UCD CASL

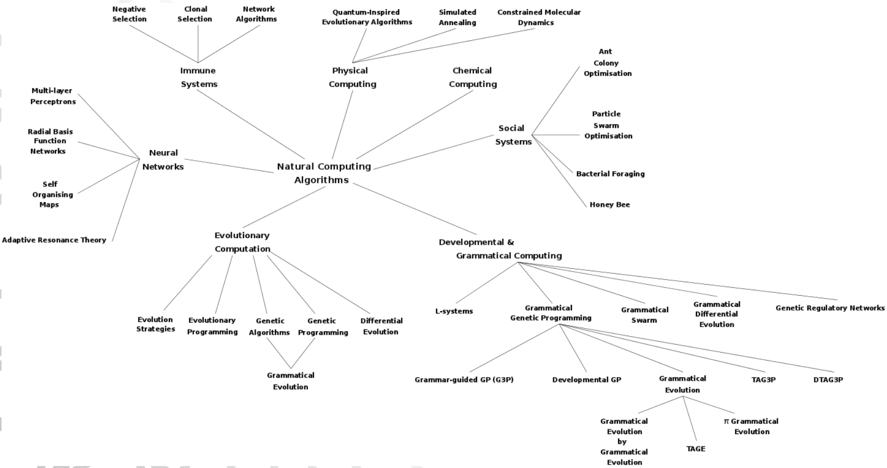
Complex & Adaptive Systems Laboratory

Dr. Michael O'Neill
Dr. Miguel Nicolau

Ant Colony Optimisation

COMP30290 Natural Computing
COMP41190 Natural Computing and Applications

Natural Computing Algorithms

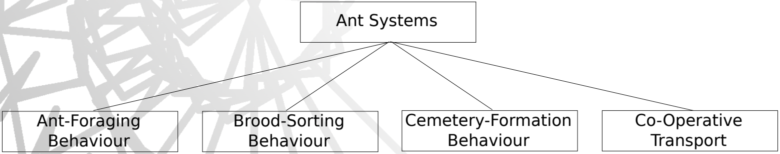


- Natural Computing Algorithms
  - Immune Systems
    - Negative Selection
    - Clonal Selection
    - Network Algorithms
  - Physical Computing
    - Quantum-Inspired Evolutionary Algorithms
    - Simulated Annealing
    - Constrained Molecular Dynamics
  - Chemical Computing
  - Social Systems
    - Ant Colony Optimisation
    - Particle Swarm Optimisation
    - Bacterial Foraging
    - Honey Bee
  - Neural Networks
    - Multi-layer Perceptrons
    - Radial Basis Function Networks
    - Self Organising Maps
    - Adaptive Resonance Theory
  - Evolutionary Computation
    - Evolution Strategies
    - Evolutionary Programming
    - Genetic Algorithms
    - Genetic Programming
      - Grammatical Evolution
    - Differential Evolution
  - Developmental & Grammatical Computing
    - L-systems
    - Grammatical Genetic Programming
      - Grammar-guided GP (G3P)
      - Developmental GP
      - Grammatical Evolution
        - Grammatical Evolution by Grammatical Evolution
        - TAGE
        - π Grammatical Evolution
      - TAG3P
      - DTAG3P
    - Grammatical Swarm
    - Grammatical Differential Evolution
    - Genetic Regulatory Networks

Social Algorithms

Inspiration

- PSO: School of fish / Flock of birds behaviour;
- Ant, Bee and Termite colonies;
- No Top-Down control;
- Interactions → Emergence.



Ant Systems

Ant-Foraging Behaviour

Brood-Sorting Behaviour

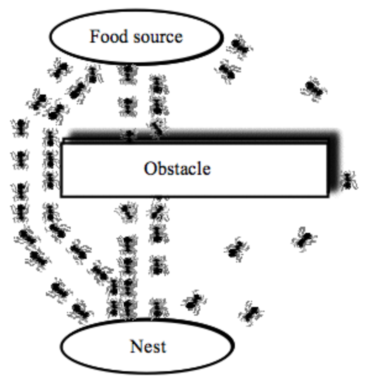
Cemetery-Formation Behaviour

Co-Operative Transport

Ant Colony Optimisation

Ant-Foraging Behaviour

- Ant Colonies:
  - Find shortest path between nest and food source.
- No memory;
- No cognitive maps;
- Colony builds map with pheromone trails;
- Stigmergy (indirect communication between agents through environment).



Ant Colony Optimisation

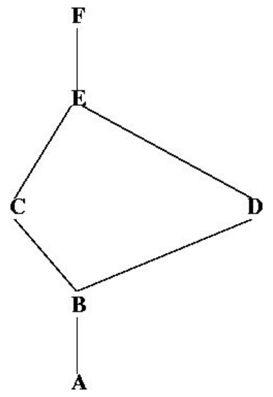
ACO

- Marco Dorigo (1991);
- Combinatorial Optimisation;
- TSP;
- Network Routing;
- Scheduling;
- Family of Algorithms:
    - Extensions of Ant System (AS).

Ant Colony Optimisation

Representation

- Problem-dependent;
- Generally graph-based;
- Ant System (AS).



Ant Colony Optimisation

Path Choice

- Transition probability!

$$\eta_{ij} = \frac{1}{d_{ij}}$$

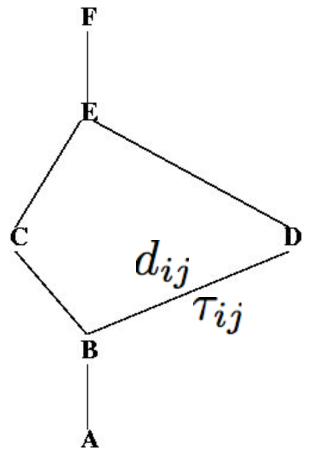
  - d_{ij} = distance from i to j.

$$p_{ij}^k(t) = \frac{\tau_{ij}(t)}{\sum \tau_{ik}(t)}$$

  - τ_{ij} = pheromone between i and j.

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha.[\eta_{ij}]^\beta}{\sum([\tau_{ik}(t)]^\alpha.[\eta_{ik}]^\beta)}$$

  - Parameterised rate of distance and pheromone.



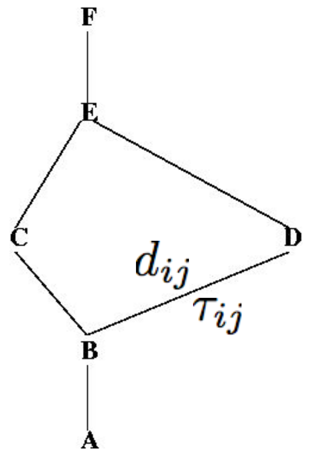
Ant Colony Optimisation

Pheromone

- Trail Intensity:

$$\tau_{ij}(t+1) = \tau_{ij}(t)(1-p) + d_{ij}$$

  - p = Evaporation Rate.



ACO - Algorithm

- 24,978 Cities;
- 72,500Km tour (2004).

SCIENCE

Santa Claus and the traveling salesman problem

Old Saint Nick has one night to deliver gifts to children around the world. To save time and wear and tear on his sleigh, Santa plots the shortest journey possible among thousands of cities. It sounds easy enough, and for Santa it probably is, but for scientists, who call this the traveling salesman problem (TSP), it is a challenge that grows increasingly difficult as more and more cities are added to the list.

Traveling salesman problem explained

30 Solution for Santa

Solving problem for more cities requires computers

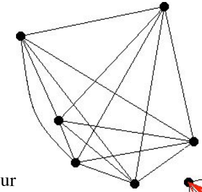
Years ago

Everyday uses for TSP

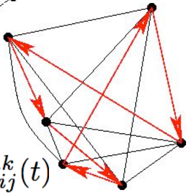
13,509 A record

The hardware involved

ACO - TSP Representation



- Construction Graph
- Each ant builds a tour
- Follows pheromones
- Update pheromone trail after tour



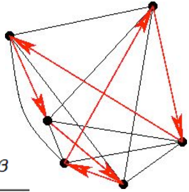
$$\tau_{ij}(t+1) = \tau_{ij}(t)(1-p) + \delta_{ij}$$

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

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \frac{Q}{L^k(t)}, & \text{if } (i,j) \in T^k(t); \\ 0, & \text{otherwise.} \end{cases}$$

ACO - TSP Representation...

- Tabu List



$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum[\tau_{ik}(t)]^{\alpha}.[\eta_{ik}]^{\beta}}$$

$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha}\cdot[\eta_{ij}(t)]^{\beta}}{\sum_{c\in C_i^k}[\tau_{ic}(t)]^{\alpha}\cdot[\eta_{ic}(t)]^{\beta}}, j \in C_i^k$$

ACO

[Mason Demo]

Ants for clustering

- Brood Sorting
  - Leptothorax unifasciatus
- Cemetery Formation
  - Lasius niger

- Ants on x,y grid
  - up, down, left, right

$$P_{pick} = \left(\frac{k_1}{k_1+f}\right)^2$$

$$P_{drop} = \left(\frac{f}{k_2+f}\right)^2$$

Ants for clustering...

- pick_drop (Deneubourg Model)
- + Dissimilarity (Lumar & Faieta Model)

$$f(o_i) = max\left\{0, \frac{1}{s^2}\sum_{o_j \in Neigh_{(s*s)}(r)}\left[1 - \frac{d(o_i, o_j)}{\alpha}\right]\right\}$$

$$P_{pick}(o_i) = \left(\frac{k_1}{k_1 + f(o_i)}\right)^2$$

$$P_{drop}(o_i) = 2f(o_i), \text{ if } f(o_i) < k_2$$
$$P_{drop}(o_i) = 1, \text{ if } f(o_i) \geq k_2$$

Project Ideas

ACO

- Parameter study;
- Variations on Pheromone Update Equation;
- Applications:
    - Clustering;
    - Quadratic Assignment;
    - Vehicle Routing;
    - Network Routing;
    - Graph Colouring;
    - Knapsack;
    - . . .
- Global Heuristics.

Project Proposal

Checklist

- Run proposal idea past Mike and/or Miguel;
- Use template on website (max 2 pages);
- Submit printout to the School of CSI office;
- Deadline 3pm next Thursday 10th October;

Next Classes

- Tuesday 8th October - Project Clinic
- Thursday 10th October - Submission Deadline / No Lecture
- Tuesday 15th October - Natural Computing & Creativity (Jonathan Byrne)
- Thursday 17th October - Individual Proposal Feedback with Mike & Miguel (3-5pm)