fractal and fractional

Review

Metaheuristic-Based Algorithms for Optimizing Fractional-Order Controllers—A Recent, Systematic, and Comprehensive Review

Ahmed M. Nassef [1,2,*], Mohammad Ali Abdelkareem [3], Hussein M. Maghrabie [4] and Ahmad Baroutaji [5]

1 Department of Electrical Engineering, College of Engineering in Wadi Alddawasir, Prince Sattam Bin Abdulaziz University, Wadi Alddawasir 11991, Saudi Arabia
2 Computers and Automatic Control Engineering Department, Faculty of Engineering, Tanta University, Tanta 31733, Egypt
3 Department of Sustainable and Renewable Energy Engineering, University of Sharjah, Sharjah P.O. Box 27272, United Arab Emirates; mabdulkareem@sharjah.ac.ae
4 Department of Mechanical Engineering, Faculty of Engineering, South Valley University, Qena 83521, Egypt; hussein_mag@eng.svu.edu.eg
5 School of Engineering and Applied Science, Aston University, Aston Triangle, Birmingham B4 7ET, UK; a.baroutaji@aston.ac.uk
* Correspondence: a.nasef@psau.edu.sa

Abstract: Metaheuristic optimization algorithms (MHA) play a significant role in obtaining the best (optimal) values of the system's parameters to improve its performance. This role is significantly apparent when dealing with systems where the classical analytical methods fail. Fractional-order (FO) systems have not yet shown an easy procedure to deal with the determination of their optimal parameters through traditional methods. In this paper, a recent, systematic. And comprehensive review is presented to highlight the role of MHA in obtaining the best set of gains and orders for FO controllers. The systematic review starts by exploring the most relevant publications related to the MHA and the FO controllers. The study is focused on the most popular controllers such as the FO-PI, FO-PID, FO Type-1 fuzzy-PID, and FO Type-2 fuzzy-PID. The time domain is restricted in the articles published through the last decade (2014:2023) in the most reputed databases such as Scopus, Web of Science, Science Direct, and Google Scholar. The identified number of papers, from the entire databases, has reached 850 articles. A Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) methodology was applied to the initial set of articles to be screened and filtered to end up with a final list that contains 82 articles. Then, a thorough and comprehensive study was applied to the final list. The results showed that Particle Swarm Optimization (PSO) is the most attractive optimizer to the researchers to be used in the optimal parameters identification of the FO controllers as it attains about 25% of the published papers. In addition, the papers that used PSO as an optimizer have gained a high citation number despite the fact that the Chaotic Atom Search Optimization (ChASO) is the highest one, but it is used only once. Furthermore, the Integral of the Time-Weighted Absolute Error (ITAE) is the best nominated cost function. Based on our comprehensive literature review, this appears to be the first review paper that systematically and comprehensively addresses the optimization of the parameters of the fractional-order PI, PID, Type-1, and Type-2 fuzzy controllers with the use of MHAs. Therefore, the work in this paper can be used as a guide for researchers who are interested in working in this field.

Keywords: metaheuristics; fractional-order; optimization; PID controller; fuzzy logic; PRISMA

check for updates

Citation: Nassef, A.M.; Abdelkareem, M.A.; Maghrabie, H.M.; Baroutaji, A. Metaheuristic-Based Algorithms for Optimizing Fractional-Order Controllers—A Recent, Systematic, and Comprehensive Review. *Fractal Fract.* **2023**, *7*, 553. https://doi.org/10.3390/fractalfract7070553

Academic Editors: Xin-Guang Yang, Baowei Feng, Xingjie Yan and Carlo Cattani

Received: 20 June 2023
Revised: 11 July 2023
Accepted: 15 July 2023
Published: 17 July 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The concept of sustainability is central to preserve the natural resources and the environment. This can be accomplished by coinciding with the environment, equity, and economics [1]. Control systems play a crucial role in ensuring that environmental sustainability is maintained over time. Accordingly, control systems are usually developed to efficiently monitor, assess, and regulate the parameters that can influence the environmental system's sustainability. The applications of control systems in environmental sustainability may include waste [2] and pollution [3]; energy consumption [4]; ventilation [5]; carbon emissions [6]; air conditioning [7]; lighting [8]; renewable energy [9]; etc. In addition, control systems have a significant role in ensuring sustainability in industrial processes such as water [10] and power generators [11]. Therefore, by control systems, the use of natural resources can be optimized and hence, environmental sustainability is maintained and protected.

Recently, fractional-order controllers (FOCs) have gained increased interest due to their ability to provide better control performance in comparison to integer order controllers in many practical applications [12]. FOCs have received this significant attention because of their ability to capture the non-linear dynamics of complex systems [13]. Furthermore, they can be used to model complex systems that cannot be modelled using integer order systems [14]. Therefore, FOCs can also be used to improve the performance of existing control systems by providing better stability and faster response times.

There are several common types of FOCs. These types include fractional-order proportional-integral-derivative (FOPID) controllers, fractional-order proportional-integral (FOPI) controllers, fractional-order proportional-derivative (FOPD) controllers, and fractional-order fuzzy PID (FOFPID) controllers [15]. These controllers are applied in a wide range of engineering applications that include temperature control, motion control, process control, etc. [16].

The design of the FOCs is not a trivial task due to the non-linear and complex nature of the fractional-order systems. Therefore, obtaining the optimal FOCs with traditional methods is not easy because of the non-convexity of the optimization problem involved. This requires that the parameters of the fractional-order controllers must be adjusted appropriately in order to improve the dynamics of the system, such as the damping factor, natural frequency, steady-state time, and the system's error [17].

Due to the complexity of tuning the FOCs using the classical methods, modern techniques have been applied to optimize the parameters of FOCs based on various criteria such as stability, robustness, and performance. Metaheuristics techniques are examples of modern optimizers that have proven their efficiency in the optimization of fractional-order controllers. These techniques utilize advanced optimization algorithms that use a random-search strategy in their searching process. Examples of these algorithms are Genetic algorithms (GA) [18], Simulated annealing (SA) [19], Ant colony optimization (ACO) [20], Particle swarm optimization (PSO) [21], Gravitational swarm optimizer (GSO) [22], etc., rather than the classical optimization techniques such as gradient descent (GD) and least squared estimation (LSE). They are particularly useful in tuning fractional-order controllers because these controllers have many parameters that need to be optimized.

In this paper, a systematic as well as comprehensive review on the role of metaheuristic optimization in the optimization of fractional-order controllers is introduced. Two types of fractional-order controllers are introduced in this study: the PID and the fuzzy controllers. During the study of PID controller, the PI controller is also included. Similarly, the fuzzy controllers included both Type-1 and Type-2 fuzzy controllers. This work is prepared to help and guide the researchers who are interested in this field. Therefore, this study focuses on the recent articles that have been published during the last decade only. Throughout an extensive search, around 850 articles were identified from the most popular and scientific databases such as Scopus, Web of Science, Science Direct, and Google Scholar. Then, these articles were filtered and screened based on the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) [23] methodology to exclude the duplicate and irrelevant papers. Finally, around 80 articles were found to be the most relevant and significant articles.

The number of review papers is typically lower in comparison to the regular articles' publications. However, there are some trials that present the use of metaheuristic optimization techniques in optimizing fractional-order controllers. In this regard, Muresan and her colleagues presented a study that reviews the autotuning techniques of the fractional-order PID controllers in industrial processes [24]. They discussed some classical autotuning methods such as the Ziegler-Nichols and the Astrom and Hagglund methods. However, all the presented methods did not include any of the metaheuristic algorithms. Kumarasamy et al. [25] conducted a recent review paper on the speed control of brushless DC motor drive application. Their work presents the optimization of integer- and fraction-order PID controllers for a specific DC motor application. Jamil et al. did a review paper on the use of FOPID controller in a system that uses temperature control [26]. Joseph et al. introduced a review on the MHA approaches in the tuning of FOPID controllers in open problems [27]. Alilou et al. did a comprehensive review of the use of fractional-order controllers in renewable energy and energy-storage-integrated power systems [28]. They presented the most popular algorithms such as Ant Colony Optimization (ACO), PSO, Grey Wolf Optimizer (GWO), etc., that have been used to optimize the FOPID in the application of renewable energy. Reddy and Saha introduced a work that reviews the use of swarm-based optimizers in tuning FOPI and FOPID in the doubly fed induction generator [29]. They showed that the most popular swarm-based algorithms are PSO, GWO, Ant Lion Optimization (ALO), etc.

Based on our comprehensive literature review, this appears to be the first review paper that systematically and comprehensively addresses the optimization of the parameters of the fractional-order PI, PID, Type-1, and Type-2 fuzzy controllers with the use of MHAs. The work in this paper can be used as a systematic guide for researchers that are interested in working in this field. The contributions of this work can be highlighted as follows:

- A systematic review procedure that describes the relationship between the MHA and FOCs optimization is presented;
- The searching keywords are introduced;
- The searching resources included four popular databases such as Scopus, Web of Science, Science Direct and Google Scholar;
- The FOCs include PI, PID, Type-1 and Type-2 fuzzy controllers;
- The search domain is focused on the recent work published only during the last decade (2014:2023);
- The popular objective functions, used in the optimization process, are presented.

The remaining of this paper is organized as follows: An overview on the classical integer-order control systems including the classical PID and fuzzy controllers are presented in Section 2. However, their versions of fractional-order controllers are illustrated in Section 3. Section 4 introduces an outline of the metaheuristic algorithms. The simulation toolboxes are presented in Section 5. The methodology, followed by the results and discussion, are demonstrated in Section 6, respectively. Finally, the conclusions of the work are presented in Section 7.

2. Classical Integer-Order Control Systems

2.1. Classical PID Controller

Classical PID controllers are widely used in industrial control systems due to their simplicity and effectiveness. PID stands for Proportional–Integral–Derivative, which are the three components that make up the controller. The proportional component provides a control output that is proportional to the error between the desired setpoint and the actual process variable. The integral component integrates the error over time, which helps to eliminate steady-state errors. The derivative component provides a control output that is proportional to the rate of change of the error, which helps to improve the response time of the controller [30,31]. In other words, the PID controller takes an action based on the present (proportional), past (integral), and future (derivative) state of the system's error.

The control action and the transfer function of a PID controller as a function of the system's error can be shown as:

$$u(t) = K_P e(t) + K_I \int_0^t e(t)dt + K_D \frac{de(t)}{dt}, \tag{1}$$

$$\frac{U(s)}{E(s)} = K_P + \frac{K_I}{s} + K_D s \tag{2}$$

where $e(t)$ is the system's error; $u(t)$ is the controller output; K_P, K_I, and K_D are the proportional, integral, and derivative parameters (controller gains), respectively.

One of the main advantages of classical PID controllers is their ease of implementation. However, classical PID controllers have some limitations. They are not suitable for systems with large time delays or nonlinearities, and they can be sensitive to changes in the system dynamics [32]. Furthermore, they require manual tuning, which can be time-consuming and may not always result in optimal performance [33].

2.2. Classical Fuzzy Controller

Fuzzy controllers have been applied in several engineering applications [34].

2.2.1. Type-1 Fuzzy Logic Controller (T1FLC)

A new era in science and technology emerged with the invention of fuzzy systems. Credit and gratitude in the field of fuzzy sets go to Professor Lotfi Zadeh, who developed them in the 1960s [35]. Zadeh looked at the sets from a different perspective. Instead of using a sharp discriminator (threshold) to decide whether an instance belongs to a certain set (class) or not, he proposed a more flexible and realistic switching operator by using the concept of degree of membership. In this context, the classical set or the binary set is generalized to form a fuzzy set. The former uses the degree of membership, μ, as a binary number in the set {0, 1} however, the latter considers it as [0 1]. Accordingly, this new strategy proposes that, for any instance in a set, it is possible to accept a partial degree of belonging to a certain set with a membership value between 0 and 1, i e., $0 \leq \mu \leq 1$. This inequality implies that $\mu = 1$ denotes the fully belonging state and that $\mu = 0$ denotes the unbelonging state. These two states are the same as the classical set when using a crisp threshold.

The idea of fuzzy sets has helped create a new science with its own logic, variables, reasoning, mathematics, operators, systems, etc. For example, fuzzy logic uses MIN and MAX operators instead of AND and OR in Boolean logic, respectively. In the following, the structure of the fuzzy system and its main components will be introduced.

Structure of Type-1 Fuzzy Logic Controller

To use fuzzy systems for processing the inputs data until obtaining the final output value, some operations have to be executed. The fuzzy system is composed of three main components: Fuzzifier, Inference Engine, and Defuzzifier, as shown in Figure 1.

Fuzzifier

The first operation in a fuzzy system is the one that is responsible for transforming (mapping) the values of the inputs from its crisp values to fuzzy values. This mapping process is done through a predefined membership function (MF). Usually, the MF is a convex function that maps the inputs to a limited range in the period [0 1]. The input's domain of discourse can be classified into more than one MF depending on the problem that is under investigation. The Gaussian, triangular, and trapezoidal are examples of fuzzy MFs [36].

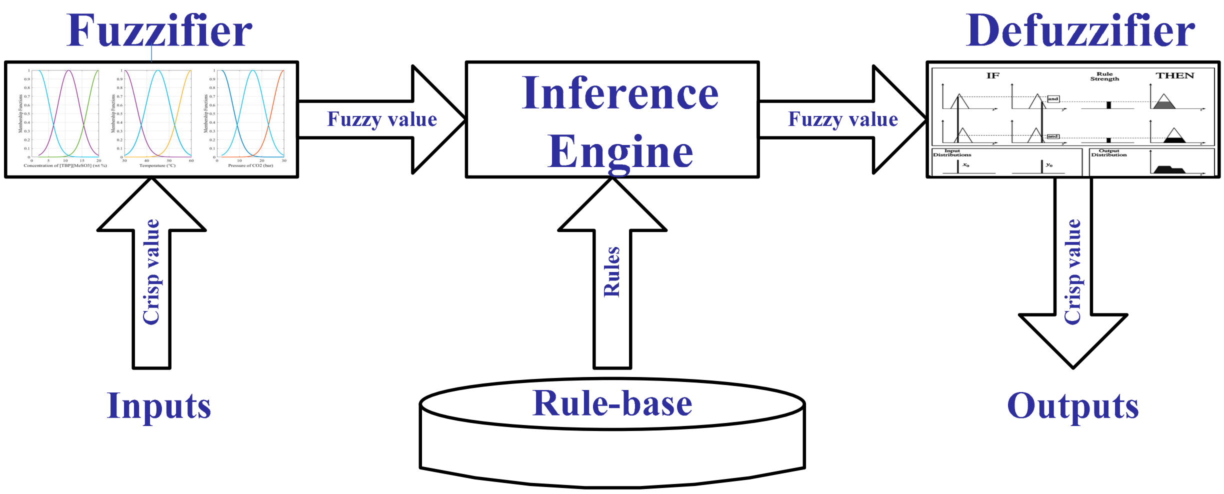

Figure 1. The components of the fuzzy logic controller.

Rule-Base

The rule-base contains the fuzzy rules that govern the relation between the system's outputs and the inputs. Regardless of the type of fuzzy rule, it takes the form of the IF-THEN structure [37]. There are two types of fuzzy rules. However, the choice of either type is problem dependent. The two forms of fuzzy rules are the Mamdani-type [38] and the Sugeno-type [39]. The rule structure, based on the two types, of a two-input single-output system, is as follows:

IF a is α and b is β THEN c is γ, (Mamdani-type)
IF a is α and b is β THEN $c = f(a, b)$, (Sugeno-type)

where a and b are the two inputs of the system; α and β are two arbitrary membership functions that are associated with the inputs a and b, respectively; c is γ are the output and its MF, respectively; $f(a, b)$ is a mathematical linear/nonlinear function of the inputs.

The fuzzy rules are usually generated either based on a systems expert or a clustering algorithm of the dataset.

Inference Engine

In this process, the IF-THEN rules are fired to infer their outputs given a certain set of inputs. Particularly, the output of each rule is obtained using the implication method. There are many techniques to apply the implication method. However, the most widely used is the Min (Intersection) operation [40].

Defuzzifier

As soon as the rules are fired, the entire rules' outputs are aggregated together to end up with a single output value. In Mamdani-type fuzzy rule, the rule's output has a fuzzy value. In this case, the Max (Union) operation is applied. On the other hand, in Sugeno-type fuzzy rule, the weighted average is used to produce the final output [41].

2.2.2. Type-2 Fuzzy Logic Controller (T2FLC)

Type-2 Fuzzy Logic Controller is an extension of the traditional or Type-1 fuzzy logic controller [42]. It was proposed to handle the uncertainties and vagueness that are present in real-world systems [43]. Unlike Type-1 fuzzy sets, which have a single membership function for each input variable, Type-2 fuzzy sets have multiple membership functions at different levels of uncertainty. The principal difference between the Type-2 and Type-1 fuzzy logic controllers lies in the definition of membership functions for a certain variable. Type-2 FLC uses two membership functions to define the fuzzy value of a variable; the upper and lower MFs. Hence, the fuzzy value is represented by an interval instead of a single value. Figure 2a,b show examples of Type-1 and Type-2 fuzzy membership functions, respectively. The MFs in Figure 2 represent the system's error and the change of error, which are the most popular inputs to the fuzzy controller. The regions in-between the upper and lower MFs define the footprint of uncertainty (FOU) [44]. However, Figure 3 shows the components of the Type-2 fuzzy logic controller.

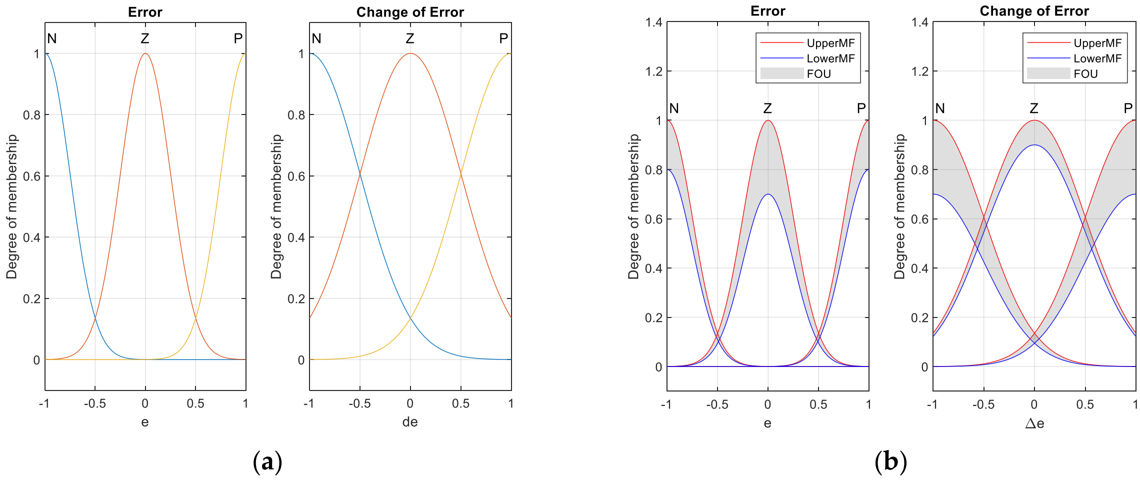

Figure 2. Example of fuzzy membership functions (**a**) Type-1 and (**b**) Type-2.

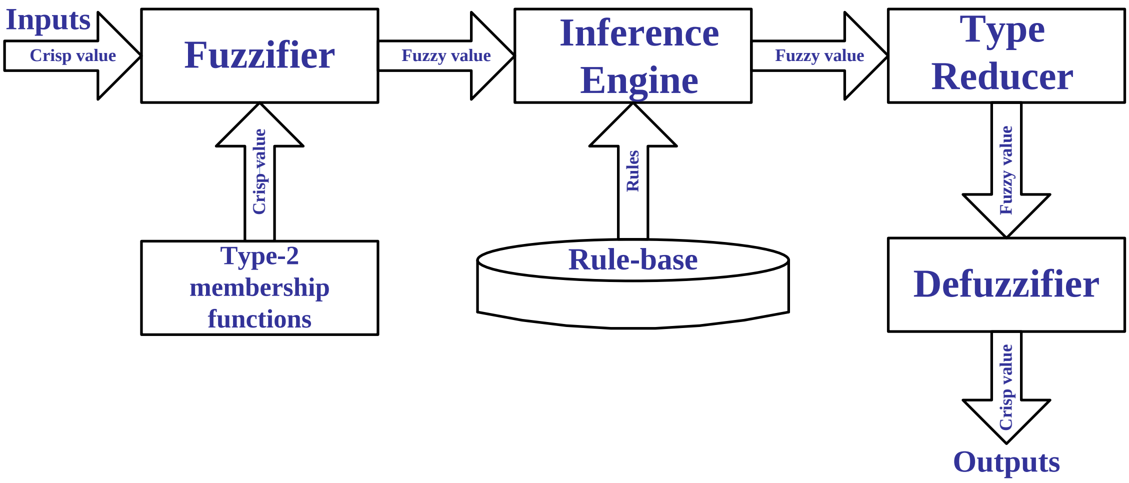

Figure 3. The components of a Type-2 fuzzy system.

The main advantage of using a Type-2 Fuzzy Logic Controller is its ability to handle more complex systems by accounting for higher-order uncertainty [45]. This means that it can perform better when dealing with noisy data or when there is incomplete knowledge about the system being controlled.

However, the downside of using Type-2 Fuzzy Logic controllers is their increased computational complexity and difficulty in tuning due to their additional parameters. In addition, they require more training data than Type-1 controllers because they rely on interval-valued membership functions rather than crisp values. Furthermore, an extra Type-Reducer block is added to the fuzzy processes to perform the Type-2 fuzzy system, as shown in Figure 3. Therefore, the block diagram that represents the PID controller based on the T2FLC is shown in Figure 4.

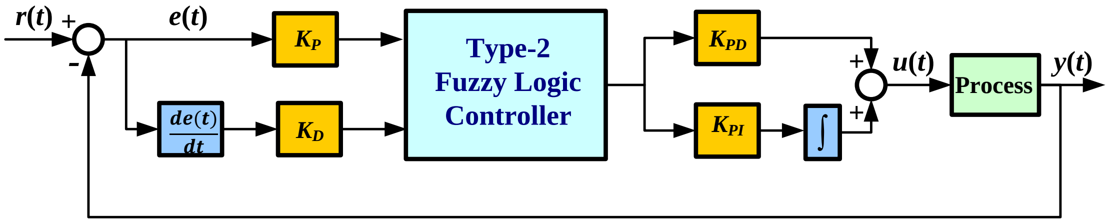

Figure 4. Structure of the integer-order Type-2 fuzzy PID control system.

3. Fractional-Order Systems

The origin of fractional-order derivatives first appeared in the 17th century when the French mathematician Guillaume de L'Hôpital asked the German mathematician Gottfried Leibniz about a particular notation in his publications for the nth-derivative, $\frac{d^n x}{dx^n}$, of the linear function $f(x) = x$. L'Hôpital postulated an amazing question: what will be the derivative of $f(x)$ if $n = 0.5$? Despite the concept of fractional order originating in 1695 [46], the use of fractional-order control systems only started to gain a significant interest in 1945 [47]. Many attempts have been conducted in the direction of finding a concrete solution to the L'Hôpital postulation. Each one of these attempts has its mathematical formulation and definition towards the final solution.

The Grünwald–Letnikov (GL), Riemann–Liouville (RL), and Caputo are the most commonly used definitions of the fractional derivative/integral in fractional calculus. Their mathematical expressions for the fractional-order derivative/integral with a positive/negative fractional number β to a function $f(t)$ can be shown as [48]:

$$ {}_aD_t^{\beta} = \lim_{h\to 0} \frac{1}{h^{\beta}} \sum_{k=0}^{\left[\frac{t-a}{h}\right]} (-1)^k \binom{\beta}{k} f(t-kh) \tag{3}$$

where $\left[\frac{t-a}{h}\right]$ is the integer part; r is the integer that satisfies $r-1 < \beta < r$; $\binom{\beta}{k}$ is the binomial coefficient defined by $\binom{\beta}{k} = \frac{\beta(\beta-1)\ldots(\beta-k+1)}{k!}$.

$$ {}_aD_t^{\beta} f(t) = \frac{1}{\Gamma(r-\beta)} \frac{d^r}{dt^r} \int_b^t \frac{f(\tau)}{f(t-\tau)^{\beta-r+1}} d\tau \tag{4}$$

$$ {}_aD_t^{\beta} f(t) = \frac{1}{\Gamma(\beta-r)} \int_b^t \frac{f^r(\tau)}{f(t-\tau)^{\beta-r+1}} d\tau \tag{5}$$

$\Gamma(\beta - r)$ is the gamma function defined by $\Gamma(z) = \int_0^{\infty} e^{-t} t^{z-1} dt$ and r is the integer that satisfies $r-1 < \beta < r$.

If the fractional-order value is positive, it denotes a differentiation operation. If it is negative, it denotes an integration operation.

The use of fractional calculus has been extended to many applications including the control systems engineering. In fact, fractional-order controllers (FOCs) are being widely used by several scientists in order to reach the most robust performance of the systems. However, the idea of using FOC for the control of dynamical systems belongs to Alain Oustaloup and his colleagues, who developed the CRONE controller in 1991 [49]. CRONE is an abbreviation of "Commande, Robuste d'Ordre Non Entier" which stands for "non-integer order robust control". The pioneer studies in this topic were done by Axtell and Bise [50], Bagley and Calico [51], Makroglou et al. [52], Podlubny [53], Matignon [54], and Matignon and d'Andréa-Novel [55], which provided very interesting ideas for the utilization of fractional derivatives in control theory and fractional-order control systems.

3.1. Fractional-Order PID Controller

Fractional-order PID controllers have gained significant attention in recent years due to their ability to provide better control performance compared to traditional integer-order PID controllers [56]. These controllers are based on fractional calculus, which allows for the use of non-integer orders of differentiation and integration. One of the main advantages of fractional-order PID controllers is their ability to provide better control performance in systems with non-linear dynamics. Traditional integer-order PID controllers are limited in their ability to handle non-linear systems, which can result in poor control performance. Fractional-order PID controllers, on the other hand, can handle non-linear systems more effectively due to their ability to capture the memory effects, which is represented by the order of the derivative of the system [57]. This results in better control performance and improved stability [58]. The FO proportional–integral–derivative (FOPID) controller is a generalized controller of conventional PID. It offers superior response and more stability compared to conventional PID. However, determining the parameters of FOPID is a dilemma [59]. Five parameters need to be identified, instead of only three in the case of conventional PID. Consequently, several tuning approaches have recently been established to determine the parameters of FOPID [59].

FOPID is an application of the utilization of fractional calculus in control systems engineering. In FO, the PID controller's transfer function is updated to include two extra parameters for the derivative and integral terms. These parameters are the μ and λ, which represent the FO derivative and integral, respectively, and can be shown as [60]:

$$\frac{U(s)}{E(s)} = K_P + \frac{K_I}{s^{\lambda}} + K_D s^{\mu} \tag{6}$$

where λ and μ are the two positive real numbers.

3.2. Fractional-Order Fuzzy Controller

The use of fractional-order systems is extended to many engineering fields including automatic control systems. Therefore, many researchers in the control engineering discipline are interested in using FO in their applications [14,15,61,62].

3.2.1. Fractional-Order Type-1 Fuzzy Controller

A block diagram that describes the configuration of a feedback system that uses the Fractional-Order Type-1 Fuzzy (FOT1FPID) as the main controller is shown in Figure 5. It can be seen in Figure 5 that the two inputs to the fuzzy controller are the error and the change-of-error to simulate the PD control action. Then, the output of the controller is integrated to produce the final PID control action. For better performance, all the gains (K_P, K_D, K_{PD}, and K_{PI}) and the FO parameters (λ and μ) have to be adjusted appropriately.

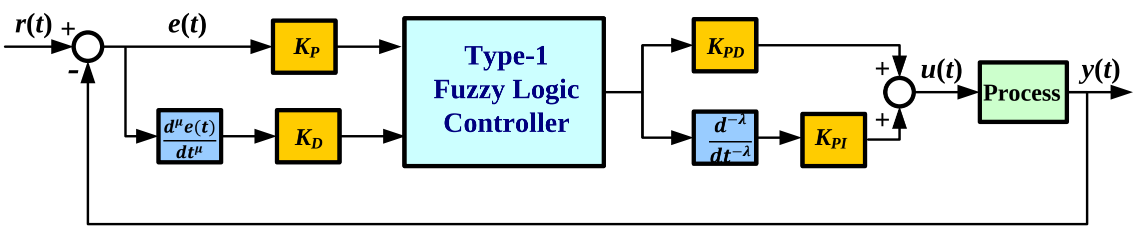

Figure 5. Block diagram of the FOT1FPID controller.

3.2.2. Fractional-Order Type-2 Fuzzy Controller

The block diagram that describes the configuration of a feedback system that uses the Fractional-Order Type-2 Fuzzy (FOT2FPID) as the main controller is shown in Figure 6. In addition, the two inputs to the fuzzy controller are the error and the change-of-error, which is the same as the FOT1FPID. Similarly, all the gains (K_P, K_D, K_{PD}, and K_{PI}) and the FO parameters (λ and μ) have to be adjusted appropriately to obtain a better performance.

It can be seen that the use of FOT2FPID is the same as the FOT21FPID. However, in FOT2FPID, the formulation of the fuzzy membership function is different, as seen in Figure 2. Furthermore, the processing of fuzzy variables is a little bit different due to the Type-Reducer, as illustrated in Figure 3.

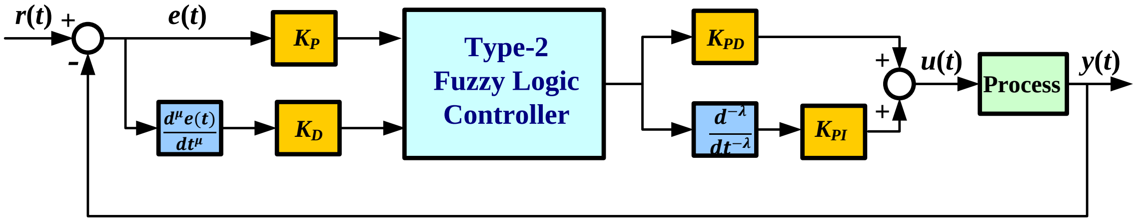

Figure 6. Block diagram of the FOT2FPID controller.

4. Metaheuristic Algorithms (MHA)

Optimization refers to the process of finding the best (maximum/minimum) value from a set of possible solutions to a specific function, with the objective of achieving a specific goal. This goal is to either maximize or minimize the objective function depending on the nature of the problem under investigation. In particular, if the objective function denotes the cost function in this case the target is to minimize this cost. However, if the objective function denotes the benefit function, the goal is to maximize it. In this regard, every function that has peaks or valleys or both can be examined to search for the locations of peaks (maxima) or valleys (minima). The function is classified as unimodal if and only if it has only a single peak or a single valley. However, the function is considered multimodal if it has many peaks and valleys. The highest peak and the lowest valley are known as the global maximum and minimum, respectively. In the same context, the other peaks and valleys are the local maxima and minima, respectively. As most of the optimization algorithms are defined and programmed as minimization problems, in this paper the word "optimization" is meant by "minimization" and vice versa. Therefore, the constrained optimization problem can be formulated as shown below:

$$\underset{X\in\mathbb{R}}{arg\,min}(f(X)) \tag{7}$$

Subject to:

$$g(X) \leq a \tag{8}$$

$$h(X) = b \tag{9}$$

where $f(X)$ is the cost (objective) function; $X = [x_1, x_2, x_3, \ldots, x_n]$ is the vector of controlling inputs of dimension n with $L \leq X \leq U$, where L and U are the lower and upper bounds of the inputs, respectively.

Equations (8) and (9) describe the inequality and equality constraints of the optimization problem, respectively; a and b are constants.

It is worth mentioning that the same form of Equation (7) can be adopted for the case of maximization problems by simply multiplying the cost function by -1. Consequently, the optimization problem can take the form shown below:

$$\underset{X\in\mathbb{R}}{arg\,min}(-f(X)) \tag{10}$$

There are many techniques and procedures to apply the optimization process. The classical one is to use analytical methods that are based on differential calculus such as the gradient descent (GD) procedure which is useful for continuous and differentiable functions [63]. Unfortunately, this procedure fails when the mathematical expression of the function (model) to be optimized does not exist specifically for black-box models.

Furthermore, the main drawback of using the classical GD technique is the possibility of falling into local minima, which is known as the premature convergence problem. This problem occurs as the solution obtained by the GD is sensitive to the initial values of the proposed solutions. On the other hand, modern optimization techniques such as metaheuristic algorithms (MHAs) can tackle these kinds of problems by using a controlled stochastic search instead. These types of searching algorithms are controlled via tuning some parameters to guarantee and accomplish the two targets of the optimization process, which are the accuracy and speed of obtaining the optimal solution. MHA refers to an iterative procedure to develop an optimization algorithm for a high-level problem-independent framework [64,65].

The No-free-Lunch (NFL) principle opens a significant window for researchers to be able to propose new optimizers. Based on this principle, the route for developing new algorithms is unlimited [66,67]. Accordingly, this endless path allows many new optimizers to be introduced every year.

In this review paper, a spotlight has been focused on the relationship between the MHA and the fractional-order controllers' design which can be taken as a guide to the researchers in the engineering sector.

4.1. History of MHA

In fact, many developments have been established since metaheuristics have been first appeared to improve the entire performance. These improvements include either the development of new algorithms or the enhancements of already established algorithms. The timeline figure that describes the historical development of metaheuristic algorithms is illustrated in Figure 7 [68]. The timeline of the progress of metaheuristics is illustrated in Table [64]. It is worth mentioning that as of 2022, over 540 optimizers have been developed, each with different classifications [69]. Table 1 presents a detailed description of the metaheuristic algorithms and their historical timeline developments.

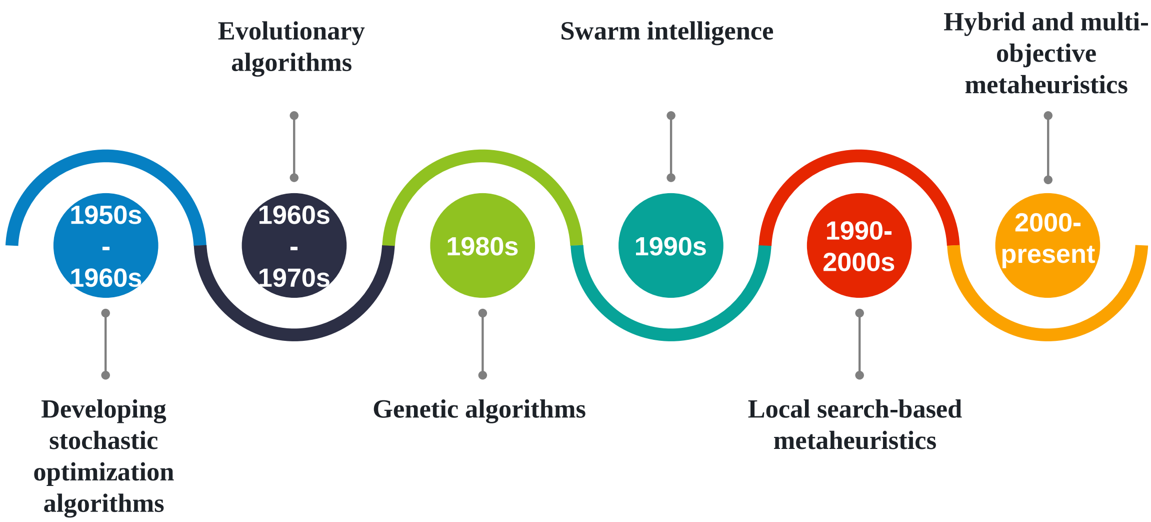

Figure 7. A timeline describing the historical developments of optimization algorithms [68].

Table 1. The detailed description of the metaheuristic algorithms and their timeline.

Timeline	Description
1953s	George Dantzig and others develop the simplex algorithm, one of the first optimization algorithms for linear programming.
1950s–1960s	Researchers begin developing stochastic optimization algorithms, such as Monte Carlo methods and simulated annealing, that use probabilistic techniques to find good solutions.
1960s–1970s	Evolutionary algorithms, inspired by the principles of natural selection, are developed by researchers such as John Holland and Ingo Rechenberg.
1980s	Genetic algorithms, a type of evolutionary algorithm, gain popularity as a means of solving optimization problems.
1990s	Swarm intelligence approaches, such as particle swarm optimization and ant colony optimization, emerge as a new class of metaheuristics that are based on the collective behaviour of simple agents.
1990s–2000s	Local search-based metaheuristics, such as tabu search and variable neighbourhood search, become popular for solving combinatorial optimization problems.
2000s–present	Hybrid and multi-objective metaheuristics gain popularity, as researchers seek to combine different classes of metaheuristics or combine metaheuristics with other optimization techniques to improve their performance on a wide range of problems.

4.2. Classifications of MHA

There are several types of metaheuristic algorithms, each of which has its own strengths and weaknesses. One type of metaheuristic algorithm is the evolutionary algorithms, which is based on the principles of natural selection and genetic recombination. Another type of metaheuristic algorithm is swarm intelligence, which is inspired by the collective behaviour of groups of simple agents. Examples of swarm intelligence algorithms include particle swarm optimization and ant colony optimization. Local search-based metaheuristics, such as tabu search and variable neighbourhood search, are another type of metaheuristic algorithm that iteratively improves a solution by exploring its neighbourhood. Finally, hybrid metaheuristics combine different types of metaheuristic algorithms or combine metaheuristics with other optimization techniques to improve their performance on specific types of problems. Each type of metaheuristic algorithm has its own strengths and weaknesses and can be applied to a wide range of optimization problems. In addition, the MHs can be classified according to the nature of obtaining the final solution either through a single or population-based strategies. In the single-solution-based strategy, the solution is randomly generated and improved iteratively until the final best result is obtained. On the other hand, the population-based solution strategy uses a set of initial random solutions. These solutions are hence updated iteratively until the optimal solution is reached among the entire set [68]. Examples of single solution-based algorithms include Tabu search, Simulated annealing, Local search, and Iterated local search. However, both the Evolutionary algorithms (EA) and the swarm intelligence (SI) algorithms are classified as population-based algorithms. Examples of EA algorithms include the GA and DE. However, the PSO, Firefly, Ant colony, Bee colony, and Bat algorithm are examples of SI algorithms. Therefore, MHA are classified into several categories [70,71]. This is because many criteria are applied. However, the popular classification criteria are based on the nature of inspiration, the population size and the updating mechanism. Figure 8 shows the popular taxonomies of metaheuristic algorithms. The most popular and widely used algorithms are of the EA and SI types. Therefore, the following section will provide a brief introduction to the SI class, which is the most popular one, along with examples.

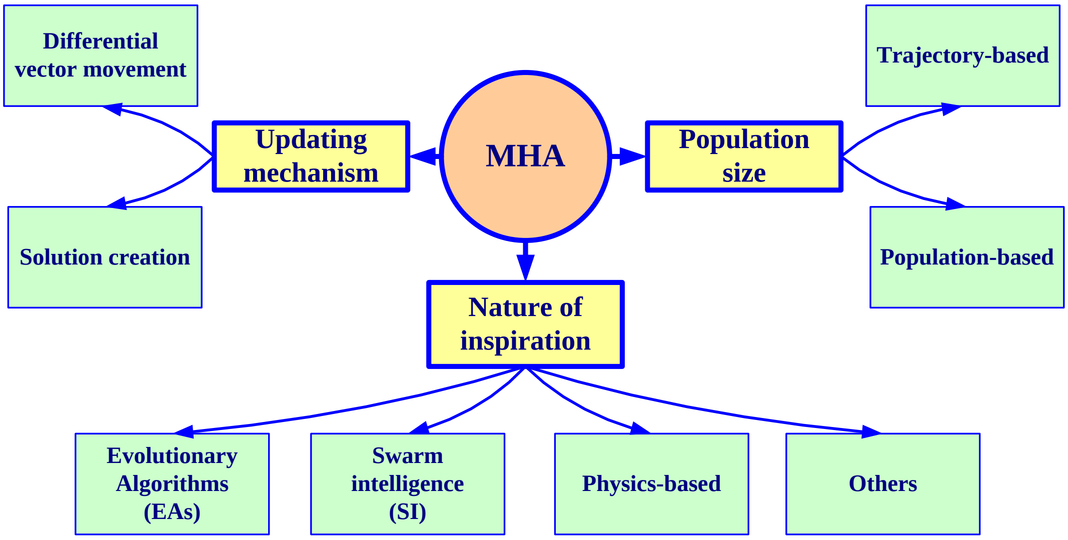

Figure 8. Popular taxonomies of metaheuristic algorithms.

4.2.1. Swarm-Based Algorithms

Swarm-based optimization algorithms are a group of algorithms that are inspired by the collective behaviour of animals or insects that exhibit swarm intelligence. These algorithms have gained a lot of interest in the field of optimization since they are capable of finding near-optimal solutions quickly. The basic idea behind these algorithms is to simulate the behaviour of the dynamic behaviour of a group of particles that interact with each other as they search for an optimal solution.

Common examples of swarm-based optimization algorithms include Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Artificial Bee Colony (ABC).

Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a population-based stochastic optimization technique that mimics the behaviours of birds flocking or fish schooling. In this algorithm, a group of particles move around in the search space, and each particle adjusts its direction and speed based on the positions of particles in its neighbourhood [72].

Particle Swarm Optimization (PSO) is a variant of the SI algorithms that mathematically simulates the movement of a flock of birds. The main idea of the algorithm is to generate a list of candidate solutions and aim for one of these solutions to reach the global best (minimum/maximum) position within the search-space. In PSO, these solutions are defined as particles and the best solution is the one that has grasped the best position [73]. Further descriptions about the PSO theory and its application can be found in Refs. [21,73]. The optimization procedure starts by setting the number of particles, N; maximum number of iterations, M; and the upper and lower values of the controlling variables, U and L, respectively. The initial N solutions are generated according to Equation (11).

$$p_{i,d} = L_{i,d} + r_{i,d}(U_{i,d} - L_{i,d}) \tag{11}$$

where $p_{i,d}$, $L_{i,d}$, $r_{i,d}$, and $U_{i,d}$ are the location of particle, lower value, random variable and upper value of the ith particle in the dth dimension.

Then, the particles' locations are updated iteratively. The updating rule, related to the ith particle in the swarm in an iteration t, for the velocity v_i^{t+1} and position p_i^{t+1} can be shown as [74,75]:

$$p_i^{t+1} = p_i^t + v_i^{t+1} \tag{12}$$

$$v_i^{t+1} = w \times v_i^{t+1} + C_1 \times R_1 \times \left(p_{pBest}^t - p_i^t\right) + C_2 \times R_2 \times \left(p_{gBest}^t - p_i^t\right) \tag{13}$$

The weight of inertia is denoted by w, which varies iteratively from 1 to 0.4 throughout the optimization process. However, the p_{pBest}^{t} and p_{gBest}^{t} represent the local best and the global best values, respectively. The local experience weight and the global experience weight are denoted by the two constants C_1 and C_2, respectively. R_1 and R_2 are two stochastic variables that vary in the range [0 1]. The flowchart describing the updating mechanism of the PSO is illustrated in Figure 9.

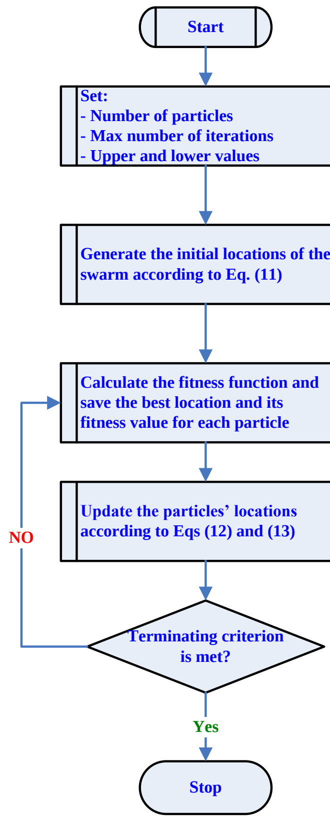

Figure 9. Flowchart describing the PSO updating mechanism.

4.3. Objective Function

The most popular performance indices used as objective functions are the Integral Absolute Error (IAE), Integral Squared Error (ISE), Integral of the Time-Weighted Absolute Error (ITAE), Integral of Time multiplied by the Squared Error (ITSE). All of the above functions are required to be minimized to obtain the best performance for the system. In other words, to minimize the error signal throughout the entire simulation time, t_s, to minimize the overshoots and settling time accordingly. The formulas of *IAE*, *ISE*, *ITAE*, and *ITSE* can be shown as:

$$ISE = \int_0^{t_s} (e(t))^2 dt, \tag{14}$$

$$IAE = \int_0^{t_s} |e(t)| dt, \tag{15}$$

$$ITAE = \int_0^{t_s} t|e(t)| dt, \tag{16}$$

$$ITSE = \int_0^{t_s} t(e(t))^2 dt. \tag{17}$$

4.4. Simulation Toolboxes

The optimal parameters of the controller can be obtained by applying an appropriate optimizer. To accomplish this task, a simulation toolbox must be used. An example of the block diagram that illustrates the configuration of the optimizer and the T2FOFPID system, in an optimization process, is presented in Figure 10. The figure illustrates that the input to the optimizer is the system's error, $e(t)$ and its output is the set of optimal parameters of the controller, i.e., the gains and the fractional-orders. The optimizer uses the error and computes one of the considered objective functions shown in Equations (14)–(17).

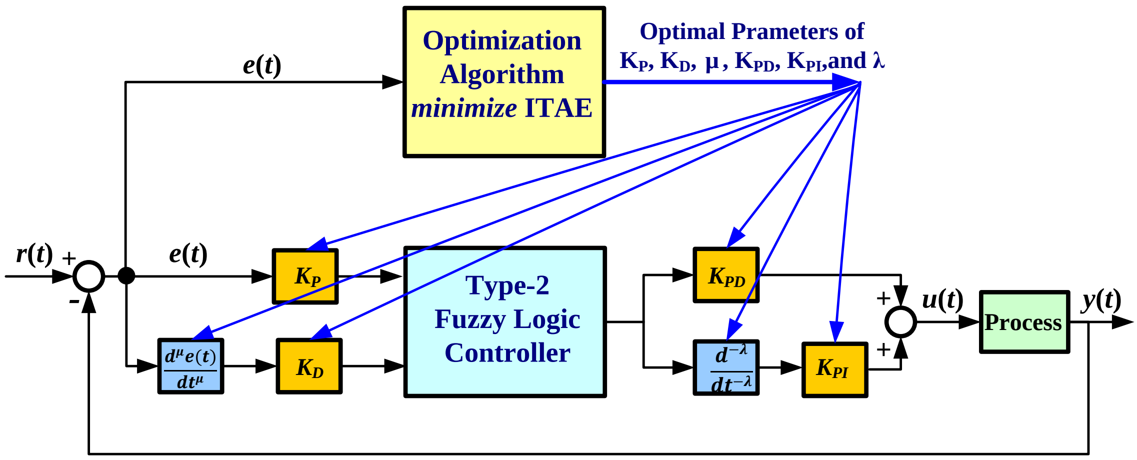

Figure 10. The block diagram representing the optimization process of T2FOFPID.

When simulating a FOPID controller, several analytical tools can be used. The popular tools are commonly utilized within the MATLAB software environment.

4.4.1. FOMCON Toolbox

In 2013, Tepljakov et al. implemented the Fractional-order Modelling and Control toolbox, FOMCON, which worked under the framework of MATLAB to handle FOPID controllers [76]. The link to download FOMCON is provided below [77]: https://www.mathworks.com/matlabcentral/fileexchange/66323-fomcon-toolbox-for-matlab?s_tid=srchtitle (accessed on 1 June 2023).

4.4.2. FOTF

Xue et al. developed the fractional-order transfer function (FOTF) [78]. The FOPID controller block can work in the Simulink toolbox and is able to use the optimization method to tune the controller. The code for FOTF can be obtained from Ref. [79]. https://www.mathworks.com/matlabcentral/fileexchange/60874-fotf-toolbox (accessed on 1 June 2023).

4.4.3. Ninteger Toolbox

In 2005, Valerio and Costa developed the script of Ninteger MATLAB code [80]. The source code is available and can be downloaded from the MathWorks website from the following [81]: https://www.mathworks.com/matlabcentral/fileexchange/8312-ninteger?s_tid=srchtitle (accessed on 1 June 2023).

4.4.4. CRONE

Oustaloup et al. [49] developed the CRONE toolbox for engineers and researchers to deal with FO systems. The link to download the CRONE software can be shown below (***Note: Registration is required with a professional email address***) [82]: http://archive.ims-bordeaux.fr/CRONE/toolbox/pages/accueilSITE.php?guidPage=home_page (accessed on 1 June 2023).

5. Methodology

To accomplish the target of this review paper, a systematic methodology was first applied by collecting the related papers from the popular databases according to the items shown in Table 2.

Table 2. Keywords used in the search process.

Item	Description
Main searching keywords	Factional order; PID; Fuzzy; Controller; Optimization
Date Range	2014–2023
Databases	Scopus (SC); Web of Science (WOS); Science Direct (SD); Google Scholar (GS)

Using the data listed in Table 2, a searching criterion was applied to every database according to its appropriate searching formula. Synonyms for each keyword were employed using the OR logical operators, as shown in Table 3. The searching formulas used for each database are presented in the following subsections.

Table 3. The keywords and their synonyms.

Keyword	Synonyms
Factional order	FO OR fractional-order
Optimization	optimal OR swarm OR tuning OR metaheuristic

5.1. Scopus

The searching criterion used with the Scopus database is as follows:

"TITLE (((PI OR PID OR fuzzy) AND controller) AND ("factional order" OR "fractional-order") AND (optimization OR optimal OR swarm OR tuning OR metaheuristic)) AND (LIMIT-TO (PUBYEAR, 2023) OR LIMIT-TO (PUBYEAR, 2022) OR LIMIT-TO (PUB/YEAR, 2021) OR LIMIT-TO (PUBYEAR, 2020) OR LIMIT-TO (PUBYEAR, 2019) OR LIMIT-TO (PUBYEAR, 2018) OR LIMIT-TO (PUBYEAR, 2017) OR LIMIT-TO (PUBYEAR, 2014))".

A screenshot of the search criterion of fractional-order controllers on the Scopus database is shown in Figure 11.

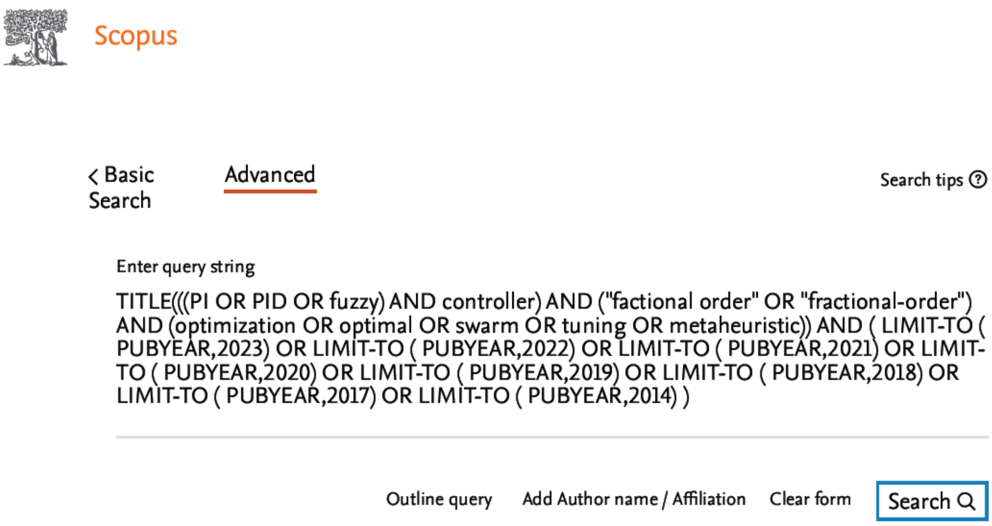

Figure 11. A screenshot of the search criterion of fractional-order controllers on the Scopus database.

5.2. Web of Science (WOS)

A screenshot of the search criterion of fractional-order controllers on the WOS is shown in Figure 12.

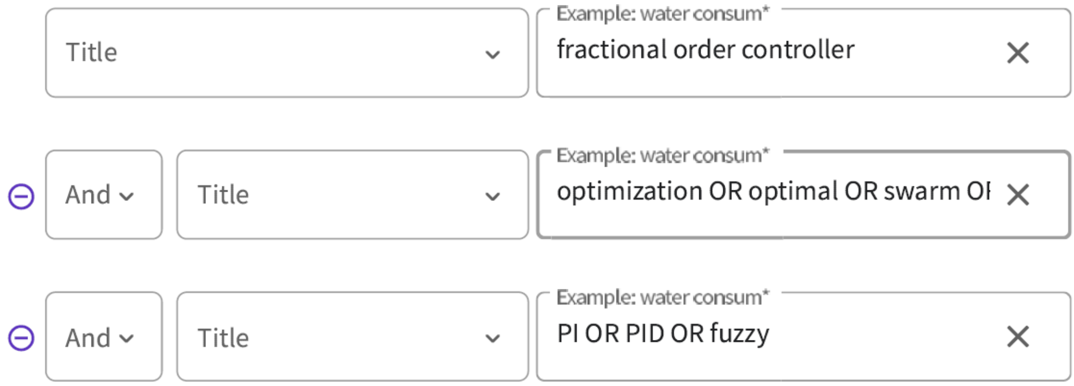

Figure 12. A screenshot of the search criterion of fractional-order controllers on the WOS.

5.3. Science Direct

The following searching criterion was used with the Science Direct (SD) database:

"(PI OR PID OR fuzzy) AND controller AND ("factional order" OR "fractional-order") AND (optimization OR optimal OR swarm OR tuning OR metaheuristic)".

5.4. Google Scholar

The following searching criterion was applied in the title field on the Google Scholar (GS) database:

"allintitle: PID controller optimization OR optimal OR swarm OR metaheuristic OR tuning "fractional-order". However, the search procedure was repeated for each controller type; PI, PID, and fuzzy. A screenshot of the search criterion of PID controller on the Google Scholar is shown in Figure 13.

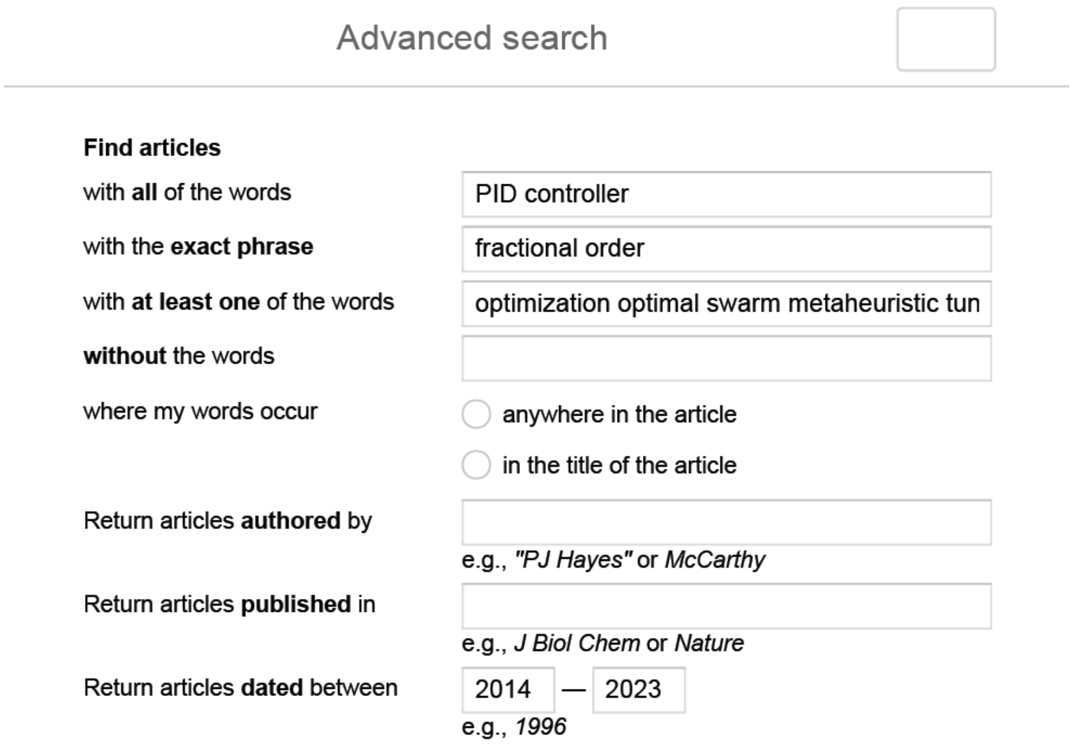

Figure 13. A screenshot of the search criterion of the PID controller on Google Scholar.

6. Results and Discussion

The searching criteria, mentioned in the previous section, have been applied and the numbers of papers that satisfied the searching criterion for each controller and obtained from each database are listed in Table 4. However, the total size of identified papers is 858 articles. A PRISMA strategy was adopted in this study [23]. Accordingly, the obtained 858 articles have been filtered to remove the redundant and duplicate papers, as shown in Table 5.

Table 4. Total number of articles extracted from the popular databases related to FO controllers.

	Database Source				
Controller Type	**Scopus**	**Web of Science**	**Google Scholar**	**Science Direct**	**Total**
FO-PI	32	44	35	12	**123**
FO-PID	158	165	181	42	**546**
FO-Fuzzy	83	38	46	22	**189**
Total	**273**	**247**	**262**	**76**	**858**

Table 5. Total number of duplicate articles extracted from the popular databases related to FO controllers.

Controller Type	Initial	Duplicates	Remaining
FO-PI	123	49	**74**
FO-PID	546	241	**305**
FO-Fuzzy	189	80	**109**
Total	**858**	**370**	**488**

Afterward, the screening (inclusion) criterion has been applied. This criterion is as follows: the article is included in the review list if it is recent, relevant, and cited more than once a year. Accordingly, Table 6 shows the number of articles included in the current review after excluding the irrelevant articles and the articles with very less citations.

Table 6. Total number of excluded articles extracted from the popular databases related to FO controllers.

Controller Type	Initial	Irrelevant Excluded	Included
FO-PI	74	58	**16**
FO-PID	305	257	**48**
FO-Fuzzy	109	91	**18**
Total	**488**	**406**	**82**

Figure 14 shows the PRISMA flowchart [23] of the identification, screening, and inclusion processes. As shown in Tables 4–6, FO-PID has the highest number of published articles, appearing to have attracted the interest of many researchers. The main reason is that the FO-PID produced better performance in comparison to the FO-PI because the derivative term is taken into consideration for generating the control action. The derivative gain is useful, especially for the systems that suffer from a high percentage overshoot. Furthermore, the FO-PID has a smaller number of tuning parameters (only five) relative to the FO-fuzzy controller, as the latter needs at least six parameters to be tuned.

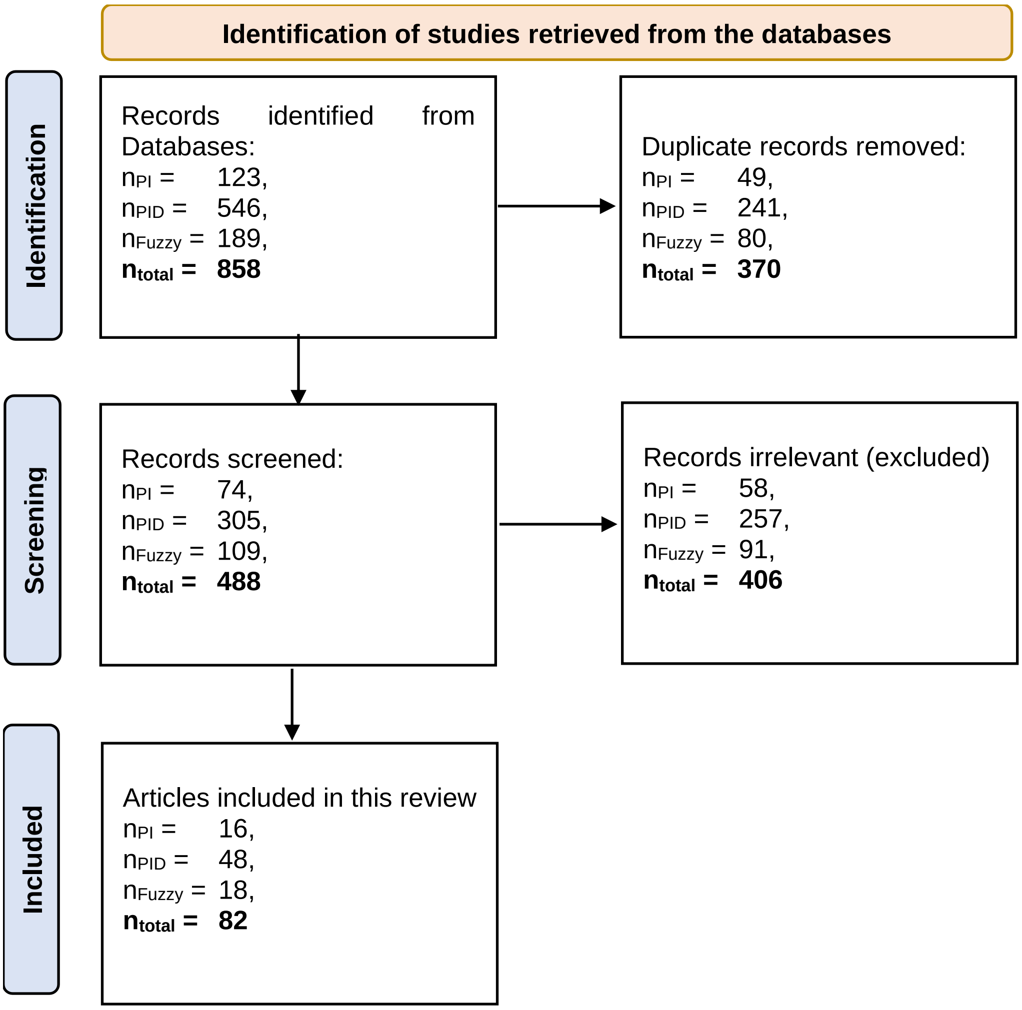

Figure 14. PRISMA flowchart for the current systematic review [23].

It can be seen from the final list of papers that 29 optimizers have been unitized throughout the last decade to optimize the FO controllers. The plot in Figure 15 shows the number of published papers, throughout the last decade, for the first 16 optimizers. For the remaining 13 optimizers, each was applied only once, so it is omitted from the plot. All variants of every single optimizer are considered the same. The list of the addressed optimizers is shown in Table 7 and the optimizers with less usage and citations have been skipped from this list. From the table, it can be noticed that the PSO is the one that gained more interest than the others, as it is applied in almost 25% of the published articles with 20 research papers out of 82.

The data of highly cited as well as recent papers that have been used the MHA in the optimization of fractional-order PI, PID, and fuzzy controllers is listed in Tables 8–10, respectively. By examining the data presented in the three tables, it can be noticed that the FO-PID controller has garnered the highest level interest among researchers compared to the others. This can be observed based on the large number of published papers, with 48 articles utilizing FO-PID as opposed to only 16 and 18 for the FO-PI and FO-fuzzy controllers, respectively. Moreover, the work with FO-PID has high citation records in comparison to the FO-PI and FO-fuzzy controller. This can be seen from the first records of "Cited by" columns in Tables 8–10. The work with FO-PI controller was cited by a maximum of 64 articles. Additionally, the work with FO-fuzzy controller was cited by a maximum of 23 articles. On the other hand, the work with FO-PID controller got 196 citation records.

Table 7. A list of algorithms used in the recent applications for the optimization of fractional-order controllers.

No.	Author	Optimizer	Abbreviation	Year	Inspiration	DOI
1	Dorigo et al. [20]	Ant Colony Optimization	ACO	2006	Ant foraging behaviour	https://doi.org/10.1109/MCI.2006.329691
2	Mirjalili [83]	Ant Lion Optimization	ALO	1992	Ant lion hunting mechanism	https://doi.org/10.1016/j.advengsoft.2015.01.010
3	Yang and Gandomi [84]	Bat Algorithm	BA	2012	Echolocation behaviour of bats	https://doi.org/10.1108/02644401211235834
4	Teodorović [85]	Bee Colony Algorithm	BCA	2009	The behaviour of bees in nature	https://doi.org/10.1007/978-3-642-04225-6_3
5	Talatahari and Azizi [86]	Chaos Game Optimization	CGO	2020	Configuration of fractals by chaos game	https://doi.org/10.1007/s10462-020-09867-w
6	Baran Hekimoğlu [87]	Chaotic Atom Search Optimization	ChASO	2019	Atomic motion in nature	https://doi.org/10.1109/ACCESS.2019.2905961
7	Ibrahim et al. [88]	Chaotic Harris Hawks Optimization	CHHO	2020	Hunting strategy of Harris's hawks	https://doi.org/10.1109/ICENCO49778.2020.9357403
8	Gandomi et al. [89]	Cuckoo Search Algorithm	CS	2013	Breed behaviour of cuckoo species	https://doi.org/10.1007/s00366-011-0241-y
9	Abedinpourshotorban et al. [90]	Electromagnetic Field Optimization	EFO	2016	Behaviour of electromagnets with different polarities	https://doi.org/10.1016/j.swevo.2015.07.002
10	Faramarzi et al. [91]	Equilibrium Optimizer	EO	2020	Mass balance models	https://doi.org/10.1016/j.knosys.2019.105190
11	Yang [92]	Flower Pollination Algorithm	FPA	2012	Pollination process of flowers	https://doi.org/10.1007/978-3-642-32894-7_27
12	Genetic Algorithm [18]	Genetic Algorithm	GA	1975	Evolution and natural selection theory	https://www.jstor.org/stable/24939139
13	Saremi et al. [93]	Grasshopper Optimization Algorithm	GOA	2017	Food source seeking by grasshopper swarms	https://doi.org/10.1016/j.advengsoft.2017.01.004
14	Mirjalili et al. [94]	Grey Wolf Optimizer	GWO	2014	Hunting behaviour of grey wolf packs	https://doi.org/10.1016/j.advengsoft.2013.12.007
15	Heidari e. al. [95]	Harris Hawks Optimization	HHO	2019	Cooperative behaviour and chasing style of Harris's hawks	https://doi.org/10.1016/j.future.2019.02.028
16	Geem and Kim [96]	Harmony Search	HS	2001	Artificial phenomenon in musical performance	https://doi.org/10.1177/003754970107600201

Table 7. *Cont.*

No.	Author	Optimizer	Abbreviation	Year	Inspiration	DOI
17	Madadi et al. [97]	Improved Moth Swarm Algorithm	IMSA	2020	Behaviour of moths in nature	https://doi.org/10.1016/j.envpol.2020.114258
18	Mirjalili [98]	Moth Flame Optimization	MFO	2015	Navigation method of moths in nature	https://doi.org/10.1016/j.knosys.2015.07.006
19	Abdel-Basset et al. [99]	Modified Flower Pollination Algorithm	MFPA	2021	Pollination process of flowers	https://doi.org/10.3390/math9141661
20	Ouaar and Boudjemaa [100]	Modified Salp Swarm Algorithm	MSSA	2021	Swimming and foraging behaviour of salps in oceans	https://doi.org/10.1007/s00521-020-05621-z
21	Sulaiman et al. [101]	Plant Propagation Algorithm	PPA	2016	Propagation of strawberry plant	Sci. Int. (Lahore), 28(1), 201–209, 2016
22	Eberhart and Kennedy [21]	Particle Swarm Optimization	PSO	1995	Social movement of flock of birds	https://doi.org/10.1109/ICNN.1995.488968
23	Mirjalili [102]	Sine Cosine Algorithm	SCA	2016	Periodical oscillations of the sine/cosine functions	https://doi.org/10.1016/j.knosys.2015.12.022
24	Gomes et al. [103]	Sunflower Optimization	SFO	2019	Motion of sunflowers in capturing solar radiation	https://doi.org/10.1007/s00366-018-0620-8
25	Satapathy and Naik [104]	Social Group Optimization	SGO	2016	Behaviour of humans in solving complex problems	https://doi.org/10.1007/s40747-016-0022-8
26	Dhiman and Kumar [105]	Seagull Optimization Algorithm	SOA	2018	Migration and attacking behaviours of a seagull	https://doi.org/10.1016/j.knosys.2018.11.024
27	Seyedali Mirjalili [106]	Salp Swarm Algorithm	SSA	2017	Swarming behaviour of salps (salp chain)	https://doi.org/10.1016/j.advengsoft.2017.07.002
28	Yu and Li [107]	Social Spider Algorithm	SSA	2015	Foraging strategy of the social spider	https://doi.org/10.1016/j.asoc.2015.02.014
29	Eskandar et al. [108]	Water Cycle Algorithm	WCA	2012	Movements of rivers and streams	https://doi.org/10.1016/j.compstruc.2012.07.010

Table 8. Data of highly cited and recent papers that used MHA in the optimization of fractional-order PI controllers.

No.	Ref	* Cited by	Year	Optimizer	# of Optimal Variables	Objective Function	Value	System	Controlled Variable
1	Maroufi et al. [109]	64	2019	BA	-	IAE	6.89	MPPT-Pitch control of a Wind Turbine	Pitch angle
2	Guha et al. [110]	63	2020	GOA	6	ITAE	-	Load frequency control	Frequency
3	Barakat [111]	13	2022	CGO	3	ITAE	-	Interconnected power systems	Frequency
4	Ramadan et al. [112]	12	2022	MFPA	3	Absolute of steady-state error	0.015	On-grid Fuel Cell	Overshoot Settling time Execution time
5	Kakkar et al. [113]	10	2021	WCA	3	ITAE, absolute steady-state error, and settling time. A multiobjective external optimization (MOEO) technique (FOMCON toolbox for MATLAB)	-	Grid-connected PWM Rectifiers	-
6	Leena et al. [114]	6	2018	SGO	3	ISE	-	SISO process	Response
7	Maamir et al. [109]	4	2015	PSO	-	-	-	Control thermal device (heat pump)	Temperature
8	Agarwal et al. [115]	3	2015	PSO	3	ITAE	0.982009	Speed control of DC motor	Speed
9	Bouderres et al. [116]	2	2022	PSO	-	-	-	Grid-connected photovoltaic system	Voltage
10	Zamee and Won [117]	2	2019	PPA	6	ITAE	0.2918	Grid-connected single-stage three-phase solar photovoltaic system	Dc link voltage
11	Hameed and Ramasubramanian [118]	1	2020	(ALO), (MFO), (WOA) (SCA)	Case 1: 6 Case 2: 12 Case 3: 12 Case 4: 12	ITAE	-	Multi-area thermal power system	Deviation in frequency
12	Özyetkin and Birdane [119]	0	2023	PSO	-	-	-	Numerical example	Response

Table 8. *Cont.*

No.	Ref	* Cited by	Year	Optimizer	# of Optimal Variables	Objective Function	Value	System	Controlled Variable
13	Altawil et al. [120]	0	2023	SSO	-	ITAE	-	Photovoltaic system	Voltage
14	Labed et al. [121]	0	2022	PSO	-	-	-	Wind turbine system	Output
15	Mehmed and Abdullah [122]	0	2022	PSO	-	Total harmonic distortion	1.65%	Bidirectional three-phase dc-ac power inverter	Output
16	Dwivedi et al. [123]	0	2022	WCA	-	(MATLAB–Simulink)	-	Three-phase EV charger	Voltage

* The number of citations listed in this table was last updated on 5 June 2023.

Table 9. Data of highly cited and recent papers that used MHA in the optimization of fractional-order PID controllers.

No.	Ref	* Cited by	Year	Optimizer	# of Optimal Variables	Objective Function	Value	Plant	Controlled Variable
1	Hekimoğlu [87]	196	2019	ChASO	-	ITSE	5.4978×10^{-5}	Dc motor	Speed
2	Aghababa [124]	91	2015	Modified PSO	-	IAE ISE ITSE	-	Five-bar linkage robot	Dynamic response
3	Vanchinathan and Valluvan [125]	75	2018	BA	-	-	-	Sensor-less BLDC Motor	Speed
4	Khan et al. [126]	70	2019	SSO	-	ITAE	0.004195	Automatic voltage regulator system	Voltage
5	Suri babu and Chiranjeevi [127]	68	2016	GA ACO	5	ITAE	-	Automatic voltage regulator (AVR) system	Voltage
6	Vanchinathan and Selvaganesan [128]	65	2021	ABC	-	IAE ITAE ISE	0.008 0.002 0.008	Brushless DC motor	Speed
7	Guhaet al. [129]	49	2020	EO	-	ISE	0.70112×10^{-4}	Load frequency control of power system	Frequency
8	Bouakkaz et al. [130]	46	2020	PSO	5	ISE	-	PV energy generation systems	Cell's output power
9	Ghamar et al. [131]	26	2021	ALO	5	-	-	Buck converter	Voltage
10	Jaiswal et al. [132]	13	2020	GA	5	IAE	0.098	Conical tank (nonlinear) system	Magnitude

* The number of citations listed in this table was last updated on 5 June 2023.

Table 10. Data of highly cited and recent papers that used MHA in the optimization of fractional-order fuzzy controllers.

No.	Ref	* Cited by	Fuzzy Type	Year	Optimizer	# of Optimal Variables	Objective Function	Value	Plant	Controlled Variable
1	Karahan [133]	23	Type-1	2021	CS	-	ITSE	0.000052	Molten salt reactors	Power control
2	Debidasi and Panda [134]	21	Type-1	2021	MSSA	-	Frequency variation plus control signal output	4.0846	Hybrid Power System with Electric Vehicle	Frequency
3	Nayak et al. [135]	14	Type-1	2021	SFO	8	ITSE	0.0091	Solar-wind integrated power system with hydrogen aqua equalizer-fuel cell unit	Frequency
4	Prusty et al. [136]	8	Type-2	2022	IMSA	8	ISE	0.0224	Microgrid	Frequency
5	Patel and Shah [137]	7	Type-2	2022	FPA	6	close loop error	6.723×10^{-1}	Nonlinear uncertain level control systems	Response
6	Bennaoui et al. [138]	7	Type-1	2020	MFO	5	IAE	0.1691	Dc-dc boost converter	Voltage
7	Kalyan [139]	5	Type-1	2022	SOA	5	ISE	5.2×10^{-3}	Multi-Area Diverse Source System with Realistic Constraints	Load frequency control
8	Sahoo et al. [140]	5	Type-2	2020	CHHO	8	ITAE	2.33×10^{-2}	Power system	Frequency deviation and Deviance in tie-line power
9	Ghaleb et al. [141]	0	Type-1	2023	SSA (Social Spider)	5	Rise Time, Settling Time, Peak Time, Peak value	-	Inverted pendulum	Angle

* The number of citations listed in this table was last updated on 5 June 2023.

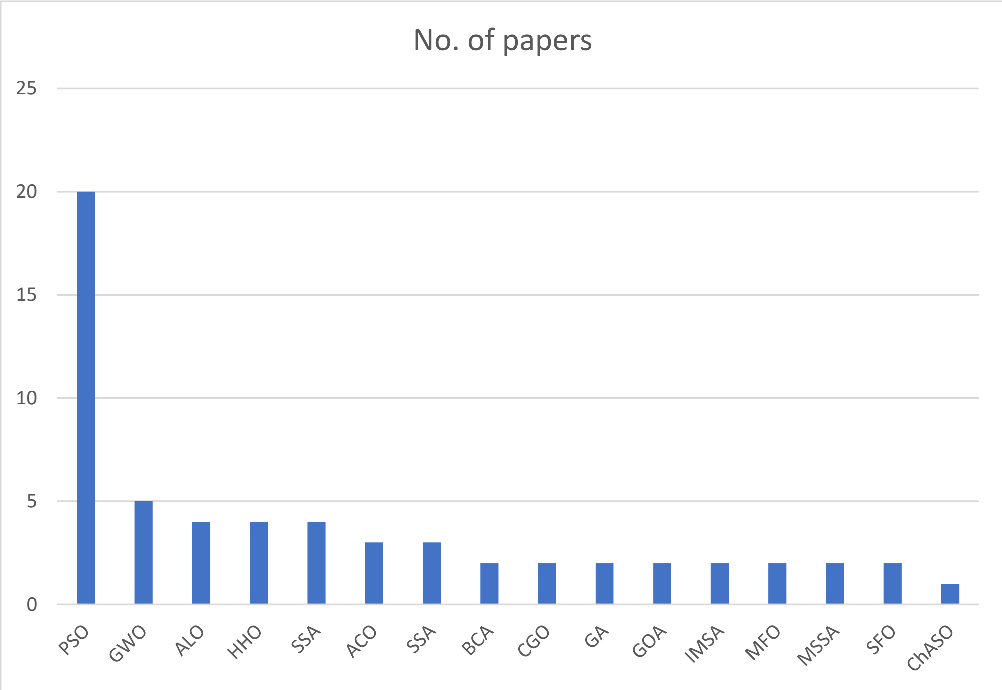

Figure 15. The list of highly used addressed optimizers and their number of published papers.

In terms of the MHA and based on the 82 relevant papers, the PSO algorithm is the most preferable optimizer. This is because of its simplicity and acceptable accuracy. By referring to Figure 15, PSO accommodates almost 25% of the works that use MHA to optimize FO controllers. Despite that the ChASO has a high citation record with 196 citations, the PSO is better in terms of the number of published papers.

Regardless of the application type, the objective function of ITAE appears to be the best formula to optimize the parameters of the controllers, as seen in Tables 8–10 as it has been used 11 times out of 27 with a percentage that exceeds 40%. Statistically, the second nominated objected function is the ISE, followed by IAE, then the ITSE, which have been used 7, 5, and 4 times out of 27. Despite that the ITSE gained less attraction to the researchers, it produced the lowest cost function value of 0.000052. The second-best cost function value of 0.002 was obtained by the ITAE. However, the third and fourth ranks were for the IAE and ISE that produced the same cost function value of 0.08.

It can be noticed that the use of ITSE produces the most accurate values with the lowest cost function, but it consumes a lot of memory due to the summation of squared errors over the simulation time. On the other hand, the ITAE can be considered the best nominated cost function as it uses less memory and at the same time produces very acceptable results.

In conclusion, for researchers who are interested in using the MHA in optimizing FOC, it is recommended to use the configuration illustrated in Figure 10 regardless of the type of controller, optimizer, or objective function. The ITAE introduced in Equation (16) is the best nominated cost function. Furthermore, the PSO is the best attractive optimizer according to the resulting statistical data. This is because of its simplicity and how easy it is to implement its mechanism.

7. Conclusions

The use of metaheuristics techniques has been proven to be very effective in the optimization of fractional-order controllers. This has urged many researchers to apply such techniques in the tuning of fractional-order controllers. Therefore, this paper introduces a recent, systematic, and comprehensive review on the use of MHA in optimizing the gains and orders of the FOCs.

The work starts by presenting an introduction of the four popular FOCs such as the FOPI- FOPID, FO-T1FLC, and FO-T2FLC. This introduction presents the structure and block diagram of the system when using such controllers. Afterward, an introduction about the MHAs and their history is presented. The PSO algorithm and its flowchart are introduced as it is the most popular, simple, and easy-to-use optimizer. In addition, the popular objective functions and the FO toolboxes are illustrated.

The systematic review is conducted by defining the search keywords and four scientific databases have been included. These databases are Scopus, Web of Science, Science Direct, and Google Scholar. The initial identified papers reached 850 papers. These papers were then filtered and screened using the PRISMA methodology. The final list contains the 82 papers that are the most appropriate works. The relevant data is then extracted and tabulated for further discussion. The most popular optimizers and their publication resources (DOI) are presented to assist researchers in easily access them.

The results showed that the FO-PID controller has garnered the highest level of interest among researchers compared the others. This can be observed from the significantly larger number of published papers, as the FO-PID controller is used in 48 articles as opposed to only 16 and 18 for FO-PI and FO-fuzzy controllers, respectively. In addition, the PSO occupies around 25% of the work that uses MHA to optimize FO controllers. However, the ChASO has a high citation record of 196 citations, up to the publication date of this work. Therefore, the PSO is better in terms of the record of published papers.

In conclusion, the selection of an appropriate algorithm and its parameters is crucial for obtaining good results. This review paper presented the most popular optimizers used in recent publications and the way these optimizers can be applied. For researchers who are interested in using the MHA in optimizing FOC, it is recommended to use the configuration illustrated in Figure 10 regardless of the type of controller, optimizer, or objective function. The ITAE introduced in Equation (16) has the best nominated cost function. Furthermore, the PSO is the most attractive optimizer according to the resulting statistical data. This is because of its simplicity and how easy it is to implement its mechanism. Therefore, future research can be conducted to focus on the development of new and more efficient metaheuristic algorithms for FOC design, as well as the investigation of their performance in real-world applications. Furthermore, other types of FOCs can also be reviewed.

Author Contributions: Conceptualization, A.M.N. and M.A.A.; methodology, A.M.N. and A.B.; formal analysis, A.M.N., H.M.M. and M.A.A.; investigation, A.M.N.; resources, A.M.N., A.B. and M.A.A.; writing original draft preparation, A.M.N. and A.B., H.M.M. and M.A.A.; writing review and editing, A.M.N., A.B., H.M.M. and M.A.A.; supervision, A.M.N. and M.A.A.; project administration, A.M.N.; funding acquisition, A.M.N. All authors have read and agreed to the published version of the manuscript.

Funding: This study was sponsored by the Prince Sattam bin Abdulaziz University via project number 2023/RV/016.

Data Availability Statement: Not applicable.

Acknowledgments: This study was sponsored by the Prince Sattam bin Abdulaziz University via project number 2023/RV/016.

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

References

1. DesRoches, C.T. The Concept of Sustainability. In *Environmental Ethics for Canadians*, 3rd ed.; Williston, B., Ed.; Oxford University Press: Oxford, UK, 2023; pp. 385–390.
2. Dli, M.; Puchkov, A.; Vasiliev, A.; Kirillova, E.; Selyavskiy, Y.; Kulyasov, N. Intelligent Control System Architecture for Phosphorus Production from Apatite-Nepheline Ore Waste. *Energies* **2021**, *14*, 6469. [CrossRef]
3. Kelektsoglou, K.; Karali, D.; Stavridis, A.; Loupa, G. Efficiency of the Air-Pollution Control System of a Lead-Acid-Battery Recycling Industry. *Energies* **2018**, *11*, 3465. [CrossRef]

4. Rezk, H.; Nassef, A.M.; Abdelkareem, M.A.; Alami, A.H.; Fathy, A. Comparison among various energy management strategies for reducing hydrogen consumption in a hybrid fuel cell/supercapacitor/battery system. *Int. J. Hydrogen Energy* **2021**, *46*, 6110–6126. [CrossRef]
5. Saber, E.M.; Chaer, I.; Gillich, A.; Ekpeti, B.G. Review of Intelligent Control Systems for Natural Ventilation as Passive Cooling Strategy for UK Buildings and Similar Climatic Conditions. *Energies* **2021**, *14*, 4388. [CrossRef]
6. Nassef, A.M.; Houssein, E.H.; Rezk, H.; Fathy, A. Optimal Allocation of Biomass Distributed Generators Using Modified Hunger Games Search to Reduce CO_2 Emissions. *J. Mar. Sci. Eng.* **2023**, *11*, 308. [CrossRef]
7. Yayla, A.; Świerczewska, K.S.; Kaya, M.; Karaca, B.; Arayici, Y.; Ayözen, Y.E.; Tokdemir, O.B. Artificial Intelligence (AI)-Based Occupant-Centric Heating Ventilation and Air Conditioning (HVAC) Control System for Multi-Zone Commercial Buildings. *Sustainability* **2022**, *14*, 16107. [CrossRef]
8. Ding, X.; Yu, J. The Design of Intelligent Building Lighting Control System Based on CNN in Embedded Microprocessor. *Electronics* **2023**, *12*, 1671. [CrossRef]
9. Rezk, H.; Inayat, A.; Abdelkareem, M.A.; Olabi, A.G.; Nassef, A.M. Optimal operating parameter determination based on fuzzy logic modeling and marine predators algorithm approaches to improve the methane production via biomass gasification. *Energy* **2022**, *239*, 122072. [CrossRef]
10. Martirosyan, A.V.; Ilyushin, Y.V.; Afanaseva, O.V. Development of a Distributed Mathematical Model and Control System for Reducing Pollution Risk in Mineral Water Aquifer Systems. *Water* **2022**, *14*, 151. [CrossRef]
11. Bonfiglio, A.; Cacciacarne, S.; Invernizzi, M.; Lanzarotto, D.; Palmieri, A.; Procopio, R. A Sliding Mode Control Approach for Gas Turbine Power Generators. *IEEE Trans. Energy Convers.* **2019**, *34*, 921–932. [CrossRef]
12. Daou, R.A.Z.; Moreau, X. Comparison between integer order and fractional order controllers. In Proceedings of the MELECON 2014-2014 17th IEEE Mediterranean Electrotechnical Conference, Beirut, Lebanon, 13–16 April 2014; pp. 292–297.
13. Caponetto, R.; Machado, J.T.; Murgano, E.; Xibilia, M.G. Model Order Reduction: A Comparison between Integer and Non-Integer Order Systems Approaches. *Entropy* **2019**, *21*, 876. [CrossRef]
14. Al-Dhaifallah, M.; Nassef, A.M.; Rezk, H.; Nisar, K.S. Optimal parameter design of fractional order control based INC-MPPT for PV system. *Sol. Energy* **2018**, *159*, 650–664. [CrossRef]
15. Nassef, A.M.; Rezk, H. Optimal Tuning of FOPID-Like Fuzzy Controller for High-Performance Fractional-Order Systems. *CMC-Comput. Mater. Contin.* **2022**, *70*, 171–180. [CrossRef]
16. Sondhi, S.; Hote, Y.V. Fractional Order Controller and its Applications: A Review. In Proceedings of the IASTED Asian Conference Modelling, Identification and Control/770: Advances in Computer Science and Engineering, Phuket, Thailand, 2–4 April 2012.
17. Chaib, L.; Choucha, A.; Arif, S. Optimal design and tuning of novel fractional order PID power system stabilizer using a new metaheuristic Bat algorithm. *Ain Shams Eng. J.* **2017**, *8*, 113–125. [CrossRef]
18. Holland, J.H. Genetic Algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
19. Pardalos, P.M.; Mavridou, T.D. Simulated Annealing. In *Encyclopedia of Optimization*; Floudas, C.A., Pardalos, P.M., Eds.; Springer: Boston, MA, USA, 2009; pp. 3591–3593.
20. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [CrossRef]
21. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; pp. 1942–1948.
22. Yadav, A.; Deep, K.; Kim, J.H.; Nagar, A.K. Gravitational swarm optimizer for global optimization. *Swarm Evol. Comput.* **2016**, *31*, 64–89. [CrossRef]
23. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*, n71. [CrossRef]
24. Muresan, C.I.; Birs, I.; Ionescu, C.; Dulf, E.H.; De Keyser, R. A Review of Recent Developments in Autotuning Methods for Fractional-Order Controllers. *Fractal Fract.* **2022**, *6*, 37. [CrossRef]
25. Kumarasamy, V.; KarumanchettyThottam Ramasamy, V.; Chandrasekaran, G.; Chinnaraj, G.; Sivalingam, P.; Kumar, N.S. A review of integer order PID and fractional order PID controllers using optimization techniques for speed control of brushless DC motor drive. *Int. J. Syst. Assur. Eng. Manag.* **2023**. [CrossRef]
26. Jamil, A.A.; Tu, W.F.; Ali, S.W.; Terriche, Y.; Guerrero, J.M. Fractional-Order PID Controllers for Temperature Control: A Review. *Energies* **2022**, *15*, 3800. [CrossRef]
27. Joseph, S.B.; Dada, E.G.; Abidemi, A.; Oyewola, D.O.; Khammas, B.M. Metaheuristic algorithms for PID controller parameters tuning: Review, approaches and open problems. *Heliyon* **2022**, *8*, e09399. [CrossRef] [PubMed]
28. Alilou, M.; Azami, H.; Oshnoei, A.; Mohammadi-Ivatloo, B.; Teodorescu, R. Fractional-Order Control Techniques for Renewable Energy and Energy-Storage-Integrated Power Systems: A Review. *Fractal Fract.* **2023**, *7*, 391. [CrossRef]
29. Reddy, K.; Saha, A.K. A review of swarm-based metaheuristic optimization techniques and their application to doubly fed induction generator. *Heliyon* **2022**, *8*, e10956. [CrossRef] [PubMed]
30. Ogata, K. *Modern Control Engineering*; Prentice Hall PTR: Hoboken, NJ, USA, 2001.
31. Golnaraghi, F.; Kuo, B.C. *Automatic Control Systems*; Wiley Publishing: Hoboken, NJ, USA, 2009.
32. Maddi, A.; Guessoum, A.; Berkani, D. Design of nonlinear PID-smith predictor controllers with large time delays. In Proceedings of the 2015 Third World Conference on Complex Systems (WCCS), Marrakech, Morocco, 23–25 November 2015; pp. 1–4.

33. Mashor, M.Y.; Mahdi, M.C. Performance of Manual and Auto-Tuning PID Controller for Unstable Plant-Nano Satellite Attitude Control System. In Proceedings of the 2018 6th International Conference on Cyber and IT Service Management (CITSM), Parapat, Indonesia, 7–9 August 2018; pp. 1–5.
34. Elhosseini, M.A.; El Sehiemy, R.A.; Salah, A.H.; Abido, M.A. Modeling and control of an interconnected combined cycle gas turbine using fuzzy and ANFIS controllers. *Electr. Eng.* **2018**, *100*, 763–785. [CrossRef]
35. Zadeh, L.A. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [CrossRef]
36. Passino, K.M.; Yurkovich, S. *Fuzzy Control*; Addison Wesley Longman: Menlo Park, CA, USA, 1998.
37. Yan, J.; Ryan, M.; Power, J. *Using Fuzzy Logic: Towards Intelligent Systems*; Prentice-Hall, Inc.: Hoboken, NJ, USA, 1995.
38. Rahman, S.M.A.; Rezk, H.; Shaikh, B.; Abdelkareem, M.A.; Olabi, A.G.; Nassef, A.M. Prediction of mass transfer during osmotically treated zucchini fruit product using advanced fuzzy inference system. *Neural Comput. Appl.* **2023**, *35*, 3125–3141. [CrossRef]
39. Nassef, A.M.; Rezk, H.; Alahmer, A.; Abdelkareem, M.A. Maximization of CO_2 Capture Capacity Using Recent RUNge Kutta Optimizer and Fuzzy Model. *Atmosphere* **2023**, *14*, 295. [CrossRef]
40. Reznik, L. *Fuzzy Controllers Handbook*; Biddles Ltd.: Norfolk, VA, USA, 1997.
41. Gilda, K.; Satarkar, S. Analytical overview of defuzzification methods. *Int. J. Adv. Res. Ideas Innov. Technol.* **2020**, *6*, 359–365.
42. Shvedov, A.S. On Type-2 Fuzzy Sets and Type-2 Fuzzy Systems. *J. Math. Sci.* **2021**, *259*, 376–384. [CrossRef]
43. De, A.K.; Chakraborty, D.; Biswas, A. Literature review on type-2 fuzzy set theory. *Soft Comput.* **2022**, *26*, 9049–9068. [CrossRef]
44. Mendel, J.M.; Hagras, H.; Tan, W.-W.; Melek, W.W.; Ying, H. *Introduction to Type-2 Fuzzy Logic Control*; Wiley-IEEE Press: Piscataway, NJ, USA, 2014.
45. Linda, O.; Manic, M. Uncertainty-Robust Design of Interval Type-2 Fuzzy Logic Controller for Delta Parallel Robot. *IEEE Trans. Ind. Inform.* **2011**, *7*, 661–670. [CrossRef]
46. Miller, K.S.; Ross, B. *An Introduction to the Fractional Calculus and Fractional Differential Equations*; Wiley: Hoboken, NJ, USA, 1993.
47. Petráš, I. Chapter Three-Fractional-order control: New control techniques. In *Fractional Order Systems*; Radwan, A.G., Khanday, F.A., Said, L.A., Eds.; Academic Press: Cambridge, MA, USA, 2022; Volume 1, pp. 71–106.
48. Sun, Z.-Z.; Gao, G.-h. *Fractional Differential Equations*; De Gruyter: Berlin, Germany, 2020.
49. Oustaloup, A.; Cois, O.; Lanusse, P.; Melchior, P.; Moreau, X.; Sabatier, J. The crone aproach: Theoretical developments and major applications. *IFAC Proc. Vol.* **2006**, *39*, 324–354. [CrossRef]
50. Axtell, M.; Bise, M.E. Fractional calculus application in control systems. *IEEE Conf. Aerosp. Electron.* **1990**, *562*, 563–566.
51. Bagley, R.L.; Calico, R.A. Fractional order state equations for the control of viscoelasticallydamped structures. *J. Guid. Control Dyn.* **1991**, *14*, 304–311. [CrossRef]
52. Makroglou, A.; Miller, R.K.; Skaar, S.B. Computational results for a feedback control for a rotating viscoelastic beam. *J. Guid. Control Dyn.* **1994**, *17*, 84–90. [CrossRef]
53. Podlubny, I. Fractional-order systems and fractional-order controllers. *Inst. Exp. Phys. Slovak Acad. Sci. Kosice* **1994**, *12*, 1–18.
54. Matignon, D. Stability results for fractional differential equations with applications to control processing. *Comput. Eng. Syst. Appl.* **1996**, *2*, 963–968.
55. Matignon, D. Some Results on Controllability and Observability of Finite-dimensional Fractional Differential Systems. *Comput. Eng. Syst. Appl.* **1996**, *2*, 952–956.
56. Shalaby, R.; El-Hossainy, M.; Abo-Zalam, B.; Mahmoud, T.A. Optimal fractional-order PID controller based on fractional-order actor-critic algorithm. *Neural Comput. Appl.* **2023**, *35*, 2347–2380. [CrossRef]
57. He, J.; Xiao, M.; Lu, Y.; Wang, Z.; Zheng, W.X. Fractional-order PID control of tipping in network congestion. *Int. J. Syst. Sci.* **2023**, *54*, 1873–1891. [CrossRef]
58. Chipipop, B.; Puangdownreong, D. Designing of Fractional Order PID Controller for Stable Fractional Order System. In Proceedings of the 2019 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT-NCON), Nan, Thailand, 30 January–2 February 2019; pp. 77–80.
59. Pradhan, R.; Majhi, S.K.; Pradhan, J.K.; Pati, B.B. Optimal fractional order PID controller design using Ant Lion Optimizer. *Ain Shams Eng. J.* **2020**, *11*, 281–291. [CrossRef]
60. El-Khazali, R. Fractional-order PIλDμ controller design. *Comput. Math. Appl.* **2013**, *66*, 639–646. [CrossRef]
61. Al-Dhaifallah, M. Fuzzy fractional-order PID control for heat exchanger. *Alex. Eng. J.* **2023**, *63*, 11–16. [CrossRef]
62. Mohammadzadeh, A.; Kumbasar, T. A new fractional-order general type-2 fuzzy predictive control system and its application for glucose level regulation. *Appl. Soft Comput.* **2020**, *91*, 106241. [CrossRef]
63. Rao, S.S. Classical Optimization Techniques. In *Engineering Optimization Theory and Practice*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2019; pp. 57–108.
64. Sörensen, K.; Sevaux, M.; Glover, F. A History of Metaheuristics. In *Handbook of Heuristics*; Springer: Cham, Switzerland, 2018; pp. 791–808.
65. Sörensen, K.; Glover, F.W. Metaheuristics. In *Encyclopedia of Operations Research and Management Science*; Kluwer Academic: New York, NY, USA, 2013; pp. 960–970.
66. Ho, Y.C.; Pepyne, D.L. Simple Explanation of the No-Free-Lunch Theorem and Its Implications. *J. Optim. Theory Appl.* **2002**, *115*, 549–570. [CrossRef]

67. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [CrossRef]
68. Almufti, S.M. Historical survey on metaheuristics algorithms. *Int. J. Sci. World* **2019**, *7*, 1–12. [CrossRef]
69. Rajwar, K.; Deep, K.; Das, S. An exhaustive review of the metaheuristic algorithms for search and optimization: Taxonomy, applications, and open challenges. *Artif. Intell. Rev.* **2023**. [CrossRef]
70. Rajabi Moshtaghi, H.; Toloie Eshlaghy, A.; Motadel, M.R. A comprehensive review on meta-heuristic algorithms and their classification with novel approach. *J. Appl. Res. Ind. Eng.* **2021**, *8*, 63–89. [CrossRef]
71. Stegherr, H.; Heider, M.; Hähner, J. Classifying Metaheuristics: Towards a unified multi-level classification system. *Nat. Comput.* **2020**, *21*, 155–171. [CrossRef]
72. Tilahun, S.L.; Tawhid, M.A. Swarm hyperheuristic framework. *J. Heurist.* **2018**, *25*, 809–836. [CrossRef]
73. Rezk, H.; Fathy, A. Simulation of global MPPT based on teaching–learning-based optimization technique for partially shaded PV system. *Electr. Eng.* **2017**, *99*, 847–859. [CrossRef]
74. Diab, A.A.Z.; Rezk, H. Global MPPT based on flower pollination and differential evolution algorithms to mitigate partial shading in building integrated PV system. *Sol. Energy* **2017**, *157*, 171–186. [CrossRef]
75. Yousef, B.A.A.; Rezk, H.; Abdelkareem, M.A.; Olabi, A.G.; Nassef, A.M. Fuzzy modeling and particle swarm optimization for determining the optimal operating parameters to enhance the bio-methanol production from sugar cane bagasse. *Int. J. Energy Res.* **2020**, *44*, 8964–8973. [CrossRef]
76. Tepljakov, A.; Vunder, V.; Petlenkov, E.; Nakshatharan, S.S.; Punning, A.; Kaparin, V.; Belikov, J.; Aabloo, A. Fractional-order modeling and control of ionic polymer-metal composite actuator. *Smart Mater. Struct.* **2019**, *28*, 084008. [CrossRef]
77. Tepljakov, A. FOMCON Toolbox for MATLAB. Available online: https://github.com/extall/fomcon-matlab/releases/tag/v1.50.4 (accessed on 1 June 2023).
78. Xue, D.Y.; Bai, L. *Fractional Calculus: Numerical Algorithms and Implementations*; Tsinghua University Press: Beijing, China, 2022.
79. Xue, D. FOTF Toolbox. Available online: https://www.mathworks.com/matlabcentral/fileexchange/60874-fotf-toolbox (accessed on 1 June 2023).
80. Valério, D.; Costa, J.S.d. Ninteger: A non-integer control toolbox for matlab. In *Proceedings of Fractional Differentiation and Its Applications*; Springer: Bordeaux, France, 2004.
81. Valério, D. Ninteger. Available online: https://www.mathworks.com/matlabcentral/fileexchange/8312-ninteger (accessed on 1 June 2023).
82. Agence de Protection de Programmes (APP). *CRONE Toolbox*; Agence de Protection de Programmes (APP): Paris, France, 2000.
83. Mirjalili, S. The Ant Lion Optimizer. *Adv. Eng. Softw.* **2015**, *83*, 80–98. [CrossRef]
84. Yang, X.S.; Hossein Gandomi, A. Bat algorithm: A novel approach for global engineering optimization. *Eng. Comput.* **2012**, *29*, 464–483. [CrossRef]
85. Teodorović, D. Bee Colony Optimization (BCO). In *Innovations in Swarm Intelligence*; Lim, C.P., Jain, L.C., Dehuri, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 39–60.
86. Talatahari, S.; Azizi, M. Chaos Game Optimization: A novel metaheuristic algorithm. *Artif. Intell. Rev.* **2020**, *54*, 917–1004. [CrossRef]
87. Hekimoğlu, B. Optimal Tuning of Fractional Order PID Controller for DC Motor Speed Control via Chaotic Atom Search Optimization Algorithm. *IEEE Access* **2019**, *7*, 38100–38114. [CrossRef]
88. Ibrahim, A.; Ali, H.A.; Eid, M.M.; El-kenawy, E.S.M. Chaotic Harris Hawks Optimization for Unconstrained Function Optimization. In Proceedings of the 2020 16th International Computer Engineering Conference (ICENCO), Cairo, Egypt, 29–30 December 2020; pp. 153–158.
89. Gandomi, A.H.; Yang, X.-S.; Alavi, A.H. Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems. *Eng. Comput.* **2011**, *29*, 17–35. [CrossRef]
90. Abedinpourshotorban, H.; Mariyam Shamsuddin, S.; Beheshti, Z.; Jawawi, D.N.A. Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm. *Swarm Evol. Comput.* **2016**, *26*, 8–22. [CrossRef]
91. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* **2020**, *191*, 105190. [CrossRef]
92. Yang, X.-S. Flower Pollination Algorithm for Global Optimization. In Proceedings of the Unconventional Computation and Natural Computation, Orléan, France, 3–7 September 2012; pp. 240–249.
93. Saremi, S.; Mirjalili, S.; Lewis, A. Grasshopper Optimisation Algorithm: Theory and application. *Adv. Eng. Softw.* **2017**, *105*, 30–47. [CrossRef]
94. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
95. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* **2019**, *97*, 849–872. [CrossRef]
96. Zong Woo, G.; Joong Hoon, K.; Loganathan, G.V. A New Heuristic Optimization Algorithm: Harmony Search. *Simulation* **2016**, *76*, 60–68. [CrossRef]
97. Madadi, M.R.; Akbarifard, S.; Qaderi, K. Improved Moth-Swarm Algorithm to predict transient storage model parameters in natural streams. *Environ. Pollut.* **2020**, *262*, 114258. [CrossRef]
98. Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl.-Based Syst.* **2015**, *89*, 228–249. [CrossRef]

99. Abdel-Basset, M.; Mohamed, R.; Saber, S.; Askar, S.; Abouhawwash, M. Modified Flower Pollination Algorithm for Global Optimization. *Mathematics* **2021**, *9*, 1661. [CrossRef]
100. Ouaar, F.; Boudjemaa, R. Modified salp swarm algorithm for global optimisation. *Neural Comput. Appl.* **2021**, *33*, 8709–8734. [CrossRef]
101. Sulaiman, M.; Salhi, A.; Fraga, E.S.; Mashwani, W.K.; Rashidi, M.M. A Novel Plant Propagation Algorithm: Modifications and Implementation. *Sci. Int.* **2016**, *28*, 201–209.
102. Mirjalili, S. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowl.-Based Syst.* **2016**, *96*, 120–133. [CrossRef]
103. Gomes, G.F.; da Cunha, S.S.; Ancelotti, A.C. A sunflower optimization (SFO) algorithm applied to damage identification on laminated composite plates. *Eng. Comput.* **2018**, *35*, 619–626. [CrossRef]
104. Satapathy, S.; Naik, A. Social group optimization (SGO): A new population evolutionary optimization technique. *Complex Intell. Syst.* **2016**, *2*, 173–203. [CrossRef]
105. Dhiman, G.; Kumar, V. Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems. *Knowl.-Based Syst.* **2019**, *165*, 169–196. [CrossRef]
106. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **2017**, *114*, 163–191. [CrossRef]
107. Yu, J.J.Q.; Li, V.O.K. A social spider algorithm for global optimization. *Appl. Soft Comput.* **2015**, *30*, 614–627. [CrossRef]
108. Eskandar, H.; Sadollah, A.; Bahreininejad, A.; Hamdi, M. Water cycle algorithm–A novel metaheuristic optimization method for solving constrained engineering optimization problems. *Comput. Struct.* **2012**, *110–111*, 151–166. [CrossRef]
109. Maamir, F.; Guiatni, M.; El Hachemi, H.M.S.M.; Ali, D. Auto-tuning of fractional-order PI controller using particle swarm optimization for thermal device. In Proceedings of the 2015 4th International Conference on Electrical Engineering (ICEE), Boumerdes, Algeria, 13–15 December 2015; pp. 1–6.
110. Guha, D.; Roy, P.K.; Banerjee, S. Grasshopper optimization algorithm scaled fractional order PI-D controller applied to reduced order model of load frequency control system. *Int. J. Model. Simul.* **2019**, *40*, 217–242. [CrossRef]
111. Barakat, M. Novel chaos game optimization tuned-fractional-order PID fractional-order PI controller for load–frequency control of interconnected power systems. *Prot. Control Mod. Power Syst.* **2022**, *7*, 16. [CrossRef]
112. Ramadan, H.S.; Padmanaban, S.; Mosaad, M.I. Metaheuristic-based Near-Optimal Fractional Order PI Controller for On-grid Fuel Cell Dynamic Performance Enhancement. *Electr. Power Syst. Res.* **2022**, *208*, 107897. [CrossRef]
113. Kakkar, S.; Maity, T.; Ahuja, R.K.; Walde, P.; Saket, R.K.; Khan, B.; Padmanaban, S. Design and control of grid-connected PWM rectifiers by optimizing fractional order PI controller using water cycle algorithm. *IEEE Access* **2021**, *9*, 125941–125954. [CrossRef]
114. Leena, J.G.; Sundaravadivu, K.; Monisha, R.; Rajinikanth, V. Design of Fractional-Order PI/PID Controller for SISO System Using Social-Group-Optimization. In Proceedings of the 2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA), Pondicherry, India, 6–7 July 2018; pp. 1–5.
115. Agarwal, S.; Mathur, S.; Mishra, P.; Kumar, V.; Rana, K.P.S. Online tuning of fractional order PI controller using particle swarm optimization. In Proceedings of the International Conference on Computing, Communication & Automation, Greater Noida, India, 15–16 May 2015; pp. 1026–1031.
116. Bouderres, N.; Kerdoun, D.; Djellad, A.; Chiheb, S.; Dekhane, A. Optimization of Fractional Order PI Controller by PSO Algorithm Applied to a Grid-Connected Photovoltaic System. *J. Eur. Des Systèmes Autom.* **2022**, *55*, 427–438. [CrossRef]
117. Zamee, W. A Novel Plant Propagation-Based Cascaded Fractional Order PI Controller for Optimal Operation of Grid-Connected Single-Stage Three-Phase Solar Photovoltaic System. *Appl. Sci.* **2019**, *9*, 4269. [CrossRef]
118. Sahul Hameed, J.B.; Ramasubramanian, J. Optimal Fractional Order PI Controller for Frequency Ancillary Services in Restructured Power System. *Energy Syst.* **2020**, *13*, 79–109. [CrossRef]
119. Özyetkin, M.M.; Birdane, H. The processes with fractional order delay and PI controller design using particle swarm optimization. *Int. J. Optim. Control Theor. Appl. (IJOCTA)* **2023**, *13*, 81–91. [CrossRef]
120. Altawil, I.; Momani, M.A.; Al-Tahat, M.A.; Al Athamneh, R.; Al-Saadi, M.A.; Albataineh, Z. Optimization of fractional order PI controller to regulate grid voltage connected photovoltaic system based on slap swarm algorithm. *Int. J. Power Electron. Drive Syst. (IJPEDS)* **2023**, *14*, 1184–1200. [CrossRef]
121. Labed, N.; Attoui, I.; Makhloufi, S.; Bouraiou, A.; Bouakkaz, M.S. PSO Based Fractional Order PI Controller and ANFIS Algorithm for Wind Turbine System Control and Diagnosis. *J. Electr. Eng. Technol.* **2022**, *18*, 2457–2468. [CrossRef]
122. ÇElebİ, M.; BaŞÇİ, A. Parameter Optimization of Bidirectional Three-Phase DC-AC Power Inverter by an Improved Particle Swarm Optimization based Fractional Order PI Controller for the Grid Forming Operation. *Int. J. Eng. Technol. IJET* **2022**, *8*, 25–30. [CrossRef]
123. Dwivedi, R.; Singh, S.; Singh, B. Design and development of three-phase EV charger based on the integration of voltage-oriented pulse width modulation control and water cycle algorithm fractional-order PI controller. *Int. J. Ambient Energy* **2022**, *44*, 157–170. [CrossRef]
124. Aghababa, M.P. Optimal design of fractional-order PID controller for five bar linkage robot using a new particle swarm optimization algorithm. *Soft Comput.* **2015**, *20*, 4055–4067. [CrossRef]
125. Vanchinathan, K.; Valluvan, K.R. A Metaheuristic Optimization Approach for Tuning of Fractional-Order PID Controller for Speed Control of Sensorless BLDC Motor. *J. Circuits Syst. Comput.* **2017**, *27*, 1850123. [CrossRef]

126. Khan, I.A.; Alghamdi, A.S.; Jumani, T.A.; Alamgir, A.; Awan, A.B.; Khidrani, A. Salp Swarm Optimization Algorithm-Based Fractional Order PID Controller for Dynamic Response and Stability Enhancement of an Automatic Voltage Regulator System. *Electronics* **2019**, *8*, 1472. [CrossRef]
127. Babu, A.G.S.; Chiranjeevi, B.T. Implementation of Fractional Order PID Controller for an AVR System Using GA and ACO Optimization Techniques. *IFAC-Pap.* **2016**, *49*, 456–461. [CrossRef]
128. Vanchinathan, K.; Selvaganesan, N. Adaptive fractional order PID controller tuning for brushless DC motor using Artificial Bee Colony algorithm. *Results Control Optim.* **2021**, *4*, 100032. [CrossRef]
129. Guha, D.; Roy, P.K.; Banerjee, S. Equilibrium optimizer-tuned cascade fractional-order 3DOF-PID controller in load frequency control of power system having renewable energy resource integrated. *Int. Trans. Electr. Energy Syst.* **2020**, *31*, e12702. [CrossRef]
130. Bouakkaz, M.S.; Boukadoum, A.; Boudebbouz, O.; Fergani, N.; Boutasseta, N.; Attoui, I.; Bouraiou, A.; Necaibia, A. Dynamic performance evaluation and improvement of PV energy generation systems using Moth Flame Optimization with combined fractional order PID and sliding mode controller. *Sol. Energy* **2020**, *199*, 411–424. [CrossRef]
131. Ghamari, S.M.; Narm, H.G.; Mollaee, H. Fractional-order fuzzy PID controller design on buck converter with antlion optimization algorithm. *IET Control Theory Appl.* **2021**, *16*, 340–352. [CrossRef]
132. Jaiswal, S.; Suresh Kumar, C.; 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]
133. Karahan, O. Design of optimal fractional order fuzzy PID controller based on cuckoo search algorithm for core power control in molten salt reactors. *Prog. Nucl. Energy* **2021**, *139*, 103868. [CrossRef]
134. Mohanty, D.; Panda, S. Modified Salp Swarm Algorithm-Optimized Fractional-Order Adaptive Fuzzy PID Controller for Frequency Regulation of Hybrid Power System with Electric Vehicle. *J. Control Autom. Electr. Syst.* **2021**, *32*, 416–438. [CrossRef]
135. Nayak, P.C.; Nayak, B.P.; Prusty, R.C.; Panda, S. Sunflower optimization based fractional order fuzzy PID controller for frequency regulation of solar-wind integrated power system with hydrogen aqua equalizer-fuel cell unit. *Energy Sources Part A Recovery Util. Environ. Eff.* **2021**, 1–19. [CrossRef]
136. Prusty, U.C.; Nayak, P.C.; Prusty, R.C.; Panda, S. An improved moth swarm algorithm based fractional order type-2 fuzzy PID controller for frequency regulation of microgrid system. *Energy Sources Part A Recovery Util. Environ. Eff.* **2022**, 1–23. [CrossRef]
137. Patel, H.R.; Shah, V.A. A metaheuristic approach for interval type-2 fuzzy fractional order fault-tolerant controller for a class of uncertain nonlinear system*. *Automatika* **2022**, *63*, 656–675. [CrossRef]
138. Bennaoui, A.; Saadi, S.; Ameur, A. Performance Comparison of MFO and PSO for Optimal Tuning the fractional order fuzzy PID Controller for A DC-DC Boost Converter. In Proceedings of the 2020 International Conference on Electrical Engineering (ICEE), Istanbul, Turkey, 3–5 July 2020; pp. 1–5.
139. Naga Sai Kalyan, C.H.; Goud, B.S.; Reddy, C.R.; Udumula, R.R.; Bajaj, M.; Sharma, N.K.; Elgamli, E.; Shouran, M.; Kamel, S. Seagull Optimization Algorithm–Based Fractional-Order Fuzzy Controller for LFC of Multi-Area Diverse Source System with Realistic Constraints. *Front. Energy Res.* **2022**, *10*, 921426. [CrossRef]
140. Sahoo, D.K.; Sahu, R.K.; Panda, S. Chaotic Harris Hawks Optimization based type-2 Fractional Order Fuzzy PID controller for frequency regulation of power systems. *Int. J. Ambient Energy* **2020**, *43*, 3832–3844. [CrossRef]
141. Ghaleb, A.F.; Oglah, A.A.; Humaidi, A.J.; Al-Obaidi, A.S.M.; Ibraheem, I.K. Optimum of fractional order fuzzy logic controller with several evolutionary optimization algorithms for inverted pendulum. *Int. Rev. Appl. Sci. Eng.* **2023**, *14*, 1–12. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.