Competitive Coevolution of Robots to Play Ultimate Frisbee

Abigail Henderson & Bernard Koch

Abstract

This paper describes the coevolution of neural controllers for robots in a simulated Ultimate Frisbee play. The goal of these experiments was to reproduce established strategies for both offensive and defensive players in Ultimate, as well as to possibly discover novel strategies that could be implemented by humans. In each experiment, an offensive robot and a defensive robot compete for optimal positioning to receive/intercept a frisbee pass. Two closely-related fitness functions were designed to reward each robot for positioning with respect to its opponent and a stationary robot representing the player with the disc. Defenders evolved against a pre-programmed cutter displayed behavior resembling good man-on-man defense, and in coevolution, cutters developed the ability to make deep cuts when the defender was focused on guarding the in-cuts.

1 Evolving Intelligent Players

In recent years, artificial intelligence and developmental robotics have taken on many challenges. Some, such as problem-solving and simple categorization, have been met with outstanding results, while others, such as natural language processing and computer vision, have proven quite complex. In this paper, we present a new challenge. Can robots learn to play a sport which is young enough to still be evolving itself? And if they can, might they also be able to develop effective strategies that humans have not yet invented? The answer to the first question will be of great import to the world of computer science, and the answer to the second will be of great import to the world of Ultimate Frisbee. If our robots succeed, it will help demonstrate the ability of machines to participate in and understand a small part of human activity, and it may help Ultimate players to think critically about established strategies.

1.1 Ultimate Frisbee: The Game

Ultimate Frisbee is a team sport played on a rectangular field with end zones at each end. The goal of the offensive team is to advance a circular disc from one end of the field to the other, and a point is scored when an offensive player catches the disc within the opposing end zone. The disc is advanced by being thrown from one player to a teammate. A player may not run while in possession of the disc and has 10 seconds to make a pass or a change of possession occurs. The goal of the defensive team is to force a turnover of possession, either by intercepting passes or by preventing the offense from completing the pass such that the disc touches the ground. Seven players from each team are on the field during play, and the defense typically matches up one-to-one on each offensive player or distributes to guard regions of the field. The offensive team is divided between two roles: handling the frisbee (2-3 players), and cutting to receive passes (4-5 players). Handlers are typically more-experienced players who have developed throwing skills and good field vision.

They are responsible for passing the frisbee to the cutters, who spend their time cutting to avoid defenders and receive passes to advance the disc. After the cutter has received the pass, they will typically make a short pass to a handler to begin a new play.

We expect the robots in our experiments to evolve reasonable strategies for cutting and defending, similar to those that human players have created. Some cutting strategies used by humans include fakes (starting to move one direction, then changing directions) and position awareness (if a player is far away from the disc, they should cut in, and vice versa). Some defensive strategies include zone defense and man defense (stay as close as possible to the offensive player). Defenders must also be aware of the concept of a light side and a dark side. The defender guarding the handler blocks passes by the handler from one side, making it difficult to throw to that side (the dark side). Because the cutter is more likely to receive a pass on the handler's open side (the light side), the cutter's defender can increase their chances of blocking that pass by always staying to the light side of the cutter (called light-side defense).

Ultimate is a relatively new sport in which players are still developing strategies for defending, cutting, and organizing offensive plays. In this paper, we focused specifically on the interaction between one offensive player and one defensive player. We sought to evolve neural network controllers for a robotic cutter and a robotic defender that would reproduce some of the strategies described above. We elected to use a coevolutionary scenario because it most closely resembles how human players learn to play Ultimate, by being aware of your opponent and trying new strategies until something works, while your opponent is doing the same. It was our hope that novel, effective behaviors for both robots might emerge through coevolution as well.

1.2 Related Work

Competitive coevolution confers three major theoretical advantages over single population evolution. Most importantly, coevolution presents an appropriate but increasingly difficult task to the evolving population. This allows a robot brain to learn simple strategies against simple opponents initially and expand to more sophisticated strategies as the opponent evolves counter-strategies. This 'arms race' should drive better and better solutions.

Secondly, the presence of an evolving opponent creates a dynamic environment for the robot population. Constant changes in environment will affect changes in the fitness landscape, potentially preventing evolution from stagnating within local fitness minima.

Finally, the opportunity for a robot brain to compete against multiple individuals in the opposing population allows for the simultaneous exploration of several possible solutions. Without coevolution, a brain that scored a low fitness score against one opponent would be discarded. In coevolution, this brain might have been uncompetitive with two out of three opponents, but could do really well against the third. In this case, the promising brain would still be rewarded fitness and the strategy could be maintained for further exploration and development in future generations.

In one of the first competitive coevolution studies, Sims simulated two robots competing for control of a square in the middle of a square map. This study demonstrated that neural controllers and body plans could be coevolved to develop several novel effective strategies. [6]

Unlike Sims' duel configuration, Floreano and Nolfi used a predator-prey scenario and experimented with the effects of sensors on coevolution. They first equipped the predators with long range cameras and the prey with only short-range infrared sensors. In these experiments, the predators' strategies evolved while the prey did not progress beyond trying a variety of radically different strategies. They then added cameras to the prey as well, allowing them to evolve and refine strategies, thus restoring the arms race.[5]

Bason et al extend Floreano and Nolfi's predator-prey experiments by exploring the added dimensions of body morphology and environment evolution.[2] In his experiments, Bason gradually allowed his robots to evolve the positioning of their sensors and select for environment size, concluding that arbitrary choices by the designer on both morphology and environment are not necessarily optimal, and may not reflect natural evolution where species' select their environments as much as they adapt to them.

Stanley and Miikkulainen devised a zero-sum robot duel in which two robots with limited energy move around a map until they collide.[7] While both robots have sufficient energy for the duration of the match, movement costs energy. If the robots collide, the robot with more energy wins. These robots were evolved under the NEAT framework for coevolution which incorporated measures to overcome many of the problems formerly described in coevolution theory. In particular, NEAT attempts to perpetuate the arms race through constant complexification of the neural network controller. The authors found that complexification allowed the controllers to develop new, more sophisticated strategies on top of the old ones. We used NEAT in this study, and the advantages of the framework will be described briefly in the next section.

In their attempts to coevolve robots that play Capture the Flag, Nelson and White developed a strategy to overcome the problem of initial populations that were sub-minimally fit.[3] In their experiments, they encountered populations that were unable to complete the task against even the simplest competitors. To avoid this, initial behaviors were hard-coded and but the neural controller was able to overwrite these behaviors as evolution progressed. Controllers evolved under this approach in simulation were successfully transferred to teams of two embodied robots.[4]

2 Our Study: Coevolving Ultimate Frisbee Players

This study is unique from those previously described for several reasons. First, with the exception of Nelson and White, it is the only study to emulate a human game or behavior. Second, the frisbee play scenario is significantly more nuanced than either the duel configurations in Sims or Stanley and Miikkulainen, or the predator-prey scenario in Floreano and Nolfi or Bason et al. Like the duels in Stanley and Miikkulainen, the robots have identical morphologies, but the fitness functions are not identical. Like the predator-prey experiments, the defender and the cutter have different roles, but they also have identical morphologies and similar fitness functions.

Lastly, fitness in our experiments is monitored continuously rather than at the end of the trial. This is because a pass could be completed at any moment, and the quality of cutting or defense is assessed by the trajectory of movement and not the final position. We think this last discrepancy in particular makes this study an interesting contribution to the competitive coevolution literature. Furthermore, this paper also compares competitive coevolution to single population evolution experiments designed to compensate for sub-minimal fitness.

2.1 Experimental Setup

The frisbee play environment was a simulated open square, with the defender and the cutter placed in the center of the map. The only other obstacles on the map are two stationary robots along the center of the bottom edge, representing the handler and, to its left, a defender guarding the handler

(Fig. 1A). The evolving defender is situated to the right of the cutter. The cutter is initially oriented vertically facing the handler, while the defender is oriented at an angle intermediate between facing the cutter and the handler, reflective of a human defender's orientation such that both the cutter and the handler are in peripheral vision. The defender is positioned to the light side of the cutter, in this case the right, where the handler is more likely to pass since he is being guarded on his left (the dark side). This setup is typical of the start of an Ultimate play.

The environment was scaled to 15 by 15 meters with respect to the size of the robots, representing a passing range of a typical Ultimate play. The environment initially had walls, but these were later found to be unnecessary and removed. Experiments were conducted within Pyrobot, a Python programming environment containing basic artificial intelligence data structure objects, as well as APIs and simulators for the control of several commercial robots.[1]

While based on the simulated body of the Pioneer robot, our robots only have abstracted sensors which are intended to capture the situational awareness of an Ultimate defender or cutter. Because both teams are trying to gain possession of the disc, it seems reasonable that each robot be unconditionally aware of the position of the handler, so both robots are given their angle and distance to the handler at each time step. Similarly, defenders and cutters on an Ultimate field should always be aware of each other; the primary goal of the defender is to prevent the cutter from getting the frisbee, while the cutter can increase its chances of pass completion by escaping the defender. To implement this awareness, each robot is also given the angle and distance to its opponents. In our earlier experiments, the robots had a fifth binary sensor which fired when the robot was within one body-length of a wall (Table 1, Series A), but this was removed with the removal of walls after the A-series experiments.

Robots were evolved using the NEAT framework described in Stanley and Miikkulainen. [7] NEAT has multiple benefits, including the design of a system for neural network crosssover that is minimally destructive of neural network topology and a speciation system that protects structurally innovative brains from immediate out-competition by more evolved strategies. However, its primary advantage is the gradual complexification of topolgies over time. By structurally complexifying artificial neural networks continuously rather than just elaborating weights, NEAT encourages the evolution of new strategies on top of existing strategies rather than instead of them. This discouragement of catastrophic forgetting should decrease the stagnation or cycling of the arms race.

NEAT initiated evolution with the four or five sensor input nodes feeding directly into two output nodes for motor translation and rotation using a sigmoidal activation function. Our competition setup differed from the one used by Stanley and Miikkulainen in that each robot population competes against the single fittest individual in the previous generation of its opponent as well as two others. We could not use a full hall of champions and/or dominance hierarchy due to computational constraints. Instead, each population after generation 4 competes against three other randomly selected previous opponent champions in addition to the most recent one. For similar reasons, we limited each matchup to one trial. We ran each trial for 10 steps, analogous to the 10 seconds the offensive team has to complete a pass before the disc changes possession. Alternatively, the trial ends if either robot's fitness drops below zero. This can only occur from accumulating stall penalties from hitting the opponent robot or the stationary robots.

Besides the coevolution experiments in series A and D (Table 1), we also explored single-population evolution. Experiment series B evolved a defender against a cutter making different pre-programmed cuts. In these experiments, the trial ends when the pre-programmed cut ends

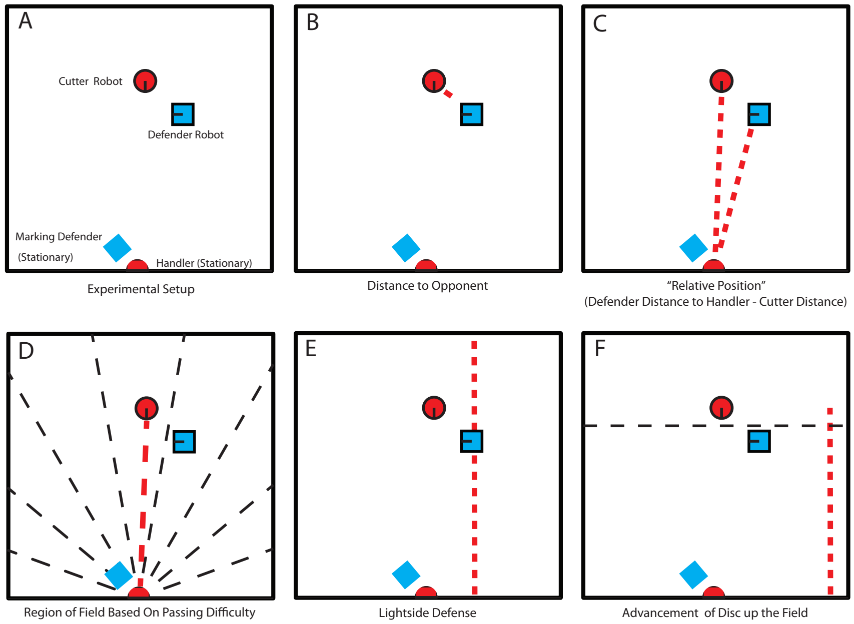

Figure 1: Frisbee play environment and components of the fitness function.

after 6 or 8 steps or if the defender stalls out and has zero fitness. Experiments in the C series then evolved a defender against the fittest cutter evolved in B4. The end conditions for these trials were identical to those of the coevolution experiments described above.

2.2 Fitness Function

For both Ultimate offense and defense, the objective of a single play is to achieve positioning that maximizes potential to complete or intercept a pass, respectively. In this section, we describe the fitness criteria chosen to quantify good positioning in an ultimate play. For most criteria, the scale used to calculate the cutter's fitness is simply the reverse of the scale used to calculate the defender's fitness. In Figure 2, we have outlaid the proportion of fitness contributed by each of the criteria below for both the cutter and defender. For the defender, these contributions are approximated by averaging data from the last generation of D1 (Table 1). This is necessary because the lightside defense criterion is a flat reward so its proportion of fitness is relative. This highlights the fact that it is hard get a maximum fitness for either robot, because fitness is so dependent on the opponent's behavior.

Distance to Opponent (Fig 1B)

In man-on-man defense, a defender is useless if he drifts too far away from his man. Therefore the largest component of the defender's fitness function, approximately 62%, is devoted to achieving the minimum distance to the cutter (Fig 2). For the cutter, maximizing distance to the defender accounts for only 32.6% of overall fitness, for while getting away from the defender is important, having good positioning is more important.

Contributions of Fitness Criteria to Overall Fitness

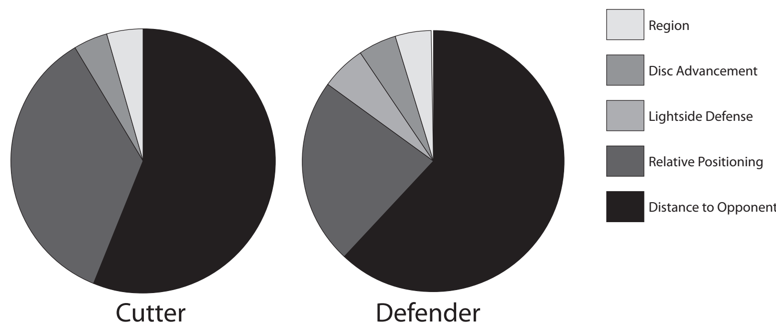

Figure 2: Weights of the components of cutter and defender fitness functions.

Relative Positioning (Fig 1C)

Good ultimate defense goes beyond just being close to the cutter; positioning oneself with respect to the handler and the cutter is critical. If the cutter is cutting in towards the handler, it is advantageous for the defender to be between the handler and the cutter. However, if the cutter is cutting deep, a better move for the defender would be to get in front of the cutter to have a better opportunity of catching a long pass.

In our fitness function, we quantify positioning for the defender as the difference between the cutter's distance to the handler and the defender's distance to the handler. In the lower 8.5 meters of the map, the defender is rewarded for being between the handler and cutter, but beyond 8.5 meters, the cutter is considered to be going deep and fitness is gained by being farther out than the cutter. The reverse, of course, is true for the defender's fitness. This encourages the robots to learn the changes in positioning that occur for short passes versus long passes. Relative positioning accounts for approximately 23% of the defender's total fitness and 54% of the cutter's.

Region and Light-side Defense (Fig 1D,E)

Because the handler is being guarded, not all possible passes from the handler are equally difficult or likely to be completed. To reflect this, the region factor in our fitness function corresponds to the angle of the player robot to the handler. In this model, standing directly to the left of the handler (on the dark side) gives the least reward because the handler must pass the disc directly through its guard. Conversely, standing directly to the right of the handler receives the most fitness because the handler is wide open for a high-completion pass.[1] Region accounts for 6.5% of the cutter's total fitness and 4.7% of the defender's fitness.

[1] It should be noted that it can often be advantageous to cut for a throw from the dark side in a game of Ultimate. For the purposes of our simlation, however, we have simplified the task by assuming that this handler has difficulty completing dark-side throws.

In order to encourage the defender to stay on the lightside, the defender receives a flat bonus of 7 added to its fitness when it is to the right of the cutter. In trial D1, this ends up accounting for about 5.4% of the defender's total fitness. This is the most contrived component of our fitness function and while it allowed sub-minimally fit defenders to improve their fitness in other areas, we decided not to consider it for the cutter's fitness.

Disc Advancement (Fig 1F)

Advancing the disc up the field is the general goal of the offense in Ultimate. While shorter passes are less risky, longer passes cover more distance and should yield a higher reward. A small portion of both the cutter and the defender's fitness is calculated relative to the cutter's advancement up the field (defender: 4.6%; cutter: 6.5%).

Stalling

In Ultimate, physical contact to prevent another player from moving, called setting a pick, is illegal. When a Pyrobot robot runs into another robot (or an obstacle), it stalls and can no longer move. Besides the fact that this is illegal in a game, it can prevent an accurate assessment of performance through fitness. Therefore, whenever an evolving robot stalls, it receives a fitness of negative ten.

3 Experimental Results

The conditions and results of our experiments are very briefly outlined in Table 1. In the following sections, discussion and analysis will focus on experiments B-D, and the results of A1-8 will be briefly summarized. The results will be analyzed using primarily behavioral performance, but also fitness measurements and neural network topology. Much of the analysis is qualitative simply because of the abstract nature of what constitutes "good positioning."

3.1 Experiments A1-A8: Summary of Results

This first set of experiments involved manipulating many different parameters in search of productive results. Our first experiment, A1, resulted in both players going in circles backwards for the entirety of every trial. The defender would sometimes run into the cutter and both robots would stall. In order to produce more variation and complexity in behavior, we tried changing the NEAT parameters in A2 and A3 to increase the probability of mutation and species size or to start with one hidden node. None of these changes produced a noticeable difference in the behavior or fitness of the robots. We briefly tried only allowing the robots to move forward with A4, but this only resulted in both players going in circles forwards. A5, A6, and A7 focus on refining the fitness functions and avoiding stalling, with limited success after evolution. A8 evolves both players over 250 generations instead of 50, but the results are no different.

While the poor outcomes of these experiments were likely in part due to bugs in the program which were later fixed, these early experiments were integral in fully developing our fitness functions and in framing our later experiments.

Exp	Conditions	Motivation	Behavioral Results
A1	50 generations. Defender's fitness function is the negative inverse of the cutter's (+60 to make it positive). No light-side defense reward/penalty.	Try coevolution of defender and cutter.	Both players circle backwards. Defender sometimes runs into cutter, resulting in a stall.
A2	Built from A1. Increase probability of mutation and species size.	Encourage more varied behaviors.	No noticeable difference.
A3	Built from A1. Start with one hidden node.	Encourage more complex behaviors.	No noticeable difference.
A4	Built from A3. Disable backward movement.	Force forward movement to more closely mimic real players.	Both players circle forwards.
A5	Built from A1. Increase stall penalty. Modify defender's fitness function: include explicit reward/punishment of light-side defense, increase positioning weight, decrease spacing weight.	Avoid stalling; encourage light-side defense.	Defender slightly improved. Both still circling.
A6	Built from A5. Modify cutter fitness weights: increase position and spacing.	Encourage cutter to get open.	No noticeable difference. Still stalls.
A7	Built from A6. Increase stall penalty.	Avoid stalling.	More varied behavior to avoid stalls. Not particularly good behavior, however.
A8	Built from A7. Run for 250 generations.	Allow more time for evolution.	No noticeable difference.
FIX	Between the A and B series experiments, we fixed several bugs in sensors, inputs, and outputs. Wall sensor and walls were also removed.		
B1	Built from A7. Only evolve defender, with pre-programmed cutter making an in-cut.	Let defender evolve without competing with an evolving cutter.	Decent behavior, but not great.
B2	Built from B1. Decrease stall penalty.	Perhaps strong stall penalty "scares" defender, causing it to stay far away from cutter.	Defender runs into cutter immediately; stalls through rest of trial.
B3	Built from B2. Add stall sensor input to neural network.	Help defender learn not to stall.	No noticeable difference.
B4	Built from B1. Increase population size.	Greater genetic diversity (due to increased population size) should encourage more different types of behavior.	Displays great strategy, light-side defense, and decision-making.
B5	Built from B4. Add 2 more cuts to pre-programmed cutter (an out-cut and a dark-side in-cut). Increase population size.	Evolve a good general defender.	Unsatisfactory behavior; defender moves around the center region regardless of the cutter's positioning.
C1	Built from B4. Only evolve cutter, with best defender brain from B4 controlling defender.	Let cutter evolve without competing with an evolving defender.	Cutter responds to weaknesses in defender and cuts deep. Defender does poorly.
D1	Built from C1, B1.	Textbook coevolution once more.	Great improvement in cutter behavior, including fakes and deep cuts. Slight improvement in defender.

Table 1: Overview of the experiments.

3.2 Fixing Bugs and Branching Out

After the A series of experiments, we spent a long time discovering and fixing bugs that were hindering evolution. The angle sensor, which senses the relative angle from one robot to another, was not calculating relative angles correctly, and the equation calculating the region of the field was also incorrect. We also discovered that Pyrobot's time steps were not consistent from trial to trial (ranging from .02 seconds to close to 3 seconds), making results difficult to reproduce or compare. We solved this problem by waiting the remainder of 3 seconds in between steps. Lastly, our neural network inputs and outputs were not normalized for the tanh evaluation function we were using with NEAT. We changed to an exponential sigmoidal function which uses values between 0 and 1, and normalized all of our inputs and outputs accordingly. Lastly, the wall sensor and its corresponding input node were not activated very often, and when they were, the result was a stall which incapacitated the robot for the rest of the trial. The sensor, the input node, and the walls were all removed for future experiments to encourage a more natural flow of game.

With all these fixes in place, we decided to branch out and experiment with evolving each player alone as well as continuing our experiments in coevolution. The next three sections elaborate on these experiments.

3.3 Experiments B1-B5: Evolving a Defender with a Pre-Programmed Cutter

If the goal of our project is to model human player behavior, it must be noted that sometimes a new player can learn best by defending an experienced cutter, rather than trying to figure out good positioning while the cutter is also learning and developing strategies. To model this, we ran a series of experiments in which a defender is evolved to compete against a pre-programmed cutter. B1-B4 use a cutter brain which "fakes" backward and then cuts in, forward diagonally and to the light side. This most basic cut in frisbee is an ideal candidate for training a defender. B1 evolved a defender that correctly changed speed and direction with the defender's fake, but after following the defender for a short amount of time, it stayed upfield on the light side making circles. This caused us to question whether the stall penalty was too high, killing any individuals who stayed close to their defender (and accidentally got too close) and leaving only half-decent defenders like the ones we observed.

In order to test this, we lowered the stall penalty considerably so that instead of returning a fitness of -10 for every stalled time step, fitness was still calculated normally, and 25 was subtracted from the total fitness. The total fitness received during a stall, then, is comparable to the fitness received by a defender who has strayed a bit from its cutter, as the defender in B1 had. B4 used only this reduced stall penalty, and B5 added an additional neural network input node which was a binary stall sensor. Both experiments resulted in the defender running into the cutter immediately and stalling for the rest of the trial. This was, in fact, the most reliable way for the defender to get a reasonable fitness score, but it was poor behavior for an ultimate frisbee play.

Returning the stall penalty to its original high stakes, we tested another theory for the mediocrity of the defender's behavior: small population diversity. In order to encourage more varied behavior from multiple species, we increased the population size from 15 to 25 for experiment B4. The results of this experiment were considerably better and can be viewed in detail in Figure 3A. Increasing the population size had the desired effect, for while only 2 species were created over the course of B1, 6 species were created over the course of B4. However, the task is not terribly difficult, so it is not surprising to see no hidden nodes in the phenotype of the evolved neural network, even though the experiment was only run for 25 generations (Fig. 3A).

After the good behavior achieved in B4, we decided to try to evolve a more general defender by using a pre-programmed cutter with 3 different cuts (1 trial for each). Building off of the light-side in-cut of B1-4, we added a deep cut (”fakes” forward, turns, and goes deep, curving to the light side) and a dark-side in-cut (no fake because the cutter starts with good positioning away from the defender). Evolving a defender against these cuts in B5 did not produce great results. Rather than adapt three separate strategies for the three different types of cuts, it compromised by staying in the middle of the field regardless of the cutter's behavior

The best results of the B-series experiments were achieved in B4 evolving a defender against a pre-programmed in-cut cutter. The next question, then, is can we evolve a cutter to play well against this defender?

3.4 Experiment C1: Evolving a Cutter with a Pre-Evolved Defender

In a game of Ultimate, a human cutter takes the defender's behavior into account when deciding how to cut. For example, if the defender spends most of the game in front of the cutter, trying to cut off in-cuts, an intelligent cutter will start cutting deep instead. This type of behavior is what we hoped to see from C1.

C1 uses a population size of 25 and evolves a cutter from scratch against the top defender from B4. This defender, as is to be expected, is not a great general defender, since it was evolved only for in-cuts. When the cutter does not make an in-cut, the defender follows in its general direction, but not in the smoothest, straightest path possible (see Fig. 3B). The defender evolves to take advantage of this weakness and gain higher fitness by making a deep cut to the light side.

The neural network brain evolved in this experiment (see Fig. 3B) is more complex than that of B4, adding a hidden node which takes input from the angle-to-opponent sensor and the distance-to-handler sensor. C1's task is more complex than B4's task, because the evolved defender brain used in C1 is not as predictable as the pre-programmed cutter brain in B4. The added complexity in the evolved neural network reflects this.

3.5 Experiment D1: Back to Coevolution

After evolving a cutter and a defender separately, the important question is how those results compare to simultaneous competitive coevolution instead. The parameters of D1 are taken from B4 and C1, with 10-step trials for populations of 25 run for 50 generations. The cutter and defender are co-evolved from scratch. The resultant cutter behavior is quite good. The cutter goes slightly in as it turns around and then switches quickly to a relatively straight deep cut, curving to the light side in an ideal path to receive a disc from the thrower. The defender's performance is not so good. It starts out following the defender deep, but then it turns and effectively goes in circles for the rest of the trial (see Figure 3C).

The fitness graphs of D1 are typical of coevolution fitness graphs. In Figure 3C, we have overlayed the cutter and defender fitness functions to fit on one graph. Because the fitnesses are calculated differently and on slightly different scales, this graph should be read looking for trends and not for individual fitness comparisons. One characteristic trend is that because the defender and cutter are both competing for good positioning to catch the disc, their fitnesses are inversely proportional. As the cutter's fitness increases (e.g. because he gets farther away from his defender), the defender's fitness decreases (e.g. because he has let his cutter get farther away). This trait of

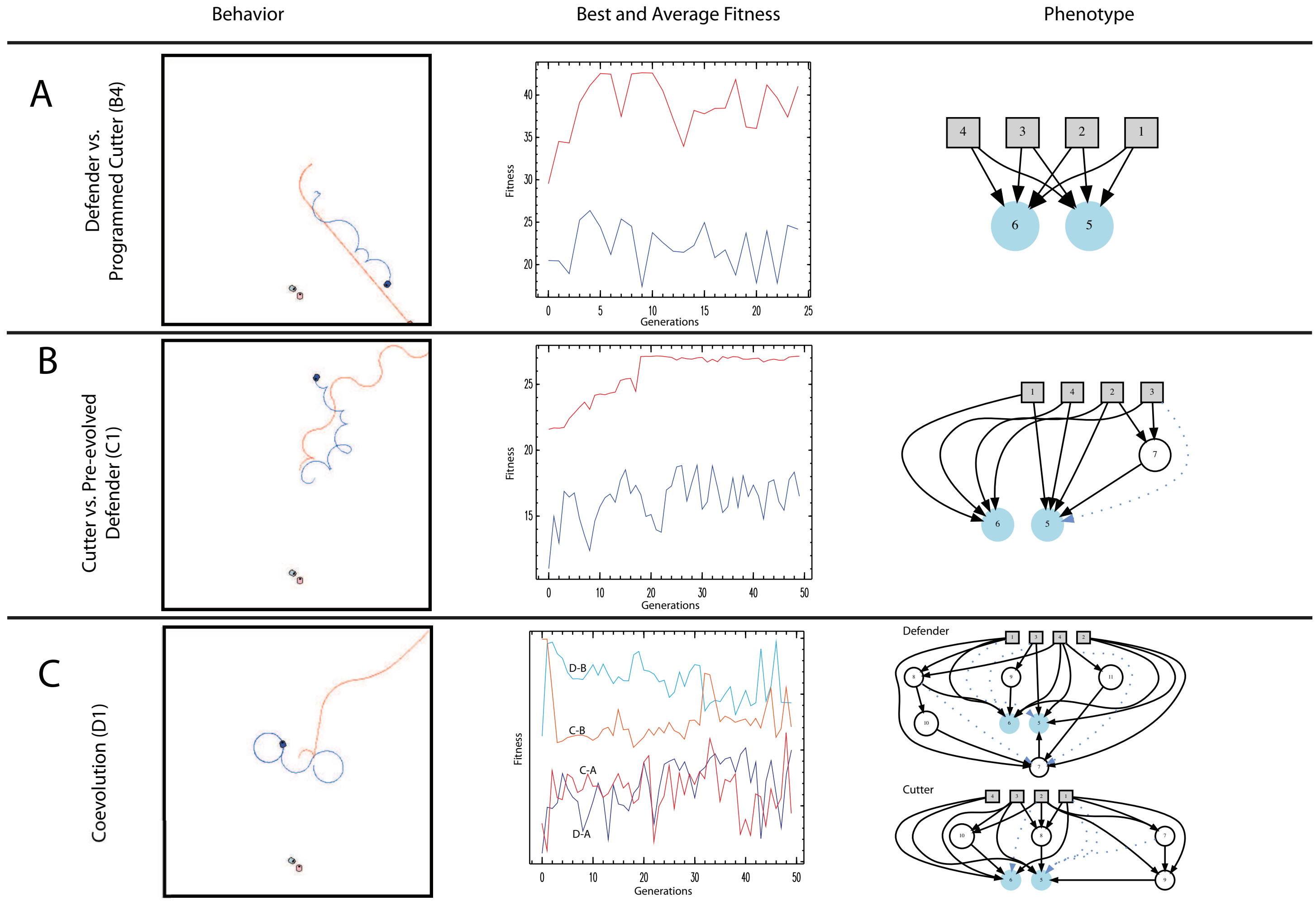

Figure 3. Behavior paths, fitness graphs, and NN phenotypes after evolution in 3 scenarios. In the behavior paths, blue represents the defender and red represents the cutter. In the fitness graphs, blue represents the average fitness and red represents the best fitness of each generation. The co-evolution fitness graph plots the cutter and defender's best and average fitness on the same graph (C-A = Cutter Average, etc.). Because the defender's and cutter's fitnesses are measured in different ways and on different scales, the graph in 2C should be analyzed for general trends rather than exact measurements.

competitive coevolution can be extended to the trends in fitness over time – there is no overall rise in fitness because as each player evolves better strategies, it is are also competing against a better opponent, resulting in no net gain or loss of fitness over time. The neural network phenotypes evolved in coevolution are noticeably more complex than those produced in single-player evolution because reacting to the movement of another intelligent opponent is a more complex task.

4 Summary and Conclusions

The aim of these experiments was to develop a fitness function and scenario that would enable coevolving robots to reproduce or invent strategies used in in standard Ultimate plays. We also explored single population evolution possibilities using either programmed opponents or opponents evolved against programmed robots. We achieved moderate success with coevolution. As seen in Figure 3C, the coevolved cutter from experiment D1 learned to cut deep to the lightside when it was poorly guarded. This behavior was also exhibited in single population cutters evolved against a pre-evolved defender (C1).

The defender seems to have had less success in coevolution, but it did learn to play man-on-man defense when evolved against a programmed cutter making in-cuts. The failure of the defender to progress in coevolution may have similar causes to those described by Nelson and White in their Capture the Flag experiments.[3] Like Capture the Flag, Ultimate is a complex task where the trajectory of the behavior is more important than exploiting a fitness function at the end of the trial. Furthermore, Ultimate Frisbee is complex enough that developing a fitness function is challenging. Further exploration of coevolution using starting populations that have already been evolved independently could be promising. However, we also believe that with further refinement of the fitness function and other parameters, unassisted coevolution has a lot of potential to develop interesting strategies for both offense and defense in the real world of Ultimate Frisbee.

Acknowledgments

We would like to thank Lisa Meeden for her leadership and assistance throughout this project. Our classmates in Swarthmore College's Adaptive Robotics class also provided inspiration and support, for which we are quite grateful. Special thanks goes to the men's and women's Ultimate Frisbee teams at Swarthmore for their consultation, encouragement, and enthusiasm throughout this project.

References

[1] Douglas Blank, Lisa Meeden, and Deepak Kumar. Python robotics: An environment for exploring robotics beyond legos. In *In Proceedings of the SIGCSE Conference*, pages 317–321. ACM Press, 2003.

[2] Gunnar Buason, Nicklas Bergfeldt, and Tom Ziemke. Brains, bodies, and beyond: Competitive co-evolution of robot controllers, morphologies and environments. *Genetic Programming and Evolvable Machines*, 6(1):25–51, March 2005.

[3] A.L. Nelson, E. Grant, G.J. Barlow, and M. White. Evolution of autonomous robot behaviors using relative competitive fitness. pages 145–150. IEEE International Conference, Oct 1-3 2003.

[4] A.L. Nelson, E. Grant, and T.C. Henderson. Evolution of neural controllers for competitive game playing with teams of mobile robots. *Robotics and Autonomous Systems*, 46(3):135 – 150, 2004.

[5] Stefano Nolfi and Dario Floreano. Coevolving predator and prey robots: Do ”arms races” arise in artificial evolution? *Artif. Life*, 4(4):311–335, Oct 1998.

[6] Karl Sims. Evolving 3d morphology and behavior by competition. *Artificial Life*, 1(4):353–372, 1994.

[7] Kenneth O. Stanley and Risto Miikkulainen. Competitive coevolution through evolutionary complexification. *Journal of Artificial Intelligence Research*, 21:63–100, 2002.