mathematics

Article

Experimental Validation of Fractional PID Controllers Applied to a Two-Tank System

Felipe de J. Sorcia-Vázquez [1], Jesse Y. Rumbo-Morales [1], Jorge A. Brizuela-Mendoza [2], Gerardo Ortiz-Torres [1], Estela Sarmiento-Bustos [3,*], Alan F. Pérez-Vidal [1], Erasmo M. Rentería-Vargas [1], Miguel De-la-Torre [1] and René Osorio-Sánchez [1]

[1] Computer Science and Engineering Department, University of Guadalajara, Ameca 46600, Mexico; felipe.sorcia@academicos.udg.mx (F.d.J.S.-V.); jesse.rumbo@academicos.udg.mx (J.Y.R.-M.); gerardo.ortiztorres@academicos.udg.mx (G.O.-T.); alan.perez@academicos.udg.mx (A.F.P.-V.); erasmo.renteria@academicos.udg.mx (E.M.R.-V.); miguel.dgomora@academicos.udg.mx (M.D.-l.-T.); rene.osorio@academicos.udg.mx (R.O.-S.)

[2] Exact Sciences and Methodologies Department, University of Guadalajara, Ciudad Guzmán 49000, Mexico; jorge.brizuela@cusur.udg.mx

[3] Academic Division of Industrial Mechanics, Emiliano Zapata Technological University of the State of Morelos, Emiliano Zapata 62760, Mexico

* Correspondence: estelasarmiento@utez.edu.mx

Abstract: An experimental validation of fractional-order PID (FOPID) controllers, which were applied to a two coupled tanks system, is presented in this article. Two FOPID controllers, a continuous FOPID (cFOPID) and a discrete FOPID (dFOPID), were implemented in real-time. The gains tuning process was accomplished by applying genetic algorithms while considering the cost function with respect to the tracking error and control effort. The gains optimization process was performed directly to the two-tanks non-linear model. The real-time implementation used a National Instruments PCIe-6321 card as a data acquisition system; for the interface, we used a Simulink Matlab and Simulink Desktop Real-Time Toolbox. The performance of the fractional controllers was compared with the performance of classical PID controllers.

Keywords: fractional PID control; coupled tanks system; real-time control

MSC: 93-05

Citation: Sorcia-Vázquez, F.d.J.; Rumbo-Morales, J.Y.; Brizuela-Mendoza, J.A.; Ortiz-Torres, G.; Sarmiento-Bustos, E.; Pérez-Vidal, A.F.; Rentería-Vargas, E.M.; De-la-Torre, M.; Osorio-Sánchez, R. Experimental Validation of Fractional PID Controllers Applied to a Two-Tank System. *Mathematics* **2023**, *11*, 2651. https://doi.org/10.3390/math11122651

Academic Editors: Ying Luo, Fudong Ge and Jorge Muñoz

Received: 19 April 2023
Revised: 30 May 2023
Accepted: 7 June 2023
Published: 10 June 2023

1. Introduction

The research community has deeply studied the classical PID controller. These controllers have been applied to industrial processes, electromechanical systems, electric and electronic systems, and aeronautical and aerospace systems. The PID controller's success is focused on its simplicity and robustness. One important characteristic of this controller is that the integral term offers robustness against disturbances and parameter uncertainty. Several tuning methods are applied to these controllers, e.g., root locus, frequency, or graphical methods, which are relatively easy to implement. These facts make the controller attractive for implementing the regulation of almost any system or process. A weak characteristic of PID controllers is related to the derivative term. This term is sensitive to the noise originating that the resulting control signal is noisy. This fact negatively affects the performance of the closed-loop system. Different proposals of PID structures have been developed in place of classical PIDs, such as fuzzy PID controllers [1,2], adaptive PID controllers [3,4], sliding mode PID controllers [5,6], intelligent PID controllers [7,8], and generalized PI and PID controllers [9–11]. All of these developments try to improve the performance of classical PIDs and give better characteristics that classical PIDs lack.

Another variation of the classical PID controller is the fractional-order PID controller (FOPID). This variation is based on fractional calculus. During the last decades, fractional calculus has been the focus of automatic control researchers. Fractional calculus has been applied to modeling systems that have fractional-order differential equations. At first, models were developed based on fractional order equations for mechanical [12,13] and electrical systems [14,15]. Later, the first continuous FOPID (cFOPID) controller was introduced by Podlubny in 1999 [16]. This controller has two additional parameters compared with classical PIDs. These parameters are the constants of the fractional-order integral and fractional-order derivative. This new characteristic added to the classical PID generates a more flexible controller and provides the opportunity to better adjust the dynamical properties of the closed-loop system [16]. After this first development, the use of the FOPID controller to regulate different types of systems increased. Examples of the applications in engineering systems that are regulated by FOPIDs are: smart reactors [17], electronic power converters [18–20], rehabilitation systems [21]; automatic voltage regulators [22,23], industrial process models [24,25], robotic systems [26], power systems using synchronous generators [27], and wind turbines [28].

The gains-tuning process is essential in the implementation of FOPIDs for regulating dynamic systems, which is a challenging aspect. This process can be manually facilitated, but the performance of the closed-loop system could be better. Different methods for tuning FOPID controllers have been developed to solve this problem, e.g., rule-based, analytical, and numerical methods [29]. We focus on numerical methods, specifically evolutionary algorithms. An evolutionary algorithm is an optimization algorithm that is inspired by a natural, biological, or physical phenomenon. Among the evolutionary algorithms, genetic algorithms (GAs) and their improvements have been an attractive solution used to tune fractional-order PID controllers. In order to start the optimization process, it is necessary to establish a suitable objective function. Many popular objective functions have been defined and used for tracking error minimizations, such as the integral squared error (ISE), integral of the time-weighted squared error (ITSE), integral absolute error (IAE), and integral time-weighted absolute error (ITAE) indices [27], with the aim of obtaining the minimum value of these functions. In [30], the authors used a GA to tune the fractional factors of a cFOPID while considering the ISE index as the objective function and setting the fractional order bound as ranging from 0 to 100. A third-order continuous transfer function was utilized as a controlled system. The results showed that the cFOPID controller can regulate the system, even if the fractional factor values exceed the general bound for these two parameters [29]. In [31], the authors implemented a simulation of a cFOPID controller with a three-DOF robot system that is driven by DC motors. In this work, the objective function considered the overshoot, settling time, and IAE index. The optimization was performed to find all FOPID parameters, and the simulation results show that GAs can find optimal values for all FOPID parameters. In [32], the authors discussed the control of a CD motor fed by a buck–boost converter using a cFOPID, and its performance was compared with the classical PID controller; in this work, the controller was tuned by applying a GA, and the tuning process was conducted while considering three different cost functions: the ISE, ITSE, and mean square error (MSE). The results obtained show that, for this study case, the ISE function achieved the best performance for the cFOPID. In [33], the authors addressed the tuning of a cFOPID device using GAs and particle swarm optimization for a two-DOF robot trajectory control. The authors used three different cost functions during the tuning process: the root mean squared error (MRSE), mean absolute error (MAE), and mean minimum fuel and absolute error (MMFAE). Each cost function is used individually to obtain the optimal values of both controllers. The results show that the MRSE cost function provides the best controller parameters with the lowest fitness values. In [34], the authors addressed the application of a multi-objective genetic algorithm (MOGA) fractional-order PID controller for semi-active magnetorheological damped seat suspension. The authors selected the gain crossover frequency and phase margin for the optimization problem of the FOPID controller as objective functions. The simulations performed show that cFOPID offers superior performance over the integer controllers. An improvement of GAs called multi-objective evolutionary non-dominated sorting genetic algorithm II (NSGAII), which is used to tune a cFOPID, was applied to a hydraulic turbine and addressed in [35]. The performance of the cFOPID was compared with the performance of a PID. The authors considered two objective functions for optimizing the controller parameters in the system: the ISE as the cost function one and ITSE as cost function two. The results show that the NSGAII algorithm is an efficient algorithm for tuning a cFOPID. In [36], the authors implemented a novel optimization algorithm called cloud-model-based quantum genetic algorithm (CQGA) to tune a cFOPID to control the motion of an autonomous underwater vehicle (AUV) in real time, and the authors considered the linear transfer functions of the AUV dynamics. CQGA is a fusion of cloud-model theory and the quantum genetic algorithm (QGA). The QGA algorithm is based on the principles and concepts of quantum computing. The authors adopted the ITAE as the objective function. The results show that the cFOPID performs the best for the heading control, diving control, and path-following systems compared with a classical PID. The authors established the limits of the fractional order factors as $\lambda, \mu \in (0, 1)$, but their optimization result shows values of λ and μ out of this interval. The authors in [37] constructed the design of a cFOPID device tuned using GAs for a conical tank and compared its performance with a PID device. From the nonlinear model of the tank, the authors analyzed the dynamical response in an open-loop system at different operating points. After this process, a linear transfer function at the height of 40 cm was selected to perform the tuning process of the cFOPID. The ITAE was chosen as the objective function. The cFOPID achieved the best transient response and performance at the operation point of 40 cm compared with the results obtained by applying the PID. In [38], the authors presented the application of a robust queen bee assisted genetic algorithm (QBGA) for tuning a cFOPID to control a boost converter with a non-minimum phase behavior. The authors selected the ISE as the objective function to minimize. The tests performed show that the cFOPID performs better than two PID controllers that were tuned based on the transient response and ISE using the QBGA. The robustness of the controllers of the closed-loop system was tested by applying parameter variations to the model. The results show that the cFOPID has better robustness than the PID controllers.

From a general point of view, the FOPID tuning process using GAs employs objective cost functions depending on the tracking error (e.g., ISE, ITAE, and ITSE) and simplified linear models of the system under control. This motivated our study to explore the behavior of GAs by using a cost function that incorporates the weighted quadratic error and control effort, as well as the nonlinear system model, to perform the optimization process. Considering the control input in the cost function allows us to penalize large control inputs, which improves the dynamic performance [39]. In this research work, we present the study of the performance of cFOPID and dFOPID controllers in real time. The performance of the FOPID controllers was compared with the performance of continuous and discrete classical PID controllers, which were also tuned using GAs. The work contribution is summarized as follows:

- The tuning process of the FOPID using GAs that considers an objective function including the weighted quadratic error and control effort;
- An offline simulation for the optimization process that uses the nonlinear model of the system for control;
- The experimental results demonstrate that the proposed controllers' optimization results are adequate for testing the FOPIDs tuned using a GA and the cost function defined by Equation (19) in real time.

The remaining sections of the paper are arranged in the following order: Section 2 shows the two-tank system model and includes the experimental validation of the proposed model. Section 3 presents the structures of the fractional and classical PID controllers that were implemented in this work. Section 4 provides the results of the controllers applied in the two-tank system in real time. Additionally, this section provides a discussion of these results. Finally, Section 5 presents the conclusions of this work.

2. Two-Tanks System

2.1. Two-Tank Description

Academic and research studies have frequently used tank systems to test controller structures. The tank system is a modular process that can be easily configured or modified. Coupled tank systems can be found in arrangements of two [40], three [41], four [42], or more tanks; this is an attractive feature of tank systems. The system used to test the FOPID controllers consisted of a two-coupled tank system. Two cylindrical tanks conformed to the system, one over the other. Figure 1 shows the physical setup of the two tanks. The system had the following hardware implemented:

- Two ultrasonic sensors to measure the levels of each tank;
- A direct current pump that supplied water flow to fill the tank system;
- A PCIe-6321 Data Acquisition (DAQ) unit;
- A host PC to implement the controllers in real time.

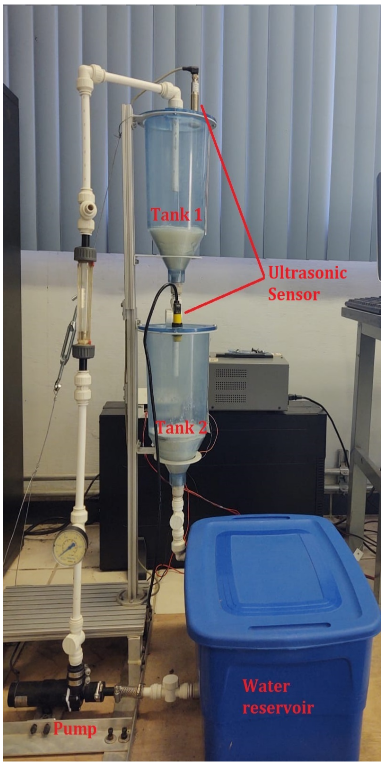

Figure 1. Two-tank system.

The conical bottom section of the two tanks was eliminated using cones created by a 3D printer, which allowed us to measure the tank levels from the top of these conical sections.

2.2. Two-Tank Model System

Figure 2 depicts the two-tank system scheme. The system consists of a water reservoir where a pump draws the water to fill the tanks and two ultrasonic sensors measure levels h_1 and h_2. The pump flow fills tank 1, and tank 1's outlet flow supplies tank 2.

The dynamic model for the two tanks' liquid levels is derived as follows. The time rate of change of the volume in each tank is described by

$$\frac{dV_i(t)}{dt} = F_{i,in}(t) - F_{i,out}(t), \quad i = 1, 2, \tag{1}$$

where $V_i(t)$ is the volume of the tank and $F_{i,in}(t)$ and $F_{i,out}(t)$ are the inlet and outlet flow for the i-th tank, respectively. Equation (1) can be expressed in terms of the liquid level, considering the tank volume as $V_i(t) = A_i h_i(t)$:

$$\frac{dh_i(t)}{dt} = \frac{1}{A_i}(F_{i,in}(t) - F_{i,out}(t)), \quad i = 1,2, \tag{2}$$

where A_i denotes the cross-sectional area of the i-th tank. The inlet flow in tank 1 is defined by

$$F_{1,in} = K_f u(t), \tag{3}$$

where K_f is the pump constant (cm^3/sV) and $u(t)$ is the voltage applied to the pump. The outlet flow in each tank is modeled using Bernoulli's law, which describes the outlet flow velocity of tanks with small orifices. This velocity is defined as

$$v_{i,out} = \sqrt{2gh_i(t)} \quad i = 1,2, \tag{4}$$

Therefore, the outlet flow rate for each tank, considering the cross-sectional area of the outlet orifice, is described by

$$F_{i,out} = a_i\sqrt{2gh_i(t)} \quad i = 1,2, \tag{5}$$

where a_i is the cross-sectional area of the outlet flow orifice in each tank and g is the gravitational acceleration. Finally, from Figure 2, the inlet flow to tank 2 is the outlet flow of tank 1,

$$F_{1,out} = F_{2,in}, \tag{6}$$

From this consideration, the two-tank model system can be derived from Equation (2), (3), (5), and (6) as follows

$$\begin{aligned} \frac{dh_1(t)}{dt} &= -\frac{a_1}{A_1}\sqrt{2gh_1(t)} + \frac{K_f}{A_1}u(t), \\ \frac{dh_2(t)}{dt} &= \frac{a_1}{A_2}\sqrt{2gh_1(t)} - \frac{a_2}{A_2}\sqrt{2gh_2(t)}. \end{aligned} \tag{7}$$

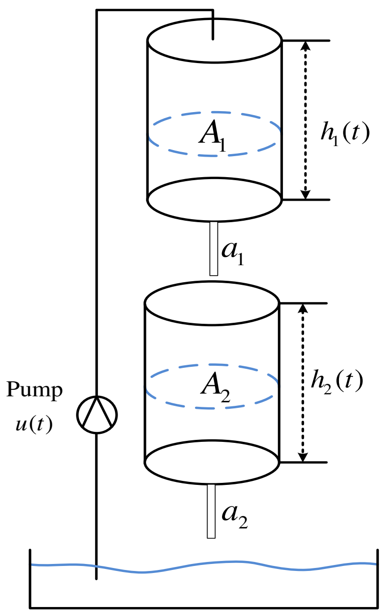

Figure 2. Two-tank system scheme.

2.3. Experimental Validation of the Two-Tank System Model

An essential step in designing controller schemes and implementing them in real time is the model validation of the plant wanting to be controlled. The model validation was performed by simulating Equation (7) and using the parameters shown in Table 1. During the test, a fixed voltage input of $u(t) = 0.5$ was applied to the pump. Under this same condition, the physical tank system was exited to capture its dynamic. It is relevant to establish that the pump's input voltage is normalized in the following interval: $u(t) \in [0 \quad 1]$.

Figure 3 depicts the validation of the dynamics of the tank levels. The bottom graph shows the dynamic of level h_2, and the top graph shows the dynamic of level h_1. It can be seen in this figure that at the beginning of the level evolutions, the model cannot reproduce the dynamics of the tank. This issue originated from a turbulent flow in the tank's outlets when the water began entering each tank. After that, the outlet flow in each tank becomes a laminar flow, and the model can reproduce the tank's dynamics better. With this result, it is considered that the two-tank model, which is represented by Equation (7), can reproduce the dynamics of the physical two-tank system.

Table 1. Two-tank system parameters.

Parameter	Variable	Value	Units
Pump constant	K_f	116.66	cm^3/sV
Area of tank 1 and 2	(A_1, A_2)	630	cm^2
Discharge constant of tank 1	a_1	0.75	cm^2
Discharge constant of tank 2	a_2	0.532	cm^2
Gravitational constant	g	981	cm/s^2
Level of tank 1	h_1	3.08	cm
Level of tank 2	h_2	6.12	cm

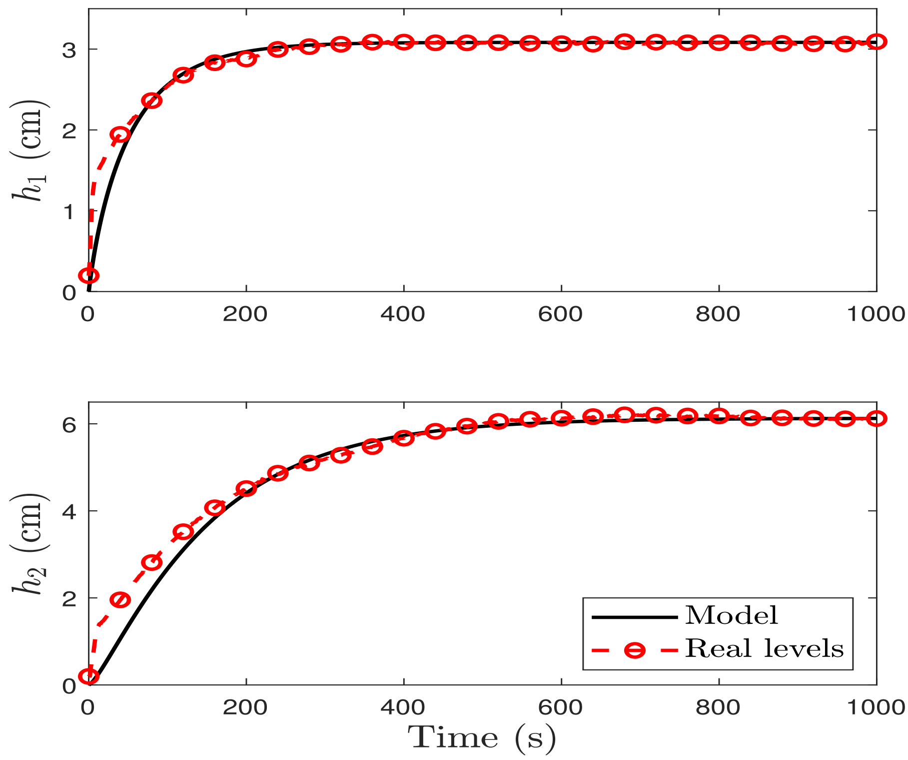

Figure 3. Real-time validation of the two-tank system model.

Table 1 shows the final values reached for each tank.

3. PID Control Structures Applied to the Two-Tank System

3.1. Continuous Fractional PID Controller

Fractional-order Systems are based on the generalized non-integer order fundamental operator

$$
{}_a\mathscr{D}_t^\alpha = \begin{cases} \frac{d^\alpha}{dt^\alpha}, & \alpha > 0, \\ 1, & \alpha = 0. \\ \int_a^t (dt)^{-\alpha}, & \alpha < 0, \end{cases} \tag{8}
$$

where a and t are the limits of the operation. In general, it is assumed that $\alpha \in \mathbb{R}$, but it may also be a complex number. Several definitions of the integro-differential operator have been developed. The application of the cFOPID considers the Grünwald–Letnikov definition because of its applicability to the numerical evaluation of fractional-order derivatives

$$
{}_a\mathscr{D}_t^\alpha f(t) = \lim_{h \to 0} \frac{1}{h^\alpha} \sum_{j=0}^{k} (-1)^j \left(\begin{array}{c} \alpha \\ j \end{array} \right) f(t - jh), \tag{9}
$$

where $a = 0$, $t = kh$, k is the number of computation steps, and h is the step size. Considering zero initial conditions, the Laplace transform of the α-order derivative is given by

$$
\int_0^\infty e^{-st} {}_0\mathscr{D}_t^\alpha f(t) dt = s^\alpha F(s), \tag{10}
$$

in this case, $\alpha \in \mathbb{R}^+$, and s is the usual Laplace variable.

The generalized cFOPID controller, which was proposed by Podlubny in 1999 [16], is called $\mathrm{PI}^\lambda \mathrm{D}^\mu$. This controller has an integrator with order λ and a differentiator with order μ. The cFOPID controller has the following time domain equation

$$
u_c(t) = K_p e(t) + K_i D^{-\lambda} e(t) + K_d D^\mu e(t), \tag{11}
$$

where $D^{-\lambda} e(t)$ and $D^\mu e(t)$ represent the fractional-order integral and fractional-order derivative, respectively. Applying the Laplace transform to Equation (11), the general cFOPID controller transfer function takes the form

$$
\frac{u_c(s)}{e(s)} = K_p + \frac{K_i}{s^\lambda} + K_d s^\mu. \tag{12}
$$

If the fractional order constants are $\lambda = \mu = 1$, the result is the classical PID controller. Figure 4 shows the cFOPID controller plane, where four points mark out the classical PID controller and its variations. The FOPID controller has more degrees of freedom than the classical PID. This characteristic allows for a better adjustment of the dynamical response of the fractional-order control system [16].

3.2. Discrete Fractional PID Controller

For the implementation of the dFOPID controller, the following structure of the cFOPID will be considered:

$$
u_c(t) = \left(K_p + K_i \nabla^{-\lambda}(q) + K_d \nabla^\mu(q) \right) e(t), \tag{13}
$$

where $\nabla^{-\lambda}(q)$ and $\nabla^\mu(q)$ are the backward (nabla) discrete-time fractional-order integrator and difference, q is the forward shift operator, and h is the sample time [43]. The nabla fractional-order difference can be defined by the use of the discrete-time Grünwald–Letnikov operator as

$$\nabla^{\mu}e(t) = h^{-\mu}\sum_{j=0}^{t/h}(-1)^j\binom{\mu}{j}e(t-hj) = \nabla^{\mu}(q)e(t), \tag{14}$$

where $\nabla^{\mu}(q) = h^{-\mu}\sum_{j=0}^{t/h}(-1)^j\binom{\mu}{j}q^{-j}$. The $\mathscr{Z}$-transform of $\nabla^{\mu}(q)$ is defined as

$$\nabla^{\mu}(z) = h^{-\mu}\sum_{j=0}^{\infty}(-1)^j\binom{\mu}{j}z^{-j}.$$

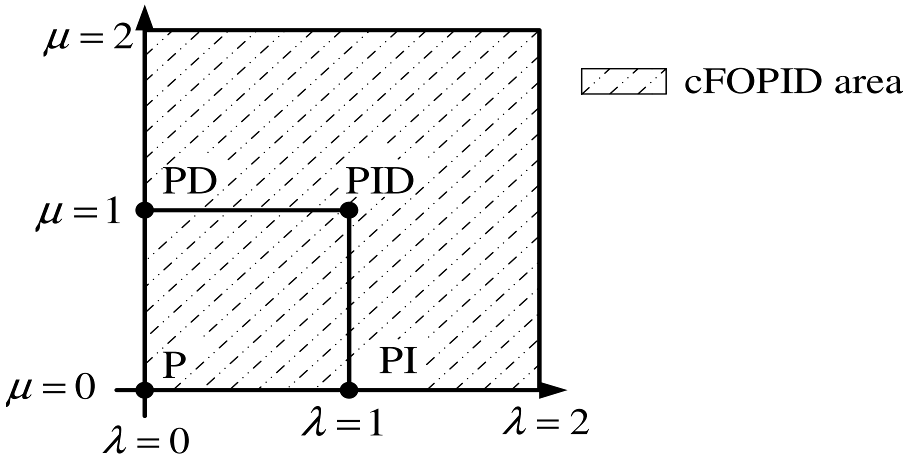

Figure 4. cFPID controller plane.

In practice, $\nabla^{\mu}(z)$ cannot be implemented because of the infinite sum; therefore, in practical applications, their finite-length implementations can be employed in the form of

$$\nabla^{\mu}(z) \approx \nabla^{\mu}_{L}(z) = h^{-\mu}\sum_{j=0}^{L}(-1)^j\binom{\mu}{j}z^{-j}, \tag{15}$$

with L being the finite length of the sum. To calculate the fractional-order integrator $\nabla^{-\lambda}(q)$, μ is substituted by $-\lambda$ in Equation (15). Therefore, the operator $\nabla^{-\lambda}(z)$ can be obtained as follows

$$\nabla^{-\lambda}_{L}(z) = \frac{1}{\nabla^{\lambda}(z)} = \frac{1}{h^{-\lambda}\sum_{j=0}^{L}(-1)^j\binom{\lambda}{j}z^{-j}}. \tag{16}$$

Finally, the discrete fractional-order PID (dFOPID) controller has the form

$$\frac{u_d(z)}{e(z)} = K_p + K_i\nabla^{-\lambda}_{L}(z) + K_d\nabla^{\mu}_{L}(z). \tag{17}$$

3.3. Continuous and Discrete PID Controller

As stated, the continuous PID controller is a particular case of the fractional-order PID controller. The general form of the discrete PID (dPID) controller is obtained by discretizing Equation (12) while considering the fractional-order factor as $\lambda = \mu = 1$. The transfer function in the $\mathscr{Z}$-transform of the dPID has the form

$$\frac{u_d(z)}{e(z)} = K_p + \frac{K_i}{1-z^{-1}} + K_d(1-z^{-1}). \tag{18}$$

It is important to emphasize that the same gain variables and fractional-order factors (K_p, K_i, K_d, μ, λ) are used in the continuous controllers and discrete controllers, but these gains and factors are different in each controller.

3.4. Genetic Algorithm for Tuning Controller Gains

Evolutionary algorithms are an important field of machine learning techniques and are of great interest to researchers. The genetic algorithm (GA) is a population-based stochastic algorithm inspired by natural selection, which is the basis of the Darwinian theory of evolution. The elements of GA are chromosome representation, fitness selection, and biological-inspired operators [44,45]. In the execution of the GA, a random initial population, called a generation, competes at a given task with a defined cost function. Specific rules or genetic operators determine how successful individuals advance to the next generation. After the first generation is populated with individuals, each individual is evaluated and assigned a fitness based on their performance on the cost function [46]. The GA can be applied to the gains-tuning process in almost any control structure. Specifically, GAs can be applied in the tuning process fractional-order PID controllers to obtain the optimal values of the gains K_p, K_i, K_d, integral λ, and derivative μ factors, defining a specific cost function (the reader can refer to [47] for more information about meta-heuristics algorithms for tuning FOPIDs). The cost function considered in this work has the form:

$$J(y(k), u(k), r(k)) = \sum_{k=1}^{k_f} [Q(r(k) - y(k))]^2 + \sum_{k=1}^{k_f} [Ru(k)]^2, \tag{19}$$

where $y(k)$ is the system output, $u(k)$ is the control input, $r(k)$ is the reference, and $Q > 0$ and $R > 0$ are the cost function weights. The optimization problem is established as follows

$$\begin{aligned} x_{opt} &= \arg\min_{x} J(y(k), u(k), r(k)), \\ &\text{subject to} \quad \underline{x} \leq x \leq \overline{x}, \end{aligned} \tag{20}$$

where x is the decision variable representing the controller gains and the fractional-order factors, and $\underline{x}$ and $\overline{x}$ denote the lower and upper bounds of x. Algorithm 1 shows the offline optimization process to obtain the optimal gains for each classical PID controller and FOPID controller. Figure 5 depicts the diagram of the closed-loop system; $h_{2r}(t)$ is the desired level of tank 2. It can be seen in this figure that the data needed to launch the GA and perform the offline optimization is obtained by simulating the closed-loop systems with the optimal values of the current generation before performing an optimization process.

Algorithm 1 (Controllers tuning process)

1: Initialize the gains ($[K_p, K_i, K_d]$ or $[\lambda, \mu]$) equals to zero.
2: *GA*: search for controller gains considering Equation (20).
3: Update the gains with the new value found.
4: Perform the closed-loop simulation with the optimal values of the current generation using Simulink and Equation (7) to update $h_2 \rightarrow y(k)$ and $V \rightarrow u(k)$.
5: **If** the stop criterion reached? Foud the minimum of Equation (19) through evaluate Equation (20), get the values of the optimal gains, and finish the optimization process.
6: **Else**, if Gen $<$ MaxGen and the stop criterion has not been reached, increment the generation index $Gen = Gen + 1$ and return to step 2.

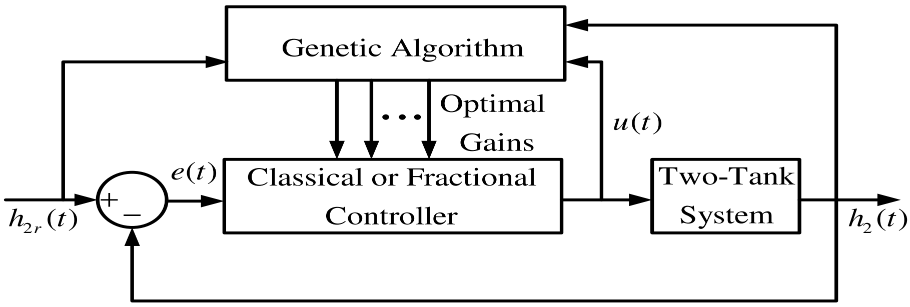

Figure 5. Process for the optimization of the DFOPID parameters.

4. Results and Discussion

4.1. Tuning Results

The tuning process using a GA in the proposed controllers for implementation in real time was performed while following these considerations

- The classical PID controllers were first tuned to obtain the optimal gains K_p, K_i, and K_d for each controller;
- The FOPID controllers used the optimal K_p, K_i, and K_d from the classical PID controllers as fixed gains, and only the fractional-order factors λ and μ were optimized;
- The experiment was set to a population of 10, and the maximum number of generations was defined depending on the best results obtained in the optimization process;
- The experiment considered a simple time of $h = 2$ ms (for the simulations and real time tests).

The optimization process considers bounds for the gains and the fractional-order factor. The bounds are set based on the physical limits of the voltage applied to the pump and on testing the continuous and discrete PIDs in real time using pole placement as the tuning method. These bounds have the form:

- Discrete and continuous classical PID bounds:

$$\begin{gathered} 0 \leq K_p \leq 1, \\ 5 \times 10^{-4} \leq K_i \leq 5 \times 10^{-3}, \\ 0 \leq K_d \leq 3. \end{gathered} \tag{21}$$

- Discrete and continuous FOPID bounds

$$\begin{gathered} 0 \leq \lambda \leq 2, \\ 0 \leq \mu \leq 1.5. \end{gathered} \tag{22}$$

The values of the cost function weights were selected to ensure an acceptable performance in the real-time application. After several tests, the best values found were $Q = 10$ and $R = 0.001$.

Figure 6 presents the obtained results of applying a GA to the continuous classical and fractional-order PID. As can be seen in this figure, the optimization of the cPID controller needed sixty generations to find the optimal gains. On the other hand, the optimization process of the fractional-order factors of the cFOPID needed only twenty generations.

Figure 7 depicts the obtained results of applying a GA to the discrete classical and fractional-order PID. As the figure shows, in this case, the optimization of the dPID controller needed forty generations to find the optimal gains. On the other hand, the optimization process of the fractional-order factors of the dFOPID controller needed only twenty generations. Table 2 concentrates the optimal values for the gains and fractional factors found by the GA for all controllers. In this table, it can be seen that the parameters of the controllers as found by the GA vary in a small proportion. However, these variations are enough to generate different responses in the closed-loop systems.

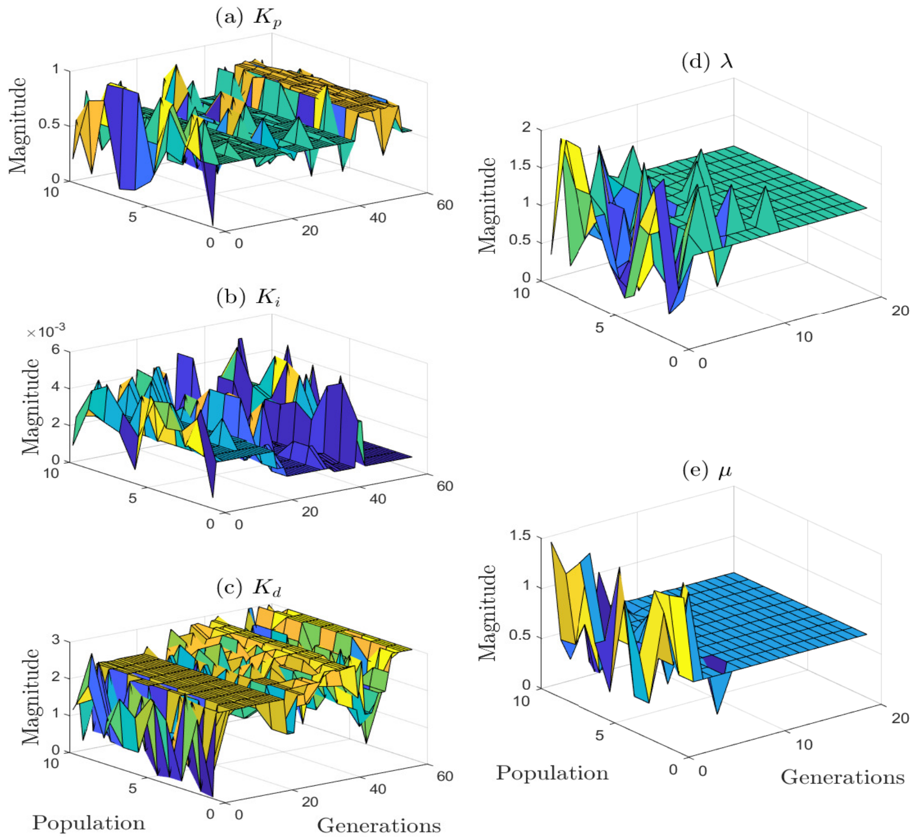

Figure 6. Optimization results of the parameters for the continuous classical and fractional-order PID controllers. (**a**) K_p optimization results. (**b**) K_i optimization results. (**c**) K_d optimization results. (**d**) λ optimization results. (**e**) μ optimization results.

Table 2. Optimal gains and optimal fractional-order factors of the PID controllers and FOPID controllers.

Controller	K_p	K_i	K_d	λ	μ
cPID	0.5214	6.516×10^{-4}	2.99	-	-
cFOPID	0.5214	6.516×10^{-4}	2.99	1.0918	0.6321
dPID	0.5246	6.485×10^{-4}	2.8432	-	-
dFOPID	0.5246	6.485×10^{-4}	2.8432	1.0908	0.6307

Regarding the computational cost in the optimization process, when optimizing the gains of the PID controllers, the optimization time between generations yielded an average of 16.5 s. The computational cost for optimizing the fractional-order factors was 15.5 s between generations. We performed a test that considered both the gains and fractional factors in the optimization, obtaining a computational cost of 18.5 s between generations. In this last test, satisfactory results were not obtained in real time; this is why we optimized the gains of the PID controllers first and then optimized the fractional-order factors. All of these times include the simulation time of the closed-loop system with the optimal controller parameters of the current generation.

4.2. Experimentel Results

The experimental tests were performed while considering the parameters shown in Table 2. The real-time implementation was completed on a computer with the following features:

- CPU: Intel core i5.

- RAM memory: 16 GB;
- Operating system: Windows 10;
- Data acquisition card: National Instruments PICe-6321;
- Matlab 2020b software with the Simulink Desktop Real-Time Toolbox.

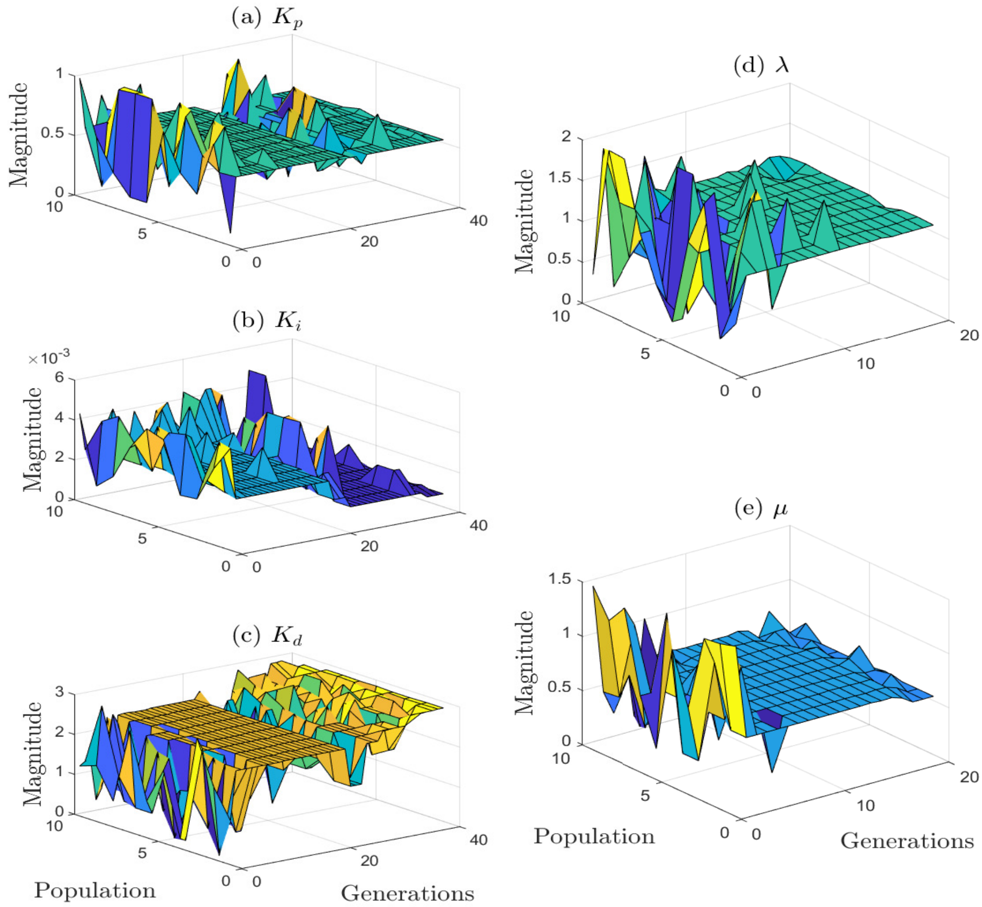

Figure 7. Optimal gains for the discrete classical and fractional-order PID controllers. (**a**) K_p optimization results. (**b**) K_i optimization results. (**c**) K_d optimization results. (**d**) λ optimization results. (**e**) μ optimization results.

The real-time application was performed while considering the steady-state levels reached with the constant signal being applied to the pump. When the two-tank system was at its steady-state point, the controller that was to be tested was turned on. The reference changes considered in the experiments were

$$h_{2r}(t) = \begin{cases} 6.12, & 0 \leq t \leq 500, \\ 7.12, & 500 \leq t \leq 1500, \\ 6.12, & 1500 \leq t \leq 2500, \\ 6.52, & 2500 \leq t \leq 3500. \end{cases} \tag{23}$$

These reference changes consider an increment of 1 cm at the 500 s; at 1500 s, a decrement of 1 cm is applied. Finally, an increment of 0.5 cm is applied at 2500 s.

Figure 8 shows the results obtained after applying the cPID and cFOPID to the two-tank system in real time. It can be observed in this figure that the cFOPID has a better transient response than the cPID (zooms were performed in the figure for better clarity). The tank level reached a value of 7.39 cm with the cPID and 7.21 cm with the cFOPID. With these level values, therefore, the cPID generated an overshoot of 27%, while the cFOPID generated an overshoot of 9%. In the second zoom, it is observed that the cFOPID did not generate an undershoot when the reference changed from 7.2 cm to 6.2 cm. In terms of the settling time, considering an error criteria of 5% and the level change at 500 s, the cPID stabilized the level in 177.63 s, while the cFOPID stabilized the level in 100.9 s. Another relevant fact is with respect to the control signal. The control signal of the cPID was noisier than the control signal of the cFOPID.

Figure 9 depicts the results obtained by implementing the dPID and dFOPID in the two-tank system in real time. As can be observed in this figure, the dFOPID had a better transient response than the dPID, in the same way as in the results obtained with the continuous controllers. In these tests, the tank level reached a value of 7.4 cm with the dPID and 7.24 cm with the dFOPID (this can be observed in the zooms performed in the figure). The dPID generated an overshoot of 28%, while the dFOPID generated an overshot of 12%. When the reference change was performed at 1500 s, it was observed that the dFOPID, as in the result obtained when using the cFPIOD, did not generate undershoot (second zoom in the figure). With respect to the settling time in these tests, considering the same conditions as with the continuous controllers, the dPID stabilized the level in 180.11 s, whereas the dFOPID stabilized the level in 114.55 s. Concerning the control signal obtained in these results, it can be observed that the control signal of the dPID was noisier than the control signal of the dFOPID. Table 3 summarizes the transient response data analyzed for the four controllers.

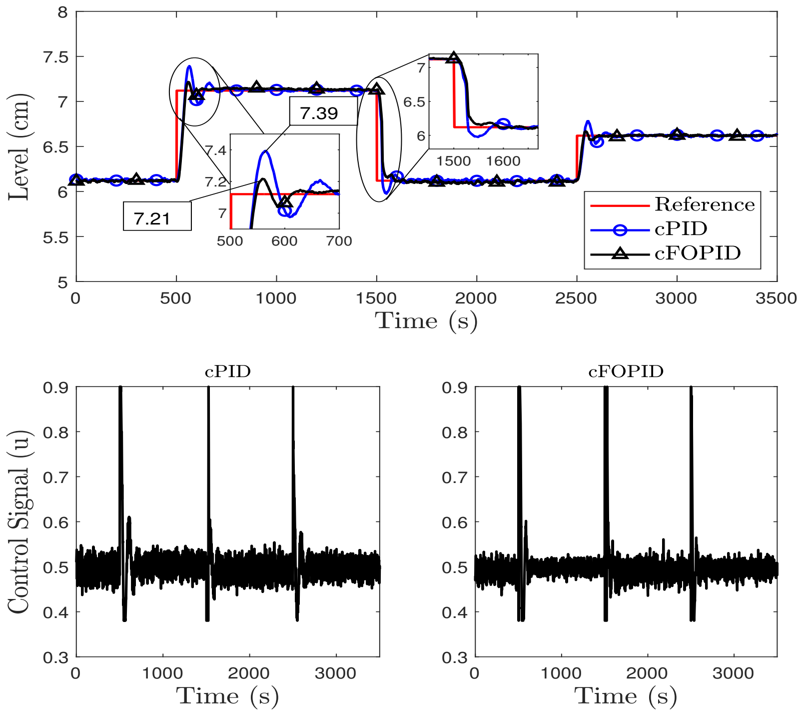

Figure 8. Experimental results obtained by applying the cPID and cFOPID controllers to the two-tank system.

Table 3. Transient response of the classical PID and FOPID controllers.

Controller	Overshoot	Settling Time (Seconds)
Continuous PID	29%	177.63
Continuous FOPID	7%	100.9
Discrete PID	28%	180.11
Discrete FODPID	12%	114.55

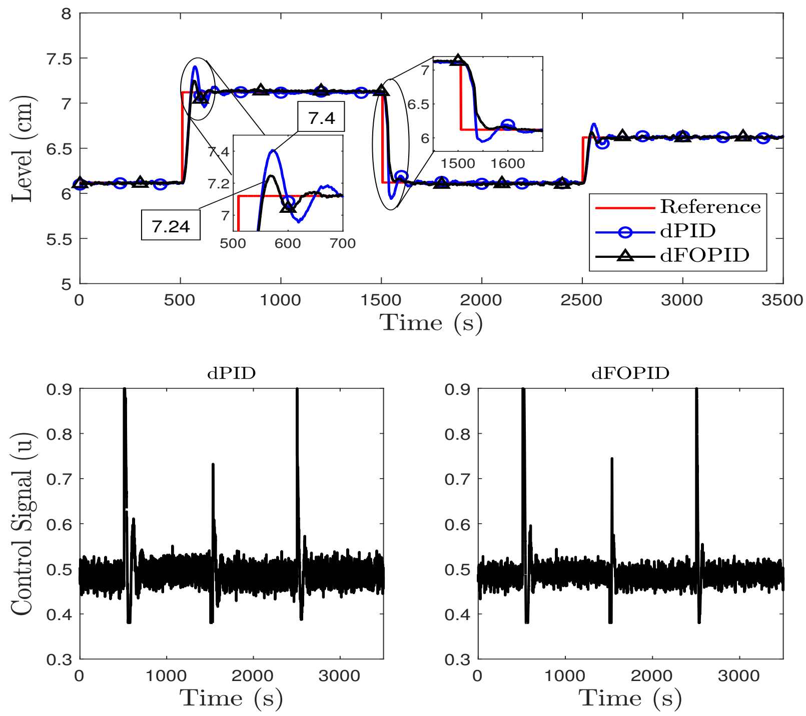

Figure 9. Experimental results obtained by applying the dPID and dFOPID controllers to the two-tank system.

4.3. Discussion

The results obtained by applying the classical PID controllers and fractional-order PID controllers show that the FOPIDs achieved better transient responses than the PIDs. The FOPIDs have more parameters that offer more degrees of freedom in the tuning process, which contribute to them improving the closed-loop dynamic response. In order to quantify the controllers' performance, the integral squared error (ISE), integral absolute error (IAE), and integral time-weighted absolute error (ITAE) indices were evaluated. These performance indices are defined as follows

$$ISE = \int e^2(t), \quad IAE = \int |e(t)|, \quad ITAE = \int t|e(t)|. \tag{24}$$

Table 4 concentrates the performance results obtained after testing all of the controllers. Observing the performance results shown in this table, the cFOPID controller had the best results among all controllers, except in the ISE index; the dFOPID controller had better performance but with a minimal variation in contrast to the ISE obtained by the cFOPID. The dPID controller had the worst performance, generating the highest performance values in all cases.

Table 4. Performance of the classical PID and FOPID controllers.

Controller	ISE	IAE	ITAE
Continuous PID	47.37	106.1	9.465×10^4
Continuous FOPID	47.09	102.8	9.112×10^4
Discrete PID	49.65	108.6	9.915×10^4
Discrete FODPID	47.02	103.1	9.712×10^4

Considering the tuning process results obtained using the GA, the controllers' performances could be different with better results because the GA cannot always reach the global minimum in the optimization processes. Another significant fact is that if no bounds are set in the optimization problem, the GA could expend much computational time to reach the optimal solution with the population and generations setup. It could be considered that the possible values for the decision variables are infinity, as in the case of the gains' values for the controllers tested in this work, where it could be considered that if the gain values obtained by the GA were different, the controllers could have better performance; however, evaluating all possible value combinations is impractical or virtually impossible.

5. Conclusions

The present work addresses the experimental validation of fractional-order PID controllers applied to a two-tank system. The study was limited to optimizing the parameters of the PID and FOPID controllers while considering the cost function described in Equation (19). The performance of the fractional controllers was compared with the performance of classical PID controllers. All of the controllers were tuned using genetic algorithms to obtain optimal values for the controllers' gains and the fractional-order factors. The GAs were applied to each controller considering the nonlinear model of the two-tank system. After testing all controllers in real time, the results show that the FOPID controllers performed better than the classical PIDs. The controllers' parameters obtained with the GA adequately controlled the two-tank system; using an objective cost function that includes the weighted quadratic error and control effort is an acceptable alternative to the typical cost objective functions that are based on the tracking error, which are only optimal for minimizing the error tracking. The FOPIDs reduced the transient response characteristic as the overshoot and settling time. Another interesting result is related to the control signal: the FOPIDs generated less noisy control signals compared with the PIDs.

In conclusion, the FOPID controllers are an attractive alternative to the classical PIDs for real-time control systems. FOPID controllers, having fractional-order factors in their structures, bring more degrees of freedom than classical PIDs. This characteristic presents the opportunity of allowing better adjustment to the controller parameters, improving the closed-loop performance or transient response when the controllers are applied to nonlinear systems; this is a feature that classical PIDs lack. In a future work, we will consider implementing a FOPID controller to regulate ethanol production in a complex dynamical system such as the pressure swinging adsorption process; moreover, we will consider testing the proposed cost function with other optimization algorithms and contrasting the optimization result with the conventional error-based performance indices.

Author Contributions: Conceptualization, F.d.J.S.-V. and J.Y.R.-M.; methodology, J.A.B.-M. and F.d.J.S.-V.; software, G.O.-T. and R.O.-S.; validation, A.F.P.-V. and E.M.R.-V.; formal analysis, F.d.J.S.-V. and G.O.-T.; investigation, J.Y.R.-M. and A.F.P.-V.; resources, E.S.-B. and M.D.-l.-T.; supervision, F.d.J.S.-V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The simulations and experimental data can be shared after a reasonable request to the authors.

Acknowledgments: The authors wish to thank the Centro Universitario de los Valles of the University of Guadalajara for the access and use of the Remote Automation and Computing Laboratory, in which the two-tank System is located.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

PID	Proportional-integral-derivative
FOPID	Fractional-order proportional-integral-derivative
GA	Genetic algorithm
ISE	Integral of the square error
IAE	Integral of the absolute value of the error
ITAE	Integral of the absolute value of the error pondered by the time

References

1. Carvajal, J.; Chen, G.; Ogmen, H. Fuzzy PID controller: Design, performance evaluation, and stability analysis. *Inf. Sci.* **2000**, *123*, 249–270. [CrossRef]
2. Zhu, Z.; Liu, Y.; He, Y.; Wu, W.; Wang, H.; Huang, C.; Huang, C. Fuzzy PID Control of the Three-Degree-of-Freedom Parallel Mechanism Based on Genetic Algorithm. *Appl. Sci.* **2022**, *12*, 11128. [CrossRef]
3. Qiao, L.; Zhao, M.; Wu, C.; Ge, T.; Fan, R.; Zhang, W. Adaptive PID control of robotic manipulators without equality/inequality constraints on control gains. *Int. J. Robust Nonlinear Control* **2022**, *32*, 9742–9760. [CrossRef]
4. Mandava, R.K.; Vundavilli, P.R. An adaptive PID control algorithm for the two-legged robot walking on a slope. *Neural Comput. Appl.* **2020**, *32*, 3407–3421. [CrossRef]
5. Nguyen, D.N.; Nguyen, T.A. A Novel Hybrid Control Algorithm Sliding Mode-PID for the Active Suspension System with State Multivariable. *Complexity* **2022**, *2022*, 9527384. [CrossRef]
6. Ghani, M.F.; Ghazali, R.; Jaafar, H.I.; Soon, C.C.; Sam, Y.; Has, Z. Improved Third Order PID Sliding Mode Controller for Electrohydraulic Actuator Tracking Control. *J. Robot. Control* **2022**, *3*, 219–226. [CrossRef]
7. Fliess, M.; Join, C. Intelligent PID controllers. In Proceedings of the 2008 16th Mediterranean Conference on Control and Automation, Ajaccio, France, 25–27 June 2008; pp. 326–331.
8. Fliess, M.; Join, C. Model-free control. *Int. J. Control* **2013**, *86*, 2228–2252. [CrossRef]
9. Sira-Ramirez, H.; Marquez, R.; Fliess, M. On the generalized pid control of linear dynamic systems. In Proceedings of the 2001 European Control Conference (ECC), Porto, Portugal, 4–7 September 2001; pp. 166–171.
10. Becedas, J.; Trapero, J.R.; Feliu, V.; Sira-Ramirez, H. Adaptive Controller for Single-Link Flexible Manipulators Based on Algebraic Identification and Generalized Proportional Integral Control. *IEEE Trans. Syst. Man Cybern. Part B* **2009**, *39*, 735–751. [CrossRef]
11. Zurita-Bustamante, E.W.; Linares-Flores, J.; Guzman-Ramirez, E.; Sira-Ramirez, H. A Comparison Between the GPI and PID Controllers for the Stabilization of a DC-DC "Buck" Converter: A Field Programmable Gate Array Implementation. *IEEE Trans. Ind. Electron.* **2013**, *58*, 5251–5262. [CrossRef]
12. Oldham, K.B.; Zoski, C.G. Analogue instrumentation for processing polarographic data. *J. Electroanal. Chem. Interfacial Electrochem.* **1983**, *157*, 27–51.
13. Westerlund, S.; Ekstam, L. Capacitor theory. *IEEE Trans. Dielectr. Electr. Insul.* **1994**, *1*, 826–839. [CrossRef]
14. Bagley, R.; Calico, R. Fractional order state equations for the control of viscoelasticallydamped structures. *J. Guid. Control Dyn.* **1991**, *14*, 301–311. [CrossRef]
15. Makroglou, A.; Miller, R.K.; Skaar, S. Computational results for a feedback control for a rotating viscoelastic beam. *J. Guid. Control Dyn.* **1994**, *17*, 84–90. [CrossRef]
16. Podlubny, I. Fractional-order systems and $PI^{\lambda}D^{\mu}$-controllers. *IEEE Trans. Autom. Control* **1999**, *44*, 208–214. [CrossRef]
17. Safarzadeh, O.; Noori-kalkhoran, O. A fractional PID controller based on fractional point kinetic model and particle swarm optimization for power regulation of SMART reactor. *Nucl. Eng. Des.* **2021**, *377*, 111137. [CrossRef]
18. Pereira, L.F.D.S.C.; Batista, E.; de Brito, M.A.G.; Godoy, R.B. A Robustness Analysis of a Fuzzy Fractional Order PID Controller Based on Genetic Algorithm for a DC-DC Boost Converter. *Electronics* **2022**, *11*, 1894. [CrossRef]
19. Warrier, P.; Shah, P. Optimal Fractional PID Controller for Buck Converter Using Cohort Intelligent Algorithm. *Appl. Syst. Innov.* **2021**, *4*, 50. [CrossRef]
20. Seo, S.-W.; Choi, H.H. Digital Implementation of Fractional Order PID-Type Controller for Boost DC-DC Converter. *IEEE Access* **2019**, *7*, 142652–142662. [CrossRef]
21. Wang, H.; Lu, J. Research on Fractional Order Fuzzy PID Control of the Pneumatic-hydraulic Upper Limb Rehabilitation Training System Based on PSO. *Int. J. Control Autom. Syst.* **2022**, *20*, 310–320. [CrossRef]
22. Abood, L.H.; Oleiwi, B.K. Design of fractional order PID controller for AVR system using whale optimization algorithm. *Indones. J. Electr. Eng. Comput. Sci.* **2021**, *23*, 1410–1418. [CrossRef]
23. Altbawi, S.M.A.; Mokhtar, A.S.B.; Jumani, T.A.; Khan, I.; Hamadneh, N.N.; Khan, A. Optimal design of Fractional order PID controller based Automatic voltage regulator system using gradient-based optimization algorithm. *J. King Saud Univ. Eng. Sci.* 2021, *in press*. [CrossRef]
24. Koszewnik, A.; Pawluszewicz, E.; Ostaszewski, M. Experimental Studies of the Fractional PID and TID Controllers for Industrial Process. *Int. J. Control Autom. Syst.* **2021**, *19*, 1847–1862. [CrossRef]
25. Zhang, F.; Yang, C.; Zhou, X.; Gui, W. Optimal Setting and Control Strategy for Industrial Process Based on Discrete-Time Fractional-Order $PI^{\lambda}D^{\mu}$. *IEEE Access* **2019**, *7*, 47747–47761. [CrossRef]
26. Mohammad, A.F.; Abdulsalam, M.A. Fractional order PID controller tuned by bat algorithm for robot trajectory control. *Indones. J. Electr. Eng. Comput. Sci.* **2012**, *21*, 74–83.

27. Saadatmand, M.; Gharehpetian, G.B.; Kamwa, I.; Siano, P.; Guerrero, J.M.; Haes Alhelou, H. A Survey on FOPID Controllers for LFO Damping in Power Systems Using Synchronous Generators, FACTS Devices and Inverter-Based Power Plants. *Energies* **2021**, *14*, 5983. [CrossRef]
28. Paducel, I.; Safirescu, C.O.; Dulf, E.-H. Fractional Order Controller Design for Wind Turbines. *Appl. Sci.* **2022**, *12*, 8400. [CrossRef]
29. Shah, P.; Agashe, S. Review of fractional PID controller. *Mechatronics* **2016**, *38*, 29–41. [CrossRef]
30. Padhee, S.; Gautam, A.; Singh, Y.; Kaur, G. A Novel Evolutionary Tuning Method for Fractional Order PID Controller. *Int. J. Soft Comput. Eng.* **2011**, *1*, 1–9.
31. Lazarević, M.P.; Batalov, S.A.; Latinović, T.S. Fractional PID Controller Tuned by Genetic Algorithms for a Three DOF's Robot System Driven by DC motors. In Proceedings of the 6th Workshop on Fractional Differentiation and Its Applications Part of 2013 IFAC Joint Conference SSSC, FDA, TDS, Grenoble, France, 13–15 February 2013; pp. 385–390.
32. Khather, S.I.; Almaged, M.; Abdullah, A.I. Fractional order based on genetic algorithm PID controller for controlling the speed of DC motors. *Int. J. Eng. Technol.* **2018**, *7*, 5386–5392.
33. Bingül, Z.; Karahan, O. Fractional PID controllers tuned by evolutionary algorithms for robot trajectory control. *Turk. J. Electr. Eng. Comput. Sci.* **2012**, *7*, 1123–1136. [CrossRef]
34. Gad, S.; Metered, H.; Bassuiny, A.; Abdel Ghany, A.M. Multi-objective genetic algorithm fractional-order PID controller for semi-active magnetorheologically damped seat suspension. *J. Vib. Control* **2017**, *23*, 1248–1266. [CrossRef]
35. Chen, Z.; Yuan, X.; Ji, B.; Wang, P.; Tian, H. Design of a fractional order PID controller for hydraulic turbine regulating system using chaotic non-dominated sorting genetic algorithm II. *Energy Convers. Manag.* **2014**, *84*, 390–404. [CrossRef]
36. Wan, J.; He, B.; Wang, D.; Yan, T.; Shen, Y. Fractional-Order PID Motion Control for AUV Using Cloud-Model-Based Quantum Genetic Algorithm. *IEEE Access* **2019**, *7*, 124828–124843. [CrossRef]
37. Jaiswal, S.; Chiluka, S.K.; Seepana, M.M.; Babu, G.U.B. Design of Fractional Order PID Controller Using Genetic Algorithm Optimization Technique for Nonlinear System. *Chem. Prod. Process. Model.* **2020**, *15*, 20190072. [CrossRef]
38. Devaraj, S.V.; Gunasekaran, M.; Sundaram, E.; Venugopal, M.; Chennianppan, S.; Almakhles, D.J.; Subramaniam, U.; Bhaskar, M.S. Robust Queen Bee Assisted Genetic Algorithm (QBGA) Optimized Fractional Order PID (FOPID) Controller for Not Necessarily Minimum Phase Power Converters. *IEEE Access* **2021**, *9*, 93331–93337. [CrossRef]
39. Cao, J.; Cao, B. Design of Fractional Order Controllers Based on Particle Swarm Optimization. In Proceedings of the 1st IEEE Conference on Industrial Electronics and Applications, Singapore, 24–26 May 2006; pp. 1–6.
40. Gouta, H.; Said, S.H.; Barhoumi, N.; M'Sahli, F. Observer-Based Backstepping Controller for a State-Coupled Two-Tank System. *IETE J. Res.* **2015**, *63*, 259–268. [CrossRef]
41. Join, C.; Sira-Ramírez, H.; Fliess, M. Control of an uncertain three-tank system via on-line parameter identification and fault detection. *IFAC Proc. Vol.* **2005**, *38*, 251–256. [CrossRef]
42. Sorcia-Vázquez, F.; García-Beltrán, C.; Valencia-Palomo, G.; Brizuela-Mendoza, J.; Rumbo-Morales, J. Decentralized robust tube-based model predictive control: Application to a four-tank system. *Rev. Mex. Ing. QuḾica* **2020**, *19*, 1135–1151. [CrossRef]
43. Stanislawski, R.; Rydel, M.; Li, Z. A New Reduced-Order Implementation of Discrete-Time Fractional-Order PID Controller. *IEEE Access* **2022**, *10*, 17417–17429. [CrossRef]
44. Mirjalili, S. Genetic Algorithm. In *Evolutionary Algorithms and Neural Networks: Theory and Applications*; Kacprzyk, J., Ed.; Springer International Publishing: Warsaw, Poland, 2019; pp. 43–55.
45. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126. [CrossRef]
46. Duriez, T.; Brunton, S.L.; Noack, B.R. *Machine Learning Control—Taming Nonlinear Dynamics and Turbulence*; Springer: Cham, Switzerland, 2017; pp. 11–46.
47. Tepljakov, A.; Alagoz, B.B.; Yeroglu, C.; Gonzalez, E.A.; Hosseinnia, S.H.; Petlenkov, E.; Ates, A.; Cech, M. Towards Industrialization of FOPID Controllers: A Survey on Milestones of Fractional-Order Control and Pathways for Future Developments. *IEEE Access* **2021**, *9*, 21016–21042. [CrossRef]