"Ant Colony Optimization: The Traveling Salesman Problem"

Section 2.3 from *Swarm Intelligence: From Natural to Artificial Systems*
by Bonabeau, Dorigo, and Theraulaz

Traveling Salesman Problem (TSP)

- Goal is to find a closed tour of minimal length connecting n given cities.
- The problem can be though of as a graph, with each city as a node and the paths between them as edges.

Traveling Salesman Problem

- Ant colony optimization approach to TSP was initiated by Dorigo, Colorni, and Maniezzo
- The researchers chose the TSP for several reasons:
  - It is a shortest path problem to which the ant colony metaphor is easily adapted.
  - It is a very difficult (NP) problem
  - It has been studied a lot and therefore many sets of test problems are available, as well as many algorithms with which to run comparisons.
  - It is a didactic problem: it is very easy to understand and explanations of the algorithm behavior are not obscured by too many technicalities.

Ant System (AS)

- Ants build solutions to TSP by moving on the problem graph from one city to another until they complete a tour.
- During an iteration of the AS algorithm each ant builds a tour executing one step for each node (city).
- For each ant, transitions from one city to another depend on:
  - Whether or not the city has been visited
  - The heuristic desirability (“visibility”) of connected cities.
  - The amount of pheromone trail on the edge connecting two cities

Transition Rule

$$p_{ij}^{k}(t) = \frac{\left[\tau_{ij}(t)\right]^{\alpha} \cdot \left[\eta_{ij}\right]^{\beta}}{\sum_{l \in J_i^k} \left[\tau_{il}(t)\right]^{\alpha} \cdot \left[\eta_{il}\right]^{\beta}},$$

- k: ant, t: tour, i,j: neighboring cities
- J: list of cities still to be visited
- α, β: tuning parameters
- τ_{ij}: Pheromone strength between *i* and *j*
- η_{ij}: visibilty between *i* and *j*, defined as 1/distance$_{ij}$

Algorithm 2.1 High-level description of AS-TSP

/* **Initialization** */
For every edge (i, j) **do**
  $\tau_{ij}(0) = \tau_0$
End For
For $k = 1$ to m **do**
  Place ant k on a randomly chosen city
End For
Let T^+ be the shortest tour found from beginning and L^+ its length
/* **Main loop** */
For $t = 1$ to $t_{\max}$ **do**
  For $k = 1$ to m **do**
    Build tour $T^k(t)$ by applying $n - 1$ times the following step:
    Choose the next city j with probability

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta},$$

    where i is the current city

Pheromone Trail

- At the end of a tour, each ant lays pheromones on each edge it has used
- The amount of pheromone is proportional to the performance of the ant
- Pheromones intensity on each edge decays over time

End For
For $k = 1$ to m **do**
  Compute the length $L^k(t)$ of the tour $T^k(t)$ produced by ant k
End For
If an improved tour is found **then**
  update T^+ and L^+
End If
For every edge (i, j) **do**
  Update pheromone trails by applying the rule:
  $\tau_{ij}(t) \leftarrow (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) + e \cdot \Delta\tau_{ij}^e(t)$ where

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

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

Q: Tuning parameter. Value should be in vicinity of expected length of optimal solution. Use nearest neighbor heuristic or similar to determine its value.

Improvements: Elitist Ants

- “Elitist” ants introduced to improve performance
- Elitist ants reinforce the edges belonging to the best tour found from the beginning of the trial
- Elitist ants are added at every iteration to reinforce the best path

Elitist Ants

For every edge (i,j) **do**

Update pheromone trails by applying the rule:

$\tau_{ij}(t) \leftarrow (1-\rho)\cdot\tau_{ij}(t) + \Delta\tau_{ij}(t) + e\cdot\Delta\tau_{ij}^{e}(t)$ where

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

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

and

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

End For

End of Algorithm

For every edge (i,j) **do**

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

End For

End For

Print the shortest tour T^{+} and its length L^{+}

Stop

/* Values of parameters used in experiments */

$\alpha = 1, \beta = 5, \rho = 0.5, m = n, Q = 100, \tau_0 = 10^{-6}, e = 5$

AS Conclusions

- For small problems, AS performs comparably to other TSP algorithms
- Quickly converged to good solutions for larger problems, but could not find optimal solutions
- Performance level is much lower than specialized TSP algorithms

	Best tour	Average	Std. Dev.
AS-TSP	420	420.4	1.3
TS	420	420.6	1.5
SA	422	459.8	25.1

AS Conclusions

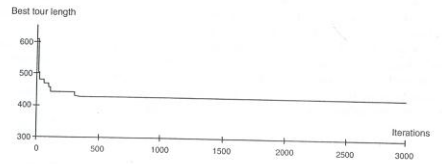

FIGURE 2.12 Evolution of best tour length (Test problem: Oliver30). Typical run. After Dorigo et al. [109]. Reprinted by permission © *IEEE Press*.

- Does not converge to a single optimal solution
- Continues to produce new, possibly improving, solutions
  - Avoids getting trapped in local optima
  - AS is promising for applications to dynamic problems