Resolving the Travelling Salesman Problem with Ants

By Samuel Guilhem-Ducléon

The problem

I developed the Evolife module for the Travelling Salesman Problem using ants.

➔ I wanted to know how the parameters influence exploration and exploitation.

Let us make a brief reminder of the problem : we are given a set of nodes.
All those nodes are the vertices of a graph where all the vertices are connected together.
We want to find the shortest path which goes through each node.

This problem is NP-hard although polynomial algorithms find good solutions, i.e. short paths but not necessary the shortest. We will see that using ants provides good solutions and also a history of all paths found : we can store a list of alternative paths.

I used the formulas that are given in the course :

$$P_{i \to j} \propto \frac{\text{pheromone}_{i,j}^{\alpha}}{\text{distance}_{i,j}^{\beta}}$$

That means that the probability to go from node i to node j is proportional to the quantity of pheromone between node i and node j exponent alpha over the distance between node i and node j exponent beta.

$$\text{pheromone} = (1 - \rho) \cdot \text{pheromone} + \frac{1}{L^k}$$

When an ant ends its trip, i.e. it has gone through all nodes, we update the quantity of pheromone between each node of the path. The shorter the path the more the ant deposits pheromone.

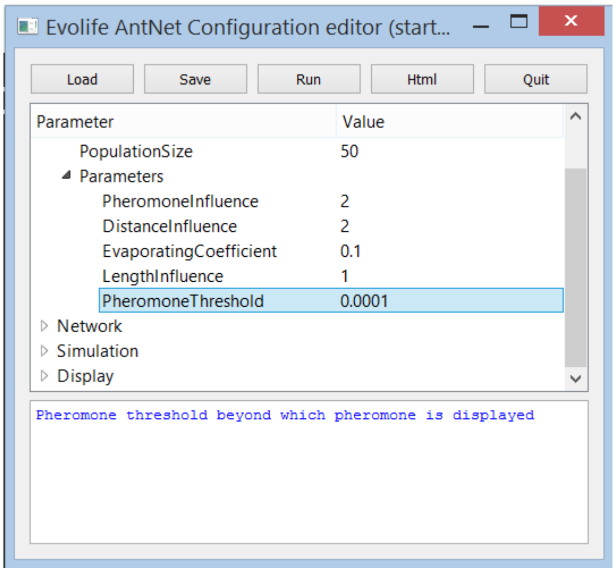
The pho coefficient is the evaporating coefficient : the pheromone evaporates over time.

The program

We see that four coefficients are arbitrary here : alpha, beta, pho and k.

I update the configuration file to add those parameters.

I also added another parameter which is the threshold of pheromone beyond which we display the pheromone graphically.



Parameter	Value
PopulationSize	50
Parameters	
PheromoneInfluence	2
DistanceInfluence	2
EvaporatingCoefficient	0.1
LengthInfluence	1
PheromoneThreshold	0.0001
Network	
Simulation	
Display	

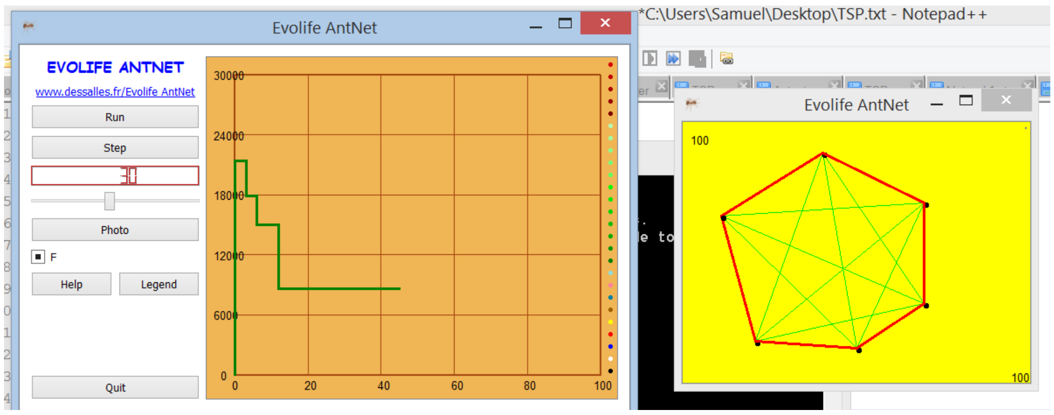
```
Pheromone threshold beyond which pheromone is displayed
```

I started reading the python code of the AntNet module to know how to handle graphics in my code.

Finally as AntNet does, the module can load graphs from a file or can generate random graphs.

The red lines represent the current shortest path found. The green lines represent the pheromone (they are displayed if there is more pheromone that the pheromone threshold).

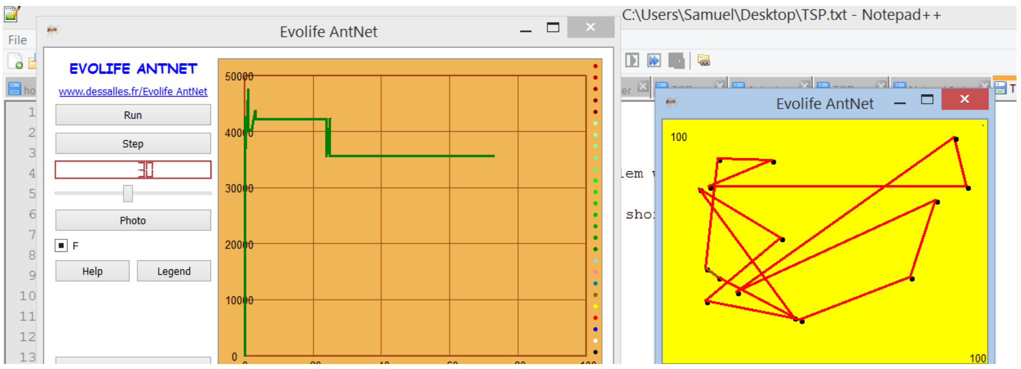
The curve represents the length of the shortest path found in function to the number of iterations.



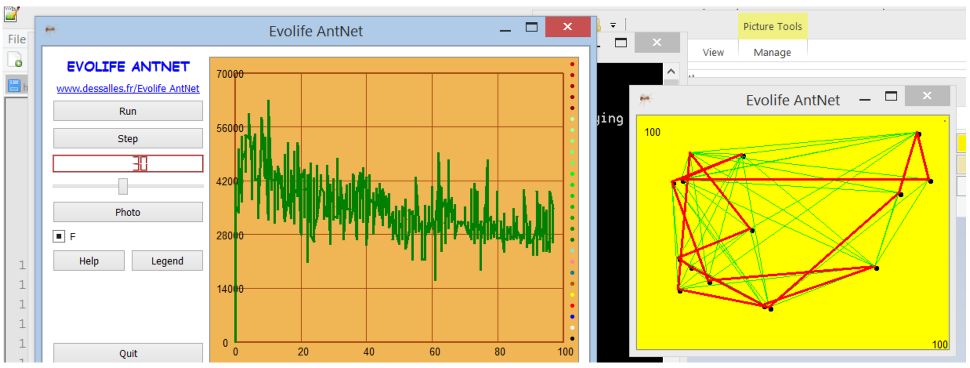
The ants are spawned randomly on the graph and at each iteration, one ant is taken randomly and goes to the next most probable node.

I generated a graph with 15 nodes randomly then stored it into a file and see how the problem varied when changing the parameters.

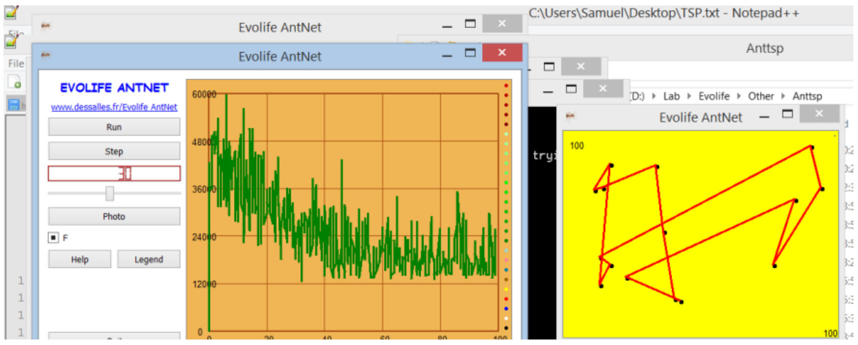
First of all, I increased the pheromone influence with no evaporation (α = 10, β = 1, ρ = 0) : exploitation is dominant and the program does not find a good solution. Pheromone is positive feedback here, and it is a memory of previous paths borrowed by the ants. Without evaporation the ants rapidly tend to behave the same way without trying new solutions. I did not display pheromone.


Evolife AntNet
EVOLIFE ANTNET
www.dessalles.fr/Evolife AntNet
Run
Step
Photo
Help
Legend
C:\Users\Samuel\Desktop\TSP.txt - Notepad++

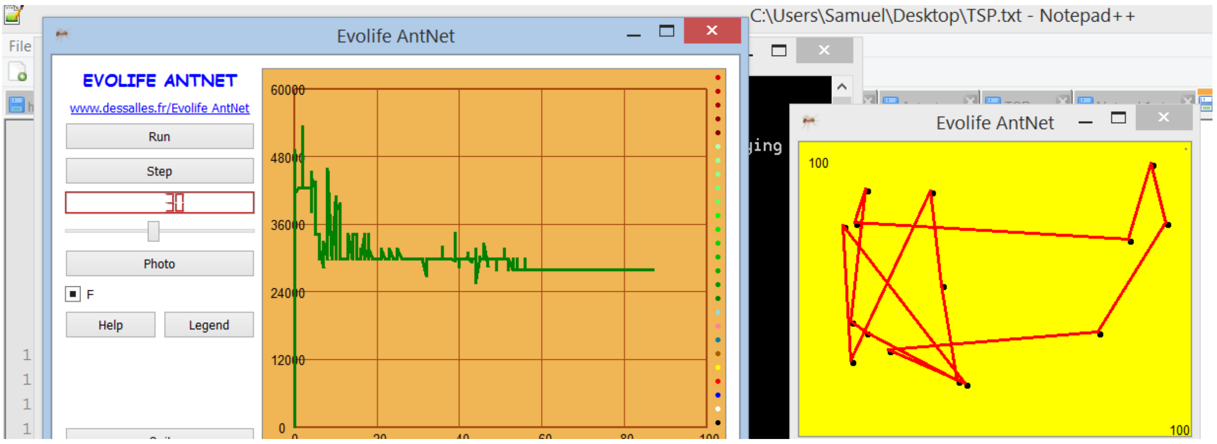
Then I suppressed the distance influence (α = 2, β = 0, ρ = 0.4) and I increased evaporation. The curve is far more unstable. Exploration is dominant : the ants do not stick to previous tried paths. The curve decreases slightly on average because the quantity of pheromone increases if the path is shorter.



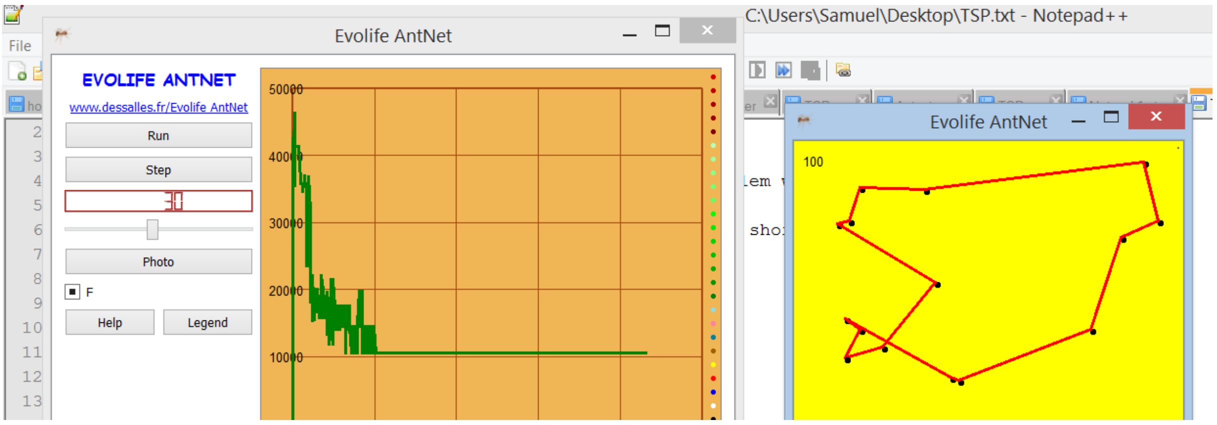
Now I added the influence of distance compared to the previous situation (α = 2, β = 1, ρ = 0.5). I stopped displaying pheromone (no specific reason for this). The curve is still very unstable because of the important evaporation rate but the global trend is more obvious because of the influence of distance in the probability of choosing next node.


Evolife AntNet
EVOLIFE ANTNET
www.dessalles.fr/Evolife AntNet
Run
Step
Photo
Help
Legend
C:\Users\Samuel\Desktop\TSP.txt - Notepad++
Anttsp

Now I set very few evaporation and a similar influence of pheromone and distance (α = 2, β = 1, ρ = 0.05). The curve is more stable but there is few exploration : the program is stuck in an intermediary solution. If the next shorter solution is far away, since the is no evaporation and thus few exploration, the program will not find it.



Eventually, with a balanced influence of pheromone and distance and a compromise in the evaporation rate, the program finds a very good solution, it is quite impressive.



Conclusion

The evaporation rate controls the exploration we want. Unbalance between distance and pheromone leads to more exploitation or more exploration.

The problem is very adaptative : if a node disappear, the new solution is found quickly.

The solution found lastly is not the best one and I could rerun the program several times to finally get the best solution, or modify the parameters until finding the best solution. But the goal is to be adaptative to many situations and we cannot change the parameters every time … Let us be honest, the ant solution does not necessary give the best solution. But it finds very good ones and the main advantage of solving the problem with ants is that we can keep an history of very good alternatives !