

Article

Interval Type-2 Fuzzy PID Controller Using Disassembled Gradational Optimization

Yongzhi Chu ^{1,2,3} , Hasiaoqier Han ^{1,2,*}, Tianjiao Ma ^{1,2}, Mingchao Zhu ^{1,2}, Zhongcan Li ^{1,2}, Zhenbang Xu ^{1,2}  and Qingwen Wu ^{1,2,3}

- ¹ Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, No. 3888, Dong Nanhu Road, Changchun 130033, China; chuyongzhi@ciomp.ac.cn (Y.C.); matianjiao@ciomp.ac.cn (T.M.); zhumingchao@ciomp.ac.cn (M.Z.); lizhongcan20@mailsucas.ac.cn (Z.L.); xuzhenbang@ciomp.ac.cn (Z.X.); wuqw@ciomp.ac.cn (Q.W.)
- ² Center of Materials Science and Optoelectronics Engineering, University of Chinese Academy of Sciences, Beijing 100049, China
- ³ CAS Key Laboratory of On-Orbit Manufacturing and Integration for Space Optics System, Chinese Academy of Sciences, Changchun 130033, China
- * Correspondence: hanhasiaoqier@yahoo.com; Tel.: +86-177-0431-3194

Abstract: This paper presents an interval type-2 fuzzy proportional–integral–derivative (IT2F-PID) controller that is designed using a new disassembled gradational optimization (D-GO) method. A PID controller is first optimized using the D-GO method and then connected to a type-1 fuzzy logic system (T1-FLS). The parameters of the T1-FLS are optimized, and the T1-FLS is blurred into the interval type-2 fuzzy logic system (IT2-FLS). Finally, the IT2F-PID controller is formed. The proposed method is compared with the concurrent and general optimization methods. The simulation results show that the D-GO method reduces the optimization time by over 90% compared with the general method, and decreases the integral-of-time-absolute-error (ITAE) by 30%. Beyond that, compared with the concurrent optimization method, the D-GO method reduces time by over 25%, and the ITAE value by about 95%. In the normal case, model uncertainty, target uncertainty, and external disturbance, the control ability of the IT2F-PID controller designed using the D-GO method is verified via simulations using a nonlinear forced closed-loop system. The results show that the overshoot is reduced by 80% and the fluctuation is reduced by 67% compared with a traditional PID controller and an IT2F-PID controller built using the general method.

Keywords: disassembled gradational optimization method; interval type-2 fuzzy logic system; interval type-2 fuzzy PID controller; uncertain system; forced closed-loop system



Citation: Chu, Y.; Han, H.; Ma, T.; Zhu, M.; Li, Z.; Xu, Z.; Wu, Q. Interval Type-2 Fuzzy PID Controller Using Disassembled Gradational Optimization. *Sensors* **2023**, *23*, 9067. <https://doi.org/10.3390/s23229067>

Academic Editor: Mohammed Chadli

Received: 29 September 2023
Revised: 1 November 2023
Accepted: 3 November 2023
Published: 9 November 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

Conventional proportional–integral–derivative (PID) controllers have a simple structure, convenient adjustment, good stability, and reliable operation. Hence, they have been widely used in industrial control applications such as synchronous generator control, aircraft automatic navigation system control, computer numerical control, machine tool control, robot control, temperature control, and liquid level control [1–5]. Many intelligent controllers are based on conventional PID controllers. However, when the controlled object is a high-order complex system with a time delay, the performance of PID controllers is greatly reduced, and the control ability may even be lost [6,7].

To overcome this problem, many researchers have combined fuzzy logic systems (FLS) with conventional PID controllers [8,9]. Type-1 fuzzy PID (T1F-PID) and interval type-2 fuzzy PID (IT2F-PID) controllers have recently been developed [10,11]. The structure of the T1-FLS is similar to the IT2-FLS, with the IT2-FLS formed by adding uncertainties to the membership function of the T1-FLS [12,13]. Therefore, the IT2-FLS has a stronger control ability than that of the T1-FLS and can effectively control uncertain systems with

inherent modeling nonlinearities and time delays [14,15]. The combination of the IT2-FLS and PID controllers is a current research hotspot [16]. For example, IT2F-PID controllers have been successfully applied to high-precision electro-optical tracking [17], a low-cost microcontroller [18], a highly complex nonlinear double-link robot operating system [19], and a mobile robot [20].

At present, the common structure of IT2F-PID controllers includes cascade and parallel types. The cascade-type IT2F-PID controllers connect the IT2-FLS with a proportional–integral controller in series and perform the proportional–integral operation on the output of the IT2-FLS [21–23]. This method does not control the three parameters of the PID controller separately when establishing the fuzzy control rules. So it does not take advantage of the independent control of the three parameters of the PID controller. The parallel IT2F-PID controller structure combines the IT2-FLS and the PID controller in parallel [6]. Compared with cascade-type controllers, the parallel-type IT2F-PID controller not only retains the benefits of PID controllers but also utilizes the powerful control ability of the IT2-FLS to adjust the control parameters of the PID controller in real time, thus enabling the PID controller to control high-order complex systems. Therefore, the parallel-type structure is more conducive to engineering applications. However, at present, the design of parallel IT2F-PID controllers employs an empirical method, and there is no other optimization approach [24].

By analyzing the optimal design of cascade-type IT2F-PID controllers, it appears that the construction of IT2F-PID controllers requires appropriate input and output factors and membership function parameters [23]. Therefore, the appropriate parameters must be selected for parallel-type IT2F-PID controllers. Several researchers have found that the input and output factors have significant effects on the performance of IT2F-PID controllers [25]. Therefore, on the basis of a fixed membership function, only the input and output factors need to be optimized, which is an efficient method of constructing the controller [26]. Parallel-type IT2F-PID controllers retain the structure of PID controllers, so there are eight parameters to be optimized, namely, the three coefficients of the PID controller and five scaling factors of the IT2-FLS. Cascade-type IT2F-PID controllers usually have only four parameters to be optimized; even so, it is very difficult to optimize four parameters at the same time using regular intelligent optimization algorithms [27]. Optimizing the eight parameters, therefore, presents a real difficulty in constructing a parallel-type IT2F-PID controller.

In this paper, a practical disassembled gradational optimization (D-GO) method is proposed for the design of a parallel-type IT2F-PID controller. The main feature of this method is the optimization of eight parameters in two steps. The first step is to optimize the proportional, integral, and differential coefficients of the PID controller. The second step then optimizes the scaling factor of error input, error change rate, proportional coefficient output, integral coefficient output, and differential coefficient output.

The merit of this method is that the optimal PID controller is determined in the first step, so the IT2-FLS only controls the increment of the proportional, integral, and differential coefficients. Compared with the general optimization method, which uses an intelligent optimization algorithm to optimize the parameters at the same time, the optimization range of the parameters is greatly reduced, which significantly reduces the optimization time. The D-GO method provides support for the IT2F-PID controller to realize online real-time parameter adjustment and control time-varying nonlinear complex systems. This paper compares the D-GO method with the concurrent and general optimization methods. Under the same conditions, the controller built using the D-GO method achieves better control performance and has a shorter optimization time. Through simulations, the validity of the parallel-type IT2F-PID controller designed using the D-GO method is verified using a forced closed-loop control system.

The balance of this paper is organized as follows. In Section 2, the IT2F-PID controller is introduced and the process of constructing a parallel IT2F-PID controller with the D-GO method is described in detail. Section 3 presents the forced closed-loop control system and

the corresponding IT2F-PID controller constructed using the D-GO method. The simulation results are analyzed in Section 4, and the conclusions of this study are summarized in Section 5.

2. Interval Type-2 Fuzzy PID Controller

The parallel-type IT2F-PID controller is composed of an upper-level IT2-FLS controller and a lower-level classical PID controller. The upper intelligent controller provides a parameter change selection mechanism for the lower classical PID controller. The parameters of the lower classical PID controller are adjusted using the upper controller. The structure of the parallel IT2F-PID controller is shown in Figure 1.

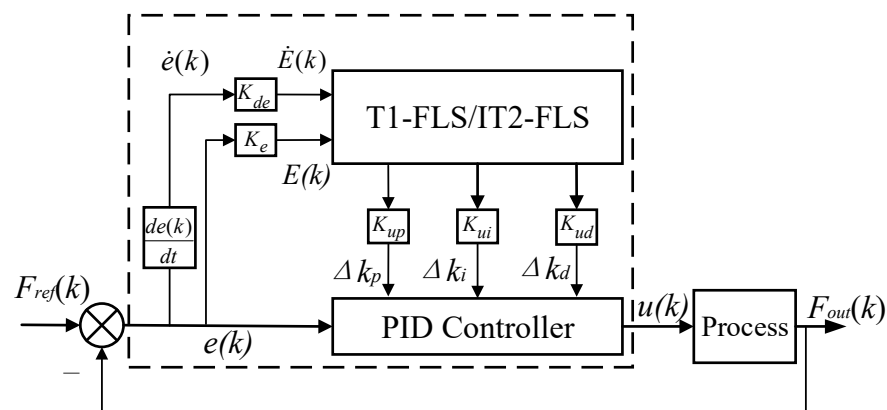


Figure 1. Interval type-2 fuzzy PID controller structure.

The structure of the traditional PID controller is as follows [6]:

$$\begin{aligned} u(k) &= k_p e(k) + k_i \int e(k) dk + k_d \frac{de(k)}{dk}, \\ e(k) &= F_{ref}(k) - F_{out}(k) \end{aligned} \quad (1)$$

where k_p , k_i , and k_d are the proportional, integral, and derivative gains, respectively, and $u(k)$, $F_{ref}(k)$, $F_{out}(k)$, and $e(k)$ are the controller output, expected target value, system output value, and the difference between the target value and the system output value, respectively. In the classic PID controller, the values of k_p , k_i , and k_d are adjusted by technicians according to changes in the system. The control ability of the controller depends heavily on the experience of these technicians, which is not conducive to stable and accurate control [7]. Therefore, an intelligent IT2-FLS based on PID control rules is designed to adjust the parameters of the classical PID controller online, enabling better adaptation to changes in the system without excessive operator intervention and further enhancing the application range of the traditional controller. The composition and working process of the IT2-FLS is introduced below, and the proposed D-GO method is described in detail.

2.1. Interval Type-2 Fuzzy Logic System

The structure of the upper-level IT2-FLS consists of four parts: a fuzzifier, an inference engine, a rule base, and an output processor. The specific process is shown in Figure 2 [28].

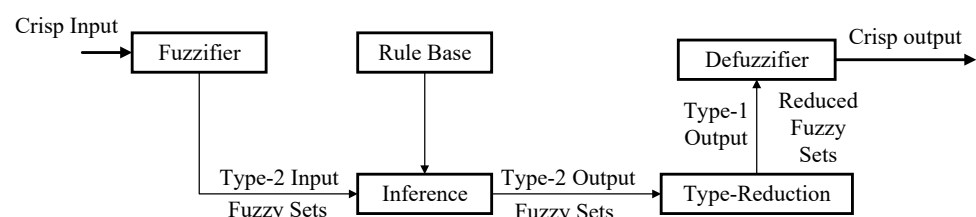


Figure 2. Structure of the type-2 fuzzy logic system.

The working process of the IT2-FLS is as follows [14]: the crisp input $e(k)$ and $de(k)/dt$ from the input sensor are first multiplied by the input factors (K_e and K_{de}) and fuzzified into the input interval type-2 fuzzy sets (IT2-FSs). The input IT2-FSs then activate the inference engine and rule base to generate the output IT2-FSs. The IT2-FSs rules are the same as in the T1-FLS, but the antecedents and/or consequents are represented by the IT2-FSs. The inference engine combines the fired rules and gives the mapping from the input IT2-FSs to the output IT2-FSs. The IT2 fuzzy outputs of the inference engine are processed using a type reducer, which combines the output sets and performs a centroid calculation to obtain the T1-FSs, called the type-reduction sets. After the type-reduction process, the type-reduction sets are defuzzified to obtain a clear output. The product of the crisp output and the scale factors (K_{up} , K_{ui} , and K_{ud}) gives the adjustment values (Δk_p , Δk_i , and Δk_d) in the classic PID controller.

2.1.1. Interval Type-2 Fuzzy Set

An IT2-FS $\tilde{A}$ can be expressed as [29]:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} 1/(x, u), \quad (2)$$

where x and u are the primary and secondary variables, respectively. J_x is the primary membership of x , $J_x \subseteq [0, 1]$. $\iint$ means taking all possible values of x and u . The secondary grades of $\tilde{A}$ are 1, $f_x(u) = 1(\forall u \in J_x \subseteq [0, 1])$. Figure 3 describes the footprint of uncertainty (FOU) of the fuzzy set $\tilde{A}$:

$$FOU(\tilde{A}) = \{(x, u) | x \in X \text{ and } u \in [\underline{\mu}_{\tilde{A}}(x), \bar{\mu}_{\tilde{A}}(x)]\}, \quad (3)$$

where $\underline{\mu}_{\tilde{A}}(x)$ and $\bar{\mu}_{\tilde{A}}(x)$ are the lower membership function (LMF) and the upper membership function (UMF), respectively, and $\underline{\mu}_{\tilde{A}}(x) \leq \bar{\mu}_{\tilde{A}}(x)$. The LMF and UMF are type-1 membership functions. Like type-1 fuzzy sets, the interval type-2 membership function (IT2MF) can take different forms, such as triangular, trapezoidal, Gaussian, or bell. The membership function used in this paper is triangular, as shown in Figure 3.

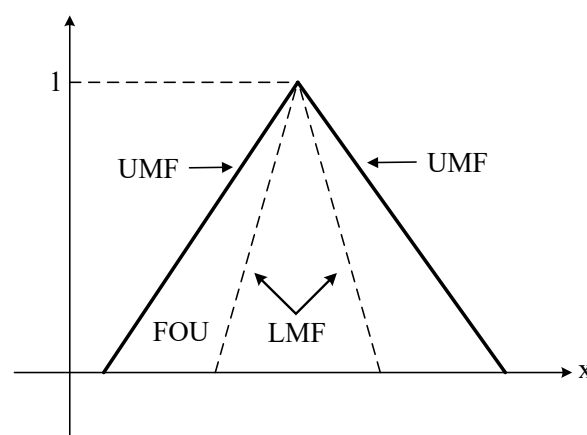


Figure 3. Interval type-2 fuzzy set.

2.1.2. Rule Base

The rules of the IT2-FLS are generally IF-THEN rules that formulate the knowledge of technicians. The rules determine how the IT2-FLS adjusts the parameters of the PID controller. The parameters k_p , k_i , and k_d mainly influence the system through the rise time, overshoot, and settling time. The influence of each parameter is described in Table 1.

Table 1. Effects of gain parameters.

Gain Parameters	Effects of Increasing Gain		
	Rise Time	Overshoot	Settling Time
k_p	Decrease	Increase	Small change
k_i	Decrease	Increase	Increase
k_d	Decrease	Decrease	Decrease

The function of the k_p is to speed up the response of the system. As k_p increases, the response speed of the system becomes faster and the adjustment accuracy is enhanced. However, it is easy to overshoot the target value, potentially leading to system instability. If the value of k_p is too small, the adjustment accuracy of the system will be reduced, the response speed will be slow, the adjustment time will be prolonged, and the static and dynamic characteristics of the system will deteriorate.

The function of k_i is to eliminate the steady-state error of the system. With increasing k_i , the static error of the system can be eliminated faster, but if the value is too large, the integral saturation phenomenon will occur at the initial stage of the response process, resulting in a large overshoot. If k_i is too small, it will be difficult to eliminate the static error, thus affecting the adjustment accuracy of the system.

The function of k_d is to improve the dynamic characteristics of the system. Its main function is to suppress changes in the deviation in any direction during the response process and predict the changes in deviation in advance. However, if k_d is too large, the adjustment process will be overdamped, which will prolong the adjustment time and reduce the anti-interference performance of the system.

Based on the above PID parameter characteristics, the fuzzy control rules are formulated. The rule base takes the error and the change in error as the premise and the proportional, integral, and differential gains as the result. The structure of the fuzzy rules is as follows:

If $E(k)$ is $\tilde{A}_1$ and $\dot{E}(k)$ is $\tilde{A}_2$
 then Δk_p is $\tilde{Y}_1$ and Δk_i is $\tilde{Y}_2$ and Δk_d is $\tilde{Y}_3$
 where $\tilde{A}_1$, $\tilde{A}_2$, $\tilde{Y}_1$, $\tilde{Y}_2$, and $\tilde{Y}_3$ are IT2-FSs, as described in Figure 3.

2.1.3. Fuzzification and Inference

When $E(k) = x'_1$, the vertical line at x'_1 crosses the $FOU(\tilde{A}_1)$ region to create an interval $[\underline{\mu}_{\tilde{A}_1}(x'_1), \bar{\mu}_{\tilde{A}_1}(x'_1)]$. When $\dot{E}(k) = x'_2$, the vertical line at x'_2 crosses the $FOU(\tilde{A}_2)$ region, creating an interval $[\underline{\mu}_{\tilde{A}_2}(x'_2), \bar{\mu}_{\tilde{A}_2}(x'_2)]$. After obtaining two sets of interval values, two firing levels are computed. The lower and upper firing levels are calculated as $\underline{f}(x') = \min[\underline{\mu}_{\tilde{A}_1}(x'_1), \underline{\mu}_{\tilde{A}_2}(x'_2)]$ and $\bar{f}(x') = \min[\bar{\mu}_{\tilde{A}_1}(x'_1), \bar{\mu}_{\tilde{A}_2}(x'_2)]$, respectively. After obtaining the upper and lower firing levels, the firing interval is obtained as $F(x') = [\underline{f}(x'), \bar{f}(x')]$.

2.1.4. Type Reduction and Defuzzification

The type-reduction process is part of the IT2-FLS that differs from the T1-FLS. The process of type reduction involves transforming IT2-FSs into T1-FSs [30]. The type reduction of IT2-FSs outputs T1-FSs, which are collections of all embedded T1-FSs. In the proposed controller, the most common center-of-set method based on the Karnik–Mendel (KM) algorithm is selected to map IT2-FSs to T1-FSs [31]. The calculation process is as follows:

$$y_l = \min_{k \in [1, n-1]} \frac{\sum_{n=1}^k \bar{f}^n y^n + \sum_{n=k+1}^n \underline{f}^n y^n}{\sum_{n=1}^k \bar{f}^n + \sum_{n=k+1}^n \underline{f}^n}, \quad (4)$$

$$y_r = \max_{k \in [1, n-1]} \frac{\sum_{n=1}^k f^n \bar{y}^n + \sum_{n=k+1}^n \bar{f}^n \bar{y}^n}{\sum_{n=1}^k f^n + \sum_{n=k+1}^n \bar{f}^n}, \quad (5)$$

$$Y_{\cos}(x') = \bigcup_{\substack{f^n \in F^n(x') \\ y^n \in Y^n}} \frac{\sum_{n=1}^n f^n y^n}{\sum_{n=1}^n f^n} = [y_l, y_r], \quad (6)$$

where $Y_{\cos}(x)$ is the T1-FS, y_l and y_r are the left and right endpoints of the IT2-FS, respectively. y_l and y_r can be calculated using the KM algorithm [32]. After the type-reduction process is applied, the clear output value of the IT2-FLS is calculated as:

$$y(x) = \frac{y_l + y_r}{2}, \quad (7)$$

2.2. Proposed D-GO Method

Based on the structure of the IT2F-PID controller (Figure 1), at least eight parameters ($k_p, k_i, k_d, K_e, K_{de}, K_{up}, K_{ui},$ and K_{ud}) should be optimized to establish a suitable controller. To reduce the parameter optimization time and the hardware requirements, this paper disassembles the parameter adjustment into two categories and performs the optimization gradationally as shown in Figure 4.

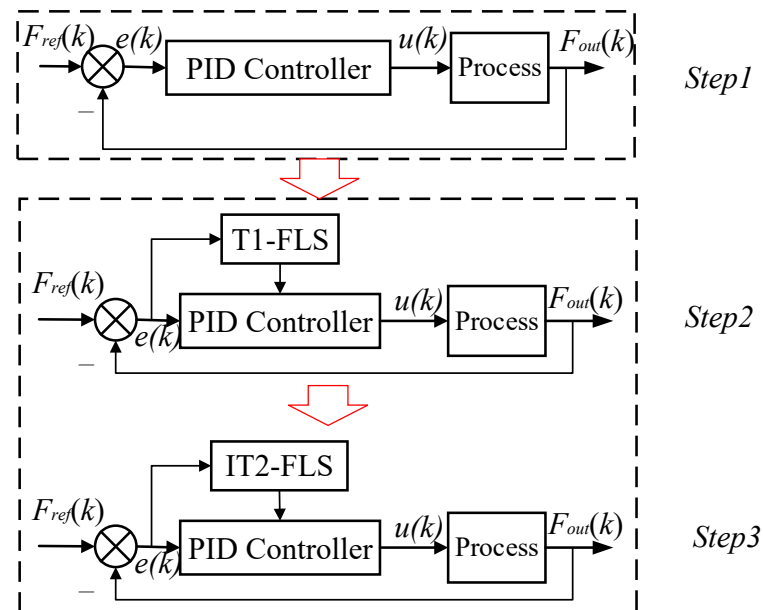


Figure 4. Diagram of the D-GO method.

The PID controller of the IT2F-PID controller plays the control role. The IT2-FLS adjusts the PID control parameters. Therefore, we optimize the PID controller parameters ($k_p, k_i,$ and k_d) first, and then optimize the IT2-FLS parameters. This effectively reduces the optimization time. The specific optimization process, optimization algorithm, and objective function are introduced below.

2.2.1. Specific Process of the D-GO Method

The D-GO method proceeds as follows. First, the IT2F-PID controller is simplified to a PID controller, and the PID controller is connected to the controlled system as shown in Figure 4 (step 1).

At this time, only the PID controller parameters need to be optimized. After the PID parameters (k_p , k_i , and k_d) have been optimized, the IT2-FLS can be established and optimized. Once the optimal PID controller parameters have been determined, the IT2-FLS can adjust Δk_p , Δk_i , and Δk_d . Compared with the cascade-type IT2F-PID controller parameters, the optimization range of these parameters is greatly reduced, which significantly reduces the optimization time.

The membership function of the IT2-FLS is uncertain. Hence, optimizing the input and output factors directly will take a lot of time, which affects work efficiency. The structure and mathematical foundations of the IT2-FLS are similar to those of the T1-FLS. Typically, the IT2-FLS is constructed by adding uncertainty to the membership function based on the T1-FLS [33–35]. The method of expanding a T1-FLS to an IT2-FLS is theoretically feasible and takes less time.

Therefore, we first establish a T1-FLS, select an appropriate membership function for the fuzzy sets E , $\dot{E}$, U_p , U_i , and U_d , and then optimize the scale factors (K_e , K_{de} , K_{up} , K_{ui} , and K_{ud}) of the T1-FLS as shown in Figure 4 (step 2). After that, the uncertainty of the membership functions in the T1-FLS is added, and then the IT2-FLS is established (step 3). To verify the effectiveness of the proposed D-GO method, k_p , k_i , k_d , K_e , K_{de} , K_{up} , K_{ui} , and K_{ud} are optimized under the condition that the T1-FLS and IT2-FLS membership functions have been determined. Various intelligent algorithms can be used to optimize the above parameters, including genetic algorithms [36], particle swarm optimization (PSO) algorithm [37], grey wolf optimization algorithm [38], and imperialist competition algorithm [39]. This paper uses the PSO algorithm as an example to demonstrate the implementation of the D-GO method and optimize the PID controller and IT2-FLS. The optimization objective function is the integral-of-time absolute error (ITAE).

2.2.2. PSO and Objective Function

PSO is a global optimization algorithm inspired by the study of birds' predation behavior [37]. The algorithm starts with randomly generated particles in the search space. In each iteration, particles move around the search space at the specified speed, hoping to find the optimal solution. The position of each particle is updated based on the particle's current position, the particle's individual optimal solution position, and the global optimal solution position. The particle's position is then iteratively updated based on the new speed until the stop criterion is reached [37]. The mathematical expression of the above process is:

$$\begin{aligned} v_i^{t+1} &= w \times v_i^t + c_1 \times r_1 \times (p_{i,d} - x_{i,d}^t) + \\ & c_2 \times r_2 \times (p_{g,d} - x_{i,d}^t) \\ x_{i,d}^{t+1} &= x_{i,d}^t + v_i^{t+1} \end{aligned} \quad (8)$$

where $x_{i,d}^t$ and $x_{i,d}^{t+1}$ are the current and future positions of the i th particle, respectively. The parameter v_i^t is the current speed of the particle, and w is a weight function that controls the impact of the particle's current speed on its future speed. The variable $p_{i,d}$ represents the best solution of particle i at iteration t , and $p_{g,d}$ is the best global solution attained so far. The weight factors c_1 and c_2 determine the importance of the current local optimal solution and the global optimal solution. Finally, r_1 and r_2 are random numbers in the range $[0, 1]$, which further increase the randomness of the particle search in the whole search space.

The selection of a fitness function in the optimization algorithm directly affects the optimization effect; therefore, the commonly used ITAE [17] is selected as the fitness function. The ITAE is calculated as follows:

$$ITAE = \int_0^\infty t|e(t)|dt, \quad (9)$$

3. Forced Closed-Loop System

To verify the effectiveness of the IT2F-PID controller established using the D-GO method proposed in this paper, a forced closed-loop system is established based on a single leg of a parallel manipulator. The control ability of the IT2F-PID controller established using the proposed method is tested when the controlled system and the input signal are uncertain. The controlled system and the IT2F-PID controller designed for the system are introduced in detail below.

3.1. System Model

The common leg structure of parallel manipulators incorporates a permanent magnet synchronous motor (PMSM) connected with a reducer and a ball screw [40]. To realize the forced control closed-loop system, a force sensor is installed at the end of the ball screw. The PMSM is the power output mechanism and is, therefore, the core of the whole system.

The mathematical model of the motor is established and the PMSM is regarded as an ideal motor. It is assumed that the surface-mounted PMSM adopted does not consider the core reluctance, the loss of hysteresis and eddy current, the higher harmonics of the magnetic field, and the parameter changes in inductance and resistance. There is no damping winding in the rotor or the permanent magnet. The mutual inductance coefficient between the rotor winding and the stator winding is a sine (or cosine) function of the rotor position angle, and the friction coefficient is 0 [41]. The dynamic mathematical model of the PMSM is:

$$\begin{cases} u_d = L_d \frac{d}{dt} i_d + R_s i_d - L_q p_n \omega_e i_q \\ u_q = L_q \frac{d}{dt} i_q + R_s i_q + L_d p_n \omega_e i_d + p_n \omega_e \psi_f \\ T_e = \frac{3}{2} p_n i_q [i_d (L_d - L_q) + \psi_f] \\ T_e = T_d + J \frac{d\omega_m}{dt} + B \omega_m \end{cases}, \quad (10)$$

where u_d and u_q are the d - q axis components of the stator voltage, i_d and i_q are the d - q axis components of the stator current, R_s is the stator resistance, ω_e is the electrical angular velocity of rotor rotation, ψ_f is the flux chain, L_d and L_q are the d - q axis inductance components, and T_d , J , ω_m , B , and T_e are the load torque, motor rotational inertia, motor rotor mechanical angular velocity, damping coefficient, and output torque, respectively. The whole forced closed-loop system model is established on the basis of the dynamic mathematical model of the PMSM [42]. Figure 5 shows a block diagram of the whole forced closed-loop system:

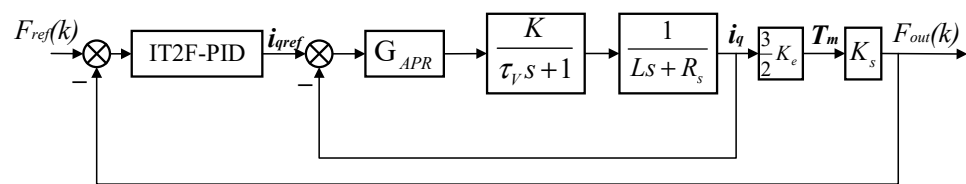


Figure 5. Block diagram of the forced closed-loop system.

where G_{APR} is a PI controller, $\frac{K}{\tau_V s + 1}$ is the simplified transfer function of the PWM inverter [43], k is the amplification ratio of the inverter, τ_V is the time constant, $\frac{1}{Ls + R_s}$ is the simplified transfer function of the PMSM, L is the armature inductance, R_s is the stator resistance, K_e is the torque coefficient, and K_s is the ratio coefficient of the ball screw and reducer.

Analysis of Equation (10) shows that the output torque is proportional to the current, so the inner loop of the forced closed-loop system is the current control loop. In this example, the force sensor signal is used for closed-loop force control, which is provided

using the force (IT2F-PID) controller. Once the reducer and ball screw are connected to the motor, the motor output force is:

$$F_{out} = T_m \times \frac{2\pi \cdot \eta \cdot i}{P_h}, \quad (11)$$

where T_m is the output torque of the motor, P_h is the lead of the ball screw pair, η is the efficiency of the ball screw pair without preload, i is the reduction ratio of the reducer, and F_{out} is the outward output force of the ball screw pair. The transfer coefficient of the force sensor is 1.

3.2. Proposed Controller for the Forced Closed-Loop System

Figure 5 shows the forced closed-loop system with the IT2F-PID controller. The goal is for the output force of the structure to satisfy the design requirements. The steps involved in designing the IT2F-PID controller are as follows:

- First, the PID controller is designed. Then, the following steps are carried out: input the expected force, define the initial optimization range, optimize k_p , k_i , and k_d using the PSO algorithm and ITAE of the fitness function, and obtain the optimal rounding for the PID parameters.
- Next, a Sugeno-type T1-FLS as the design basis of the IT2-FLS is established. The error and error change rate are selected as the input variables of the IT2-FLS, and the proportional, integral, and differential gains are selected as the output variables. After the input and output variables have been determined, a T1-FLS is established. Then, the linguistic variables are divided into quantitative levels. A finer quantization level will have more corresponding rules, and the control will become more complex. When the number of selected levels is small, there are fewer rules and the control effect will become rougher. Based on experience, we divided E , $\dot{E}$, U_p , U_i , and U_d into three levels, namely, positive (P), zero (ZO), and negative (N). The relationships between the fuzzy linguistic values (E , $\dot{E}$, U_p , U_i , and U_d) and the actual input and output values (e , $\dot{e}$, Δk_p , Δk_i , and Δk_d) are:

$$\begin{cases} E = K_e \cdot e \\ \dot{E} = K_{de} \cdot \dot{e} \\ \Delta k_p = K_{up} \cdot U_p \\ \Delta k_i = K_{ui} \cdot U_i \\ \Delta k_d = K_{ud} \cdot U_d \end{cases} \quad (12)$$

The membership functions used in the fuzzy sets of input variables E and $\dot{E}$ are shown in Figure 6. The membership function is a trigonometric function with a range of $[-1, 1]$. The parameters of N are $[-1, -1, -0.2]$, the parameters of ZO are $[-0.8, 0, 0.8]$, and the parameters of P are $[0.2, 1, 1]$.

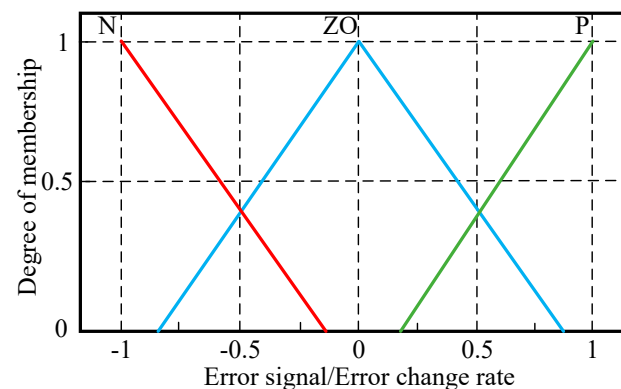


Figure 6. Membership functions for E and $\dot{E}$.

The membership functions used for the output (U_p , U_i , and U_d) are shown in Figure 7.

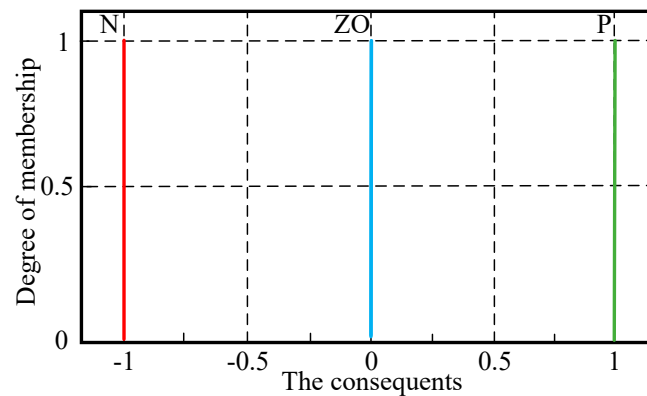


Figure 7. Membership functions for U_p , U_i , and U_d .

The membership functions of the consequents are single values, and the corresponding values of $\{N, ZO, P\}$ are $\{-1, 0, 1\}$. The fuzzy rules are written based on Table 1 and experience. Based on the influence of the proportional, integral, and derivative gains on the force control system, the corresponding fuzzy rules are listed in Tables 2–4.

Table 2. Rule base for the proportional gain.

Error Signal	Change in Error Signal		
	N	ZO	P
N	N	N	N
ZO	N	P	P
P	P	P	P

Table 3. Rule base for the integral gain.

Error Signal	Change in Error Signal		
	N	ZO	P
N	ZO	ZO	ZO
ZO	P	P	P
P	ZO	ZO	ZO

Table 4. Rule base for the derivative gain.

Error Signal	Change in Error Signal		
	N	ZO	P
N	N	N	P
ZO	N	ZO	P
P	N	N	P

- After establishing the T1-FLS, the scale factors (K_e , K_{de} , K_{up} , K_{ui} , and K_{ud}) of the T1-FLS are optimized. According to the PID parameters, the optimization ranges are defined for these five parameters, and the optimal values are obtained and rounded. Based on experience, the uncertainty of the membership functions in the T1-FLS is added, and then the IT2-FLS is established. The input membership function of the IT2-FLS is shown in Figure 8. At the bottom of the membership function of the T1-FLS, a blurring value of 0.2 is applied. The membership functions used for output (U_p , U_i , and U_d) are shown in Figure 7.

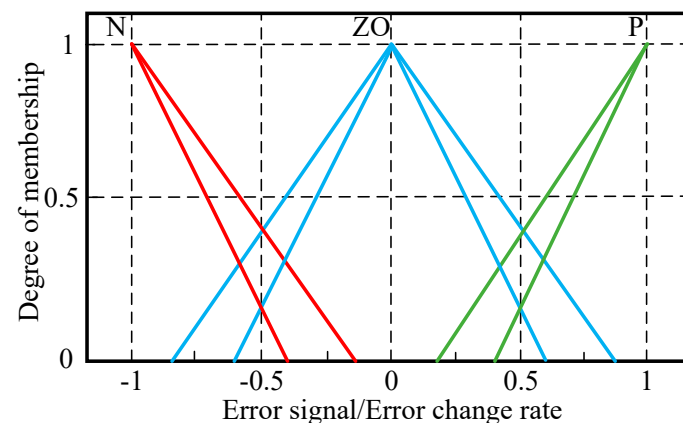


Figure 8. Membership functions of IT2-FLS for E and $\dot{E}$.

4. Simulation Results

This section describes the application of the IT2F-PID controller established using the D-GO method to simulate and control a forced closed-loop system. At present, IT2F-PID controller construction mostly uses the general optimization method, which optimizes the parameters at the same time [27]. Beyond that, the concurrent method is added for comparison. Comparative analysis was conducted on the optimization time and ITAE of the D-GO and the concurrent and general optimization methods. Four different tasks were performed and the performance was compared with that of a PID controller and IT2F-PID controllers established using the general optimization method and the concurrent optimization method, respectively. To ensure the effectiveness of the comparison, the optimization algorithm and the objective function are the same for each controller. The parameters of the forced closed-loop system are listed in Table 5.

Table 5. Parameters of the forced closed-loop control system.

Parameter	Meaning	Value	Unit
L	Rotator inductance	0.01225	H
R	Rotator resistance	12.4	Ω
Pn	Number of pairs	5	/
Ke	Torque constant	0.1125	/
τ_V	Inverter time constant	0.001	s
K	Inverter gain	1	/
Ks	Proportionality coefficient	113,100	/
G_{APRP}	The proportional parameter of G_{APR}	39	/
G_{APRI}	The integral parameter of G_{APR}	8563	/

4.1. Task 1: Normal Case

First, we compare the performance of the PID controller and the IT2F-PID controllers established using the general optimization method, the D-GO method, and the concurrent optimization method, respectively, for a normal case with a 3000 N expected output force. The results are presented in Figure 9. The parameters (k_p , k_i , and k_d) were optimized in the range $[0-1, 0-500, 0-1]$ and the optimal PID parameters $[0.04, 222, 0]$ were obtained after rounding. The scale factors (K_e , K_{de} , K_{up} , K_{ui} , and K_{ud}) of the T1-FLS were then optimized. The initial optimization range of these five parameters was defined as $[0-0.001, 0-0.000001, 0-0.1, 0-200, 0-0.01]$, and the optimal parameters were rounded. The IT2F-PID controller was then established as described in Section 3.2.

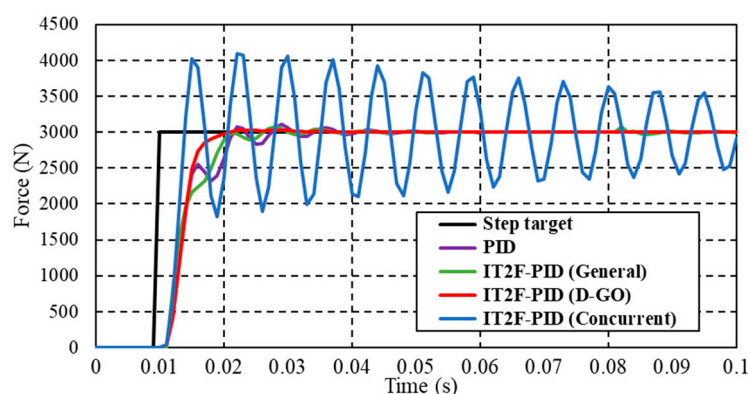


Figure 9. Response of the forced closed-loop system for a normal case.

The general method is to set up the IT2F-PID controller directly and then optimize its eight parameters (k_p , k_i , k_d , K_e , K_{de} , K_{up} , K_{ui} , and K_{ud}) together. The general optimization method is different from the D-GO method. It does not optimize the PID controller first, so it has no preliminary understanding of the system error, error rate of change, or changes in the proportional, integral, and derivative gains. Therefore, the initial parameters (k_p , k_i , k_d , K_e , K_{de} , K_{up} , K_{ui} , and K_{ud}) are estimated values. The initial optimization range is [0–1, 0–500, 0–1, 0–1, 0–1, 0–1, 0–500, 0–1] and the optimal parameters are rounded.

The concurrent optimization is to connect the PID controller and IT2-FLS into the controlled system, respectively, for parameter optimization, and then combine them. The PID parameter optimization process is the same as that of the D-GO method, so it is not described here. The IT2-FLS parameter optimization process is similar to that of the general method, which is also not described here. The parameters to be optimized are K_e , K_{de} , K_{up} , K_{ui} , and K_{ud} and the initial optimization range is [0–1, 0–1, 0–1, 0–500, 0–1]. The IT2F-PID controller built using the D-GO method has obvious advantages over the controllers built using the concurrent and general optimization methods, as shown in Figure 9. Compared with the other controllers, the controller reduces the overshoot by 80% and the fluctuation by 67%. It has only a small overshoot, no oscillation, and no steady-state error after stabilization. Compared with the IT2F-PID controller, the PID controller has a longer rise time, larger overshoot, and longer settling time. The parameters of the PID controller used are consistent with those of the IT2F-PID controller built using the D-GO method; therefore, the reason for the poor performance is that the PID controller is not connected with the IT2-FLS. Similarly, the IT2F-PID controller established using the general method also has problems like the PID controller and it exhibits sudden fluctuations even after the system has stabilized. Although the fluctuations quickly disappear, they have a certain impact on the stability of the system. The reason for the above problems is that compared with the D-GO method, the initial parameter selection is not accurate enough, resulting in the optimized parameters being not optimal and the control performance being poor. For the D-GO method, a better parameter range was preliminarily determined in the PID parameter optimization step. The controller constructed using the concurrent method does not converge. The control result of the IT2F-PID controller is poor because the PID controller and IT2-FLS are connected to the controlled system, respectively, for parameter optimization during the construction process. The combination of the two causes the IT2F-PID controller to lose control of the controlled system. This illustrates that the proposed controller achieves superior performance.

To illustrate the advantages of the D-GO method, it is compared with the concurrent and general optimization methods in Table 6. Taking the normal case as an example, the D-GO method has a significantly shorter optimization time than the concurrent and general optimization methods in constructing the IT2F-PID controller and outperforms the concurrent and general optimization methods in terms of the ITAE. The D-GO method reduces the optimization time by over 90% and the ITAE value by about 30% compared

with the general method. Compared with the concurrent optimization method, the D-GO method reduces time by over 25% and the ITAE value by about 95%.

Table 6. Comparison between the D-GO method and the concurrent and general optimization methods.

Method	Time (s)	ITAE
The D-GO	1020	0.1462
The concurrent optimization	1370	2.8541
The general optimization	13,242	0.2087

By comparing the optimization time of the three methods, it can be found that when the IT2F-PID controller is disassembled into the PID controller and IT2-FLS, fewer parameters are optimized at the same time, and a lot of optimization time can be saved. By comparing ITAE values, it can be found that if the relationship between the PID controller and IT2-FLS is not considered and the controller is only optimized and recombined, the controller will lose control ability. The optimization method that saves time while ensuring control ability is the D-GO method, which first disassembles the controller and analyzes the relationship between the PID controller and the IT2-FLS, and then carries out gradational optimization. It ensures that fewer parameters are optimized at the same time, and the relationship between the two controllers is considered to ensure the control performance.

4.2. Task 2: Model Uncertainty

In practical control applications, the existence of uncertainties such as temperature and friction [44] makes it difficult to obtain accurate values from the physical model of the forced closed-loop system. Therefore, there is a mismatch between the simulation results and the actual model output. To verify the effectiveness of the D-GO method, the motor inductance parameter and the winding resistance were set to 1.1 times their nominal values. All control parameters remained unchanged.

The results in Figure 10 show that, in the case of model mismatch, the force tracking performance of the controller of the D-GO method is consistent with that of model matching. The force response curve has only a small overshoot, and there is no steady-state error after a slight oscillation, so the required value is quickly attained and accurately tracked. The PID controller and the IT2F-PID controller established using the general optimization method exhibit worse force tracking ability than the IT2F-PID controller established using the D-GO method. In addition, the IT2F-PID controller established using the general method still exhibits sudden fluctuations and it is more obvious after the system has stabilized. Although the fluctuations quickly disappear, they have a certain impact on the stability of the system. The reasons for the above phenomena are the same as that in normal cases. The PID controller is not connected with the IT2-FLS, so it does not have real-time tuning ability, and its control performance is inevitably poor. The parameters of the IT2F-PID controller established using the general method are not optimal, which leads to poor control performance when the controlled system parameters are uncertain, and the fluctuation after stability is more obvious. In normal cases, the concurrent method has not converged, and it still does not converge in the model uncertainty; therefore, it is not discussed here. This again illustrates that the proposed controller achieves superior performance.

4.3. Task 3: Target Uncertainty

In the actual force control process, there is great uncertainty about the expected output force. Thus, we performed experiments to verify the adaptability of the D-GO method to different expected forces. The expected input was a pulse force with an amplitude of 3000 N, a period of 0.1 s, and a pulse width of 50%. This input is significantly different from that used to establish the controller and is intended to reflect the effectiveness of the proposed method.

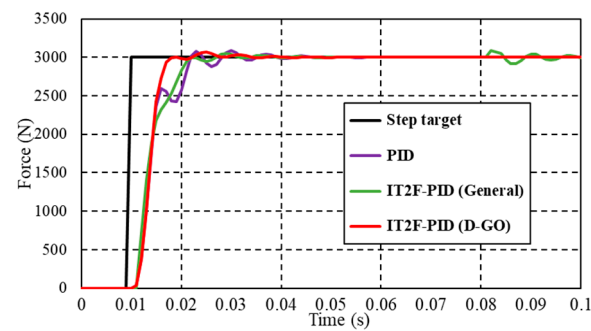


Figure 10. Force tracking curve under model mismatch.

The results in Figure 11 show that the IT2F-PID controller designed using the D-GO method achieves excellent force tracking performance under the peak of pulse force. The force tracking fluctuates at the bottom, so it can be seen that the optimization parameters obtained in the normal case cannot be applied to any situation. Compared with the other controllers, the proposed IT2F-PID controller reduces the overshoot by 80% and the fluctuation by 67%. The force response curve has a small overshoot, and quickly tracks the required value thereafter. The performance is obviously better than that of the PID controller and the IT2F-PID controller designed using the general optimization method. Because the PID controller does not have the ability to adjust its own parameters, the parameters of the IT2F-PID controller designed using the general method are far from the optimal value, resulting in an obvious overshoot and a significant increase in the stability time of both.

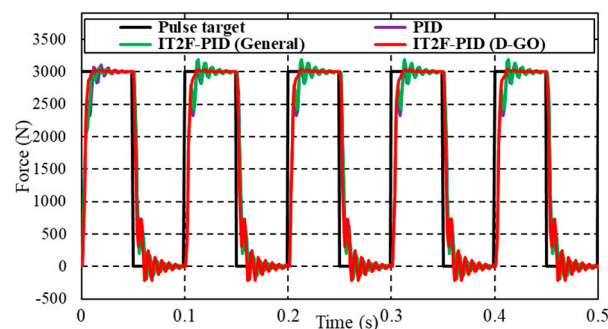


Figure 11. Force tracking curve under pulse signal.

4.4. Task 4: External Disturbance

The forced closed-loop control system is inevitably subjected to external force interference during operation. Under external disturbance, the validity of the D-GO method is tested. The increase in disturbance force was $F_d = 300$ N in the normal case. The parameters of each controller remain unchanged.

The force tracking performance is shown in Figure 12. Under external disturbance, the controller constructed using the D-GO method has a short rise time, small overshoot, and short settling time in the process of force tracking, and the overall tracking performance is obviously better. Compared with the PID controller and the controller constructed using the general method, the overshoot was reduced by 80%, the fluctuation was reduced by 67%, and especially the recovery speed after interference was obviously better. The reason for the poor performance of the controller established using the general method is similar to the normal case. The PID has no ability to adjust its own parameters, and the controller parameters established using the general method have a wide range of optimization, which leads to a long distance from the optimal value and poor control performance. The parameters of the controller constructed using the D-GO method are closer to the optimal value and the control performance is better.

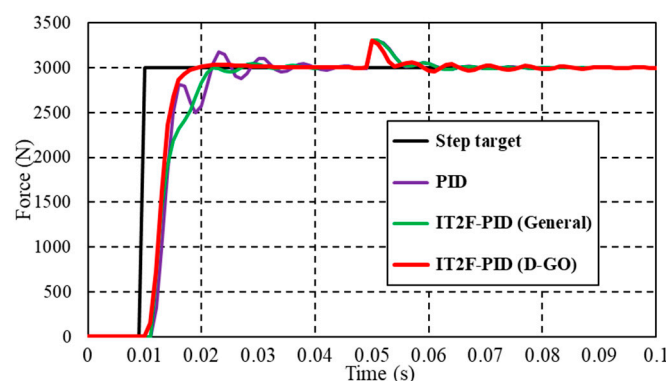


Figure 12. Force tracking curve under external disturbance.

5. Conclusions

This paper has described the D-GO method for designing a parallel-type IT2F-PID controller. The proposed method disassembles the parallel IT2F-PID controller into a PID controller and a T1-FLS for parameter optimization and then blurs the membership function of the T1-FLS to construct a parallel IT2F-PID controller. Compared with the general optimization method, the D-GO method reduces the optimization time by over 90% and the ITAE value by about 30%. Compared with the concurrent optimization method, the D-GO method reduces time by over 25% and the ITAE value by about 95%. The D-GO method significantly reduces the optimization time and supports the IT2F-PID controller to realize online real-time parameter adjustment and control a time-varying nonlinear complex system.

Simulations were conducted to verify the control performance of the D-GO method. Through comparisons with a PID controller and an IT2F-PID controller established using the concurrent and general optimization methods, we found that in the normal case, model uncertainty, target uncertainty, and external disturbance, the D-GO method's controller achieves superior control performance for the forced closed-loop system. The overshoot is reduced by 80% and the fluctuation is reduced by 67% compared with the PID controller and the parallel IT2F-PID controller established using the concurrent and general optimization methods. In addition, compared with the other controllers, the D-GO method's controller does not produce sudden fluctuations after the system has been stable and the recovery speed after interference is obviously better. This illustrates that the D-GO method's controller achieves superior performance. The results presented in this paper are sufficient to demonstrate that the D-GO method is suitable for the establishment of IT2F-PID controllers. By extension, it also provides inspiration for the establishment of other intelligent controllers.

Author Contributions: Conceptualization, Y.C. and T.M.; methodology, Y.C.; software, T.M.; validation, M.Z., Z.L. and H.H.; formal analysis, Y.C.; investigation, Y.C.; resources, H.H.; data curation, T.M.; writing—original draft preparation, H.H.; writing—review and editing, H.H.; visualization, T.M.; supervision, Z.X.; project administration, Q.W.; and funding acquisition, H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number No. 52005478 and the Youth Innovation Promotion Association, Chinese Academy of Sciences grant number No. 2022215.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available within the article.

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

References

- Skogestad, S. Simple analytic rules for model reduction and PID controller tuning. *J. Process Control* **2003**, *13*, 291–309. [\[CrossRef\]](#)
- Toscano, R. Robust synthesis of a PID controller by uncertain multimodel approach. *Inf. Sci.* **2007**, *177*, 1441–1451. [\[CrossRef\]](#)
- Reznik, L.; Ghanayem, O.; Bourmistrov, A. PID plus fuzzy controller structures as a design base for industrial applications. *Eng. Appl. Artif. Intell.* **2000**, *13*, 419–430. [\[CrossRef\]](#)
- Xie, T.C.; Nan, X. Research and Applications of Fuzzy-PID Control on NC Machine. *Key Eng. Mater.* **2009**, 392–394, 650–654. [\[CrossRef\]](#)
- Yamamoto, T.; Kawada, K.; Kugemoto, H.; Kutsuwa, Y. Design and Industrial Applications of a Control Performance Assessment Based PID Controller. *IFAC Proc. Vol.* **2009**, *42*, 729–734. [\[CrossRef\]](#)
- El-Bardini, M.; El-Nagar, A.M. Interval type-2 fuzzy PID controller for uncertain nonlinear inverted pendulum system. *ISA Trans.* **2014**, *53*, 732–743. [\[CrossRef\]](#)
- Kim, J.H.; Oh, S.J. A fuzzy PID controller for nonlinear and uncertain systems. *Soft Comput.* **2000**, *4*, 123–129. [\[CrossRef\]](#)
- Patel, A.V.; Mohan, B.M. Analytical structures and analysis of the simplest fuzzy PI controllers. *Automatica* **2002**, *38*, 981–993. [\[CrossRef\]](#)
- Mohan, B.M.; Patel, A.V. Analytical structures and analysis of the simplest fuzzy PD controllers. *IEEE Trans. Syst. Man Cybern. Part B* **2002**, *32*, 239–248. [\[CrossRef\]](#)
- Güzelkaya, M.; Eksin, I.; Yeşil, E. Self-tuning of PID-type fuzzy logic controller coefficients via relative rate observer. *Eng. Appl. Artif. Intell.* **2003**, *16*, 227–236. [\[CrossRef\]](#)
- Kumbasar, T. A simple design method for interval type-2 fuzzy PID controllers. *Soft Comput.* **2014**, *18*, 1293–1304. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Lu, X.G.; Liu, M.; Liu, J.X. Design and optimization of interval type-2 fuzzy logic controller for delta parallel robot trajectory control. *Int. J. Fuzzy Syst.* **2017**, *19*, 190–206. [\[CrossRef\]](#)
- Jammeh, E.A.; Fleury, M.; Wagner, C.; Hagnas, H.; Ghanbari, M. Interval type-2 fuzzy logic congestion control for video streaming across IP networks. *IEEE Trans. Fuzzy Syst.* **2009**, *17*, 1123–1142. [\[CrossRef\]](#)
- El-Nagar, A.M.; El-Bardini, M.; El-Rabaie, N.M. Intelligent control for nonlinear inverted pendulum based on interval type-2 fuzzy PD controller. *Alex. Eng. J.* **2014**, *53*, 23–32. [\[CrossRef\]](#)
- Hagnas, H.A. A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots. *IEEE Trans. Fuzzy Syst.* **2004**, *12*, 524–539. [\[CrossRef\]](#)
- Tong, W.; Zhao, T.; Duan, Q.; Zhang, H.; Mao, Y. Non-singleton interval type-2 fuzzy PID control for high precision electro-optical tracking system. *ISA Trans.* **2022**, *120*, 258–270. [\[CrossRef\]](#)
- El-Nagar, A.M.; El-Bardini, M. Practical implementation for the interval type-2 fuzzy PID controller using a low cost microcontroller. *Ain Shams Eng. J.* **2014**, *5*, 475–487. [\[CrossRef\]](#)
- Kumar, A.; Kumar, V. A novel interval type-2 fractional order fuzzy PID controller: Design, performance evaluation, and its optimal time domain tuning. *ISA Trans.* **2017**, *68*, 251–275. [\[CrossRef\]](#)
- Kumbasar, T.; Hagnas, H. Big Bang–Big Crunch optimization based interval type-2 fuzzy PID cascade controller design strategy. *Inf. Sci.* **2014**, *282*, 277–295. [\[CrossRef\]](#)
- Khooban, M.H.; Alfi, A.; Abadi, D.N.M. Teaching–learning-based optimal interval type-2 fuzzy PID controller design: A nonholonomic wheeled mobile robots. *Robotica* **2013**, *31*, 1059–1071. [\[CrossRef\]](#)
- AbouOmar, M.S.; Su, Y.; Zhang, H.; Shi, B.; Wan, L. Observer-based interval type-2 fuzzy PID controller for PEMFC air feeding system using novel hybrid neural network algorithm-differential evolution optimizer. *Alex. Eng. J.* **2022**, *61*, 7353–7375. [\[CrossRef\]](#)
- El-Nagar, A.M.; El-Bardini, M. Derivation and stability analysis of the analytical structures of the interval type-2 fuzzy PID controller. *Appl. Soft Comput.* **2014**, *24*, 704–716. [\[CrossRef\]](#)
- Araujo, H.; Xiao, B.; Liu, C.; Zhao, Y. and Lam, H.K. Design of type-1 and interval type-2 fuzzy PID control for anesthesia using genetic algorithms. *J. Intell. Learn. Syst. Appl.* **2014**, *6*, 70.
- Yesil, E. Interval type-2 fuzzy PID load frequency controller using Big Bang–Big Crunch optimization. *Appl. Soft Comput.* **2014**, *15*, 100–112. [\[CrossRef\]](#)
- Kumar, A.; Kumar, V. Evolving an interval type-2 fuzzy PID controller for the redundant robotic manipulator. *Expert Syst. Appl.* **2017**, *73*, 161–177. [\[CrossRef\]](#)
- 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. [\[CrossRef\]](#)
- Mendel, J.M. Type-2 fuzzy sets and systems: An overview. *IEEE Comput. Intell. Mag.* **2007**, *2*, 20–29. [\[CrossRef\]](#)
- Mendel, J.M.; John, R.I.B. Type-2 fuzzy sets made simple. *IEEE Trans. Fuzzy Syst.* **2002**, *10*, 117–127. [\[CrossRef\]](#)
- Wu, H.; Mendel, J.M. Uncertainty bounds and their use in the design of interval type-2 fuzzy logic systems. *IEEE Trans. Fuzzy Syst.* **2002**, *10*, 622–639.
- Mendel, J.M. *Uncertain Rule-Based Fuzzy Systems: Introduction and New Directions*; Springer: Berlin, Germany, 2017; pp. 391–396.
- Karnik, N.N.; Mendel, J.M.; Liang, Q. Type-2 fuzzy logic systems. *IEEE Trans. Fuzzy Syst.* **1999**, *7*, 643–658. [\[CrossRef\]](#)
- Mendel, J.M.; John, R.I.; Liu, F. Interval type-2 fuzzy logic systems made simple. *IEEE Trans. Fuzzy Syst.* **2006**, *14*, 808–821. [\[CrossRef\]](#)

34. Kumbasar, T.; Hagrass, H. A self-tuning zSlices-based general type-2 fuzzy PI controller. *IEEE Trans. Fuzzy Syst.* **2014**, *23*, 991–1013. [[CrossRef](#)]
35. Wu, D. On the fundamental differences between interval type-2 and type-1 fuzzy logic controllers. *IEEE Trans. Fuzzy Syst.* **2012**, *20*, 832–848. [[CrossRef](#)]
36. Noshadi, A.; Shi, J.; Lee, W.S.; Shi, P.; Kalam, A. Genetic algorithm-based system identification of active magnetic bearing system: A frequency-domain approach. In Proceedings of the 11th IEEE International Conference on Control & Automation (ICCA), Taichung, Taiwan, 18–20 June 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 1281–1286.
37. Eberhart, R.; Kennedy, J. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
38. Saremi, S.; Mirjalili, S.Z.; Mirjalili, S.M. Evolutionary population dynamics and grey wolf optimizer. *Neural Comput. Appl.* **2015**, *26*, 1257–1263. [[CrossRef](#)]
39. Atashpaz-Gargari, E.; Lucas, C. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 4661–4667.
40. Yu, Y.; Xu, Z.B.; Wu, Q.W.; Yu, P.; He, S.; Wang, G.Q. Kinematic analysis and testing of a 6-RR RP RR parallel manipulator. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2017**, *231*, 2515–2527. [[CrossRef](#)]
41. Wang, Q.; Yu, H.; Wang, M.; Qi, X. An improved sliding mode control using disturbance torque observer for permanent magnet synchronous motor. *IEEE Access* **2019**, *7*, 36691–36701. [[CrossRef](#)]
42. Wen, S.; Zhang, B.; Hao, P.; Lam, H.K.; Wang, H. Fuzzy fractional order force control of 6PUS-UPU redundantly actuated parallel robot based on inner model position control structure. *Eng. Appl. Artif. Intell.* **2017**, *65*, 200–211. [[CrossRef](#)]
43. Jan, R.M.; Tseng, C.S.; Liu, R.J. Robust PID control design for permanent magnet synchronous motor: A genetic approach. *Electr. Power Syst. Res.* **2008**, *78*, 1161–1168. [[CrossRef](#)]
44. Wen, S.; Qin, G.; Zhang, B.; Lam, H.K.; Zhao, Y.; Wang, H. The study of model predictive control algorithm based on the force/position control scheme of the 5-DOF redundant actuation parallel robot. *Robot. Auton. Syst.* **2016**, *79*, 12–25. [[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.