

Swarm Intelligence Techniques as Efficient Tools for Problem Solving

Manjula. KA

*Research Scholar,
Research & Development Centre,
Bharathiar University, Tamilnadu, India.*

Abstract - Swarm intelligence is the study of computational systems inspired by the 'collective intelligence' Which is emerged through the cooperation of large numbers of homogeneous agents in the environment like schools of fish, flocks of birds, and colonies of ants. Such intelligence is decentralized, self-organizing and distributed throughout an environment and this has led to the development of a new computing methodology in solving complex problems and finding optimum solutions in many areas including the research areas in information technology.

Keywords – natural computing, swarm intelligence, ant colony optimization, particle swarm optimization, bee colony optimization.

1. INTRODUCTION

Nature does things in an astonishing way and the information processing in nature is performed in a distributed, self-organized and optimal manner without any central control. Nature can provide very good examples to solve problems in efficient and effective manner and it has always been a source of inspiration and stimulating research on new computing paradigms. Computation methods for human problem solving may be designed by applying knowledge from natural systems and this idea has led to the emergence of new fields like Nature Inspired Computing. The researchers have been trying to devise computational methods that can help human in solving complex problems for the past few decades. Nature inspired computing techniques such as swarm intelligence, genetic algorithm, artificial neural network, DNA computing, membrane computing and artificial immune system have helped in solving complex problems and providing optimum solutions. Parallel, dynamic, decentralized, asynchronous and self organizing behaviour of nature inspired algorithms are best suited for soft computing applications.

The swarm intelligence techniques - Ant colony optimization, particle optimization and Bee colony optimization - are discussed in this paper. A few important applications of swarm intelligence in information technology are discussed in the last section of this paper.

II. SWARM INTELLIGENCE TECHNIQUES

Nature inspired techniques can be examined from two perspectives. First, as a way to understand and characterize the underlying mechanism(s) of complex real-world phenomena or systems behaviour by formulating

computing models and testing hypotheses through controlled experimentation. The end product is at least a deep understanding of the working mechanism(s) of the modeled system. Second, as a way to design and develop computing solutions to solve complex problems by reproducing life like behavior in solving computing problems [1]. Nature inspired computing draws on the principles of emergence, self-organization and complex systems. It aims to develop new techniques, algorithms and computational applications by observing how nature behaves in those situations of solving complex problems. Projects based on the nature inspired computing create and apply algorithms based on natural phenomena such as functioning of the human brain, darwinian evolution and swarms of insects. These algorithms can be applied to a wide variety of problems including problems in business management, robotics, engineering and bio-informatics. The complex and often coordinated behavior of swarms fascinate not only biologists but also experts from other fields like computer science and management. Bird flocking and fish schooling are impressive examples of coordinated behavior that emerges without central control. Social insect colonies show complex problem solving skills resulted from the actions and interactions of non sophisticated individuals.

Swarm intelligence, one of the nature inspired computing technique, is the study of computational systems inspired by the 'collective intelligence'. Collective Intelligence emerges through the cooperation of large numbers of homogeneous agents in the environment. Examples include schools of fish, flocks of birds, and colonies of ants. Such intelligence is decentralized, self-organizing and distributed throughout an environment. In nature, such systems are commonly used to solve problems such as effective foraging for food, prey evading, or colony re-location. The information is typically stored throughout the participating homogeneous agents, or is stored or communicated in the environment itself such as through the use of pheromones in ants, dancing in bees, and proximity in fish and birds.

The swarm intelligence paradigm consists of different dominant sub-fields of which three are discussed here 1) *Ant Colony Optimization* that investigates probabilistic algorithms inspired by the stigmergy and foraging behavior of ants, and 2) *Particle Swarm Optimization* that investigates probabilistic algorithms inspired by the flocking, schooling and herding. 3) *Bee Colony*

Optimization that investigates probabilistic algorithms inspired by the collective intelligence applied by the honey bees during nectar collecting process.

Like evolutionary computation, swarm intelligence algorithms are considered as adaptive strategies and are typically applied to search and optimization domains.

A. Ant Colony Optimization

The attempt to develop algorithms inspired by one aspect of ant behavior, the ability to find what computer scientists would call shortest paths, has become the field of ant colony optimization (ACO), the most successful and widely recognized algorithmic technique based on ant behavior. Introduced by Marco Dorigo in 1992, the Ant Colony Optimization field has experienced a tremendous growth. It stands today as an important nature-inspired computing technique which works very well in graphs with changing topologies and the examples of such systems include computer networks and artificial intelligence. This meta-heuristic approach is suitable for solving hard combinatorial optimization problems.

This behavior of real ants is the inspiring source of ACO. At first, the ants wander randomly. When an ant finds a source of food, it walks back to the colony leaving pheromones that show the path leading to food. When other ants come across these markers, they are likely to follow this path with a certain probability. If they do, they then populate this path with their own markers as they bring the food back. As more ants follow the same path, it gets stronger until there are a couple streams of ants traveling to various food sources near the colony. Because the ants drop pheromones every time they bring food, shorter paths are more likely to be stronger, hence optimizing the solution. In the meantime, some ants may be still randomly searching for closer food sources. Once the food source is depleted, the route is no longer populated with pheromones and slowly decays.

ACO is based on the indirect communication of a colony of simple agents, called (artificial) ants, mediated by (artificial) pheromone trails. The pheromone trails in ACO serve as a distributed, numerical information which the ants use to probabilistically construct solutions to the problem being solved and which the ants adapt during the algorithm's execution to reflect their search experience [2]. Ant system was first applied to the travelling salesman problem(TSP) and it provided inspiration for the development of computational algorithms for the solution of difficult mathematical problems [3] – [5]. In the application of ACO in TSP, initially each ant is randomly put on a city. During the construction of a feasible solution, ants select the following city to be visited through a probabilistic decision rule. When an ant *k* visits city *i* and constructs the partial solution, the probability moving to the next city *j* neighboring on city *i* is given by the following equation.

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} [\tau_{iu}(t)]^\alpha [\eta_{iu}]^\beta} & \text{if } j \in J_k(i) \\ 0 & \text{otherwise} \end{cases}$$

where τ_{ij} is the intensity of trails between edge (i,j) and $\eta_{ij} = 1/d_{ij}$. $J_k(i)$ is a set of cities which remain to be visited when the ant is at city *i*. α and β are two adjustable positive parameters that control the relative weights of the pheromone trail and of the heuristic visibility. After each ant completes its tour, the pheromone amount on each path will be adjusted with the following equation.

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \Delta \tau_{ij}(t)$$

In this equation,

$$\Delta \tau_{ij}(t) = \sum_{k=1}^m \Delta \tau_{ij}^k(t)$$

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k} & , \text{if } (i, j) \in \text{tour done by ant } k \\ 0 & \text{otherwise} \end{cases}$$

(1- ρ) is the pheromone decay parameter ($0 < \rho < 1$) where it represents the trail evaporation when the ant chooses a city and decide to move. L_k is the length of the tour performed by ant *k*, *Q* is a constant and *m* is the number of ants.

Several variants of ant algorithms for optimizations have been introduced and applied in various domains [6] – [9].

B. Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling and modeled on swarm intelligence. The principle underlying this technique is that, 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/her 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/her. This tightening pattern continues until one of the birds happens upon the food. 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 is not 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 above, the data would be the X, Y, Z coordinates of each bird. In this case, the individuals furthest from the food would make an effort to keep up with the others by flying faster toward the gBest bird. If the

data is a pattern or sequence, the velocity would describe how different the pattern is from the target, and thus, how much it needs to be changed to match the target. Each particle's pBest value only indicates the closest the data has ever come to the target since the algorithm started. The gBest value only changes when any particle's pBest value comes closer to the target than gBest. Through each iteration of the algorithm, gBest gradually moves closer and closer to the target until one of the particles reaches the target. The pseudocode for PSO algorithm is given below.

```

For each particle do
{ Initialize particle }
Repeat until maximum iterations or minimum error
criteria
{
For each particle do
{ Calculate Data Fitness Value
If the Fitness Value is better than pBest
{ pBest ← current Fitness Value }
If pBest is better than gBest
{ gBest ← pBest }
}
For each particle do
{ Calculate particle Velocity
Use gBest and Velocity to update particle Data
}
}
}

```

The velocity and position update step is responsible for the optimization ability of the PSO algorithm and the velocity of each particle in the swarm is updated using the following equation[10].

$$v_i(t+1) = wv_i(t) + c_1r_1[x_i^l(t) - x_i(t)] + c_2r_2[g(t) - x_i(t)]$$

The index of the particle is represented by i . Thus, $v_i(t)$ is the velocity of particle i at time t and $x_i(t)$ is the position of particle i at time t . The parameters w , c_1 and c_2 ($0 \leq w \leq 1.2$, $0 \leq c_1 \leq 2$, and $0 \leq c_2 \leq 2$) are user-supplied coefficients. The values r_1 and r_2 ($0 \leq r_1 \leq 1$ and $0 \leq r_2 \leq 1$) are random values regenerated for each velocity update. The value $x_i^l(t)$ is the individual best candidate solution for particle i at time t , and $g(t)$ is the swarm's global best candidate solution at time t .

Once the velocity for each particle is calculated, each particle's position is updated by applying the new velocity to the particle's previous position:

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

This process is repeated until some stopping condition is met where common stopping conditions include, a preset number of iterations of the PSO algorithm, a number of iterations since the last update of the global best candidate solution, or a predefined target fitness value.

In past several years, PSO has been successfully applied in many research and application areas. PSO gets better results in a faster, cheaper way compared with other methods and also there are very few parameters to adjust.

C. Bee Colony Optimization

BCO is the name given to the collective food foraging behavior of honey bee and in this swarm intelligent system, the low level agent is the bee. The bee system is a standard example of organized team work, well coordinated interaction, coordination, labor division, simultaneous task performance, specialized individuals, and well-knit communication.

A colony of honey bees can exploit a large number of food sources in big fields and they can fly up to 11 km to exploit food sources. The colony employs about one-quarter of its members as forager bees. The foraging process begins with searching out promising flower patches by scout bees. The colony keeps a percentage of the scout bees during the harvesting season. When the scout bees have found a flower patch, they will look further in hope of finding an even better one. The scout bees search for the better patches randomly. The scout bees inform their peers waiting in the hive as to the quality of the food source, based amongst other things, on sugar levels. The scout bees deposit their nectar and go to the dance floor in front of the hive to communicate to the other bees by performing their dance, known as the 'waggle dance'.

The waggle dance is named based on the wagging run (in which the dancers produce a loud buzzing sound by moving their bodies from side to side), which is used by the scout bees to communicate information about the food source to the rest of the colony. The scout bees provide the following information by means of the waggle dance: the quality of the food source, the distance of the source from the hive and the direction of the source [11] – [13].

The waggle dance path is shaped as that of figure eight. Initially the scout bee vibrates its wing muscles which produces a loud buzz and runs in a straight line the direction which is related to the vertical on the hive and indicates the direction of the food source relative to the sun's azimuth in the field. The scout then circles back, alternating a left and a right return path. The speed/duration of the dance indicates the distance to the food source; the frequency of the waggles in the dance and buzzing convey the quality of the source. This information will influence the number of follower bees [14].

Artificial Bee Colony optimization (ABC) was proposed by Karaboga[15]. The algorithm consists of the following bee groups: employed bees, onlooker bees and scout bees as in nature. Employed bees randomly explore and return to the hive with information about the landscape. This explorative search information is shared with onlooker bees. The onlooker bees evaluate this information with a probabilistic approach such as the roulette wheel method to start a neighborhood search. Meanwhile, the scout bees perform a random search to carry out the exploitation.

Pham et. al. proposed another algorithm named as 'Bees Algorithm' which is very similar to the ABC in the sense of having local search and global search processes [16]. However there is a difference between both algorithms during the neighborhood search process. As mentioned above, ABC has a probabilistic approach during the neighborhood stage; however the Bees Algorithm does not use any probability approach, but instead uses fitness

evaluation to drive the search. The Bees Algorithm has both local and global search capability utilizing exploitation and exploration strategies, respectively. The pseudo-code of the algorithm is given below.

C. Pseudo-code of the basic Bees Algorithm

Generate the initial population size as n , set the best patch size as m , set the elite patch size as e , set the number of forager bees recruited to the of elite sites as nep , set the number of forager bees around the non-elite best patches as nsp , set the neighborhood size as ngh , set the maximum iteration number as $MaxIter$, and set the error limit as $Error$.

$i = 0$

Generate initial population.

Evaluate Fitness Value of initial population.

Sort the initial population based on the fitness result.

While $i \leq MaxIter$ or $FitnessValue_i -$

$FitnessValue_{i-1} \leq Error$

1. $i = i + 1$;

2. Select the elite patches and non-elite best patches for neighborhood search.

3. Recruit the forager bees to the elite patches and non-elite best patches.

4. Evaluate the fitness value of each patch.

5. Sort the results based on their fitness.

6. Allocate the rest of the bees for global search to the non-best locations.

7. Evaluate the fitness value of non-best patches.

8. Sort the overall results based on their fitness.

9. Run the algorithm until termination criteria met.

End

III. APPLICATIONS OF SWARM INTELLIGENCE IN INFORMATION TECHNOLOGY

There are several areas in information technology where swarm intelligence can be applied for problem solving. A few applications are listed below.

1. Scheduling: Scheduling plays a significant role in computing since various resources are to be allocated to various tasks in an optimized way. For example, virtual computation elements such as threads, processes or data flows may need to be scheduled onto hardware resources such as processors, network links or expansion cards. There are several algorithms based on swarm intelligence to manage scheduling effectively and these are highly relevant in grid computing environments.

2. Clustering: Clustering is the unsupervised classification of patterns such as observations, data items or feature vectors into groups known as clusters. Each cluster is a collection of agents which are similar and are dissimilar to the agents in other clusters. The clustering problem has been addressed in many contexts in computer science such as datamining, image processing etc. This capability of clustering is exhibited in the formation of cluster of corpses to clean up the ants' nests and algorithms based on this are used in addressing clustering problems in computer science.

3. Routing: Routing is the process of forwarding of a packet in a network so that it reaches its intended destination. There are several routing algorithms based on different principles aiming for optimization. There are optimized algorithms which are designed based on ant colony optimization. This is making use of the principle that backward ants utilize the useful information gathered by the forward ants on their trip from source to destination during their food foraging. The routing algorithm for MANETs is designed on the basis of the capability of swarm intelligence of ants and it is named as AntHocNet.

4. Image edge detection: Edge detection is an image processing technique used for finding the boundaries of objects within images by detecting the discontinuities in brightness. Edge detection is useful in image segmentation and image data extraction. For designing optimized edge detection algorithms, swarm intelligence techniques like ant colony optimization and particle swarm optimization are used.

5. Swarm robotics : The application of swarm principles to robots is called swarm robotics and in swarm robotics, a large number of robots are coordinated in a distributed and decentralized way. Large number of simple robots can perform complex tasks in a more efficient way than a single robot, giving robustness and flexibility to the group as that of the collective behaviour of social insects. The knowledge of the swarm is distributed throughout all the agents, where an individual is not able to accomplish its task without the rest of the swarm.

IV. CONCLUSIONS

Swarm intelligence is the discipline of Computational Intelligence inspired by the behavior analysis of groups of animals. This is a group of bio-inspired algorithms which is modeled on the collective behaviour of a group of social organisms in nature such as ants, termites, bees, fishes and birds. The striking characteristic of swarm intelligence is that it emerges from many simple interactions between individuals which show no level of intelligence at all as discussed in the three optimization techniques based on swarm intelligence described here - ant colony optimization, particle swarm optimization and bee colony optimization. Swarm intelligence has been a source of successful models for the solution of optimization problems in many fields of science, engineering and management. Several research and application areas of information technology find swarm intelligence based optimization techniques highly relevant and it is expected that swarm intelligence will definitely play a significant role in the future development of computing technology.

REFERENCES

- [1]. Batouche, Mohamed, and Souham Meshoul. "Nature Inspired intelligent techniques for problem solving." (2010).
- [2]. Dorigo, Marco, and Thomas Stützle. "The ant colony optimization metaheuristic: Algorithms, applications, and advances." In *Handbook of metaheuristics*, pp. 250-285. Springer US, 2003.
- [3]. Dorigo, Marco, and Luca Maria Gambardella. "Ant colony system: a cooperative learning approach to the traveling salesman problem." *IEEE Transactions on evolutionary computation* 1, no. 1 (1997): 53-66.

- [4]. Dorigo, Marco, Vittorio Maniezzo, and Alberto Colomi. "Ant system: optimization by a colony of cooperating agents." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 26, no. 1 (1996): 29-41.
- [5]. Dorigo, Marco, Mauro Birattari, and Thomas Stutzle. "Ant colony optimization." *IEEE computational intelligence magazine* 1, no. 4 (2006): 28-39.
- [6]. Hlaing, Zar Chi Su Su, and May Aye Khine. "An ant colony optimization algorithm for solving traveling salesman problem." In *International Conference on Information Communication and Management*, vol. 16, pp. 54-59. 2011.
- [7]. J. Tian, W. Yu, and S. Xie, "An Ant Colony Optimization Algorithm For Image Edge Detection", IEEE Congress on Evolutionary Computation, 1-6, pp. 751-756, June. 2008.
- [8]. Li, X. Y., Peng Tian, and Stephen CH Leung. "An ant colony optimization metaheuristic hybridized with tabu search for open vehicle routing problems." *Journal of the Operational Research Society* 60, no. 7 (2009): 1012-1025.
- [9]. Parpinelli, Rafael S., Heitor S. Lopes, and Alex Alves Freitas. "Data mining with an ant colony optimization algorithm." *IEEE Transactions on evolutionary computation* 6, no. 4 (2002): 321-332.
- [10] Blondin, James. "Particle swarm optimization: A tutorial." Available from: http://cs.armstrong.edu/saad/csci8100/pso_tutorial.pdf (2009).
- [11] Seeley, Thomas D. *The wisdom of the hive: the social physiology of honey bee colonies*. Harvard University Press, 2009.
- [12] Gould, James L., and Carol Grant Gould. *The honey bee*. Scientific American Library, 1988.
- [13] Frisch, Karl von. *Bees-Their Vision, Chemical, Senses And Language*. Cornell University Press; New York, 1950.
- [14]. Talbi, El-Ghazali. *Metaheuristics: from design to implementation*. Vol. 74. John Wiley & Sons, 2009.
- [15] Karaboga, Dervis. *An idea based on honey bee swarm for numerical optimization*. Vol. 200. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.
- [16]. Pham, D. T., A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi. "The bees algorithm-A novel tool for complex optimisation." In *Intelligent Production Machines and Systems-2nd I* PROMS Virtual International Conference (3-14 July 2006)*. sn, 2011.