A Brief Review of Bio-Inspired Algorithms in Computational Perspective

Bio Inspired Algorithms

Dr. Pankaj Lathar
Asst. Professor, Department of Information Technology
Ch. Brahm Prakash Government Engineering College
Delhi, India
e-mail: pankajlathar@gmail.com

Abstract— Computing over the years has evolved from being simplex mathematical processing machine to more sophisticated problem solving entity pushing limits around reasoning and intelligence. Along the way, lots scientists and engineers have closely observed some of the biological processes achieving certain things in a more efficient and simple fashion than traditional computational mechanisms. This has led to development of various techniques and algorithms which try and mimic these biological processes and are categorised under, Bio-Inspired Computing.

Keywords- *Bio-ispired Computing, Machine Learning, Computational Modeling*

*****

I. Introduction

Need for bio-inspired computing mechanisms? Well, more than the question of need, its a question of capabilities and explanations provided in the classical computing world. The following captures best the capabilities of the classical computing paradigm.

Classical computing paradigm is **good** at:

- Number-crunching
- Thought-support (glorified pen-and-paper)
- Rule-based reasoning
- Constant repetition of well-defined actions.

Classical computing paradigm is **bad** at:

- Pattern recognition
- Robustness to damage
- Dealing with vague and incomplete information;
- Adapting and improving based on experience

The term bio-inspired has been introduced to demonstrate the strong relation between a particular system or algorithm, which has been proposed to solve a specific problem, and a biological system, which follows a similar procedure or has similar capabilities.

- Bio-inspired computing represents a class of algorithms focusing on efficient computing, e.g. for optimization processes and pattern recognition.
- Bio-inspired systems rely on system architectures for massively distributed and collaborative systems, e.g. for distributed sensing and exploration.
- Bio-inspired networking is a class of strategies for efficient and scalable networking under uncertain conditions, e.g. for delay tolerant networking.

The process of arriving at a bio-inspired solution typically involves the following steps:

- Identification of analogies
  - Analogy of genesis used in many prediction problems to identify anomalies
  - Swarm to design fault tolerant and distributed computing solutions trying to optimize resources
- Understanding
  - Modelling a realistic biological behavior e.g. understanding the various decisive steps involved in genesis - survival of the fittest, crossover, mutation, selection etc.

II. SWARM INTELLIGENCE

Living organism are complex systems which act together to produce very desirable characteristics few of which include fault tolerance, distributed, evolution and growth. These characteristics are harder to achieve in classical computing paradigm. A class bio-inspired solution mimic the collective/group solutioning process and intelligence exhibited by several organisms and are referred to as swarm intelligence.

Swarming is a collective behaviour exhibited by animals of similar size which aggregate together, perhaps milling about the same spot or migrating in some direction. This behavior is mainly seen in smaller insects and in the bird family.

E.g.:

- Termites swarm to build colonies
- Birds swarm to find food
- Bees swarm to reproduce

The reasoning behind these creatures showing the swarm behaviour is mainly due to the following:

- Finding food is best achieved when done collectively - better forage
- Migration is a collective/social activity
- Increases the strength of defence and can aid in guarding against the predators

Plenty of swarming examples can be found in nature. Swarming is a powerful mechanism which can yield results which is out of reach of an individual entity if it were performing the same activity in isolation. Incase of birds, flocking is a swarm activity and is performed by following 3 simple rules.

- Collision Avoidance
  - Rule 1: Avoid Collision with neighboring birds

- Velocity Matching
  - Rule 2: Match the velocity of neighboring birds
- Flock Centering
  - Rule 3: Stay near neighboring birds

[TBD - Pictures]

Looking at the rules, one can deduce that the amount of work/compute and the kind of compute performed by every individual in the groups is very similar with different compute parameters. Because of the simplistic design principles involved, the swarm characteristics are very appealing and can be used to simplify complex computing systems performing similar tasks. The following are few of the characteristics of the swarm solutions:

- **Simple rules for each individual** : Individual participant node performs simple compute which is necessary to meet the requirements to be in swarm to achieve the end goal. This way the compute capabilities required are minimal or just enough to follow rules of the group activity
- **No central control :** Lot of the distributed systems built in this computing era have certain drawbacks, one of them is the master slave architecture and single point of failure. Because there is a master node controlling most of the distributed activity, if the master were to fail the entire group would fail. The fault tolerance mechanism built involves use of another distributed system altogether and increases the complexity of the system. Emerging distributed architectures have the shared nothing concept which can be thought of as an inspiration from these swarm solutions. Swarm solutions do not have central authority which increases the robustness of the entire solution by reducing impact of failure.
- **Emergent :** Emergent structures are a common strategy found in many animal groups: colonies of ants, mounds built by termites, swarms of bees, shoals/schools of fish, flocks of birds, and herds/packs of mammals. An example to consider in detail is an ant colony. The queen does not give direct orders and does not tell the ants what to do. Instead, each ant reacts to stimuli in the form of chemical scent from larvae, other ants, intruders, food and build up of waste, and leaves behind a chemical trail, which, in turn, provides a stimulus to other ants. Here each ant is an autonomous unit that reacts depending only on its local environment and the genetically encoded rules for its variety of ant. Despite the lack of centralized decision making, ant colonies exhibit complex behavior and have even been able to demonstrate the ability to solve geometric problems. For example, colonies routinely find the maximum distance from all colony entrances to dispose of dead bodies [wiki]

Swarm Intelligence - Definition: “any attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies” [Bonabeau, Dorigo, Theraulaz: Swarm Intelligence]

Application of Swarm Intelligence:

- The Lord of the Rings film trilogy made use of similar technology, known as Massive, during battle scenes
- In telecommunication networks in the form of ant based routing
- Swarm robotics is a new approach to the coordination of multirobot systems which consist of large numbers of mostly simple physical robots. These employ a majority of swarm intelligence solutions

III. PRATICLE SWARM OPTIMIZATION

Particle Swarm Optimisation(PSO) is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. It imitates human or insects social behavior. Individuals interact with one another while learning from their own experience, and gradually move towards the goal. It is easily implemented and has proven both very effective and quick when applied to a diverse set of optimization problems. One of the best example of PSO found in nature is that of Bird Flocking.

Over a number of iterations, a group of variables have their values adjusted closer to the member whose value is closest to the target at any given moment. Imagine a flock of birds circling over an area where they can smell a hidden source of food. The one who is closest to the food chirps the loudest and the other birds swing around in his direction. If any of the other circling birds comes closer to the target than the first, it chirps louder and the others veer over toward him. This tightening pattern continues until one of the birds happens upon the food. It's an algorithm that's simple and easy to implement. The algorithm keeps track of three global variables:

Target value or condition

Global best (gBest) value indicating which particle's data is currently closest to the Target

Stopping value indicating when the algorithm should stop if the Target isn't found

Each particle consists of:

Data representing a possible solution

A Velocity value indicating how much the Data can be changed

A personal best (pBest) value indicating the closest the particle's Data has ever come to the Target

The particles' data could be anything. In the flocking birds example, the data would be the X, Y, Z coordinates of each bird. The individual coordinates of each bird would try to move closer to the coordinates of the bird which is closer to the food's coordinates (gBest). If the data is a pattern or sequence, then individual pieces of the data would be manipulated until the pattern matches the target pattern.

IV. ANT COLONY OPTIMIZATION

Foraging has been seen as one of the primary reasons for exhibiting swarm behavior in insects and birds. Ants, especially, work in a very robust and distributed fashion where they optimise for time and energy at every step. Ant Colony Optimization is inspired by the foraging behavior shown by ants.

To begin with ants wander randomly out of their colony in search of food. Once a food source is located, ants lay down pheromone trails on their way back to the colony. It is very likely that there would be multiple trails for ants to follow at a given point in time. The decision for choice of trail is based on the density of pheromone deposited for a particular trail. Pheromone also evaporates over time which reduces density of the trail and thus leaving it as one of the poor choice for ants. Dense implies many ants have travelled the same path and less evaporation which implies the food source must be closer. Less dense implies that most likely the food source must be slightly far apart and the low density could be an indicator of more time exposed for evaporation due to longer distance of travel and less frequent and travel overlap. Pheromone now becomes the only means of communication for ants and this kind of indirect communication via the local environment is called stigmergy. The above described mechanism has better adaptability to local conditions, robustness and fault tolerance by means of redundancy.

The objective of the strategy is to exploit historic and heuristic information to construct candidate solutions and fold the information learned from constructing solutions into the history. Solutions are constructed one discrete piece at a time in a probabilistic step-wise manner. The probability of selecting a component is determined by the heuristic contribution of the component to the overall cost of the solution and the quality of solutions from which the component has historically known to have been included. History is updated proportional to the quality of the best known solution and is decreased proportional to the usage if discrete solution components.

Generic ACO:

- Formalized into a metaheuristic.
- Artificial ants build solutions to an optimization problem and exchange info on their quality vis-à-vis real ants.
- A combinatorial optimization problem reduced to a construction graph.
- Ants build partial solutions in each iteration and deposit pheromone on each vertex.

IV. NATURAL COMPUTING

Natural computing, is a terminology introduced to encompass three classes of methods: 1) those that take inspiration from nature for the development of novel problem-solving techniques; 2) those that are based on the use of computers to synthesize natural phenomena; and 3) those that employ natural materials (e.g., molecules) to compute. The main fields of research that compose these three branches are artificial neural networks, evolutionary algorithms, artificial immune systems, fractal geometry, artificial life, DNA computing, and quantum computing, among others. [wiki]

V. ARTIFICIAL IMMUNE SYSTEMS

Our body's immune system is a perfect example of a learning system. It is able to distinguish between good cells and. potentially harmful ones (Antigens). Artificial Immune system is a learning mechanism inspired by the human immune system. Following are some of the appealing factors of an immune system, which can help a computation problem

- The immune system promotes diversification. That is, it does not attempt to focus on a global optima, instead it evolves antibodies which can handle different antigens (situations)
- The immune system is a distributed system with no central controller. That is, the immune system is distributed throughout our bodies via its constituent cells and molecules.
- The immune system is a naturally occurring event–response system which can quickly adapt to changing situations.
- The immune system possesses a self-organizing memory which is dynamically maintained and which allows items of information to be forgotten. It is thus adaptive to its external environment.
- The immune system's memory is content addressable, thus allowing antigens to be identified by the same antibody. It is thus tolerant to noise in the antigens presented to it.

AIS has successfully been applied to a wide variety of problems, following are some of the examples:

- Computer Security,
- Pattern Recognition,
- Bridge Fault Detection
- Data Mining

Human immune system is a very sophisticated process of things involving multiple agents (cells and antibodies) working together for a cause. Before proceeding with the inner working of a learning algorithm, it is important to understand the terminologies involved and their corresponding purpose in the entire process.

There are two primarily types of lymphocytes:

- B-lymphocytes (B cells)
- T-lymphocytes (T cells)

Others types include macrophages, phagocytic cells, cytokines, etc.

Lymphocytes

Lymphocytes are small leukocytes that possess a major responsibility in the immune system. There are two main types of lymphocytes: B lymphocyte (or B cell), which, upon activation, differentiate into plasmocyte (or plasma cells) capable of secreting antibodies; and T lymphocyte (or T cell). Most of the lymphocytes is formed by small resting cells, which only exhibit functional activities after some kind of interaction with the respective antigens, necessary for proliferation an specific activation. The B and T lymphocytes express, on their surfaces, receptors highly specific for a given antigenic determinant. The B cell receptor is a form of the antibody molecule bound to the membrane, and which will be secreted after the cell is appropriately activated.

B cells and antibodies

The main functions of the B cells include the production and secretion of **antibodies** (Ab) as a response to exogenous proteins like bacteria, viruses and tumor cells. Each B cell is programmed to produce a specific antibody. The antibodies are specific proteins that recognize and bind to another particular protein. The production and binding of antibodies is usually a way of signaling other cells to kill, ingest or remove the bound substance. They are also responsible for persisting the **memory** of recent infections.

T cells and lymphokines

The T cells are so called because they mature within the thymus (Dreher, 1995). Their functions include the regulation of other cells' actions and directly **attack** the host-infected cells. The T lymphocytes can be subdivided into three major subclasses: T helper cells (Th), cytotoxic (killer) T cells and suppressor T cells. The T helper cells, or simply Th cells, are essential to the activation of the B cells, other T cells, macrophages and natural killer (NK) cells. They are also known as CD4 or T4 cells. The killer T cells, or cytotoxic T cells, are capable of eliminating microbial invaders, viruses or cancerous cells. Once activated and bound to their ligands, they inject noxious chemicals into the other cells, perforating their surface membrane and causing their destruction. The suppressor T lymphocytes are vital for the maintenance of the immune response. They are sometimes called CD8 cells, and inhibit the action of other immune cells. Without their activity, immunity would certainly loose control resulting in allergic reactions and autoimmune diseases (Janeway Jr. & Travers, 1997).

[reference]

Immune System principles

- Immune network theory (Network of B cells)
- Negative Selection (Creation of detector set)
- Clonal Selection theory (Cloning of fit population)

Immune Network Theory

The immune Network theory had been proposed in the mid-seventies (Jerne 1974). The hypothesis was that the immune system maintains an idiotypic network of interconnected B cells for antigen recognition. These cells both stimulate and suppress each other in certain ways that lead to the stabilization of the network. Two B cells are connected if the affinities they share exceed a certain threshold, and the strength of the connection is directly proportional to the affinity they share.

Negative Selection mechanism

The purpose of negative selection is to provide tolerance for self cells. It deals with the immune system's ability to detect unknown antigens while not reacting to the self cells. During the generation of T-cells, receptors are made through a pseudo-random genetic rearrangement process. Then, they undergo a censoring process in the thymus, called the negative selection. There, T-cells that react against self-proteins are destroyed; thus, only those that do not bind to self-proteins are allowed to leave the thymus. These matured T-cells then circulate throughout the body to perform immunological functions and protect the body against foreign antigens.

Clonal Selection Principle

The clonal selection principle describes the basic features of an immune response to an antigenic stimulus. It establishes the idea that only those cells that recognize the antigen proliferate, thus being selected against those that do not. The main features of the clonal selection theory are that:

- The new cells are copies of their parents (clone) subjected to a mutation mechanism with high rates (somatic hypermutation);
- Elimination of newly differentiated lymphocytes carrying self-reactive receptors;
- Proliferation and differentiation on contact of mature cells with antigens.

When an antibody strongly matches an antigen the corresponding B-cell is stimulated to produce clones of itself that then produce more antibodies. This (hyper) mutation, is quite rapid, often as much as "one mutation per cell division" (de Castro and Von Zuben, 1999). This allows a very quick response to the antigens. It should be noted here that in the Artificial Immune Systems literature, often no distinction is made between B-cells and the antibodies they produce. Both are subsumed under the word 'antibody' and statements such as mutation of antibodies (rather than mutation of B- cells) are common.

There are many more features of the immune system, including adaptation, immunological memory and protection against auto-immune attacks, not discussed here. Some of the application of immune systems include Intrusion Detection Systems and Data mining problems - Collaborative filtering and clustering

V. GENETIC ALGORITHMS

Genetic algorithm is a family of computational models based on principles of evolution and natural selection. These algorithms convert the problem in a specific domain into a model by using a chromosome-like data structure and evolve the chromosomes using selection, recombination, and mutation operators. In computer security applications, it is mainly used for finding optimal solutions to a specific problem.

Genetic algorithm comprises of a population of strings (called chromosomes or the genotype of the genome), which encode candidate solutions (called individuals, creatures, or phenotypes).The process of a genetic algorithm usually begins with a randomly selected population of chromosomes. These chromosomes are representations of the problem to be solved. According to the attributes of the problem, different positions of each chromosome are encoded as bits, characters, or numbers. These positions are sometimes referred to as genes and are changed randomly within a range during evolution. The set of chromosomes during a stage of evolution are called a population. An evaluation function is used to calculate the "goodness" of each chromosome. During evaluation, two basic operators, crossover and mutation, are used to simulate the natural reproduction and mutation of species. The selection of chromosomes for survival and combination is biased towards the fittest chromosomes. Finally, the best individual (chromosome) is picked out as the final result once the optimization criterion is met.

Genetic algorithms are used to solve various optimization problems. Following are few of the applications of GA to various computer science problems:

- Intrusion Detection systems: The mechanism of detection intrusions in a network setup is envisioned as an evolutionary problem and genetic algorithms are used as a way to generate dynamic rule sets to identify intrusions

ISSN: 2454-4248
31 – 35

- Search Optimization and as an optimisation mechanism in travelling salesmen problem.
- In Robotics, GAs can be programmed to search for a range of optimal designs and components for each specific use, or to return results for entirely new types of robots that can perform multiple tasks and have more general application.
- Other applications can be found in the field of control systems, engineering design, strategy development in many finance, marketing and security industry

VI. CONCLUSIONS

In this series of articles we will try and understand the nature of these algorithms and their close relationship with biological processes. We've divided these bio inspired computational equivalents into two, (i) Swarm Intelligence and (ii) Natural Computing.

REFERENCES

[1] Fogel, David B. "Evolutionary algorithms in theory and practice." *Complexity* 2.4 (1997): 26-27.

[2] Holland, John H. "Genetic algorithms and the optimal allocation of trials." *SIAM Journal on Computing* 2.2 (1973): 88-105.

[3] Olariu, Stephan, and Albert Y. Zomaya. *Handbook of bioinspired algorithms and applications*. Chapman and Hall/CRC, 2005.

[4] Mahale, Rashmi A., and S. D. Chavan. "A survey: evolutionary and swarm based bio-inspired optimization algorithms." *International Journal of Scientific and Research Publications*2.12 (2012): 1-6.

[5] Binitha, S., and S. Siva Sathya. "A survey of bio inspired optimization algorithms." *International Journal of Soft Computing and Engineering* 2.2 (2012): 137-151.

[6] Fister Jr, Iztok, et al. "A brief review of nature-inspired algorithms for optimization." *arXiv preprint arXiv:1307.4186*(2013).

[7] Kar, Arpan Kumar. "Bio inspired computing–A review of algorithms and scope of applications." *Expert Systems with Applications* 59 (2016): 20-32.

[8] Zang, Hongnian, Shujun Zhang, and Kevin Hapeshi. "A review of nature-inspired algorithms." *Journal of Bionic Engineering*7.4 (2010): S232-S237.