Social Insect Inspired AI
Ant Algorithms

CSc355

Alan Dix
dix@comp.lancs.ac.uk
Manolis Sifalakis
mjs@comp.lancs.ac.uk

Lecture Overview

• What is Ant Algorithms?
• A brief history – timeline
• Self Organising systems
• Deneubourg’s simple experiment
• Computational Model
  • Sample iteration of the algorithm
• Application: The Travelling Salesman Problem
• Other applications: Network Routing
• Reference List

What is Ant Algorithms

• Ant optimisation algorithms [Dorigo 1996] are multi-agent systems, which consist of agents with the collective behavior (stigmergy) of ants for finding shortest paths
  • Alternative to applying complex algorithms to static datasets
  • A set of artificial ants implement a simple algorithm collectively to solve a combinatorial problem by a cooperative effort
  • Originated from Aife research

A Brief History - Timeline

Self Organising systems: Bénard cellular convection

• Layer of fluid heated from below
• In right conditions, perfect temperature gradient is formed vertically across the fluid
• System becomes ‘top heavy’, warmer (light) molecules at the bottom rise to the top where they cool (heavy) and uniformly flow to the bottom again
• The liquid does not simply bubble away without a pattern or organisation... but instead...
  • an ordered system is formed with millions of molecules self-organising in a hexagonal pattern
• Most efficient convection for energy-dispersion.

Deneubourg’s Simple Experiment
Terminology Index

- Node: a vertex
- Edge: a line connecting 2 vertices
- Graph: Diagram connecting nodes with edges
- Weighted graph: Edges have weights
- Path: Sequence of nodes-edges from between two nodes
- Hamiltonian path: A path without any node revisited
- Sigma: a sum of terms
- Delta: a difference between terms

Computational Model

- The environment
  - Weighted graph representing distances between nodes
- The ants
  - Init population randomly distributed
  - Travel across the graph
  - Ordered tabu list of visited nodes
  - Follow a Hamiltonian path

Sample Iteration

Application: Travelling Salesman Problem

- A set of cities. A salesman needs to travel through all the cities following an optimal route (Hamiltonian path) that minimises the distance travelled.
  - First studied in 1930s
  - NP-hard: Not been found an algorithm that solves the general problem (for any number of cities at any arrangement), in polynomial time.
  - (Sub-)optimal solutions are possible for specific instances of the problem

TSP: The (pseudo-) code

```c
Main ()
{
  Initialise ()
  while time < MAX_TIME
  {
    SimulateAntsOnce ()
    UpdateTrail ()
    time++
  }
}
```
TSP: The (pseudo-) code

```plaintext
SimulateAntsOnce ( )
foreach ant
    ant.tabu_list += current_city
    next_city := MoveAntToNextCity (ant)
    ant.tour_length += distance [current_city] [next_city]
    current_city = next_city

UpdateTrail ( )
foreach city1 city2 pair
    pheromone [city1] [city2] = EvaporatePheromone ( )
foreach ant
    foreach city1 city2 in ant.tabu_list
        pheromone [city1] [city2] += IncreasePheromone ( )
```

TSP: Sample run (30 cities)

TSP: Sample run (50 cities)

TSP: Parameter tuning

- Alpha ($\alpha$), Beta ($\beta$)
  - $\alpha$: pheromone level on the path
  - $\beta$: distance across an edge
  - A number of combinations yield good solutions
- Rho ($\rho$)
  - $\rho$: concentration of pheromone on edge over time
  - $\rho > 0.5$ yields good solutions
- Number of ants
  - number of ants = number of cities, yield the best solutions

Other Applications: Net Routing (AntNet)

- Communications networks are unpredictable.
  - Sudden interest in a particular web site or a local crisis will lead to surges of network activity
  - Efficient traffic re-routing needed, minimising delays and congestion through quieter network sections
- Congestion resembles food source depletion near an ant colony
  - Ants must search for new routes, dynamically updating the virtual pheromone trail between nodes
- [Di Caro, Dorigo] developed AntNet, an ant-based routing algorithm
  - Outperformed all other routing methods

AntNet Overview

- Internet Routing
  - Distributed activity of building routing tables in each network node
  - Tell incoming data packets which outgoing link to follow to continue their travel to destination with maximum performance
- AntNet Routing Tables

• AntNet Routing Tables
AntNet Algorithm

1. From every network node a mobile agent (ant $t$) is started every $\Delta t$ towards a destination node $d$ with probability $P_d$.
   - Ants travel like normal data packets to discover new low-cost paths.
   - Ants store tabu-list of visited nodes to avoid loops and record paths.

2. At node $k$ next hop node $n$ is selected from routing table with probability $P_{td} = \text{function}(P_{tn}, \text{queue length})$.

3. On arrival at node $d$ ant follows same route backwards.

4. At each node $k$ on reverse path arriving from neighbor $m$, updates the routing table $T_k$ & traffic table $M_k$ for entries related to destination $d$, if ant $t$ has better statistics to report than those in $M_k$:
   - $T_{k,n}$: Increment $P_{tn}$.
   - $M_{k,n}$: Decrement all other $P_{tn}$.
   - $M_k$: Update stats w.r.t. travel time from $k$ to $d$.

Reference List

- Seminal paper

- Ant algorithms and Internet routing

- Ant Algorithms and the Travelling Salesman Problem
  - An Introduction to Ant Colony Optimization http://www.irit.fr/~Monmarche/Papers/Book/Book.pdf

- Ant colony optimisation web archive
  - http://www.aco-metaheuristic.org/

- Book chapter on Ant algorithms