Andreas Hagen

Evolving altruism through kin recognition in evolutionary robotics

Master's thesis, Spring 2014

Artificial Intelligence Group
Department of Computer and Information Science
Faculty of Information Technology, Mathematics and Electrical Engineering

NTNU – Trondheim
Norwegian University of
Science and Technology

Abstract

In this paper the different mechanisms that enable the evolution of altruistic behavior in evolutionary computation are explored. The research is motivated by the need for systems using embodied evolution to evolve relevant traits needed to avoid extinction in a dynamic environment. In particular interest is the subject of self-sacrifice where individuals relinquish the chance of reproduction altogether to assist others. The paper contains a structured literature review on altruism in simulated environments in artificial evolution. The literature review results in the question of whether or not kin-selection is a prerequisite for the evolution of self-sacrifice.

Preface

This report was written as a master's thesis in the spring of 2014 at the Norwegian University of Science and Technology. It was officially supervised by Pauline Haddow with assistance by Jean-Marc Montanier.

Andreas Hagen
Trondheim, June 17, 2014

Contents

Chapter 1

Introduction

One main area of interest in altruism in evolutionary robotics is within the field of multi agent systems, especially within embodied evolution where agents are continuously adapting to the environment. One of the long-term goals in this field is to create agile populations of robots that are able to create new solutions to previously unseen problems that arise and have the necessary mechanisms for ensuring the survival of already functioning genotypes in situations where the temporal changes in the environment could lead to the extinction of desired traits. To achieve this it is important to understand how to design and facilitate the evolutionary process in such a way that not only behavior that can be predicted to be desired in advance is evolved, but also the behavior that is needed, even when it is counter-productive for the individual phenotype.

Understanding the mechanisms that allow self sacrifice is a step towards understanding the mechanisms that can allow evolutionary computation to be truly adaptive through developing generalized algorithms for evolving behavior that favors solutions that benefit the entire population, or even sub-groups of the population. In evolutionary robotics there is there is always a need to better our understanding of how control mechanisms can be designed to be adaptable. The goal is to design the robots in such a way that when they encounter new problems that prevent them from doing the task they have been assigned, they are able to device solutions to the problem by continously changing the control mechanism without interference from the designers. One way to a better understanding of how such mechanisms can be designed is to examine what needs to be present in such a system for a given behaviour to evolve. Altruism is an interesting behavioural trait in this regard because it is a trait that can be useful for the population in a multi agents system a whole. Altruism can be seen as being evolutionary counterintuitive and is of interest because of this. Understanding the mechanisms that make individuals evolve behaviour that maximizes the utility of the population rather than maximizing the utlity of the individual is important to be able to create control mechanisms in multi agent systems that can make the system perform better as a whole.

A specific instance where the most acute form of altruism could be of use is a group of autonomous robots working on a glacier trapped on a sheet of ice where further exploration is only possible if one of the robots drives into a crevasse to form a bridge that the others can run over.

1.1 What is Altruism?

This section gives a brief introduction to what altruism is and presents the most authoritative works on the subject in biology.

1.1.1 Altruism in Biology

Altruism can in short be described as one individual sacrificing it's own fitness, meaning chances of survival to increase another's. In the classic theory of evolution individuals are thought to maximize their own fitness to ensure the survival of their own genes, yet this behaviour where a phenotype actively decreases their own chances of survival is often seen in nature. There has been proposed many explanations on how this behavior is evolved through natural selection when individuals seek to maximize their own fitness and the most prevalent theory in literature is the notion of 'inclusive fitness' outlined in the classic texts by Hamilton such as W. D. Hamilton [1963], Hamilton [1964a] and Hamilton [1964b]. Inclusive fitness includes not only the fitness of the individual, but also the number of offsprings it has and is able to sustain. In short, inclusive fitness measures the success of the individuals in ensuring the survival of their genes. To be altruistic and individual must perform and altruistic action. An altruistic action is characterized by the cost C in form of decreased benefit, the benefit B given to the receiver and the relatedness r between the parties involved. Hamilton characterized the relationship between the three in the equation given in 1.1

$$C/B < r \tag{1.1}$$

It is common to distinguish between reciprocal and non-reciprocal altruism. The latter means that the altruists gets no immediate benefit from the transaction.Trivers [1971] gives an explanation for the mechanisms behind reciprocal altruism. This text focuses on non-reciprocal altruism. Lehmann and Keller [2006] develops a method of classifying models of what they call 'helping' in which there is a distinction between the act of cooperation and the act of altruism. The transaction between two individuals is seen as cooperation if there is an exchange of fitness benefits, either directly or indirectly over time through repeated interactions. To be altruistic, the exchange has to lead to a direct or indirect decrease in fitness for one of the individuals. Although the focal point of this article is altruism, many of the same mechanisms that evolve cooperation apply and are sometimes referenced. Montanier [2013] presents a partial review of the most recognized mechanisms that account for the emergence of altruism. In this review, the mechanisms are divided into four categories:

Kin-selection The individuals that benefit from the altruistic deed are closely related to the altruist and are also harbors this capability, thus ensuring the survival of the gene.

Group-selection Groups are created randomly containing altruistic and egotistical individuals and the altruists help ensure the survival of the group as a whole. The groups are reorganized at random after some predefined amount of time has passed and without this the altruists would go extinct within their own group.

Tag-recognition Phenotypic traits are used to identify similarities in the genome, which altruistic individuals use to identify each other to gain selective advantage. Tag-recognition was first proposed by Hamilton and further explored and named the 'Green Beard' effect in Dawkins [2006].

Environment-viscosity In viscous populations, there is a greater chance that the benefit of altruistic actions goes to closely related individuals. This can be seen as a mechanism that ensures kin-selection.

1.2 Research on Altruism and Cooperation in multi-agent systems

Floreano et al. [2008] presents four different algorithms that may lead to altruistic cooperation. Both selection at the level of the individual and team selection is tested. The experiments simulate ants foraging for food items where two ants can bring back more food by cooperating than the two separately can by bringing a food item each. The associated cost is that each ant gets less food in return than when cooperating. Higher levels of altruism was observed when using team-level selection and more homogeneous teams had higher overall fitness. Experiments on kin selection in viscous populations were done by Dulk and Brinkers [2000] exploring the effect it has on the evolution of altruism. This concept is also explored from the viewpoint of theoretical biology in Joshua Mitteldorf and Wilson [2000]. Montanier and Bredeche [2011] uses an experimental setup where autonomous robotic agents must forage for food and there is a chance that the situation of the tragedy of commons might occur. The fitness function is implicit by having the robots exchange genomes with every other robot it meets during a generation. The robot then chooses a genome to use at random from its list of genomes and uses a slightly modified version of this. This is interesting because low viscosity increases the fitness at the population level. Altruism is still observed and to a certain degree tuned by introducing a mechanism for kin-selection. Turner and Kazakov [2003] explores how different mechanisms for sharing affect the spreading of altruism in a MAS. The agents have no explicit fitness function and their survival is dependant on a stochastic process. The altruistic gene is seeded into the population and they explore different degrees of kinship recognition, the most interesting of which being a scenario where the agents' Phenotypic traits are determined from their genetic makeup, save the gene that determines altruism. The agents use this to judge how likely it is that they are closely related. This is similar to a 'green beard' effect, except that it's not discriminated against non-altruists, only those of sufficient genetic distance. Ozisik and Harrington [2012] includes tags in the selective fitness model to account for some of its shortcomings. Hales [2005] proposes that tag-mechanisms obviate the need for repeated interactions or genetic relatedness to evolve altruistic behavior. The paper presents the hypothesis that mutating tags at a much higher rate than the behavioral strategy is a precondition for tag-mechanisms to work to avoid being exploited by free-riders. This hypothesis is tested experimentally with one-off prisoner's dilemma-games and the results support the hypothesis. Spector and Klein [2006] demonstrates experiments using tags where the cost of the altruistic acts exceeds the benefits of the recipient, an important step towards self-sacrifice. Mayoh [2000] evolves altruistic strategies in iterated games inspired by game theory where the possible strategies are predefined in the experiments. The interesting point made in this paper is that it provides a model showing that reciprocal altruism can be a good strategy for maximizing utility even in interactions where the other's strategy is unknown, ie. without the use of a tag. Cooperation among non-kin in organisms that lack the capacity to distinguish other altruists are accounted for in Barta et al. [2010] by the introduction of the concept of generalized reciprocality. The paper makes the argument that internal state is a factor and that some organisms are more likely to cooperate if they were cooperated with in the last encounter. This is similar to the tit-for-tat strategy in prisoner's dilemma and the results are shown experimentally by introducing state and evolving this strategy under a range of conditions. On the other end of the spectrum, Dessalles [1999] explains this behavior through complex political constructs and sub-group competition. Cooperation in situations where individuals have the capacity to assess the intentions of others are described in Han et al. [2011] where the results support the conclusion that intention recognition promotes cooperation.

1.3 Most related work

Martijn Brinkers and Dulk [1999] did simulations of the evolution of non-reciprocal altruism with kin-recognition where the altruistic act was indeed self-sacrifice. Agents were placed on a grid, and the grid had parts with land and parts with water. The goal was to forage for food, and agents could drive into the water forming a bridge between two pieces of land so that others could reach the food that existed on the other side. However, this was based on a very simple simulation where the genome evolved was the probability that an agent would drive straight ahead when there was water in front of it. An important point here is that although the results showed that the probability increased when closely related individuals could benefit from it, the act itself was not a behavior emerging from necessity, rather than by design.

1.4 Research question

Exploring the research on the artificial evolution of altruistic behavior and in particular the relationship between the evolution of altruism and the recognition of related individuals leads to the question of whether or not kin-recognition can be used as a way of ensuring the successful evolution of altruism in evolutionary robotics. At this point, it is important to keep in mind that recreating the conditions that evolce altruism in nature is but a mean to achieve the desired results in evolutionary computing and not goal in itself. This means that the research question and the proposed research is not geared towards explaining the observations from nature, it is directed towards finding mechanisms that can be used to solve a problem.

The research question that arose is presented here:

Research question *Do kin-selection and kin recognition help the evolution of self-sacrifice in evolutionary robotics in mutli-agent system?*

In this context, self sacrifice is thought of as the focal individual relinquishing its own possibility of further dissemination of ones own genes in order to enhance the the possibility of spreading the recipient's genes. Recognition of kin implies that the benefactor has a way of discriminating between those who have a genetic composition that is close to its own and those who do not. The literature shows that large degrees of altruism is rarely displayed without some form of explicit kin-recognition and there is reason to believe that evolving self sacrifice without kin recognition isn't possible. In this case, the null hypothesis is that it is impossible to evolve self-sacrifice without kin-recognition and failure to reject it would also be a result. The most promising of the simulations in the reviewed literature is that of Barta et al. [2010] where cooperation between individuals is evolved using the concept of generalized reciprocality.

Chapter 2

Method

To see if kin recognition has a positive effect on altruistic behaviour different means of recognizing kin will be tested against a baseline without any form of recognition. This chapter describes the experiments to be used to determine if and how different kin-recognition mechanisms lead to different degrees of altruism.

2.1 Basic setup

To observe altruism a population of robots that can perform an altruistic action and an environment for the robots to interact in is needed. The robots need to have a control mechanism where the control of all the actuators are evolved by having the agents exchange genotypes that are combined and mutated to serve as the basis for the next generation of individuals. A simple way to model this is to make the robots dependent on a form of sustenance for survival and give them the the ability to give away sustenance. In the experiment a population of robots will forage for sustenance by collecting and consuming "energy points" in the environment. The Robots consume energy indiscriminately, meaning that they will consume all the energy they come across, and can not continue to survive in the environment without energy. A fixed maximum lifetime duration will be used to simulate the robots dying of old age to ensure the continuing evolution of the population. The robots will be able to mate with other robots they encounter which is how the mixing of successful genotypes is ensured.

2.1.1 Initial period

To separate the evolution of the foraging behaviour from the evolution of altruism the population will first be evolved without the ability to give away energy. In this preliminary period, the agents will consume energy points that are distributed in and generated by the environment. When an energy point is harvested by an agent it becomes inactive so that no agents can benefit from it. After a delay, the energy point is re-activated and can again be used. The populations that have a form of kin selection or kin recognition will be evolved with this trait active also in the initial period.

2.1.2 Second period

When the initial period is over and effective harvesting patterns have been established the agents will be given the ability to create energy points of their own. The robot generated energy points are like the energy points that are generated by the environment, but the agent who has generated a point can not consume it. The energy points created by the robots can only be consumed once and are never replenished. In this part of the experiment all the energy points created by the environment are removed to simulate a period of food shortage. This situation will be used to see if the different mechanisms implemented will lead to different strategies for sharing the available resources.

2.1.3 Desired behaviour

A succesful outcome of the experiment would be if more robots survive for a longer period of time than in the baseline. The desierd effect is that related robots share the energy resources in the environment in a more effective way than before.

2.2 The scenarios

2.2.1 Baseline experiment

In the baseline experiment the robots have no notion of kin-recognition or even that there are other robots in the environment at all. This experiment establishes the behaviour that is expected from the robots when they are simply given the ability to donate energy after first having evolved basic harvesting behaviour.

2.2.2 Kin recognition of nearest individual

In this experiment the robots are given the ability to assess how closely related they are to the robot that is the closest to them in geographical distance. The hypothesis is that if the robots that are more prone to donate energy when they are close to a related robot helps related robots survive, a group of related robots will help each other to survive in the environment and in that way ensure that the altruistic trait is carried one while the group reimans alive. The representation of the genotypes of the robots will be given as a vector of numbers an thus the measure of the distance will be based on the formula in

$$\sum_{i=1}^{N} |RobotGenome_i - ClosestGenome_i|$$

2.2.3 Recognition of direction of closest related individual

In this experiment the robots are given the ability to discern the location of the closest related individual in the environment. The hypothesis is that the robots that thend to gravitate towards related individuals and are prone to altruism will have an advantage in that related individuals benefit from the altruistic deeds. The input value of this sensor will be the difference in the angle between the orientation of the robot and an absolute reference point and the angle between the robot and the closest related robot. i This value will be given in degrees between -180 and 180. The closest related individual will be found by using the same measure of relatedness as in the kin recognition.

2.2.4 Kin-Selection in the evolutionary algorithm

In this experiment the robots will select the individual they have encountered among the other robots that is the closest to them genetically to use as a basis for procreation. The hypothesis is that since kin selection tend to make robots that are in close proximity of each other more related, the robots that are prone to altrusim will give benefit to related individuals ensuring that the trait survives. Again, the same measure will be used for determining the most related genotypes.

2.3 The Robots

The robots that are simulated in the environment are based on the epuck model and are thought to be compact differential wheeled robots with sensors that sense distance in eight directions as shown in figure **??** .

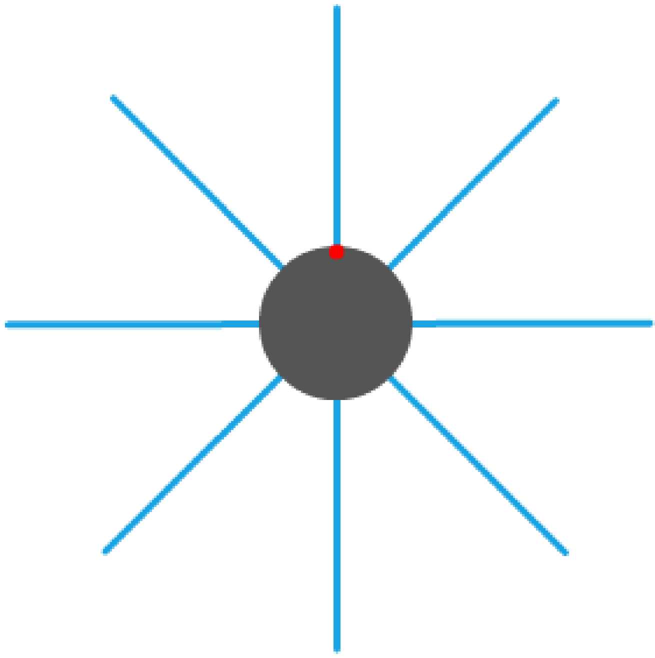


Figure 2.1: A single robot seen from above. The lines emanating from the body represents the reach of the distance sensors.

The robots also have a sensor that can sense if they are directly on top of an energy point. The sensory input the agents get are fed into the neural network that controls the actuators of the robots. There are four actuators: The two wheels, the ”mouth” that absorbs food and the output that creates energy points. Each robot is controlled by an artificial neural network that has 13 inputs that will be connected to the robot’s sensors.

2.3.1 Sensors and actuators

The robot has 12 sensors that all provide numeric input to the control mechanism. The function of the each sensor is given below:

- 1-8: Distance sensors
- 9: Direction to nearest energy point

- 10: Distance to nearest energy point
- 11: Energy level
- 12: Dependant on scenario

The robot has three actuators, two wheels used for locomotion and an output for the energypoints.

2.4 Artificial Neural Networks

ANNs are computational entities that are inspired by how the brain does computation. In the brain, a network of neurons acquire knowledge through the body's receptors and maps the perceptions it receives to a given action. Learning is achieved by strengthening the interneuron connections, known as synaptic weights. Haykin [1994]

One of the key reasons using a neural network is beneficial in this problem is that one of the great advantages of neural networks is that contextual information is taken into consideration. All the neurons in the network are affected by what is happening on a global level and therefore a given response may be elicited according to context. In this case, the choice of relinquishing energy should be affected by whether or not there are other robots nearby and perhaps also how closely they are related.

Artificial neurons have a series of inputs that are altered according to the weight of the input. This weight is analogous to the strength of the synaptic connection. There is also often an extra input that is constant and is known as a bias-weight. The summation of this is fed to an activation function that determines the output of the neuron. The output of one neuron can be fed as part of the input to another neuron and this is how the network is built. A neuron can also use its own output as an input, effectively giving the neuron memory.

The ANN that the robots are controlled by is a multi-layer perceptron with three hidden neurons and three output neurons. The output neurons control each of the wheels and has a binary output that decides if energy should be dropped or not.

Figure 2.4 shows the schematic drawing of the artificial neuron used by the robots in the baseline and the kin selection scenario. The unused sensor input is set to be 0 at all times.

The ANN for the kin recognition and kin seeking scenarios are show in figure **??** and 2.4 respectively.

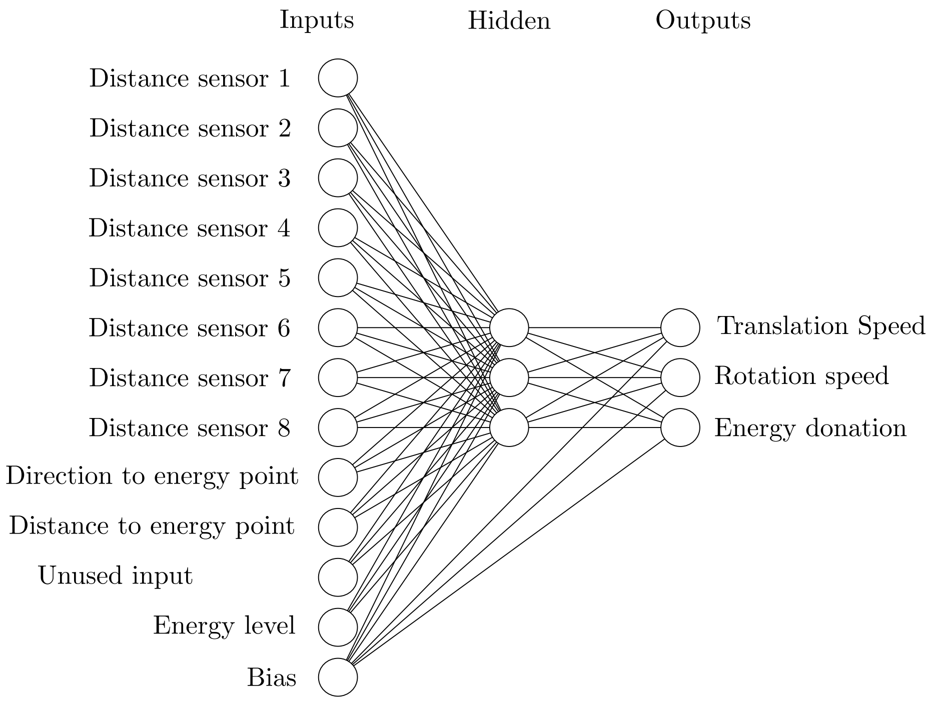


Figure 2.2: ANN used by the baseline and kin selection

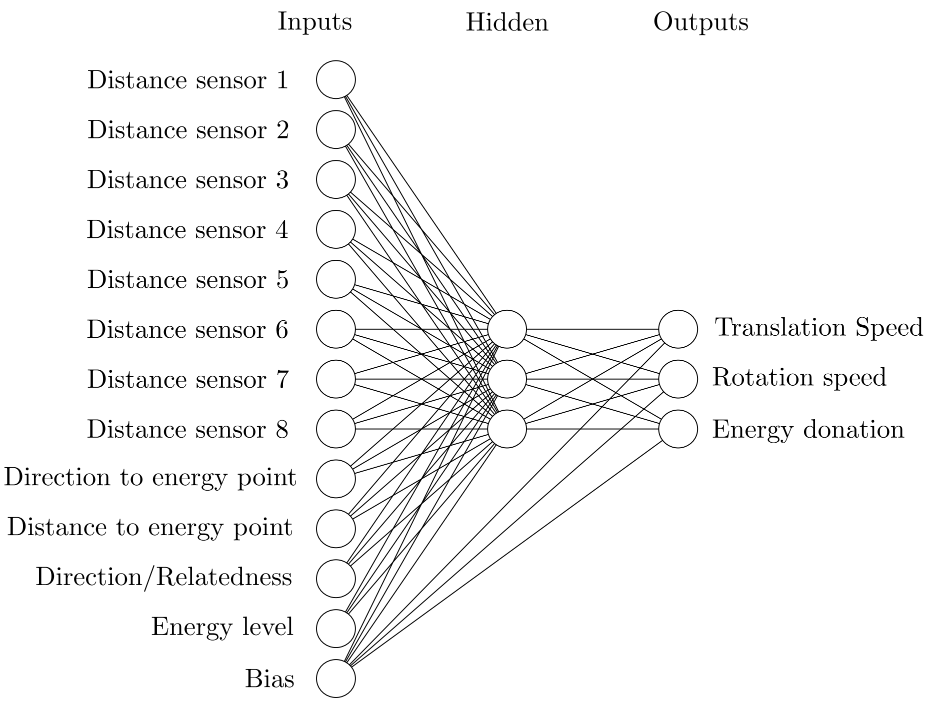


Figure 2.3: ANN used by the kin seeking/recognizing population

2.5 The Evolutionary algorithm - mEDEA

The evolutionary algorithm to be used needed to be both robust to change and have a fitness function that rewards survival and spreading of genes. The reason for this is that the algorithm should be designed to reward selfish individuals on the individual level so that the altruistic behaviour arises as an evolutionary respones that increases inclusive fitness. For this reason the mEDEA-algorithm was chosen. The mEDEA algorithm is presented in pseudo code in figure 1

Algorithm 1 The mEDEA algorithm

```
genome.randomInitialize()
while forever do
    if genome.notEmpty() then
        agent.load(genome)
    end if
    for iteration = 0 to lifetime do
        if genome.notEmpty() then
            agent.move()
            broadcast(genome)
        end if
    end for
    genome.empty()
    if genomeList.size > 0 then
        genome = applyVariation(select_random(genomeList))
    end if
    genomeList.empty()
end while
```

In the mEDEA algorithm each agent has a list of genomes. Every time an agent encounters another agent they add that agent's genome to their list of genomes and when an agent is reactivated it choses a genome from it's list of genomes that will serve as the basis for the new genome of the agent. The genomes are mutated as in regular evolutionary algorithms and the selection scheme can also vary. The fact that survival and dissemination of genes is an imlicit demand of the mEDEA algorithm makes it a great fit for the experiment. The mEDEA-algorithm has an implicit fitness function. The better the individuals are at spreading their genes, the greater are their chances to pass on their genes. This closely mimics the way evolution works in real life. Having an implicit fitness function that partially rewards for the trait we are after is having traits appear by design rather than by necessity. An agent is deactivated when it runs out of energy and reactivated when another agent passes nearby.

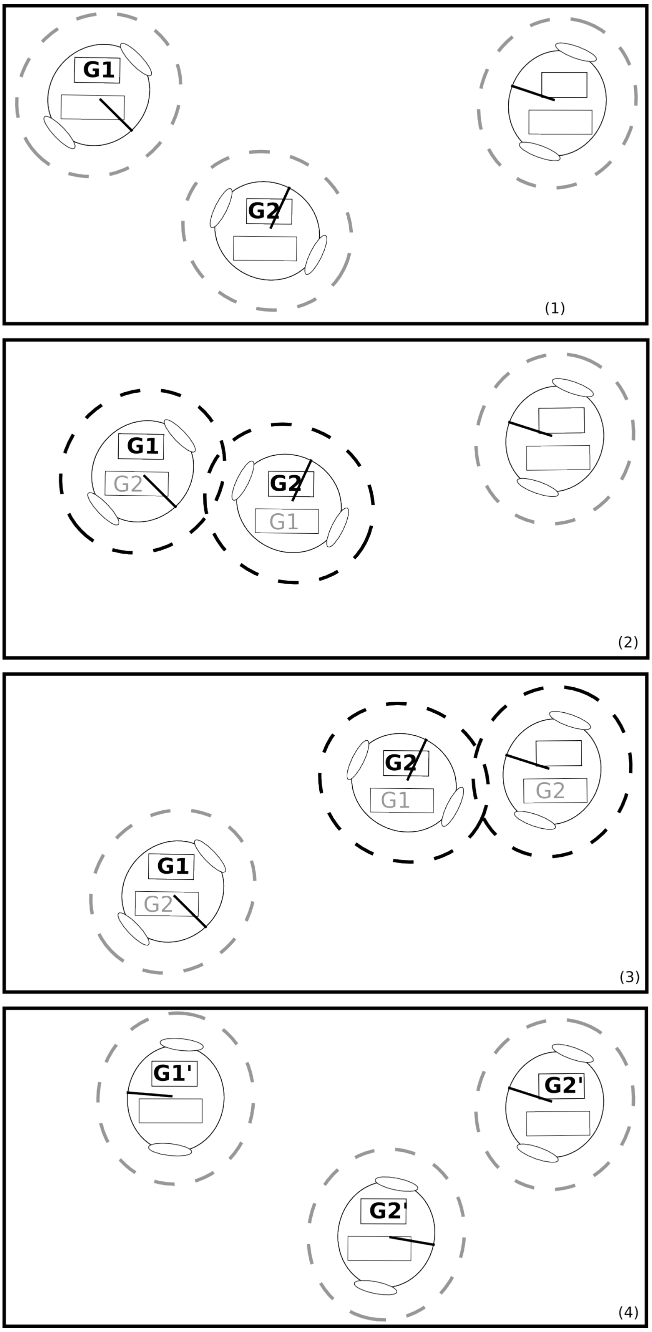


Figure 2.4: Sequence showing how genomes are exchanged in the mEDEA-algorithm

2.6 Donating Energy

The agents each have an output neuron that outputs a floating point value. If the value is above a predetermined threshold the agent creates an energy point in the environment that it cannot utilize for itself. The energy points that are created are never replenished unlike the energy points that are intrinsic to the environment. The amount of energy donated is set to be a predetermined number. This was chosen to simplify the experiments, but ideally the amount of energy donated should be determined by the output value of the neuron.

Using a predetermined value relieves the model of biological accuracy, but it allows for greater control and understanding of the parameters that are needed for the wanted behaviour to occur, which is the object of the experiment. Early initial tests showed that linking the amount of energy donated in each timestep unsurprisingly led to the first generation of agents commiting mass suicide in the first few timesteps since the output value from the beginning is random. It is a reasonable to assume that the insects or animals the robots model already have evolved an inclination towards not dropping food and that this should only happen when a certain sensory input is provided.

2.6.1 The environment

The environment the robots inhabit is a large two dimensional square with a few obstacles scattered around. Very little exists in the environment and the robots roam around freely. A screenshot of the environment can be seen in figure

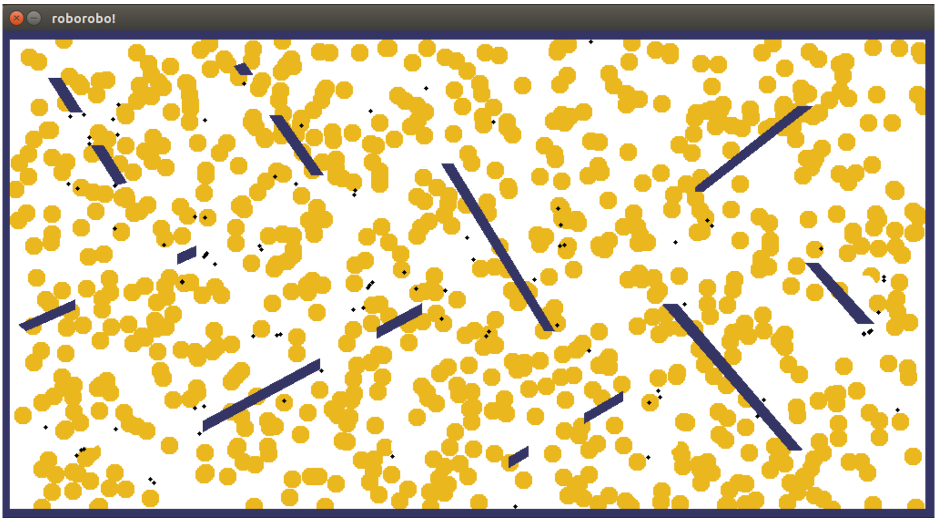


Figure 2.5: Screenshot of the robots in the initial period. The colored dots are the the energy points created by the evironment, the black dots are robots and the lines are walls

2.7 Important parameters for the initial period

The complete properties file for the experiment can be found in the appendices.

The robots start out with a total of 100 units of energy and a generation lasts 400 iterations. 1 unit of energy is spent per iteratiion which gives the genom 1/4 of generation to prove itself.

Energy expenditure	1
Initial energy	100
Iterations/generation	400
Energy points	100
Energy point value	50
Revival energy	400

Table 2.1: Important paramters for the first experiment

2.8 Important parameters for the final period

The threshold that the output of the donation neuron needs to exceed for a donation to occur is set to 0.5. Since the initial output value of this neuron is not a factor in the initial evolution of the agents it is to be expected that half of the agents will become donators immediately and the other half won't Introducing the trait in this way is artificial but it if the trait has a negative enough impact on the fitness of the individuals the trait will soon disappear. The energy expenditure in the altruism part of the experiment is set to 0.005 per iteration. The initial energy of the robot is set four times higher than in the initial period so that there will be enough time for the altruistic agents to meet other agents and propagate their genes as the initial tests showed that the altruistic agents would often give away so much energy in the beginning causing them to die out before being able to spread their altruistic genes. This also caused the entire population to go extinct shortly after. In this second period the robots spend 0.005 per iteration which makes the total expenditure of energy per generation 20 units for each agent. This means that they can survive for 200 generations without needing food provided they don't donate energy. This value is set low to ensure that the agents survive long enough that evolution can occur.

For the first experiment the number value of each energy point to be generated by each agent was set to be 50. This means that the agents can run out of energy if they create more than 16 energy points in the first generation. This number was chosen to be low enough that not all agents that are inclined to donate energy will die, but high enough so that the energy point provides useful sustainance for other agents.

The experiment is run for 400 generations to see what happens when the robots are close to running out of energy altogether, but the main period of interest is before this when the agents need to distribute the energy they have among them in order to ensure that all the active agents surive as long as possible.

Donation Threshold	0.5
Energy Point value	50
Energy expenditure	0.005
Initial energy	400
Iterations/generation	400

Table 2.2: Important paramters for the final period

2.9 Experimental setup

All the experiments were run on an Intel Centrino 2 clocked at 2.26 MhZ. For each setting, each experiment was run 100 times and the results computed as an average over those runs. On average, a single complete run of one experiment with both the initial period and the final period took approximately 30-35 minutes. To implement the experimental environment described an existing system that fulfilled many of the requirements was modified. The system that was used was Roborobo which is a 2D robot simulator based on the epuck/kephra model written mainly by Nicolas Bredeche with assitance from Jean-Marc Montanier and Leo Cazenille. Roborobo is written in C++ and is described in detail in Bredeche et al. [2013] This system was chosen because it was available in open source and because it provieded a lot of the functionality that was needed for the experiment:

- Ready made environment for simulating small robots
- Integrated code for neural networks
- Built-in evolutionary algorithm functionality ...

The graphs were created using the Java library JFreeChart available at `http://www.jfree.org/jfreechart/`

Chapter 3

Results

In this chapter the results from the experiments are presented. First the results of the baseline is presented to show the behaviour that is expected when no kin-recognition is present in the system The results from the three experiments are presented in turn and the same graphs are shown for all the experiments.

- The amount of agents over time
- The amount of agent generated points created and consumed
- The amount of energy available in agents and points

For all graphs the standard deviation is shown as a shaded area beneath the plot.

3.1 Baseline

Figure 3.1 shows the graph displaying the number of active robots in the environment at the end of each generation. The number of robots first drops dramatically as there is an extreme amount of self sacrifice in the begin when the donations happen at random. The number of robots quickly rises again as the remaining active robots who are not giving away energy consume the energy that has been dropped. The robots then gradually become inactive as the robots with the least amount of energy expend all their energy while there is less and less energy available in the environment as shown in figure 3.3 which shows the sum total of energy in the robots and the sum total of energy in energy points in the environment over time. Figure 3.2 shows the number of points created and consumed. It is clear that after the initial period there is very little energy being brought into the environment as very few new energy points are created.

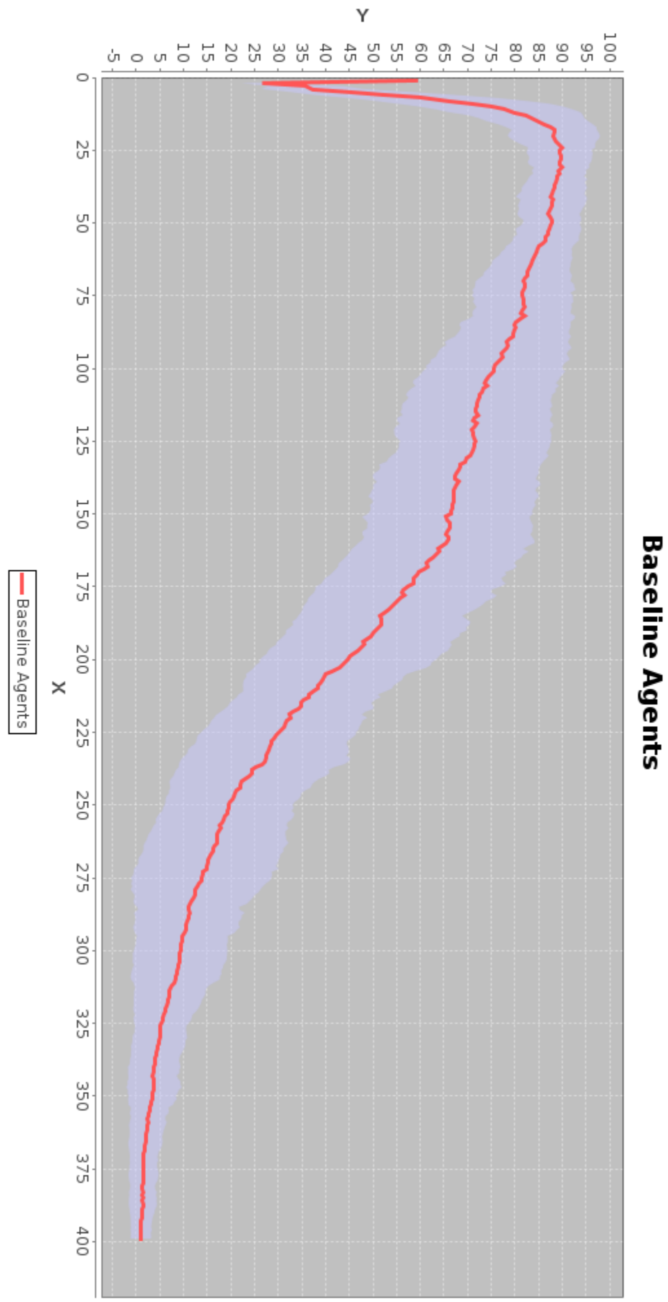


Figure 3.1: Graph showing the number of active robots in each generation for the baseline

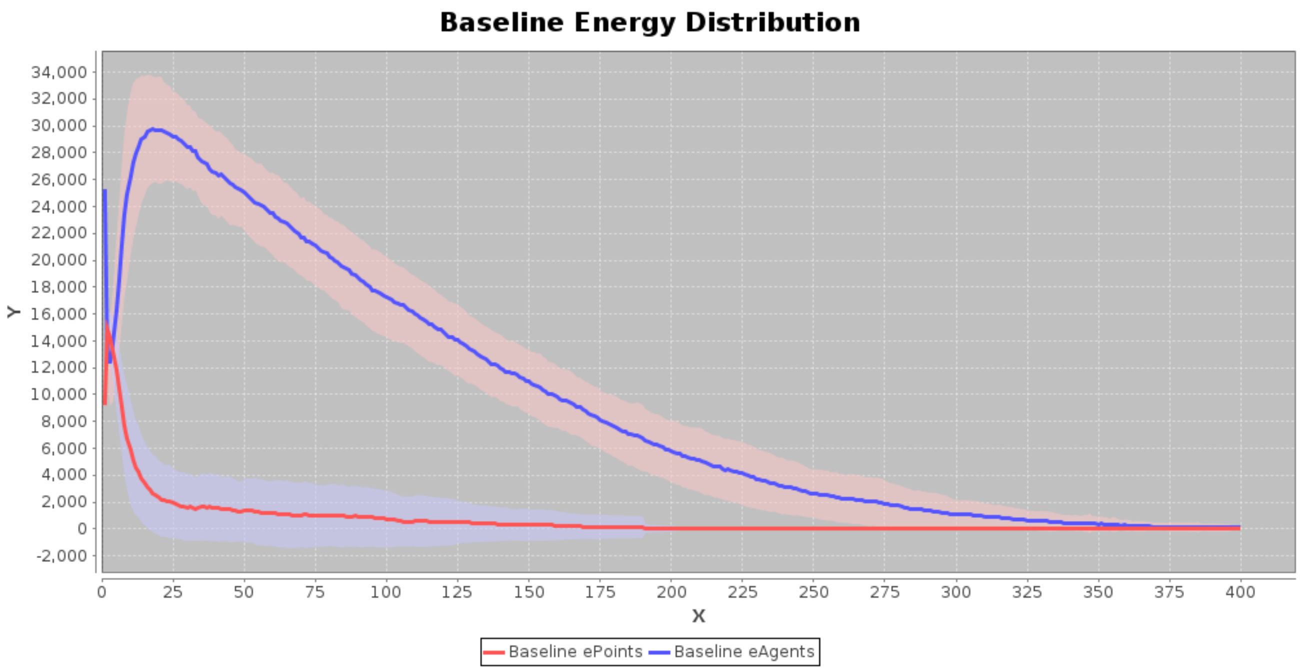


Figure 3.2: Graph showing the amount of energy in the system and the amount of energy in the robots for the baseline

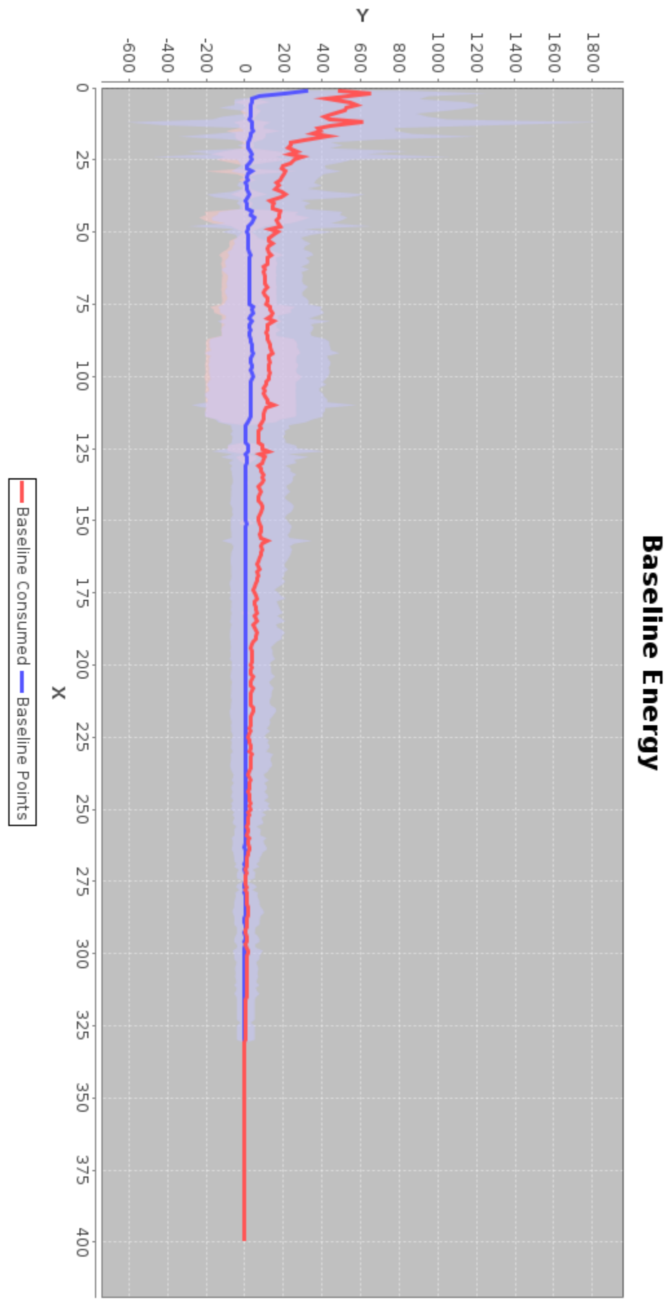


Figure 3.3: Graph showing the number of points created and consumed each generation for the baseline

3.2 Kin recognition

Figure 3.4 shows the graph with the number of active robots in the kin recognition scenario superimposed on the graph with the number of active robots in the baseline from 3.1 for reference. The drop in number of robots is less dramatic than in the baseline, but in return the rise in number active agents in the subsequent period is substantially less. The cause of this is unknown, an hypothesis could be that the kin seeking population is more homogeneous than the baseline. This could lead to less of an impact on the population because the surviving robots have a similar genetic make-up as the ones that die, so more robots keep on creating energy points after the initial period. The graph in figure 3.5 supoorts this by showing that roughly 50 % more points are created in the first 25 generations which is the time it takes the baseline to reach its maximum population. There were however no measurements of the homogenity of the populations at the start of the second period made that can be used to support this. The graph in figure 3.6 shows that for a large section of the duration of the experiment the amount of energy in the system is constant although points are being consumed.

3.3 Kin seeking

Kin seeking writeup

Figure 3.9 shows the amount of energy in the robots and the amount of energy in the environment in the form of energy points over time. Figure 3.7 Figure 3.8

3.4 Kin selction

kin selection writeup

Figure 3.12 shows the amount of energy in the robots and the amount of energy in the environment in the form of energy points over time. Figure 3.10 Figure 3.11

3.5 Graph overlay

When putting all the graphs on top of each other as can be seen in figure 3.13 the effect of the kin oriented algorithms become visible. The number of active agents never rise to the same level as in the baseline with the maximum number among the kin oriented algorithms at 45 active robots opposed to nearly 70. However, the amount of agents drops a lot more rapidly in the non-kin oriented

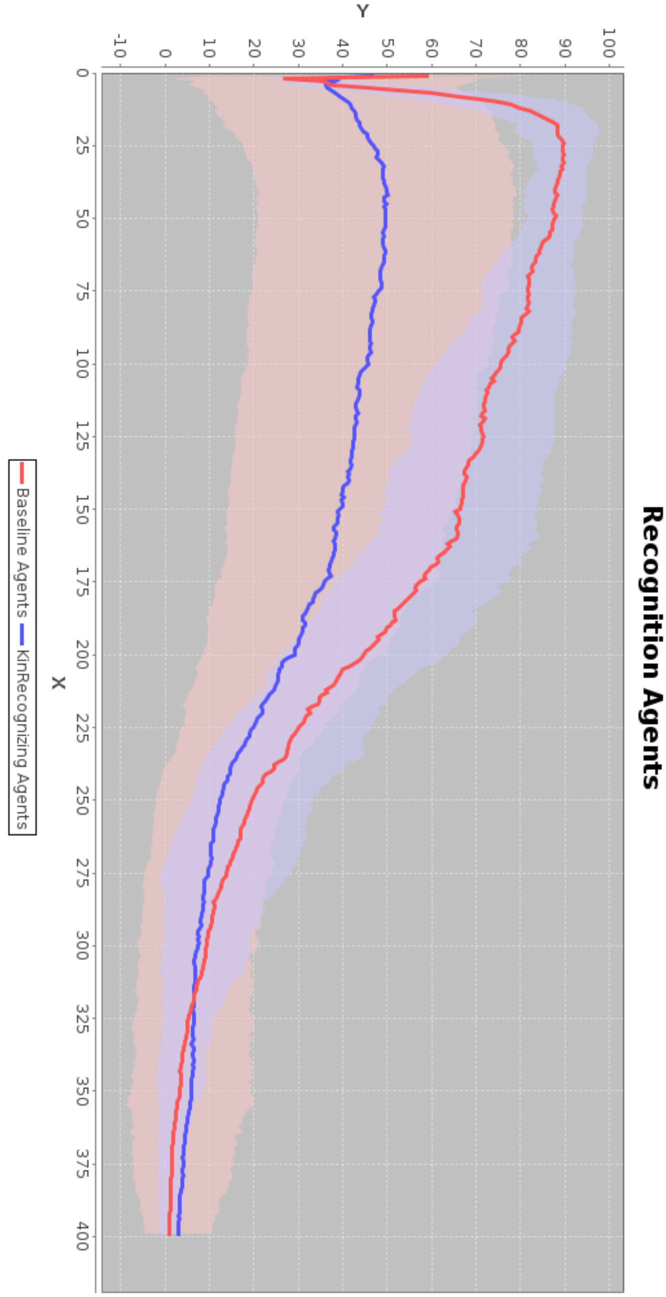


Figure 3.4: Graph showing the number of active robots in each generation for the kin recognition scenario

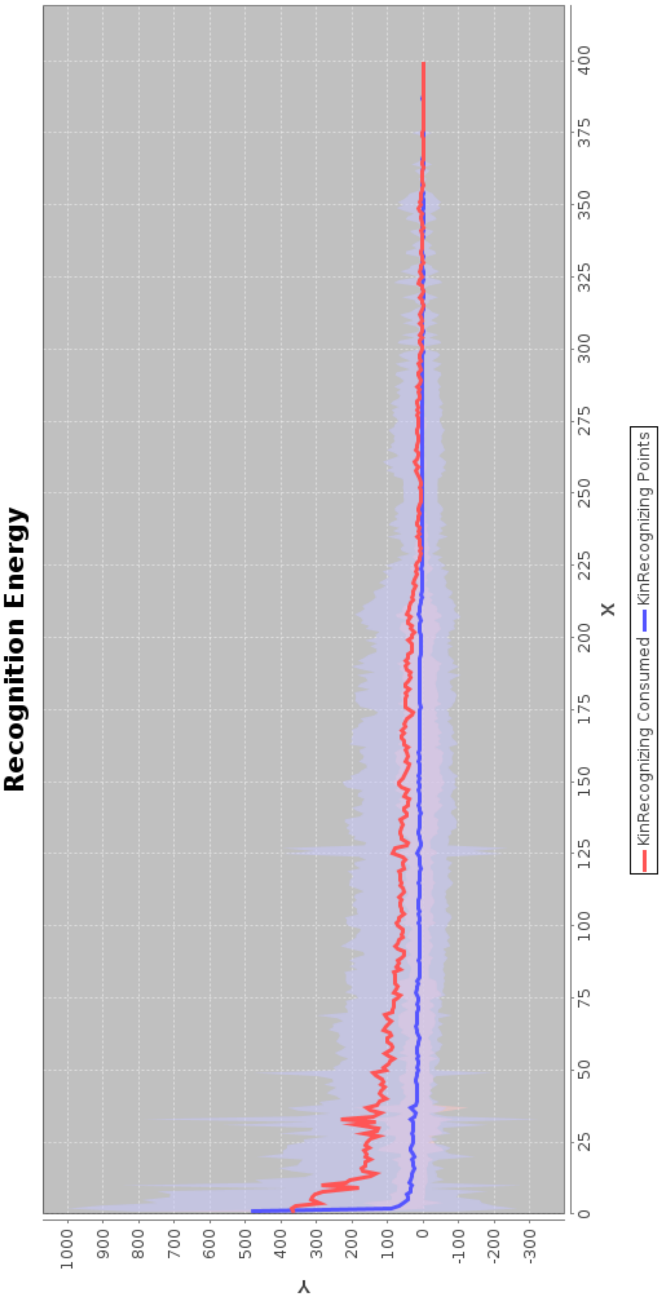


Figure 3.5: Graph showing the number of points created and consumed each generation for the kin recognition scenario

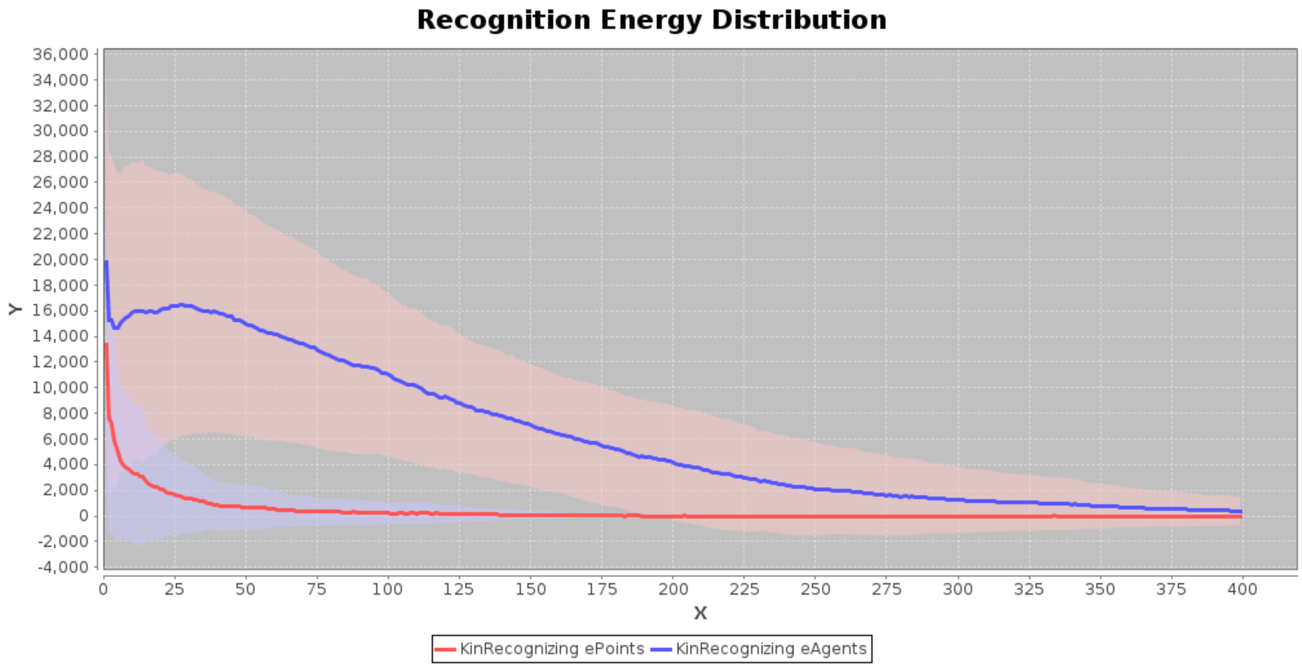


Figure 3.6: Graph showing the amount of energy in the system and the amount of energy in the robots for the kin recognition scenario

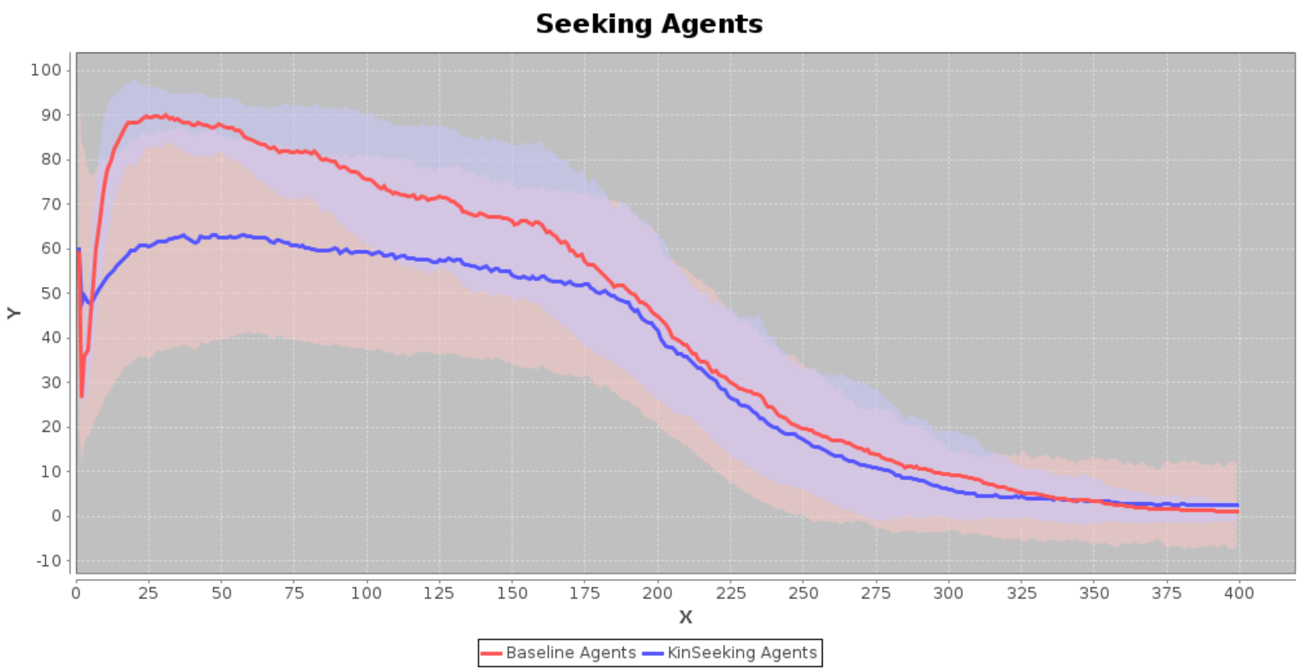


Figure 3.7: Graph showing the number of active robots in each generation for the baseline

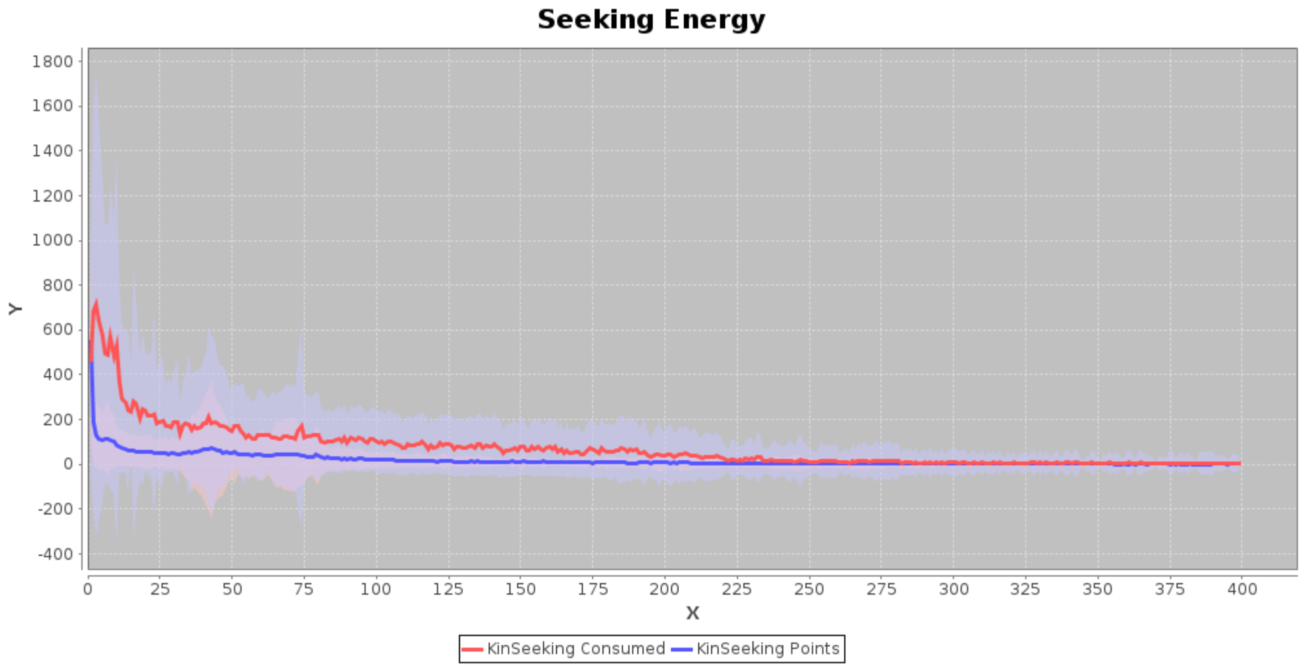


Figure 3.8: Graph showing the number of points created and consumed each generation for the baseline

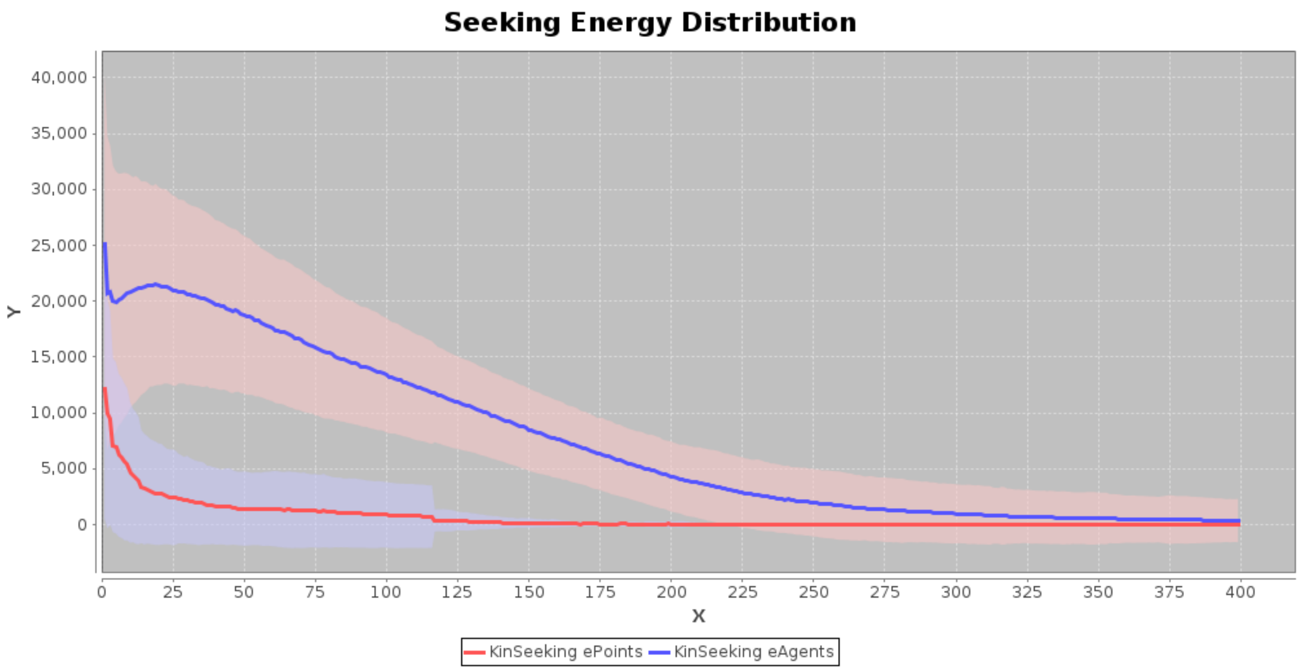


Figure 3.9: Graph showing the amount of energy in the system and the amount of energy in the robots for the baseline

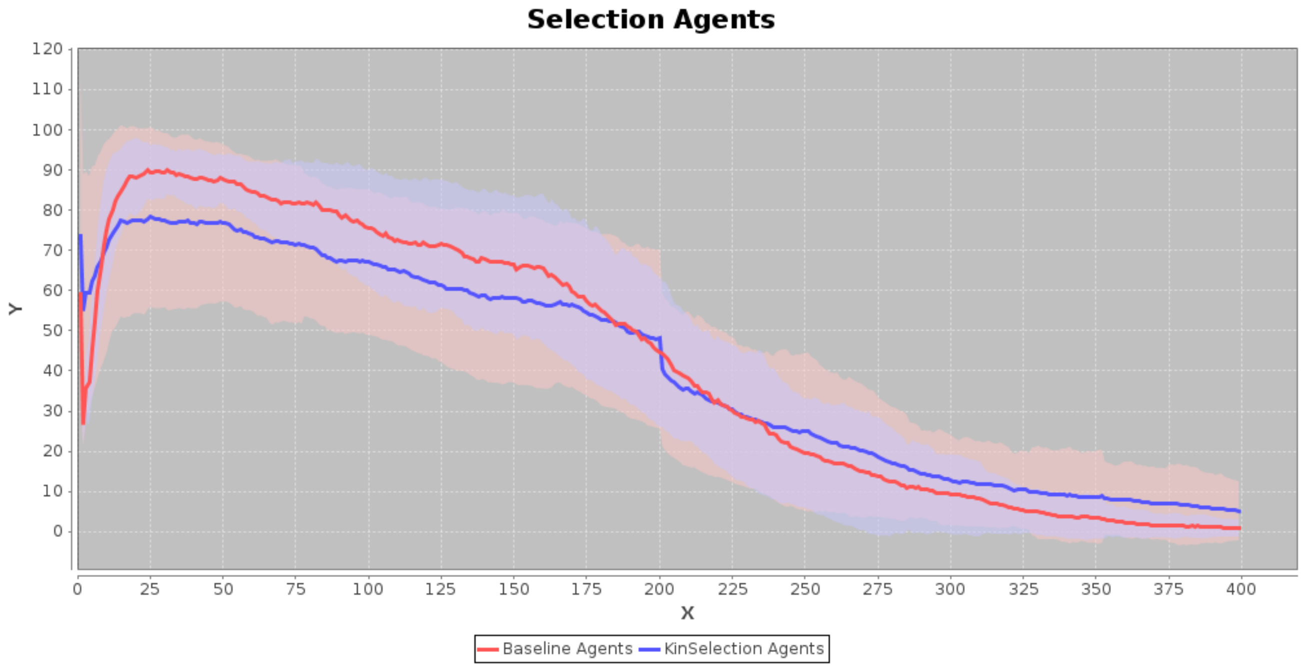


Figure 3.10: Graph showing the number of active robots in each generation for the baseline

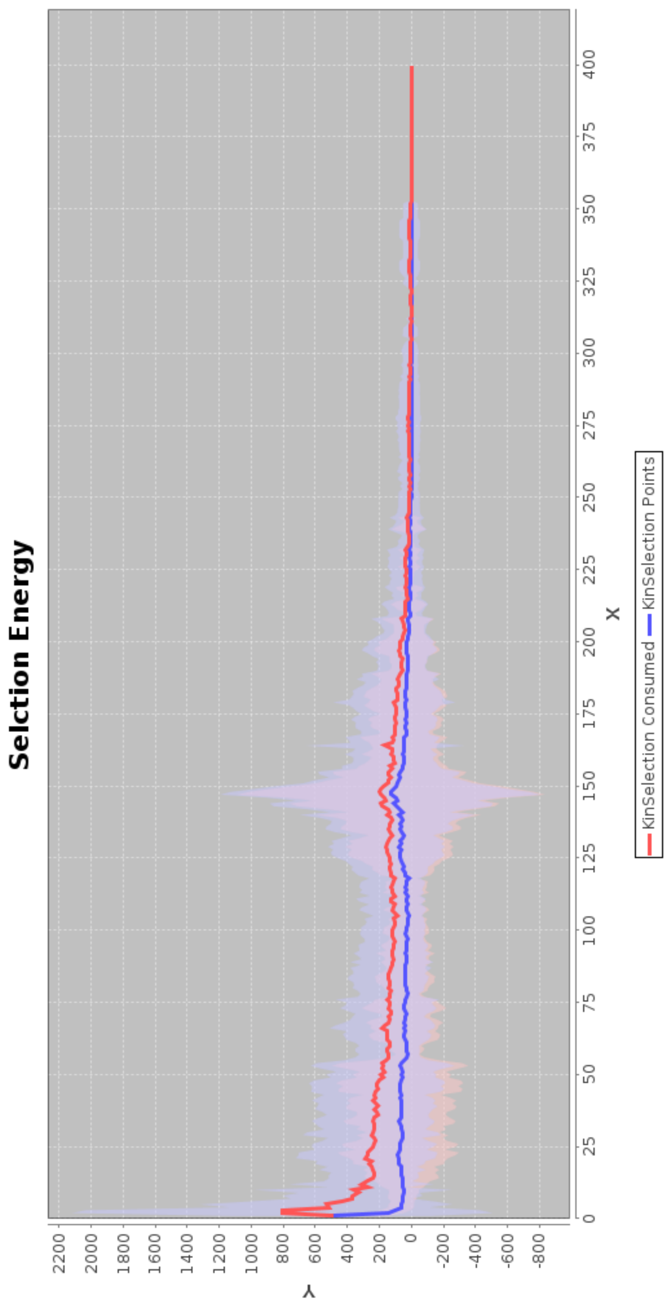


Figure 3.11: Graph showing the number of points created and consumed each generation for the baseline

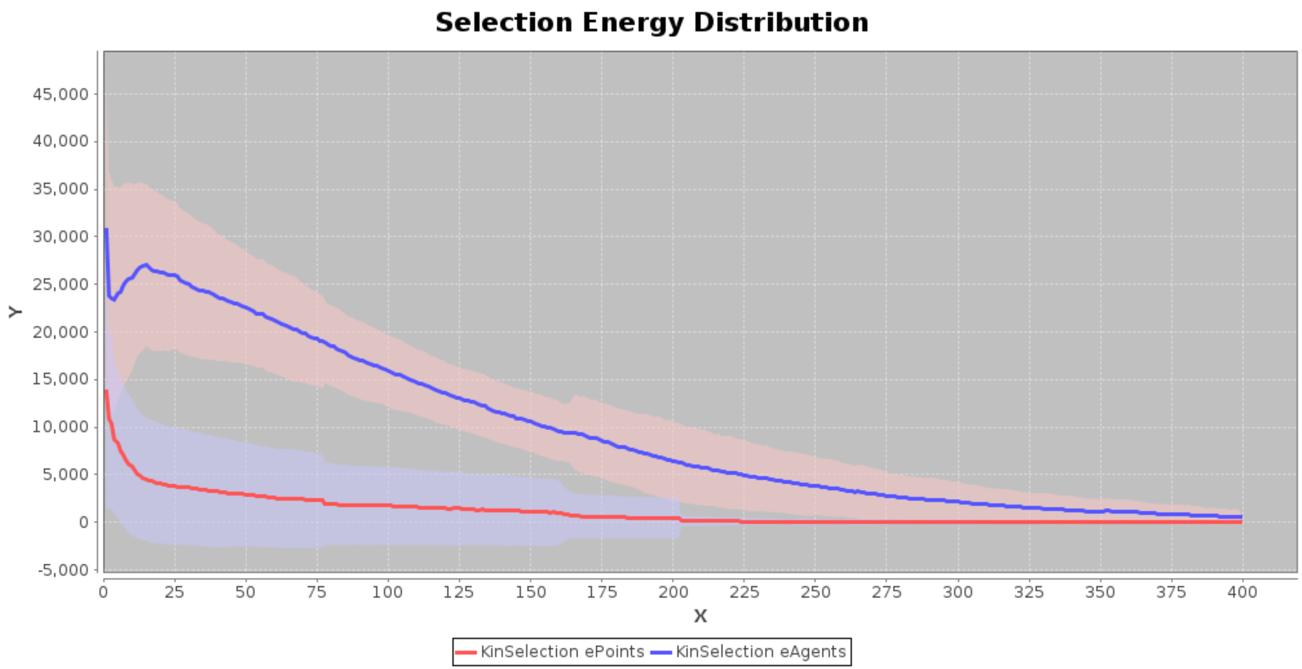


Figure 3.12: Graph showing the amount of energy in the system and the amount of energy in the robots for the baseline

algorithms and soon after a 100 generations all the kin oriented populations are more successful.

Figure 3.14 shows the overlay of the number of points created. The kin seeking population creates slightly more energy points than the others explaining why they remain the most successful population for a large portion of the experiment. In the period from 120 to 170 the kin selection population creates more points than the others which explains why it ends up as the most successful population towards the end of the simulation. A detail of this is shown in figure 3.15

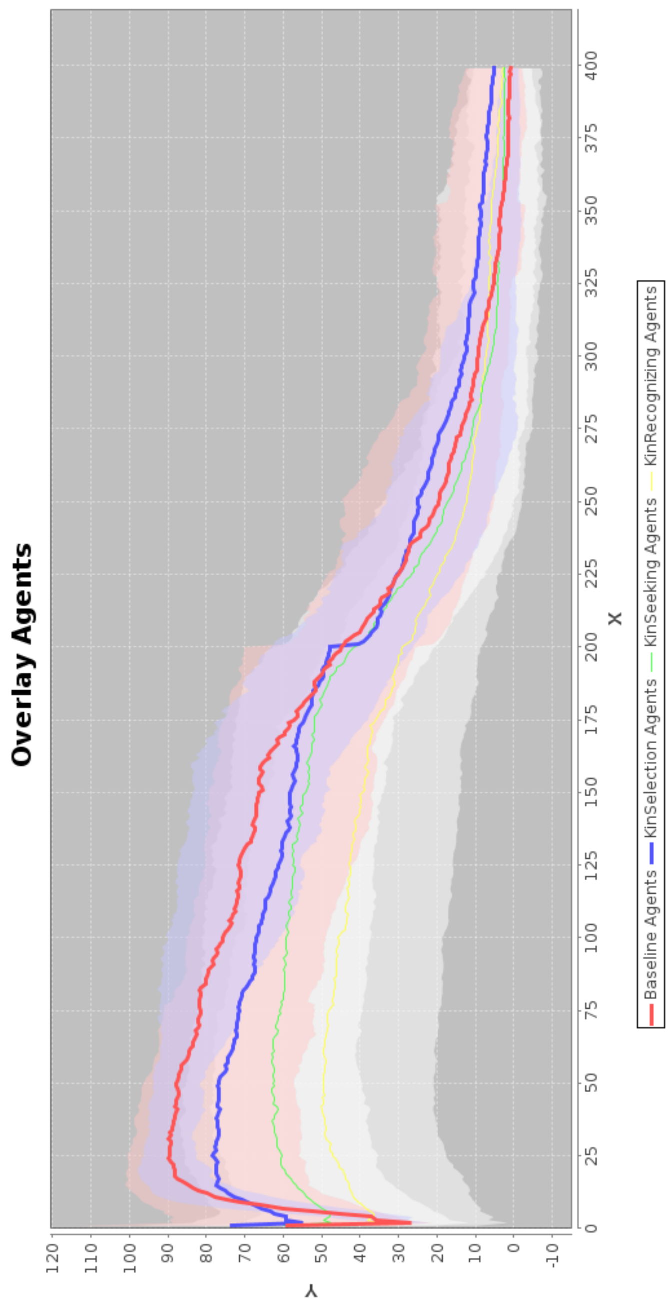


Figure 3.13: Graph showing the number of agents in all the runs

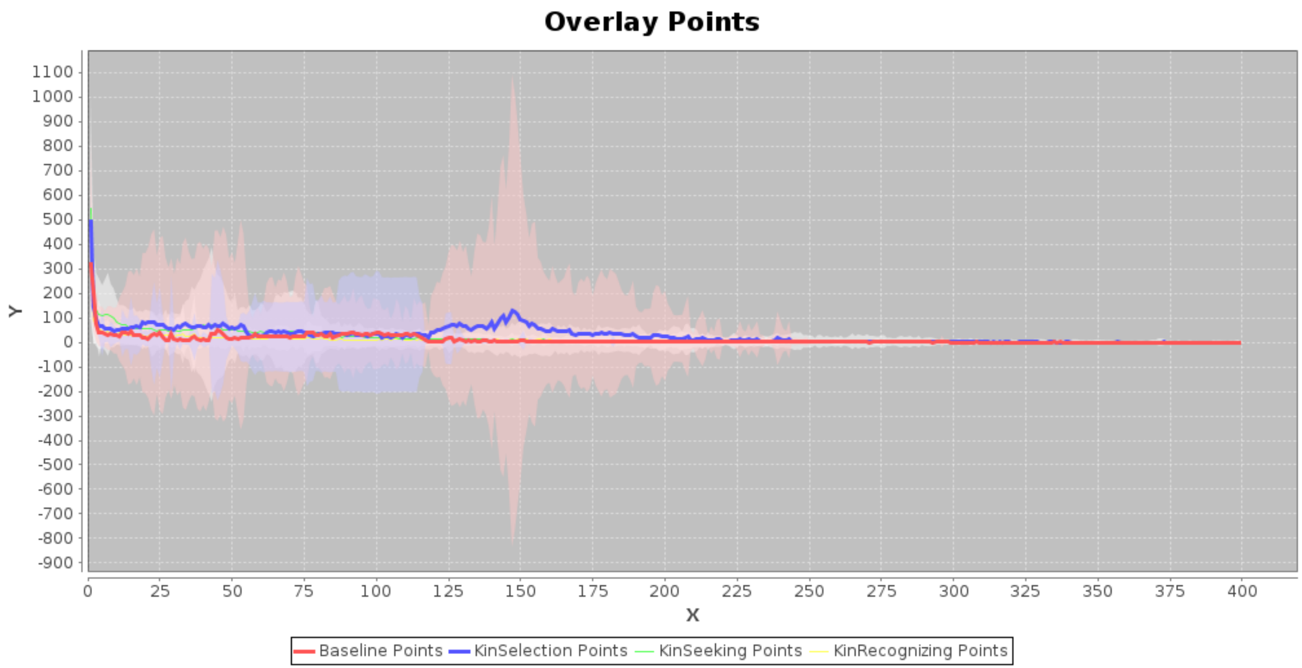


Figure 3.14: Graph showing the number of agents in all the runs

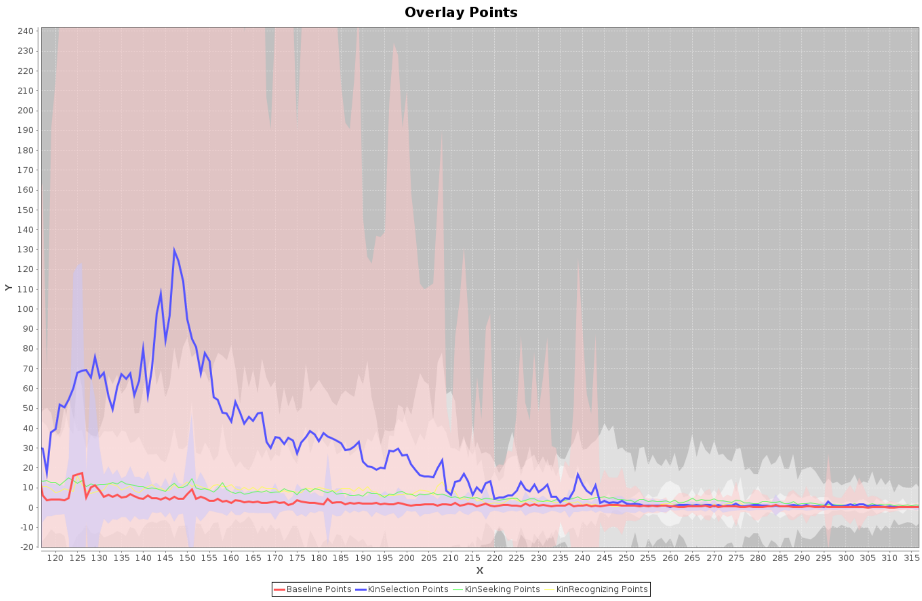


Figure 3.15: Overlay graph of the number of points created for each scenario

Chapter 4

Discussion

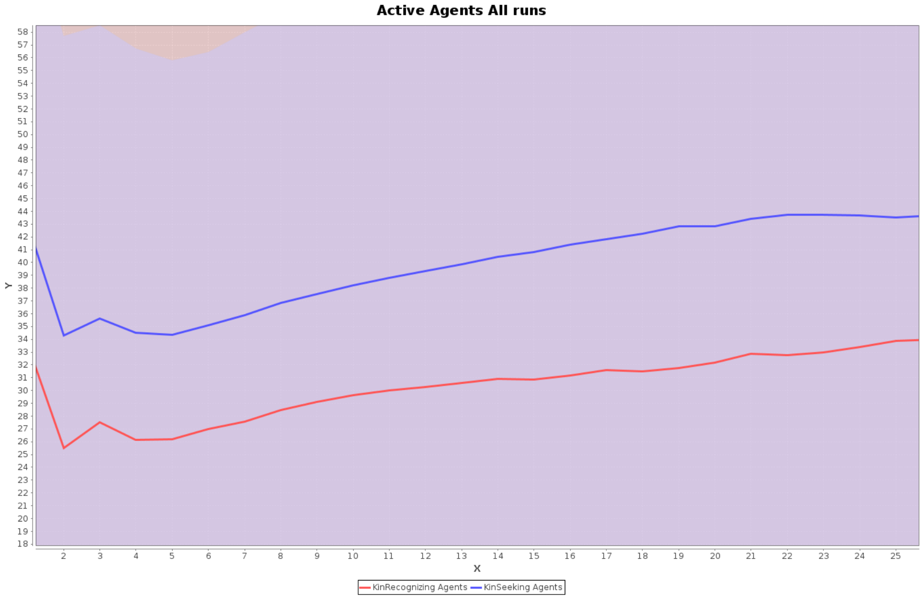
This chapter contains of the results presented in the previous chapter and adresses the implications they have regarding the research question.

4.1 How succesful were the kin oriented mechanisms?

All the kin mechanisms have more robots alive than the baseline after the initial period of 100 generations and all the kin mechanisms have on average more robots alive at the end of the 400th generation. A contributing factor to this could simply be that the baseline has a much larger population during the first 100 generations and this more energy is spent and less energy is available to the remaining generations. It's difficult to say why the baseline donates much more energy than the other populations in the beginning and it could be due to an error in the model although no error has been found.

4.2 Simliarities of kin seeking and recognition

The graphs displaying the number of active agents reveals that the shape of the curves for kin seekin and kin recognition look remarkably similar. The detail of the graph shown in figure 4.2 shows that they display the same pattern of falling before rising slightly and then falling again before they ascend to the highest point. This could be an indication that the populations evolved by the kin seeking population and the kin recognizing population may have a simliar constitution.



4.3 Future work

The results from the experiments leave room for much to be explored. This section gives a few suggestions on what should be pursued further.

4.3.1 More advanced version of kin seeking algorithm

The kin seeking algorithm seems promising and it would be interesting to run an experiment where the robots are given the direction of the largest concentration of related individuals. In this scenario the robot may seek out a group of robots that has a similar genotype. The hypothesis is that this would lead to more robots with altruistic inclinations profitting from the altruism when altruistic individuals group together. The challenge would be to find a purposful way of grouping the robots together when assessing which area has the largest concentration of related individuals. One way of doing this could be to partition the environment into sections where the average degree of relatedness is calculated for each agent for each section. However, this method makes it more challenging to transfer the method to situated agents as it requires each of the robots to have a mapping function. Assuming the robots broadcast their whereabouts this could be solved by sectioning a circle surrounding the agents.

4.3.2 More advanced version of the kin recognizing algorithm

The kin recognizing algorithm could be expanded upon by having the agents take into account how related the are to all agents that are within close enough distance to transfer their own genome. This could provide a higher chance that the robots would give away energy when related agents are close enough that they may benefit from the donation. Another benefit of this method is that it makes use of the already necessary functionality in the robots to broadcast their genome to all agents who are close enough. This makes it a more practial method than the kin seeking method as less elaborate methods of

4.4 Energy transferring

In the experimental set up used the robots donate energy to the environment without any assurance that the energy is consumed by the intended beneficiary. This increases the chance that related individuals will be deprived of the energy and in as a consequence giving away energy is less likely to increase the inclusive fitness. In future experiments the way of donating energy could be changed to a direct transfer of energy between the robots. T his would eliminate the problem and also provide a higher degree of realism since creating a way to exchange energy between actual robots would be best deviced with a form of aggregation or throug wireless energy transmission. In both cases, proximity is a factor that must be taken into account.

Appendices

Appendix A

Systematic Literature Review Protocol

The systematic literature review was performed using the guidelines for systematic literature review in software engineering presented in Keele [2007]. The review protocol is presented along with the documentation of each step. The literature review process is divided into 8 steps:

Step 1 *Defining review questions*

Step 2 *Defining the systematic literature review protocol*

Step 3 *Search for relevant studies*

Step 4 *Selection of studies*

Step 5 *Quality assessment*

Step 6 *Data Collection*

Step 7 *Data synthesis and analysis*

Step8 *Dissemination*

A.1 Defining the Review Questions

The first step in the systematic review process was to formalize the the goal of the review into review questions that the review is meant to answer. The goal of the review was to answer the following questions:

RQ1 *What are the mechanisms that allow altruistic behavior to evolve?*

RQ2 *What are the most important factors in determining the degree of altruism displayed?*

RQ3 *Which methods show the most promise in achieving self sacrificial behaviour in artificial evolution?*

A.2 Search for Relevant Studies

To perform the search in a systematic way I compiled a list of of relevant sources which would be the subject to systematic query. I decided to use the list compiled in Lillegraven and Wolden [2010] as a starting point as it presented a list of relevant sources both for research on computer science in general and had already been used to find sources in Artificial Intelligence.

Source	Type	URL
ACM Digital Library	Digital Library	`http://portal.acm.org/dl.cfm`
IEEE Xplore	Digital Library	`http://ieeexplore.ieee.org/`
CiteSeerX	Digital Library	`citeseerx.ist.psu.edu`
Web of Knowledge	Digital Library	`http://wokinfo.com/`
Journal of AI Research	Journal	`http://jair.org/`
References in papers	N/A	N/A

Table A.1: Sources considered in the online search

A.2.1 Searching the online resources

Following the methodology in Oates [2005] I created groups of search terms that were synonyms or similar in meaning. The purpose of this was to exploit the possibility of using boolean search strings in modern digital libraries. The search for relevant literature is a continuous process and I went through a number of different tables of search terms. The table of search terms presented in table A.2.1 is the one I ended up using. The sparsity of the table is a conscious choice as having a general search query and then narrow the results down based on research subject proved a more effective method for finding relevant literature. The search terms were combined in the boolean search string in equation A.1

	Group 1	Group 2
Term 1	Altruism	Evolution
Term 2	Self-Sacrifice	Natural Selection
Term 3		Evolving
Term 4		Evolutionary

Table A.2: Search terms used

$$(Altruism \vee Altruistic) \wedge \\ (Evolution \vee NaturalSelection \vee Evolving \vee Evolutionary) \quad \text{(A.1)}$$

ACM Digital Library

For the ACM Digital Library, the number of results on the original search query was so large that it had to be further limited by only including entries from relevant publications. Of the publications that returned matches for the query, these were included in the final search:

- Proceedings of the 9th annual conference on Genetic and evolutionary computation
- Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion
- Proceeding of the fifteenth annual conference companion on Genetic and evolutionary computation conference companion
- Autonomous Agents and Multi-Agent Systems
- Evolutionary Computation
- Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems
- Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2

- Artificial Life and Robotics
- Proceedings of the 2004 international conference on Multi-Agent and Multi-Agent-Based Simulation
- Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume Two
- Artificial Intelligence
- Autonomous Robots
- Neural Networks
- Artificial Intelligence Review

Springer Link

Springer Link allows filtering on research field, so the search was limited to Artificial Intelligence.

IEEE Xplore

The search string for IEEE Xplore was also limited to the publications

- Evolutionary Computation, IEEE Transactions on
- Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on
- Intelligent System and Knowledge Engineering, 2008. ISKE 2008. 3rd International Conference on

Web of Knowledge

The search on Web of Knowledge was refined to include only results from the research domains Science Technology and computer science.

Search Results

Applying the search string in A.1 to the sources in A.2 yielded the results shown in table A.2.2 In addition to exploring the vast online resources I also searched available literature in the University Library and checked reference lists in the articles I read that were of particular interest if the theme they referenced fit some of my inclusion criteria or if the title alone fit one or more of my inclusion criteria.

Source	Hits
Springer Link	39
CiteSeer	26
ACM Digital Library	25
IEEE Xplore	4
Web of Knowledge	23
Journal of AI Research	0
Other	2

Table A.3: Search results for the search string in equation A.1

A.2.2 Selection of Studies

After applying the search strategy I began selecting the studies that were relevant for my research questions. To filter the number of studies found I employed a three stage screening process where the set of found articles were gradually culled according to a set of inclusion criteria. The three stage process was:

- screening based on title
- Screening based on contents in the Abstract
- Screening based on full-text reading
- Screening based on quality

Screening based on title

The first level of screening was based on excluding articles based on the following criteria:

EQ1 *The main focus of the title is not within the field of computer science*

EQ2 *It can be quickly determined from the title that the focus of the research is neither AI nor theoretical biology related to altruism*

Abstract inclusion criteria screening

The inclusion criteria that were used for the screening based on the contents in the abstract were:

IC1 *The paper focuses mainly on evolving altruistic behavior using artificial evolution*

IC2 *The paper focuses mainly on one of the mechanisms behind the evolution of altruistic behavior in nature*

Before the full text inclusion criteria screening, the search results were as follows:

Source	Hits
Springer Link	3
CiteSeer	4
ACM Digital Library	5
IEEE Xplore	1
Web of Knowledge	5
Journal of AI Research	0
Other	2
Total	20

Table A.4: Search results after applying EQ1, EQ2, IC2 and IC2

Full text inclusion criteria screening

IC4 *The paper focuses mainly on evolving altruistic behavior using artificial evolution*

IC6 *The paper recreates one or more of the settings in which altruistic behavior evolves*

IC7 *The paper studies the genetic preconditions for the evolution of altruistic behavior*

Full text quality criteria screening

QC1 *There is a clear statement of the aim of the research*

QC2 *The Study is put into context of other studies and research*

A.3 Data Collection

Given the exploratory nature of this literature review the data collection consisted of reading the material and noting interesting points.

A.4 Dissemination

Dissemination means communicating the results, in this instance the review was handed in as part of a project.

Appendix B

Property files

B.1 Initial period

```
#
# Properties for roborobo
#


# general file information

#gLogFilename =                 logs/log.txt

gAgentMaskImageFilename =       data/miniagent-mask.png
gAgentSpecsImageFilename =      data/miniagent-specs.png

gForegroundImageFilename =      data/simple\_foreground-2.png
gEnvironmentImageFilename =     data/simple\_environment-2.png
gBackgroundImageFilename =      data/simple\_background-2.png
gZoneImageFilename =            data/simple\_zones.png
gZoneCaptionPrefixFilename =    data/zonecaption

# general purpose

gRandomSeed =                   -1

gVerbose =                      false
gBatchMode =                    true

gFramesPerSecond =              60
gParallaxFactor =               1

gMaxIt =                        80000  # gen*lifeduration
```

```
# general data

gNbOfAgents =                 100

gDisplayZoneCaption =         false

gPauseMode =                  false
gInspectorMode =              false
gInspectorAgent =             false

ConfigurationLoaderObjectName = MedeaAltruismConfigurationLoader

# artificial neural net
nbLayer = 1 #should always remain to 1
nbHiddenNeurons = 5

gEvaluationTime = 400

gEnergyInit = 100
gEnergyMax = 800
gEnergyRevive = 400
gDeadTime = 1.0
gDonationThreshold = 1.1

gZoneEnergy\_maxHarvestValue = 100
gZoneEnergy\_minHarvestValue = 1.1
gZoneEnergy\_maxFullCapacity = 10
gZoneEnergy\_saturateCapacityLevel = 40
gMaxPenalizationRate = 0.5
g\_xStart\_EnergyZone = 0 #700
g\_yStart\_EnergyZone = 212 #0
g\_xEnd\_EnergyZone = 1023
g\_yEnd\_EnergyZone = 535

VisibleEnergyPoint = true

gEnergyMode = true
gMaxEnergyPoints = 800
gEnergyPointRadius = 10.0
gEnergyPointValue = 50.0
gEnergyPointRespawnLagMaxValue = 200 # not used here

gDynamicRespawn = true
gThresholdIncreaseRespawn =  100
gLowestBoundRespawn = 0
gHighestBoundRespawn = 25
exponentialFactor = 4

selectionScheme = pureRandom
gNbMaxGenomeSelection = 3
harvestingScheme = dynCost
fixedCost = 5
```

```
# if respawnlag>0, use non locked version.

VisibleLockedEnergyPoint = true
initLock = 0.0
iterationMax = 40

gEnergyPolar = false

#        gEnergyPointValue = 150.0


# general parameters for the self-adaptive alg. and experiment
gSwarmOnlineObsUsed = true
gDynamicSigma = true
gSigmaMin = 0.01
gProbAdd = 0.5
gProbSub = 0.5
gDynaStep = 0.35
gSigmaRef = 0.1
gSigmaMax = 0.5
gProbRef = 0.5
gProbMax = 0.5
gDriftEvaluationRate = 1.0
gInitLock = 0.0
gDriftLock = 2.0
gMaxKeyRange = 4
gDeltaKey = 2.0
gSynchronization = true



gAgentCounter =                 0
gAgentIndexFocus =              0

gScreenWidth =                  1024
gScreenHeight =                 536


gMoveStepWidth =                1
gMoveStepHeight =               1

gInspectorAgentXStart =         100
gInspectorAgentYStart =         355

# agent dynamics and structure

gMaxTranslationalSpeed =        2 # wednesday 101110 : 2
gMaxTranslationalDeltaValue =   2 # wednesday 101110 : 2
gMaxRotationalSpeed =           30
gSensorRange =                  64

gMaxSpeedOnXaxis =              2
```

```
gMaxSpeedOnYaxis =              10

gLocomotionMode =               0

gInspectAgent =                 false

SlowMotionMode =                false

gAgentRegistration =            true

gNiceRendering =                true

gDisplayMode =                  0
gFastDisplayModeSpeed =         60

gUserCommandMode =              false

# not used
gAgentWidth =                   0
gAgentHeight =                  0
gAreaWidth =                    0
gAreaHeight =                   0



# radio com network info

gRadioNetwork =                 true
gMaxRadioDistance =       32

# danger zone specific parameters (not be displayed in debug.properties)

DangerZone\_InfluenceRadius      100
DangerZone\_RobotDensityThreshold         2
DangerZone\_MaximumVelocityPenalizationFactor    0.5
```

Bibliography

Barta, Z., McNamara, J. M., Huszar, D. B., and Taborsky, M. (2010). Cooperation among non-relatives evolves by state-dependent generalized reciprocity. *Proceedings of the Royal Society B: Biological Sciences*, 278(1707):843–848.

Bredeche, N., Montanier, J.-M., Weel, B., and Haasdijk, E. (2013). Roborobo! a fast robot simulator for swarm and collective robotics. *CoRR*, abs/1304.2888.

Dawkins, R. (2006). *The Selfish Gene: 30th Anniversary edition*. Oxford University Press.

Dessalles, J.-l. (1999). *Coalition Factor in the Evolution of Non-Kin Altruism*.

Dulk, P. d. and Brinkers, M. (2000). Evolution of altruism in viscous populations: Effects of altruism on the evolution of migrating behavior. In Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J. J., and Schwefel, H.-P., editors, *Parallel Problem Solving from Nature PPSN VI*, number 1917 in Lecture Notes in Computer Science, pages 457–466. Springer Berlin Heidelberg.

Floreano, D., Mitri, S., Perez-Uribe, A., and Keller, L. (2008). Evolution of altruistic robots. In Zurada, J. M., Yen, G. G., and Wang, J., editors, *Computational Intelligence: Research Frontiers*, number 5050 in Lecture Notes in Computer Science, pages 232–248. Springer Berlin Heidelberg.

Hales, D. (2005). Change your tags fast! - a necessary condition for cooperation? In *Proceedings of the 2004 International Conference on Multi-Agent and Multi-Agent-Based Simulation*, MABS'04, page 89–98, Berlin, Heidelberg. Springer-Verlag.

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

Hamilton, W. (1964b). The genetical evolution of social behaviour. II. *Journal of Theoretical Biology*, 7(1):17–52.

Han, T. A., Pereira, L. M., and Santos, F. C. (2011). The role of intention recognition in the evolution of cooperative behavior. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, page 1684–1689, Barcelona, Catalonia, Spain. AAAI Press.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition.

Joshua Mitteldorf and Wilson, D. S. (2000). Population viscosity and the evolution of altruism. *Journal of Theoretical Biology*, 204(4):481–496.

Keele, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical report, EBSE Technical Report EBSE-2007-01.

Lehmann, L. and Keller, L. (2006). The evolution of cooperation and altruism – a general framework and a classification of models. *Journal of Evolutionary Biology*, 19(5):1365–1376.

Lillegraven, T. N. and Wolden, A. C. (2010). *Design of a Bayesian recommender system for tourists presenting a solution to the cold-start user problem*. PhD thesis, Norwegian University of Science and Technology.

Martijn Brinkers and Dulk, P. d. (1999). The evolution of non-reciprocal altruism. In Floreano, D., Nicoud, J.-D., and Mondada, F., editors, *Advances in Artificial Life*, number 1674 in Lecture Notes in Computer Science, pages 499–503. Springer Berlin Heidelberg.

Mayoh, B. (2000). *Evolution of Altruism*.

Montanier, J.-M. (2013). Environment-driven evolution for robotic swarms.

Montanier, J.-M. and Bredeche, N. (2011). Surviving the tragedy of commons: Emergence of altruism in a population of evolving autonomous agents.

Oates, B. J. (2005). *Researching Information Systems and Computing*. SAGE.

Ozisik, A. P. and Harrington, K. I. (2012). The effects of tags on the evolution of honest signaling. In *Proceedings of the Fourteenth International Conference on Genetic and Evolutionary Computation Conference Companion*, GECCO Companion '12, page 345–352, New York, NY, USA. ACM.

Spector, L. and Klein, J. (2006). Genetic stability and territorial structure facilitate the evolution of tag-mediated altruism. *Artificial Life*, 12.

Trivers, R. (1971). The evolution of reciprocal altruism. *The Quarterly Review of Biology*, 46(1):35–57.

Turner, H. and Kazakov, D. (2003). Stochastic simulation of inherited kinship-driven altruism. In Alonso, E., Kudenko, D., and Kazakov, D., editors, *Adaptive Agents and Multi-Agent Systems*, number 2636 in Lecture Notes in Computer Science, pages 187–201. Springer Berlin Heidelberg.

W. D. Hamilton (1963). The evolution of altruistic behavior. *The American Naturalist*, 97(896):354–356. ArticleType: research-article / Full publication date: Sep. - Oct., 1963 / Copyright © 1963 The University of Chicago.