Ant Algorithms

Ant Algorithms, (ACO: Ant Colony Optimization) is a meta heuristics for solving Combinatorial optimization Problems (hard ones)

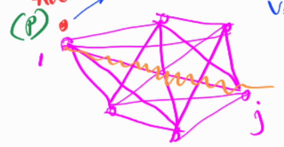
"Double Bridge" Experiment.

(How Ants control where to go?)

→ Initialization: (need an admissable solution)

The Pheromone values are initialized to $\tau_{ij}(0)$, which maybe chosen to the user's discretion.

Ant P has already visited the nodes $N_p(t)$ (at time t)
(It is not allowed to visit same node again)



$d_{ij} = d_{ji}$
= dist/Pheromone value.
(Phero value for each edge)

$\beta \rightarrow$ Prefer smaller edges

$$p^{prob\,(P)}_{ij} = \begin{cases} \dfrac{(\tau_{ij})^{\alpha} \cdot (\eta_{ij})^{\beta}}{\sum_{s \notin N_p(t)} (\tau_{is})^{\alpha} \cdot (\eta_{ij})^{\beta}} & \text{if } j \notin N_p(t) \quad \text{where } \alpha : [0,1] \\ 0 & \text{otherwise} \end{cases}$$

η_{ij} is called Visibility, or heuristic info.
(long edge is less visible, prefer short edges)

Here, we could use $\eta_{ij} = \dfrac{1}{d_{ij}}$

- Generate Solution candidates by ants, starting for ant P, with $N_p(0) = \emptyset$ (or $N_p(0) = \{\text{starting position}\}$)

At time $t+1$:
Ant P at position i goes to $j \notin N_p(t)$ with prob $prob^{(P)}_{(ij)}$

$$N_p(t+1) = N_p(t) \cup \{j\}$$

- α, β control the influence of Pheromone values and heuristic Info.

* Update Pheromone:

Fixed number Param.

$\{ \; Q_1$ if ant P goes from i to j between time $(t, t+1)$

- **Ant Density**: $g_{ij}^{(P)}(t, t+1) = \begin{cases} \ldots \\ 0 & \text{otherwise} \end{cases}$

↳ Influence on edge pheromones (independent of length)

spread out over whole edge.

- **Ant Quantity**: $g_{ij}^{(P)}(t, t+1) = \begin{cases} \overbrace{\dfrac{Q_2}{dist(ij)}} & \text{if ant P goes} \ldots \\ 0 \end{cases}$

- **Ant Cycle**: $g_{ij}^{(P)}(t, t+1) = \begin{cases} \dfrac{Q_3}{L^{(P)}} & \text{if antP goes} \ldots \\ 0 \end{cases}$

$L^{(P)}$ is length of ant P's best solution found so far.

→ Pheromone update now,

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

parameter ρ denotes the decay, or evaporation

$$\Delta\tau_{ij}(t, t+1) = \sum_{\forall p} g_{ij}^{(P)}(t, t+1)$$

Goal: To find Hamiltonian Path at back at So. (Travel Salesman Problem) decay of already seen pheromone value

→ Cannot find soln after one walk through Graph and returning & So need → Ant Cycle → after completion, update Phero values, and restart Cycle while remembering old solution

(Ant always reach So after N steps where N = no of nodes)

→ Solves Min Spanning Tree prob, without being aware of it, if we come close (in running time) to special MST problem, then the solution is more general than the algo that is not even invented.