Coevolution of Multiagent Systems using NEAT

Matt Baer

Abstract

This experiment attempts to use NeuroEvolution of Augmenting Topologies (NEAT) create a Multiagent System that coevolves cooperative learning agents with a learning task in the form of three predators hunting the prey. Since the prey is hardcoded in a way that eludes the nearest predator, methods to capture the prey elude supervised learning techniques, i.e. cooperation is required to succeed. This is in some part an extension of Yong and Miikkulainen's experiment *Coevolution of Role-Based Cooperation in Multiagent Systems*, except the Multiagent Systems use NEAT for the adaptive method rather than Symbiotic, Adaptive NeuroEvolution Enforced SubPopulations (SANE ESP). The NEAT multiagent system consisted of coevolving three heterogeneous networks with a common fitness function that rewards being close to the prey upon capture or the time limit imposed. The fitness scores among the population promisingly grew, but stagnation proved to be a large obstacle during later generations. Emergent behaviors of teamwork and clear evidence of heterogeneity leading to different roles were all found in the behavior and topology. This experiment made use of past research to create a rudimentary NEAT system that wouldn't have been able to accomplish the same goal with a homogeneous or single agent system.

1 Introduction

Cooperative multiagent problem solving involves agents working together with a shared goal. By working in a parallel fashion, multiagent systems can solve problems with more efficiency, robustness, and flexibility than single-agent systems could. Cooperation can be implemented with the consideration of factors such as coordinating the agents with a single controller or having separate diverse controllers, i.e. homogeneity vs. heterogeneity, and whether communication is truly necessary.

Yong and Miikkulainen demonstrates in their paper *Coevolution of Role-Based Cooperation in Multiagent Systems*[4] the implementation of a multiagent system that solves a predator-prey capture task with 3 predators chasing 1 prey using *role-based cooperation*, which is defined as non-communicating cooperation based on stigmergic coordination. The cooperative learning task is solved by evolving neural networks in a number of ways to determine the most effective setup. The optimal set-up was found to be coevolving heterogeneous, autonomous networks without communication.

It is the goal of this paper to recreate Yong and Miikkulainen's multiagent system using a different adaptive method in NEAT[1] , rather than SANE ESP. The learning task is the same: 3 predators evolving to chase and eventually capture the prey hardcoded to run away from the

[1]We use the python implementation of Kenneth O. Stanley which can be found on his website.[1] www.cs.ucf.edu/∼kstanley/neat.html

predators. A simulator was created to copy the same task domain as Yong and Miikkulainen's discrete grid world. However, the simulator utilizes a toroidal world and has a continuous coordinate system. This is arguably better because steps become more natural since the predators can head in any direction. The main variable changed for analysis is simply the type of neural network used to coevolve the predators. By changing the type of neural network but possibly recreating the same result of role-based coordination, we can confirm the level of robustness with applying concepts of heterogeneity and stigmergy.

SANE ESP, or Symbiotic, Adaptive NeuroEvolution Enforced SubPopulations, is an extension of SANE which evolves a population of neurons by having each chromosome represent the connection of one neuron instead of its own entire network. SANE ESP evolves a separate population for each hidden layer neuron in the network, and the neuron populations can be solved simultaneously. ESP acts as a cooperative coevolution method, with neurons hopefully converging to the roles that bring the highest fitness.

NEAT, or NeuroEvolution of Augmenting Topologies, is an adaptive method that evolves networks starting with a topology of only inputs and outputs. Over generations of evolution, new hidden layers of nodes and connections are added. By complexifying the space of potential solutions and elaborating on the simple strategies to form more complex ones, the network is more likely to build on existing solutions rather than alter them. Since the experiment makes use of a complexifying method in NEAT, no incremental task is needed for the learning agents to converge to a solution. For example, the speed of the prey does not need to be incremented to allow the predators to get accustomed first. However, SANE ESP should prove to be more efficient since each chromosome only has to represent the connections of one neuron instead of that of a whole neural network like this experiment does. However, we can hope to, even with a bulkier chromosome population, converge to a high-level solution.

Stanley and Miikkulainen's paper *Competitive Coevolution through Evolutionary Complexification*[3] shows that complexification, defined as the incremental elaboration of solutions through adding new structure, through NEAT develops more complex strategies that can reach higher level of optimal solutions.

Communication in the context of the experiment consists of giving each predator the position offsets of the other predators as input for each of their neural networks. The question arises when it comes to determining the inputs we give our learning agents. In addition to giving the agents the prey offsets, do we have them broadcast their locations to each other?

Yong and Miikkulainen concluded in their paper by direct comparison that communication is actually detrimental to performance when stigmergic coordination is possible. Stigmergy refers to the indirect coordination between agents done by observing the direct change in the environment caused by the actions of the agents. When stigmergy can be achieved, cooperation becomes much less costly due to the cost of requiring additional resources to make meaningful connections out of these largely noisy inputs.[4] Since this experiment also makes room for stigmergic coordination, the NEAT networks will be coevolved without communication and will instead rely on stigmergy.

Heterogeneity when it comes to developing learning agents involves separately evolving the cooperative agents, rather than having homogeneity which means using the same singular network for each agent. Heterogeneity has been found to be more effective since it opens up the possibility of having different strategies among the agents in the form of specializing roles.

In his paper *Behavioral Diversity in Learning Robot Teams*, Balch concludes that heterogeneity works better in situations in which there has to be a division of labor, i.e. the task cannot be solved by one agent alone.[2] The skills required to accomplish the goal, e.g. capturing the prey in our predator-prey task, involve cornering the prey in a way that requires multiple agents to move in for the kill at once.

More relevant is Yong and Miikkulainen's test of robustness regarding having heterogeneous agents. Homogeneity can only be accomplished through communication, since their behavior will be too similar especially since their initial positions are close together. With homogeneous teams, there was a success rate of only 3% since establishing roles and breaking symmetry were both too difficult for the homogeneous teams. The homogeneous teams just could not differ well enough in a way to break off and go their own way, while heterogeneous agents are found to be naturally effective at breaking symmetry.[4]

The main hypothesis of this paper is that it is possible to demonstrate role-based cooperation when coevolving multiagent systems with the same learning task but with a different adaptive method in NEAT, effectively emulating Yong and Miikkulainen's most effective system of heterogeneous, non-communicating coevolution. The hypothesis that the costs of communication outweigh the benefits will also be tested.

Two important differences in the implementation involve (1)the switch from a discrete grid to a continuous world where each agent can move in any direction, rather than North, South, East, West, for each step and (2)taking advantage of NEAT's complexification by using a non-incremental task, which requires less supervision from the user which is a step forward in neuroevolution. Using a different adaptive method to achieve the level of cooperation in Yong and Miikulainen's experiment would do well to confirm the effectiveness of applying the underlying concepts of heterogeneity, stigmergy, as well as even simply the idea of using multiagent systems.

The evaluation of this hypothesis will be done by looking for cases of heterogeneity and cooperation in the emergent behaviors of the Multiagent system. There are too many differences in the implementation to able to directly compare the results to those of Yong and Miikkulainen's experiment.

2 Experimental Setup

2.1 Prey-Capture Task

The learning task will follow the lead of Yong and Miikulainen's experiment, with three predators in the bottom left corner chasing after the prey. Yong and Miikkulainen used a toroidal discrete 100 x 100 grid in which agents are allowed to overlap and at each step each move either North, South, East, West, or not move at all. This experiment instead uses a 300 x 300 toroidal world created using the Zelle graphics library with a continuous coordinate system in which agents of radius 10 can move in any direction.

Commands are given to the agents in the form of an ordered pair consisting of numbers representing translation and rotation. These commands of course will act as the outputs of the NEAT networks used to control each predator. The agent representing the prey is captured in when it becomes in contact with one of the predator agents, i.e. the predator overlaps with the prey. The prey is hardcoded to run away from the closest predator.

To speed up evolution, Yong and Miikkulainen incremented the difficulty of capturing the prey

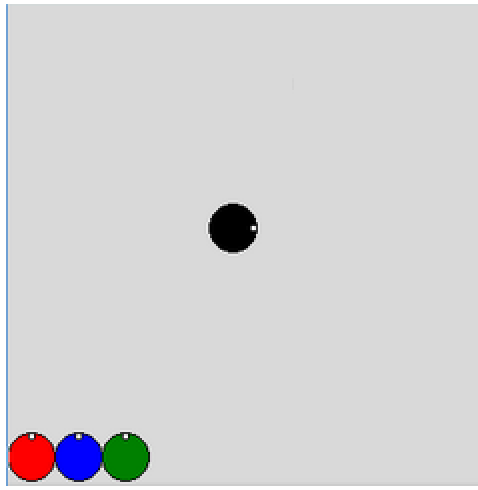

Figure 1: In this experiment, the predators are initially in the bottom left corner with the prey in the center of the board. The predators are rewarded for being close to the prey and for capture.

going from not moving to always moving away from the closest predator in terms of probabilities. This experiment varies in that the prey will always run away from the nearest point since the very beginning, but at 70% of the speed. This is a good deal harder to accomplish, but the hope is that the complexification involved in NEAT will allow the predators to do better than with SANE ESP.

A huge difference is the initial position of the prey is fixed at the center of the board as in Figure 1., unlike in Yong and Miikkulainen's where the prey was placed randomly in one of nine boxes of area in their 100 x 100 discrete grid. Those nine areas also were used to serve as benchmark tests for how robust their predators were. Because of the level of difficulty and time involved in evolving such a robust system, the NEAT multiagent system will instead be confined to one simple goal. It must able to track down the prey which always starts at the centings that I found over time of using this feature. Both are mentioned in the linked pdf, but I mier of the board, which, because of the toroidal world, is the maximum initial distance.

The toroidal world adds to the layer of difficulty by ensuring that simply running after the prey won't work. The prey has to be cornered and approached from different directions at once. Having a toroidal world requires the offsets, distance between agents, and angle between agents to account for the idea that the shortest distance path may lead off-grid. The prey offsets used as NEAT inputs account for this as well.

There will also be a time limit set at 75 steps, since it was found to be the time it takes for a predator to go from corner to corner. This is so as to eliminate accidental captures since bonuses may go to solutions that deserve a lower score. With a time limit, the predators are forced to evolve a strategy that doesn't waste time. This is also following Yong and Miikkulainen's lead.

2.2 Coevolution of NEAT Populations

Each NEAT network has 2 inputs and 2 outputs, with the parameters as shown in Table 2.2. The two inputs are the prey's x and y offsets (accounting for toroidal distance), and the two outputs are the translation and rotation which are used to update the predator's position for the next step. The networks start out with no hidden layer and then complexify over generations.

Parameter	**Setting**
Generations	50
Population size	100
Input nodes	2
Hidden nodes	0
Output nodes	2
Prob. to add link	0.1
Prob. to add node	0.05

Table 1: NEAT parameter settings used in the experiments. See appendix for full config file.

There are 3 NEAT populations each of size 100. They are coevolved in a way such that 3 random chromosomes are selected from the populations, and each chromosome will attempted to be used 10 times. When the 3 chromosomes are used for the brains of the predators, the fitness will capture how far along the predators were relative to the prey and whether there was a capture. The fitness score will then be added to the list of each of the three chromosome's performances, and the average fitness score over trials each generation will be computed.

The fitness function will follow the same structure as Yong and Miikkulainen's fitness function which minimized the average final distance from the predators to the prey while rewarding captures. Since one of the predators are going to have to go the long way anyway to corner the prey, time doesn't vary and therefore it's excluded from the fitness function. The max fitness threshold is 1 so the fitness function is as follows:

$$f = \begin{cases} 1 - \frac{d_f}{d_{max}} & \text{, if prey is caught} \\ 0.5 * (1 - \frac{d_f}{d_{max}}) & \text{, if prey is not caught} \end{cases}$$

where f is the fitness metric, d_{max} is the maximum distance possible on the grid, and d_f is the average final distance between the predators and prey at either capture or when the time limit has run out.

There were 10 runs of 50 generations each. The average and best fitness scores are recorded for each population over each generation.

3 Results

With the ten runs, the average fitness score was recorded every 10 generations and is shown in Table 3. The progression of the average and best fitness of two different runs are visualized in Figure 2. The best fitness was noticeably noisy, but still improved on its best captures. The average fitness was steadily improved. These results show a steady increase in average fitness score, and steady improvement was found to lead to more complex solutions that had the prey cornered from its first move.

Stagnation was found to increase past the 50th generation, but this steady rise in fitness remains

Trials	Gen 10	Gen 20	Gen 30	Gen 40	Gen 50
1	0.19	0.20	0.22	0.22	0.26
2	0.19	0.20	0.21	0.22	0.26
3	0.20	0.27	0.31	0.34	0.35
4	0.19	0.25	0.26	0.28	0.31
5	0.16	0.18	0.18	0.19	0.20
6	0.19	0.20	0.20	0.21	0.22
7	0.17	0.21	0.22	0.24	0.27
8	0.18	0.20	0.20	0.21	0.24
9	0.19	0.20	0.22	0.22	0.28
10	0.19	0.20	0.22	0.22	0.27
Avg	0.19	0.20	0.22	0.22	0.27

Table 2: Average fitness scores for each run over generations. The average fitness score for each generation is calculated as well.

promising. Finding the right difficulty however requires finesse, because if the prey's too fast, the networks struggle to evolve, too short and the task becomes closer to meaningless in terms of requiring homogeneity and stigmergy. Stagnation is the barrier to success for the learning agents. In fact it was Yong and Miikkulainen's measure of failure for their SANE ESP Multiagent System. Getting to generation #50 without stagnating completely is a good start towards further work.

The phenotype of the predators have also developed in an interesting fashion, as you can see in the evolution of the topologies of each population over generations shown in Figure 3. The topologies clearly develop over time, each different from one another. This demonstrates the heterogeneity of the learning agents as well as the roles they take on. One predator usually takes the long away across while the other two go the other direction. The prey was found to be slippery since it's in the middle, but the best solutions managed.

4 Discussion

In the context of cooperative learning, this experiment demonstrated the use of a NEAT multiagent system to complexify towards a solution that consists of heterogeneous, cooperative agents working towards a common goal by coevolving together. In that sense, teamwork was able to be found in the solutions. The ultimate goal of demonstrating stigmergic coordination remains to be completely realized, due to the problem of stagnation in later generations. However, the modest goal of recreating a fairly successful multiagent system using NEAT and a graphics simulator was reached. The topology developed was fairly complex, and the difference in amount of connections and nodes between predators in the same generation indicated an accepted role taken with each predator.

Future work on this experiment could involve a lot of variations. Keep in mind that the difficulty of the task remains a limiting factor. However, the more difficult the task, the more robust the solutions. There's the option of making the prey's position random, as well as evaluating the network using tests for each random location just like Yong and Miikkulainen did. The prey can easily be made harder to capture by making it the same speed as the predator. It is also possible to modify the prey to be able to escape two predators at once, thus requiring even more teamwork and less room for homogeneity. The prey could have an invisible force field around it such that it runs away from all predators in the force field at the best possible angle, as opposed to alternating angles in a way that greatly slows it down. The communication, coevolution, and heterogeneity could be investigated for their robustness using this experiment. Finding out whether each of these concepts still applies to multiagent systems with NEAT would be an interesting find for even more future work.

NEAT Parameters

```
[phenotype]
input_nodes         = 2
output_nodes        = 2
max_weight          = 30
min_weight          = -30
feedforward         = 1
nn_activation       = tanh
hidden_nodes        = 0
weight_stdev        = 0.9

[genetic]
pop_size              = 100
max_fitness_threshold = 1
prob_addconn          = 0.1
prob_addnode          = 0.05
prob_mutatebias       = 0.2
bias_mutation_power   = 0.5
prob_mutate_weight    = 0.9
weight_mutation_power = 1.5
prob_togglelink       = 0.01
elitism               = 1

[genotype compatibility]
compatibility_threshold = 3.0
compatibility_change    = 0.0
excess_coeficient       = 1.0
disjoint_coeficient     = 1.0
weight_coeficient       = 0.4

[species]
species_size        = 10
survival_threshold  = 0.2
old_threshold       = 30
youth_threshold     = 10
old_penalty         = 0.2
youth_boost         = 1.2
max_stagnation      = 15
```

References

[1] The NEAT users page. www.cs.ucf.edu/∼kstanley/neat.html. Accessed: 2014-05-01.

[2] Tucker Balch. *Behavioral Diversity in Learning Robot Teams*. PhD thesis, Georgia Institute of Technology, 1998.

[3] Kenneth O. Stanley and Risto Miikkulainen. Competitive coevolution through evolutionary complexification. *Journal of Artificial Intelligence Research*, 21(63-100), 2004.

[4] Chen Han Yong and Risto Miikkulainen. Coevolution of role-based cooperation in multiagent systems. *IEEE Transactions on Autonomous Mental Development*, 1(3), October 2009.

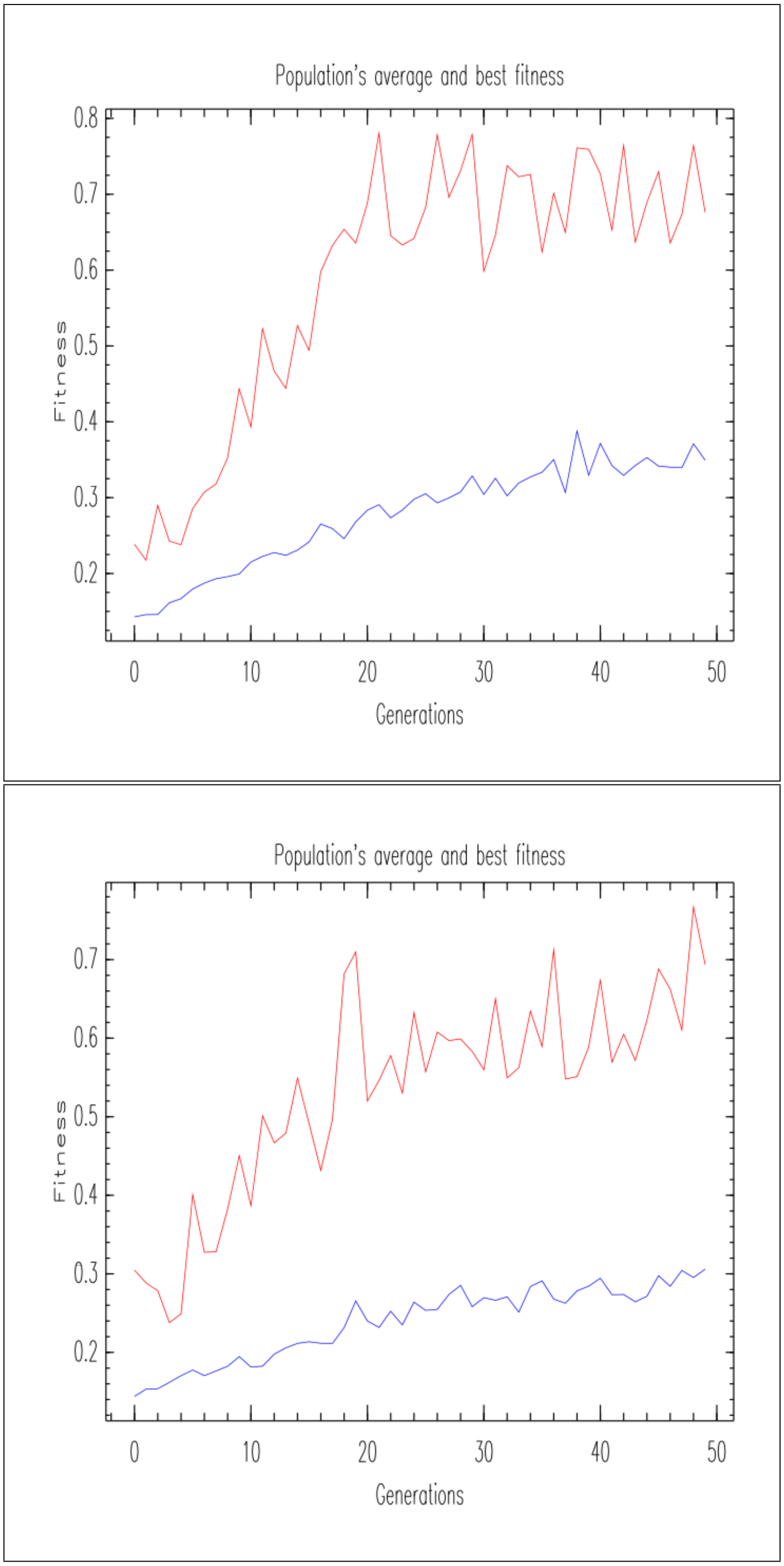

Figure 2: The average and best fitness graphs from Run #3 and Run #5, where red represents the best fitness and blue the average fitness over generations.

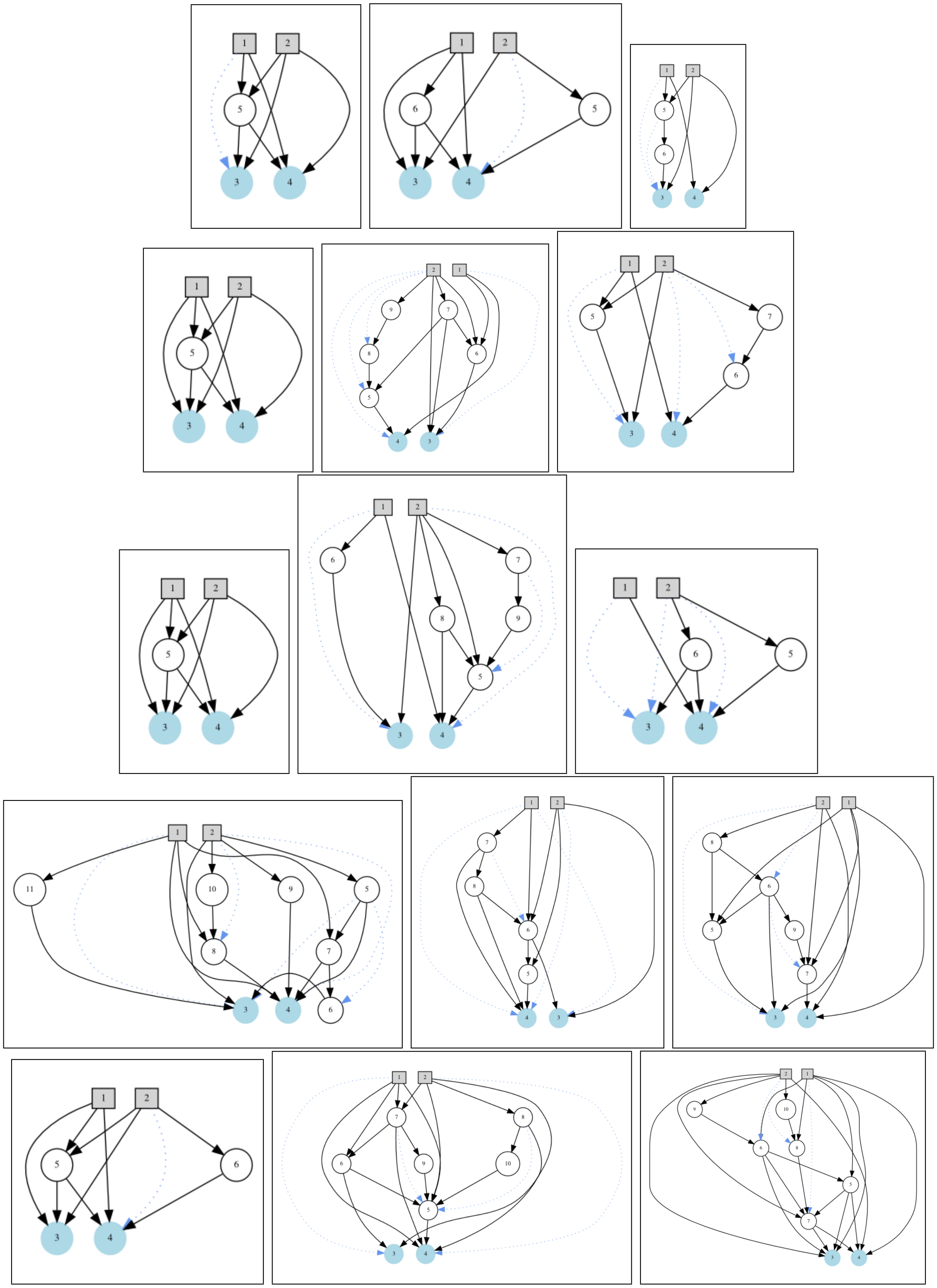

Figure 3: The topologies of Predator A, Predator B, and Predator C as developed in generations. Row 1 represents the topologies in Generation 10; 2 represents those in 20 and so on until Gen 50.