International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 12 Issue 1–DECEMBER 2014.

Nature Inspired Algorithms: A Review

M. Bhuvaneswari[a], S. Hariraman[b], B. Anantharaj[b], and N. Balaji[b]

[a] PG Student, Department of Computer Science and Engineering, Thiruvalluvar College of Engg & Tech, Vandavasi, Tamil Nadu
[b] Asst. Professor, Department of Computer Science and Engineering, Thiruvalluvar College of Engg & Tech, Vandavasi, Tamil Nadu
bhuvanasince1980@gmail.com,
hariraman.vs@gmail.com,
ananthu_arun72@yahoo.com,
nbalajime1983@gamil.com

ABSTRACT - Swarm intelligence and bio-inspired algorithms form recent trends in the increase of new nature inspired algorithms. This paper presents a brief survey about the existing Algorithms in Nature Inspired Algorithms. The existing algorithms based on Evolutionary Algorithm (EA), Physical Algorithms (PA), Swarm Intelligence (SI), Bio-Inspired Algorithms (BIA) and Other Nature Inspired Algorithms. However, the study reveals that the existing algorithms to improve the optimization performance in different analysis. The intention of this review is to present a broad classification of all the nature inspired algorithms and to motivate for the research.

Keywords -Nature Inspired Algorithms, Evolutionary Algorithm, Physical Algorithms, Swarm Intelligence, Bio-Inspired Algorithms.

I. INTRODUCTION

The most established "classical" nature-inspired models of computation are cellular automata, neural computation, and evolutionary computation. More recent computational systems abstracted from natural processes include swarm intelligence, artificial immune systems, membrane computing, and amorphous computing. In fact, all major methods and algorithms are nature-inspired metaheuristic algorithms [1] including cellular automata, evolutionary computation, swarm intelligence, artificial immune systems, membrane computing, and amorphous computing.

A. Evolutionary Computation

Evolutionary computation [2] is a computational paradigm inspired by Darwinian evolution. An artificial evolutionary system is a computational system based on the notion of simulated evolution. It comprises a constant- or variable-size population of individuals, a fitness criterion, and genetically inspired operators that produce the next generation from the current one. The initial population is typically generated randomly or heuristically, and typical operators are mutation and recombination. At each step, the individuals are evaluated according to the given fitness function (survival of the fittest). The next generation is obtained from selected individuals (parents) by using genetically inspired operators. The choice of parents can be guided by a selection operator which reflects the biological principle of mate selection. This process of simulated evolution eventually converges towards a nearly optimal population of individuals, from the point of view of the fitness function.

B. Swarm intelligence

Swarm intelligence, [3] sometimes referred to as collective intelligence, is defined as the problem solving behavior that emerges from the interaction of individual agents (e.g., bacteria, ants, termites, bees, spiders, fish, birds) which communicate with other agents by acting on their local environments. Particle swarm optimization applies this idea to the problem of finding an optimal solution to a given problem by a search through a (multi-dimensional) solution space. The initial set-up is a swarm of particles, each representing a possible solution to the problem. Each particle has its own velocity which depends on its previous velocity (the inertia component), the tendency towards the past personal best position (the nostalgia component), and its tendency towards a global neighborhood optimum or local neighborhood optimum (the social component). Particles thus move through a multidimensional space and eventually converge towards a point between the global best and their personal best. Particle swarm optimization algorithms have been applied to various optimization problems, and to unsupervised learning, game learning, and scheduling applications.

C. Artificial immune systems

Artificial immune systems (a.k.a. immunological computation or immunocomputing) are computational systems inspired by the natural immune systems of biological organisms. Viewed as an information processing system, the natural immune system of organisms performs many complex tasks in parallel and distributed computing fashion. [4] These include distinguishing between self and nonself, [5] neutralization of nonself pathogens (viruses, bacteria, fungi, and parasites), learning, memory, associative retrieval, self-regulation, and fault-tolerance. Artificial immune systems are abstractions of the natural immune system, emphasizing these computational aspects. Their applications include computer virus detection, anomaly detection in a time series of data, fault diagnosis, pattern recognition, machine learning, bioinformatics, optimization, robotics and control.

D. Membrane computing

Membrane computing (or MC) is an area within computer science that seeks to discover new computational models from the study of biological cells, particularly of the cellular membranes. It is a sub-task of creating a cellular model. Membrane computing deals with distributed and parallel computing models, processing multisets of symbol objects in a localized manner. Thus, evolution rules allow for evolving objects to be encapsulated into compartments defined by membranes. The communications between compartments and with the environment play an essential role in the processes. The various types of membrane systems are known as P systems after Gheorghe Păun who first conceived the model in 1998.[6]

Applications of membrane systems include machine learning, modelling of biological processes (photosynthesis, certain signaling pathways, quorum sensing in bacteria, cell-mediated immunity), as well as computer science applications such as computer graphics, public-key cryptography, approximation and sorting algorithms, as well as analysis of various computationally hard problems.

E. Amorphous computing

In biological organisms, morphogenesis (the development of well-defined shapes and functional structures) is achieved by the interactions between cells guided by the genetic program encoded in the organism's DNA. Inspired by this idea, amorphous computing aims at engineering well-defined shapes and patterns, or coherent computational behaviours, from the local interactions of a multitude of simple unreliable, irregularly placed, asynchronous, identically programmed computing elements (particles).[7] As a programming paradigm, the aim is to find new programming techniques that would work well for amorphous computing environments. Amorphous computing also plays an important role as the basis for "cellular computing".

II. NATURE INSPIRED ALGORITHMS

The Fig 1. Presents the classification of the existing nature inspired algorithms like Evolutionary Algorithms (EA), Physical Algorithms (PA), Swarm Intelligence (SI), Bio-Inspired Algorithms (BIA), and Other Nature Inspired Algorithms.

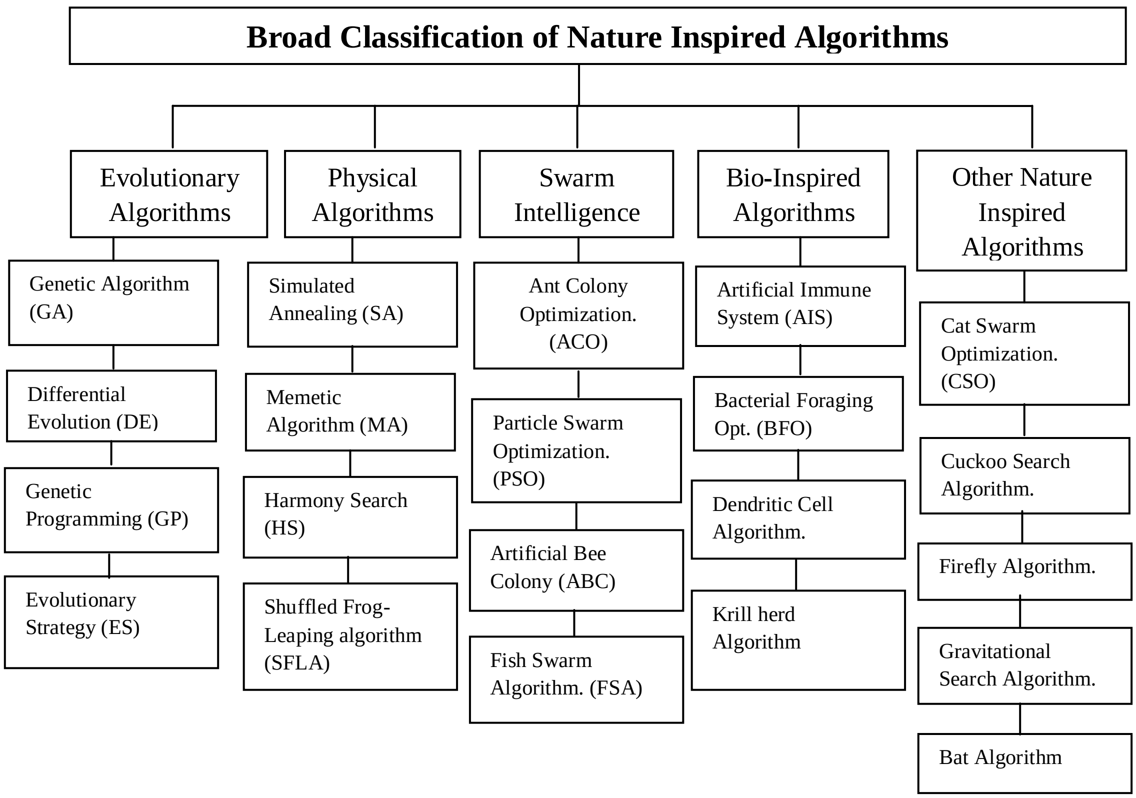

Fig 1. Broad Classification of Nature Inspired Algorithms

1. Evolutionary Algorithms (EA)

Evolutionary algorithms (EAs) are extensively used for solving many diverse domains of science and real-time engineering applications to find an optimum solution of multimodal functions [8, 9]. The term Evolutionary algorithm [8] is used to designate a collection of optimization techniques whose functioning is loosely based on metaphors of biological processes. Evolutionary computation is a paradigm in the artificial intelligence that involves collective phenomena in adaptive populations of problem solvers utilizing the iterative progress comprising growth, development, reproduction, selection and survival as seen in a population. EAs are the most well known, classical and established algorithms among nature inspired algorithms, based on the biological evolution in nature which is responsible for the purpose of all living beings on earth.

They have different functional components as the fitness function, initialization, selection, recombination, mutation and replacement.

The EA family members are Genetic algorithm (GA), Differential Evolution (DE), Genetic programming (GP), Evolutionary strategy (ES), and Granular Agent Evolutionary Algorithm.

A. Genetic Algorithm (GA)

In the field of artificial intelligence, a Genetic Algorithm (GA) is a search heuristic that mimics the process of natural selection [10]. This heuristic (also sometimes called a metaheuristic) is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

Genetic algorithms find application in bioinformatics,phylogenetics, computational science, engineering, economics, chemistry, manufacturing, mathematics, physics, pharmacometrics and other fields[11].

B. Differential Evolution (DE)

In computer science, Differential Evolution (DE) is a method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. [12] Such methods are commonly known as metaheuristics as they make few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. However, metaheuristics such as DE do not guarantee an optimal solution is ever found.

DE is used for multidimensional real-valued functions but does not use the gradient of the problem being optimized, which means DE does not require for the optimization problem to be differentiable as is required by classic optimization methods such as gradient descent and quasi-newton methods. DE can therefore also be used on optimization problems that are not even continuous, are noisy, change over time, etc.

C. Genetic Programming

In artificial intelligence, Genetic Programming (GP) is an evolutionary algorithm-based methodology inspired by biological evolution to find computer programs that perform a user-defined task. Essentially GP is a set of instructions and a fitness function to measure how well a computer has performed a task. It is a specialization of genetic algorithms (GA) where each individual is a computer program. It is a machine learning technique used to optimize a population of computer programs according to a fitness landscape determined by a program's ability to perform a given computational task.

D. Evolutionary Strategy (ES)

In computer science, an Evolution Strategy (ES) is an optimization technique based on ideas of adaptation and evolution. It belongs to the general class of evolutionary computation or artificial evolution methodologies.

The 'evolution strategy' optimization technique was created in the early 1960s and developed further in the 1970s and later by Ingo Rechenberg, Hans-Paul Schwefel and their co-workers.

(ES, see Rechenberg, 1994) evolve individuals by means of mutation and intermediate or discrete recombination. ES algorithms are designed particularly to solve problems in the real-value domain. They use self-adaptation to adjust control parameters of the search. De-randomization of self-adaptation has led to the contemporary Covariance Matrix Adaptation Evolution Strategy (CMA-ES).

2. Swarm Intelligence (SI)

Swarm intelligence (SI) is a sub-field of evolutionary computing. Swarm intelligence (SI) is the collective behavior of decentralized, self-organized systems, natural or artificial. The concept is employed in work on artificial intelligence. The expression was introduced by Gerardo Beni and Jing Wang in 1989, in the context of cellular robotic systems.[13] SI systems consist typically of a population of simple agents or boids interacting locally with one another and with their environment. The inspiration often comes from nature, especially biological systems. The agents follow very simple rules, and although there is no centralized control structure dictating how individual agents should behave, local, and to a certain degree random, interactions between such agents lead to the emergence of "intelligent" global behavior, unknown to the individual agents. Examples in natural systems of SI include ant colonies, bird flocking, animal herding, bacterial growth, and fish schooling. The definition of swarm intelligence is still not quite clear. In principle, it should be a multi-agent system that has self-organized behaviour that shows some intelligent behaviour.

The application of swarm principles to robots is called swarm robotics, while 'swarm intelligence' refers to the more general set of algorithms. 'Swarm prediction' has been used in the context of forecasting problems.

The SI family members are Ant Colony Optimization (ACO), Particle Swarm Optimization. (PSO), Artificial Bee Colony (ABC) and Fish Swarm Algorithm. (FSA).

A. Ant Colony Optimization (ACO)

Ant colony optimization (ACO), introduced by Dorigo in his doctoral dissertation, is a class of optimization algorithms modeled on the actions of an ant colony. ACO is a probabilistic technique useful in problems that deal with finding better paths through graphs. Artificial 'ants'—simulation agents—locate optimal solutions by moving through a parameter space representing all possible solutions. Natural ants lay down pheromones directing each other to resources while exploring their environment. The simulated 'ants' similarly record their positions and the quality of their solutions, so that in later simulation iterations more ants locate better solutions.[14]

B. Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space. Hypotheses are plotted in this space and seeded with an initial velocity, as well as a communication channel between the particles. [15, 16] Particles then move through the solution space, and are evaluated according to some fitness criterion after each timestep. Over time, particles are accelerated towards those particles within their communication grouping which have better fitness values. The main advantage of such an approach over other global minimization strategies such as simulated annealing is that the large number of members that make up the particle swarm make the technique impressively resilient to the problem of local minima.

C. Artificial Bee Colony (ABC)

Artificial bee colony algorithm (ABC) is a meta-heuristic algorithm introduced by Karaboga in 2005, [17] and simulates the foraging behaviour of honey bees. The ABC algorithm has three phases: employed bee, onlooker bee and scout bee. In the employed bee and the onlooker bee phases, bees exploit the sources by local searches in the neighbourhood of the solutions selected based on deterministic selection in the employed bee phase and the probabilistic selection in the onlooker bee phase. In the scout bee phase which is an analogy of abandoning exhausted food sources in the foraging process, solutions that are not beneficial anymore for search progress are abandoned, and new solutions are inserted instead of them to explore new regions in the search space. The algorithm has a well-balanced exploration and exploitation ability.

D. Fish Swarm Algorithm (FWA)

Fish Swarm Algorithm (FSA): The FSA algorithm is derived from the schooling behavior of fish. Cheng et al. [18] applied the FSA for cluster analysis. The algorithm operates by mimicking three important behavior of natural fish: searching behavior (tendency of fish to look at food), swarming behavior (fish assembles in swarms to minimize danger) and following behavior (when a fish identify food source, its neighboring individuals follow based on fish 0s visual power). Tsai and Lin [19] have reported improved solution provided by FSA compared to PSO for several optimization problems.

3. Bio-Inspired Algorithms (BIA)

Obviously, SI-based algorithms belong to a wider class of algorithms, called Bio-Inspired Algorithms (BIA). In fact, bio-inspired algorithms form a majority of all nature-inspired algorithms. From the set theory point of view, SI-based algorithms are a subset of bio-inspired algorithms, while bio-inspired algorithms are a subset of nature-inspired algorithms [20].

Many bio-inspired algorithms do not use directly the swarming behaviour. Therefore, it is better to call them bio-inspired, but not SI-based. For example, genetic algorithms are bio-inspired, but not SI-based. However, it is not easy to classify certain algorithms such as differential evolution (DE). Strictly speaking, DE is not bio-inspired because there is no direct link to any biological behaviour. However, as it has some similarity to genetic algorithms and also has a key word 'evolution', we tentatively put it in the category of bio-inspired algorithms.

The BIA family members are Artificial Immune System (AIS), Bacterial Foraging Optimization (BFO), Dendritic Cell Algorithm and Krill herd Algorithm.

A. Artificial Immune System (AIS)

In computer science, Artificial Immune Systems (AIS) are a class of computationally intelligent systems inspired by the principles and processes of the vertebrate immune system. The algorithms typically exploit the immune system's characteristics of learning and memory to solve a problem. The field of Artificial Immune Systems (AIS) is concerned with abstracting the structure and function of the immune system to computational systems, and investigating the application of these systems towards solving computational problems from mathematics, engineering, and information technology [5]. AIS is a sub-field of Biologically-inspired computing, and Natural computation, with interests in Machine Learning and belonging to the broader field of Artificial Intelligence. Artificial Immune Systems (AIS) are adaptive systems, inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving.

B. Bacterial Foraging Optimization (BFO)

BFO-based approaches: Passino [21] proposed the Bacterial Goraging Optimization (BFO) algorithm in 2002 which imitates the foraging strategies of E. coli bacteria for finding food. An E. coli bacterium can search for food in its surrounding by two types of movements: run or tumble. These movements are possible with the help of flagella (singular, flagellum) that enable the bacterium to swim. If the flagella move counter clockwise, their effects accumulate in the form of a bundle which pushes the bacterium to move forward in one direction (run). When the flagella rotate clock wise, each flagellum separates themselves from the others and the bacterium tumbles (it does not have any set direction for movement and there is almost no displacement). The bacterium alternates between these two modes of operation throughout its entire lifetime. After the initial development by Passino the algorithm gradually has become popular due to its capability to provide good solution in dynamic [22] and multi-modal [23] environments.

C. Dendritic Cell Algorithm

The Dendritic Cell Algorithm (DCA) is an example of an immune inspired algorithm developed using a multi-scale approach. This algorithm is based on an abstract model of dendritic cells (DCs). The DCA is abstracted and implemented through a process of examining and modeling various aspects of DC function, from the molecular networks present within the cell to the behaviour exhibited by a population of cells as a whole. Within the DCA information is granulated at different layers, achieved through multi-scale processing.[24]

D. Krill herd Algorithm

A novel biologically-inspired algorithm, namely krill herd (KH) is proposed for solving optimization tasks. [25] The KH algorithm is based on the simulation of the herding behavior of krill individuals. The minimum distances of each individual krill from food and from highest density of the herd are considered as the objective function for the krill movement. The time-dependent position of the krill individuals is formulated by three main factors: (i) movement induced by the presence of other individuals (ii) foraging activity, and (iii) random diffusion. For more precise modeling of the krill behavior, two adaptive genetic operators are added to the algorithm.

The formation of the krill herd after predation depends on many parameters. The herding of the krill individuals is a multi-objective process including two main goals: (1) increasing krill density, and (2) reaching food. In the present study, this process is taken into account to propose a new metaheuristic algorithm for solving global optimization problems. Density-dependent attraction of krill (increasing density) and finding food (areas of high food concentration) are used as objectives which finally lead the krill to herd around the global minima.

III. CONCLUSION

This paper report the characteristics of the existing nature inspired algorithms like Evolutionary Algorithm (EA), Physical Algorithms (PA), Swarm Intelligence (SI), Bio-Inspired Algorithms (BIA) and Other Nature Inspired Algorithms have been discussed briefly. However, the study reveals that the existing nature inspired algorithms to improve the optimization performance has been analyzed. In future the above discussed nature inspired algorithms can be efficiently applied in Multidimensional 0-1 Knapsack.

IV. REFERENCES

[1]Yang, X.-S., Nature-inspired metaheuristic algorithms, Luniver Press, (2008).
[2] T., Fogel, D., Michalewicz, Z., editors. Handbook of Evolutionary Computation. IOP Publishing, U.K., 1997.
[3]Engelbrecht,A .Fundamentals of Computational Swarm Intelligence.Wiley and Sons, 2005.

[4] Dasgupta, D. editor. Artificial Immune Systems and Their Applications. Springer, 1998.

[5] de Castro, L., Timmis, J. Artificial Immune Systems: A New Computational Intelligence Approach. Springer, 2002.

[6] Paun, G. Membrane Computing: An Introduction. Springer, 2002.

[7] Abelson, H., Allen, D., Coore, D., Hanson, C., Homsy, G., Knight Jr., T., Nagpal, R., Rauch, E., Sussman, G., Weiss, R. Amorphous computing. Communications of the ACM 43, 5 (May 2000), 74-82.

[8] Rainer Storn, Kenneth Price, Differential evolution a simple and efficient heuristic for global optimization over continuous spaces, J. Glob.Optim.11 (4) (1997)341–359.

[9]S. Das,P.Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Trans.Evol.Comput.60(4)(2011)1469–1479.

[10] Mitchell 1996, p. 2.

[11] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison - Wesley, 1989, ISBN0201157675.

[12] 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.

[13] Beni, G., Wang, J. Swarm Intelligence in Cellular Robotic Systems, Proceed. NATO Advanced Workshop on Robots and Biological Systems, Tuscany, Italy, June 26–30 (1989).

[14] Ant Colony Optimization by Marco Dorigo and Thomas Stützle, MIT Press, 2004. ISBN 0-262-04219-3.

[15] Parsopoulos, K. E.; Vrahatis, M. N. (2002). "Recent Approaches to Global Optimization Problems Through Particle Swarm Optimization". Natural Computing 1 (2-3):235–306.

by Maurice Clerc, ISTE, ISBN 1-905209-04-5, 2006.

[17] Karaboga, Dervis (2010). "Artificial bee colony algorithm". Scholarpedia 5 (3): 6915. doi:10.4249/scholarpedia.6915.

[18] Y.Cheng, M.Jiang, D.Yuan, Novel clustering algorithms based on improved artificial fish swarm algorithm, IEEE Sixth International Conference on Fuzzy Systems and Knowledge Discovery, vol.3, 2009, pp.141–145.

[19] H.C. Tsai, Y.H.Lin, Modification of the fish swarm algorithm with particle swarm optimization formulation and communication behavior, Appl.Soft Comput. 11(2011)5367–5374.

[20]. Iztok Fister Jr., Xin-She Yang, Iztok Fister, Janez Brest, Duˇsan Fister, “A Brief Review of Nature-Inspired Algorithms for Optimization”, Elektrotehni Ski Vestnik 80(3): 1–7, 2013, English Edition.

[21] K. Passino, Bio mimicry of bacterial foraging for distributed optimization and control, IEEE Control Syst.Mag. 22 (3) (2002) 52 – 67.

[22] W.J.Tang, Q.H.Wu, J.R.Saunders, Bacterial foraging algorithm for dynamic environments, in:IEEE Congress on Evolutionary Computation, CEC2006, pp. 1324–1330.

[23] S. Mishra, Ahybrid least square—fuzzy bacterial foraging strategy for harmonic estimation, IEEE Trans.Evol.Comput. 9 (1) (2005) 61–73.

[24] Greensmith, J.; Aickelin, U. (2009). "Artificial Dendritic Cells: Multi-faceted Perspectives" (PDF). Human-Centric Information Processing Through Granular Modelling: 375–395.

[25] A.H.Gandomia, A.H.Alavib, Krillherd: a new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer.Simul.17 (12)(2012)4831–4845