Advanced Engineering Technology and Application
An International Journal

http://dx.doi.org/10.18576/aeta/130101

Experimental Investigation of Second-Order Transfer Function Parameters

Monzer M. Krishan[1*]

[1] Department of Mechatronics Engineering, Faculty of Engineering Technology Al-Balqa Applied University, Amman, Jordan.

Received: 27 Jul. 2024, Revised: 2 Aug. 2024, Accepted: 5 Aug. 2024.
Published online: 1 Sep. 2024.

Abstract: This study investigates the fundamental characteristics of the quadratic transfer function, widely acknowledged as the most basic function exhibiting complex poles in control system theory. The transfer function assumption becomes very significant, mainly when designing the control systems in analog devices. For this specific topic, the research is planned to systematically collect and analyze critical data regarding the RTU and its importance in the overall design of the power plant. Moreover, the study undertakes the quadratic transfer function as the appropriate initial filter design followed by other filter types, namely ones of second-order divisions. This study proposes integrating the Proportional-Integral-Derivative (PID) controller to improve the dynamic aspects of reducing a second-order transfer function. The endpoint is to find one particular parameter configuration of the PID controller appropriate for system stability during the transition, which is a significant task in control system design. The investigation considers a change of transfer function parameter to find out the implications on transfer function unit step response parameters. The analysis contributes to the acquired knowledge of the system's behavior and competes with the theoretical concepts of control system dynamics.

Keywords: SOTF, PID, optimization, response, parameters, Response specification, steady state, error.

1 Introduction

Transfer functions are mathematical models that estimate the signal change from the input to the output signal. Therefore, they are commonly used in physics, mathematics, engineering, chemistry, and others to mimic and describe different systems. For instance, the transfer functions can describe the dynamics of the linear systems covering the mechanics, the electrical circuits, and the chemistry. They are also applied to the investigation of the dynamic system response to a given input signal and the system's stability and mode of behavior. Transfer functions can control the category of controllers and amplify a system's performance.

Moreover, transfer functions model the behavior of nonlinear systems, including those with some feedback frequently. Through it, the system's control and stability performance are analyzed, which is called the transfer function. Such investigations investigate transfer functions to determine and represent actual physical events [1-5]. As stated in [6–9], identification is vital for industrial operations. For modeling and control applications in [10], a fractional transfer function is utilized to create a low-order algorithm. The studies above demonstrate that transfer functions may represent thermal, hydraulic, electrical, and hybrid systems. Second-order transfer functions (SOTFs) hold a unique position as they can represent a broad range of physical systems due to their inherent characteristics. In particular, the studies cited in references [11-15] employ systems featuring a SOTF, for which the transfer function structure is known. Still, the function parameters are often unknown and must be derived from the parameters of the original system's differential equation model. Several approaches are available for calculating these parameter values, with one option being using metaheuristic algorithms as described in the reference [16]. Although such methods may require iterative processing, they offer the advantages of simplicity and ease of implementation. At the same time, the slow-moving parameters of the process do not pose a significant obstacle to accurate parameter estimation [17,18].

The estimation of parameters for a transfer function is a widely discussed topic. In the case of an electrohydraulic servo transfer function, various relevant studies, such as [19], are used to estimate the function's parameters based on amplitude-frequency characteristics. The problem of finding the exact solution is addressed through implementing different methodologies such as heuristic, metaheuristic, and exact methods. For instance, [20] can be utilized to estimate the system's transfer function by means of Matlab's System Identification Toolbox and Vector Fitting based on the frequency of the system's response. Therefore, on the one hand, he [Neuman] utilizes the second-order dynamic

*Corresponding author e-mail: drkrishan@bau.edu.jo

system Newton identification method for parameter estimation, and on the other hand, he [Neuman] uses the heuristic algorithm. [22,23] The parameters are estimated through the Newton method, in which the least squares technique and the damping cyclic parameterization method are applied. Some studies, such as [24-26], suggest using the impulse response as an alternative to the step response for parameter estimation.

In contrast, others, such as [27,28], focus on using the gradient for parameter identification. In [1], the authors determine the transfer function parameters based on Green's function and test their hypothesis using an experimental setup. Finally, [29] focuses on locating a Flame Transfer Function to find the Wiener-Hopf inversion motif while avoiding skewed feedback system results. To do this, the writers evaluate several identifying techniques and compare them. Some investigation on the determination of transfer function parameters using frequency response. In [30], the properties of a high-frequency induction machine are determined through analysis of its frequency response. In [31], an imprecise frequency response function and modal data are utilized to identify structural degradation.

In contrast, specific attempts, such as the one shown in [32], utilize the reaction in the temporal domain, as with the current investigation. Much study has been conducted on the use of metaheuristic algorithms for the determination of parameters. Several publications, such as [2,33], apply diverse identification techniques and algorithms. There are several bio-inspired metaheuristic algorithm types. This method type is often adaptable to various applications and easily parallelizable. Yet, its results are frequently approximations based on the search space. Genetic algorithms are commonly used as an algorithm type to suggest a locally optimal alternative when the answer is beyond the scope of the search. Two examples of its widespread application are the optimization of motor control [2] and crude oil operations in refineries [34]. Unfortunately, their efficacy depends on several crucial characteristics that are difficult to estimate. Thus, studies such as [35] rely only on generic parameter methods (population, stop condition, and search range). The particle swarm optimization approach is a popular alternative with various versions and uses. Regrettably, the original version of the algorithm tends to achieve irrational optimizations quickly. Therefore, many authors have suggested significant algorithmic improvements [36].

Conversely, approaches like cuckoo search utilize bird flight patterns to locate targets [37]. Although the cuckoo algorithm yields similar outcomes to genetic algorithms, it necessitates fewer setup parameters. Nevertheless, its convergence is gradual.

There is a significant amount of curiosity in proportional integral derivative (PID) controllers, which are utilized in several industrial applications. They are architecturally straightforward and offer strong performance throughout various operational conditions. Without a thorough understanding of the process, these controls represent the most effective options. These controllers consist of three main elements: the proportionate (P), integral (I), and derivative (D) components. The proportionate component is responsible for maintaining the selected set point, known as reference tracking. Meanwhile, the integral and derivative sections measure the sum of previous errors and the rate of error change in the system.

PID controllers are often more prevalent than other types of control. It is even referred to as the "bread and butter" of control engineering as it is an integral part of the toolset of any control engineer. Moreover, it is the most general feedback and a versatile and straightforward technique. Almost ninety-five percent of control loops in process control are PID-type. The majority of loops utilize PID control. Nowadays, PID controllers are employed anywhere where control is required. The controllers are available in various formats [38-43]. Equation 1 describes the overall structure of a PID controller.

Tuning a PID controller discusses the modification (or update) of its many settings to get an optimal overall response (P. I and D). Essential response criteria include stability, peak time, overshoot, and intended rising time.

A significant modification of the PM settings can satisfy various processes' demands. The tuning process entails examining the system's step-response to ascertain the PID parameters if the system may be turned off. However, in most industrial applications, the system must remain online, and competent employees must manually perform the tuning, which can be prone to human error and ambiguity. Another tuning approach is the Ziegler-Nichols method [38, 39], which is suitable for online computations but can involve trial and error and may not be ideal for specific situations. Equation 1 demonstrates the output function based on the set point.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de(t)}{dt} \quad (1)$$

The SOTF [38], [39] practically represent a dynamic system with linear time constants. A transfer function is a job Task variant in mathematics. The transfer function for finite-dimensional approaches is merely a Boolean function of a complex variable [40, 41]. Examination of basic algebraic manipulation of differential equations representing systems can provide the transfer function. Transfer functions can explain processes with extremely high order, including

controlled infinite-dimensional systems with partial binary equations [42, 43]. The system's transfer function can be determined through system experiments. As indicated in Figure 1, SOTF response is characterized by the following characteristics [44], [45], and [46]:

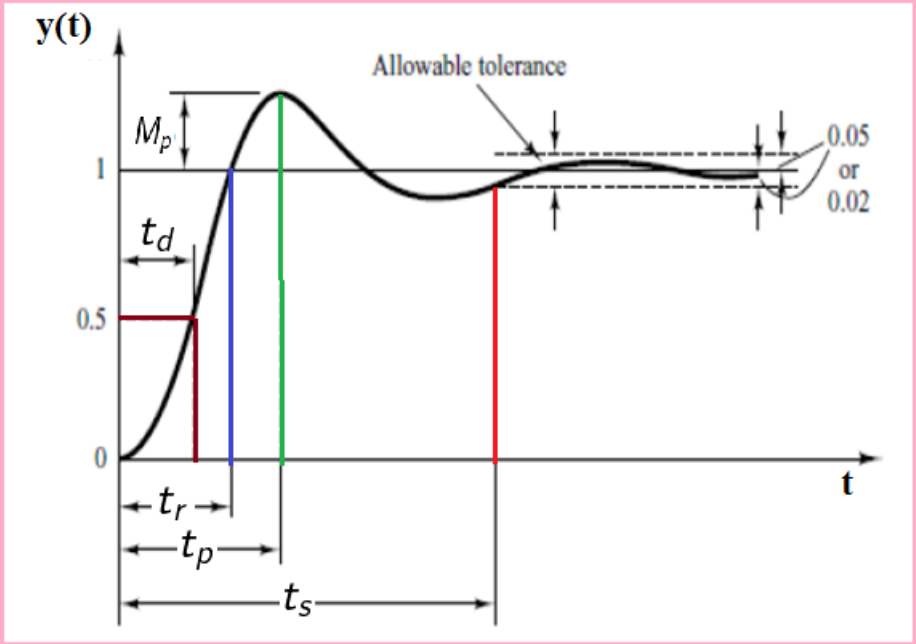

Fig. 1: SOTF specifications.

- **Delay time** t_d: The time required for the reaction to reach 50 percent of its starting value.
- **Rise time** t_r**:** The time needed for a response's final value to grow by 10%,5%, or 100%. Typically, the 0%-to-100%rise time is employed for second-order systems that are dampened. The 10% to 90% rise time is generally used for systems with excessive damping.
- **Peak time** t_b**:** The time needed for a reaction to reach the initial peak of an overshoot.
- **Maximum overshoot** M_p**:** Maximum overshoot is the most outstanding value measured from unity at the apex of the response curve. Commonly, the maximum percent overrun is utilized if the final steady-state value of the response departs from the agreement. The most remarkable (percent) overrun quantity reflects the system's relative stability.
- **Settling time** t_s**:** The time necessary for the response curve to approach and maintain inside a band of the final value of size, expressed as a percentage of the final value (often 2 or 5 percent). The settling time is proportional to the control system's most crucial time constant (the time required to move the system response to a steady state).

2 Context and Typical Applications of a Second-Transfer Function

The purpose of this section is to discuss the SOTF and the systems it is typically used to represent. Moreover, A second-order function can represent both a higher-order and a first-order system, allowing for a broad range of engineering systems to be characterized. Thus, the SOTF can represent numerous engineering systems. Equation (2) describes its typical form:

$$G(s) = \frac{D}{As^2 + Bs + C} \tag{2}$$

Equation (2) indicates a vast number of non-zero physical systems. If the equation is rearranged to account for the dynamic response, the identical equation is rewritten as follows:

$$G(s) = K\frac{\omega_n^2}{s^2 + 2\mathfrak{T}\omega_n s + \omega_n^{2\prime}} \tag{3}$$

where the dc gain is represented by K, which is the proportion between both the steady-state and amplitude response when a step input is applied to a stable system, the damping factor is defined by $\mathfrak{T}$, and the undamped natural

frequency is given by ω_n . In comparison, a transfer function is characterized by Equation (4) as the output over the input:

$$G(s) = \frac{Output(s)}{Input(s)} \tag{4}$$

Equation (5) defines the output of a system:

$$Output(s) = \frac{D}{As^2 + Bs + C} * Input(s) \tag{5}$$

Given the above, [A, B, C, D] is defined as the unknown parameters vector. Assuming that the input signal to the system is a step signal with a magnitude of 'a', the transfer function can be expressed in the following manner:

$$Output(s) = \frac{D}{As^2 + Bs + C} * \frac{a}{s} \tag{6}$$

In contrast, the FVT enables the computation of the end value of a transfer function even when its transitional period is not defined. Equation (7) also demonstrates that the theorem applies to stable systems.

$$\lim_{s\to 0} sF(s) = \lim_{t\to\infty} F(t) \tag{7}$$

Equation (8) describes the calculated value of the transfer function, such as equation (7), when the FTV is utilized:

$$Output(s) = \frac{D}{C} * a \tag{8}$$

Therefore, the ultimate value of this sort of function is decided through the values of the factors *C* and *D*. The FVT was employed as a constraint to which the algorithms essentially adhere when looking for factors.

2.1 Electrical System Transfer Function

Using the equations of Ohm and Kirchhoff, it is known that the dynamic model of an RLC circuit, like the one seen in Figure 1, is governed by equation (9).

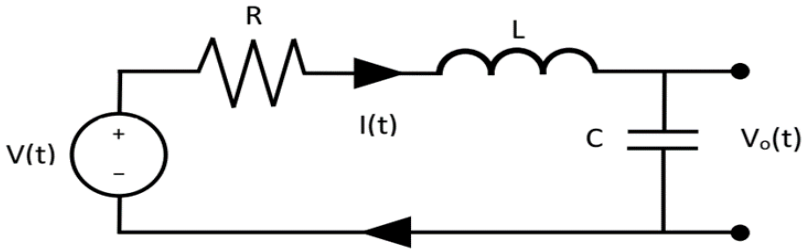

Fig. 2: RLC circuit employed to determine the SOTF in electrical systems.

$$V(t) = V_0 + LC\frac{d^2V_0}{dt^2} + RC\frac{dV_0}{dt} \tag{9}$$

Equation (9) presents the dynamic model of the circuit when considering the input as the voltage source V(t) and the output as the capacitor voltage (Vo). The circuit consists of a power source voltage given by V(t), a mesh current represented by I(t), a resistor denoted by R, an inductor value indicated by L, and a capacitor value represented by C. The results were as follows after applying the location transform while taking into account null initial conditions:

$$G(s) = \frac{V_0(s)}{V(s)} = \frac{1}{LCs^2s + CR + 1} \tag{10}$$

Equation (11) provides the predictable system output when a voltage input of magnitude *a* is supplied.

$$V_0(s) = \frac{1}{LCs^2s + CR + 1} * \frac{a}{s} \tag{11}$$

Equation (11) represents the second-order transfer function that characterizes the circuit. However, it is possible to obtain the same expression using metaheuristic techniques and the standard structure of a SOTF without knowledge of the system's dynamic model or parameter values. This approach is especially advantageous when only the system's output is observable, and its specifics are unknown.

Another illustration of the electrical system is the R.L. circuit in Figure 3. The voltage across the resistor serves as a representation of the system's output, and it is referred to as a first-order transfer function. A differential equation describes equation (12)'s dynamic model of the R.L. circuit, and equation (13) provides its associated transfer function.

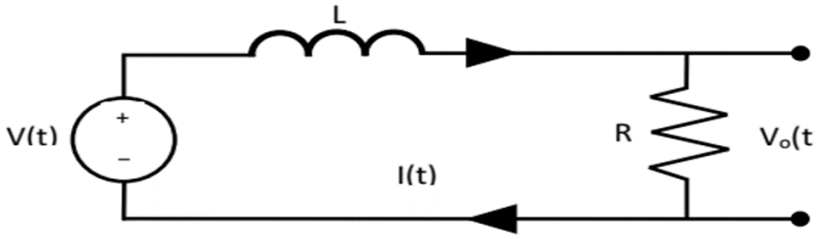

Fig. 3: The R.L. circuit utilized in electrical systems for obtaining the first-order transfer function.

$$V(t) = V_0 + \frac{R}{L}\frac{dV_0}{dt} \tag{12}$$

$$G(s) = \frac{V_0(s)}{V(s)} = \frac{R}{Ls + R} \tag{13}$$

2.2. Mechanical Systems Transfer Function

The transfer function of a mechanical system is demonstrated in Figure 4 with a mass-spring-damper (MSD) system. The dynamic model is generated by Newton's second law, resulting in the model specified by Equation (14).

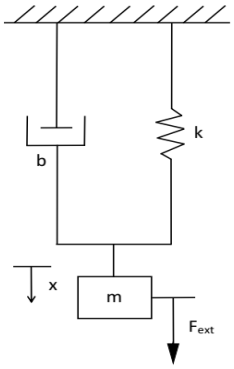

Fig. 4: A mass spring damper was utilized to determine the SOTF in mechanical systems.

$$kx + m\frac{d^2x}{dt^2} + b\frac{dx}{dt} = F_{ext} \tag{14}$$

where Fext represents the external force, the displacement *is* represented by *x*, the mass is represented by m, b is the friction coefficient, and finally, the spring constant is given by k. Equation (14) represents the transfer function of the system:

$$G(s) = \frac{X(s)}{F_{ext}(s)} = \frac{1}{ms^2 + bs + k} \tag{15}$$

The proposed approach becomes particularly valuable in mechanical systems, where measurements of variables and coefficients are often more intricate than in electrical systems. Coefficients such as the friction coefficient and spring constant can be challenging to estimate and necessitate specialized testing. Figure 5 illustrates a different mechanical system: a high-order (h_o) mechanical system. The differential equation represents the mechanical circuit, while equation (16) provides the transfer function corresponding to the displacement in m_1 provided in equation (20).

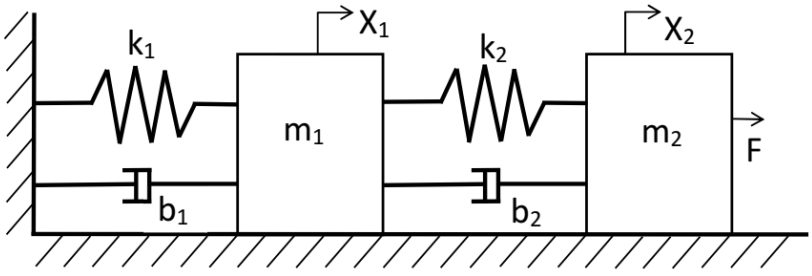

Fig. 5: Mass-spring-damper for obtaining the high-order transfer function in mechanical systems.

$m_1\ddot{x}_1 = k_2(x_2 - x_1) + b_2(\dot{x}_2 - \dot{x}_1) - k_1x_1 - b_1\dot{x}_1 m_2\ddot{x}_2 = F - k_2(x_2 - x_1) - b_2(\dot{x}_2 - \dot{x}_1)$ (16)

$$G(s) = \frac{X_1(s)}{F(s)} = \frac{b_2S + k_2}{l} \quad (17)$$

where l is given by $[m_1m_2s^4 + (b_2m_1 + m_2(b_1 + b_2))\, s^3 + (m_2(k_1 + k_2) + k_2m_1 - b^2_2 + b_2(b_1 + b_2))s^2 + (k_2(b_1 + b_2) - 2b_2k_2 + b_2(k_1 + k_2))s + k_2(k_1 + k_2) - k^2_2]$. Despite including a zero, the system's order is larger than 2.

2.3. Electromechanical System Transfer Function

Figure 6 demonstrates a typical example of an electromechanical system: the direct current motor. This type of system consists of both mechanical and electrical components. As shown by Equation (18), its model comprises two differential equations, one for the mechanical component and one for the electrical component.

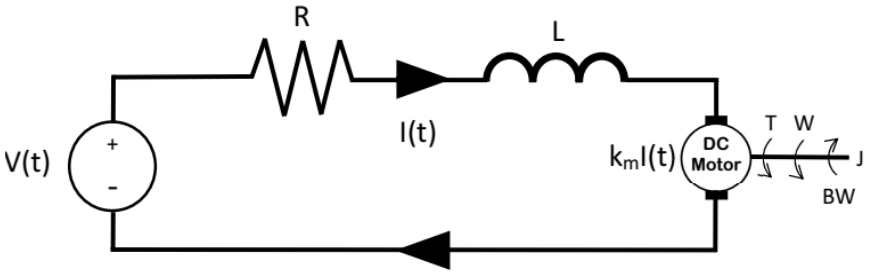

Fig. 6: Scheme for D.C. motors used to derive the SOTF in electromechanical systems.

$$V(t) = RI(t) + L\frac{dI(t)}{dt} + K_e\omega(t) \quad (18)$$

$$K_mI(t) = J\frac{d\omega(t)}{dt} + B\omega(t) \quad (19)$$

The voltage is represented by V(t), while I(t) represent the current in the direct current motor. The system also comprises other elements, such as the armature resistance (R), armature inductance (L), electrical (K_e) and mechanical (K_m) constants, inertial momentum (J), and coefficient of friction (B). Combining the motor equations and assuming zero initial conditions makes it possible to obtain the speed-voltage transfer function represented by equation (19). This transfer function facilitates the regulation of a mechanical variable, such as the speed, based on an electrical input, such as voltage.

$$G(s) = \frac{\omega(s)}{V(s)} = \frac{K_m}{LJs^2 + (RJ + LB)s + (RB + K_mK_S)} \quad (20)$$

Several of these metrics' direct measurements are complex or intrusive. Hence, parametric estimating approaches are applied.

3 Modeling SOTF

SOTF (shown in equation 1) [47], [48] is a mathematical representation of the control plant (such as a D.C. motor), as shown in Figure 7. It can be easily used to model the plant and analyze the effects of SOTR parameters (A, B, C, and D) on the response specification [49].

$$G(s) = \frac{A}{Bs^2 + Cs + D} \quad (21)$$

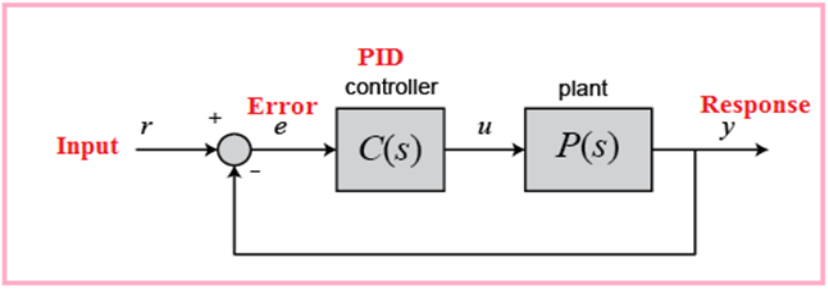

Fig. 7: Plant modeling.

A proportional integral derivative (PID) controller could be incorporated into the model to provide a steady state response by minimizing the error between the input and system response. Equation 22 represents the output of the PIDC [50].

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \quad (22)$$

Industrial Controls use a PID (Proportional-Integral-Derivative), a feedback mechanism that helps regulate certain process variables like flow pressure and temperature. The controller has three parts, proportional, integral, and derivative, which work together to solve the problem.

PID controllers use a regulatory mechanism ensuring that process variable changes meet the set output range. The error has a proportional effect on the output to be changed, while the sum of all past errors determines the magnitude of the integral component for the required output correction. The adjustable part of the derivative section interacts with the rapid change of the error and creates the output proportionately. Proportional Integral Derivative controllers play significant roles in process control, ensuring excellent and steady control. Achieving consistent product quality requires a system response within a specific range. Any adjustments involving introducing or eliminating a product or implementing a ramping function necessitate precise regulation. Despite the straightforward premise of PID control, its underlying mathematics are intricate and achieving optimal performance mandates careful selection of process-specific values across various interconnected parameters.

The procedure of determining these parameters is known as tuning. When a PID controller is optimally tuned, the device minimizes deviation from the set point and responds promptly to disturbances or changes in the set point with a slight overshoot.

Even when the equipment is equivalent, each process control has its standards. SPTF parameters vary, as will the ambient reaction. As depicted in Figure 8, the PID parameters (namely the gain applied to the correction factor and the time used for integral and derivative computations, dubbed "reset" and "rate") must be adjusted to account for these regional variations.

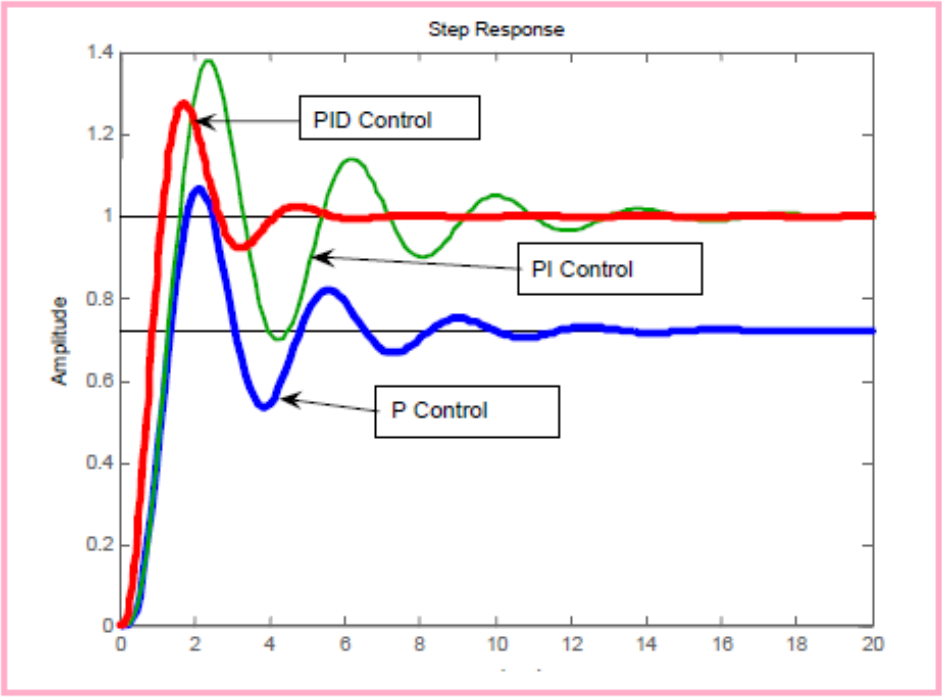

Fig. 8: PID tuning.

Suggested model to analyze SOTF parameters

Based on Equation 21, a Simulink model shown in Figure 4 was built. A PIDC controller was created using Equation 22 (see Figure 9) and added to the model to enhance SOTF specifications and minimize errors between the input step uint function and the output response.

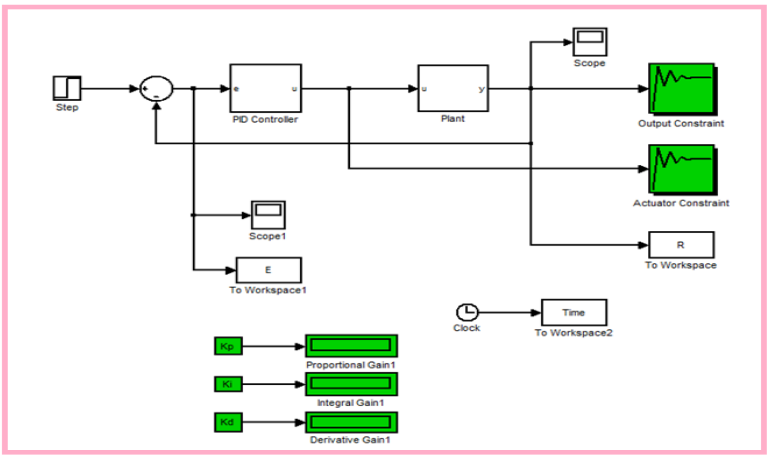

Fig. 9: SOTF Simulink model.

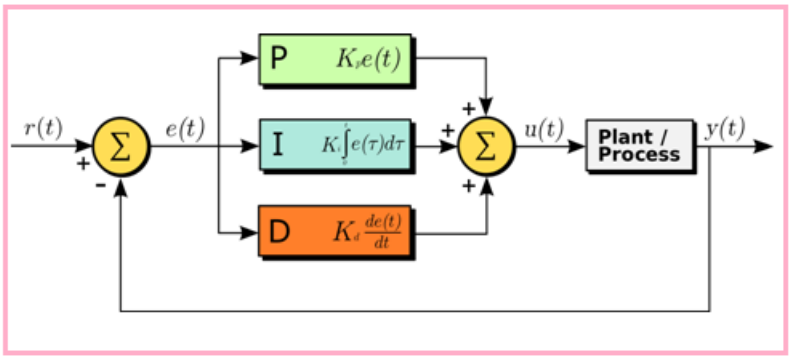

Fig. 10: PIDC Simulink model.

The model was implemented using various values for SOTR parameters each time the model was tuned, and the required optimized values for K_p, K_i, and K_d were obtained; these values were used to get the optimal steady state as shown in Figure 11:

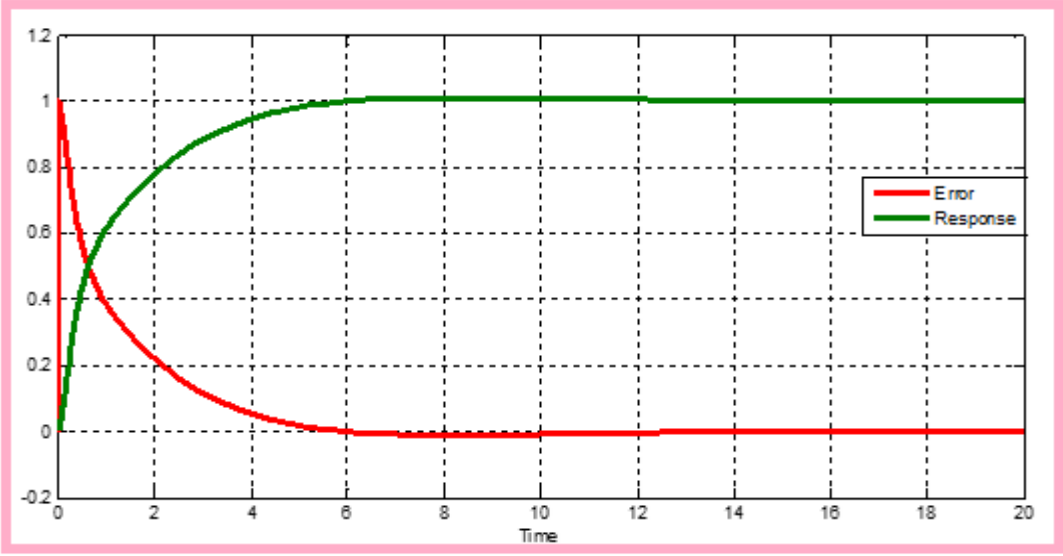

Fig. 11: Optimal response steady state.

Table 1 shows the steady states and the optimal values for PIDC controller parameters using various values for SOTF parameters:

Table 1: Steady states using multiple values for SOTF parameters.

A	B	C	D	K_p	K_i	K_d	t_s
5	1	3	1	0.3096	0.1741	0.4810	6
1	1	1	1	1.934	0.2253	0.224	10
2	1	1	1	2.2177	0.7206	0.198	10
3	1	1	1	1.4597	0.5630	0.2542	10
6	0.3	1	1	1.8009	0.5927	0.2073	5
6	0.4	1	1	1.5805	0.5544	0.2305	5
6	0.5	1	1	1.4401	9.5275	0.2320	5
6	0.6	1	1	1.3661	0.5202	0.2544	5
6	0.6	2	1	2.1893	0.5786	0.1207	5
6	2	1	1	0.5036	0.3159	0.5036	10
6	3	1	1	0.1221	0.1287	0.3620	5
6	4	1	1	0.1021	0.1073	0.3823	5
6	5	1	1	0.1022	0.0984	0.3903	15
6	0.1	1	1	2.579	0.6991	0.0961	7
6	0.2	1	1	1.8932	0.6048	0.1047	5
6	0.6	3	1	1.2238	0.3160	-0.174	5
6	0.6	4	1	2.0311	0.3491	-0.2019	5
6	0.6	5	1	1.4717	0.3328	-0.2410	5
6	0.6	0.1	1	0.3287	0.3658	0.1980	7
6	0.6	0.2	1	0.5019	0.3260	0.2799	7
6	0.6	0.3	1	0.7426	0.4016	0.3167	7
6	0.6	0.4	1	0.5681	0.3643	0.2167	7
6	0.6	1	2	0.6346	0.5275	0.3275	7
6	0.6	1	3	0.8605	0.7171	0.2838	7
6	0.6	1	5	0.9533	1.0756	0.2753	7
6	0.6	1	0.1	0.7262	0.8246	0.3184	5
6	0.6	1	0.2	0.7282	0.8246	0.3184	5
1	1	1	1	3.8773	1.5104	12.3365	20
6	0.6	1	0.2	0.2649	0.0985	1.9426	40
6	0.6	1	5	0.5453	0.8578	0.3563	5
6	0.6	0.4	1	0.1224	0.1937	0.2079	15
6	0.6	4	1	0.2566	0.0809	1.0496	40
6	0.6	1	1	0.4893	0.2071	0.9282	16
6	0.1	1	1	0.5092	0.4435	1.1202	16
6	4	1	1	0.9378	0.4158	2.2082	15
6	3	1	1	1.9636	0.7653	3.5150	15
6	2	1	1	1.3990	0.4771	5.5334	20
7	1	1	1	0.3838	0.3009	1.7150	20
6	1	1	1	0.4975	0.2109	0.6074	10
5	1	1	1	0.5832	0.1398	0.8868 1	8
4	1	1	1	0.0846	0.1189	0.1366 1	8
3	1	1	1	0.3220	0.1461	2.2114 3	5
2	1	1	1	1.6325	0.2231	0.8019 3	8
7	1	1	1	0.2699	0.2097	0.3576	5
6	1	1	1	0.918	0.5669	0.2967	3
5	1	1	1	0.9327	0.5981	0.2977	4
4	1	1	1	0.9327	0.5981	0.2977	5

4 Experimental Results

Fixing the values of the PIDC controller to the first steady state 1n table and changing the values of SOTF parameters, we have to discuss the following cases:

Case 1: Changing the values for parameter A:

Fixing all parameters values and changing the values of A parameters, Figure 12 shows the results obtained by running the model.

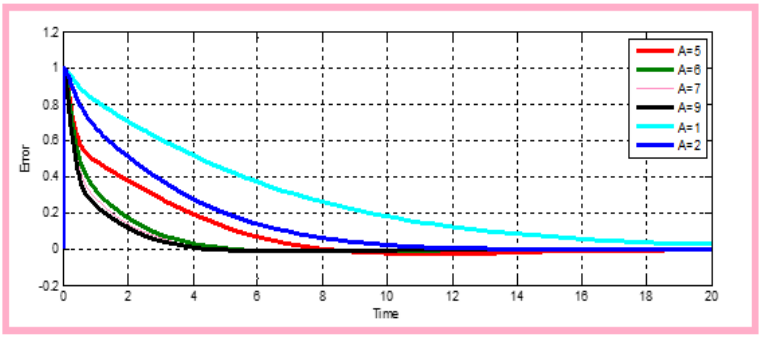

Fig. 12: Response error when changing A.

From Figure 12, we can see that decreasing A increases the settling time and increases the error while setting A to a negative value produces an unsteady response state by rapidly expanding the mistake, as shown in Figure 13:

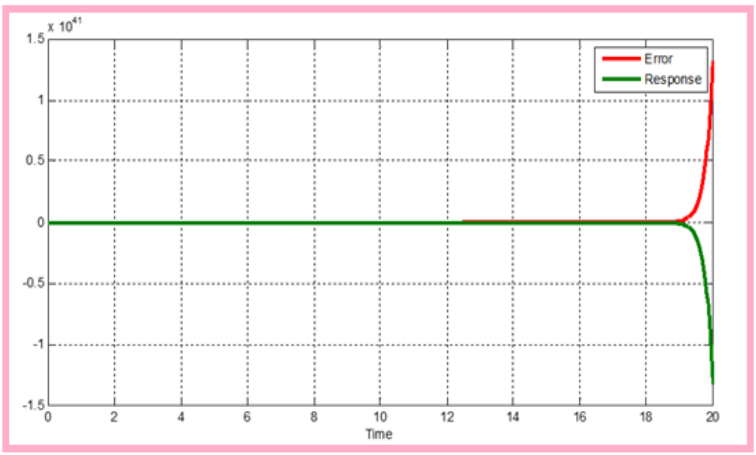

Fig. 13: The effects of using a negative value for A.

Case 2: Changing the values of the B parameter:

Fixing all parameter values and changing the values of B parameters, Figure 14 shows the results obtained by running the model.

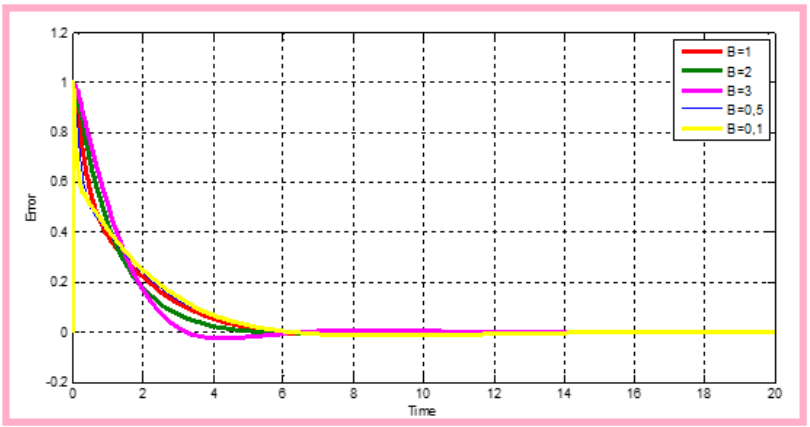

Fig. 14: Response error when changing B.

From Figure 14, we can see that increasing B increases the under-damp and does not affect the settling time, while setting B to a negative value produces an unsteady response state by rapidly expanding the error, as shown in Figure 15:

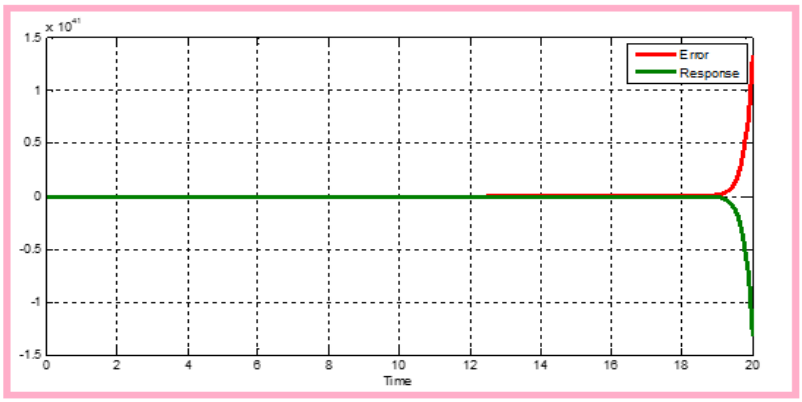

Fig. 15: The effects of using a negative value for B.

Case 3: **Changing the values of the C parameter:**

Fixing all parameter values and changing the values of C parameters, Figure 16 shows the results obtained by running the model

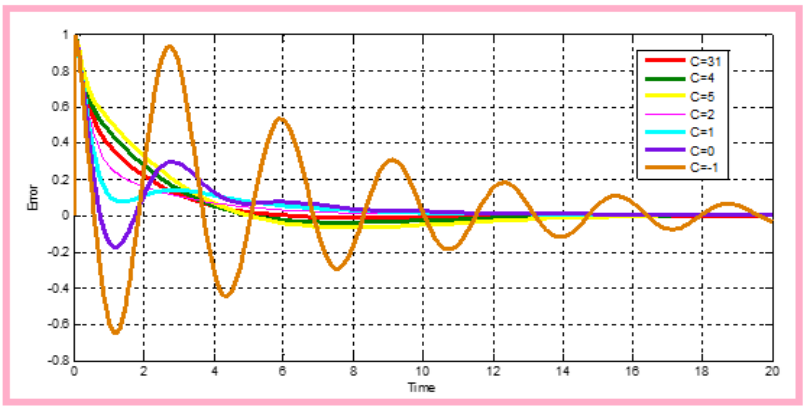

Fig. 16: Response error when changing C.

From Figure 16, we can see that increasing the values of the C parameter does not affect the system response, but decreasing the values, even for negative ones, makes the response over-damp and under-damp.

Case 4: **Changing the values of the D parameter:**

Fixing all parameter values and changing the values of D parameters, Figure 17 shows the results obtained by running the model.

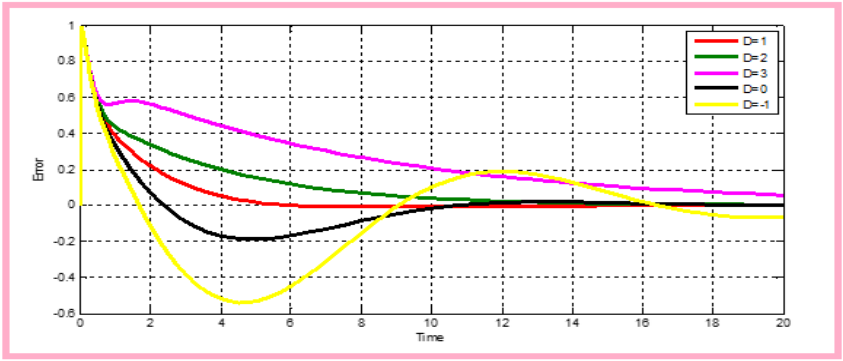

Fig. 17: Response error when changing D.

From Figure 17, we can see that increasing the value of the D parameter negatively affects the steady state and the settling time. Decreasing the deal generates an under-damp and negative error.

5 Conclusion

In this study, the correlation between the SOTF parameters and the system behavior was simulated by modeling the dynamics of the second-order system using the Simulink model. The experiments' results revealed the intrinsic vital importance of elements C and D inside a Proportional-Integral-Derivative (PID) Controller (PIDC), which, in turn, highlights the imperative of rigorous parameter selection when theorizing a PID controller. The investigation analyzed the problem of the parameters restraints and discordances between them and the influence of parameter modification on the system. The model's running experimental appraisal involved applying multiple load disturbances to find and exploit the best combination of SOTF and PIDC parameter settings that would guarantee the highest possible performance of the system in real-world conditions. It provided the opportunity to collect information about the oscillatory behavior of the system and elaborated on which parameters, SOTF and PIDC values can be reached for an optimally resilient and stable configuration.

The research brought to light the characteristics of SOTF, and additionally, they also stressed crucial elements to consider during the selection of PIDC settings. Under real-time load disturbances of sufficient magnitude, the proposed model was subjected to rigorous testing to confirm its practicality for implementation. The findings show that not only quite precisely but also the specified coordination of the SOTF parameters (Setpoint Optimization and Tracking Filter) dramatically increases the speed of external disturbance response without affecting stability characteristics. With such findings, developing proper regulating systems and their engineering becomes crucial.

Reference

[1] Fernandes, A.; dos Santos, M.; Guimarães, G. An analytical transfer function method to solve inverse heat conduction problems. Appl. Math. Model. 2015, 39, 6897–6914. [Google Scholar] [CrossRef]

[2] Rodriguez-Abreo, O.; Rodriguez-Resendiz, J.; Fuentes-Silva, C.; Hernandez-Alvarado, R.; Falcon, M. Self-Tuning Neural Network PID with Dynamic Response Control. IEEE Access 2021, 9, 65206–65215. [Google Scholar] [CrossRef]

[3] Torres-Salinas, H.; Rodríguez-Reséndiz, J.; Estévez-Bén, A.; Cruz Pérez, M.; Sevilla-Camacho, P.; Perez-Soto, G. A hands-on laboratory for intelligent control courses. Appl. Sci. 2020, 10, 9070. [Google Scholar] [CrossRef]

[4] Cruz-Miguel, E.; García-Martínez, J.; Rodríguez-Reséndiz, J.; Carrillo-Serrano, R. A new methodology for a retrofitted self-tuned controller with open-source FPGA. Sensors 2020, 20, 6155. [Google Scholar] [CrossRef] [PubMed]

[5] Macias-Bobadilla, G.; Becerra-Ruiz, J.; Estévez-Bén, A.; Rodríguez-Reséndiz, J. Fuzzy control-based system feedback by OBD-II data acquisition for complementary injection of hydrogen into internal combustion engines. Int. J. Hydrogen Energy 2020, 45, 26604–26612. [Google Scholar] [CrossRef]

[6] Mendoza-Mondragon, F.; Hernandez-Guzman, V.; Rodriguez-Resendiz, J. Robust Speed Control of Permanent Magnet Synchronous Motors Using Two-Degrees-of-Freedom Control. IEEE Trans. Ind. Electron. 2018, 65, 6099–6108. [Google Scholar] [CrossRef]

[7] H. Attar, A. Tahseen Abu-Jassar, A. Amer, V. Lyashenko, V. Yevsieiev, and M. R. Khosravi, "Control System Development and Implementation of a CNC Laser Engraver for Environmental Use with Remote Imaging," 2022, doi: 10.1155/2022/9140156.

[8] Garduno-Aparicio, M.; Rodriguez-Resendiz, J.; Macias-Bobadilla, G.; Thenozhi, S. A Multidisciplinary Industrial Robot Approach for Teaching Mechatronics-Related Courses. IEEE Trans. Educ. 2018, 61, 55–62. [Google Scholar] [CrossRef]

[9] Martínez-Prado, M.A.; Rodríguez-Reséndiz, J.; Goméz-Loenzo, R.A.; Herrera-Ruiz, G.; Franco-Gasca, L.A. An FPGA-Based Open Architecture Industrial Robot Controller. IEEE Access 2018, 6, 13407–13417. [Google Scholar] [CrossRef]

[10] De Keyser, R.; Muresan, C.; Ionescu, C. An efficient algorithm for low-order direct discrete-time implementation of fractional order transfer functions. ISA Trans. 2018, 74, 229–238. [Google Scholar] [CrossRef]

[11] Secer, G.; Saranli, U. Control of Planar Spring–Mass Running Through Virtual Tuning of Radial Leg Damping. IEEE Trans. Robot. 2018, 34, 1370–1383. [Google Scholar] [CrossRef]

[12] Zhao, K.; Cheng, L.; Li, S.; Liao, A. A new updating method for the damped mass-spring systems. Appl. Math. Model. 2018, 62, 119–133. [Google Scholar] [CrossRef]

[13] Odry, A. An open-source test environment for effective development of marg-based algorithms. Sensors 2021, 21, 1183. [Google Scholar] [CrossRef]

[14] Odry, A.; Kecskes, I.; Sarcevic, P.; Vizvari, Z.; Toth, A.; Odry, P. A novel fuzzy-adaptive extended Kalman filter for real-time attitude estimation of mobile robots. Sensors 2020, 20, 803. [Google Scholar] [CrossRef][Green Version]

[15] Odry, A.; Tadic, V.; Odry, P. A Stochastic Logic-Based Fuzzy Logic Controller: First Experimental Results of a Novel Architecture. IEEE Access 2021, 9, 29895–29920. [Google Scholar] [CrossRef]

[16] Hashim, F.A.; Hussain, K.; Houssein, E.H.; Mabrouk, M.S.; Al-Atabany, W. Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems. Appl. Intell. 2021, 51, 1531–1551. [Google Scholar] [CrossRef]

[17] Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. Adv. Eng. Softw. 2014, 69, 46–61. [Google Scholar] [CrossRef][Green Version]

[18] Venkata Rao, R. Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int. J. Ind. Eng. Comput. 2016, 7, 19–34. [Google Scholar] [CrossRef]

[19] Sokolov, V.; Krol, O. Determination of Transfer Functions for Electrohydraulic Servo Drive of Technological Equipment. In Design, Simulation, Manufacturing: The Innovation Exchange; Springer: Cham, Switzerland, 2019; pp. 364–373. [Google Scholar]

[20] Arda Ozdemir, A.; Gumussoy, S. Transfer Function Estimation in System Identification Toolbox via Vector Fitting. IFAC-PapersOnLine 2017, 50, 6232–6237. [Google Scholar] [CrossRef]

[21] Xu, L.; Chen, L.; Xiong, W. Parameter estimation and controller design for dynamic systems from the step responses based on the Newton iteration. Nonlinear Dyn. 2015, 79, 2155–2163. [Google Scholar] [CrossRef]

[22] Xu, L.; Ding, F.; Zhu, Q. Hierarchical Newton and least squares iterative estimation algorithm for dynamic systems by transfer functions based on the impulse responses. Int. J. Syst. Sci. 2019, 50, 141–151. [Google Scholar] [CrossRef][Green Version]

[23] Xu, L. The damping iterative parameter identification method for dynamical systems based on the sine signal measurement. Signal Process. 2016, 120, 660–667. [Google Scholar] [CrossRef]

[24] Xu, L.; Ding, F. Parameter estimation for control systems based on impulse responses. Int. J. Control Autom. Syst. 2017, 15, 2471–2479. [Google Scholar] [CrossRef]

[25] Xu, L.; Ding, F.; Yang, E. Separable Recursive Gradient Algorithm for Dynamical Systems Based on the Impulse Response Signals. Int. J. Control Autom. Syst. 2020, 18, 3167–3177. [Google Scholar] [CrossRef]

[26] Xu, L.; Ding, F.; Zhu, Q. Decomposition strategy-based hierarchical least mean square algorithm for control systems from the impulse responses. Int. J. Syst. Sci. 2021, 1–16. [Google Scholar] [CrossRef]

[27] Xu, L.; Ding, F. Parameter estimation algorithms for dynamical response signals based on the multi-innovation theory and the hierarchical principle. IET Signal Process. 2017, 11, 228–237. [Google Scholar] [CrossRef]

[28] Xu, L.; Ding, F.; Wan, L.; Sheng, J. Separable multi-innovation stochastic gradient estimation algorithm for the nonlinear dynamic responses of systems. Int. J. Adapt. Control Signal Process. 2020, 34, 937–954. [Google Scholar] [CrossRef]

[29] Jaensch, S.; Merk, M.; Emmert, T.; Polifke, W. Identification of flame transfer functions in the presence of intrinsic thermoacoustic feedback and noise. Combust. Theory Model. 2018, 22, 613–634. [Google Scholar] [CrossRef]

[30] Miloudi, H.; Bendaoud, A.; Miloudi, M.; Dickmann, S.; Schenke, S. A novel method of transfer-function identification for modeling D.M. impedance of A.C. motor. In Proceedings of the 2017 International Symposium on Electromagnetic Compatibility—EMC EUROPE, Angers, France, 4–7 September 2017; pp. 1–5. [Google Scholar] [CrossRef]

[31] Farshadi, M.; Esfandiari, A.; Vahedi, M. Structural model updating using incomplete transfer function and modal data. Struct. Control Health Monit. 2017, 24, e1932. [Google Scholar] [CrossRef]

[32] Hu, X.; Siew, W.H.; Judd, M.D.; Peng, X. Transfer function characterization for HFCTs used in partial discharge detection. IEEE Trans. Dielectr. Electr. Insul. 2017, 24, 1088–1096. [Google Scholar] [CrossRef][Green Version]

[33] Yang, B.; Wang, J.; Zhang, X.; Yu, T.; Yao, W.; Shu, H.; Zeng, F.; Sun, L. Comprehensive overview of meta-heuristic algorithm applications on P.V. cell parameter identification. Energy Convers. Manag. 2020, 208, 112595. [Google Scholar] [CrossRef]

[34] Hou, Y.; Wu, N.; Zhou, M.; Li, Z. Pareto-Optimization for Scheduling of Crude Oil Operations in Refinery via Genetic Algorithm. IEEE Trans. Syst. Man Cybern. Syst. 2017, 47, 517–530. [Google Scholar] [CrossRef]

[35] Yu, K.; Qu, B.; Yue, C.; Ge, S.; Chen, X.; Liang, J. A performance-guided JAYA algorithm for parameters identification of photovoltaic cell and module. Appl. Energy 2019, 237, 241–257. [Google Scholar] [CrossRef]

[36] Cao, Y.; Zhang, H.; Li, W.; Zhou, M.; Zhang, Y.; Chaovalitwongse, W.A. Comprehensive Learning Particle Swarm Optimization Algorithm With Local Search for Multimodal Functions. IEEE Trans. Evol. Comput. 2019, 23, 718–731. [Google Scholar] [CrossRef]

[37] Cao, Z.; Lin, C.; Zhou, M. A Knowledge-Based Cuckoo Search Algorithm to Schedule a Flexible Job Shop With Sequencing Flexibility. IEEE Trans. Autom. Sci. Eng. 2021, 18, 56–69.

[38] Wang-Xiao Kan, Sun Zhong-Liang, Wnglei and Feng Dong-Qing, 2008. "Design and Research-Based on Fuzzy Self-Tuning PID Using Matlab". Proceedings of International Conference on Advanced Computing Theory and Engineering.

[39] Liu, Fan and Er Meng Joo, 2009. "Design for AutoTuning PID Controller Based on Genetic Algorithms". IEEE Conference on Industrial Electronics and Applications (ICIEA 2009).

[40] Surekha Bhanot, J. Nagrath and M. Gopal, 2002. "Process Control Principles and Applications". Control System Engineering, New AgeInternational Publications, 3 rd edition (2002).

[41] Vaishnav, S.R. and Z.J. Khan, 2007. "Design and Performance of PID andFuzzy Logic Controller with Smaller Rule Set for Higher Order System".WCECS, San Francisco, USA (October 24-26, 2007).

[42] Gade, S.S., S.B. Shendage and M.D. Uplane, 2010. "PerformanceComparison of Online Auto Tune PID Controller with Conventional PIDController". International Journal of Computer Science and Communication, 1(1): 273-277, (January-June 2010).

[43] Astrom, K.J. and T. Haglund, 1995. "PID Controller Theory Design andTuning".Research Triangle Park, NC: Instrum. Soc. Amer, 2 nd edition (1995).

[44] Mann, G.K.I., B.G. Hu and R.G. Gosine, 2001. "Time-Domain BasedDesign and Analysis of New PID Tuning Rules". Proceedings of Inst. Elect. Eng., Control Theory and Applications, 148(3): 251-261.

[45] Pradeep Kumar Juneja, A.K. Ray and R. Mitra, 2010."Various ControllerDesign and Tuning Methods for a First-Order Plus Dead Time Process". International Journal of Computer Science and Communication, 1(2): 161-165 (July-December 2010).

[46] Kalaiselvan, E. and J. Dominic Tagore, 2013. "A Comparative Novel Method of Tuning of Controller for Temperature Process". International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(11).

[47] Kalaiselvan, E. and J. Dominic Tagore, 2013. "A Comparative Novel Method of Tuning of Controller for Temperature Process". International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(11).

[48] Finn Haugen, 2010. "Comparing PI Tuning Methods in a Real Benchmark Temperature Control System". Modeling, Identification and Control, 31(3): 79-91.

[49] Haitham Alasha'ary, Abdullah Al-Hasanat, Khaled Matrouk and Ziad Al-Qadi, 2015, Analysis and Optimization of PID Controller Parameters for a Second-Order Transfer Function, Middle-East Journal of Scientific Research 23 (8).

[50] Hani Attar, Mehrdad Ahmadi Kamarposhti, Ahmed Amin Ahmed Solyman, "Impacts of integration of wind farms on voltage stability margin," International Journal of Electrical and Computer Engineering Vol. 12(No. 5): pp 4623-4631. DOI: 10.11591/ijece.v12i5.pp4623-4631