Machine Learning
(CSE4020)

Review III

A review on Bio-Inspired computing in Finance Management

Faculty: D.P. Archarjya

Slot: F1

Group Members:

Shreyansh Kothari -17BE0833
Munish Kohar - 17BCE0774
Anjali Vashisth – 17BCE0452
Sameeran Bandishti – 17BCE0267
Priyanshu Mittal – 17BCE0027

School of Computer Sciences and Engineering
(SCOPE)

April, 2019

A review on Bio-Inspired computing in Finance Management

Munish Kohar, Shreyansh Kothari, Anjali Vashisth, Sameeran Bandishti and Priyanshu Mittal
VIT University, 17BCE0774, 17BCE0833, 17BCE0452, 17BCE0267, 17BCE0027

***Abstract* - Bio-inspired evolutionary algorithms are probabilistic search methods that mimic natural biological evolution. They show the behaviour of the biological entities interacting locally with one another or with their environment to solve complex problems. This paper aims to analyze the most predominantly used bio-inspired optimization techniques that have been used for stock market prediction and hence present a comparative study between them.**

Index Terms – Firefly, Bee Colony, KNN, Ant Colony, Genetic algorithm, Differential Evolution, Bio-Inspired Computing, Artificial Algae Algorithm.

INTRODUCTION

A stock market or equity market is a public entity at an agreed price for trading corporate stock (shares) and derivatives[1]. The stocks are listed and traded on stock exchanges that are entities of a corporation or mutual organization specialized in bringing together buyers and organization sellers to a list of stocks and securities. Financial forecasting or stock market prediction is one of the hottest research fields recently due to its commercial applications due to the high stakes and the attractive benefits it has to offer.

For common investors, businessmen, brokers and speculators, predicting price movements in stock markets was a major challenge. As more and more money is being invested in the market, the investors are becoming anxious about future stock price trends. However, financial time series includes the most 'noisiest' and 'non-stationary' signals that are very hard to predict. Financial time series is highly volatile and with time the time series changes. There are different techniques available for stock forecasting, but the neural networks[2] are outperforming the statistical techniques due to their ability to follow nonlinear data.

When combined with neural networks, the optimization techniques make the calculation derivative free by reducing the complexity of the computation to a greater extent. An optimization problem attempts to find the best possible solution among all feasible solutions in mathematics as well as in computer science. With large no variables, large search space and non-linear objective functions, the optimization techniques commonly associated with mathematics often fail to solve hard problems. Evolutionary algorithms have been proposed for finding near-optimal solutions to problems to overcome these problems[3-9].

LITERATURE SURVEY

I. Genetic Algorithm

- **Genetic algorithm approach to feature discretization in artificial neural networks for prediction of a stock price index (2000)**: In order to predict the stock price index, this paper proposes genetic algorithms (GAs) approach to feature discretization and determining connection weights for artificial neural networks (ANNs). Previous research proposed many ANN and GA hybrid models for the network training method, selection of subset features, and optimization of topology. However, GA is only used in most of these studies Stock Market Prediction Using Artificial Neural Networks: This study was aimed at finding the best model for the prediction of Istanbul Stock Exchange market index values. Total of 8 sets of predictions, that result from the application of 6 ANN models and two MA were performed. Results were compared using the coefficients of determination for ANN models and using mean relative percentage errors for all of the models.
- **An investigation of genetic algorithms for the optimization of multi-objective fisheries bioeconomic models**: This paper examined the potential applicability of genetic algorithms to the application of bioeconomic models for fisheries. The ultimate goal is to promote the development of broader and more comprehensive fishing models for use in decision-making in management. Such a tool will both contribute to the methodological development and immediate practice of bioeconomic modelling Such a tool will both contribute to the methodological development of bioeconomic modelling and will have immediate practical advantages in increasing the range of management issues that such models can address.
- **Artificial neural networks with evolutionary instance selection for financial forecasting, Expert Systems with Applications**: In this paper, it is proposed an approach to a genetic algorithm (GA) for instance selection for financial data mining in artificial neural networks (ANNs). ANN has preeminent ability to learn but often displays inconsistent and unpredictable performance for noisy data. Furthermore, if the amount of data is so large, it may not be possible to train ANN or

the training task cannot be effectively performed without data reduction. The GA optimizes the connection weights between layers and a selection task for relevant instances simultaneously in this paper. The weights evolved globally to mitigate the well-known limitations of the algorithm of gradient descent. Genetically selected instances also shorten the learning time and improve the performance of predictions. This study applies the proposed model for the analysis of the stock market.

- **Financial forecasting using genetic algorithms, Applied Artificial Intelligence**: This article presented a new inductive machine learning system based on genetic algorithms. The system is applied to financial forecasting, in particular, to predict individual stock performance. The genetic algorithm system is benchmarked against a neural network system on roughly 5000 stock-prediction experiments. Both systems significantly outperform a market in the experiments conducted, i.e. the genetic algorithm system yields better final results. A combined approach individually outperforms either algorithm.
- **Improved Optimization Technique using Hybrid ACO-PSO (2016):** This paper shows the comparison between the optimization of three particle swarm optimization algorithms, the hybrid GA-PSO and the new ACO-PSO hybrid. The mind behind using PSO is to advance the attributes in the ACO, which explains that parameter selection does not depend on experience but on the search for the particles in the PSO.

II. Differential Evolution

- **Hybridizing extreme learning machine and bio-inspired computing approaches for improved stock market forecasting**: The focus of this study is to emphasize how evolutionary computation and ELM methods based on swarm intelligence are used to solve complex problems. Considering the complex nature of the stock market, we proposed four different hybrid methods for accurate effective forecasting to extract meaningful statistics from the stock market. Assessment of the commonly used performance measures in stock market forecasting is presented as a guideline for investors. The results show that four methods proposed are capable of accurately forecasting stock prices.
- **Stock Selection with a Novel Sigmoid-Based Mixed Discrete-Continuous Differential Evolution Algorithm**: This paper proposes a novel stock selection model with discrete and continuous variables, i.e. selection of features and weight optimization, introducing and extending the traditional DE algorithm to a sigmoid-based DE algorithm for this mixed discrete-continuous issue. The novel sigmoid-based DE algorithm makes contributions from two main perspectives compared to the existing DE variants for discrete or mixed discrete-continuous optimization.
- **A New Highly Efficient Differential Evolution Scheme and Its Application to Waveform Inversion**: A better evolutionary strategy called the PES was proposed in this letter, which divides the entire process of evolution into two phases, i.e., full evolution and selective evolution. We proposed a new global optimization method called the HEDE by integrating the PES into CCDE and applying it to waveform inversion. Unlike the CCDE, HEDE individuals are treated adaptively in a generation based on their contributions. In the selective evolution phase, the HEDE finds some individuals after every three generations by the criterion defined in the deletion operation and keeps them from evolving over the next three generations.
- **Eye illusion enhancement using interactive Differential Evolution**: The proposed approach in this paper can be applied to other fields of image processing for the purpose of customizing the proposed algorithm to a targeted application such as enhancing or filtering image contrast. In this area, it builds another direction of research. This work can be assumed as the first step in computerized designing of eye illusion images with higher qualities and amazing characteristics. In fact, the user can define the type of illusion which he/she is interested; then computers can design images which include the targeted illusion with the highest deceptiveness.
- **Fundamentals of Differential Evolution**: The originators observe that differential evolution is much more sensitive to mutation intensity selection than to crossover probability selection. The likelihood of crossover is more like a fine-tuning element.
- **Opposition-based differential evolution:** In this paper, OBL's concept was used to speed up DE. OBL was used to introduce population initialization based on opposition and generation jumping based on opposition. ODE was proposed by embedding these two steps into DE.

III. Artificial Algae Algorithm

- **Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications (Author: Ashraf Darwish)**: This paper takes a look at 9 upcoming bio-inspired optimization algorithms. It performs a GAP analysis of each of these algorithms and also gives real-life applications of these algorithms. Artificial Bee Algorithm is one of those 9 upcoming bio-inspired algorithms.
- **Hybrid Artificial Algae Algorithm for global optimization (Mohit Kumar; J. S. Dhillon)**: This paper introduces a new kind of AAA algorithm called HAAA: Hybrid Algae Algorithm. It uses the simplex search method along with the AAA. The paper also talks about the experimental results and the benchmarks of HAAA.
- **Artificial algae algorithm (AAA) for nonlinear global optimization (Authors: Sait Ali Uymaz, Gulay Tezel,**

EsraYel): This paper is the introductory paper for the AAA algorithm. It consists of the parameters that AAA operates on. It compares the AAA algorithm with other prominent algorithms.

- **An artificial algae algorithm with stigmergic behaviour for binary optimization (Authors: Sedat Korkmaz, Mustafa Servet Kiran)**: This paper focuses on modifying the AAA algorithm for binary problems. It discusses the algorithm required to imitate an XOR gate. The paper then discusses the benchmarks of the modified AAA Algorithm.
- **Artificial algae algorithm with multi-light source for numerical optimization and applications (Authors: Sait Ali Uymaz, Gulay Tezel, EsraYel)**: This paper proposes a solution to improve the efficiency of the AAA algorithm by providing a new light source to every dimension of the dataset. This increases efficiency. This new method gives a different solution for each dimension. The paper also shows experimental results and benchmarks.
- **A new optimization algorithm for solving wind turbine placement problem: Binary artificial algae algorithm (Authors: Mehmet Beşkirli, İsmail Koç, Hüseyin Haklı, Halife Kodaz)**: This paper focuses on a particular application of the AAA algorithm: Wind Turbine Placement. This paper shows how the AAA algorithm could be used to find the optimum way to place wind turbines in a given area. It then discusses and compares the results of two hybrid AAA algorithms.

IV. Particle Swarm Optimization

- **Particle Swarm Optimization: Development, Applications and Resources (2001)**: This is one of the first papers written on Particle Swarm Optimization which describes the underlying theory behind the PSO algorithm and explains the mathematics of the algorithm. Because the algorithm was in its early stages at that time, the paper also describes some of the applications that were thought of at that time, like used to evolve neural network weights
- **Adapting Particle Swarm Optimization to Stock Markets (2005)**: This paper proposes a decision–making model which combines two AI tools: ANN and PSO algorithm. The objective of this model is to analyse the daily stock returns and to make one day forward decisions related to the purchase of the stocks. The variations of different model realizations in the paper conclude that decision– making model with two layers cannot ensure stable results while making decisions in the stock market.
- **A Survey of Particle Swarm Optimization (2009)**: This paper presents a summary of PSO usage in power systems. It depicts many applications in which PSO was efficiently applied, yet it reveals some additional unexplored areas where it can be furthermore employed like safety, restoration, etc. Also, it explains how deregulating all major sections of the electric power sector led to the emergence of a new operation philosophy that will reformulate many optimization problems.
- **Analysis of Particle Swarm Optimization Algorithm (2010)**: According to this paper, Particle swarm optimization is a new heuristic optimization technique based on swarm intelligence. Compared with the other algorithms, the method is very easy, simply completed and it needs fewer parameters, which made it fully developed.
- **Intelligent Parameter Selection MethodParticle Swarm Optimization Algorithm (2011)**: Test results about optimum function in this paper depicts that this design parameters optimization algorithm will help effective realization of PSO algorithm the parameter choosing, and the parameters set by this way are better than the previous parameters in the optimal fitness, the mean of optimal fitness, convergence rate, and etc, especially optimal rate.
- **Improved Optimization Technique using Hybrid ACO-PSO (2016):** This paper shows the comparison among three optimization algorithm particle swarm optimization, hybrid GA-PSO and new hybrid ACO-PSO. The mind behind to use PSO is to advance the attributes in the ACO, which explains that the selection of parameter doesn't depend on experience but on the strong search on the particles in the PSO.

V. Firefly Algorithm

- **Firefly Algorithm, Stochastic Test Functions and Design Optimisation (2010)**: This paper successfully uses the Firefly Algorithm to carry out nonlinear optimisation. The paper validated the algorithms by some set test operations. After designing some new test functions with singularity and stochastic parameters, then used the FA to solve these unrestricted stochastic functions. Also applied it to find a more optimized global solution to the pressure vessel design optimisation.
- **Evaluation Performance Study of Firefly Algorithm (2013)**: According to this paper, FFA performs better, especially on the functions that have multiple peaks. The complexity of the functions had no effect on the FFA as needed except on some functions. FFA and ABC sometimes perform near about the same at the levels of complexity. However, execution time in each replication is dramatically higher when they are compared, especially when both local and global optima found. The parameters can be adjusted to suit for solving various problems with different scales.
- **Firefly Algorithm: Recent Advances and Applications (2013)**: This paper has used the firefly algorithm to reach optimal balance and depict that firefly algorithm can provide a fine balance of exploitation and searching. It is

also shown that the firefly algorithm requires far fewer function testing. However, the huge differences between intermittent search theory and the technique of metaheuristics in practice also state that there is still a large difference between our understanding of algorithms and the actual behaviour of metaheuristic.

- **Performance Evaluation of a New Modified Firefly Algorithm (2014)**: In this work, a new improved Firefly algorithm is proposed and its efficiency is compared with an established version of Firefly algorithm. Also, the new algorithm performs better than other algorithms with more accuracy and quicker convergence. This conclusion is based on the four benchmark functions used in the experiment and the results of the new algorithm may vary for some other set of benchmark functions.
- **A Hybrid Firefly Algorithm (2017)**: In this paper, mating behaviour is added to the original firefly algorithm. This paper presents a Hybrid firefly algorithm (HFA) algorithm, in which the crossover operator of Genetic Algorithm is introduced in to improve the original FA algorithm. With the new parameter, information is exchanged fully between fireflies and the good individuals are used.
- **Intelligent Firefly Optimization System (2017)**: In this paper, the new intelligent firefly technique has been developed for optimization problems. The potential of the system has been tested using different benchmark functions. The experimental result depicts that the technique could solve the different dimensions problems in small time and in fewer iterations. Moreover, the system has been applied to handle with constraining engineering problems.

VI. Artificial Fish Swarm Optimization

- **A survey of the state-of-the-art, hybridization, combinatorial and indicative applications(2014)**: This paper is a review of AFSA algorithm and describes the evolution of this algorithm along with all improvements, its combination with various methods as well as its applications. There are many optimization methods which have an affinity with this method and the result of this combination will improve the performance of this method.
- **Fish swarm optimization algorithm applied to engineer system design(2014)**: In this work, Fish Swarm Optimization Algorithm (FSOA) based on the social behaviour of fish colonies, was applied to solve different design problems. The simulation results were compared with those obtained from other competing evolutionary algorithms.
- **A review of artificial fish swarm optimization methods and applications (2012)**: This algorithm is capable of solving the problems by inspiration from the en masse movement of fishes. Fishes show different behaviors including seeking for food, following other fishes, protecting the group against threats and stochastic search. These behaviours have been employed in the AFSA and an acceptable result has been obtained.
- **A weak signal detection method based on artificial fish swarm optimized matching pursuit(2009)**: A new improved matching pursuit algorithm is proposed. The mathematical model of searching algorithms based on artificial fish swarm is established; the artificial fish swarm with the advantages of distributed parallel searching ability, strong robustness, good global astringency, and insensitive preferences are employed to search the best matching atoms.
- **A self-adaptive control algorithm of the artificial fish formation(2009)**: With the deep study of swarm intelligence, biologists found that fish swarm changes information gradually in time during their movement. This formation change leads to better and more effective access to evade predator an opportunity to capture food so that the group's overall performance is improved. The architecture of artificial fish formation is established based on the behavioural model of an artificial fish swarm.
- **Notice of Retraction Artificial Fish-Swarm Algorithm with Chaos and Its Application(2010)**: This paper is a review of AFSA algorithm and describes the evolution of this algorithm along with all improvements, its combination with various methods as well as its applications.

VII. Bacterial Foraging Optimization

- **A review of bacterial foraging optimization and its applications (2012)**: This paper gives an extensive survey of the Bacterial Foraging Optimization Algorithm and its variations as it exists to date. The worldwide search approach of BFO results in bigger convergence time.
- **Analysis and improvement of the bacterial foraging optimization algorithm (2014)**: The calculation is improved in the disposal dispersal and chemotaxis tasks, in view of the essential BFO. By restricting the scope of the elimination- dispersal of bacteria, the escape formula can be maintained a strategic distance from and the convergence velocity of the calculation is viably improved. Besides, the impact of the progression estimate in the chemotaxis activities on the calculation has been examined.
- **Improved bacteria foraging optimization algorithm for flexible job shop scheduling problem (2015)**: An improved bacterial foraging algorithm is proposed in this paper and connected to look for the ideal arrangement on FJSP. Contrasted and the conventional calculation, the enhancement capacity of this technique is increasingly precise. Contrasted and the improved hereditary calculation, the improved strategy can lessen the iterative time and incredibly diminish the settling time.

- **Bacterial foraging optimization and adaptive version for economically optimum sitting, sizing and harmonic tuning orders setting of LC harmonic passive power filters in radial distribution systems with linear and nonlinear loads (2015)**: This paper presents bacterial foraging optimization (BFO) algorithm and its versatile form to improve the arranging of inactive symphonious channels (PHFs). The imperative issue of utilizing PHFs is deciding area, size and consonant tuning requests of them, which is achieved standard dimensions of harmonic distortion with applying the base expense of detached channels.
- **Bacterial Foraging Optimization Algorithm for neural network learning enhancement (2011)**: This paper encourages us to give an alternative answer for the examination presented, Bacterial Foraging Optimization Algorithm to be used in feedforward neural system to upgrade the learning procedure and improve its combination rate and grouping exactness. The created Bacterial Foraging Optimization Algorithm Feedforward Neural Network (BFOANN) is analyzed against Particle Swarm Optimization Feedforward Neural Network (PSONN).
- **Improved algorithm of bacterium foraging and its application (2012)**: This paper structured an improved versatile advance and stop condition for taking care of nearby ideal and untimely issues, and connected this improved calculation to the adaptable employment shop booking Problem (FJSP).

VIII. K Nearest Neighbour

- **Safar, M., Ibrahimi, D., Taniar, D.: Voronoi-based reverse nearest neighbour query processing on spatial networks**: This paper presented four different RNN query types, and proposed algorithms to efficiently process them. The proposed algorithms are novel and are not based on Euclidean distance. The proposed approaches that used network expansion mechanism like PINE together with NVD, offer many benefits, including reducing the number of computations required to answer an RNN query.
- **Continuous nearest neighbour monitoring in road networks**: • In this paper, they present a novel directional graph model for road networks to simultaneously support these two kinds of continuous k-NN queries by introducing unidirectional network distance and bidirectional network distance metrics. Considering the computational capability of the mobile client to locate the edge containing it, they use a memory-resident hash table and linear list structures to describe the moving objects and store the directional model.
- **The k-nearest neighbour join: turbo charging the KDD process**: In this paper, they propose an important, third similarity join operation called the k-nearest neighbour to join, which combines each point of one point set with its k nearest neighbours in the other set. They discover that many standard algorithms of Knowledge Discovery in Databases (KDD) such as k-means and k-medoid clustering, nearest neighbour classification, data cleansing, postprocessing of sampling-based data mining, etc. can be implemented on top of the k-nn join operation to achieve performance improvements without affecting the quality of the result of these algorithms.
- **Efficient evaluation of all-nearest-neighbour queries**: In this paper, they challenge the common practice of using R*-tree index for speeding up the ANN computation. They propose an enhanced bucket quadtree index structure, called the MBRQT, and using extensive experimental evaluation show that the MBRQT index can significantly speed up the ANN computation. In addition, we also present the MBA algorithm based on a depth-first index traversal and bi-directional node expansion strategy.
- **All-nearest-neighbors queries in spatial databases**: In this paper, they study alternative methods for processing ANN queries depending on whether A and B are indexed: Our algorithms are evaluated through extensive experimentation using synthetic and real datasets. The performance studies show that they are an order of magnitude faster than a previous approach based on closest-pairs query processing.
- **Efficient index-based KNN join processing for high-dimensional data. Inf. Softw. Technol**: In this paper, they examine the problem of processing K-nearest neighbour similarity join (KNN join). KNN join between two datasets, R and S, returns for each point in R its K most similar points in S. They propose a new index-based KNN join approach using the iDistance as the underlying index structure. They first present its basic algorithm and then propose two different enhancements. In the first enhancement, they optimize the original KNN join algorithm by using approximation bounding cubes. In the second enhancement, they exploit the reduced dimensions of data space.

IX. ANT Colony Optimization

- **Ant algorithm for the multi-dimensional knapsack problem.In: International Conference on Bioinspired Optimization Methods and their Applications**: In this paper, they show that ACO algorithms are known to have scalability and slow convergence issues, here we have augmented the traditional ACO algorithm with a unique random local search, which not only produces near-optimal solutions but also greatly enhances convergence speed. A comparative analysis with other state-of-the-art heuristic algorithms based on public MMKP dataset shows that in all cases our approaches outperform others. We have also shown that our algorithms find near optimal (within 3% of the optimal value) solutions within milliseconds, which makes our approach very attractive for large scale real-time systems

- **An adaptive tabu-simulated annealing for concave cost transportation problems**: • In this study, they propose a hybrid algorithm based on the concepts borrowed from tabu search (TS) and simulated annealing (SA) to solve the ccTP. This algorithm, called ATSA (adaptive tabu-simulated annealing), is an SA approach supplemented with a tabu list and adaptive cooling strategy. The effectiveness of ATSA has been investigated in two stages using a set of TPs with different sizes. The first stage includes performance analysis of ATSA using SA, ASA (adaptive simulated annealing) and TS, which are basic forms of ATSA. In the second stage, ATSA has been compared with the heuristic approaches given in the literature for concave cost transportation problems (ccTP).
- **An ant colony algorithm for solving budget constrained and unconstrained dynamic facility layout problems**: In this paper, they make use of the ant colony optimization (ACO) algorithm to solve the DLP by considering the budget constraints. The paper makes the first attempt to show how the ACO can be applied to DLP with the budget constraints. In the paper, example applications are presented and computational experiments are performed to present suitability of the ACO to solve the DLP problems. Promising results are obtained from the solution of several test problems
- **An improved ant colony optimization for vehicle routing problem**: This paper proposes an improved ant colony optimization (IACO), which possesses a new strategy to update the increased pheromone, called ant-weight strategy, and a mutation operation, to solve VRP. The computational results for fourteen benchmark problems are reported and compared to those of other metaheuristic approaches
- **New metaheuristic approaches for the edge-weighted k-cardinality tree problem**: In this paper, they propose three metaheuristic approaches, namely a Tabu Search, an Evolutionary Computation and an Ant Colony Optimization approach, for the edge-weighted k-cardinality tree (KCT) problem. According to them, this problem is an NP-hard combinatorial optimization problem that generalizes the well-known minimum weight spanning tree problem.
- **An ant-based algorithm for finding a degree-constrained minimum spanning tree:** In this paper, they propose a learning automata-based heuristic algorithm to solve the minimum spanning tree problem in stochastic graphs wherein the probability distribution function of the edge weight is unknown. The proposed algorithm taking advantage of learning automata determines the edges that must be sampled at each stage. As the presented algorithm proceeds, the sampling process is concentrated on the edges that constitute the spanning tree with the minimum expected weight. The proposed learning automata-based sampling method decreases the number of samples that need to be taken from the graph by reducing the rate of unnecessary samples.

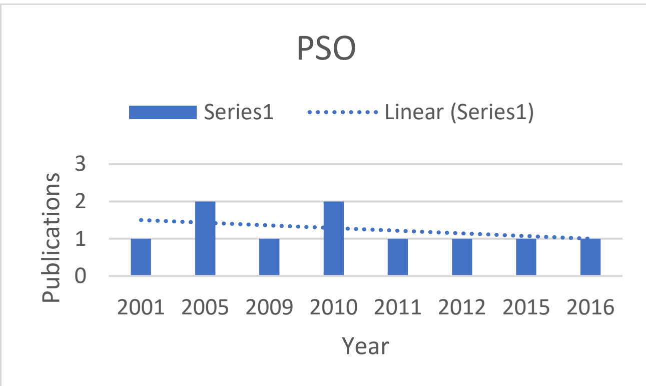
X. Artificial Bee Colony

- **Bio-inspired computing**: This paper takes a look at 9 upcoming bio-inspired optimization algorithms. It performs a GAP analysis of each of these algorithms and also gives real-life applications of these algorithms. Artificial Bee Algorithm is one of those 9 upcoming bio-inspired algorithms
- **Artificial bee colony algorithm**: This paper is exclusively centred on the Artificial Bee Colony Algorithm. It introduces the concept behind the ABC algorithm along with mathematics. It then goes ahead to analyse all the significant recent research taking place in this field. It ends with a proposal of the challenged and future research highlights for ABC
- **Overview of Artificial Bee Colony (ABC) algorithm and its applications**: This paper is a review paper that encompasses all the research made in ABC's till the year 2012. It also talks about the key features of the algorithm and discusses it's performance characteristics.
- **Self-adaptive constrained artificial bee colony for constrained numerical optimization**: This paper focuses on a hybrid method of ABC that has the same core idea as ABC. It talks about an ABC with constrained parameters which helps in faster convergence. It then discusses the experimental results of this hybrid ABC.
  Optimizing architectural properties of Artificial Neural Network using proposed Artificial Bee Colony algorithm: This paper proposes to use the ABC algorithm to find the appropriate weights and Activation functions for an ANN so as to maximize accuracy and minimize error. It then discusses experimental evidence of the increased efficiency.
- **Velocity based artificial bee colony algorithm for high dimensional continuous optimization problems:** This paper proposes a Hybrid ABC called VABC. ABC's have a slow and inefficient onlooker phase. This can be overcome using VABC, which uses a PSO in the onlooker phase of the algorithm

RESEARCH METHODOLOGY

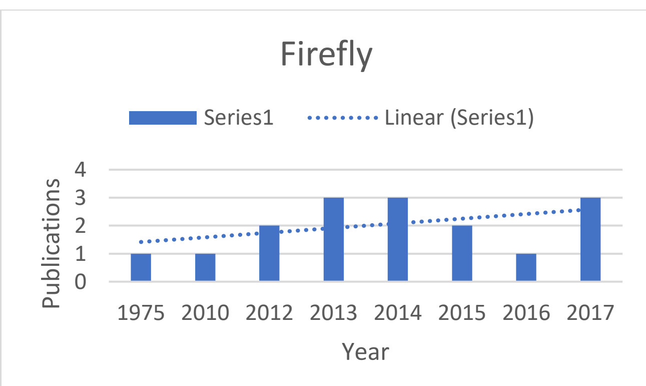
I. Particle Swarm Optimization (Introduced in 1995)

Over the years PSO has gained much popularity and is used in various fields for optimization. However, any new advancement with respect to Particle Swarm Optimization has not come to light. Researchers, in recent years, have explored other bio-inspired algorithms that have yet to be understood completely trying to provide insight. PSO is one of the oldest Bio-Inspired Algorithm but yet very efficient and it is applied in various departments of not only sciences but also management.



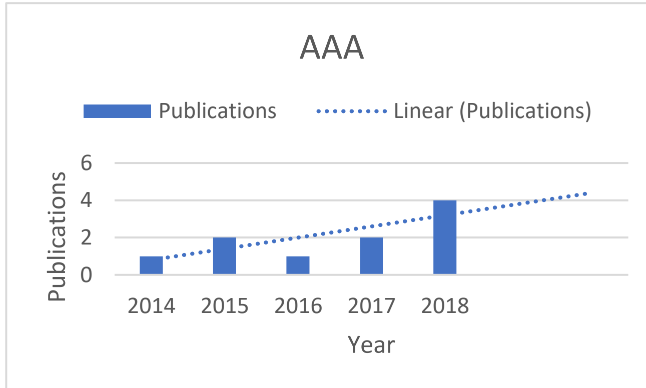
II. Firefly Algorithm (Introduced in 2008)

Though the algorithm was introduced in 2008, it was thought of and mathematically formulated in 1975. Since then, no researcher was keen to create a firefly algorithm until Xin-She Yang did it in 2008. Since then there have been considerable developments in the research of firefly algorithm. Researchers are trying to apply it in different fields and trying to optimize its fitness.



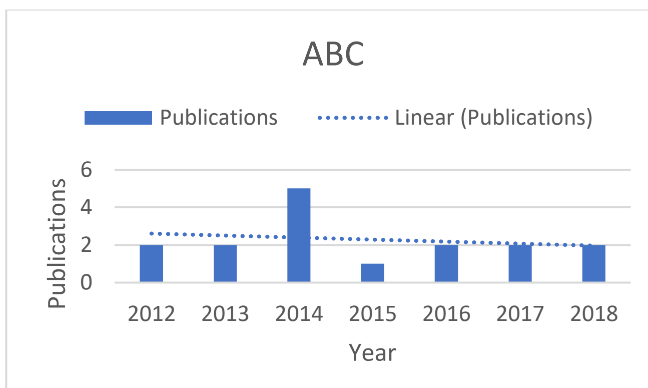
III. Artificial Algae Algorithm

The title and author data are in one-column format, while the rest of the paper is in a two-column format. To accomplish this, most word processors have a section break that is installed to separate the one and two-column format. There are two ways to set up this format: 1) Use this template as a guide, 2) make your own. If you wish to make your own, it is suggested that you open a new document and begin by inserting the title and author information in the standard one-column format. Please adhere to the following style guidelines:



IV. Artificial Bee Colony

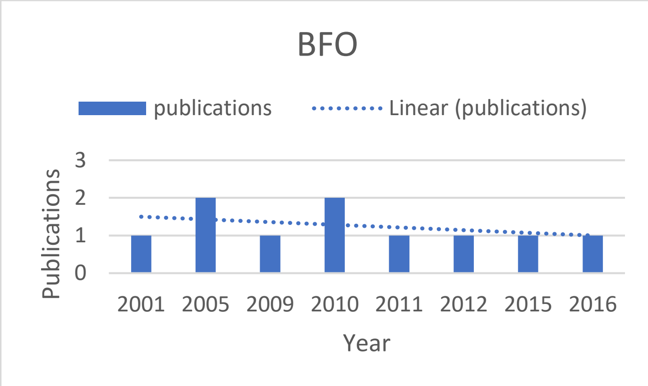
The research and evaluation of the ABC algorithm were done through previously published papers in similar fields. As can be seen in the graph, the ABC algorithm took off in 2012 and had its peak in 2014. The number of papers published has been consistent ever since. ABC algorithm has seen a lot of success over the years with more papers being published proposing modified ABC algorithms that cover the shortcomings of the basic ABC algorithm.



V. Bacterial Foraging Optimization

Bacterial Foraging Optimization (BFO) algorithm was first proposed by Passino in 2002. until 5 decades many other evolutionary strategies including genetics were brought up making this algorithm to lag behind. However, since its origin, BFOA has drawn the consideration of scientists from differing fields of learning particularly because of its biological inspiration and graceful structure. They are trying to hybridize it with several other newly formed genetic algorithms. Many improved application-based variations of

BFO have likewise come up prompting an exceptional decrease in assembly time and with higher accuracy. But the complete capability of BFOA optimizer is yet to be explored.

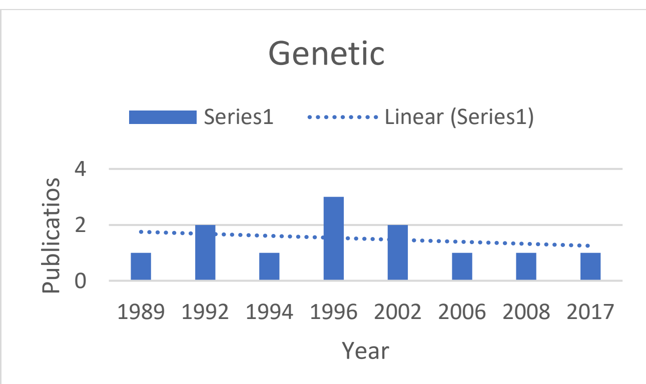


VI. Genetic Algorithm

There are many publications which give excellent introductions to genetic algorithms: see Holland (1975), Davis (1987), Goldberg (1989b), Davis (1991), Beasley et al. (1993), Forrest (1993), Reeves (1995), Michalewicz (1996), Mitchell (1996), Falkenauer (1998), Coley (1999), and Man et al. (1999). An excellent work which brings together the early pioneering work in the field is Fogel (1998). There are a number of journals and conferences which publish papers concerned with genetic algorithms. The key conferences and journals are listed below, but remember that papers on Genetic Algorithms are published in many other outlets too.

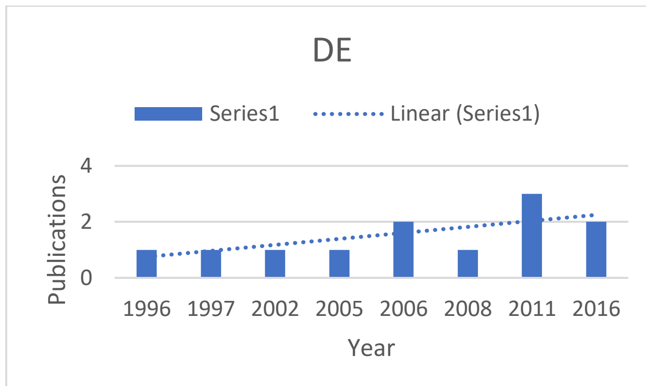
Conferences:

- Congress on Evolutionary Computation (CEC)
- Genetic and Evolutionary Computation Conference (GECCO)
- Parallel Problem Solving in Nature (PPSN)
- Simulated Evolution and Learning (SEAL)



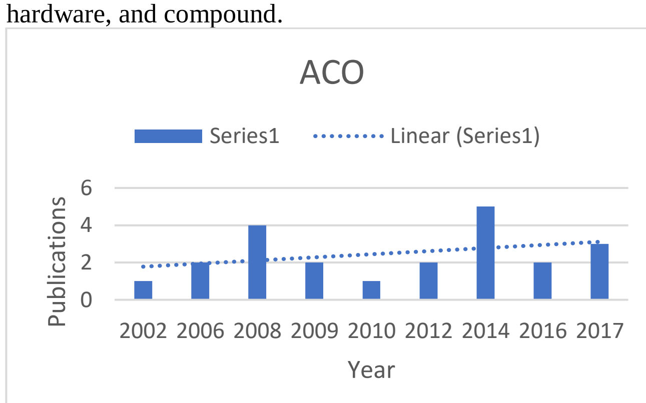
VII. Differential Evolution

Mustafa E. Abdual-Salam et al[6] presented a comparison between two stochastic, population-based and real-valued algorithms. These algorithms are namely Differential Evolution (DE) and Particle Swarm Optimization (PSO). These algorithms are used in the training of feed-forward neural network to be used in the prediction of the daily stock market prices David Enke, Manfred Grauer and Nijat Mehdiyev[17] introduced three-stage stock market prediction system. In the first phase, Multiple Regression Analysis is applied to define the economic and financial variables which have a strong relationship with the output. In the second phase, Differential Evolution-based type-2 Fuzzy Clustering is implemented to create a prediction model. For the third phase, a Fuzzy type-2 Neural Network is used to perform the reasoning for future stock price prediction. Nizar Hachicha et al[18] investigated the development of a new modelling technique using fuzzy sets optimized through differential evolution. Tetsuyuki Takahama et al [19] applied differential evolution to the structural learning of neural networks and shown that DE can reduce the proper number of ineffective parameters and find better estimation models, which have smaller estimation errors for test data.



VIII. ANT Colony Optimization

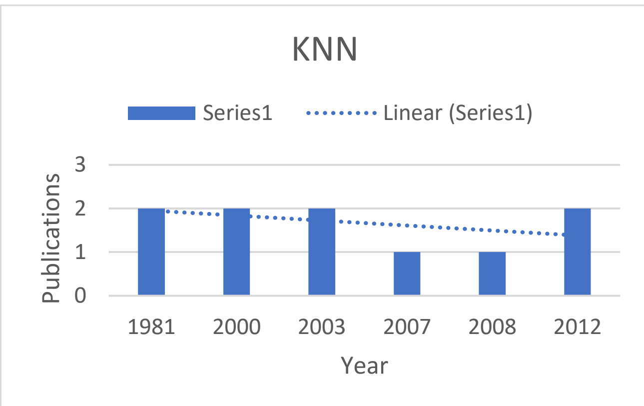
In this phase, parallel applications for this algorithm are introduced. They fall under three categories: software, hardware, and compound.



IX. K Nearest Neighbour

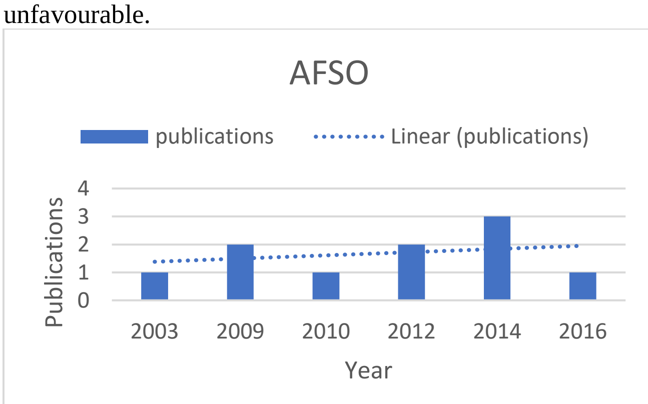
KNN modelling works around four parameters:

- Features: The variables based on which similarity between two points is calculated
- Distance function: Distance metric to be used for computing similarity between points
- Neighbourhood (k): Number of neighbours to search for
- Scoring function: The function which combines the labels of the neighbours to form a single score for the query point



X. Artificial Fish Swarm Optimization

Nowadays, this algorithm in nature have been used in the development of methodologies for solving a variety of real-world optimization problems. Although, it has not yet been unanimously recognized by the international academic community. By 34 Benchmark Functions tested with AFSA, the result evaluation for functions that are applicable and not applicable by AFSA is summarized. After gaining the optimal solution researchers need the influence of parameter values. The system is started first in a set of randomly generated potential solutions, and then performs the search for the optimum one interactively in 2006 after 2002. This new method is considered best among other all warm intelligence algorithms. Yet it's high complexity makes it bit unfavourable.



ALGORITHMS

I. Differential Evolution

Algorithm:

1. The first step is the random initialization of the parent population. Randomly generate a population of (say) NP vectors, each of N dimensions.
2. Calculate the objective function value f(Xi) for all X
3. Select three points from the population and generate perturbed individual
4. Calculate the objective function value.
5. Choose better of the two (function value at target and trial point) for the next generation.
6. Check whether convergence criterion is met, if yes then stop else repeat.

II. Artificial Fish Swarm Optimization

Algorithm:

Step 1: Initialize
Step2: Calculate the fitness value
Step3: Each artificial $fish_i$ $_{(i= 1,2,….,N)}$
Step3.1: Following; check if the previous state is better than the current one after following, and if so, go to Step4, else go to Step3.2;
Step3.2: Clustering; check if the previous state is better than the current one after clustering, and if so, go to Step4, else go to Step3.3;
Step3.3: Foraging;
Step4: now best value currently is updated;
Step5: distance among fish swarm is update, d $_{i,j, (I,j =1,2,…,N)}$
Step6: if maximum evolution algebra is reached, then exit; else go to Step3

This administrator adds to the individual and aggregate developments of fishes in the swarm. Each fish refreshes its new position by utilizing the Equation (1):

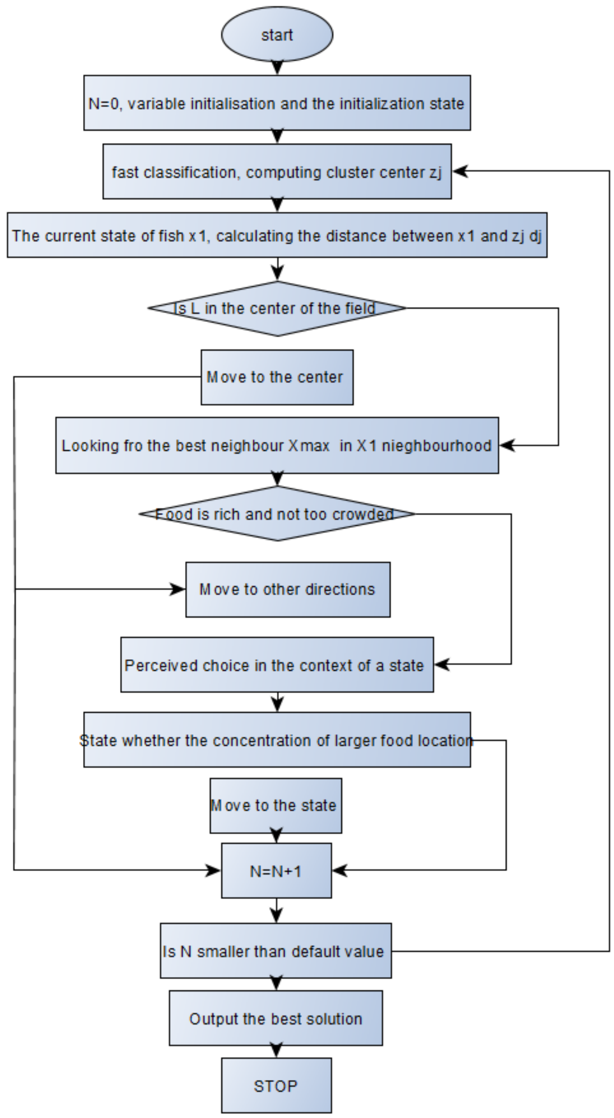
x_i^(t+1)=x_i^t+rand ×s_ind

where xi is the final position of fish i at current generation, rand is a random generator and sind is a weighted parameter.

III. Bacterial Foraging Optimization

D = Dimension of search. It is a number of parameters to be upgraded. if you have three parameters to be enhanced, state m1,m2,m3, then at that point D will be equivalent to three.

B= Number of bacteria in the population. It ought to be equivalent to a number of sets of focuses acquired by discretizing the advancement parameter. Assume m1, m2, m3 every parameter is discretized to give ten values in range [1, 2]. Then each set will represent a point in space (m1,m2,m3-



coordinates). Hence there will be ten locations in the optimization domain. So, ten bacteria are required to be placed at these points to start the search.

NC = Number of chemotaxis steps that a bacterium takes in a complete chemotaxis procedure before opting any reproduction.
Ns = Number of swimming steps
Nre = Number of reproduction steps
Ned = Number of elimination and dispersal steps
Ped = Elimination and dispersal probability
C(i) = Unit run-length

Step 1: Elimination and dispersal loop l = l+1

Step 2: Reproduction loop l = l+1

Step 3: Chemotaxis loop j = j+1
For i= 1,2,3.....B, a chemotaxis step for ith bacterium will be
as follows:

Calculate fitness function Pi,j,k,l.
Save this value in Plast= Pi,j,k,l so that we can find better fitness (cost) via the run. Tumble: Generate direction vector del (i) is assigned a new value which is a random number lying between [-1,1].
Move using equation (1) …
fi,j+1,k,l(m1,m2,m3) = fi,j,k,l(m1,m2,m3) + C(i) (del(i))/√(〖del〗 ^(T) del(i))
Calculate fitness function Pi,j,k,l
Swim : (i) Initialize swim counter sc = 0.
(ii) If sc< Ns
If Pi,j,k,l < Plast , Let Plast= Pi,j,k,l, and use equation (1) given in step e) to move in the same direction.
Use the newly formedlocation fi,j,k,l for new values of m1,m2,m3 to calculate Pi,j,k,l and continue in the loop.
Else sc=Ns
Do the same process for next bacterium i=i+1, go step b) if i≠S.

Step 4: If j< Nc, go to step 3 for the next chemotaxis step as the chemotaxis process not complete.

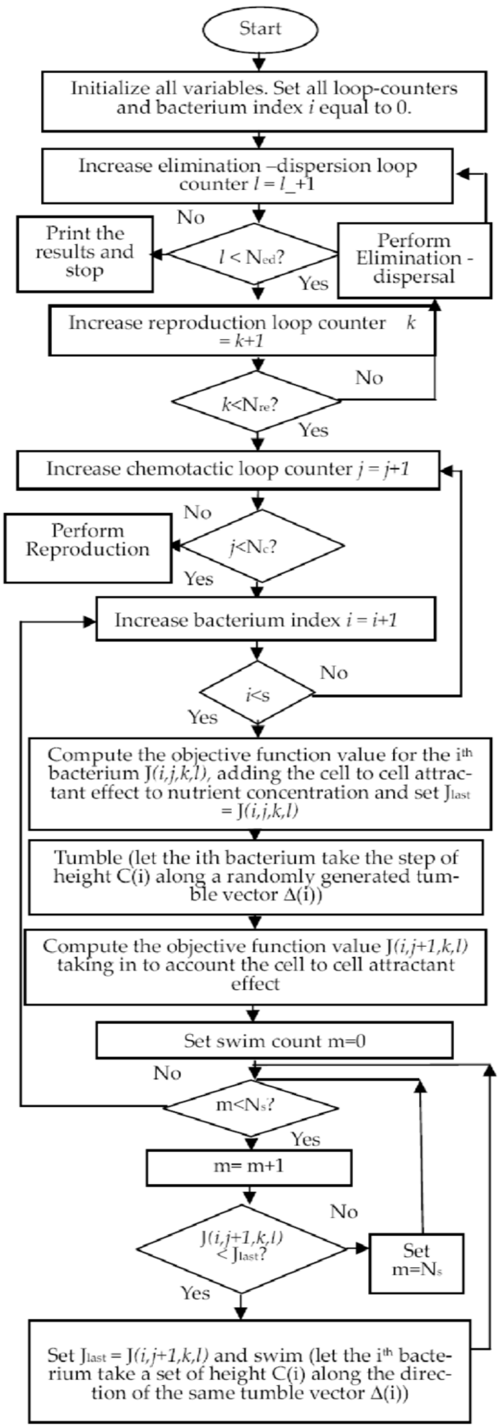
Step 5: Reproduction. With (current) values of k,l, calculate overall fitness (cost function) ∑_(j=1)^Nc P^(i,j,k,l) for each ith bacterium and arrange the fitness in descending order. Lesser value of cost function means high fitness.

Step 6: Half the bacteria with less fitness will die and the other half reproduces. They are split into two parts and are placed at their parents location. So, crowd number remains the same.

Step 7: If k< Nre, go to step 2. increase the counter for reproduction and restart new chemotaxis process.

Step 8: Elimination-dispersion. Eliminate the bacterium with probability Ped and disperse one at a random location in the optimization space.

Step 9: If l< Ned, go to step1. Else end.*II. Genetic*



IV. K Nearest Neighbour

Since KNN predictions are based on the intuitive assumption that objects close in distance are potentially similar, it makes good sense to discriminate between the K nearest neighbours when making predictions, i.e., let the closest points among the K nearest neighbours have more say in affecting the outcome of the query point.

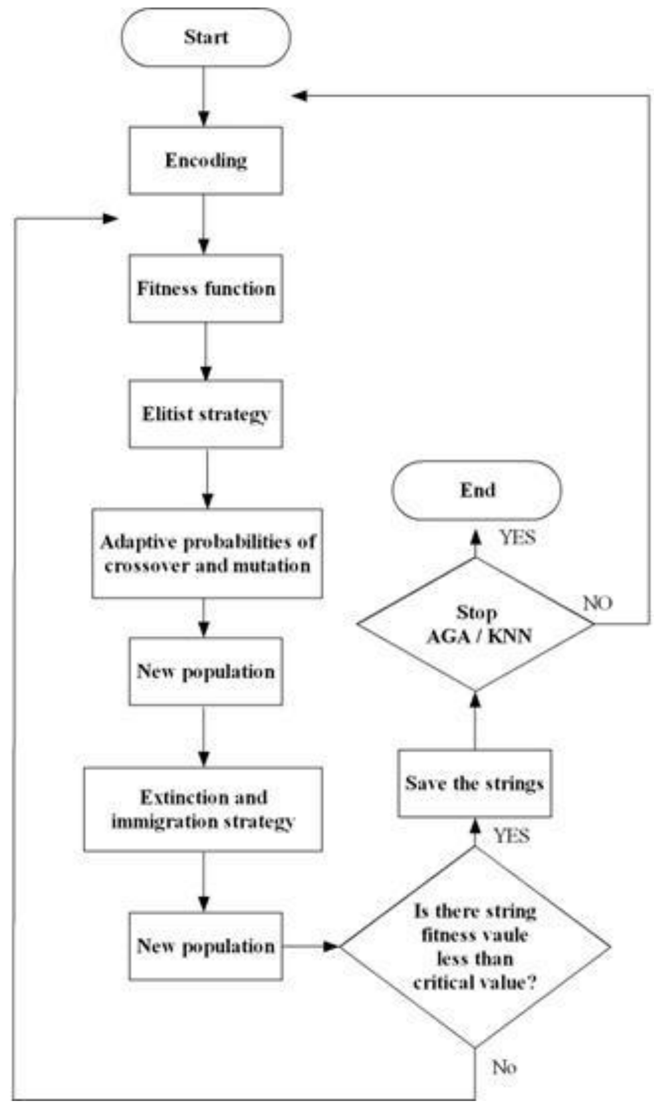
$$W(x, p_i) = \frac{exp(-D(x, p_i))}{\sum_{i=1}^{k} exp(-D(x, p_i))}$$

$$\sum_{i=1}^{k} W(X_0, X_i) = 1$$

Thus, for regression problems, we have:

$$y = \sum_{i=1}^{k} W(X_0, X_i) y_i$$

Flowchart:



and proposed the first Genetic algorithm in 1989. It provides excellent solutions to search and optimization problems and relies on bio-inspired operators like crossover, selection and crossover.

V. ANT Colony Optimization

```
Begin
    Initialize
    While stopping criterion not satisfied do
        Position each ant in a starting node
        Repeat
            For each ant do
                Choose next node by applying the state transition rule
                Apply step by step pheromone update
            End for
        Until every ant has built a solution
        Update best solution
        Apply offline pheromone update
    End While
End
```

Mathematical Notation:

$$\Delta\tau_{i,j}^{k} = \begin{cases} \dfrac{1}{L_k} \\ 0 \end{cases}$$

VI. Artificial Bee Colony

1. Initialize the food sources randomly
2. Employed bee goes to a food source and evaluates it
3. Onlooker bees choose best food sources by taking info from employees bees
4. After onlookers abandon food sources, scouts find new food sources
5. The best food source yet is stored.
6. Repeat from 2 until requirements are met

1. Initialization of the population:

First, the algorithm creates a set of uniformly distributed solutions. The solutions are vectors with the same number of dimensions as the number of attributes.

2. Employees Bees Phase:

In this phase, the assumed solutions are modified according to the amount of food employees bees find in that area. Old solutions are discarded as new and better ones are found

3. Onlooker Bee Phase:

All the employed bees transmit the information of their current food sources to the onlooker bees who decide which is the best food source. Others are discarded. It uses the above formula to calculate the probability of a certain food source being the best solution.

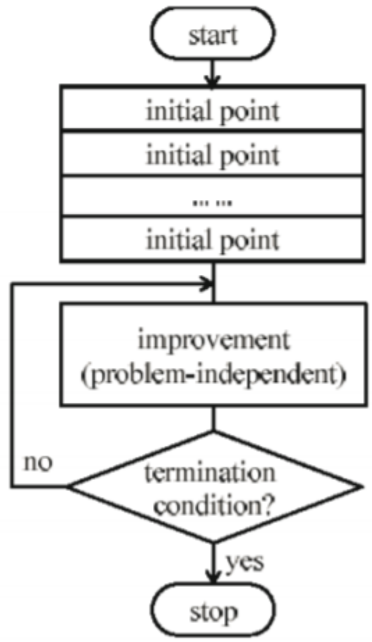
4. Scout Bee Phase:

A bee on an abandoned food source becomes a scout bee that is responsible for finding new sources of food. The bee waits for a certain number of cycles of not being selected as the best food source and this number is called the ‘limit for abandonment’.

VII. Genetic Algorithm

The genetic algorithm uses three main types of rules at each step to create the next generation from the current population:

• *Selection rules* select the individuals, called parents, that contribute to the population at the next generation.
• *Crossover rules* combine two parents to form children for the next generation.
• *Mutation rules* apply random changes to individual parents to form children.



(b) genetic algorithm

(1) Generate a random population of ‘n’ chromosomes
(2) Evaluate the fitness of each chromosome in the population
(3) Create a new population by repeating the following steps:
i. Select two parent chromosomes from the population according to their fitness.
ii. With a cross over probability cross over the parents to form new offspring(children)
iii. With a mutation, probability mutate the new offspring at each locus.
iv. Place the new offspring in the population
(4) Use the new generated population for a further run.
(5) Stop if stopping criteria is met.

VII. Particle Swarm Algorithm

For each particle
{
  Initialize particle
}
Do until maximum iterations or minimum error criteria
{
  For each particle
  {
    Calculate Data fitness value

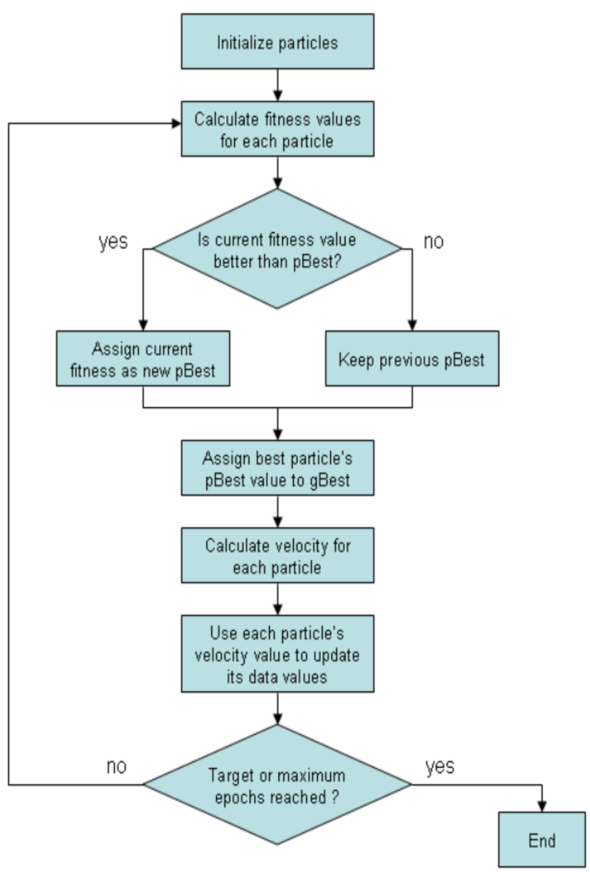
    If the fitness value is better than pBest

    {

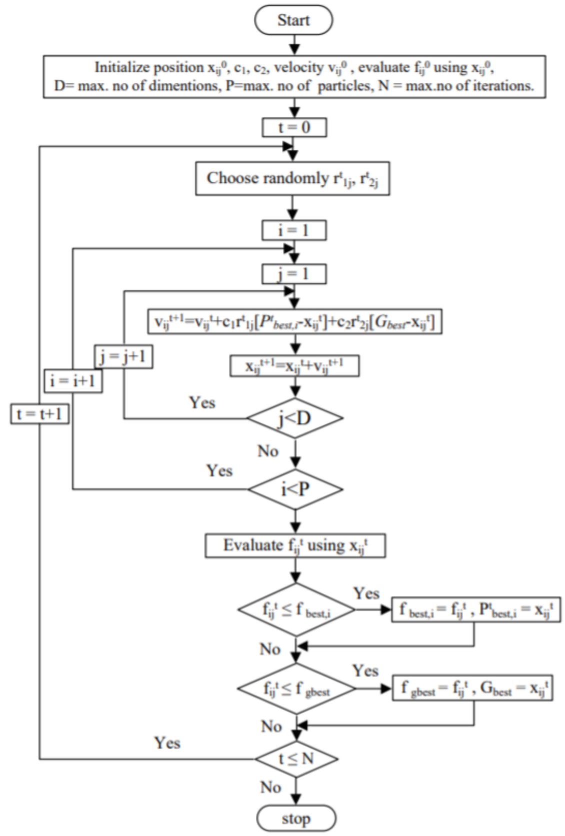
      Set pBest = current fitness value

    }
  If pBest is better than gBest
    {
      Set gBest = pBest
    }
  }

  For each particle
  {
    Calculate particle Velocity
    Use gBest and Velocity to update particle Data



Model to find the Global Minima Using PSO:



Each particle's velocity is updated using this equation:

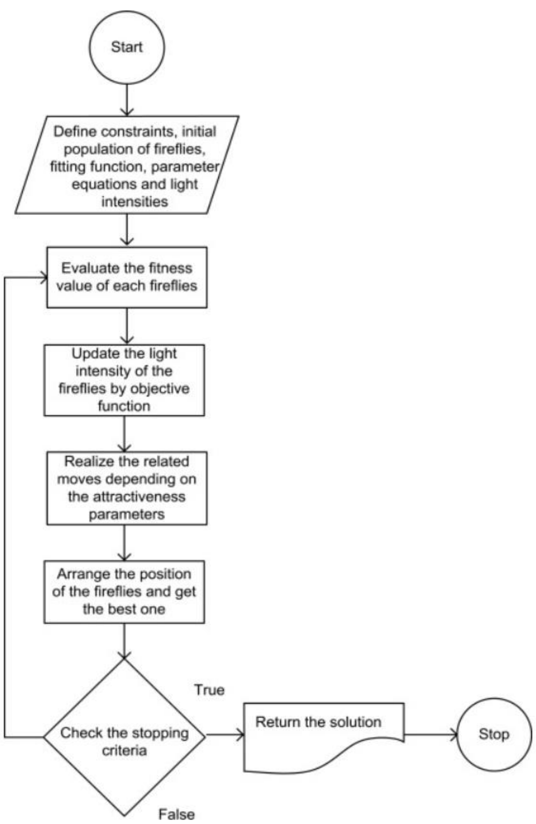
vi (t+1) = wvi (t) + c1r1[xvi (t) − xi (t)] + c2r2[g (t) − xi (t)]

- i is the particle index
- w is the inertial coefficient
- c_1, c_2 are acceleration coefficients, $0 \le c_1, c_2 \le 2$
- r_1, r_2 are random values ($0 \le r_1, r_2 \le 1$) regenerated every velocity update v_i (t) is the particle's velocity at time t
- x_i (t) is the particle's position at time t
- x_{vi} (t) is the particle's individual best solution as of time t
- g (t) is the swarm's best solution as of time t

Each particle's position is updated using this equation:
xi(t +1) = xi(t) + vi(t+1)

VIII. Firefly Algorithm

Begin
Objective function $f(x), x = (x_1, \ldots, x_d)^T$
Generate initial population of n fireflies $x_i, i = 1, 2, \ldots, n$
Formulate light intensity I so that it is associated with $f(x)$
While ($t <$ MaxGeneration)
Define absorption coefficient γ
for $i = 1 : n$ (n fireflies)
for $j = 1 : n$ (n fireflies)
if ($I_j > I_i$),
move firefly i towards j
end if
Vary attractiveness with distance r via $\exp(-\gamma r$^2)
Evaluate new solutions and update light intensity
end for j
end for i
Rank the fireflies and find the current best
end while
Post-processing the results and visualization
end



In the simplest case for maximum optimization problems, the brightness I of a firefly at a particular location x can be chosen as I(x) ∝ f(x). However, the attractiveness β is relative, it should be seen in the eyes of the beholder or judged by the other fireflies. Thus, it will vary with the distance rij between firefly i and firefly j. In addition, light intensity decreases with the distance from its source, and light is also absorbed in the media, so we should allow the attractiveness to vary with the degree of absorption. In the simplest form, the light intensity I(r) varies according to the inverse square law I(r) For a given medium with a fixed light absorption coefficient γ, the light intensity I vary with the distance r. That is

$$I = I_0 e^{-\gamma r},$$

where I0 is the original light intensity. As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the attractiveness β of a firefly by

$$\beta = \beta_0 e^{-\gamma r^2}$$

where β0 is the attractiveness at r = 0. The distance between any two fireflies i and j at xi and xj can be the Cartesian distance rij = ||xi −xj ||2 or the ℓ2-norm. For other applications such as scheduling, the distance can be a time delay or any suitable forms, not necessarily the Cartesian distance. The movement of a firefly i is attracted to another more attractive (brighter) firefly j is determined by-

$$\mathbf{x}_i = \mathbf{x}_i + \beta_0 e^{-\gamma r_{ij}^2} (\mathbf{x}_j - \mathbf{x}_i) + \alpha \boldsymbol{\epsilon}_i$$

DISCUSSIONS

I. Genetic Algorithm

The Darwinian principle of "survival of the fittest" and natural evolution inspires the genetic algorithm. John Holland presented algorithms developed on Darwin's 1960 hypothesis and subsequently worked on it by his Goldberg understudy and in 1989 proposed the first Genetic algorithm. It offers excellent search and optimization solutions and relies on bio-inspired operators such as crossover, selection and crossover.

Advantages:

- A Genetic algorithm can solve any optimization problem that can be described with genetic encoding.
- This technique is not based on the surface of errors, so this technique can solve multi-dimensional, non-differential and even non-continuous issues.
- This can be transferred easily to existing models.

Limitations:

- This technique does not ensure constant response time for optimization. Sometimes the difference between the shortest and longest response times for optimization is significantly greater than conventional gradient methods.
- Genetic algorithm applications in real-time controls are limited due to random solutions and convergence, meaning that the entire population is improving, but this cannot be said for an individual in this population. The use of genetic algorithms for online controls in real systems is therefore unreasonable without first testing them on a simulation model.

Applications:

- Evolutionary Computation
- Genetic Programming and Evolvable Machines

II. Differential Evolution

Stock market prediction is an important research area where BPNN was seen as a familiar learning algorithm in so many applications, but some researchers used DE as a criterion for optimization. Here Lin Wang et al. used DE in the time series data forecast to take the initial weights and [16] threshold. It's the same as GA using the same operator, but the fundamental difference between them is that DE relies on mutation operator while GA relies on the operator of a crossover. Considering DE Mustafa E. Abdul Salam et al. conducted a comparison study between DE and PSO applying to [17] Feed-Forward Neural Network (FFNN) training on stock market prediction and found DE provides better accuracy than PSO. Six technical indicators were used here for the input layer FFNN.

Advantages:

- Find the true global minimum irrespective of the initial parameter values
- Quick and rapid convergence
- Using a few parameters of control.

Limitations:

- Convergence is unstable
- It may be questionable to argue that differential evolution is not good for epistatic problems as all evolutionary algorithms perform equally badly on epistatic issues.

Applications:

- Evolutionary Computation
- Genetic Programming and Evolvable Machines

III. Particle Swarm Algorithm

Since the critical management decisions in financial markets need to be accurate and fast, the two main components of the current research are algorithm design and developing and implementation of the algorithm on the faster computer platform. For algorithm design, we have used nature-inspired algorithm known as Particle swarm optimization (PSO) and for speed, we have used state-of-the-art graphics processing units (GPUs).[28]

Advantages:

- Can be simple to implement
- Have few parameters to adjust
- Able to run the parallel computation
- Can be robust
- Have higher probability and efficiency in finding the global optima
- Can converge fast
- Do not overlap and mutate
- Have short computational time
- Can be efficient for solving problems presenting difficulty to find accurate mathematical models

Limitations:

- Can be difficult to define initial design parameters
- Cannot work out the problems of scattering
- Can converge prematurely and be trapped into a local minimum especially with complex problems

Applications:

- Multimodal biomedical image registration - Biomedical image registration, or geometric alignment of two-dimensional and/or three-dimensional (3D) image data, is becoming highly important in diagnosis, treatment planning, functional studies, computer-guided therapies, and in biomedical research.[25]

- Iterated prisoner's dilemma-The dilemma of the iterated prisoner is an extension of the general form except that the same participants repeatedly play the game. The dilemma of an iterated prisoner differs from the original concept of the dilemma of a prisoner because participants can learn about their counterparty's behavioural tendencies [22].

- Power system optimization problem - Power system, where there are many places where optimization is used a lot these days. Distributed Generation (DG) are used in the distribution system. There are many things to improve in our distribution system, such as reducing losses, improving voltage, improving stability limits, etc. So, if we put DG at certain buses different results will be obtained. So here comes the role of optimization. We should place the DG at that particular bus where our losses are less and voltage is improved more so out of several sets of solutions obtained whichever will give a maximum improvement in voltage and minimum losses are said to be Optimized.[22]

IV. Firefly

The firefly algorithm is one of the latest bio-inspired algorithms that proved its performance in solving continuous and discrete problems of optimization. Management of the supply chain, the most complex part of the inventory management process. A complexity resulting from the conflict between cost minimization and service level maximization. For these reasons, the traditional lot size control methods have to address the explosion of new supply chain-related needs. FA's optimal solutions are far better than the best solutions obtained through the literature's analyzed deterministic methods[33].

Advantages:

- FA can deal with highly non linear, multi-modal optimization problems naturally and efficiently.
- FA does not use velocities, and there is no problem associated with velocity in PSO.
- The speed of convergence of FA is very high in the probability of finding the global optimized answer.
- Flexibility to integrate hybrid tools with other optimization techniques.
- To start its iteration process, it does not require a good initial solution.

Applications:

- For solving the Travelling Salesman Problem
- Digital image compression and image processing
- Feature Selection and fault detection
- Antenna design
- Structural design
- Scheduling
- Chemical phase equilibrium
- Dynamic problems

V. Artificial Algae Algorithm

The research and evaluation of the AAA algorithm were done through previously published papers in similar fields. The references for which have been written at the end of the document.

Advantages:

- Parallelly computes solutions
- Can work on multiple dimensions at once
- Works on the whole sample space at once, no solutions are left out

Limitations:

- Very inefficient for small data sets
- requires a lot of computational power

Applications:

- Global Optimization
- Binary Optimization
- Economic Load Dispatch

VI. Artificial Bee Colony

Artificial Bee Colony (ABC) is one of Dervis Karaboga's most recent algorithms in 2005, motivated by honey bees' intelligent behaviour. It is as simple as algorithms for Particle Swarm Optimization (PSO) and Differential Evolution (DE) and uses only common control parameters like colony size and a maximum number of cycles.

Advantages:

- Simplicity, Flexibility and Robustness
- Ability to explore local solutions
- ability to handle the objective cost
- ease of implementation
- Broad applicability

Limitations:

- Lack of use of secondary information
- New fitness tests on new algorithm parameters
- The high number of objective functions
- Slow in sequential processing

Applications:

- Benchmark Optimization
- Bioinformatics
- Scheduling
- Clustering and Mining
- Image Processing
- Economic Dispatch Problems

VII. Bacterial Foraging Optimization

Bacterial Foraging Optimization (BFO) algorithm[21] was first proposed by Passino in 2002. It is inspired by the foraging and chemotactic behaviours of bacteria, especially the Escherichia coli (E. coli) By smooth running and tumbling, The E. coli can move to the nutrient area and escape from poison area in the environment. The chemotactic is the most attractive behaviour of bacteria.

Advantages:

- It is used in parallel distributed processing, insensitivity to an initial value, and global optimization
- The high-performance optimizer because of its faster convergence and global search approach is provided.
- This can be easily transferred to existing models.

Limitations:

- The problem of optimal power flow brings a lag for this algorithm.
- The Convergence in this algorithm is very slow.

Applications:

- Improved Bacteria Foraging Optimization Algorithm for Solving Flexible Job-Shop Scheduling Problem
- Genetic Programming and Evolvable Machines applied BFO to RFID (radio frequency identification) network scheduling. A variant of BFO, self-adaptive bacterial

foraging optimization (SABFO) has been developed in which swim length of individual bacterium adjusts dynamically during the search to balance the exploration/exploitation trade off.
- An Enhanced Bacterial-Foraging, Optimization-Based Machine Learning Framework to Predict Somatization Disorder Severity

VIII. Artificial Fish Swarm Optimization

AFSA inspired by the collective movement of the fish and their various social behaviours. To bring about accurate forecasting result researchers have applied so many techniques. Here a Forecasting model using Radial Basis Function Neural Networks optimized by AFSA is developed by Wei Shen et al, where the proposed model is executed on the data of Shanghai Composite Indices.

Advantages:
- This algorithm has many advantages including high convergence speed, flexibility, tolerance of failures and high precision
- There are numerous advanced techniques which have a fondness with this strategy and the aftereffect of this mix will improve the execution of this strategy.

Limitations:
- Its disadvantages include high time complexity, lack of balance between global and local search, in addition to lack of benefiting from the experiences of group members for the next movements.
- Although it expects a high number of objective function evaluations in the present version of the FSOA algorithm.

Applications:
- A weak signal detection method based on an optimized matching pursuit of the artificial fish swarm. In: Computer Science and Information Engineering World Congress.
- Application of artificial fish-swarm algorithm based on a probabilistic causal-effect model for fault diagnosis in mine hoist.
- The welded beam design problem is taken from Rao (1996) and He and Wang (2007), in which a welded beam is designed for minimum cost subject to constraints on shear stress (τ), bending stress in the beam, buckling load on the bar (Pc), end deflection of the beam (δ), and side constraints. Fish swarm optimization algorithm used for the design of the engineering system.

IX. Ant Colony Optimization

Ant colony optimization (ACO) is a metaheuristic popultion based approach that can be used to find approximate solution s to difficult problems of optimization. A set of software age nts in ACO called artificial ants seeking good solutions to a given problem of optimization.

Advantages:
- Inherent parallelism
- Positive feedback means that good solutions are quickly discovered
- Efficient for Travelling Salesman Problem and similar problems.
- Can be used in dynamic applications

Limitations:
- Theoretical analysis is difficult
- Sequences of random decisions
- Probability distribution changes by iteration
- Research is experimental rather than theoretical
- Time to convergence uncertain

Applications:
- Routing in telecommunication networks
- Travelling Salesman
- Graph Coloring
- Scheduling
- Constraint Satisfaction

X. K nearest neighbours

A genetic algorithm is inspired by the Darwinian principle of" survival of the fittest" and natural evolution. John Holland presented algorithms developed on Darwin's hypothesis in 1960 and thereafter his understudy Goldberg worked on it and proposed the first Genetic algorithm in 1989. It provides excellent solutions to search and optimisation problems and relies on bio-inspired operators like crossover, selection and crossover.

Advantages:
- Robust to noisy training data(especially if we use an inverse square of weighted distance)
- Effective if the training data is large

Limitations:
- Need to determine the value of parameter K
- Distance-based learning is not clear which distance to use and which attribute to use to achieve the best outcome.
- The cost of computing is quite high because we need to calculate each query's distance.

Applications:
- Text Mining
- Agriculture
- Currency Exchange Rate
- Trading Futures
- Loan management
- Medicine

REFERENCES

[1] Goldberg, David (2002). The Design of Innovation: Lessons from and for Competent Genetic Algorithms. Norwell, MA: Kluwer Academic Publishers. ISBN 978-1402070983.

[2] Goldberg, David (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley Professional. ISBN 978-0201157673.

[3] Kyoung-jae Kim,Ingoo Han(2000):Genetic algorithm approach to feature discreetization in artificial neural networks for prediction of stock price index. Expert systems with applications, ELSEVIER, 125-132

[4] Birgul Egeli, Meltem Ozturan, Bertan Badur:Stock Market Prediction Using Artificial Neural Networks.

[5] Kyoung-jaeKim(2006):Artificial neural networks with evolutionary instance selection for financial forecasting, Expert Systems with Applications, ELSEVIER ,519–526

[6] Sam Mahfoud,Ganesh Mani(1996) :Financial forecasting using genetic algorithms,Applied Artificial Intelligence, Taylor and Francis.

[7] Koza, John (1992). Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA: MIT Press. ISBN 978-0262111706.

[8] Yadav, P. K., & Prajapati, N. L. (2012). An overview of genetic algorithm and modeling. *International Journal of Scientific and Research Publications*, *2*(9), 1-4.

[9] Mitchell, Melanie (1996). An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press. ISBN 9780585030944.

[10] Mardle, S. J., Pascoe, S., & Tamiz, M. (2000). An investigation of genetic algorithms for the optimization of multi-objective fisheries bioeconomic models. *International Transactions in Operational Research*, *7*(1), 33-49.

[11] Rocca, P.; Oliveri, G.; Massa, A. (2011). "Differential Evolution as Applied to Electromagnetics". IEEE Antennas and Propagation Magazine. 53 (1): 38–49. doi:10.1109/MAP.2011.5773566.

[12] Anyong Qing, "Fundamentals of Differential Evolution," in *Differential Evolution: Fundamentals and Applications in Electrical Engineering* , , IEEE, 2009, pp.6-7

[13] Z. Gao, Z. Pan and J. Gao, "A New Highly Efficient Differential Evolution Scheme and Its Application to Waveform Inversion," in *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1702-1706, Oct. 2014.

[14] Z. S. Mohamad, A. Darvish and S. Rahnamayan, "Eye illusion enhancement using interactive Differential Evolution," *2011 IEEE Symposium on Differential Evolution (SDE)*, Paris, 2011, pp. 1-7.

[15] Z. Gao, Z. Pan and J. Gao, "A New Highly Efficient Differential Evolution Scheme and Its Application to Waveform Inversion," in *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1702-1706, Oct. 2014.

[16] L. Yu, L. Hu and L. Tang, "Stock Selection with a Novel Sigmoid-Based Mixed Discrete-Continuous Differential Evolution Algorithm," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1891-1904

[17] Mustafa E. Abdual-Salam, Hatem M. Abdul-Kader and Waiel F. Abdel-Wahed: Comparative Study between Differential Evolution and Particle Swarm Optimization Algorithms in Training of Feed-Forward Neural Network for Stock Price prediction.

[18] Enke David,Grauer Manfred,Mehdiyev Nijat(2011):Stock Market Prediction with Multiple Regression, Fuzzy Type-2 Clustering and Neural Networks, Procedia Computer Science ,ELSEVIER, pp:201–206

[19] Hachicha Nijam , Jarboui Baseem , Siarry Patric(2011) :A fuzzy logic control using a differential evolution algorithm aimed at modelling the financial market dynamics,Information Sciences 181pp: 79–91.

[20] Rahnamayan, S., Tizhoosh, H. R., & Salama, M. M. (2008). Opposition-based differential evolution. *IEEE Transactions on Evolutionary computation*, *12*(1), 64-79.

[21] Russell C. Eberhart and Yuhui Shi (2001) Particle Swarm Optimization: Developments, Applications and Resources 0-7803-6657-3/01/$10.00 2001 IEEE

[22] Xiaohui Cui, Thomas E. Potok, Paul Palathingal (2005). Document Clustering using Particle Swarm Optimization 0-7803-8916-6/05/$20.00 2005 IEEE

[23] Jovita Nenortaite, Rimvydas Simutis Adapting Particle Swarm Optimization to Stock Markets 0-7695-2286-06/05 $20.00 © 2005 IEEE

[24] M. R. AlRashidi A Survey of Particle Swarm Optimization 1089-778X/$26.00 © 2009 IEEE

[25] Kalyanmoy Deb and Nikhil Padhye Improving a Particle Swarm Optimization Algorithm Using an Evolutionary Algorithm Framework KanGAL Report Number 2010003 feb 21, 2010

[26] Qinghai Bai Analysis of Particle Swarm Optimization Algorithm vol.3, No. 1 feb 2010 CCSE

[27] Yuntao Dai and Liqiang Liu, Ying Li An Intelligent Parameter Selection Method for Particle Swarm Optimization Algorithm 978-0-7695-4335-2/11 $26.00 © 2011 IEEE

[28] Steffen Freitag, Rafi L. Muhanna and Wolfgang Graf. A Particle Swarm Optimization Approach for Training Artificial Neural Networks with Uncertain Data SBN 978-80-214-4507-9. Published by Ing. Vladislav Pokorný −LITERA

[29] TAKUYA AOKI and CLAUS ARANHA, HITOSHI KANOH PSO Algorithm with Transition Probability Based on Hamming Distance for Graph Colouring Problem 978-1-4799-8697-2/15 $31.00 © 2015 IEEE

[30] Dheeraj Pal, Pratima Verma, Divya Gautam, Priyanka Indait Improved Optimization Technique using Hybrid ACO-PSO 978-1-5090-3257-0/16/$31.00 ©2016 IEEE

[31] DONALD L. SNYDER, MEMLIER, IEEE, AND PHILIP M. FISHMAN, MEMBER, IEEE How to Track a Swarm of fireflies by observing their flashes 1975

[32] Xin-She Yang Firefly Algorithm, Stochastic Test Functions and Design Optimisation arxiv:1003.1409v1 2010

[33] Siddharth Agarwal,M-tech(CS)Amrit Pal Singh,(Asst Prof) Nitin Anand, M-Tech (IS) Evaluation Performance Study of Firefly Algorithm – 2013

[34] Changnian Liu, Yafei Tian, Qiang Zhang, Jie Yuan, Binbin Xue Adaptive Firefly Optimization Algorithm 978-0-7695-5079-4/13 $26.00 © 2013 IEEE

[35] Xin-She Yan Firefly Algorithm: Recent Advances and Applications 2013

[36] Mukesh Gupta S.N.Tazi, Akansha Jain, Deepika Edge Detection Using Modified Firefly Algorithm 978-1-4799-6929-6/14 $31.00 © 2014 IEEE

[37] Sankalap Arora and Satvir Singh Performance Research on Firefly Optimization Algorithm with Mutation 2014

[38] Shruti Goel and V. K. Panchal Performance Evaluation of a New Modified Firefly Algorithm 978-1-4799-6896-1/14/$31.00 ©2014 IEEE

[39] Xiangbo Qi, Sihan Zhu and Hao Zhang A Hybrid Firefly Algorithm 978-1-4673-8979-2/17/$31.00 ©2017 IEEE

[40] Tawfeeq Al-Kanhel Intelligent Firefly Optimization System 978-1-5090-5443-5/17/$31.00 ©2017 IEEE

[41] Mehdi Neshat, Ghodrat Sepidnam, Mehdi Sargolzaei, Adel Najaran Toosi/ Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, ombinatorial and indicative applications/Artif Intell Rev (2014) 42:965–997 DOI 10.1007/s10462-012-9342-2

[42] Fran Sérgio Lobato; Valder Steffen Jr./ Fish swarm optimization algorithm applied to engineering system design/Lat. Am. j. solids struct. vol.11 no.1 Rio de Janeiro Jan. 2014

[43] Neshat, M., Adeli, A., Sepidnam, G., Sargolzaei, M., & Toosi, A. N. (2012)/ A review of artificial fish swarm optimization methods and applications/ International Journal on Smart Sensing & Intelligent Systems, 5(1).

[44] Ai-ling, Q., Hong-wei, M., & Tao, L. (2009, March)/ A weak signal detection method based on artificial fish swarm optimized matching pursuit/ In 2009 WRI World Congress on Computer Science and Information Engineering (Vol. 6, pp. 185-189)/ IEEE.

[45] Ban, X., Yang, Y., Ning, S., Lv, X., & Qin, J. (2009, August)/ A self-adaptive control algorithm of the artificial fish formation/ In 2009 IEEE International Conference on Fuzzy Systems (pp. 1903-1908)/ IEEE.

[46] Chen, Z., & Tian, X. (2010, March)/ Notice of Retraction Artificial Fish-Swarm Algorithm with Chaos and Its Application/ In 2010 Second International Workshop on Education Technology and Computer Science (Vol. 1, pp. 226-229)/ IEEE.

[47] Duan, Q., Mao, M., Duan, P., & Hu, B. (2016)/ An improved artificial fish swarm algorithm optimized by particle swarm optimization algorithm with extended memory/ Kybernetes, 45(2), 210-222.

[48] Lobato, F. S., & Steffen Jr, V. (2014)/ Fish swarm optimization algorithm applied to engineering system design/ Latin American Journal of Solids and Structures, 11(1), 143-156.

[49] Bansal, V., Sakizlis, V., Ross, R., Perkins, J. D., & Pistikopoulos, E. N. (2003)/ New algorithms for mixed-integer dynamic optimization/ Computers & Chemical Engineering, 27(5), 647-668.

[50] Neshat, M., Sepidnam, G., Sargolzaei, M., & Toosi, A. N. (2014)/ Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, combinatorial and indicative applications/ Artificial intelligence review, 42(4), 965-997.

[51] Sharma, V., Pattnaik, S. S., & Garg, T. (2012)/ A review of bacterial foraging optimization and its applications/ International Journal of Computer Applications (IJCA).

[52] Li, J., Dang, J., Bu, F., & Wang, J. (2014)/ Analysis and improvement of the bacterial foraging optimization algorithm/ Journal of Computing Science and Engineering, 8(1), 1-10.

[53] Wu, X. L., Zhang, Z. Q., Du, Y. H., & YAN, J. (2015)/ Improved bacteria foraging optimization algorithm for flexible job shop scheduling problem/ Computer Integrated Manufacturing Systems, 21(5), 1262-1270.

[54] Mohammadi, M. (2015)/ Bacterial foraging optimization and adaptive version for economically optimum sitting, sizing and harmonic tuning orders setting of LC harmonic passive power filters in radial distribution systems with linear and nonlinear loads. Applied Soft Computing, 29, 345-356.

[55] Al-Hadi, I. A. A., Hashim, S. Z. M., & Shamsuddin, S. M. H. (2011, December)/ Bacterial Foraging Optimization Algorithm for neural network learning enhancement/ In 2011 11th International Conference on Hybrid Intelligent Systems (HIS)(pp. 200-205)/ IEEE.

[56] Yang, D., Li, X., & Jiang, L. (2012). Improved algorithm of bacterium foraging and its application. Computer Engineering and Applications, 48(13), 31-34.

[57] Jing-jinga, C. U. I., Yan-minga, S. U. N., & Lan-xiub, C. H. E. (2011)/ Improved bacteria foraging optimization algorithm for Job-Shop scheduling problems [J]/ Application Research of Computers, 9.

[58] Yang, C., Ji, J., Liu, J., Liu, J., & Yin, B. (2016)/ Structural learning of Bayesian networks by bacterial foraging optimization/ International Journal of Approximate Reasoning, 69, 147-167.

[59] Majhi, R., Panda, G., Majhi, B., & Sahoo, G. (2009)/ Efficient prediction of stock market indices using adaptive bacterial foraging optimization (ABFO) and BFO based techniques/ Expert Systems with Applications, 36(6), 10097-10104.

[60] Mishra, S. (2005)/ Hybrid least-square adaptive bacterial foraging strategy for harmonic estimation/ IEE Proceedings-Generation, Transmission and Distribution, 152(3), 379-389.

[61] Audibert, J.Y. & Tsybakov, A.B. (2007) "Fastlearning rates for plug-in classifiers under themargin condition", Ann. Statist, 35: 608 – 633.

[62] Bailey, T. & Jain, A. (1978) "A note ondistance-weighted k-Nearest Neighbor rules",IEEE Trans. Systems, Man, Cybernetics, 8:311-313.

[63] BaoliBaoli, L., Shiwen, Y. & Qin, L. (2003) "AnImproved k-Nearest Neighbor Algorithm for Text Categorization, ArXiv Computer Sciencee-prints.

[64] Bauer, M.E., Burk, T.E., Ek, A.R., Coppin,P.R. Lime, S.D., Walsh, T.A., Walters, D.K.,Befort, W. & Heinzen, D.F. (1994) "Satellite Inventory of Minnesota’s Forest Resources",

[65] [5] Bax, E. (2000) "Validation of nearest neighbor classifiers", IEEE Trans. Inform. Theory, 46:2746– 2752.

[66] Benetis, R., Jensen, C., Karciauskas, G. &Saltenis, S. (2006) "Nearest and Reverse Nearest Neighbor Queries for MovingObjects", The International Journal on VeryLarge Data Bases, 15(3): 229 – 250.

[67] Bermejo, T. & Cabestany, J. (2000) "Adaptivesoft k-Nearest Neighbor classifiers", PatternRecognition, 33: 1999-2005.

[68] Chitra, A. & Uma, S. (2010) "An EnsembleModel of Multiple Classifiers for Time SeriesPrediction", International Journal of Computer Theory and Engineering, 2(3): 1793-8201.

[69] Cover, T.M. (1968) "Rates of convergence for nearest neighbor procedures", In Proceedingsof the Hawaii International Conference onSystem Sciences, Univ. Hawaii Press,Honolulu, 413 – 415.

[70] Cover, T.M. & Hart, P.E. (1967) "Nearestneighbor pattern classification", IEEE Trans.Inf. Theory, 13: 21 – 27.

[71] Afshar MH (2005) A new transition rule for ant colony optimization algorithms: application to pipe network optimization problems. Engineering Optimization 37(5):525–540

[72] Alaya I, Solnon C, Gh´edira K (2004) Ant algorithm for the multi-dimensional knapsack problem. In: International Conference on Bioinspired Optimization Methods and their Applications (BIOMA 2004, pp 63–72)

[73] Baykasoglu A, Dereli T, Sabuncu I (2006) An ant colony algorithm for solving budget constrained and unconstrained dynamic facility layout problems. Omega 34:385–396

[74] Bin Y, Zhong-Zhen Y, Baozhen Y (2009) An improved ant colony optimization for vehicle routing problem. European Journal of Operational Research 196:171–176

[75] Blum C, Blesa MJ (2005) New metaheuristic approaches for the edge-weighted k-cardinality tree problem. Computers & Operations Research 32:1Artificia

[76] Babeau E, Dorigo M, Theraulaz G (1999) Swarm Intelligence - from Natural to Artificial Systems. Oxford University Press, New York

[77] Bouhafs L, Hajjam A, Koukam A (2006) A combination of simulated annealing and ant colony system for the capacitated location-routing problem. In: KES (1), pp 409–416

[78] Bullnheimer B, Hartl RF, Strauß C (1997) A new rank based version of the ant system - a computational study. Central European Journal for Operations Research and Economics 7:25–38

[79] Cordon O, Herrera F, St¨utzle T (2002) A review on the ant colony optimization metaheuristic: Basis, models and new trends. Mathware & Soft Computing 9:141–175

[80] Crawford B, Castro C (2006) Integrating lookahead and post processing procedures with aco for solving set partitioning and covering problems. In: ICAISC, pp 1082–1090