Nature-Inspired Algorithms: State-of-Art, Problems and Prospects

Parul Agarwal
Department of Computer Science & Engineering
Jaypee Institute of Information Technology
Noida

Shikha Mehta
Department of Computer Science & Engineering
Jaypee Institute of Information Technology
Noida

ABSTRACT

Nature-inspired algorithms have gained immense popularity in recent years to tackle hard real world (NP hard and NP complete) problems and solve complex optimization functions whose actual solution doesn't exist. The paper presents a comprehensive review of 12 nature inspired algorithms. This study provides the researchers with a single platform to analyze the conventional and contemporary nature inspired algorithms in terms of required input parameters, their key evolutionary strategies and application areas. A list of automated toolboxes available for directly evaluating these nature inspired algorithms over numerical optimization problems indicates the need for unified toolbox for all nature inspired algorithms. It also elucidates the users with the minimum and maximum dimensions over which these algorithms have already been evaluated on benchmark test functions. Hence this study would aid the research community to know what all algorithms could be examined for large scale global optimization to overcome the problem of 'curse of dimensionality'.

Keywords

Nature inspired algorithm, key evolutionary strategies, automated toolboxes, benchmark test function.

1. INTRODUCTION

Nature inspired computing is a technique that is inspired by processes, observed from nature. These computing techniques led to the development of algorithms called Nature Inspired Algorithms (NIA). These algorithms are subject of computational intelligence. The purpose of developing such algorithms is to optimize engineering problems. As the world is moving towards industrialization, engineering problems are becoming more and more complex and difficult to optimize. This is because of increasing dimensions, variables, time complexity, space complexity etc. To cope up with such situation, nature inspired algorithms are designed to optimize numerical benchmark functions [1], multi objective functions and solve NP-hard problems for large number of variables, dimensions, etc. NIA are mainly categorized into evolutionary algorithms and swarm intelligence based algorithms. Evolutionary algorithms are based on the evolutionary behavior of natural systems. These algorithms are inspired by Charles Darwin theory [30]: Survival of the fittest. A population of only those species reproduces who fits in the natural habitat – natural selection. These algorithms employ recombination and mutation operators to optimize the complex problems e.g. genetic algorithm and differential evolution etc. Swarm intelligence (SI) based algorithms, also called as swarm optimization techniques; optimize the certain

problem by mimicking the collective behavior of natural swarms' .e.g. particle swarm optimization, ant colony algorithm, artificial bee colony algorithm, bacterial foraging algorithm etc. Besides these two categories, NIA have various other classifications depending on the source of inspiration [2] e.g. intelligent water drop algorithm is inspired from flowing of natural water drops in the river.

The paper is organized as follows: Section 2 highlights the related work. Section 3 presents the relative comparison of 12 nature inspired algorithms on the basis of their control parameters, evolutionary strategies and the application areas. The minimum and maximum dimensions over which nature inspired algorithms have been evaluated are discussed in Section 4, along with list of automated tools available for these algorithms. Conclusion and future work are discussed in section 5.

2. RELATED WORK

Due to the immense popularity of nature inspired algorithms in last few years, a competition on evolutionary computation is organized every year by IEEE [41] to welcome numerous algorithms developed for large scale optimization. As the size of problem increases, its complexity gets enhanced and hence algorithms are no longer able to provide optimal solution. This problem is termed as "curse of dimensionality" [31]. On exponential growth of solution space, an efficient and reliable algorithm is required to diversify its attention in almost all promising areas within a given time period and financial budget. Therefore, nature inspired algorithms are being either developed or modified to overcome this problem. To assess the efficacy of these algorithms, they are evaluated over continuous unimodal or multimodal benchmark test problems. Now a day's NIA are being tested for up to 2000 dimensions for large scale optimizations.

Various studies have been carried out to explore the research performed in the field of nature inspired algorithms. Fister [2] presented a study focused on classification of various nature inspired algorithms. All these algorithms are clustered into number of groups depending upon their source of inspiration. However, the paper did not discuss the applications and performance of these algorithms. Elbeltagi [10] performed the comparative analysis of 5 evolutionary algorithms for continuous and discrete optimization. The paper evaluated algorithms on small dimensions but it did not discuss the effects of high dimensions on these algorithms. Moreover only 5 algorithms are considered for comparative study which did not include recently developed algorithms. Yang [11] discussed the search strategy of various meta-heuristic algorithms, importance of large scale problems and pointed

out new challenges in nature inspired algorithms. Sureja [39] evaluated the performance of some recently proposed nature inspired algorithms such as cuckoo algorithm, firefly algorithm and bat algorithm for small dimensions on various benchmark functions. However, the study lacks comparison with established algorithms such as particle swarm optimization, ant colony algorithm etc. as well as evaluation over higher dimensions. Binitha [40] explored the significance of various bio-inspired algorithms on basis of their control parameters and application. But this paper did not specify the actual dimension range on which different nature inspired algorithms are evaluated. Thus, all these papers provide limited information to the researchers and practitioners about the problems and prospects of nature inspired algorithms. In this respect, this paper presents an intensive review of various nature inspired algorithms on the basis of their control parameters, key updating formula and applied applications. An important contribution of the paper is that it provides an insight on the minimum and maximum dimension search space over which these algorithms have already been evaluated. It also provides a list various automated tools available for direct implementation of algorithms.

3. REVIEW OF NATURE INSPIRED ALGORITHMS

The main objective of nature inspired algorithms is to find global optimal solution for a given problem. Two key factors common in all nature inspired algorithms are intensification and diversification commonly termed as Exploration and Exploitation. Exploration leads to a random search of a new solution space for finding global optima and exploitation finds local optima in explored solution space. Intense exploration does not give optimal solution while deep exploitation traps an algorithm in local optima. A balance between these two factors is very essential for any nature inspired algorithm. Table 1 depicts the input parameters, evolutionary mechanism and applications of 12 nature inspired algorithms. The study presents applications of these algorithms in the computer science field only. An overview of nature inspired algorithms mentioned in Table 1 is as follows:

3.1 Genetic Algorithm (GA)

Genetic algorithm was given by John Holland during 1960s and 1970s [20] [21]. It is a biologically inspired computational technique motivated from Darwin's theory of evolution. It mimics the process of natural selection. Performance of GA is based on four factors: population size, mutation rate, crossover rate and number of generations. To determine the best individual among population, genes of corresponding individuals are evaluated against the objective function. The evolutionary mechanisms through which a new offspring (individual) is produced are crossover and mutation. In crossover mechanism, an offspring is produced by combining the genes of selected individuals among population while mutation causes some random changes in genes of an individual thereby producing new genetic individual. The evolutionary mechanism is carried out till convergence criteria are satisfied.

3.2 Ant Colony Optimization (ACO)

In 1992, Marco Dorigo [24] proposed a new algorithm in his PhD thesis, Ant colony optimization. Ants are small creatures that can intelligently find the shortest path between their nest and food source. Each ant represents a potential solution to an objective function. Ants establish their communication through trails of pheromone. When an ant comes out of their

nest to search for food source, they move randomly in any direction, leaving pheromone in their path. On reaching the food source, ants return back with food, leaving pheromone again on same path. Hence, the path with the highest amount of pheromone represents the shortest route from ant's nest to the food source. Pheromone concentration in each path represents the quality of solution (goodness of fitness value). The process is continued until stopping criteria is met.

3.3 Particle Swarm Optimization (PSO)

PSO was developed by J. Kennedy and R. Ederhart in 1995 [23]. PSO mimics the flocking behavior of birds. The birds fly in a solution space and their flocking behavior determines the optimum solution. They follow some path to reach their food destination. The shortest path followed by a bird is considered to be local or particle best solution. Particles tend to move towards its local best position (solution) (lbest) found by them so far. They also keep the track of global best (gbest) solution, the best (shortest) path found by any particle at particular instance. Each particle is associated with a velocity, through which it gets accelerated towards local and global best path, the position in 'n' dimension space and the current position of particle with respect to gbest and lbest. Birds communicate with each other to find the most optimum (best) path to reach its food sources. Hence, they learn from the experience of their local best solutions and global best solutions. The algorithm continues till global optimum solution is achieved.

3.4 Memetic Algorithm (MA)

Memetic algorithm is also inspired by Dawkin's theory of evolution [22]. Instead of genes, a set memes are considered to form chromosomes. In genetic algorithm, crossover and mutation operations begins immediately after the selection of individuals while in MA, an individual takes time to learn experience and then crossover and mutation operations are applied. When crossover between two individuals takes place, a new offspring is produced. This offspring undergoes local search to produce local optimum solution. Individual with best fitness value (solution) is selected and rests are abandoned. Thus in MA, evolutionary mechanism includes local search along with crossover and mutation.

3.5 Bacterial foraging optimization Algorithm (BFOA)

BFOA was developed by Kevin M. Passino in 2002 [15]. There are three main steps followed by bacteria to achieve global optimum solution: chemotactic step, reproduction step, elimination and dispersal step. In chemotactic step of BFOA, bacteria swims in direction of high nutrient surface while they tumble when noxious surface is encountered. Objective of BFOA is to minimize the cost of bacteria's movement in high nutrient surface. At the end of this step, all bacteria are arranged in descending order of their fitness value. In reproduction step, first half of the bacteria with high cost die as they did not get sufficient nutrient to survive, while other half (bacteria getting sufficient amount to nutrient) are split into two parts thereby maintaining constant population size. In elimination and dispersal step, bacteria get dispersed into complete surface thus controlling them to get trapped in local optima. The newly produced bacteria occupy position of eliminated bacteria. The bacteria with best fitness value, that is, minimum cost finally represent the solution to an objective function. The process continues till desired number of generations gets exhausted.

3.6 Shuffled Frog Leaping Algorithm (SFLA)

SFLA was given by Muzaffar Eusuff and Kevin Lansey in 2003 [25]. This algorithm combines the best properties of two algorithms: MA and PSO. SFLA is inspired by leaping and shuffling behavior of frogs to exchange information among them in order to search for food. Each frog represents a potential solution for the specified problem. After computing the fitness value of each frog in SFLA, all frogs are arranged in descending order of their fitness value. Thereafter all frogs are divided into number of subsets called memeplexes in some special manner; each memeplex contains equal number of frogs. After this division local best solution is computed within each memeplex. In SFLA, local evolution is carried out within each memeplex and after defined number of memetic evolutions, all frogs are shuffled together for global evolution. The process continues unless desirable solution is achieved or stopping criteria is met.

Table 1. List of Nature Inspired Algorithms with input parameters, evolutionary strategies and applications

S.No	Algorithm	Input parameters	Evolutionary Mechanism	Applied Application Area
1.	Genetic Algorithm (GA)	Crossover rate and Mutation rate.	Selection, Recombination and Mutation	Machine Learning [21], Code Breaking, Computer automated design , Computer architecture, Bayesian inference [27], forensic science, Data Center/Server Farm, File allocation for a distributed system, game theory, robot behavior etc.
2.	Ant Colony Optimization (ACO)	$\tau(e)$ is pheromone update of a path e (edge), ρ is evaporation rate $0 \leq \rho \leq 1$	$\tau(e)=$ $\begin{cases} (1-\rho)\tau(e) \\ (1-\rho)\tau(e) + \text{new pherom} \end{cases}$	The generalized assignment problem (GAP) [48], and the set covering problem (SCP) [49], Classification [50], AntNet for network routing applications [51], Multiple Knapsack Problem [52]
3.	Particle Swarm Optimization (PSO)	Learning factors C1 and C2, inertia weight w, maximum change of a particle velocity Vmax, Pg is position of best particle (g), Xi is the current position of particle (i), Pi is best position of particle in previous cycle.	New Velocity Vi = w * current Vi + C1*rand() * (Pi – Xi) + C2 * Rand() * (Pg-Xi) New position Xi = current position Xi + New Vi; Vmax >=Vi >=Vmax	conjunction with a back propagation algorithm, to train a neural network system design [61], multi-objective optimization, classification, pattern recognition and image processing [69] , image clustering [62], robotic applications, decision making, simulation and identification, time-frequency analysis, image segmentation etc.
4.	Memetic Algorithm (MA)	Crossover rate, mutation rate, Local search step constant value	Selection, Recombination and Mutation, Local Search mechanism	Travelling salesman problem [22], training of artificial neural networks, pattern recognition, robotic motion planning, multidimensional knapsack problem, clustering [63] of gene expression profiles, feature/gene selection etc.
5.	Bacterial Foraging Algorithm (BFOA)	Step size Z(i), position vector of ith of bacterium O(i) at jth chemotactic step and kth reproductive step, Δ indicates a vector in the random direction whose elements lie in [-1, 1].	$O(i,j+1,k) = O(i,j,k) +$ $Z(i)\frac{\Delta(i)}{\sqrt{\Delta t(i)\Delta(i)}}$	Job Scheduling [42], machine learning [44], to train a Wavelet-based Neural Network (WNN), face recognition [43], Image Edge detection [45], Image Segmentation [46], Color Image Quantization [47]
6.	Shuffled frog Leaping Algorithm (SFLA)	Number of memeplexes, number of memeplex iterations and maximum change in position Dmax	Change in frog position (Di) = rand () *(Xb- Xw) New Position Xw = Xw + Di (Dmax >= Di >=-Dmax)	multi-user detection in DS-CDMA Communication System [72], multivariable PID controllers and web document classification, image watermarking [73], clustering [74]
7.	Artificial Bee Colony Algorithm (ABC)	Φ_{ij}, a uniformly distributed real random number in the range [-1, 1]. i,j is food source having different values and j is dimension	New position $POS_{ij} = X_{ij} + \Phi_{ij}(X_{ij} - X_{kj})$	train neural networks [32], medical pattern classification [33] and clustering problems [34], solving TSP [35], leaf-constrained minimum spanning tree [36], network reconfiguration problem in a radial distribution system [37]

8.	Firefly Algorithm (FFA)	β is variation of attractiveness, r is distance between two files x_i and x_i. α_t is step size, γ is the light absorption coefficient, $\in$ is a random variable	$x_i^{t+1} = x_i^t + \beta exp[- \gamma\, r_{ij}^2]$ $(x_j^t - x_i^t) + \alpha_t \in$	Digital Image Compression [56] and Image Processing, Feature selection and fault detection [57], trail neural network [58], Semantic Web Composition, Classification [59] and Clustering [60], Rigid Image Registration Problems, Parameter Optimization of SVM
9.	Biogeography Based Optimization (BBO)	P_s is probability of exactly S species, μ_s and λ_s are emigration and immigration rate respectively. Δt is change in time.	$P_s(t + \Delta t) = P_s(t)(1- \lambda_s \Delta t - \mu_s \Delta t) + P_{s-1}\, \lambda_{s-1}\Delta t + P_{s+1}$ $_{s+1}\Delta t$ And mutation factors	Color image segmentation [53], Knapsack problem [54],satellite image dissipation [55]
10.	Cuckoo Search Algorithm (CSA)	$X_i(t+1)$ is new solution (nest) of ith cuckoo, $\oplus$ entry wise multiplication of α step size and levy flight	$X_i(t+1) = X_i(t) + \alpha \oplus L\bar{e}vy(\lambda)$	Spring design optimization [19], Welded Beam Design [19], software testing and data generation, wireless sensor network [65], Knapsack problems [66], train neural network [67]
11.	Bat Algorithm (BA)	F_i is frequency of ith bat, V_i is velocity and X_i is position vector of bat, β ia random vector between (0,1)	$F_i = F_{min} + (F_{max} - F_{min})\beta$ $V_i^t = V_i^{t-1} + (X_i^t - X^*)\, F_i$ $iX_i^t = X_i^{t-1} + V_i^t$	Continuous optimization, classification, clustering [68] and data mining, inverse problem and parameter estimation, combinatorial optimization and scheduling, image processing [70], fuzzy logic and other applications [71]
12.	Flower Pollination Algorithm (FPA)	x_i^t is the pollen i at t^{th} iteration, x_j^t and x_k^t are pollens of different flowers, $\in$ is random number over [-1,1]	$x_i^{t+1} = x_i^t + \in (x_j^t - x_k^t)$	Pattern recognition, Design of disc brake

3.7 Artificial Bee Colony Algorithm (ABC)

ABC algorithm was developed by Dervis Karaboga in 2005. The algorithm imitates the foraging behavior of honey bees [13] [14]. The term artificial bee is used because the behavior of actual bee is quite different from the behavior of bees assumed in ABC algorithm. Scout bee randomly searches food source position. These food positions are exploited by employed bees and they communicate nectar amount found in particular food source to onlooker bees. Each food source is a potential solution of an objective function. Onlooker bees memorize the best food source position (one having highest nectar amount) and send employee bees to find better food position in neighborhood of best food source found so far. Hence optimized solution is the food source having highest nectar amount. The algorithm terminates when desired solution is achieved.

3.8 Firefly Algorithm (FFA)

Firefly algorithm was first introduced by Xin-She Yang in 2007[17]. The algorithm was motivated by mimicking the flashing behavior of fireflies for the purpose food acquisition [16]. Each firefly is attracted towards other firefly. This attraction is represented by their brightness, which increases or decreases depending on distances between the flies. For all fireflies, the light intensity (brightness) of each firefly is compared with other firefly. Low light intensity flies move towards high light intensity, thereby decreasing the distance and updating its own brightness. The firefly with high brightness and least distance is the best solution of an objective function.

3.9 Biogeography Based Optimization (BBO)

BBO is a new bio-inspired metaheuristic algorithm proposed by Dan Simon in 2008 [26]. It is based on migrating behavior of species in habitat. Each habitat represents a probable solution of a problem. Habitat suitability index (HSI) is an attribute of the habitat which gives the desirability of living in that habitat. Habitat immigration (species arriving in habitat) and emigration (species leaving habitat) rates are decided by HSI value. High HSI habitat (said to be good solution) has suitable environment for feeding and reproduction, hence contain large number of species than low HSI habitat. Emigration rate is higher than immigration rate in high HSI habitat and vice versa in low HSI habitat. Migrating species passes features of high HSI habitat to low HSI habitat. Each habitat suitability is characterized by SIV i.e. suitability index variables. Mutation (migration of some unexpected species into the habitat) causes disturbance in the equilibrium state (emigration rate μ and immigration rate λ are equal). This causes the change in SIV of any habitat. The process continues unless convergence criteria are satisfied.

3.10 Cuckoo Search Algorithm (CSA)

Cuckoo search algorithm is a metaheuristic algorithm developed by Xin-She Yang and Suash Deb in 2009 [19]. Cuckoo search algorithm (CSA) is inspired by breeding behavior of cuckoo bird. They select their home nest by randomly taking over the nest of some other birds for reproduction. They lay their eggs in selected nest of host bird and drop the host bird's egg. The host bird either drop cuckoo

bird's egg or abandon the whole nest. Some female cuckoo can imitate their eggs like host bird's egg and lay their eggs just before the laying of host bird's egg. This increases the probability of their chick survival. Each egg in nest represents one solution and cuckoo bird's egg represents a new solution. Fitness for each solution is computed and nest with high quality of eggs (best fitness value) represents the best solution [38]. The process is continued unless global optimum solution is achieved.

3.11 Bat Algorithm

Bat algorithm was introduced by Yang in 2010 [18]. It simulates the echolocation behavior of microbats as microbats can generate high echolocation. The Bat produces a very high sound to detect its prey which echoes back with some frequency. Echolocation is a process of detecting an object by reflected sound. It is used to know how far the prey is from background object. By observing the bounced frequency of sound, bats are able to distinguish between the prey and obstacle and can sense the distance between them in their nearby surroundings. They fly randomly with some velocity, frequency and sound (loudness) to search for food. Solution of objective function is to find prey at minimum distance. The frequency and zooming parameters maintain the balance between exploration and exploitation processes. The algorithm continued till convergence criteria are satisfied.

3.12 Flower Pollination Algorithm (FPA)

Flower pollination algorithm is a latest bio-inspired algorithm proposed by Xin-She Yang in 2012 [29]. It is inspired by fertilization (pollination) process of flowers. In FPA, abiotic and self pollination are considered for local pollination while biotic and cross pollination is considered for the global pollination between the flower plants. The algorithm maintains a balance between local and global pollination. Yang [28] assumed that each plant can have only one flower and each flower can have only one pollen grain for the purpose of optimizing the benchmark functions. The process of pollination is done by pollinators such as flies, insects or wind. Thus, each flower (or pollen) can be considered as a potential solution of an objective function. The objective function finds the best flower, which is capable of doing maximum pollination. The process continues unless stopping criteria is met.

4. REVIEW OF EVALUATED DIMENSIONS AND AUTOMATED TOOLS OF NATURE INSPIRED ALGORITHMS

Various studies have been performed to estimate the efficiency of nature inspired algorithms (discussed in section 3) over benchmark test problems in order to solve the "curse of dimensionality" problem. Table 2 depicts the minimum and maximum dimensions over which these algorithms have already been tested. It can be observed from the table that most of the basic NIA algorithms such as GA, PSO, MA, SFLA have been evaluated upto 1000 dimensions and other algorithms such as BBO, FFA, CCA and FPA have been assessed for around 100 dimensions only. Algorithms like ABC, BFOA and ACO are evaluated over few dimensions varying from 10 to 30. On the contrary, "curse of dimensionality" problem considers the search space with minimum 100 dimensions and maximum dimensions vary in thousands. Thus, the efficiency of these algorithms on high dimensions (approximately 2000) is a big challenge [31]. Therefore, there is scope for evaluating the performance of these algorithms over higher dimensions.

Besides, plenty of time and efforts with good programming expertise are required for implementing these nature-inspired algorithms on benchmark functions. It is not feasible for any researcher to devote that much amount of time and efforts on existing algorithms. To overcome this problem, several automated tools have been developed to directly analyze these algorithms by providing values of various input parameters. Table 2 portrays the various existing automated tools available to examine and evaluate the various nature-inspired algorithms.

Table 2. Review of Evaluated Dimensions and Automated Tools of Nature Inspired Algorithms

Nature Inspired Algorithm	Max Dimensions	Min Dimension	Number of benchmark functions	Automated Tool for Implementation
GA [76]	1000	2	14	GEATbx [4],SEVO [3]
ACO mixed variable [64]	10	2	6	ACO_TOOLBOX [7]
PSO [76]	1000	100	6	MOEA [6], PSO toolbox [5], SEVO [3]
MA [76]	1000	100	6	SEVO [3]
BFOA [75]	10	4	4	None
SFLA [76]	1000	100	6	SEVO [3]
ABC[13]	30	10	5	ABC2 and ABCScatter [8]
FFA [16]	256	16	10	None
BBO [26]	100	2	14	Nature Inspired Matlab based Toolbox [9]
CCA [19]	128	16	7	None
BA [18]	NA	NA	NA	None
FPA [28]	256	16	10	None

SEVO [3] is an automated nature-inspired tool that evaluates algorithm on various unimodal and multimodal benchmark functions. The tool can be used to compare SFLA, GA, PSO and MA over various benchmark functions. SEVO is based on JAVA and is developed using JDK1.6.0 tool kit. Genetic and evolutionary tool box [4] provides sophisticated ways of solving various problems through genetic algorithm. The tool box is implemented on Matlab platform and can evaluate algorithm on various input parameters. ACO tool box [7] provides the solution of Travelling Salesman Problem and Capacitated P-Medians Problem through ACO algorithm but not on benchmark functions. PSO toolbox makes use of Matlab's optimization toolbox for implementing PSO on various types of benchmark problems. It can be easily used by students, researchers, scientist etc. involved in the field computational intelligence. Karaboga and his scholars developed the software (ABC2 and ABCScatter) for evaluating ABC algorithm on benchmark functions by supplying various input parameters [8]. ABC2 provides support for 5 benchmark problems while ABCScatter can be used to evaluate ABC over 4 benchmark function. BBO algorithm is implemented in nature inspired Matlab based tool box [9]. This toolbox also provides support for PSO along with BBO. However, there are no automated tools available for other contemporary algorithms such as FFA, CCA, BA and FPA   for direct evaluation over benchmark test functions.

Thus, there is no single toolbox available for evaluating all nature-inspired algorithms over benchmark functions; this provides an opportunity for the research community to develop the unified toolbox that could be used to simulate the performance of these algorithms over benchmark test problems. Further, toolbox can also be extended for solving various other types of combinatorial, NP hard, complex optimization problems etc. Moreover, there is scope for evaluating the performance of contemporary nature-inspired algorithm on large scale global optimization problems and identify the need to extend these algorithm to solve "curse of dimensionality" problem.

5. CONCLUSION AND FUTURE WORK

Nature inspired algorithms are motivated from natural ecosystem and simulate the behavior of natural living and non living things. This paper presented an extensive review of 12 nature-inspired algorithms. The work highlighted the important features of these algorithms in terms of their input parameters, evolutionary mechanism and applications. The main focus of this article was to enlighten the research community with the optimization capability of contemporary algorithms over multi-modal and unimodal continuous functions for large scale global optimization. It also helps researcher or practitioner to gain insight of the various toolboxes available for simulating nature inspired algorithms over benchmark problems. Thus, this paper would act as a boon to the research community in identifying the research prospects in the field of nature inspired algorithms. In future, the work will focus on more comprehensive evaluation of nature inspired algorithms with graphical and tabular analysis.

6. REFERENCES

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

[2] I. Fister, X. S. Yang, J. Brest, and D. Fister, "A Brief Review of Nature-Inspired Algorithms for Optimization", Elektrotehnişki Vestnik, Jul, 2013.

[3] H. Banati and S. Mehta, "SEVO: Bio-inspired Analytical Tool for Unimodal and Multimodal Optimization", International Conference on Soft Computing for Problem Solving, Advances in Intelligent and Soft Computing, Springer, 2011, pp. 557-566.

[4] H. Pohlheim, "Genetic and Evolutionary Algorithm Toolbox (GEATbx) for Use with Matlab" http://www.geatbx.com, 1998.

[5] http://www.mathworks.com/matlabcentral/fileexchange/ 7506

[6] K. C. Tan, T. H. Lee, D. Khoo, and E. F. Khor, "A Multiobjective Evolutionary Algorithm Toolbox for Computer-Aided Multiobjective Optimization", IEEE transactions on systems, man and cybernetics-PartB: cybernetics, Vol. 31, No. 4, 2001.

[7] Toolbox - Ant Colony Optmization , Available: http://www.lbic.free.unicamp.br/homepage/downloads/ac oPOR.htm

[8] ABC Algorithm Information, Available: http://mf.erciyes.edu.tr/abc/software.htm

[9] Nature Inspired MATLAB based Toolbox!, Available: http://www.drsatvir.in/tools.php#

[10] E. Elbeltagi, T. Hegazy and D. Grierson, "Comparison among five evolutionary-based optimization algorithms", Advanced Engineering Informatics, Elsevier, 2005, pp. 43–53.

[11] X. S. Yang, "Nature- Inspired Mateheuristic Algorithms: Success and New Challenges", J Comput. Eng. Inf. Technol., Vol. 1, Issue 1, 2012, pp.1-3.

[12] X.S. Yang, Z. Cue, R. Xiao, A. H. Gandomi et al., "Swarm Intelligence and Bio-Inspired computation, Theory and Applications", Elsevier, Waltham, Mass, USA, 2013.

[13] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm", Applied Soft Computing, Elsevier, 2007, pp. 687-697.

[14] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", Journal of Global Optimization, Springer, 2007, pp. 459-471.

[15] K. M. Passino, "Biomimicry of Bacterial Foraging for Distributed Optimization and Control", IEEE Control Systems Magazine, 2006, pp. 52-67.

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

[17] X. S. Yang and X. He, "Firefly Algorithm: Recent Advances and Applications", Int. J. Swarm Intelligence, Vol. 1, No. 1, 2013, pp. 36–50.

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

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

[20] J. G. Digalakis and K. G. Margaritison, "Benchmarking Functions for Genetic Algorithms", Intern. J. Computer Math., Vol. 00, 2000, pp. 1-27.

[21] DE. Goldberg, "Genetic algorithms in search, optimization and machine learning", Reading, MA, Addison-Wesley, 1989.

[22] P. Moscato and MG. Norman, "A memetic approach for the traveling salesman problem implementation of a computational ecology for combinatorial optimization on message-passing systems", International conference on parallel computing and transputer application, 1992, pp. 177–86.

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

[24] M. Dorigo, V. Maniezzo and A. Colorni, "Ant system: optimization by a colony of cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics, Vol.26, No.1, 1996, pp.1-13.

[25] M. Eusuff, K. Lansey and F. Pasha, "Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization", Engineering Optimization, Vol. 38, No. 2, 2006, pp.129-154.

[26] D. Simon, "Biogeography-Based Optimization", IEEE Transactions on Evolutionary Computation, Vol. 12, No. 6, DEC 2008, pp.702-713.

[27] V. Kadirkamanathan, "Bayesian Inference for Basis Function Selection in Nonlinear System Identification using Genetic Algorithms", Maximum Entropy and Bayesian Methods, Fundamental Theories of Physics, Vo. 70, 1996, pp. 135-142.

[28] X. S. Yang, "Flower pollination algorithm for global optimization", Unconventional Computation and Natural Computation, Lecture Notes in Computer Science, Vol. 7445, 2012, pp. 240–249.

[29] X.S. Yang, M. Karamanoglu and X. He, "Multi-objective Flower Algorithm for Optimization", International Conference on Computational Science, Elsevier Science, Vol. 18, 2013, pp.861-868.

[30] A. Parashar and K. K. Swankar, "Genetic algorithm using to the solution of unit commitment", International Journal of Engineering Trends and Technology, Vol. 4, No. 7, 2013, pp. 2986-2990.

[31] K. Tang, X. Yao, P. N. Suganthan, C. MacNish, et al. "Benchmark Functions for the CEC'2008 Special Session and Competition on Large Scale Global Optimization", Technical Report, University of Science and Technology of China, 2008.

[32] D. Karaboga and B. Akay, "An artificial bee colony (abc) algorithm on training artificial neural networks", 15th IEEE Signal Processing and Communications Applications, SIU, June 2007, pp. 1–4.

[33] D. Karaboga, C. Ozturk and B. Akay, "Training neural networks with ABC optimization algorithm on medical pattern classification", International Conference on Multivariate Statistical Modelling and High Dimensional Data Mining, TURKEY, June 2008.

[34] C. Ozturk and D. Karaboga, "Classification by neural networks and clustering with artificial bee colony (ABC) algorithm", Sixth International Symposium on Intelligent and Manufacturing Systems Features, Strategies and Innovation Turkiye, October 2008.

[35] L. Fenglei, D. Haijun and F. Xing, "The parameter improvement of bee colony algorithm in TSP problem", Science Paper Online, November 2007.

[36] A. Singh, "An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem", Applied Soft Computing, Vol. 9, No.2, 2008, pp. 625–631.

[37] D. H. Kumar, P S. Babu and M P. Lalitha, "Distribution System Network Reconfiguration by using Artificial Bee Colony Algorithm", IOSR Journal of Electrical and Electronics Engineering, Vol. 9, No.1, Jan. 2014, pp. 48-52.

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

[39] Nitesh Sureja, "New Inspirations in Nature: A Survey", International Journal of Computer Applications & Information Technology, Vol. 1, No. 3, November 2012, pp. 21-24.

[40] S. Binitha and S. S. Sathya, "A Survey of Bio inspired Optimization Algorithms, International Journal of Soft Computing and Engineering, Vol.2, No. 2, May 2012, pp. 137-151.

[41] 2014 IEEE Congress on Evolutionary Computation (CEC), Available: http://www.ieee.org/conferences_ events/conferences/conferencedetails/index.html?Conf_I D=32438

[42] C Wu, N Zhang, J Jiang, J Yang and Y Liang, "Improved Bacterial Foraging Algorithms and Their Applications to Job Shop Scheduling Problems", 8th International Conference, ICANNGA 2007, Springer Berlin Heidelberg, Vol. 4431, 2007, pp. 562-569.

[43] [43] R Jakhar, N Kaur and R Singh, "Face Recognition Using Bacteria Foraging Optimization-Based Selected Features", International Journal of Advanced Computer Science and Applications, pp. 106-111.

[44] L Tan, H Wang, X Liang and K Xing, "An Adaptive Comprehensive Learning Bacterial Foraging Optimization for Function Optimization", Communications in Computer and Information Science, Vol. 375, 2013, pp. 194-199.

[45] O. P. Verma, R. Sharma and D. Kumar, "Binarization based Image Edge Detection using Bacterial Foraging Algorithm", 2nd International Conference on Communication," Elsevier, Vol. 6, 2012, pp. 315–323.

[46] Beenu, S. Kaur, "Image Segmentation using Improved Bacterial Foraging Algorithm", International Journal of Science and Research, Vol. 2, No. 1, 2013, pp. 63-69.

[47] R. Kaur, A. Girdhar and S Gupta, "Color Image Quantization based on Bacteria Foraging Optimization", International Journal of Computer Applications, Vol. 25, No.7, 2011, pp. 33-42.

[48] R. Lourenço and D. Serra, "Adaptive search heuristics for the generalized assignment problem", Mathware & soft computing, vol.9, no.2-3, 2002.

[49] R. Hadji, M. Rahoual, E. Talbi and V. Bachelet, "Ant colonies for the set covering problem", Abstract proceedings of ANTS2000, pp. 63-66.

[50] D. Martens, M. De Backer, R. Haesen, J. Vanthienen, M. Snoeck, B. Baesens, "Classification with Ant Colony Optimization", IEEE Transactions on Evolutionary Computation, Vol. 11, No. 5, 2003, pp. 651-665.

[51] G. D. Caro and M. Dorigo, "Extending AntNet for best-effort quality-of-service routing," Proceedings of the First Internation Workshop on Ant Colony Optimization (ANTS'98), 1998.

[52] S. Fidanova, "ACO algorithm for MKP using various heuristic information", Numerical Methods and Applications, vol.2542, 2003, pp.438-444.

[53] S. Gupta , K. Bhuchar and P. S. Sandhu, "Implementing Color Image Segmentation Using Biogeography Based Optimization" , International Conference on Software and Computer Applications, vol.9, 2011, pp. 79-86.

[54] B. Zhao, C. Den, Y. Yang and H. Peng, "Novel Binary Biogeography-Based Optimization Algorithm for the Knapsack Problem", 2012, pp. 217–224.

[55] V.K. Panchal, "Biogeography based Satellite Image Classification", International Journal of Computer Science and Information Security, Vol. 6, No.2, 2009.

[56] M. H. Horng, "Vector quantization using the firefly algorithm for image compression", Expert Systems with Applications, 2012, pp. 1078-1091.

[57] H. Banati and M. Bajaj, "Firefly based feature selection approach", Int. J. Computer Science Issues, Vol. 8, No. 2, 2011, pp. 473-480.

[58] S. Nandy, P. P. Sarkar and A. Das, "Analysis of nature-inspired firefly algorithm based back-propagation neural network training", Int. J. Computer Applications, Vol. 43, No. 22, 2012, pp. 8-16.

[59] A. Rajini, V. K. David, "A hybrid metaheuristic algorithm for classification using micro array data", Int. J. Scientific & Engineering Research, Vol. 3, No.2, 2012, pp. 1-9.

[60] J. Senthilnath, S. N. Omkar and V. Mani, "Clustering using firely algorithm: performance study", Swarm and Evolutionary Computation, Vol. 1, No. 3, 2011, pp. 164-171.

[61] S. S. Travessa, W. P. Carpes and M. A. Nunes Filho, "Use of an Artificial Neural Network-based metamodel in the optimization by Particle Swarm Optimization method", Universidade Federal de Santa Catarina.

[62] G M. Omran, A. P. Engelbrecht, and A. Salman, "Particle Swarm Optimization Method for Image Clustering, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 19, No. 03, pp. 297-321.

[63] J. Ni, L. Li, F. Qiao, Q. Wu, "A novel memetic algorithm and its application to data clustering", Memetic Computing, March 2013, Vol. 5, No. 1, pp 65-78.

[64] T. Liao, K. Socha, M. A. M. de Oca, Thomas Stutzle, et al., "Ant Colony Optimization for Mixed-Variable Optimization Problems", IEEE transactions on evolutionary computing, 2013.

[65] M. Dhivya and M. Sundarambal, "Cuckoo search for data gathering in wireless sensor networks", Int J Mobile Communication", 2011, pp. 642–656.

[66] A. Layeb, "A novel quantum-inspired cuckoo search for Knapsack problems", International Journal of Bio-inspired Computation, Vol. 3, No. 5, 2011, pp. 297–305

[67] E. Valian, S. Mohanna and S. Tavakoli, "Improved cuckoo search algorithm for feedforward neural network training" International Journal of Artificial Intelligence Application, Vol. 2, No. 3, 2011, pp.36–43

[68] G. Komarasamy and A. Wahi, "An optimized K-means clustering technique using bat algorithm", European J. Scientific Research, Vol. 84, No. 2, 2012, pp. 263-273.

[69] M. G. H. Omran, A. P. Engelbrecht and A. Salman, "Particle Swarm Optimization for Pattern Recognition and Image Processing", Swarm Intelligence in Data Mining, Studies in Computational Intelligence, Vol. 34, 2006, pp 125-151.

[70] A. Rahman, E. M. Ahmad and A. R. Akhtar, "A metaheurisic bat inspired algorithm for full body human pose estimation", Ninth Conference on Computer and Robot Vision, 2012, pp. 369–375.

[71] T. A. Lemma, B. M. Hashim, "Use of fuzzy systems and bat algorithm for exergy modelling in a gas turbine generator", IEEE Colloquium on Humanities, Science and Engineering, 2011, pp. 305–310.

[72] Z. Zhijin, Y. Keqiang and Z. Zhidong, "Discrete shuffled frog leaping algorithm for multi-user detection in DS-CDMA communication system," in 2008 11th IEEE International Conference on Communication Technology, ICCT 2008, Hangzhou, China, pp. 421-424, 2008.

[73] Z. Wang and X. Sun, "Image Watermarking Scheme Based on Shuffled Frog Leaping Algorithm", Knowledge Acquisition and Modeling Workshop, IEEE International Symposium on, 2008, pp. 239 – 242.

[74] B. Amiri, M. Fathian and A. Maroosi, "Application of shuffled frog-leaping algorithm on clustering", The International Journal of Advanced Manufacturing Technology, November 2009, Vol. 45, No. 1-2, pp 199-209.

[75] H. Chen, Y. Zhu, and K. Hu, "Adaptive Bacterial Foraging Optimization", Abstract and Applied Analysis, Hindawi, Vol. 2011, pp. 1-27.

[76] S. Mehta and H. Banati, "Improved shuffled frog leaping algorithm for continuous optimization adapted SEVO toolbox", International. J. of Advanced Intelligence Paradigms, Inderscience, Vol. 5, No.1/2, 2013, pp. 31 – 44.