# Complex Behavior Emerging from Primitive Reward-Cost Dynamics

Armaghan Behlum and Alexander Moore

*Abstract*—**Predator / prey relationships are pretty common in nature. Their existence serves as a pressure on both predator and prey to evolve and perform better, which is what we call co-evolution. Previous research has used co-evolution to optimize solutions for common problems with results that outperform regular evolutionary algorithms and solutions by people. Our work shows how co-evolution can be used to generate complex behavior in simulations. We see artificial animals herd prey, rush predators, and kill their competitors for food and all of this is achieved through evolution using genetic algorithms on a simple neural network.**

## I. INTRODUCTION

Co-evolution is the process by which two species evolve in relation to one another. A change in one of the species can act as a pressure for the other to evolve as well. More complex behavior might arise from these pressures, but the initial conditions of the world needed to develop complex behavior seem to be very simplistic in nature. We used a simulated world in which there were two species, A and B, and two types of plants, PL1 and PL2. Both species have an internal neural network which is used to make decisions about the world. We evolved the weights on the neural network to see how our species would pressure each other. Also, each of the species have mappings of the reward they will receive when they eat something in the world. The goal was to see if complex species relationships and interactions could arise from just the simple mappings of food to reward and the neural networks.

### A. Inter-Species Dynamics

There are some inter-species dynamics in nature, including:

1)  Predator/prey - one species hunts and eats another. An example of co-evolution would be the prey species evolving to be faster and in response the predator species evolving to either to be faster or develop a strategy to reduce the advantage of speed. In nature, the fringed rue developed an oil to repel bug eating insects but then the old world swallowtail caterpillar developed a resistance to the oil. This caused it to have the exclusive ability to eat the rue.
2)  Parasitism - one species gains benefits with negative effects on the other. Co-evolution would occur if the host species develops a defensive layer against letting the parasite species in, but then the parasite species evolves to use better tools to invade the hosts. This is similar to predator / prey relationships but it is not advantageous for the parasite to kill the host species.

3)  Mutualism - both species benefit from one another. Clownfish and sea anemones co-evolved their mutualistic behavior since clownfish are provided protection from predators by the sea anemone while the sea anemone is protected from its parasites by the clownfish which eats the parasites.
4)  Commensalism - one species benefits from another without affecting that species. This would be similar to just seeing regular evolution of the first species.
5)  Competition - two species compete for the same resource. Interesting evolutionary scenarios might involve one harming the other or in some way making the other unable to access the shared resource.

This project mostly focused on developing complex predator/prey behavior from primitive costs and rewards in our simulated world, though we also explored a little bit of competition. We wanted to see how each of the species would evolve in relation to one another given some simplistic mapping of food to costs in our game world. Also, we had already seen instances of a predator / prey dynamic being used in research, such as the work by Hillis (1992), in which he evolved a sorting algorithm by evolving it against a generating algorithm that made test cases for the sorting algorithm. Given the context of the foundational research into coevolution, we wanted to see how easy it was to develop the complex behavior we saw in nature.

## II. PREVIOUS WORK

Artificial co-evolution has already shown significant promise as an optimization procedure and research tool. In one of the first papers concerning artificial co-evolution, Ray (1992) evolved self-replicating machine code without the use of a fitness function. Sub-populations of hosts and parasites soon emerged and began evolving with each other in accordance with biologically observed host-parasite dynamics. Another paper by Hillis (1992) co-evolved sorting networks and test cases, eventually generating a sorting network that performed only slightly worse than the best networks developed by humans. Artificial co-evolution has also been used to study of numerous biological phenomena for research purposes, including mutation rates and gene complexity (Floreano et al. 2008, pg 559).

In co-evolution, the continuously changing fitness landscape has promising qualities. Intuitively one hopes to encounter an evolutionary "arms race" whereby each population performs better than they would have otherwise as a result of competition. In a similar manner, this process results in

increased genetic diversity and thus a lower likelihood of getting stuck in local minima. However, there are several disadvantages to this changing landscape which must be circumvented. The first is called the *Red Queen Effect*. The problem is that the fitness landscape is dependent on both populations, and thus changes with each generation. Because of this, a solution that was effective in one generation can become ineffective in later ones (Floreano et al. 2008, pg. 561). Moreover, the meanings of fitness values become ambiguous. A decrease in fitness for population A could mean that A got worse or B got better. Even worse, generation to generation fitness values could be decreasing or stagnant while overall effectiveness is actually increasing (for example, two evenly matched competitors evolve increasingly effective strategies together). From this we see that fitness values no longer have absolute, global meaning.

One solution for establishing global fitness is the Hall of Fame method, proposed by Rosin and Belew (1997). This method involves recording a copy of the best individuals with each generation and determining future fitness based on performance against this population. Future fitness can be determined by testing the individual against all members of the Hall of Fame or against a fixed-size random sample. Since our individuals are actually populations with homogeneous genotypes, we would record the genotype of the best population with each generation and test future populations against those.

Floreano and Nolfi (1997) also noted that there were multiple unnecessary difficulties in other research that explored co-evolution. Among them were the ideas that the "genetic encoding" could be too complex (resulting in complex fitness landscapes), and that the agents were too similar to actually exploit weaknesses in one another. The problem of genetic complexity can be addressed through the use of a feed-forward neural network (Floreano and Nolfi, 1997). Using this encoding, a genetic string has only as many parameters as the network has weights. By varying the number of hidden layers, an arbitrary level of complexity can be achieved. The problem of similarity can be addressed by creating populations with heterogeneous abilities, just as Floreano and Nolfi (1997) varied the sensorimotor abilities of their agent populations.

## III. MODEL

### A. Neural Networks

To facilitate decision making for each animal, a fully connected feed-forward neural network with no hidden layer (perceptron) was implemented. Previously a hidden layer had been included, but convergence to intelligent behavior was never observed under this model, probably due to excessive complexity in the genetic representation. The perceptron model included 16 inputs and 5 outputs. There were 4 directions to consider relative to the agent (North, South, East and West), and for each direction there were 4 inputs, one for each type

of object in the world (Species A, Species B, Plant 1, Plant 2). This resulted in 16 different inputs, where each input stored the count of a particular object in a particular direction. Counts were discounted according to distance such that an object located $d$ squares away from an agent added a value $\frac{1}{d}$ to the count. Including separate inputs for each level of distance was considered, but that idea was discarded since each distance would double the complexity of the network. The network contained 5 outputs, one for each possible action (move North, move South, move East, move West, Stay). The action to take was simply the action corresponding to the output node with the highest value. The genome contained the network weights and was thus directly related to network complexity. For our case, there were 80 weights to evolve on. Weights were initialized with random values in $[-1, 1]$. A diagram of the network is given in Figure 1.
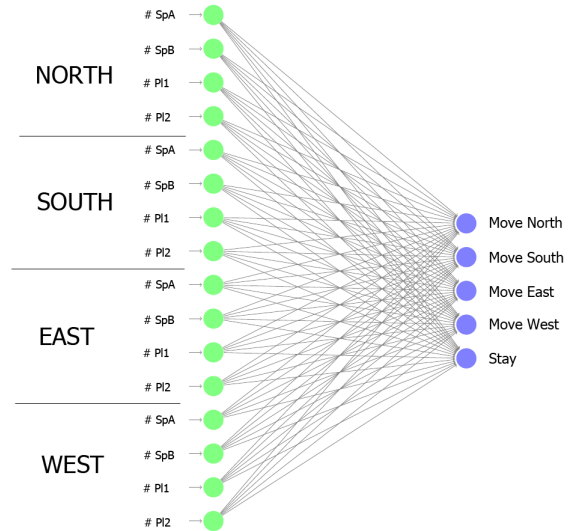


Fig. 1: A diagram of the neural network. The input nodes are on the left. Note how there are 4 directions with 4 counts each (the number of objects of each type in each direction weighted by distance) for a total of 16 nodes. There are 5 output nodes on the right, and the node with the highest value is the action to take. Each edge from an input to an output is weighted.

### B. Evolution

The evolutionary process was governed by a genetic algorithm. A population of 40 genomes for species A competed against a population of 40 genomes for species B. Each genome for A competed against every genome for B, and a fitness value was computed for each species in each competition. This fitness value for a genome in an individual competition was equal to the average number of points acquired by all animals using that genome. The total fitness value was simply the largest individual competition fitness value for a given genome. With each generation, the genome with the highest fitness for a given species was added to a Hall of Fame. The

Hall of Fame held at most 5 genomes, and when it was full one was randomly selected to be removed. With each generation, a random member of the Hall of Fame was re-introduced into the population.

## C. Genetic Algorithm

Selection was performed in proportion with fitness, such that a genome $i$ with fitness $f_i$ in a group of $N$ genomes had probability $\frac{f_i}{\sum_{j \in N} f_j}$ of being selected. To keep the probabilities between 0 and 1, all fitness values were translated by a constant such that the smallest fitness value was 1. Upon selection, an 8-point crossover was used, and mutation was performed independently on each weight according to a Gaussian distribution with the current weight as the mean and a variance of $0.5$. Mutation occurred with probability $p = 0.10$ for each weight, and two offspring were produced from each parent pairing. With each generation, the population size remained constant.

## D. Game world

Our game world was represented as a 10 x 10 grid with species having the ability to show up on the other side of the grid by crossing its edge. In a single game, there are 5 instances of species A, 5 of species B, 10 of PL1, and 10 of PL2. Each animal of species A has the same encoding for the weights on their internal neural network while the animals of species B have a different encoding from A. These encodings are their "genome" which we evolved. Each of the 40 genomes of A played a game with each of the 40 genomes of species B. Each of the species also had an associated food mapping. Species A might receive 100 points for eating an animal of Species B, 50 points for eating PL1, 0 points for eating PL2, and -100 points for eating an animal of species B. An eating action was taken whenever one animal in the game moved into a space of the world where something else already existed. A single animal instance would be able to take in as input information about what existed in the space that formed a 2 block radius around it. This input was fed to our neural network and from there the animal made a decision about going North, South, East, West, or staying still. Movement had a very small associated cost while dying had a very large associated cost (a medium explicit cost plus an implicit opportunity cost). Internally, all the animals kept track of how many points they collected in their life in the world. The duration of life was the 100 steps in each game though some animals had their life cut short if they were eaten by another one. The ordering of actions was shuffled each time, so no one animal had the advantage of going first in the world. Once each of the 40 genomes of each species finished playing each other, we called that a single generation of the evolution. The next generation was then produced according to the evolutionary procedure previously outlined. The evolution process would run for 1000 generations given a single food mapping parameter set. Then, the evolution process was started from scratch for each food mapping. That means the neural networks were randomized once again and we allowed the species to develop different strategies for what

might be a very different world. These were the normal games we played, but we also had special games where the species might have different abilities, such as one species might move faster than another or see farther than the other.

## E. Food mappings and Abilities

The following are the food mappings we used:

| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 0 | -100 | 100 | 0 | 0 | 0 | -100 | 100 | 50 |
| 1 | -100 | 0 | 100 | 50 | 100 | -100 | 100 | 50 |
| 2 | -100 | 0 | 100 | 50 | 50 | -100 | 100 | 100 |
| 3 | -100 | -100 | 0 | 50 | -100 | -100 | 50 | 0 |
| 4 | -100 | -100 | -50 | 50 | -100 | -100 | 50 | -50 |
| 5 | -100 | 0 | 50 | 0 | 0 | -100 | 50 | 0 |
| 6 | -100 | 50 | 50 | 0 | 50 | -100 | 50 | 0 |
| 7 | -100 | 50 | 100 | 0 | 50 | -100 | 100 | 0 |
| 8 | -100 | -50 | 100 | 0 | -50 | -100 | 100 | 0 |
| 9 | -100 | -100 | 50 | 0 | -100 | -100 | 50 | 0 |
| 10 | 100 | 100 | 0 | 0 | -100 | -100 | 50 | 0 |
| 11 | 100 | -100 | 0 | 0 | 100 | -100 | 0 | 0 |
| 12 | -100 | -100 | 0 | 100 | -100 | -100 | 100 | 0 |

So, in mapping 0 when an animal of species B eats an instance of PL1 it gets 100 points. These settings were picked to influence certain types of worlds. In mapping 0, A is encouraged to eat B and B eats plants. In 1 and 2, A is now the herbivore while B eats both A and plants but with slightly different reward schemes. For 3 and 4, both species were driven to eat different kinds of plants while 4 also makes it a penalty to eat the other species' plants. Mappings 5 to 9 have them competing for the same food but differ in how they penalize eating each other. For 10 and 11 they both like to eat A, which means A is at least a little cannibalistic. And finally, in the last case they strongly prefer to eat different food. These settings were run with the normal ability set:

| Abilty | SPA | SPB |
|---|---|---|
| sight range | 2 | 2 |
| speed | 1 | 1 |

Having a sight range of 2 means that the species can see 2 blocks around them in all directions. Having a speed of 1 means the animal can move 1 block per step.

We also ran the following two food mapping sets with 6 varying ability sets:

| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 13 | -100 | 100 | 0 | 0 | -100 | -100 | 100 | 50 |
| 14 | -100 | -100 | 100 | 50 | 100 | -100 | 0 | 0 |

The idea in these cases was to influence one species to heavily hunt the other and we wanted to see if interesting hunting behavior would arise given different sets of abilities. The different ability settings were:

|   | SPA | | SPB | |
|---|---|---|---|---|
|   | Sight Range | Speed | Sight Range | Speed |
| A | 2 | 2 | 4 | 1 |
| B | 4 | 2 | 4 | 1 |
| C | 2 | 2 | 2 | 2 |
| D | 4 | 1 | 2 | 1 |
| E | 2 | 2 | 2 | 1 |
| F | 2 | 1 | 2 | 2 |

So in some cases SPA can go faster than SPB or look farther or the reverse might be true. And finally we had these final two cases:

|   | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
|   | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 15 | -100 | 100 | 0 | 0 | 0 | -100 | 100 | 50 |
| 16 | -100 | -50 | 100 | 0 | -50 | -100 | 100 | 0 |

For mapping 15 we ran a set where SPB moves faster. For mapping 16 we ran a set where SPB sees farther and SPA moves faster. We also ran another case that was exactly the same but SPA could also see farther.

## IV. DATA

At the end of the evolution, we recorded the final generation of genomes and their associated fitness values as well as the diet of each of species in each of the games. The diet consisted of the total count of what a generation of one species ate in the game.

### A. Records and Project Code

All files recording the diet and fitness of the last generations of genomes can be found in our Github repository, linked here[1] along with instructions on how to generate bar graphs, simulations, and read the most interesting files.

### B. Videos

The repository also contains a link to a YouTube playlist that contains all the videos of the simulations we thought produced interesting behavior. You can find the link on the README. All videos were recorded in a 50x50 world with 100 members of each species and 100 of each plant type. While the genomes themselves evolved in a 10x10 world, we thought that given the limited sight range of the animals, the evolved behaviors could extend to a world of arbitrary size.

## V. DATA ANALYSIS

### A. Herding, Rushing and Packs

Our first simulations used the normal set of abilities. That means at each time step, an animal could move one block and for input it used information from squares that were in a 2 block radius around it. We saw very interesting behavior from food mapping 0, defined as so:

---
[1]https://github.com/Armi23/AICoevolution

|   | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
|   | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 0 | -100 | 100 | 0 | 0 | 0 | -100 | 100 | 50 |

The simulation for this can be found in the "Control Simulation" video. In this case, Species A is rewarded for eating Species B and Species B likes to eat plants. Their dietary results can be found in Figure 2 from which we see that A ate a large number of B while B mostly ate plants in the top performing final generations. However, it is interesting to note that B also ate plenty of animals that belong to species A. When we view the simulation, we see that some interesting behavior evolved in both species. Members of Species A learned to crowd around members of Species B. This would cause the B animals to flee while the A animals herded them in a fixed direction. Doing so meant that eventually when an A animal appeared in front of the B animal, the B animal would have no where to escape and it would get eaten. This herding behavior by A is a pretty effective hunting pattern which eventually causes the extinction of Species B. However, B also learned to fight back against A and did what we call rushing species A. When single instances of A and B encounter each other in the world and A makes the first move closer to B, B is a step away from A. In these cases, we noted that B decides to take the step to eat A. It does this because it is not penalized in this world for eating A (it receives 0 reward for it) but the penalization for having to run or dying are both high enough that its more advantage for the B to just eat the A animal. That is why A is such a large part of Species B's diet. This rushing mechanism was clearly evolved as a response to A's evolution for hunting B. Both herding and rushing behaviors were interesting because we can also trace where they are not as useful.

We first tried to adjust the costs to see if herding behavior would persist. To do so, we rewarded A for eating other A's, which is food mapping 10 like so:

|   | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
|   | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 10 | 100 | 100 | 0 | 0 | -100 | -100 | 50 | 0 |

The simulation for this is under the "No Herding Behavior" video. When we did so we saw that A animals were afraid to go near one another. They did not group together as well as they used to in our previous example and that affected their overall performance at hunting B animals since they received lower averages (Figure 3). Also, interestingly B animals no longer evolved the rushing behavior we saw before compared to the control, most likely because they were not pressured into doing so by the ineptitude of the A animals.

On the other end of the spectrum, we saw a variant form of herding that we called packing. This arose when we had B hunt A and but it also ate plants. This was food mapping 2:
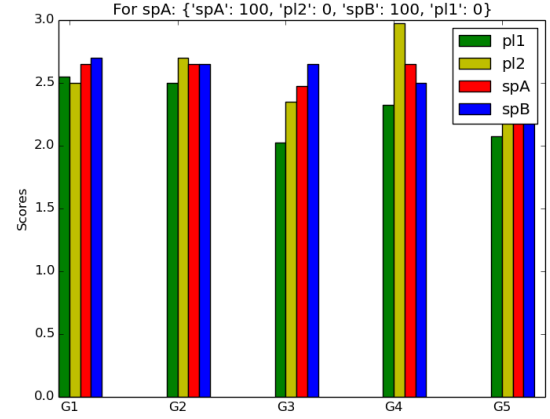
|   | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
|   | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 2 | -100 | 0 | 100 | 50 | 50 | -100 | 100 | 100 |

While with herding, the predators would form a group around the prey and drive them in one direction, in packing
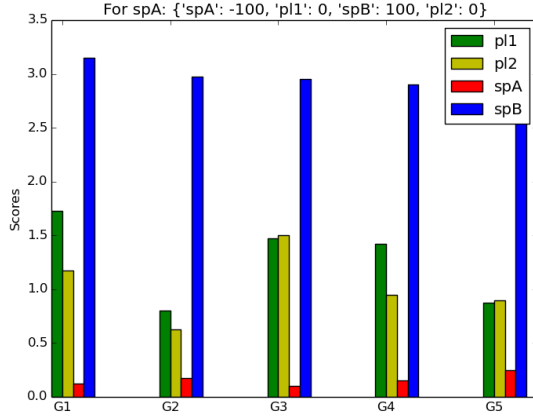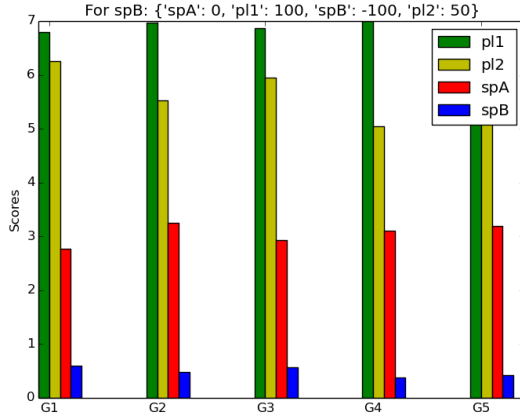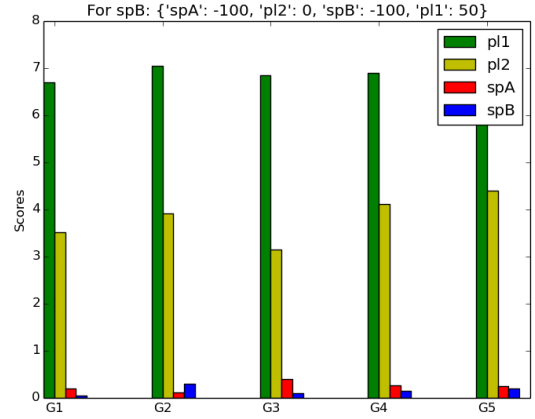
(a) The average amount of each object consumed by the top 5 genomes of species A



(b) The average amount of each object consumed by the top 5 genomes of species B
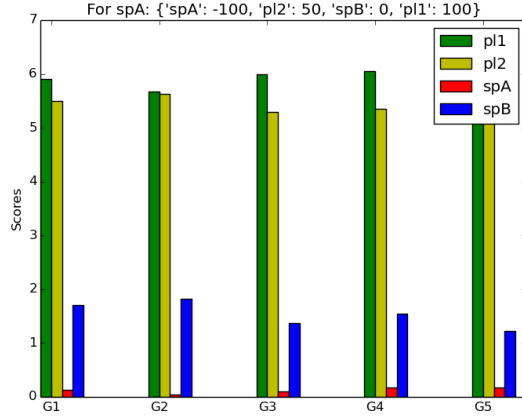
Fig. 2: Results for Control Simulation



(a) The average amount of each object consumed by the top 5 genomes of species A



(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 3: Results for No Herding Behavior

what we saw instead was the predators forming big groups that would in unison move in one direction. This behavior did not seem to be as useful as herding for just eating the prey since the predators (in this case species B) had a lower average for animals they ate (Figure 4) and in the simulation we see a lot more of the prey alive. However, it was definitely a useful strategy because the predators were able to attack prey together which reduced its chances of escaping and exploring the world in this way was very effective at finishing the plants in the world. The simulation for this one can be found in the "Hunting Pack" video. Again though, we did not see rushing behavior form, most likely because the predator was not as effective at hunting its prey.

## B. Food and Penalties

Beyond just predator / prey relationships, we wanted to also see how our simulated animals would evolve if they competed for food. First we tried having the animals hunt for food but avoid each other, which we hoped would be mapping 3:

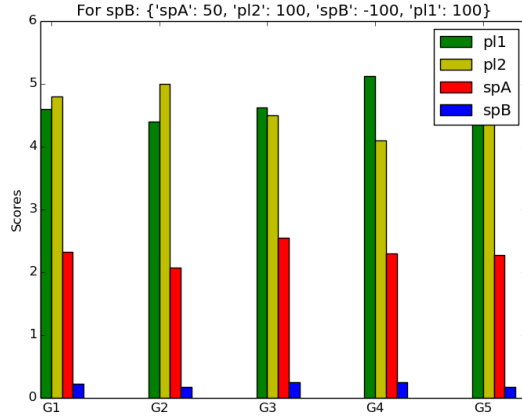| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 3 | -100 | -100 | 0 | 50 | -100 | -100 | 50 | 0 |

However, this lead to the creation of a boring world in which all the animals just moved around a little bit in a local area. The animals were too afraid to eat one another and be heavily penalized so it was better to not move around too much. The simulation for this can be found in the video titled "Over Penalized." Having learned from this, we decided to reduce
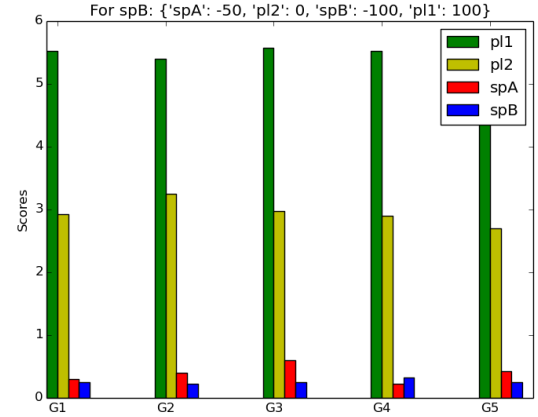
(a) The average amount of each object consumed by the top 5 genomes of species A
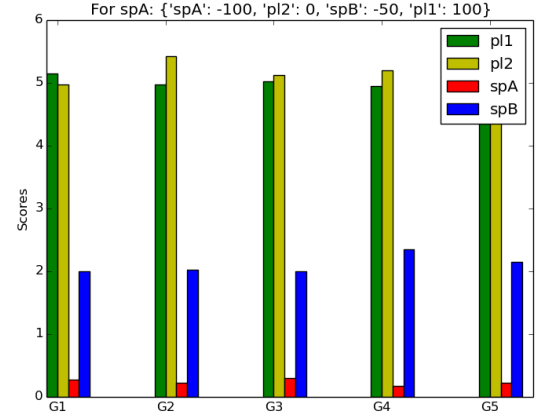


(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 4: Results for Hunting Pack



(a) The average amount of each object consumed by the top 5 genomes of species A



(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 5: Results for Avoiding Each Other

the penalties for eating members of the other species. We also had them compete for the same resource in this scenario. This was mapping 8:

| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 8 | -100 | -50 | 100 | 0 | -50 | -100 | 100 | 0 |

Watching the simulation of this world ("Avoiding Each Other"), what we see is that PL1 is quickly extinguished and A and B do in fact try to avoid each other, as we had wanted in the last simulation. Because of this, the world continues to be a dense world to the end. This is also shown in their diet graph (Figure 5) where both species ate only small amounts of each other. We were curious about what would happen if we reduced the penalties even more which was mapping 5:

| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 5 | -100 | 0 | 50 | 0 | 0 | -100 | 50 | 0 |

With this parameter, we saw some surprising behavior. As we see in Figure 6 and the "Kill Competition" simulation video, species A still learned to hunt species B, even though it did not directly receive points for doing so in this world. In fact, in the simulation we see A is surprisingly aggressive in attacking B and B runs away from A, which means we surprisingly evolved a predator / prey dynamic in a world that does not even directly reward the predator for hunting.
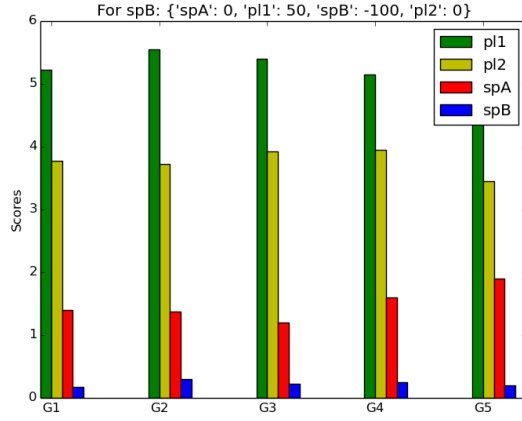
### C. The Effect of Speed

We will now examine the case where A can move twice as fast as usual (2 squares at a time), while B remains
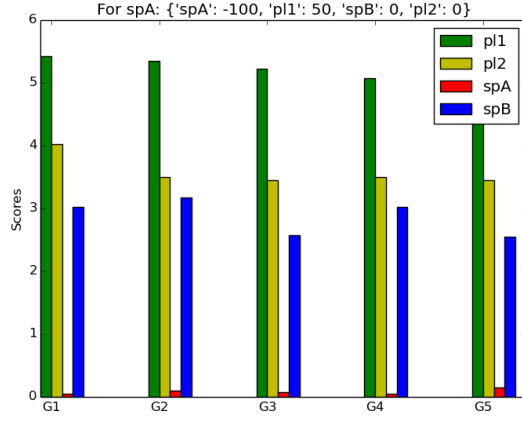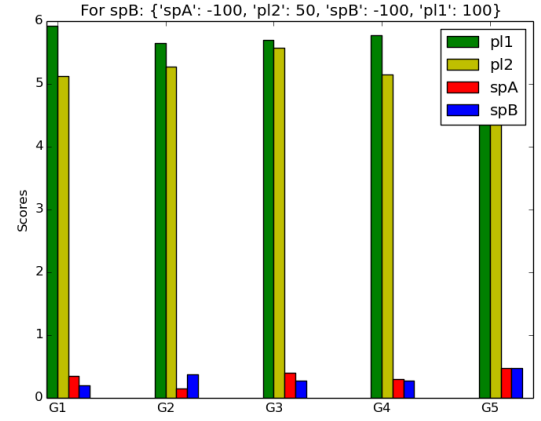
(a) The average amount of each object consumed by the top 5 genomes of species A
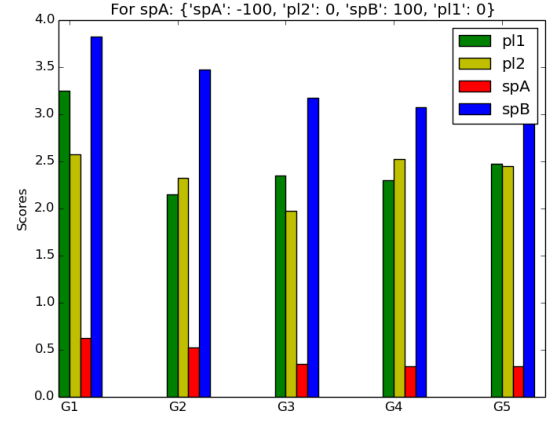


(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 6: Results for Killing Competition



(a) The average amount of each object consumed by the top 5 genomes of species A



(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 7: Results for Fast Predator

unchanged. Our results show that this new ability can be either an advantage or a harm depending on the role of Species A in the system.

In the first case we consider, Species A receives a reward for eating Species B, while Species B only receives a reward for eating plants. Thus, A is a predator of B. The food reward mappings are as follows (13 from the earlier list):

| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 13 | -100 | 100 | 0 | 0 | -100 | -100 | 100 | 50 |

The results can be seen in Figures 7a and 7b (video: "Fast Predator"). From these figures, note that A tends to consume

more B's than it does plants or A's, while B consumes more plants than A's or B's. Thus, each species exhibits preferences towards objects that provide rewards as expected. In the video, we see that A has learned to quickly overtake B. Since A moves two squares at a time, it is able to consume B in the next step after appearing at the edge of B's vision. As a result, B has little time to react. Although the B's do try to flee, A is able to quickly clear the board of B's using this strategy.

In the second case we consider, the roles are flipped. B is now a predator of A, and A still moves quickly. The food reward mappings are as follows (14 from the earlier list):

| | When SPA eats | | | | When SPB eats | | | |
|---|---|---|---|---|---|---|---|---|
| | SPA | SPB | PL1 | PL2 | SPA | SPB | PL1 | PL2 |
| 14 | -100 | -100 | 100 | 50 | 100 | -100 | 0 | 0 |
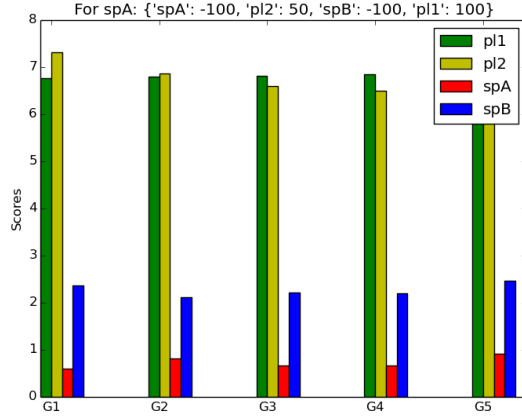
(a) The average amount of each object consumed by the top 5 genomes of species A
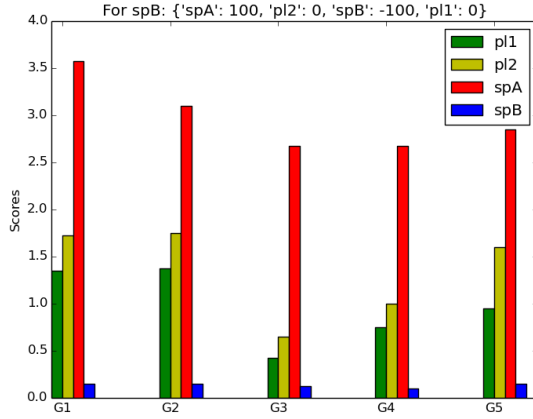


(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 8: Results for Waiting

The results can be seen in Figures 7a and 7b (video: ”Waiting”). From these figures, note that B is still quite successful at consuming A despite A's increased speed. We found this counterintuitive– a faster prey should have no trouble evading a slower predator. Upon watching the simulation, however, it becomes clear that B has actually used A's speed to its advantage. Rather than chasing after A in a pack or herd, B tends to sit still (or stay in a constrained area) and quickly consume any A's that pass by. Since A has normal sight range but moves quickly, A cannot see beyond its immediate destination when moving, rendering it more susceptible to a waiting strategy like the one executed by B. This strategy of waiting to strike is rather common in nature. Venus flytraps and other camouflaged predators will often wait for their prey to arrive before catching them by surprise.

*D. The Effect of Speed on Rushing*

Recall the ”rushing” behavior mentioned in the first section, whereby the prey ate the predator in order to avoid being eaten. Now we observe that when that prey has increased speed, this rushing behavior is even more prominent.

We first consider when A is a predator of B under parameter set 13 again. In this case, B has increased movement speed.



(a) The average amount of each object consumed by the top 5 genomes of species A



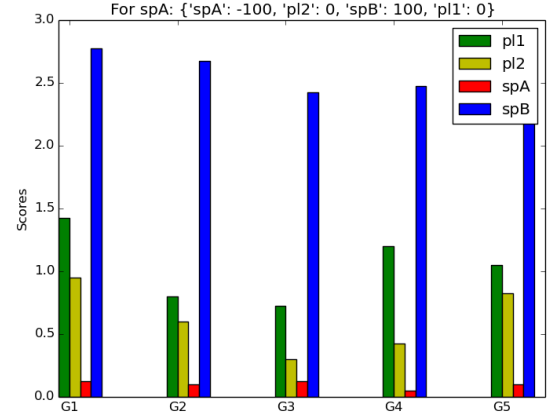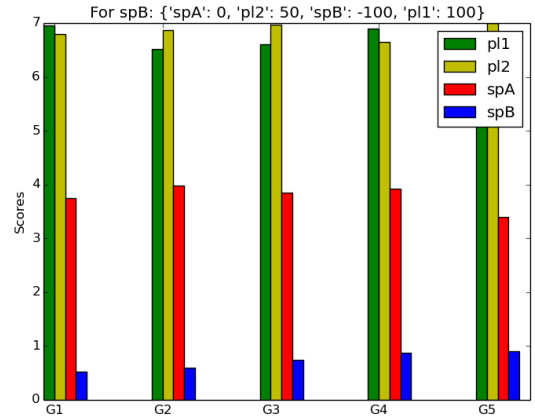(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 9: Results for Rushing: Speed Advantage

The results can be seen in Figures 9a and 9b (video: ”Rushing: Speed Advantage). Note that even though A was the predator, species B ate more A's than species A ate B's. There are several reasons to consider. First, rushing the predator becomes less risky with higher speed. If an A and a B come within sight range of each other and each decides to eat the other, B will always win due to speed. Second, rushing is cost-effective. Even though there is a penalty for eating

members of species A, the opportunity cost of dying is greater. Additionally, the penalty can be offset by eating other plants. Since rushing is guaranteed to succeed, A's are consumed by B's at a faster rate than in the more ambiguous rushing case where both species moved at the same speed.

We now consider when B is a predator of A under food mapping set 14. In this case, A has increased speed and vision.



(a) The average amount of each object consumed by the top 5 genomes of species A



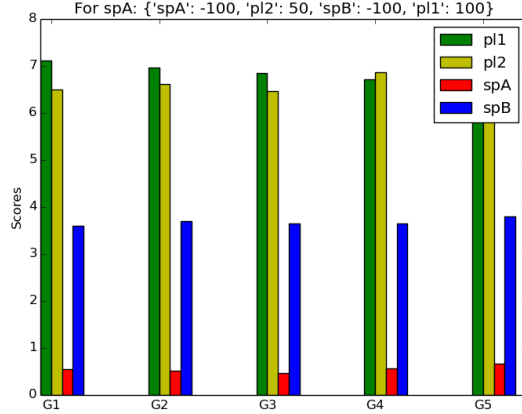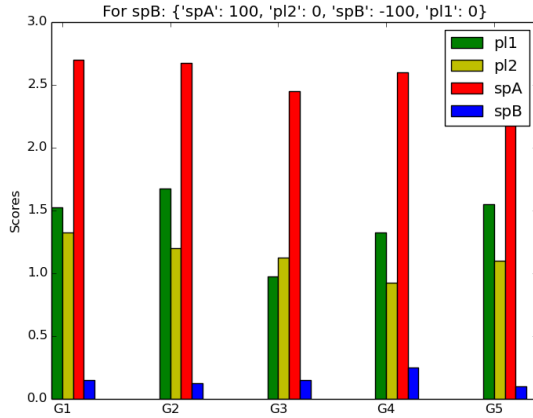(b) The average amount of each object consumed by the top 5 genomes of species B

Fig. 10: Results for Rushing: Speed and Sight Advantage

The results can be seen in Figures 10a and 10b (video: "Rushing: Speed and Sight Advantage"). Similarly to before, species A, despite being the prey, consumes many members of species B. Watching the video, A typically avoids B, but in some cases A will consume B. Given A can see farther and move faster than B, this strategy is especially likely to succeed.

*E. Miscellaneous Behavior*

There were a couple other interesting behaviors we observed. In one case, we wanted to see how a predator with a longer sight range would interact with a faster prey (video: "Moving diagonals"). Here, Species A was the predator under food mapping set 13. Species A could see farther, while species B could move faster. The hypothesis was that A would be able to employ a more complex strategy than usual, taking advantage of its sight advantage to compensate for B's speed advantage. From the video, we see that A was successful in hunting B. Interestingly, A tended to form groups of roughly diagonal lines together, eating members of species B as they came by, a strategy reminiscent of a moving fishing net. Thus, there was evidence that improved vision led to an ability to create more complex strategies than simple packing or herding behavior.

Another interesting scenario occurred when both Species A and B moved quickly and A was a predator of B under food mapping set 13 (video: "Aggressive Speed") . In this case, B was defeated very swiftly by an aggressive A. In the case where only the prey could move faster, we posited that the increased movement speed was prone to traps due to limited sight range. In this case, however, this limitation did not seem to affect the predator. Species A was still capable of quickly dispatching B, perhaps suggesting that the disadvantages cancelled each other out.

## VI. DISCUSSION AND FUTURE WORK

This project mostly focused on predator / prey relationships but we were not able to develop them to the extent that we wanted. Though we saw interesting behaviors in the last generations of lots of the simulations, we had a lot more simulations that did not develop anything interesting. This might just be a limit in the number of iterations that we used to get our final results, but given the scope of the project and the time we had to work on it, we were forced to limit our steps.

The parameter settings were not picked randomly but instead we specifically picked the ones that we thought would lead to interesting behavior eventually. Originally we wanted to iterate through a larger space of parameters but realized that doing so would lead to redundant runs because if we swapped the parameters between two species that would technically be another set but we would not expect a different world or interesting insight. Also, there were just too many parameter sets to run. A parameter set could take up to 4 hours to run and so picking the ones that looked most promising was very important. Given more time, we might have allowed iterations.

We would have also wanted to run more matches between genomes and rerun parameter sets more to verify results. Again, time constraints were the greatest limitations on what we could accomplish with this, but this metric might have lead to better overall results.

We picked perceptron neural networks because larger neural networks explode in the number of parameters. This slows down evolution and would have meant an infeasible number of iterations necessary just to get simple expected behavior. In fact, this is what we tried for the first few cycles of our project and in all of those cases we could not differentiate our evolved genomes and the randomly created ones we had at the start of each simulation. Additionally, note that our perceptron model was completely deterministic given a set of inputs. One interesting variation might have been a probabilistic model whereby an action was taken probabilistically weighted by the values of each output node. Perhaps introducing confidences in this manner would give rise to better informed decisions (ex. perhaps all actions are equally good while in open space, so acting probabilistically could lead to better exploration of the world).

There were many cases in which increased sight range led to worse performance (these cases were deemed "uninteresting" but can be viewed on GitHub). We suspect that this has something to do with our encoding of distance information. Under our model, one nearby food item could give the same input as two farther away food items, potentially leading to ambiguous inputs for the neural network. Future investigations might include new input nodes for each distance, removing any ambiguity but possibly increasing the number of genetic algorithm iterations necessary for convergence. Another suspicion is that the evolutionary world was too small to accommodate large sight ranges. In a 10x10 world, seeing 4 squares on either side means most of the world is visible. It also means that objects can suddenly appear on either side in potentially unexpected ways, potentially hindering the evolutionary process.

Another interesting route for exploration would be to differentiate the species even further. We tried to do a little bit of this when we allowed some animals to see farther or move faster. This lead to some interesting results but there were a lot more ideas that we had. We were thinking about allowing species to be stationary as they processed and ate food. Think might lead to either predators actively looking for stationary animals to eat or parasites waiting for other species to process food before they eat the food. Another idea would have been to allow for some form of communication between the animals to see if they coordinate. It would have made an interesting comparison to what we termed as herding and packing in this project because what we saw in the earlier simulations was done without explicit communication. It was instead movement in unison derived from just knowledge of the presence of other animals. However, aside from this making the simulations more computationally expensive it would also escape the scope of this project which aimed to derive complex behavior from simple worlds.

Finally, we are concerned about our results because quantifying behaviors is difficult and subjective. It is entirely possible that we have projected existing processes we see in nature onto the patterns we generated. We tried to quantify the information objectively in terms of diets of the creatures, but that does not capture the entire story of their interactions. Two species can have the same bar charts even if one just spreads randomly across the world and one actively hunts its food but has a strategy that does not perform too well in a sparse world. To capture that information we had to visually observe our simulations, but doing so is subjective and takes a lot of time, which limits the number of simulations we can observe. A better way to do this would be to calculate distances between animals and so define interesting behaviors based numbers calculated from these interactions. However, we were not sure what patterns we could derive from these numbers since we did not really know what what types of numbers we would need to see a behavior we would classify as say herding. Therefore, we just made observations about the simulations and did our best to only keep the ones we felt strongly represented some type of behavior.

## VII.   CONCLUSION

In conclusion, it is possible to develop complex relationships we see in nature from very simple dynamics. This is immediately clear when we see how easy it was to derive herding, rushing, and packing behavior from our earliest simulations. Those are calculated and coordinated patterns we would generally expect from more intelligent creatures, but instead we see them appear relatively quickly in the evolutionary process of our simplistic models. Importantly, we also observed the rise of predator / prey behavior even when the predator was not rewarded directly for hunting its prey. Therefore, despite not having direct advantages from it, the simulation evolved to realize it could kill its competition for its own benefit and was as effective at the job as a normal predator from our first simulation. Seeing such promising results, it would be interesting to investigate what other behavior patterns arise simply from nature.

## VIII.   PROJECT CONTRIBUTIONS

Armaghan handled most of the set up for the game world and the simulation while Alex handled the neural networks and genetic algorithm. Once the simulations were running, both ran multiple cases on their respective laptops and analyzed the data together. This report was also written jointly.

### REFERENCES

[1] Floreano, Dario, and Claudio Mattiussi. Bio-inspired artificial intelligence: theories, methods, and technologies. MIT press, 2008.

[2] Hillis, W. Daniel. "Co-evolving parasites improve simulated evolution as an optimization procedure." Physica D: Nonlinear Phenomena 42.1 (1992): 228-234.

[3] Nolfi, Stefano, and Dario Floreano. "Coevolving predator and prey robots: do arms races arise in artificial evolution?." Artificial life 4.4 (1997): 311-335.

[4] Ray, Thomas S. "An approach to the synthesis of life." (1992): 371-408.

[5] Rosin, Christopher D., and Richard K. Belew. "New methods for competitive coevolution." Evolutionary Computation 5.1 (1997): 1-29.

[6] Waibel, Markus, Laurent Keller, and Dario Floreano. "Genetic team composition and level of selection in the evolution of cooperation." Evolutionary Computation, IEEE Transactions on 13.3 (2009): 648-660.