EVOLUTION OF SOCIALNESS

Project within the module
"Complex Adaptive Systems" by Claudia Pahl-Wostl
Universität Osnabrück

Johannes Knabe and Hauke Strasdat
July 2004

Contents

1 Introduction

1.1 Motivation

Among other species, humans often help other, needy people they come across - despite of the fact that a return is not for sure and sometimes even highly improbable. How could such a social, cooperative behaviour outlast the strong pressures of evolution when it is seemingly a disadvantage for the individual? Additionally, according to classic theories, the rational human (homo economicus) would always maximise its

personal wealth and thus not predict such a behaviour.
Having made these considerations we came to the following observations:

- Cooperating agents should be able to better exploit their environment. This means more resources and thus more agents.
  Thesis 1: *The grid's carrying capacity is higher for social agents than for selfish ones.*

- Being more productive, social communities should grow and such be able to aquire more (or better) land:
  Thesis 2: *In competing situations social agents prevail over selfish agents.*

- We assume that nature did not start immediately with some cooperating species, instead we think this idea has started in a few and in the run of evolution proven more successful for some species.
  Thesis 3: *Selfish agents become social in the long run or die out.*

In the following we describe an agent based model which reproduces the above observations from simple basic rules.

1.2 Model Idea

There are many types of cooperation, but we think that it should be appropriate to have a single resource representing all types of goods and services like e.g. our monetary system.
In order for an agent to become needy we need it to have steady consumption from an environment. Since we want some agents to be richer and thus able to help, the environment should be *diverse*, i.e. it should be very fruitful at some places while being meagre in other areas.
We want agent communities that might possibly shrink or grow, easiest realised by making it possible for agents to starve and reproduce.

2 Implementation

For the implementation we used the Quicksilver tool which allows for quick development and testing of agent models.[1] It is implemented in Java and such is our code. Quicksilver provides the basic notion of an active object in discrete time, models are forced to have an initial state and a state transition function. Starting from the initial state, the state transitions are recurrently applied in each step.

Normally the transition function is called on every object sequentially and since the successor state of an object partially depends on the states of other objects; this means that objects which are called later in such a sequence work on mixed data: Some objects may already have been updated while others are not. To prevent this we have introduced two modes; in every even time step all objects calculate their successor states. But they do not change their data, instead the next state is saved and only applied in the next (odd) time step. Like this we simulate that all objects are updated 'at once'.

Our model world is designed as a 2D grid composed of 40 times 40 cells. Every cell gets assigned a state of either 'STONE' or 'CORN' initially. Opposed to stone cells, corn cells can hold and produce ressources. The `density` of corn can be given as a parameter to simulate environments rich of resources as well as desert-like worlds. A corn cell is initialised with the number of resources it can maximally hold (`field_capacity`). To model a diverse environment, this number is heterogenous: It is a natural number randomly taken from the interval [`max_cap`/2, `max_cap`], where `max_cap` has to be given as a parameter. The number of resources actually available from a corn cell is at first set to half its `field_capacity` and afterwards in every cycle increased by the parameter `regrowth_rate` until it reaches its capacity.

$$\text{resources}_{(i,j,t)} = min(\text{resources}_{(i,j,t-1)} + \texttt{regrowth\_rate}_{(i,j)}, \texttt{field\_capacity}_{(i,j)})$$

But this would be quite a boring world if it would not be populated by some agents. We call our agents `farmer`. They are located in space, i.e. in every time step at most one agent can inhabit one cell. A farmer collects resources from the field which it

[1] http://www.usf.uos.de/projects/quicksilver

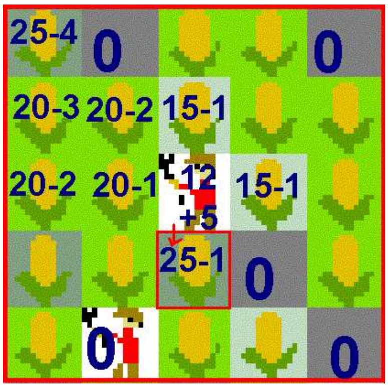

Figure 1: movement rule

inhabits and can store the harvest in a backpack, up to a maximum, the parameter `backpack_cap`. In order to survive farmers need to consume resources, thus in every cycle the number of resources in the farmers backpack gets reduced by the parameter `metabolism` if it has enough stored, otherwise the agent starves and is removed.

In order to collect enough resources farmers move on their grid world according to certain rules. But moving is costly and only local information is available, i.e. every farmer is endowed with a `sight range` in which it gets full information on the cells, thus farmers have to deliberate about moving carefully.

At the beginning of every cycle a farmer evaluates all neighbouring cells in its `sight range`. The value of a cell is zero if it is occupied by another farmer; otherwise it is the number of resources the cell holds minus the Manhattan distance to the current cell (figure 1). In order to simulate a kind of settledness, a `laziness` addend is introduced which is added to the value of the current cell.

$$\text{value}_{(i,j)} = \begin{cases} 0 & \text{IF CellIsOccupied}_{(i,j)} \\ \text{resources}_{(i,j)} - \text{ManDist}_{(i,j)} + \texttt{laziness} & \text{ELSE IF } (i,j) = \textit{my\_pos} \\ \text{resources}_{(i,j)} - \text{ManDist}_{(i,j)} & \text{ELSE} \end{cases}$$

$$\text{ManDist}_{(i,j)} := (|\text{my\_pos}.x - i| + |\text{my\_pos}.y - j|)$$

Actually all position values are additionally taken modulo 40 since we use *periodic border conditions*, i.e. an agent leaving the grid to the right would reappear on the

left of the grid.
Additionally, if an agent has filled up its backpack to its capacity, it reproduces. Reproduction is asexually, so the child inherits all the properties of its parent. Giving birth to a child costs the parent half its resources, the remains are shared: The child gets one third while the parent keeps two thirds. Now the cells in the agents `sight range` are evaluated as described in the movement section above and the child is placed on the best surrounding cell.
Finally farmers can interact with other agents depending on their similarity and the environment.
Agents are endowed with a boolean parameter `socialness` determining whether they are altruitic or not. This value is inherited from the parent or, for the first generation, given as a parameter specifying the probability of an agent to be altruistic. Farmers can recognise agents in an area of twice their sight range and determine the amount of resources those agents own. Social agents can give a needy farmer resources in case they have filled their backpack to its capacity instead of getting a child. Currently a needy farmer is defined as having one third or less of its backpack filled and in that case half of the donator's resources are transfered to the recipient. Thus the behavioral rule for altruistic farmer is:

```
IF (my_resources=backpack_cap) THEN
   FOR ALL x ∈ neighbourhood
        IF (x.is_hungry) THEN
             give(x, my_resources/2)
             EXIT
   produceOffspring()
```

A table of all important parameters and their default values are listetd in figure 2.

Grid World	default	Cell	default	Farmer	default
density	0.8	state	CORN	sight_range	2
max_cap	20	field_capacity	[max_cap/2, max_cap]	metabolism	2
		regrowth_rate	1	backpack_cap	15
		farmer	*false*	laziness	5
				socialness	*false*

Figure 2: `parameters and their default values`

3 Experiments

3.1 Thesis I

> The grid's carrying capacity is higher for social agents than for selfish ones.

3.1.1 Settings

To verify the first thesis it is necessary to conduct a experiment consiting of two turns. Starting with a initial number of agents one should performe a farmer world simulation as long as the total number of agents stabilises. This number mirrors the grid's carring capacity. In the first turn all agents are egoists, whereas in the second turn they are altruists. Finally both carring capacities are compared. All parameters except for the socialness have to be constant (used parameters: figure 2).
It is quite valuable to find out whether the experimental outcome is robust - i.e. it reflects a general property - or whether it relies on the specific model design. To test the robustness one should perform the experiment with different parameter values. The most relevant parameter according to the grid capacity seems to be the corn `density`: The fact, whether a environment is fruitful or meagre, is likely to influence the value of social behaviour. Therefore, in addition to the default environment (density 0.8) we performed the experiment in a dry-land environment (density 0.5) and a desert one (density 0.2).

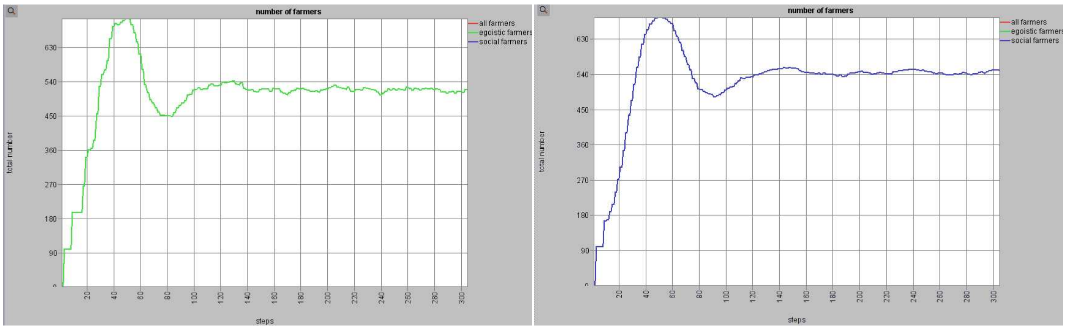

Figure 3: *left:* egoistic agents, average grid capacity approx. 525; *right:* social agents, average grid capacity approx. 545 *density:* 0.8 *time scale:* 300 steps

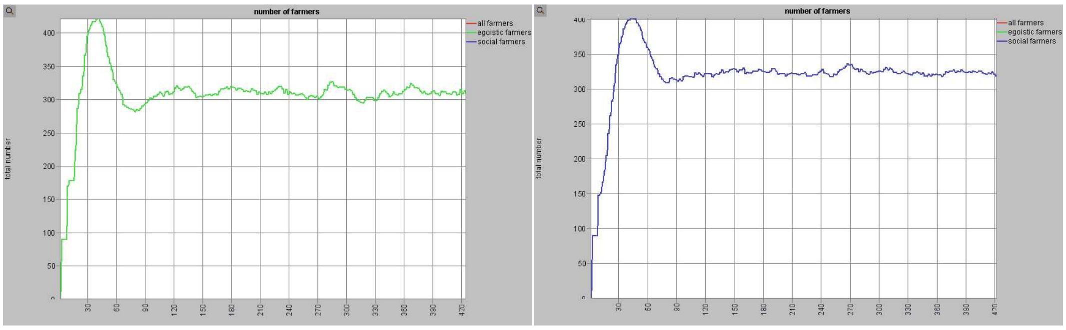

Figure 4: *left:* egoistic agents, average grid capacity approx. 315; *right:* social agents, average grid capacity approx. 325 *density:* 0.5 *time scale:* 420 steps

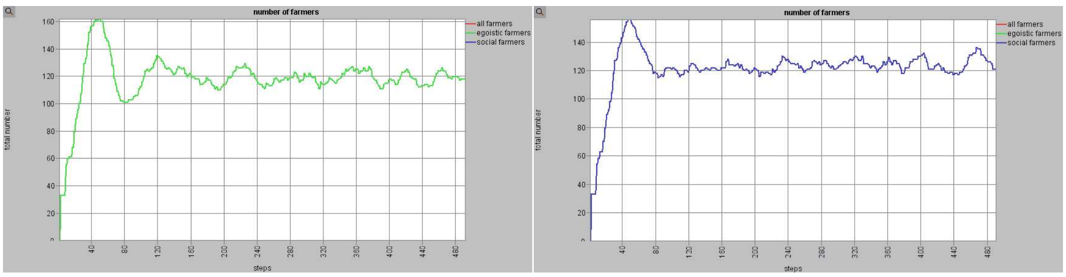

Figure 5: *left:* egoistic agents, average grid capacity approx. 120; *right:* social agents, average grid capacity approx. 122 *density:* 0.2 *time scale:* 480 steps

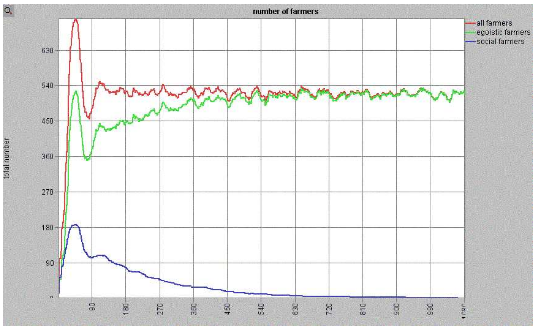

Figure 6: *initial population:* 50 egoists, 50 altruists *density:* 0.8 *time scale:* 1080 steps

3.1.2 Results

In the normal environment and in the dryland one (figure 3 and 4) the grid capacity is higher for social agents: 525 vs. 545 and 315 vs. 325. This differnce is not as high as we expected, but it is significant anyway. In the desert environment (figure 5) there is only a hair bredth distance between the grid capacity of social and selfish agents. This results from the fact that the distances between the agents are higher in the desert, so social agents have problems to stay in contact with each other.

3.2 Thesis II

In competing situations social agents prevail over selfish agents.

3.2.1 Settings

For this competitive test a single-turn experiment is required, in which the initial probability of being social is set to 0.5. The simulation is run until one fraction prevails over the other permanently.

3.2.2 Interim Results and Discussion

Figure 6 presents an unexpected result. Unlike Theory II egoistic agents dominate the environment and the altruistic farmers die out after 1080 time steps.

Our simple model which should mirror the advantage of social behaviour generates

contrary results. Do complex human characteristics like socialness be actually simulatable with such a simple rule-based model? The answer is yes, but it seems to be that some details in our model design have to be revised.
The behavioural rule of altruistic agents is rather rigorous:

```
FOR ALL x ∈ neighbourhood
    IF (x.is_hungry)
        give(x, my_resources/2)
        EXIT
```

This rule forces altruistic agents to give their food to egoistic one, which never would share their resources. It is indeed altrustic behaviour, but it is self-destructive anyway. Maybe it is possible to define social behaviour without sticking totally to altruism. A revised behavioural rule could be the following one:

```
FOR ALL x ∈ neighbourhood
    IF (x.is_hungry AND x.is_social)
        give(x, my_resources/2)
        EXIT
```

Agents, which this new rule are applied for, are called *social* (not altruistic) in the following. It is important to remark that this rule does not imply that social agents only give resources to individuals who will give or gave resources to him. It rather means that social agents share only resources with other agents who would do the same, if they were in the same situation, i.e. they are also social.[2]

3.2.3 Revised Settings

The settings are the same as in the previous experiment despite from the fact that altruistic agents are replaced with social ones. In adition the experiment is performed

[2]The rule is consistent with a setting in which one social agent donate his whole life without any reward, whereas another agent is permanently the receiver(, although it is very unlikely). Thus our definition of social behaviour is diffrent to 'tit for tat' strategies.

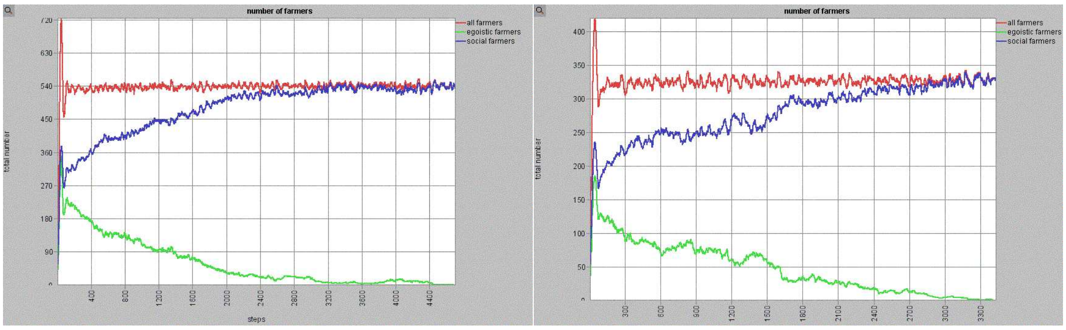

Figure 7: *initial population:* 50 egoists, 50 altruists *left:* density 0.8, time scale 4600 *right:*density 0.5, time scale 3400

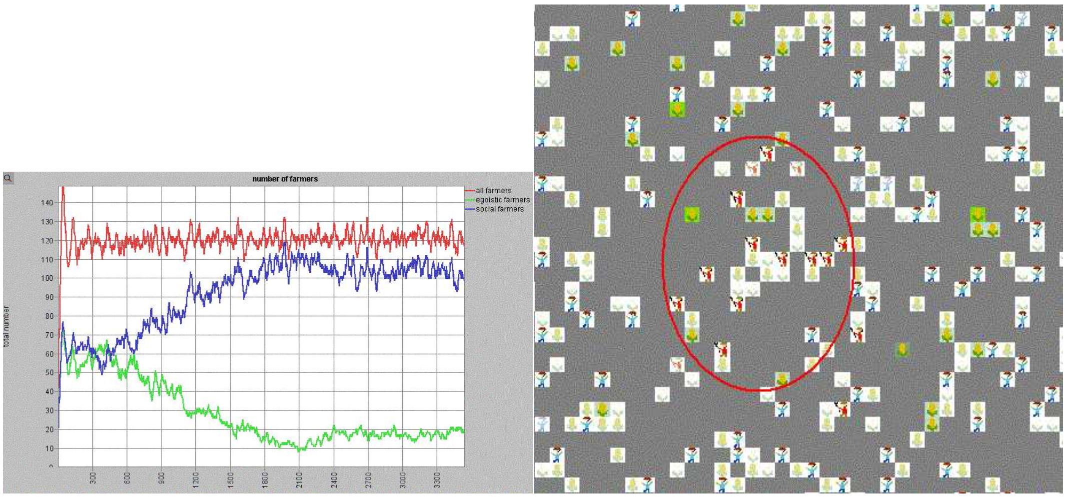

Figure 8: *initial population:* 50 egoists, 50 altruists *left:* graph: time scale 3500 *right:* snapshot, pattern of egoistic agents *density* 0.2

in the dryland and desert environment.

3.2.4 Final Results

As one can see in figure 7 social farmer dominate, whereas selfish farmers die out after a while. This result holds for the normal and dry-land environment. In the desert environment it is a bit different (figure 8, *left*). The social farmer dominate, but a minor fraction of selfish farmers also survive. This seems to be a stable situation. Why does not prevail the best strategy completely? One explanation would be that

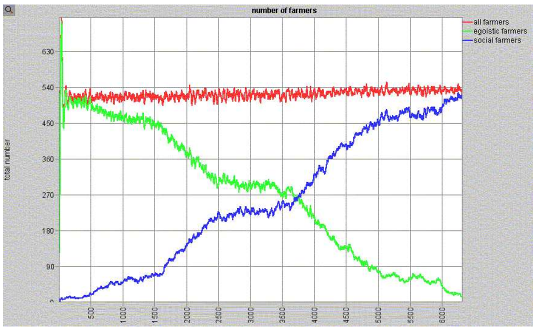

Figure 9: *initial population:* 100 selfish agents *mutation rate:* 0.01 *density* 0.8 *time scale* 6250 time steps

social behaviour is not so often applicable, since their are not as many agents in a social neighbourhood (= double sight range) as in a environment with a higher corn density (as mentoined in the result section of Thesis I). Additional information could be gathered if one analise the grid for patterns: Egoistic survivors often lives on 'islands' which are quite hard or even impossible to capture. (figure 8, *right*)

3.3 Thesis III

Thesis 3: Selfish agents become social in the long run or die out.

3.3.1 Settings

There are two ways of modelling mutation: By the aid of sexual reproduction and cross-over or by the aid of mutation (or both). Since the presented model contains *asexual* reproduction, a *mutation rate* is implemented: With the probability of 0.01 an offspring of a egoistic agent becomes social and vice versa. The simulatation is started with an initial population of 100 selfish agents.

3.3.2 Results

As one can see it takes a while before the social society overstep the value of 90 agents, but then the number increases quite rapid (figure 9). At time step 6250 the selfish society became extinct.

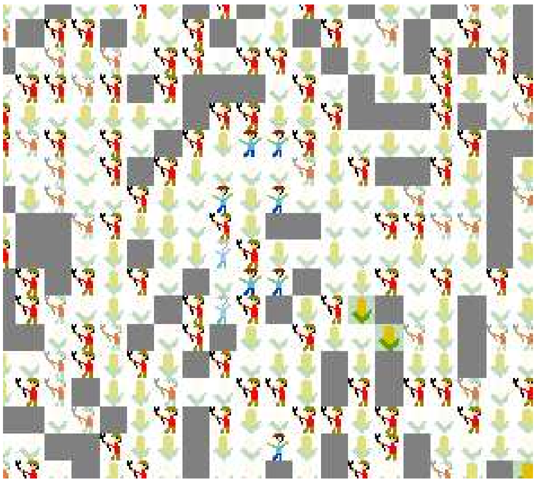

Figure 10: cluster of social agents

4 Discussion

It has been shown that from few assumptions and with simple rules situations result in which social behaviour dominates over egoistic behaviour. Seemingly such behaviour results in a better use of the resources the environment provides, which is probably due to the fact that the overproduction of 'good' cells is distributed to other agents. An isolated social agent can of course not expect any advantages from its being social. But from the moment only two social agents are close enough to realize each other this could be the rescue when resources are low in the area of one social agent. This explains the observed 'clusters', i.e. social agents sticking together, although the movement rule only states to go to the best field in the agents sight range. (10) This is probably the explanation for the positive feedback loops as well. When by chance a critical threshold in the number of social agents is overcome their numbers grow steadily and sometimes even more than linear.

4.1 Criticism and Outlook

Our conclusions are merely based on single experiments with random seed. To get more meaningful results, a number of experiments - each with different a seed value

- should be performed for each setting. In addition we only analysed the model's robustness in relation to the `density` of corn. If one would like to validate our result more precisely, it is necessary to vary also other parameters like `field_capacity`, `regrowth_rate`, `backpack_cap` and `metabolism`. It is possible to do this automatically within discretised intervals by running the Monte Carlo method.

For future work it could be of interest to introduce a second resource which could be traded. Trade might reduce the advantage of being social since it is then possible to acquire needed resources by other means. On the other hand it might show that trade only delays the time it takes until social agents dominate.

Additionally more heterogeneity than only social or not social might be included in the agents. Interesting would be to code sociality of an agent in a fuzzy way in its genes and then let it only help a needy agent if it is

a) less social than the donator

b) social in a range close to the donator's socialness or

c) more social than the donator

and to compare the results.

When the genome is extended, heterogenous backpack capacities, sight ranges and metabolisms could be added as well - being rich would then depend on a rich parent, good genes and a good starting cell.

Seasonality, i.e. varying the amount of resources produced every round on a cell, and times of drought, are not only more realistic to assume than a constant regrowth, but would probably increase the dominance of social agents since then a staving agent could receive help to survive the 'bad times'.

Finally it would be interesting to have sexual reproduction and a natural death and to see, whether and if, how, this changes the results we observed and discussed above.