Radboud University Nijmegen

Bachelor degree thesis in Artificial Intelligence

Sharing Blood

A decentralised trust and sharing ecosystem based on the Vampire Bat

August 25, 2012

Author:

Wouter Bulten
WouterBulten@student.ru.nl
Student number: s3040585

Supervisors:

Dr. W.F.G. (Pim) Haselager
Dr .I.G. (Ida) Sprinkhuizen-Kuyper

Sharing Blood: A decentralised trust and sharing ecosystem based on the Vampire Bat

Wouter Bulten

Abstract

Vampire bats manage to live longer by donating to fellow roost mates and trusting that they will, in the future, return the favor. I have modelled this interaction by creating a biological plausible, decentralised trust and sharing ecosystem. In a simulated 3D environment the performance has been tested by groups of artificial bats, which shows a significant increase in life span as a result of the bat trust ecosystem. To further test the system, groups of cheaters were added to influence the population of trusters. Even though cheaters have a negative influence on the population of trusters, the results show that this is (for the most part) not the result of their cheating behaviour. In other words, trust pays and is robust.

1 Introduction

Trust is intertwined with our whole life. We trust our family and friends to support us, doctors to take care of us and governments to protect us. Not only in our lives, but also in the lives of other animals, trust pays a significant part. Numerous cases have been reported of animals trusting and cooperating [5]. Think of a plover picking food fragments from a crocodiles mouth, the bird 'trusts' the crocodile not to eat it.

Unfortunately, because trust is so broad, we first have to define the part of trust we are interested in. Trusting that a particular document is correct differs substantially from trusting someone with your life. McKnight and Chervany [6] make a distinction based on the trust referent and formed four high level categories: *benevolence*, *integrity*, *competence* and *predictability*. *Benevolence*, "caring and being motivated to act in one's interest", is the trust type that will be referred to in this thesis. It describes that part of trust that includes "I depend or believe in your goodwill towards me".

One of the species that uses this kind of trust are vampire bats. These blood-drinking bats need their trust system to survive, without it their lives would be a lot shorter. Vampire bats help each other by means of altruistic food sharing. Wilkinson, who studies the behaviour of vampire bats, used simulations [15] to calculate the benefit of these altruistic acts. In this thesis I will further extend this research on the vampire bat and use computational modelling of altruistic food sharing.

The Monte Carlo simulation Wilkinson used included fixed association values between bats, fixed chances of finding food (always 90%) and only 11 bats. It is my aim to, first, improve his simulation and, second, extend it by introducing a new group of agents: cheaters. A first block of computer simulations will be used to answer the first main research question: What does the trust system contribute in terms of bat performance, i.e. to what extent is their lifetime prolonged, compared to a control group, and how, specifically, is this contribution influenced by the availability of food?

The second block of simulations focuses on cheaters and will try to answer the second research question: Do cheaters shorten the life span of trusting agents, is this caused by their cheating behaviour or just by the presence of a non-sharing group and how does the size of the cheater population influence this effect?

1.1 Dimensions of plausibility

When modelling a natural occurring system it is required to define boundaries on what will be modelled. Especially due to time constraints and complexity it is impossible to model every aspect of the living thing, let alone match how everything works internally. Fortunately it is not necessary to create a direct copy, some simplifications can be made without having too much of an influence. The boundaries that are used will be defined using the seven dimensions of Webb [10]: *relevance*, *level*, *generality*, *abstraction*, *structural accuracy*, *performance match* and *medium*.

1. **(Biological) Relevance:** To what extent the model "generates hypotheses for biology". The modelled system and the simulations will, hopefully, give more insight into the effects of trust on the life span of vampire bats. Also, the simulations of Wilkinson did not include the effect of cheating agents. With these simulations I will measure the effect of both their presence and their behaviour. This tries to give insight in how the ecosystem reacts when agents stop cooperating.

   Where Webb focusses on the biological relevance, the system and the results can also be relevant for other disciplines. When the results show that the system is able to prolong the life span of trust agents, it could be used in (artificial) systems where (equal) sharing is required.

2. **Level:** The social and population processes of the vampire bat will be modelled, there will be no focus on the underlying processes. This high level ensures that the model will not be over complex and easy to test and adapt.

3. **Generality:** The ecosystem will not be general. To properly test how good the trust system of vampire bats works we have to try to mimic it as closely as possible, therefore making it more general would be counterproductive. It is not our goal to measure the influence of trust in general, just for this specific case. However, the part of the system that decides whether to trust someone or not is general, it is not limited to bats or food and only assumes that something is shared (and what that is, does not matter).

4. **Abstraction:** Where the ecosystem will not be very general it will be abstract. As described in the previous points, the only interest lays with the trust system. Due to a time constraint and the massive task it would be to model a complete bat, only relevant systems are modelled. Moreover, additional systems could mean that there is more noise in the data, as these systems will probably have an influence on the simulated bats.

5. **Structural Accuracy:** The system that will be presented is not structural accurate. The system is developed so that it reflects the behaviour performed by vampire bats and uses the same constraints, but the internal mechanisms are represented in a different way. This will be further explained in Section 2.1.

6. **Performance Match:** In a perfect situation the system will perform perfectly the same as a real vampire bat, however, I expect that this will not be the case. The level of abstraction and the structural mismatch will probably have an effect and make the results differ from real world observations. On the other hand, I hope to find an similar match in performance in terms of improvement.

7. **Medium:** The system will be tested using computer simulations which will be explained further in the Methods section.

1.2 Reciprocal altruism, Reciprocity and Reciprocal Cooperation

Cooperation between relatives is favoured by natural selection. When a parent helps a child, for example by giving it some of its food, the chances of survival of that child increase as do the chances for the parent of successfully passing through its genes. Cooperation between non-relatives or even other species has at first sight no such clear advantage for both parties; it benefits the one that needs help but not the other one. Donating some of your meal to the neighbour's offspring does not directly improve something for you. It would only help you if, in the long or short run, you would gain something from your 'altruistic' act. Trivers [9] calls this *reciprocal altruism* and argues that cooperation between non-relatives can in fact be favoured by means of reciprocal interactions.

1.2.1 The Naming Problem

The problem with this term is that, depending on your point of view, it is not altruistic [11]. The behaviour, which we want to define and that will be used in this thesis, can be described as:

> *An agent A donates some of its resources to agent B, which as a result lowers the fitness of A but increases the fitness of B. At a later stage, B returns the favour which increases the fitness of A and likewise lowers the fitness of B.*

To correctly name this behaviour we must ask our self: is this behaviour altruistic? Hamilton [4] defines behaviour altruistic when it implicates a loss for the actor and a gain for the recipient. As agent *A* loses resources it is plausible to term this 'altruistic'. However, in the future, agent *A* gains from its initial act as it receives from *B*.The net loss (or profit) for both agents is zero, is this still altruistic?

The choice whether or not to name this behaviour altruistic depends on the timeframe of the behaviour, whether you only look at short-term influences (first interaction) or the long-term (both interactions). I reckon that, when you look at both interactions separately, you cannot say that the behaviour is reciprocal, there is nothing done in return. The only possibility is to look at the long term, which shows that the reciprocating behaviour has a benefit for both agents.

This would lead to the conclusion that the behaviour is not really altruistic and that *reciprocal altruism* is not the best term. Unfortunately, other terms (*reciprocity* and *reciprocal cooperation*) also do not quite fit the bill. What the behaviour makes different from normal cooperation is that the act has a negative effect for the performing agent. It is not just doing something in return, it is doing something in return that is negative for you (until you receive something back).

Brosnan and de Waal [2] partially address this by making a distinction between low-cost and high-cost reciprocity. Low cost reciprocity is that of services and losing an opportunity. High-cost reciprocity, which Brosnan and de Waal call altruistic, corresponds to the behaviour defined above.

No matter what term is used, it is questionable whether the performing agent is able to understand its behaviour. We as humans will see the use of helping someone and can predict how beneficial it is for us. But are animals who perform this behaviour also able to do this? And in this particular case, do vampire bats understand it? If not, it is acceptable that the behaviour could be called altruistic as the agent is not aware of the long term benefit. On the other hand, if it is not aware of the long-term benefits, is it aware of the short-term loss?

To conclude, there is probably no single term for this behaviour that is without discussion. A perfect name should incorporate both the altruistic aspect of the short term, the non-altruistic aspect of the long term and maybe even the perspective of the performing agent. As there is no such term available, in this thesis, *reciprocal altruism* is used as it is consistent with many of the literature regarding the vampire bat.

1.2.2 Criteria of Reciprocal Altruism

Wilkinson [15] mentions five criteria which should be met before behaviour qualifies as reciprocal altruism:

1. "The behaviour must reduce a donor's fitness relative to the selfish alternative."
2. "The fitness of the recipient must be elevated relative to the non-recipient."
3. "Performance of the behaviour must not depend on receipt of an immediate benefit."
4. "A mechanism for detecting individuals who receive benefits but never pay altruistic costs has to exist."

5. "A large but indefinite number of opportunities to exchange aid must exist within each individual's lifetime."

The first two criteria require the behaviour to be altruistic or high-cost reciprocity. If the behaviour is immediately advantageous for the donor it cannot be called altruistic. The third criterion divides reciprocal altruism from mutualism, cooperation where both parties benefit. Wilkinson comments to this third criterion that the time between encounters should be large enough for an agent to cheat. If the time is shorter the two encounters would better be considered one.

Criterion five corresponds to the iterated prisoner's dilemma. If the number of opportunities is known the best strategy is to cooperate except for the last move, defecting has a higher benefit if it has no consequence. An agent could cooperate at all moves but defect the last, this would result in the highest outcome. If that strategy is determined, the best counter strategy is to cooperate expect for the last two moves as you know that the other agent will not cooperate at the last move, eventually both agents will behave egoistic.

1.3 Vampire Bat

Vampire bats, who are named after the folkloric vampires, are small blood drinking bats who live mostly in the regions of South America. During the night vampire bats hunt for food using infrared radiation from their potential targets, who are usually livestock (and occasionally humans).

1.3.1 Diet of Blood

As blood consists mostly of water, vampire bats need a lot of it to fulfil their nutritional needs. Experiments with captive vampire bats show that they can easily consume 33% of their own weight in blood [17]. For non-captive bats it is a lot more difficult to measure how much blood they drink, these creatures start to excrete only moments after starting to feed. Simply measuring differences in weight will not give a clear result.

Nonetheless, calculating an estimate is possible. When weighing bats as soon as they return to the nest, taking excretion statistics from captive bats result into account, the original blood consumption can be derived. For natural vampire bats the average consumption of blood lays at 53% of their own body weight and differs a lot from bat to bat with values ranging from 15% to 132%.

1.3.2 Regurgitation

What is truly interesting about their blood drinking behaviour is what they do with the blood after they have fed. Research on the common vampire bat (the *desmodus rotundus*) showed that bats younger than two years have a 30% chance to fail in finding a sufficient meal. For mature bats this is lower, but still 7% [16]. It is important to note that there were little bats who failed repeatedly in finding food (in contrast to other bats) and therefore did not bias the data. Also the failure to find an appropriate meal was not synchronised within the colony [12]. In other words, not all bats fail at the same time in finding food.

Vampire bats can only survive for no more than 48 to 60 hours without food. Based on the probability of finding an appropriate meal, it seems unlikely that these bats, especially the females, can reach the observed ages of up to eighteen years. Wilkinson noticed from these numbers that the annual mortality, based on the probabilities, should be 82% whereas the observed mortality with real bats is only 24%. So what happens?

1.3.3 Reciprocal Altruism

When a bat has sufficient blood available it regularly, when certain criteria are met, donates blood to a less fortunate roost mate. As blood is difficult to carry bats drink it immediately, therefore this sharing is done by regurgitation. To show that the behaviour qualifies as reciprocal altruism we compare it with the five criteria of Wilkinson (see Sec. 1.2.2):

1. **"The behaviour must reduce a donor's fitness relative to the selfish alternative":**
A donor bat regurgitates about 5 millilitres of blood and with that action loses approximate six hours of life time. When it would choose for the selfish alternative it would have more time to find a new food source.

2. **"The fitness of the recipient must be elevated relative to the non-recipient":**
The recipient gains about 18 hours due to the donation and therefore benefits more than the donor. This is the result of a non-linear relation between body weight and time. The more food an agent has, the higher is its decrease in body weight per hour.

3. **"Performance of the behaviour must not depend on receipt of an immediate benefit":**
The donor bat cannot receive blood from the recipient because the recipient has not enough food to share, otherwise it wouldn't need the donation in the first place. Also, when the recipient immediately responds with a donation the agents would return to the original situation.

4. **"A mechanism for detecting individuals who receive benefits but never pay altruistic costs has to exist":**
It has been suggested that vampire bats use grooming to detect cheaters [14]. During grooming bats would inspect stomach sizes of other bats and with that keep track of the feeding records.

5. **"A large but indefinite number of opportunities to exchange aid must exist within each individual's lifetime":**
The social structure of female vampire bats is fairly stable [13]. Female offspring stay in their natal group and usually only move when their mother moves or dies.

2 Method

The system will be modelled and tested in a virtual environment that resembles a simplified version of the environment vampire bats live in. The bats are modelled using the *Python*[1] language and the open-source software *Breve* [7] which has also been used to run the simulations. Some inspiration has been taken from existing Breve simulations (including [8]) though their influence was small as these were often written in *Steve*, the standard language used in Breve.

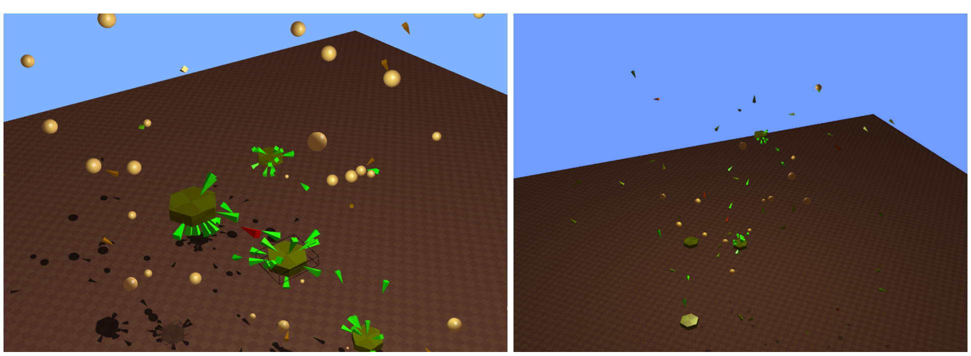


Figure 1: Screenshots of the simulations. Balls represent food sources. Coloured polygon cones are agents. An agent's colour shows his current energy level, which ranges from green (saturated) to dark red (starving). The polygon disks are nests.

[1]Python Programming Language: `www.python.org`

The environment consisted of a 3D world with a fixed size and agents were only able to navigate within that area. Each simulation used the same 3D world, an example can be seen in Fig. 1. Before the simulation started the world was filled with the following objects:

Nests: Each world contains a fixed amount of nests (or roosts). Agents return to the nest after feeding or when the night ends. At start, as there were four nests, each nest contains a quarter of all agents.

Food sources: To stay alive agents must find food sources and extract food from them. When a food source is depleted it is removed from the world. After each night the food sources are reset and placed at (new) random locations.

Agents: Each simulation consists of a certain population of agents, the distribution of the type of agents depends on the simulation. Agents can die but no new agents can be born, so the maximum amount of agents is fixed.

2.1 Agents

All agents in the simulations share a set of common behaviours, these include exploring, locating food, feeding, returning and leaving the nest and dying. These common or *base* behaviours are implemented in a *base agent* which has been used as an abstract class for the other agents. By further implementing this base agent in total three different types of agents were created to live in the environment:

Control Agents only contain the behaviours included in the base agent. They will try to find food but will never share food with another agent.

Trust Agents resemble real vampire bats in the sense that they have the ability to trust and are able to donate and receive food from other agents.

Cheating Agents will never donate food to another agent. Instead, cheating agents will try to receive as much food possible (from other agents).

2.1.1 Basic Behaviour

The most important influence on the basic behaviour of agents is the day-night cycle. When the night falls agents will leave their nest and start searching for food and, whether they have found food or not, return to the nest before sunrise.

When wandering through the world an agent can detect a food source when it is in the line of sight of the agent. Each agent has a maximum view distance and a two radians wide angle ($\pm$ 114 degrees) it can detect objects in. When a food source is detected and not yet occupied by another bat an agent will fly to the source until it is within feeding distance. After it arrived it will slowly extract food from the source until the source is depleted or the agent is fully saturated.

Besides returning to the nest at dawn, there are two more situations in which an agent returns to the nest. First of all, if an agent is fully saturated, because of a successful hunt, it will stop exploring and return to the nest. This is advantageous over continuing with flying as resting in the nest consumes less energy. The second occasion is when an agent is almost dying, and its only chance to survive is to beg for and receive food from a donator. Trust and cheating agents will return to the nest if their energy drops below a certain threshold (4% of maximum energy).

The energy consumption of all bats is based on the decrease in body weight of real vampire bats (see Figure 2) [12, 16]. The non-linear relation between weight and time after feeding formed the basis for a table of energy consumption values (see Figure 3). These values were then tested in the simulation by a group of control agents and scaled to let agents survive for approximately 60 hours without food. This incorporates the difference in energy consumption due to resting in nests.

All agents will start within a given nest in the world and are assumed to be females as they have a stable social structure. On average a bat moves to another nest each seven days (as used by [15]). Each nest has

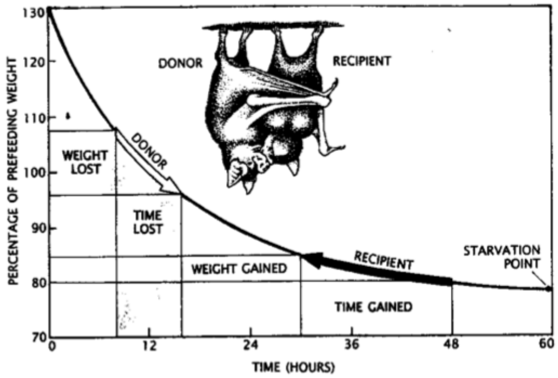


Figure 2: Relation between body weight and time since last feeding. The graph shows the non-linear decrease of a vampire bat's weight over time. Figure by Wilkinson [16].

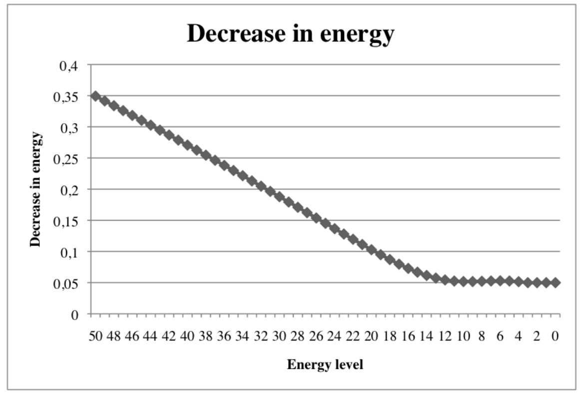


Figure 3: Plot of the decrease in energy for the simulated bats. The values are derived from the relation between the body weight and time since last feeding.

the same chance of being chosen and this moving behaviour is simulated by assigning new nest ids to bats after they leave their nest with a probability of $\frac{1}{7}$.

2.1.2 Behaviour of Trust Agents

Trust agents inherit all basic behaviours but have the extra ability of trust. The most important aspect of a trust agent is its own limited memory, which it can use to store information about other agents in its surroundings. For each agent it can store the *association*, *share rate* and *foraging success*. Only when an agent has enough food available and the thresholds of all these values are met it will donate food to another agent. An agent can store up to 20 other agents in its memory. When an unknown agent donates food, the recipient will always remember that agent, regardless the memory limit. This is however unlikely to happen, the donor needs to know the recipient before it shares and it would be strange if that was not vice versa.

Association. When a trust agent returns to its nest, it will detect the other agents that are present. For each agent that is in the same nest it will increase the association value for that particular agent in its memory. Also, it will lower the association value of agents that are not in the same nest. If the trust agent sees a new and unknown agent, it will add it to its memory if there is enough space for it. It will always keep the agents with the highest association value. An agent is considered 'known' after several encounters.

Share Rate. Trust agents keep track of agents who they donate to and from which they receive food. When they donate food to another agent the share rate will be lowered. The recipient will increase the share rate associated with the donor. A trust agent will donate food to another agent (provided the other constraints are met) if the share rate is greater then zero. When the share rate is precisely zero (none or equal previous sharing interactions) the agent shares with a chance of 25%, so agents are slightly optimistic.

Foraging success. Wilkinson used foraging success in its simulation by associating each bat with a foraging success value. A bat didn't share when it's own foraging success was too low. In our simulations finding food is not a binary issue, an agent could find an abundant food source but also one with only a few drops of blood left. So instead of depending on foraging success, the agents shared when they had equal or more than 40% of their total energy level left.

In addition, foraging success is used to assess other agents. Each agent inspects the current food level of the other agents in the nest. This can be seen as a simplified form of grooming which real vampire bats use to assess the amount of blood in the stomach of another bat (as suggested by Wilkinson [14]). Each time an agent inspects the food levels of an agent x it combines this with information from previous encounters using the following formula:

$$\mathrm{fs}(x,t) = 0.8\,\mathrm{fs}(x,t-1) + 0.2\,\mathrm{fl}(x,t) \tag{1}$$

In this formula $\mathrm{fl}(x,t)$ is the current energy of agent x and $\mathrm{fs}(x,0) = 0$. An agent will only share with another agent if this value is equal or higher than 20% of the maximum energy.

Begging. When the energy level of a trust agent drops below 14% of the maximum energy it will beg other agents for food by approaching them one by one, each only once. This threshold is the same as with real vampire bats and translates to about 24 hours of lifetime (including resting). An agent only begs agents that are in the same nest and will not approach agents that are flying. If begging is successful the energy levels and memory of both the donor and recipient are updated. If all agents refuse to donate food to an agent it will stop begging for that night.

2.1.3 Behaviour of Cheating Agents

Like trust agents, cheating agents include all basic behaviour. In addition, a cheating agent is also able to beg for food when running low on energy. The part that makes a cheating agent 'cheating' is that it will never donate food to another agent. When another agent begs a cheating agent for food it will always refuse. It is important to note that agents cannot detect cheaters by any means other than its interactions. Even cheaters themselves do not know who the cheaters are and could easily beg one of them for food.

2.2 Simulations

The simulations are split into three parts, which all test a different use or part of the system:

1. **Utopia:** Resembles a perfect world, where everyone shares freely and food is abundant (30 food sources). The thresholds for sharing are all set to zero, which means that agents will share if they have food to spare regardless of the association, share rate or foraging success. In total 128 simulations have been performed (64 with control agents and 64 with trust agents) which gives information about 10240 agents.

2. **Life span:** Tests the influence of trust on the life span of agents. Groups of control or trust agents were placed in an environment with different amount of food sources (see Table 1). The food sources are used to determine the influence of the chances of finding food on the trust system. In total 100 simulations were ran with control agents (resulting in information of 8000 control agents) and 72 with trust agents (5760 agents).

3. **Cheaters:** The third block of simulation is used to test the influence of cheaters. A portion (see Table 2) of the trust agents was replaced with cheating agents ('Cheaters') or control agents ('Cheaters control'). The simulations with trust and control agents are used to determine the influence of the presence of another group, those with cheaters and trust agents to determine the influence of the cheating behaviour. All together, the behaviour of 2640 control, 10720 trust and 2640 cheating agents were simulated.

Table 1: Overview of the simulations in the life span block. Each simulation setup has a different amount of food sources and is tested by control and trust agents. The amount of agents (at start) is always the same.

Simulation block	Life span						Life span control					
Simulation number	1	2	3	4	5	6	7	8	9	10	11	12
Number of food sources	5	10	15	20	25	30	5	10	15	20	25	30
Number of trust agents	80	80	80	80	80	80	0	0	0	0	0	0
Number of control agents	0	0	0	0	0	0	80	80	80	80	80	80

Table 2: Overview of the different simulations in the cheaters block. A portion of the trust agents is replaced with cheating or control agents. The amount of food sources and the total amount of agents is the same for each simulation setup.

Simulation block	Cheaters					Cheaters control				
Simulation number	13	14	15	16	17	18	19	20	21	22
Number of food sources	20	20	20	20	20	20	20	20	20	20
Number of trust agents	76	72	60	40	20	76	72	60	40	20
Number of cheating agents	4	8	20	40	60	0	0	0	0	0
Number of control agents	0	0	0	0	0	4	8	20	40	60

Each simulation lasted one simulated year (219000 iterations, 365 day/night cycles) and always started with 80 agents. A day or resting period lasted at least 225 iterations, a night or hunting period maximal 375. For each simulation the same statistics were saved, variables that are not applicable are set to zero. At the end the agent type, lifetime, times it had a successful hunt, amount of food it received from others, amount of food it donated, times it begged for food and memory size are stored for each agent.

3 Results

The simulations with only control agents established a base line of performance. For each food source density the average chance of a successful hunt (finding food at night) by a control agent was calculated, these were respectively: 26% for 5, 56% for 10, 79% for 15, 88% for 20, 92% for 25 and 95% for 30 food sources. The distribution of the chance (per agent) can be found in Figure 4. As the number of simulated bats per simulation setting differed all figures display the number of bats in percentages.

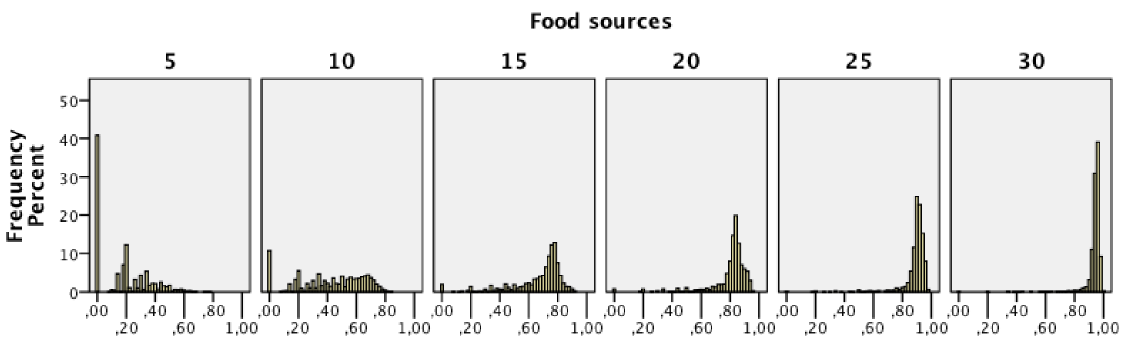


Figure 4: Distribution of the percentage of nights agents that found food.

3.1 Utopia

The results of the utopia simulations are, as expected, very clear (see Fig. 5). About 55% of the control agents and almost all trust agents (99%) survived the 365 simulated days. As the data is not normally distributed a Mann-Whitney U test was performed. The tests show a significant difference between the control (rank average of 4001,11 and a mean of 268) and trust group (rank average of 6239,89 and a mean of 365) with a p value of 0,000 (U = 7375903,0 and Z = -52,342).

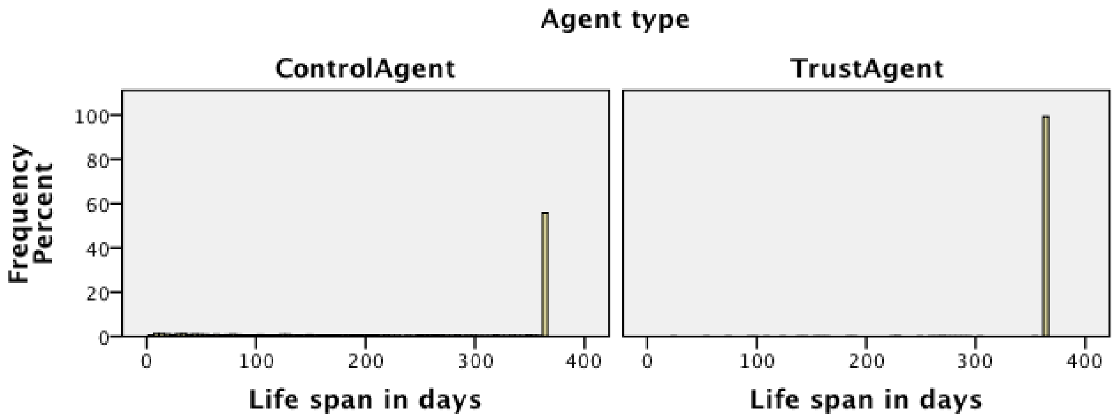


Figure 5: Distribution of the life span of control and trust agents in the utopia simulations. Sharing has a large influence on the life span of these agents. The utopia simulations used a fixed amount (30) of food sources.

3.2 Life span

When a trust system is introduced we see the life span increase in comparison to the control agents. Table 3 and Figure 6 display that, except for the low amounts of food, the trust system performs far better than the control system. When the amount of food sources rise we see that the trust system stabilises a bit due to the time constraint of one year. In the simulations an agent couldn't get any older than one year. Eventually, the control and trust agents will have the same average life span, when the amount of food rises so that the chance of finding food is 100% no agent will ever die.

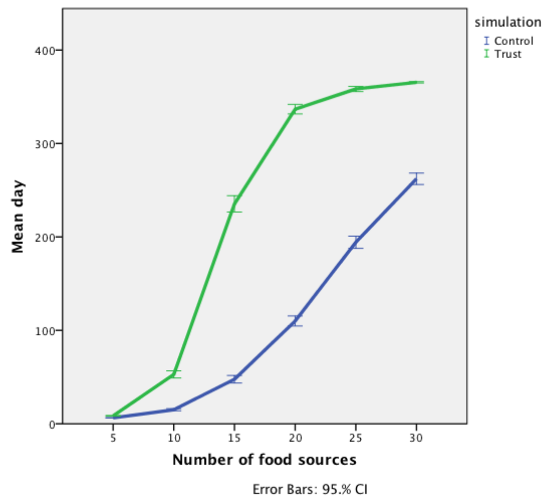
Table 3: Mortality rate of trust and control agents in the life span simulations per food source.

Food sources	5	10	15	20	25	30
Trust agents	100%	99,8%	57,4%	14,8%	4,1%	0,3%
Control agents	100%	100%	99,0 %	85,1%	70,6%	46,3%

As the data is not normally distributed (see Figure 8) a Mann-Whitney U test has been performed for each combination of food source density between control agents and trust agents. The results of the trust agents differ significantly from the control agents, as seen in Table 4. The effect sizes are the largest with 15 and 20 food sources. Figure 7 confirms this, the amount of times trust agents beg and receive food is the largest at a food density of 15 and second for 20.

Table 4: Comparison of the life span in days of control versus trust agents in relation to the amount of food sources.

	N		Mean		Mean Rank					
Food	Control	Trust	Control	Trust	Control	Trust	U	Z	p	Effect size
5	880	960	6	8	771,69	1056,91	291445,5	-11,745	0,000	-0,273
10	880	960	15	53	666,89	1152,98	199223,0	-19,624	0,000	-0,457
15	960	960	48	235	597,27	132,73	112098,0	-28,860	0,000	-0,658
20	1760	960	110	337	965,73	2084,24	150008,0	-36,363	0,000	-0,697
25	1760	960	194	358	1036,95	1953,67	275354,0	-31,514	0,000	-0,604
30	1760	960	262	365	1140,37	1764,08	457365,0	-24,430	0,000	-0,468



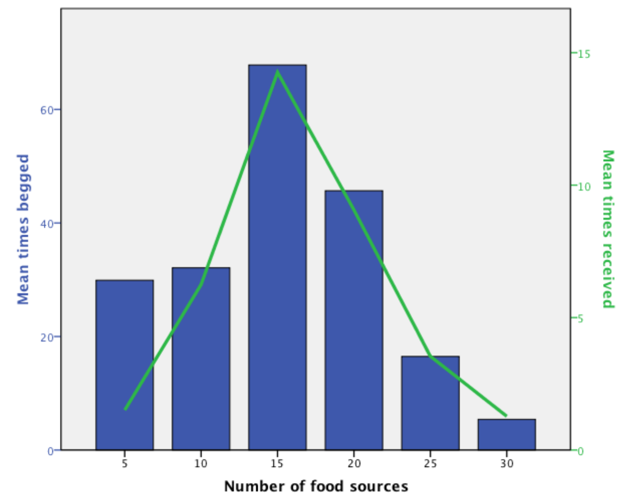


Figure 6: Average life span in days of control and trust agents in environments with different amount of food sources. The difference between the two groups is the smallest around the edges, here the trust system is not needed or is not able to lift the performance of the agents.

Figure 7: Relation between the amount of food sources and the times trust agents beg (bars) and receive food (line). The peak of both values lays in the middle which can explain why the effect between the two groups is the largest around these numbers.

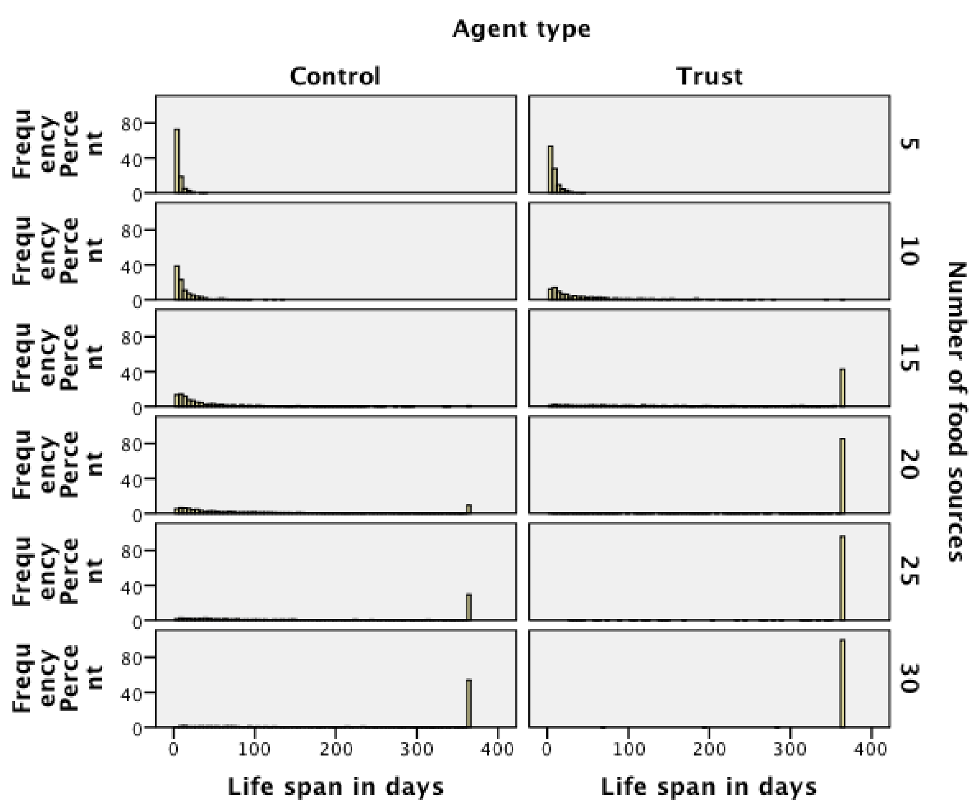


Figure 8: Distribution of the life span of control and trust agents for different amount of food sources. The histograms show that the data is not normally distributed which guided the choice of using Mann-Whitney U tests for the statistical analysis.

3.3 Cheaters

The influence of control agents on a population of trust agents can be seen in Figure 9. It is clear that trust agents perform better than the control agents but that the control agents, when their number rises, have a negative influence on the population of trust agents. Nonetheless, the control agents do not have such an influence that the trust agents perform worse or equal. The difference between trust and control agents is significant for all five cases (see Table 5).

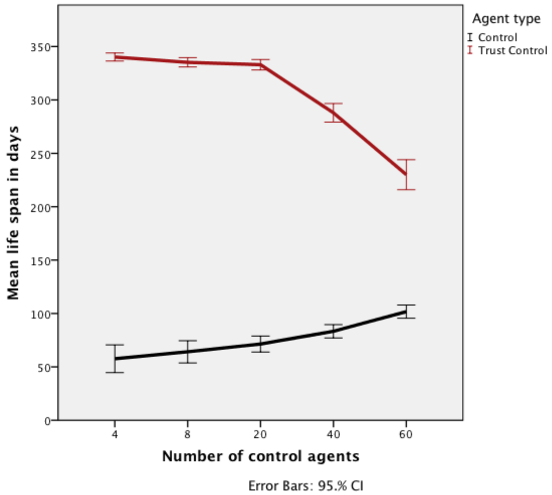


Figure 9: Average life span in days of control and trust agents together in the same simulation. The amount of control agents was varied and shows the influence of a growing competing group

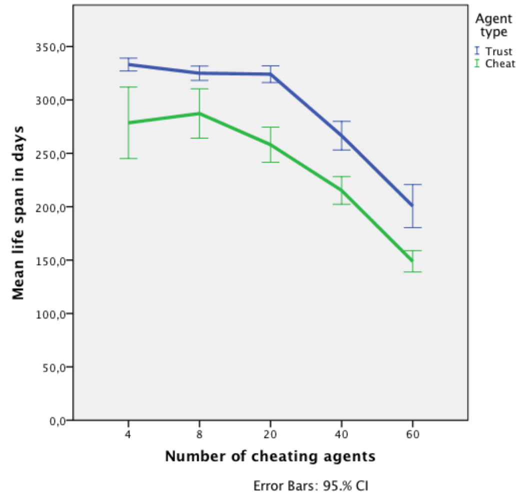


Figure 10: Average life span in days of trust and cheating agents placed together in the same simulation. The influence of a growing cheater population is quite clear.

Results of cheating agents show that their survival rate is a lot higher than that of control agents and that their presence, like the control agents did, influences the trust agents. An overview can be seen in Figure 10. There seems to be a tipping point around 20 (25%) cheaters or control agents, the average age of trust agents begins to start declining a lot faster after this point. Cheating agents perform, on average, less than trust agents, though better than control agents, and their expected lifespan lowers when their numbers grow. The effect sizes (see Table 6) are not very high (ranging from 0,15 to 0,35), especially in comparison to the effect sizes from the comparison of trust agents with control agents.

Table 5: Results of control agents versus trust agents who were placed together in the same environment. Although the control agents have a negative influence on the trust agents there is still a large difference between the two groups.

Control	N		Mean		Mean Rank					
agents	Control	Trust	Control	Trust	Control	Trust	U	Z	p	Effect size
4	80	1520	58	340	82,92	838,27	3393,5	-20,932	0,000	-0,523
8	160	1440	64	335	127,61	875,27	7537,0	-25,951	0,000	-0,648
20	400	1200	71	333	250,66	938,78	20064,0	-31,597	0,000	-0,789
40	800	800	83	288	501,48	1099,52	80782,5	-26,475	0,000	-0,661
60	1200	400	102	230	702,48	1094,57	122374,0	-14,371	0,000	-0,359

The difference between trust agents coexisting with few and many cheaters can be used to compute the effect of a growing cheater population. A Mann-Whitney U test was performed on the means of the trust agent population in the 5% and 75% cheaters condition. This showed a fairly large effect of -.646 (U = 1547058,0, Z = -51,687 and p = 0,000). The effect size between trust agents combined with 5% or 75% control agents is -0,428 (U = 166768,0, Z = 18,751 and p = 0,000). A comparison of trust agents who coexist with cheating agents and trust agents together with control agents shows differences that are not always significant and have very small effect sizes (see Table 7).

Table 6: Results of cheating and trust agents placed together in the same environment. The effect sizes are smaller than with control agents but the difference is still significant for all cases.

Number of cheaters	N Cheat	N Trust	Mean Cheat	Mean Trust	Mean Rank Cheat	Mean Rank Trust	U	Z	p	Effect size
4	40	760	279	333	263,34	407,72	9713,5	-5,612	0,000	-0,183
8	80	720	287	325	322,26	409,19	22541,0	-4,287	0,000	-0,152
20	200	600	258	324	287,39	438,20	37378,0	-9,889	0,000	-0,350
40	400	400	215	267	352,85	448,15	60939,0	-6,096	0,000	-0,216
60	600	200	149	201	380,67	460,00	48100,0	-4,220	0,000	-0,149

Table 7: Comparison of the life span of trust agents coexisting with cheating agents ('cheat') versus trust agents together with control agents ('control'). The displayed results are only of trust agents; cheaters and control agents are not included.

Other Agents	N Cheat	N Control	Mean Cheat	Mean Control	Mean Rank Cheat	Mean Rank Control	U	Z	p	Effect size
4	760	1520	333	340	1118,60	1151,45	560958,0	-1,786	0,074	-0,037
8	720	1440	325	335	1039,02	1101,24	488531,5	-3,305	0,001	-0,071
20	600	1200	324	333	881,94	909,78	348861,5	-1,600	0,110	-0,037
40	400	800	267	288	562,22	619,64	144688,0	-3,148	0,002	-0,091
60	200	400	201	230	278,44	311,53	35588,5	-2,277	0,023	-0,092

3.3.1 Trust statistics

Cheaters receive less food than the two groups of trust agents do (see Figure 11) and when the amount of other agents rise (whether these are cheaters or control agents) the amount of food received by all agents drops. The trust agents in the control condition receive less food than the trust agents who coexist with cheaters, these trust agents do however also beg a lot less (see Figure 12).

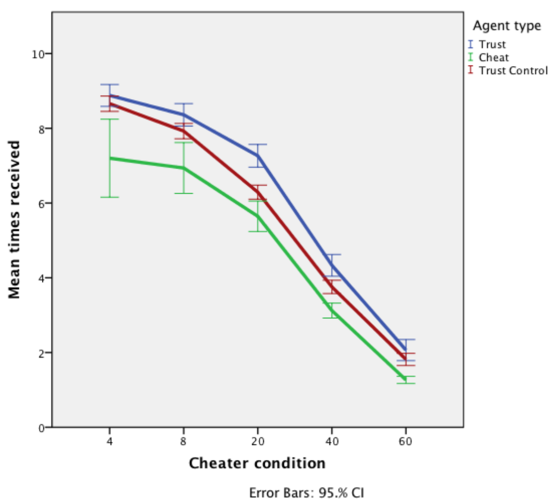


Figure 11: Average amount of times an agent received food. These averages decrease when the amount of other agents rise which could give an explanation why the life span of agents drop. Trust agents who where placed in an environment with cheaters have a higher average than agents that are placed in an environment with control agents.

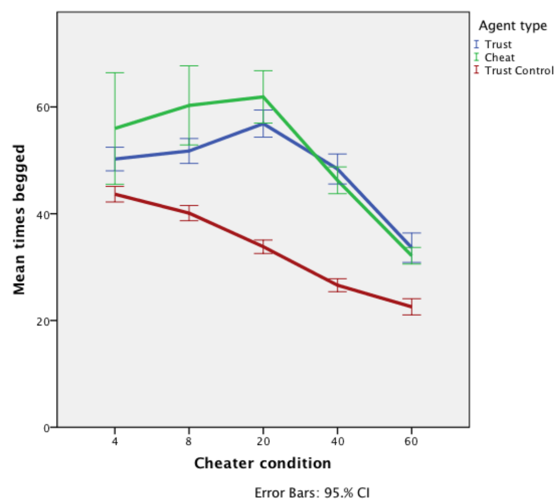


Figure 12: Average amount of times an agent begged another agent for food. That the control trust agents beg less than the two other groups could be caused a larger group of possible recipients in the cheater condition. Available food has to be shared with more agents than in the control condition, but the amount of potential donors is the same.

In Figure 12 we can see that the times agents beg decreases when the amount of cheaters rise, possibly due to the shortened life span. However, the amount of times agents beg per day (total amount divided by the amount of days they lived) increases when the amount of non-trust agents becomes larger (Figure 13). So, although the decrease in lifespan lowers the amount of possibilities agents have to beg, this is not enough to compensate for the effect of a growing population of competitors, which requires agents to beg more.

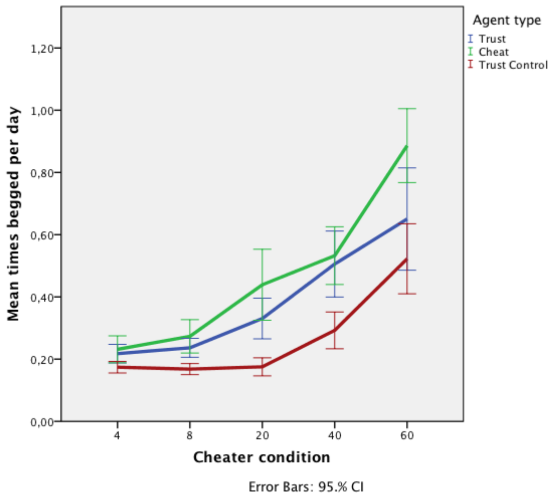


Figure 13: Average amount of times an agent begged per day. When the amount of cheaters increase, which lowers the amount of agents that can donate, the times agents beg increase.

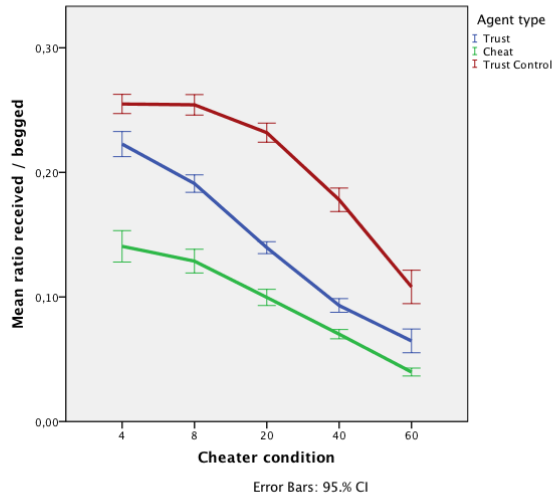


Figure 14: The ratio of the times an agent actually receives food after begging. Cheaters have the lowest ratio; from the three groups they are refused the most.

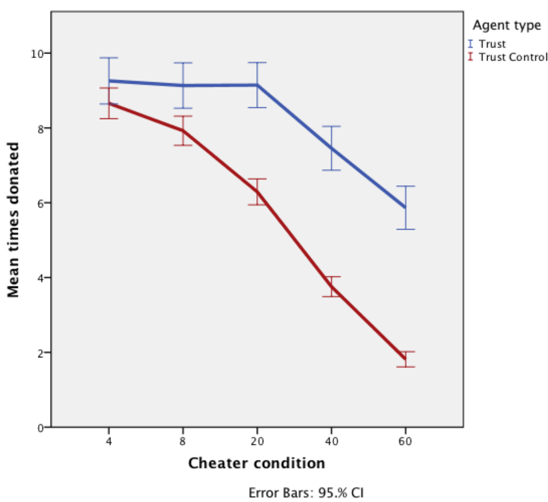


Figure 15: Average amount of times an agent donated in its whole lifetime. Both lines drop but trust agents in the control condition donate less.

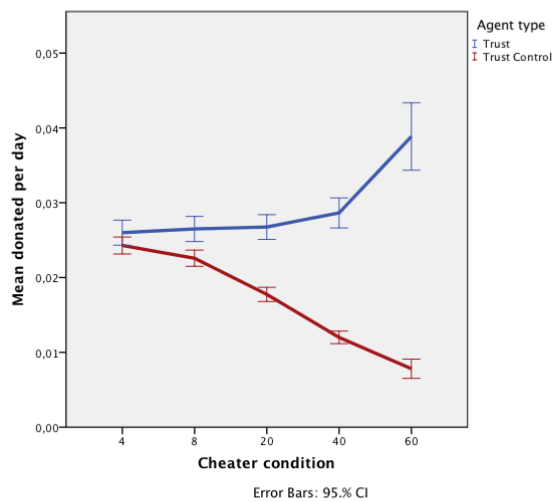


Figure 16: Average amount of times an agent donated per day. The trust agents in the control condition start donating less when the amount of cheaters rise, on the other hand, trust agents in simulations with cheaters start donating more per day.

Figure 14 shows the ratio between begging and receiving for the three groups of agents (control agents cannot beg or received food so are not shown). The ratio drops when the amount of competing agents increases, this is a logical result from the previous figures which showed an increase in begging and a decrease in receiving. Cheaters' begging is answered the least, they have the lowest ratio, so the trust system is able to prevent cheaters from receiving more food than agents who cooperate. The trust control group (trust agents with control agents) has a larger ratio than the trust agents in the cheater condition; probably some food is lost to cheaters.

Evaluating the number of times trust agents donate we see that their is a growing difference between agents in the control group and agents confronted with cheaters (see Figure 15). Agents in the cheater condition donate more food to other agents, partly to other trust agents but also to cheating agents. When the amount of cheaters increase there are more mouths to feed and less donating agents, therefore an increase in donations is expected. Interestingly, where the total amount of donations decreases, the amount of donations per day increases for trust agents that lived in a world together with cheaters (Figure 16). For the agents in the control condition this is not the case, their average donations per day decreases.

4 Discussion

Agents with the ability of trust live significantly longer in comparison to a group without this ability. The size of the effect relates to the amount of food sources and an optimum lays likely between 15 and 20 (a 79% and 88% chance of finding food), here agents still have a hard time finding a steady food supply, but the trust system can compensate for this lack. This optimum lays below the 93% chance of finding food (mature) bats have in real life and the 90% chance Wilkinson used in his simulations.

The utopia simulations were used to see the effect of a perfect world, where every trust agent shares freely. The differences between the trust agents in the utopia simulations and those in the life span simulations are small; the mortality rate for the 30 food sources condition is practically the same. At least for that condition the trust system performs equal to a non-restricted sharing system.

4.1 Introducing another group

When we combine trust agents with control agents in the same environment we saw that the life span of trust agents stayed a lot higher than that of control agents. However, in contrast to a population with only trust agents there is a decrease. In a second condition control agents were replaced with cheating agents to measure the influence of their cheating behaviour, i.e. beyond their mere presence. The difference between the trust agents and the cheaters remains significant although the effect size is a lot lower than with control agents. Almost all cheaters perform better than control agents. Finally, on average a cheater lives shorter than a trust agent, although, there are a few cheaters who live equal to or even longer than the trust agents.

For both cases of increasingly added agents, when the numbers of the other agent rise, the average life span of trust agents declines. This can be explained by three (not mutually exclusive) causes:

1. When the number of the other agents increases trust agents have a smaller chance of finding one another and when they find another trust agent it must also be willingly to share some of its food. It is coherent that when the amount of trust agents drops these factors have larger effects.

   We see this in the results by looking at the amount of donations done by agents. Especially in the control condition, an increase in the other competing group results in a strong decline in the number of donations (see Figure 15). Also, the amount of times an agent begged per day increased, which resulted in a lower receive / begged ratio (Figure 14)

2. When a trust agent finds food, it will help him, but possibly also another trust agent through a donation. Because a starving bat benefits more from a donation than that the donor loses, the food source has the potential to be ‘more’, in terms of hours of life span, than its initial value. In other words, donating to a fellow trust agent is an investment in the population. When there are fewer donations, because trust agents, for example, do not know each other, this has a negative effect on the population.

3. From the perspective of the population, it is counterproductive that some agents do not share their food, this is a decrease in the potential availability of the food. But the population suffers even more when some of the collected food is shared with cheaters because there is an additional diminishing of

the available food. This donation will not only lower the fitness of the donor, it will also decrease the fitness of the whole (trusting) population, as the energy is 'lost'. In the simulations this has occurred, cheaters receive food from trust agents, especially because they are in the beginning not yet classified as cheaters.

4.2 The influence of cheating

The control agents showed us that the mere presence of another group could already influence the life span of trust agents. So what portion of the influence of cheaters is caused by cheating and what by their presence in the world? As cheaters keep all the food they collect for themselves and get some help during rough nights it is pretty straightforward that they have a higher chance of survival than control agents.

For trust agents it is beneficial that the other agents die as quickly as possible, this increases the food/agent ratio, but due to the longer life span of cheating agents they will use the available food for a longer period than control agents do. Also, because they share more nights with trust agents they have more opportunities to beg for food and possibly receive some food that was better used, from the populations perspective, for starving trust agents.

These two aspects together we can call *'the influence of cheating'* as it is the direct effect on trust agents due to the cheating behaviour. This influence becomes larger when the amount of cheaters rise, the effect size between 5% and 75% cheater is -0,646 where the base level is -0,428 (from the control agents). The effect sizes of the difference between the two groups of trust agents (those living together with cheaters and those that live together with the control agents) are, however, only ranging from -0,03 to -0,09.

Apparently there is only a small difference between the two groups of trust agents. Cheating agents behave a bit like parasites, they need the trust agents to survive (longer) but their cheating *behaviour* does not have a big influence on the population of trust agents. This is caused by two factors: sharing is not always required and the begging of cheaters is not always successful. Therefore, the absolute quantity of food lost to cheaters is minor.

4.3 Future Research

The simulations conducted for this thesis give a good insight in the working and performance of the sharing and trust system of the vampire bat. However, the simulations were limited and concessions had been made in terms of biological plausibility and structural accuracy, which made the results differ from the observed results by (for example) Wilkinson. With a 92% chance of finding food the mortality rate for the simulated trust agents is only 4,1% where with real bats this is around 24%. In other words, the created and simulated ecosystem is not a perfect model of the behaviour and environment of the vampire bat.

A first improvement that can be made to these simulations is trying to minimise the difference between the model and the real system, this would make it easier to translate the specific results from the simulations to real vampire bats. For this to be accomplished it could be possible that more aspects of the vampire bats have to be included in the simulations, for example behaviours that are not directly related to the trust system but who do have an effect on the amount of time agents have to hunt or interact with each other.

From a biological point of view it is interesting how such a trust system evolved over time and how it achieved to be such an influential behaviour. The introduction of offspring and an evolutionary algorithm could give insight in this. Not only the trust behaviour but also the cheating behaviour could be included in this research. Will cheating agents eventually disappear, because cooperating is the optimum policy, or could a mixed form between sharing and cheating be formed? Possibly, when the food supply contained more variation, it would be better to cooperate in times of abundance and defect when the chance of finding food is low.

Second, in the current version of the simulations, agents are not aware of sharing interactions within their vicinity. It could however be possible that bats are in fact aware of this and use that to assess the trustworthiness of other agents. An agent could decide not to donate because that particular agent received to much in relation too how much it donated. This could be an effective addition to the trust system for preventing cheaters.

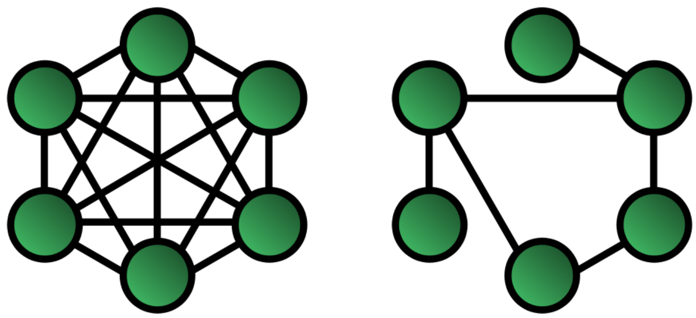


Figure 17: A mesh network is often not fully (left) but only partially connected (right). *(Source: GW Simulations / Wikimedia Commons / Public Domain)*

A third approach to improving these simulations is that of optimisation. In what conditions does the system perform the best? We saw that the amount of sharing interaction was highest around 20 food sources and lowest on the edges. When we look at cheaters, how many cheaters can be present in the environment before the population collapses and the life span drastically decreases, so what is the true tipping point in terms of population distribution? These optimums are interesting for possible applications of the system as they could result in a higher performance.

4.4 Application

The results from the simulations showed us that the trust system works an is able to extend the life span of agents but also prevents cheaters of having too much of an influence. From a more practical aspect the question arises how we could use our gained insights of the food sharing system for practical applications. Therefore we will look at one of the possible uses of this system: the field of (wireless) mesh networks.

At its basics, a mesh network is a set of nodes where each node acts as a relay. The network does not have to be fully connected (Fig 17) so a node could only be connected to one other node. When a node 'A' requests information from a node 'B' the information has to traverse through the network.

A mesh network can be used for various purposes but, in my opinion, the most interesting is when it is used to bypass existing networks or architecture or to create one where there is a lack. It is sometimes said that we wouldn't need "the Internet" if everyone would open up their mobile devices (smartphones, routers, etc.) to form one massive global mesh network. Wether this will be the future or not, (wireless) mesh networks are already used in current projects, for example in *one laptop per child*[2], *FreedomBox*[3] or *Byzantium*[4]. Whether their goal is to create a connection because there is no infrastructure or to protect the privacy and anonymity of surfers, they all use a decentralised system instead of the centralised system we use today.

When these projects become larger, too large to know everyone in a mesh, or when users want to be totally anonymous it becomes difficult to regulate bandwidth. 'Free-riding' or cheaters are often very difficult to stop or prevent [3]. Solutions, for example the 'community approach' by Antoniadis et al. [1] cannot be used when anonymity is required. When bandwidth is scarce, you only want to open up your node to people who would do the same for you, here the sharing system of the vampire bat could come in handy.

The analogy is quite easy to make. Each user (a node) can be represented as a bat that is part of a nest (sub network). A user has a fluctuating amount of bandwidth available which depends on the connections with other nodes and external systems[5]. When a user's bandwidth starts running low the value for bandwidth

[2]Website of *one laptop per child* project: `one.laptop.org`

[3]Website of *FreedomBox*: `www.freedomboxfoundation.org`

[4]Website of project *Byzantium*: `www.hacdc.org`

[5]To give the whole mesh access to resources on the web, at least one of the nodes has to be connected to the internet.

will rise for that particular user, this translates to the non-linear relation between body weight and energy consumption of the vampire bat. A user that will be granted some bandwidth from a nearby node (which has plenty) will value that bandwidth higher than the donor.

The advantage of such a system is that it is decentralised and user based. There is no central authority that decides who can be trusted or not. The system is able to automatically trust and distrust others and make sure that cooperating is the optimal policy. There are, of course, also drawbacks of using this system, mainly because they need further research. For example, how will the system react on users changing their identity (which is a lot easier in a digital world)? Users with a freshly generated identity (also called *whitewashing* [3]) will be treated as new and this could give them the possibility to benefit more from cheating. So, before an application as this can be fully used, there needs to be more research to the effect of (selective) cheaters, something which lays outside of the scope of this thesis.

4.5 Conclusion

To conclude, the trust system of the vampire bat is a simple but effective decentralised sharing system that is able to elevate the fitness of a group of agents that use it. Agents get to know each other and decide, based on association and previous interactions, who to share with.

More research has to be performed to further improve the model before the results can one-to-one be translated to real vampire bats. However, as many of the features are directly derived from real behaviour, the results can, already in this state, give an indication of the effects of the trust system and cheaters on the life span of vampire bats.

The system works best when there is some deficiency in food supply and can really be of influence, even with low amount of sharing interactions between agents. While cheaters do have some influence this is mainly caused by their presence and not by their behaviour. If the number of cheaters does not get out of hand, trusters should be fine.

References

[1] P. Antoniadis, B. Le Grand, A. Satsiou, L. Tassiulas, R. Aguiar, J. Barraca, and S. Sargento. Community building over neighborhood wireless mesh networks. *Technology and Society Magazine, IEEE*, 27(1):48–56, 2008.

[2] S. Brosnan and F. de Waal. A proximate perspective on reciprocal altruism. *Human Nature*, 13:129–152, 2002. 10.1007/s12110-002-1017-2.

[3] M. Feldman and C. John. Overcoming free-riding behavior in peer-to-peer systems. *ACM SIGecom Exchange*, 5(4):41–50, July 2005.

[4] W. Hamilton. The genetical evolution of social behaviour. i. *Journal of Theoretical Biology*, 7(1):1 – 16, 1964.

[5] A. H. Harcourt. Help, cooperation and trust in animals. In *Cooperation and prosocial behaviour*, pages 15–26. Cambridge University Press, 1991.

[6] D. Harrison McKnight and N. Chervany. Trust and distrust definitions: One bite at a time. In R. Falcone, M. Singh, and Y.-H. Tan, editors, *Trust in Cyber-societies*, volume 2246 of *Lecture Notes in Computer Science*, pages 27–54. Springer Berlin / Heidelberg, 2001.

[7] J. Klein. Breve: a 3D environment for the simulation of decentralized systems and artificial life. In *Proceedings of the eighth international conference on Artificial life*, pages 329–334, 2003.

[8] L. Spector, J. Klein, C. Perry, and M. Feinstein. Emergence of collective behavior in evolving populations of flying agents. In *Genetic and Evolutionary Computation—GECCO 2003*, pages 200–200. Springer, 2003.

[9] R. L. Trivers. The evolution of reciprocal altruism. *The Quarterly Review of Biolog*, 46(1):35–57, March 1971.

[10] B. Webb. Can robots make good models of biological behaviour? *Behavioral and brain sciences*, 24(6):1033–1049, 2001.

[11] S. A. West, A. S. Griffin, and A. Gardner. Social semantics: altruism, cooperation, mutualism, strong reciprocity and group selection. *Journal of Evolutionary Biology*, 20(2):415–432, 2007.

[12] G. S. Wilkinson. Reciprocal food sharing in the vampire bat. *Nature*, 308:181–184, March 1984.

[13] G. S. Wilkinson. The social organization of the common vampire bat. *Behavioral Ecology and Sociobiology*, 17(2):123–134, 1985.

[14] G. S. Wilkinson. Social grooming in the common vampire bat, desmodus rotundus. *Animal Behaviour*, 34(6):1880–1889, 1986.

[15] G. S. Wilkinson. Reciprocal altruism in bats and other mammals. *Ethology and Sociobiology*, 9:85–100, 1988.

[16] G. S. Wilkinson. Food sharing in vampire bats. *Scientific American*, 262(2):64–70, 1990.

[17] W. A. Wimsatt. Transient behavior, nocturnal activity patterns, and feeding efficiency of vampire bats (desmodus rotundus) under natural conditions. *Journal of Mammalogy*, 50(2):pp. 233–244, 1969.