Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Lecture with Computer Exercises:
Modelling and Simulating Social Systems

Project Report

Vector based navigation of desert ants

Josua Graf, Noah Zarro

Zurich
December 9, 2018

Agreement for free-download

We hereby agree to make our source code for this project freely available for download from the web pages of COSS. Furthermore, we assure that all source code is written by ourselves and is not violating any copyright restrictions.

Josua Graf

Noah Zarro

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Eigenständigkeitserklärung

Die unterzeichnete Eigenständigkeitserklärung ist Bestandteil jeder während des Studiums verfassten Semester-, Bachelor- und Master-Arbeit oder anderen Abschlussarbeit (auch der jeweils elektronischen Version).

Die Dozentinnen und Dozenten können auch für andere bei ihnen verfasste schriftliche Arbeiten eine Eigenständigkeitserklärung verlangen.

Ich bestätige, die vorliegende Arbeit selbständig und in eigenen Worten verfasst zu haben. Davon ausgenommen sind sprachliche und inhaltliche Korrekturvorschläge durch die Betreuer und Betreuerinnen der Arbeit.

Titel der Arbeit (in Druckschrift):

Vector based navigation of desert ants

Verfasst von (in Druckschrift):

Bei Gruppenarbeiten sind die Namen aller Verfasserinnen und Verfasser erforderlich.

Name(n):	**Vorname(n):**
Graf	Josua Andreas
Zarro	Noah

Ich bestätige mit meiner Unterschrift:

- Ich habe keine im Merkblatt „Zitier-Knigge“ beschriebene Form des Plagiats begangen.
- Ich habe alle Methoden, Daten und Arbeitsabläufe wahrheitsgetreu dokumentiert.
- Ich habe keine Daten manipuliert.
- Ich habe alle Personen erwähnt, welche die Arbeit wesentlich unterstützt haben.

Ich nehme zur Kenntnis, dass die Arbeit mit elektronischen Hilfsmitteln auf Plagiate überprüft werden kann.

Ort, Datum	**Unterschrift(en)**
Winterthur, 9.12.18	J. Graf

Bei Gruppenarbeiten sind die Namen aller Verfasserinnen und Verfasser erforderlich. Durch die Unterschriften bürgen sie gemeinsam für den gesamten Inhalt dieser schriftlichen Arbeit.

Contents

1 Abstract

In this project we simulated the behavior of desert ants, as stated by R. S. Wehner in his report (Collett et al. [1998]). He conducted experiments with real ants and proposed a theory how these ants handle the orientation in the desert. We used his results to simulate an ant, which behaves as Wehner stated in his thesis and conducted his experiments again in simulation. It seams that our simulation matches the results of his real tests pretty well.

2 Individual contributions

N.Z and J.G set up a basic concept for the simulation, N.Z. and J.G. programmed the simulation, J.G. run the simulation, N.Z and J.G. wrote the report.

3 Introduction and Motivations

3.1 Motivation

Desert ants (*Cataglyphis*) has a difficult task to navigate since there natural habitat has not much landmarks helping to find there way. Different scientific papers (Müller and Wehner [1988]), (Collett et al. [1998]), (Knaden and Wehner [2005]) are postulating that desert ants use path integration as one component of there navigation. Because many technical applications are inspired by nature it is of high interest to have a clear idea how ants navigate in a unrecognizable landscape like desert. We could imagine a technical use in satellite steering or the navigation of evacuation robots in.

3.2 Base Paper

This project is based on the paper "Local and global vectors in desert ant navigation" (Collett et al. [1998]). Through simulating an ant which follows the concepts of local and global vector described by Collett et al. [1998] and comparison with the actual behavior we wanted to find out if the concept postulated is complete or needs adjustments. The goal was to implement the ant following the concepts as accurately as possible, then simulate the same test setup as it was used for the paper and compare the simulated plots with the plots from the experiments.

3.3 Hypothesis

The hypothesis for our simulation is that if we implement all the in the running text of the paper (Collett et al. [1998]) described navigation tools of desert ants in our simulation we would get the same plots than Collett and his collaborators got in there real life experiment. To prevent our simulation to be a self-fulfilling prophecy the basis of implementation are only the described concepts in the running text and not the results of there experiment.

4 Description of the Model

4.1 Behavior of the ant

For our simulation we aimed to implement our model as closely as possible to the postulated model by Collett and his participants (Collett et al. [1998]). Their model splits the navigation of the ant in two parts. First, if there are no known landmarks in sight (which is the usual case in the desert, the ants natural habitat) the ants orient themselves with a process named 'path integration'. An ant leaving their nest to gather food counts each step it takes and measures its direction with aid of the sunlight, which it uses as a sort of compass. Like this it is always aware of its position relative to the nest and is able to find its way home. With this practice it is also less vulnerable to changes in the environment. The integrated value, respectively its negative which is pointing to the nest, is called the `global vector` by Collett and his colleagues. However, this practice becomes unreliable if the sun is not visible or the ant makes a mistake counting its steps. Therefore the ant also uses a different form of finding their way. It uses known landmarks, as for example rocks, bushes or other static objects for his orientation. Collett uses the word `local vector` to describe this form of orientation. Here it is important to say that as Collett says he does not calculate its `local vector` according to the direction of travel, like "after the gray rock go left", but relative to the cardinal direction, like "after the gray rock go south".

4.2 Conducted experiments

To prove their theory the four researchers conducted several experiments. They built a training ground for the ants, where they had a nest, a feeding ground and a so called 'channel'. The channel was a tunnel dug in the ground just deeply enough that the ants could not see it form the surface. The ants were left in this training area, where they continuously gathered food and brought it back to the nest. So they got used to the setup and memorized the local landmarks and therefore set up their `local vector`. After a few days training the researchers started to conduct several different experiments, where they changed the setup partly. In some experiments they changed the the length and position

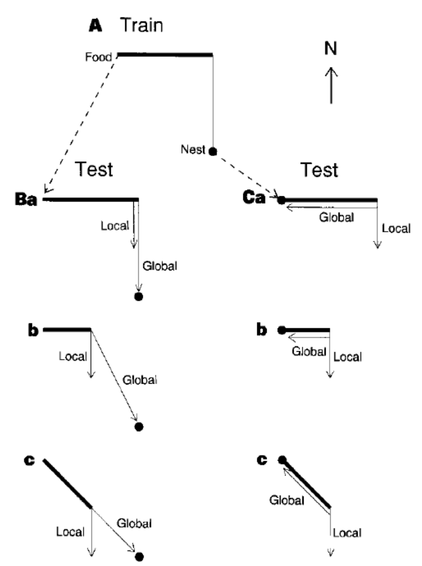

Figure 1: Test setup and some experiments conducted by Colletts. Collett et al. [1998]

of the channel, which disordered the ant's `local vector` and in some other cases they picked up the ant at their nest or on their way home and released it somewhere else, in order to give them a false estimation of their position according to the `global vector`, see 1 on the preceding page

Furthermore the group conducted an other kind of experiments, where they used no channel, but six highly visible black cylinders. They formed some kind of alley, where the ants had to get trough in order to reach their food. With this arrangement the ants were trained again, see **??** on page ??. In the experiment conducted with those ants, they were picked up at the nest and released them at the feeding station.

5 Implementation

5.1 Main Concept

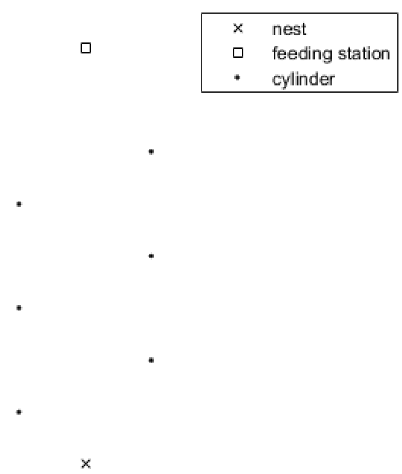

Figure 2: Training ground with cylinders.

Our goal was to set up a simulation environment which was as general as possible, so we could easy switch between the test cases. The simulation is actually a huge loop, where in every iteration the ant does a step on a grid. It can just move horizontally, vertically and diagonal. This of course limits our ant, but if we simulate it with a big enough resolution it does not really matter. The ant can not reach every point on the map, there are several restrictions. First there is the channel, which can only be entered at specified channel exit or enter points. On the other hand the ant can not leave the map, so we constructed a virtual wall around the whole map, which it can not pass. Finally there is the nest, which we placed in the middle of the map.

In every iteration, the desired direction of the ant gets calculated according to the global and local vector in the `desired_direction` function. Then we calculate all possible directions in the `possible_direction` function. At the end the function `actual_step` lets the ant walk one step and the global vector gets adjusted. If the ant did a certain amount of steps the simulation stops and the path gets plotted.

Our whole simulation environment is highly configurable, in the `conifg.py` file one can import any test file, in which all things like the map configuration, the start position of the ant etc. can be specified.

5.2 Global Vector

The global vector starts with an initial value, which can be edited in the test file. In every iteration of the main loop the step conducted by the ant gets subtracted form the global vector. Additionally to model the imperfectness of the ants 0.1 mg brain we introduced two random factors. One gives credit to the fact, that a normal ant never walks in a straight line but rather zig-zags its way home. For this reason before a step is executed by the ant, the step gets randomly rotated, but with a gaussian distribution around the straight forward direction, see lst 1. The second randomization mirrors the fact that an ant never can memorize exactly how many millimeter it walked north or east. Thus the global vector gets slightly rotated randomly after every step, which means the ant does not exactly remember which way it took to get to its current position.

Listing 1: Randomization of desired step

```
# choose a random angle
sigma = con.sigma
randomAngle = np.random.normal(0, sigma)

# construct rotation matrix
rotationMatrix = [[math.cos(randomAngle), -math.sin(randomAngle)], [math.sin(random
math.cos(randomAngle)]]

# rotate vector
rotated = np.matmul(rotationMatrix, desired)
```

5.3 Local Vector

The local vector gets calculated every iteration and depends on the close surroundings of the ant. The influence and the position of these surroundings can be configured in the test files. The channel exit for example pushes the ant to the south. All these objects have a circular area of influence and the influence lowers quadratically with the ants distance to the object. All influences of all objects get summed up to the local vector and are merged with the global vector to the desired direction with the following formula:

$$desired\_direction = local\_vector * local\_weight + global\_vector * (1 - local\_weight)$$

To calculate the local vector in the area of a familiar path the path is represented by a straight line with start and end point. Mathematically the straight is a linear function $y(x) = m \times x + q$ and the vertical slope n is calculated by $n = \frac{-1}{m}$. This implementation leads to errors for familiar paths with slope 0 or ∞. Since the familiar path does not have to be a such exact defined line we can use this simple implementation for adjusting the local vector in the region of objects.

6 Simulation Results and Discussion

6.1 Simulation Results

Due to our simulation setup the results are plots of traced ant paths on a square map. To compare them with the experiments conducted by Collett et al. [1998] we take e look on there plots and define some critical characteristics in the plots, that we would like to recognize in the simulation.

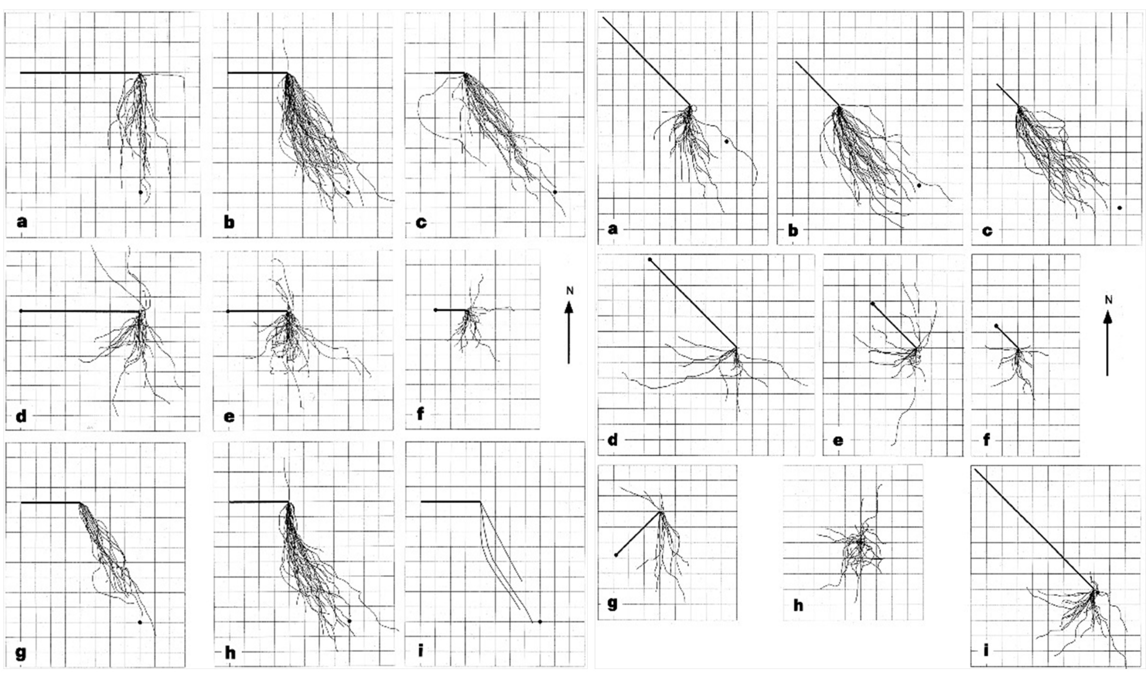

Figure 3: Trajectories of ants 'homing' on the test area. **left a)** results of test setup Ba. **left b)** results of test setup Bb. **left d)** results of test setup Ca. **left e)** results of test setup Cb. **right b)** results of test setup Bc. **right e)** results of test setup Cc. Experiments conducted by Colletts (Collett et al. [1998])

6.1.1 Test setup Ba

The setup Ba is a horizontal channel starting at the feeder/launching position in the northwest and has its exit exactly in the north of the nest. In this run the launched ants had been picked up at the feeder on the test ground therefore they have the correct global vector pointing form there position to the nest.
In the plot of the experiment it can be seen that all the ants leaving the channel in south direction and then loose the direct path in a kind of cone shape. Comparing the simulation with the experiment the simulated ants leave the channel collectively in south direction and spread later than the real ants.

6.1.2 Test setup Ca

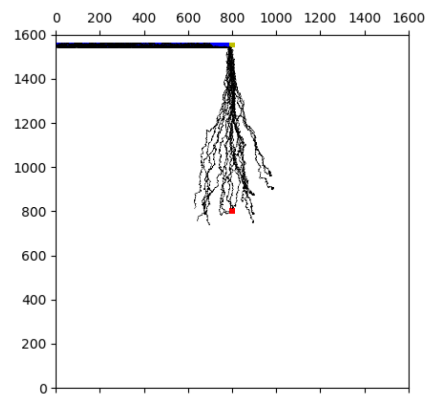

Figure 4: Simulation setup Ba

The setup Ca is almost the same than Ba but the launched ants had been picked up at there nest. That means they do not have a global vector at there launching position/feeder and allocate this position as the origin to calculate the global vector.
In the experiment the ants leave the channel mostly in south direction as trained but then do a kind of searching walk. The ants in the simulation heading back to the feeder but outside of the channel, because after leaving the channel in south direction the the local vector is missing and the global vector takes control of navigation. The global vector is total is useless since the ants do not know there position compared to the nest and just walk back to there launching position. In this simulation it can be seen, that Collett et al. [1998] did not describe the ant navigation if the global vector is disturbed by different pick up and launching position. We did not implement a searching walk since there is no idea mentioned in the paper how the ants do that.

6.1.3 Test setup Bb

In the setup Bb the horizontal channel from the feeder ends at half horizontal distance to the nest. The training ground for the ant is the the same than before.
In the experiment the ants walked from the exit

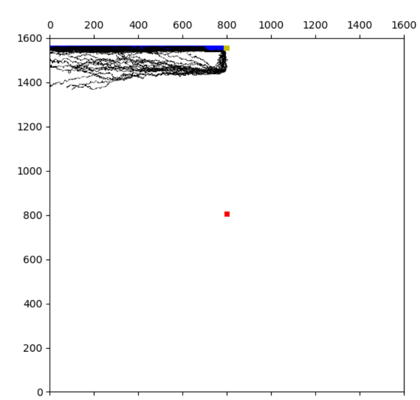

Figure 5: Simulation setup Ca

to the nest in a slightly south curved diagonal. This can be seen better in the experiments plots **left g-i** there are the ants "segregated according to whether ants went to the left (**g**) or to the right (**h**) of a point 2m south and 0.5m east of the channel exit." (Collett et al. [1998] Figure 2) All the ants walking to the right (**h**) did the same as they trained actually heading south after the exit just when the local vector ends the global vector leads the ants home. In the simulation this behavior is stronger but similar.

6.1.4 Test setup Bc

The setup Bc is a diagonal channel of half the distance from feeder to the nest.
In this experiment plot a similar behavior as in Bb can be recognized, the slightly south curved diagonal, but much less clear. In the simulation just one ant does not perform this behavior.

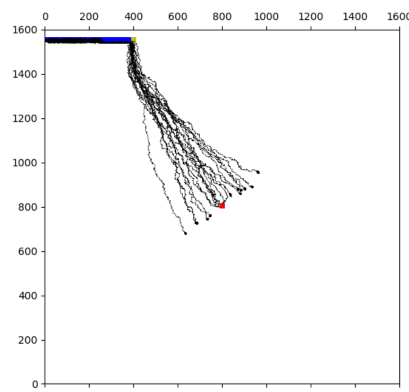

Figure 6: Simulation setup Bb

6.1.5 Familiar path near objects

The results from the
what did we get, is it the same

6.2 Discussion

what was different, probably the randomization was no as in real life

7 Summary and Outlook

7.1 Summary

it worked pretty well, Wehner did a good job, his experiments corresponds to our simulation, which was based on his theory. We do not know if its biologically correct

7.2 Outlook

we probably will not continue our research about desert ants, maybe Wehner will

8 References

M. Collett, T. S. Collett, S. Bisch, and R. Wehner. Local and global vectors in desert ant navigation. *Nature Magazin*, 394, July 1998.

Martin Müller and Rüdiger Wehner. Path integration in desert ants, *Catglyphis fortis*. *Proc. Natl. Acad. Sci. USA*, 85, July 1988.

Markus Knaden and Rüdinger Wehner. Ant navigation: resetting the path integrator. *The Journal of Experimental Biology*, 209, 26-31, November 2005.

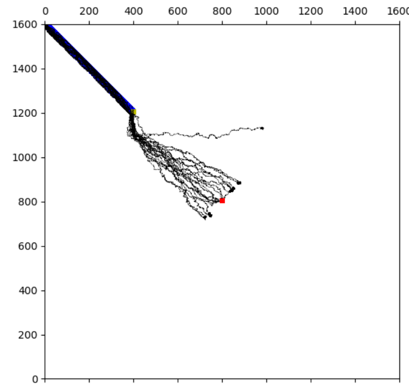

Figure 7: Simulation setup Bc

9 Latex-Stuff

Beispielverweisung:

Hier wird auf dieses Bild verwiesen (fig. 8 on the next page)

Python Code:

Listing 2: Python code example

```
import numpy as np

# show numbers 0-9
print ('Numbers from 0-9')
for i in range(10):
        print(i)
```

Beispielliste:

- erster Eintrag
- zweiter Eintrag

Figure 8: Beispielbild, in Ordner images