Department of Mathematics and Computer Science
University of Southern Denmark, Odense

October 27, 2006
Marco Chiarandini

DM63 - Heuristics for Combinatorial Optimization Problems – Lecture Notes

Lecture 10, Fall 2006

Lecture October 26

The Lecture focused entirely on Ant Colony Optimization.

Contrary to what announced there will not be a Task 4 in the Competition. Task 4 is substituted by Exercise 1 below. The analysis of results are left to the students after the Lecture on Experimental Analysis which will come the next week.

Bibliographical Notes

Ant Colony Optimization is described in the article 8 of the Notes.

Exercises

Exercise 1

The exercise consists of two parts. Some methods for Part 2 will be introduced in Lecture 12.

Part 1. Implement and configure one of the following population-based metaheuristics. There is freedom in the choice of components. Those in parenthesis are however some suggestions.

- Memetic Algorithm [MA]

  (*Initial solution*: a randomized form of Greedy Heuristic (Multiple Fragment Heuristic). Add $n/4$ ($n = |V|$) edges as follows: select a vertex v randomly and grow a partial tour of length $n/4$ by adding with probability $2/3$ the shortest edge and with probability $1/3$ the second shortest edge. Complete the construction with the Greedy Heuristic.

  Crossover: GX operator which generates one cross offspring in four steps:

  1. Copy edges that are common to the two parents to the offspring; the fraction of common edges to be copied is determined by a parameter $p_e (p_e = 1)$.
  2. Add new short edges that are not contained in any of the parents. For a vertex u_i one of the five nearest neighbors is chosen such that edge $u_i u_j$ is not contained in any of the parents and the addition of edge $u_i u_j$ is feasible; the fraction of the edges to be chosen in this way is determined by a third parameter, p_n.
  3. Copy edges from the parents where edges are ordered according to increasing length. Only edges that do not lead to a violation of the TSP constraints are considered add edges bit common to the parents may be included; the fraction of the edges added in this way is determined by a third parameter, p_c.
  4. If necessary, the candidate tour is completed using a randomized version of the Greedy Heuristic.

Selection for recombination: $n/2$ pairs at random.

Mutation: Double bridge

New Population: $(\mu + \lambda)$ method

Restart: if average bond distance between tours in the population falls below 10.

- Ant Colony Optimization [ACO]

  (In the $\mathcal{MAX} - \mathcal{MIN}$ *Ant System* version:

  Construction graph: corresponds to the original set of vertices, completely connected.

  Construction mechanism: starting from a randomly chosen vertex u_0 in each construction step, the current partial tour p is extended with a vertex u_j chosen probabilistically (transition rule) from a candidate list comprising the l nearest neighbors.

  Pheromone: indicates the desirability of going from one vertex to another hence assigned on the edges of the graph.

  Heuristic: also on the edges of the graph: $\eta_{ij} = \frac{C^{NN}}{nc_{ij}}$

  Pheromone update: only the best candidate solution found during the search updates the pheromone with $\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}^{best}$, $\Delta\tau_{ij}^{best} = C^{NN}/f(s^{best})$, where m is the number of ants.

  $\tau_0 = 1$, $\tau_{min} = 1$, $\tau_{max} = m$,

Part 2. Configure and tune the algorithm on the instances of Task 3. Compare then the obtained algorithm, thoroughly defined in the components and parameters, with the algorithms of Task 3 using the instances of Task 1. (The separation of instances into *training* instances, used for parameter tuning, and *test* instances, on which the algorithms are compared, is a procedure used to avoid over-fitting.)