



Article Modified Flower Pollination Optimization Based Design of Interval Type-2 Fuzzy PID Controller for Rotary Inverted Pendulum System

Mukhtar Fatihu Hamza 回

Department of Mechanical Engineering, College of Engineering in Alkharj, Prince Sattam bin Abdulaziz University, Alkharj 11942, Saudi Arabia; mh.hamza@psau.edu.sa

Abstract: The Type 2 Fuzzy Logic System (T2FLS) is an enhanced form of the classical Fuzzy Logic System (FLS). The T2FLS based control technics demonstrated a lot of improvements for the past few decades. This is based on the advantage of its membership function (MF). Many experimental studies indicated the superiority of Type 2 Fuzzy Logic Controller (T2FLC) over the ordinary Type 1 Fuzzy Logic Controller (T1FLC), particularly in the event of non-linearities and complex uncertainties. However, the organized design method of T2FLCs is still an interesting problem in the control engineering community. This is due to the difficulties in computing the parameters associated it. A novel application of the Modified Flower Pollination (MFP) optimization algorithm in the design of T2FL is presented. The optimized Cascade Interval Type 2 Fuzzy PID Controller (IT2FPIDC) structure is proposed in this study. The best values of the parameters of the antecedent MFs and the PID gains of IT2FPIDC are found using the MFP algorithm. The MFP optimization technique was used because of its lower computational effort and high convergence speed, in view of the higher number of variables to be optimized in cascaded IT2FPIDC. The MFP-based Type-1 Fuzzy Proportional Integral Derivative Controller (T1FPIDC) is compared with the proposed MFP-based cascade-optimized IT2FPIDC. The rotary inverted pendulum (RIP) which is a non-minimum phase, non-linear, and unstable system is employed as a benchmark for testing the proposed controller. Balance and trajectory-tracking controls of the RIP are considered. Furthermore, the disturbance rejection ability of the proposed controller is analysed. The presented control methos is evaluated on the RIP manufactured by Quanser over many simulations and real-world experiments. The performance characteristics considered are steady state error (E_{ss}) , settling time (t_s) , maximum overshoot (M_p) and rise time (t_r) . The improvement of the effectiveness and robustness proposed controller in the presence of load disturbance, noise effects and parameter variation is shown.

Keywords: fuzzy logic controller; rotary inverted pendulum; Flower Pollination algorithm; PID

1. Introduction

Real-world industrial systems show a momentous amount of unpredictability and complexity, making it challenging to design controllers to regulate them [1]. Researchers have proposed multiple controller designs to address these difficulties [2]. In recent decades, contemporary control techniques such as nonlinear, variable structure, adaptive, and optimal methods are employed, but they tend to be complicated and not easily implemented [3,4]. As a result, the control engineering community has become interested in using Type 2 fuzzy logic controllers (T2FLC) for nonlinear systems control, which has shown promising results in various successful applications [5,6]. It is hoped that these strategies will be generalized to other difficult control problems in the future.

There are two kinds of T2FLC: Interval Type-2 Fuzzy Logic Controller (IT2FLC) and General Type-2 Fuzzy Logic Controller (GT2FLC). The former uses Interval Type-2 Fuzzy Sets (IT2FSs) while the later uses wide-ranging Type-2 Fuzzy Sets [7]. This study employs



Citation: Hamza, M.F. Modified Flower Pollination Optimization Based Design of Interval Type-2 Fuzzy PID Controller for Rotary Inverted Pendulum System. *Axioms* 2023, 12, 586. https://doi.org/ 10.3390/axioms12060586

Academic Editor: Oscar Castillo

Received: 14 May 2023 Revised: 3 June 2023 Accepted: 7 June 2023 Published: 13 June 2023



Copyright: © 2023 by the author. 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/). IT2FLC because it is more practical and less computationally complex [8]. Furthermore, IT2FLC can create a very intricate control surface that a T1FLC with the same rule base cannot achieve because of the Footprint of Uncertainty (FOU) advantage inherent in the subsequent membership function of IT2FLC [9]. The construction methodology of the structure of IT2FS can be categorized into two: constructing it from an existing T1FS and directly designing IT2FSs using clustering methods or artificial neural network structures with collected investigational data. Experimental proof suggests that IT2FLC is significantly more efficient and accurate than T1FLC [10–12].

The primary challenge in designing IT2FLC is the difficulty and time-consuming nature of computing suitable values of parameter and structure [13]. This challenge has inspired researchers to explore the use of metaheuristics optimization algorithms, like Genetic Algorithm [14], Ant Colony Optimization [15], Big-Bang Big-Crunch Optimization [16], Particle Swarm Optimization [14], Biogeography Optimization [17], Bacterial Foraging Optimization [18], Simulated Annealing [19], Tabu Search Optimization [20], Firefly [21], Bee Colony Optimization [22], Cuckoo search algorithm [23], and hybrid algorithms [24], to automate the design process [23,25]. While these algorithms can provide near-optimal parameter values for IT2FLC, the computation time required is still quite high, and there is no clear consensus on which method is best. A FPA based interval type-2 fuzzy fractional-order controller is presented in [15]. This controller consists of fractional order TID controller and IT2FLC. The proposed controller was tested simulatically on Two nonlinear conical dual-tank level systems. a hybrid traffic signal control system with phase and time optimization based on IT2FLC which was optimized with Crow Search Algorithm and FPA methods, was developed in [25]. A hybrid method for fire outbreak detection based on IT2FL FPA, using environmental parameters is presented in [11]. The proposed controller is experimentally applied to detect a fire outbreak and compared with same controller without optimization which indicates the importance of the optimization technics.

This research proposes a new approach to optimizing the Membership Functions (MFs) structure and scaling factors of the Cascade Interval Type-2 Fuzzy PID Controller (IT2FPIDC) using Modified Flower Pollination (MFP) algorithm. The MFP algorithm was selected because it is both fast and accurate compared to other bio-inspired optimization algorithms [26]. The study employs the use of center-of-set type reduction of Takagi-Sugeno (T-S) type 2 fuzzy systems, which is an effective method for complex nonlinear systems, as mentioned in reference [27]. The design of the IT2FPID controller does not optimize the rule base or resulting MFs; only the PID scaling factors and the preceding MFs elements are optimized. This decision was made to demonstrate the impact of IT2FPIDC's additional degrees of freedom offered by its FOU.

This research introduces a time domain cost function that integrates four essential performance indicators: settling time, steady-state error, maximum overshoot and rise time. It is verified that employing MFP enhances IT2FPIDCs by searching for an optimal solution, leading to a superior controller when compared to T1FPIDC based on the proposed cost function. Both IT2FPIDC and T1FPIDC adopt a cascade structure, which proves effective for systems experiencing significant timing errors and high noise levels [28]. To validate the IT2FPIDC and T1FPIDC approaches, this study employs the hardware-in-loop (HIL) structure provided by MATLAB's QUARC target libraries, enabling a real-time controller interface.

The Rotary Inverted Pendulum (RIP) systems perform in an wide range of real life applications, such as flexible systems, locomotive systems, marine systems, mobile systems, pointing control aerospace systems and robotics. In addition, the study of dynamic model and control algorithms in controlling the RIP plays an important role in controlling spacecraft and rockets, maintaining the equilibrium state for two legs robots and skyscraping buildings. Moreover, when the pendulum of RIP is at hanging position, it represents real model of the simplified industry crane application [28]. The RIP is characterized by being non-linear, non-minimum phase, and unstable, making it an ideal system to test the proposed controller experimentally. The RIP has four primary control objectives: stabilization, swing-up, switching, and path tracking controls [29].

This study addressed the stabilization and path tracking controls objectives of the RIP using an MFP based cascade IT2FPIDC. In addition to that, the disturbance rejection capability of the proposed MFP-based cascade IT2FPIDC was analysed. The research began by presenting simulation studies that compared the performance of the optimized IT2FPIDC and optimized T1FPIDC structures. Next, a real experiment was conducted using the Quanser RIP to authenticate the presented cascade control methods. The MFPbased IT2FLC exhibited superior performance compared to the previous techniques in handling parameter variation, load disturbances, and noise effects, as indicated by both simulation and experimental results. Furthermore, it was shown that the path tracking and disturbance rejection achievement of the optimized cascade IT2FPIDC was superior to that of the optimized cascade T1FPIDC in the presence of parameter variations, uncertainties, and noise. These results were confirmed by both simulation and experimental outcomes, which demonstrated the efficacy and robustness of the suggested control methods. The use of MFP as a design strategy was also found to be effective in achieving high-quality solutions with less computing time, which can be advantageous in future applications requiring excellent optimization results in a short period.

The main contributions of the present study concerning the current state of the art and existing papers for optimizing type-2 fuzzy controllers are listed as follows:

- 1. Parameters optimization of the cascade interval type-2 fuzzy logic controller (PID gains of inner and outer controllers, and MFs parameters) using the Modified Flower Pollination (MFP) algorithm is proposed and explained in detail.
- Comparisons of optimized IT2FPIDC and optimized classical T1FPIDC in the presence of parameter variations, uncertainties, and noise are presented. This demonstrates the ability of the proposed controller in handling parameter variation, load disturbances, and noise effects.
- 3. Experimental validations of the simulations are presented.

The rest of the paper is organized as follows. Section 2 describes the Interval Type-2 Fuzzy Logic Systems. Section 3 Presents the Rotary Inverted Pendulum dynamic model and the experimental setup. Section 4 contains a more exhaustive information of Flower pollination algorithm and the modified Flower pollination algorithm. Section 5 contains the explanation of Cascade Control Method including the performance criterion, The internal structure of the Proposed IT2FPIDC and T1FPIDC, the optimization of the T1FPIDC cascade structure using the RIP Algorithm. Section 6 shows the Results and Discussion for both simulation and experiments while Section 7 presents the conclusion.

2. Interval Type-2 Fuzzy Logic Systems

This section presents several key characteristics and concepts related to IT2FSs. The concept of fuzzy logic systems and T2FS was originally proposed by Zadeh in 1965 and 1975, respectively. In an IT2FS, all secondary levels are uniform and equal to 1, and are completely determined by the Upper MF (UMF) and Lower MF (LMF) functions [30]. Figure 1 illustrates the construction of an IT2FS when $\mu_{\tilde{A}}(x, u) = 1$ for $\forall u \in J_x \subseteq [0, 1]$.

A T2FS $\stackrel{\sim}{A}$ is characterized by T2-MF $\mu_{\stackrel{\sim}{A}}(x, u)$, for $x \in X$ and $u \in J_x \subseteq [1, 0]$, that is,

$$\widetilde{A} = \left\{ \left((x, u), \mu_{\widetilde{A}}(x, u) \right) \middle| \forall u \in J_x \subseteq [0, 1] \right\}, \text{ in which } 0 \le \mu_{\widetilde{A}}(x, u) \le 1$$
(1)



Figure 1. Illustration of an IT2FS.

In this context, the primary membership is denoted as $J_x \subseteq [0, 1]$, while the secondary collection is indicated by $\mu_A(x, u)$, and is a type-1 fuzzy set (T1FS). Consequently, any T2 membership grade must lie within the range of 0 to 1 inclusively [31]. For every primary membership, there exists a corresponding secondary membership that also falls within the range of [1, 0], which characterizes its possibilities. The FOU region is used to represent uncertainty. The generic structure of type-2 fuzzy logic system is shown in Figure 2.



Figure 2. Structure of Type-2 fuzzy logic system.

2.1. Fuzzification

The purpose of the fuzzifier in type 1 fuzzy and type 2 fuzzy is to transform numeric vector entries $x = (x_1 \dots x_p)^T \in X_1 * X_1 * \dots * X_p \equiv X$ into a fuzzy set $_x$ (type 2 fuzzy set) defined in X. If the input values are singletons, the mapping can be carried out in the following way [32]:

$$\mu_{\tilde{A}_x}(x) = 1/1$$
 with $x = x'$, and $\mu_{\tilde{A}_x}(x) = 1/0$, for $\forall x \in x$ with $x \neq x'$ (2)

Equation (2) shows that $\mu_{\tilde{x}_i}(x_i) = 1/1$ when $x_i = x'_i$ and $\mu_{\tilde{x}_i}(x_i) = 1/0$ when $x_i \neq x'_i$ for all i = 1, ..., p.

2.2. Rules

The use of IF-THEN rules is common in both type 1 fuzzy and type 2 fuzzy systems. In type 2 fuzzy systems, the preceding and consequent MFs are depicted using type 2 fuzzy sets. The *i*th rule in a type 2 fuzzy system can be expressed using Equation (3) [33]:

$$R^i$$
: IF x_1 is \widetilde{F}_1^i , and ... and x_p is \widetilde{F}_p^i , THEN $Y^i = C_0 + C_1^i x_1 + \ldots + C_p^i$ (3)

where Y^i is the output of the *i*th IF THEN rule, i = 1, ..., M; for M = number of rules; $C_j^i(j = 0, ..., p)$ are the following fuzzy sets of type $F_k^i(k = 1, ..., p)$ are the preceding fuzzy sets of type -2.

2.3. Inference

In Type 2 fuzzy systems, the deduction mechanism is comparable to Type 1 fuzzy systems. It involves merging rules to create a transformation from an input type 2 fuzzy set to an output type 2 fuzzy set. To accomplish this conversion, it is essential to compute the union, intersection, and composition of type 2 associations [34]. In T2FLS, $F^i(x)$ represents the triggering level of the rule, and it can be expressed as shown in Equation (4).

$$F^{i}(\boldsymbol{x}) = \bigcap_{k=1}^{p} \mu_{\widetilde{F}^{i}_{k}}(\boldsymbol{x}_{k}) \tag{4}$$

The productivity of a T2FLS (TSK-type) can be stated as in Equation (5).

$$Y(\mathbf{x}) = \int_{y^{l} \in Y^{l}}^{\cdot} \dots \int_{y^{M} \in Y^{M}}^{\cdot} \times \int_{f^{l} \in F^{l}}^{\cdot} \dots \int_{f^{M} \in F^{M}}^{\cdot} \left[T_{i=1}^{M} \mu_{Y^{i}} \left(y^{i} \right) \cap T_{i=1}^{M} \mu_{F^{i}} \left(f^{i} \right) \right] / \frac{\sum_{i=1}^{M} f^{i} Y^{i}}{\sum_{i=1}^{M} f^{i}}$$
(5)

If interval T2FSs and T1FSs are used for the antecedents and sequences, respectively, of the type 2 rule, then $\mu_{\substack{x_i \\ F_k}}(x_k)$ as well as C_j^i interval fuzzy sets. Thus,

$$\mu_{\widetilde{F}_k^i}(x_k) = \left[\underline{\mu}_{\widetilde{F}_k^i}(x_k), \overline{\mu}_{\widetilde{F}_k^i}(x_k) \right], \quad k = 1, \dots, p$$
(6)

And

$$C_j^i = \left[c_j^i - s_j^i, c_j^i + s_j^i\right] \tag{7}$$

where s_j^i shows the spread of C_j^i , and c_j^i shows the center (mean) of C_j^i , C_j^i (i = l, ..., M and j = 0, 1, ..., p). For IT2FLS, the antecedent rule is an interval T1FS and can be portrayed as:

$$F^{i}(\boldsymbol{x}) = \left[\underline{f}^{i}(\boldsymbol{x}), \overline{f}^{i}(\boldsymbol{x})\right]$$
(8)

where

$$\underline{f}^{i}(\mathbf{x}) = \underline{\mu}_{F_{l}^{i}}(\mathbf{x}) \cap \ldots \cap \underline{\mu}_{F_{p}^{i}}(\mathbf{x}_{p})$$
(9)

$$\overline{f}^{i}(\boldsymbol{x}) = \overline{\mu}_{F_{i}^{i}}(\boldsymbol{x}) \cap \ldots \cap \overline{\mu}_{F_{p}^{i}}(\boldsymbol{x}_{p})$$
(10)

where \cap is a conjunction operator, is a t-norm, which can be an algebraic product (*) or a minimum (). The interval value of the consequence can be formulated as follows:

$$Y^{i} = \begin{bmatrix} y_{l}^{i}, y_{r}^{i} \end{bmatrix}$$
(11)

where

$$y_l^i = \sum_{i=1}^p c_k^i x_k + c_0^i - \sum_{k=1}^p |x_k| s_k^i - s_0^i$$
(12)

$$y_r^i = \sum_{i=1}^p c_k^i x_k + c_0^i - \sum_{k=1}^p |x_k| s_k^i - s_0^i$$
(13)

2.4. Output Processing

To acquire a precise output, the output processing step includes type reduction which results in a type 1 fuzzy set and defuzzification. The type reduction technique used here is the center-of-set method.

$$Y_{cos}(Y^{1},...,Y^{M},F^{1},...,F^{M}) = [y_{l},y_{r}] = \int_{y^{1}}^{\cdot}...\int_{y^{M}}^{\cdot}.\int_{f^{1}}^{\cdot}...\int_{f^{M}}^{\cdot}.1/\frac{\sum_{i=1}^{M}f^{i}y^{i}}{\sum_{i=1}^{M}f^{i}}$$
(14)

where Y_{cos} represents the interval set defined by y_l and y_r (two end points); $f^i \epsilon F^i = \lfloor \underline{f}^i, \overline{f}^i \rfloor$; $y^i \epsilon Y^i = \lfloor \underline{y}^i, \overline{y}^i \rfloor$; i = 1, ..., M; and M is the number of rules. Research has demonstrated that IT2FLC defuzzification can be performed as follows [8]:

$$y = \frac{y_l + y_r}{2} \tag{15}$$

where y_l and y_r are the endpoints of the type-reduced sets. These two endpoints can be calculated by rearranging $Y_n\zeta_n$ to set $Y_1\zeta_1 \leq Y_2\zeta_2 \leq \ldots Y_N\zeta_N$ and the equivalent escape interval. The y_l and y_r can be considered as follows [31]:

$$y_l = \frac{\sum_{n=1}^{L} \overline{f}_n Y_n \zeta_n + \sum_{L+1}^{N} \underline{f}_n Y_n \zeta_n}{\sum_{n=1}^{L} \overline{f}_n \zeta_n + \sum_{L+1}^{N} \underline{f}_n \zeta_n}$$
(16)

$$y_r = \frac{\sum_{n=1}^{R} \underline{f}_n Y_n \zeta_n + \sum_{R+1}^{N} \overline{f}_n Y_n \zeta_n}{\sum_{n=1}^{M} \underline{f}_n \zeta_n + \sum_{M+1}^{N} \overline{f}_n \zeta_n}$$
(17)

The calculation of the switching points *L* and *R* is performed by employing the Karnik-Mendel reduction method.

3. Rotary Inverted Pendulum

Figure 3a,b describe the experimental configuration and schematic diagram of the RIP. Two optical encoders were used to measure the pendulum's and arm's angles, and a data collection device was used to gather the encoder information and send it to the computer. Additionally, a data logger was employed to receive control signals from the computer and amplify them through a power amplifier before transmitting them to the motor. This study utilizes the RIP developed by Quanser. The direction of the arm's movement is considered positive when moving counter clockwise, while the pendulum is considered positive when moving clockwise. The initial angles for the arm and pendulum are 0 and -180 degrees, correspondingly, during real-time testing. The sampling time for the experiments is 0.01 s. The stabilization controller is programmed to activate once the sway angle reaches $\pm 10^{\circ}$.



Figure 3. (a) Experiment setting of RIP, (b) experiment's schematic diagram of the RIP's.

The experiments focused on examining the stabilization control and trajectory tracking control of the optimized T1FPIDC and optimized IT2FPIDC controllers. Moreover, the real-time robustness assessment of the proposed optimized controllers was conducted. For this purpose, an extra rod, having identical length and weight as the original pendulum, was affixed to the free end of the pendulum. This modification aimed to modify the physical characteristics of the pendulum by altering the position of its center of mass and

diminishing its rigidity. Specifically, the additional rod had a length of 0.1685 m and a weight of 0.0635 kg.

The RIP technology has numerous practical uses in various industries including aerospace, robotics, ships, target control, mobile systems, and locomotive systems [28]. Moreover, the analysis of dynamic models and control methods for RIPs is crucial for controlling rockets and spacecraft, maintaining the balance of biped robots and skyscraper buildings. Additionally, when the pendulum of the RIP is in a hanging position, it can serve as a simplified industrial crane model for industrial applications [35].

Dynamic Model of RIP

To describe the RIP, it can be divided into two components, which are the pendulum part and the arm part. The arm pivot is connected to the motor, and the angle φ increases in a counter clockwise direction about the *z*-axis. When the control voltage is positive (i.e., $V_m > 0$), φ becomes positive. The reference for φ is based on the *x*-axis. The pendulum is linked to the arm's free end. As it rotates counter clockwise around an axis that passes through the arm, the pendulum angles ρ and α increase positively. The reference for ρ and α is vertical down and vertical up, respectively. These components are illustrated in Figure 4.



Figure 4. Mathematical model of RIP.

The state space model (18) and (19) represents the linear dynamics of the RIP. It was achieved by replacing $sin(\alpha)$ with α and $cos(\alpha)$ with 1 in the nonlinear dynamic model, and assigning specific values to the RIP elements, which are listed in Table 1.

$$\begin{bmatrix} \dot{q}_1\\ \dot{q}_2\\ \ddot{q}_1\\ \ddot{q}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 73.96 & -117.05 & -0.08766\\ 0 & 101.83 & -4.34 & -1.2561 \end{bmatrix} \begin{bmatrix} q_1\\ q_2\\ \dot{q}_1\\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} 0\\ 0\\ 52.49\\ 2.624 \end{bmatrix} V_m$$
(18)
$$\begin{bmatrix} q_1\\ q_2\\ q_1\\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_1\\ q_2\\ \dot{q}_1\\ \dot{q}_2 \end{bmatrix}$$
(19)

The RIP system has open loop poles located at w = [0, 9.346, -10.594, -118.272], and it's evident that the system is unbalanced due to the presence of a pole on the right side of the S-plane. Therefore, before applying any control action, it's crucial to test the

controllability of the system. This is attainable by evaluating the rank of the matrix *R* using Equation (20). If the rank of *R* equals 4, then the system is entirely controllable.

$$R = \left[B, AB, A^2B, A^3B \dots A^{n-1}B\right]$$
(20)

Table 1. Description and values of RIP system.

Symbol	Explanation	Value	Unit
тр	Pendulum's mass	0.124	kg
Jp	Pendulum's moment of inertia about its center of mass	0.00117	kg⋅m ²
B _p	Viscous damping coefficient of the pendulums	0.0023	N·m·s/rad
lp	Length of pendulum's center of mass	0.152	m
L _r	Arm's length	0.164	m
B _r	Arm's viscous damping coefficient as seen at the pivot axis	0.0023	N·m·s/rad
Jr	Arm's moment of inertia about its center of mass	0.000978	kg⋅m ²
R_m	Armature resistance of the motor	2.5	Ω
k _t	Current-torque constant of the motor	0.00753	N-m/A
<i>k</i> _m	Back-emf constant of the motor	0.00753	V/(rad/s)
Kg	High-gear to total gear ratio	72	
η_m	Efficiency of the motor	71	%
η_g	Efficiency gear box	91	%

4. Flower Pollination Algorithm (FPA) Optimization

The concept of the flower pollination algorithm (FPA) was initially introduced by Yang et al. in 2012 [36,37] and was influenced by the pollination process of flowering plants that occurs in nature. In flowering plants, the flowers play a crucial role in the reproduction process through pollination. Pollination can be categorized as self-pollination, where the flower is fertilized by its own pollen, or cross-pollination, where the pollen from one flower is transferred to another flower belonging to a distinct plant. Cross-pollination typically takes place over considerable distances, and the assistance of pollinators like bees, birds, and flies is essential for facilitating this phenomenon. Bees and birds exhibit flight behavior that adheres to a Levy distribution. Additionally, bloom constancy serves as an incremental step to assess the similarity or dissimilarity between two blooms. The traditional FPA relies on the behavior of flower constancy, where pollinators visit elite flower species while bypassing other flowers. This type of constancy enhances the transfer of pollen to the same flower species, increasing its reproduction. The fundamental principles of the FPA are described as follows:

Rule 1: The process of cross-pollination can be considered as a form of global pollination, where pollinators follow flight patterns that adhere to the Levy distribution.

Rule 2: The procedure of self-pollination can be characterized as local pollination.

Rule 3: Pollinators, such as insects, acquire the ability to persist in flower pollination, which can be equated to a probability of reproduction. This probability is determined by evaluating the resemblance between the two flowers engaged in the process.

Rule 4: The transition from local pollination to global pollination can be controlled by a switching probability, denoted as p, which can take values of either 0 or 1. It is slightly inclined towards favoring local pollination. By incorporating the aforementioned rules, the update equations for FPA (Flowering Persistence Algorithm) can be deduced. The mathematical expression for the first and third rules can be represented as shown in Equation (21).

$$u_n^{i+1} = u_n^i + \delta L(x) \left(u_{best} - u_n^i \right)$$
(21)

where u_n^t represents the *n*th pollen or solution u_n at iteration *i*, u_{best} is the optimal solution out of all the solutions obtained thus far, δ is a scaling factor to regulate the step size, and L(x) is the Levy-based step size indicating the amount of pollination. Thus L(x) as in Equation (22).

$$L \approx \frac{x\Gamma(\gamma)\sin\left(\frac{\pi x}{2}\right)}{\pi} \frac{1}{s^1 + x}, \quad s >> s_0 > 0$$
(22)

where $\Gamma(\gamma)$ are the typical gamma function and this distribution is satisfactory for large steps s > 0. Although $s \gg 0$ is required, in practice s_0 is as small as 0.1. However, it is not easy to generate pseudo-random step sizes that correctly follow the Levy distribution [38]. Therefore, an effective algorithm available in the literature, the Mantegna algorithm [39], is used in FPA to arrive at these random values. The step size *s* can be calculated using two Gaussian distributions G_u and G_v Equation (23):

$$s\frac{G_u}{|G_v|^{\frac{1}{x}}}, G_u \sim N(0, var^2), G_v \sim N(0, 1)$$

$$(23)$$

 $G_u \sim N(0, var^2)$ implies that the samples are drawn from a Gaussian distribution with a mean and variance both set to 0 and var^2 , respectively. The variance var^2 is obtained with Equation (24)

$$var^{2} = \left[\frac{\Gamma(1+x)}{x\Gamma((1+x)/2)} \cdot \frac{\sin \pi x/2}{2^{(x-1)/2}}\right]$$
(24)

$$\Gamma'(1+\lambda) = 1, \Gamma'((1+\lambda)/2) = 1$$
 (25)

When the value of x is equal to 1, the gamma functions become and var^2 becomes equal to 1. From rules 2 and 3, the update equation for the local pollination can be expressed as in Equation (26):

$$u_n^{i+1} = u_n^i + \in \left(u_a^i - u_b^i\right) \tag{26}$$

where u_a^i and u_b^i stands for pollen from two various flowers of the same plant. If u_a^i and u_b^i are taken mathematically from the same plant species, this distribution becomes a local random walk throughout when \in drawn from a uniform distribution in [0, 1]. The pollination process of flowers occurs both locally and globally, but in reality, neighbouring flowers are more likely to be pollinated by local pollen than by pollen from a significant distance. This feature can be incorporated into the algorithm by introducing a switching or proximity probability (p) as described in Rule 4. This probability can be used to shift from global pollination to strict local pollination, as required. A value of $\rho = 0.5$ can be set as the initial probability.

4.1. Modified FPA

The M-FPA method proposes the use of adaptive orientation Gaussian (AOG) mutation to optimize controller parameters. This mutation-based flower pollination process modifies the properties of pollen during pollination to speed up the optimization algorithm. The AOG mutation method is used in FPA to modify certain particle properties and reach the solution faster. The proposed M-FPA algorithm achieves faster convergence while maintaining the traditional FPA's property consistency. In the conventional FPA, the AOG mutation is applied to global pollination's produced pollen after the pollination process. As a result, Equation (21) is transformed into Equation (27) following the mutation process [40]:

$$u_{n\ mut}^{i+1} = u_n^i + \delta L(x) \left(u_{best} - u_n^i \right) + \varphi \{AOG\}$$
⁽²⁷⁾

where φ is the probability factor. AOG mutation function is given by Equation (27).

$$AOG(x) = (G_{\sigma}(x\sin\theta) - G_{\sigma}(x\cos\theta))$$
(28)

$$G_{\sigma} = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-1}{2}\left(\frac{x-m}{\sigma}\right)^2}$$
(29)

where, m, σ , $G_{\sigma}(x)$ and θ mean, variance, Gaussian function of the variable x and orientation to which the function must be rotated. Likewise, the mutation is applied to p. The one from Equation (25) is modified as Equation (30) after applying the mutation process

$$u_{n\ mut}^{i+1} = u_n^i + \in \left(u_a^i - u_b^i\right) + \varphi\{AOG\}$$
(30)

The probability factor that determines the likelihood of a change in solution is defined as follows: If the characteristics of pollen obtained from global or local pollination in the current iteration (i + 1) match those obtained in the previous iteration (i), the probability factor is 1. If not, the probability factor is zero, and the solution remains unaffected.

$$\varphi = \begin{cases} 1 & if \left(u_n^{i+1} - u_n^i = 0\right) \\ 0 & else \end{cases}$$
(31)

The algorithm for the AOG-FPA is given below [40].

4.2. Implementation of M-FPA for Optimizing IT2FLC Controller

To begin with, a pollen matrix which has the size of $n \times 4$ is taken into account. The controller's parameters are represented by the pollen, and n indicates the total number of plants. The specific arrangement of the matrix can be found in Equation (32).

$$u_{n} = \begin{bmatrix} K_{P1} & K_{I1} & K_{D1} & K_{a1} \\ K_{P2} & K_{I2} & K_{D2} & K_{a2} \\ \vdots & \vdots & \vdots & \vdots \\ K_{Pn} & K_{In} & K_{Dn} & K_{an} \end{bmatrix}$$
(32)

Firstly, use the information in each row to calculate the fitness function and identify the plant with the best fitness. The performance criterion from Equation (13) will be used for both the inner and outer controllers. Then, update the pollen for the next iteration by employing either global or local pollination, depending on the switch probability. The pollen for the *n*th plant is represented by u_n and has two positions, u_{n_m} , where *m* is 4 for each controller in this particular application. In global pollination, all rows' pollen is used for pollination, whereas in local pollination, only the same row's pollen is used because the pollen from a row belongs to the same plant flower. Thus, this algorithm finds the optimal solution set when the convergence criterion is met.

5. Proposed Control Method

The RIP system is a type of SIMO system, meaning that it has multiple outputs controlled by a single input [41]. When one output is affected by disturbances, it can disrupt the control of the other output. Due to the non-linear behavior of the RIP system, determining the elapsed settling time can be difficult. The MFP system also presents challenges, such as a large time constant and elevated noise levels. To address these issues, a cascade control method is recommended. This method is advantageous in reducing the impact of disturbances and improving the dynamics of the entire control loop [42]. By integrating the characteristics of IT2FLC with a cascade control architecture, a stronger and more resilient control response can be attained. Figure 4 illustrates the general structure of the cascade control and the optimization of the controller parameters.

In this particular situation, the system that requires control is made up of two subsystems, referred to as subsystem 1 and subsystem 2, as displayed in Figure 5. The cascade control structure contains two control loops, each with its own controller loop. The discrepancy between the desired input signal (y_{ref}) and the output of subsystem 2 (y_0) serves as

the input for the outer controller. The input for the inner controller is the disparity between the output of the outer controller 0 and the output of subsystem 1 (y_i). The output of the internal regulator (r_i) serves as a control input for both subsystem 1 and subsystem 2. Tuning the control elements in the cascade control approach can be performed separately, following the process described in the literature [41]. This involves developing the inner loop controller utilizing the suggested objective function first, and then designing the outer controller after tuning the inner controller. This technique is applied in this work. The IT2FPIDC is utilized as both the inner and outer controller in this study. Additionally, the T1FPIDC is utilized as both the inner and outer controller for comparison purposes.



Figure 5. General cascade control structure with parameters optimization.

5.1. Proposed Performance Criterion

This study implements the optimized IT2FPIDC and T1FPIDC in cascade form and introduces a design method for the MFP-based controller related to the performance criterion C(*t*) expressed in Equation (33). Optimization methods for T2FLC design typically aim to reduce the integrated absolute error (IAE), integral squared error (ISE), or integrated time-weighted squared error (ITSE), which are commonly utilised performance indices in control system design because of their straightforward assessment in the frequency domain [43]. However, these indices have their benefits and drawbacks. For example, minimizing ISE and IAE can lead to a response with a small overshoot but a long settling time because ISE weighs all errors equally, regardless of when they occur. The ITSE can address this issue, but its analysis formula is complex and time-consuming to derive [43]. To overcome these limitations, this study adopts a time-domain performance criterion that encompasses four different control performance indices:

$$C(t) = \frac{e^{-\gamma}}{2}(t_s - t_r) + \frac{1 - e^{-\gamma}}{2}(E_{ss} + M_p)$$
(33)

The performance criterion used in this study includes four performance indices: E_{ss} (steady-state error), t_r (rise time), t_s (settling time), M_p (overshoot), and a weighing factor γ . By adjusting the value of γ , the performance criterion can meet the design requirements. According to Gaing [43], setting $\gamma > 0.7$ reduces steady-state error and overshoot, while setting $\gamma < 0.7$ reduces settling time and rise time. The study considers two different values of γ (i.e., $\gamma = 1$ and 1.5) to analyse the impact of the performance criterion and explore possible solutions.

The performance of the RIP is analyzed in terms of its dynamic behavior and convergence characteristics by studying the average (λ) and variability (σ) of the performance criterion across all subjects in the calculation process. The accuracy of the algorithm is determined using the average value, whereas the speed of convergence of the algorithm is determined using the standard deviation value [44]. The formulas for calculating the variability (σ) and average (λ) are presented in Equations (34) and (35) respectively [45].

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(C(t)_{p_i} - \lambda \right)^2}$$
(34)

$$\lambda = \frac{\sum_{i=1}^{n} C(t)_{p_i}}{n} \tag{35}$$

where *n* denotes the population size and $C(t)_{p_i}$ denotes the individual performance criterion value.

5.2. Design of T1FPIDC and IT2FPIDC in Cascade Form

In this section, the MFP algorithm-based design of the structures for both T1FPIDC and IT2FPIDC, along with an introduction to their internal structures are portrayed.

5.2.1. The Internal Structure of the Proposed IT2FPIDC and T1FPIDC

Using an MFP algorithm, the methodology optimizes the scaling factor and MFs parameters of the proposed IT2FPIDC for stabilization and path tracking controls objectives of the RIP with negligible stability error. In this situation, the performance of an IT2FPIDC is measured and the result is plot based on the Proposed Performance Criterion. This process is continued to optimize until a stopping criterion or a predefined number of iterations is met. Figure 5 shows a simplified form of the process [46]. The fuzzy system employed in both inner and outer controllers have two inputs and one output. The input parameter is "simulation iterations (SI)" and the output parameter is "change probability (P')" as adapted from [47,48]. In Equation (36), SI represents the generations of the MFP algorithm. The present simulation denotes the present generations and the maximum number of simulations represents the maximum number of generations. Equation (36) is used to input the trailed IT2FPIDC according to the following method:

$$SI = \frac{Present Simulation}{maximum number of simulations}$$
(36)

After computing the input of tailed type-2 fuzzy inference system the outputs Switching "Probability (P')" fuzzy FP algorithm can be found. Consequently, after proposing the trilled T2FIS for fuzzy MFP algorithms, the optimized IT2FLS for a RIP control system was realized. The IT2FLC optimized for RIP which is nonlinear systems was tested both with and without uncertainties.

Figure 6 displays the standard fuzzy PID structure with two inputs and one output. The outer controller takes error $e_1(t)$ and error change $\Delta e_1(t)$ as its inputs, while its output is denoted as U. These inputs are adjusted to $E_1(t)$ and $\Delta E_1(t)$ within the range defined by the outer controller's input MFs, using the scale factors K_{P_1} and K_{D_1} in that order. To transform the signal U(t) into in u(t), output scale factors of K_{I_1} and K_{a_1} are utilized. The process of normalization is based on the equations below:

$$E_1(t) = K_{P_1}e_1(t) = K_{P_1}(q_r(t) - q_2(t)),$$
(37)

$$\Delta E_1(t) = K_{D_1} \Delta e_1(t) = K_{D_1}(e_1(t) - e_1(t-1)), \tag{38}$$

$$u(t) = K_{I_1} \int U(t)dt + K_{a_1}U(t).$$
(39)



Figure 6. Standard Fuzzy PID controller in cascade form with parameter optimization.

If the current sampling time is *t*, the parameter U(t) is the output of the outer control loop $q_2(t)$ and $q_r(t)$ is the sway angle or reference signal.

In a similar fashion, the inner controller takes error $e_2(t)$ and change of error $\Delta e_2(t)$ as its inputs, and its output is denoted as *V*. These inputs are adjusted to $E_2(t)$ and $\Delta E_2(t)$, respectively, by normalization within the domain defined by the inner controller input MFs, using scaling factors K_{P_2} and K_{D_2} . The output V(t) is transformed into the control voltage of the servo motor, $v_m(t)$, through scaling factors K_{I_2} and K_{a_2} . This normalization process is based on the following equations:

$$E_2(t) = K_{P_2}e_2(t) = K_{P_2}(u(t) - q_1(t)),$$
(40)

$$\Delta E_2(t) = K_{D_2} \Delta e_2(t) = K_{D_2}(e_2(t) - e_2(t-1)), \tag{41}$$

$$V_m(t) = K_{I_2} \int V(t) dt + K_{a_2} V(t).$$
(42)

where V(t) is the output of the inner loop and $q_1(t)$ is the arm angle.

In Table 2, the technique that employs an unequal rule is utilized for managing both IT2FPIDC and T1FPIDC. Equations (43) and (44) present the usual configuration of IT2FPIDC and T1FPIDC correspondingly.

$$R_n$$
: IF E is $\stackrel{\sim}{A_j}^1$ and ΔE is $\stackrel{\sim}{A_j}^2$ THEN y is Y_n with ζ_n (43)

$$R_n$$
: IF E is A_i^1 and ΔE is A_i^2 THEN y is Y_n with ζ_n (44)

where n = 1, ..., 9 is the number of rules, ζ_n denotes the weighting factor employed to indicate the importance of the respective rule, and $\zeta_n \in [0, 1]$. Triangular MFs are employed to describe the entrance of both IT2FLC and T1FLC structures as indicated in Figure 7a,b respectively. These MFs are called negative, zero, and positive, reprents by N, Z, and P respectively. Parameters l_{ij} , $l_{jj}c_{ij}r_{ij}$ and r_{jj} are used to describe the previous IT2FSs of the IT2PIDC as shown in Figure 7a, where i = 1, 2 and j = 1, 2, 3. To be fair in comparisons, the output of IT2FPIDC is the same as that of T1FPIDC (five Singleton Crisp Consequents) as

14 of 30

shown in Figure 7c. Also, the weighting factor and the output MFs are not optimized in this study. The weight for all rules is set to 1

Table 2. Modified Flower Pollination.

- 1. Initialized the number of pollens with random solutions
- 2. Calculate the finest solution from the initial population
- 3. Set a switch probability p
- 4. while (i < max. No. of generation)
- 5. for n = 1:N
- 6. if rand < p
- 7. Draw a d-dimensional step vector that follows Levy distribution
- 8. Perform global pollination using Equation (21)
- 9. Compute mutation probability factor using Equation (31)
- 10. If $\phi = 1$
- 11. Determine new pollen in global pollination using Equation (25)
- 12. Else
- 13. New pollen is calculated using Equation (26)
- 14. Draw from a uniform distribution in [0, 1]
- 15. Perform local pollination using Equation (24)
- 16. Compute mutation probability factor using Equation (31)
- 17. If $\phi = 1$
- 18. Determine new pollen in local pollination using Equation (30)
- 19. Else
- 20. New pollen is calculated using Equation (25)
- 21. End
- 22. Compute new solutions
- 23. If new solutions are better, update them in the population
- 24. End
- 25. Find the current best solution
- 26. End
- 27. Best solution is the fitness solution



Figure 7. Membership Functions Used.

The proposed optimized IT2FPIDC system utilizes the Matlab/Simulink toolbox for the interval type-2 fuzzy logic system, which has been updated based on the proposal made by Taskin and Kumbasar [49]. This toolbox is employed to initialize the internal configuration of the system.

Correspondingly, three parameters $(l_{i_j}, c_{i_j} \text{ and } r_{i_j})$ are employed to describe the T1FSs of the T1FPIDC as shown in Figure 6b Where i = 1, 2 and j = 1, 2, 3.

5.2.2. The Optimization of the T1FPIDC Cascade Structure Using the RIP ALGORITHM

In order to minimize the suggested performance measure, the MFP optimization algorithms are employed to optimize the parameters of the preceding MFs and the scale factors for both inner and outer controllers. Table 3 shows the three parameters for each input that define the three antecedent MFs of T1FLC (N, Z, and P).

(a)
$$l_{i_1}$$
, c_{i_1} , r_{i_1} for N,
(b) l_{i_2} , c_{i_2} , r_{i_2} for Z,
(c) l_{i_2} , c_{i_3} , r_{i_3} for P

where i = 1, 2. For the T1FPIDC two-input design, the previous MF has a total of 18 structural parameters that need to be optimized, since each input has 9 parameters. Additionally, four scale factors for both the input and output of the T1FPIDC must also be optimized. Therefore, a collection of 22 factors must be optimized using MFP to minimize the performance criterion. The inner loop's optimization variables are defined as follows:

$$x_{T1FPIDC_{\text{inner loop}}} = (l_{1_1}, c_{1_1}, r_{1_1} l_{1_2}, c_{1_2}, r_{1_2} l_{1_3}, c_{1_3}, r_{1_3}, l_{2_1}, c_{2_1}, r_{2_1} l_{2_2}, c_{2_2}, r_{2_2} l_{2_3}, c_{2_3}, r_{2_3}, K_{P_2}, K_{D_2}, K_{I_2}, K_{A_2})$$
(45)

Ε/ΔΕ	Ν	Z	Р
Ν	$N(\zeta_1)$	NM (ζ_2)	$Z(\zeta_3)$
Z	NM (ζ_4)	$Z(\zeta_5)$	PM (ζ_6)
Р	$Z(\zeta_7)$	$PM(\zeta_8)$	$P(\zeta_8)$

Table 3. Rule base of T1FLC and IT2FLC with weights.

The same variables that are used in the inner loop are also optimized using MFP in the outer loop. The optimization process for the T1FPIDC is divided into two stages. In the first stage, MFP is used to optimize the inner loop controller parameters guided by the reference trajectory generated by the outer loop controller. In the second stage, the outer loop controller parameters are optimized using MFP, considering the desired reference trajectory specified by the designer, and the parameters of the inner controller are determined accordingly. Both stages aim at minimizing the performance criterion. The parameters of the preceding MFs of both the outer and inner control loops are optimized in order to have normal convex T1FSs, according to the constraints provided by the following equations:

$$c_{i_1} < c_{i_2} < c_{i_3} \tag{46}$$

$$l_{i_1} < c_{i_1} < r_{i_1} \tag{47}$$

$$l_{i_2} < c_{i_2} < r_{i_2} \tag{48}$$

$$l_{i_3} < c_{i_3} < r_{i_3} \tag{49}$$

5.2.3. The Optimization of IT2FPIDC Cascade Structure Using Meta-Heuristic Optimization Algorithms

To reduce the specified performance criterion, the MFP is employed to optimize the scale factors and parameters of the preceding MFs for both inner and outer controllers of IT2FPIDC. Each input of the three previous MFs of IT2FPIDLC (N, Z, and P) is defined with five parameters. These parameters are:

(a)
$$L_{u_{i1}}$$
, $L_{l_{i1}}$, C_{i_1} , $R_{l_{i1}}$, $R_{u_{i1}}$ for N,
(b) $L_{u_{i2}}$, $L_{l_{i2}}$, C_{i_2} , $R_{l_{i2}}$, $R_{u_{i2}}$ for Z,
(c) $L_{u_{i3}}$, $L_{l_{i3}}$, C_{i_3} , $R_{l_{i3}}$, $R_{u_{i3}}$ for P.

where i = 1, 2. For the IT2FPIDC two-input design, the legacy MF has a total of 30 parameters that need to be optimized, since each input has 15 parameters. This is more than the 18 structure parameters of the T1FPIDC, meaning that the IT2FPIDC has an additional

design degree of freedom compared to the T1FPIDC. As with the T1FPIDC, the IT2FPIDC also consists of four scaling factors for both input and output that require optimization. Therefore, a total of 34 parameters need to be optimized using MFP to minimize the performance criterion. Unlike T1FPIDC, the rule base and resulting MFs are not optimized in the IT2FPID controller design, as the goal is to demonstrate the efficiency of the additional DOF of IT2FSs produced by the FOU in IT2FSs in a closed-loop system performance. Therefore, the optimization variables for MFP are defined as:

$$x_{IT2FPIDC_{\text{inner loop}}} = (L_{u_{11}}, L_{l_{11}}, C_{1_1}, R_{l_{11}}, R_{u_{11}}, L_{u_{12}}, L_{l_{12}}, C_{1_2}, R_{l_{12}}, R_{u_{12}}, L_{u_{13}}, L_{l_{13}}, C_{1_3}, R_{l_{13}}, R_{u_{13}}, L_{u_{21}}, L_{l_{21}}, C_{2_1}, R_{l_{21}}, L_{u_{22}}, L_{u_{22}}, C_{2_2}, R_{l_{22}}, R_{u_{22}}, L_{u_{23}}, L_{l_{23}}, C_{2_3}, R_{l_{23}}, R_{u_{23}}, K_{P_2}, K_{D_2}, K_{I_2}, K_{a_2})$$
(50)

The same variables as for the inner control loop are also subject to optimization for the outer control loop. Furthermore, the weighting element ζ_n is considered as 1 for all rules, just similar to T1FPIDC, and is not optimized. The optimization process for IT2FPIDC follows similar approach as described for T1FPIDC. In order to achieve normal convex IT2FSs, the parameters of the preceding outer and inner loop MFs are optimized according to the limitations specified in the subsequent equations:

$$L_{u_{i1}} < L_{l_{i1}} < C_{i_1} < R_{l_{i1}} < R_{u_{i1}}$$
(51)

$$C_{i_1} < C_{i_2} < C_{i_3} \tag{52}$$

$$L_{u_{i1}} < L_{l_{i1}} < C_{i_1} < R_{l_{i1}} < R_{u_{i1}}$$
(53)

$$L_{u_{i2}} < L_{l_{i2}} < C_{i_2} < R_{l_{i2}} < R_{u_{i2}}$$
(54)

$$L_{u_{i3}} < L_{l_{i3}} < C_{i_3} < R_{l_{i3}} < R_{u_{i3}}$$
(55)

The pseudo codes for MFP methods guided by IT2PIDC and T1FPIDC are given in Table 3.

5.2.4. Configuration Values for Optimization Algorithms

This research used specific values for three parameters—the number of nests, the step size scaling factor, and the switching attribute—which were selected through rigorous experimentation. The authors of the study explain that in a system with multiple solutions, the nests (i.e., potential solutions) are located at different local optima. Therefore, if there are more nests than local optima, the algorithm used in the study can find all of the optima at the same time. This is particularly important for multi-objective and multi-modal optimization problems, which are the focus of this research. The MFP algorithm was implemented in Matlab and tested in 60 different runs with different initial test solutions. The simulations were carried out using Matlab R2013a on a 2.4 GHz processor with 8 GB RAM. The authors also provide a Simulink diagram for implementing the optimized controllers in cascade form, which applies to both T1FPIDC and IT2FPID. The initial angles for the arm and pendulum in the simulations were 0 and 0.97 degrees, respectively.

6. Results and Discussion

This section contains the simulations and test outcomes, including the optimized antecedent membership functions and the optimal scaling factor values for both optimized T1FPIDC and optimized IT2FPIDC. The convergence properties of the proposed optimization algorithms are also discussed. Furthermore, the results include the analysis of stabilization control, trajectory tracking control, disturbance rejection, and validation of the simulation results against the real-time results.

6.1. Simulation Results

The optimization of the inner and outer controllers aimed to reduce the suggested cost function. Figure 8 and Figure 10 display the optimized antecedent membership functions for the inner loop E_2 and ΔE_2 , as well as the outer loop E_1 and ΔE_1 , for T1FPIDC and IT2FPIDC using the MFP algorithm. The weighting factor γ was set to 1 for these optimizations. Meanwhile, Figure 9 and Figure 11 show the optimized antecedent membership functions for the inner loop E_2 and ΔE_2 , as well as the outer loop E_1 and ΔE_1 , for T1FPIDC and IT2FPIDC using the MFP algorithm. The weighting factor γ was set to 1 for these optimizations. Meanwhile, Figure 9 and Figure 11 show the optimized antecedent membership functions for the inner loop E_2 and ΔE_2 , as well as the outer loop E_1 and ΔE_1 , for T1FPIDC and IT2FPIDC using the MFP algorithm with a weighting factor of $\gamma = 1.5$.



Figure 9. Optimized antecedent MFs of T1FPIDC (MFP-based), $\gamma = 1.5$.

To reduce the suggested cost function, the inner and outer controllers were optimized. The optimized antecedent membership functions (MFs) for the inner loop E_2 and ΔE_2 and outer loop E_1 and ΔE_1 of T1FPIDC and IT2FPIDC, using the MFP algorithm, were presented in Figures 8 and 10, respectively, by setting the weighting factor $\gamma = 1$. Similarly, Figures 9 and 11 show the optimized antecedent MFs for the inner loop E_2 and ΔE_2 of T1FPIDC and IT2FPIDC, as well as the outer loop E_1 and ΔE_1 , using the MFP algorithm with $\gamma = 1.5$.



Figure 11. Optimized antecedent MFs of IT2FPIDC (MFP-based), $\gamma = 1.5$.

Two statistical simulations were conducted to evaluate the convergence properties of the proposed T1FPIDC. The acquisition value's standard deviation and mean were used as indicators of speed and accuracy, respectively. Results presented in Figure 12a,b indicate that both MFP-based IT2FPIDC and FP-based T1FPIDC maintain a stable average cost value for weighting factors $\gamma = 1$ and $\gamma = 1.5$ under the same simulation conditions and performance criteria. However, MFP-based IT2FPIDC controllers outperformed MFP-based T1FPIDC in terms of mean and cost value, demonstrating better accuracy for both $\gamma = 1$ and $\gamma = 1.5$.



Figure 12. Tendency of convergence of mean value of performance criterion with The optimized scaling factors for the optimized cascade IT2FPIDC and optimized cascade T1FPIDC are shown in Table 4.

Table 5 displays the optimal runtime for T1FPIDCs and IT2FPIDCs after 100 iterations. The results reveal that IT2FPID has a longer runtime than T1FPIDC for each optimization algorithm, which is attributed to the greater number of optimization parameters in IT2FPIDC than T1FPIDC.

Controllers			Outer Controller				Inner Controller			
	γ	K_{P_1}	K_{I_1}	K_{D_1}	K_{a_1}	K_{P_2}	K_{I_2}	K_{D_2}	K_{a_2}	
IT2FPIDC	1	7.015	5.137	0.0019	1.662	0.8275	0.6883	0.0205	0.4062	
	1.5	6.914	4.863	0.0074	2.186	0.8135	0.7008	0.0393	0.3916	
T1FPIDC	1	6.517	4.593	0.0138	1.615	0.8519	0.6583	0.0078	0.3653	
	1.5	6.103	4.916	0.0342	2.217	0.7994	0.7182	0.0062	0.4466	

Table 4. Optimized scaling factors for cascade IT2FPIDC and T1FPIDC.

 Table 5. Average computational time for IT2FPIDCs and T1FPIDCs.

IT2FP	PIDC	T1FPIDC		
γ=1	γ =1.5	γ =1	γ =1. 5	
174.38	209.08	142.94	157.06	

6.1.2. Stabilization Control Using Optimized Cascade IT2FPIDC

This section presents an analysis of the stabilization control of RIP using two optimized cascade controllers, namely IT2FPIDC and T1FPIDC. The designated swing angle was established as zero, which is considered to be the unstable upright position. Figure 13 displays the best results obtained for various parameters, such as pendulum angle, arm angle, outer loop output, and inner loop output for a specific value of $\gamma = 1$. To assess the impact of changing the value of γ , it was increased to 1.5, and the corresponding results

are presented in Figure 14. The simulation results for 60 runs with different values of γ are summarized in Table 6. Based on the results presented in Table 6, it can be concluded that the optimized cascade IT2FPIDC designed using the MFP method outperforms the optimized IT2FPIDC designed using GA and PSO proposed in [14] in various performance indices, such as rise time t_r , settling time t_s (less than 2%), steady-state error E_{ss} , and maximum overshoot M_p .



Figure 13. Comparative simulation results for the MFP-based cascade IT2FPIDC and the MFP basic cascade T1FPIDC with $\gamma = 1$, (**a**) pendulum angle, (**b**) arm angle, (**c**) Outer loop output (**d**) Inner loop output.

Table 6.	Comparative simulation	analysis results for	r optimized cascad	le IT2FPIDC and	optimized
cascade	T1FPIDC.				

		IT2FPIDC				T1FPI	DC		
γ			1	1	.5	1	l	1	.5
$t_r(Sec)$	Pendulum (α)	Best	Average	Best	Average	Best	Average	Best	Average
	Arm (ϕ)	0.31	0.39	0.33	0.42	0.51	0.65	0.45	0.67
$t_s(Sec)$	Pendulum (α)	0.14	0.18	0.12	0.15	0.21	0.29	0.17	0.25
	Arm (ϕ)	1.05	1.36	0.65	0.91	1.03	1.48	1.01	1.52
E _{ss}	Pendulum (α)	1.34	1.67	1.08	1.47	1.51	2.19	1.44	2.25
	Arm (ϕ)	0.0065	0.0066	0.0068	0.0071	0.0072	0.0078	0.0072	0.0082
$M_p(\%)$	Pendulum (α)	0.0066	0.0069	0.0073	0.0078	0.0072	0.0094	0.0081	0.0084
	Arm (ϕ)	0.09	0.11	0.15	0.21	0.10	0.15	0.16	0.27
Coast		1.10	1.12	0.95	1.38	1.44	2.33	0.99	1.72



Figure 14. Comparative simulation results for the MFP-based cascade IT2FPIDC and the MFP-base cascade T1FPIDC with $\gamma = 1.5$, (a) pendulum angle, (b) arm angle, (c) Outer loop output (d) Inner loop output.

6.1.3. Disturbances Rejection Analysis in Stabilization Control for MFP-Based Cascade IT2FPIDC and T1FPIDC

To test the durability of the proposed IT2FPIDC/T1FPIDC optimized cascade, both internal and external disruptions are incorporated into the RIP system. A load with a length of 0.1965 m and a mass of 0.0741 kg is also added to the pendulum. Moreover, the output of the procedure (feedback) is subjected to random noise of 0.00739 power and 10% parameter value alterations, as depicted in Figure 15. The sway angle, arm angle, outer loop output, and inner loop output under disruptive circumstances are illustrated in Figure 16, demonstrating the effectiveness and resilience of the IT2FPIDC. Although there are instances of oscillations in the arm and pendulum angles, ranging from -0.1 to 0.2 degrees and -0.04 to 0.04 degrees, respectively, which are absent in the disturbance-free simulation, all controllers successfully manage to swing the pendulum into an upright position and stabilize it in an unstable state with minimal vibration. Table 7 displays the controller's performance in the event of faults, clearly indicating the superiority of IT2FPIDC over T1FPIDC in the presence of interference.



Figure 15. MFP-based Cascade IT2FPIDC/T1FPIDC including the source of noise.



Figure 16. Comparativesimulation results for disturbance rejection ability of MFP-base cascade IT2FPIDC and MFP-base cascade T1FPIDC. (**a**) pendulum angle, (**b**) arm angle, (**c**) Outer loop output (**d**) Inner loop output.

Table 7.	Comparative	simulation	analysis	results	for	disturbance	rejection	ability	of	optimized
cascade I	T2FPIDC and o	optimized c	ascade T	IFPIDC.						

T1FPIDC		IT2FPIDC	T1FPIDC
t_r (Sec)	Pendulum (α)	0.45	0.62
	Arm (ϕ)	0.20	0.23
t_s (Sec)	Pendulum (α)	1.16	1.24
	Arm (ϕ)	1.42	1.68
E_{ss}	Pendulum (α)	0.0074	0.0087
	Arm (ϕ)	0.0077	0.0085
<i>M_p</i> (%)	Pendulum (α)	0.17	0.21
	Arm (ϕ)	2.41	3.27
Coast		0.186	0.184

6.1.4. Trajectory Tracking Control for Optimized Cascade T1FPIDC

One of the key aims of RIP control is to manage the trajectory, which indicates how well the controller can guide the RIP arm to follow a specific time-dependent path while the pendulum remains in an unstable state. Achieving this objective involves creating a controller $\tau \epsilon R$ that can keep both the trajectory error ϕ and the sway angle within certain limits [50]. The controller design should ensure that these limits remain uniform over time. The designed controller guarantees the following:

$$\left\| \begin{array}{c} \widetilde{\phi}(t_0) \\ \alpha(t_0) \end{array} \right\| \le a \Rightarrow \left\| \begin{array}{c} \widetilde{\phi}(t) \\ \alpha(t) \end{array} \right\| \le b, \forall t \ge t_0 + T \tag{56}$$

were *a* and *b* are constant, T > 0 and $\phi = \phi_d - \phi$. In this research, the intended path function utilized is a time-dependent square function. This function is both smooth and has a derivative. Furthermore, ϕ_d , $\dot{\phi}_d$, and $\ddot{\phi}_d$ are constrained for all $t \ge 0$. The output function is defined as:

$$y = \phi - \alpha \tag{57}$$

A thorough description and development of RIP trajectory tracking control can be located in [51,52].

Figure 17 illustrates the optimal simulation outcomes of RIP trajectory tracking control for the optimized cascade IT2FPIDC/T1FPIDC at $\gamma = 1$ for arm and weave angles in Figure 17a,c, respectively. Furthermore, the simulation result for the RIP trajectory tracking control using the optimized cascade IT2FPIDC/T1FPIDC at $\gamma = 1.5$ is shown in Figure 17b,d for arm and weave angles, respectively. Observing both Figure 17a,b, it is evident that the MFP-based cascade IT2FPIDC achieves the target value rapidly in comparison to the MFP-based cascade T1FPIDC. The detailed performance comparison is presented in Table 8.



Figure 17. Trajectory tracking control by optimized cascade IT2FPIDC/T1FPIDC. (**a**) pendulum angle, (**b**) arm angle, (**c**) Outer loop output (**d**) Inner loop output.

6.2. Experimental Validation

This section presents experimental outcomes that validate the simulation findings. These outcomes comprise of stabilization control, trajectory tracking control, and clutter rejection analysis, all of which were obtained at $\gamma = 1$ for both MFP-based IT2FPIDC and MFP-based T1FPIDC. The reason for choosing the MFP-based controllers and weighting factor $\gamma = 1$ for validation is because they exhibited superior action compared to other controllers in the simulation studies.

Performance Measures	Coordinates	IT2FPIDC		T1FPIDC			
γ		1	1.5	1	1.5		
t_r (Sec)	Pendulum (α)	0.37	0.36	0.49	0.47		
	Arm (ϕ)	0.29	0.25	0.38	0.35		
t_s (Sec)	Pendulum (α)	0.73	0.84	0.99	1.11		
	Arm (ϕ)	0.95	1.05	1.23	1.38		
t_d (Sec)	Pendulum (α)	0.46	0.47	0.51	0.53		
	Arm (ϕ)	0.34	0.36	0.39	0.42		
E_{ss}	Pendulum (α)	0	0	0	0		
	Arm (ϕ)	0	0	0	0		
<i>M</i> _p (%)	Pendulum (α)	1.42	1.53	1.88	1.96		
	Arm (ϕ)	2.15	4.99	4.56	5.98		

Table 8. Comparative trajectory tracking results for optimized cascade T1FPIDC (Simulation).

6.2.1. Stabilization Control (Experiments)

Figure 18a,b depicts the experimental results in the real world, comparing the proposed MFP-based optimized cascade IT2FPIDC, the MFP-based optimized cascade T1FPIDC, and a traditional energy-based controller for pendulum and arm angles, where the equilibrium state is to be established. The actual results obtained from the physical system indicate initial oscillations, which arise due to the required swinging motion needed to position the pendulum in a vertically unstable state. The two controllers, MFP-based optimized cascade IT2FPIDC and MFP-based optimized cascade T1FPIDC, demonstrate a notable level of resilience. However, the conventional energy-based controller exhibits multiple oscillations before achieving stability. The conventional energy-based controllers also require more time to reach a steady state compared to MFP-based optimized cascade IT2FPIDC and MFP-based optimized cascade T1FPIDC. Furthermore, MFP-based optimized IT2FPIDC and MFP-based optimized cascade T1FPIDC.



Figure 18. Cont.



Figure 18. Experimental results for stabilization control.

6.2.2. Trajectory Tracking Control (Experiments)

A reference signal in the form of a square wave with an amplitude of $\pm 40^{\circ}$ is utilized to guide the arm's movement. Figure 19a,b depict the reaction of the arm angle and pendulum angle, respectively. The optimized cascade MFPs based on IT2FPIDC and T1FPIDC exhibit a smooth ability to direct the motor shaft to follow the desired trajectories while also maintaining the pendulum's upright stance (approximately 0). Moreover, the systems are capable of continuously compensating for any mechanical disturbances, such as friction.

6.2.3. Disturbance Rejection Analysis (Experiments)

To test the resilience of the proposed cascade-optimized MFP-based IT2FPIDC and T1FPIDC controllers, internal noise and external disturbances were introduced to the system in real-world experiments. A pendulum load with a length of 0.1685 m and a mass of 0.0635 kg was added, and the process output (feedback) was subjected to random noise with a power of 0.00634 and 10% parameter value changes. Figure 20a,b display the empirical findings pertaining to the trajectory control of the RIP system with the arm and pendulum angles. The experimental and simulation outcomes showed that the optimized cascade IT2FPIDC outperformed the optimized cascade T1FPIDC in terms of load disturbances, parameter variations, and noise effects. Although there were slight variations in the arm and pendulum angles compared to the experimental results without perturbations, both controllers effectively directed the movement of the motor shaft to track the intended paths while simultaneously stabilizing the pendulum in an upright stance (around 0) and compensating for mechanical and introduced disturbances with minimal vibrations. Grounded on the experimental outcomes, it can be inferred that the optimized cascade IT2FPIDC improved the performance of the cascade-optimized T1FPIDC and the conventional energy-based controller on the RIP. Table 9 demonstrates the superior performance of the optimized cascade IT2FPIDC over the optimized cascade T1FPIDC in the presence of faults without a faulted state.



(b)

Figure 19. Experimental result for trajectory tracking control (comparing optimized cascade MFP-based IT2FPIDC and T1FPIDC).

Table 9. Comparative trajectory tracking results for optimized cascade MFP-based IT2FPIDC and T1FPIDC (Experiment).

Controllers	Disturbances	t_r (sec)	t_s (sec)	t_d (sec)	M_p %	E_{ss}
IT2FPIDC	No disturbance	1.60	5.05	1.45	9.94	0
	With disturbance	1.85	6.80	1.62	10.3	0
T1FPIDC	No disturbance	2.07	7.65	1.36	12.1	0
	With disturbance	2.34	8.74	1.65	13.0	0



Figure 20. Experimental result for disturbed trajectory tracking control (comparing optimized cascade MFP-based IT2FPIDC and T1FPIDC).

7. Conclusions

The main goal of this study is to explore the potential benefits of utilizing the MFP algorithm approach to design IT2FPIDC and to examine the merits of IT2FLC over T1FLC. The study compares the optimization performances of IT2FPIDC in a cascade structure with those of T1FPIDC in a cascade structure. The comparison is based on four performance measures: steady-state error (E_{ss}), settling time (t_s), rise time (t_r), and maximum overshoot (M_p).

The results of the study indicate that using the MFP-based design approach can produce high-quality solutions with a significantly reduced computing time of 196.33 min. Consequently, this approach can be used for designing more intricate IT2FLCs with an increased number of input/output parameter applications, requiring optimized results in a shorter time frame. To demonstrate the benefits of using FOU in IT2FPIDC, the study kept the rule-based, rule-weighted elements, and the resulting MFs factors fixed.

The study only optimized the scaling factors and the parameters of the preceding MFs parameters, and this approach was also applied to the T1FPIDC for an equitable comparison. The main limitation of the present study is limited number of MFs used and the time taken for the multi-objective optimization cost function. The experimental and simulation outcomes indicated that the optimized cascade IT2FPIDC outperformed the optimized cascade T1FPIDC in terms of the four performance measures (E_{ss} , t_s , t_r , and M_p) regardless of the optimization method used.

For instance, the optimized cascade IT2FPIDC exhibited a progress between 6.1% to 33.3%, 5.7% to 35.2%, and 6.6% to 20.8% in t_r , t_s , and E_{ss} , in the corresponding order, when compared to its T1FPIDC counterpart in the presence of disturbances. This is possibly a result of FOU present in IT2FPIDC, that enables it to handle complex systems which T1FPIDC cannot control with the same set of rules.

In the future, a more complex benchmark will be used to test the proposed controller. Also, optimization of general type-2 fuzzy logic controller (GT2FLC) using MFP should be tested. This will enable us to compare the performance of T1FLC, IT2FLC, and GTFLC.

Funding: This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444).

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

- 1. Rigatos, G.G. Modelling and control for intelligent industrial systems. In *Adaptive Algorithms in Robotics and Industrial Engineering;* Springer: New York, NY, USA, 2011.
- Hamza, M.F.; Yap, H.J.; Choudhury, I.A. Recent advances on the use of meta-heuristic optimization algorithms to optimize the type-2 fuzzy logic systems in intelligent control. *Neural Comput. Appl.* 2015, 28, 979–999. [CrossRef]
- 3. Zhang, Y.; Li, S.; Liao, L. Near-optimal control of nonlinear dynamical systems: A brief survey. *Annu. Rev. Control.* **2019**, 47, 71–80. [CrossRef]
- 4. Liu, D.; Xue, S.; Zhao, B.; Luo, B.; Wei, Q. Adaptive Dynamic Programming for Control: A Survey and Recent Advances. *IEEE Trans. Syst. Man, Cybern. Syst.* 2020, *51*, 142–160. [CrossRef]
- Mittal, K.; Jain, A.; Vaisla, K.S.; Castillo, O.; Kacprzyk, J. A comprehensive review on type 2 fuzzy logic applications: Past, present and future. *Eng. Appl. Artif. Intell.* 2020, 95, 103916. [CrossRef]
- Cervantes, L.; Caraveo, C.; Castillo, O. Performance Comparison between Type-1 and Type-2 Fuzzy Logic Control Applied to Insulin Pump Injection in Real Time for Patients with Diabetes. *Mathematics* 2023, 11, 730. [CrossRef]
- Mendel, J.M. General Type-2 Fuzzy Logic Systems Made Simple: A Tutorial. *IEEE Trans. Fuzzy Syst.* 2013, 22, 1162–1182. [CrossRef]
- 8. Liang, Q.; Mendel, J. Interval type-2 fuzzy logic systems: Theory and design. IEEE Trans. Fuzzy Syst. 2000, 8, 535–550. [CrossRef]
- Mendel, J.M.; John, R.I.; Liu, F. Interval Type-2 Fuzzy Logic Systems Made Simple. *IEEE Trans. Fuzzy Syst.* 2006, 14, 808–821. [CrossRef]
- Castillo, O.; Amador-Angulo, L.; Castro, J.R.; Garcia-Valdez, M. A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems. *Inf. Sci.* 2016, 354, 257–274. [CrossRef]
- Montazeri-Gh, M.; Yazdani, S. Application of interval type-2 fuzzy logic systems to gas turbine fault diagnosis. *Appl. Soft Comput.* 2020, 96, 106703. [CrossRef]
- 12. Ochoa, P.; Castillo, O.; Soria, J. Optimization of fuzzy controller design using a Differential Evolution algorithm with dynamic parameter adaptation based on Type-1 and Interval Type-2 fuzzy systems. *Soft Comput.* **2020**, *24*, 193–214. [CrossRef]
- 13. Valdez, F.; Castillo, O.; Cortes-Antonio, P.; Melin, P. A survey of Type-2 fuzzy logic controller design using nature inspired optimization. *J. Intell. Fuzzy Syst.* 2020, *39*, 6169–6179. [CrossRef]
- 14. Hamza, M.F.; Yap, H.J.; Choudhury, I.A. Genetic Algorithm and Particle Swarm Optimization Based Cascade Interval Type 2 Fuzzy PD Controller for Rotary Inverted Pendulum System. *Math. Probl. Eng.* **2015**, 2015, 695965. [CrossRef]
- 15. Zhang, Z.; Wang, T.; Chen, Y.; Lan, J. Design of type-2 Fuzzy Logic Systems Based on Improved Ant Colony Optimization. *Int. J. Control. Autom. Syst.* **2019**, *17*, 536–544. [CrossRef]
- Leon-Garza, H.; Hagras, H.; Pena-Rios, A.; Conway, A.; Owusu, G. A big bang-big crunch type-2 fuzzy logic system for explainable semantic segmentation of trees in satellite images using hsv color space. In Proceedings of the 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Glasgow, UK, 19–24 July 2020; IEEE: New York, NY, USA, 2020.
- 17. Mohammadikia, R.; Aliasghary, M. Design of an interval type-2 fractional order fuzzy controller for a tractor active suspension system. *Comput. Electron. Agric.* 2019, *167*, 105049. [CrossRef]

- 18. Kiani, M.; Mohammadi, S.M.A.; Gharaveisi, A.A. A bacterial foraging optimization approach for tuning type-2 fuzzy logic controller. *Turk. J. Electr. Eng. Comput. Sci.* 2013, 21, 263–273. [CrossRef]
- 19. Türk, S.; Deveci, M.; Özcan, E.; Canıtez, F.; John, R. Interval type-2 fuzzy sets improved by Simulated Annealing for locating the electric charging stations. *Inf. Sci.* 2021, 547, 641–666. [CrossRef]
- Almaraashi, M.; Hedar, A.-R. Optimization of interval type-2 fuzzy logic systems using tabu search algorithms. In Proceedings of the 2014 Sixth World Congress on Nature and Biologically Inspired Computing (NaBIC 2014), Porto, Portugal, 30 July–1 August 2014; IEEE: New York, NY, USA, 2014.
- Ray, P.K.; Paital, S.R.; Mohanty, A.; Foo, Y.S.E.; Krishnan, A.; Gooi, H.B.; Amaratunga, G.A.J. A Hybrid Firefly-Swarm Optimized Fractional Order Interval Type-2 Fuzzy PID-PSS for Transient Stability Improvement. *IEEE Trans. Ind. Appl.* 2019, 55, 6486–6498. [CrossRef]
- 22. Amador-Angulo, L.; Castillo, O. A new fuzzy bee colony optimization with dynamic adaptation of parameters using interval type-2 fuzzy logic for tuning fuzzy controllers. *Soft Comput.* **2016**, *22*, 571–594. [CrossRef]
- 23. Hamza, M.F.; Yap, H.J.; Choudhury, I.A. Cuckoo search algorithm based design of interval Type-2 Fuzzy PID Controller for Furuta pendulum system. *Eng. Appl. Artif. Intell.* **2017**, *62*, 134–151. [CrossRef]
- 24. De (Maity), R.R.; Mudi, R.K.; Dey, C. Nature-inspired and hybrid optimization algorithms on interval Type-2 fuzzy controller for servo processes: A comparative performance study. *SN Appl. Sci.* **2020**, *2*, 1292.
- Kumar, A.; Kumar, V. Performance analysis of optimal hybrid novel interval type-2 fractional order fuzzy logic controllers for fractional order systems. *Expert Syst. Appl.* 2018, 93, 435–455. [CrossRef]
- 26. Chiroma, H.; Shuib, N.L.M.; Muaz, S.A.; Abubakar, A.I.; Ila, L.B.; Maitama, J.Z. A Review of the Applications of Bio-inspired Flower Pollination Algorithm. *Procedia Comput. Sci.* **2015**, *62*, 435–441. [CrossRef]
- 27. Wu, D.; Mendel, J.M. Recommendations on designing practical interval type-2 fuzzy systems. *Eng. Appl. Artif. Intell.* **2019**, *85*, 182–193. [CrossRef]
- Hamza, M.F.; Yap, H.J.; Choudhury, I.A.; Isa, A.I.; Zimit, A.Y.; Kumbasar, T. Current development on using Rotary Inverted Pendulum as a benchmark for testing linear and nonlinear control algorithms. *Mech. Syst. Signal Process.* 2019, 116, 347–369. [CrossRef]
- 29. Nguyen, N.P.; Oh, H.; Kim, Y.; Moon, J. A nonlinear hybrid controller for swinging-up and stabilizing the rotary inverted pendulum. *Nonlinear Dyn.* **2021**, *104*, 1117–1137. [CrossRef]
- Khanesar, M.A.; Kaynak, O. Recurrent interval type-2 neuro-fuzzy control of an electro hydraulic servo system. In Proceedings of the 2016 IEEE 14th International Workshop on Advanced Motion Control (AMC), Auckland, New Zealand, 22–24 April 2016; IEEE: New York, NY, USA, 2016.
- Karnik, N.N.; Mendel, J.M. Introduction to type-2 fuzzy logic systems. In Proceedings of the 1998 IEEE International Conference on Fuzzy Systems Proceedings. IEEE world Congress on Computational Intelligence (Cat. No. 98CH36228), Anchorage, AK, USA, 4–9 May 1998; IEEE: New York, NY, USA, 1998.
- 32. Mendel, J.M. Advances in type-2 fuzzy sets and systems. Inf. Sci. 2007, 177, 84–110. [CrossRef]
- Mendel, J.; Hagras, H.; Tan, W.W.; Melek, W.W.; Ying, H. Introduction to Type-2 Fuzzy Logic Control: Theory and Applications; John Wiley & Sons: New York, NY, USA, 2014.
- Tan, W.W.; Chua, T.W. Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions (Mendel, J.M.; 2001) [book review]. *IEEE Comput. Intell. Mag.* 2007, 2, 72–73. [CrossRef]
- 35. Hamza, M.F.; Yap, H.J.; Choudhury, I.A.; Isa, A.I. Application of Kane's Method for Dynamic Modeling of Rotary Inverted Pendulum System. In Proceedings of the 2016 MNTMSim Conference, Kuala Lumpur, Malaysia, 1–3 March 2016.
- 36. Yang, X.-S. Flower pollination algorithm for global optimization. In Proceedings of the Unconventional Computation and Natural Computation: 11th International Conference, UCNC 2012, Orléan, France, 3–7 September 2012; Springer: New York, NY, USA, 2012.
- 37. Yang, X.-S.; Karamanoglu, M.; He, X. Flower pollination algorithm: A novel approach for multiobjective optimization. *Eng. Optim.* **2014**, *46*, 1222–1237. [CrossRef]
- 38. Pavlyukevich, I. Lévy flights, non-local search and simulated annealing. J. Comput. Phys. 2007, 226, 1830–1844. [CrossRef]
- 39. Mantegna, R.N. Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes. *Phys. Rev. E* 1994, 49, 4677–4683. [CrossRef] [PubMed]
- 40. Rajeswari, C.; Santhi, M. Modified flower pollination algorithm for optimizing FOPID controller and its application with the programmable n-level inverter using fuzzy logic. *Soft Comput.* **2021**, *25*, 2615–2633. [CrossRef]
- 41. Kumar, R.; Sikander, A. A novel load frequency control of multi area non-reheated thermal power plant using fuzzy PID cascade controller. *Sādhanā* **2023**, *48*, 25.
- 42. Campos-Rodríguez, A.; García-Sandoval, J.; González-Álvarez, V. Hybrid cascade control for a class of nonlinear dynamical systems. J. Process. Control. 2019, 76, 141–154. [CrossRef]
- 43. Gaing, Z.-L. A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System. *IEEE Trans. Energy Convers.* **2004**, *19*, 384–391. [CrossRef]
- 44. Hassanzadeh, I.; Mobayen, S. Controller Design for Rotary Inverted Pendulum System Using Evolutionary Algorithms. *Math. Probl. Eng.* **2011**, 2011, 572424. [CrossRef]
- 45. Haupt, R.L.; Haupt, S.E. Practical Genetic Algorithms; John Wiley & Sons: New York, NY, USA, 2014.

- Korkmaz, E.; Akgungor, A.P. A Hybrid Traffic Controller System based on Flower Pollination Algorithm and Type-2 Fuzzy Logic Optimized with Crow Search Algorithm for Signalized Intersections. *Res. Sq.* 2022. [CrossRef]
- Umoh, U.A.; Inyang, U.G.; Nyoho, E.E. Soft Computing for Problem Solving 2019: Proceedings of SocProS. In 2019A Hybrid Framework for Fire Outbreak Detection Based on Interval Type-2 Fuzzy Logic and Flower Pollination Algorithm; Springer: Singapore, 2020; Volume 2, pp. 27–45.
- 48. Patel, H.R.; Shah, V.A. Shadowed Type-2 Fuzzy Sets in Dynamic Parameter Adaption in Cuckoo Search and Flower Pollination Algorithms for Optimal Design of Fuzzy Fault-Tolerant Controllers. *Math. Comput. Appl.* **2022**, *27*, 89. [CrossRef]
- Taskin, A.; Kumbasar, T. An open source Matlab/Simulink toolbox for interval type-2 fuzzy logic systems. In Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence, Cape Town, South Africa, 7–10 December 2015; IEEE: New York, NY, USA, 2015.
- 50. Khalil, H.K.; Grizzle, J. Nonlinear Systems; Prentice Hall: Hoboken, NJ, USA, 1996; Volume 3.
- 51. Aguilar-Avelar, C.; Moreno-Valenzuela, J. A composite controller for trajectory tracking applied to the Furuta pendulum. *ISA Trans.* **2015**, *57*, 286–294. [CrossRef]
- 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 Časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije 2022, 63, 656–675. [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.