Chapter

Bio-inspired Optimization: Algorithm, Analysis and Scope of Application

G. Devika and Asha Gowda Karegowda

Abstract

In the last few years, bio-inspired optimization techniques have been widely adopted in fields such as computer science, mathematics, and biology in order to optimize solutions. Bio inspired optimization problems are usually nonlinear and restricted to multiple nonlinear constraints to tackle the problems of the traditional optimization algorithms, the recent trends tend to apply bio-inspired optimization algorithms which represent a promising approach for solving complex optimization problems. This work comprises state-of-art of ten recent bio-inspired algorithms, gap analysis, and its applications namely; Particle swarm optimization (PSO), Genetic Bee Colony (GBC) Algorithm, Fish Swarm Algorithm (FSA), Cat Swarm Optimization (CSO), Whale Optimization Algorithm (WOA), Artificial Algae Algorithm (AAA), Elephant Search Algorithm (ESA), Cuckoo Search Optimization Algorithm (CSOA), Moth flame optimization (MFO), and Grey Wolf Optimization (GWO) algorithm. The previous related works collected from Scopus databases are presented. Also, we explore some key issues in optimization and some applications for further research. We also analyze in-depth discussions on the essence of these algorithms and their connections to self-organization and their applications in different areas of research are presented. As a result, the proposed analysis of these algorithms leads to some key problems that have to be addressed in the future.

Keywords: particle swarm optimization, genetic bee colony algorithms, Fish swarm algorithm, artificial algae algorithm, Chicken swarm optimization, Grey wolf algorithm, Cat swarm optimization

1. Introduction

Bio-inspired algorithms nowadays resolve application problems in decision-making, information handling, and optimization purposes from different domains of science and engineering. Many techniques developed fields expected to next few years intelligent optimization algorithms more effective in solving different problems for anomaly and failure detection areas [1]. Optimization plays a major role in more single or multi-objective problems with deterministic or stochastic algorithms [2]. The focus of NP-hard problem-based deterministic or stochastic algorithms to intensification and diversification of meta-heuristic optimization algorithm. Compared to conventional methods, bio-inspired algorithms are intelligent, improved, easy to test, and flexible [3].

In computer networks, security, mechanical problems, electronics image processing, electrical, robotics, production engineering, management, planetary and others are applying bio-inspired algorithms in new era to solve problems easily [4, 5]. Hence it is an emerging field, authors aim to review the discussion and future scope of bio-inspired algorithms. Bio-inspired algorithms concern definitions, principles models, processing steps, merits and demerits reviewed for the most frequently applied bio-inspired algorithms in this chapter. The study discusses bio-inspired algorithms which are purely inspired by identifiable or special behaviour of biological organisms. This chapter covers both emerging and well-known techniques. Ten bio-inspired algorithms: Particle swarm optimization (PSO), Genetic Bee Colony (GBC) Algorithm, Fish Swarm Algorithm (FSA), Cat Swarm Optimization (CSO), Whale Optimization Algorithm (WOA), Artificial Algae Algorithm (AAA), Elephant Search Algorithm (ESA), cuckoo Search Optimization Algorithm (CSOA), Moth flame optimization (MFO), and Grey Wolf Optimization (GWO) algorithm are analysed deeply in this work along with their future scope. Authors have restricted to ten potential algorithms few more potential bio-inspired algorithms is dealt in detail for authors other publications [6, 7]. The work carried on in two phases, in initial phase aims in recognizing algorithms and second phase in depth study of identified algorithms is performed. The chapter noticeably aid in identification of significant bio-inspired solutions for various problems. In section 1, overview of optimization technique and types are presented. Section 2 covers core part of authors work which gives in-depth information on ten bio-inspired algorithms. Section 3 focus on current observation of algorithms and in next section further scope and conclusion are briefed.

2. Overview of optimization

Optimization methods execute and compare iteratively to find solutions for the optimum solutions to be searched. Optimization is part of all problems in all fields. Common types of optimization methods adopted to find solutions are briefed.

2.1 Stochastic optimization

Stochastic optimization (SO) computation involves more vagueness and impreciseness because of randomness in function of minimization or maximization to lend for real-life scenarios. The involved unpredictability exists in form of noise in process of search by Monte Carlo randomness [8]. Stochastic annealing, approximation, programming, swarm-based algorithms are common involved techniques of SO. They include high non-linearity system noise and dimensional models. These models are present to analyze, solve, derive, and numerical extraction of information in resolving decision-making problems. Major investment of SO is in specific applications oriented towards long and short programs. Aircrafts, missile, drug design, and network traffic control applications are getting advantage of SO. Stochastic application tool can be applied as a powerful modelling tool in a few applications, but estimation of real-life problems is another major uncertainity where solving through SO involves practical limitations. Another problem of SO is complete dependency on data available and modelling of it [3, 9].

2.2 Robust optimization

The optimization model is robust based to deal with data to regulate uncertainty. Key features are deterministic, easy computational tractability and set based. Model includes global or local or non- probabilistic or probabilistic models. Any given problem will get involve all the features of robust optimization in order to search for a solution. The technique is also known as the min-max or worst-case approach. Provide a guarantee for solutions to problem application which involves more uncertainty in data. The parameters involved in process of estimation are to resolve estimation errors. One improved model for definition and interpretation is setting more robust constraints [10]. Engineering optimization design results mainly in reliability optimization and feasible input possible values to robust solution structure. Robust optimization gives same weight and values for parametric values in collection of uncertain data. Problems will be resolved with the formulation of cost savings and stability, qualitative and quantitative. Complex problems considered for optimization may extend complexity to a more significant level [6, 7].

2.3 Dynamic optimization

Dynamic programming is another name for dynamic optimization which processes optimal profile of more than one parameter of a system used to find possible solutions for a problem given. Variations of dynamic optimization with optimization discrete time, calculus variation and extended static optimization. The implementation includes a system controller to perform criteria with algorithm to execute control. Dynamic optimization involves a system controller performing optimal substructure and overlapping sub-problems [8]. Dynamic optimization characterizes structure, recursively defines value, computes value and constructs an optimal solution for computation. Dynamic programming optimizes problems and recursively divide problem into sub-problems which can solve either bottom-up or top-down approach. The logic used is general and supple. It solves computation time and storage space [9]. Classification optimization based on different factors is summarized in **Table 1**.

3. Bio inspired optimization algorithms

This section is brief on bio-inspired algorithms detailed. Concept advantage algorithm, flowchart and applications are briefed.

3.1 Particle swarm optimization (PSO)

In proposed particle swarm optimization (PSO) algorithm inspired by intelligent behaviour of birds [11], Craig Reynolds simulated flock social birds behaviour for the first time and later studied by Frank Heppner [12]. PSO search for optimal solution similar to flyting birds with specific velocities determined from previous results and

	Optimization	Factor	Taxonomy
1	Stochastic	Constraints	Unconstrained Constrained
		Nature of equation	Non-Linear Polynomial Linear Quadratic
2	Robust	Physical structure	Optimal control Non-optimal control
		Decision variable permissible value	Integer programming Real-valued programming
3	Dynamic	Variable type	Deterministic Non-deterministic
		Function splitting	Separable Non-separable
		Objective function	Single objective Multi-objective

Table 1.
Classification of optimization.

neighbours in identified search areas [13]. Given a problem identified in search space represent solution in different n-dimension as result in PSO as n particles. The particle moves in n dimension solution space with different velocities. Particles move and store previous behaviours of it and share experiences to store search space. The key merit of POS is its experience to share particle communicate to part or complete swarm to lead motion to detect search space [14]. Each particle will compare the current fitness value with previous optimized results and neighbours in every iteration. Entire particles' global and local algorithm is considered. Each particle best global value is stored as a local value. The entire search space particles' best result is stored as the global best optimal solution. In further iterations value will be adjusted to best optimal if current is best when compared to previous results.

3.1.1 PSO concept

Each particle is running in PSO to identify a feasible solution to the optimization problem in a given search space. The behaviour of flight of particles is considered as search of individual particle. Velocity of particles is dynamically updated based on position of particle and optimal swarm population. The swarm population is composed of M particles in D dimensional space and historical optimal position of the ith particle is represented by pi, iϵ {1,2,3 … M} and optimal position of swarm population is denoted by p_g. In every step velocity and position of each particle are updated dynamically tracking its corresponding previous positions and optimal position of swarm population. The detailed equations are expressed as follows,

$$V_{i_j}^{t+1} = wV_{i_j}^{t+1} + c_1 r_1^t \left(pbest_{i_j} - X_{i_j}^t\right) + c_2 r_2^t \left(gbest_j - X_{i_j}^t\right) \tag{1}$$

$$X_{i_j}^{t+1} = X_{i_j}^t + V_{i_j}^{t+1} \tag{2}$$

In Eqs. (1) and (2) t indicates iteration number , d ϵ {1, 2, 3, … .. D} indicates dimension, $xi,d(t)$ is the *dth* dimension variable of the *ith* particle in the *tth* iteration, and variables $vi,d(t)$, $pg,d(t)$, and $pi,d(t)$ have the similar meanings in turn. ? is inertial weight, and *c1* and *c2* denote acceleration coefficients, *r1* and *r2* are random numbers uniformly distributed in interval [0, 1]. The objective function to be set, and resultant objective values of each particle correspond to fitness values. These fitness values are used to measure position of particles, historical optimal position of particles and the optimal position of swarm population.

The main concept of PSO is clear from the particle velocity equation that a constant balance between three distinct forces hauling on each particle: (i) particles previous velocity (inertia), (ii) Distance from the individual particles' best-known position (cognitive force) and (iii) Distance from the swarms best known position (social force). These forces are dependent on c1 and c2 weight constants and randomly concerned by r1 and r2 constants. Three forces are shown in vector form as in **Figure 1a** where weight values are specified in vector magnitude. The particles will continue to explore as in search space similar to bird as shown in **Figure 1b** to converge to best position.

PSO shows sufficient better performance on optimization related problems of small scale. The original POS later on improved versions of PSO have been proposed by many researchers. Few incremental works of PSO has been discussed in this sub section which support for large scale and multiple optima [14].

Opposition based PSO discussed by Jabeen et al. [11]. Particle has been classified into two class bad and good. Population of two class generated with fitness computation then original PSO applied. The opposite particle computed using equation

$$Popi = a + b - pi \tag{3}$$

Where in Eq. (3) D is the dimension and R is real number. Quasi-oppositional comprehensive learning particle swarm optimizers (QCLPSO) proposed Chang et al. [7]. Swarm initialization applied by qausi opposite number. The constriction factor balance incremental approach to proposed by Clerc [8]. The equation with constriction factor velocity updated equation is summarized in Eqs. (4)–(6),

$$v_{ij}(t+1) = x\left[v_{ij}(t+1)v_{ij}(t) + \varphi\left(v_{ij}(t)\right) - x_{ij}(t) + \varphi\left(y_{ij}(t)\right) - x_{ij}(t)\right] \tag{4}$$

$$x = 2/|\, 4 - \varphi - \sqrt{2\varphi - 4\varphi}| \tag{5}$$

$$\varphi = c1 + c2, \varphi 1 = c1r1 \tag{6}$$

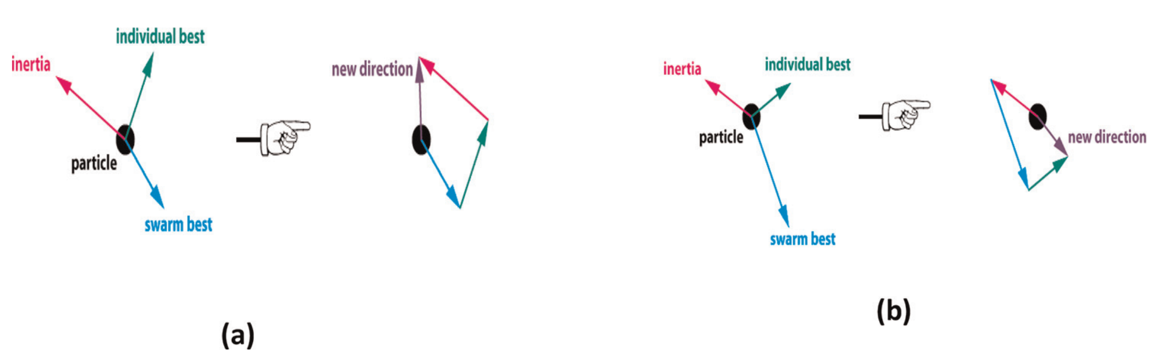


Figure 1.
(a) Exploration of PSO (b) search of new position [4].

```
Initialize the population randomly
 While (Population Size)
{
Calculate fitness If fitness value is better from the best fitness value (pbest) in history then
Update pbest with the new pbest
Select the particle with the best fitness value from all particles as gbest
While maximum iterations or minimum error criteria is not attained
{
 For each particle Calculate particle velocity by equation (1)
Update particle position according to equation (2)
}
 }
```

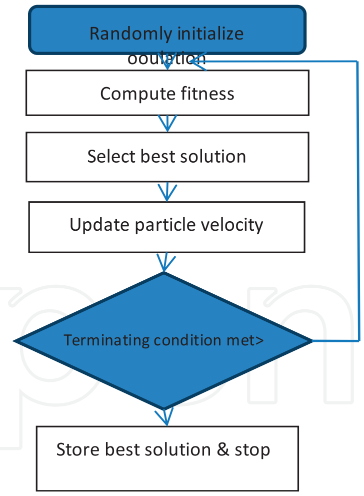


Figure 2.
PSO algorithm & flowchart.

A random value is distributed between [0, 1] for particles by Zhang et al. [13]. The dependency inertia weight to maximize number of iterations to another one is applied to avoid problems in original PSO to search local ability to end. Speed- and accumulation-based inertia weight computation is proposed in Wei et al. [15]. A Cauchy mutation an improved PSO proposed by Wang et al. [14]. The original fitness of particle is selected to mutate the particle to distribute with increase in parameter scale t = 1. It is defined between test function to choose randomly to assign velocity and pop size for particles in swarm. A variation in computation of power distribution among particles applies global best value with power mutation function. The fitness calculation for both particles select appropriate one in Wu et al. [13]. Power mutation function based another opposition-based power mutation function applied for PSO by Imran et al. [15]. Two times mutation being applied on opposite swarms and global best particle power mutation. Global selection for best mutation avoids stagnation. Still improved PSO presented by Imran et al. [16] with student T mutation. Global best particle T student particle identified to work over adaptive and Cauchy mutation (**Figure 2**).

3.1.2 Merits of PSO

- Communication capability: particles can communicate efficiently each other as positions of best particle of all previous iterations are stored.
- Faster Convergence: accelerates more towards optimal solution in optimization
- Simplicity: Updation of velocity and position equations are simple to calculate
- Adaptable to environment: have ability to choose best optimal solutions in changing environment.

3.1.3 Demerits of PSO

- PSO fails to resolve problem which lack in storage and not able to may clear distinction between previous and next particle positions.

- Assume all particles are same hence inertial and velocity also remain same
- PSO do not identify multiple optima.
- Convergence is harder with varied inertia weight

3.1.4 Applications

PSO has been applied in most domains to optimize solutions from agriculture to industry. PSO has been extensively applied in different geotechnical engineering aspects such as slope stability analysis, pile and foundation engineering, rock and soil mechanics, and tunnelling and underground space design [17]. PSO has been widely used in various kinds of planning problems, especially in the area of substation locating and sizing [14]. But in the area of heating supply, PSO is mainly applied in heating load forecasting [18–20], but rarely used in Heat System Planning. PSO can be applied for various optimization problems, for example, Energy-Storage Optimization. PSO can simulate the movement of a particle swarm and can be applied in visual effects like those special effects in Hollywood film.

3.2 Genetic bee colony (GBC) algorithm

Bee food identification and collection intelligent swarm technique is defined in artificial bee colony. The best bee for the required problem is selected through parameters communication link, task allocation, reproduction, dance, placement mating and movement. GBC is optimised towards solution iteratively in an attempt to increase efficiency for any critical problem. Bee swarm is categorized as employee, onlooker and scouts. The employee bee identifies fresh sources of food. Scout bees job is to assign fitness quotient to entrust job of random search for employee bee identified spots. The assignment is random-based. If freshly identified food is better than earlier findings then, bees will collect from fresh location. Constantly employee bees look for best site for food collection. The onlooker bee is responsible to identify the best food source considering quantitative factor of food availability [21, 22].

3.2.1 Concept

The ABC algorithm consists of four main steps: initialization, employed bee phase, onlooker bee phase, and scout bee phase. After the initialization step, the other three main steps of the algorithm are carried out repeatedly in a loop until the termination condition is met. The main steps of the ABC algorithm are as follows.

Step 1 (initialization). In the initialization step, the ABC generates a randomly distributed population of *SN* solutions (food sources), where *SN* also denotes the number of employed or onlooker bees. Let φ represent the ith food source, where is the problem size. Each food source is generated within the limited range of ith index by where$\varphi = 1{,}2, \ldots SN, j = 1{,}2, \ldots .D, \varphi i, j$, is a uniformly distributed random real number in , x_{min} and x_{max} and are the lower and upper bounds for the dimension , respectively. Moreover, a trial counter for each food source is initialized as in Eq. (7).

$$x_{i,j} = x_j^{min} + \varphi_{i,j}\left(x_j^{max} - x_j^{min}\right) \tag{7}$$

Step 2 (employed bee phase). In the employed bee phase, each employed bee visits a food source and generates a neighboring food source in the vicinity of the selected food source. Employed bees search a new solution, by performing a local search around each food source as follows: where is a randomly selected index and is a randomly chosen food source that is not equal to; that is, is a random number within the range generated specifically for each and combination. A greedy selection is applied between and by selecting the better one as in Eq. (8).

$$v_{i,j} = x_{i,j} + \text{Ø}\left(x_{i,j} - x_{r1,j}\right) \tag{8}$$

Step 3 (onlooker bee phase). Unlike the employed bees, onlooker bees select a food source depending on the probability value, which is determined by nectar amount associated with that food source. The value is calculated for the food source as follows considering Eqs. (9) and (10): where the fitness value of solution and calculated as in (4) for minimization problems. Different fitness functions are employed for maximization problems. By using this type of roulette wheel based probabilistic selection, better food sources will more likely be visited by onlooker bees. Therefore, onlooker bees try to find new candidate food sources around good solutions. Once the onlooker bee chooses the food source, it generates a new solution using (2). Similar to the employed bee phase, a greedy selection is carried out between.

$$p_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_i} \tag{9}$$

$$fit_i = \begin{cases} \dfrac{i}{1 + fit_i}\, fi \geq 0 \\ 1 + abs(fi)\, fi < 0 \end{cases} \tag{10}$$

Step 4 (scout bee phase). A trial counter is associated with each food source, which depicts the number of tries that the food source cannot be improved. If a food source cannot be improved for a predetermined number of tries (limit) during the onlooker and employed bee phases, then the employed bee associated with that food source becomes a scout bee. Then, the scout bee finds a new food source using (1). By implementing the scout bee phase, the ABC algorithm easily escapes from minimums and improves its diversification performance.

It should be noted that, in the employed bee phase, a local search is applied to each food source, whereas in the onlooker bee phase better food sources will more likely be updated. Therefore, in ABC algorithm, the employed bee phase is responsible for diversification whereas the onlooker bee phase is responsible of intensification. The flow chart of the ABC is given in **Figure 3**.

3.2.2 Merits of GBC

The ABC algorithm is a population-based algorithm with the advantages of finding global optimization solution, being simple and flexible, and using very few control parameters. The ABC algorithm has been applied to many real-world applications, for example, function optimization, real-parameter optimization, digital filter design, clustering, and neural network training. ABC algorithm-based applications are easy to

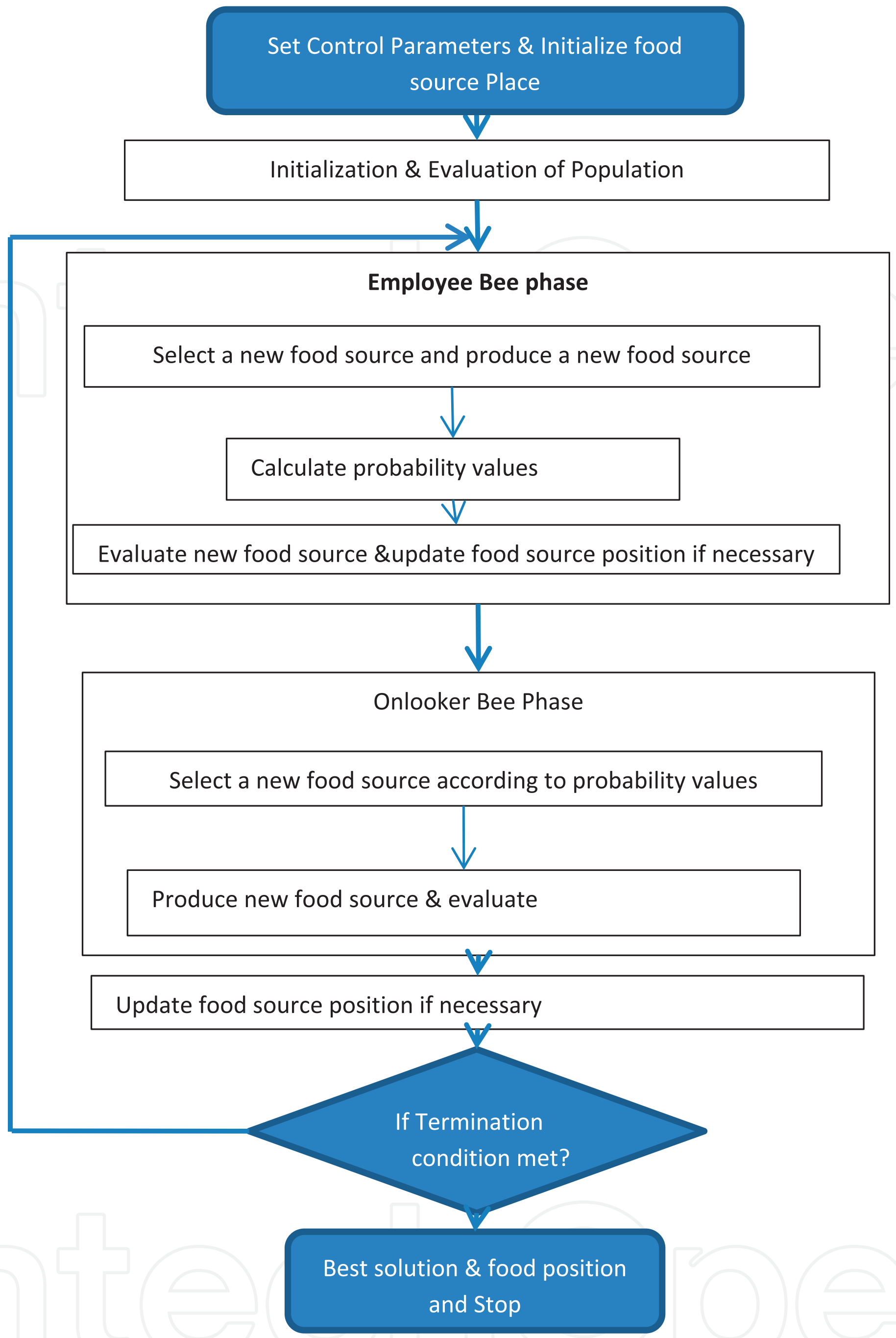


Figure 3.
GBC flowchart.

build, robust, converge fast, flexible and time efficient. Compared to PSO ACO parameters considered in ABC is less.

3.2.3 Demerits of GBC

Inspite of mentioned advantage GBC has few problems also. GBC are slow convergent speed in large computations and accuracy is less. GBC may face premature convergence problem for more duration application. The population size is fixed, and size is variations or non-autonomous. Individuals can extend the searching space and increase the probability of finding global optimization solution; however, it costs much time in each generation; oppositely, it may obtain a local minimum.

3.2.4 Applications of artificial bee colony (ABC) algorithm

GBC are more affine towards single-objective numerical optimization problems but not limited and can be extended to multi-parametric. Decision making, time schedule, assignment, search, inception, boundary setting, and network issued are other more applied fields of GBC [21]. GBC is capable to handle constrained and unconstrained, continuous and discrete, differential and non-differential oriented problems [22]. GBC is not specific in domain, applicable from agriculture to industry, and rural area to military field.

3.3 Fish swarm algorithm (FSA)

Fish swarm algorithm inspired by the behaviour of movement of aquatic fish in liquid medium. The target is picked randomly and moves toward in an iterative manner. Visually shorter distance considered an initial step, influences on final step. Initial values remain constant and considered along parameters. Suitable initial value selection leads toward the best optimum solution. Fishes are capable of venturing into bigger steps in search of a larger environment where they exist. So, fish is capable of escaping from unfavourable circumstances at any stage. But some deficiencies in large values may cause low steadiness. Global search is potential factor in generating local search with a larger visual position of FSA. Better fitness can be found for better fitness to search for parameters to make the algorithm steady and accurate. Fish are capable of moving quickly towards the target and can get passed from local best search results. FSA algorithm design has undergone many changes in design in order to fulfil the needs of different types of problems. The variation in algorithm can be grouped into solutions of FSA for continuous and discrete, combinatorial and binary, multi-parametric and hybrid FSA. Fei et al. [23] selected nine search positions to initialize the Afs for motion estimation. Zhu et al. [24] and Gao et al. [25] used the chaotic transformation [26] method to generate a more stable and uniform population. Kang et al. [27] used a uniform initialization method to initialize the population, while Liu et al. [28] initialized the

Afs based on the optimization problem in hand. The MSAFSA [29] model introduced both the leaping and swallow behaviors to escape from the local optima and reduce, Yazdani et al. [30, 31] introduced mNAFSA for optimization in dynamic environments.

3.3.1 Concept

Fish Swarm algorithm and background is discussed. Notations used are X, V, S, X_v indicate current position of fish, distance, step, position, respectively. N visual fishes are indicated as X1, X2, X3 … … .. Xn. Y = f(X) denotes the food concentration of the AF at the current position, di;j =||Xi – Xj ||. The FSM involves four key operations: preying behaviour, swarming behaviour, following behaviour and random behaviour. Preying is fish behaviour to move itself towards high concentration of food. It is represented mathematically as in Eq. (11) considering with in visual distance ith fish. The fish preying will continue to try number of times if not satisfied within, then

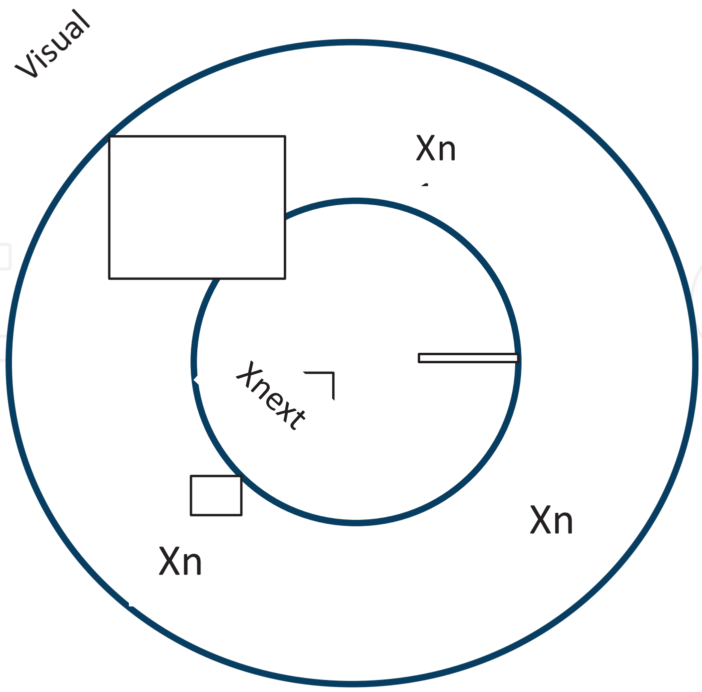


Figure 4.
FSA visual.

randomly computed using Eq. (12). Fishes group among themselves to from any danger situations against them. Mathematically, central position in fish swarm is computed as in Eq. (13). Fish when a locates good concentration of food. The preying movement for fish in step movement is represented in Eq. (14). Few fishes randomly move freely if lie in sparely concentrated food. This behaviour is modeled as in Eq. (15) (**Figure 4**).

$$X_j = X_i^{(t)} + V * rand() \tag{11}$$

$$X_i^{(t+1)} = X_i^{(t)} + V * rand() \tag{12}$$

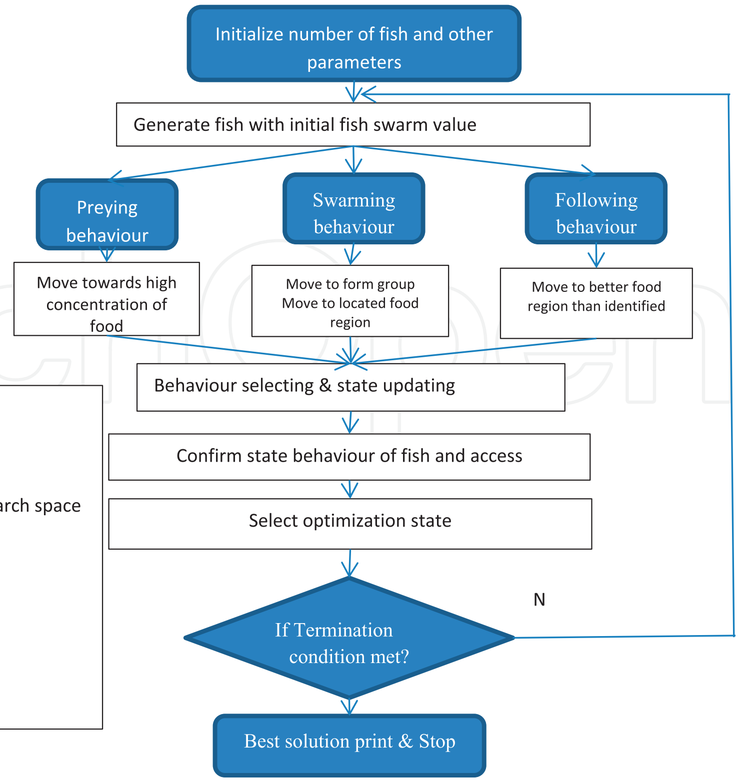
$$x_{cd} = \frac{\sum_{j=1}^{nf} x_{jd}}{n_f} \tag{13}$$

$$X_i^{(t+1)} = X_i^{(t)} + S * rand() * \left(\frac{X_j - X_i^{(t)}}{\|X_j - X_i^{(t)}|(t)|} \right) \tag{14}$$

$$X_i^{(t+1)} = X_i^{(t)} + V * rand() \tag{15}$$

3.3.2 Algorithm and flowchart

FSA perform record one if new. This search continues until end is not met following four operational steps as mentioned in previous section. The algorithm FSA is shown in briefed in **Figure 5**.




```
Algorithm: FSA
Input: Initialize necessary parameters
Output: Optimal fish state as solution

Generate n random fishes within the given search space
while not m steps then
    for n fishes do
        Compute fitness value
        Evaluate behaviour
    end for
    update best solution
end while
```

Figure 5.
FSA algorithm & flowchart.

3.3.3 Merits

FSA similar to GBC has got increased convergence power and flexible. In addition, it exhibits fault to tolerant and accuracy feature. Global search ability, tolerance of parameter setting and robustness are other merits of FSA. It solves nonlinear and multi modal problems.

3.3.4 Demerits

FSA exhibits high complexity, lack of balance among one is ineffective if lack balance between local search. Not suitable for global convergence problem. The information transfer if experience low search rate. As good robustness, global search ability, tolerance of parameter setting, and it is also proved to be insensitive to initial values

3.3.5 Applications

FSA has been applied for network related problems, control of resources, image processing related problems. In order to increase evolutionary capabilities of FSA, in few swarm solutions hybrid to FSA.

FSA incorporate to solutions in wireless sensor networks [30, 32, 33], tracking [34], medical estimations [35–37], segmentation [32], clustering [33], regression [38], image processing [39, 40], calibration [35], localization [41], power systems [36, 42].

3.4 Cat swarm optimization (CSO)

Chu et al. introduced cat swarm optimization technique to solve most of engineering problems inspired by the movement of cats. The process is carried on in two different modes seeking and tracing modes. Nodes virtually move in fixed areas as cats to determine optimal solution. Number of virtual cats are fixed in both modes and predefined in few cases ratio known as MR. the N virtual cats is placed randomly. Processed and unprocessed cats are identified for each dimension based on its value of MR set either to 0 or 1 for tracing or seeking in coming rounds. Every cat compute fitness function in evaluation then among the existing best will be chosen initially existing is compared if It best retained for fitness function otherwise coordinates will be changed to new best cat. The movement of cat adjusted towards solution space identified as identified in initially. Choose for unprocessed cats in tracing mode through permutation. Tracing mode ends if no more cats are left. Traced coordinate nodes will be selected as best solution at end. In seeking mode, cats' movement will be slow and conformist. Essential parameters of seeking mode are seeking memory pool, Ra range of identified dimension, counts of dimensions to change and self-position. Improved CSO algorithm proposed by Tsai et al. supports parallel information exchange in tracing mode. The parallelizing of virtual agents is adopted in PCSO [43, 44]. PCSO finds application in parallel processing inspired by colonies of cats tracing for food.

3.4.1 Concept

CSO identification of optimized solution is described in this sub section step by step. The seeking feature of cats carried in five processing steps. In first step, j copies of cat generated is recognized applying Eq. (16). Addition or decrement of SRD value on selected search space defined by Eqs. (17)–(19) in step two. In next step, fitness value for all candidates is selected. In step four, calculation of probability of cat performed by Eq (20). Sort and select best solution by roulette wheel selection in last step. In tracking mode cats imitate movement of prey during tracing. This process can be discrete into three operational steps. In first step, velocity of each cat is updated as in equation 6 for given search space. The random value for cat adjusted in range 0-1. In step 2, the valued are rearranged based on velocities of cat. Velocities are set to maximum velocity value. Position of cat is updated selecting by Eq. (7) in last step.

$$j = \begin{cases} SMP & SPC = \text{"}true\text{"} \\ SMP - 1 & SPC = \text{"}false\text{"} \end{cases} \tag{16}$$

$$M = Modify \cup (1 - Modify) \tag{17}$$

$$|Modify| = CDC * M \tag{18}$$

$$x_{jd} = \begin{cases} x_{jd} & d \nexists Modify \\ (1 + rand * SRD) * x_{jd} & d \in Modify \end{cases} \tag{19}$$

$$p_i = \begin{cases} 1 & when\ FS_{max} = FS_{min} \\ \dfrac{|FS_i - FS_b|}{FS_{max} - FS_{min}}, & whare\ 0 < i < j,\ otherwise \end{cases} \tag{20}$$

$$v_{k,d} = v_{k,d} + r.c.(x_{best,d} - x_{k,d}), d = 1,2 \ldots ..M \tag{21}$$

$$x_{k,d} = x_{k,d} + v_{k,d} \tag{22}$$

3.4.2 Algorithm and flowchart

The algorithms and flow of operations of CSO is summarized in **Figure 6**.

3.4.3 Merits

COA is Simple to construct and have minimal parameters to adjust. COA has got ability to execute in parallel system. The design is robust. Can converge fast, find global solution, overlap and mutate. Have computational time less. Find accurate mathematical models. Discover good and rapid solutions. Adapt changes in new system and dependent on random decisions.

3.4.4 Demerits

Definition of initial parameters is time consuming. COA does not work better for scattering problems and can converge at faster rate if trapped in complex problems. The time to converge and towards convergence for multi objective and larger sized problem is more.

```
Algorithm  : CSO
Input: Number of cat,
Output: Best cat as solution
Begin
    Determine best MR applying minimal rounds
    Create cat population
    While not termination condition do
        For size n do
            If cat is in seeking mode then
                Apply seeking mode
            Else
                Tracing mode
        End for
    Evaluate best fit
    End while
    Store best solution
end
```

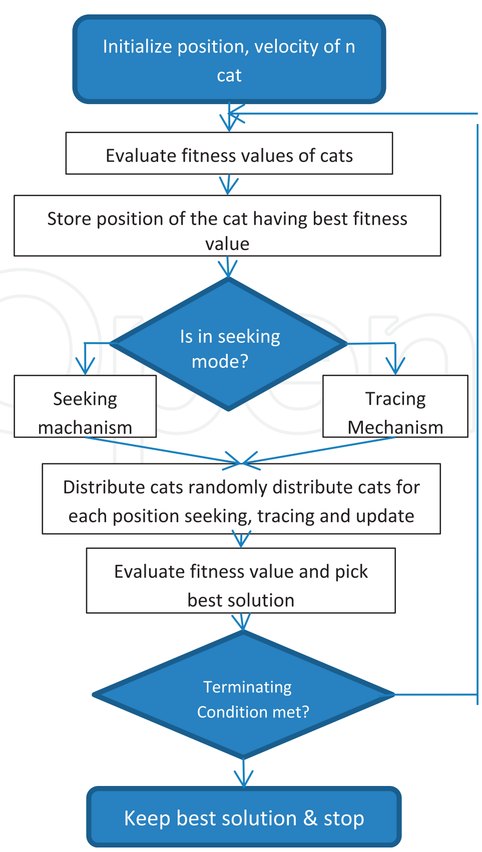


Figure 6.
COA algorithm & flowchart.

3.4.5 Applications

CSO optimization is being incorporated in media for information hiding [44], aircraft scheduling recovery in limited processing time [45]. Voltage stability, economically dispatch in transmission system, hybrid generation systems, task allocation, data mining, project scheduling, optimal contract capacity, global numeric optimization problems. Applied for clustering technique in green expression classification, travelling salesman problem, data hiding, graph coloring, SVM, K means. CAO even find its application in stock market and supply chain in currency exchange rate analysis and stock prediction. COA is applied in image processing for machinery fault detection, plant modeling, image edge enhancement, water marking and single bit map. COA extended application in electronics for cognitive radio engine cooperative, spectrum sensing, linear antenna array synthesis, aircraft maintenance, routing for wireless sensor network.

3.5 Whale optimization algorithm (WOA)

Whale optimization algorithm proposed by Mirjalili et al. WOA is also based on population of whale. It simulates bubble-net attacking method of humpback whales when hunting their preys. Whales are intelligent due to the spindle cells in their brain. They live in group and are able to develop their own dialect. Whale optimization algorithm consists of two modes of operation. The two modes of operation named as exploitation and exploration. In first prey encircling and position update in spiral manner carried on. Searching for prey randomly done in second phase [46–48]. WHO exploitation phase for prey encircling is mathematically equalized as bubble net attack system. Humpback characteristics of whales considered for phase one behaviour. Whales encircle prey with identification of them in an undefined search space. Initial solution of nearby prey or ideal assumed as best further best solution will be updated once exploration begins. Distance between prey and whales calculated initially then, updates for spiral positioned distance to it. WOA has modified and incorporated improvements by many researchers [49–53]. Few notable changes included in AWOA, IWOA, chaotic WOA, ILWOA, and MWOA research work. WOA hybridized with other meta-heuristic algorithms PSO, BA, and others in order to improve local search [34, 54–56].

3.5.1 Concept

Whale has a special hunting mechanism which is called bubble-net feeding method. This foraging behaviour is done by creating special bubbles in a spiral shape or nine shape path. Humpback whales know the location of prey and encircle them. They consider the current best candidate solution is best obtained solution and near the optimal solution. After assigning the best candidate, the other agents try to update their positions towards the best search agent as computed by Eq. (23). In Eqs. (23) and (24), t is the current iteration, A and C are coefficients vectors, X^* is the position vector of the best solution. The vector A and C are calculated using Eqs. (25) and (26). In Eqs. (25) and (26) a are linearly decreased from 2 to 0 over the course of iterations and r is random vector in [0, 1]. The humpback whales attack the prey with the bubble-net mechanism in exploitation phase. In shrinking encircling mechanism, the value of A is a random value in interval [-a, a] and the value of a is decreased from 2 to 0 over the course of iterations. Spiral updating position mechanism calculate the distance between the whale location and the prey location is calculated then the helix-shaped movement of humpback is created using Eq. (28). D' = |X*(t) – X(t)| is distance between the prey and the ith whale, be is a constant, l is random number in [−1, 1]. Whale selectively applies swim around prey techniques suitably. The mathematical model of these two mechanisms assumes to choose between these two mechanisms to update the position of whale as in Eq. (29). In steady exploitation phase the humpback whales search for prey and change their position of whale. The force the search away from reference whale the mathematical model of exploration is computed as in Eqs. (29) and (30).

$$X(t+1) = \begin{cases} X*(t) - A > D \; if < 0.5 \\ D^{t}.e^{bl}.\cos(2\pi l) + X*(t) \; if \; p \geq 0.5 \end{cases} \tag{23}$$

$$D = |C.X*(t) - X(t)| \tag{24}$$

$$X(t+1) = X*(t) - A.D \tag{25}$$

$$A = 2a.r.a \tag{26}$$

$$C = 2.r \tag{27}$$

$$\mathrm{X(t+1)} = D^{t}e^{bl}.\cos(2\pi l) + X*(l) \tag{28}$$

$$D = |c.X_{rand.X}| \tag{29}$$

$$X(t+1) = X_{rand} - A.D \tag{30}$$

3.5.2 Algorithm and flowchart

The detailed workflow and algorithms is presented in **Figure 7**.

```
Algorithm :Whale Swarm Algorithm
Input: Number of whale, position,variable v=0,u =infinity
Output:  Optimal solution
Begin:
    Initialize parameters
    Initialize whale positions
    Compute fitness value of all whale
    while not final criterion do,
        for n whales do,
            Find the better and nearest whale  among all n
            For i in m objective values do
                If u not equal to m then
                    If f(m_i) < 0.5 then
                        If dist(m_i, m_v) < u then
                                v= i & u dist(m_i, m_v)
                        end if
                    end if
                end if
            end for
            if Solution exist then
                N moves under guidance of best near whale
                Evaluate fitness value
            end if
        end for
    end while
store better optimal solution
end
```

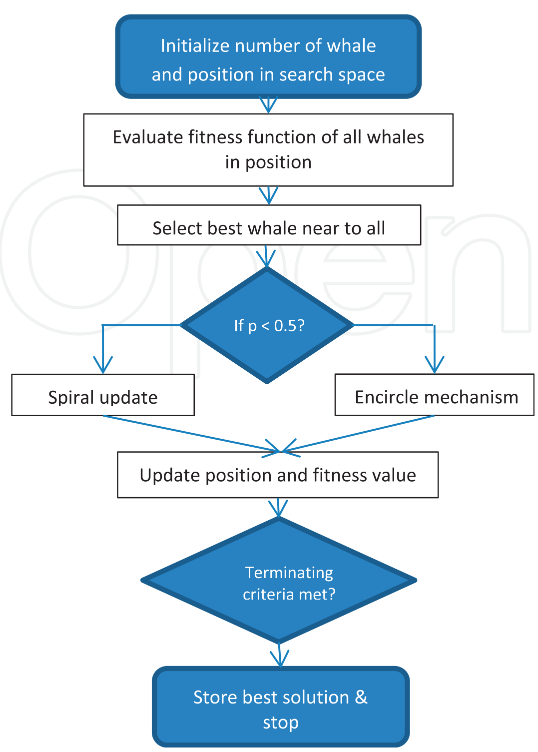


Figure 7.
WOA algorithm & flowchart.

3.5.3 Merits

Whale optimisation avoid problem of local optima have got ability to compute local and global optima for any constrained or unconstrained optimization applications. During the process, it even does not require any structural are parametric rearrangement or alteration in value. The exploration for best solution computed simply and easily at faster rate. Improve quality of generated population and converge at faster rate.

3.5.4 Demerits

WOA is not suitable for larger spaced problem incurs more time to explore and converge. The accuracy of solution is questionable. The optimal solution cannot be recognized for optimization of problems to solve high dimensional problem. Randomization technique of core WOA solution is complex. Balance among process of exploration and exploitation is lacking. The encircle mechanism slows WOA to jump from one local optima to another yielding low performance. Application problems of classification and dimensional reduction problem

3.5.5 Applications

WOA has ability to incorporate in dynamic applications. Most of researchers applied WOA for electrical, mechanical and management problems. WOA has been used to solve problem in engineering, multi objective, binary, identification classification and scheduling. WOA has found problems Power plants and systems scheduling [57] has confirmed to standard radial systems. Verify test system in execution of IEEE 30-bus [58, 59]. Size of pillars and optimization to increase efficiency of building is analysed [60]. Energy rise of solar energy to get importance in design of photovoltaic cells. WOA benefit solar cell and photovoltaic cells [61] by calculating internal parameters automatically. The partially cloudy atmosphere traced to get highest power region by a modified artificial killer whale optimization algorithm MAKWO [62]. Medical image analysis for classification and diagnose liver and cluster based abdominal to avoid intensity values to overlap [63]. WOA incorporated in economic and emission dispatch [64], vehicle fuel consumption [65], mobile robot path planning [66], optimal allocation of an ameliorative of water resource [67], design problem [68], heat and power economic dispatch [69].

3.6 Artificial Algae Algorithm (AAA)

Artificial Algae Algorithm initially proposed in 2015 by Uymaz et al. is also a meta-heuristic bio inspired algorithm. Microalgae growth and reproduction in presence of sunlight behaviour are considered in algorithm AAA. Algae swim towards presence of sunlight for food production following process as photosynthesis. The movement of algae towards sunlight will be in helical manner. They live in groups as algae colonies. The algae identify best sunlight presence to carry on photosynthesis itself considering largest size and reproduce algae's with highest energy. In case sunlight presence is less, then size of algal colony and energy level is less and starts for high starvation level. If sunlight is less algae colony tries to adopt itself in environment for its survival otherwise algae cells die because of starvation. The adaptation of algae cells in unsupportive environment is known as evolution [70]. Uymaz et al. developed AAA then they modified to perform better [71]. From then many researchers contributed for AAA by incorporating AAA in different fields. Multi-objective optimization for AAA designed by Babalık et al. [72]. Binary version presented by Zhang et al. [73]. AAA applied in various fields from processing to manufacturing and in applications ranging from agriculture to home [74]. Few researcher improved AAA through hybridization [75].

3.6.1 Concept

AAA proposed for first by Uymaz et al. deals with considering advantages of research area of the properties found in algae. Algae moves from helically towards lighter sources. Algae adopt in nature to adapt and reproduce forming colonies which represent a solution. Colony of algae consists set of cells which dwell together. The colony exposed to external forces. The algae are divided into group and each become new colony as can move jointly, under in appropriate circumstance to from new colony. AAA process incorporated by three parts: Evolutionary, adaptation and helical movement. In evolutionary process, algae colony grows and flourish to get sufficient light, and benefit conditions. The algae undergo mitosis to result in two new algae. If not algae will perish under less nutrition and lighter conditions. In few scenarios if algae cannot grow in an environment due to lack of supporting factors. In such environments algae adapt by itself to environment in order to survive as other species. Finally, algae if it could not adapt then moves toward large grouped algae. If starvation occurs algae stop to adopt. Algaes move in helical movement by swim. In order to live they try to remain close to surface of water to get light. The search capacity will not remain same. Algae growth is more in region where frictional surface is more. The chance of algae movement is more in fluid. Helical motion supports to move algae at higher rate. The energy in different surfaces is not constant and is directly proportional to quantity of food and type of nutrient available in the environment. Capability and survival of algae existence depend on its adaptation and movement. The algae survival process mathematically applied in functional parts. Initially fix size of algae by Eq. (31). Evaluate fitness value of algae and size of colony by Eq. (32). Adaptation of algae is through growth of algae and use of nutrients by Eq. (33). The energy of algae computation inclusion of frictional force is computed applying Eqs. (33) and (34). During adaptation process algae build itself under non favorable or movement to nearby stronger and larger algae colony part. The optimization for given problem can be computed by Eqs. (35)–(38). The three subgroups of algae considered for adaptation. Identification of starvation be Eqs. (39) and (40). Section of best solution is selected by Eq. (41).

$$X_{ij} = LB_j + \left(UB_j LB_j\right).RAND\ i = 1,2,3 \ldots ..N; j = 1,2,3 \ldots .D \tag{31}$$

$$\mu_i = \frac{S}{K_s + S} \tag{32}$$

$$G_i^{t+1} = \mu_i^t G_i^t \ \mathrm{i} = 1,2,3, \ldots \ldots .\mathrm{N} \tag{33}$$

$$\tau(x_i) = 2\pi \sqrt[3]{\frac{3G_i}{4\pi}}^2 \tag{34}$$

$$GE^{t+1} = norm\left(\left(rank\left(G^t\right)\right)^2\right) \tag{35}$$

$$X_{i_m}^{t+1} = X_{i_m}^t + \left(X_{j_k}^t - X_{i_k}^t\right)\left(\Delta - \tau^t(X_i)\right)P \tag{36}$$

$$X_{i_k}^{t+1} = X_{i_k}^t + \left(X_{j_k}^t - X_{i_k}^t\right)\left(\Delta - \tau^t(X_i)\right)\cos\alpha \tag{37}$$

$$X_{i_l}^{t+1} = X_{i_l}^t + \left(X_{j_l}^t - X_{i_l}^t\right)\left(\Delta - \tau^t(X_i)\right)\sin\beta \tag{38}$$

$$Starving^t = maxA_i^t \; i = 1,2,3, \dots \dots .N \tag{39}$$

$$Starving^{t+1} = Starving^t + \left(Biggest^t - Starving^t\right).rand \tag{40}$$

$$biggest^t = maxA_i^t \; i = 1,2,3 \dots \dots \dots .N \tag{41}$$

3.6.2 Algorithm and flowchart

The algorithm and flow of operation of AAA is shown in **Figure 8**.

3.6.3 Merits

AAA exhibits accuracy for identified colonies. Converge faster towards local and global solution compared to ACO or PSO. Algorithm is convenient and efficient. The method helps find efficient and high accurate result. Produce robust algorithm for real-time optimization problems. Main benefit for gradient-based problems provide by an efficient optimize in few steps and simple to generate.

```
Algorithm : Artificial Algae Algorithm
Input: number of algae and colony
Output: Best solution and colony

Begin
Generate an initial population of n algal colonies with random solution
evaluate f(xi), i = 1, 2, …, D
while stopping condition not reached do
for n algae do
while energy of ith colony not finished do
modify the colony with helical movement
end while
end for
apply evolutionary
apply adaptation
apply helical movement
end while
keep best solution
end
```

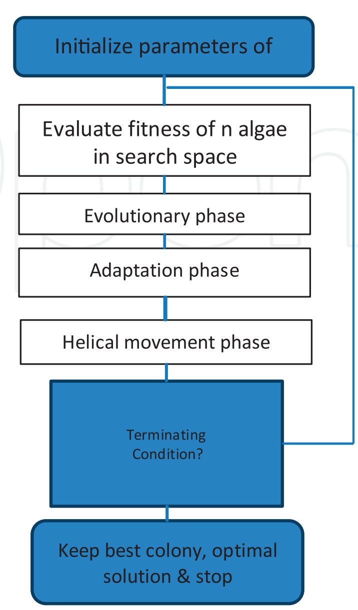


Figure 8.
AAA algorithm & flowchart.

3.6.4 Demerits

Major problem of AAA is its expensive apparatus, consumption of time and specialized operator. If data and input size increases accuracy will be minimized. They tend to stick to local optima, increased dependency. AAA apply randomness by this methodology is simple but result accuracy is questionable. Hence applications involved in AAA are complex and provide unstable result.

3.6.5 Applications

In optimal placement distributed power flow controller (DPFC) with MCFC, optimal coverage, routing and selection of cluster head in wireless sensor network.

3.7 Elephant search algorithm (ESA)

Elephant search algorithm developed by Adams et al. is inspired by elephant search for water. Normally elephants search for water in drought within swarm. Elephant swarm together search water source. Each elephant swarm consists of leader responsible to make decision regarding movement of whole group. Elephant is identified by its particular position and velocity in each group very similar to other swarm techniques. Leader elephant informs rest of elephants in group in case best water source is identified. The communication is through chemical, tactile, acoustic or visual means. The fitness function is computed considering water source quality and quantity. The elephants' group can move from one water source to another and visits previous also if necessary as they got good memory. Group visit previous water source in case older identified is best solution in compassion to new water source. Elephants search for best solution locally and globally then best solution will be identified in given solution space following long and short distance communication. Switching probability is key controller in considering water search either local or global.

3.7.1 Concept

EHO is meta-heuristic simulated behaviour in herds of elephants [24] introduced by Wang. Optimize solution for global optimization tasks [5]. Each solution I in each clan ci is updated considering current information such as position and matriarch. The generations are updated by algorithm execution through separating operators. Each individual in heard represent vectors in 2D. The dimensions in unknown population are included. The population is divided into n clans. Updating operator is modeled by increment or decrement each solution i in the clan by ci by influence of ci to identify best fitness value in generation. Fitness update solution in each clan c_i represented in Eq. (42). New and old position in clan, incremental factor based on influence of matriarch are parameters included of Eq. (43). In 2D the central clan is computed through Eq. (44). It updates individual value of elephants in heard. The total search space indicates number of solutions in clan. The separating search space and n_{ci} indicates number of solutions in clan in c_i. The separate operator is applied at each generation for execution on worst individual in population. Choose random population [0-1] be uniform distribution range within lower and upper limits of the position of the individual by Eq. (45).

$$x_{new,c_i,j} = x_{ci,j} + \alpha.\left(x_{best,c_i} - x_{c_i,j}\right).r \tag{42}$$

$$x_{new,ci,j} = \beta.x_{center,c_i} \tag{43}$$

$$x_{center,ci,d} = \frac{1}{n_{ci}}.\sum_{j=1}^{d} x_{c_ij,d} \tag{44}$$

$$x_{worst,c_i} = x_{min} + x_{max} - x_{min} + 1) * rand \tag{45}$$

3.7.2 Algorithm and flowchart

The detailed algorithms and flow of operation of EHOA is presented in **Figures 9** and **10**.

3.7.3 Merits

EHOA is more performance stable than other meta-heuristic algorithms such as PSO. Convergence is faster because they are in herd. Have ability to search a population in parallel. Rapidly discover good solutions similarly adapt to changes such as distance. The computation is simple. EHOA is efficient in solving problem which are difficult to find accurate mathematical models. Computational time is less and overlap is avoided.

3.7.4 Demerits

Probability can change for each iteration, theoretical analysis is difficult, and sequence of random decisions are major hindering factors of EHOA. Time requirement for convergence is uncertain.

```
Algorithm 1 Pseudo-code of EHO algorithm
Input: n elephant and max. generation
Output: Best solution
Begin
Generate population and calculate fitness
 Divide population into m clans
Calculate fitness of each individual
For n individuals
While not termination condition do
Sort all solution according to their fitness
For all clans do
Update generation and find fitness
End for
Update best
End while
End for
Select best if existing is best retain
end
```

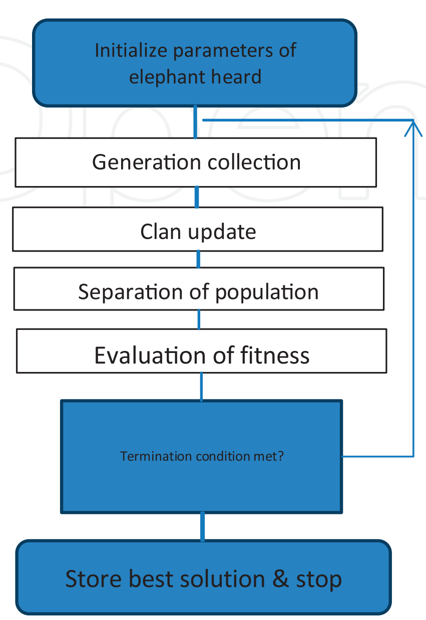


Figure 9.
EHOA algorithm & flowchart.

Algorithm : Cuckoo search algorithm
Input: n nests with eggs
Output: best egg as solution
Begin
Objective function f(X), $X = (x1, ..., xd)^T$
Generate initial population of n host nests X_i (i = 1, 2, ..., n)
while (t <MaxGeneration) or (stop criterion)
Get a cuckoo randomly by L´evy flights
evaluate its quality/fitness F_i
Choose a nest among n (say, j) randomly
if ($F_i > F_j$),
replace j by the new solution;
end if
A fraction (p_a) of worse nests are abandoned and new ones are built;
Keep the best solutions
Rank the solutions and find the current best
end while
Postprocess results and visualization
end

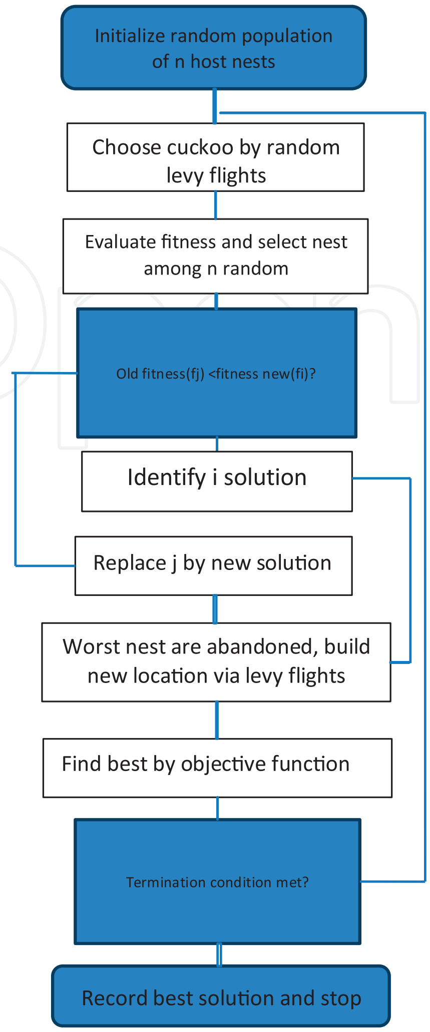


Figure 10.
CSO algorithm & flowchart.

3.7.5 Applications

EHO applied to optimize training artificial neural network, selection structure and weight for neural networks, training neural netwosk, optimizing underwater sensor networks, unmanned aerial vehicle path planning, clustering, support vector machine, control problem.

3.8 Cuckoo search optimization algorithm (CSOA)

Yang and Deb introduced cuckoo optimization in 2009 a meta-heuristic algorithm. Later Gandomi, Yang, and Alavi and Yang and Deb extended to solve single or multi-objective problems involved in any constraints or complexity. The solution is capable to resolve potential solutions of any randomly selected population in habitants of cuckoo. The function of CSOA is global optimality, real-world problems are NP-hard for problem used in any problem. Construct workable solution required to be globally optimal solution replicating behaviour of cuckoos. They lay eggs in nest of other birds and obliterate eggs of birds to guarantee hatching of its breed. Cuckoos brood parasitism is simulated in three different ways: Intra-specific brood parasitism, nest take over and co-operative breed. The basic cuckoo search algorithm has undergone changes convergence speed of cuckoo search algorithm is increased in modified cuckoo search [74] by avoiding cross overs. Binary version of cuckoo search algorithm is presented in [75] to increase accuracy by reducing problem to binary coordinated feature. In [76] improves cuckoo search by resetting position and random vector value of eggs rather considering as static parameter value.

3.8.1 Concept

CS algorithm is based on the obligate brood parasitic behaviour of some cuckoo species in combination with the levy flight behaviour of some birds and fruit flies. Some species of Cuckoo birds lay their eggs in communal nests. If a host bird discovers the eggs are not their own, they will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere. CS, can be described using following three idealized rules:

a. Each cuckoo lays one egg at a time, and dump its egg in randomly chosen nest;

b. The best nests with high quality of eggs will carry over to the next generations;

c. The number of available host nests is fixed, and the egg laid by a cuckoo is discovered by the host birth a probability pa ∈ [0, 1].

3.8.2 Algorithm and flowchart

The algorithm and flow of operations of CSOA is presented in **Figure 10**.

3.8.3 Merits

A meta-heuristic method exhibits several advantages as easier for applications to change parameters to meet requirement of applications. It is very easy fall for optima of local solution to slow convergence rate. In addition, cuckoo search is simple and easy to follow with real-world engineering applications. Cuckoo search algorithm easy to implement in comparison to other population algorithms.

3.8.4 Demerit

CSA is about easy to fall into local optima solution due to its simplicity. Slow the convergence rate randomness is still a problem. Self-adaptability may be limitation under certain problems. Low efficiency, less accuracy can be experienced while dealing with multi-peak function.

3.8.5 Applications

Cuckoo search optimization algorithm applied for different problems in various domains. Power generations to minimise the cost of flues , n power with probability to generate in different values, Cloud computing security frameworks are-Gathering information, Network mapping, vulnerabilities exploration, audits and penetration tests, vulnerabilities enumeration and categorization, technology selection for vulnerability remediation, security solutions implementation. The security technology is used to decrease the vulnerability and costs are called Set covering problem [75] that is the Distribution systems will have more power loss and poor voltage regulation and voltage stability. VANET protocols design [76], electromagnetic and antenna arrays [77], classification of IDS [78]. Self-adaptive algorithm for search accuracy of the CSA [79], Compression factor to build [80], dynamic appropriate step-size [81]. CSA have been applied in many researchers in different application problems such as multilevel image thresholding, flood forecasting, wireless sensor networks, data fusion, cluster in wireless networks, clustering, ground water expedition, supplier selection, load forecasting, surface roughness identification, DG allocation in network, BPNN neural network, web service composition, speaker recognition, face recognition, training neural networks [82–85].

3.9 Moth flame optimization (MFO)

Mirjalili proposed moth flame optimization algorithm a swarm algorithm inspired by movement of moths in spiral path around light source. Moth flames randomly start searching in solution space. The fitness value estimated based on position by each moth in group. Falling category to best position flame by all is optimal solution. The function category updates following spiral movement function to achieve better division towards light source. The best position can be individual positions and repeats updating moth's distance and position generate new position to terminate criteria to be met. The variations in moth flame design in oder to improve are for multi-objective, binary and hybridization

3.9.1 Concept

Mirjalili proposed meta-heuristic algorithm based on population. MFO moths randomly with in space recognize fitness value and identify position suitable without flame. The movement is continuous and repeated to recognize better position. Update position suitably until termination criteria is met. The process MFO is carried on in three main steps. In first step, initialization of population and parameters are assumed in hyper dimensional space. The difference in way updates and treats in iterations. The position of each moth is stored. The selection of best moth is also performed so that results are stored longer time. In second step, three main functions converge to global result in Eq. (46). The identification to optimization is implemented randomly. Movement is spiral in moths applying logarithmic spiral function by Eq. (47). Moth and flame fixed position and indicate [−1, 1] ranges. It balances between exploitation and exploration to guarantee moths circulation in search space guarantee in spiral motion. The fly of moth is traps of the local optima. Moth positioned near flame represented in matrix. In step 3, number of flames is updated; Moths locations search the exploitation in search space. Decrease and solve issue based on Eq. (48).

$$M(i,j) = (ub(i) - lb(j) * rand() + lb(i)) \tag{46}$$

$$S(M_i, F_j) = D_i e^{bt}.\cos(2\pi t) + F_j \tag{47}$$

$$flamecount = |N - l * \frac{N-l}{T}| \tag{48}$$

3.9.2 Algorithm and flowchart

The algorithm and flow of operations of MFOA is presented in **Figure 11**.

3.9.3 Merits

MFO similar to most population-based algorithm flexible and robust. The local opitma for individuals is avoided. Construction is easy and flexible in design. Moth has been incorporated to solve many engineering problems.

3.9.4 Demerits

Convergence is major issue in MFO.

3.9.5 Applications

MFO advantages have been incorporated in many domains. Navigation approach to solve the inequality and equality constrained optimization are real problem, to optimize real function for constrained selected variables. Chemical identification to improve single level production which can be extended to incorporate as include in determination of optimal production portfolio in other industries, applied in agriculture based to recognize problems of tomato [52]. Applied for medical field to improve time consuming Alhemeris disease, detection and diagnosis of breast cancer, to train networks RBFN [42], deployment of Wifi, determination of optimal solution in placement, location problem solution.

```
Algorithm : Moth-flame optimization Algorithm
Input: Number of moth-flame
Output: Best optimized solution
Begin
Parameter math-flam initialization
Moth position Pi randomly
For n moths do
Calculate fitness of all moths
end for
While not termination condition do
update position of Pi
calculate flames number
evaluate fitness function
if  initial round then
sort all moths & store best and second best
else
sort excluding previous best moth end if
For n moths do
For d flames do
Update parameters of moth flame
compute number of flame near to each moth and update respectively
end for
end for
end while
store best solution
end
```

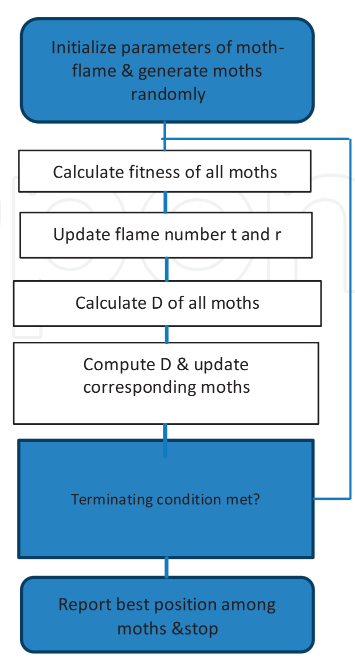


Figure 11.
MFO algorithms & flowchart.

3.10 Grey wolf optimization (GWO) algorithm

Grey wolf optimization, a meta-heuristic swarm technique, introduced for first time by Simon Fong [86]. Hunting behaviour in pack of wolf inspired in design of grey wolf optimization. Wolves in pack will not communicate physically during hunting, each wolf identify and attack prey individually silently. They follow levy flights model in search of food during hunting. Wolves unify to another pack of wolves or to new location if they find new food location better and suitable compared to their current dwelling place. A random hunter will be selected among pack to hunt for prey. The hunter identifies potential position itself to catch prey from current line of sight.

3.10.1 Concept

The social hierarchy consists of four levels in GWO. The level one called Alpha. They are the leaders of the pack, and they are male and female. They are responsible for making decisions about hunting time to walk, sleeping place and soon. The pack members have to dictate the alpha decisions and they acknowledge the alpha by holding their tails down. The alpha wolf is considered the dominant wolf in the pack and all his/her orders should be followed by the pack members. Next level group is labeled as Beta. The betas are subordinate wolves, which help the alpha in decision making. They can be either male or females. If consider the best candidate to both alpha when the alpha passes away or becomes very old. The beta reinforces the alpha's commands throughput the pack and gives the feedback to alpha. The third group of wolves is called Delta. They are subordinates. They need to submit their work report to alpha and beta. Scouts are responsible for watching boundaries of the territory and warning the pack in case of any danger. Sentinels are responsible for protecting the pack. Hunters are response got helping the alphas and beta involves beta in hunting and provide food for the pack. They are not important individuals in the pack, and they are allowed wolves were outwards. They are fighting i the case of loss.

Wolf search has been used to select two relay nodes: inter and intra relay nodes. Within a cluster, cluster members sense and transmit sensed data directly to the CH irrespective of their distance from CH. Hence, the nodes far away from CH dissipate more energy resulting in reduced network lifespan. To overcome this problem, the Wolf search is used in order to identify intra relay nodes for every cluster. The cluster member will send the sensed data to intra relay node and it in turm to CH. Similarly, all CHs communicate directly to BS irrespective of distance between CHs and BS. Hence, the CHs far away from BS dissipate more energy which leads to selection of new CHs resulting in next iteration, resulting very low network lifespan. To overcome this PEGASIS protocol introduced inter relay node as final node to communicate with BS. In proposed work, Wolf search is used to identify the inter relay nodes. The working principles of Wolf search for identification of inter and intra relay nodes are described in this section. The pseudo code of Wolf search is described in Algorithm 2.11.2.

(X, Y) are the coordinates of unknown node/target node and (x_i, y_i) are the coordinates of the i^{th} anchor node in the neighbourhood. The computations of WS) for encircling, and hunting process are shown below.

Eqs. (49)–(54) used in WSO are as follows.

$$d_i = \sqrt{(X - x_i)^2} + \sqrt{(Y - y_i)^2} \tag{49}$$

$$D = |C * X_p(t) - X(t)| \tag{50}$$

$$C = 2 * r \tag{51}$$

$$A = 2 * a * r - a \tag{52}$$

$$X(t+1) = X_p(t) - A * D \tag{53}$$

$$r = 0.5 + \frac{\sin 2\sqrt{x^2 + y^2 - 0.5}}{(1 + 0.001X(x^2 + y^2))^2} \tag{54}$$

where t represents the current iteration, A and C are coefficient vectors, position vector of the prey is represented X_p, X the position vector, | | is the absolute value, and is an element-by-element multiplication, a is linearly decreased from 2 to 0 in each iteration and r is a random vector in $[0, 1]$.

3.10.2 Flowchart and algorithm

The algorithm and flow of operation of GWO is presented in **Figure 12**.

Algorithm : Grey wolf optimization Algorithm
Input:
Output:
Begin
Initialize population of n candidate solutions Xi(I = 1, 2, 3......n)
Initialize α->best agent, β-> second best agent, δ->third best agent
While termination condition not met do
For each candidate wolf do
Update value of current candidate solution
End for
Update α β δ
Calculate fitness value
End while
end

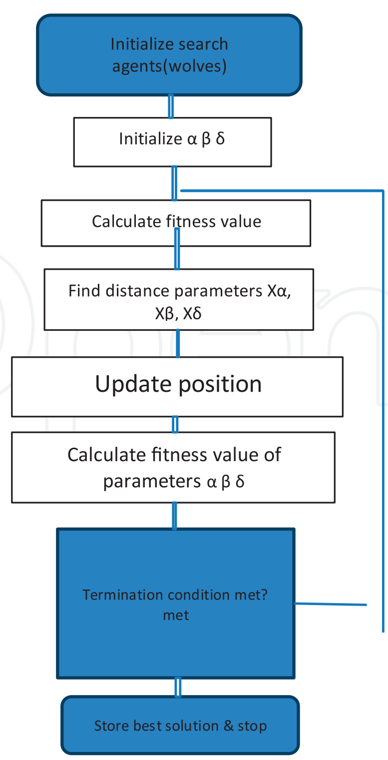


Figure 12.
GWO algorithm & flowchart.

3.10.3 Merits

GWO experience alpha and beta experience is good for complex problems. More heads identification and building are experienced wolves is built. The goodness is identified better through GWO. It is not easy to apply local optima compared to meta-heuristic has lesser parameters. It searches in local search space. The convergence is faster. GWO is easy to implement in any platform. In more iteration avoids local optima and provide higher performance in search problems.

3.10.4 Demerits

In short problem slow convergence and easy premature can be expected. Piece-wise linear cost approximation and update of equation to build local exploration ability are problem in GWO. The accuracy solving can be considered a research challenge. Bad local search ability and slow velocity and falling optimum behaviour and position update are required. Easy falling of velocity and fall into local optimum.

3.10.5 Applications of wolf-based algorithm

GWO algorithm finds application adaption in different domains. Fault system estimation and prediction, hydro-power optimal operation station, Optimization in multi-layer perception. Electronics based domain to find optimal allocation to determine system power loss, link functional net construction by q-Gaussian radial basis, Control operation of DC motors. Fault detection in power systems. Prioritization of problem, selection problem, solve combined economic emission dispatch problem to find optimum allocation. Multi-input and multi-output contingency management problem. Multi-input multi-output contingency management problems and for detection of faulty sections in power systems, to name a few.

4. Comparison of algorithms

Literature Survey reveals complex problems can be resolved in simple steps by applying bio-inspired principles and rules effectively by giving importance to each relationship. The discussed social and population based ten algorithms are involved in processing stages they include,

i. Identification of natural behaviour and responses of biological organism

ii. Replica model to simulate behaviour of biological organism

iii. Translating developed model to mathematical model with certain required assumptions

iv. Pseudocode generation for behaviours of biological organism

v. Experimenting practically and theoretically both models of biological organism for guaranteed performance improvements in real-world problem.

5. Issues, challenges and future direction

This section briefs on bio-inspired algorithm current challenges, issues and further direction for next works to be carried in this direction.

5.1 Literary issues

The database identification was first challenge to identify supporting literature. Scopus a largest database of academic articles was primary focus in collection of articles from journals. The published articles on specific bio-inspired algorithm searched for publication number from 2008 to 2020. During search process name was considered as keyword. Obtained results were analysed for algorithm, document-wise and subject wise. Documents are categories are article, conference paper, review, book and others. More research publication in different categories can be found based on bio-inspired algorithms PSO and GBA compared to other emerging algorithms which is plotted in **Figure 13**. To have clear view of published article year wise plot was plotted as shown in **Figure 14**. The evolution of algorithms is clearly shown in **Figure 14**. More publications have established algorithm PSO and GBA and on other remaining algorithms publications are comparatively low hence more research can be carried on to identify suitable optimization position for this algorithms.

5.2 Challenges

Bio-inspired algorithms face challenges in design of competitive and interactive component design. Biological systems have found lack in information exchange so algorithm has to be developed in absence of data. Improve or develop bio-inspired algorithms to design solution to adapt for any real-world problem. Performance of bio-inspired algorithm is another issue which need to be sorted in working environment.

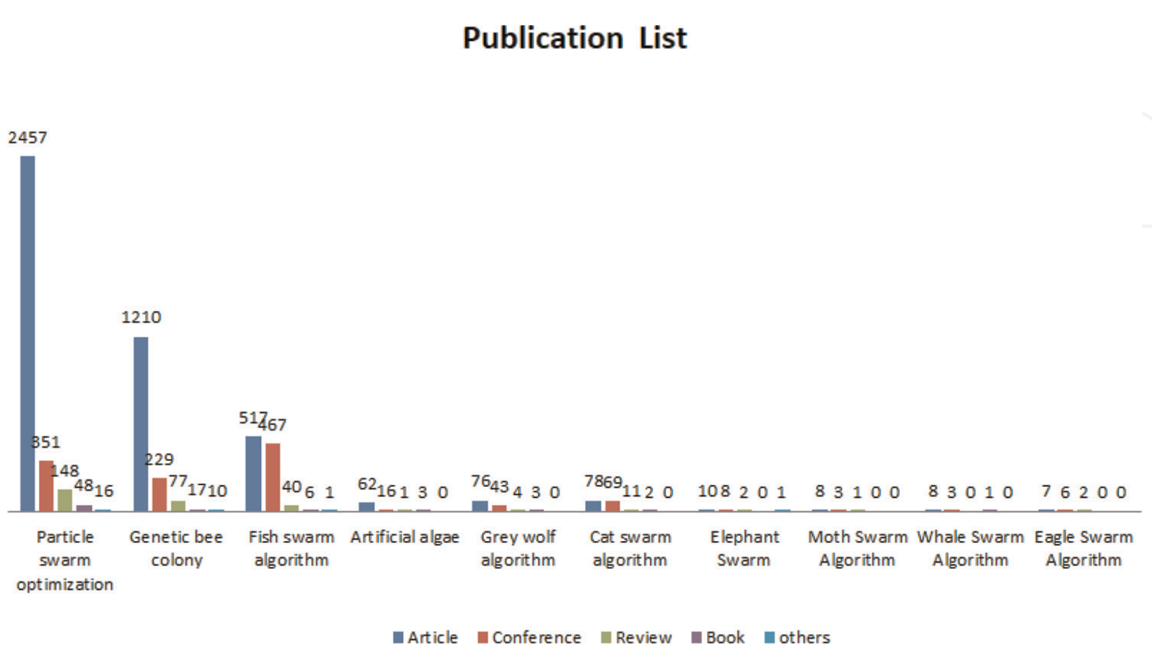


Figure 13.
List of publications on different bio-optimization.

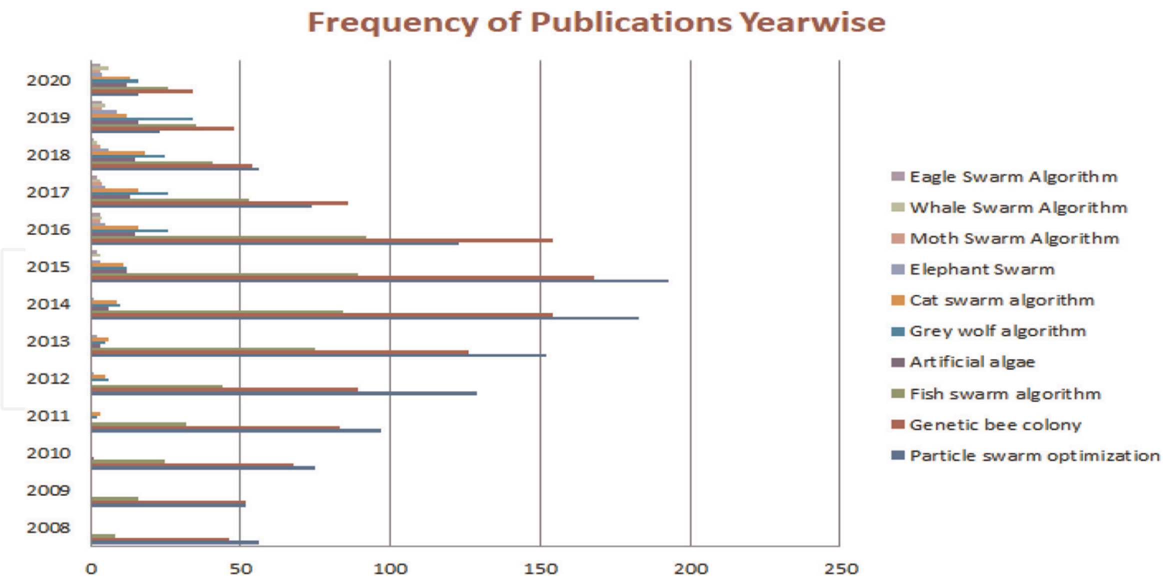


Figure 14.
Frequency of research articles published on various based algorithms till 2021 bio-inspired algorithms year wise.

5.3 Future scope

Bio-inspired algorithms brought revolutionary changes in different domains as well got power to impact further generation computing. The application coverage area is vast compared conventional methods includes modeling, algorithm, engineering and computing. Generally, optimization techniques based on swarm search procedures incorporate random changes and identification and still has capacity grow which is attracting many young researchers. Bio-inspired algorithms still require addressing new technologies along with it by exploring new ways to adopt algorithms. In order to achieve they need to be collaborated with research communities like computer science, biology, artificial intelligence, ecology, quantum and others. Currently, many bio-inspired algorithms exist, and application field is also extensive and obviously work require further exploration,

- -solution for specific application suitability of selection of parameters
- optimization in range and value of parameters.
- theoretical analysis of convergence of algorithm
- new application of bio-inspired algorithms needs to be explored
- identify suitable hybridization of algorithms with function or algorithms either convention or bio-inspired based.

6. Conclusion

Bio-inspired algorithms have got roots in both pure science and engineering domains. Methods and related theories are mature got huge practical potential benefits to provide in different domain problems. To conclude ten bio-inspired algorithms, FSA imitates food search behaviour of fish considering three parameters distance, length, crowd factor among them first two influence function much. WOA whale inspired algorithm has three operators applied to model search, encircle and foraging behaviour of whales. CSO includes two key operations seeking and tracing in computation of optimum solution. AAA control parameters influence whole functionality. MFO accuracy is based on spiral movement towards artificial light. ESA is based on exploration and exploitation in searching. GWO algorithm simulates wolves by dividing into four group; alpha, beta, delta, and omega.

Author details

G. Devika[1]* and Asha Gowda Karegowda[2]

1 Government Engineering College, K. R. Pet, Karnataka, India

2 Siddaganga Institute of Technology, Tumkur, Karnataka, India

*Address all correspondence to: sgdevika@gmail.com

References

[1] Mishra KK, Tiwari S, Misra AK. Bio inspired algorithm for solving optimization problems. In: 2011 2nd International Conference on Computer and Communication Technology (ICCCT-2011). Allahabad, India: Motilal Nehru National Institute of Technology; 2011. pp. 653-659. DOI: 10.1109/ICCCT.2011.6075211

[2] Hammod WA, Zamil K, Ali A. A review of bio-inspired algorithms. In: A Conference: (SOFTEC Asia 2017). Malaysia: Kuala Lumpur Convention Centre; 2017

[3] Das S, Suganthan PN. Differential evolution: A survey of the state-of-the-art. IEEE Transactions on Evolutionary Computation. 2011;**15**(1):4-31

[4] Fan X, Sayers W, Zhang S, et al. Review and classification of bio-inspired algorithms and their applications. Journal of Bionic Engineering. 2020;**17**: 611-631

[5] Kar AK. Bio-inspired computing—A review of algorithms and scope of applications. Expert Systems with Applications. 2016;**5**:20-32

[6] Swarm intelligence based energy efficient clustering algorithms for WSN: A review of algorithms, analysis and scope of applications. Chapter 5, Resource optimization using swarm intelligence and the IoT

[7] Devika G, Ramesh D, Karegowda AG. Analysis of binary and grey wolf optimization algorithms applied for enhancing performance of energy efficient. In: IEEE Proceedings. M S Raimaiah Institute of Technology; 2019

[8] Eftimov T, Korošec P, Koroušić Seljak B. A novel approach to statistical comparison of meta-heuristic stochastic optimization algorithms using deep statistics. Information Sciences. 2017; **417**:186-215

[9] Eftimov T, Korošec P, Koroušić Seljak B. Comparing multi-objective optimization algorithms using an ensemble of quality indicators with deep statistical comparison approach. In: Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI 2017). Hawaii, USA; 2017

[10] Gourtani A, Nguyen TD, Xu H. A distributionally robust optimization approach for two-stage facility location problems. EURO Journal of Computation Optimization. 2020;**8**:141-172

[11] Jabeen H, Jalil Z, Baig AR. Opposition based initialization in particle swarm optimization. In: Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers; New York, NY, USA. 2009. pp. 2047-2052

[12] Zhang C et al. A novelswarm based Quasai oppositional PSO. In: 5th International Forum of Information Technology and Applications. 2009. pp. 325-330

[13] Clarccle C, Kennedy J. A particle swar-explosive, stability and convergence in multi-dimensional complex space. IEEE Transactions on Evolutionary Computation. 2002;**6**:58-73

[14] Imran M, Jabeen H, Ahmad M, Ababs Q, Bangyal W, Ababs Q. Opposition based PSO and mutation operators. In: 2nd International Conference on Education Technology

and Computer. Shanghai; 2010. pp. 506-508

[15] Zhang L, Yu H, Hu S. A new approach to improve particle swarm optimization. In: Proceedings of the 2003 International Conference on Genetic and Evolutionary Computation. 2003. pp. 134-139

[16] Weiand Wang J. A dynamical particle swarm optimization with dimensional mutation. IJCSNS International journal of computer Science and Network Security. 2006;**6**: 221-224

[17] Wang H. A hybrid PSO with Cauchely mutation. In: Proceeding of IEEE Transactions on Swarm Intelligence Symposium. Honolulu, HI, USA; 2007. pp. 356-360

[18] Imran M, Manzoor Z, Ali S, Ababs Q. Modified particle swarm optimization with student T mutation. In: International Conference on Computer Networks and Information Technology. Abbottabad; 2011. pp. 283-286

[19] Hajihassani M, Jahed Armaghani D, Kalatehjari R. Applications of particle swarm optimization in geotechnical engineering: A comprehensive review. Geotechnical and Geological Engineering. 2018;**36**:705-722

[20] Ma R-J, Yu N-Y, Hu J-Y. Application of particle swarm optimization algorithm in the heating system planning problem. Scientific World Journal. 2013; **2013**:718345

[21] Karaboga D. An Idea Based on Honey Bee Swarm for Numerical Optimization. Erciyes: Computer Engineering Department, Erciyes University; 2005

[22] Alshamlan HM, Badr GH, Alohali YA. Genetic Bee Colony (GBC) algorithm: A new gene selection method for microarray cancer classification. Computational Biology and Chemistry. 2015;**56**:49-60

[23] Feng X, Yin J, Xu M, Zhao X, Wu B. The algorithm optimization on artificial fish-swarm for the target area on simulation robots. In: IEEE 2nd International Conference on Signal Processing Systems (ICSPS). Zhenjiang, Jiangsu, China; 2010. pp. 87-89

[24] Alam F, Saadi HS, Alam MS. A novel comparative study between dual population genetic algorithm and artificial bee colony algorithm for function optimization. In: 2016 19th International Conference on Computer and Information Technology (ICCIT). Dhaka; 2016. pp. 333-338

[25] Xiao L. A clustering algorithm based on artificial fish school. In: 2nd International Conference on Computer Engineering and Technology (ICCET). Chennai, India: Contemporary Engineering and Technology; 2010. pp. 766-769

[26] Zhang M, Cheng S, Li F, Gan Y, Sun J. Evolving neural network classifiers and feature subset using artificial fish swarm. In: IEEE International Conference on Mechatronics and Automation. China; 2006. pp. 1598-1602

[27] Zhuang D, Ma K, Tang C, Liang Z, Wang K, Wang Z. Mechanical parameter inversion in tunnel engineering using support vector regression optimized by multi-strategy swarm algorithm. Tunnelling and Underground Space Technology. 2019;**83**:425-436

[28] Yazdani D, Nasiri B, Sepas-Moghaddam A, Meybodi M, Akbarzadeh-Totonchi M. mNAFSA: A

novel approach for optimization in dynamic environments with global changes. Swarm and Evolutionary Computation. 2014;**18**:38-53

[29] Kang C, Wang S, Ren W, Lu Y, Wang B. Optimization design and application of active disturbance rejection controller based on intelligent algorithm. IEEE Access. 2019;7:59862-59870

[30] Feng Y, Zhao S, Liu H. Analysis of network coverage optimization based on feedback k-means clustering and swarm algorithm. IEEE Access. 2020;**8**:42864-42876

[31] Yan L, He Y, Huangfu Z. A fish swarm inspired holes recovery algorithm for wireless sensor networks. International Journal of Wireless Information Networks. 2020;**27**(1):89-101

[32] Du T, Hu Y, Ke X. Improved quantum artificial fish algorithm application to distributed network considering distributed generation. Computational Intelligence and Neuroscience. 2015;**2015**:91

[33] Li X, Keegan B, Mtenzi F. Energy efficient hybrid routing protocol based on the artificial fish swarm algorithm and ant colony optimisation for wsns. Sensors. 2018;**18**(10):3351

[34] Kalaipriyan T, Amudhavel J, Sujatha P. Whale optimization algorithm for combined heat and power economic dispatch. Advances and Applications in Mathematical Sciences. 2017;**17**(1):197-211

[35] He N, Belacel A, Chan H, Hamam Y. A hybrid artificial fish swarm simulated annealing optimization algorithm for automatic identification of clusters. International Journal of Information Technology & Decision Making. 2016;**15**(05):949-974

[36] Sathya DJ, Geetha K. Hybrid ANN optimized artificial fish swarm algorithm based classifier for classification of suspicious lesions in breast DCE-MRI. Polish Journal of Medical Physics and Engineering. 2017;**23**(4):81-88

[37] Fei T, Zhang L. Application of BFO-AFSA to location of distribution centre. Cluster Computing. 2017;**20**(4):3459-3474

[38] Mao M, Duan Q, Duan P, Hu B. Comprehensive improvement of artificial fish swarm algorithm for global mppt in pv system under partial shading conditions. Transactions of the Institute of Measurement and Control. 2018;**40**(7):2178-2199

[39] Yan W, Li M, Pan X, Wu G, Liu L. Application of support vector regression cooperated with modified artificial fish swarm algorithm for wind tunnel performance prediction of automotive radiators. Applied Thermal Engineering. 2020;**164**:114543

[40] Zhu Y, Xu W, Luo G, Wang H, Yang J, Lu W. Random forest enhancement using improved artificial fish swarm for the medial knee contact force prediction. Artificial Intelligence in Medicine. 2020;**103**:101811

[41] Serapiao AB, Correa GS, Goncalves FB, Carvalho VO. Combining K-means and K-harmonic with fish school search algorithm for data clustering task on graphics processing units. Applied Soft Computing. 2016;**41**:290-304

[42] Pei-Wei T, Pan J-S, Chen S-M, Liao B-Y, Hao S-P. Parallel cat swarm optimization. In: International Conference on Machine Learning and Cybernetics. 2008. pp. 3328-3333

[43] Liu Y, Wu X, Shen Y. Cat swarm optimizing clustering: A cat swarm optimization in advances in swarm intelligence. In: Lecture Notes in Computer Science (LNCS 6782). 2011. pp. 321-328

[44] Saha SK, Ghoshal SP, Kar R, Mandal D. Cat Swarm Optimization Algorithm for Optimal Linear Phase FIR Filter Design. ISA Transactions in Press, Elsevier Ltd; 2013

[45] Abdel-Basset M, El-Shahat D, El-henawy I, Sangaiah AK, Ahmed SH. A novel whale optimization algorithm for cryptanalysis in merkle-hellman cryptosystem. Mobile Networks and Applications. 2018;**23**(4):723-733

[46] Ong P. Adaptive cuckoo search algorithm for unconstrained optimization. Scientific World Journal. 2014;**2014**:1-8

[47] Li-Jiang Y, Tian-Lun C. Application of chaos in genetic algorithms. Communications in Theoretical Physics. 2002;**38**:168

[48] Xu Z, Yu Y, Yachi H, Ji J, Todo Y, Gao S. A novel memetic whale optimization algorithm for optimization. In: International Conference on Swarm Intelligence. Cham, Switzerland: Springer; 2018

[49] Pecora LM, Carroll TL. Synchronization of chaotic systems. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2015;**25**:097611

[50] Naik MK, Panda R. A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition. Applied Soft Computing. 2016;**38**:661-675

[51] Thi-Kien D, Pan T-S, Pan J-S. A multi-objective optimal mobile robot path planning based on whale optimization algorithm. In: Signal Processing (ICSP), 2016 IEEE 13th International Conference on IEEE. 2016

[52] Faseela CK, Vennila H. Economic and Emission Dispatch using Whale optimization Algorithm (WOA). In: 2018 in IJECE. 2018

[53] Fong M. A multi-objective optimal vehicle fuel consumption based on whale optimization algorithm. In: Advances in Intelligent Information Hiding and Multimedia Signal Processing: Proceeding of the Twelfth International Conference on Intelligent Information Hiding and Multimedia Signal Processing. Kaohsiung, Taiwan; 2016, 2016

[54] Marler RT, Arora JS. Survey of multi-objective optimization methods for engineering. Structural and Multidisciplinary Optimization. 2004;**26**(6):369-395

[55] Yan Z, Sha J, Liu B, Tian W, Lu J. An ameliorative whale optimization algorithm for multi-objective optimal allocation of water resources in Handan, China. Water. 2018;**10**:87

[56] Mafarja MM, Mirjalili S. Hybrid Whale Optimization Algorithm with simulated annealing for feature selection. Neurocomputing. 2017;**260**:302-312

[57] Trivedi IN, Jangir P, Kumar A, Jangir N, Totlani R. A novel hybrid PSO–WOA algorithm for global numerical functions optimization. In: Advances in Computer and Computational Sciences. Cham, Switzerland: Springer; 2018

[58] Jadhav AN, Gomathi N. WGC: Hybridization of exponential grey wolf optimizer with whale optimization for

data clustering. Alexandria Engineering Journal. 2017;**57**(3):1569-1584

[59] Revathi ST, Ramaraj N, Chithra S. Brain storm-based Whale Optimization Algorithm for privacy-protected data publishing in cloud computing. Cluster Computing. 2018;**21**:1-10

[60] Prakash D, Lakshminarayana C. Optimal siting of capacitors in radial distribution network using Whale Optimization Algorithm. Alexandria Engineering Journal. 2017;**56**(4):499-509

[61] Touma HJ. Study of the economic dispatch problem on IEEE 30-bus system using whale optimization algorithm. International Journal of Engineering, Science and Technology. 2016;**5**(1):1

[62] Tirkolaee EB, Alinaghian M, Hosseinabadi AAR, Sasi MB, Sangaiah AK. An improved ant colony optimization for the multi-trip Capacitated Arc Routing Problem. In: Computers & Electrical Engineering. 2018

[63] Kaveh A, Ghazaan MI. Enhanced whale optimization algorithm for sizing optimization of skeletal structures. Mechanics Based Design of Structures and Machines. 2017;**45**(3):345-362

[64] Oliva D, El Aziz MA, Hassanien AE. Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. Applied Energy. 2017;**200**:141-154

[65] Gupta S, Saurabh K. Modified artificial killer whale optimization algorithm for maximum power point tracking under partial shading condition. In: Proceedings of the 2017 International Conference on Recent Trends in Electrical, Electronics and Computing Technologies (ICRTEECT). Warangal, Telangana, India; 2017. pp. 87-92

[66] Mostafa A, Hassanien AE, Houseni M, Hefny H. Liver segmentation in MRI images based on whale optimization algorithm. Multimedia Tools and Applications. 2017;**76**(23):24931-24954

[67] Karaboga D, Akay B, Ozturk C. Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks. MDAI. 2007;**7**:318-319

[68] Rakitianskaia AS, Engelbrecht AP. Training feedforward neural networks with dynamic particle swarm optimisation. Swarm Intelligence. 2012;**6**(3):233-270

[69] Wang L, Zeng Y, Chen T. Back propagation neural network with adaptive differential evolution algorithm for time series forecasting. Expert Systems with Applications. 2015;**42**(2):855-863

[70] Aljarah I, Faris H, Mirjalili S. Optimizing connection weights in neural networks using the whale optimization algorithm. Soft Computing. 2018;**22**(1):1-15

[71] Uymaz SA, Tezel G, Yel E. Artificial algae algorithm (AAA) for nonlinear global optimization. Applied Soft Computing. 2015;**2015**:153

[72] Liu CP, Ye CM. Solving permutation flow shop scheduling problem by firefly algorithm. Industrial Engineering Management. 2012;**17**:56-59

[73] Fernandez-Viagas V, Framinan JM. A bounded-search iterated greedy algorithm for the distributed permutation flowshop scheduling problem. International Journal of Production Research. 2015;**53**:1111-1123

[74] Mustafa İ, Salih D, Hamid T, Kenan K, Fatih M. Optimization of filter by using support vector regression machine

with Cuckoo search algorithm. Radioengineering. 2014;**23**:790-797

[75] Yang XS, Deb S. Cuckoo search: Recent advances and application. Neural Computing and Applications. 2014;**24**: 169-174

[76] G. Zheng, S. P. Mohanty, E. Kougianos, "Metamodel-assisted fast and accurate optimization of an OP-AMP for biomedical applications", in Proceedings of the 11th IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2012; 273–278

[77] Chifu VR, Pop CB, Salomie I, Suia DS, Niculici AN. Optimizing the semantic web service composition process using cuckoo search. Computational Intelligence. 2012;**382**: 93-102

[78] Yildiz AR. Cuckoo search algorithm for the selection of optimal machine parameters in milling operations. International Journal of Advanced Manufacturing Technology. 2012;**64**: 55-61

[79] Ahmed T, Obaidi S. Improved Scatter Search Using Cuckoo Search. International Journal of Advanced Research in Artificial Intelligence. 2013; **2**:61-67

[80] Naik MK, Maheshwari RN, Wunnava A, et al. A new adaptive cuckoo search algorithm. In: IEEE International Conference on Recent Trends in Information Systems. Kolkata; 2015. pp. 1-5

[81] Burnwal S, Deb S. Scheduling optimization of flexible manufacturing system using cuckoo search-based approach. International Journal of Advanced and Manufacturing Technology. 2012;**64**:954-959

[82] Wang M, Chen H, Yang B, Zhao X, Hu L, Cai Z, et al. Toward an optimal kernel extreme learning machine using a chaotic moth-flame optimization strategy with applications in medical diagnoses. Neurocomputing. 2017;**267**: 69-84

[83] Singh P, Prakash S. Optical network unit placement in fiber-wireless (fiwi) access network by moth-flame optimization algorithm. Optical Fiber Technology. 2017;**36**:403-411

[84] Shehab M, Khader AT, Laouchedi M. Modified cuckoo search algorithm for solving global optimization problems. In: International Conference of Reliable Information and Communication Technology. Springer; 2017c. pp. 561-570

[85] Hassanien AE, Gaber T, Mokhtar U, Hefny H. An improved moth flame optimization algorithm based on rough sets for tomato diseases detection. Computers and Electronics in Agriculture. 2017;**136**:86-96

[86] Faris H, Aljarah I, Mirjalili S. Evolving radial basis function networks using moth–flame optimizer. In: Handbook of Neural Computation. Vol. 28. Elsevier; 2017. pp. 537-550