Distribution Function Based- Arithmetic Optimization Algorithm for Global optimization and Engineering Applications

Abdullah Ates (abdullah.ates@inonu.edu.tr)
Inonu Universitesi Muhendislik Fakultesi https://orcid.org/0000-0002-4236-6794
Laith Abualigah
Amman Arab University
Mohamed Abd Elaziz
Zagazig University

Research Article

Keywords: Distribution function, Arithmetic Optimization, 3 DOF Hover, global Optimization.

Posted Date: May 27th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-822763/v1

Distribution Function Based- Arithmetic Optimization Algorithm for Global optimization and Engineering Applications

Abdullah Ates[*1], Laith Abualigah[2,3], Mohamed Abd Elaziz[4]

[1]Inonu University, Computer Engineering Department, Malatya, Turkey

[2]Amman Arab University, Software Engineering Department, Amman, Jordan

[3]School of Computer Sciences, University Sains Malaysia, Pulau Pinang, 11800, Malaysia.

[4]Zagazig University, Department of Mathematics, Zagazig, Egypt

abdullah.ates@inonu.edu.tr, aligah.2020@gmail.com, abd_el_aziz_m@yahoo.com

Abstract

In this study, Modified Arithmetic Optimization Algorithm (MAO) is proposed by updating the basic arithmetic optimization (AO) algorithm with different random distribution functions. The AO algorithm is a stochastic swarm-based algorithm that uses main mathematics operators (multiplication, division, subtraction, and addition) during the updating process. In the basic AO algorithm, random coefficients derived according to uniform distribution are used, especially in the generation of the initial population, exploration, and exploitation phases. In this study, these random coefficients are updated with Chi-Square, gamma, logistic, half normal, exponential, normal, extreme value, inverse Gaussian distribution functions. The efficacy of the developed MAO is evaluated using a set of experimental series including global benchmark optimization and real engineering applications named 3 DOF Hover flight system. For global optimization, the proposed MAO algorithm was run according to 100, 500, and 1000 dimensions for 23 different benchmark functions, and the results are compared with each other. As can be seen from the results, the proposed method produced better results than the classical AO results and well-known metaheuristic techniques. It is seen that MAO performs much better, especially in cases where the number of dimensions' increases. In addition, 3 DOF Hover Experiment sets, which is an important problem in flight control systems, were used for the engineering application of the proposed method. Linear Quadratic Regulator (LQR) control structure is used to control this experiment set. In the LQR control structure, the Q and R matrices must be optimal. A total of 10 parameters were optimized, and the results were compared with Darwinian particle swarm optimization, fractional-order Darwinian particle swarm optimization, and classical AO algorithms. For comparison, first of all, optimization has been made on the simulation model of the system. As a result of this optimization, it was determined that the results of the MAO algorithm optimized according to the half-normal and exponential distribution functions have better control performance. Then, the optimization parameters obtained for the simulation model were tested in real-time 3 DOF Hover systems and it was shown that the results found work in real-time 3 DOF Hover systems.

Keyword: Distribution function; Arithmetic Optimization; 3 DOF Hover; global Optimization.

1. Introduction

Recently, optimization problems have more attractive since they have been applied to different real-world applications. For example, Fuel cell (Fathy et al. 2020), PV solar cell (Oliva et al. 2017; Abd Elaziz and Oliva 2018; Fathy et al. 2019), medical image processing (Chakraborty and Mali 2021)(Kandhway et al. 2020), cloud computing (Abd Elaziz and Attiya 2020), machine learning (Aljarah et al. 2020; Ouadfel and Abd Elaziz 2020; Abd Elaziz and Yousri 2021), forecasting problems (Essa et al. 2020; Al-qaness et al. 2021), engineering problems (Ewees and Elaziz 2020; Mehrabi et al. 2021; Abd Elaziz et al. 2021a), and others. There are different optimization techniques that have been presented to tackling optimization problems. These methods can be classified into main categories, including traditional and Metaheuristic techniques. These traditional methods depend on gradient information collected from search space. This kind of optimization method including, gradient descent and the Newton method. Those methods are simple and easy to be implemented; However, they suffer from several drawbacks, such as easily stuck in local points, and they have only one single solution at each iteration. This leads to degradation of the performance of the final solution.

Moreover, the Metaheuristic (MH) techniques avoid these limitations since they depend on the population of solutions which leads to improving the convergence rate. These MH techniques simulate the behavior of animal, bird, fish, physical rules, and chemical components. According to these behaviors, MH can be divided into four groups; the first group is called evolutionary algorithms, which emulate the evolution theory that depends on selection, crossover, and mutation operations. Genetic algorithm (Holland 1992), differential evolution (Storn and Price 1997), genetic programming (Turky et al.). The second group is named natural-based technique, which simulates the natural phenomena in nature and they are including Simulated Annealing and, Gravitational Search Algorithm (GSA), Chemical Reaction Optimization, a modified intelligent water drops (IWD) (Alijla et al. 2014), water cycle algorithm (WCA) (Sadollah et al. 2015), spiral optimization (SO) (Tamura and Yasuda 2011), symbiotic organisms search (SOS) (Cheng and Prayogo 2014). The third group aims to emulate the behavior of humans, so named human-inspirit MH technique. This group include teaching learning-based optimization (TLBO) (Khanduzi et al. 2018), seeker optimization algorithm (SOA) (Dai et al. 2007), soccer league competition (SLC) (Moosavian and Kasaee Roodsari 2014), volleyball premier league algorithm (VPL) (Abd Elaziz et al. 2021b), and cultural algorithm (CA) (Coello Coello and Becerra 2004).

The fourth group emulates the behavior of swarm in nature such as bird, fish, animal and other, so it is called swarm-based optimization. This group is considered one of the most popular MH techniques, and it includes Particle Swarm Optimization ([CSL STYLE ERROR: reference with no printed form.]), Ant Colony (Dorigo and Di Caro), Sailfish Optimization (SO) (Shadravan et al. 2019), Marine Predators Algorithm (MPA) (Faramarzi et al. 2020), Aquila Optimizer (AO) (Abualigah et al. 2021b) and others.

Most of these MH techniques established their performance in several problems since they consist of two explorations and exploitation, which leads to an increase in the convergence rate towards the optimal solution. However, the main critical point in these swarm optimization methods is the factor of balancing between these two phases. This led to developing several improvements and applied them to real-world problems (Ates et al. 2017; Ates 2021; Ates and Akpamukcu 2021).

In the same context, a new MH technique named Arithmetic Optimization Algorithm (AOA) has been developed in (Abualigah et al. 2021a). This technique uses the basic arithmetic operators (i.e., addition, subtraction, division, and multiplication) to perform the exploration and exploitation stages similar to MH techniques. AOA has been established its performance

to solve different optimization problems such as CEC2020, and engineering problems (Abualigah et al. 2021a). However, AOA still require improvement to enhance the balancing between exploration and exploitation. Besides, the No Free Lunch theorem (Adam et al. 2019) which assumed that there is no one algorithm that can solve all optimization problems with the same quality (i.e., absence of universality of MH techniques).
It is known that stochastic processes are derived according to a uniform distribution in stochastic-based optimization algorithms. However, the performance of optimization algorithms can be increased by using different distribution functions instead of uniform distribution. For example, Akdag et al. (Akdag et al. 2020) presented harris hawk optimization algorithm has been modified with different distribution functions and applied in optimum power flow problem. Akpamukcu and Ates (AKPAMUKÇU and ATEŞ 2020) updated the stochastic multi-parameter divergence optimization (SMDO) algorithm with different distribution functions, increasing the performance of the existing algorithm, and the results are shown on the benchmark functions. In (Ates et al. 2020) fractional-order controllers are designed for fractional-order systems with modified SMDO algorithm with distribution functions is presented. Novel algorithms are constantly being proposed today. Then, hybrid algorithms are proposed by combining the superior sides of existing algorithms. For example, the MAPO algorithm is proposed by combining the Base optimization algorithm with the Artificial physic optimization algorithm (Ateş and Yeroğlu 2018). A multi-objective hybrid particle swarm optimization algorithm is proposed by combining particle swarm with salp swarm (El Sehiemy et al. 2020). But algorithms may lose their characteristics or basic philosophies when hybridizing. The main purpose of using different distribution functions is to increase the performance of the existing algorithm without losing its philosophy. The newly proposed arithmetic optimization algorithm is a population-based optimization algorithm that uses basic mathematical operators such as addition, subtraction, multiplication, division operators, and uniform random variables in the initial population generation, exploration, and exploitation phases (Abualigah et al. 2021a). This has been motivated us to introduce an enhancement AOA algorithm using five distributions, including (gamma, chi-square, logistic, half normal, exponential, normal, extreme value, inverse Gaussian). In this study, without changing the basic philosophy of the AO algorithm, a modified arithmetic optimization algorithm is proposed by using different distribution functions in generating the initial population, exploitation, and exploration processes of basic AO. The main objective of using these distributions is to improve the balancing between the exploration and exploitation and this achieved by update the Math Optimizer Accelerated (MOA) and Math Optimizer probability (MOP) parameters in traditional AOA based on these distributions.

The main contribution of this study can be summarized as:

- Proposed an alternative method to solve global optimization and engineering problems.
- Improve the performance of the Arithmetic Optimization Algorithm using distribution functions.
- Evaluate the ability of developed method using classical benchmark functions and Q and R weight matrix optimization for control of 3 DOF Hover flight control system.
- Compare the results of the developed method with other well-known MH techniques.

2. Background

2.1 Basic Arithmetic Optimization Algorithm

Usually, population-based algorithms begin their improvement stages with a random number of candidate solutions [1]. A series of iterations and rules incrementally enhance the created solutions. It is then evaluated iteratively by a fixed objective function, which is the core of the optimization technique. Since population-based algorithms aim to find the best solution to optimization problems in a stochastically efficient way, finding a solution in a single run is not guaranteed. An adequate collection of potential solutions iteratively for the given problem, on the other hand, increases the possibility of finding the best global solution.

The conventional Arithmetic Optimization Algorithm (AOA) [2], which is stimulated by the Arithmetic operators in math science (i.e., Multiplication (M), Division (D), Subtraction (S), and Addition (A), is discussed and exploited in this portion. AOA is a population-based, gradient-free search approach for solving any optimization problem with a given objective function. The AOA's used quest phases are depicted in Figure 1 and are defined as follows.

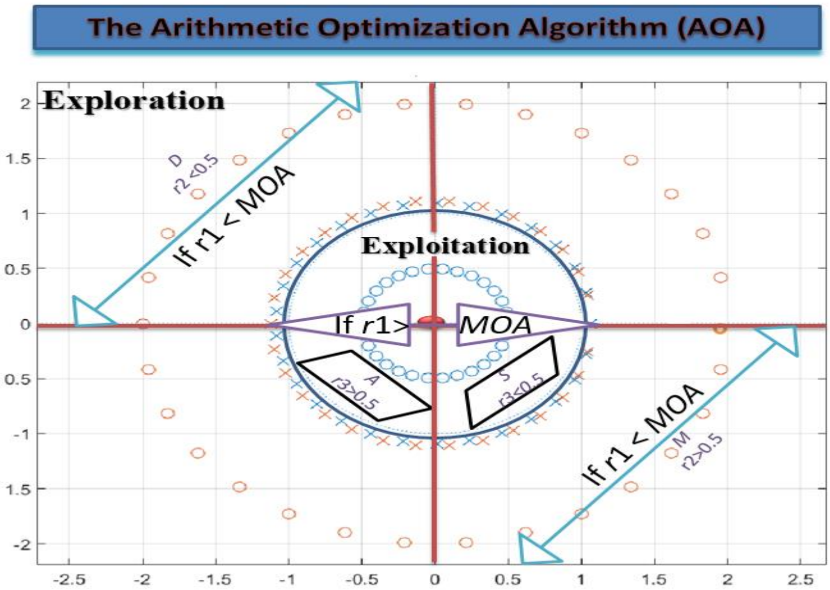

Figure 1. The optimization processes of the AOA.

Arithmetic is a necessary part of the numerical method. Along with geometry, arithmetic, and inquiry, it is an essential aspect of functional mathematics. The standard consideration measures used to evaluate the numbers are arithmetic operators (multiplication, division, subtraction, and addition). These basic operators are used as part of a mathematical optimization technique to find the best solution given a set of constraints. From the boundary to the heart, Figure 2 illustrates the ladder of Arithmetic operators and their rules—the mathematical form of the AOA as seen below.

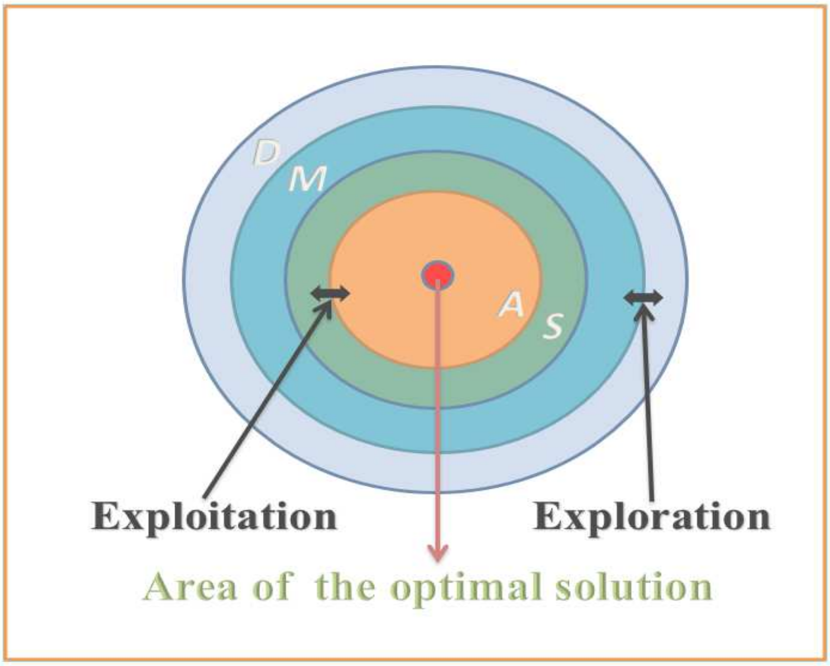

Figure 2. The ladder of Arithmetic operators and their rules.

2.1. Initialization phase

As Matrix (1) stated, the optimization rules in the AOA function on several generated random solutions (X). The best-obtained solution is saved as close to the optimum so far in each iteration.

$$X_i = \begin{bmatrix} x_1^1 & x_1^1 \cdots & x_{1,D}^1 \\ x_1^2 & x_2^2 \; \cdots & x_D^2 \\ \vdots & \ddots \quad \ddots & \vdots \\ x_1^N & x_D^N \; \cdots & x_D^N \end{bmatrix} \tag{1}$$

For each iteration, the search process (exploration or exploitation) should be chosen based on the Math Optimizer Accelerated (MOA) function determined using Equation (2).

$$MOA(C\_Iter) = Min + C\_Iter \times \left(\frac{Max - Min}{M\_Iter}\right) \tag{2}$$

where MOA(C_Iter) is the MOA value determined by Equation (2) for the current iteration, M_Iter is the cumulative number of iterations. C_Iter is the current iteration, which is between [1 M_Iter]. The accelerated function's minimum and maximum values are called Min and Max, respectively.

2.2 Exploration phase

The exploratory operation of AOA is defined in this section. The phase's optimization method used either the Division (D) rule or the Multiplication (M) rule to find high-distributed decisions dependent on the Arithmetic operators. However, unlike other rules, these rules (i.e., D and M) cannot smoothly advance the goal area due to their high distribution (i.e., S and A). The effect of Arithmetic operators on complex mathematical formulas is depicted in Figure 3. As a result, the discovery quest seeks a near-optimal solution that can be sought after multiple iterations.

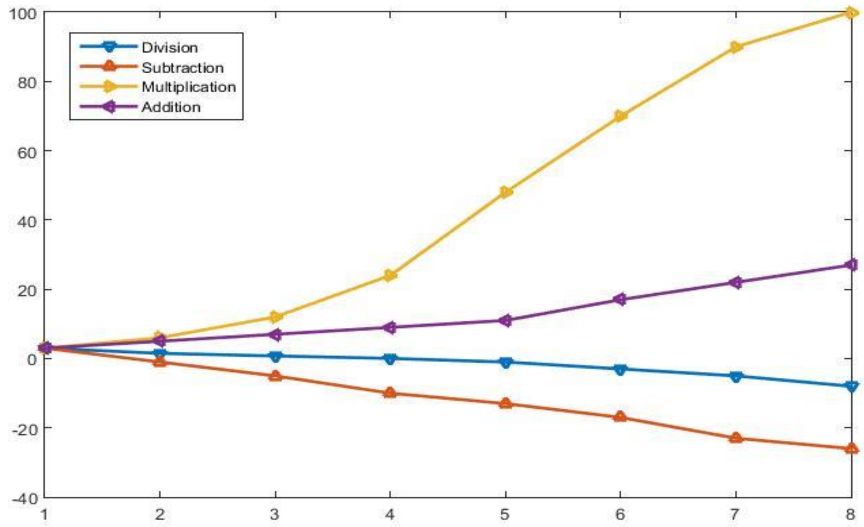

Figure 3. The influence of Arithmetic operators for specific mathematical calculations.

AOA's exploration phase searches the search space in many places at random. It employs two search procedures (D and M), which are modeled in Equation (3), to arrive at a better solution. If r1>MOA, where r1 is a random number and MOA is determined using Equation (4), this step is enabled. The D operator is conditioned by r2<0.5 in this phase; otherwise, the other operator (M) would perform the current operation, with r2 being a random number. To have further diversification, a stochastic scaling coefficient is used.

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) \div (MOP+\epsilon) \times \left((UB_j - LB_j) \times \mu + LB_j\right), r2 < 0.5 \\ best(x_j) \times MOP \times \left((UB_j - LB_j) \times \mu + LB_j\right), otherwise \end{cases} \tag{3}$$

where $x_i(C\_Iter+1)$ is the i_{th} solution positions in the upcoming iteration, $x_{i,j}(C\_Iter)$ is the j_{th} position of the i_{th} solution of the candidate solutions, and $best(x_j)$ is the j_{th} in the best-obtained position in the i_{th} solution. μ is a tuning parameter utilized to determine and control the search process, which is set equal to 0.5 according to the original paper [2].

$$MOP(C\_Iter) = 1 - \frac{C\_Iter^{\left(\frac{1}{\alpha}\right)}}{M\_Iter^{\left(\frac{1}{\alpha}\right)}} \tag{4}$$

where MOP is a coefficient value, $MOP(C\_Iter)$ is the value of the MOP calculated at the t_{th} iteration, and C_Iter is the current iteration of the search process, and M_Iter is the fixed maximum number of iterations. α is a tuning parameter utilized to control and specify the exploitation accuracy throughout the optimization process, which is set equal to 5 according to the original paper [2].

2.3 Exploitation phase

The AOA exploitation protocol is defined in this section. To get high-dense results based on the Arithmetic operators, mathematical formulas using these operators (i.e., S or A) are used.

Compared to other operators, these operators can accurately identify the target region due to their low distribution. As a result, the exploitation process determines the near-optimal solution that can be identified over several iterations. By improved coordination between the search processes, the exploitation operators are used to facilitate the exploitation process.

If r1 ≤ MOA(C Iter), this step is conditioned by the MOA value. The manipulation operators of AOA (i.e., S and A) analyze the quest area intensively on many thick regions and achieve a better solution, which is modeled in Equation 1. (5). This process uses the search space by performing an in-depth search, as seen in Figure 1. If r3 < 0.5, the S operator in Equation (5) will be used; otherwise, the A operator will be used to complete the current mission. This phase's procedures are identical to the previous phase's separations.

However, exploitation operators (i.e., S and A) typically aim to stay away from the local optimum. This technique deals with exploring search methods in order to find the right solution while maintaining the variety of applicant solutions. The parameter is set to have a random value at each iteration, allowing for experimentation of both the first and last iterations. This part of the search helps prevents local optima stagnation, especially in the final iterations.

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) - MOP \times \left((UB_j - LB_j) \times \mu + LB_j\right), r3 < 0.5 \\ best(x_j) + MOP \times \left((UB_j - LB_j) \times mu + LB_j\right), otherwise \end{cases} \quad (5)$$

Figure 4 illustrates the updating process in AOA using math operators and how a search solution renews its locations in a 2-Dimensional search space using math operators. The final-obtained region can be in a stochastic place within an area specified by math operators' positions as the search range, which can be recognized. Figure 4 shows updating process using the math operators in AOA

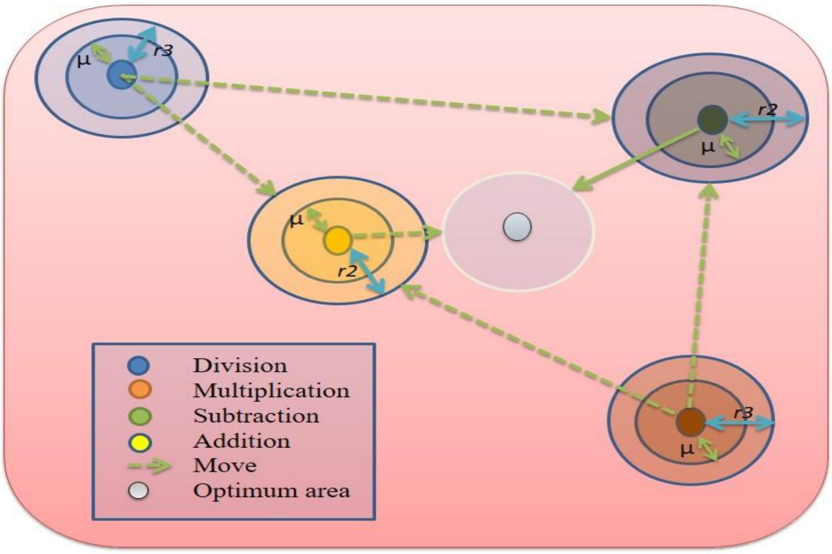

Figure 4: Updating process using the math operators in AOA

3 Distribution Function

The uniform random numbers are always used in stochastic algorithms. It is known that the characteristics of the random number are specified by the used distribution function. Five different distribution function is tried on AO method. These are Chi-Square distribution, gamma distribution, logistic distribution, half-normal distribution, and exponential distribution. Used distribution definitions are given as follows:

3.1 Chi-Square Distribution (CSD)

The chi-square distribution has one parameter for generating curves. The chi-square distribution is usually used in hypothesis testing. (Probability, 2020).

$$\text{y=f(x | v)} = \frac{x^{(v-2)/2}\, e^{-x/2}}{2^{\frac{v}{2}}\Gamma(v/2)} \tag{6}$$

3.2 Gamma Distribution (GD)

The gamma distribution is two-parameter to generate curve. This models sums of the chi-square and exponential distributions (Probability, 2020).

$$\text{y=f(x|a,b)} = \frac{1}{b^a\Gamma(a)}x^{a-1}e^{\frac{-x}{b}} \tag{7}$$

3.3 Logistic Distribution (LD)

The logistic distribution has two parameters for generating values. It uses for logistic regression. It is second statistical moment is higher than the normal distribution (Probability, 2020).

$$f(x\,|\,\mu,\sigma) = \frac{\exp\left\{\frac{x-\mu}{\sigma}\right\}}{\sigma(1+\exp\left\{\frac{x-\mu}{\sigma}\right\})^2} \tag{8}$$

3.4 Half Normal Distribution (HND)

The half-normal has also two parameters. It is special case of the normal and truncated normal distributions. The probability density function (pdf) of the half-normal distribution is (Probability, 2020);

$$\text{y = f(x}|\mu\text{) = } f(x\,|\,\mu,\sigma) = \sqrt{\frac{2}{\pi}}\frac{1}{\sigma}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)};x\geq\mu \tag{9}$$

3.5 Exponential Distribution (ED):

The exponential distribution has a single parameter for generating curve. Probability density function of the exponential distribution (pdf) (Probability, 2020):

$$y = f(x|\mu) = \frac{1}{\mu}e^{\frac{-x}{\mu}} \tag{10}$$

3.6 Extreme Value Distribution (EVD):

The exponential distribution has two parameters for generating curve. Probability density function of the exponential distribution (pdf) (Probability, 2020). The probability density function is always used for modelling minımum value.

$$y = f(x|\mu) = y = f(x|\mu,\sigma) = \sigma^{-1}\exp\left(\frac{x-\mu}{\sigma}\right)\exp\left(-\exp\left(\frac{x-\mu}{\sigma}\right)\right) \tag{11}$$

3.7 Inverse Gaussian Distribution (IGD):

The Inverse Gaussian Distribution has two parameters for generating curve. Density function of the inverse Gaussian distribution is given as follows (Probability, 2020):

$$y = f(x|\mu) = \sqrt{\lambda/2\pi x^3}\exp\left\{-\frac{\lambda}{2\mu^2 x}(x-\mu)^2\right\} \tag{12}$$

4 Modified Arithmetic Optimization Algorithm

The basic arithmetic optimization algorithm is a population-based algorithm worked on the multiplication, division, subtraction, and addition operators. In this algorithm, random coefficients derived from uniform distribution are used for parameter movements in exploitation and exploration processes, especially in the generation of the initial population. These coefficients affect the performance of the algorithm. The modified arithmetic optimization algorithm is obtained by updating stochastic processes with different distribution functions in the basic arithmetic optimization algorithm. First of all, distribution functions to be used in this study were determined according to the benchmark function and their use in the real engineering problem. Chi-square Distribution (CSD), Gama Distribution (GD), Logistic Distribution (LD), Half Normal Distribution (HND) and Exponential Distribution (ED), Normal Distribution (ND), Extreme Value Distribution (EVD), Inverse Gaussian Distribution (IGD) distributions were used in the study according to their better performance in AO method. While using these distributions, a distribution function (DF) vector has been defined for use in this study. This vector is given as below.

$$\text{DF= [CSD GD LD HND ED EVD IGD ND]} \tag{12}$$

Initially, the distribution function that should be used according to the related problem is selected from this vector with selector operator φ, and all equations with stochastic processes in the basic AO algorithm are updated as follows. First, the initial population is derived according to the distribution function selected at the beginning of the algorithm.

$$X_j = \begin{bmatrix} DF_1^1(\varphi,1) & DF_2^1(\varphi,1) & \cdots & \cdots & DF_D^1(\varphi,1) \\ DF_1^2(\varphi,1) & DF_1^2(\varphi,1) & \cdots & \cdots & DF_D^2(\varphi,1) \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ DF_1^N(\varphi,1) & DF_D^N(\varphi,1) & \cdots & \cdots & DF_D^N(\varphi,1) \end{bmatrix} \tag{13}$$

where φ show distribution function selector. This value is selected arbitrarily. In the MAO optimization algorithm, instead of the r2 coefficient in the exploration phase given in Eq. 3, it is modified with the $DF(\varphi,1)$ vector given in Eq. 13. All other rules are the same as for the basic AO algorithm.

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) \div (MOP + \in) \times ((UB_j - LB_j) \times \mu + LB_j), DF(\varphi,1) < 0.5 \\ best(x_j) \times (, MOP \times ((UB_j - LB_j) \times \mu + LB_j), otherwise \end{cases} \tag{14}$$

A similar process has been implemented for the exploitation phase (in Eq. 5). In this process, the coefficient r3 was updated with the DF vector given in Eq. 12, and Eq. 15 was obtained for MAO.

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) - MOP \times ((UB_j - LB_j) \times \mu + LB_j), DF(\varphi,1) < 0.5 \\ best(x_j) + (, MOP \times ((UB_j - LB_j) \times \mu + LB_j), otherwise \end{cases} \tag{15}$$

The purpose of the modified arithmetic optimization algorithm is to increase the performance of the algorithm without changing the basic rules of the AO algorithm. Therefore, the performance of the algorithm is increased by using different distribution functions instead of random coefficients in the basic AO algorithm. In this way, it has been shown that analytical contributions can be made to increase the performance of existing algorithms instead of constantly learning new algorithms to increase the performance of the algorithm. The pseudo-code of the modified arithmetic optimization algorithm can be given as follows. To summarize, the optimization rule in MAO produces a random set of competitor solutions (population). The math operators determine the feasible positions of the near-optimal solution using the trajectory of repetition. From the best-obtained solution, each solution renews its place. The MOA value is increased linearly from 0.2 to 0.9 to sustain discovery and extraction. Competitor solutions try to turn from the near-optimal solution to the near-optimal solution as $DF(\varphi,1)$ >MOA and vice versa. In MAO method all uniform random distribution definition was modified with DF matrix. Finally, until the end requirement is fulfilled, the MAO optimizer is terminated.

Algorithm 1: Modified Arithmetic Optimization Algorithm (MAOA) (Abualigah et al. 2021a)

Initialize the Modified Arithmetic Optimization Algorithm Parameters N, α, μ, Distribution **function parameter** etc.
Initialize the solutions' positions randomly **with selected distribution function definition**. (Solutions: i= 1... N).
WHILE (C_Iter < M_Iter)
  Calculate the Fitness Function for the given solutions.
  Find the best solution so far.
  Update the MOA parameter using Equation (2).
  Update the MOP parameter using Equation (4).
  FOR (i=1 to Solutions)
    FOR (j=1 to Positions)
Generated random values according to used **distribution function definitions with [DF] matrix** for (r1, r2, and r3)
      IF ($DF(\varphi,1)$ **>MOA)**
        IF $DF(\varphi,1)$ **>0.5)** **Exploration phase**
        Update the i_{th} solutions' positions using the first part in Equation (3).
      ELSE
        Update the i_{th} solutions' positions using the second part in Equation (3).
      ENDIF
    ELSE
      IF ($DF(\varphi,1)$ >0.5) **Exploitation phase**
        Update the i_{th} solutions' positions using the first part in Equation (5).
      ELSE
        Update the i_{th} solutions' positions using the second part in Equation (5).
        ENDIF
      ENDIF
    ENDFOR
  ENDFOR
  C_Iter=C_Iter+1
ENDWHILE
Return the best solution (best_x).

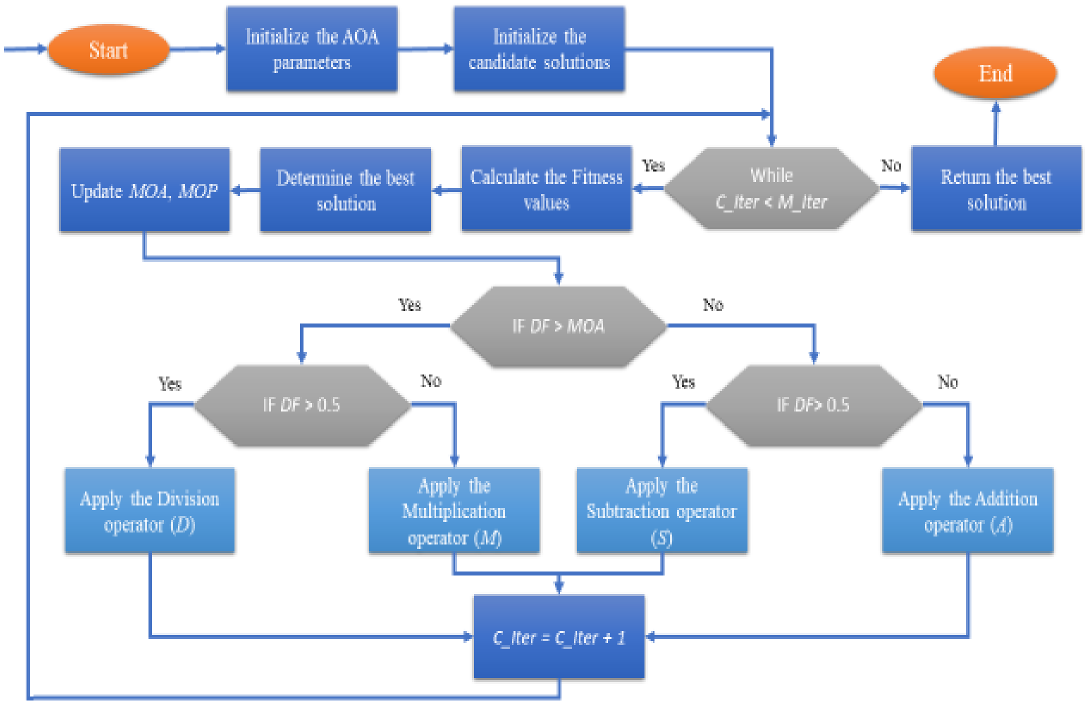

Figure 5: Flowchart of the Modified MAO Algorithm (Abualigah et al. 2021a) .

5. Results and Discussion

5.1 Benchmark Function Comparisons

In this section, the proposed modified Arithmetic Optimization Algorithm (MAOA) is tested using twenty-three (i.e., F1-f23) benchmark functions from CEC2005 to illustrate the ability of the proposed method in solving versions benchmark functions. For the F1-F13, three high dimensional sizes (i.e., 100, 500, and 1000) are used to prove the performance of the proposed MAOA in solving high dimensional problems. Note, the number of used solutions and a maximum number of iterations are 30 and 1000, respectively.

Table 1 shows the results of the comparative methods using thirteen benchmark functions (F1-F13) using various distribution functions, where the dimension size is fixed to 100. The proposed MAOA got almost the best results in all the test functions except F8. This reflects the ability of the proposed method to solve problems of large dimensions more efficiently than the traditional method. Also, the employment of distribution functions has an apparent effect on the results of the proposed method for clearly finding optimal solutions. The MAOA got the best results for the F1 using Chi-square distribution where value is 0.1 and value1 is 1.4. As well, for the functions F2, F4, F7, F9, F10, and F11, the proposed MAOA got the best results using Chi-square distribution where value is 0.1 and value1 is 1.4. The MAOA got the best results for the F3 using Normal distribution where value is 0.05 and value1 is 0.05. For the functions F5, F6, and F13, the proposed MAOA got the best results using Gamma distribution where value is 0.005 and value1 is 0.005. The MAOA got the best results for the F12 using Extrem distribution where value is 0.1 and value1 is 1.4.

Table 2 shows the results of the comparative methods using ten fixed dimension benchmark functions (F13-F23). The proposed MAOA got the best results in all the test functions (F14-F23). The MAOA got the best results for the F1 using Half normal distribution where value is 0.1 and value1 is 1.4. As well, for the functions F15, F16, and F17, the proposed MAOA got the best results using Normal distribution where value is 3 and value1 is 3. The MAOA got the best results for the F18 and F19 using Inverse Gaussion distribution where value is 0.08 and value1 is 0.08. For the functions F20, F21, F22, and F23, the proposed MAOA got the best results using Extrem Value distribution where value is 3, value1 is 2, and value2 is 2. This represents the proposed method's potential to solve large-scale problems more quickly than the conventional method. Furthermore, the use of distribution functions seems to have an effect on the effects of the proposed approach for identifying optimal solutions. In these tables "value", "value1" and "value2" shows that coefficients of the corresponding distribution functions. These values were adjusted for each benchmark function in order to improve the performance of the distribution.

Table 1: Comparisons of Benchmark Function (F1 to F13) with 100 Dimensional according to Modified AOA (MAOA)				
Benchmark Function		Distribution Type	AOA Distribution Based Results (MAOA)	Basic AOA Result
F1 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	2.41E-06
	Std		0	6.31E-06
F2 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	3.58E-08
	Std		0	5.25E-08
F3 Min=0	Ave	Normal Value=0.05 Value1=0.05	0	1.99E-04
	Std		0	2.46E-02
F4 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	3.40E-03
	Std		0	3.23E-03
F5 Min=0	Ave	Gamma Value=0.005 Value1=0.005	9.79e+01	9.79E+01
	Std		2.57e-05	6.55E-01
F6 Min=0	Ave	Gamma Value=0.005 Value1=0.005	2.10+01	2.38E+01
	Std		3.88e-04	2.64E+01
F7 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	5.08e-05	2.03E-04
	Std		4.77e-05	3.24E-03
F8 Min=-418982	Ave	Chisquare Value=0.1 Value1=1.4	-1.897e+04	-1.21E+04
	Std		2.18e+03	-2.45E+04
F9 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	8.46E-06
	Std		0	9.24E-06
F10 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	8.88e-16	2.02E-04
	Std		4.14E-05	4.14E-03
F11 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	1.22E+00
	Std		0	1.22E+00
F12 Min=0	Ave	Extrem Value Value=0.1 Value1=1.4	1.88e-01	2.40E-02
	Std		2.42e-02	3.84E-01
F13 Min=0	Ave	Gamma Value=0.005 Value1=0.005	4.40	4.42E+00
	Std		4.77e-02	9.54E+00

Table 2: Comparisons of Benchmark Function (F14 to F23) according to Modified AOA (MAOA)						
Benchmark Function		Distribution Type	AOA Distribution Based Results (MAOA)	Basic AOA Result	Best Results and Method for Benchmark Function in Basic AOA Study	Benchmark Function
F14 Min=1	Ave	Half_normal Value=0.1 Value1=1.4	0.4670	9.98E-01	9.98E-01	with AOA
	Std		7.3757e-05	5.54E-01	5.54E-01	
F15 Min=0.0003	Ave	Normal Value=3 Value1=3	0.0003152	3.12E-04	3.12E-04	with AOA
	Std		2.370e-02	2.64E-04	2.64E-04	
F16 Min=-1.0316	Ave	Normal Value= 3 Value1= 3	-1.03E+00	-1.03E+00	-1.03E+00	with AOA
	Std		2.3e-10	5.48E-05	5.48E-05	
F17 Min=0.398	Ave	Normal Value=3 Value1=3	3.98E-01	3.98E-01	3.98E-01	with AOA
	Std		1.51e-05	2.54E-06	2.54E-06	
F18 Min=3	Ave	Inverse Gaussion Value=0.8 Value1=0.8	3.00E+00	3.00E+00	3.00E+00	with AOA
	Std		2.03e+01	1.00E-02	1.00E-02	
F19 Min=-3.86	Ave	Inverse Gaussion Value=0.8 Value1=0.8	-3.862E+00	-3.862E+00	-3.862E+00	with AOA
	Std		2.25e-03	4.29E-04	4.29E-04	
F20 Min=-3.32	Ave	Extrem Value Value=3 Value1=2 Value2=2	-3.322	-3.32E+00	-3.32E+00	with AOA
	Std		5.96E-02	1.25E+01	1.25E+01	
F21 Min=-10.1532	Ave	Extrem Value Value=3 Value1=2 Value2=2	-10.1517	-8.85E+00	-8.85E+00	with AOA
	Std		2.53	1.25E+01	1.25E+01	
F22 Min=-10.4028	Ave	Extrem Value Value=3 Value1=2 Value2=2	-10.4022	-1.04E+01	-1.04E+01	with AOA
	Std		2.85	2.21E+00	2.21E+00	
F23 Min=-10.5363	Ave	Extrem Value Value=3 Value1=2 Value2=2	-10.5324	-1.05E+01	-1.05E+01	with AOA
	Std		3.66e+00	1.02E+00	1.02E+00	

Table 3 shows the results of the comparative methods using thirteen benchmark functions (F1-F13) using various distribution functions, where the dimension size is fixed to 500. We tested the proposed method with higher dimensional size (i.e., 500) to evaluate the scalability of the proposed method. The proposed MAOA got almost the best results in all the test functions. The MAOA got the best results for the F1 and F3 using Normal distribution where value is 0.1 and value1 is 1.4. Also, for the functions F2, F4, F7, F9, F10, F11, and F12, the proposed MAOA got the best results using Chi-Square distribution where value is 0.1 and value1 is 1.4. For the functions F5, F6, and F13, the proposed MAOA got the best results using Gamma distribution where value is 0.005 and value1 is 0.005. The MAOA got the best results for the F8 using Logistic distribution where value is 0.5 and value1 is 1.4.

Table 4 shows the results of the comparative methods using thirteen benchmark functions (F1-F13) using various distribution functions, where the dimension size is fixed to 1000. We tested further the proposed method with higher dimensional size (i.e., 1000) to evaluate the scalability of again for harder dimension. The proposed MAOA got almost the best results in all the test functions. For the functions F1, F2, F4, F7, F8, F10, F11, F12, and F13, the proposed MAOA got the best results using Chi-Square distribution where value is 0.1 and value1 is 1.4. Also, the MAOA got the best results for the F3 using Normal distribution where value is 0.1 and value1 is 1.4. For the functions F5 and F5, the proposed MAOA got the best results using Gamma distribution where value is 0.005 and value1 is 0.005. The MAOA got the best results for the F9 using Exponential distribution where value is 0.7 and value1 is 0.7. Thus, the proposed MOAO method has proven its distinct ability and strength in solving various problems. It is also capable of solving all problems of large dimensional size with high efficiency.

Table 3: Comparisons of Benchmark Function (F1 to F13) with 500 Dimensional according to Modified AOA (MAOA)				
Benchmark Function		Distribution Type	AOA Distribution Based Results (MAOA)	Basic AOA Result
F1 Min=0	Ave	Normal Value=0.1 Value1=1.4	3.20e-09	2.70E-02
	Std		1.22e-08	3.14E-02
F2 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	1.09E-02
	Std		0	1.54E-03
F3 Min=0	Ave	Normal Value=0.05 Value=0.05	0	3.57E-01
	Std		0	3.01E-02
F4 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	2.40e-05	4.00E-02
	Std		3.73e-05	4.42E-02
F5 Min=0	Ave	Gamma Value=0.005 Value1=0.005	4.96e+02	4.97E+02
	Std		1.27e-03	5.32E-01
F6 Min=0	Ave	Gamma Value=0.005 Value1=0.005	7.44 E+01	8.22E+01
	Std		3.013 E+00	2.21E+01
F7 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	6.69e-07	2.98E-05
	Std		6.88e-05	2.95E-05
F8 Min=-418*500	Ave	Logistic Value=0.1 Value1=1.4	-7.46e+04	-1.73E+04
	Std		2.92e+03	1.24E+04
F9 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	1.58E-02
	Std		0	1.58E-02
F10 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	8.88e-16	1.93E-03
	Std		2.23e-12	3.50E-04
F11 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	1.86e-09	1.50E-02
	Std		2.97e-09	3.20E-02
F12 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	1.88e-01	2.09E-01
	Std		2.42e-02	2.14E-01
F13 Min=0	Ave	Gamma Value=0.005 Value1=0.005	4.95e+01	4.98E+01
	Std		5.36e-04	6.25E+01

Table 4: Comparisons of Benchmark Function (F1 to F13) with 1000 Dimensional according to Modified AOA (MAOA)						
Benchmark Function		Distribution Type	AOA Distribution Based Results (MAOA)	Basic AOA Result	Best Results and Method for Benchmark Function in Basic AOA Study	
F1 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	0	1.55E+00	1.55E+00	with AOA
	Std		0	2.06E-01	2.06E-01	
F2 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	1.0304e-04	1.60E+00	1.60E+00	with AOA
	Std		1.0621e-04	1.02E-01	1.02E-01	
F3 Min=0	Ave	Normal Value=0.05 Value=0.05	6.3103e-04	3.80E+01	3.70E+01	with FA
	Std		3.9116e-04	1.43+00	1.49E+00	
F4 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	9.3460e-05	1.40E+00	1.40E+00	with AOA
	Std		5.2003e-05	1.12E-01	1.12E-01	
F5 Min=0	Ave	Gamma Value=0.005 Value1=0.005	1.6923E+00	1.70E+00	1.65E+00	with BAT
	Std		4.0813e-06	1.12E-01	1.16E-01	
F6 Min=0	Ave	Gamma Value=0.005 Value1=0.005	1,2544 E+00	1.44E+00 (241.8468)	1.44E+00	with AOA
	Std		6.7692e-04	1.02E-01 (0.1507)	1.02E-01	
F7 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	6.6311e-05	3.99E+00	3.99E+00	with AOA
	Std			8.16E-02	8.16E-02	
F8 Min=-418982	Ave	Chisquare Value=0.1 Value1=1.4	-92756	2.56E+00	2.45E+00	with BAT
	Std		1.763e-12	3.13E-01	2.88E-01	
F9 Min=0	Ave	Exponential Value=0.7 Value1=0.7	4.3430e-04	2.30E+00	2.30E+00	with AOA
	Std		6.6170e-05	2.71E-01	2.71E-01	
F10 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	8.8818e-16	2.42E+00	2.42E+00	with AOA
	Std		5.0493e-18	4.35E-01	4.35E-01	
F11 Min=0	Ave	Chisquare Value=0.1 Value1=1.4	1.8174e-08	2.66E+00	2.61E+00	with BAT
	Std		2.0443e-08	4.05E-01	3.95E-01	
F12 Min=0 okul	Ave	Chisquare Value=0.1 Value1=1.4	0.4655	8.56E+00	8.56E+00	with AOA
	Std		0.0166	1.36E+00	1.36E+00	
F13 Min=0 server	Ave	Chisquare Value=0.1 Value1=1.4	8.6665	8.71E+00	8.47E+00	with BAT
	Std		0.4085	1.33E+00	1.38E+00	

4.2 Controller Design for 3 DOF Quad Copter System

Control design problems with optimization algorithms are still up to date research area for the researcher. Because if the controller designed system is a real-time system, it is necessary to adjust the control parameters of the related algorithm quickly and effectively due to many environmental distributions. There are many studies in the literature in this area. For example, a study in which a fractional and integer degree controller was designed with the enhanced Equilibrium optimization algorithm is presented in (Ates 2021). With optimization to optimization approach, monarchy butter fly and stochastic multi parameter divergence optimization algorithm are presented for the controller design study for load frequency control problem in (Ates and Akpamukcu 2021). Fractional and integer order controllers are designed for twin rotor MIMO system with stochastic multi parameter divergence algorithm in (Alagoz et al. 2013; Yeroğlu and Ateş 2014). Fractional order and integer order controllers are designed for integer and fractional order systems with the modified artificial physics optimization algorithm in (Ateş and Yeroğlu 2018). Fractional and integer order controllers are designed with Tabu search based optimization algorithm in (Ateş and Yeroglu 2016). As can be seen from the examples, controller design with stochastic-based optimization algorithms is still a current research area in the literature.

In this study, the optimization of the Q and R weight matrices in the LQR controller structure used in the control of 3 DOF hover systems, which is a 4-motor helicopter prototype, is carried out. In this way, a better control performance was obtained by finding suitable Q and R matrices.

4.2.1 3 DOF Hover System

In this study 3 DOF Hover system experiment set was used. This set was manufactured by Quanser. The system has four propellers and four DC motors. The set is depicted in Fig. 6. It works by rotating the pitch, yaw and roll angles. This system mimics to quad copter helicopter system. And it is compatible with MATLAB. ([CSL STYLE ERROR: reference with no printed form.]).

Figure 6. 3 DOF Hover Experimental Setup (ATEŞ and AKPAMUKÇU 2021)

4.2.2 Optimization of Q and R Vectors with MAOA Method

Quadrotor helicopter systems are frequently used in many areas such as military, search, security and imaging due to some advantages (Rahman et al. 2018). According to the purpose of use of these vehicles, the desired number of propeller models can be selected. However, this ease of use brings the difficulty of control mechanisms. First of all, the models of these systems have nonlinear complex components. Vehicles are exposed to aerodynamic disturbances as they are in direct contact with the environment. This adds complex, nonlinear components to the system model that change over time. The system has many nonlinear components in total and some of them change over time. Therefore, controller structures that can control multi-input multi-output systems are used in the control of these systems. In addition, since the system is in the air, it is exposed to all kinds of environmental effects and the control of these vehicles appears as a difficult control problem. In the literature, analytical methods such as Linear Quadratic Regulator (LQR) (Prach et al. 2016), Linear Quadratic Gaussian (LQG) (Prach et al. 2016), $H\infty$ control (Zhang et al. 2014), Amplified Linear Quadratic Regulator ALQR (Pereira and Kienitz 2015) are used for the control of these tools.

LQR control structure is widely used in the control of such systems. In the LQR control structure, a K gain matrix is derived that minimizes the 2nd order Riccati equation by using the Q and R weight matrices given at the beginning and utilizing the state space model of the system. The system can be controlled using this K gain matrix. Hence, optimizing the appropriate Q and R weight matrices determining the K gain matrix is important for control performance. In this study, the Q and R weight matrices in LQR structure used for the control of 3 DOF Hover systems were determined by MAO algorithm and the results were compared with classical AO and DPSO and KDDPSO (Içen et al. 2017) methods. The K gain matrix is calculated by minimizing the objective function in Eq. 16, which is created with the 2nd order Riccati equation by using Q and R weight matrices.

$$J = \frac{1}{2}\int_0^{\infty}\left[x^T Qx + u^T Ru\right]dt \tag{15}$$

The calculation of the K gain matrix is defined as the optimal control problem. This gain matrix, which is obtained later, is used in simulation and real-time tests as described in other sections. In (Içen et al. 2017), determination of Q and R gain matrices used in determination of K gain matrix used in the control of 3 DOF hover systems was made with Darwinian and fractional order Darwinian particle swarm optimization algorithms. In this study, first of all, the optimization of the Q and R matrices was done with the classical AO algorithm. During the optimization process, multi-objective function structure given below was used. In addition to these, the optimization structure given in Figure 7 was used during the optimization.

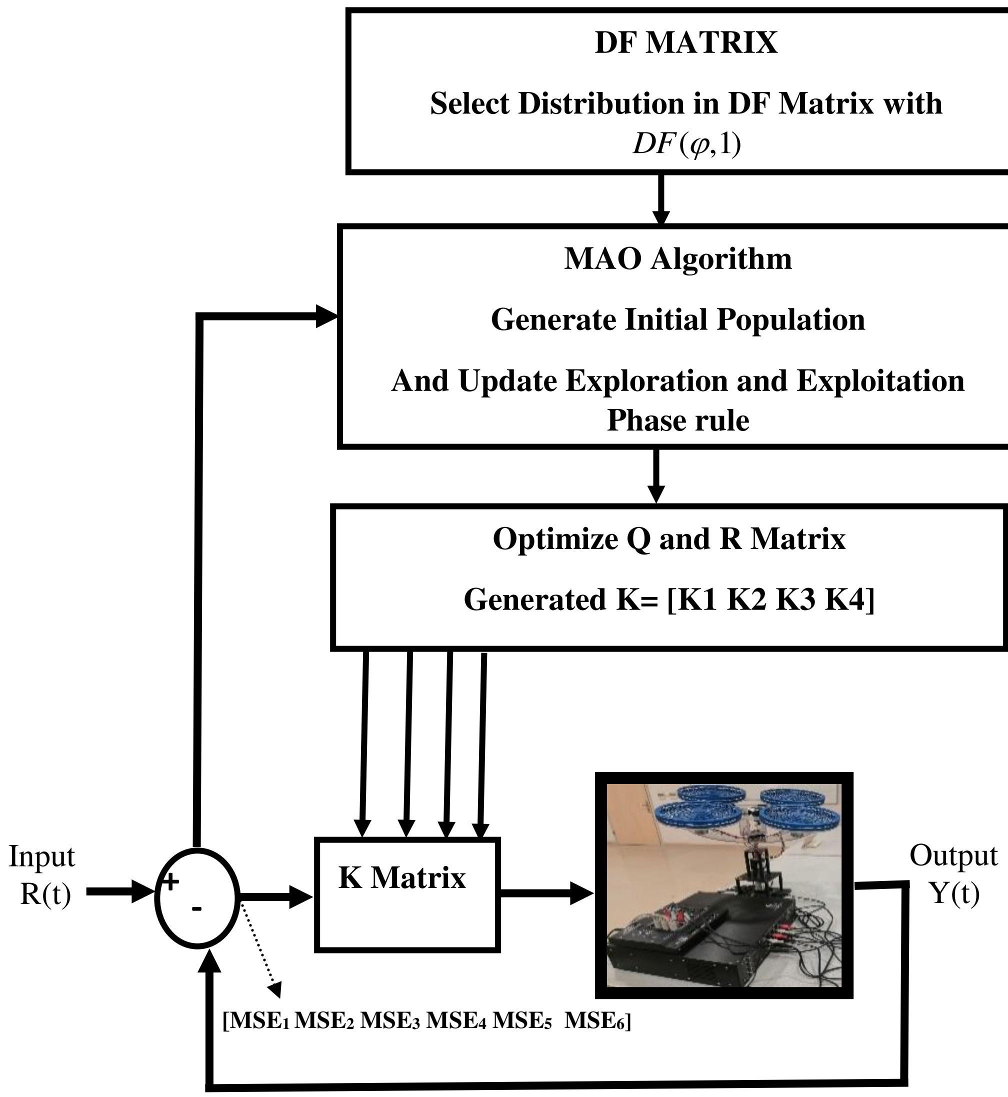

Figure 7: Optimization structure of the MAO

$$OF = w_1MSE_1+w_2MSE_2+w_3MSE_3+w_4MSE_4+w_5MSE_5+w_6MSE_6 \tag{16}$$

In this objective function, all weight coefficients are taken equal (w1 to w6). MSE error structure was used during the study. Later, the AO algorithm was modified according to the half normal and exponential distribution functions, and the MAO algorithm was used. Q and R matrices were obtained by using these algorithms and the results obtained are presented in Table 5. 3 DOF hover system has three angles as pitch yaw and roll. The results obtained with AO and MAO algorithms are presented in comparison with DPSO and FODPSO algorithms. First of all, the results obtained for Pitch angle are presented in Figure 8 comparatively. As can be seen from the results, the Q and R matrices obtained with the classical AO optimization algorithm have better control performance than the current results, DPSO and FODPSO algorithms. This is an indication that the newly proposed AO algorithm is an effective algorithm that can be used in real engineering problems. However, when the AO algorithm is updated according to half normal and exponential distributions, the results

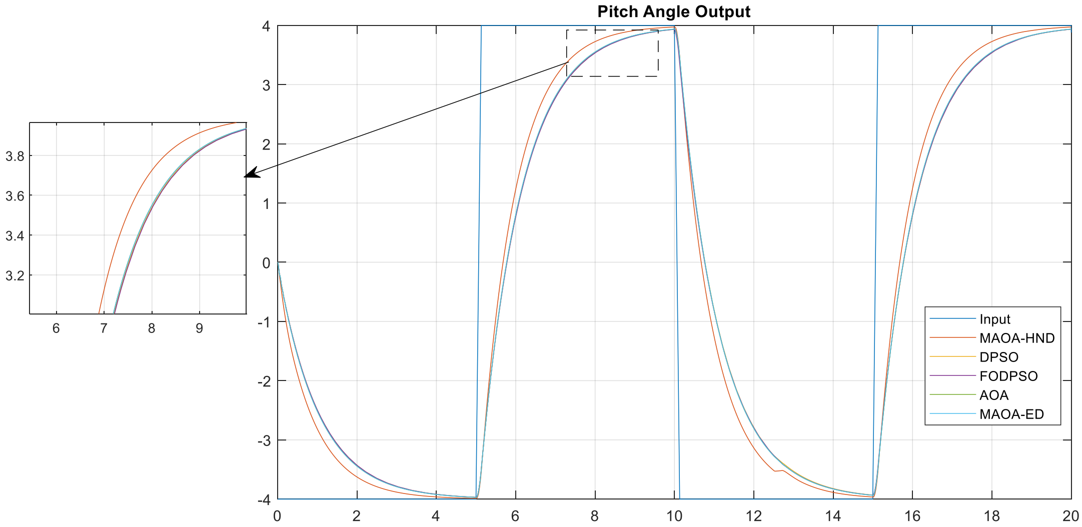

Figure 8: Pitch Angle Comparisons

obtained have better control performance, settling and rising time in DPSO and FODPSO algorithms than classical AO. These results show that the approach of using different distribution functions can be easily used in real engineering problems.

Comparisons of yaw and roll angles are shown in Figures 9 and 10. As can be seen from the results, the proposed MAO structure has better control performance than the current results. With the MAO algorithm, it has been seen that the performance of the approach of using different distribution functions can be increased without changing the basic philosophy of an existing algorithm. For 3 DOF Hover systems, the AO method has been updated with half normal and exponential distributions and the results are presented comparatively. As can be seen from the results, it has been shown that the approach of using different distribution functions can be used in real engineering problems.

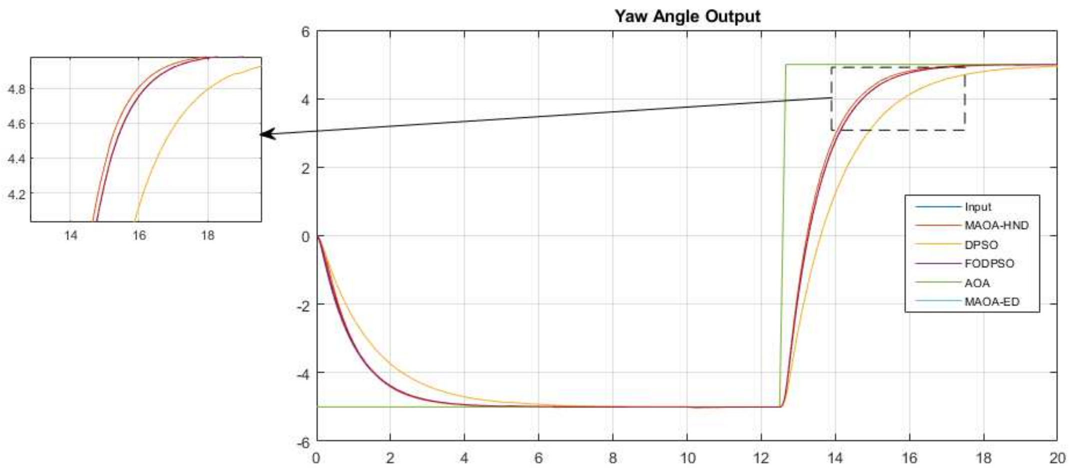

Figure 9: Yaw Angle comparisons

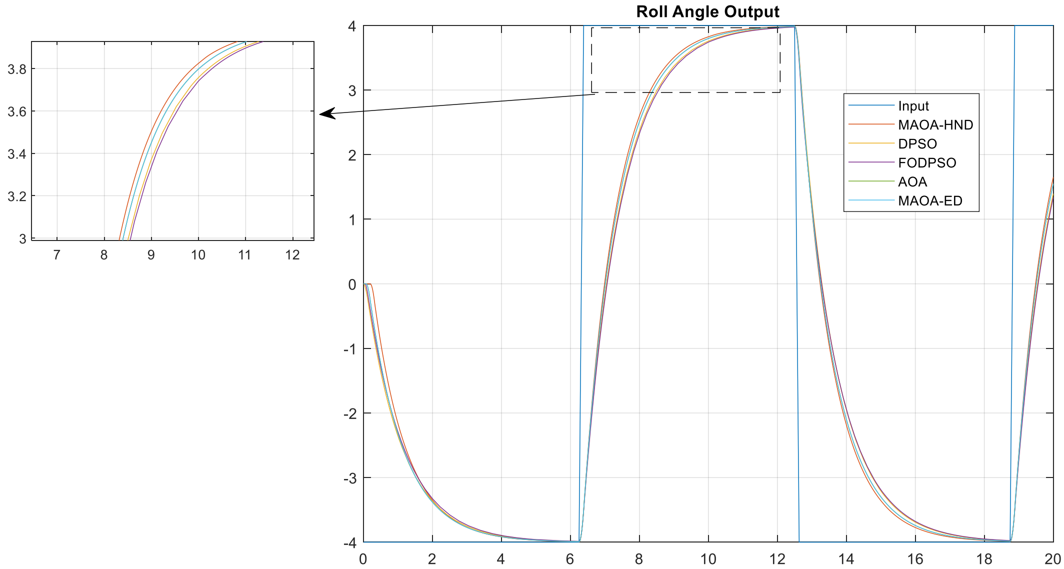

Figure 10: Roll Angle comparisons

Table 5. Optimized Q and R matrix value for 3 DOF Hover

Method	Q1	Q2	Q3	Q4	Q5	Q6	R1	R2	R3	R4
DPSO	16.1303	9.6459	8.8070	30.5133	10	9.6660	0.1013	0.0279	0.0609	0.0285
FODPSO	67.8073	9.5458	7.8138	55.0860	10	8.9105	0.0803	0.0325	0.0920	0.0215
AO	88.6741	24.0549	17.6776	71.873	24.5583	17.4327	0.212702	0.0832926	0.0832926	0.0651809
MAO with Half normal Distribution (HND)	159.0888	25.54199	16.08867	115.8188	18.74083	14.66382	1.017507	0.1349155	0.09527338	0.1371927
MAO with Exponential Distribution (ED)	88.6741	24.0549	17.6776	71.873	24.5583	17.4327	0.212702	0.0832926	0.0832926	0.0651809

4.2.3 Usage of Optimized Q and R Matrix in Experimental 3 DOF Hover System

In this section, the values obtained in Table 5 were directly tested on the real-time system given in Figure 6. Pitch roll and yaw angles comparisons are presented in Figures 11, 12 and 13, respectively. As can be seen from these results, the MAO algorithm updated with different distribution functions is an algorithm that can work in real-time engineering problems. In addition to these, it has been shown over the real-time control problem that the performance of an algorithm can be increased without changing its basic philosophy.

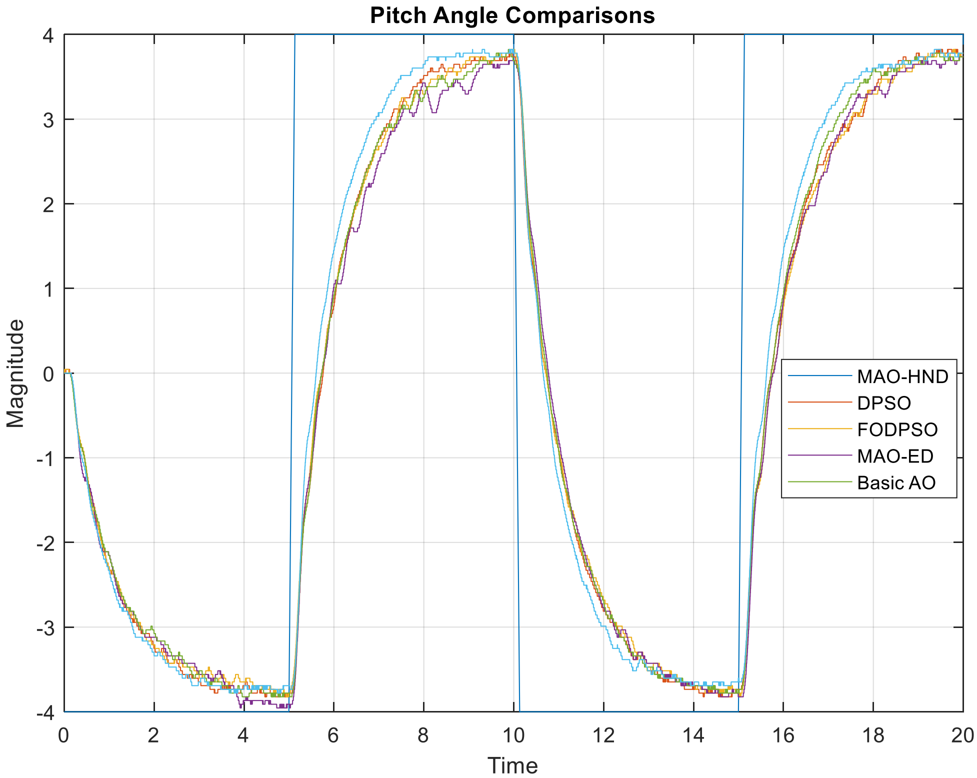

Figure 11: Real time Hover pitch angle comparisons

As can be seen from the results, the Q and R matrices obtained with the distribution function-based AO algorithm can also work in the real-time 3 DOF Hover system. It is very important to choose the appropriate distribution function in the method. Therefore, many distribution functions have been tried during this study and it has been determined that Half normal and Exponential distribution functions are suitable for the 3 DOF Hover simulation model. In addition to these, determining the parameters of distribution functions appropriately is important for the performance of the algorithm. After the distributions were determined, precise adjustments of the parameters were made with multiple tests. Furthermore, it has been determined that the Q and R matrices produced by the Half normal distribution based AO method in the 3 DOF Hover experimental system have better control performance. In future studies, it is planned to propose algorithms that can determine distribution function parameters with other algorithms and work simultaneously with the modified AO algorithm.

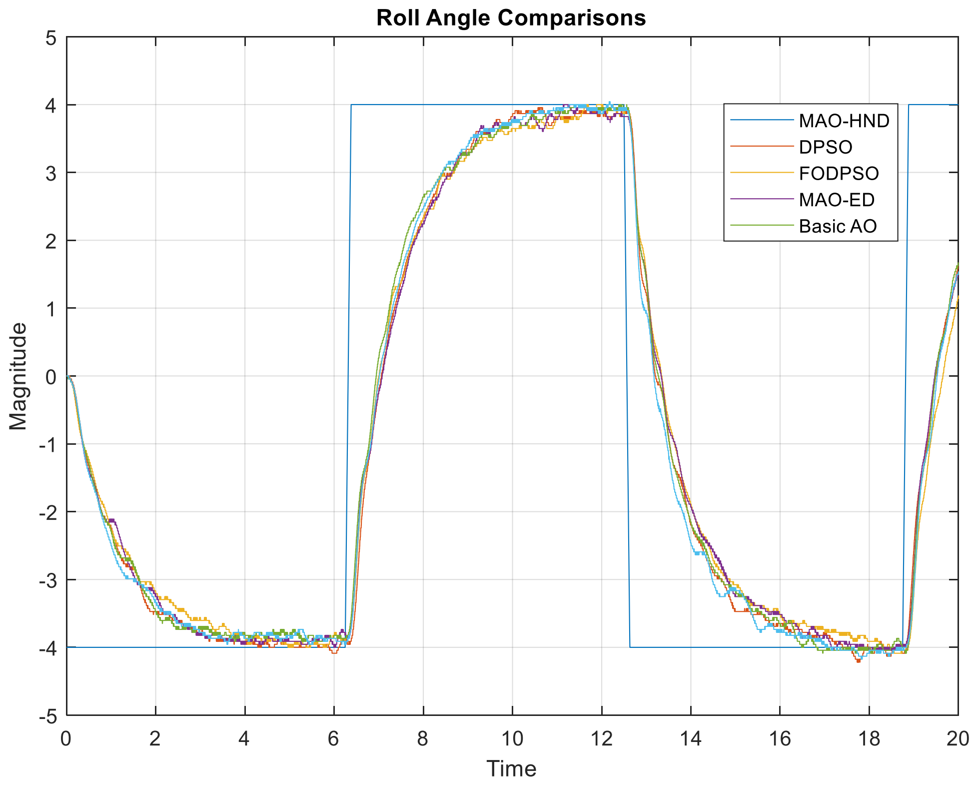

Figure 12: Real time Hover roll angle comparisons

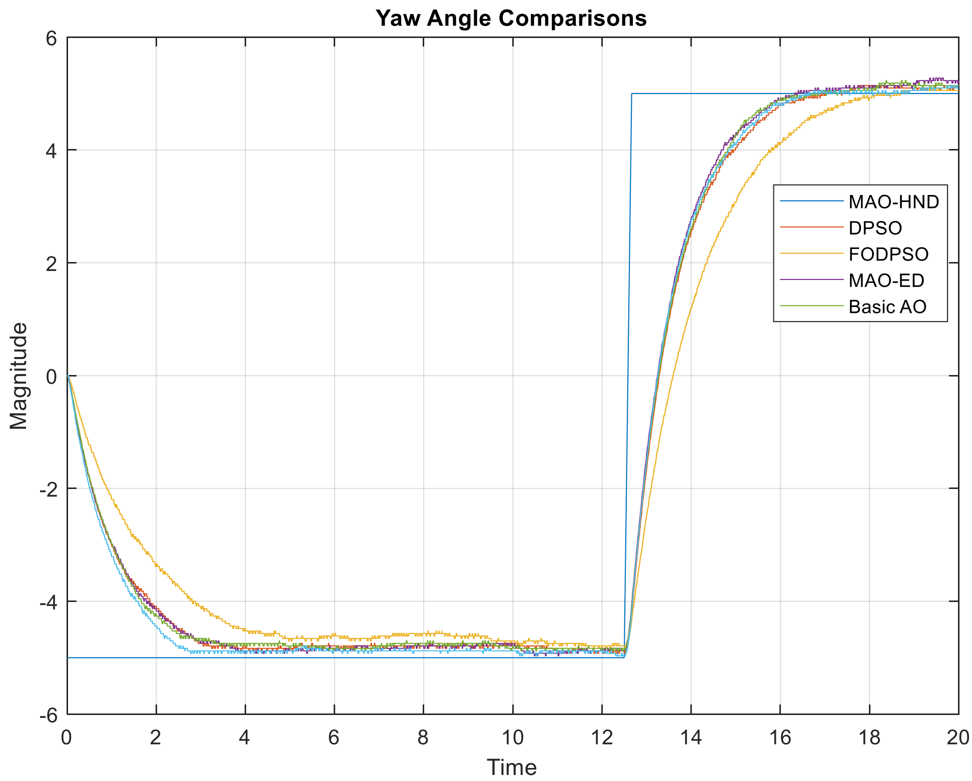

Figure 13: Real time Hover yaw angle comparisons

5 Conclusion

In this study, firstly, the generation of the initial population of the AO algorithm using basic mathematical operators, the random coefficients derived according to the uniform distribution in the exploration and exploitation phases are updated with the different distribution functions in the DF vector and the MAO algorithm is proposed. MAO algorithm is an algorithm obtained without changing the basic philosophy of the basic AO algorithm. In this study, Chi-Square, Gamma, Logistic, half normal, Exponential, Normal, Extreme Value, Inverse Gaussian distributions were used.

Benchmark functions were used for performance analysis in the study. The appropriate distribution function was determined for each benchmark function. And the results are compared with the results of the literature available in the tables. Especially, running benchmark functions according to 100, 500 and 1000 dimensions has shown the improvements in the performance of the algorithm presented. It has been observed that the proposed MAO algorithm performs better as the number of dimension increases. In this way, it has been shown that the approach of using different distribution functions can increase the performance of algorithms.

In addition to these, the Q and R weight matrices of the LQR control structure were optimized with the MAO algorithm to control the 3 DOF Hover quadcopter systems and the results were compared with the DPSO and FODPSO methods. Then, by using these weight matrices, the K gain matrix is calculated to minimize the 2^{nd} order linear error function and applied to the simulation model and real-time model of the system. Simulation results are obtained for pitch, roll and yaw angle and compared with the values on the system. Later, these values were applied to the real time system. As can be seen from the results, it is seen that the MAO method suggested in the experimental results in the simulation response has a better control performance. Especially, it has been observed that the modified versions of MAO with half normal and exponential distribution functions work better in 3 DOF hover problems. In addition, it is observed that the choice of weight matrices Q and R of the 2^{nd} order linear regulator to be applied to the system has a great influence on the system response. It has been shown that much better results can be obtained with Q and R weight matrices obtained by MAO methods, which are numerical optimization algorithms, especially in real time application.

In this way, it has been shown that by using different distribution functions in an algorithm, the optimization performance can be increased without changing the basic philosophy of the algorithm, using Benchmark functions and the MAO algorithm for real-time engineering problems. In future studies, it is planned to obtain better optimization performances by determining the parameters of distribution functions with the optimization to optimization (OtoO) approach.

Funding None. The authors declare that they have no conflicts of interest to disclose.

Declarations

Data availability The authors used own data and own coding.

Human or animal rights Humans and animals are not involved in this research work.

6 References

Abd Elaziz M, Attiya I (2020) An improved Henry gas solubility optimization algorithm for task scheduling in cloud computing. Artif Intell Rev. https://doi.org/10.1007/s10462-020-09933-3

Abd Elaziz M, Elsheikh AH, Oliva D, et al (2021a) Advanced Metaheuristic Techniques for Mechanical Design Problems: Review. Arch Comput Methods Eng. https://doi.org/10.1007/s11831-021-09589-4

Abd Elaziz M, Nabil N, Moghdani R, et al (2021b) Multilevel thresholding image segmentation based on improved volleyball premier league algorithm using whale optimization algorithm. Multimed Tools Appl 80:12435–12468. https://doi.org/10.1007/s11042-020-10313-w

Abd Elaziz M, Oliva D (2018) Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm. Energy Convers Manag 171:1843–1859. https://doi.org/10.1016/j.enconman.2018.05.062

Abd Elaziz M, Yousri D (2021) Automatic selection of heavy-tailed distributions-based synergy Henry gas solubility and Harris hawk optimizer for feature selection: case study drug design and discovery. Artif Intell Rev. https://doi.org/10.1007/s10462-021-10009-z

Abualigah L, Diabat A, Mirjalili S, et al (2021a) The Arithmetic Optimization Algorithm. Comput Methods Appl Mech Eng 376:113609. https://doi.org/10.1016/j.cma.2020.113609

Abualigah L, Yousri D, Abd Elaziz M, et al (2021b) Aquila Optimizer: A novel meta-heuristic optimization algorithm. Comput Ind Eng 157:107250. https://doi.org/10.1016/j.cie.2021.107250

Adam SP, Alexandropoulos S-AN, Pardalos PM, Vrahatis MN (2019) No Free Lunch Theorem: A Review. pp 57–82

Akdag O, Ates A, Yeroglu C (2020) Modification of Harris hawks optimization algorithm with random distribution functions for optimum power flow problem. Neural Comput Appl 1–27. https://doi.org/10.1007/s00521-020-05073-5

AKPAMUKÇU M, ATEŞ A (2020) Analyzing of Usage Effect of the Distribution Functions for SMDO Algorithm via Benchmark Function with Matlab Toolbox. DÜMF Mühendislik Derg. https://doi.org/10.24012/dumf.721670

Al-qaness MAA, Saba AI, Elsheikh AH, et al (2021) Efficient artificial intelligence forecasting models for COVID-19 outbreak in Russia and Brazil. Process Saf Environ Prot 149:399–409. https://doi.org/10.1016/j.psep.2020.11.007

Alagoz BB, Ates A, Yeroglu C (2013) Auto-tuning of PID controller according to fractional-order reference model approximation for DC rotor control. Mechatronics 23:. https://doi.org/10.1016/j.mechatronics.2013.05.001

Alijla BO, Wong L-P, Lim CP, et al (2014) A modified Intelligent Water Drops algorithm and its application to optimization problems. Expert Syst Appl 41:6555–6569. https://doi.org/10.1016/j.eswa.2014.05.010

Aljarah I, Habib M, Faris H, et al (2020) A dynamic locality multi-objective salp swarm algorithm for feature selection. Comput Ind Eng 147:106628.

https://doi.org/10.1016/j.cie.2020.106628

Ates A (2021) Enhanced equilibrium optimization method with fractional order chaotic and application engineering. Neural Comput Appl 1–28. https://doi.org/10.1007/s00521-021-05756-7

Ates A, Akpamukcu M (2021) Optimization to optimization (OtoO): optimize monarchy butterfly method with stochastics multi-parameter divergence method for benchmark functions and load frequency control. Eng Comput 1:3. https://doi.org/10.1007/s00366-021-01364-0

ATEŞ A, AKPAMUKÇU M (2021) Modified Monarchy Butterfly Optimization with Distribution Functions and its Application for 3 DOF Hover Flight System. (Under Rev Syst with Appl

Ates A, Alagoz BB, Chen YQ, et al (2020) Optimal Fractional Order PID Controller Design for Fractional Order Systems by Stochastic Multi Parameter Divergence Optimization Method with Different Random Distribution Functions. 9–14. https://doi.org/10.1109/iccma46720.2019.8988599

Ates A, Alagoz BB, Kavuran G, Yeroglu C (2017) Implementation of fractional order filters discretized by modified Fractional Order Darwinian Particle Swarm Optimization. Meas J Int Meas Confed 107:. https://doi.org/10.1016/j.measurement.2017.05.017

Ateş A, Yeroglu C (2016) Optimal fractional order PID design via Tabu Search based algorithm. ISA Trans 60:. https://doi.org/10.1016/j.isatra.2015.11.015

Ateş A, Yeroğlu C (2018) Modified Artificial Physics Optimization for Multi-parameter Functions. Iran J Sci Technol - Trans Electr Eng 42:465–478. https://doi.org/10.1007/s40998-018-0082-4

Chakraborty S, Mali K (2021) SuFMoFPA: A superpixel and meta-heuristic based fuzzy image segmentation approach to explicate COVID-19 radiological images. Expert Syst Appl 167:114142. https://doi.org/10.1016/j.eswa.2020.114142

Cheng M-Y, Prayogo D (2014) Symbiotic Organisms Search: A new metaheuristic optimization algorithm. Comput Struct 139:98–112. https://doi.org/10.1016/j.compstruc.2014.03.007

Coello Coello CA, Becerra RL (2004) Efficient evolutionary optimization through the use of a cultural algorithm. Eng Optim 36:219–236. https://doi.org/10.1080/03052150410001647966

Dai C, Zhu Y, Chen W (2007) Seeker Optimization Algorithm. pp 167–176

Dorigo M, Di Caro G Ant colony optimization: a new meta-heuristic. In: Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406). IEEE, pp 1470–1477

El Sehiemy RA, Selim F, Bentouati B, Abido MA (2020) A novel multi-objective hybrid particle swarm and salp optimization algorithm for technical-economical-environmental operation in power systems. Energy 193:116817. https://doi.org/10.1016/j.energy.2019.116817

Essa FA, Abd Elaziz M, Elsheikh AH (2020) An enhanced productivity prediction model of active solar still using artificial neural network and Harris Hawks optimizer. Appl Therm

Eng 170:115020. https://doi.org/10.1016/j.applthermaleng.2020.115020

Ewees AA, Elaziz MA (2020) Performance analysis of Chaotic Multi-Verse Harris Hawks Optimization: A case study on solving engineering problems. Eng Appl Artif Intell 88:103370. https://doi.org/10.1016/j.engappai.2019.103370

Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH (2020) Marine Predators Algorithm: A nature-inspired metaheuristic. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2020.113377

Fathy A, Elaziz MA, Alharbi AG (2020) A novel approach based on hybrid vortex search algorithm and differential evolution for identifying the optimal parameters of PEM fuel cell. Renew Energy 146:1833–1845. https://doi.org/10.1016/j.renene.2019.08.046

Fathy A, Elaziz MA, Sayed ET, et al (2019) Optimal parameter identification of triple-junction photovoltaic panel based on enhanced moth search algorithm. Energy 188:116025. https://doi.org/10.1016/j.energy.2019.116025

Holland JH (1992) Genetic Algorithms - Computer programs that “evolve” in ways that resemble natural selection can solve complex problems even their creators do not fully understand. Sci. Am. 66–72

Içen M, Ateş A, Yeroğlu C (2017) Optimization of LQR weight matrix to control three degree of freedom quadcopter. In: IDAP 2017 - International Artificial Intelligence and Data Processing Symposium

Kandhway P, Bhandari AK, Singh A (2020) A novel reformed histogram equalization based medical image contrast enhancement using krill herd optimization. Biomed Signal Process Control 56:101677. https://doi.org/10.1016/j.bspc.2019.101677

Khanduzi R, Maleki HR, Akbari R (2018) Two novel combined approaches based on TLBO and PSO for a partial interdiction / fortification problem using capacitated facilities and budget constraint. Soft Comput. https://doi.org/10.1007/s00500-018-3005-4

Mehrabi P, Honarbari S, Rafiei S, et al (2021) Seismic response prediction of FRC rectangular columns using intelligent fuzzy-based hybrid metaheuristic techniques. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s12652-020-02776-4

Moosavian N, Kasaee Roodsari B (2014) Soccer league competition algorithm: A novel meta-heuristic algorithm for optimal design of water distribution networks. Swarm Evol Comput 17:14–24. https://doi.org/10.1016/j.swevo.2014.02.002

Oliva D, Aziz MA El, Hassanien AE (2017) Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. Appl Energy 200:141–154

Ouadfel S, Abd Elaziz M (2020) Enhanced Crow Search Algorithm for Feature Selection. Expert Syst Appl 159:113572. https://doi.org/10.1016/j.eswa.2020.113572

Pereira RL, Kienitz KH (2015) Experimental investigation of nonlinear controllers applied to a 3DOF hover: SMC via ALQR approach. In: 2015 23rd Mediterranean Conference on Control and Automation, MED 2015 - Conference Proceedings. Institute of Electrical and Electronics Engineers Inc., pp 520–524

Prach A, Kayacan E, Bernstein DS (2016) An experimental evaluation of the forward propagating Riccati equation to nonlinear control of the Quanser 3 DOF Hover testbed. In: Proceedings of the American Control Conference. Institute of Electrical and

Electronics Engineers Inc., pp 3710–3715

Rahman MFA, Radzuan SM, Hussain Z, et al (2018) Performance of loiter and auto navigation for quadcopter in mission planning application using open source platform. In: Proceedings - 7th IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2017. Institute of Electrical and Electronics Engineers Inc., pp 342–347

Sadollah A, Eskandar H, Bahreininejad A, Kim JH (2015) Water cycle algorithm with evaporation rate for solving constrained and unconstrained optimization problems. Appl Soft Comput J 30:58–71. https://doi.org/10.1016/j.asoc.2015.01.050

Shadravan S, Naji HR, Bardsiri VK (2019) The Sailfish Optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems. Eng Appl Artif Intell 80:20–34. https://doi.org/10.1016/j.engappai.2019.01.001

Storn R, Price K (1997) Differential Evolution :A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. J Glob Optim 11:341--359

Tamura K, Yasuda K (2011) Spiral optimization -A new multipoint search method. In: 2011 IEEE International Conference on Systems, Man, and Cybernetics. IEEE, pp 1759–1764

Turky A, Sabar NR, Sattar A, Song A Evolutionary Learning based Iterated Local Search for Google Machine Reassignment Problems

Yeroğlu C, Ateş A (2014) A stochastic multi-parameters divergence method for online auto-tuning of fractional order PID controllers. J Franklin Inst 351:. https://doi.org/10.1016/j.jfranklin.2013.12.006

Zhang KK, Ye D, Zhao XG (2014) Fault tolerant tracking control for quad-rotor helicopter via robust adaptive technique. In: Proceedings of the 33rd Chinese Control Conference, CCC 2014. IEEE Computer Society, pp 3233–3237

Probability density function - MATLAB pdf. https://www.mathworks.com/help/stats/prob.normaldistribution.pdf.html. Accessed 12 Mar 2020b

3 DOF Hover - Quanser. https://www.quanser.com/products/3-dof-hover/. Accessed 27 Jul 2020c