IFERP
connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 11, November 2017

A Review on Nature-Inspired Swarm Intelligence based Optimization Techniques

[1] Shiv Kumar Agarwal, [2] Surendra Yadav
[1] Research Scholar, [2] Professor
[1][2] Department of Computer Science & Engineering, Career Point University, Kota, Rajasthan, India.

***Abstract*— Human beings are deeply inspired by nature. Nature has the capability to solve very large complex problems in its own classical way. Nature gives some of the logical and effective ways to find solution to these problems. Nature acts as self optimizer for solving the complex problems. These nature-inspired metaheuristic algorithms can be based on swarm-intelligence-based, bio-inspired, physics-based and chemistry-based, depending on the sources of inspiration. Swarm intelligence and bio-inspired algorithms form a hot topic in the developments of novel algorithms inspired by nature. In current decades, meta-heuristic algorithms have been developed to overcome the problem that most of them are inspired from nature. For the past decades, various research efforts have been determined in this particular area. In this paper, the algorithms which are discussed imitate the processes running in nature. Though not all of them are efficient, a few algorithms have proved to be very efficient and thus have become trendy tools for solving real-world problems. In this paper we depict the various Swarm Intelligence Optimization techniques and analysis of these techniques, on the basis of analysis some research issues are given in this paper. This paper also provides the comparison of these techniques and conclusion of the overall paper. This paper proposes the high extent for the development of latest, better and efficient techniques and application in this area. This paper highlights the comparative analysis of nature inspired swarm intelligence based optimization techniques based on literature analysis and the areas where these algorithms have been most successfully applied.**

Keywords— Nature Inspired Algorithm, bio-inspired, Optimization technique, Swarm Intelligence, metaheuristic

I. INTRODUCTION

Every one of us experiences the glory of nature. Nature inspired computing is a technique that is inspired by processes, observed from nature. These computing techniques led to the development of algorithms called Nature Inspired Algorithms (NIA). These algorithms are subject of computational intelligence. Nature inspired computing is the computing which has its foundation in the biological components of the nature i.e., humans and animals. Nature has four powerful features which are basic building blocks are self optimization, self healing, self learning and self processing. Nature as Self optimizer is that it can automatically manage its resources in an efficient manner to meet enterprise need. Nature as self healer is as the components of nature on seeing any problem finds a solution and come out of it. Self learning and self processing are two related terms. As the individuals of nature have the capability to evolve according to the changing environment so in present scenario it is indeed required that computers and their intelligence to learn and involve as per changing conditions and solve highly complex problems as nature does. In reality, the so-called simple nature is very much complex. Technological advancements provided us with computers to carry out complex tasks. To solve Real-world optimization problems, optimization tools have to be used, though there is no guarantee that the optimal solution can be obtained. In addition, new algorithms have been developed to see if they can cope with these challenging optimization problems.

It is very difficult to imitate nature since nominal information is available in direct form. In spite of these technical hitches, researchers have tried to connect the nature with computation and the nature-inspired algorithms have resulted as an outcome of some of the finest research work. This paper discusses the classification of set of nature-inspired algorithms particularly brief explanation of swarm based algorithm. To fulfill this desire, we want our algorithms to adopt the techniques and features from nature and become more effective and efficient too. The purpose of developing such algorithms is to optimize engineering problems. As the world is moving towards industrialization, engineering problems are becoming more and more complex and difficult to optimize. This is because of increasing dimensions, variables, time complexity, space complexity etc. To handle such situation, nature inspired algorithms are designed to optimize numerical benchmark functions [1], multi objective functions and solve NP-hard problems for large number of variables, dimensions, etc.

NIA are mainly categorized into evolutionary algorithms and swarm intelligence based algorithms. Evolutionary algorithms are based on the evolutionary behavior of natural systems e.g. genetic algorithm and differential evolution etc. Swarm intelligence (SI) based algorithms, also called as swarm optimization techniques; optimize the certain problem by mimicking the collective behavior of natural swarms e.g. particle swarm optimization, ant colony algorithm, artificial bee colony algorithm, bacterial foraging algorithm, bat algorithm, cuckoo search, firefly algorithm etc. Therefore, the rest of this paper is organized as follows: Section 2 describes the classification of nature inspired algorithms which are further classified into four categories- Swarm intelligence based,Bio-inspired (not SI-based),Physics and Chemistry based and Other algorithms. Section 3 shows various applications of nature inspired algorithms. Section 4 provides a brief and yet comprehensive list of swarm intelligence based optimization techniques. Finally, Section 5 concludes with some suggestions.'

II. CLASSIFICATION OF NATURE INSPIRED ALGORITHMS

The main objective of nature inspired algorithms is to find global optimal solution for a given problem. Based on the above discussions, Nature inspired algorithms are broadly categorized into four categories based on the fundamental natural process that they possess: swarm intelligence (SI) based, bio-inspired (but not SI-based), physics/chemistry-based, and others. We will summarize them briefly in the rest of this paper. Two key factors common in all nature inspired algorithms are intensification and diversification commonly termed as Exploration and Exploitation. Exploration leads to a random search of a new solution space for finding global optima and exploitation finds local optima in explored solution space. Intense exploration does not give optimal solution while deep exploitation traps an algorithm in local optima. A balance between these two factors is very essential for any nature inspired algorithm. Nature presents many diverse phenomenons. However, we will focus here on swarm intelligence (SI) based algorithms. It is worth pointing out the classifications here are not distinctive as some algorithms can be categorized into different categories at the same time. The categorization depends on the real perspective and inspirations. Therefore, the categorization here is just one probable attempt, though the stress is placed on the sources of inspiration.

2.1 Swarm intelligence based

Swarm intelligence (SI) concerns the collective, emerging behavior of multiple, interacting agents who follow some simple rules. While each agent may be considered as unintelligent, the whole system of multiple agents may show some self-organization behavior and thus can behave like some sort of collective intelligence. Many algorithms have been developed by drawing inspiration from swarm-intelligence systems in nature. Swarm intelligence can be described as the collaborative conduct of a group of animals, especially insects such as ants, bees and termites, that are each following very basic rules but when seen in the field of computer science, swarm intelligence is a simulated way to problem solving using algorithms formed on the concept of self managed collective behavior of social insects. Swarming has been defined “distributed problem solving devices inspired by the collective behavior of social insect colonies and other animal societies.” Swarm intelligence (SI) is a relatively novel field of nature-inspired algorithms for multi-agent search and optimization. They usually use decentralized controls to coordinate and self-organization. The behaviors of a single animal such as ant, bee, termite, fish, or wasp often are too simple, but their swarm and social behavior is superior matter like smart population. All SI based algorithms use multi-agents, inspired by the cooperative actions of social insects like ants, bees, as well as from other animal societies like flocks of birds or fish. SI systems are typically made up of a population of simple agents that everyone interacts locally with another and with their environment [2]. In SI there is no centralized control unit to dictate how individual agents should behave; but local interactions between such agents often lead to the emergence of global behavior and match their position or speed with regards to the new situations.

Bonabeau et al., 1999 [3] gave a simple definition for better understanding of swarm intelligence i.e., "the popular way of simple and common intelligence of social agents". Swarm intelligence is an emerging new domain that visualizes intelligence as a method of communication between independent agents. The classical particle swarm optimization (PSO) uses the swarming behaviour of fish and birds, while firefly algorithm (FA) uses the flashing behaviour of swarming fireflies. Cuckoo search (CS) is based on the brooding parasitism of some cuckoo species, while bat algorithm uses the echolocation of foraging bats. Ant colony optimization uses the interaction of social insects (e.g., ants), while the class of bee algorithms are all based on the foraging behaviour of honey bees. SI-based algorithms are amongst the most accepted and extensively used. There are many reasons for such popularity; one of the reasons is that SI-based algorithms usually sharing information among multiple agents, so that self organization, co-evolution and

IFERP
connecting engineers... developing research

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 11, November 2017

learning during iterations may help to provide the high efficiency of most SI-based algorithms. In this paper, the authors discussed some of the swarm intelligence optimization techniques, which are more effective for solving linear/nonlinear problem compared to other techniques.

2.2 Bio-inspired (not SI-based)

In true sense, bio-inspired algorithms form a majority of all nature-inspired algorithms. From the set theory point of view, SI-based algorithms are a subset of bio-inspired algorithms, while bio-inspired algorithms are a subset of nature-inspired algorithms. That is SI-based ⊂ bio-inspired ⊂ nature-inspired On the other hand, not all nature-inspired algorithms are bio-inspired, and some are wholly physics and chemistry based algorithms as we will see in next section. Many bio-inspired algorithms do not use directly the swarming behaviour. Therefore, it is superior to call them bio-inspired, but not SI-based. However, it is not easy to classify certain algorithms such as differential evolution. Thoroughly speaking, differential evolution is not bio-inspired because there is no direct link to any biological behaviour. On the other hand, as it has some similarity to genetic algorithms which are bio-inspired, but not SI-based and also has a key word 'evolution', we uncertainly put it in the category of bio inspired algorithms.

2.3 Physics and Chemistry based

Not all algorithms are bio-inspired, because their sources of inspiration often come from physics and chemistry. For the algorithms that are not bio-inspired, most have been developed by imitating certain physical and/or chemical laws, including electrical charges, gravity, river systems, etc. As unlike natural systems are relevant to this class, we can even subdivide these into many subcategories like physics and chemistry based algorithms. After all, many fundamental laws are the same. So we simply group them as physics and chemistry based algorithms.

2.4 Other algorithms

While researchers develop new algorithms, some may look for initiative away from nature. Therefore, some algorithms are not bio-inspired or physics/chemistry-based; it is sometimes difficult to put some algorithms in the above categories, because these algorithms have been developed by using various uniqueness from dissimilar sources, such as social, emotional, etc. In this case, it is better to put them in the other category.

TABLE I: EXAMPLES OF SWARMING BEHAVIOR IN NATURE

Swarming Behavior	Entities
Path formatting	Ants
Food source selection	Ants, Bees
Flocking	Birds
Cooperative transport	Ants
Brooding	Cuckoo
Path optimizing	Natural rivers
Schooling	Fish
Synchronization	Fireflies
Task allocation	Wasps
Herding behavior	Krill individuals
Echolocation	Bat
Pattern generation	Bacteria
Web construction	Spiders
Thermo regulation	Bees
Law of gravity	Mass interactions
Nest sorting	Ants
Hive Construction	Bees, Wasps, Hornets
Prey Surrounding	Wolves

III. APPLICATIONS

Nature-inspired algorithms have been applied to a wide range of diverse applications; solve various optimization problems in real-world applications across domains. An optimization problem is the problem of finding the best solution from all possible solutions. Optimization problems can be divided into two categories depending on whether the variables are continuous or discrete [4]. Classification of optimization algorithm can be carried out in many ways. A straightforward way is to look at the nature of the algorithm, and this divides the algorithm into two categories: deterministic algorithms, and stochastic algorithms. Deterministic algorithms pursue a rigorous process, and its path and values of both design variables and the function are repeatable. On the other hand, stochastic algorithms always have some randomness and every individual path towards a possible solution is not exactly repeatable. Optimization problems are classified according to the mathematical characteristics of the objective function, the constraints and the control variables. The most significant characteristic is the nature of the objective function. Major application areas of optimization are Hard problems, Telecommunications, Image processing, Engineering design, Vehicle routing.

A classic example of a hard problem is the Travelling Salesman Problem, which has been solved using metaheuristic approaches such as ant colony optimization. Nature-inspired algorithms have also been used to optimize local access networks to maximize quality of service and minimize overall energy consumption. Image processing often concerns time-consuming computational tasks. When traditional techniques are combined with nature-inspired algorithms, features can be extracted more accurately for many applications Many engineering design problems are highly nonlinear—such as structural design and wireless sensor networking—and traditional methods do not handle such nonlinearity well. Modern studies show that nature-inspired algorithms can frequently produce better design options more competently because they use landscape information to search design-space regions and share information among different agents. The use of metaheuristic approaches was shown to be more effective than traditional algorithms in solving vehicle routing problems and transport costs were lower. The scheduling of aircraft, departure slot allocation, and airspace management can also be solved satisfactorily by nature-inspired metaheuristics, with solutions leading to reduced running costs and more effective use of departure slots.

IV. SWARM INTELLIGENCE BASED OPTIMIZATION TECHNIQUES

4.1 Ant Colony Optimization

Ants are social insects, living in organized colonies, which can have as many as 25 million ants in each colony. Ant colony optimization algorithm takes its inspiration from the real world of ant colonies to solve optimization problems. Ant colony optimization that is a probabilistic technique to solve complex problems is initially proposed by Marco Dorigo in1992 [5], [6] originally was applied to travelling salesman problem and then applied later to various hard optimization problems. ACO that can reduce finding best and shortest paths through the graphs is based on ant behavior for seeking a path to achieve the source of food. For finding food, ants start out from their colony and move randomly in all directions. Once a ant find food, it returns to colony and leave a trail of chemical substances called pheromone that shows the trace of an ant [7] that permit them to communicate to each other. To find the best way to food, ant employs heuristic information. Other ants of the swarm can sense pheromone trails and move on the same path.

They leave the nest and move randomly to find food but when they find a pheromone trail that made by other ants, they decide whether or not to follow it. If they decide to follow it they make own pheromones over the trail. Quality of pheromone in a path makes more chance to the path to be selected by ant over the other paths and gradually the amount of pheromone on the path would be highlighted among the others. The interesting point is that how often the path visit by ants is determined by the concentration of pheromone along the path. Since pheromone will naturally disperse over time, the length of the path is also a factor. Therefore under these conditions, a shorter and best path will be chosen because ants moving on that path keep adding pheromone to it which makes the concentration strong enough against evaporation. Pheromone concentration in each path represents the quality of solution (goodness of fitness value). The process is continued until stopping criteria is met. As a result, the shortest and best path from colony to food emerges. ACO is a simulation of the colony of ant to find the shortest path as shown in the Fig.1.

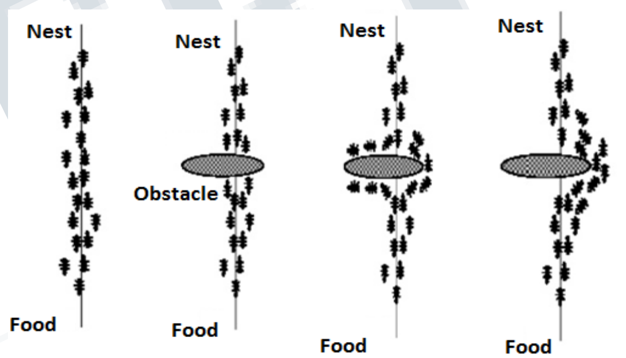

Fig.1. Ants' behavior to find food

In the past decades substantial amount of research has been done to both develop the ACO algorithm itself and practical applications to solve the relevant problems in the real world. The initial ACO algorithm, Ant System (AS), was proposed by Marco Dorigo in 1992 in his PhD thesis [6]. Dorigo and Gambardella introduced Ant Colony System (ACS) as a variant of AS in 1997[8]. In parallel, Stützle and Hoos invented the MAX-MIN Ant System in 1996[9].

The ACO meta heuristic was developed (Dorigo & Di Caro, 1999 ;) to generalize, the overall method of solving combinatorial problems by approximate solutions based on the generic behavior of natural ants. In the early twenties, Iredi et al. published the first multi-objective algorithm which is a very popular extension to the original ACO algorithm [10].

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 11, November 2017

The ant colony optimization is graph based technique which was used to get the optimum solutions in various applications. Initially it was used to solve NP hard problems like traveling salesman, job shop scheduling problems etc. Later on it was used to solve many engineering problems like Face identification systems, Vehicle routing Prob¬lem, detecting optimal path for routing in WSN etc.

4.2 Artificial Bee Colony Optimization

Artificial bee colony (ABC) is a swarm-based meta-heuristic algorithm that introduced in 2005 by Karaboga [12] and opens a new direction in the field of optimization algorithms in complex problems. It is the most popular algorithm based on bee colony optimization. Artificial bee colony (ABC) Algorithm is an optimization algorithm based on the intelligent behavior of honey bee foraging. The term artificial bee is used because the behavior of actual bee is quite different from the behavior of bees assumed in ABC algorithm. As the ants search their food using pheromones, similar food collecting behaviors are found in honey bees. Instead of pheromones, the algorithm relies on the foraging behavior of honey bees.

The first step in this process is to sending bees in different directions to search for good quality food. When the bees have located the desired food source they return to their colony and tell other bees about the food source using a technique known as waggle dance. This dance tells other bees about three information i.e., distance of food source from colony, the direction in which all other bees have to go and last the quality of food source. The bees are attracted to that bee only which has brought information about the best quality food source. This process gives the best food source. (ABC) Algorithm is based on inspecting the behaviors of real bees on finding nectar amounts and sharing the information of food sources to the other bees in the hive. The algorithm divides bees into employed, scout, and onlooker bees. These specialized bees try to maximize the nectar amount stored in the hive by performing efficient division of labour and self-organization [13]. The employed bees are associated with the specific food sources, onlooker bees watching the dance of employed bees within the hive to choose a food source, and scout bees searching for food sources randomly [14]. The onlooker bees and the scout bees are the unemployed bees. The Employed bees which are selected to search for food source, Onlooker bee that follows the paths provided by best employed bee and The Scout bee to found the lowest quality of food source and is now responsible for finding new food, the new food sources.

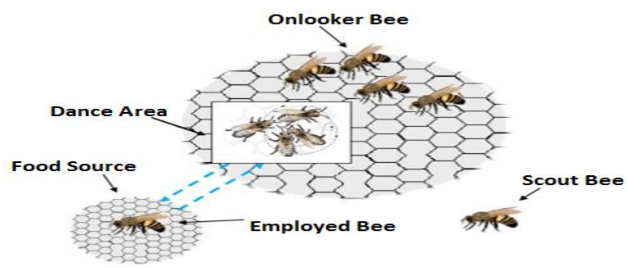

Fig.2. Bees' behavior to find food

Initially, the scout bees discover the positions of all food sources, thereafter, the job of the employed bee starts. An artificial employed bee probabilistically obtains some modifications on the position in its memory to target a new food source and find the nectar amount or the fitness value of the new source. Later, the onlooker bee evaluates the information taken from all artificial employed bees and then chooses a final food source with the highest probability related to its nectar number. If the fitness value of new one is higher than that of the previous one, the bee forgets the old one and memorizes the new position [15]. This is called as greedy selection. Then the employed bee whose food source has been exhausted becomes a scout bee to search for the further food sources once again. Hence optimized solution is the food source having highest nectar amount. The algorithm terminates when desired solution is achieved.

The Artificial Bee Colony optimization differs from a real Bee Colony Optimization because in ABC we use only scout bees and onlooker bees in equal quantity as initial set of solution (population). ABC based on approach of examining the behaviors of real bees on finding food and shares this information of food sources to the bees in the area. This behavior can be applied to many complicated engineering problems including computational, control, optimization, transportation, etc. This algorithm solves multidimensional and multimodal optimization problems. It has been used to solve unconstrained numerical optimization problems and constrained optimization problems as well as to train neural networks.

ABC algorithm works in this way:

1. Initialization: Create an initial set of solutions in which scout bess will search for optimal solution.

2. *Fitness Evaluation:* Scout bees which are chosen to search the set for optimal solution, visit sites, and then the fitness value is evaluated based on some criterion for those visited sites.
3. *Evaluating Best Value:* The bees which have higher fitness value are chose as "selected bees" and the sites which they visit are chosen for neighborhood search.
4. *Iteration:* If the optimal solution is not found, the remaining scout bees are sent to search for another solutions near to those sites which are chosen as best sites.

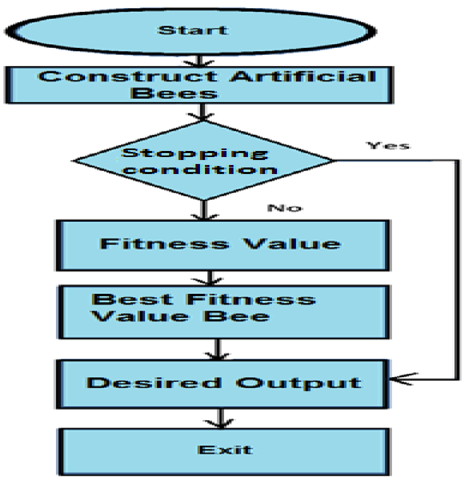

Fig.3. Flow chart of ABC algorithm

4.3 Bacterial Foraging Optimization

The bacterial foraging optimization algorithm [16] is a computational intelligence and Meta heuristics type. It was first developed by Kevin M. Passino in 2002 [17]. BFA is a newly introduced evolutionary optimization algorithm that mimics the foraging behavior of Escherichia coli bacteria [18]. According to the gradient of chemicals in the environment, bacteria seek direction for food. Bacteria can tumble or swim according to its flagella [17]. An E. coli bacterium moves using a pattern of two types of movements: tumbling and swimming. Tumbling refers to a random change in the direction of movement, and swimming refers to moving in a straight line in a given direction. A bacterium in a neutral medium alternates between tumbling and swimming movements. Using its swimming behavior bacteria moves in different directions in order to get food. These Bacteria cells are treated like agents. Depending on the agent to agent interactions, it may form a group or it may ignore each other.

There are three main steps followed by bacteria to achieve global optimum solution: chemotactic step, reproduction step, elimination and dispersal step. In chemotactic step of bacterial foraging optimization algorithm, bacteria swims in direction of high nutrient surface while they tumble when noxious surface is encountered. Objective of BFOA is to minimize the cost of bacteria's movement in high nutrient surface. At the end of first step, all bacteria are arranged in descending order of their fitness value. In reproduction step, first half of the bacteria with high cost die as they did not get sufficient nutrient to survive, while other half (bacteria getting sufficient amount to nutrient) are split into two parts thereby maintaining constant population size. In last step, bacteria get dispersed into complete surface thus controlling them to get trapped in local optima. The newly produced bacteria occupy position of eliminated bacteria. The bacteria with best fitness value, that is, minimum cost finally represent the solution to an objective function. The process continues till desired number of generations gets exhausted. There are successful applications of bacterial foraging optimization in optimization problems such as economic load dispatch and power systems.

4.4 Bat algorithm

Bat algorithm was introduced by Yang in 2010 [19]. Bat algorithm is inspired by the research on the social behavior of bats. It simulates the echolocation behavior of microbats as microbats can generate high echolocation for navigation. Microbats use echolocation to detect prey, avoid obstacles, and locate their roosting crevices in the dark. The Bat produces a very high sound to detect its prey which echoes back with some frequency. Echolocation is a process of detecting an object by reflected sound. It is used to know how far the prey is from background object. By observing the bounced frequency of sound, bats are able to make a distinction between the prey and obstacle and can sense the distance between them in their nearby surroundings. They fly randomly with some velocity, loudness (sound) and frequency to search for food. When a bat finds an insect and is homing in on its prey, the pulse emission rate can accelerate to 200 pulses per second with a higher frequency. Echolocation allows the bats to more accurately measure a flying insect's size, position, range, speed, and direction.

Solution of objective function is to find prey at minimum distance. The frequency and zooming parameters maintain the balance between exploration and exploitation processes. The algorithm continued till convergence criteria are satisfied. The bat algorithm uses characteristics of pulse emission and frequency tuning and considers the bat's location as a solution vector in the search space. Among the entire bat group, there is a global best solution, and other bats tend to cloud toward it. Consequently, convergence is relatively fast, controlled by frequency tuning, pulse emission, and loudness. The bat algorithm has been applied in many real-world applications such as engineering optimization, image processing, training neural networks, and solving the travelling salesman problem.

4.5 Cuckoo Search Algorithm

Most of the birds care a lot until the development of young ones to the bird right from laying eggs. However, there are birds so as certain species, like cuckoo that depends on other birds for raising their young ones. This process of relying on other organism to raise their young ones is called brood parasitism. Brood parasitism of three types namely Intra specific, Cooperative breeding and nest takeover. Cuckoo search algorithm is a metaheuristic algorithm developed by Xin-She Yang and Suash Deb in 2009 [20], considers a cuckoo's egg as a solution vector and the nesting field as the search space. Cuckoo search algorithm (CSA) is inspired by breeding behavior of cuckoo bird. They select their home nest by randomly taking over the nest of some other birds for reproduction. They lay their eggs in selected nest of host bird and drop the host bird's egg. The host bird either drop cuckoo bird's egg or discard the whole nest. Some female cuckoo can imitate their eggs like host bird's egg and lay their eggs just before the laying of host bird's egg. This increases the probability of their chick survival.

In addition to this, cuckoo even takes at most care so that all its young ones are properly raised by the host. Soon after expelling eggs of host bird the first cuckoo chick comes out. Thus it ensures greater share of food to the cuckoo chicks by host mother birds. The other most interesting feature of a cuckoo chick is that it can mimic the call of host chicks. The brood parasitism of cuckoo which is used to find the best nest for his younger ones to be bring up as bird. Each egg in nest represents one solution and cuckoo bird's egg represents a new solution. Fitness for each solution is computed and nest with high quality of eggs (best fitness value) represents the best solution [21].

The process is continued unless global optimum solution is achieved. Initially they generate the population of cuckoo, after that select the best nest which depends on fitness value to lay the eggs. The nest with the worst fitness is abandoned and the remaining is ranked to find the optimum solution. The cuckoo search algorithm has been successfully applied to engineering optimization and image-processing problems.

4.6 Firefly Algorithm

Fire flies, also called glow worms or lightning bugs are found all over the world. There are around two thousand firefly species and most fireflies produce short and rhythmic flashes of light. The pattern of flashes is often unique for a particular species. The fundamental functions of such flashes are to attract mating partners and preys. In today's world, more promising swarm optimization technique is Firefly Algorithm [22], which was introduced by yang in 2008. The algorithm was motivated by mimicking the flashing behavior of fireflies for the purpose food achievement [23]. It uses a randomization by searching for a set of solution so it belongs to a stochastic algorithm. In this algorithm, he incorporated the nature of real fireflies. The fireflies [22] are one type of insects, which produces flash light with different color. Each firefly produces flash light with different intensity. They use flashing for communication, and to attract other flies for mating. Each firefly is attracted towards other firefly. This attraction is represented by their brightness, which increases or decreases depending on distances between the flies. Because light intensity decreases with distance and increased air pollution, the flashing light is visible only a few hundred meters away. The air absorbs light, which becomes weaker and weaker as the distance increases. For all fireflies, the light intensity (brightness) of each firefly is compared with other firefly. Low light intensity flies move towards high light intensity, thereby decreasing the distance and updating its own brightness.

The algorithm is formulated by assuming (i) All fireflies are unisexual, so that one firefly will be magnetized to all other fireflies. (ii) Attractiveness is proportional to their brightness, and for any two fireflies, the less bright one will be attracted by (and thus move to) the brighter one; however, the brightness can decreases as the distance between them increases. (iii) If there are no fireflies brighter than a given firefly, it will move randomly. The brightness is associated with the objective function and the associated constraints along with the local activities carried out by the fireflies [24]. The firefly with high brightness and least distance is the best solution of an objective function. Because short-distance attraction is stronger than long-distance attraction, the whole firefly algorithm can be automatically subdivided into subgroups

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 11, November 2017

or subswarms, and each subswarm will swarm around a local model. The main advantages of fireflies algorithm are its subdivision of population automatically and finding the best global solutions when the size of A position vector and a velocity are nominated to each particle and they are adjusted in any iteration with regards to local and global best found in the whole swarm. subdivision population has larger size compare to number of modes. It is a population based metaheuristic algorithm and used in nonlinear multimodal optimization in dynamic environment.

4.7 Particle Swarm Optimization

Particle swarm optimization (PSO) is one of the swarm-based global optimization algorithms that can move particles (as solutions) through feasible problem space to find the new optimum solutions. Initially, PSO inspired from flocks of birds, schools of fish, and even human social behavior. This nature-based meta-heuristic algorithm was developed by J. Kennedy and R. Ederhart in 1995 [25]. PSO adopts the nature's general behavior, rapidly changing movements and interaction among social agents such as birds or fishes. This algorithm uses a number of agents (particles) that constitute a swarm moving around in the search looking for the best solution. In this optimization technique, individual particles of a swarm is represented as potential solutions, which budge through the problem search space for finding an optimal or the best solution for the problem. PSO mimics the flocking behavior of birds. The birds fly in a solution space and their flocking behavior determines the optimum solution. They follow some path to reach their food destination. The shortest path followed by a bird is considered to be local or particle best solution. Particles tend to move towards its local best position (solution) (lbest) found by them so far. They also keep the track of global best (gbest) solution, the best (shortest) path found by any particle at particular instance. When flying through the solution space all particles try to follow the current optimal particles. A position vector and a velocity are nominated to each particle and they are adjusted in any iteration with regards to local and global best found in the whole swarm.

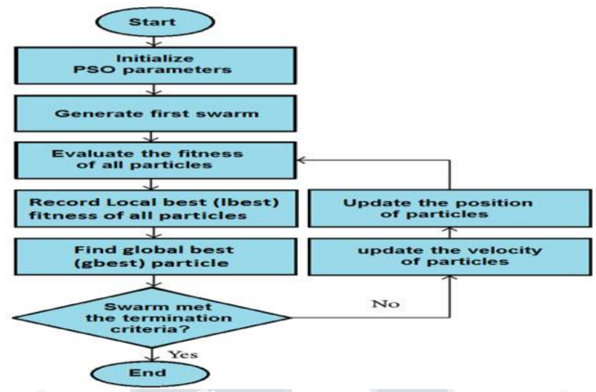

Fig.4.Flow chart of PSO Algorithm

In optimization science, PSO is computational algorithm that tries to improve a candidate feasible solution with attention to a given amount of quality. Then, move these particles around the search space regarding the candidate's solution's position and velocity. Each particle is associated with a velocity, through which it gets accelerated towards local and global best path, the position in 'n' dimension space and the current position of particle with respect to gbest and lbest. Birds communicate with each other to find the most optimum (best) path to reach its food sources. Hence, they learn from the experience of their local best solutions and global best solutions. The algorithm continues till global optimum solution is achieved.

TABLE II: Comparison of Swarm Intelligence Techniques

S.No	Algorithm	Proposed By	Principle	Advantages	Disadvantages	Application Area
1	Ant Colony Optimization (ACO)	Marco Dorigo 1992	works on foraging behaviour of the ant for the food	It is useful in solving optimization problems and also helps in solving the Clustering Problems, NP Hard Problems.	It consumes more energy and It is only for Global Optimization problems not for local.	traveling salesman, job shop scheduling problems Face identification systems, Vehicle routing Problem, detecting optimal path for routing in WSN

2	Artificial Bee Colony Algorithm (ABC)	Karaboga and Basturk 2005	works on foraging behaviour of a honey bee	Easy to implement, used to improve software cost estimation problem. It has fast convergence speed.	Exploitation process of bees is poor, and bees have no knowledge about the location of food	train neural networks, medical pattern classification and clustering problems, solving TSP
3	Bacterial Foraging Algorithm (BFOA)	Passino 2002	works on behaviour of bacteria	It is useful in parallel distributed processing, and global optimization.	Very specific type of constraint handling mechanism and the high number of parameter values	Job Scheduling ,machine learning, face recognition, Image Edge detection, Image Segmentation
4	Bat Algorithm (BA)	Yang 2010	works on echolocation behaviour of microbats	It is very easy to implement algorithm and it produces reliable results	Accuracy may be limited if the number of function evaluations is not high	Continuous & combinatorial optimization, classification, clustering and data mining, fuzzy logic, and image processing,
5	Cuckoo Search Algorithm (CSA)	Yang and Deb 2009	works on breeding behaviour of a cuckoos	Easy to implement and simple, can deals with multi criteria optimization problem	Algorithm may be efficient for a specific class of optimization problems but may not be for some other class of optimization	Spring design optimization, Welded Beam Design, software testing and data generation, wireless sensor network, Knapsack problems, train neural network
6	Firefly Algorithm (FFA)	Yang 2008	works on lighting behaviour of firefly	It can deal with highly non-linear, multi-modal optimization problems naturally and efficiently.	getting trapped into several local optima, parameters are set fixed and they do not change with the time, does not memorize any history of better situation	Digital Image Compression and Image Processing, Feature selection and fault detection, trail neural network, Classification and Clustering,
7	Particle Swarm Optimization (PSO)	Kennedy 1995	Works on flocking behavior of birds.	It gives potential solution of the problem, overlapping is not done. It helps to detect the brain tumor through magnetic resonance imaging	It is not suitable for local solutions and also for non coordinate systems	train a neural network system design , multi objective optimization, classification, pattern recognition and image processing , image clustering, robotic applications, decision making, simulation and identification, etc.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 11, November 2017

V. CONCLUSION

The sources of inspiration for algorithm development are very diverse, and accordingly the algorithms are equally diverse. The nature inspired algorithms are flexible and work in changing environment to organize and grow accordingly. In this paper, we have briefly summarized all the algorithms into four categories. This can be a comprehensive source of information to form a basis or starting point for further research. This paper provides an overview of a range of nature inspired swarm Intelligence based optimization techniques. Usually speaking, almost all of the SI algorithms perform with heuristic population-based search procedures that incorporate random variation and selection. These algorithms have the ability to self learn, self train, self organize and self grow. They can find best optimal solutions to complex problems using simple conditions and rules of nature. Therefore, our aim is to inspire more research to gain better insight into efficient algorithms and solve large-scale real-world problems. So researchers suggest that, the concept of hybrid algorithms to overcome the shortcomings which occur in various individual algorithms. Hybrid algorithms take the best features of the individual algorithms, combine with each other and give the more effective results than the individual algorithms. It has been observed that the applications and growth of natural computing in the last years is very extreme and has been applied to many optimization problems in computer networks, data mining, bioinformatics, control systems, biometrics, power systems, image processing,

REFERENCES

[1] M. Molga and C. Smutnicki, “Test functions for optimization needs”, kwietnia, 2005.

[2] Dyke Parunak and S. Brueckner, "Engineering swarming systems,"Methodologies and Software Engineering for Agent Systems, pp.341-376, 2004.

[3] Bonabeau, E., Dorigo, M. and Theraulaz, G.1999: Swarm intelligence. Oxford University Press.

[4] A. Khanna and A. Mishra "A literature based survey on swarm intelligence inspired optimization technique," International Journal of Advanced Technology in Engineering and Science Volume No 03, Special Issue No. 01, March 2015.

[5] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic," 1999.

[6] M. Dorigo, "Optimization, learning and natural algorithms," Ph. D. Thesis, Politecnico di Milano, Italy, 1992.

[7] R. Beckers, J. L. Deneubourg, and S. Goss, "Trails and U-turns in the selection of a path by the ant Lasius niger," Journal of theoretical biology, vol. 159, pp. 397-415, 1992.

[8] Dorigo M, Gambardella L M. Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation, 1997, 1, 53–66.

[9] Stützle T, Hoos H H. Max-Min ant system, future generation computer systems. Future Generation Computer Systems, 2000, 16, 889–914,

[10] Iredi S, Merkle D, Middendorf M. Bi-criterion optimization with multi colony ant algorithms, evolutionary multi-criterion optimization. First International Conference EMO 2001, Zurich, Switzerland, 2001, 359–372.

[11] Manish Dixit,Nikita Upadhyay and Sanjay Silakari “An Exhaustive Survey on Nature Inspired Optimization Algorithms,” International Journal of Software Engineering and Its Applications Vol. 9, No. 4 (2015), pp. 91-104

[12] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, Journal of Global Optimization 39 (2007) 459–471.

[13] Artificial bee colony (abc), harmony search and bees algorithms on Numerical optimization D. Karaboga, b. Akay Erciyes University, the dept. Of computer engineering, 38039, melikgazi, kayseri, turkiye.

[14] Karaboga, D. Artificial Bee Colony Algorithm. Scholarpedia 2010, 5, 6915.Availableonline: http://www.scholarpedia.org/article/Artificial_bee_colony_algorithm/ (accessed on 27 May 2011).

[15] Chaotic Bee Swarm Optimization Algorithm for Path Planning of Mobile Robots Jiann-Horng Lin and Li-

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 11, November 2017

Ren Huang Department of Information Management I-Shou University, Taiwan 2009.

[16] Das S, Biswas A, Dasgupta S, Abraham A. Bacterial forag¬ing optimization algorithm: theoretical foundations, analy¬sis, and applications. 2009; 203(1):23–55.

[17] Thomas. Survey of Bacterial Foraging Optimization Algo¬rithm. IJISME. 2013 Mar; 1(4):11–2. ISSN: 2319-6386.

[18] Raghavendra V. Kulkarni, and Ganesh Kumar Venayagamoorthy, "Bio-inspired Algorithms for Autonomous Deployment and Localization of Sensor Nodes", PART C: APPLICATIONS AND REVIEWS(to appear in issue 5, volume 40, 2010)

[19] X. S. Yang, "Bat algorithm: literature review and applications", Int. J. Bio-Inspired Computation, Vol. 5, No. 3, 2013, pp. 141–149.

[20] X. S. Yang and S. Deb, "Engineering Optimisation by Cuckoo Search", Int. J. Mathematical Modelling and Numerical Optimisation, Vol. 1, No. 4, 2010, pp. 330–343.

[21] X. S. Yang and S. Deb, "Cuckoo search: recent advances and applications," Neural Computing and Applications, vol. 24, no.1, 2014, pp. 169–174.

[22] Fister I, Fister IJR, Yang XS, Brest J. A comprehensive re-view of firefly algorithms swarm and evolutionary compu¬tation. Elsevier; 2013 Dec; 13(13):34–46.

[23] X. S. Yang, "Firefly Algorithm, Stochastic Test Functions and Design Optimisation", Int. J. Bio-Inspired Computation, Vol. 2, No. 2, 2010, pp.78–84.

[24] Xin-She Yang, Chaos-Enhanced Firefly Algorithm with Automatic Parameter Tuning, International Journal of Swarm Intelligence Research, December 2011.

[25] J. Kennedy and R. Eberhart, "Particle swarm optimization", IEEE international conference on neural networks, 1995, pp. 1942–1948.