NAT Tutorial 4: Genetic programming

1. A GP system is employed to evolve a controller for a mobile robot. The fitness function evaluates the robot performance starting from 50 initial positions. In a long series of tests the system is observed to produce a satisfactory controller in 70% of runs.
   The system designer decides to improve the GP system by speeding it up, and reduces the number of fitness cases to 25. The system now produces a satisfactory controller in only 50% of runs. Is this an improvement?
   Hint: Consider the amount of work the system must do to produce a satisfactory controller.

Answer: For a 70% chance we will need on average 10/7 trials, for 50% it will be 2, i.e. using the hint: P x 50 / 0.7 = P x R and P x 25 / 0.5, i.e. there will be about 70 x P vs. 50 x P trials necessary, such that the second option is preferable (P is the size of the population). Note that it may be risky to use too few test cases, but here the explanation says that the controller will be satisfactory. Note also that in numerical problem the test cases may be not very time consuming, but in a robot task almost all time is used in the hardware tests.

2. What is Meta-Genetic Programming?

Answer: see http://en.wikipedia.org/wiki/Genetic_programming#Meta-Genetic_Programming

3. Which of the existing implementations of GP would you recommend? Why?

Answer: Obvious advantages are if the system is free, versatile and possibly supports also related algorithms in order to combine the strength of a number of approaches. Since you may like to modify the system according to your own purposes it should be written in a language that you like and and run under the OS of your choice. Wikipedia (Genetic_programming) give a long list of Implementations. A classical choice might be beagle.sourceforge.net/ while recently implementation in Python seem to be preferred.

4. Consider the following (very small) TSP:
   $d(A,B) = 2$, $d(A,C) = 3$, $d(A,D) = 5$, $d(B,C) = 3$, $d(B,D) = 3$, $d(C,D) = 4$.
   What is the optimal (shortest) tour? How many different tours are possible? How many tours are possible with a TSP containing *N* cities? How long does it take at least to solve the above problem by an ant system? Assume a convergence probability of 50% and establish an (possibly trivial) upper bound based on the assumption of a minimal pheromone level at each link of the graph.

Answer: (4 choices for first city) x (3 for 2nd) x (2 for third) = 4! = 24
Circular permutations are the same so / by 4, and ABCD is the same as DCBA so / by 2. So 3 in total., i.e. N!/(2N) = (N-1)!/2 (assuming you go back to the starting city).
The three tours have lengths of 12 (ABDCA), 14 (ADCBA) and 14 (ADBCA). Here avoiding the longest link (AD) specifies already the optimum.
Use the Ant Colony Optimisation formula given below and a pheromone matrix initialised with tau(i,j) = 1.0 for all i\=j, (and 0.0 for i=j).

$$Pr(i,j) = \frac{tau(i,j).[eta(i,j)]\text{^}beta}{sum\{all\ legal\ j\}\ tau(i,j).[eta(i,j)]\text{^}beta} \quad \text{with } eta(i,j) = 1/d(i,j) \text{ and } beta = 2$$

Calculate the probabilities that an ant placed initially on city A will move to B, C or D.
tau(A,B) = tau(A,C) = tau(A,D) = 1.0
eta(A,B) = 1/2
eta(A,C) = 1/3
eta(A,D) = 1/5
sum = 361/900 = .401111

Pr(A,B) = 1.0*(1/2)^2/.40111 = .62327
Pr(A,C) = 1.0*(1/3)^2/.40111 = .27701
Pr(A,D) = 1.0*(1/5)^2/.40111 = .09972

Now use the following pheromone values and recalculate the probabilities for Pr(A,B), Pr(A,C) and Pr(A,D). What about Pr(B,A)?

tau(A,B)=4.0,	tau(A,C)=4.0,	tau(A,D)=0.2
	tau(B,C)=0.4,	tau(B,D)=2.0
		tau(C,D)=4.0

sum = 1.452
Pr(A,B) = 0.688
Pr(A,C) = 0.306
Pr(A,C) = 0.006

Pr(B,A) = 0.789 (sum = 1.26666)

Assume that A-B-D-C-A is the fittest of the current iteration, that the evaporation (\rho) is 0.75 and that the reinforcement value is 1.0. Update the values above using the pheromone update rule:

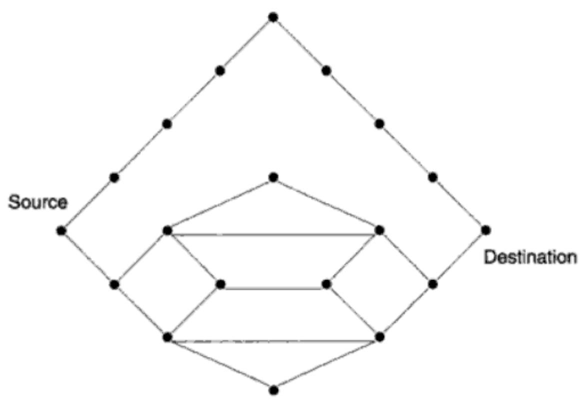
tau(i,j) = [tau(i,j) * rho] + delta(i,j)

where \delta(i,j) is 1.0 if i-j or j-i is a link in the best solution and 0.0 otherwise.

Shortest path is ABDC(A). So can update
tau(A,B) = 4.0*0.75 + 1.0 = 4.0
tau(B,D) = 2.0*0.75 + 1.0 = 2.5
tau(D,C) = tau(C,D) = 4.0*0.75 + 1.0 = 4.0
tau(C,A) = tau(A,C) = 4.0*0.75 + 1.0 = 4.0

Obviously if we'd had rho larger than 0.75 the pheromone values on the paths AB, DC, CA would have increased.

5. The figure on the right shows an example from the ACO book by Dorigo and Stuetzle. What results do you expect for an ant colony algorithm that does not use taboo lists (except for inhibition of immediate return to the previous node)?



Answer: Assume the visibility is given by the inverse of the lengths of the links in the picture. The densely connected path in the lower part of the graph will most likely lead to
cycles. Cycles can be discouraged by a high evaporation rate, but then the algorithm will not generate a memory of good solutions. So without a taboo list, cycles seem to be unavoidable. Some ants will travel the upper part which leads them straight to the goal such that in most of the runs the algorithm will end up with the suboptimal solution along the upper part (8 vs. 5 steps in the lower part). Because the problem is small there is still some chance to find the optimal solution. Conclusion: Taboo lists are useful.

6. Assume that ants are allowed to lay pheromone on a path at every time step, so that the pheromone update rule is applied at each time step. Come up with a combination local/global updating scheme that encourages exploration and exploitation– consider what parameters influence this.

Answer: The previous example shows that laying pheromones at each step can be problematic as it may lead to cycles. We should therefore use a taboo list. However, in a high-dimensional problem this might be not sufficient. If we assume that the ants perform a random walk initially which is know to have a low probability to return in dimensions higher than two.
Consider therefore an implementation of the taboo list in a soft way by negative pheromones: If the ant senses pheromones from the same cycle it is counted as reduction of the global desirability, while “old” pheromones remain attractive. Old and new pheromones need to be stored in separate matrices, and after each round the new pheromones are added to the old ones. In this way the ant spread out and explore the space of solutions well. In order to include exploitation, good ants should influence the new pheromones more than others, as usual in ACO.

7. How would you apply ACO to finding the cheapest way to fly from Edinburgh airport to the airport of Bora Bora (Leeward group of Society Islands of French Polynesia)?

Answer: There are other approaches, but a standard ant system should be able to find a good solution. Use the price for each leg of the flight as local heuristic (you may want to include also hotel costs etc. when a route requires waiting for a connection).
The problem is here that i.e. the nodes in the graph that are easy to reach (major airports) have a large degree (lots outgoing flights), and that the cheap flights (e.g. from Edinburgh) are unlikely to get you anywhere near Bora Bora.
The large degree implies the use a larger number of ants than usual. We can, however, expect that soon pheromone trails will indicate the larger airports which is probably good.
A further idea is to first find any route to Bora Bora (not necessarily a cheap one) such that ants can be terminated when the exceed the price of this route by a certain factor, which will focus the search. This reference route will later be replaced by any better route found by the algorithm.
In addition, consider: taboo list, using geographical distance as an additional factor to the local heuristics, same for price per reduction of distance, use a small value of beta (in order to be able to use locally also expensive flights e.g. for a long distance) which may be increased later.

8. Discuss the application of ACO to the eight-queens puzzle. This puzzle is the problem of putting eight chess queens on an 8x8 chessboard such that none of them are able to capture any other using the standard chess queen's moves, cf. http://en.wikipedia.org/wiki/Eight_queens_puzzle.

Answer: (s. S. Khan et al."Solution of n-Queen problem using ACO". In proc. of 13th IEEE International Multi-topic Conference (INMIC 2009), Islamabad, Pakistan.)
Each ant simply place queen by queen on the chessboard, which can be done in a search space of n times nxn nodes. Think of nxn rows and n columns: each ant runs from left to right (i.e. n steps including the initial placement) each time selecting one of the nxn fields of the chessboard). Local heuristic is whether an queen can "kill" any one of the queens that are already there for this ant. The paper also studies alpha and beta and finds alpha slightly >1 and beta about 1.5 to be good values.

9. Computer exercise: Run the standard ACO on the travelling salesperson problem with N cities. You may use code from http://www.aco-metaheuristic.org/aco-code/ or elsewhere or partially reuse your code from the 1st assignment. Start with n_{ants}=N, alpha=1, beta=2, rho=0.75. How can you influence the quality of the stationary solution. Consider the standard deviation of the tour length over the ants during one iteration.

Answer: s. lecture slides. Strictly speaking, the quality of the stationary solution can only be influenced by the choice (or changes) of the parameters before the solution becomes stationary. However, theoretical studies often consider quasi-stationary solutions: run the algorithm for some parameters and wait until the characteristics (effective decree, mean and variance of the solutions) do not change anymore, then change parameters. In physics this is called adiabatic approximation, it allow us to tell whether the parameters are responsible for a certain change or whether it had happened anyway).
Smaller values of alpha and beta should help, as well as higher evaporation, i.e. larger rho. Generally a low standard deviation is a sign of (premature) convergence, while high standard deviation is not a problem if you make sure to keep the overall-best ant (and to reinsert it should the mean tend to increase or to use some other form of elitism)