Swarm-intelligence with ants

Daniel Wurzer, FK09, Wirtschaftsingenieurwesen

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN MÜNCHEN

Executive Summary

This paper is about ants. Ants are individuals which are on their own not really intelligent actually. But they can act very intelligent as a group. This is called swarm intelligence.

In this simulation, these nice animals are modeled and problems like route-planning or finding a good path within many paths can be iterated quickly, even if the underlying network structure is changing over time. This process is called heuristic.

Introduction

Multi agent simulations are getting more and more popular. Where normal continuous mathematical models are getting to the limits due to many input variables this kind of algorithms helps. Applications are for example economic markets where many people are trading shares. Also crowd simulations are a good simulation to pre-test panic situations. There are many individuals with different discrete options. These are so called agents.

Here ants are used. The move along certain paths and try to find heuristically the best solution. The advantages are that they are very quick. Also they return good results even when the parameters are changed over time.

Description of the problem / use case

Find the best solution when there are so many options that you never can compute every option.

For example, finding the optimal path between many destinations. This problem is known as the Traveling Salesman Problem. The pattern can be used for everything where networks are used:

In logistics, telephone industry, production lines or even computer games where programmed individuals, also known as bots, have to react on their environment and should find optimal solutions (path planning). The ant itself should be robust against disturbance variables from the environment or other influences. With few simple rules programmed in each agent it is very robust and highly adaptable to many real world problems.

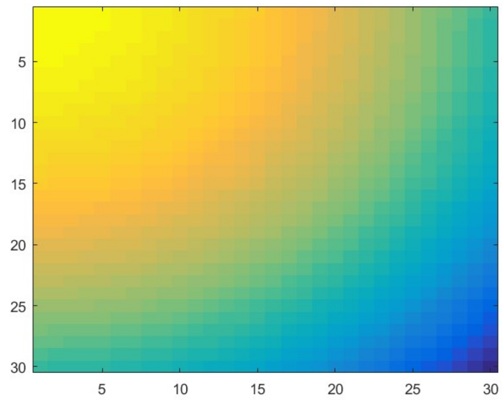
Model

The whole model is a discrete model. This means the ants are represented as an element in a 2 dimensional matrix and there is nothing continues like a known mathematical function like the Sinus.

The environment

The 2-dimensional matrix represents the space or the world where the ants are moving. In this model the nest is in the upper left corner of the world in 2D space. In this model the world is planar and bordered left right and top down. There also exist toroidal worlds where when you go too far left you start on the right side again. And the same with up and down. This fact is the difference between Snake 1 and Snake 2 on old Nokia phones.

Furthermore, a layer of nest scent is spread out over the entire map, so that the ants find back to their nest. They just go in the direction with the highest nest scent.



The nest scent over the entire map, yellow is 100% and blue is 0%

This is similar to the gradient function in mathematics. To visualize how the ants are moving along the gradient, you can think of mountain climbers which are heading towards the highest point. The ants are doing nothing different. They just move in the direction of the highest nest scent. This nest scent is necessary, so that the ants will find back to their nest when they found food.

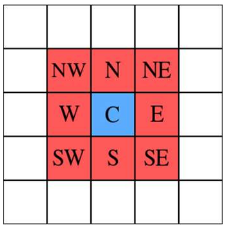
The Ant

At first we have to transfer a real ant into a mathematical model. The ants are represented as one element in the 2D matrix (world) and can move in 8 directions.

These directions are **up, down, left, right**

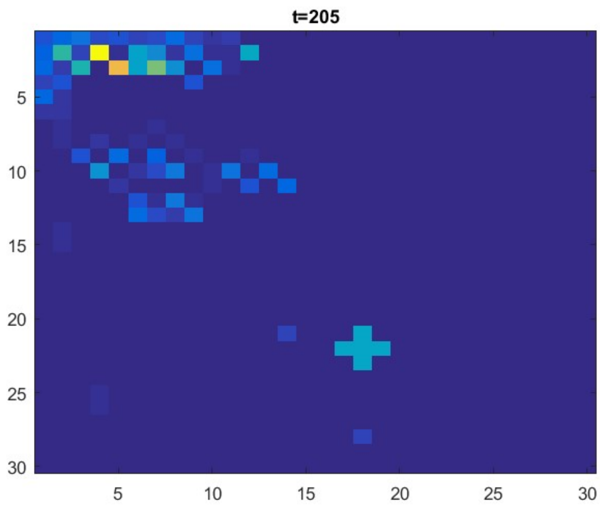
and the diagonals **upper left, upper right, lower left** and **lower right**.

In every time step they can move only by a maximum of 1 unit within these 8 directions. This is also considered as Moore neighborhood after Edward F. Moore, a pioneer in cellular automata theory.



Ant individual *The Moore neighborhood (moving directions in red)*

In the first place an ant is born in the nest and is looking for food. This means that the ant moves in random directions until it finds something to eat. Furthermore, every individual has a limited amount of pheromones carried within its body and is constantly spreading some of them along the path. If the ant is moving too far away from the nest without finding food it will lose its ability to distribute pheromones in the world matrix. This has to be the case because otherwise other ants will also follow the ant and die far away from the nest due to the lack of something to eat.



The nest is in the upper left corner, some of the ants are moving and the light blue parts are the pheromones they leave behind while moving along

In the case that an ant finds something to eat it distributes more pheromones on its way than before. This is because now the ant produces more scent because of the recent food supply.

After it found something to eat it heads back to the nest (like the mountain climbers). It is assumed that the ant brings some of the food back to the nest or want to get more other ants to the food source to supply everyone.

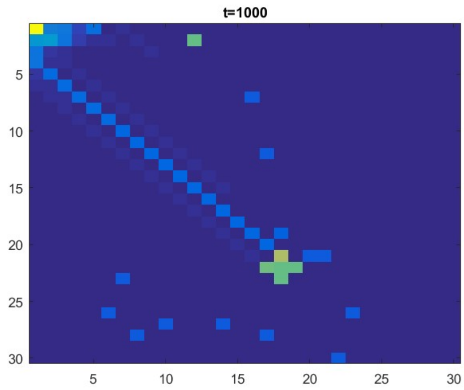
Simulation/ Solution of the problem

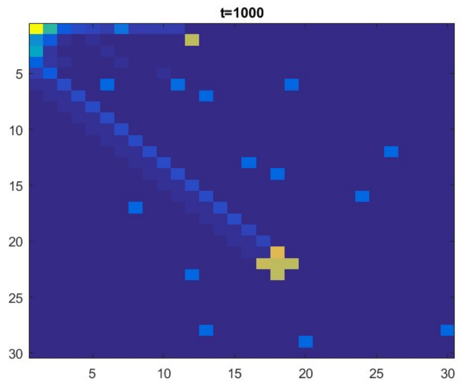
The simulation is inspired by the Game of Life. This is one of the simplest cellular automata. It was invented by John Horton Conway in 1970. There are few very simple rules. But this rules can lead to very complex systems. Someone even realized a whole computer with registers and a CPU which can be programmed and bit-wise operations like AND, OR and so on. Many other cool things can be made too.

This simulation is a multi-agent simulation realized in the script language called Matlab. It is also good to find a good start and learn object orientated programming with this kind of tasks. The ants can be easily modified and made more complex. New rules can be appended and instructions can directly be implemented in the ant object itself.

Note that every ant has the same code. But they react differently in different situations. This leads to complex structures and represents the way swarm intelligence works.

Below you can see some simulation results.

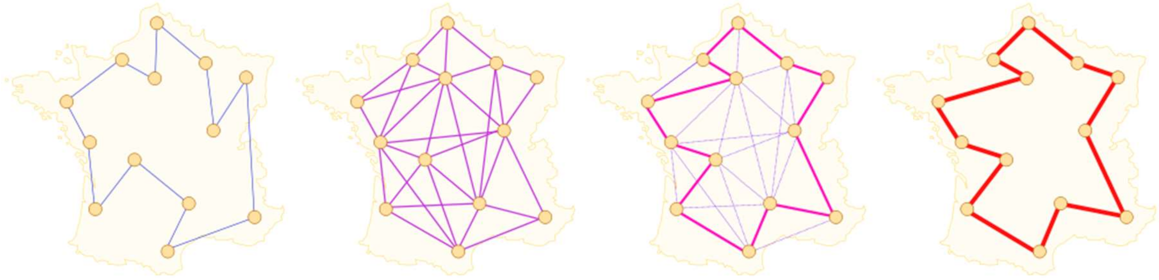




The result looks a little bit different every time; this is because the highly use of random numbers. They have a big influence like in Monte Carlo simulations. But over time they converge to the same results at the same starting parameters. Here you see the same initial parameters over 1000 time steps. The solutions look nearly the same, but in the right image you can see that the shorter path on top is better detectable because of more ants moving on this path near the 1000th time step.

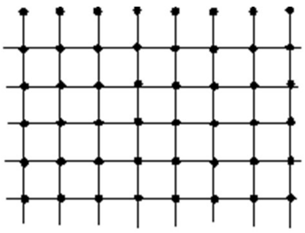
Evaluation of the simulation / solution results / Analysis of results / Sensitivity Analysis

Normally a graph, in sense of many connected dots or also a network, is used and the ants move along the edges of the graph. The destinations are called vertices.



The graph is not planar here. The shortest path is marked red

But if you consider each pixel or each element in the 2D world as node or vertices then an evenly distributed planar graph can be estimated. This makes it possible to load images with landscapes or data visualization more beautiful and a little bit more understandable.



In this simulation the pixels can be estimated as an evenly distributed planar graph

Further Improvements

- The ants are born all at the same time, which is a bit unrealistic. Better would be if they were born one after another to bring a little bit more realism in the simulation.
- In the simulation there aren't yet any unpassable areas which would be really interesting if the ants would find a path around these obstacles or how they would react to such difficulties.
- A better heat map/visualization would be nice to see the differences between the ants, food and pheromones better. Also the food status of the ants could be visualized. If an ant has found food already or not.