2020 17th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)
Mexico City, Mexico. November 11-13, 2020

BFO-GA Interval Type-2 Fuzzy PD Control applied to a Magnetic Levitation System.

Jesús Salazar de León
Tecnológico Nacional de México
Instituto Tecnológico de La Laguna
Torreón, Coahuila de Zaragoza.
Email: jesus_de_leon@outlook.com

Ramón García Hernández
Tecnológico Nacional de México
Instituto Tecnológico de La Laguna
Torreón, Coahuila de Zaragoza.
Email: rgarciah@correo.itlalaguna.edu.mx

Miguel Ángel Llama Leal
Tecnológico Nacional de México
Instituto Tecnológico de La Laguna
Torreón, Coahuila de Zaragoza.
Email: mllama@correo.itlalaguna.edu.mx

***Abstract*—This paper presents the optimization of two fuzzy logic control (FLC) schemes applied to a magnetic levitation (MagLev) system. The controller optimization is performed using a hybrid algorithm that combines the techniques of genetic algorithms (GA) and Bacterial Foraging Optimization (BFO) algorithm. Comparisons with a type-1 fuzzy PD (T1-FPD) and an interval type-2 fuzzy PD (IT2-FPD) controllers applied to a MagLev system are addressed via simulations in MATLAB - Simulink. The obtained type-2 controllers are based on the design of type-1 fuzzy logic controllers; from these, the uncertainty is generated to have the type-2 membership functions. The control law is directly calculated from the error position signal of the steel ball, as well as its respective velocity error, which presents interesting control qualities.**

I. Introduction

In the control area it is usually required that some plant or system performs a particular task; for this reason, it is necessary to design controllers that optimally fulfill such a task. However, doing this is not easy due to the uncertainty associated to the model dynamics of the plant. Fuzzy logic allows us to represent common knowledge in a mathematical language through the theory of fuzzy sets and membership functions (MFs) associated with them. It has been shown that type-2 fuzzy systems are capable of dealing with sources of uncertainty such as: multiple meanings of linguistic labels, noisy training information and noise-contaminated measures. In fact, type-2 fuzzy logic is a better option than traditional fuzzy logic for dealing with phenomena whose nature is non-linear and stochastic at the same time.

Type-2 fuzzy sets (T2FSs) are used to model uncertainty and inaccuracy in a better way. These T2FSs were originally presented by Zadeh in 1975 and are essentially sets where the fuzzy degree of membership is a type-1 fuzzy set (T1FS) [1]. New concepts were introduced by Mendel and Liang [2], allowing the characterization of a T2FS with an upper membership function (UMF) and a lower membership function (LMF). Each of these two functions can be represented by a membership function of a T1FS.

Magnetic levitation system is a well-studied plant, and control solutions are proposed in [3], where the authors present a fuzzy control scheme for regulation problem applied to this system. The main contribution is the real-time application of this controller and its comparison with other linear and non-linear schemes. However, bio-inspired algorithms have become an important research area due to intent to emulate nature in order to help us solve real-life complex problems, particularly for optimization. Bio-inspired algorithms for engineering propose novel algorithms to solve real-life complex problems in classification, approximation, vision, pattern recognition, identification, and control. Such algorithms include combined well-known bio-inspired algorithms and new ones, including both rigorous analyses, as well as unique applications [4]. For this reason, a type-2 fuzzy logic approach combined with a Bacterial Foraging Optimization - Genetic Algorithm (BFO-GA) is used in order to solve this problem.

II. System description

The magnetic levitation (MagLev) system can be seen as a magnetic circuit with air gap, which is composed of a magnetic object in levitation (steel ball), which moves below an electromagnet that is fixed to a base, as it is shown in Fig. 1 where $u(t)$ is the electromagnet voltage [V], $i(t)$ is the

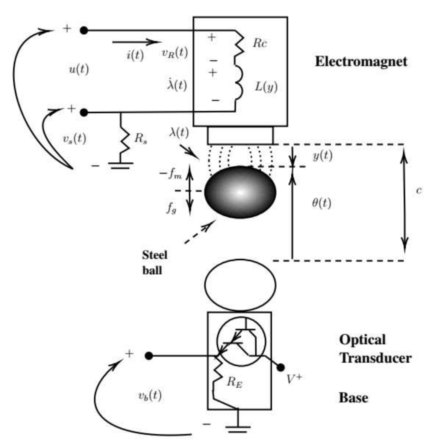

Figure 1. Magnetic levitation system (MagLev).

electromagnet current [A], $\dot{\lambda}(t)$ is the induced electromotive force in the electromagnet [V], $\lambda(t)$ is the magnetic flux in the air gap [Wb], f_m is the attraction force of the electromagnet

[N], $f_g = mg$ is the force due to the earth gravity [N]. Table I shows the main parameters (e.g. mechanical and electrical specifications, conversion factors, constants, etc.) associated with the MagLev specialty plant. Some of these parameters can be used for mathematical modeling of the MagLev system as well as to obtain the steel ball's equation of motion (EOM). The variable $\theta(t)$ is the position of the steel ball

Table I
MAGLEV SYSTEM PARAMETERS

Notation	Description	Value	Units
$i_{max}(t)$	Maximum continuous coil current	$\pm$ 3	A
L_c	Coil inductance	412.5 $\times 10^{-3}$	H
R_c	Coil resistance	10	Ω
l_c	Coil length	0.0825	m
R_s	Current sense resistance	1	Ω
m	Steel ball mass	0.068	Kg
g	Gravitational constant	9.81	m/s^2

measured with respect to the nominal air gap denoted by the constant $c = 0.014$ [m]. The MagLev system has a variable inductance described by

$$L(y) = \frac{k}{y(t)} = \frac{k}{c - \theta(t)} \quad (1)$$

where $k = 6.5308 \times 10^{-5}$ is the electromagnet force constant $[H - m]$. The relationship between $\lambda(t)$ and $i(t)$ for MagLev system is expressed as:

$$\lambda(t) = L(y)i(t) \quad (2)$$

The dynamic model in terms of the state vector $[\ y(t)\ i(t)\ \dot{y}(t)\ ]^T$ results as:

$$\frac{d}{dt}\begin{bmatrix} y(t) \\ i(t) \\ \dot{y}(t) \end{bmatrix} = \begin{bmatrix} \dot{y}(t) \\ \frac{y}{k}[u(t) - Ri(t)] + \frac{i(t)\dot{y}(t)}{y(t)} \\ -\frac{k}{2m}\frac{i^2(t)}{y^2(t)} + g \end{bmatrix} \quad (3)$$

where the equilibrium point for a constant $u(t) = u_{eq}$ in open loop is $[y_{eq}\ i_{eq}\ 0]^T$, where $R = R_c + R_s$, y_{eq} is the operating position; $i_{eq} = k_{ff}\ y_{eq}$ is the operating current, with $k_{ff} = \sqrt{\frac{2mg}{k}} = 142.92\ [A/m]$ a constant feedback, and $u_{eq} = R_{ieq}$ the constant input voltage, y is the position of the steel ball and $\dot{y}$ is the velocity of displacement of the steel ball, which is computed numerically.

III. TYPE-1 FUZZY SETS

A fuzzy set can contain elements partially, that is, the property that a x element belongs to the A set, such that $x \in A$ can be true with a partial degree of truth. This degree of membership may be a proposition in the context of fuzzy logic, and not of the standard binary logic, which only admits two values: true or false. The degree of membership varies within the range of $[0, 1]$ and labels are used to facilitate the classification of each entry to the MFs.

A. Type-1 Fuzzy PD control (T1-FPD)

The block diagram of T1-FPD controller applied to a MagLev system is shown in Fig. 2. The control law is

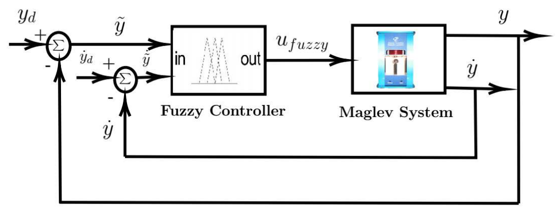

Figure 2. Block diagram of the T1-FPD controller.

analytically expressed as

$$u_{fuzzy}(t) = K_{pc}(i_d(t) - i(t)) \quad (4)$$

where $i_d(t) = K_p(\tilde{y}, \dot{\tilde{y}})\tilde{y}(t) + K_d(\tilde{y}, \dot{\tilde{y}})\dot{\tilde{y}}(t)$ is the desired current in the inner loop; $K_p(\tilde{y}, \dot{\tilde{y}})$ and $K_d(\tilde{y}, \dot{\tilde{y}})$ are the fuzzy position and velocity gains respectively; $\tilde{y}(t) = y_d - y$, $\dot{\tilde{y}} = \dot{y_d} - \dot{y}$, y_d is a sinusoidal signal with an amplitude of 0.012 [m]; K_{pc} is the proportional gain current.
The fuzzy rule base for MagLev system of two inputs, $\tilde{y}$ and $\dot{\tilde{y}}$, and one output u_{fuzzy} is formed by taking all possible combinations of the MFs for each entry.

The MFs used for the design of T1-FPD controller are shown in Figs. 3 and 4. The selected rules follow the structure

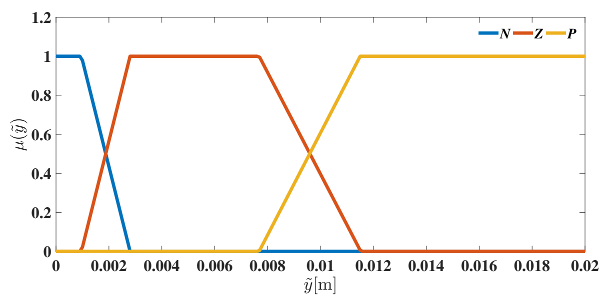

Figure 3. Type-1 MFs for $\tilde{y}$ input.

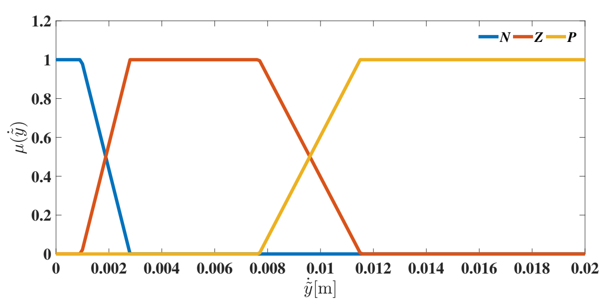

Figure 4. Type-1 MFs for $\dot{\tilde{y}}$ input.

Table II
TYPE-1 FUZZY RULE BASE.

$\tilde{y}$ \ $\dot{\tilde{y}}$	N	Z	P
N	Kp, Kd → BIG	Kp, Kd → BIG	Kp, Kd → $MEDIUM$
Z	Kp, Kd → $MEDIUM$	Kp, Kd → $MEDIUM$	Kp, Kd → $MEDIUM$
P	Kp, Kd → $MEDIUM$	Kp, Kd → $SMALL$	Kp, Kd → $SMALL$

shown in table II. The minimum inference engine along with weighted average defuzzification are used in order to evaluate the fuzzy rule base.

IV. TYPE-2 FUZZY SETS

In some contexts, it is feasible to identify sources of uncertainty associated with natural language, where this uncertainty can be weighted with degrees of membership that are in turn fuzzy, thus considering them as type-2 fuzzy uncertainty. In general, a T2FS is a generalization of a T1FS along with a secondary source of uncertainty associated with the definition of a word or set A. This additional source of uncertainty is represented as a secondary MF associated with each value of $x \in X$ (or universe of discourse); in other words:

$$\tilde{A} = X \to F([0,1]) \tag{5}$$

where $F([0,1])$ is the set of all T1FSs that can be defined within the universe of discourse $x \in X$.
T2FSs, whose secondary MF is an interval (that is, the uncertainty associated with language is treated fairly), are called interval type-2 fuzzy sets or IT2FSs [2], since the degrees of membership assigned to the elements of the universe of discourse are an interval.
In [5], the author explains that an IT2FS is expressed by two membership vector functions, where one represents the degree of membership in X and the other gives a weighting to each of the T1FSs as

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) | x \in X\} \tag{6}$$

Specifically, an IT2FS is defined as

$$\tilde{A} = \{(x,u), J_x, f_x | x \in X; u \in [0,1]\} \Leftrightarrow f_x(u) = 1 \tag{7}$$

where the primary fuzzy set J_x is weighted by the fuzzy set $f_x(u)$ as a secondary MF, x is the primary variable, J_x is an interval $[0,1]$ representing the primary membership of x, u is the secondary variable and $f_x(u)$ is the secondary membership function of x. The Footprint of Uncertainty (FOU) of T2FSs defines the uncertainty of $\tilde{A}$ as the union of all primary memberships; this can be noted in Fig. 5 and is described by

$$FOU(\tilde{A}) = \cup_{x \in X} J_x \tag{8}$$

which is limited by two MFs: an Upper Membership Function (UMF) $\mu_{\tilde{A}\ max}$ and a Lower Membership Function (LMF) $\mu_{\tilde{A}\ min}$, both are T1FSs and have e embedded sets A_e; consequently, the degree of membership of each element of an IT2FS is an interval $[\mu_{\tilde{A}\ min},\ \mu_{\tilde{A}\ max}]$ [6].
The set has an expected value obtained through a reduction of

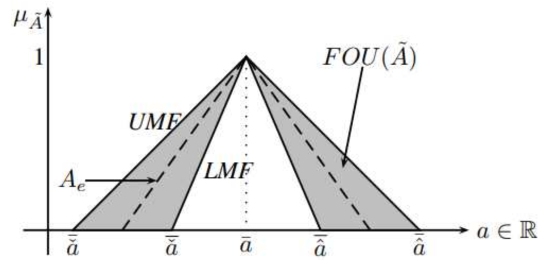

Figure 5. Type-2 Membership Function

type that gives information about the possible behavior of the problem. This measure is the centroid that corresponds to a closed interval formed by all the centroids of the type-1 fuzzy embedded assemblies contained in the FOU of $\tilde{A}$ as follows

$$C(\tilde{A}) = \frac{1}{[C_l(\tilde{A}); C_u(\tilde{A})]} = \frac{1}{[C_l, C_u]} \tag{9}$$

where $C_l(\tilde{A})$ and $C_u(\tilde{A})$ are the lower and upper levels of the centroid. The IT2FSs have an infinite amount of centroids embedded in their FOU. Their calculation is an iterative process where the general equations for calculating the dimensions are

$$C_l(\tilde{A}) = \frac{(\sum_{i=1}^{L} A_i \mu_{\tilde{A}\ max}(x_i) + \sum_{i=L+1}^{N} A_i \mu_{\tilde{A}\ min}(x_i))}{(\sum_{i=1}^{L} \mu_{\tilde{A}\ max}(x_i) + \sum_{i=L+1}^{N} \mu_{\tilde{A}\ min}(x_i))} \tag{10}$$

$$C_u(\tilde{A}) = \frac{(\sum_{i=1}^{U} A_i \mu_{\tilde{A}\ max}(x_i) + \sum_{i=U+1}^{N} A_i \mu_{\tilde{A}\ min}(x_i))}{(\sum_{i=1}^{U} \mu_{\tilde{A}\ min}(x_i) + \sum_{i=U+1}^{N} \mu_{\tilde{A}\ max}(x_i))} \tag{11}$$

where $\mu_{\tilde{A}\ max}(x_i)$ and $\mu_{\tilde{A}\ min}(x_i)$ are the upper and lower values of the MF, L the point that marks the change from $\mu_{\tilde{A}\ max}(x_i)$ to $\mu_{\tilde{A}\ min}(x_i)$, and U is the point that marks the change of $\mu_{\tilde{A}\ min}(x_i)$ and $\mu_{\tilde{A}\ max}(x_i)$.

A. *Interval Type-2 Fuzzy PD control*

Recently, researchers began investigating interval type-2 (IT2) fuzzy logic controllers that have demonstrated significant improvements in control performance compared to their type-1 counterpart. In this subsection, the general structure of the proposed interval type-2 fuzzy PD (IT2-FPD) controller will be presented. In the design of the IT2-FPD, the same control law in (4), the 3×3 symmetric rule base and the consequent MFs of the T1-FPD will be used. Therefore, the structure of the IT2-FPD rule is defined as follows:

$$R_q : IF\ \ \tilde{Y}\ \ is\ \ \tilde{A}_{1j}\ and\ \ \dot{\tilde{Y}}\ is\ \ \tilde{A}_{2j}$$
$$THEN\ \ U\ \ is\ \ C_q j,\ \ j = 1,2,3; q = 1,\dots,Q = 9$$

where C_q is the consistent set, Q is the number of rules, $\tilde{A}_{1j}$ and $\tilde{A}_{2j}$ are the IT2FS of the inputs, j is the number of input and output MFs of the IT2FS. The IT2FS will be denoted as N (Negative), Z (Zero) and P (Positive). The antecedent can be described in terms of UMF ($\mu_{\tilde{A}_1\ max}$ and $\mu_{\tilde{A}_2\ max}$) and in LMF ($\mu_{\tilde{A}_1\ min}$ and $\mu_{A_2\ \tilde{}min}$) that creates the FOU (which provides an additional degree of freedom). The implemented controller uses the weighted average defuzzification method. The MFs used are shown in Figs. 6 and 7. The fuzzy rule base applied to the system with $\tilde{Y}$ and $\dot{\tilde{Y}}$ entries, and an output u,

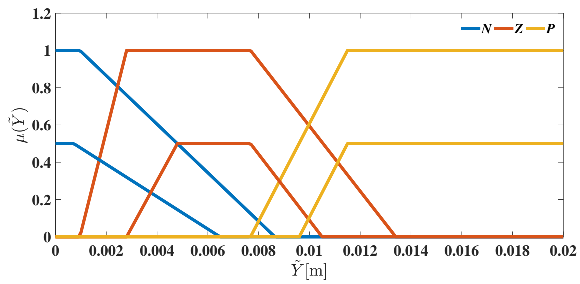

Figure 6. Interval Type-2 Membership functions for $\tilde{Y}$ input.

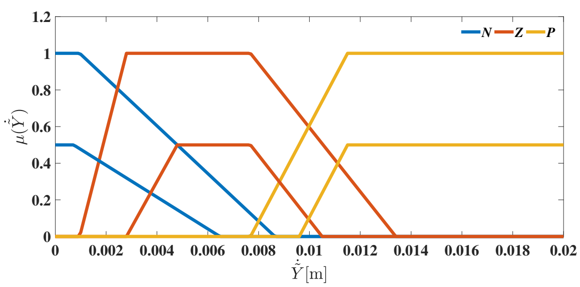

Figure 7. Interval Type-2 Membership functions for $\dot{\tilde{Y}}$ input.

is formed by taking all possible combinations of membership functions for each entry; in this way, table III is developed:

Table III
INTERVAL TYPE-2 FUZZY RULE BASE.

$\tilde{Y}$ \ $\dot{\tilde{Y}}$	N	Z	P
N	Kp, Kd $\rightarrow$ BIG	Kp, Kd $\rightarrow$ BIG	Kp, Kd $\rightarrow$ $MEDIUM$
Z	Kp, Kd $\rightarrow$ $MEDIUM$	Kp, Kd $\rightarrow$ $MEDIUM$	Kp, Kd $\rightarrow$ $MEDIUM$
P	Kp, Kd $\rightarrow$ $MEDIUM$	Kp, Kd $\rightarrow$ $SMALL$	Kp, Kd $\rightarrow$ $SMALL$

The block diagram of IT2-FPD controller is shown in Fig.8:

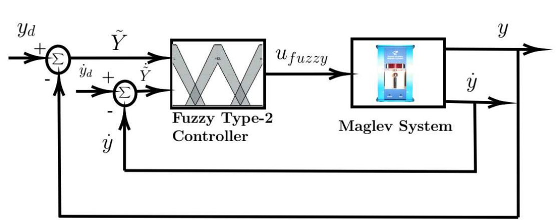

Figure 8. Block diagram of the IT2-FPD controller.

V. BFO-GA OPTIMIZATION ALGORITHM

There are totally different optimization strategies and algorithms; they are classified into deterministic and stochastic approaches [7], [8].

A. Bacterial Foraging Optimization

The Bacterial Foraging Optimization (BFO) algorithm was first proposed in 2002 by Passino. It is inspired by the foraging behavior and chemotactic bacteria, especially Escherichia coli (E. coli) in our intestine. By running smooth and tumbling, it can be moved to the escape area of nutrients and poison zone in the environment. Chemotaxis is more attractive behavior of bacteria, and it has been studied by many researchers [9], [10]. Due to their group response, the social behavior of the colony of E. coli is very interesting for engineering; it allows them to get quickly and easily the best food supply with the lowest possible risk. These bacteria can communicate through chemical exchanges. The bacteria that have achieved a safe place to feed, communicate it to others who come to such place; the greater the amount of existing food, the stronger is the signal. Similarly, if the bacteria are in a dangerous place, with agents that may threaten the colony, they warn others to stay away from that place. This behavior can be represented mathematically forage as a kind of swarm intelligence [9]. So, BFO mimics the four principal mechanisms of real bacterial systems: chemotaxis, swarming, reproduction and elimination-dispersal to solve optimization problems.

B. Genetic Algorithm

Genetic algorithms are a category of algorithms supported the abstraction of Darwin's evolution of biological systems, pioneered by Holland and his collaborators within the 1960s and 1970s [11]. Holland used genetic operators corresponding to the crossover and recombination, mutation, and choice within the study of adaptational and artificial systems. Every resolution is encoded in a very string (often binary or decimal) referred to as chromosome.

GA is mainly based on the fitness function, and the selection, crossover and mutation operation are performed. Individual with best performance will be obtained in the process of continuous iteration.

C. BFO-GA

BFO algorithm is combined with GA to form BFO-GA, and BFO algorithm is regarded as the local search method. The scheme of BFO-GA is shown in Fig. 9. In the first block of the BFO-GA scheme, the data is submitted to the GA scheme, in this stage the coding of the data to be optimized is presented; in the next block, the initial population is declared, they are evaluated based on the optimization criteria that are proposed, a selection of the values closest to the optimal ones is then made; then the process of crossing these values is carried out to finally subject them to the mutation phase. The new values obtained are sent to the BFO optimization scheme.

These obtained data in GA undergo the chemotactic calculation process, then undergo the bacterial reproduction process; finally, the obtained values undergo the *elimination-dispersion* step. Once the desired values are obtained in this part, they are sent again to the proficiency calculation proces. This process continues until the optimal system values are found. The programming of the proposed controller was carried out by implementing dependent code using the conventional MATLAB platform. This code respect the structure of the diagram shown in Fig. 9 which is a modification of the algorithm proposed in

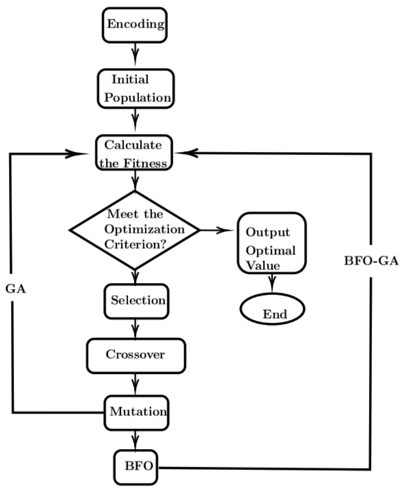

Figure 9. Block diagram of the BFO-GA.

[12]. The block diagram for optimized IT2-FPD controller is shown in Fig. 10. For the T1-FPD controller the new values of

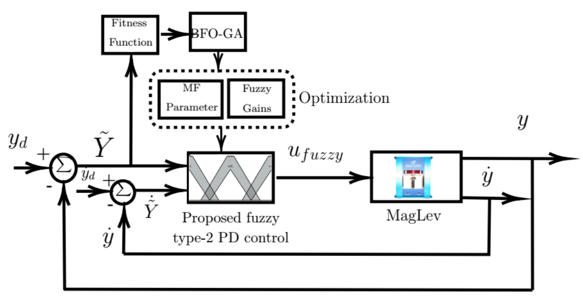

Figure 10. Block diagram of the Optimized Type-2 Fuzzy Controller.

the MFs obtained are shown in figures 11 and 12. The fuzzy

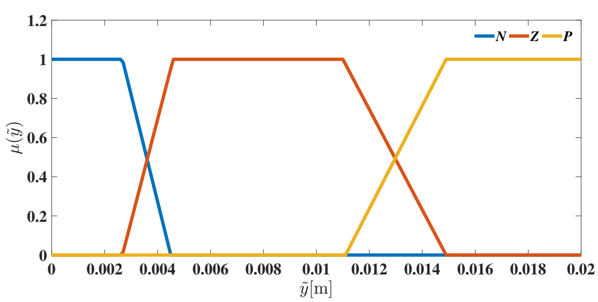

Figure 11. Optimized type-1 membership functions $\tilde{y}$.

rule base remains the same, however the gains were modified by the BFO-GA. For the IT2-FPD controller the new values of the MFs obtained are shown in figures 13 and 14. The fuzzy rule base remains the same, however the gains were modified by the BFO-GA.

VI. Results

This section shows the obtained results for trajectory tracking and position error when each used is applied.

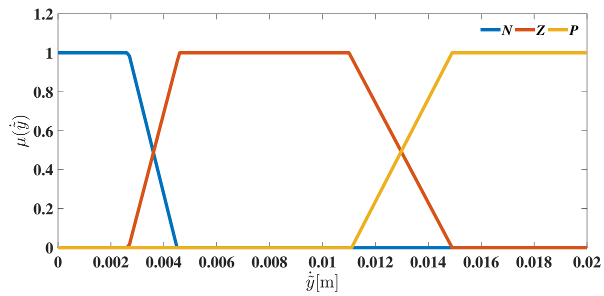

Figure 12. Optimized type-1 membership functions $\dot{\tilde{y}}$.

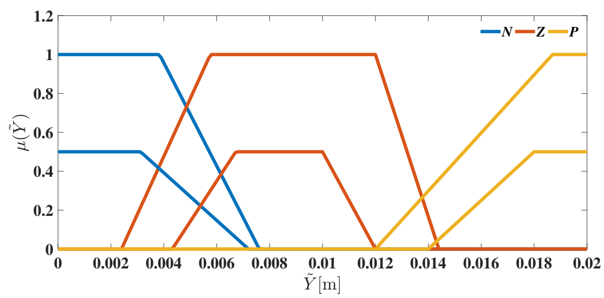

Figure 13. Optimized type-2 membership functions $\tilde{Y}$.

The gain tuning for T1-FPD and IT2-FPD controllers was carried out with the analytical method of location of the poles and zeros for the linearized system, however an empirical retuning was carried out during the simulation process to increase the speed of the system response. The gains that were obtained for each controller are shown in Table IV. The

Table IV
Used gains for each controller.

Controller	Kp	Kd
T1-FPD	[150 100 50]	[15 5 2]
IT2-FPD	[140 80 60]	[12 6 2]
BFO-GA T1-FPD	[60 45 25]	[4 2 1]
BFO-GA IT2-FPD	[100 70 45]	[8 4 1]

computation of gains K_p and K_d for T1-FPD controller were performed following the criteria seen in [13] and then were used as a basis to determine the gains of IT2-FPD controller.

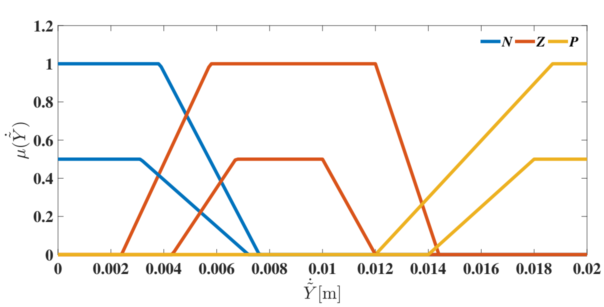

Figure 14. Optimized type-2 membership functions $\dot{\tilde{Y}}$.

The simulation of the IT2-FPD controller was possible using the Toolbox developed in [14].
Fig. 15 shows the position of the steel ball in regulation, which is subject to various changes in the sinusoidal desired reference signal for all controllers.
It is important to note the use of a sinusoidal reference signal because the implemented BFO-GA requires a continuous signal for execution and sampling.
As can be seen in Fig. 15, the response of the BFO-GA IT2-FPD controller is smoother when it reaches the reference and manages to reach it before the BFO-GA T1-FPD controller, which has oscillations in the first 0.35 seconds of the simulation, reaching a maximum peak of approximately 1 cm above the reference signal. Fig. 16 shows the position error for each controller, it can be determined at a glance that the BFO-GA IT2-FPD controller has lower oscillations and a lower error rate compared to the others controllers included in this work.

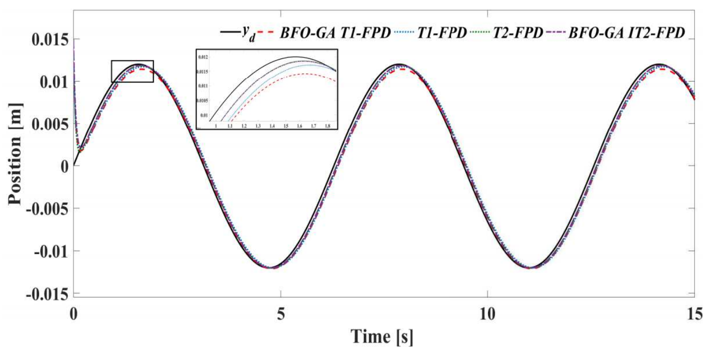

Figure 15. Ball position response.

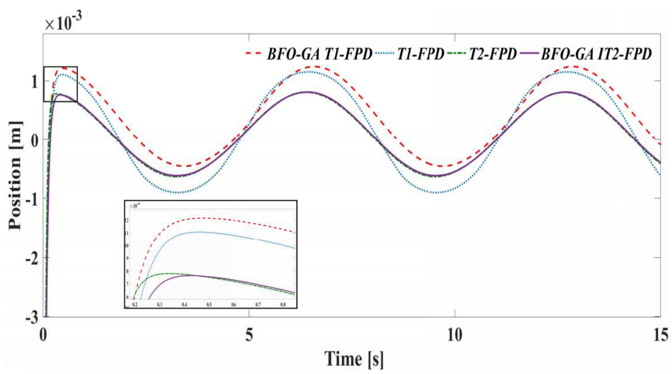

Figure 16. Position error.

VII. Comparative analysis

This section shows a comparison analysis for the mean square error (MSE) of the position error. As can be seen in Table V, the application of the BFO-GA hybrid optimizer considerably improved the behavior of the IT2-FPD controller.

Table V
Performance index.

T1-Controller	MSE ($\tilde{y}$)	T2-Controller	MSE ($\tilde{Y}$)
T1-FPD	6.21×10^{-4}	IT2-FPD	4.12×10^{-4}
BFO-GA T1-FPD	9.41×10^{-5}	BFO-GA IT2-FPD	3.92×10^{-5}

VIII. Conclusion

The main objective of this research is the implementation of bio-inspired algorithms for optimization of fuzzy logic controllers and explore ways to further improve their performance to reach an optimal solution. After a careful analysis of the existing algorithms, this research work proposed the BFO-GA algorithm to perform a multiple optimization, where the optimal values are obtained, both for the MFs and the gains of each controller; which for the best knowledge of the authors, is the first work that uses the application of a hybrid algorithm to carry out this task.
Based on the simulations results obtained in Figs. 15-16, it can be seen that the response of the BFO-GA IT2-FPD controller is softer than the others when monitoring the reference, since it presents a lower overshoot, a smaller error, and although it reachs the reference slightly later, it presents a better behavior, which for a real-time application will have a better performance and lower damage to the system actuators. In the same way it can be seen that the position error presents lower shocks for type-2 fuzzy controller than for type-1 fuzzy controller. In the future, the proposed algorithm may be applied for an experimental test.

Acknowledgements

The authors thank for the support received through the projects of the Tecnologico Nacional de Mexico and CONACyT (957486).

References

[1] L.A. Zadeh Fuzzy logic. Computer, vol. 21(4) pp. 83-93 1988.
[2] Q. Liang J. Mendel "Interval type-2 fuzzy logic systems: Theory and design" IEEE Trans. on Fuzzy Systems vol. 8 pp. 535-550 2000.
[3] J. Ollervides, V. Santibáñez, M. Llama and A. Dzul, Aplicación de Control Borroso a un Sistema de Suspensión Magnética: Comparación Experimental. Revista Iberoamericana de Automática e Informática industrial, 7(3), pp. 63-71, 2010.
[4] A.Y. Alanis, N. Arana-Daniel, and C. Lopez-Franco, Bio-inspired Algorithms for Engineering, Butterworth-Heinemann ELSEVIER, USA, 2018.
[5] O. Castillo, Interval Type-2 Fuzzy Logic for Control Applications, 2010 IEEE International Conference on Granular Computing, 14-16 Aug. 2010.
[6] G. Klir and B. Yuan, Fuzzy sets and fuzzy logic, vol. 4. Prentice Hall New Jersey, 1995.
[7] C. A. Floudas, Deterministic Global Optimization: Theory, Methods and Applications, vol. 37 of Nonconvex Optimization and Its Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.
[8] J. C. Spall, Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, Wiley-Interscience Series in Discrete Mathematics and Optimization, Wiley-Interscience, Hoboken, NJ, USA, 2003.
[9] B. Bhushan , M. Singh Adaptive control of nonlinear systems using bacterial foraging algorithm. Int J Comput Electr Eng 2011;3(3):335.
[10] S. Das, S. Dasgupta, A. Biswas, A. Abraham, A. Konar On stability of the chemotactic dynamics in bacterial-foraging optimization algorithm. Trans Syst Man Cyber Part A May 2009; 39(3): 670–9.
[11] J. Salazar, R. García, E. J. Ollervides, "Desarrollo de un algoritmo de control inteligente para un sistema de levitación magnética", Núm. 39 (2019): Revista Estudios de la Ciénega - Enero-Junio 2019, pag 65-82.
[12] Z. Lan, M. Hongmei, S. Changyin and W. Xinqiao, "A Novel Indoor Intelligent Location Algorithm Based on GA-BFO," 2016 3rd International Conference on Information Science and Control Engineering (ICISCE), Beijing, 2016, pp. 1222-1225.
[13] H. Mühlenbein, (1997). Genetic algorithms.
[14] A. Taskin y T. Kumbasar, 2015. An open source Matlab/Simulink Toolbox for Interval Type-2 Fuzzy Logic Systems, IEEE Symposium Series on Computational Intelligence – SSCI 2015, Cape Town, South Africa.