{RG}}(s)$ by approximate discretization, using the approximation (3)

$$H_{RG}(z) = H_{RG}(s)\big|_{s=\frac{2}{h}\cdot\frac{1-z^{-1}}{1+z^{-1}}} \tag{3}$$

with

$$s = \frac{1}{h}\ln z \approx \frac{2}{h}\cdot\frac{z-1}{z+1} = \frac{2}{h}\cdot\frac{1-z^{-1}}{1+z^{-1}} \tag{4}$$

This being Tustin transformation, the Padé approximation of the exponential function.

3. Example

3.1 Controller design

A common example in practice is that of a second-order process. Controlled with a PI controller designed with Kessler's module criterion. The process, seen as a link block L like in **Figure 4**, has the transfer function:

$$H_L(s) = \frac{K_L}{(T_1 s+1)(T_2 s+1)},\quad T_1 > T_2 \tag{5}$$

The PI controller has the transfer function:

$$H_{RG}(s) = K_R\left(1+\frac{1}{sT_R}\right) \tag{6}$$

According to Kessler's method, the design equations are:

$$T_R = T_1, K_R = \frac{T_1}{2K_L T_2} \tag{7}$$

The values of the efficiency of the regulation indicators of the control system to be obtained are:

$$\sigma_1\% = 4,3\%, t_r = 8,4T_2 \tag{8}$$

The digital controller may be obtained by discretization by approximation:

$$\begin{aligned} H_{RG}(z) &= H_{RG}(s)\big|_{s=\frac{2}{h}\frac{1-z^{-1}}{1+z^{-1}}} = \\ &= K_R\left(1 + \frac{h}{2T_R}\cdot\frac{1+z^{-1}}{1-z^{-1}}\right) = \frac{K_R}{2T_R}\cdot\frac{2T_R + h - (2T_R - h)z^{-1}}{1-z^{-1}} \\ u_c(z) &= H_{RG}(z).e(z) \\ (1-z^{-1})u_c(z) &= \frac{K_R}{2T_R}\left[(2T_R + h)e(z) - (2T_R - h)z^{-1}e(z)\right] \end{aligned} \tag{9}$$

The digital control algorithm has the algorithm:

$$\begin{aligned} u_c(k) &= u_c(k-1) + a_1K_R(2T_R + h).e(k) + a_2K_R(2T_R - h).e(k-1) \\ a_1 &= K_R\left(1 + \frac{h}{2T_R}\right), a_2 = -K_R\left(1 - \frac{h}{2T_R}\right) \end{aligned} \tag{10}$$

And it may be implemented in a digital equipment, exactly in this form. The implementation only requires memorization, additions and multiplications.

The minimum number of samples may be computed with the formula:

$$N_r = \frac{T_r}{h} > 4\ldots10 \tag{11}$$

where T_r is the rise time.

3.2 Case studies

3.2.1 The linear control system

In **Figure 7** shows the block diagram of the simulation of the linear regulation system with a regulator sized as above.

The parameters have the following values: KL = 1, T1 = 0,1 s, T2 = 0,01 s, TR = 0,1 s, KR = 5.

The response to the step signal applied to the prescription input is shown in **Figure 8**.

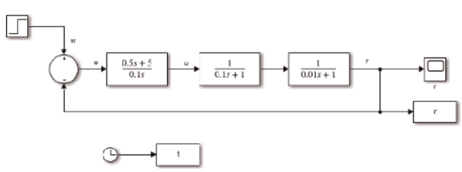

Figure 7.
The block diagram of the linear control system.

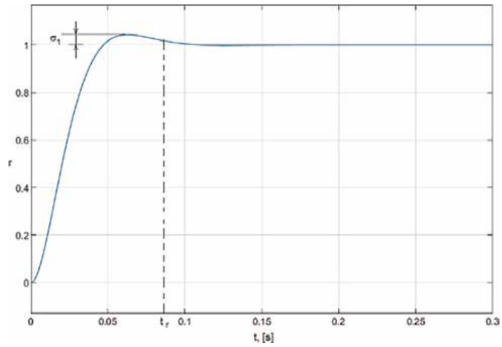

Figure 8.
The response to the step signal.

3.2.2 Digital control system

In **Figure 9** shows the block diagram of the digital control system with the digital controller designed for h = 0,001 s.

The responses to the step signal applied to the prescription input, for different values of the sampling periods, are shown in **Figure 10**. For larger values of the sampling period, the quality of the control decreases, causing errors at the output.

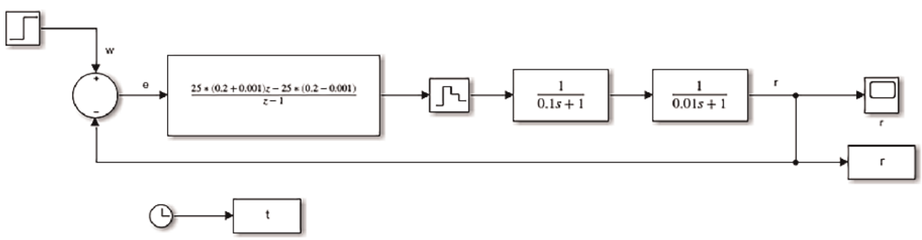

Figure 9.
The block diagram with digital controller for h = *0,001 s.*

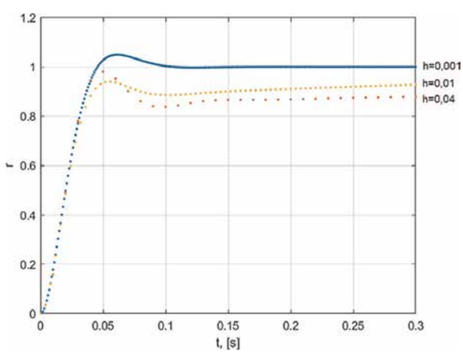

Figure 10.
The responses to the step signal.

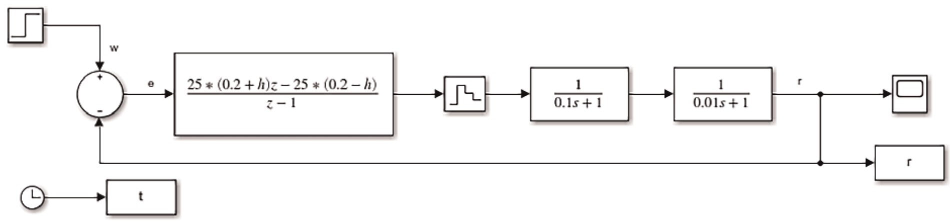

Figure 11.
The block diagram of the digital control system at different h.

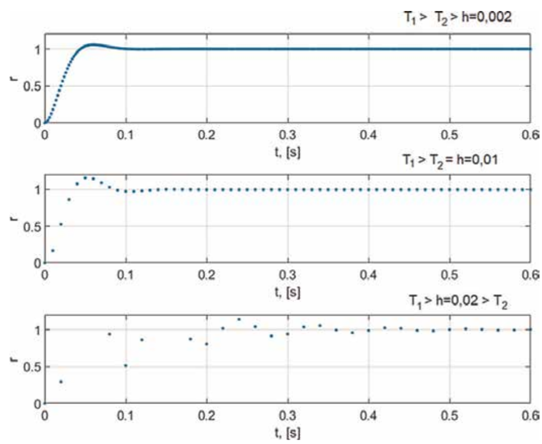

Figure 12.
The responses to the step signal.

3.2.3 Digital system at different sampling periods

In **Figure 11** shows the block diagram used for simulations with several values of the sampling period.

The responses to the step signal applied to the prescription input, for different values of sampling period, are shown in **Figure 12**.

For larger values of the sampling period, the quality of the control decreases, not fulfilling the condition according to the relationship (11).

4. Discussion

In theory, the linear regulation system with an appropriately sized regulator ensures the required values of the regulation efficiency indicators. Also, the regulation system with a regulator obtained by discretization with an appropriate sampling step faithfully follows the regulation characteristics of the system in continuous time. If the sampling period is shorter, adjustment errors appear and the adjustment efficiency indicators decrease in quality. For higher values of the sampling period compared to the time constants of the process, the quality of the regulation decreases, especially in the transient regime.

The proper choice of the sampling interval is very important. Too low sampling frequency may lose so much information that the control performance deteriorates and the system dynamics is lost. Too high sampling rate increases the burden of the processor; also, it may lead to discrete representation with bad numerical properties. For oscillating systems, the sampling interval is often tied to the frequency of the dominating oscillation. For damped systems, the sampling interval is usually chosen to be in relation to the time constant.

5. Conclusion

Digital control systems have many practical applications. They can be analyzed using mathematical models in discrete time. Mathematical models of processes in discrete time can be obtained from mathematical models in continuous time by exact discretization. Discrete time controllers can be designed using continuous time criteria and then discretized by approximation. The implementation of regulators can be done on numerical equipment using numerical regulation algorithms. Proper choice of the sampling interval is very important. Increasing the discretization period leads to instability and also to control errors, altering the performance indicators of the control systems. The digital control algorithm may be implemented in microprocessors using a developing software. The controller coefficients may be introduced in the digital equipment using a user's interface. The hidden technology of control systems consists in fact that the electric and electronic equipment could be developed using the analysis and synthesis methods presented.

Author details

Constantin Voloșencu
"Politehnica" University, Timisoara, Romania

*Address all correspondence to: constantin.volosencu@aut.upt.ro

References

[1] Solangi H, Hafeez K, Mekhilef S, Seyedmahmoudian M, Stojcevski A, Khan L. Advanced multi-sampling PWM technique for single-inductor MIMO DC-DC converter in electric vehicles. Energies. 2024;**17**:3633. DOI: 10.3390/en17153633

[2] Arciuolo TF, Faezipour M, Xiong X. Day/night power generator station: A new power generation approach for lunar and Martian space exploration. Electronics. 2024;**13**:2859. DOI: 10.3390/electronics13142859

[3] Shi H, Xu G, Lu W, Ding Q, Chen X. An electric gripper for picking brown mushrooms with flexible force and *in situ* measurement. Agriculture. 2024;**14**:1181. DOI: 10.3390/agriculture14071181

[4] Shamsuzzoha M, editor. PID Control for Industrial Processes. London: InTech; 2018. 218 p. DOI: 10.5772/intechopen.69592

[5] Vagia M, editor. PID Controller Design Approaches - Theory, Tuning and Application to Frontier Areas. London: InTech; 2012. 300 p. DOI: 10.5772/2628

[6] Orrante-Sakanassi J, Santibanez V, Camp R. On saturated PID controllers for industrial robots: The PA10 robot arm as case of study. In: Ehsan S, editor. Advanced Strategies for Robot Manipulators. Rijeka: Sciyo; 2010. DOI: 10.5772/10196

[7] Volosencu C. Stabilization of fuzzy control systems. WSEAS Transactions on Systems and Control. 2008;**10**(3):879-896

[8] Volosencu C. Identification of distributed parameter systems, based on sensor networks and artificial intelligence. WSEAS Transactions on Systems. 2008;**6**(7):785-801

[9] Mahmood Al-Rawi OY. Enhancing control systems response using genetic PID controllers. In: Popa R, editor. Genetic Algorithms in Applications. London: InTech; 2012. DOI: 10.5772/2675

[10] Luis J, Ferreiro R, Couce A, Quintian-Pardo H, Alaiz-Moreto H. Conceptual model development for a knowledge base of PID controllers tuning in open loop. In: Vizureanu P, editor. Expert Systems for Human, Materials and Automation. London: Intech; 2011. DOI: 10.5772/16551

Chapter 2

Design and Development of Servo Drive Control System Based on DSP

Ming-Yen Wei

Abstract

This chapter proposes the development of a motion control system combining a three-degree-of-freedom platform with immersive full-view 3D stereoscopic images. Initially, the visual effects computer and the three-degree-of-freedom rotating turret define network packet messages based on different scenes. When the visual effects computer receives joystick commands, it transmits control parameters to the image generator as control data. The turret servos and the visual effects computer then respectively report actual positions and joystick commands, exchanging relevant information via Ethernet. Subsequently, a three-axis control architecture is proposed, with a digital signal processor responsible for outputting pulse commands to the turret servos. This system integrates a controller to calculate pulse quantities and frequencies internally, ensuring good dynamic response, loading capacity, and tracking ability during manned operation and manipulation. The drive system utilizes the Texas Instruments TMS320F28377 digital signal processor as its core, performing logic operations, controller functions, and digital communication transmission. Hence, the hardware circuitry is relatively simple. Finally, experimental results demonstrate the effective application of the proposed system, confirming its feasibility and correctness in practical applications.

Keywords: digital signal processor (DSP), servo motor, motion control system, mechatronics, interactivity

1. Introduction

Simulation training, by simulating the sensory stimuli that humans can experience in real-world environments through human-machine interfaces, possesses three main characteristics: immersion, interactivity, and imagination [1]. Immersion makes individuals feel as if they are in a real-life scenario, interactivity creates continuous interactive states through pre-designed actions, and imagination stimulates human creativity through graphic design, motion decomposition, scene planning, light distribution, shadow configuration, and 3D imagery. Consequently, virtual reality (VR) technology is generally divided into the establishment of three-dimensional models, visual input devices, auditory input devices, tactile input devices, and mind input devices [2]. In modern times, VR technology is widely utilized, with associated peripheral hardware devices being well-developed, such as Google's Google Cardboard [3], Meta's Oculus [4], HTC's VIVE series [5], Samsung's Gear VR [6], and Sony's PlayStation VR [7].

VR applications are increasingly widespread for creating virtual prototypes, 3D models, and visualizations. For example, literature [8] proposed a comprehensive framework to achieve immersive services in augmented remote operation of unmanned aerial vehicles (UAVs). The work deeply analyzes various key performance indicators describing system performance and addresses VR-based remote and immersive control issues. Literature [9] proposed vehicle-type virtual reality (Vehicle VR), requiring the system to accurately simulate real environments and specific operating interfaces and equipment. In terms of driving simulation, the instruments operated are the same as those in actual flight cockpits. When users operate them, the VR system simulates the real-time responses of various instruments to users. Such systems are mainly designed to reduce training costs and are mostly applied in training scenarios requiring high fidelity. Literature [10] discussed the application of virtual reality (VR) in pistol shooting training in the Brazilian Army, focusing on the TAT VR simulator. It overcomes the limitations of traditional methods and compares the results with reference values from related work. Notable features include emphasis on convenience, accessibility, and flexibility, as well as samples from experienced army and navy officers.

Considering the widespread use of motion platforms and their significant applications in the transportation industry, training personnel can experience operations more realistically through virtual environments [11]. Literature [12] studied a four-degree-of-freedom motion chair to enhance flight simulation training for an immersive experience. The chair adopts a modular design and uses kinematics for control, achieving closed-loop control. Generally, for multi-degree-of-freedom control systems, electromechanical design is required to achieve trajectory tracking [13], with servo motors paired with reducers to meet the load capacity of the platform. It is evident that the mechanical structure of multi-axis motion platforms is highly complex, with uncertainties in control including inertia, efficiency, and friction coefficients within reducers, as well as the nonlinear characteristics caused by gear backlash, all while considering the weight of the mechanical structure and the combined weight of the experienced personnel. Therefore, system model establishment becomes quite challenging. On the other hand, motor control modes need to be switched according to different operating conditions; thus, the output of control commands will affect the precision of three-axis motion control systems. Literature [14–16] researched load estimation for unknown loads. For instance, the applied neural networks and adaptive theory to embedded permanent magnet motors, calculating controller weight values online to maintain precise speed control considering parameter variations and additional loads [14]. Wai and Lee designed intelligent optimal controllers, combining them with fuzzy neural networks for the control of permanent magnet motors [15]. Literature [16] applied fuzzy control to systems with uncertain inertia, ensuring that permanent magnet motors have better steady-state responses. Additionally, when the servo system performs reciprocating motions, nonlinear friction periodically appears. Therefore, this article adopts iterative learning control to adjust parameter uncertainties.

Generally, proportional-integral-derivative (PID) control is often used to improve the transient performance of systems and eliminate periodic nonlinear disturbances. The compensation amount adjusts using the system's error amount without needing to know the various parameters of complex systems beforehand, making it suitable for nonlinear, time-varying, and complexly modeled systems [17]. Given this, the highlight of the three-degree-of-freedom platform system proposed in this article is the first design of an interactive shooting simulator, combined with iterative learning theory, to improve system operational performance. It dynamically adjusts based on the system's error amount, capable of eliminating periodic nonlinear disturbances without the need to measure the parameters of complex systems. This is suitable for nonlinear, time-varying, and complexly modeled systems. The DSP clock used in the core control is 200 MHz, allowing participants to enjoy the thrill of controlling gun turrets and shooting freely while their feet hang outside the turret, experiencing a tense thrill as if on the edge of an abyss. Additionally, this article utilizes the Oculus Quest as the VR gaming device, running VR shooting games created by this study on the Unity system. Through joystick manipulation, participants experience and operate rotating gun turrets and stereophonic sound while shooting at continuously generated intelligent targets.

2. Control system

The internal controller of the servo drive adopts a proportional-integral-derivative (PID) controller, which features simple structure, convenient design, and easy adjustment advantages and is widely used in control loop design. It calculates straightforward mathematical equations. Servo motors often require the use of gearboxes, racks, or lead screws to generate actual mechanical motion.

In this paper, the implementation of the three-axis motion system was based on the PID controller, as shown in **Figure 1**, because of its simplicity and because it is widely used in industrial applications. In addition, by suitably adjusting the gains: K_p, K_I and K_D, the controller can achieve the required performance of a closed-loop

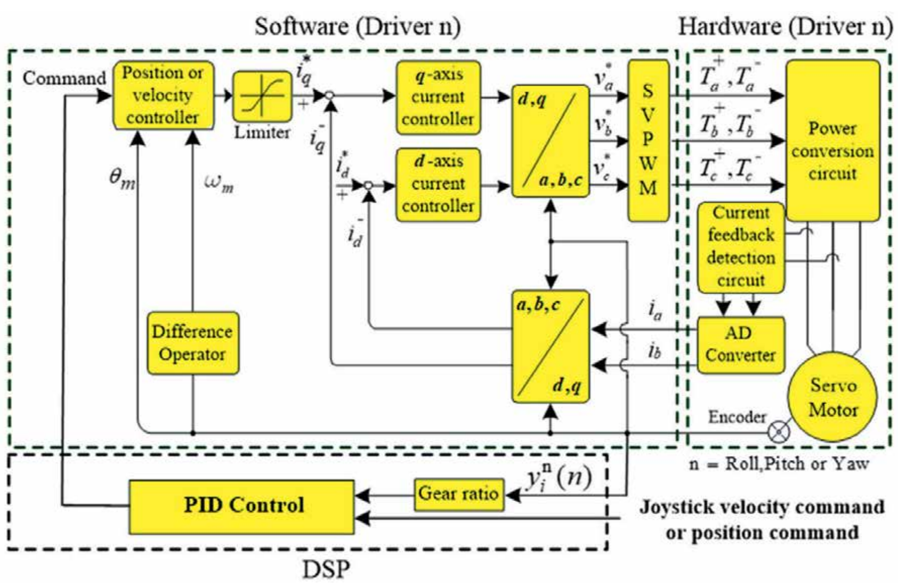

Figure 1.
The closed-loop control system.

control system, as shown in **Figure 1**. The internal speed or position bandwidth of the driver is set to 1 kHz, and the frequency of packet transmission from the visual effects computer to the digital signal processor network is set to 50 Hz.

Based on the error between y_d and y_i, the pulse command is determined to be either activated or deactivated. When the pulse command is activated, it indicates that the motor can be controlled to run. With a fixed frequency, the motor will maintain constant speed, while with varying frequency, the motor will accelerate or decelerate. When the pulse command is deactivated, the motor stops all motion, indicating that positioning has been completed. The PID control law is

$$u = K_P e(t) + K_I \int e(t)dt + K_D \frac{d}{dt} e(t) \tag{1}$$

where K_p, K_I, and K_D are the proportional coefficient, integral coefficient and differential coefficient, respectively. The closed-loop control system is designed and applied independently to each actuator in the three degrees of freedom. The command represents the reference command.

3. Design of artificial intelligence for enemy aircraft

Design of the Aircraft is divided into three types: target aircraft, companion aircraft, and enemy aircraft. The first type of target aircraft generates the required sensory feedback through cockpit personnel operating the joystick. The second type, companion aircraft, primarily accompanies the target aircraft without any specific actions. The third type, enemy aircraft, is designed using artificial intelligence to attack the target aircraft and search for targets. Unity game development visual effects software is utilized, employing the AI Planner package to enable autonomous decision-making for the enemy aircraft. Programmatic coding designs rules and logic, and decision trees are employed at each decision node to select the optimal branch node, as shown in **Figure 2**.

Additionally, finite state machines (FSM) are used to establish reactive AI to implement the behavior of enemy aircraft. The primary development language for Unity Engine is C#, and Microsoft's Visual Studio Code is utilized as the development environment. FSM design divides the behavior of enemy aircraft into four modes: patrol mode, attack mode, evasion mode, and idle mode, as illustrated in **Figure 3**. At the initial stage, the AI system defaults to the patrol mode to search

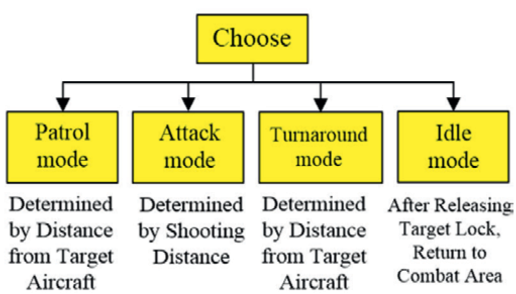

Figure 2.
Reactive AI decision nodes.

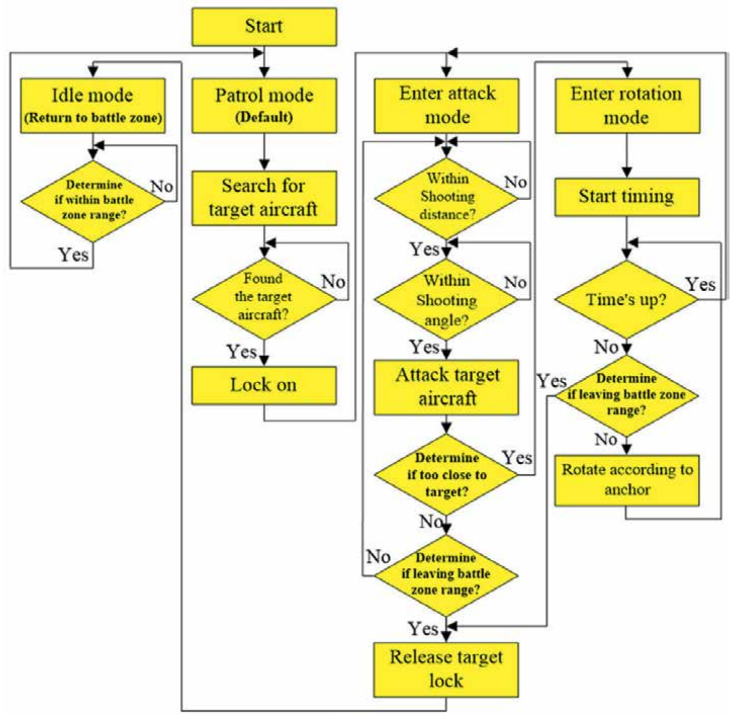

Figure 3.
FSM for reactive AI.

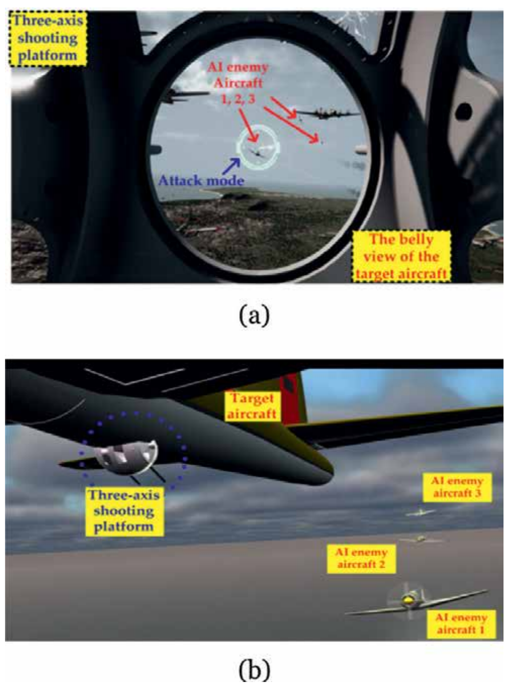

Figure 4.
AI interface design: (a) cockpit view; (b) relationship between target aircraft and AI enemy aircraft.

for targets and locks onto the nearest target unit. Upon locking onto the target, the AI system begins automatic navigation to pursue the target, transitioning to attack mode. Depending on the distance and angle to the target, the enemy aircraft adjusts its speed and determines the optimal time to shoot. If the distance between the enemy and the target aircraft becomes too close, the evasion mode is executed to assess whether to leave the combat zone. If not, it continues pursuing the target based on anchor points and re-enters attack mode after 7 seconds to continue chasing the target. Otherwise, it enters idle mode to return to the combat zone and then switches back to patrol mode. **Figure 4(a)** illustrates the belly view of the target aircraft, clearly showing the screen after entering attack mode. **Figure 4(b)** depicts the relationship between the position of the three-axis shooting platform and the target aircraft and AI enemy aircraft.

4. System overview

This section provides an overview of the three-axis shooting platform drive and control system. Using a digital signal processor as the control core, combined with relevant hardware circuits, it achieves speed and position control. **Figure 5** illustrates the block diagram of the control system, consisting of three main parts. The first part employs a digital signal processor as the control computation core. The second part

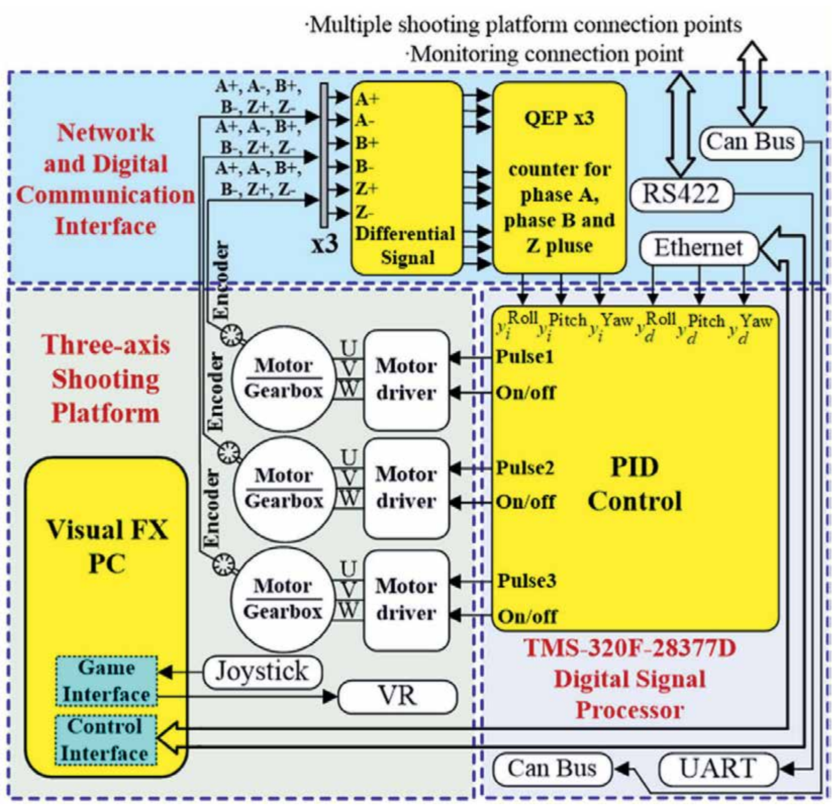

Figure 5.
Three-axis motion platform control system block diagram.

comprises the hardware of the three-axis shooting platform, including a set of visual effect computers, three sets of servos, joysticks, and VR headsets. The third part primarily serves as the interface for receiving digital and network signals, including three sets of encoder signals: Can Bus, RS422, and network communication. The TMS-320F-28377D digital signal processor is a 32-bit floating-point module executing the required instructions for computation. The software program includes position and velocity calculations, iterative learning algorithms, visual effect computer network interfaces, monitoring computers, and multi-shooting platform interface programs. The visual effect computer selected is a fanless computer produced by Advantech paired with an RTX2080 Ti graphics card. The VR headset chosen is the Oculus model from Meta's Quest series. **Figure 6** depicts the internal perspective, where personnel manipulate a three-axis joystick to track AI enemy aircraft. Upon targeting, shooting is initiated by pressing the trigger button. The gearbox is manufactured by SPT with a gear ratio of 240. The motors selected are three TECO PBC08ABKB servo motors, each rated at 750 watts, with parameters detailed in **Table 1**.

To validate the proposed shaft angle estimation method and related control strategies, this study employs the digital signal processor developed by Texas Instruments as the development platform. The control board uses the TMS-320F-2877D digital signal processor chip produced by Texas Instruments as the control core, featuring the advantages of fast computation by the microprocessor and parallel processing capabilities, utilizing floating-point arithmetic. The physical architecture is shown in **Figure 7(a)** and **(b)**. In **Figure 7(a)**, item A is the chip socket, item B is SCIB and SCIC, item C is CANA and CANB, item D is the analog and digital interface, item E is the digital interface, and item F is the power input. **Figure 7(b)** shows the TMS-320F-2877D chip inside the chip socket.

Figure 8 illustrates the operational flowchart. Upon pressing the power button on the control box, power is supplied to the cockpit. Subsequently, the computer initiates and enters the human-machine interface. The operator executes the zeroing button, as depicted in the diagram. Once the personnel confirm readiness by being seated, the preparation button is activated. Then, through communication between the servo and the visual computer, pressing the start button enables control of the platform

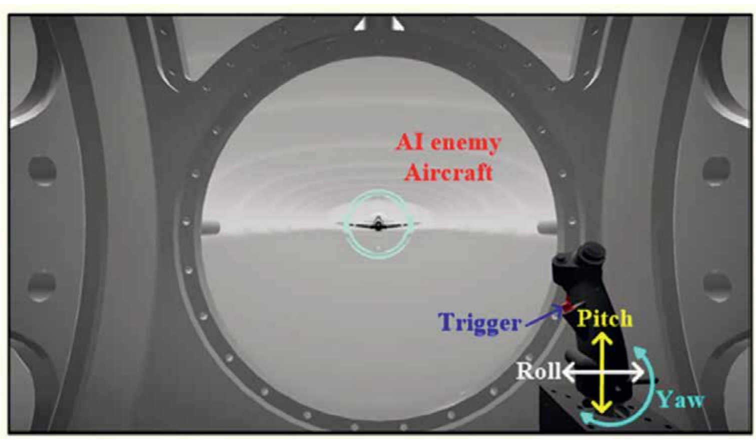

Figure 6.
Aiming at AI enemy aircraft with joystick.

Parameter	Units	Values
Pole number	poles	8
Power consumption	kW	0.75
Rated voltage	Volt	220
Rated current	Arms	4.3
Rotor inertia	kg m^2	0.000164
Rated speed	rpm	3000
Maximum speed	rpm	5000
Rated torque	N m	2.39
Maximum torque	N m	7.16

Table 1.
Parameters of the motor.

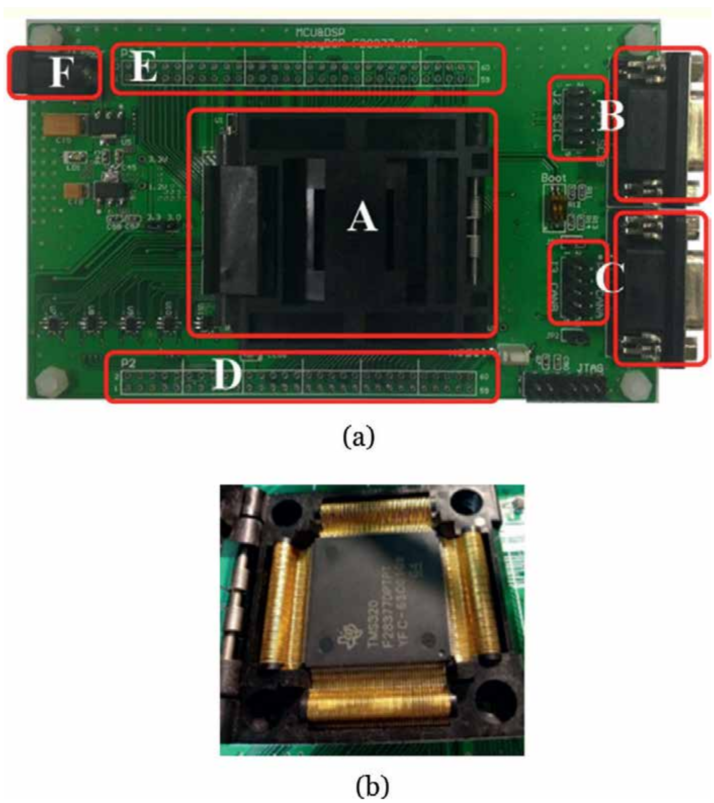

Figure 7.
Physical diagram of TMS-320F-28377D digital signal processor control board: (a) control board; (b) chip.

via the joystick. Upon pressing the end button to conclude the game, the platform automatically returns to its original position. Pressing the reset button during startup or operation retains only the zero state, while all other states are cleared, awaiting new instructions.

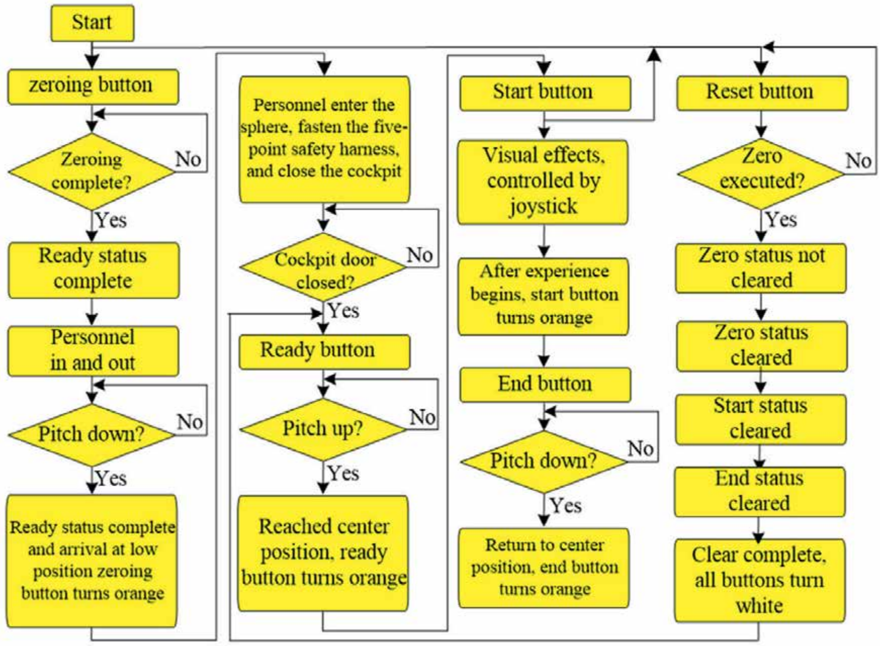

Figure 8.
Operation process diagram.

Figure 9(a–c) depicts the wiring of three rotating axes. In **Figure 9(a)**, the wire enters the yaw axis bracket hole and passes through to the top exit hole, leaving a length of 500 mm from the bracket to the rotating body. Then, it follows the servo wiring along the roll axis bracket (**Figure 9(b)**) and the pitch axis bracket (**Figure 9(c)**), while the power and signal for in-cabin use will enter from the middle of the bearing on the other side of the pitch axis for internal use. **Figure 10(a–c)** illustrate the internal configuration of the control box, divided into three sections.

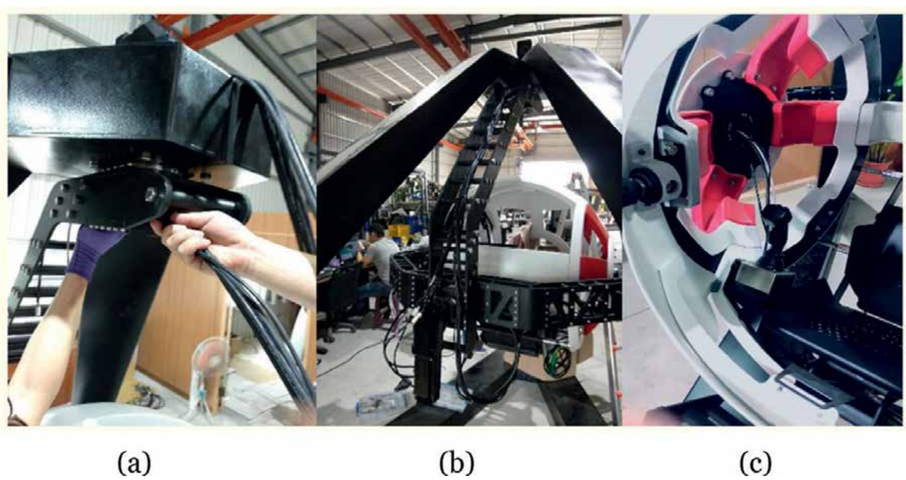

Figure 9.
Illustrates the wiring of the three rotating axes of the platform: (a) yaw axis; (b) roll axis; (c) pitch axis.

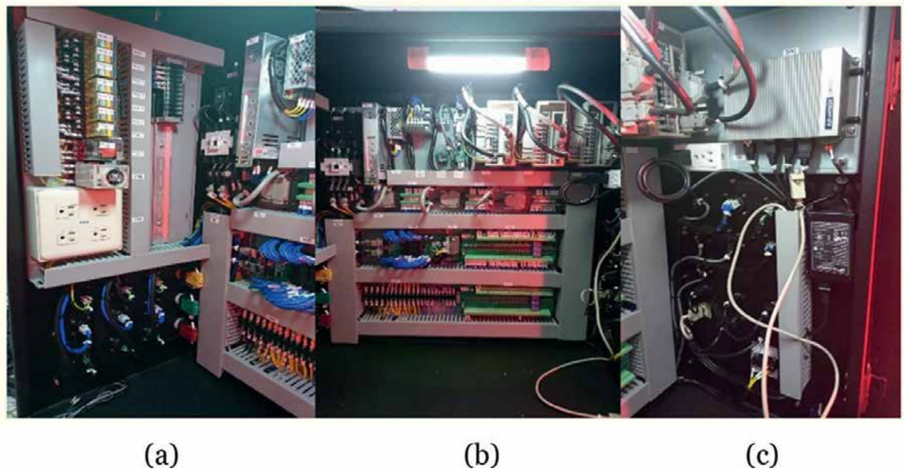

Figure 10.
Depicts the internal configuration of the electrical control box: (a) left distribution panel; (b) main distribution panel; (c) right distribution panel.

The left section in **Figure 10(a)** consists of toggle switches and relays. As shown in **Figure 10(b)**, it comprises circuit breakers, digital signal processors, interface boards, and drivers. **Figure 10(c)** serves primarily as the power source for in-cabin use and for network communication transmission and reception.

5. Experimental results

The servo software is written and debugged using the CCS8.2 software and then executed. The visual effects computer is connected to update, compute, and actuate the servos via the network interface. It also constantly receives instructions from the operating computer and transmits internal status to the servos. **Figure 11** depicts the visual

Figure 11.
Visual effects scene flight path.

scene, set in an island location, to make the flight path more challenging. As the flight attitude varies with different flight path planning, slight rolling effects are observed, as indicated by the fluctuation in height between different green dots. **Figure 12(a)** shows the response of three servo systems when controlling the three-axis platform using a joystick. **Figure 12(b)** present the experimental results of the proposed method. A joystick

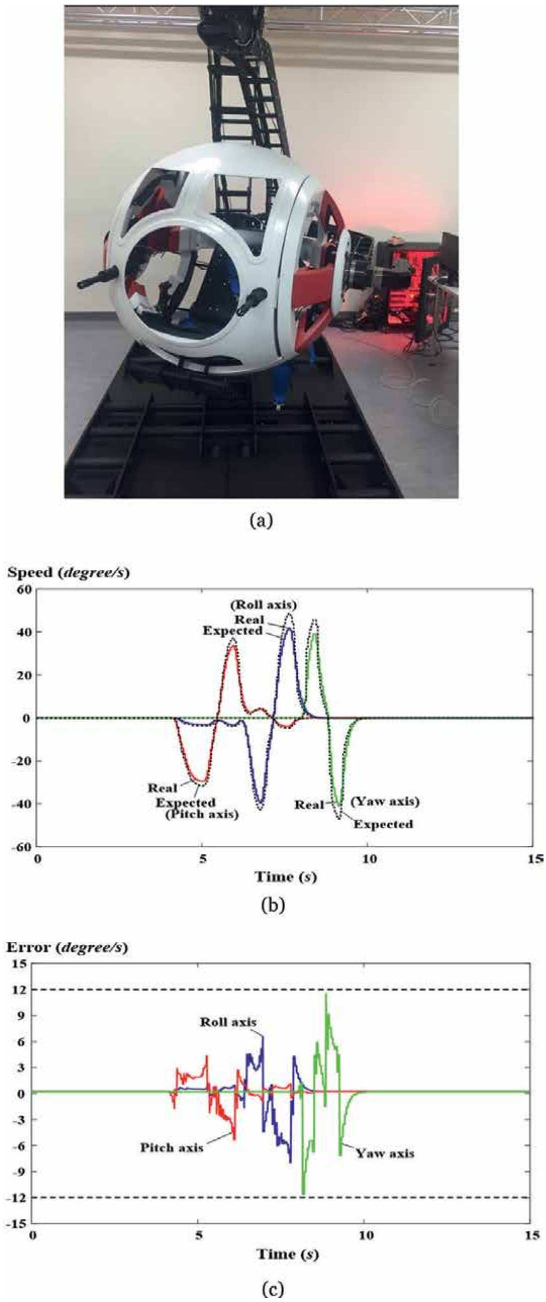

Figure 12.
Measured response of angular velocity response for the three-axis platform controlled by joystick: (a) test photo image; (b) PID; (c) error.

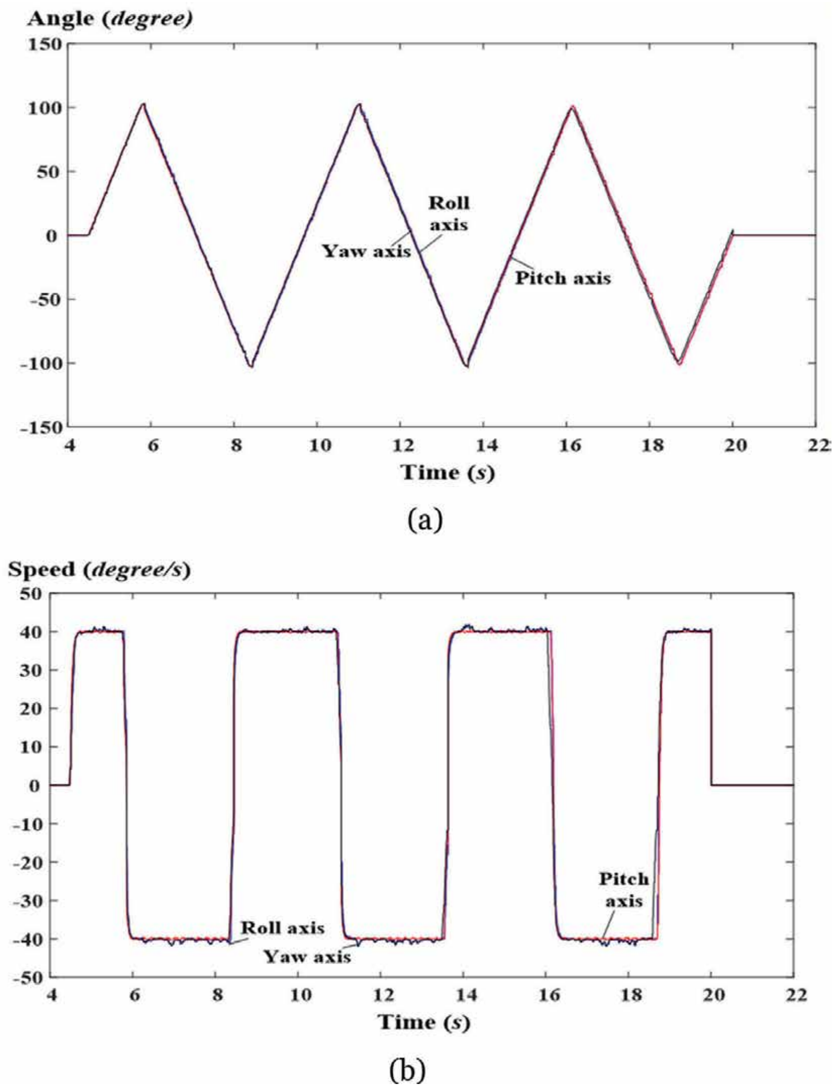

Figure 13.
Measured response of position-controlled three-axis platform: (a) angle; (b) angular velocity.

is utilized to command the velocity, and the actual speeds of the three servos are measured. **Figure 12(c)** depict the errors corresponding to **Figure 12(b)**. It can be noted that the proposed method yields errors of approximately ±12 degrees per second. **Figure 13(a** and **b)** illustrates the control test of the position loop. When a ±100-degree forward and reverse motion is provided, the response diagram of the three-axis angle (**Figure 13(a)**) and speed (**Figure 13(b)**) is obtained.

6. Conclusions

This study developed a three-degree-of-freedom shooting platform that combines interactivity with virtual reality to achieve high-fidelity effects. Participants can control pitch and yaw axes using a joystick, while the roll axis motion is determined by a visual effects computer based on the target aircraft's attitude. When encountering attacks from AI opponents, the platform generates a sense of shaking on the corresponding axis, enhancing the participant's immersion in the virtual reality environment. Control of the three-degree-of-freedom servos employs PID control to mitigate the effects of mechanical backlash, coupled with network connectivity and integration with visual effects computer scenes to create fully immersive 3D stereoscopic images, making participants feel as if they are in the real world. The experimental results presented in this paper demonstrate the simulator's capabilities in servo and visual effects interaction, three-degree-of-freedom control, and hardware development. In the realm of entertainment, this technology can be integrated into theme park attractions or home gaming systems, offering users a more immersive and interactive experience.

Author details

Ming-Yen Wei
Department of Electrical Engineering, National Formosa University, Huwei, Yunlin, Taiwan

*Address all correspondence to: wei@nfu.edu.tw

References

[1] Burdea GC, Coiffet P. Virtual Reality Technology. 2nd ed. New York: John Wiley and Sons; 2003

[2] Mandal S. Brief introduction of virtual reality and its challenges. International Journal of Scientific and Engineering Research. 2013;**4**:304-309

[3] Google. Google Cardboard. Available from: https://arvr.google.com/intl/en_us/cardboard/ [Accessed: August 4, 2024]

[4] Meta. Meta Quest. Available from: https://www.meta.com/tw/quest/ [Accessed: August 4, 2024]

[5] HTC. VIVE. Available from: https://www.vive.com/us/ [Accessed: August 4, 2024]

[6] Samsung. Gear VR. Available from: https://www.samsung.com/tw/support/model/SM-R322NZWABRI/ [Accessed: August 4, 2024]

[7] Sony. PlayStation VR2. Available from: https://www.playstation.com/en-tw/ps-vr2/?emcid=pa-co-506610&gad_source=1&gclid=CjwKCAjwouexBhAuEiwAtW_Zx51Dr_nmVBTFzMpxJf2l1qGqtuP1ibtkZ8NK0mB7DdAATPP8fsUBmBoC9G0QAvD_BwE&gclsrc=aw.ds [Accessed: August 4, 2024]

[8] Taleb T, Sehad N, Nadir Z, Song J. VR-based immersive service management in B5G mobile systems: A UAV command and control use case. IEEE Internet of Things Journal. 2022;**10**:5349-5363

[9] Rhiu I, Kim YM, Kim W, Yun MH. The evaluation of user experience of a human walking and a driving simulation in the virtual reality. International Journal of Industrial Ergonomics. 2020;**79**:103002

[10] Girardi R, Filho AAP, Teodoro GP, Oliveira JCD. TAT VR: A virtual reality simulator for military shooting training. In: Proceedings of the 24th Symposium on Virtual and Augmented Reality; 24-27 October 2022; Natal, RN Brazil

[11] Casas-Yrurzum S, Portalés-Ricart C, Morillo-Tena P, Cruz-Neira C. On the objective evaluation of motion cueing in vehicle simulations. IEEE Transactions on Intelligent Transportation Systems. 2021;**22**:3001-3013

[12] Wei MY, Yeh YL, Chen SW, Wu HM, Liu JW. Design, analysis, and implementation of a four-DoF chair motion mechanism. IEEE Access. 2021;**9**:124986-124999

[13] Yue X, Vilathgamuwa M, Tseng KJ. Robust adaptive control of a three-axis motion simulator with state observers. IEEE/ASME Transactions on Mechatronics. 2005;**10**:437-448

[14] Yi Y, Vilathgamuwa DM, Rahman MA. Implementation of an artificial neural network based real time adaptive controller for an interior permanent magnet motor drive. IEEE Transactions on Industry Applications. 2003;**39**:96-103

[15] Wai RJ, Lee MC. Intelligent optimal control of single link flexible robot arm. IEEE Transactions on Industrial Electronics. 2004;**51**:201-220

[16] Li S, Liu Z. Adaptive speed control for permanent magnet synchronous

motor system with variations of load inertia. IEEE Transactions on Industrial Electronics. 2009;**56**:3050-3059

[17] Mohammed IK, Abdulla AI. Elevation, pitch and travel axis stabilization of 3DOF helicopter with hybrid control system by GA-LQR based PID controller. International Journal of Electrical and Computer Engineering. 2020;**10**:1868-1878

Chapter 3

A Novel Approach for the Identification and Control of Unstable Systems Using PITOPS: Simplifying Complex Processes beyond Traditional Methods

Davarapalli Kishore, Sabbarapu Ganesh and Vaska Lokesh

Abstract

Most of the unstable systems exist in the chemical process such as bio-reactor, distillation column, CSTR, fluidized bed reactor. The control of the unstable systems was very difficult due to their multiple steady states, complexity, and dynamic behavior. There are several methods available in the literature for identifying and controlling unstable systems. But all these methods require extensive mathematical complexity and in-depth knowledge of the processes. Moreover, there is much more complexity associated with identification and controller design for multiple-input and multiple-output (MIMO) systems using the conventional methods. In the present work, we are incorporating the simplest method for identification and control the unstable systems using the modified relay feedback approach along with one data-driven approach. Instead of using the MATLAB routine for the purpose of identification and controller design, we are trying to identify and control using the new software PITOPS. A data-driven approach has been introduced to overcome the difficulties with conventional methods with the help of PITOPS. A comparative study has been made among the methods proposed in this work.

Keywords: identification, relay feedback, unstable, MIMO, stable, system identification

1. Introduction

Identification and control play a very prominent role in process control applications. Most of the processes in chemical industries are complex, multi-input and multi-output (MIMO), and nonlinear and precise, mathematical modeling is also challenging due to the presence of strong interactions among the process variables. For any process to be controlled and optimized, it should be identified with prior knowledge. This identification has to be obtained either through mathematical modeling through the transfer function approach or else by using the standard identification methods in the literature such as open-loop identification methods and closed-loop identification methods. Among the two identification methods closed-loop identification methods are pioneered because of their accuracy and less noise. Even though there are various closed-loop methods available, the system identification and relay methods are more prominent among the methods available.

System identification is the process of building mathematical models of dynamic systems based on measured input-output data. The goal of system identification is to understand and describe the behavior of a system without necessarily understanding its underlying physical principles. System identification is used in various fields, including control engineering, signal processing, and machine learning. The need for identification is to model, control, predict, and optimize the performance of a system.

Among time domain, frequency domain and relay feedback identification, relay feedback [1] is widely used for its simplicity and accuracy. The drawback of this method is that it fails when the order of the systems increases and involves extensive simulations and mathematical modeling. The use of relay feedback is to induce a limit cycle in the process output. The limit cycle is an important behavior that is used to extract process information [2]. This method overcomes the difficulties associated with the earlier method but this approach is not applicable to the unstable systems. The identified parameters decide the good tuning of PID controllers. Closed-loop identification is considered in this work as it overcomes the drawbacks of open-loop identification, that is, sensitive to disturbance, more computational time, and sometimes process output deviates from the set point [3]. Astrom and Hagglund [1] and Luyben [4] pioneered the use of relay feedback combined with a describing function approximation as a simple means to determine the ultimate gain and ultimate frequency. An integrating system is a type of system in control theory where the output signal is the integral of the input signal concerning time. The main limitation of these methods is they are all applicable to the process which has well-defined transfer functions and lower order processes.

The effective tuning of Proportional Integral Derivative (PID) controllers in process industries heavily relies on accurately identifying process models. Traditionally, open-loop methods have been used, involving the introduction of controlled excitation to observe output responses. However, these methods suffer from sensitivity to disturbances, prolonged computational time, and deviation from set points. In contrast, closed-loop identification methods particularly relay feedback identification; have gained traction due to their ability to overcome these drawbacks. By utilizing a relay instead of a controller, sustained oscillations or limit cycles are generated, providing valuable insights into the process dynamics.

A system is said to be unstable if bounded input produces an unbounded (infinite) output. In other words, a system whose transfer function has at least one pole lying in the right half plane is known as an unstable system. Open-loop instability means the system will move away from the steady state even for a small perturbation of the system parameters or operating conditions. The poles of the transfer function and the eigenvalues of the system matrix are related. The eigenvalues of the system matrix A are the poles of the transfer function of the system in the state-space domain. Every pole of the transfer function is an eigenvalue of the system matrix A. However, not every eigenvalue of axis is a pole of the transfer function. There are some disadvantages in the unstable systems like the performance of the control systems is limited for unstable transfer function models. The overshoot and the settling time are larger for the unstable systems compared to the stable systems. Rao [5] has presented enormous research on unstable systems. Chidambaram and Vivek Sathe [6] have reported the various methods for identification and control of the systems using the relay feedback approach. Kishore et al. [7] have presented a novel method of identification, which is a combination of both subspace and relay feedback approaches. But all the above methods are not applicable when there is theoretical aspects are involved and complexity also increases when the order of the process increases.

Most of the methods mentioned in the above literature involved extensive mathematical calculations and needed prior information on the process, control, prediction, and optimization will be too complex. Moreover, if the order of the system increases one needs a thorough understanding of the dynamics of the system, even it becomes more complex if it applies to unstable systems. Since most of the systems in chemical process are unstable and nonlinear and multiple-input and multiple-output (MIMO). To address the above issues, an attempt has been made by introducing data-driven approach such as Process Identification and Controller Tuning Optimizer Simulator (PITOPS) and Advance Process Control (APC) methods. At first, the identification is carried out through the relay feedback approach for unstable system data which has been considered from the literature and compared with other data-driven approaches. The rest of the chapter includes the application of the relay feedback method and system identification methods using PITOPS and APC methods and comparative studies have been made among the above methods.

2. Relay feedback approach

We performed a series of advanced control operations to fine-tune a PID controller for a control system using relay feedback. The relay feedback was employed to induce oscillations, allowing us to determine the ultimate gain and period. Subsequently, we utilized the Ziegler-Nichols tuning rules to calculate the initial PID parameters. Initially, the fundamental steps in the relay feedback are illustrated with simple mathematics [8].

Relay feedback control:

$$\mathrm{u(t)} = \begin{cases} relay_{bias} + relay_{amplitude}, \mathit{if}\ e(t) > relay_{hysteresis} \\ relay_{bias} - relay_{amplitude}, \mathit{if}\ e(t) < -relay_{hysteresis} \\ relay_{bias}, otherwise \end{cases}$$

where, e(t) = setpoint−measured value.

First-order plant response:

$$\mathrm{y(t)} = \left(1 - \frac{\Delta \mathrm{t}}{\tau}\right).\mathrm{y(t - \Delta t)} + \frac{K \cdot u(t) \cdot \Delta t}{\tau} \quad (1)$$

where:

- y(t) is the plant output,
- u(t) is the control signal,
- τ is the time constant,
- K is the plant gain.

Ziegler-Nichols tuning parameters:
Kp =0.6·Ku.
Ti =0.5·Pu.
Td = 0.125·Pu.
where:

- Kuis the ultimate gain,
- Pu is the ultimate period.

PID controller output:

$$\mathrm{u(t)} = \mathrm{Kp}\left[e(t) + \frac{1}{Ti}\int_0^t e(\tau)d\tau + T_d\frac{de(t)}{\mathrm{dt}}\right] \tag{2}$$

where:

- Kp is the proportional gain,
- Ti is the integral time,
- Td is the derivative time,
- e(t) is the error at time t.

Performance indices:

- Integral of absolute error (IAE):

$$\mathrm{IAE} = \int_0^{tf} |e(t)|dt$$

- Integral of squared error (ISE):

$$\mathrm{ISE} = \int_0^{tf} \mathrm{e(t)}^2 dt$$

- Integral of time-weighted absolute error (ITAE):

$$\mathrm{ITAE} = \int_0^{tf} t\cdot|e(t)|dt.$$

Process parameters (Theta and Tau):

- **Theta** (θ): Dead time
- **Tau** (τ): Time constant

These are typically calculated from the response to a step input:
τ = time constant

We defined an objective function incorporating integral absolute error (IAE), integral squared error (ISE), and integral time absolute error (ITAE) to measure control performance. This function was then minimized using the sequential least squares quadratic programming (SLSQP) optimization method with constraints ensuring that the performance indices remained within acceptable limits.

The optimization process yielded the following PID parameters: Kp = 1.2927, Ti = 1.9109, Td = 0.4777. The optimized controller's performance indices were IAE = 1.3865e+37, ISE = 1.1239e+74, ITAE = 1.0991e+39, and NRMSE = 5.0524e+35. Additionally, the system's time delay (theta) and time constant (tau) were determined to be 0.0 and 0.307, respectively.

These results, depicted in various plots, demonstrate the effectiveness of the optimization procedure in tuning the PID controller for improved performance, as evidenced by the calculated performance indices and the system's response.

The optimized PID parameters and performance indices were as follows:

- Ultimate Gain (Ku): 2.154431
- Ultimate Period (Pu): 3.8217
- Proportional Gain (Kp): 1.292658
- Integral Time (Ti): 1.91085
- Derivative Time (Td): 0.477713
- IAE: 1.386477e+37
- ISE: 1.123906e+74
- ITAE: 1.099141e+39
- NRMSE: 5.052398e+35

These results indicate that the PID controller was successfully tuned to achieve optimal performance, albeit the extraordinarily high values of IAE, ISE, ITAE, and NRMSE suggest potential issues with the system's data or initial conditions that might need further investigation.

The relay feedback approach, with both the relay amplitude and hysteresis set to 2, proved effective for inducing oscillations and determining the ultimate gain and period. The subsequent optimization using the Ziegler-Nichols tuning rules and the SciPy minimize function resulted in a set of PID parameters that significantly enhanced the control system's performance.

However, the high error indices suggest the need for further refinement, possibly involving a review of the initial conditions and the accuracy of the transfer function model. As experts in process control, we recommend iterative testing and validation to ensure the robustness and reliability of the optimized control system. The responses obtained through relay feedback are shown in **Figure 1**. The parameters calculated with this method are shown and controller settings are obtained and errors also calculated.

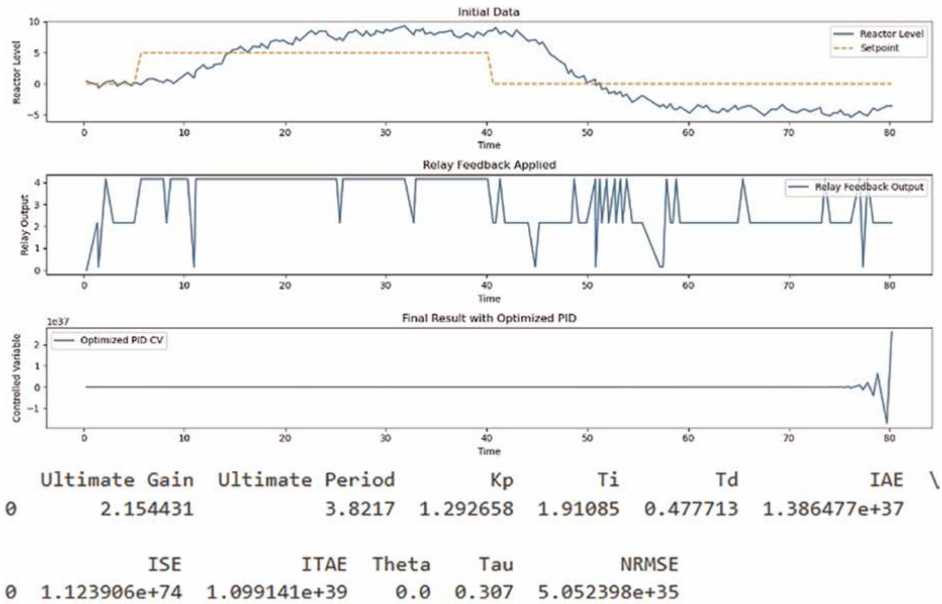

Figure 1.
Relay feedback response.

3. System identification using PITOPS

In this section, we present a systematic method for performing system identification in PITOPS, focusing on obtaining accurate first-order transfer function parameters. Our analysis was carried out in two distinct steps, utilizing specific time windows to refine our model.

Step 1: First data set analysis

For the initial analysis, we selected a time window from 0 to 2400 seconds with a sampling time of 30 seconds. The identified first-order transfer function parameters were given below. The response obtained through this method is shown in **Figure 2** and parameters obtained through this method are as follows:

- Delay: 219.18521 seconds
- Gain: 1.81674
- Time constant (Tau 1): 366.62186 seconds
- Time constant (Tau 2): 0.00 seconds

The performance metrics for this identification were:

- Fit (FIT): 95.77%
- Normalized root mean square error (NRMSE): 0.208
- Integral of absolute error (IAE): 0.500 EU

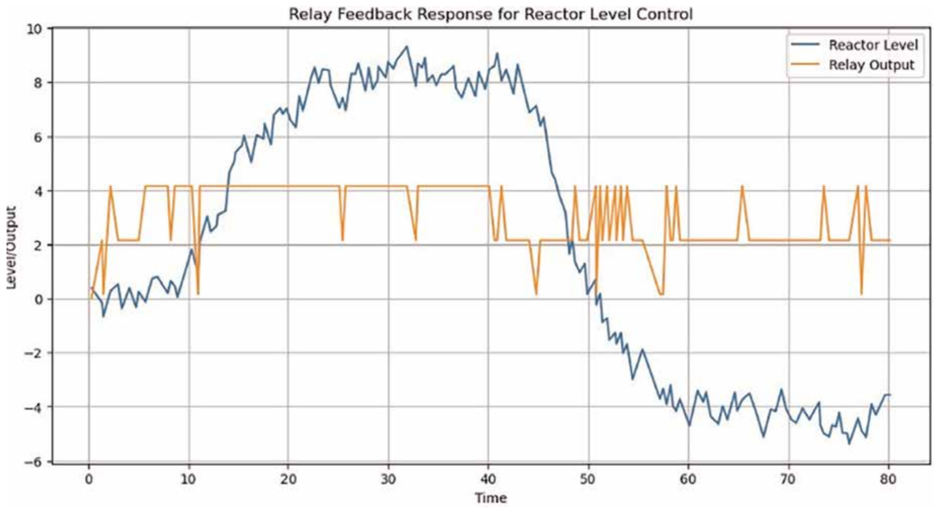

Figure 2.
Relay feedback response for reactor level control.

These results indicate a robust model, effectively capturing the dynamics of the process based on the selected data.

Step 2: Second data set analysis

In the second step, we focused on a time window from 2190 to 4770 seconds, maintaining the same sampling time of 30 seconds. The identified transfer function parameters for this analysis were:

- Delay: 229.04309 seconds
- Gain: 2.53672
- Time constant (Tau 1): 476.05698 seconds
- Time constant (Tau 2): 0.00 seconds

The results for this data set were even more favorable:

- Fit (FIT): 98.49%
- Normalized root mean square error (NRMSE): 0.124
- Integral of absolute error (IAE): 0.44 EU

This improvement in fit and reduction in error metrics confirm that the second data set provided a clearer representation of the system's dynamics. The dynamics in the responses are shown in **Figures 3** and **4** below.

Through a methodical approach using PITOPS, we successfully identified first-order transfer function parameters over two carefully selected time intervals. The results from both analyses demonstrate high accuracy and reliability, laying a strong

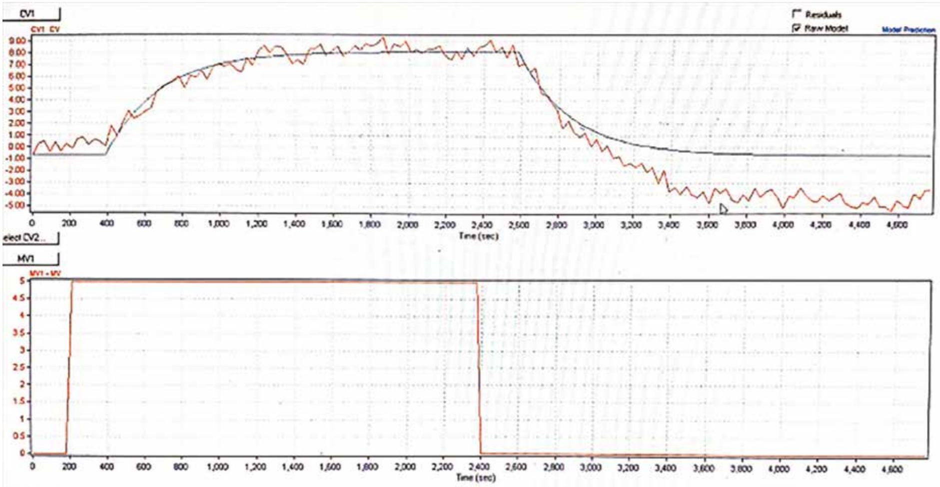

Figure 3.
Representation of dataset and its dynamics.

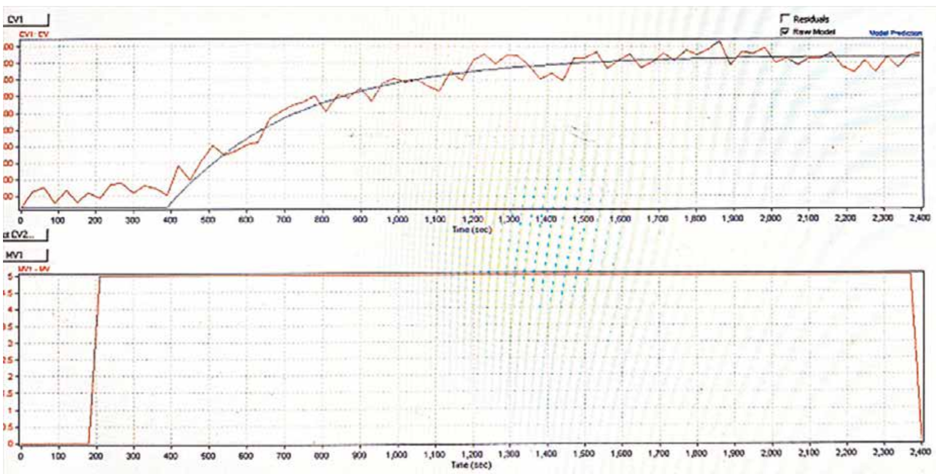

Figure 4.
Dynamics of the first-order transfer function.

foundation for further development of control systems. By leveraging quality data and strategically chosen time windows, we enhance our understanding and modeling of complex system behaviors.

4. APC/PID optimization in PITOPS: A step-by-step guide

PITOPS is an advanced tool designed to simplify and expedite the PID tuning process. Compared to traditional methods such as relay feedback, PITOPS requires less knowledge and time to achieve stable and robust control. Here is a straightforward guide to using PITOPS for PID tuning.

Step 1: Set initial parameters

- Heuristic methods: PITOPS employs heuristic methods to set initial PID parameters, providing a solid starting point for further optimization.

Step 2: Optimize through simulation

- Simulation engine: Use the simulation engine in PITOPS to adjust PID settings based on real-world scenarios. This helps fine-tune parameters effectively.

Step 3: Address non-linearities

- Compensate for issues: PITOPS can address non-linearities such as control valve stiction and dead band, improving the robustness of PID control.

Step 4: Use advanced strategies

- Advanced techniques: Employ advanced control techniques like internal model control (IMC) and direct torque control (DTC) to further optimize PID settings.

Step 5: Iterative refinement

- Continuous improvement: Continuously refine PID parameters using simulations and real-time data to achieve the best performance.

Step 6: Validate and implement

- Integration: Validate the optimized settings and seamlessly integrate them into your control systems.

Easy tuning process

- Automatic tuning: Load the data into PITOPS, select the tuning method, and let the system automatically tune the PID controller for better stability. This process is straightforward and time-efficient.

Customizable tuning

- Adjustable parameters: Further tuning is possible by manually adjusting the values of P, I, and D. PITOPS also allows the selection of specific PID controllers for tuning.

Tuning methods in PITOPS.
PITOPS offers a variety of tuning methods, including:

1. Internal model control (IMC)
   - PI parameters
   - PID parameters

2. Ziegler-Nichols method
    - P parameter
    - PI parameters
    - PID parameters
3. Cohen-Coon method
    - P parameter
    - PI parameters
    - PID parameters
4. Lambda method
    - PI parameters
    - PID parameters
5. PITOPS internal methods
    - Based on integral of absolute error (IAE)
    - Based on integral of squared error (ISE)
    - Based on integral of time-weighted absolute error (ITAE)
    - Based on reduced overshoot

These internal methods provide comprehensive tuning parameter combinations, ensuring the best possible outcomes for various control scenarios.

PITOPS simplifies the PID tuning process, making it an excellent choice for quickly achieving stable and robust control with minimal effort. Some of them are given here. The controller responses obtained through this data-driven approach of PITOPS are shown in **Figures 5–10** for various tuning methods.

PITOPS is not only providing general known methods available in the literature, but it also offers its own proprietary methods to fine-tune systems for achieving optimal steady-state performance. These methods include:

- PITOPS internal method based on integral of absolute error (IAE): This method minimizes the absolute error over time, ensuring a smoother and more stable control response.

- PITOPS internal method based on integral of squared error (ISE): By focusing on reducing the squared error, this method gives higher priority to significant deviations, ensuring more precise control.

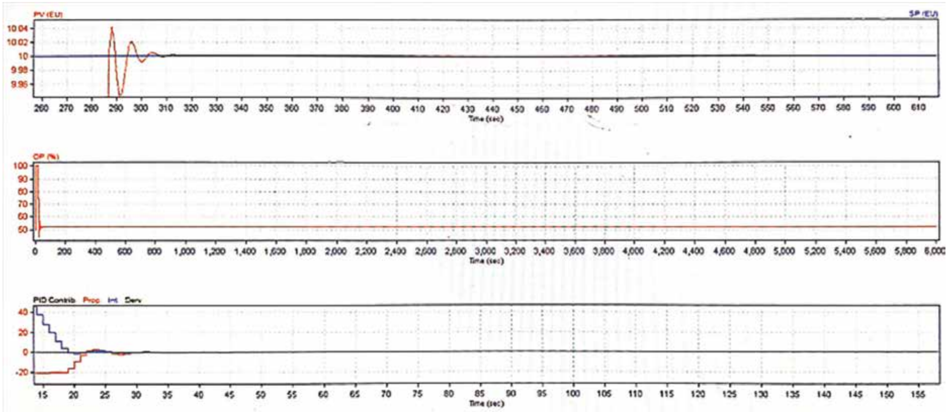

Figure 5.
PID control response for Cohen-Coon method.

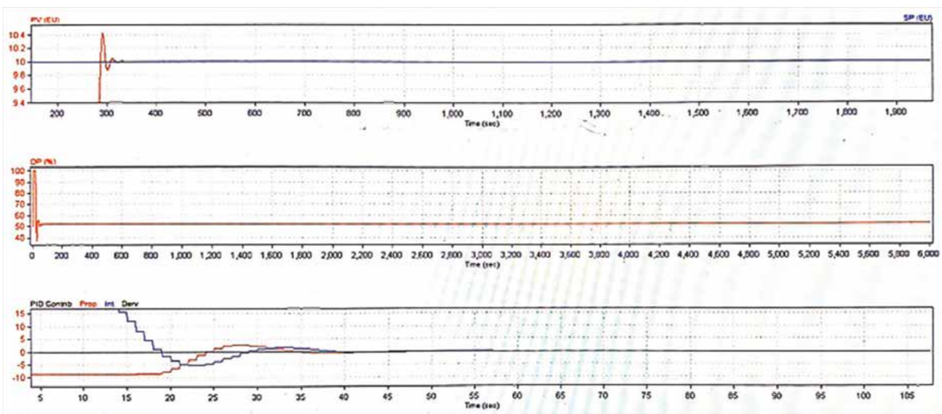

Figure 6.
PID control response for Lambda method.

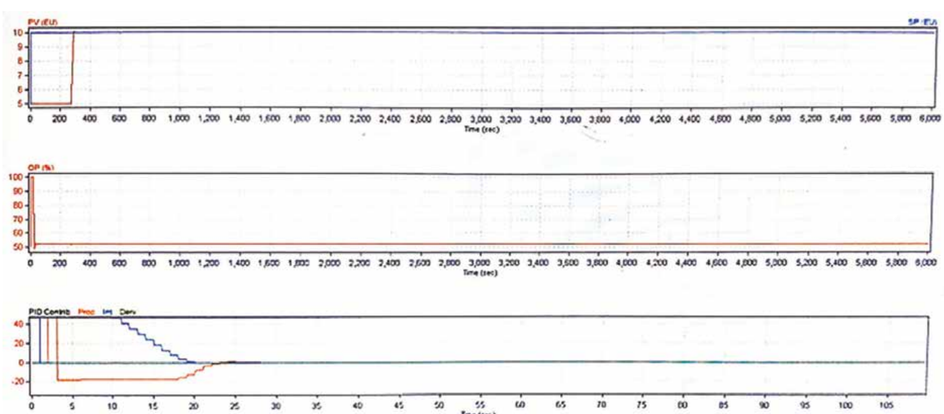

Figure 7.
PITOPS internal method based on IAE.

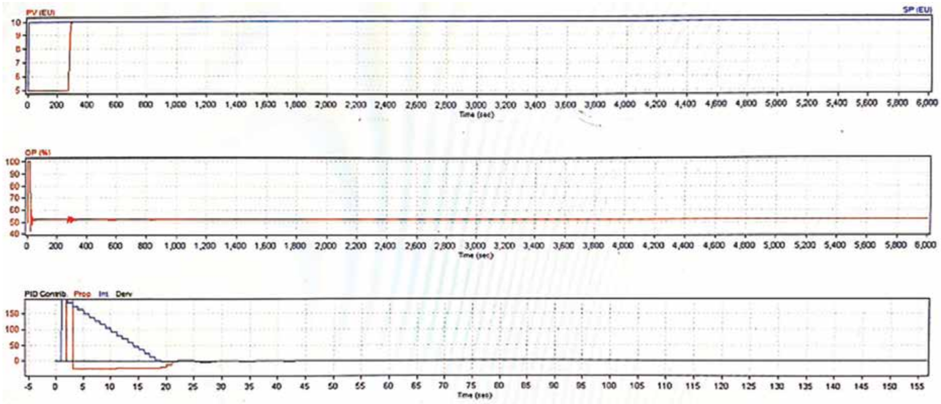

Figure 8.
PITOPS internal method based on ISE.

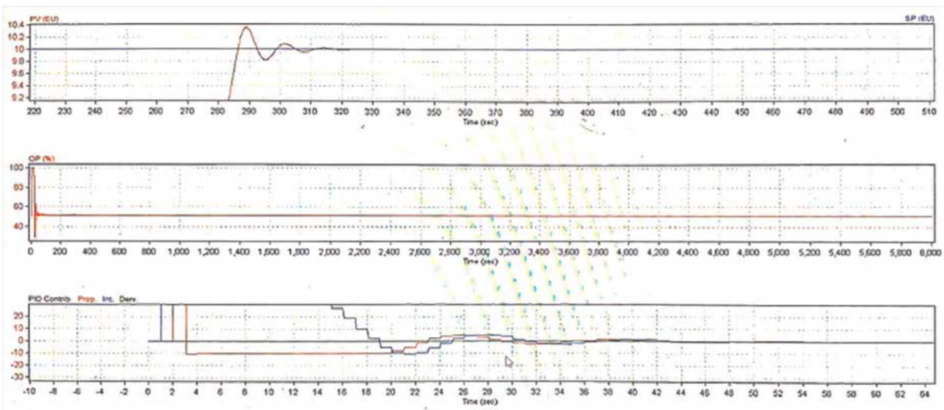

Figure 9.
PITOPS internal method based on ITAE.

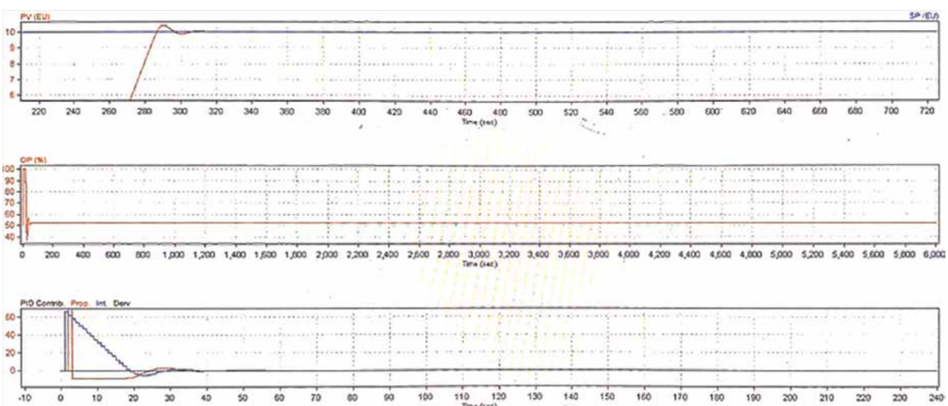

Figure 10.
PITOPS internal method based on reduced overshoot.

- PITOPS internal method based on integral of time-weighted absolute error (ITAE): This approach minimizes errors with a time-weighted strategy, enhancing long-term stability and performance.

- PITOPS internal method based on reduced overshoot: Specifically designed to minimize overshoot, this method improves the system's responsiveness and stability, ensuring robust control.

In conclusion, PITOPS stands out for its user-friendly interface and powerful capabilities, even for users without prior knowledge of PID tuning. The system is designed to automatically tune itself to achieve stability, requiring minimal user intervention. For those looking to further refine the system, slight adjustments can be made to reduce error. Compared to other methods like MATLAB and relay feedback, PITOPS significantly reduces the mathematical complexity and time required for operation.

Furthermore, while traditional methods such as internal model control (IMC) and the Ziegler-Nichols method are available, PITOPS' proprietary methods—based on integral of absolute error (IAE), integral of squared error (ISE), integral of time-weighted absolute error (ITAE), and reduced overshoot—have been observed to provide superior stability and reduced error. This makes PITOPS a highly efficient and effective tool for PID tuning.

5. APC Ziegler-Nichols format for tuning the PID parameters

Advance Process Control **APC Ziegler-Nichols format for tuning the PID parameters**

PID controller equation:

$$\mathrm{u(t)} = \mathrm{Kp}\left[e(t) + \frac{1}{Ti}\int_{0}^{t} e(\tau)d\tau + T_d\frac{de(t)}{\mathrm{dt}}\right] \quad (3)$$

Where:

- u(t) is the control signal.

- e(t) = Setpoint−Measured value is the error signal.

- Kp is the proportional gain.

- Ti is the integral time constant.

- Td is the derivative time constant.

Plant response (discrete-time simulation):

$$\mathrm{CV[i]} = \mathrm{CV[i-1]} + \Delta\mathrm{t}\cdot(-\mathrm{CV[i-1]} + \mathrm{MV[i]}) \quad (4)$$

Where:

- CV[i] is the controlled variable at the i-th time step.

- MV[i] is the manipulated variable (control output) at the i-th time step.
- Δt is the time step size.
- Integral of absolute error (IAE):

$$\mathrm{IAE} = \int_0^{tf} |e(t)|dt$$

- Integral of squared error (ISE):

$$\mathrm{ISE} = \int_0^{tf} \mathrm{e(t)}^2 dt$$

- Integral of time-weighted absolute error (ITAE):

$$\mathrm{ITAE} = \int_0^{tf} t \cdot |e(t)|dt$$

- Normalized root mean squared error (NRMSE):

$$\mathrm{NRMSE} = \frac{1/\mathrm{N}\sum_{\mathrm{i=1}}^{\mathrm{N}} \mathrm{e(i)}^2}{\mathrm{Setpoint} - \min(\mathrm{Measured\ Values})}$$

Ziegler-Nichols tuning parameters:

- Proportional gain: Kp = 0.6·Ku
- Integral time: $\mathrm{Ti} = \frac{Tu}{2}$
- Derivative time: $\mathrm{Td} = \frac{Tu}{8}$

Where, Ku is the ultimate gain and Tu is the ultimate period.
Ziegler-Nichols Kp: 3.0
Ziegler-Nichols Ti: 0.005
Ziegler-Nichols Td: 0.00125
IAE: 1.4694423643476442
ISE: 2.7801348679486675
ITAE: 0.718563981608584
NRMSE: 0.05671185574602207

In this study, we undertook a series of structured operations to fine-tune a PID controller for optimal performance using the Ziegler-Nichols method. Initially, we collected control variable data and set a specific set point value. We then defined a PID controller model to handle proportional, integral, and derivative calculations. To identify an optimal set of PID parameters, we employed differential evolution for global optimization followed by local optimization using the L-BFGS-B algorithm.

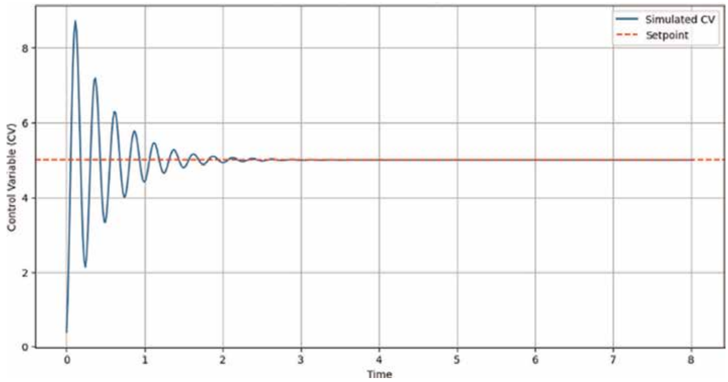

Figure 11.
Simulated control system response with Ziegler-Nichols PID parameters.

Subsequently, we applied the Ziegler-Nichols tuning rules, using the ultimate gain (Ku) and ultimate period (Tu) derived from the optimized parameters, to compute the final PID settings. These operations yielded the following PID parameters: Kp = 3.0, Ti = 0.005, Td = 0.00125. To evaluate the performance of these parameters, we conducted a discrete-time simulation, resulting in performance indices of IAE = 1.4694, ISE = 2.7801, ITAE = 0.7186, and NRMSE = 0.0567. The control response obtained with data-driven approach is shown in **Figure 11**.

These indices indicate that the tuned PID controller achieved a balanced trade-off between responsiveness and stability, providing effective control performance for the given process. The low NRMSE value, in particular, highlights the controller's accuracy in set-point tracking, reflecting successful PID tuning.

6. APC manual

In this study, we performed a series of advanced optimization operations to fine-tune a PID controller. By initially employing differential evolution for global optimization, followed by the L-BFGS-B algorithm for local optimization, we effectively identified the optimal PID parameters. This process included defining a PID controller, formulating an objective function based on various error metrics, and conducting a discrete-time simulation to evaluate performance. The mathematical calculations involved in this method are shown below.

PID controller update equation (APC manual):

$$\text{Control signal} = \text{Kp}\left[e(t) + \frac{1}{Ti}\cdot integral + Td \cdot derivative\right] \tag{5}$$

Where:

- error = setpoint−measured_value
- integral $= \sum_{\text{i}=1}^{\text{N}} error \times \text{dt}$ (cumulative sum of the error over time)

- derivative $= \frac{error-previous\_error}{dt}$
- Kp, Ti, Tdare the proportional, integral, and derivative gains, respectively.

System dynamics (discrete-time simulation):

$$CV[i] = CV[i-1] + dt \cdot (-CV[i-1] + MV[i])$$

Where:

- CV[i] is the controlled variable at step iii.
- MV[i] is the manipulated variable (output of the PID controller) at step iii.
- dt is the time step size.
- Integral of absolute error (IAE):

$$\mathrm{IAE} = \int_0^T |e(t)| dt$$

- Integral of squared error (ISE):
- $\mathrm{ISE} = \int_0^{tf} \mathrm{e(t)}^2 dt$
- Integral of time-weighted absolute error (ITAE):
- $\mathrm{ITAE} = \int_0^{tf} t \cdot |e(t)| dt$
- Normalized root mean squared error (NRMSE):

$$\mathrm{NRMSE} = \frac{1/\mathrm{N} \sum_{i=1}^{N} \mathrm{e(i)}^2}{\text{Setpoint} - \min(\text{Measured Values})} \tag{6}$$

The fine-tuning resulted in the following PID parameters: Kp = 4.9983, Ti = 0.0100, Td = 0.1341. The corresponding performance indices were IAE = 0.6403, ISE = 0.8995, ITAE = 0.2245, and NRMSE = 0.0323. These results demonstrate that the optimized PID controller achieved low error metrics and effective control performance, with the low NRMSE indicating precise set point tracking and minimal deviation. The control response obtained for this method is shown in **Figure 12** and the parameters obtained with method are calculated and it was observed that compared to other methods it yields the best results in terms of IAE, ISE, ITAE, NRMSE (**Table 1**).

Fine-tuned Kp: 4.998262673424511
Fine-tuned Ti: 0.01000098344062561
Fine-tuned Td: 0.1340666385289393

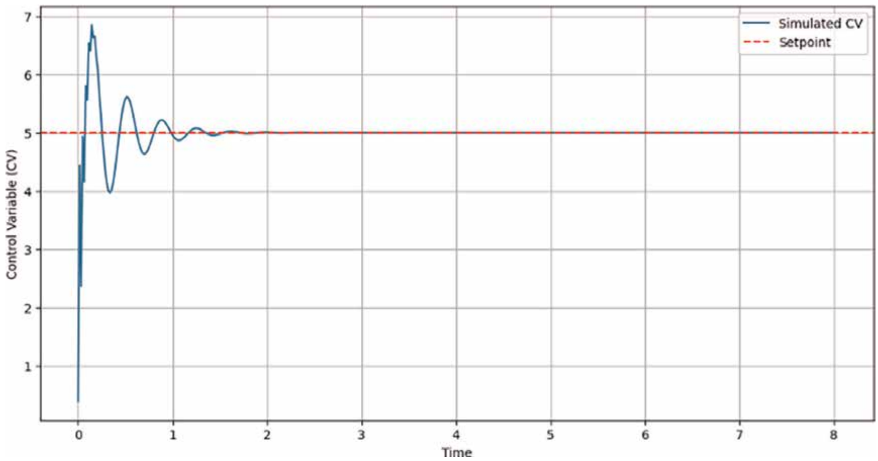

Figure 12.
Simulated control system response with fine-tuned PID parameters.

Parameters	Relay feedback	PITOPS system identification		APC Ziegler-Nichols format for tuning the PID parameters	APC manual
		First data set analysis	Second data set analysis		
IAE	1.386477e +37	0.500 EU	0.44 EU	1.4694423643476442	0.6402990671633989
ISE	1.123906e +74	—	—	2.7801348679486675	0.8995189644873076
ITAE	1.099141e +39	—	—	0.718563981608584	0.22451696055648687
NRMSE	5.052398e +35	0.208	0.124	0.05671185574602207	0.03225864255732831

Table 1.
Comparison of the various methods.

IAE: 0.6402990671633989
ISE: 0.8995189644873076
ITAE: 0.22451696055648687
NRMSE: 0.03225864255732831

7. Conclusion

In comparing different methods for tuning PID controllers and system identification, we found that PITOPS and APC manual optimization methods are the most effective, but they have different strengths. PITOPS is user-friendly and straightforward, making it accessible even to those without extensive knowledge in programming or mathematical optimization. It automatically tunes the PID controller and performs accurate system identification, providing high accuracy and low error metrics. We can further fine-tune the PID parameters in PITOPS to achieve even lower error rates, making it a highly preferable option. On the other hand, APC manual optimization delivers the best overall performance with the lowest error rates and highest control precision. However, it requires advanced programming skills and a strong understanding of complex mathematical algorithms. While traditional relay feedback and Ziegler-Nichols methods offer some improvements, they are more complicated and less efficient. Therefore, PITOPS is ideal for those seeking an easy-to-use solution, with the added benefit of fine-tuning for better performance and effective system identification, while APC manual optimization is best for those who can manage its complexity to achieve optimal results.

Acknowledgements

The authors would like to acknowledge the support of the Registrar, Principal, HOD-Department of EIE and faculty for their continuous support in the preparation of this book chapter.

Conflict of interest

The authors declare no conflict of interest.

Author details

Davarapalli Kishore[1*], Sabbarapu Ganesh[2] and Vaska Lokesh[3]

1 Department of EIE, Adikavi Nannaya University, Rajamahendravaram, Andhra Pradesh, India

2 CSIR-IICT, Hyderabad, Telangana, India

3 CSIR-NGRI, Hyderabad, Telangana, India

*Address all correspondence to: kishorenit.trichy@gmail.com

References

[1] Astrom KJ, Hagglund T. Automatic tuning of simple regulators with specifications on phase and amplitude margins. Automatica. 1984;**20**(5):645-651

[2] Bajarangbali R, Majhi S. Relay based identification of systems. International Journal of Scientific & Engineering Research. 2012;**3**(6):1-4

[3] Rahul G, Pandyan KRR. A Systematic Approach for Identification of SOPTD Processes using a Relay Feedback with a Fractional Order Integrator. Acta Polytechnica-Hungarica. 2023;**20**(2): 243-263

[4] Luyben WL. Derivations of transfer functions for highly non-linear distillation columns. Industrial and Engineering Chemistry Research. 1987; **26**:2490-2495

[5] Chinta Sankar rao. Subspace identification of unstable systems [thesis]. India: IIT-Madras; 2015

[6] Chidambaram M, Sathe V. Relay Autotuning for Identification and Control. 1st ed. Cambridge, United Kingdom: Cambridge University Press; 2014. ISBN: 978-1-107-05871

[7] Kishore D, Smruti Rajan S, Anand Kishore K, Panda RC. An improved identification and control of 3×3 multi-input multi-output system using relay and subspace method. Indian Chemical Engineer. 2018;**61**(1):87-101. DOI: 10.1080/00194506.2018.1486237

[8] Kishore D, Lokesh V. Perspective chapter: Modeling and identification of two tank system using relay feedback – An experimental approach [internet]. In: PID Control for Linear and Nonlinear Industrial Processes. London, UK: IntechOpen; 2023. DOI: 10.5772/ intechopen.106568

Chapter 4

Analytical Design of a Closed Control Loop Controller, Based on a Suboptimal Kalman-Busy Filter

Yuri V. Kim

Abstract

The Analytical Design (AD) of a closed, negative feedback control loop, when only single design criteria (potentially achievable accuracy) can be considered at the first stage of development for the synthesis of desired system dynamics. Such an approach, based on a modified suboptimal Kalman-Busy Filter (KBF) with Bounded Grows of Memory (FBGM), was presented in several previous author's papers. In some cases, the required optimal controller should work, mainly, in the stationary stabilization mode in stationary conditions and, actually, is a regulator. In these cases, FBGM can be essentially simplified to a stationary Kalman's state estimator, with a switched matrix weight coefficient (transient/stationary). The coefficient can, practically, be found rather from the conventional conditions for providing the system sufficient dynamics, than from the solution of KBF Riccati eq. A successful tuning makes the steady state accuracy be close to the optimal, provided by the KBF. The estimator is used for the estimation/filtering and control/regulation purposes simultaneously. This approach is considered in the below chapter to draw developer's attention. A simple example of the 2nd order unit, assuming regulation of system angular position and angular velocity is presented.

Keywords: analytical design, dynamic system, state equation, Kalman filter, Riccati equation, white noise, quality criterion

1. Introduction

As far as the author knows, the term "Analytical design (AD)" appeared in the seventieth-century in the former USSR. There were published [1, 2], where authors applied optimal filtering [3, 4] (KBF) and optimal control theory [5, 6] to find an optimal dynamic for the typical aerospace applications. The fundamental base for this approach was presented by R. Kalman in Ref. [3], as a "dual (separation) principle". Further general consideration of above-mentioned optimal estimation/filtering and control theories and applications can be found in the following publications [7–11]. It must be mentioned that despite the term AD was not directly used in the Western scientific literature, in fact, a similar approach was presented in some publications [7, 10]. This approach can be called a synthesis of optimal closed control loop dynamic with minimum of the quadratic quality criterion. To be short, in the below consideration, the term AD is used. What is essential for the engineering practice is that the AD is very consistent with developing generic International methodology and tools (MATLAB/Simulink), developed by MathWorks under the concept of Model-Based Design [12].

Talking about the direct using of KBF in AD, we need to take into account that stochastic system model, assumed in KBF theory [3, 4], is usually very idealistic. In reality takes place more complex than centered Gaussian white noises (the mathematical abstraction) disturbances and measured errors. We can clearly see this when comparing experimental data with KBF idealistic model, especially for a long-time observed intervals. Often, exact tuning the KBF in accordance with its idealistic model can lead to an unstable filter that cannot be practically used. An additional problem is the limited calculation capability of used real-time computers. Even today, for essentially developed sophisticated Aerospace on-board computers [13, 14] computational and software debugging difficulties for the direct KBF implementation have not been completely overcome. However, if implemented, then often it can result in useless wasting of computational power and idle processing where simpler and more reliable results can be achieved. All above says that almost always is worth sacrificing theoretical minimum, provided by KBF, for the robust and simple implementation of a suboptimal KBF. Such suboptimal versions of KBF the author began developing in the early seventieth-century. At this time he was urged by the necessity to simplify and implement it in the first generation of USSR aviation on-board digital computers (OBC) that had then very modest computational capabilities. Then was developed a sufficient suboptimal KBF form- the FBGM, where the filter weight coefficient matrix K was presented as consisting from two components; not stationary $\tilde{K}(t)$ and stationary K^* (constant). Both parts were pre-calculated and the not stationary one ($\tilde{K}(t)$) was approximated by simple polynomial functions. A special analytical procedure, based on using of the ratio signal/noise (estimation indicators), was developed. Working in the steady state with the stationary coefficient K^*, filter periodically can be restarted to upgrade the estimates. Detailed information about this can be found in [15–17]. In this final analytical form, the filter coefficients were "stitched"into the OBC memory. Some examples of using the filter as the estimator and the controller simultaneously were published in [18] (pair of satellites, flying in formation (FF)) and [19] (navigation accelerometer with the electric spring). The general approach of using FBGM for the AD was published in [20].

The following approach is presented below. The filter is suggested to be used as the filter and controller at the same time. It is in the scope of the linear, time-invariant systems under the assumptions of the linear KBF theory. It is considered a special, however, a wide class of control systems: the stabilization systems, when the controller mainly acts as a regulator, stabilizing regulated parameters with respect to the desired set level. Two modes are typical for such a system: the initial activation (transient process) and the stabilization (stationary/steady filtering process). For such a case the FBGM can be represented (degenerats) as a simple form: a stationary filter with switching in time bandwidth: wide at the transient stage and narrow at the filtering stage. With proper tuning, the matrix of the filter coefficients is robust and can provide estimation (control) accuracy close to the theoretically achievable maximum (minimum of estimation errors), as with KBF.

As with regards to the AD and Model-based design, the filter synthesis can be considered as the first step of the design. Preliminary (conceptual model – CM or Low Fidelity Model - LFM) design model. It can be used to synthesize the system state estimator/observer and the controller and to evaluate system feasibility and potentially achievable performance. Further steps of the design take into account some realistic constraints and restrictions, and develop this LFM, by adding more detail elements, to complex, non-linear, and not-stationary High Fidelity Model (HFM).

More recent work about AD taking into account some additional constraints can be found in [21].

It must be added that recent developments in the Control Theory introduced new effective methods for syntheses controllers that can work in more complex than stabilization operational modes and conditions, demonstrating better than LTI - KBF dynamics. At least, two new approaches should be mentioned here: the Fuzzy Logic (FL) and the Artificial Intelligence (AI).

Works, dedicated to using FL for synthesis of PID controller and for Fault detection and diagnostic, were published by Prof. C. Volosenku in Refs. [22–24].

Broad survey of using AI in various engineering applications for technical diagnostic readers can be found in Ref. [25].

2. Suboptimal Kalman-Busy filter

Let us assume that we have had acceptable for further analytical design system conceptual model, which is LFM; linear time invariant (LTI), fully observable and controllable model of a stochastic system, presented by the following state equation [4].

$$\begin{aligned} \dot{x} &= Fx + Gw, \\ z &= Hx + v, \end{aligned} \tag{1}$$

where: x is an n state vector, w is an m -external disturbances vector, z is a p measurement vector, v is a p is measurement noise vector, F is $(n \times n)$ matrix of system, G is an $(n \times m)$ matrix of disturbances, H is $(p \times n)$ matrix of measurements.

Let us assume about (1) that F, G, H are given matrixes and

$$\begin{aligned} &E[x(t_0)] = 0, \;\; E[w(t)] = 0, \;\; E[v(t)] = 0, \\ &E\left[x(t_0)x^T(t_0)\right] = P_0, \\ &E\left[w(t)v^T(\tau)\right] = E\left[v(t)w^T(\tau)\right] = E\left[w(t)x^T(t_0)\right] = 0, \\ &E\left[w(t)w^T(\tau)\right] = Q(t)\delta(t-\tau), \\ &E\left[v(t)v^T(\tau)\right] = R(t)\delta(t-\tau), \end{aligned} \tag{2}$$

where: P_0 is the covariance matrix of the initial state, $R(t)$ is the measurement noise covariance matrix, $Q(t)$ is the disturbance noise covariance matrix, $\delta(t-\tau)$ is the Dirac's standard pulse. In other words, $w(t)$ and $v(t)$ are normal Gaussian white noise processes. Often, Q and R are diagonal and constant matrixes of the white noises $w(t)$ and $v(t)$ accordingly. Instead of the idealistic white noises, in a wide frequency band, they can be interpreted as correlated processes with the following matrixes spectral densities

$$\begin{aligned} Q &= \left[2\sigma_{wi}^2 T_{wi}\right]_{diag}, \; {\scriptstyle i=1,2,..m} \\ R &= \left[2\sigma_{vj}^2 T_{vj}\right]_{diag}, \; {\scriptstyle j=1,2,..m} \end{aligned} \tag{3}$$

where σ_{wi} and σ_{vj} are the standard deviations, T_{wi} and T_{vj} are the correlation times of these stochastic processes w and v.

We want to find an optimal $\hat{x}$ estimate of the system state vector x, using measurements z.

The estimation error $\tilde{x} = \hat{x} - x$ covariance matrix is as follows $P = E\left[\tilde{x}(t) \cdot \tilde{x}^T(t)\right]$.

Let us use the following optimization criteria

$$J_{\min} = \min(traceP) \tag{4}$$

Then, the Kalman-Busy filter (KBF) is to be considered as the optimal estimator [4–6].

Let us consider KBF in its continuous form [6].

$$\begin{aligned} &\dot{\hat{x}} = F\hat{x} + K(z - H\hat{x}), \ \ \hat{x}_0, \\ &K = PH^T R^{-1}, \\ &\dot{P} = FP + PF^T - PH^T R^{-1} HP + GQG^T, P_0 \end{aligned} \tag{5}$$

where: $K = K(t)$ is matrix weight coefficient, $P = P(t)$ is estimated state vector errors covariance matrix.

Matrix P is determined by the 3-rd equation of (5), aka as the differential equation Riccati of the KBF filter theory).

Eq. (5) provides an optimal solution with minimum (4), when measurements z are available $z \neq 0$, then the weight coefficient $K(t)$ in (5) is not equal to zero and takes place continuous update of the current estimates. This mode of working (5) is called the "filtering mode". If, for some reason, the measurement vector z is not available ($z \equiv 0$) then, for this time interval the filter (5) can also be switched to another mode of operation, called the "prediction mode". In this mode weight coefficient in (5) should be set to zero $K(t) \equiv 0$ and initial conditions should be taken as the last available past estimates: $\hat{x}_p(t_0) = \hat{x}_f\left(t_f\right)$. Then original KBF equations degenerate to the following:

$$\begin{aligned} &\dot{\hat{x}} = F\hat{x}, \ \ \hat{x}_0 = \hat{x}_p, \\ &K \equiv 0, \\ &\dot{P} = FP + PF^T, \quad P_0 = P_p \end{aligned} \tag{6}$$

The KBF, providing optimal system state estimation, was presented by R. Kalman in the early sixtieth-century, in the recursive, (convenient for direct digital computer implementation) [5] and analogue [6] forms. Right away, its presentation attracted attention of many developers to apply it for diverse technical applications. However, low computing power of electronic analog and digital computers at that time hardly allowed implementing the KBF practically, specifically for real-time aerospace applications in on-board computers. And even today, despite significant progress, achieved with on-board computer characteristics [7, 8], implementation of KBF still is quite a challenging problem for a programmer, requiring from him a certain experience and skill. It is worth noting that for many practical applications, direct KBF implementation would follow just grinding the steady state data in idle and/or creating the computational divergence trend for the filter, followed by the system instability. Therefore, the author's position, related to the considered problem (using KBF for analytical design) is that is almost always worth analyzing and implementing a suboptimal KBF (sacrificing by the theoretical maximum estimation accuracy, but making the filter robust and simple for its implementation and debugging) than direct optimal KBF (5) implementation (Quid pro quo).

Such sub-optimal form of KBF - the Filter with Bounded Growth of Memory (FBGM) is presented below. The simplest, but practically common and important case is when the model (1) and the filter (5) are linear stationary (time-invariant) systems (LTI).

The filter development stages and it's not-stationary form for suboptimal estimation in transient process readers can be found in Refs. [1, 2, 9].

The filter operation in time is consecutively divided, by switching its weight coefficient K, in two periods: estimation of the transient process $t < t_*$ (initial filter-IF) and estimation of the steady process $t > t_*$ (steady filter-SF). It can be presented by the following equations

$$\dot{\hat{x}} = F\hat{x} + K(z - H\hat{x})$$
$$K = \begin{cases} K_{IF}^* \text{ if } t_0 \le t \le t_*, \\ K_{SF}^* \text{ if } t > t_*, \\ 0 \quad if \;\; z \equiv 0 \end{cases} \tag{7}$$

In (7) K_{IF}^* and K_{SF}^* are filter matrix weight coefficients.

K_{IF}^* for IF and K_{SF}^* for SF filtering mode, and $K = 0$ for the prediction mode. These coefficients and boundary time t_* can be found empirically or from the KF sub-optimization procedure [1, 2, 9]. The suboptimal coefficient for the transient process $K_{IF}^* = \tilde{K}(t)$ can be found as the polynomial approximation of KBF solution (3-rd differential equations, Riccati type) in (5). And for the stationary (steady) state $(t \to \infty, \dot{P}(t \to \infty) = 0)$ estimation K_{SF}^* is, in fact, optimal and can be found from the equations below

$$K_{SF}^* = P^* H^T R^{-1} \tag{8}$$

$$F^* P^* + P^* F^{*T} - P^* H^T R^{-1} H P^* + GQG^T = 0 \tag{9}$$

where $F^* = F - K_{SF}^* H$.

In this steady state case, the differential non-linear Riccati equation in [5] degenerates to the algebraic eq. (9) that can be solved numerically and required P^* and $K^* = K_{SF}^*$ can be found.

3. Closed loop, negative feedback control, and estimation

The estimation error $\tilde{x}$ is presented by the following formula

$$\tilde{x} = x - \hat{x} \tag{10}$$

Let us assume that all state vector x components are directly measured (in some cases this assumption is not directly applicable).

Then, using eqs. (1) and (5), we can get the following matrix equation

$$\dot{\tilde{x}} = \tilde{F}\tilde{x} + Gw - Kv \tag{11}$$

where $\tilde{F} = F - KH$ and K can be determined as in (9).

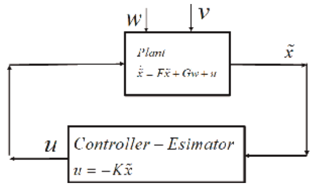

Figure 1.
Sate estimator (filter) as system closed loop controller.

Closed negative feedback control loop with the state estimator performing both: estimation and control tasks is shown in **Figure 1**.

Let us talk about the selection of the appropriate control-estimation matrix coefficient K for (7), (11).

As it can be seen from (7), the KBF optimal matrix weight coefficient is entirely dependent on the matrixes Q and R that in many cases not known. System disturbance $w(t)$ (matrix Q) usually, hardly can be approximated by white noise and is rather almost a deterministic signal (for example, close to the bias).

However, the measuring error $V(t)$ (matrix R) often can be indeed considered as a Gaussian wide bandwidth white noise with a small correlation time (τ).

The following simple idea for choosing suboptimal coefficients K_{IF}^{*} and K_{SF}^{*} in (7) is used. Both are constant and are chosen by rather conventional engineering practices than optimal KBF formulas. Two filter studies are considered: a-Transient; IF mode with high band-with and b-Stationary; SF mode with low bandwidth Physically, a closed control loop (11) can be considered as an LTI filter having a big cut frequency ω_{IF}^{*} (wide bandwidth) during the transient process $t_0 \leq t \leq t_*$, and a low cut frequency $\omega_{SF}^{*} < \omega_{IF}^{*}$.

Its weight matrix coefficient K^* is switched by a step at the time $t = t_*$ accordingly to provide desired ω_{SF}^{*} and ω_{IF}^{*}. By choosing an appropriate ω_{IF}^{*} (wide filter bandwidth) we can achieve a high performance and a short duration of the filter transient process t_*. After the bandwidth is narrowed, a steady state filtering process will take place, which has the purpose of passing through the filter slowly changing in time (almost deterministic) input signal and cutting high frequency wide range measured noise.

In [19] author proposes to apply a rather conventional engineering than pure mathematical optimization approach: splitting system operation time into two typical parts transient process ($t_0 < t < t_*$) and steady operation (filtering) process ($t_* < t$).

- At the first (transient) stage original system model (1) is considered as "quasi-deterministic"

$$\begin{aligned} \dot{x} &= Fx, x_0 \\ z &= Hx \end{aligned} \tag{12}$$

and neglecting by noises w and v choosing the gate K_{IF}^{*} as being aimed by the single criterion: fast decaying of the transfer process, caused by the initial conditions x_0.

- At the second (steady) stage, the original system model [1] is considered as "purely statistic", when after decaying of the transferring process the steady state filtering takes place: passing through the low-frequency bandwidth almost deterministic signal measuring with an additive random white noise.

$$\begin{aligned} \dot{x} &= Fx, x_0 = 0 \\ z &= Hx + v \end{aligned} \tag{13}$$

Then, K_{SF}^* can be found from the equation

$$F^*P + F^*P + K_{SF}^{*T} R K_{SF}^* = 0 \tag{14}$$

where $F^* = F - K_{SF}^* H$,

This equation is covariance equation for the filter estimation errors caused by the measuring noise $v(t)$

$$\dot{\tilde{x}} = F^*\tilde{x} - K_{SF}^* v \tag{15}$$

In fact, in practice the real physical processes $w(t)$ and $v(t)$ have more complex than Gaussian stationary white noise structure and hardly can be expressed by the matrixes Q and R, assumed in KBF theory. However, this abstraction could be helpful for practical needs if some "appropriate" levels of Q and R are taken for the AD that would result in a solution that is compatible with conventional engineering practice.

It is presented that both matrixes control/filtering gates K_{IF}^* and K_{SF}^* can be chosen from usual for the Automatic Control system theory [4, 10] engineering criteria, such as stability, overshooting, and filtering bandwidth, rather than from solving KBF Riccati equation.

Let us consider the filter SF characteristic polynomial

$$\Delta(s) = sI - (F + K^*H) = s^n + b_1 s^{n-1} + \ldots\ldots + b_{n-1}s + b_0 \tag{16}$$

and choosing its coefficients b_i to have in (3.6) desired roots. For example, for the multiple real number roots λ

$$\Delta(s) = (s + \lambda)^n \tag{17}$$

where λ is the physical meaning of system bandwidth.

The matrix coefficient K_{IF}^* can be found, considering desired bandwidth λ and standard polynomial coefficients in (17).

It also should be noted that if the filter realization (estimation process only) is digital, then the following trick to eliminate transient process can be applied

$$\hat{x}_1 = H^{-1} z_1 \tag{18}$$

The estimate vector at the first step is equal to the first measurement.

4. Example

Let us consider simple single-axis X flat rotation of a spherical rigid body, with angular dynamics equation

$$\ddot{x} = \frac{M}{J} \tag{19}$$

where $\ddot{x}$ is point acceleration, J is its axial inertia $J = J_x$, M is external disturbing torque.

This equation can be rewritten in the state equation form as follows

$$\begin{cases} \dot{x}_1 = w_1 \\ \dot{x}_2 = x_1 \end{cases} \tag{20}$$

where x_1 is point velocity $x_1 = \dot{x} = V$, x_2 is point position $x_2 = x$, w_1 is the specific external disturbing force, applied to the point $w_1 = \frac{M}{J}$.

Let us also assume that system for (10) (plant) states vector x both components are measured by some position and angular velocity sensors

$$z = Hx + v \tag{21}$$

where.

$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ is measurement matrix, $v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ is measuremnt error vector

Gaussian white noises having covariance matrixes $Q = \begin{bmatrix} q_1 & 0 \\ 0 & 0 \end{bmatrix}$ and

$R = \begin{bmatrix} r_1 & 0 \\ 0 & r_2 \end{bmatrix}$.

In this example, we assume that both state vector components x_1 and x_2 are measured and estimation filter will be used for the control purposes in the closed control loop as well. In other words, (4) is a common quality criterion for the estimation and the control.

If minimum diagonal P^* is a common quality criterion for estimation and control of (19), then.

Applying to the system KBF and subtracting from the system vector its estimation, we can get equation of estimated errors

$$\begin{cases} \dot{\tilde{x}}_1 = -k_{12}\tilde{x}_2 - k_{12}v_2 + w_1 \\ \dot{\tilde{x}}_2 = \tilde{x}_1 - k_{22}\delta\tilde{x}_2 - k_{22}v_2 \end{cases} \tag{22}$$

that can be interpreted as the negative feedback closed control loop. Converting (22) to the second-order differential equation, we will get the following 2-nd order differential equation unit

$$\ddot{\tilde{x}}_2 = -\frac{1}{T^2}\dot{\tilde{x}}_2 - 2\frac{d}{T}\tilde{x}_2 - \frac{1}{T^2}\dot{v}_2 - 2\frac{d}{T}v_2 + w_1 \tag{23}$$

where $T = \frac{1}{\sqrt{k_{12}}}$ is the unit time constant, $d = \frac{k_{22}}{2\sqrt{k_{12}}}$ is specific damping coefficient.

The above example was simulated with Simulink simulator.

The following data were used:

$J = 50$ kgm^2 is satellite inertia, $\sigma_M = 10^{-6}$ Nm is satellite disturbing torque (noise) standard deviation (STD), $T_M = 10$ s is disturbing torque correlation time,

$\sigma_{v1} = 0.01$ deg/sis satellite angular rate measured noise STD and correlation time $T_{v1} = 0.1$ s.

$\sigma_{v2} = 0.1$ degis satellite attitude measured noise STD and $T_{v2} = 0.1$ s is satellite attitude error correlation time.

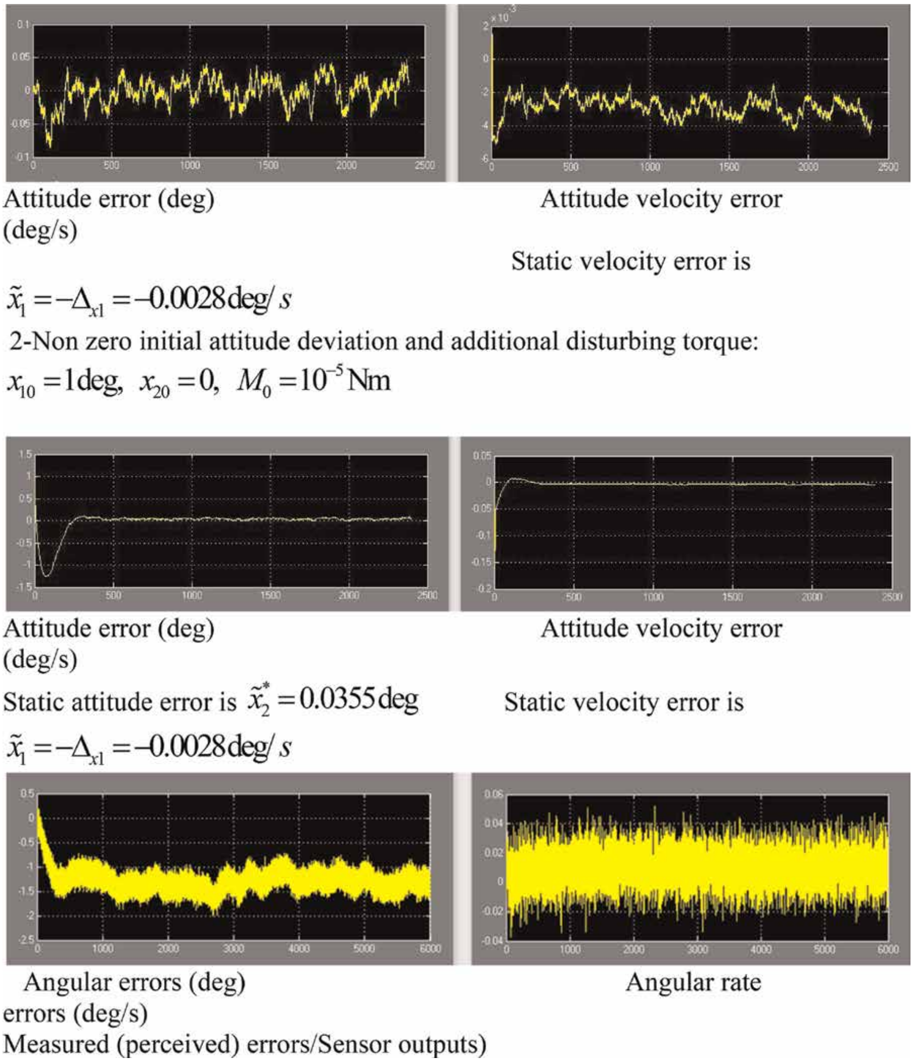

Figure 2.
Attitude and attitude velocity are measured. Control with optimal estimator, acting simultaneously also as the controller.

For this example, steady state algebraic Riccati eq. (9) cannot be solved analytically. The numerical solution, which was found, gives the following parameters:

$\sigma_1 = 0.0008$ deg$/s$ $\quad \sigma_2 = 0.05$deg are STD of estimation errors and $k_{11} = 0.0054, k_{12} = 0.000272, k_{21} = 0.272, k_{22} = 0.0236$ are the filter/controller coefficients.

These coefficients can be converted into the time constant $T = \frac{1}{\sqrt{k_{11}k_{22} - k_{12}k_{21} + k_{12}}} = 55.3$ s and the specific damping coefficient $d = \frac{k_{11}+k_{22}}{2T} = 2.62 \cdot 10^{-4}$. So, the calculated optimal damping coefficient for the KBF cannot practically be used as not providing sufficient damping.

Therefore, to demonstrate the applicability from the practical point of view a different (suboptimal) approach.

The coefficient d was set as $d = 0.707$, but chosen filter time constant - is optimal $T = 55.3$ s.

There were added some additional factors to demonstrate essential differences with the idealistic model for KBF; constant disturbing torque $\sigma_{M_0} = 10^{-5}$ Nm and constant angular velocity measurement error (bias) $\Delta_{x1} = 10\text{deg}/h = 0.0028\text{deg}/s$.

For IF mode was taken that: $T_{IF} = 3$ s, $d = 0.707$. Switching IF/SF time was $t^* = 9$ s.

The simulation results are presented in **Figure 2**.

1-Zero initial conditions and additional disturbing torque:
$x_{10} = 0,\ \ x_{20} = 0,\ \ M_0 = 0.$

The time of FBGM switch from IF to SF mode is $t^* = 9$ s.

5. Conclusion

AD, based on using the sub-optimal KBF in closed control loop for the filtering and control-stabilization purposes simultaneously, can be considered as a helpful synthesis method for a broad spectrum of engineering applications. Simple implementation of the two stages stationary filter with switched: wide/narrow bandwidth can guarantee a satisfactory system dynamic with close to optimal steady state KBF accuracy.

Acknowledgements

Author acknowledges the Academic Editor Prof. C. Volosenku and the Publisher IntechOpen contributions to the chapter editorial improvement and publishing.

Author details

Yuri V. Kim
Canadian Space Agency, Canada

*Address all correspondence to: yurikim@hotmail.ca

References

[1] Krasovsky A. Automatic Flight Control Systems and Analytical Design. Moscow: Nauka, Rus. M; 1973

[2] Letov A. Flight Dynamics and Control. Moscow: Nauka, M; 1979

[3] Kalman RE. A new approach to linear filtering and prediction problems. ASME J. Basic Eng. 1960;**82**:34-45

[4] Kalman RE, Bucy RS. New results in linear filtering and prediction theory. ASME J. Basic Eng. 1961;**80**:193-196

[5] Bellman R. Dynamic Programming. Princeton, N.J: Princeton Univ. Press; 1957

[6] Pontriagin L, Boltiasky V, Gamkrelidze R, Mischenko E. Mathematical Theory of Optimal Processes, Moscow: Rus. Nauka, M; 1976

[7] Bryson AE, Yu-Chi Ho J. Applied Optimal Control. Levittown, PA: Taylor & Francis; 1975. pp. 364-373,369,457

[8] Chernousko FL, Kolmanovskiy VB. Optimal Control under Random Disturbances. Moscow: Nauka, M; 1978. p. 37, (Rus.)

[9] Gelb A et al. Applied Optimal Estimation. Cambridge, MA: The M.I.T. Press; 1974. p. 2. 67, 131

[10] Kwakernaak H, Sivan R. Linear Optimal Control Systems, (Rus. Transl.). MMoscow: Mir, M; 1972. pp. 83-88

[11] Zarach P, Missof H. Fundamentals of Kalman filtering: Practical approach. In: AIAA, Progress in Astronautics and Aeronautics. Ch. 4 ed. Vol. 190. Reston, VA:AIAA; 2000

[12] Aarenstrap R. Managing Model-Based Design. Natic, MA: Math Works, Inc.; 2015

[13] Eickhoff J. Onboard Computers, Onboard Software and Satellite Operations. Berlin: Springer; 2012. pp. 1-278

[14] You Z, editor. Space Microsystems and Micro/Nano Satellites. Ch. 4 ed. UK/US: Butterworth-Heineman; 2018. pp. 120-146

[15] Kim YV. An approach to suboptimal filtering in applied problems of information processing, (Rus.). In: Technicheskaya Kibernetica, USSR J. Science Academy. Vol. 1. NY: Engl. Transl. Scripta Technica Inc.; 1990. pp. 109-123

[16] Kim YV, Kobzov PP. Optimal filtering of a polynomial signal, (Rus.). In: Technicheskaya Kibernetica, USSR J. Science Academy. Vol. 2. NY: Engl. Transl. Scripta Technica Inc.; 1991. pp. 120-133

[17] Kim YV. Kalman filter decomposition in the time domain using Observabiliy index. In: IFAC, Seul, July 6–11, 2008, International Conference Proceedings. Vol. 2. NY: Curran Associates Inc.; 2009. pp. 625-630

[18] Vukovich G, Kim Y. The Kalman filter as controller: Application to satellite formation flying problem. Int. J. Space Science and Engineering. 2015; **3**(2):148-170, ISSN print 2048-8459

[19] Kim YV. Review of Kalman filter developments in analytical engineering design. In: Kim YV, editor. Kalman Filter, Engineering Applications. Ch. 1 ed. London, UK: IntechOpen; 2023

[20] Kim YV. Kalman filter and satellite attitude control system analytical design. Int. J. Space Science and Engineering. 2020;**6**(1):82-103

[21] Dozortsev V, Folomkin U. Analytical design of regulators with limited control, (Rus.). In: Proceedings of 23 International Scientific Conference “Mathematical Methods in Technic and Technology”, Russia, Smolensk. Vol. 2. 2010. pp. 106-109

[22] Volosencu C. Tuning Fuzzy PID controller. In: Panda RC et al., editors. Introduction to PID Controllers-Theory, Tuning and Application to Frontier Areas. Ch. 7 ed. London, UK: IntechOpen; 2012. pp. 171-190

[23] Volosenku C. Fault detection and diagnosis. In: Volosenku C, editor. Fuzzy logic. London, UK: IntechOpen; 2016

[24] Volosenku C, editor. Control Theory in Engineering. London, UK: IntechOpen; 2020. pp. 1-374. DOI: 105772/IntechOpen,83289

[25] Travé-Massuyès L. Bridging control and artificial intelligence theories for diagnosis: A survey. Engineering Applications of Artificial Intelligence. 2013;**27**:1. DOI: 10.1016/j. engappai.2013.09.01,hal-01131188

Chapter 5

A Study on Tuning Methods of Fuzzy PI Controllers

Constantin Voloșencu

Abstract

The question addressed is placed in the broad context of fuzzy control, based on fuzzy PI controller, of industrial processes, the equivalence of fuzzy PI controllers with linear PI controllers, tuned with given methods, for the improvement of performance of control systems. The purpose of the study is to analyze the possibility of these implementation and to find the behavior of the resulted fuzzy control systems to compare with the linear control systems. The tuning criteria chosen are the modulus criterion in Kessler's variant for second-order processes with two time constants, the symmetric criterion in Kessler's variant, for the second order processes with purely integrator character and Ziegler-Nichols criterion for slow processes. The equivalence of the fuzzy controllers with the linear controllers was made using the input-output transfer characteristics of the fuzzy block and its linearization in the origin to obtain the gain in the origin. The results consist of an equivalence formula for the controller, simulation characteristics for the control systems, comparisons and analyses. The main conclusions are that fuzzy regulators can be designed by equating linear regulators with the presented method and better regulation performances can be obtained.

Keywords: fuzzy PI controller, Kessler criteria, Ziegler-Nichols criterion, modeling, simulation

1. Introduction

The proportional-integrator controller is the most commonly known control solution used in practice. It adjusts the process variables by modifying the control system variables based on the error between the reference variables and the measured variable. It assures a behavior of the process on an optimal state trajectory characterized by a short rise time and a zero error. Regulators are implemented numerically in computing equipment with microprocessors by discretizing the regulators in continuous time. The discretization is done with a discretization step – a sampling period – chosen appropriately. PI regulators are used for processes that do not have a purely integrative component. The design of PI controllers is done through a suitable choice of their parameters, with methods that take into account the characteristics of the driven process and the desired performances for the regulation system. The methods of tuning the regulators have been developed for decades; in the specialty literature, there are numerous works on this topic. Scientists continue to research in the field, always bringing improvements to these methods. Some works in the field can be highlighted. The basic book [1] presents process modeling notions, such as dynamic and static modeling, frequency response, PID regulation notions and their numerical implementation, design of regulators with Ziegler-Nichols methods, disturbance rejection and others. PI regulators are still in use today in industry, in many fields, being considered conventional regulators, being characterized by simplicity and having satisfactory performances for a wide range of processes, being standard regulators. The modulus (magnitude) and symmetrical optimum criteria in Kessler's variants are used to design regulators in practice, such as current and speed regulators for electric motors [2]. The paper [3] presents a study of several types of Ziegler-Nichols algorithms for tuning PI controllers. The use of fuzzy PI controllers is done by using pseudo-equivalences of these controllers with linear PI controllers. The use of the term pseudo is due to the fact that fuzzy controllers are nonlinear, and the calculation of their coefficients using values obtained through tuning methods specific to linear controllers is only indicative. In practice, additional tuning through simulations is necessary. Several works that address this issue can be listed several works that address this issue can be listed [4–7]. The paper [8] presents basic properties of fuzzy systems useful in the design of fuzzy controllers, such as description of fuzzy blocks by input-output characteristic, variable nonlinear amplification and linearization around the origin. The paper [9] presents examples of equating fuzzy PID controllers with linear PID controllers using the properties highlighted above. The papers [10, 11] present examples of applications of fuzzy PI controller and analyses of their behavior in different systems and processes.

This chapter appeals as preliminaries to conventional, standard methods for tuning PI regulators, both for fast processes and for slow processes, for processes with or without a purely integrative character. We remember the results that these methods can give in relation to the performance indicators of the regulation systems by modeling and simulating some basic systems. The tuning methods of the fuzzy controllers from the works [8, 9] are used, and modeling and simulations are done with the PI fuzzy controllers obtained by equalization on the same basic processes. The results obtained with linear regulation systems can be compared with the results obtained with fuzzy regulation systems.

2. Tuning methods for linear PI controllers

2.1 The modulus criterion in Kessler's version

The block diagram of a simplified control system is d in **Figure 1**.

The regulated system is equated with the transfer function:

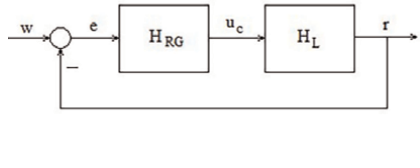

Figure 1.
The simplified block diagram of a control system.

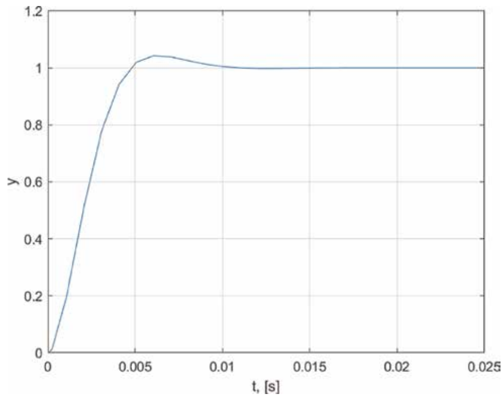

Figure 2.
The response of the linear system tuned with the modulus criterion in Kessler's variant.

$$H_L(s) = \frac{K_L}{(1 + sT_1)(1 + sT_S)} \tag{1}$$

The regulator parameters are calculated with the formulas:

$$H_R(s) = K_R\left(1 + \frac{1}{sT_i}\right), T_i = T_1, K_R = \frac{T_1}{2K_L T_S} \tag{2}$$

Experimental values are: $K_L = 1, T_1 = 0.01$ s, $T_S = 0.001$ s, $K_R = 5, T_i = 0.01$ s.

The values of the performance criteria that are obtained are: $\sigma_1\% = 4.3\%$; $t_1 = 4.7$. $T_S = 0.0047$ s, $t_r = 8{,}4\ T_S = 0.0084$ s.

The response at the unity step input is presented in **Figure 2**.

2.2 The symmetric criterion in Kessler's variant

The regulated system is equated with the transfer function:

$$H_L(s) = \frac{K_L}{s(1 + sT_S)} \tag{3}$$

The regulator parameters are calculated with the formulas:

$$H_{RG}(s) = K_R\left(1 + \frac{1}{sT_i}\right), T_i = 4T_S, K_R = \frac{1}{2K_L T_S} \tag{4}$$

The value of the overshoot that is obtained is $\sigma_1\% = 43\%$. To reduce this value at 8%, a smoothing element placed at the system input is used:

$$H_N(s) = \frac{1}{1 + sT_N} \tag{5}$$

where $T_N = T_S$.

Experimental values are: $K_L = 1$, $K_R = 500, T_i = 0.004$ s, $T_N = 0.004$ s.

The response at the unity step input is presented in **Figure 3**.

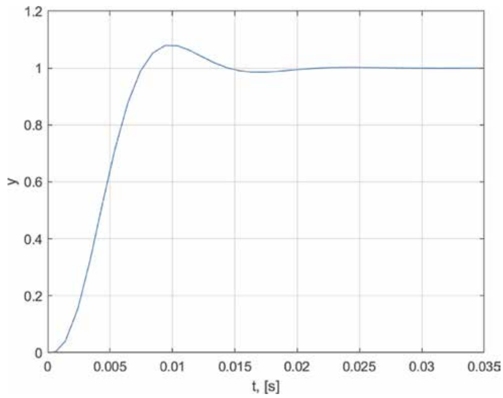

Figure 3.
The response of the linear system tuned with the symmetric criterion in Kessler's variant.

2.3 Slow processes

The regulated system is equated with the transfer function:

$$H_L(s) = \frac{K_L}{1 + sT_e} e^{sT_{me}} \tag{6}$$

Where

$$\rho = \frac{T_{me}}{T_e} \tag{7}$$

According to Ziegler-Nichols criterion, the regulator parameters are calculated with the formulas:

$$K_R K_L \rho = 0,9; \frac{T_i}{T_{me}} = 3,3 \tag{8}$$

Experimental values are: $K_L = 1, T_{me} = 10$ s, $T_e = 0.1$ s, $\rho = 100$, $K_R = 0.009, T_i = 33$ s. A dedicated design can be applied to optimize the regulator, to obtain better regulation efficiency criteria, as a compromise.

The response at the unity step input for an improved design case is presented in **Figure 4**.

2.4 Simulations

The general structure of the regulation system includes the regulator and the process and it is described by the transfer function:

$$H(s) = \frac{y(s)}{u(s)} = \frac{H_R(s)H_L(s)}{1 + H_R(s)H_L(s)} \tag{9}$$

The behaviors of the output $y(t)$ of the systems in the time domain are analyzed based on the response to the unitary step signal applied to the entrance $u(t) = \sigma(t) = 1$.

The answers of the analyzed systems are presented as follows.

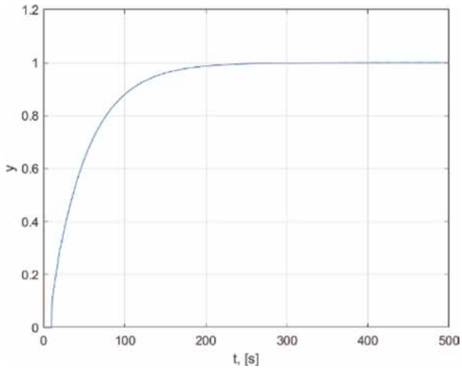

Figure 4.
The response of the linear system tuned with Ziegler – Nichols criterion.

3. Tuning the fuzzy PI controller

3.1 The structure of the fuzzy PI controller

The basic structure of the fuzzy PI controller with integration at the output FC-PI-OI, as presented in [8–10], is shown in **Figure 5**.

The variables are: w – reference variable, r – measured, feedback variable, e – the regulation error, de – the derivative of the error, x_{e} and x_{de} – crisp input variables, u_{d} – defuzzified output variable and u – the output variable of the controller which is the command input for the process.

The fuzzy block has the well-defined Mamdani structure, with fuzzification, with three fuzzy values, N, Z and P, for each variable, with trapezoid, triangle and trapezoid membership functions, rule base with nine rules, inference with the max-min method and defuzzification with the center of gravity method. This structure brings a powerful nonlinear behavior of the fuzzy controller. The fuzzy block has no inertia in time. A PI behavior may be imposed as the dynamic of the controller by an adequate choice of the input derivative filter and the output integrator filter.

Limitations of the input variables are introduced because, in general cases, the numerical calculus of the inference is done only on the scaled universe of discourse $[-1, 1]$.

In particular, the control system must be stable and ensure good amortization.

The filter from the controller input, placed on the low channel, realizes the operation of digital derivation at its output; we obtain the derivative *de* of the error *e*:

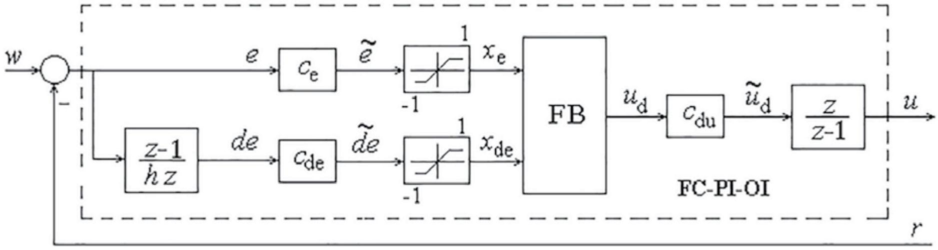

Figure 5.
The block diagram of the fuzzy PI controller.

$$de(t) = \frac{d}{dt}e(t) \circ - \bullet \; de(z) = \frac{z-1}{hz}e(z) \tag{10}$$

The error e and its derivative de are scaled with two scaling coefficients c_e and c_{de}, as it follows:

$$\tilde{e}(t) = c_e e(t), \tilde{de}(t) = c_{de} de(t) \tag{11}$$

In the case of the PI fuzzy controller with integration at the output, the scaled variable $\tilde{u}_d$ is the derivative of the output variable u of the controller. The output variable is obtained at the output of the second filter, which has an integrator character, and it is placed at the output of the controller:

$$u(t) = \int_0^t \tilde{u}_d(\tau)d\tau \circ - \bullet u(z) = \frac{z}{z-1}\tilde{u}_d(z) \tag{12}$$

The defuzzified value of the output variable u_d is scaled with an output scaling coefficient c_{du}.

A composite variable x_t is calculated:

$$x_t = \tilde{e} + \tilde{de} \tag{13}$$

The fuzzy block BF may be describe using its input-output transfer characteristics, its variable gain and its gain in origin as a linear function around the origin [8, 9]:

$$\tilde{e} = 0, \tilde{de} = 0, u_d = 0 \tag{14}$$

An example of the u_d characteristic as a function of x_t is given in **Figure 6a**, from which the characteristic of the nonlinear amplification factor K_{BB} of the fuzzy block in **Figure 6b** can be extracted.

The gain K_{FB0} in origin is

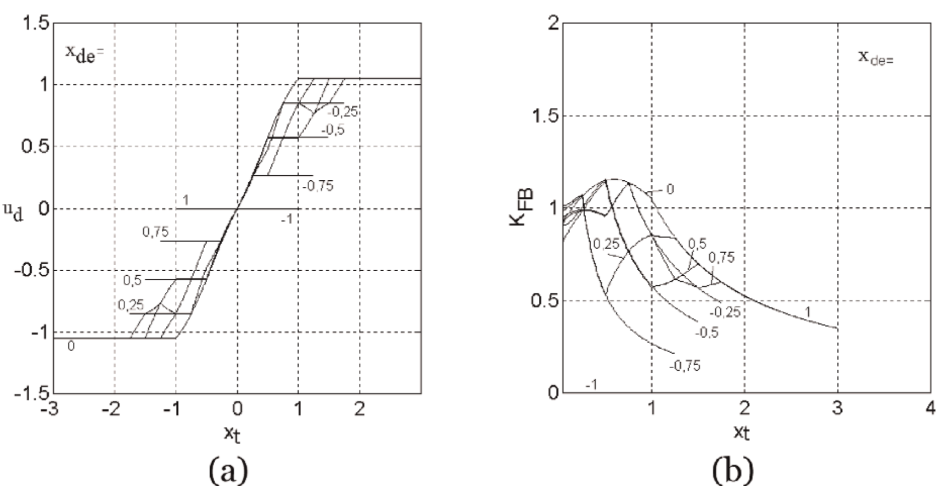

Figure 6.
(a) $u_d = f(x_t; de)$, (b) $K_{FB} = f(x_t; de)$.

$$K_{BF0} = \lim_{x_t \to 0} K_{FB}(x_t) \tag{15}$$

In the case of the linearization of the fuzzy block BF in the origin, the output variable ud is calculated with the relations:

$$u_d = K_{BF0}x_t, , , \tilde{u}_d = c_{du}K_{BF0}x_t\tilde{u}_d = \tilde{c}_{du}x_t\tilde{c}_{du} = c_{du}K_{BF0} \tag{16}$$

3.2 Pseudo-equivalence of the fuzzy PI controller with the linear PI controller

Next, a pseudo-equivalence of the fuzzy PI regulator with the linear PI regulator is presented.

For the fuzzy controller FC-PI-OI with the fuzzy block BF linearized around the origin we may write the following input-output relation in the z-domain:

$$H_{RF}(z) = \frac{u(z)}{e(z)} = \frac{z}{z-1}\tilde{c}_{du}\left(c_e + c_{de}\frac{z-1}{hz}\right) \tag{17}$$

with

$$z = \frac{1 + sh/2}{1 - sh/2} \tag{18}$$

were h is the sampling period.

The equivalence of the fuzzy PI controller with a linear PI controller is a false one, because the fuzzy controller is not linear, so we use the word "pseudo."

The transfer function in s is:

$$H_{RF}(s) = \frac{u(s)}{e(s)} = H_{RF}(z)\Big|_{z=\frac{1+sh/2}{1-sh/2}} = \frac{\tilde{c}_{du}}{h}\left(c_{de} + \frac{h}{2}c_e\right)\left[1 + \frac{c_e}{(c_{de} + c_e h/2)s}\right] \tag{19}$$

From the identification of the coefficients of the function (19) with the coefficients of the transfer function of the PI linear controller, the following relations result:

$$K_{RG} = \frac{\tilde{c}_{du}}{h}\left(c_{de} + \frac{h}{2}c_e\right), T_{RG} = \frac{c_{de} + \frac{h}{2}c_e}{c_e} \tag{20}$$

At the limit, for $h \to 0$, the gain and time coefficients of the fuzzy controller have the expressions:

$$K_{RG} = c_{de}K_{BF0}c_{du}/h, T_{RG} = c_{de}/c_e \tag{21}$$

The relations for designing the scaling coefficients based on the parameters of the linear PI controller are:

$$c_e = \frac{hK_{RG}}{c_{du}K_0T_{RG}}, c_{de} = c_e(T_{RG} - h/2) \tag{22}$$

The coefficient c_{du} and c_{de} may be chosen starting with the value 1 and then modified so that the regulation system is stable and has good values of the performance indicators of the regulation.

The coefficient $K_0 \approx 1$ is taken as the value in origin from **Figure 6b**.

The linear PI controller may be designed with the methods from Section 2, or other methods from literature.

3.3 Calculation examples

3.3.1 Equivalence in the first case, the modulus criterion in Kessler's version

The values of the parameters of the linear regulator are those calculated with eq. (2).

The values of the parameters of the fuzzy controller are calculated with equation (22). Thus, the regulator parameters have the initial values h = 0.001 s, c_e = 0.5, c_{de} = 0.00475, c_{du} = 1, and they are adjusted through repeated attempts so as to obtain good values of the performance indicators of the regulation.

3.3.2 Equivalence in the second case, the symmetric criterion in Kessler's variant

The values of the parameters of the linear regulator are those calculated with eq. (4).

The values of the parameters of the fuzzy controller calculated with equation (22) are: c_e = 125, c_{de} = 0.4375.

3.3.3 Equivalence in the third case, the slow processes

The values of the parameters of the linear regulator are those calculated with eq. (8).

The values of the parameters of the fuzzy controller calculated with equation (22) are: c_e = 2,72.10^{-7} and c_{de} = 9.10^{-6}.

4. Results of the simulation of fuzzy control systems

For simulations, the values of the scaling coefficients of the resulting fuzzy PI controllers are calculated for each of the four controllers in Section 2, using the above

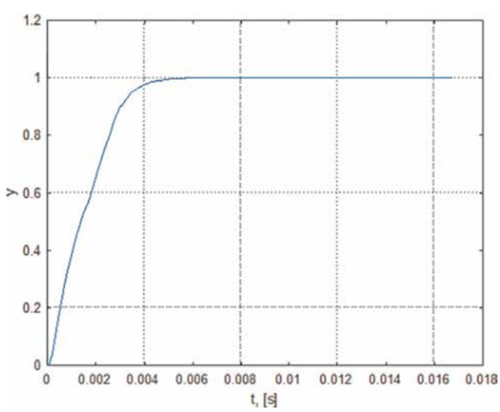

Figure 7.
The response of the linear system tuned with the modulus criterion in Kessler's version.

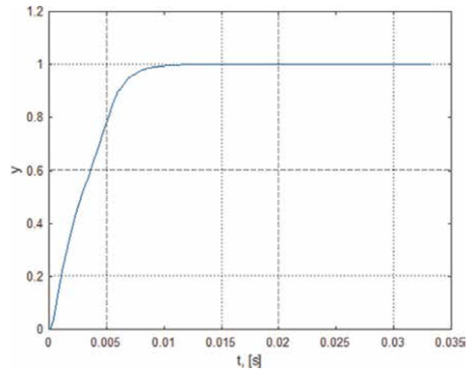

Figure 8.
The response of the linear system tuned with the symmetric criterion in Kessler's version.

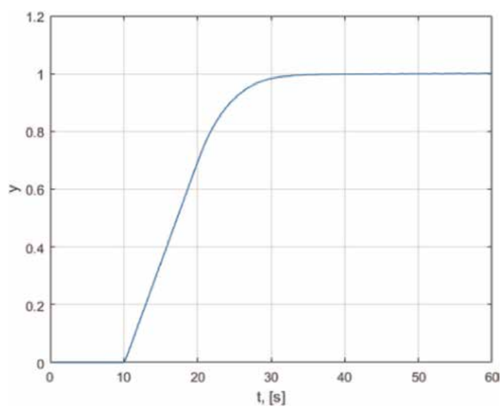

Figure 9.
The response of the linear system tuned with Ziegler-Nichols criterion.

formulas. Then, the fuzzy controllers are used to command the above processes, and time analyses are done, calculating the responses of the systems to the same step signals from above, applied to the inputs.

The simulations were done using MATLAB, Simulink and fuzzy toolbox. The responses of the four systems with fuzzy regulators are presented in **Figures 7–9**.

In cases of tuning with Kessler's criteria, the responses of the systems are aperiodically damped, with zero overshoot and with approximately the same rising time. In the case of tuning with the Ziegler-Nichols criterion, the fuzzy tuning ensures also a damped aperiodic response with zero overshoot and with a shorter response time.

5. Conclusion

The paper presents a study on tuning fuzzy PI controllers with integration at the output based on equivalence with linear PI controllers designed with three design methods, two specific to fast processes, namely Kessler's criteria and one specific to slow processes, namely the criterion of Ziegler-Nichols. The equivalence method of

the fuzzy PI controller is presented, using the input-output transfer characteristics of the nonlinear fuzzy block and its linearization around the origin. The calculation equations of the coefficients of the fuzzy PI controllers based on the coefficients of the linear PI controllers are presented. Analyzes were performed in the time domain through simulations of linear and fuzzy control systems. The step signal responses of the developed regulation systems were compared. Fuzzy regulators improve, in a certain way, the values of the performance indicators of the regulation systems for the prescription input signal.

Author details

Constantin Voloșencu
Politehnica University, Timisoara, Romania

*Address all correspondence to: constantin.volosencu@aut.upt.ro

References

[1] Astrom K, Hdgglund T. PID Controllers. 2nd ed. Research Triangle Park: Instrument Society of America; 1995. 338p

[2] Ufnalski B. DSFOC Drive and Kessler's Criteria. 2024. Available from: https://www.mathworks.com/matlabcentral/fileexchange/52706-dsfoc-drive-and-kessler-s-criteria [Accessed: July 27, 2024]

[3] Bhuyan S. A Matlab Approach to Study Different Types of Ziegler-Nichols PID Controller Tuning Algorithm. 2024. Available from: https://www.mathworks.com/matlabcentral/fileexchange/55908-a-matlab-approach-to-study-different-types-of-ziegler-nichols-p-i-d-controller-tuning-algorithm [Accessed: July 27, 2024]

[4] Buhler H. Reglage par logique floue. Lausanne: Press Polytechniques et Universitaires Romands; 1994. p. 20

[5] Bouallegue S, Haggege J, Benrejeb M. A new method for tuning PID-type fuzzy controllers using particle swarm optimization. In: Iqbal S, editor. Fuzzy Controllers-Recent Advances in Theory and Applications. London: Intech; 2012. DOI: 10.5772/48305

[6] Adel T, Abdelkader C. PID Control for Takagi-Sugeno Fuzzy Model. In: Shamsuzzoha M, editor. PID Control for Industrial Processes. London: InTech; 2018. DOI: 10.5772/intechopen.74295

[7] Sinthipsomboon K, Hunsacharoonroj I, Khedari J, Po-ngaen W, Pratumsuw P. A hybrid of fuzzy and fuzzy self-tuning PID controller for servo electro-hydraulic system. In: Iqbal S, editor. Fuzzy Controllers-Recent Advances in Theory and Applications. London: InTech; 2012. DOI: 10.5772/48614

[8] Volosencu C. Properties of fuzzy systems. WSEAS Transactions on Systems. 2009;**2**(8):210-228

[9] Volosencu C. Pseudo-equivalence of fuzzy PID controllers. WSEAS Transactions on Systems and Control. 2009;**4**(4):163-176

[10] Volosencu C. Stabilization of fuzzy control systems. WSEAS Transactions on Systems and Control. 2008;**10**(3):879-896

[11] Volosencu C. Identification of distributed parameter systems, based on sensor networks and artificial intelligence. WSEAS Transactions on Systems. 2008;**6**(7):785-801

Edited by Constantin Voloşencu

The subjects in the book *PID Control - New Design Methods and Applications* chapters range from fundamental aspects of PID (Proportional–Integral–Derivative) controller design theory to industrial applications and complex process control systems. The book covers topics such as basic considerations for the digital implementation of PID Controllers, tuning methods of fuzzy PI controllers, analytical design of a closed control loop controller, identification and control of unstable systems using PITOPS (Process Identification and Controller Tuning Optimizer Simulator), and the design and development of servo drive control system based on DSP (Digital Signal Processor) The book highlights several advantages, including the efficiency of PID (Proportional–Integral–Derivative) controllers, which is demonstrated both theoretically and practically, showcasing their fast and stable response. It also emphasizes their ability to reduce errors and improve the performance of control systems, as well as their simplicity, ease of tuning, and the practical methods presented to enhance PID controllers. The book is intended for a broad audience, including academics and industrial specialists such as professors, researchers, designers, and students.

Published in London, UK

IntechOpen

ISBN 978-0-85466-477-1