REVISTA ELECTRO, Vol. 44 pp. 83-87, Oct 2022, Chihuahua, Chih. México
http:// electro.itchihuahua.edu.mx/revista
ISSN 1405-2172

COMPARISON BETWEEN EVOLUTIONARY ALGORITHMS IN HEIGHT ADJUSTMENT IN A PNEUMATIC LEVITATOR

García-Mejía Juan Fernando[1], López-Pérez Miguel A[1]. Garduño-Ruíz Genaro[1], Segundo Lovera José Felix[1],
Granda Gutiérrez Everardo Efrén[1], Vázquez-Robledo Ricardo Arturo[2], Colín Rivas Laura[1]
[1]Universidad Autónoma del Estado de México
Centro Universitario UAEMEX Atlacomulco
Carretera Toluca-Atlacomulco km.60
[2]Agencia Espacial Mexicana
Centro Regional de Desarrollo Espacial Atlacomulco
Parque Industrial Santa Barbara
Tel: 7121220446
e-mail: fgarciam@uaemex.mx

ABSTRACT.

This proposal presents a comparative study between three heuristic methods applied in the gain tuning of a Proportional-Integral-Derivative controller, applied to the height regulation of a pneumatic levitator, this is a mechanical device that aims to raise an object from the variations of a certain air flow supplied by a fan. The purpose of this comparison is to determine the use feasibility in automatic control of a new evolutionary algorithm called Cuckoo Search, a heuristic supported by the behavior of Cuckoo birds, which is modelable by Levy's flight, a stochastic process that depends only on its present value. The results show that the Cuckoo Search is more stable with respect to the number of executions than the genetic algorithms and differential evolution used in this case of study.

Keywords: Cuckoo Search, Pneumatic levitator, Levy's flight

1. INTRODUCTION

The basic principle of aerodynamic levitation is based on the suspension of an arbitrary object in the air using air flow. The process of suspending an object in the air in an open or closed environment without any physical force applied directly to the body is called levitation, which results from the existence of a force that counteracts the weight of the levitating body or object [1].

These systems includes magnetic, acoustic, pneumatic or electrostatic levitation in which the challenge is to suspend an object in the air for a given time or position. Some problems are disturbances that are set points of the movement or that are transmitted from the environment [2].

Because the pneumatic levitator requires a continuous flow of air to keep the object in suspension, control is required at the source of the air flow, specifically at the voltage source of the fan used for this purpose. Therefore, by biasing the source with a series of pulses it is possible to monitor the position of the object and the applied voltage to obtain the transfer function that mathematically models the response of the levitator. The general model of the pneumatic levitator can be seen in Fig. 1, the fan is located at the bottom, the air flow is maintained in a tube and as long as there is air flow the object levitates at the top of the tube, on the other hand the model has a solid wooden structure.

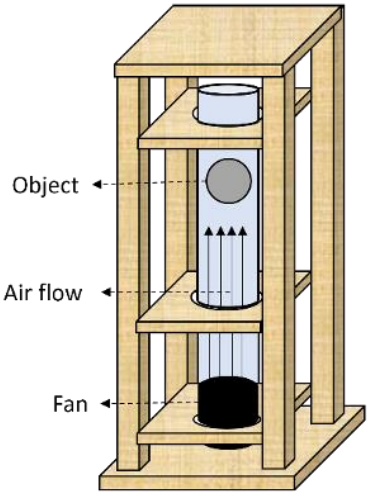

Fig. 1. Pneumatic levitator structure

It is possible to classify the works related to the present research proposal in some axes, in the first of them, optimal control techniques are used such as integral predictive actions [3], classic and robust PID controllers based on the standard [4], another example is the regulation of the object position to be levitated by means of a non-linear type PID controller [5]. A second axis is constituted by regulating the height of the levitating object, it is regulated by means of a PID type controller whose gains are tuned by means of evolutionary algorithms such as genetic algorithms, its reduced version is known as genetic microalgorithm [6] in addition the problem of tuning these gains is treated by means of evolution differential algorithms [7]. The approach of height regulation by means of evolutionary algorithms is the object of study of this proposal, therefore, it is proposed to document the differences between differential evolution algorithms (it should be noted that in [7] only one version of differential evolution algorithm was analyzed), genetic algorithms and a newfangled evolutionary algorithm: the Cuckoo Search

2. IMPLEMENTED ALGORITHMS

Evolutionary algorithms have their antecedent in "Esempi Numerici di processi di evoluzione", a work published in 1954 by Nils Aall Barricelli from the Charles Darwin's theory of evolution of species and Mendel's laws of inheritance. The flowchart of an evolutionary algorithm is shown in figure 4

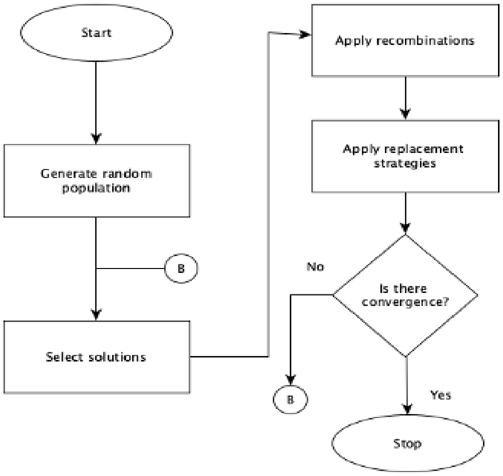

Figure 2 Generic Evolutionary Algorithm.

2.1. Genetic Algorithm

The Genetic algorithms are methods used to find the most optimal solutions to search and optimisation problems. Based on Darwin's theory of evolution of species, these algorithms are based on the genetic process of the species, so their main idea is to generate simulations that originate possible solutions to a given problem. Fig 3 shows a graphical representation of a genetic algorithm [8].

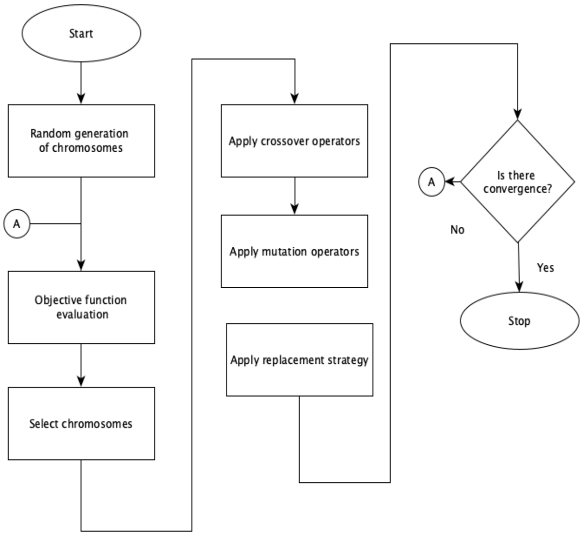
Start
Random generation of chromosomes
Apply crossover operators
A
Is there convergence?
A
No
Apply mutation operators
Yes
Objective function evaluation
Stop
Apply replacement strategy
Select chromosomes

Fig. 3. Flowchart of a genetic algorithm

The operators implemented in the genetic algorithm are the Blend Crossover (BLX-∝), expressed by chromosomes C^{H1}y C^{H2} expressed in equation 1 [8]

$$C^H = rand[(h_{min} - I * \propto), (h_{max} - I * \propto)] \qquad (1)$$

Where C^H represents the descendant chromosome, h_{min} is the minimum value of chromosomes C_i^1, C_i^2 , h_{max} is the maximum value of the chromosomes C_i^1, C_i^2 and the parameter $\propto$ is represented as a randomly generated value with a uniform distribution $rand[0,1]$.Thus, the arithmetic crossover operator is obtained, and it is expressed in equation 2 [8]

$$\begin{aligned} C^{H1} &= C_1^N *\propto + C_2^N * \ (1-\propto) \\ C^{H2} &= C_2^N *\propto + C_1^N * \ (1-\propto) \end{aligned} \qquad (2)$$

Where N represents the n-th position of the chromosome vector, in the same way $\propto$ *=rand[0,1]* following a uniform distribution.

For the mutation, a modification operator with Gaussian distribution was implemented with mean 0 and variance defined by each of the genes, expressed in (3) [8]

$$\sigma_k = \frac{T-t}{T} \frac{(g_k^{max} - g_k^{min})}{3} \qquad (3)$$

Where t is the occurrence of each generation, T is the maximum number of generations given the mutation of the chromosomes in the algorithm, expressed in (4)

$$C' = C + N(0, \sigma_k) \qquad (4)$$

2.2. Differential Evolution Algorithms

The differential evolution algorithms, seen in Fig 4, are a subset of evolutionary algorithms and are stochastic search tools, mainly population-based, in order to obtain possible candidate solutions given the objective function through iterations to evolve the initial population.

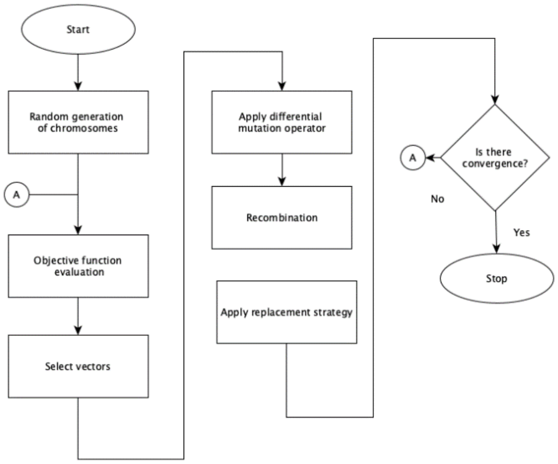
Start
Random generation of chromosomes
Apply differential mutation operator
Is there convergence?
A
A
No
Recombination
Yes
Objective function evaluation
Stop
Apply replacement strategy
Select vectors

Fig. 4. Flowchart of a Differential evolution algorithm

Recombination and mutation are the variation operators used to generate new solutions, and a replacement mechanism provides the ability to maintain a population size. The replacement strategy is based on the competition between the vectors descended from the recombination process (children) with those of the original population, which is generated by means of a normal probability distribution. The mutation aims to generate variations that displace the solution vectors in the correct direction and magnitude and in its simplest form is represented in equation 5 [8]

$$\vec{v}_g = \vec{x}_{best,G} + F\left(\vec{x}_{r1,G} - \vec{x}_{r2,G}\right) \qquad (5)$$

Where $F \in [0,1]$ is a scale factor that controls the vector difference described in $\vec{x}_{r1,G} - \vec{x}_{r2,G}$, G is the current generation and $r_1 \neq r_2 \neq r_3$ represent the indices of the vectors used in the mutation operator.

Another way to perform the mutation is the one expressed in equations 6 and 7

$$\vec{v}_g = \vec{x}_{i,G} + F\left(\vec{x}_{best,G} - \vec{x}_{current,G}\right) + F\left(\vec{x}_{r1,G} - \vec{x}_{r2,G}\right) \qquad (6)$$

$$\vec{v}_g = \vec{x}_{i,G} + F\left(\vec{x}_{r1,G} - \vec{x}_{r2,G}\right) + F\left(\vec{x}_{r3,G} - \vec{x}_{r3,G}\right) \qquad (7)$$

Where *i* represents the current index of the vectors used in the mutation. Equations 5,6 and 7 are the essence of differential evolution algorithm schemes called DE/best/1, DE/current to best, and DE/best/2 On the other hand, recombination allows the exchange of information between the parent vector and the mutant vector by generating a descendant, where each of the elements of the child can be taken from the parent vector or the mutation vector $\vec{u}$ with a probability determined by the parameter CR which is within the range [0,1], this operator is described in equation 8. [11].

$$\vec{u}_{i,G} = \begin{cases} \vec{u}_{i,G} & \forall\, rand[0,1] < CR \\ \vec{x}_{i,G} & another\ case \end{cases} \qquad (8)$$

The replacement scheme applied to a maximization problem takes care of choosing which vectors $\vec{u}_{i,G}$ will pass to the next generation $G + 1$, it is defined by equation 9. Figure 2 shows the flowchart of a Differential Evolution algorithm [8].

$$\vec{x}_{i,G+1} = \begin{cases} \vec{u}_{i,G} & \forall\ f\left(\vec{u}_{i,G}\right) \geq f\left(\vec{x}_{i,G}\right) \\ \vec{x}_{i,G} & \forall\ f\left(\vec{u}_{i,G}\right) \leq f\left(\vec{x}_{i,G}\right) \end{cases} \qquad (9)$$

2.3. Cuckoo Search

One of the most recent bio-inspired algorithms is the one proposed by She Yang and Suash Deb called Cuckoo Search which consists of emulating the parasitic behavior of Cuckoo birds, which are native to Europe and North Africa and have a migratory behavior to Sub-Saharan Africa and Southeast Asia. A peculiarity of its parasitic behavior is that the female cuckoo deposits its eggs in different bird nests mimicking them with some that are similar.

This heuristic is based on the updating of nests (possible solutions) by means of a random number generator called Levy's flight in honor of the French mathematician Paul Pierre Lévy, who describes the movements of birds, bees and even sharks as a behavior of the fractal type. Levy's flight can be described by equation 10 [9]

Where $\beta = [0.25{,}3]$, is the size of the step that determines the scan in the search space T is the Gamma function, on the other hand the step size of the random generator is determined as $S = \frac{u}{|v|^{1/B}} \forall\, u \sim N(0, {\sigma_u}^2), v \sim N(0,1)$. From the above it is possible to update the nests by means of the expression shown in equation 11 [9]

$$\sigma_u = \left[\frac{\mathrm{T}(1+\beta) * sin\left(\frac{\pi * \beta}{2}\right)}{\mathrm{T}\left(\frac{1+\beta}{2}\right) * \beta * 2^{(\beta+1)/2}}\right]^{\frac{1}{\beta}} \qquad (10)$$

$$nest_i^{(t+1)} = nest_i^{(t)} + \propto * \, S\left(\max(nest^t) - min(nest^t)\right) * r \qquad (11)$$

The procedure of the Cuckoo Search algorithm consists of randomly selecting a nest, updating it by means of the expressions described above, then choosing a new nest, which will be replaced if when evaluating it, it is smaller than the modified nest. Figure 5 shows the Cuckoo Search flowchart

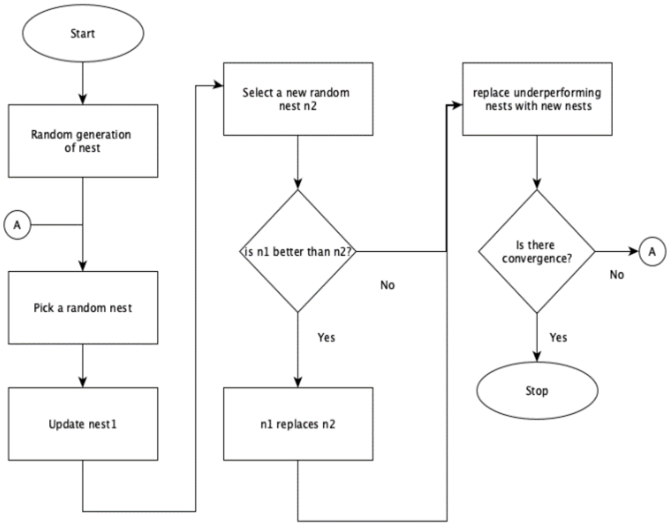

Fig. 5. Flowchart of a Cuckoo Search algorithm

3. METHODOLOGY

For the development of this proposal are used the pneumatic levitator implemented in [6] which is shown in fig 1. It has the characteristics of a 0.9-metre-high acrylic tube with a diameter of 50.8 mm, a 40 gram unicel sphere and a 6000 rpm fan, these elements have been placed in a wooden structure. The open-loop levitator transfer function is shown in the equation and its response to a 4.5 volt step input is shown in (14)

$$G(s)=\frac{16.1742e^{-3.4945s}}{(1+4.0512s)(1+4.6443s)(1+4.7953s)} \qquad (14)$$

From what is stated in section 2 of this document it is possible to construct the experiments shown in Table 1

Table 1 Description of experiments

Test 1	DE/best/1
Test 2	DE/current to best
Test 3	DE/best/2
Test 4	GA BLX-∝
Test 5	GA Aritmethic
Test 6	Cuckoo Search

To carry out this research proposal, the methodological scheme shown in Fig 6 is used and the following objective function shown in equation 4 is proposed.

$$f_{obj}(k_p,k_i,k_d)=\frac{1}{1+e_{rms}} \qquad (14)$$

Where $e_{rms}=ref-sal;\ e_{rms}=ref-G(s)*PID$

It should be noted that the statistical study is carried out from 40 iterations for each proposed algorithm

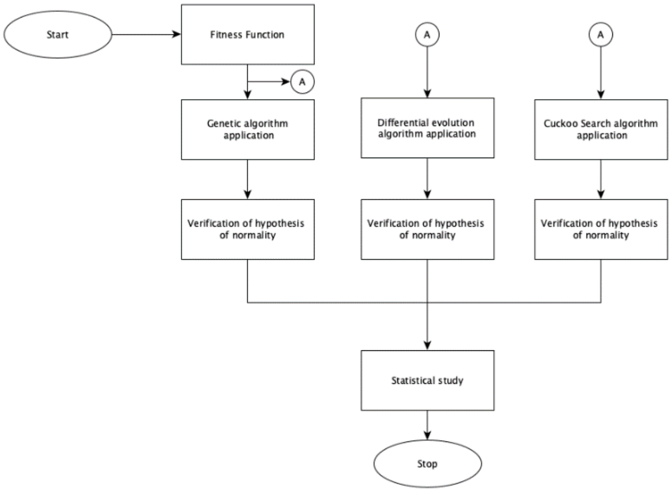
Start
Fitness Function
A
A
A
Genetic algorithm application
Differential evolution algorithm application
Cuckoo Search algorithm application
Verification of hypothesis of normality
Verification of hypothesis of normality
Verification of hypothesis of normality
Statistical study
Stop

Fig. 6. Methodological scheme

4. RESULTS

The table 3 shows the implementation of the Shapiro test, which indicates whether the test data follow a normal distribution

Table 2 Anova test results

Test	Shapiro test (p-value)
Test 1	0.004165
Test 2	0.002536
Test 3	0.1239
Test 4	1.799582e-26
Test 5	0.05322077
Test 6	0.2088

From Table 3 it is possible to affirm that Test 6, Test 5, Test 3 have a normal distribution that is, the Cuckoo Search, GA Arithmetic and DE/best/2 algorithms, have that behavior.

Table 3 Statistics normal distribution experiments

Test	Mean	Standar Desviation	variation coefficient
Cuckoo Search	22.71	0.1108472	0.00488
GA Aritmethic	23.31	0.3289706	0.01411
DE/best/2	22.11	0.7773264	0.03515

It is observed that the lowest coefficient of variation is found in the Cuckoo Search algorithm, but with a lower average in the DE/best/2 algorithm, with this in mind the Anova test is calculated for these two experiments in order to determine if they are statistically different. The p-value of this statistic is 5.65e-06, therefore being less than 0.05 means that they are two different algorithms. From Table 3 it is possible to affirm that Tests 1, Test 2, Test 4 have a non-normal distribution, that is, the algorithms DE/best/1, DE/current to best and GA BLX-∝, have that behavior. comportamiento

Table 4 Statistics experiments of non-normal distribution

	Median	3rd Quartile	1st Quartile
DE/best/1	22.36	22.76	21.93
DE/current to best	22.63	22.72	22.26
GA BLX-∝,	17.20	17.20	17.20

Given the absence of normality of the data reported to verify if there are differences between these algorithms, the Barlet test is used with a p-value = 2.2e-16, therefore being less than 0.05 means that they are different algorithms. The behavior of these aalgorithms with respect to their distribution is shown in Figure 7

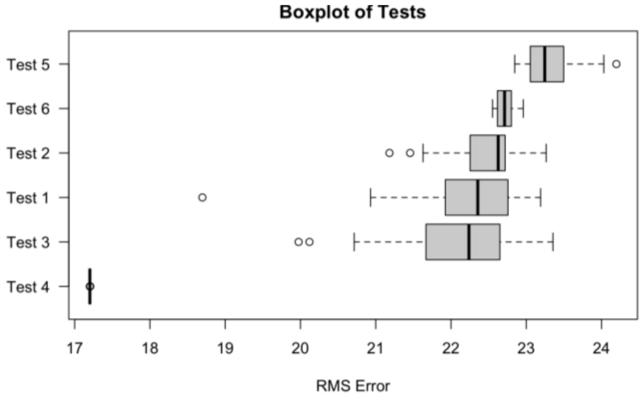

Fig. 7. Boxplot of Test

According to Figure 7 and graphics tables 3 and 4 it can be inferred that the best adjustment algorithm is the Genetic with BLX-∝ cross, according to the median of the executions, but generates a response with several oscillations. Alternatively, seeking to have a response with fewer oscillations, a comparison is made in closed-loop response with the controllers tuned by Cuckoo Search, and DE/best/2. This is shown in Figure 8

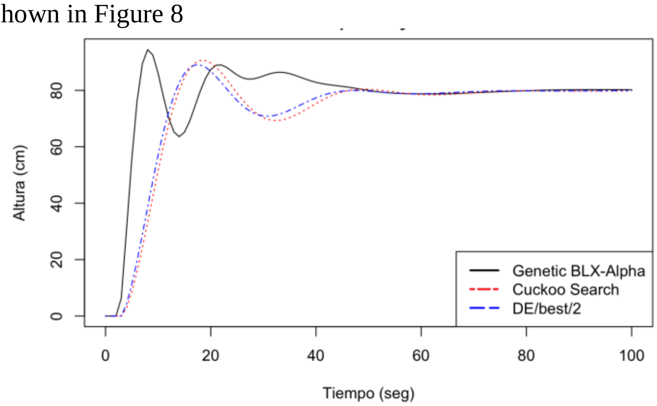

Fig. 8. Response of the PID-Levitator systems

It should be noted that the BLX-∝ algorithm is compared with DE/best/2, since it has the lowest rms error value, after the BLX-∝ algorithm. The reason for choosing Cuckoo Search is that this is the most stable of the studied algorithms, this statement is supported in Figure 7 and Table 4.

5. CONCLUSIONS

The gain tuning of a controller is one of the areas of opportunity of evolutionary algorithms, since they can perform deeper explorations in search spaces than analytical solutions such as the Zigler-Nichols method. It is important to study the statistical behavior of the algorithms since this is an indicator of their stability and the quality of solutions obtained.

New heuristics such as the Cuckoo Search are usually proposed for specific problems, but it has not been tested in applications such as the height control of a pneumatic levitator, therefore it is feasible to propose as future work to carry out a design of experiments that allows to verify if when varying parameters of Levy's flight there are variations in the stability of the algorithm or its convergence, that is, the speed at which it finds a solution

6. REFERENCIAS

[1] T. Tkáˇcik, M. Tkáˇcik, S. Jadlovská y A. Jadlovská, «Design of Aerodynamic Ball Levita-tion Laboratory Plant,» Laboratory Plant. Processes, vol. IX, nº 11, p. 1950, 2021

[2] A. Peijnenburg, J. Vermeulen y J. Van Eijk, «Magnetic levitation systems,» Microelectron-ic Engineering 83, vol. 83, nº 4-9, p. 1372–1375, 2006.

[3] J. M. Escano, M. G. Ortega y F. R. Rubio, "Position Control of a Pneumatic Levitation System", en 2005 IEEE Conference on Emerging Technologies and Factory Automation, Catania, Italy. IEEE.

[4] O. A. Wilson, P. N. Wellington y P. A. Carlos, "PID and LQI control algorithms using WNCS applied to a levitator", en 2021 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Valparaíso, Chile, 6–9 de December de 2021. IEEE, 2021.

[5] J. Chacón, H. Vargas, S. Dormido y J. Sánchez, "Experimental Study of Nonlinear PID Controllers in an Air Levitation System", IFAC-PapersOnLine, vol. 51, n.º 4, pp. 304–309, 2018.

[6] J. F. García-Mejía, L. Colin-Rivas, C. E. Torres-Reyes y J. A. Pérez-Martínez, "Control de un Levitado Neumático con base en un PID y microalgoritmos genético", CIM, vol. 4, n.º 4, pp. 456–462, 2016.

[7] Rivas, L. C., García Mejía, J. F., & Flores Fuentes, A. A. (2021). Diseño e implementación de un regulador pid del flujo de un levitador neumático. Tecnología, Diseño E Innovación, 6(1), 39–50

[8] Xin-She Y.: Nature-Inspired Optimization Algorithms. Second Edition Academic Press. United States (2021)

[9] [11] Joshi, A., Kulkarni, O., Kakandikar, G., & Nandedkar, V. (2017). Cuckoo Search Optimization- A Review. Materials Today: Proceedings, 4(8), 7262–7269. h