Modified cuckoo search algorithm for fast convergence

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering

In

Computer Science

Submitted by:

Pratibha

(Roll No. 801532042)

Under the Supervision of:

Mr. Vinay Arora

(Assistant Professor)

Computer Science Engineering and Department

Thapar University

Patiala – 147004

June 2017

Certificate

I hereby certify that the work which is being presented in the thesis report entitled, **"Modified cuckoo search algorithm for fast convergence"**, submitted by me in partial fulfillment of the requirements for the award of the degree of Master of Engineering in Computer Science and Engineering at Computer Science and Engineering department of Thapar University, Patiala is an authentic record of my own work carried out under the supervision of *Mr. Vinay Arora* and refers other researcher's work which are duly listed in reference section.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Pratibha

(Pratibha)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Vinay Arora

(Mr. Vinay Arora)
Assistant Professor
CSED, Thapar University
Patiala

ACKNOWLEDGEMENT

I express my sincere and deep gratitude to my guide Mr. Vinay Arora, Assistant Professor, Computer Science and Engineering and Department, Thapar University, Patiala for the invaluable guidance, support and encouragement. He provided me all the resources and guidance throughout the thesis work.

I am also thankful to Dr. Maninder Singh, Head of department, Computer Science and Engineering Department for his kind help and cooperation. I express my gratitude to all the staff members of Computer Science and Engineering Department for providing me all the facilities required for the completion of my thesis work.

I would also like to express my appreciation to my good friend for the timely motivation and providing interesting environment. It was great pleasure in working with them during the thesis work. Last but not the least I would like to thank God and my parents for not letting me down at the time of crisis and showing me the silver lining of the dark clouds.

Pratibha

(Pratibha)

(801532042)

Abstract

In recent years, several meta-heuristic optimization techniques have been generated. Recently introduced Cuckoo Search Algorithm (CSA), has proven its outstanding capabilities for solving optimization problems, such as increased convergence rate and greater global minimum values. An optimization algorithm termed as CSA is inspired by the lifespan of a Cuckoo bird. Unique lifestyle of this bird and their typical features of laying eggs and breeding have drawn inspiration for developing a new evolutionary optimization algorithm. Alike different evolutionary approaches, CSA starts with an initial set of population (*i.e.* cuckoo with eggs). There are two types of cuckoo population in different societies: mature cuckoos and cuckoo eggs. The origin of Cuckoo Search Algorithm is established from the struggle to survive among cuckoos. A fraction of cuckoos or their eggs get destroyed during the struggle of survival. But the survived cuckoos make a society and settle into a better habitat and they start laying eggs and reproducing there. Probably the survival attempt of cuckoos converges to a state that there is only one cuckoo society that exits with same profit values. As a novel evolutionary estimation technique, CSA has drawn much attention and extensive applications, due to its easy implementation. As most population-based algorithms, CSA is good at analyzing the promising area of the search space, but not so well at tuning the approximation to the minimization.

In order to increase the efficiency and convergence rate of standard CSA, a modified cuckoo search algorithm has been introduced. Generally, all the parameters in cuckoo search algorithm are kept constant which results in the decreased efficiency of algorithm. To deal with this problem, the parameters of cuckoo search algorithm have been tuned by applying some tuning strategies to it. Considering various commonly used benchmark functions, numerical studies acknowledge that the modified algorithm can find better solution comparative to the solutions obtained by the standard algorithm. On account of this, it is expected that the modified algorithm can be applied successfully to a broad variety of optimization problems.

Table of Contents

List of figures

List of tables

Chapter 1

Introduction

It is no exaggeration to claim that optimization is all around, from the routing of the web to planning holiday and from engineering design to planning of business. In approximately all of these tasks, certain objectives are to be accomplished or to advance profit, quality and time taken. Resources like cost and time are constantly restricted in real-world problems; there is a need to discover results to ideally utilize these significant resources under some constraints. With the help of mathematical tools, mathematical optimization or writing computer programs is the study of such organizing and plan issues. For tackling such optimization problems these days' computer simulations turn into a basic tool with the help of various optimization algorithms that work efficiently.

1.1.1 Optimization

Optimization has perpetually been a dynamic area in research field, since large number of real-world problems which are based on optimization are a part of hard problems. The aim for optimization problem is to discover the best possible solution for an objective function. In other words, optimization is the technique to fine-tune the features of mathematical experiments to get the optimized result or minimum output. The input variables consists following terms namely, the objective function or cost function or fitness function as a result best fitness value of the function is obtained [1]. For evolution of many search algorithms, it was needed to solve optimization problems.

Evolutionary algorithms come under a branch of heuristic algorithms that adopt some visible features of natural evolution to solve optimization problems. The most popular evolutionary algorithms are Genetic algorithm, Genetic programming, Evolutionary strategies, Differential algorithm, and Particle swarm evolution. These are some of the methods influenced from natural processes which are used for solving optimization problems. These problems have their start by taking an initial set of variables and these methods after the start, evolves to attain the global minimum value or global maximum values of target function. One of these nature inspired algorithms is developed by Xin-She Yang *et al.* in 2009 and termed as CSA [2].

Mathematically, most of the optimization problems are expected to compose in the general form

$$\text{minimize}_{y \in \mathbb{R}n} \qquad f_i(y), \qquad (i = 1,2,\ldots,L), \tag{1.1}$$

$$\text{Subject to} \qquad h_j(y)\text{–}\ 0, \qquad (j = 1,2,\ldots.,M), \tag{1.2}$$

$$g_k(y) \leq 0, \qquad (k = 1,2,\ldots.,N), \tag{1.3}$$

Where $f_i(y)$, $h_j(y)$ and $g_k(y)$ are outline vector functions.

$$y = (y_1, y_2, \ldots., y_n)\mathrm{T}, \tag{1.4}$$

Here, the decision variables are y_i of y and these variables can be either discrete, real continuous, or combination of these two variables.
The function $f_i(y)$ is termed as the cost function or objective functions, and when L = 1, and there exists only one objective function. The search space range over by the decision variables is known as $search\,space < n$, while the space designed by the values of objective function is known as the response space or solution space. The equivalences for h_j and variations for g_k are known as constraints. It is worth bringing up that it can also be written as inequalities in another way, and we can likewise plan the objectives as maximization issue.

In a rare but extreme case where there is no objective at all, there are only constraints. This type of problem is known as a feasibility problem considering the fact that any feasible solution is an ideal solution. If optimization problems are categorized corresponding to total number of objectives, at that point there are two classifications: single target L = 1 and Multi-target L >1. Multi-objective optimization is additionally indicated as multi-criteria or even multi-traits improvement in the literature. In daily issues, most optimization problems are multi-objective.
Similarly, optimization can also be classified as number of constraints M + N. For unconstrained optimization problem there are no limitations *i.e.* M = N = 0. In the event that N = 0 and M ≥ 1, it is called a balanced-constrained problem, while M = 0 and N ≥ 1 turns into a unbalanced-constrained problem. In literature of optimization, it is beneficial

focusing on some formulations where equalities are not explicitly included, and only dissimilarities are included. This is on account of equality can be composed as two inequalities. For instance $h(y) = 0$ is equivalent to $h(y) \leq 0$ and $h(y) \geq 0$.

Actual function forms can also be used for classification. The target functions can be either be straight or nonlinear. If h_j and g_k are linear, then they turn into linearly constrained problem. In a linear programming problem, all the constraints and target functions used are linear. Here `programming' has nothing to do with processing programming, it implies arranging and additionally improvement. As all the functions f_i, h_j and g_k are generally nonlinear, we need to manage a nonlinear optimization issue.

1.1.2 Search for optimal solution

After formulating an optimization problem accurately, the key chore is to search for optimal results by some solution strategy using the accurate mathematical procedures. Allegorically, exploring an ideal solution is similar to search treasure [3]. Imagine a person is attempting to chase a secret treasure in a hilly landscape in a given time. In one case, assume he is blind-fold with no guidance; the search procedure is basically a random search, which is generally not proficient as we can expect. In another case, he has been told that the treasure is set at the most elevated peak of the given area, then he will climb up to the highest peak and attempt to reach to the topmost peak, and this situation relates to the traditional hill-climbing techniques. In most of the cases, the hunt is among these limits. There is no blind-fold, and he doesn't know where to search for treasure. It is senseless to look through each and every part of very large area in order to discover the treasure. In most expected scenario he will do an arbitrary walk, while searching for a few clues; randomly he will take a look at some place, and then move to some other place. Such irregular walk is a key characteristic of modern searching algorithms. Clearly, he can either do treasure-hunting alone, so the entire path is a trajectory-based search, and this kind of searching is known as simulated annealing. Instead, he can ask from a group of individuals to do the searching and give them clues, and this situation uses swarm intelligence and relates to the particle swarm optimization. If it is found that the treasure is actually important and the search procedure will take quite long time if the area is very large. If no time bounds are there and any area is open [4]. Theoretically it is possible to discover a perfect treasure (the worldwide ideal solution). Clearly, this search technique can be refined a little more. A few hunters are superior to others. Enroll new hunters while reserving the best ones from previous, it is somehow similar to genetic algorithm or evolutionary algorithms in which improvements are made by searching agents. Actually, after seeing all the current metaheuristic algorithms, an attempt is made to practice the best results or solutions, and substitute not-so-good solutions, though assessing fitness of every individual. With such stability, the plan is to project improved and productive optimization algorithms.

Grouping of these optimization algorithms can be done from multiple point of views. A simple way is to notice the nature of algorithm and for this the algorithm is distributed into two classes namely stochastic algorithms, and deterministic algorithms. Deterministic algorithms trace a comprehensive methodology, its way and estimations of designing variables and the functions of the function. Hill-climbing is an example of deterministic algorithm, and for a similar initial stage, they will take the similar path whenever the program will run. Then again, stochastic algorithms all the time have some randomness. Genetic algorithms are a decent illustration, the population results will be distinctive every time when the program will run meanwhile the algorithms utilize certain pseudo-random numbers, however the outcomes might not have huge dissimilarity, yet the paths of every individual are not precisely repeatable.

Besides, a third kind of algorithm is also that is a blend, or a half and half, of deterministic and stochastic calculations. For instance, hill-climbing with an arbitrary start over is a decent example. The fundamental idea is to utilize the deterministic algorithm, however begin with various initial points. This has some points of interest over a basic hill-climbing strategy, which might be stuck in a local top.

1.2 Random Walks and Lévy Flights

1.2.1 Random Walks

A random walk is an arbitrary procedure that carries a sequence of successive random steps. Numerically, S_N is denoted for summation of each successive random step X_i at that point S_N frames a random walk

$$S_N = \sum_{i=1}^{N} X_i = X_1 + X_2 + \cdots + X_N \tag{1.5}$$

where X_i is denoted for random step carried out from an irregular distribution. This correlation can be written similarly as a recursive method

$$S_N = \sum_{i=1}^{N-1} + X_N = S_{N-1} + X_N \tag{1.6}$$

Which implies the next step S_N will depend on existing step S_{N-1} and the movement or move X_N from the current step to the next step. This is normally the fundamental property of the Markov chain.

Here the length of random walk or the step size can be static or vary at times. Random walks have numerous applications in material science, economics, statistics, computer sciences, environmental science and engineering [5].

Consider a situation where an alcoholic walks on a road, at every step, he can arbitrarily walks to and fro, this results a random walk in one-dimension. If the alcoholic man walks on a soccer field, arbitrarily he can walk towards any path; this shows a two-dimensional random walk. Numerically, following calculation shows an irregular walk

$$S_{u+1} = S_u + w_u \tag{1.7}$$

Where current location is denoted by S_u or the condition at time u, and w_u is a stage or random variable with a famous distribution. If every step is completed in a space of n-dimensions, the random walk talked about earlier becomes a higher dimensional random walk and no reason is found why length of every step should be fixed.

$$S_N = \sum_{i=1}^{N} X_i \tag{1.8}$$

In actual, according to a known distribution, the step size can also fluctuate. The random walk will become Brownian motion (shown in Figure 1.1) if the step size follows the Gaussian distribution. According to the central limit theorem, the random walk must consider Gaussian distribution whenever there is an increase in value of N (number of position). Because of the zero mean of particle locations, variance is likely to increase linearly with the value of t. Variance of Brownian random walk in d-dimensional space is given as

$$\sigma^2(t) = |v_0|^2 t^2 + (2dD)t \tag{1.9}$$

Here v_0 denotes the drift velocity of the method and $D = s^2 / (2T)$ is termed as the effective diffusion coefficient and this coefficient is related to the step size s during each jump in a time interval of T.

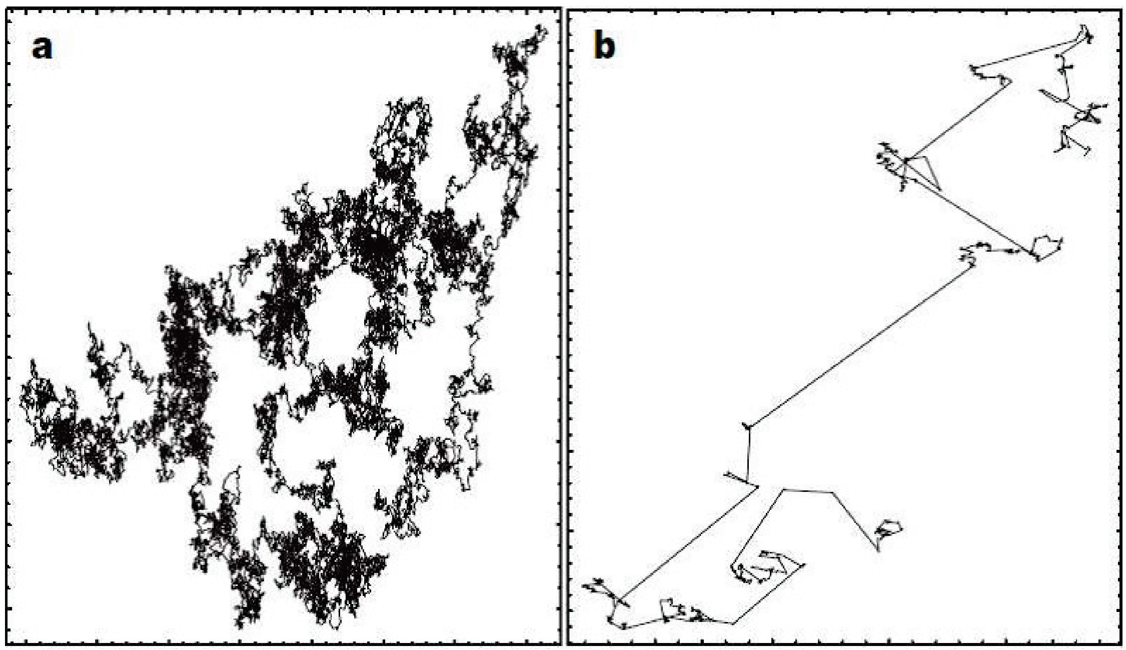

Figure 1.1: (a) Arbitrary motion: Gaussian step-size distribution for random walk
(b) Lévy flight walk used in cuckoo search algorithm

So, the Brownian motion B(t) follows a Gaussian distribution with time-dependent variance and zero mean. It can be written as, $B(t) \sim N(0, \sigma^2(t))$ where ~ symbolize the random variable. The process of diffusion can be defined as the sequence of Brownian motion, and this motion follows the Gaussian distribution. Therefore, normal diffusion is generally denoted as Gaussian diffusion.

If motion at every step is not Gaussian, the diffusion is said to be non-Gaussian diffusion. If the step size follows any other distribution, it is necessary to handle more generalized random walk. When the step size follows the lévy distribution, it comes under the category of a very special case and such type of random walk is known a Lévy walk or lévy flight.

1.2.2 Lévy walks and Lévy flights

Lévy flights are considered as a random walk where step size is fetched by the lévy distribution which is generally derived by an easy power-law principle S*(l)* ~ *|l|-1-β* where index is given as 0 < β ≤ 2. A simplest form of lévy distribution can be mathematically written as

$$S(\mathrm{l},\gamma,\mu) = \begin{cases} \frac{\sqrt{\gamma}}{2\Pi} exp\left[-\frac{\gamma}{2(l-\mu)}\right] \frac{1}{(l-\mu)^{\frac{3}{2}}}, & 0 < \mu < l < \infty \\ 0, & \text{otherwise,} \end{cases} \tag{1.10}$$

Where minimum step length is given by π > 0 and ϒ is termed as scale parameter. Clearly, as $l \rightarrow \infty$, then

$$L(l,\gamma,\mu) \approx \sqrt{\frac{\gamma}{2\Pi}} \frac{1}{8^{3/2}} \tag{1.11}$$

This is an exceptional situation of the often used lévy distribution.

Lévy flights are considered as much more efficient than Brownian random walks in inspecting unknown and a large-scale search space. This efficiency can be explained by many reasons but one of them is because in lévy flights, variance generally increases at a very faster rate than that of linear relationship (i.e., $\sigma^2(t) \sim t$) of random walks in Brownian distribution.

$$\sigma^2(t) \sim t^3 - \beta, 1 \leq \beta \leq 2 \tag{1.12}$$

Considering implementation viewpoint, it requires two steps to generate random numbers with lévy flights: first one is by selecting random paths and other is by generating steps that follow the selected lévy distribution. Uniform distribution must generate a direction, although it is quite tricky to generate steps in the second point. Comparatively there is less number of ways of obtaining this, but the most important, effective and candid way to do this is by using the well-known Mantegna algorithm for a symmetric lévy stable distribution.

The probability distribution of a random variable *V* is said to be constant if a linear arrangement of two identical copies of it (or V_1 and V_2) follows the same distribution.

Viz., $aV_1 + bV_2$ has likely the same distribution as $cV + d$ where $a, b > 0$ and c, d ∈Ṙ. If d = 0, it is known as strictly stable. Gaussian, Cauchy and lévy distributions are all comes under the category of stable distributions. In Mantegna's algorithm, the step length s can be calculated by

$$s = \frac{u}{|v|^{1/\beta}} \tag{1.13}$$

Where u and v are calculated from normal distributions. That is

$$u \sim N(0,\sigma_u^2), \qquad v \sim N(0,\sigma_v^2), \tag{1.14}$$

where

$$\sigma_u = \left\{\frac{\Gamma(1+\beta)\sin(\Pi\beta/2)}{\Gamma\left[\frac{(1+\beta)}{2}\right]\beta\, 2^{(\beta-1)/2}}\right\}^{1/\beta} \qquad \sigma_v = 1. \tag{1.15}$$

This distribution (for s) follows the expected lévy distribution for $|s| \geq |s_0|$ where s_0 is the smallest step. In principle, |s0|»0, but in reality s0 can be taken as a sensible value such as $s_0 = 0.1$ to 1.

Lévy flights have the capability to maximize the overall efficiency of resource searching in uncertain environments. In fact, lévy flights have been observed among foraging patterns of albatrosses and fruit flies, and spider monkeys. Also, lévy flights have many applications. Many physical phenomena such as the diffusion of fluorescent molecules, cooling behavior and noise could show Lévy-flight characteristics under the right conditions.

1.3 Cuckoo Search

Cuckoo search is an algorithm which is motivated from the brood parasitism of cuckoo family in nature. This algorithm was originally used as a tool for numerical function optimization and also for optimization of continuous problems [6].

The performance of this algorithm is tested by researchers using some commonly used benchmark optimization functions and comparing it with some well-known algorithms like Particle Swarm Optimization (PSO) and Genetic Algorithm (GA), and the it is proved that cuckoo search accomplished comparatively optimal results than that obtained by PSO and GA. After this, the native inventors of this cuckoo search algorithm and many other researchers have also implemented same algorithm to various optimization problems. Cuckoo search algorithm also presented better and optimal solutions over this area.

Currently, cuckoo search algorithm has already been applied in almost every area and domain of engineering optimization, function optimization, scheduling, feature selection, image processing, forecasting, planning, and real-world applications.

1.3.1 Breeding behavior of cuckoo bird

Cuckoos birds are attractive due to their melodious sounds and also due to their unique strategy of imitation. Specific species of cuckoo such as the *ani* and *Guira* cuckoos lays their eggs in other birds' nests, although they can eliminate others' eggs in order to hike the hatching probability of their own eggs. By the virtue of laying their eggs in the nests of other host birds (of different spieces), there are various species that are involved in the category of obligate brood parasitism. There are three types of brood parasitism: Cooperative breeding, Intraspecific brood parasitism, and nest takeover. Some host birds can be involved in the direct conflict with the intruding cuckoos. If host birds found that the eggs in the nest are not their own eggs, they will either destroy the foreign eggs or they will leave there own nest and make a new nest in a new place. Some of the cuckoo species such as the new world brood-parasitic *Tapera* have evolved in such a way that female parasitic cuckoos are generally very brilliant in the mimicry in color and pattern of the eggs of a chosen host species [7]. This reduces the probability of their eggs being destroyed and thus increases the probability of their reproduction.

Also, the timing of egg-laying of various species is also noticeable. Parasitic cuckoos frequently choose a nest of host bird where the host bird just laid its own eggs. Generally, cuckoo eggs are hatched earlier than their host eggs. The time when the first cuckoo chick is hatched, firstly it will expel the eggs of the host bird by blindly throwing their eggs from the nest; this hikes the chances of cuckoo chick's share of food which is provided by the owner of nest. Various studies show that in order to gain more feeding opportunity, a cuckoo chick can also mimic the sound of host chicks.

1.3.2 Lévy flights

Flying manners of several insects and animals has established the distinctive features of lévy flights. In their latest study, Reynolds and Frye, presented that fruit flies or Drosophila melanogaster discover its own land with the help of a sequence of straight ways disrupted by a rapid turn of 90^o, resulting into a lévy-flight-style irregular scale-free search arrangement. Indeed, even light can be identified with lévy flights. Consequently, this behavior has been applied to optimization problems and optimal search, and initial results show its promising proficiency [8].

Three algorithms are designed to generate random walks using lévy flights namely Mantegna's algorithm, Rejection algorithm and McCulloch's algorithm. From these three algorithms, Mantegna's algorithm and McCulloch's algorithm shows the best results. Results of both the algorithms presented that for any number of nests (N) and value of α, McCulloch's algorithm performs better than other Mantegna's algorithm [9]. In standard cuckoo search algorithm (CSA), a simplified version of Mantegna's algorithm has been used as its calculations are simpler and it outperforms when α ranges in [0.75, 1.95]. Further, by adding local best term in conventional algorithm gives the finest results in optimization problems.

1.3.3 Cuckoo search algorithm with lévy flights

Cuckoo birds draw attention because of their unique intrusive reproduction strategy. Cuckoo search algorithm gets motivated from the constrained brood parasitic kind of behavior of the cuckoo birds. The cuckoo bird lays its eggs in the nests of some other birds which act as host to the cuckoo's egg and the host can belong to different species. If the host bird found different eggs in its nest, then there will be two possibilities: either the host bird will destroy the unfamiliar egg or it will abandon its nest all at once. The origin of cuckoo eggs that mimics the host bird eggs is found from this point. To apply this, an optimization tool is used by Xin-She Yang *et al.* [11] considering three rules:

- A cuckoo bird lays eggs in a nest which it chooses randomly and only 1 egg is laid at a time by the cuckoo.

- The nests having good quality of eggs (solutions) are the best nests and would be ideal for the generations to follow.
- The host nests available are constant and there is a slight probability, $P_a \in (0, 1)$ which depicts that the host would recognize a cuckoo's egg.

In the current scenario, the host bird on recognizing a foreign egg either destroy it completely or abandons the nest where the foreign egg resides and builds a new nest at a different location.

An assumption approximates the value of P_a as the fraction percentage of the probability that a total number of nests n have been substituted with the new nests owing to a random solution. For the generation of a new solution $Y^{(m+1)}$ corresponding to cuckoo j, a lévy flight will be performed as follows

$$Y_j^{(m+1)} = Y_j^{(m)} + s \oplus \text{Lévy}(\lambda) \quad (1.16)$$

Where step size, s > 0 relates the rules of the problems corresponding to optimization techniques which need to be evaluated. In most of the cases, s = 1 is used as the common step-size. The equation written above stands for the stochastic random walk [8]. Generally, a random zigzag walk is a represents Markov chain where the next position is dependent upon the present position of cuckoo and the functional value of the transitional probability. The product $\oplus$ denotes the calculation of entry-wise multiplications. A lévy flight is a random walk and random step size is formulated by assigning corresponding to lévy flights.

$$\text{Lévy} \sim x = y^{-\mu}, \qquad (1 < \mu \leq 3) \quad (1.17)$$

This has an absolute variance along with an absolute mean. Random walk process especially takes the step with a heavy tailed power law step distribution [9]. As each nest may contain multiple eggs so the algorithm can be taken to a higher step which can handle such complicated cases.

The standard cuckoo search generates random numbers with balanced lévy distribution that can be fetched by Mantegna's algorithm for optimizing a nonlinear function.

Mantegna's algorithm makes use of two stochastic random variables that are allocated normally and these variables results in generation of a new random variables having same nature of a lévy distribution. After this it uses a nonlinear transformation to make it quickly merge to a stable lévy distribution. The algorithm uses the distribution parameters $\alpha \epsilon$ [0.3, 1.99], $c > 0$, and the total number of iterations, n.

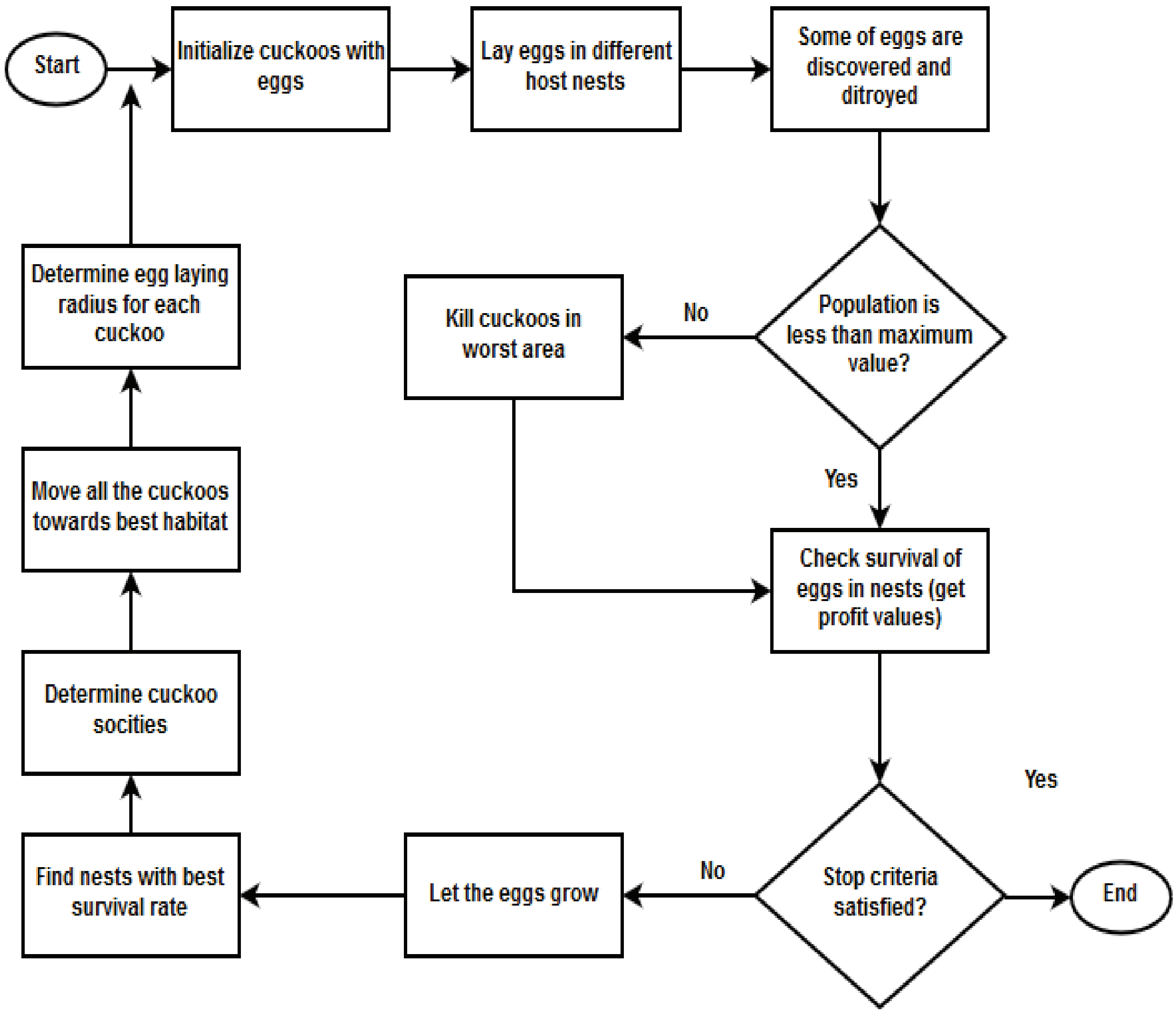

Fig. 1.3 Flowchart of cuckoo search algorithm [21]

With randomization at a large scale, it shows similarity amongst cuckoo search algorithm and hill-climbing but still there is large number of dissimilarities. Firstly, just like GA and PSO, cuckoo search is also a population-based algorithm, however it use some kind of selection criteria or elitism same as that used by harmony search. Secondly, due to heavy-tailed step size and the possibility of having large steps makes it more efficient for randomization. Third, the count of parameters that can be tuned are very less in CSA as compared to that of genetic algorithm and particle swarm optimization algorithm and therefore it is more precise to adjust into a more general class of optimization problems. In addition to this, every host nest signifies possible solutions; therefore CSA would extended to a form of meta-heuristic algorithms.

1.3.4 Advantages of cuckoo search algorithm

Advantages of cuckoo search algorithm over other optimization problems are given in Figure 1.4.

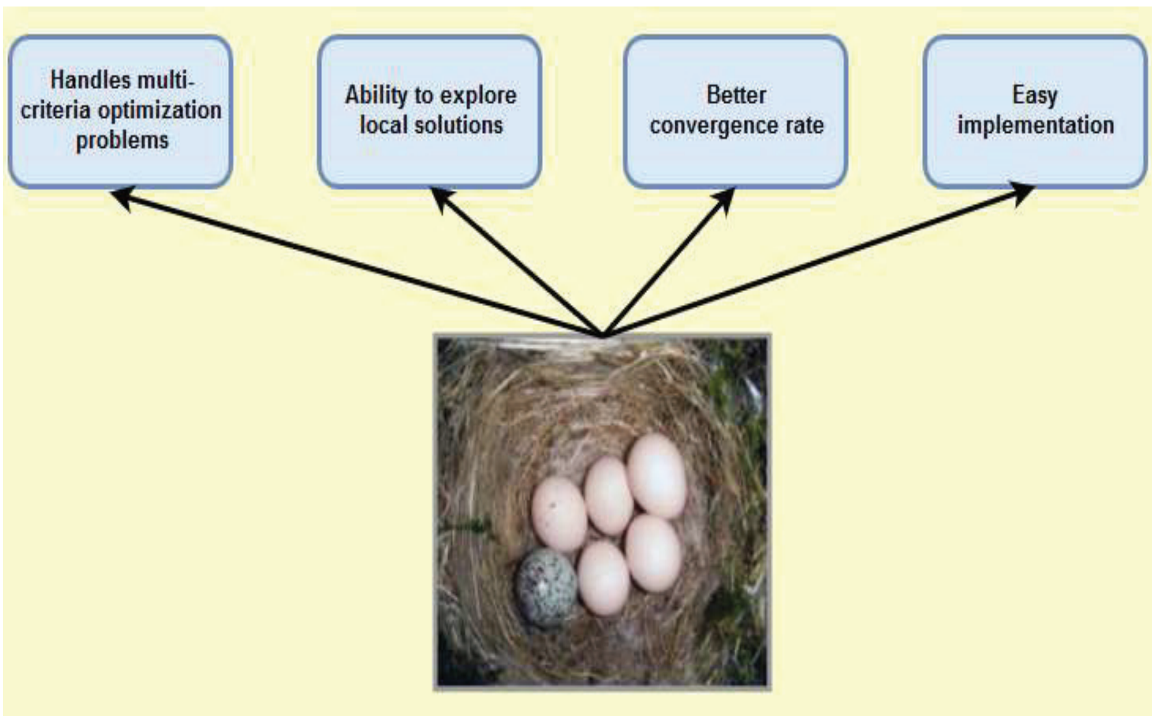

Figure 1.4 Advantages of cuckoo search algorithm

In this work, an improved cuckoo search algorithm has been proposed which is truly inspired by the way cuckoo bird lives. This improved algorithm is based upon breeding of cuckoos and their egg laying behavior. Mature cuckoos and eggs are two existing forms of cuckoos used in modeling. Since cuckoo bird does not make their own nest and lays their eggs in other's nests, the nest with eggs similar to that of cuckoo eggs, is the most preferable solution and if those eggs are not identified, they exaggerate and become a mature cuckoo. For breeding and reproduction, cuckoos need to find the best atmosphere due to the environmental attributes and the migration of societies. This paper describes the modeling and implementation of cuckoo's lifestyle.

Rest of the sections are structured as follows: Chapter 2 represents the literature review of the related work done on cuckoo search using lévy flights. Chapter 3 includes the gaps in the previous work and the problem statement of already existing work. The steps for proposed methodology and the tools used in our work are given in Chapter 4. It includes the systematic overview of how modifications are done in standard cuckoo search algorithm. In Chapter 5, whole experimental setup and results are shown. It carries the information of parameters that affect the performance of cuckoo search algorithm, improvement in values of those parameters and the results obtained by making all the changes in it. Finally, Chapter 6 presents conclusion of the proposed work and the work that can be done in future.

Chapter 2
Literature Survey

In 2009, a new metaheuristic technique was formulated by Yang *et al.* [10] named cuckoo search algorithm. With a combination of lévy flights, the performance of Cuckoo Search Algorithm is analyzed by simulating and comparing it with other optimization algorithms such as genetic algorithm and particle swarm optimization. There is no need to fine-tune parameter p_a as it does not affect the convergence rate. Results presented that CSA outperforms existing algorithms this is because CSA has few constraints to be fine-tuned. Consequently, for many optimisation problems this algorithm is more generic and robust comparative to other metaheuristic algorithms. In addition to their previous work, Yang *et al.* [11] in 2010 used cuckoo algorithm for implementing engineering design optimization problems. A comparison is made between cuckoo search algorithm, Genetic algorithm and Particle Swarm Optimization by expending some standard benchmark functions. They have also designed new stochastic test functions for the purpose which results into better performance of cuckoo search comparative to that of PSO. Goswami *et al.* [12] introduced structural optimization tasks using Cuckoo search algorithm. In this study, the CSA was firstly benchmarked using Himmelblau's problem and then it is implemented on structural engineering problems. Structural engineering problems are complex in nature that even sometimes their optimal solutions are not obtained. To evaluate the performance of cuckoo search, twelve structural engineering test problems have been solved. The performance of CSA was compared with well-known GA, DE, PSO and fireflies algorithm which conclude that CSA is superior to other metaheuristic techniques.

Cuckoo search algorithm is motivated by unique standard of living of cuckoos and their breeding behavior. Therefore, Ramin Rajabioun [13] introduced a Cuckoo Optimization Algorithm (COA) in which the aim is to find the most suitable area for cuckoos to lay their eggs in and to increase their survival rate. An Egg Laying Radius (ELR) is dedicated to each cuckoo and cuckoo starts laying eggs in the nests within this radius as shown in figure 2.1. The process is repeated till the best nest (solution) with maximum profit is found and maximum population of cuckoos is gathered near one location. The proposed algorithm was tested and evaluated on 5 benchmark functions and a comparison is made between COA, GA and PSO. Results presented that COA outperforms other two algorithms by giving fast convergence rate in less iteration. From his work Ramin concluded that COA is suitable for optimization problems but for real case study, it not necessarily the best technique.

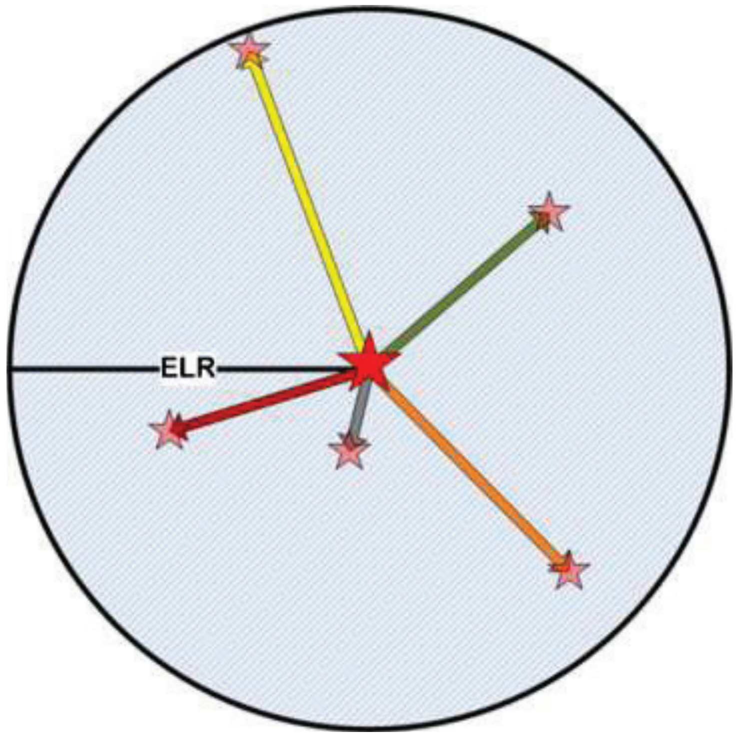

Figure 2.1 Random egg laying within ELR; red star in middle showing initial address of cuckoo with five eggs and pink stars are the nest for these five eggs.

Walton *et al.* [14] developed a new modified cuckoo search algorithm by exchanging information between the best eggs (solutions). Two modifications were made to the original algorithm by changing the step size α in lévy flight and other by exchanging the information between the best solutions. Performance of modified CSA had been compared with CSA, DE and PSO algorithms using MATLAB. In order to test the effect of these modifications, seven benchmark functions had been used which demonstrated that modified CSA shows high convergence rate and better performance than other algorithms even with a high number of dimensions. For global optimization, Valian *et al.*

[15] added an improvement in cuckoo search algorithm by boosting the accuracy and convergence rate of the algorithm. The improvements were made in the values of p_a and α as these parameters support the algorithm to search for better solutions locally and globally. By fine-tuning these two parameters, an improved value of convergence rate is obtained. The whole experiment was done in MATLAB by applying benchmark test functions to both CSA and improved CSA. After improving the CS algorithm, Valian *et al.* [16] in there further study, used this algorithm for training feedforward neural networks. The improved algorithm is then applied on two classification benchmark problems *i.e.* Iris and breast cancer. Improved CSA presented more effective results than simple CSA.

In his research work, Matteo Leccardi [17] described three algorithms for random number generation that are: rejection algorithm, McCulloch's algorithm and Mantegna's algorithm. Results presented that for any number of nests (N) and α, McCulloch's algorithm performs better than other two. In standard cuckoo search algorithm, a simplified version of Mantegna's algorithm has been used as its calculations are simpler and it outperforms when α ranges in $[0.75, 1.95]$. Using these results, Soneji *et al.* [18] compared the performance of McCulloch's algorithm, Mantegna's algorithm and simplified version of Mantegna's algorithm by applying the three in CSA to generate lévy flights. Algorithms had been implemented in MATLAB by taking population size=25 and ten benchmark problems with were implemented dimension=15. Results so obtained presented that simplified version of Mantegna's algorithm and McCulloch's algorithm gives best results. Further, by adding a local best term in the results of simplified version outperforms other optimization problems. In their future work, they added that poor nests must be prioritized to remove other than choosing random nests.

Since, the ability of searching pattern in CSA is not fully discoverable, Zhang *et al.* [19] proposed an adaptive modified cuckoo search algorithm. To take full advantage of CSA, a series of relative approach (including grouping, parallel, incentives, and information sharing and adaptive) had been designed to enhance the performance of original algorithm. Further this modified cuckoo search was evaluated using nine benchmark functions and superior performance of modified algorithm had been evaluated. Nebojsa

Bacanin [20] presented software implementation of cuckoo search and named it as CSApp. The software is a fast, robust and object-oriented system which was further tested on benchmark problems. This software was developed in JAVA language with a user-friendly GUI. Improvements in the standard algorithm were made in order to decrease its execution time and results presented that superior performance of CSApp.

In their study, Tuba *et al.* [21] presented an algorithm for unconstrained optimization problems by modifying the stand CSA. As CSA use random step size, the modification in pure code was made by changing the step size. The improvement was made in such a way that the nests matrix was sorted by the fitness values of the solutions so that solutions with higher fitness can be considered over lower fitness solutions. Software was developed for cuckoo search in JAVA programming language for the purpose and then it had been tested on ten benchmark functions to compare the performance of two CSA algorithms. Results were quite satisfying, as the modified algorithm performed slightly better than standard algorithm in seven functions. In further studies, it can be used for other benchmark functions and some real-world problems. Srivastava *et al.* [22] presented the part of CS algorithm in software coverage optimization. Optimized test sequences had been generated in order to obtain 100% software coverage. Purpose of this study was to show the testing details and its importance. The comparison of test functions had been made with that of ant colony optimization (ACO) and genetic algorithm (GA) which demonstrates the outperforming results of CSA over ACO and GA.

For solving engineering optimization problems, Valian *et al.* [23] introduced an improved algorithm for cuckoo search. Firstly, they worked on standard cuckoo search by tuning its parameters to improve the performance of the algorithm and then the improved CSA was implemented on some complex optimization problems named as a large-scale reliability optimization problem and a 15-unit system. Effectiveness of improved algorithm was compared with several well-known methods which demonstrated that improved algorithm outperforms simple CSA. Feature selection has become an active area these days by extracting selective set of features. Inspired by cuckoo bird's behavior, Rodrigues *et al.* [24] in their research proposed a new feature selection technique and named it as binary cuckoo search. The two datasets for the purpose was collected from an electric power company of Brazil. The datasets are composed of 3486 and 5645 industrial profiles with eight features each. Results of Binary cuckoo search (BCS) were compared with other evolutionary techniques like Binary bat algorithm, Binary firefly algorithm, Binary gravitational search algorithm and Binary particle swarm optimization and it presented that BCS is the most appropriate algorithm for theft recognition upto 40%. In structural engineering, design problems are mostly multiobjective and algorithms to solve these problems are different from single objective. Therefore, Yang *et al.* [25] formulated cuckoo search algorithm for multiobjective optimization problems by validating it in contradiction of a set of test functions that are multiobjective in nature and then it was applied to resolve structural design problems. Two design benchmark problems that had been used are: welded beam and disc brake. For almost all the test functions, modified CSA performed very well and quality of solution sets so obtained were also good. Nasa-ngium *et al.* [26] presented an improvement in modified CSA focusing on generation of new nests from topmost nests. In their work, to replace constant values they have used tent map chaotic sequences and Cauchy algorithm is replaced by Mantegna's algorithm. And the proposed algorithm used a combination of unimodal and multimodal benchmark functions to figure out the efficiency of improved algorithm. The conclusion of study gives that performance of improved CSA is best for wide range of optimization problems.

Considering previous work of cuckoo search, Fister Jr. *et al.* [27] summarized that due to its promising results and better efficiency, CSA had attracted great attention. Cuckoo search, its variants and its applications are practical in every field of science, engineering and industries. The review concluded that theoretical analysis should be carried out and for CS variants many other parameters can be tuned to check the effect of cuckoo's behavior. Xin-She Yang *et al.* [28] discussed the latest developments and applications of cuckoo search algorithm. They analyzed the searching behavior and mechanism of CSA and then few topics for further research have been developed. With best convergence rate, cuckoo search also had few issues to be worked on in future. First issue is that there is a critical gap between practical and mathematical analysis of optimization algorithms and second issue is parameter tuning in optimization problems. These issues were extracted from the previous studies that are to be worked in future.

To enhance convergence rate and searching ability to find optimal solution in CSA, Zhang *et al.* [29] added an improvement in standard CSA. The scale factor of step size is α, therefore value of α has been modified so as to avoid flying too far. To check the improved performance of CSA, five benchmark functions had been applied and the comparison was made between CSA, improved CSA and PSO. Results presented higher precision and better convergence rate of improved CSA. Mahmoudi *et al.* [30] used discrete cuckoo optimization algorithm for solving an NP hard problem *i.e.* graph coloring problem. Cuckoos fly only a part of distance (*i.e.* λ %) with a deviation (*i.e.* φ $radians$) towards the goal habitat as shown in Figure. 2.2. The study presented discretization of cuckoo search algorithm and the problem is evaluated on wide range of benchmark problems. Result demonstrated that in some cases, modified algorithm shows 100% success rate but still it can't be considered the best algorithm.

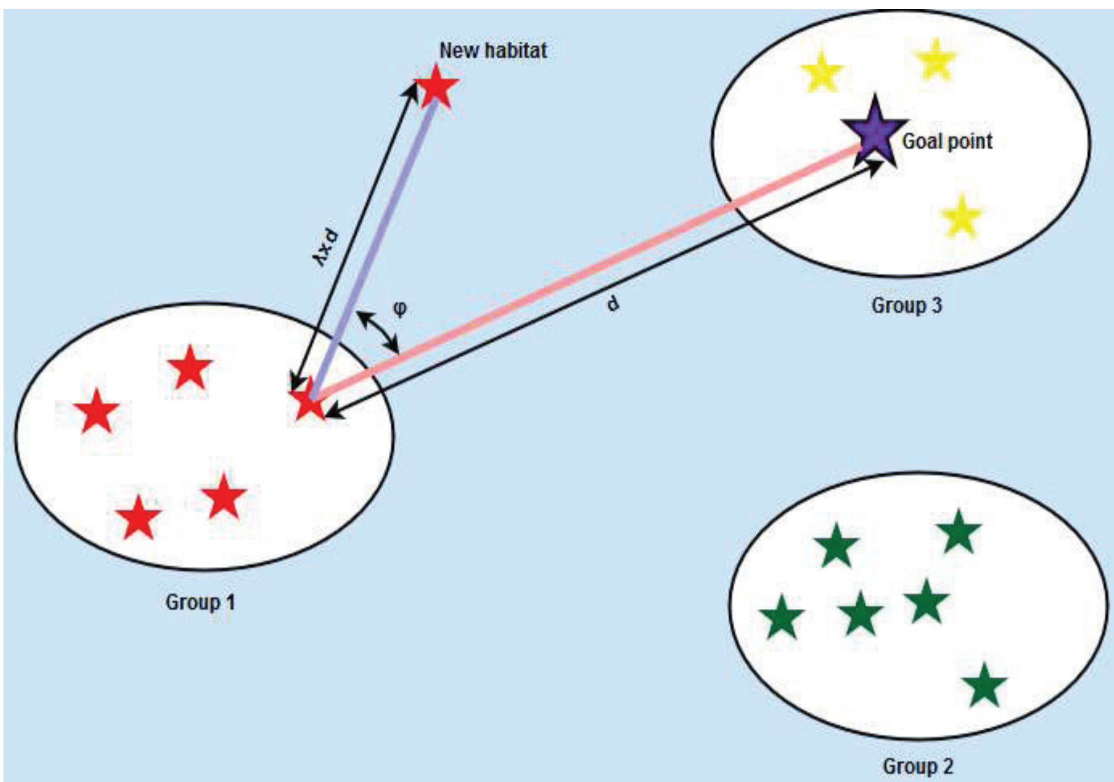

Figure 2.2 Cuckoo's migration towards goal point [21]

Praveen Ranjan Srivastava [31] in their work used cuckoo search for system analysis. In requirement coverage, cuckoo search has been used to generate optimal test cases using JAVA language. The CFG of source code that is to be tested is given in XML (extensible markup language) file format and then cuckoo search is called to calculate in-degree and out-degree at each node. Architecture of implementation of cuckoo search algorithm in software analysis is as shown in Figure. 2.3. Results so obtained were very much efficient than other optimization techniques. As cuckoo search had solved many optimization problems efficiently, in future it can be advanced to multiobjective optimization problems. In their study, Naik *et al.* [32] presented an adaptive cuckoo search aiming to find the step size from the fitness value to achieve global minima or global maxima. As the value of α is fixed in previous studies, this parameter α has been omitted and the step was given an

$$step_i(t+1) = \left(\frac{1}{t}\right)^{\left|\frac{bestfit(t)-fit_i(t)}{bestfit(t)-worstfit(t)}\right|} \tag{2.1}$$

This parameter free algorithm was tested on 23 benchmark functions and the result presented that adaptive CSA outperforms standard CSA.

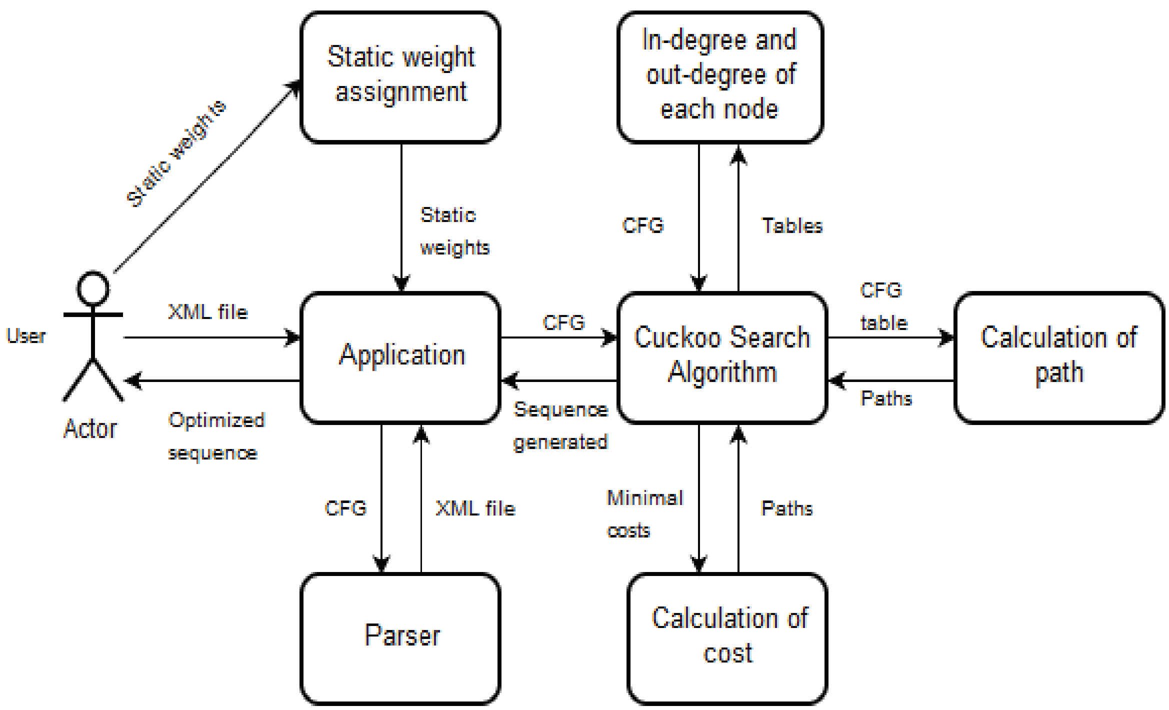

Figure 2.3 Cuckoo search used in software analysis [22]

Umenai *et al.* [33] modified an algorithm for dynamic environment. The study was carried out by considering an optimization problem that is dynamic in nature and where the global best can be cyclically changed with time. The modified algorithm holds the best solutions when solutions were searched in search space near intensive local search. The proposed method had given best performance to find good solutions when applied to four benchmark problems.

For feature selection, Aziz *et al.* [34] modified CSA using rough sets theory. Two modifications made in CSA are: first, the value of α was made to change with number of iterations and secondly a crossover was performed between cuckoos. Rough sets were used to build the fitness function by taking reduced set of features and classification quality. To test and validate the performance of algorithm, benchmark datasets were acquired from UCI repository and the dataset description is given in Table. 2.1.

S. No.	**Dataset used**	**Total samples**	**Genes**	**Classes**
1	Dermatology	34	366	6
2	Breast cancer	10	699	2
3	Hepatitis	19	155	2
4	Lung cancer	56	32	3
5	Iris	4	150	3
6	Pima Indians	8	768	2
7	Lenses	4	24	3

Table 2.1 Description of classification dataset to test the modified CSA [25]

To calculate the efficiency of the proposed approach, K-nearest neighbors and support vector machines these two learning algorithms had been used. In Particle Swarm Optimization (PSO), global best can be directly used to find the new positions of the particles. Xiangtao *et al.* [35] proposed an algorithm to find the top solution from the whole population. In the proposed algorithm, first part used the neighborhood existence to improve the distinction of the algorithm and the second part used new searching strategies to stabilize the exploitation and exploration. The performance of CSA was verified using 30 benchmark problems and it presented that modified algorithm outperforms CSA.

From above studies, it has been concluded that cuckoo search algorithm has been used for various applications. In these applications standard CSA as well as modified CSA has been used by changing value of parameters to achieve the best convergence rate. In most of the studies the main focus is on the value of parameter α and generation of new nest from topmost eggs. Various benchmark problems have also been used to make a comparison between the performances of CSA with well-known optimization algorithms namely PSO and GA.

Chapter 3

Gap analysis and problem statement

3.1 Gap analysis in existing work

In previous section, literature survey of Cuckoo Search Algorithm (CSA) using lévy flights has been given. Considering the previous studies, following gaps have been identified:

- No better technique than Mantegna's algorithm has been developed for minimizing the step size in [18].
- Till date α is taken as major parameter to modify the algorithm to reduce the fitness value of the algorithm [14].

3.2 Problem statement

After reviewing the literature, it has been analyzed that still there is some scope of improvement in cuckoo search algorithm taking in consideration the distance travelled by cuckoo using lévy flights. Since cuckoo bird does not make its own nest and lays its eggs in some other host bird's nest so a solution is adopted using lévy flights i.e. random walks which gives the optimized nest selection criteria for laying the eggs of the cuckoo bird. The nest with eggs similar to the eggs of the cuckoo bird is the most preferable nest and there is probability, $p \in [0,1]$ that the host bird will discover the new egg. Each egg in the nest signifies the possible solution, and cuckoo egg is represented as a new solution.

The main objective of the proposed algorithm is:

- To find the new solutions it is potentially better to substitute not-so-good solutions.
- To minimize the fitness value of the function used to generate lévy flight using different benchmark functions.

The best nest is found on the basis of the value retrieved by the fitness function and the nest with the minimum fitness value is chosen for the purpose. The lower the fitness value of a particular nest, the more preferable is to choose that nest for laying the eggs and that nest will be the best nest.

Chapter 4

Proposed methodology

4.1 Proposed work

The proposed work focuses on comparison of modified CSA with standard cuckoo search by testing both the algorithms on standard benchmark optimization problems [36]. Standard cuckoo search used Mantegna's algorithm for calculating step size in lévy flights. The algorithms were implemented on MATLAB and their comparative graphs were also obtained which simply show the performance of two algorithms. Two graphs are drawn with respect to each benchmark function: one is gradient descent graph for that particular problem and other is simple line graph. Flow diagram of methodology used in proposed approach is shown below in Figure 4.1.

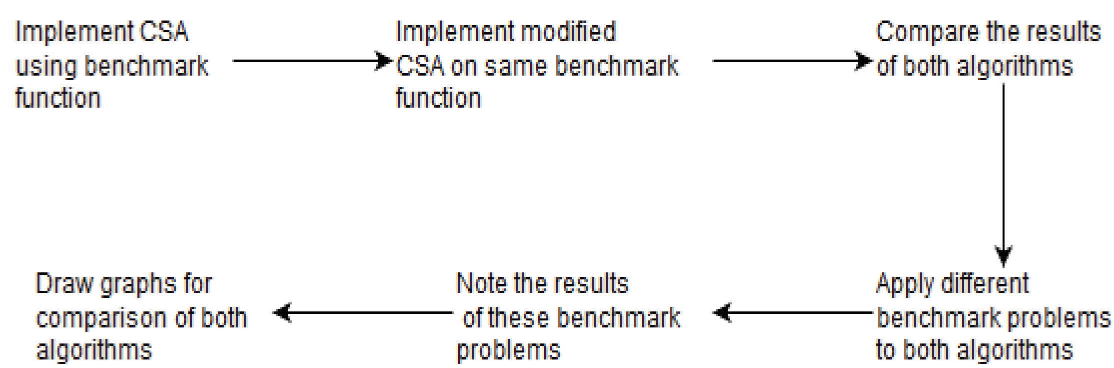

Figure 4.1 Flow diagram of proposed methodology.

Following steps have been followed for implementing the modified cuckoo search algorithm:

1. Installation of MATLAB R2013a is required for running the cuckoo search algorithm code on windows.
2. Implement conventional CS algorithm using MATLAB R2013a.
3. Change the values of parameters to improve the convergence rate. And then step size is calculated by diverting the flight to some radians in lévy flights. If the

improvements resulted in better results than conventional method, it is added to improved algorithm.

4. Implementation of improved cuckoo search algorithm code by taking same benchmark problems that has been used for standard CSA.
5. Nine benchmark problems have been introduced to MATLAB R2013a code of CSA as well as modified CSA. Following benchmark problems have been used for purpose:
   5.1. Ackley function
   5.2. Alpine function
   5.3. Beale function
   5.4. Bohachevsky function
   5.5. Booth function
   5.6. Griewank function
   5.7. Rosenbrock function
   5.8. Sphere function
   5.9. Sum square function
6. Two types of graphs have been obtained from the simulation performed using the said benchmarks:
   6.1. Gradient descent graph
   6.2. Simple line graph
7. Compare the values of fitness function and best nests obtained in results.

4.2 Modifications in cuckoo search algorithm

When sufficient calculations are given, the cuckoo search will every time meet the optimal solution but a fast convergence rate cannot be assured, due to the reason that search depends only on random walks. Therefore, few modifications to the conventional method are made so as to enhance the convergence rate, so making the approach more practical by considering the important features of the original approach [30]. As the fitness function is evaluated on the basis of lévy flight's step size, so here the target is to reduce the value of fitness function. The modification to conventional cuckoo search is made by changing the step size in lévy flight. In conventional Cuckoo search algorithm, length of step for every iteration is kept constant. As the number of generations in the standard cuckoo search increases, the value of step size reduces.

Algorithm for modified cuckoo search

Initialize population of 'm' host nests a_i where $i = 1,2, \ldots., m$.

Initialize a population of random fitness values f_i where $i = 1,2, \ldots., m$.

Tol , tolerance value

$f(x)$ is fitness function of chromosome

for all a_i **do**

Calculate fitness value $f_i = f(a_i)$

end for

$f_{min} \leftarrow$ minimum fitness value amongst all nests

$x_{min} \leftarrow$ nest with minimum fitness value

while $(f_{min} > Tol)$

for $\boldsymbol{j} = 1 \; to \; m$

Produce a cuckoo nest (a_j) using lévy flight from any randomly chosen nest

$f_j = f(a_j)$

if $(f_j < f_{min})$

$f_{min} \leftarrow f_j$

$x_{min} \leftarrow x_j$

end if

A segment p_a of worst nests are abandoned

Fitness of new nests areevaluated and the solutions are ranked

end for

end while

Initially, the value of the step size at the chosen end is set to 1 and in every generation, a step of lévy flight is evaluated using of generation number. This kind of search which is exploratory in nature is executed only on the number of host nests that are abandoned completely. However, in this study the step size is changed on the basis of the values of parameters used [34]. The parameters change their values on the basis of nest size that is the number of features a nest has as its attributes.

Another modification is in the use of benchmark functions which are available to act as the fitness functions for finding the fitness value of the nest on the basis of varying functions definitions they incorporate [9]. A total of 9 benchmark functions have been used which are explained in next section.

Chapter 5

Experimental results

This chapter draws attention towards the implementation of well-known optimization problem Cuckoo Search Algorithm (CSA) and its comparison with modified Cuckoo Search Algorithm (CSA) with lévy flights that has been proposed in this work. Both the algorithms are implemented in MATLAB [37] and there relative performance is compared in order to check the best output of the algorithm modified.

5.1 Tools used for implementing cuckoo search algorithm

5.1.1. MATLAB R2013a

Matrix based MATLAB language is the most natural way of expressing computational mathematics. MATLAB is an optimized platform to deal with engineering and scientific problems.

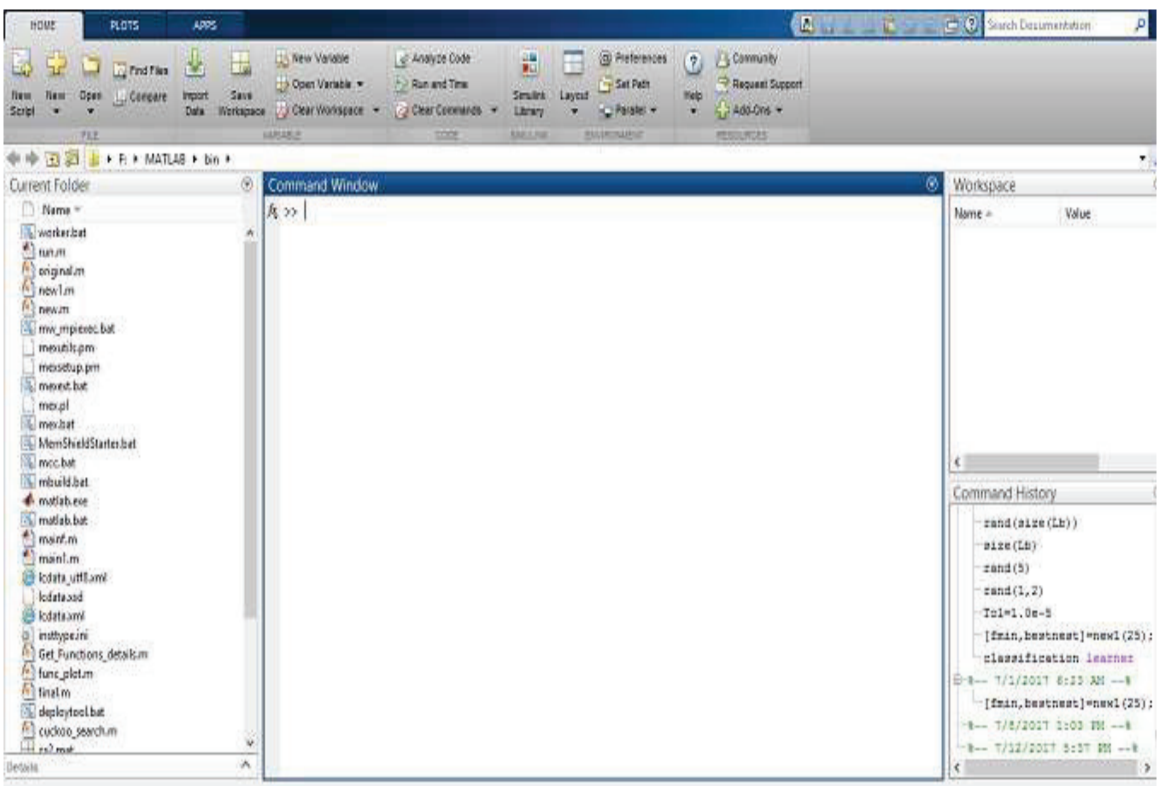

Figure 5.1 MATLAB command window in Windows 8.1

It is a platform that is being used for developing algorithms, plotting graphs for functions, manipulating matrix and creating user interface. The MATLAB works in Windows, Linux and Mac OS for IA-32 and x86-64 platforms. In our experiment, the version we are using is MATLAB R2013a in windows operating system. Install it by running the MATLAB application file. The MATLAB command window is shown in Figure.5.1.

5.2 Implementation

MATLAB editor is used to write code or MATLAB programs for any algorithm. As a new algorithm has been proposed in this research, the code for proposed work is written and enhanced in MATLAB editor as shown in Figure 5.2.

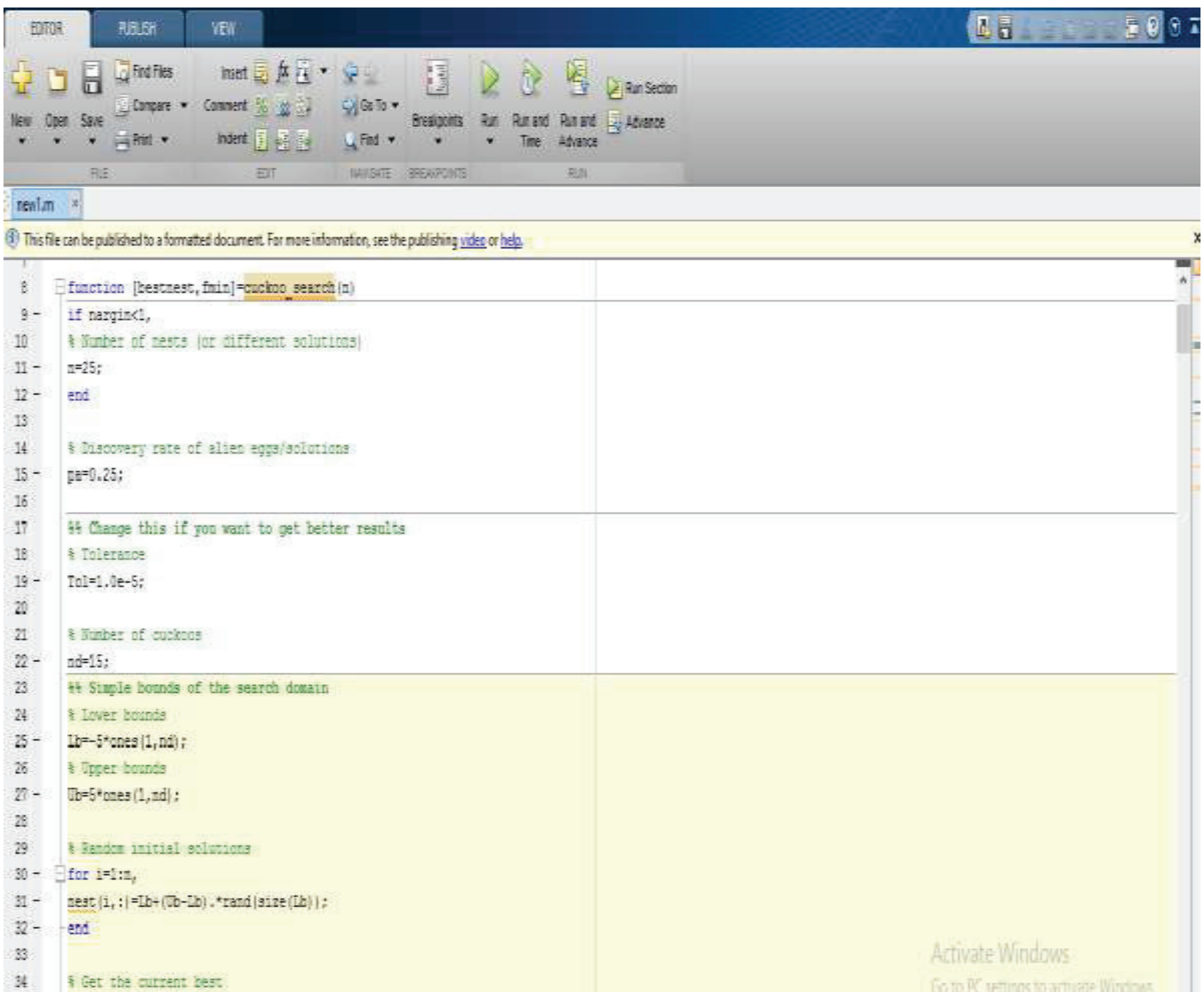

Figure 5.2 CSA code in MATLAB editor.

After implementing the code, the results of the algorithm will comes out on command line window. Here, the results of the cuckoo search algorithm gives the fitness value of the function, the best nests from the total number of nest shown in Figure 5.3 and a graphical representation of the results obtained. In proposed work, the value of *fmin* has to be decreased and for those fitness values 15 best nests has been obtained corresponding to total number of nests given (*i.e.* N). All the best nests values have been given in output. The graphical output consists of two graphs in which first graph shows the gradient graph and other shows simple line graph. The graphical results are also shown in Figure 5.4.

```
Command Window

  Columns 64 through 67

    0.8900    0.9200    0.9500    0.9800

>> [fmin,bestnest]=new1(25);
Total number of iterations=31250

fmin =

   9.4241e-06


bestnest =

  Columns 1 through 9

   -0.0020   -0.0006   -0.0015   -0.0039   -0.0012    0.0005   -0.0032   -0.0006   -0.0025

  Columns 10 through 15

   -0.0012   -0.0028    0.0008   -0.0086    0.0032    0.0021


y =

  Columns 1 through 9

   -1.0000   -0.9700   -0.9400   -0.9100   -0.8800   -0.8500   -0.8200   -0.7900   -0.7600
```

Figure 5.3 Results of proposed algorithm in MATLAB.

As shown in graphical representation below, the gradient and line graph is drawn for *Griewank function* which is categorized under 'many local minima' category. In line graph, the red line shows modified cuckoo search algorithm and the blue line shows standard CSA. As it can be clearly seen from the graph, the best score is obtained in lesser count of iterations in modified algorithm while the iterations is more in standard cuckoo search.

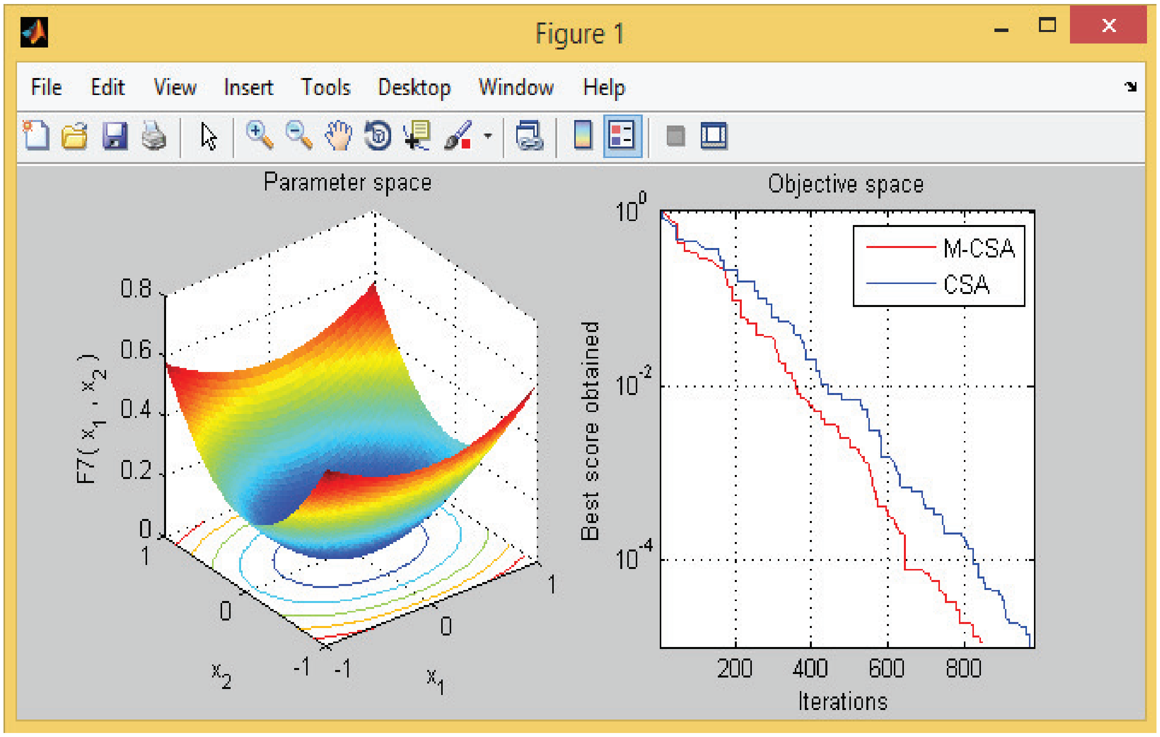

Figure 5.4 Graphical representation of results obtained through proposed algorithm.

To attain the better performance of modified cuckoo search optimization algorithm than standard algorithm for cuckoo search, the relative performance of both the algorithms have been contrasted. The algorithms are implemented on nine different benchmark optimization problems in MATLAB for testing the new optimization technique and also these problems gives remarkable results by validating and comparing its performance. The value of fitness function is calculated by these benchmark problems to get the minimum fitness value.

5.2.1. Benchmark problems

In evolutionary calculations, generally different algorithms are compared by taking a large test set specially when there is a function optimization. Benchmark problems that are also known as test optimization problem are used in applied mathematics to evaluate accuracy, convergence rate and robustness of the optimization algorithms [38]. For every optimization problem, there are some defined parameters for them like dimension is denoted by D, domain size of problem by $x_i \in [Lb, Ub]$ and the optimal solution is written as $f(x^*) = f(x_1, \ldots., x_n)$.

5.2.2. Parameter setting

In our research work, some parameters of cuckoo search algorithm need to kept constant for both algorithms. Number of trials that are performed on each benchmark problem is set to 20, the number of nests defined by N, D for dimension for cuckoo search and p_a is the probability of cuckoo egg being discovered. The results have been examined by giving variations in number of host nests (n*),* dimension of the algorithm (*D*) and in probability of discovered eggs (p_a). The value of *n* was taken as (10, 15, 20, 25, and 30), *D* is taken as (5, 10, and 15) and value of p_a is taken as (0.2, 0.25, and 0.5). After testing the algorithm on all these values, we fixed these parameter's value to $n = 25, D = 15$, and $p_a = 0.25$. Fitness function is calculated by considering these parameters and the best values of fitness function are obtained at these values. Parameter setting is shown below in Table 5.1.

Parameter	Values
Trials	20
N	25
D	15
p_a	0.25

Table 5.1: Parameter setting for benchmark functions.

After setting these parameters, performance of standard CSA code is evaluated and first modification is made in value of β and other in step size in Mantegna's algorithm. Twenty trials were performed by taking $\beta = 3/2$, and then the value is changed to 1.99 and again twenty trials were performed. The results illustrates best results when $\beta = 1.99$ and any change in value of β changes the value of σ in the algorithm. In next modification, step size of the algorithm is validated by adding an angle to the lévy flight. A number of trials were performed by changing the radians of sine and the results so obtained are noted. As the fitness function is evaluated on the basis of lévy flight's step size, so here the target is to minimize the fitness function value.

5.2.3. Test functions

In literature, there are a number of benchmark problems [39] that have been used to test the effectiveness of optimization problems. These functions are grouped according to their similarities in shapes and their physical properties. For example, many local minima, bowl-shaped, plate-shaped, valley-shaped, steep ridges and others. In this section, all the test functions are used as minimization problems. For every function, there exists an optimal objective value $f(x^*)$ corresponding to the optimal minimal solution x^*. Brief descriptions of benchmark functions that have been used for the purpose is given below.

- **Ackley function**

  It is an n-dimensional function having one global minimum and countless local minima

  Count of variables used: *n*.

  Function: $-a\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} y_i^2}\right)-\exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cy_i)\right)+a+\exp(1)$

  Search space range: $-32.768 \leq\ y_i \leq 32.768$ where $i =\ 1, 2, \ldots, n$.

  Global minimum: y^*= (0, …, 0), $f(y^*) =\ 0$.

- **Alpine function**

  It is a function is defined on n-dimensional space with one global minimum and a large number of local minima.

  Count of variables used: *n*.

  Function: $f(y) = \sum_{i=1}^{d} |y_i sin(y_i) + 0.1y_i|$

  Search space range: $-10 \leq\ y_i \leq 10$, where $i =\ 1, 2, \ldots, n$.

  Global minimum: y^*= (0, …, 0), $f(y^*) =\ 0$.

- **Beale function**

  It is a multimodal function with one global minimum.

  Count of variables used: *n* = 2.

Function: $f(y) = (1.5 - y_1 + y_1 y_2)^2 + (2.25 - y_1 + y_1 y_2^2)^2 + (2.625 - y_1 + y_1 y_2^3)^2$

Search space range: $-4.5 \leq \ y_i \leq 4.5$ where $i = \ 1, 2$

Global minimum: $y^* = (3, 0.5), f(y^*) = \ 0.$

- **Bohachevsky function**

  Count of variables used: *n* = 2

  Function: $f_1(y) = y_1^2 + 2y_2^2 - 0.3cos(3\pi y_1) - 0.4cos(4\pi y_2) + 0.7$

  $f_2(y) = y_1^2 + 2y_2^2 - 0.3cos(3\pi y_1)cos(4\pi y_2) + 0.3$

  $f_3(y) = y_1^2 + 2y_2^2 - 0.3cos(3\pi y_1 + \ 4\pi y_2) + 0.3$

  Search space range: $-100 \leq \ y_i \leq 100$, where $i = \ 1, 2$

  Global minimum: $y^* = (0, 0), f_j(y^*) = \ 0. j = 1,2,3$

- **Booth function**

  Count of variables used: *n* = 2.

  Function: $f(y) = (y_1 + 2y_2 - 7)^2 + (2y_1 + y_2 - 5)^2$

  Search space range: $-10 \leq \ y_i \leq 10$, where $i = \ 1, 2$

  Various local minima.

  Global minimum: $y^* = (1, 3), f(y^*) = \ 0$

- **Griewank function**

  Count of variables used: *n*

  Function: $f(y) = \sum_{i=1}^{d} \frac{y_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{y_i}{\sqrt{i}}\right) + 1$

  Search space range: $-600 \leq \ y_i \leq 600$, where $i = \ 1, 2, \ldots, n$.

  Global minimum: y^*= (0, …, 0), $f(y^*) = \ 0.$

- **Rosenbrock function**

  Count of variables used: *n*

  Function: $f(y) = \sum_{i=1}^{d-1}[100(y_{i+1} - y_i^2)^2 + (y_i - 1)^2]$

  Search space range: $-5 \leq \ y_i \leq 10$, where $i = \ 1, 2, \ldots, n$.

  Global minimum: y^*= (1, …, 1), $f(y^*) = \ 0.$

- **Sphere function**

  Count of variables used: *n*

  Function: $f(y) = \sum_{i=1}^{d} y_i^2$

Search space range: $-5.12 \leq y_i \leq 5.12$, where $i = 1, 2, \ldots, n$.

Global minimum: y^*= (0, …, 0), $f(y^*) = 0$.

- **Sum squares function**

  Count of variables used: *n*.

  Function: $f(y) = \sum_{i=1}^{d} i y_i^2$

  Search space range: $-5.12 \leq y_i \leq 5.12$, where $i = 1, 2, \ldots, n$.

  Global minimum: y^*= (0, …, 0), $f(y^*) = 0$.

5.3 Results

In this segment, final results are demonstrated by obtaining the fitness values (*i.e. fmin*) and the best nests values (*i.e. bestnest*) for lévy flights which shows the better values for modified cuckoo search than standard cuckoo search optimization algorithm. Results of this algorithm have been evaluated by following functions:

- value of *fmin* (fitness value)
- *bestnest* values (values of all the best nests so obtained)
- Line graphs comparing two algorithms

There are two type of graphs obtained in result that are: gradient descent graph and simple line graph. Gradient descent graphs are used to find the minimized the value of the function and find the local minimum of that function. The slope move against the gradient down towards the minimum by considering that one starts by guessing a local minimum of x_0, x_1, x_2 ... so that

$$x_{n+1} = x_n - \gamma_n \nabla F(x_n), n \geq 0$$

where we have,

$$F(x_0) \geq F(x_1) \geq F(x_2) \geq \ldots.$$

By changing the step size γ at every iteration, the series of results (x_n) converges towards the local minimum. The gradient graphs so obtained for different optimization problems used are shown in Figure 5.2.

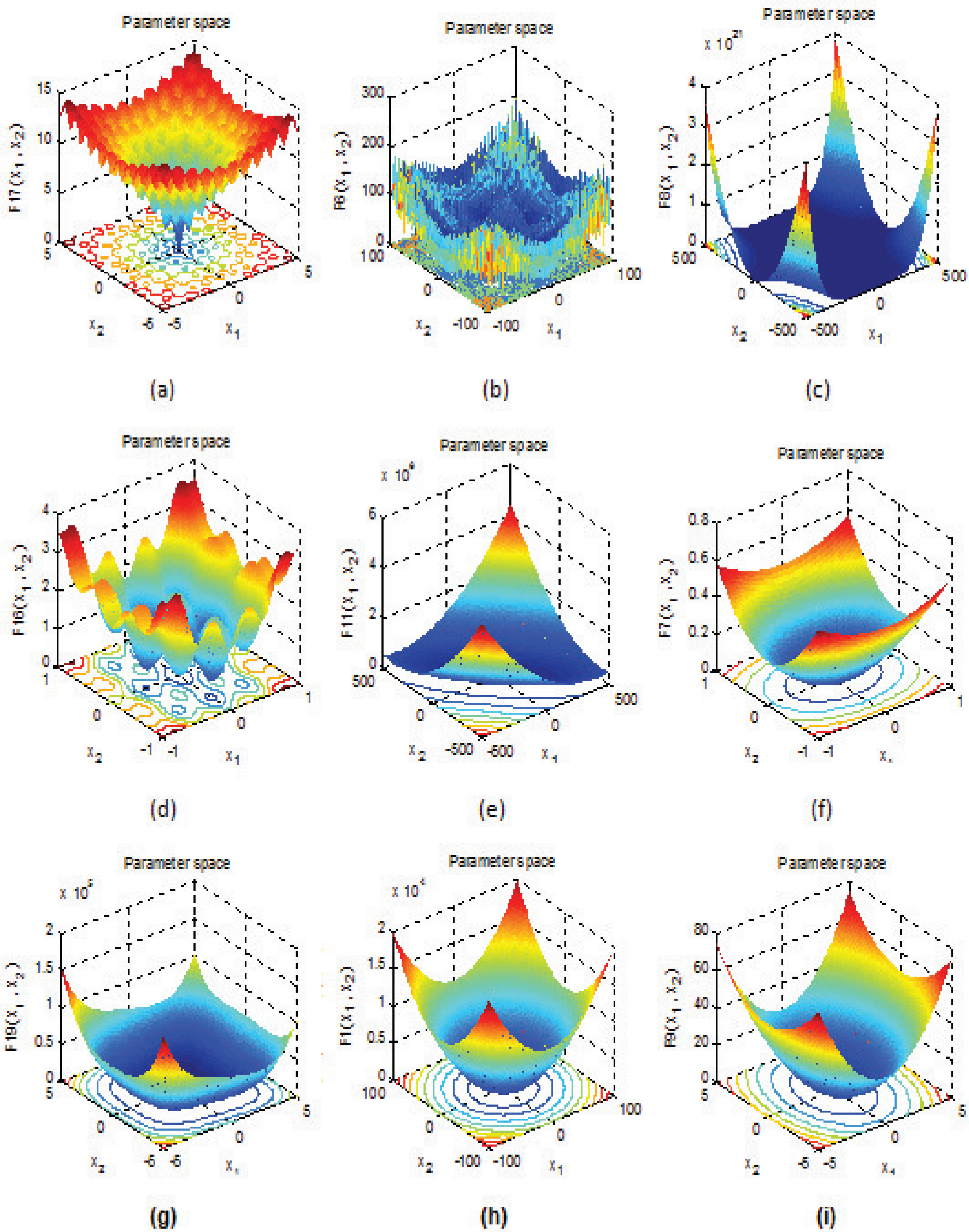

Figure 5.5 Surface plot for: (a) Ackley function, (b) Alpine function, (c) Beale function, (d) Bohachevsky function, (e) Booth function, (f) Griewank function, (g) Rosenbrock function, (h) Sphere function, (i) Sum square function

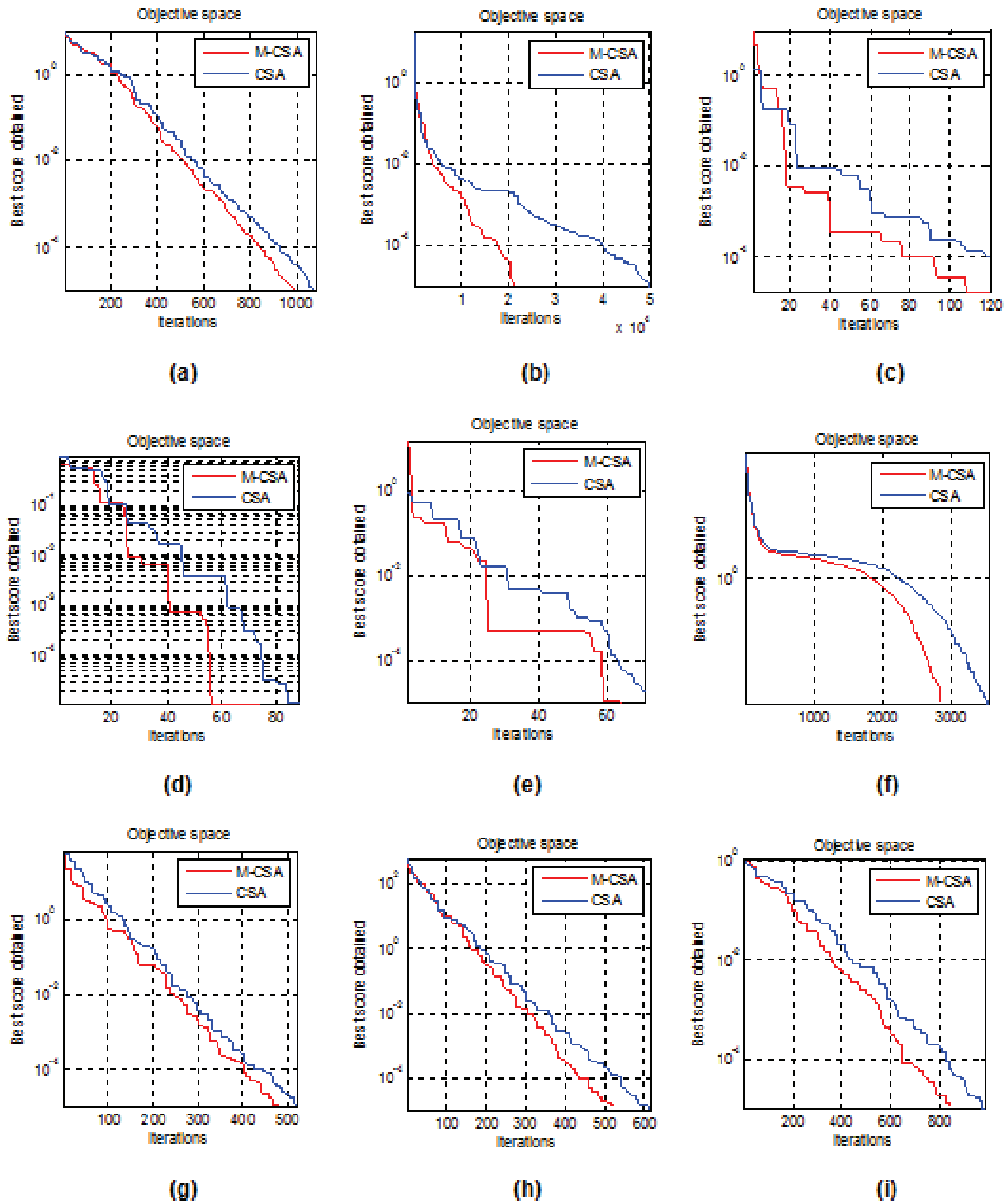

Figure 5.6 Comparison between the performance of the CSA and M-CSA algorithms: (a) Ackley function, (b) Alpine function, (c) Beale function, (d) Bohachevsky function, (e) Booth function, (f) Griewank function, (g) Rosenbrock function, (h) Sphere function, and (i) Sum square function

The line graphs are simple graphs that are used to compare the standard CSA and modified CSA. As the purpose is to reduce the fitness function, the graph clearly shows that for every function the slope (represented by red line) of modified CSA converges towards the minimum in less number of iterations than the slope (represented by blue line) of standard CSA. The line graphs are drawn between local minimum of x_0, x_1, x_2 ... and the fitness values, $F(x_0, x_1)$ of the best nests so obtained. These graphs illustrate the better convergence rate of the algorithm modified. The graphs are shown in Figure 5.6.

Chapter 6
Conclusion and future scope

In this study, a new improved cuckoo search optimization algorithm has been proposed using an improved approach for lévy flights by minimizing value of fitness function and finding the best environment for the cuckoos to lay eggs. The result so obtained shows that for better convergence rate, modified CSA has outperformed standard CSA

6.1 Conclusion

The promising results of proposed algorithm have given a major contribution to the optimization problems written below:

- Working on nine different benchmark problems the fitness of both cuckoo search and modified cuckoo search algorithms has been compared and in all cases the convergence rate of modified CSA is significantly greater than that of standard CSA.
- The value of fitness function has been decreased by adding an angle to the step size in lévy flights.

6.2 Future scope

- Exploring the performance of modified cuckoo search by applying other benchmark test problems and certain real-life problems.
- The performance of modified algorithm can be emphasized with other optimization algorithms like PSO and GA.

References

[1] Rothlauf, Franz. "Optimization problems." *Design of Modern Heuristics* (2011): 7-44.

[2] Talbi E (2009) Metaheuristics: from design to implementation. John Wiley & Sons, Hoboken.

[3] Yang, Xin-She. *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.

[4] Yang X-S (2009) Harmony search as a metaheuristic algorithm. In: Geem ZW (ed) Music-inspired harmony search: theory and applications. Springer, Berlin, pp 1–14 5508-5518.

[5] Zeitouni, Ofer. "Part II: Random walks in random environment." *Lectures on probability theory and statistics*. Springer Berlin Heidelberg, 2004. 190-312.

[6] "Cuckoo search" Available at: https://en.wikipedia.org/wiki/Cuckoo_search [as accessed on 1st March, 2017]

[7] "Cuckoo search algorithm" Available at: https://www.slideshare.net/afar1111/cuckoo-search-algorithm-40842565 [as accessed on 25th March, 2017]

[8] "Levy flight" Available at: https://en.wikipedia.org/wiki/L%C3%A9vy_flight [as accessed on 1st March, 2017]

[9] R. Mantegna. "Fast, Accurate algorithm for Numerical Simulation of LévyStable Stochastic Processes", Physical Review E, Vol. 49, No. 5, May 1994, pp. 4677-4683.

[10] Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*. IEEE, 2009.

[11] Yang, Xin-She, and Suash Deb. "Engineering optimisation by cuckoo search." *International Journal of Mathematical Modelling and Numerical Optimisation* 1.4 (2010): 330-343.

[12] Gandomi, Amir Hossein, Xin-She Yang, and Amir Hossein Alavi. "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems." *Engineering with computers* (2013): 1-19.

[13] Rajabioun, Ramin. "Cuckoo optimization algorithm." *Applied soft computing* 11.8 (2011): 5508-5518.

[14] S. Walton, O. Hassan, K. Morgan, and M.R. Brown. "Modified cuckoo search: a new gradient free optimisation algorithm." *Chaos, Solitons & Fractals* 44.9 (2011): 710-718.

[15] Valian, Ehsan, Shahram Mohanna, and Saeed Tavakoli. "Improved cuckoo search algorithm for global optimization." *International Journal of Communications and Information Technology* 1.1 (2011): 31-44.

[16] Valian, Ehsan, Shahram Mohanna, and Saeed Tavakoli. "Improved cuckoo search algorithm for feedforward neural network training." *International Journal of Artificial Intelligence & Applications* 2.3 (2011): 36-43.

[17] Leccardi, Matteo. "Comparison of three algorithms for Lévy noise generation." *Proceedings of fifth EUROMECH nonlinear dynamics conference*. 2005.

[18] Soneji, Hetal, and Rajesh C. Sanghvi. "Towards the improvement of cuckoo search algorithm." *Information and communication technologies (wict), 2012 world congress on*. IEEE, 2012.

[19] Zhang, Yongwei, Lei Wang, and Qidi Wu. "Modified Adaptive Cuckoo Search (MACS) algorithm and formal description for global optimisation." *International Journal of Computer Applications in Technology* 44.2 (2012): 73-79.

[20] Bacanin, Nebojsa. "Implementation and performance of an object-oriented software system for cuckoo search algorithm." *International Journal of Mathematics and Computers in Simulation* 6.1 (2012): 185-193.

[21] Milan Tuba, Milos Subotic and Nadezda Stanarevic. "Performance of a modified cuckoo search algorithm for unconstrained optimization problems." (2012).

[22] Praveen Ranjan Srivastava, Chandolu Sravya, Ashima, Sai Kamisetti, and Manogna Lakshmi. "Test sequence optimisation: an intelligent approach via cuckoo search." *International Journal of Bio-Inspired Computation* 4.3 (2012): 139-148.

[23] Ehsan Valian, Saeed Tavakoli, Shahram Mohanna, and Atiyeh Haghi. "Improved cuckoo search for reliability optimization problems." *Computers & Industrial Engineering* 64.1 (2013): 459-468.

[24] D. Rodrigues, L. A. M. Pereira, T. N. S. Almeida, J. P. Papa, A. N. Souza, C. C. O. Ramos and Xin-She Yang. "BCS: A binary cuckoo search algorithm for feature selection." *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*. IEEE, 2013.

[25] Yang, Xin-She, and Suash Deb. "Multiobjective cuckoo search for design optimization." *Computers & Operations Research* 40.6 (2013): 1616-1624.

[26] Nasa-ngium, Patchara, Khamron Sunat, and Sirapat Chiewchanwattana. "Enhancing modified cuckoo search by using Mantegna Lévy flights and chaotic sequences." *Computer Science and Software Engineering (JCSSE), 2013 10th International Joint Conference on*. IEEE, 2013.

[27] Iztok Fister Jr, Xin-She Yang, Dušan Fister, and Iztok Fister. "Cuckoo search: a brief literature review." *Cuckoo search and firefly algorithm*. Springer International Publishing, 2014. 49-62.

[28] Yang, Xin-She, and Suash Deb. "Cuckoo search: recent advances and applications." *arXiv preprint arXiv:1408.5316* (2014).

[29] Zhang, Zhenxing, and Yongjie Chen. "An improved cuckoo search algorithm with adaptive method." *Computational Sciences and Optimization (CSO), 2014 Seventh International Joint Conference on*. IEEE, 2014.

[30] Mahmoudi, Shadi, and Shahriar Lotfi. "Modified cuckoo optimization algorithm (MCOA) to solve graph coloring problem." *Applied soft computing* 33 (2015): 48-64.

[31] Srivastava, Praveen Ranjan. "Software Analysis Using Cuckoo Search." *Advances in Intelligent Informatics*. Springer, Cham, 2015. 239-252.

[32] Manoj Naik, Maheshwari Rashmita Nath, Aneesh Wunnava, Siddharth Sahany and Rutuparna Panda. "A new adaptive Cuckoo search algorithm." *Recent Trends in Information Systems (ReTIS), 2015 IEEE 2nd International Conference on*. IEEE, 2015.

[33] Yuta Umenai, Fumito Uwano, Yusuke Tajima, Masaya Nakata, Hiroyuki Sato, and Keiki Takadama. "A modified cuckoo search algorithm for dynamic optimization problems." *Evolutionary Computation (CEC), 2016 IEEE Congress on*. IEEE, 2016.

[34] El Aziz, Mohamed Abd, and Aboul Ella Hassanien. "Modified cuckoo search algorithm with rough sets for feature selection." *Neural Computing and Applications* (2016): 1-10.

[35] Li, Xiangtao, and Minghao Yin. "A particle swarm inspired cuckoo search algorithm for real parameter optimization." *Soft Computing* 20.4 (2016): 1389-1413

[36] Yang, Xin-She. *Nature-inspired optimization algorithms*. Elsevier, 2014.

[37] "MATLAB code" Available at: https://in.mathworks.com/matlabcentral/fileexchange/29809-cuckoo-search--cs--algorithm?focused=3778771&tab=function [as accessed on 3rd March, 2017].

[38] "Test function for optimization" Available at: https://en.wikipedia.org/wiki/Test_functions_for_optimization [as accessed on 14th April, 2017].

[39] "Benchmark problems" Available at: https://www.cs.cmu.edu/afs/cs/project/jair/pub/volume24/ortizboyer05a-html/node6.html [as accessed on 20th March, 2017].

List of Publications

Published Paper:

Pratibha Sharma, Jagmeet Kaur, Vinay Arora, and Prashant Singh Rana. "Information Retrieval in Web Crawling Using Population Based, and Local Search Based Meta-heuristics: A Review." *Proceedings of Sixth International Conference on Soft Computing for Problem Solving*. Springer, Singapore, 2017.