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

Bulten, W. a Haselager, W.F.G. b Sprinkhuizen-Kuyper, I.G. b

a Department of Artificial Intelligence, Radboud University Nijmegen, The Netherlands
b Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour, The Netherlands

Abstract

Vampire bats manage to live longer by trusting their fellow roost mates and getting donated to in return. We have modelled this interaction by creating a biological plausible decentralised trust and sharing system. In a simulated 3D environment the performance has been tested by groups of artificial bats showing 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 we have found 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 [1]. Related to this is what Trivers [4] calls reciprocal altruism: behaviour that benefits a not closely related organism while being disadvantageous for the donor. In this paper we will focus on the vampire bat and use computational modelling of altruistic food sharing.

Vampire bats live on a diet of blood, need a lot of it to fulfil their nutritional needs and can easily consume 33% of their own weight in blood [10]. Importantly, 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 chance to fail is still 7%. Also, within a colony, not all bats fail at the same time in finding food [5, 9].

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 can reach the observed ages of up to eighteen years (especially for females). 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%. How does this happen?

When a bat has sufficient blood available it regularly, when certain criteria are met, donates blood to less fortunate roost mates. Though this lowers the fitness of the donor it even occurs between non-relatives in a colony. Wilkinson defined five criteria that have to be met before behaviour can qualify as reciprocal altruism (instead of for example kin selection) [8], the vampire bat matches all of these:

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 will 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 a result of a non-linear relation between body weight and time. The more food an agent has, the higher 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 on the spot because the recipient has not enough food to share. 

A mechanism for detecting individuals who receive but never pay altruistic costs has to exist: It has been suggested that vampire bats use grooming to detect cheaters [7]. During grooming bats would inspect stomach sizes of other bats and use that to keep track of feeding records. 

A large number of opportunities to exchange aid must exist within each individual’s lifetime: The social structure of female vampire bats is fairly stable [6]. Female offspring stay in their natal group and usually only move when their mother moves or dies.

Wilkinson used simulations to calculate the benefit of altruistic acts. His Monte Carlo simulation used fixed association values between bats, fixed chances of finding food (always 0.9) and only 11 bats. It is our aim in this paper to, first, improve his simulation and, second, extend it by introducing a new group of agents: cheaters. A first block of simulations will be used to answer our 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 our second group of research questions: 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?

2 Methods

Two blocks of simulations were run where the first block of simulations test the performance of the system and tries to be biological plausible; the second block focuses on the influence of cheaters.

2.1 Environment

The system will be modelled and tested in a virtual environment that resembles a simplified version of the biome vampire bats live in. The bats are modelled using the Python language and the open-source software Breve [2] which has also been used to run the simulations. Some inspiration has been taken from existing Breve simulations (including [3]) though their influence was small as these were often written in the Breve programming language Steve.

![Figure 1: Screenshots of the setup showing food sources (balls), agents (polygons) and nests (polygon disks). An agent’s colour resembles the current energy level and ranges from green (saturated) to red (starving).](image)

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 Figure 1. Before the simulation started the world was filled with four nests, a variable amount of food sources and agents. Agents return to the nest after feeding or when the night ends. Before each hunting period food sources were reset (filled with blood) and placed at random locations.

2.2 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 a total of 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 or beg. **Trust Agents** resemble real vampire bats in the sense that they have the ability to trust another agent 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.

### 2.2.1 Basic Behaviour

The most important influence in the basic behaviour of the agents is the day-night cycle. When the night falls agents will leave their nest and start searching for food and will, 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 and fly to it when it is in the line of sight of the agent\(^1\). When a food source is detected and it is not yet occupied by another bat, an agent will fly to the source until it is within feeding distance and will feed until it is fully saturated or the source is depleted.

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 [5, 9]. The non-linear relation between weight and time after feeding formed the basis for a table with energy consumption values. 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 [8]). Each nest has 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 \(\frac{1}{7}\).

### 2.2.2 Behaviour of Trust Agents

Trust agents inherit all the basic behaviours but have the extra ability of trust. The most important aspect of the trust agent is its own limited\(^2\) 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 if the thresholds of all these values are met a trust agent will donate food to another agent.

**Association.** For each agent that is in the same nest it will increase the association value for that particular agent in its memory (with 0.3). It will lower the association value by 0.1 of agents that are not in the same nest. This way the relations between agents dynamically degrade or strengthen over time.

**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 is optimistic and will donate food to another agent (provided the other constraints are met) if the share rate is greater than zero. When the share rate is precisely zero (no previous sharing interactions) the agent shares with a chance of 0.25.

**Foraging success.** In Wilkinson's simulations 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.

---

\(^1\)Each agent has a maximum view distance and a two radian (± 114 degrees) angle it can detect objects in.

\(^2\)Agents can store up to 20 other agents in their memory. When an unknown agent donates food, the recipient will always remember that agent, regardless the memory limit.
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 [7]). Each time an agent inspects the food levels of an agent $x$ it combines this with information from previous encounters using the following formula:

$$fs(x, t) = 0.8 \cdot fs(x, t - 1) + 0.2 \cdot fl(x, t)$$  \hspace{1cm} (1)

In this formula $fl(x, t)$ is the current energy of agent $x$ and $fs(x, 0) = 0$. An agent will only share with another agent if this value is equal to 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 checking the number of bats in the nest, and approaching them one by one, each only once. 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.

### 2.2.3 Behaviour of Cheating Agents

Cheating agents include all basic behaviour and are also able to beg for food when running low on energy. The part that makes a cheating agent ‘cheating’ is that it will always refuse to donate food to another agent.

### 2.3 Simulations

The simulations are split into two main parts. The first (‘Life span’) tests the influence of trust on the life span of agents. Each simulation lasted exactly one simulated year (365 simulated days) and always started with 80 agents. Groups of control and trust agents were placed in an environment with different amount of food sources (see Table 1). Then 100 simulations were ran with control agents (resulting in information of 8000 control agents) and 72 with trust agents (5760 agents).

The second block of simulations measures 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. In total 150 simulations were ran.

<table>
<thead>
<tr>
<th>Simulation block</th>
<th>Life span</th>
<th>Life span control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation number</td>
<td>1 2 3 4 5 6</td>
<td>7 8 9 10 11 12</td>
</tr>
<tr>
<td>Number of food sources</td>
<td>5 10 15 20 25 30</td>
<td>5 10 15 20 25 30</td>
</tr>
<tr>
<td>Number of trust agents</td>
<td>80 80 80 80 80 80</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Number of control agents</td>
<td>0 0 0 0 0 0</td>
<td>80 80 80 80 80 80</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Simulation block</th>
<th>Cheaters</th>
<th>Cheaters control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation number</td>
<td>13 14 15 16 17</td>
<td>18 19 20 21 22</td>
</tr>
<tr>
<td>Number of food sources</td>
<td>20 20 20 20 20</td>
<td>20 20 20 20 20</td>
</tr>
<tr>
<td>Number of trust agents</td>
<td>76 72 60 40 20</td>
<td>76 72 60 40 20</td>
</tr>
<tr>
<td>Number of cheating agents</td>
<td>4 8 20 40 60</td>
<td>0 0 0 0 0</td>
</tr>
<tr>
<td>Number of control agents</td>
<td>0 0 0 0 0</td>
<td>4 8 20 40 60</td>
</tr>
</tbody>
</table>

### 3 Results

The simulation 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 chances were respectively: 26% for 5, 56% for 10, 79% for 15, 88% for 20, 92% for 25 and 95% for 30 food sources.

\[^3\text{This is the same as with real vampire bats and translates to about 24 hours of lifetime (including resting).}\]
3.1 Life span

When a trust system is introduced we see the life span increase in comparison to the control agents. Figure 2 and Table 3 show 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 stabilizes a bit due to the time constraint of one year. Also, because agents have more food to their disposal there is less need to beg others for food. This is also confirmed by the frequency of begging and donation, which is lowest at 30 food sources (see Figure 3).

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

<table>
<thead>
<tr>
<th>Food sources</th>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trust agents</td>
<td>100%</td>
<td>99.8%</td>
<td>57.4%</td>
<td>14.8%</td>
<td>4.1%</td>
<td>0.3%</td>
</tr>
<tr>
<td>Control agents</td>
<td>100%</td>
<td>100%</td>
<td>99.0%</td>
<td>85.1%</td>
<td>70.6%</td>
<td>46.3%</td>
</tr>
</tbody>
</table>

As the data is not normally distributed 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 size is the largest with 15 and 20 food sources, this is supported by Figure 3 which shows that for these food densities agents donate and beg the most.

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

<table>
<thead>
<tr>
<th>N</th>
<th>Food</th>
<th>Control</th>
<th>Trust</th>
<th>Mean in days</th>
<th>Mean Rank</th>
<th>U</th>
<th>Z</th>
<th>p</th>
<th>Effect size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5</td>
<td>880</td>
<td>960</td>
<td>6</td>
<td>8</td>
<td>771.69</td>
<td>1056.91</td>
<td>-11,745</td>
<td>-0.273</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>880</td>
<td>960</td>
<td>15</td>
<td>53</td>
<td>666.89</td>
<td>1152.98</td>
<td>-19,624</td>
<td>-0.457</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>960</td>
<td>960</td>
<td>48</td>
<td>235</td>
<td>597.27</td>
<td>132.73</td>
<td>-28,860</td>
<td>-0.658</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1760</td>
<td>960</td>
<td>110</td>
<td>337</td>
<td>965.73</td>
<td>2084.24</td>
<td>-36,363</td>
<td>-0.697</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>1760</td>
<td>960</td>
<td>194</td>
<td>358</td>
<td>1036.95</td>
<td>1953.67</td>
<td>-31,514</td>
<td>-0.604</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>1760</td>
<td>960</td>
<td>262</td>
<td>365</td>
<td>1140.37</td>
<td>1764.08</td>
<td>-24,430</td>
<td>-0.468</td>
</tr>
</tbody>
</table>

3.2 Cheaters

The influence of control agents on a population of trust agents can be seen in Figure 4. 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 difference between trust and control agents is still significant for all the five cases (see Table 5).

When we look at the results of the cheating agents we see that their survival rate is a lot higher than that of the control agents and that their presence influences the trust agents. An overview is shown in Figure 5.
There is 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.

When we look at the difference between trust agents living together with few and many cheaters we can 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 4 and 60 cheaters condition. This showed a fairly large effect of -0.646 (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 live together with cheating agents and trust agents living together with control agents shows differences that are not always significant and have very small effect sizes (see Table 7).

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 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 almost completely.

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 the 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 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.

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 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.

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 to 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 population as the energy is ‘lost’.

4.2 The influence of cheating

The control agents showed us that the mere presence of another group can 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.

It is beneficial for the trust agents when the other agents die as quickly as possible, 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 would have been 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 Conclusion

As the environment wherein the sharing of food was tested is limited there is room for future improvement and research. First of all, there is a structural mismatch between the agents in the simulations and real vampire bats. With a 92% chance of finding food the mortality rate for trust agents is only 4,1% where with real bats this is 24%. So from a biological aspect the simulations can be improved, for example by extending the environment or introducing more vulnerable group members, such as offspring. Such an improvement could also give insight how such a food sharing system emerged and evolved over time.

From a more practical aspect the question arises how we could use our gained insights of the food sharing system for practical applications. As the system is decentralised and able to create associations without prior knowledge, a possible application is that of (wireless) mesh networks. As these networks require users to share resources (traffic through their node) a trust system like that of the vampire bat can be used to decide whom to share with. Such an implementation will, however, need more research in the effect of cheaters, something which lays outside the scope of this paper.

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 make decisions who to share with. The system works best when there is some deficiency in food supply. While cheaters do have some influence, if their numbers does not get out of hand, trusters should be fine.

References


