Swarm Intelligence

Nisheeth Lahoti – 110050027
Mohit Gupta – 110050085
Mehul Goyal – 110050017
Deepali Adlakha – 11D170020

What is Swarm Intelligence?

The emergent collective intelligence of groups of simple agents

Examples

- Group foraging of social insects
- Co-operative transportation
- Division of labor
- Nest-building of social insects

Characteristics of Swarms

- Composed of many individuals
- Individuals are homogeneous
- Local interaction based on simple rules
- Self Organization
- Coordinating without the presence of an external controller

Motivation

- Inspired by nature
- Scalability
  - Individual's behavior is loosely influenced by swarm dimension
- Parallel Action
  - Can simultaneously take care of different aspects of a complex task
- Fault Tolerance
  - Decentralized; Failing individual can be dismissed and substituted

Applications of Swarm Intelligence

- Cooperative behavior in Swarms of Robots
- Demand for Miniaturization(nanorobotics) like distributed sensing tasks in micro machinery
- Swarm based Network Management
  - Routing and Load balancing in circuit switched networks

Particle Swarm optimization

- Simple software agents called particles
- Position of a particle represents a candidate solution
- Particle searches for better positions in search space
- Velocity is updated according to rules inspired by behavioral models of bird flocking

Particle Swarm optimization

Velocity Update Rule:

$$\vec{v}_i^{\,t+1} = w\vec{v}_i^{\,t} + \varphi_1 \vec{U}_1^{\,t}(\vec{b}_i^{\,t} - \vec{x}_i^{\,t}) + \varphi_2 \vec{U}_2^{\,t}(\vec{l}_i^{\,t} - \vec{x}_i^{\,t}),$$

- First Term – Inertia component
- Second Term – Cognitive component
- Third Term – Social component

Position Update Rule:

$$\vec{x}_i^{\,t+1} = \vec{x}_i^{\,t} + \vec{v}_i^{\,t+1}$$

Multi-Swarm optimization

- Variant of Particle Swarm Optimization
- Use of multiple sub swarms
- Each sub swarm focuses on a specific region
- Specific diversification methods on where and when to launch the sub swarms
- Fitted for the optimization on multi modal problems, where multiple (local) optima exist

Bee Colony Optimization

- Employed Bees
  - Associated with specific food sources
- Onlooker Bees
  - Watch the dance of employed bees within the hive to choose a food source
- Scout Bees
  - Search for food sources randomly

Bee Colony Meta-Heuristic

- Food source represents a possible solution and nectar amount of source corresponds to fitness of the solution
- Food source positions are discovered by scout bees
- Nectar of food sources are continually exploited by employed bees and onlooker bees which leads to their exhaustion
- The employed bee whose food source has been exhausted becomes a scout bee

Ant Colony Optimization

- It is a population-based metaheuristic to find approximate solutions to difficult optimization problems
- Software agents called ants
- Ants incrementally build solutions by moving on the graph
- Pheromone model, whose parameters are modified at runtime

Ant Colony Optimization for TSP

<u>Travelling Salesman Problem</u>

The TSP is the problem of finding a minimal length Hamiltonian circuit of the graph, where a Hamiltonian circuit is a closed tour visiting exactly once each of the n = |N| nodes of G.

Ant System (ACO Algorithm)

Tour Construction:

The probability with which ant k, currently at city i, chooses to go to city j at the t^{th} iteration of the algorithm is:

$$p(c_{ij}|s_k^p) = \begin{cases} \dfrac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{c_{il} \in N(s_k^p)} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}} & \text{if } j \in N(s_k^p), \\ 0 & \text{otherwise,} \end{cases}$$

Ant System (ACO Algorithm)

Pheromone update:

Lowering the pheromone strength on all arcs by a constant factor and then allowing each ant to add pheromone on the arcs it has visited

$$\tau_{ij} \leftarrow (1-\rho) \cdot \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \quad ,$$

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{1}{L_k} & \text{if}\ \text{ant}\ k\ \text{used}\ \text{edge}\ (i,j)\ \text{in}\ \text{its}\ \text{tour}, \\ 0 & \text{otherwise}, \end{cases}$$