Int. J. Nonlinear Anal. Appl. 11 (2020) No. 1, 137-157
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2020.4245

Woodpecker Mating Algorithm (WMA): a nature-inspired algorithm for solving optimization problems

Morteza Karimzadeh Parizi[a], Farshid Keynia*[b], Amid khatibi Bardsiri[a]

[a] *Department of Computer Engineering, Kerman Branch, Islamic Azad University, Kerman, Iran;*
[b] *Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology, Kerman, Iran;*

(Communicated by J. Vahidi)

Abstract

Nature-inspired metaheuristic algorithms have been a topic of interest for researchers to solve optimization problems in engineering designs and real-world applications, due to their simplicity and flexibility. This paper presents a new nature-inspired search algorithm called Woodpecker Mating Algorithm (WMA) and applies it to challenging problems in structural optimization. The WMA is a population-based metaheuristic algorithm that mimics the mating behavior of woodpeckers. It was inspired by the drumming sound intensity. In WMA, the population of woodpeckers is divided into male and female groups. The female woodpeckers approach the male woodpeckers based on the intensity of their drum sound. An efficiency comparison was drawn between the WMA algorithm and other metaheuristic algorithms by employing 19 benchmark functions(including unimodal, multimodal and composite functions). Moreover, the performance of WMA is compared with 8 of the best meta-heuristic algorithms using 13 high dimensional multimodal and unimodal benchmark functions. The assessments and statistical results indicate that the WMA algorithm offers promising results and is capable of outperforming the most recent and popular algorithms proposed in the literature in most of the employed benchmark functions. Moreover, a statistically significant difference was observed compared to the other assessed algorithms. The proposed algorithm produced significant results for a non-convex, inseparable, and scalable problems.

*Farshid Keynia

Email address: mkarimzadeh313@gmail.com, fkeynia@gmail.com, khatibi_amid@yahoo.com (Morteza Karimzadeh Parizi[a], Farshid Keynia*[b], Amid khatibi Bardsiri[a])

Received: Jun 2019 *Revised:* November 2019

Keywords: Metaheuristic, Optimization, Woodpecker, Drumming, Sound intensity.
2010 MSC: 68T99,68W25,68T20.

1. Introduction

Optimization is the process of obtaining optimal values for the parameters of a specific system from all of the possible values to maximize or minimize its output. The optimization problem can be observed in all research fields because it has led to the development of optimization techniques and also the emergence of intriguing areas for researchers [1]. In the past two decades, there has been a growing interest in metaheuristic algorithms in order to solve the optimization problems[2]. Metaheuristic algorithms are popular because they regard a problem as a black box [1, 3]. They avoid the local optimum due to their random nature [2, 4]. Finally, they are easy to learn and implement [5, 6].

Since metaheuristic algorithms are mainly based on the regulations of natural organisms, they are called nature inspired [7]. Depending on the nature of a simulated phenomenon, metaheuristic algorithms can be categorized into four classes: evolutionary, physics-based, swarm intelligence and inspired by human behaviour [2].

The evolutionary algorithms (EA) are generally certain optimization algorithms inspired by the Darwinian principles of nature in relation to the abilities to live organisms to evolve and adapt to environmental conditions. For instance, Genetic Algorithm(GA) [8, 9], Differential Evolution (DE) [10], Evolutionary Programming (EP), Evolution Strategies (ES) [11], are evolutionary algorithms.

The second class includes physics-based techniques. These optimization algorithms are usually the simulations of physical laws. For instance, they apply to motion, gravity, radiation, electromagnetic, weight, etc [5]. The instances are Multi-Verse Optimizer (MVO) [3], Simulated Annealing (SA) [12], Thermal Exchange Optimization (TEO) [13] and Orientation Search Algorithm (OSA) [14].

The third class of metaheuristic algorithms was inspired by human behaviour. The instances include Harmony Search (HS) [8, 15], Brain Storm Optimization Algorithm (BSOA) [16], Imperialist Competitive Algorithm (ICA) [17], Kidney-inspired algorithm (KA) [18], Interior Search Algorithm (ISA) [19], Single Seekers Society (SSS) [20], Sine Cosine Algorithm (SCA) [21], Poor and rich optimization algorithm (PRO) [22], group teaching optimization algorithm (GTOA) [23] and Teaching Learning Based Optimization (TLBO) [24].

The fourth class is called swarm intelligence (SI), which is a distributed intelligence model used to solve optimization problems. SI was inspired by the collective behaviour of a swarm of social insects and other animal swarms. SI is usually made up of a simple population of elements (an institution which is able to carry out certain operations) which operate with each other and the environment locally. With very limited capabilities, these institutions can cooperate to carry out very complicated tasks for survival [25]. Particle Swarm Optimization (PSO) [26, 27] , Artificial Bee Colony (ABC) [28], Ant Colony Optimization (ACO) [29], Satin Bowerbird Optimizer (SBO) [30], Dragonfly algorithm (DA) [31], Grey Wolf Optimizer (GWO) [5], Moth-flame Optimization (MFO) [32], Bat Algorithm (BA) [33], Firefly Algorithm (FA) [34], Cuckoo Search (CS) [35], Ant Lion Optimizer (ALO) [4], Whale Optimization Algorithm [2] ,Harris Hawks Optimizer (HHO) [36], Crow Search Algorithm (SCA) [37, 38], Donkey and Smuggler Optimization Algorithm (SDO) [39] and Artificial Acari Optimization (AAO) [40] are SI algorithms.

Exploration and exploitation are what population-based metaheuristic algorithms have in common [41]. In the exploration phase, an algorithm should be equipped with certain mechanisms for exploring the entire search space. In fact, the promising areas of the search space are identified in this phase [3]. Exploration is meant to analyze the problem at a global level and to identify certain areas of the search space with the best global solution [42]. The exploitation phase comes after exploration.

It can be defined as the process of analyzing a promising area found in the exploration phase [2]. The exploitation phase emphasizes the local search, and the algorithm converges to the promising areas found in the exploration phase [3]. Exploration and exploitation are two opposite turning points. Promoting the results of one phase downgrades the results of the other. The proper balance between these two turning points can guarantee an accurate estimation of the global optimum by using population-based algorithms [32].
In this study, a novel nature-inspired metaheuristic algorithm was introduced based on the mating behavior of woodpeckers. Woodpeckers peck at trees (drumming) to attract the opposite sex. Pecking results in a sound wave. According to the physics laws, sound waves propagate in the environment so that other woodpeckers can hear them. Therefore, the physical quantity is defined as sound intensity, on which the amount of sound received by a listener depends. Such concepts provided the inspiration for the proposed algorithm.
The rest of this paper contains four sections. In Section 2, the woodpecker mating algorithm is introduced. In Section3, mathematical benchmark functions are used to evaluate the proposed algorithm. Finally, Section 4, presents the conclusion and suggestions for future studies.

2. Woodpecker Mating Algorithm (WMA)

Woodpeckers are wonderful birds. There are nearly 200 different species of them. Woodpeckers use a specific strategy for communication. It is called drumming or pecking the trunks of trees. Drumming gives provides woodpeckers with special opportunities. They make holes into the trunks of trees to build nests and feed on insects or resin. By doing so, they can also communicate with other woodpeckers, show their territories, and scare their enemies. However, the most important purpose of drumming is to attract mates in the mating season [43]. In fact, drumming is an intra-gender competition in the mating season. Male woodpeckers compete with each other to attract female birds by drumming. Before a female woodpecker selects a mate, it listens to multiple drumming sounds and analyzes them instinctively. Then it gets attracted to the sound with the highest quality and flies toward the source of drumming (the male woodpecker). The drumming rate differs in different species. It falls down in larger species, and they make louder phones instead. However, drumming is the main non-phonetic tool for long-distance communication. The WMA algorithm was inspired by the mating behaviour of red-bellied woodpeckers (*Melanerpes carolinus*) [44]. In the proposed algorithm, it is assumed that drumming is the only communication tool used by woodpeckers.
At the beginning of the mating season, male woodpeckers start drumming. The quality of the sound produced by the male has a great effect on the attraction of female woodpeckers. Like other animals, woodpeckers try to attract and choose the best mate. Birds with the higher ability for drumming can produce stronger, higher quality drums and are regarded as ideal mates. Their drum can be heard farther away and attract more female woodpeckers. The female woodpeckers are then attracted to the source, because for them a more powerful sound connotes the male's higher ability to find food, nest and reproduce, making them a better option as mate. As a result, the acoustic power **(sound intensity)** of the male woodpecker's drum indicates its ability to attract female birds. In other words, the size of a female woodpecker's movement toward a male woodpecker depends on the quality of sound it hears. The stronger the sound, the closer the female woodpecker gets to the male bird producing it. It is worth noting that powerful or high quality sound means the physical quantity of **"sound intensity"** as described below.
A female woodpecker that is attracted to the sound of a male bird moves toward and approaches it. This process is repeated in several intervals (or several days) and each time, the female woodpecker gets closer to the male. The male woodpecker performs drumming at different time intervals, and

these periodic drums attract the female bird step by step. This approach is similar to an evolutionary process, and by each iteration, the female woodpecker gets closer to the male. On the other hand, when a male woodpecker strikes a tree, the sound is transmitted to the environment and to the ears of other females, so they move toward the male bird. Since the basis of the movement is the position of the male bird, and the female woodpeckers move toward it based on the information obtained from processing the male bird's drum, there is a process of communication and flow of information between the male and the female woodpeckers. So, this process is a swarm intelligence behavior.In the nature, at the beginning of the mating season, many male woodpeckers begin drumming to attract female woodpeckers as mates. As a result, at each time interval, a female woodpecker hears the drum sound of several males at the same time. As mentioned in the preceding paragraph, female woodpeckers look for the best males. But if another male (other than the best male) is closer to the female, the female bird will be attracted to this male because it will receive a better quality drum due to the shorter distance from the sound source.

2.1. Sound Intensity

In the physics of sound waves [45], the received sound depends on a quantity named the sound intensity. The sound intensity *(SI)* of one level is the average energy changes reaching or exceeding a level. It can be calculated through Equation (2.1):

$$SI = \frac{P}{A} \tag{2.1}$$

In this equation, P is the sound power, and A is the area meeting the sound.

It is often complicated to determine how sound intensity changes over the distance from a real source of the sound. Some of the sources (such as speakers) may send out the sound only in one direction. The environment usually generates some echoes overlapping with the direct sound wave. However, the echoes can sometimes be ignored in some cases. It can be assumed that the source of sound is a point emitting the sound in an isotropic way, i.e. with the same intensity in all directions (This assumption is true about woodpeckers.). In addition, the mechanical energy of sound waves (P) is conserved when waves are spread from a source. If a sphere is assumed with a radius of r around the source, All the energy emitted by the source must pass through the surface of the sphere. Thus, the transfer rate of sound wave energy through the perimeter should be equal to the propagation rate of the source (P_S). As a result, sound intensity (I) is defined as Equation (2.2):

$$SI = \frac{P_S}{4\pi r^2} \tag{2.2}$$

Here $4\pi r^2$is the area of the sphere. According to Equation (2.2), the sound intensity of an isotropic point source decreases by the squared distance r^2. Therefore, the intensity of the received sound depends greatly on the distance between an object and the source. The shorter the distance is, the higher and better the sound will be received. The Euclidean distance (r) can be calculated through Equation (2.3):

$$r = \|X_S - X'\| \tag{2.3}$$

Here $X_S$is the position of the sound-generating source, and X' is the position of a listener.

So far, it has been assumed that female woodpeckers fly towards a male woodpecker based on sound intensity(I). Other main factor is distance. Intensity of the received sound is depended on the distance between woodpeckers.

2.2. The WMA algorithm assumptions

The WMA algorithm is inspired by the aforementioned concepts and based on the following assumptions.
1. The only way of communication for the woodpeckers is the sound produced via drumming.
2. All female woodpeckers hear a percentage of the drum from most appropriate male per their distance and are attracted to it.
3. The fitness of a male woodpecker is calculated based on the objective function.
4. In the WMA algorithm, fitness is considered an attractiveness factor. Female woodpeckers are attracted to the more qualified male woodpeckers. The best woodpecker, also called the gpop.
5. The rate at which a female woodpecker is attracted to a male bird depends on the "sound intensity" of the drum she hears. On the other hand, the rate of attraction, according to the laws of physics in terms of acoustic waves, decreases with increasing distance from the source. The smaller the distance, the greater the amount of attraction for a female woodpecker.
6. In the WMA, woodpecker populations are considered as male and female woodpeckers based on their level of fitness. The male woodpecker population is large at the beginning of exploration (because at the beginning of the mating season the number of single males is high). But this population decreases with the increase in the number of iterations of the algorithm (as the mating season advances, the number of single male birds declines, due to successful mating). This process leads to exploitation.
7. In early iterations (at the beginning of the mating season), female woodpeckers are attracted to the most appropriate and the least distant male woodpeckers. But in the later iterations, they are attracted only to the best woodpecker.
8. For the sake of simplification, it is assumed that, besides the gpop drum, each female woodpecker will hear only one other male woodpecker at any certain point of time, that is, the male which is the closest to the female bird, and therefore, has a higher drum sound intensity.

As mentioned above, WMA is a swarm intelligence algorithm that inspired on the mating behavior of woodpeckers. There are some similarities between the WMA and the Firefly Algorithm (FA) in scientific terms. But the two are very different from technical and operator aspects. In FA, the concept of light intensity is discussed as the amount of attraction of two fireflies, whereas in the WMA, the concept of sound intensity is used as a measure of the attraction of search agents. These two concepts are quite different in terms of physics laws and formulations. In FA, the population of fireflies are considered unisex, and each firefly can be attracted by a more fitting search agent. However, the woodpecker population in the WMA is considered as male and female woodpeckers. Each female woodpecker is attracted to the best (or the gpop) and the nearest male woodpecker. Unlike FA, the WMA algorithm has several operators for sequential and efficient implementation of exploration and exploitation phases that have a significant effect on WMA performance and its global and local search capability

2.3. Mathematical model and optimization algorithm

The WMA algorithm consists of two factors: male and female woodpeckers. Male and female woodpeckers are each considered a candidate solutions. In the WMA algorithm, male and female woodpeckers do not differ based on their gender but on the degree of fitting (value of objective function). In addition, the difference between male and female woodpeckers lies in the method of exploring the problems space and updating their positions. Female woodpeckers are the main search agents that move through the problem space and the males are the best positions found so far by them. In fact, males are signs and flags found by females. Hence, female woodpeckers search around for male

```
Initialize the woodpeckers randomly
Calculate the fitness of each woodpeckers
gpop: the best of woodpeckers (solutions)
Calculate THα by Eq. (10)
While ( iteration < maximum number of iteration)
   Calculate size of male woodpecker population by Eq. (8)
   divide woodpeckers into male and female groups
   For each  woodpecker
      Select mj (jth male woodpecker)
      Calculate α_gpop and α_mj by Eq. (6)
      Calculate δ_i by Eq. (5)
      Update the position of woodpecker by Eq. (4)
      Running Away Function
         If α^i_gpop > THα
            Random Running Away  (RRA) by Eq. (11)
         Else
            Gpop Running Away (GRA) by Eq. (14)
         End if
      Evaluate the new position of female woodpecker
      Update gpop if there is a better solution
   End for
   iteration= iteration+1;
End while
Return gpop
```

Figure 1: Pseudo code of the WMA

birds and update their position based on the location of male birds. If they find a better candidate solution, they will update the position of the male woodpecker.

2.3.1. Initialization

The WMA algorithm is a population-based algorithm. Here every woodpecker is considered a candidate solution. Like other metaheuristic algorithms, the proposed algorithm initializes with a group of random woodpeckers (random candidate solutions). In fact, woodpeckers are distributed uniformly in the search space. Every woodpecker is an n-dimensional solution vector for the problem. Other parameters of the algorithm such as maximum iteration, population size, RA probability, and sound wave power are adjusted in this step. Fig. 1 shows the pseudo code of the proposed WMA algorithm.

2.3.2. Evaluation and Divide Population of the Woodpeckers

Like other birds, woodpeckers tend to select the best mate. Thus, the better young woodpeckers can be born. In this step, the population of woodpeckers is evaluated by the objective function to determine their fitness values. Then the population of woodpeckers is divided into male and female groups. At the beginning of each iteration, the woodpecker population is sorted on the basis of the rate of fitness value according to the objective function, and proportional to the size of the male woodpecker population, the most qualified woodpeckers are considered to be male birds. In each iteration, male woodpeckers are search agents with the highest degree of fitness based on the objective function. The male with the highest degree of fitness is selected as the best gpop. The gpop is the most attractive woodpecker, and all female woodpeckers hear a percentage of its drum relative to their distance, and move toward it.

2.3.3. Woodpeckers in Motion

In this step, every woodpecker updates its position through Equation (2.4).

$$x_i^{t+1} = x_i^t + r * \frac{\delta_i^t * \Big(\alpha_{gpop} * (x_{gpop}^t - x_i^t) + \alpha_{mj} * (x_{mj}^t - x_i^t)\Big)}{2} \tag{2.4}$$

Here x_i^t indicates the previous position of the ith woodpecker, and x_{gpop}^t shows the position of the best member (gpop, the best male). Moreover, x_{mj}^t is the position of the jth male woodpecker, r is a random number of a uniformly distribution selected from [0, 1], δ_i^t shows the random coefficients for the i-th woodpecker in iteration t. The values of these coefficients are calculated adaptively along the iteration cycle of the algorithm based on Equation (2.5). The $\alpha(\alpha_{gpop},\ \alpha_{mj}\ )$ is a parameter which can be determined through Equation (6(. In fact, α indicates how much a woodpecker is attracted to a male woodpecker (i.e. the target position) based on a ratio of the received sound intensity (really α determines the amount of step).

As mentioned in the preceding section, a female woodpecker may, at each time slice of the mating season, be affected by and attracted not only to the gpop but also to the drum sound of the closest male woodpecker. In equation (2.4) xmj is the indicator of the position of this male. In fact, every female woodpecker updates its position based on gpop and a male woodpecker selected as mj. Regarding the selection of an $mj$from male birds, the Euclidean distance of a female woodpecker (i) from every male bird is determined in each iteration through Equation (2.3). Then the male bird being on the shortest distance from the female woodpecker will be selected as the mj. In fact, gpop shows the best male bird in the population. As a result, it generates the highest quality of drumming sounds influencing the female woodpeckers. The male mj is near a female bird; thus, it pecks at trees to influence the female bird and make it develop certain desires. Fig. 2 shows how a female woodpecker updates its position according to gpop and mj in a 2D search space.

$$\delta_i^t = r * b \tag{2.5}$$

In this equation, r is a random number from a uniform random distribution in [0, 3]. The value of parameter b decreases from 0.77 to 0 during the iteration cycle of the algorithm. With this reduction, the exploration and exploitation phases are respectively implemented efficiently. The value of b in each iteration of the algorithm is calculated using equation (2.7).

δ_i^t has a random value in the range of [0, 3b] which decreases the value of δ_i^t as well as its fluctuation range by decreasing b during the iteration cycle of the algorithm. If the value of δ_i^t is in the range of [0, 1],the new position of a female woodpecker is placed at any random point between its current position and the target woodpecker's position (gpop or mj). In other words, female woodpeckers converge and get closer to the target woodpecker. This practice emphasizes exploitation phase particularly on end iterations. In this case, search agents are required to search locally in promising areas around male woodpeckers in order to enhance the quality of the received solutions. On the other hand, if δ_i^t>1, then female woodpeckers are required to far away the target woodpecker and diverge from it. This results in the exploration of new promising areas for better solutions. In this case, the WMA algorithm can search globally. In short, if $\delta_i^t \le 1$, then the female woodpeckers have to converge to the target woodpecker (Fig. 3a); otherwise, they have to diverge from it (Fig. 3b).

$$\alpha=\frac{1}{1+SI_j^i} \tag{2.6}$$

In this equation, α shows the probability (attraction) of male woodpecker j for woodpecker i and SI_j^i is the sound intensity of the target woodpecker (j), reaching female woodpecker i. In fact, Parameter α are amount of the pace in relation to the attractiveness (sound intensity) of the selected male woodpecker. In fact, the pace is inversely related to the sound intensity. In other words, the higher the sound intensity of a destination woodpecker is for female woodpecker, the greater the denominator will be. Therefore, the pace will reduce. The pace is selected from (0, 1]. The greatest value is obtained when the attractiveness of the destination woodpecker approaches

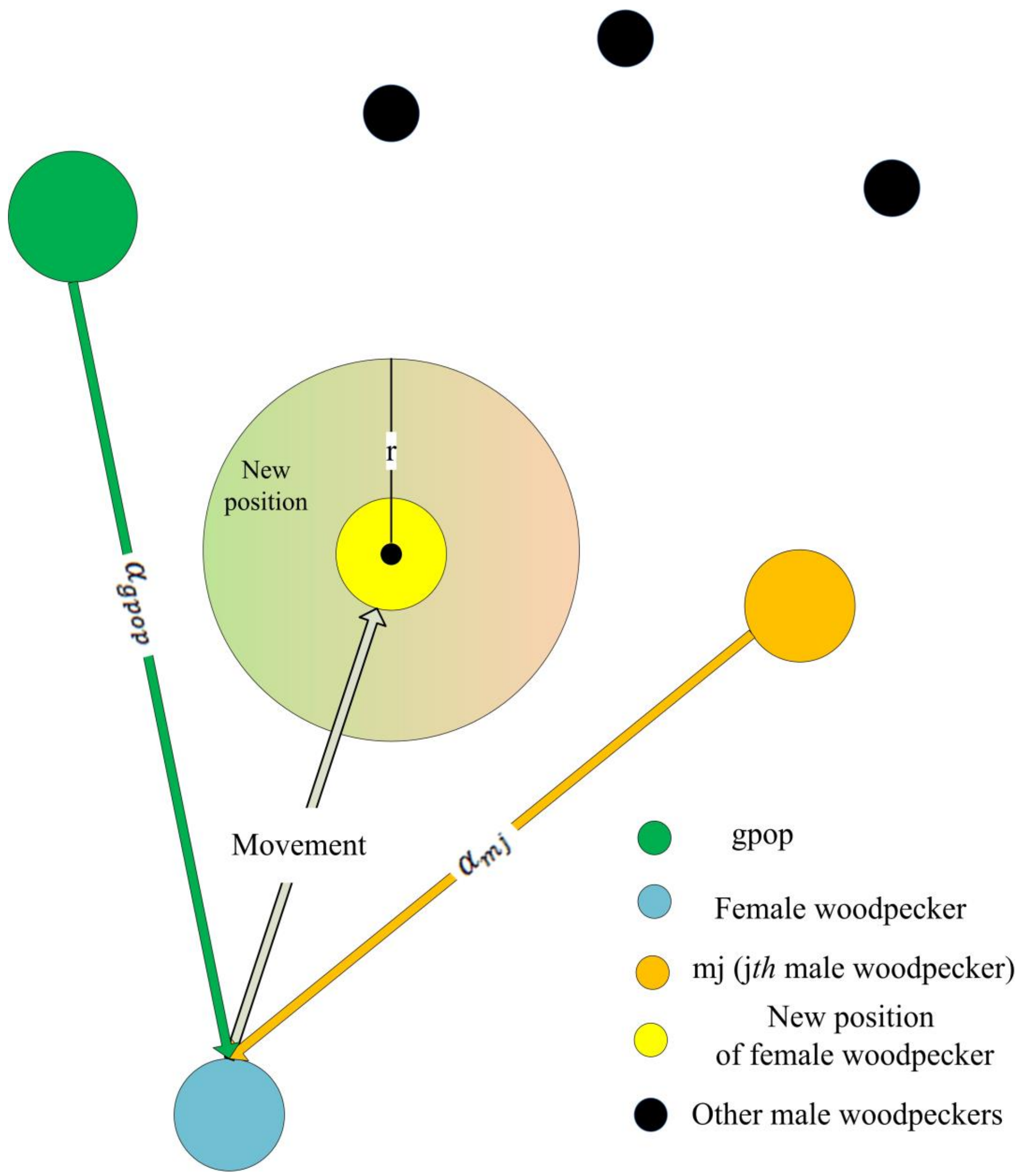

Figure 2: Position updating in WMA Algorithm

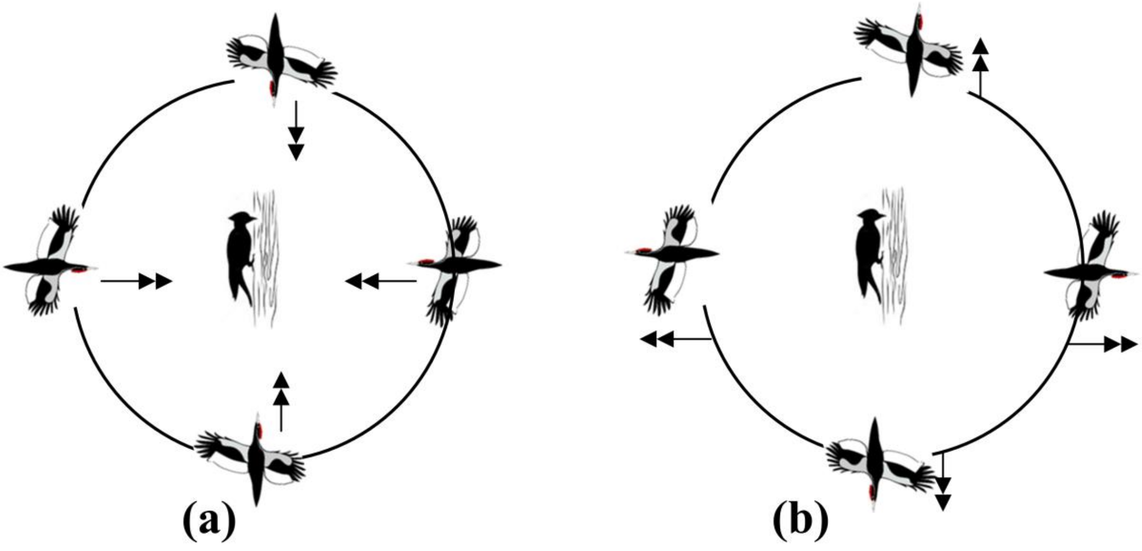

Figure 3: illustrates the influence of the parameter δ_i^t on the next position of the female woodpecker.

zero (the received sound intensity is too low). The smaller the pace is, the more accurately a woodpecker moves towards the destination. A woodpecker moves around the centralized target, something which increases exploitation and results in a more accurate estimated optimal solution. Therefore, parameter α has an effect on the exploitation of the algorithm.

$$b = Tansig\left(1 - \frac{t}{t_{max}}\right) \tag{2.7}$$

Here $Tansig$ is the tangent sigmoid function, in which t is the current iteration number, and t_{max} is the maximum number of iterations.
The population of male woodpeckers decreases during the algorithm iteration cycle adaptively. In the final iterations, only one male woodpecker will remain. The large number of male woodpeckers can increase exploration in the initial iterations. Moreover, the algorithm is prevented from being trapped in local optimums. In the final iterations, decreasing the number of male birds increases exploitation and accuracy of solutions. Equation (8(can be used for determining the number of male woodpeckers in each iteration.

$$Male_{nu} = \left\{Round\left(\frac{N}{2} * \left(1 - \frac{t}{t_{max}}\right)\right) + 1\right\} \tag{2.8}$$

Here $Male_{nu}$ is the number of male woodpeckers in each iteration, and t is the current iteration number. Moreover, t_{max} shows the maximum number of iterations, when N indicate the maximum number of woodpeckers population.
As mentioned earlier, the population of male woodpeckers in the iteration cycle of the algorithm decreases linearly with increase of iterations. The minimum population for male woodpeckers is one woodpecker and that is the gpop. In this case, equation (2.4) is simplified as equation (2.9).

$$x_i^{t+1} = x_i^t + r * (\delta_i^t * (x_{gpop}^t - x_i^t) * \alpha_{gpop}) \tag{2.9}$$

2.3.4. Running Away Function

When a female woodpecker is attracted to the sound of another male woodpecker and moves towards it, sounds may overlap in nature. Therefore, the male woodpecker may change paths unconsciously and deviates. On the other hand, a woodpecker might be attacked by other woodpeckers or hunting birds on the path. It may also make random changes to the path due to feeling danger. Such random changes are stimulated as Running Away (RA) function. In other words, path changes are regarded as random changes in the solutions. How the female woodpecker escapes can vary depending on the sound intensity it receives from the best member (gpop) of the population. In this section, two types of movement are considered as escape of woodpeckers. The criterion for selecting the type of movement is based on the sound intensity threshold received from the gpop *($TH\alpha$)*. The value of $H\alpha$ is calculated by equation (2.11). Based on what was stated in the previous section, the value of $H\alpha$ is inversely related to the sound intensity. The larger the value of α^i_{gpop}, the greater the distance between the woodpecker and the gpop and the lower the quality of the drum. If $\alpha^i_{gpop} > H\alpha$, it is assumed that the woodpecker is far from the gpop and that the drum will be received with a very low sound intensity. In this case, the female woodpecker is at an inappropriate point relative to the gpop position, so it flies quite randomly based on the equation (2.10) to another point of search space (forest). This movement is called Random Running Away (RRA).

$$TH\alpha = 0.8 * \frac{\sum_1^{n-1} \alpha^i_{gpop}}{n-1} \tag{2.10}$$

in which $H\alpha$ is the threshold for the gpop sound intensity that is calculated in the first iteration, n is size of the woodpeckers depart from, and $\alpha^i_{gpop}$is calculated proportional to the sound intensity of the woodpecker i, whose value is calculated using equation (2.6).

$$x^i_{RRA} = lb - (lb - ub) * r \tag{2.11}$$

In this equation, x^i_{RRA} is the position of a new element obtained from RRA on the ith woodpecker, r is a random number of a uniformly continuous distribution selected from [0, 1]. lb and ub are lower and upper boundaries of variables Respectively.
If $\alpha^i_{gpop} < H\alpha$, then the woodpecker has heard the sound of the gpop drums with acceptable sound intensity. Therefore, it is in an appropriate position. In this situation, the woodpecker escapes to the gpop. This escape to gpop is called GRA (Gpop Running Away). The GRA rate is controlled by the RA probability (P_{GRA}).The values of P_{GRA} decrease adaptively from γ to 0 in the algorithm iteration cycle through Equation (2.12).

$$P_{\mathrm{GRA}} = \gamma * \left(1 - \frac{t}{t_{max}} \right) \tag{2.12}$$

Here t is the current iteration number. Moreover, t_{max} shows the maximum number of iterations, when γ indicate is RA coefficient.
In the proposed algorithm, GRA is performed with two parents. $G\mathrm{RA}_{\mathrm{bit}}$ is a vector as long as the problem dimensions, the elements of which are obtained through Equation (2.13).

$$GRA_{bit} = \begin{cases} 1 \;\; if\ r \ \leq \ P_{GRA} \\ 0 \qquad\qquad else \end{cases} \tag{2.13}$$

In this equation, r is the jth element of the vector (as long as the problem dimensions). Every element is a random number of uniform distribution, selected from [0, 1]. In the WMA algorithm, GRA is done according to Equation (2.14).
In the GRA operator, the female woodpecker escapes to the gpop. Another influential factor here is the location of the random woodpecker ($\mathrm{x_r}$). This woodpecker is randomly selected from the woodpecker population, the aim of which is to randomize the escape of the female woodpecker. In

fact, the female woodpecker escapes to the best male (gpop) and the position of a random woodpecker. The new position of the female woodpecker sits at any random point between the gpop positions and the random woodpecker.

$$x^i_{GRA} = x^t_i + GRA_{bit} * \left\{ \left(x^t_{gpop} - x_r \right) * R \right\} \tag{2.14}$$

In this equation, x^i_{GRA} is the position of a new element obtained from RA on the ith woodpecker, and x_r is the position of a random woodpecker. Moreover, $\boldsymbol{x^t_{gpop}}$ shows the position of the best member (*gpop, the best male*) in t iteration. $\boldsymbol{x^t_i}$ is the new position of the ith woodpecker in iteration t, It was obtained by moving in the search space in the previous step. R is a random number of a uniformly distribution selected from [-1, 1]. t is the current iteration number. Every element of GRA_{bit} vector with a value of one changes in the second expression of Equation (2.14).
At the end of this operator (Running away function), the fitness of x^i_{GRA} or x^i_{RRA} is calculated by using the objective function. If the fitness value is better than $\mathrm{x^t_i}$, then it is replaces. Otherwise, the element that were generated by the RA operator will be ignored.
Given the $H\alpha$ value and the values larger than α in the initial iterations in the WMA algorithm life cycle, the RRA occurrence rate is higher in these iterations. Since in the RRA, female woodpeckers fly quite randomly around the search space, it results in exploration. As the algorithm cycle is repeated more and more, due to the approach of female woodpeckers to the gpop, as well as the declining rate of α, the RRA rate decreases, and the GRA rate increases. At GRA operator, the random movement of female woodpeckers around the gpop and another random woodpecker also increases exploration. Although this random motion decreases with decreasing P_{GRA}during the iteration cycle of the algorithm, it can still avoid stagnation in local optimality in the end iterations. By reducing P_{GRA}, due to the need to exploitation on the end iterations, the female woodpecker becomes more concentrated on its position, leading to exploitation. In summary, the Running Away function has a great effect on the implementation of exploration in the initial iterations and in preventing recession in the local optimal.

2.3.5. Evaluating the new position and checking terminating conditions

In this step, the new position of the ith woodpecker is compared with the previous position and the position of the best woodpecker. If the position is better than one, it will get replaced. If the termination condition of the algorithm is met, the best solution will be selected as the optimal solution to the problem. Otherwise, steps 3-5 are repeated.

3. Evaluation WMA algorithm by Benchmark Functions

In this section, the efficiency of the proposed algorithm is evaluated on 19 mathematical benchmark functions, divided into four groups: unimodal (Table 1), multimodal (Table 2), and composite (Table 3) functions. The unimodal functions have only one optimum. They are appropriate for evaluating the exploitation and convergence of the algorithm. However, multimodal functions have more than one optimum. They are more challenging than unimodal functions. One of the optimums is global, and the rest of them are local optimums. Every algorithm should avoid all local optimums to approach and identify a global optimum. Therefore, test multimodal functions can be used to evaluate the performance of algorithms in exploration and local optimum avoidance. The last class of benchmark functions includes composite functions. In fact, these functions are the combined, shifted, rotated, and biased version of multi-modal and unimodal functions. In these functions, the search space is very complicated because there are many local optimums, and different areas of the search space are in different forms. In such functions, an optimization algorithm should strike an efficient balance

Table 1: Unimodal benchmark functions

Function	Dim	Range	f_{min}
$F_1(X) = \sum_{i=1}^{n} X_i^2$	30	[-100,100]	0
$F_2(X) = \sum_{i=1}^{n} \lvert X_i \rvert + \prod_{i=1}^{n} \lvert X_i \rvert$	30	[-10,10]	0
$F_3(X) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} X_j \right)^2$	30	[-100,100]	0
$F_4(X) = \max\{\lvert X_i \rvert, 1 \le i \le n\}$	30	[-100,100]	0
$F_5(X) = \sum_{i=1}^{n-1} [100(X_{i+1} - X_i^2)^2 + (X_i - 1)^2]$	30	[-30,30]	0
$F_6(X) = \sum_{i=1}^{n} ([X_i + 0.5])^2$	30	[-100,100]	0
$F_7(X) = \sum_{i=1}^{n} iX_i^4 + \mathrm{random}[0,1)$	30	[-1.28,1.28]	0

Table 2: Multimodal benchmark functions

Function	Dim	Range	f_{min}
$F_8(X) = \sum_{i=1}^{n} -X_i \sin\left(\sqrt{\lvert X_i \rvert}\right)$	30	[-500,500]	-418*Dim
$F_9(X) = \sum_{i=1}^{n} [X_i^2 - 10\cos(2\pi X_i) + 10]$	30	[-5.12,5.12]	0
$F_{10}(X) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} X_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi X_i)\right) + 20 + e$	30	[-32,32]	0
$F_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} X_i^2 - \prod_{i=1}^{n}\cos\frac{X_i}{\sqrt{i}} + 1$	30	[-600,600]	0
$F_{12}(X) = \frac{\pi}{n}\left\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\right\} + \sum_{i=1}^{n} u(X_i, 10, 100, 4)$ $y_i = 1 + \frac{X_i + 1}{4}$ $u(X_i, a, k, m) = \begin{cases} k(X_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-X_i - a)^m & x_i < a \end{cases}$	30	[-50,50]	0
$F_{13}(x) = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + \sin^2(2\pi x_n)]\right\} + \sum_{i=1}^{n} u(X_i, 5, 100, 4)$	30	[-50,50]	0

between exploration and exploitation so that it can move towards the global optimum. Thus, the main task of these functions is to evaluate exploration and exploitation simultaneously [4, 46].
The proposed algorithm was compared to a group of new and well-known metaheuristic algorithms such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Firefly Algorithm (FA) [8], Satin Bowerbird Optimizer (SBO) [30], Artificial Bee Colony (ABC) [47], Wolf Optimizer (GWO) [5], Multi-Verse Optimizer (MVO) [3] and Bat Algorithm (BA) [33].
All of the tests were run in MATLAB 2017a. Every benchmark function was run 30 times by the optimization algorithms. The best results of 30 executions were calculated to compare the average (Ave) and standard deviation (Std). Then the Wilcoxon signed rank test was employed to analyze the significance difference between the WMA and other algorithms. The output was the statistical p_value. In this study, the significance level was considered 0.05; therefore, if the results of two algorithms are lower than 0.05, they are statistically different. In all algorithms, the size of the initial population was 50, and the maximum iteration was 500. The dimension of the problem (the number of decision-making variables) was 30. Table 4 shows the other parameters setting for each algorithm.

3.1. Evaluation of exploitation capability (functions F1-F7)

Table 5 shows the results of executing unimodal benchmark functions in optimization algorithms. Accordingly, the WMA algorithm showed a better performance in 6 out of 7 unimodal functions

Table 3: Composite benchmark functions

Function	Dim	Range	f_{min}
$F_{14}(CF_1)$ $f_1, f_2, f_3, \ldots, f_{10} = SphereFunction$ $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [5/100, 5/100, 5/100, \ldots, 5/100]$	10	[-5,5]	0
$F_{15}(CF_2)$ $f_1, f_2, f_3, \ldots, f_{10} = Griewank'sFunction$ $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [5/100, 5/100, 5/100, \ldots, 5/100]$	10	[-5,5]	0
$F_{16}(CF_3)$ $f_1, f_2, f_3, \ldots, f_{10} = Griewank'sFunction$ $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [1, 1, 1, \ldots, 1]$	10	[-5,5]	0
$F_{17}(CF_4)$ $f_1, f_2 = Rastrigin'sFunction, f_3, f_4 = WeierstrassFunction,$ $f_5, f_6 = Griewank'sFunction$ $f_7, f_8 = \text{Ackley's Function},$ $f_9, f_{10} = SphereFunction$ $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}]$=[5/32, 5/32, 1, 1, 5/0.5, 5/0.5, 5/100, 5/100, 5/100, 5/100]	10	[-5,5]	0
$F_{18}(CF_5)$ $f_1, f_2 = Rastrigin'sFunction, f_3, f_4 = WeierstrassFunction,$ $f_5, f_6 = Griewank'sFunction$ $f_7, f_8 = \text{Ackley's Function},$ $f_9, f_{10} = SphereFunction$ $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1, 1, 1, \ldots, 1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}]$=[1/5, 1/5, 5/0.5, 5/0.5, 5/100, 5/100, 5/32, 5/32, 5/100, 5/100]	10	[-5,5]	0
$F_{19}(CF_6)$ $f_1, f_2 = Rastrigin'sFunction, f_3, f_4 = WeierstrassFunction,$ $f_5, f_6 = Griewank'sFunction$ $f_7, f_8 = \text{Ackley's Function},$ $f_9, f_{10} = SphereFunction$ $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}]$=[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1] $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}]$=[0.1 * 1/5, 0.2 * 1/5, 0.3 * 5/0.5, 0.4 * 5/0.5, 0.5 * 5/100, 0.6 * 5/100, 0.7 * 5/32, 0.8 * 5/32, 0.9 * 5/100, 1 * 5/100]	10	[-5,5]	0

Table 4: Algorithm Parameters setting

Algorithm	Parameters
ABC	Limit=100
GA	roulette wheel selection, single point crossover with a crossover probability of 0.9, mutation probability of 0.01
SBO	α=0.94, z=0.02, mutation probability= 0.05
FA	β_0=1, $\alpha \in [0,1]$, γ=1
PSO	c_1=2, c_2=2, ω=1
BA	A=1, r=0.5, Q_{min}=0, Q_{max}=2, α=0.99 , γ=0.01
WMA	Ps=1, $\gamma = 0.2$

Table 5: Results of unimodal benchmark functions

Function		WMA	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F1	Ave	**1.75E-67**	5.43E-06	8.43E-02	5.90E+03	1.61E+04	1.23E+00	2.48E+00	1.07E-27	3.46E+03
	Std	2.35E-67	2.59E-05	2.88E-02	1.01E+03	3.79E+03	3.76E-01	1.16E+00	1.56E-27	9.62E+02
F2	Ave	**1.03E-34**	4.91E-03	1.08E-01	6.42E+01	1.23E+02	3.92E+00	3.43E-01	9.20E-17	8.41E+05
	Std	6.91E-35	5.13E-03	2.47E-02	5.47E+00	9.52E+01	1.63E+01	8.23E-02	7.08E-17	3.08E+06
F3	Ave	**4.26E-59**	6.53E+02	9.57E+02	4.17E+04	2.37E+04	2.07E+02	4.35E+03	5.81E-06	9.89E+03
	Std	1.41E-58	1.12E+03	2.85E+02	4.74E+03	6.98E+03	7.65E+01	1.72E+03	7.35E-06	3.44E+03
F4	Ave	**5.52E-34**	4.17E-01	1.73E+00	7.04E+01	8.55E+06	2.37E+00	7.61E+00	7.27E-07	2.41E+01
	Std	4.72E-34	1.54E-01	1.07E+00	4.81E+00	3.64E+06	1.08E+00	1.33E+00	7.27E-07	5.53E+00
F5	Ave	**3.87E+00**	3.57E+01	2.58E+02	1.56E+07	8.55E+06	3.80E+02	2.73E+02	2.69E+01	5.63E+05
	Std	9.51E+00	2.53E+01	3.50E+02	2.59E+06	3.64E+06	6.25E+02	1.42E+02	8.64E-01	3.16E+05
F6	Ave	5.46E-03	**3.81E-07**	9.73E-02	6.25E+03	1.64E+04	1.36E+00	2.35E+00	7.78E-01	3.42E+03
	Std	3.91E-03	5.71E-07	3.37E-02	9.87E+02	3.33E+03	3.71E-01	8.43E-01	3.33E-01	1.42E+03
F7	Ave	**8.96E-05**	8.90E-02	1.57E-01	7.32E+00	2.52E+00	3.46E-02	1.08E-01	2.06E-03	5.91E-01
	Std	7.58E-05	2.84E-02	3.66E-02	1.65E+00	8.50E-01	1.46E-02	4.19E-02	8.29E-04	3.73E-01

Table 6: P-values obtained from unimodal benchmark functions

Function	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F1	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F2	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F3	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F4	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F5	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	4.45E-05	1.73E-06
F6	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F7	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06

than other algorithms. Given the features of these functions, the results indicate the ability to highly exploit the proposed algorithm. Fig. 4 shows the convergence curve of optimization algorithms in the unimodal functions. According to Fig. 4, the WMA performed the convergence process faster than other algorithms in addition to a higher exploitability. The P-values of the Wilcoxon signed rank test can be seen in Table 6. Accordingly, all of the p-values were smaller than 0.05. Therefore, there was a significant difference between the WMA algorithm and other algorithms.

3.2. Evaluation of exploration capability (functions F8-F13)

Table 7 shows the results of executing 6 multimodal functions in optimization algorithms. Accordingly, in multimodal functions the WMA algorithm showed the best performance on the all of the test cases. Also Table 8 shows a comparison on optimization algorithms in the Wilcoxon signed rank test. Accordingly, the p-value was smaller than 0.05 in all cases except for one function (F_{13} in the PSO). Therefore, there was a significant difference between the WMA and other algorithms. Fig. 5 shows the convergence diagram of optimization algorithms in multimodal multimodal functions respectively . The WMA algorithm converged than other methods because of the high exploration ability and local optimum avoidance as a result of moving towards the global optimum.

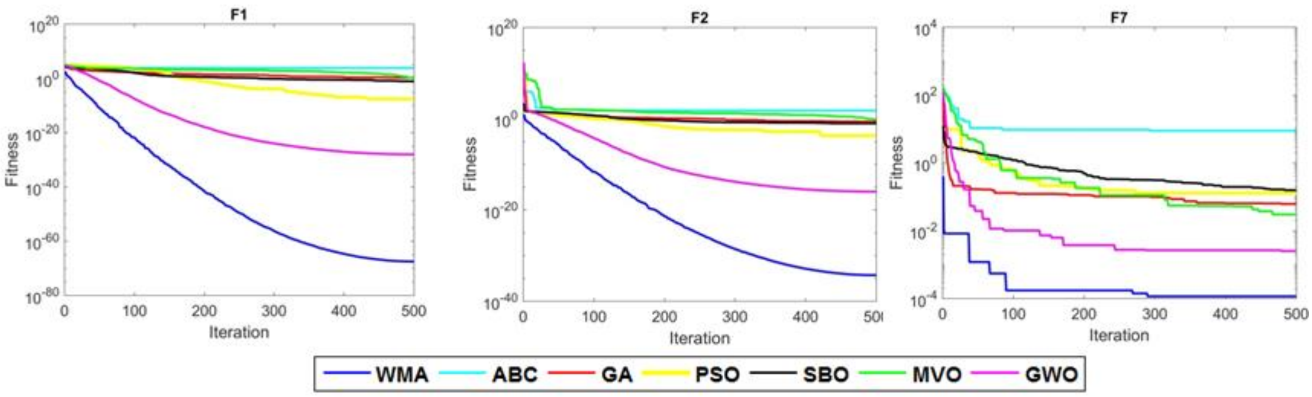

Figure 4: Convergence of algorithms on unimodal benchmark function

Table 7: Results of multimodal benchmark functions

Function		WMA	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F8	Ave	**-1.26E+04**	-2.78E+03	-5.91E+03	-4.42E+03	-7.26E+03	-7.69E+03	-1.10E+04	-5.91E+03	-2.82E+03
	Std	6.75E-03	4.24E+02	9.75E+02	2.52E+02	5.64E+02	8.37E+02	2.99E+02	7.76E+02	7.17E+02
F9	Ave	**0.00E+00**	3.88E+01	5.51E+01	2.73E+02	1.90E+02	1.17E+02	4.61E+00	2.47E+00	1.15E+02
	Std	0.00E+00	1.03E+01	1.33E+01	1.18E+01	3.07E+01	3.10E+01	2.00E+00	4.14E+00	6.24E+01
F10	Ave	**1.01E-15**	7.94E-01	1.55E-01	1.40E+01	1.90E+01	1.85E+00	5.50E-01	1.03E-13	1.06E+01
	Std	6.49E-16	1.26E+00	9.81E-02	5.72E-01	1.24E-01	4.54E-01	1.82E-01	2.25E-14	2.03E+00
F11	Ave	**0.00E+00**	8.33E+01	4.57E-01	5.80E+01	1.54E+02	8.59E-01	9.97E-01	4.26E-03	3.55E+01
	Std	0.00E+00	7.34E+00	2.13E-01	9.43E+00	3.08E+01	6.64E-02	7.41E-02	9.99E-03	7.92E+00
F12	Ave	**9.65E-05**	1.45E-01	1.36E+00	2.65E+07	2.04E+06	2.33E+00	1.83E-01	4.26E-02	5.01E+03
	Std	8.41E-05	1.99E-01	1.95E+00	8.60E+06	2.00E+06	1.29E+00	1.57E-01	2.20E-02	1.69E+04
F13	Ave	**1.12E-03**	5.80E-02	9.23E-03	6.51E+07	2.06E+07	1.81E-01	1.74E-01	6.30E-01	4.07E+05
	Std	8.66E-04	2.31E-01	4.54E-03	2.26E+07	1.30E+07	9.75E-02	6.12E-02	1.92E-01	4.33E+05

Table 8: P-values obtained from multimodal benchmark functions

Function	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F8	1.92E-06	1.73E-06	1.73E-06	3.88E-06	1.13E-05	1.73E-06	1.73E-06	1.70E-06
F9	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.66E-06	1.66E-06	1.73E-06
F10	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.70E-06	1.70E-06	1.73E-06
F11	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.70E-06	1.70E-06	1.73E-06
F12	8.73E-03	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F13	**0.643517**	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06

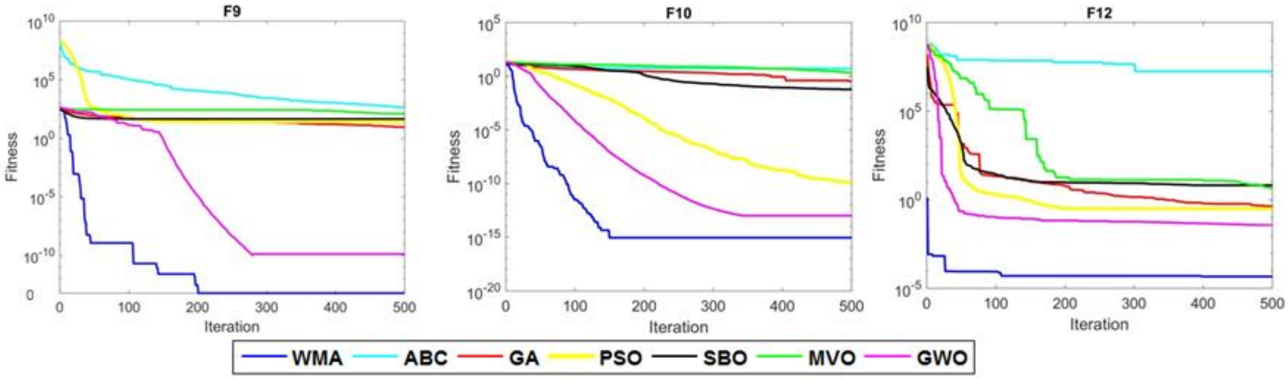

Figure 5: Convergence of algorithms on multimodal benchmark functions

Table 9: Results of composite benchmark functions

Function		WMA	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F14	Ave	**40**	80	120.0005	74.15591	151.3277	60.00288	237.7551	70.00343	151.3277
	Std	51.63978	91.89366	113.5291	22.71768	86.65939	69.91944	75.88531	105.9354	86.65939
F15	Ave	**22.806**	128.5088	160.547	103.5252	171.8722	289.5416	422.9546	119.8556	171.8722
	Std	9.147513	75.34788	119.1571	10.39313	110.6674	140.9827	158.1751	74.58779	110.6674
F16	Ave	**204.1304**	325.1323	596.0808	331.9271	683.9428	253.3962	622.7233	214.815	683.9428
	Std	59.82906	101.982	169.3054	34.43387	110.7461	118.8489	159.8559	65.25627	110.7461
F17	Ave	**347.2332**	479.8703	724.4624	381.5485	854.7304	372.8359	756.4808	374.6482	854.7304
	Std	42.62632	119.6874	146.7025	14.57331	141.4829	123.1018	145.3013	128.0835	141.4829
F18	Ave	**25.00393**	222.8516	197.5817	84.31209	664.9722	42.23124	434.6995	166.9422	664.9722
	Std	40.42662	204.4668	132.212	15.56667	240.7158	49.82718	161.7466	153.3033	240.7158
F19	Ave	692.0237	741.7132	784.2334	**533.3116**	850.0618	772.2147	876.4621	854.1923	850.0618
	Std	203.4475	207.741	190.4928	15.97211	138.4712	188.7851	108.2746	126.6481	138.4712

3.3. Ability to escape from local minima (functions F14–F19)

In composite functions, an optimization algorithm should strike an efficient balance between exploration and exploitation allows local optima to be avoided. Table 9 shows the results of executing 6 composite functions in optimization algorithms. Accordingly, in this functions the WMA algorithm showed the best performance on the majority of the test cases. Also Table 10 shows a comparison on optimization algorithms in the Wilcoxon signed rank test. Accordingly, the p-value was smaller than 0.05 in most cases except for F16. Therefore, there was a significant difference between the WMA algorithm and other algorithms. The large discrepancy of results comes from the high complexity of the search space in the evaluated composite functions. Fig. 6 shows the convergence diagram of optimization algorithms. The WMA algorithm converged than other methods as discussed earlier, exploration and exploitation phases were combined in composite functions. According to the results in Fig. 6 and Table 9, the proposed algorithm struck an efficient balance between these two phases to address the various differences of such a complicated search space.

3.4. Optimization of large-scale problems using WMA

To further demonstrate the capabilities of WMA and the importance of applying this algorithm to high-dimensional, real-world problems, the 1000 dimensional versions of the unimodal and multimodal benchmark functions (F1...F13) were used. In this test, the population size and number

Table 10: P-values obtained from composite functions

Function	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F14	0.009766	0.042128	0.001953	0.001953	0.001953	0.019531	0.019531	0.001953
F15	0.009766	0.001953	0.001953	0.001953	0.001953	0.019531	0.019531	0.001953
F16	0.009766	0.005859	0.001953	0.001953	0.001953	0.019531	**0.769531**	0.001953
F17	0.037109	0.042128	0.001953	0.001953	0.001953	0.019531	0.019531	0.001953
F18	0.039063	0.009766	0.001953	0.001953	0.001953	0.019531	0.019531	0.001953
F19	0.009766	0.005859	0.001953	0.001953	0.001953	0.019531	0.019531	0.001953

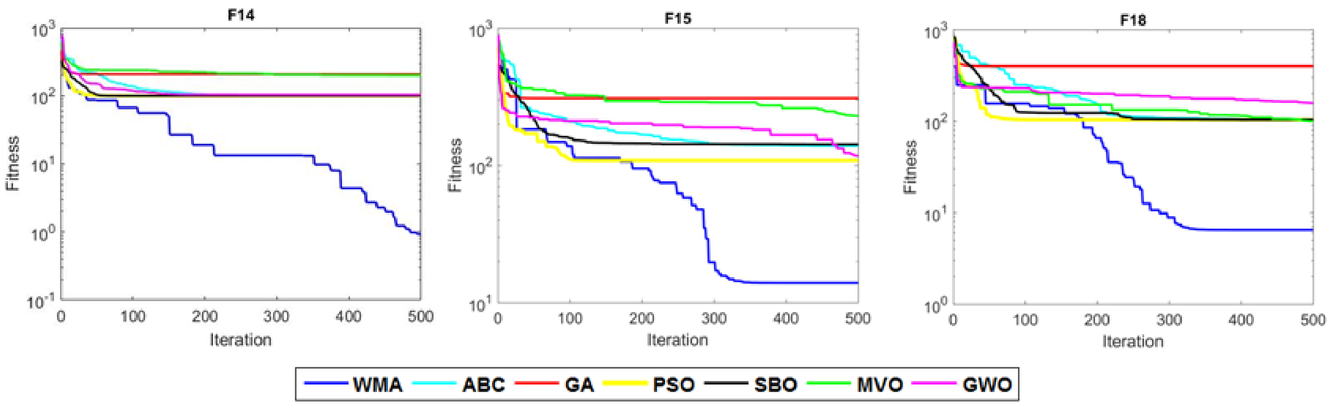

Figure 6: Convergence of algorithms on composite benchmark functions

of iterations were considered 30 and 500, respectively. The statistical results were averaged for 30 independent runs. Details of the this test problems are reported in Tables 11. The results of the proposed algorithm for high dimensions of the multimodal and unimodal functions (F1...F13) are presented in Table 11 As shown, the results are highly promising and effective in high dimensions, which is maintained as the number of problem variables (dimension) is increased. The results of 1000 dimensional search space are given in Table 11 As shown, the WMA algorithm by far outperforms the other assessed algorithms offering the best results for all F1 to F13 functions.. It should be noted that the proposed algorithm was able to obtain the global optimum for functions F9 and F11 in 30 and 1000 dimensions.

4. Conclusion

In this study, a new metaheuristic algorithm was inspired by the intelligent behavior of woodpeckers in the mating process called WMA. In the WMA algorithm, the received drum sound intensity was defined as the attractiveness for moving in another direction. The sound intensity was inspired by the concepts of sound waves physics. The proposed algorithm was equipped with multiple operators to explore and exploit the search space. In the WMA, the population is divided into male and female woodpeckers. The female woodpeckers update their positions based on the position of the best and closest male bird. In the initial iterations, this method increases exploration and diversity. As the final iterations come along, the adaptive decrease in the number of male woodpeckers increases the exploitation around the resultant solutions. At the same time, decreasing the magnitudes of steps adaptively during the algorithm iteration cycle will directly affect exploration and exploitation as well as the transfer between these two phases.

Table 11: Results of benchmark functions (F1-F13), with 1000 dimensions

Function		WMA	PSO	SBO	ABC	FA	MVO	GA	GWO	BA
F1	Ave	**1.59E-63**	3.87E+03	2.27E+05	3.05E+06	2.75E+07	7.98E+05	1.55E+05	2.59E-01	1.94E+05
	Std	2.31E-63	6.35E+02	2.38E+04	3.24E+04	2.04E+04	3.10E+04	1.30E+04	8.11E-02	7.61E+03
F2	Ave	**1.14E-32**	1.56E+03	1.63E+10	5.98E+08	2.14E+10	5.62E+07	2.04E+03	7.81E-01	2.06E+03
	Std	6.39E-33	8.92E+02	8.90E+09	2.15E+07	8.56E+09	3.62E+06	6.53E+02	4.64E-01	4.70E+02
F3	Ave	**6.94E-50**	2.24E+06	1.03E+07	4.82E+07	3.54E+07	8.01E+06	2.38E+06	1.51E+06	2.49E+06
	Std	2.23E-49	1.18E+06	2.64E+06	5.12E+06	6.23E+06	8.15E+05	4.96E+05	3.49E+05	2.05E+05
F4	Ave	**7.07E-32**	1.60E+01	4.53E+01	9.94E+01	9.45E+01	9.77E+01	5.01E+01	7.86E+01	5.24E+01
	Std	5.64E-32	7.76E-01	3.62E+00	9.05E-02	3.26E-01	5.73E-01	4.44E+00	3.15E+00	8.28E-01
F5	Ave	**3.64E+00**	3.42E+05	1.08E+08	1.42E+10	1.12E+10	2.33E+09	5.44E+07	1.03E+03	8.67E+07
	Std	4.34E+00	6.39E+04	1.93E+07	2.61E+08	2.42E+08	1.52E+08	8.91E+06	2.15E+00	5.77E+06
F6	Ave	**1.89E-01**	3.89E+03	1.08E+08	3.05E+06	2.73E+06	7.92E+05	1.50E+05	2.03E+02	1.93E+05
	Std	2.77E-01	6.75E+02	1.93E+07	3.87E+04	6.13E+04	2.98E+04	1.02E+04	2.70E+00	7.80E+03
F7	Ave	**1.80E-04**	1.76E+04	2.86E+02	2.32E+05	1.46E+05	2.96E+04	1.53E+05	1.51E-01	1.13E+03
	Std	1.01E-04	1.67E+03	4.08E+01	4.21E+03	7.07E+03	2.02E+03	1.26E+04	3.32E-02	9.33E+01
F8	Ave	**-4.19E+05**	-1.60E+04	-1.60E+04	-2.59E+04	-9.99E+04	-1.10E+05	-1.26E+05	-8.07E+04	-1.60E+05
	Std	4.33E-01	2.30E+03	3.71E+03	2.01E+03	2.01E+03	4.10E+03	7.88E+03	2.45E+04	3.55E+04
F9	Ave	**0.00E+00**	6.26E+03	6.47E+03	1.73E+04	1.22E+04	1.45E+04	5.69E+03	1.89E+02	5.66E+03
	Std	0.00E+00	2.94E+02	2.34E+02	1.59E+02	1.52E+02	2.27E+02	1.14E+02	4.42E+01	1.14E+02
F10	Ave	**1.95E-15**	8.43E+00	1.36E+01	2.11E+01	2.03E+01	2.10E+01	1.26E+01	1.76E-02	1.82E+01
	Std	1.66E-15	2.76E-01	3.06E-01	7.47E-03	2.21E-02	2.48E-02	2.89E-01	2.73E-03	2.07E-01
F11	Ave	**0.00E+00**	3.60E+03	2.06E+03	2.74E+04	2.45E+04	7.22E+03	1.34E+03	6.86E-02	1.74E+03
	Std	0.00E+00	6.70E+01	2.15E+02	3.40E+02	3.73E+02	3.00E+02	1.15E+02	8.54E-02	3.80E+01
F12	Ave	**6.69E-06**	6.62E+00	1.76E+07	3.50E+10	2.44E+10	4.20E+09	1.46E+06	1.24E+00	4.41E+06
	Std	6.88E-06	9.01E-01	7.82E+06	7.57E+08	8.99E+08	4.10E+08	6.99E+05	2.75E-01	7.28E+05
F13	Ave	**6.86E-03**	2.41E+03	1.56E+08	6.43E+10	4.85E+10	9.21E+09	5.05E+07	1.21E+02	8.80E+07
	Std	1.61E-02	6.48E+02	3.65E+07	1.53E+09	9.25E+08	6.50E+08	1.34E+07	8.25E+00	1.29E+07

The efficiency of the proposed algorithm was evaluated using 19 mathematical benchmark functions (7 unimodal, 6 multimodal and 6 composite). The WMA algorithm was compared with a group of new and well-known metaheuristic algorithms. The proposed algorithm could obtain better results in 17 benchmark functions. The proposed algorithm produced significant results in the non-convex, inseparable, and scalable benchmark functions. In addition, the convergence diagrams showed a higher convergence speed. The simulation results indicated the high abilities of the proposed algorithm to explore and exploit the search space. It can also bring about convergence quickly. As a suggestion for future studies, the WMA algorithm can be used to solve the real world problems in different engineering fields. It can also be combined with other algorithms to improve the performance of the WMA.

5. Reference

[1] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," Knowledge-Based Systems, vol. 96, pp. 120-133, 2016.
[2] S. Mirjalili and A. Lewis, "The whale optimization algorithm," Advances in Engineering Software, vol. 95, pp. 51-67, 2016.
[3] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," Neural Computing and Applications, vol. 27, pp. 495-513, 2016.
[4] S. Mirjalili, "The ant lion optimizer," Advances in Engineering Software, vol. 83, pp. 80-98, 2015.
[5] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in Engineering Software, vol. 69, pp. 46-61, 2014.
[6] V. Punnathanam and P. Kotecha, "Yin-Yang-pair Optimization: A novel lightweight optimization algorithm," Engineering Applications of Artificial Intelligence, vol. 54, pp. 62-79, 2016.
[7] A. Askarzadeh, "Bird mating optimizer: an optimization algorithm inspired by bird mating strategies," Communications in Nonlinear Science and Numerical Simulation, vol. 19, pp. 1213-1228, 2014.
[8] X.-S. Yang, Engineering optimization: an introduction with metaheuristic applications. New Jersey: John Wiley & Sons, 2010.
[9] H. R. Maier, S. Razavi, Z. Kapelan, L. S. Matott, J. Kasprzyk, and B. A. Tolson, "Introductory overview: Optimization using evolutionary algorithms and other metaheuristics," Environmental modelling & software, 2018.
[10] W. Qian, J. Chai, Z. Xu, and Z. Zhang, "Differential evolution algorithm with multiple mutation strategies based on roulette wheel selection," Applied Intelligence, pp. 1-18, 2018.
[11] S. Behera, S. Sahoo, and B. Pati, "A review on optimization algorithms and application to wind energy integration to grid," Renewable and Sustainable Energy Reviews, vol. 48, pp. 214-227, 2015.
[12] G. Erbeyoğlu and Ü. Bilge, "PSO-based and SA-based metaheuristics for bilinear programming problems: an application to the pooling problem," Journal of Heuristics, vol. 22, pp. 147-179, 2016.
[13] A. Kaveh and A. Dadras, "A novel meta-heuristic optimization algorithm: Thermal exchange optimization," Advances in Engineering Software, 2017.
[14] M. Dehghani, Z. Montazeri, O. P. Malik, A. Ehsanifar, and A. Dehghani, "OSA: Orientation Search Algorithm," International Journal of Industrial Electronics, Control and Optimization, vol. 2, pp. 99-112, 2019.
[15] T. Zhang and Z. W. Geem, "Review of harmony search with respect to algorithm structure," Swarm and Evolutionary Computation, vol. 48, pp. 31-43, 2019.
[16] A. R. Jordehi, "Brainstorm optimisation algorithm (BSOA): An efficient algorithm for finding optimal location and setting of FACTS devices in electric power systems," International Journal of Electrical Power & Energy Systems, vol. 69, pp. 48-57, 2015.

[17] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition," in Evolutionary computation, 2007. CEC 2007. IEEE Congress on, 2007, pp. 4661-4667.
[18] N. S. Jaddi, J. Alvankarian, and S. Abdullah, "Kidney-inspired algorithm for optimization problems," Communications in Nonlinear Science and Numerical Simulation, vol. 42, pp. 358-369, 2017.
[19] A. H. Gandomi, "Interior search algorithm (ISA): a novel approach for global optimization," ISA transactions, vol. 53, pp. 1168-1183, 2014.
[20] A. Baykasoğlu, A. Hamzadayi, and S. Akpinar, "Single Seekers Society (SSS): Bringing together heuristic optimization algorithms for solving complex problems," Knowledge-Based Systems, vol. 165, pp. 53-76, 2019.
[21] M. Issa, A. E. Hassanien, D. Oliva, A. Helmi, I. Ziedan, and A. Alzohairy, "ASCA-PSO: Adaptive sine cosine optimization algorithm integrated with particle swarm for pairwise local sequence alignment," Expert Systems with Applications, vol. 99, pp. 56-70, 2018.
[22] S. H. S. Moosavi and V. K. Bardsiri, "Poor and rich optimization algorithm: A new human-based and multi populations algorithm," Engineering Applications of Artificial Intelligence, vol. 86, pp. 165-181, 2019.
[23] Y. Zhang and Z. Jin, "Group teaching optimization algorithm: A novel metaheuristic method for solving global optimization problems," Expert Systems with Applications, p. 113246, 2020.
[24] R. V. Rao, V. J. Savsani, and D. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," Computer-Aided Design, vol. 43, pp. 303-315, 2011.
[25] I. BoussaïD, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," Information Sciences, vol. 237, pp. 82-117, 2013.
[26] N. A. Al-Thanoon, O. S. Qasim, and Z. Y. Algamal, "A new hybrid firefly algorithm and particle swarm optimization for tuning parameter estimation in penalized support vector machine with application in chemometrics," Chemometrics and Intelligent Laboratory Systems, vol. 184, pp. 142-152, 2019.
[27] J. Farzaneh, R. Keypour, and A. Karsaz, "A novel fast maximum power point tracking for a PV system using hybrid PSO-ANFIS algorithm under partial shading conditions," International Journal of Industrial Electronics, Control and Optimization, vol. 2, pp. 47-58, 2019.
[28] J. Ji, S. Song, C. Tang, S. Gao, Z. Tang, and Y. Todo, "An artificial bee colony algorithm search guided by scale-free networks," Information Sciences, vol. 473, pp. 142-165, 2019.
[29] S. Bromuri, "Dynamic heuristic acceleration of linearly approximated SARSA ($$\lambda $$): using ant colony optimization to learn heuristics dynamically," Journal of Heuristics, pp. 1-32, 2019.
[30] S. H. S. Moosavi and V. K. Bardsiri, "Satin bowerbird optimizer: A new optimization algorithm to optimize ANFIS for software development effort estimation," Engineering Applications of Artificial Intelligence, vol. 60, pp. 1-15, 2017.
[31] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," Neural Computing and Applications, vol. 27, pp. 1053-1073, 2016.
[32] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," Knowledge-Based Systems, vol. 89, pp. 228-249, 2015.
[33] X.-S. Yang and X. He, "Bat algorithm: literature review and applications," International Journal of Bio-Inspired Computation, vol. 5, pp. 141-149, 2013.
[34] R. Moazenzadeh and B. Mohammadi, "Assessment of bio-inspired metaheuristic optimisation algorithms for estimating soil temperature," Geoderma, vol. 353, pp. 152-171, 2019.
[35] M. Bertolini, D. Mezzogori, and F. Zammori, "Comparison of new metaheuristics, for the solu-

tion of an integrated jobs-maintenance scheduling problem," Expert Systems with Applications, vol. 122, pp. 118-136, 2019.
[36] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," Future Generation Computer Systems, vol. 97, pp. 849-872, 2019.
[37] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm," Computers & Structures, vol. 169, pp. 1-12, 2016.
[38] H. Barati and M. Shahsavari, "Simultaneous Optimal placement and sizing of distributed generation resources and shunt capacitors in radial distribution systems using Crow Search Algorithm," International Journal of Industrial Electronics, Control and Optimization, vol. 1, pp. 27-40, 2018.
[39] A. S. Shamsaldin, T. A. Rashid, R. A. A.-R. Agha, N. K. Al-Salihi, and M. Mohammadi, "Donkey and Smuggler Optimization Algorithm: A Collaborative Working Approach to Path Finding," Journal of Computational Design and Engineering, 2019.
[40] J. M. Czerniak and H. Zarzycki, "Artificial acari optimization as a new strategy for global optimization of multimodal functions," Journal of computational science, vol. 22, pp. 209-227, 2017.
[41] L. Lin and M. Gen, "Auto-tuning strategy for evolutionary algorithms: balancing between exploration and exploitation," Soft Computing-A Fusion of Foundations, Methodologies and Applications, vol. 13, pp. 157-168, 2009.
[42] B. Doğan and T. Ölmez, "A new metaheuristic for numerical function optimization: Vortex Search algorithm," Information Sciences, vol. 293, pp. 125-145, 2015.
[43] D. J. Dodenhoff, "An analysis of acoustic communication within the social system of downy woodpeckers (Picoides pubescens)," PhD dissertation PhD dissertation, The Ohio State University, 2002.
[44] H. D. Wilkins and G. Ritchison, "Drumming and Tapping by Red-Bellied Woodpeckers: Description and Possible Causation (Descripción y Posible Causa del Tamborileo y Picoteo Lento en Melanerpes carolinus)," Journal of Field Ornithology, pp. 578-586, 1999.
[45] D. Halliday, R. Resnick, and J. Walker, Fundamentals of Physics Extended, 10th Edition: Wiley, 2013.
[46] J.-J. Liang, P. N. Suganthan, and K. Deb, "Novel composition test functions for numerical global optimization," in Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005. SIS 2005., 2005, pp. 68-75.
[47] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," Journal of global optimization, vol. 39, pp. 459-471, 2007.