Competitive Coevolution of Jointly Evolving Behavior and Bodily Traits

Ibrahim Khalil Mejri, *Student, UNITN,* Kemal Alp Sezer, *Student, UNITN,*

***Abstract*—The application of artificial intelligence (AI) in the video game industry has revolutionized the dynamics of non-playable characters (NPCs) and game-playing agents, creating more engaging and adaptive gameplay experiences. This project explores the use of evolutionary algorithms, specifically the NEAT (NeuroEvolution of Augmenting Topologies) algorithm for behavior in conjunction with Evolutionary strategies for physical traits, in simulating a predator-prey ecosystem within a competitive coevolution setup. By evolving both the neural network topologies and physical attributes of agents, we aim to model and predict complex interactions and outcomes in an adaptive environment. Our implementation of Evolutionary Algorithms using the structure of Agar.io demonstrated significant learning and adaptation capabilities. Extending this approach, we simulate an ecosystem where prey and predators continuously evolve to outsmart each other, providing insights into ecological balance and evolutionary biology. The competitive coevolution framework allows for dynamic adjustments, ensuring balanced gameplay and offering a robust tool for theoretical and applied ecological studies. This project not only enhances AI-driven game development but also contributes to understanding the stability and diversity of ecosystems under evolutionary pressures.**

***Index Terms*—Evolutionary Algorithms, NEAT, Evolutionary Strategies, Co-evolution**

I. Introduction

THERE is a broad application of AI in the video game industry[1] on nonplayable characters (NPCs) and game-playing agents. AI agents can create dynamic and adaptive gameplay experiences, adjusting difficulty levels and strategies to match the player's skill, keeping the game challenging and engaging while NPCs controlled by AI provide more unpredictable interactions, enriching the game-playing experience.

AI-driven video game-playing agents have achieved remarkable milestones using a variety of advanced methods. For instance, AlphaGo[2] combined deep learning with Monte Carlo Tree Search (MCTS) even defeated advanced human players. Deep Q-Networks (DQN)[3] leveraged deep Q-learning with convolutional neural networks (CNNs) to achieve superhuman performance in Atari games by learning directly from pixel inputs. AlphaStar[4] reached Grandmaster level in StarCraft II using deep reinforcement learning and multi-agent training to master complex strategies. The NEAT algorithm [5], exemplified by MarI/O in Super Mario World allowed the agent to learn and navigate game levels effectively.

Our project takes inspiration from these successful applications of AI and extends it further by exploring the capabilities of evolutionary algorithms in game development. We began by implementing the NEAT algorithm in the classic game of Pong and achieved promising results that demonstrated their ability to learn and adapt. Building on this success, we created an agent from the popular Agar.io game, focusing on survival through food consumption. This experience sparked our curiosity – how would such an agent fare in a more complex environment where prey and predator compete for survival? This project aims to answer that question by simulating a predator-prey ecosystem using evolutionary algorithms, pushing the boundaries of AI-driven game development.

By allowing the modification of attributes as well as behavior in this competitive setup, game designers can simulate the ongoing adaptation between competing entities, such as player characters and adversaries in multiplayer games. This process helps identify and correct imbalances where one side might gain an unfair advantage by careful tuning of the parameters and the trade-off functions.

Professor Giovanni Iacca, Course coordinator of Bio Inspired AI Course.

II. Related Work

A. Predator-Prey Ecosystem

In competitive coevolution, there is a general conflict of interest whether within a species (intra-species) or between two different species (inter-species). An inter-species competition generally involves a prey-predator or a host-parasite setup. In this setting, the prey and predator are locked in a constant evolutionary arms race. The prey must not only consume food to survive but also develop cunning strategies to avoid becoming a predator's meal. Predators, on the other hand, rely solely on consuming prey for their survival, constantly honing their hunting skills to overcome increasingly evasive prey.

B. Neuroevolution using NEAT

Neuroevolution (evolving artificial neural networks using evolutionary algorithms) is particularly effective in domains where adaptive and complex behaviors are required, as it can evolve solutions that perform well in dynamic and unpredictable environments, leading to the emergence of sophisticated behaviors. NEAT (NeuroEvolution of Augmenting Topologies) [5] within this domain is a powerful approach that focuses on evolving both the weights and the structure of the networks simultaneously. It is generally useful in controlling game-playing agents and in robotics applications in which action policies play an important role which suits the context of a prey predator behavior.

C. Evolutionary Strategies

Evolutionary strategies[6], which are commonly referred to as the evolution of the evolutionary process itself, allow for adaptive strategies to control the mutation mechanism. We decided to use uncorrelated mutations with multiple σ. In this approach, each individual in the population has its own set of strategy parameters, including multiple standard deviations σ, which control the mutation strength.

III. Agents and Evolution

A. Agent characteristics

The agent is the individual that we are going to evolve, which could be either a prey or a predator.

- The agent has a circular shape with a unique mass and speed.
- The agent uses lines for vision. These vision lines are arranged around it at specific angles and radius and are equally spaced. Each line returns the distance to an object in the environment, and it will be used as input of the neural network.
- The prey uses two types of vision lines: one for detecting food and the other for detecting predators.
- The predator, on the other hand, has vision lines solely for detecting prey.
- The agent has an initial energy level that depletes over time based on its mass and speed. The agent dies when its energy is depleted. The energy increase by consuming food and at the end the survival time will be used for the fitness function.
- The agent has an initial energy level that depletes over time based on its mass and speed. The agent dies when its energy is depleted.

B. Neural Network

*1) **Prey**:* For preys the initial neural network is composed of 14 inputs (7 for the food sensors and 7 for the predators sensors) with 2 hidden layers and 3 outputs (to rotate left or right or stay at the same direction)

*2) **Predator**:* For predators the initial neural network is composed of 7 inputs for the preys sensors with 2 hidden layers and 4 outputs (to rotate left or right or stay at the same direction or to dash).
Predators have the privilege of dashing to a direction but it will consume more energy equivalent to three time of the normal consumption.

C. NEAT

Starts from a simple network and evolves more complex topologies over generations.

- **Speciation:** NEAT uses speciation to protect new innovations by grouping similar networks into species. This ensures that unique topologies have a chance to develop and mature before competing with other networks.
- **Crossover and Mutation:** NEAT uses genetic operators like crossover and mutation to evolve networks. Crossover involves combining the structures of two-parent networks, while mutation can add nodes, add connections, or alter connection weights.

D. Evolving physical properties

Simultaneously with the evolution of the neural network, which is considered the mind of the individual, the physical attributes are also evolving. These attributes include:

- Speed
- Mass
- Field of vision
- Rotation speed (determines how quickly the agent can turn)

The algorithm should find a set of physical attributes that work well with the agent's neural network because some parameters are proportional.

The final speed of the agent is equal to:

$$speed - \frac{Mass}{10}$$

The bigger the agent, the slower it gets.

The energy depletion rate of the agent is equal to:

$$\frac{2 \times (Mass + Speed)}{100}$$

Smaller agents use less energy and are faster then the bigger, but at the mean time an agent cannot eat something that is bigger then

$$1.5 \times Agent's\ mass$$

*1) **Selection**:* The parents are the same for the NEAT algorithm selected based on their fitness using the default NEAT selection which is tournament selection.

*2) **Crossover**:* The selected parent will be used to generate an offspring with simple one-point crossover.

*3) **Mutation**:* After the creation of the offspring it will go through Self-adaptive (uncorrelated mutations with multiple σ) it is one of mutation techniques used for evolutionary strategies[6].

E. Fitness Function

In each generation, the prey population (consisting of 50 individuals) will be evaluated against the predator populations from the three previous generations (n-1, n-2, and n-3). The fitness of each prey individual will be determined by averaging its performance across these three tests.
For each evaluation round:

- each time 10 prey individuals from the current population will play against 4 random samples of predators from one of the previous three generations.
- The average performance of each prey against these predator samples will be calculated to determine its fitness.

Similarly, the predator population (comprising 25 individuals) will undergo evaluation against the prey populations from the three previous generations. In this case:

- each time 2 predator individuals from the current population will play against 8 random samples of prey from one of the previous three generations.
- The average performance of each predator against these prey samples will determine its fitness.

This evaluation scheme ensures a comprehensive assessment of each generation's individuals by considering their interactions with past populations.

*1) **Survival time as fitness**:* Using only the survival time as fitness most of the time predators converge to individuals with the smallest mass and the lowest speed so they can conserve energy and die like this without trying to catch preys.
While preys show promising results with this fitness function while sometimes tend to minimize the the mass and the speed until some preys tends to converge as the predators to individual incapable of getting food.

*2) **Food Score as fitness**:* Upon consuming food, each individual's score increases by one point, accompanied by a boost in energy. In terms of fitness evaluation, we propose prioritizing the score accumulation over the duration of survival.
For this experience the predators show promising results for hunting preys.
While preys also were very good at looking for food they tend to neglect the existence of the predators and show less interest in avoiding them which lead to fast death of preys.
Also both individuals show less interest in tuning the physical attributes and maximized their speed and mass.

*3) **combined fitness function with penalties**:* This time, the fitness function is a weighted sum of survival time and food score. To ensure active behavior, additional penalties are applied:

- Predators receive a penalty if their score is zero, encouraging them to hunt.
- Preys incur a penalty if they fail to gather any food during the training period, and an additional penalty if they are killed by a predator, promoting avoidance behavior.

This fitness function has yielded the best results for both preys and predators.

*4) **Hall of fame**:* The hall of fame is comprised of individuals with the best fitness across all generations. However, if a population of prey is poor at detecting predators, the predators from that generation might have artificially high fitness levels despite not being genuinely superior. To address this, an individual is added to the hall of fame only if its fitness surpasses that of an existing member. Every 20 generations, the individuals in the hall of fame compete against the latest generation, and their fitness scores are updated accordingly. Additionally, after these updates, the hall of fame individuals are injected into the next training set to introduce variability and help avoid local maxima in the fitness function.

F. Intra-Species Conflict

Due to limited resources, individuals that move randomly or slower than others will have a lower chance of competing successfully, resulting in decreased fitness and exclusion from future generations.
Initially, this competition helped filter out less capable individuals. However, as generations progressed and most individuals became adept at detecting food, the mean fitness of the generation began to drop. This decline occurs because there isn't enough food for everyone to adequately test their skills, and randomness plays a significant role. For example, an individual that spawns in a food-rich area will have a higher fitness compared to others who might actually be more skilled but less fortunate in their spawning location.
Moreover, the algorithm does not account for intra-species conflict, as each individual moves independently of others from the same species, and the food and individuals location at the start is random.
To solve this we decided to drop the intra-species conflict:

*1) **Training Preys**:* For the preys, food will regenerate whenever the number of food cells drops to half of the initial count. This ensures that all individuals have equivalent opportunities to find food. However, this could also mean that an agent moving randomly might achieve a high fitness. This issue is mitigated by the presence of predators, which adds a layer of challenge and prevents random movement from being an effective strategy.

*2) **Training Predators**:* Predators face a greater challenge due to the limited number of preys. Increasing the prey population significantly prolongs training times as it demands substantial memory resources. Additionally, preys adapt more quickly than predators because food is readily available.
To mitigate this issue, each predator training session involves only 2 predators against 8 preys. To address the training disparity, when a prey successfully gathers a certain amount of food (20 cells for example), it is deemed a good prey. An immobile copy of this prey is then created, which predators will treat as a real prey. This helps predators hone their hunting skills more effectively specially against good preys.

IV. Results

As a result, both preys and predators can interact with their environment through their sensors.

Plotting the fitness functions may not provide a clear understanding of the dynamics.

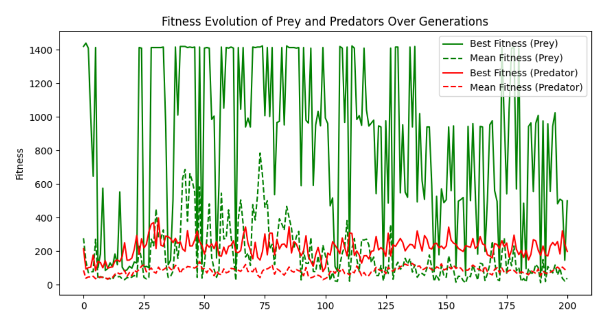


Fig. 1: Fitness Evolution of Predators and Preys over 200 Generations

Especially as each generation is compared against previous ones, the comparative analysis of prey and predator generations using this graph might be challenging.

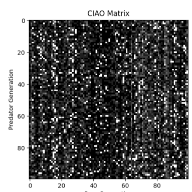
The fitness of prey and predators is interrelated; robust prey generally imply lower fitness for predators, and vice versa.

However, this relationship is not strictly proportional. It is possible to have a few highly fit prey coexisting with fit predators due to the mechanisms we've implemented to help predators close the gap in their learning rates. This complexity prevents us from creating a straightforward CIAO graph[7] to compare generations directly.

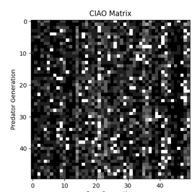
To address this, we employed various metrics to create the CIAO graph by selecting 8 prey and 4 predators from each generation to compete against each other. If all prey successfully consume 40 cells of food, they are deemed the winners, represented by white on the graph. Black indicates that none of the prey were capable of achieving this score before dying. Thus, the fitness is given by:

$$Fitness = 1 - \frac{successful\_prey}{8}$$

where $successful\_prey$ are those who achieved 40 cells of food. The darker the color, the fewer prey were able to succeed, either due to predators or insufficient ability to search for food.



(a) CIAO graph for generations 100 to 200



(b) CIAO graph for generations 150 to 200

Fig. 2: Comparison of CIAO graphs

However, the CIAO graph was not useful in this case because, as mentioned earlier, the individuals were not primarily evolved to compete against each other but to survive. Several factors could cause this distortion. For example, if the predators of one generation are smaller compared to the prey, they might not be able to attack effectively, even if they are otherwise very skilled. Conversely, even if prey are good at avoiding predators, they might not be able to escape from predators that excessively use dashing without considering energy consumption.

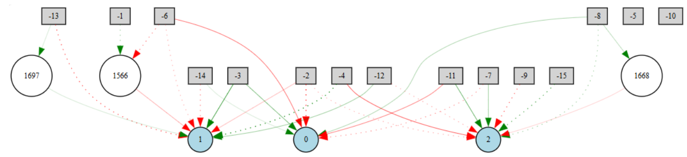
Through the graphs, we observe that the dominant color is grey, indicating more ties than wins. This means that some prey were able to succeed, while others were more likely devoured by predators.

For a more insightful evaluation, simulating the hall of fame of preys against the hall of fame of predators would be the optimal approach.

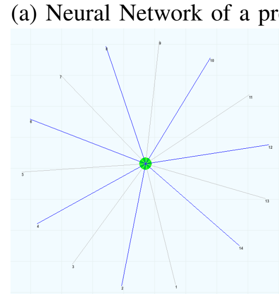
In early generations, many predators excessively used the dash ability. However, we observed that the most successful predators chose not to use dash in order to conserve energy, as dashing consumes up to three times more energy than normal movement.

While, in some experiments, we enabled preys to dash to provide them with a defensive mechanism against predators. However, due to the abundant availability of food, preys used dash excessively with every movement, leading to an imbalance and making the game unfair for the predators.

We explored alternative defense mechanisms for preys, such as creating decoy copies of their bodies to distract predators. However, this method was more energy-consuming than dashing. Consequently, we allowed the agents to choose their preferred defense mechanism during their physical evolution. After a few generations, all individuals opted for dashing because it consumes less energy and is highly effective.



(a) Neural Network of a prey



(b) Prey

Fig. 3: Prey from the hall of fame

This is a figure of a prey after training we can observe developed neural network. Inputs, depicted in grey, showcase the sensory data, while outputs, in blue, indicate the chosen actions: 0 for left turn, 1 for right turn, and 2 for maintaining the current direction.

for the shape the blue lines are the predators sensors and the grey are for food. Key attributes of the prey include a mass of 30, a speed of 13.83 units, a wide Field of vision of 180 degrees, and an agile rotation speed of 23.29 units.

V. Limitations and Future Work

A. Limitations

The biggest limitation we faced was computation time. We relied on the NEAT-Python library for neural networks, and after delving deep into the project, we discovered that it does not support GPU functions. At that point, restarting from scratch was not an option. As a result, the majority of our training was limited between 100-200 generations with each change. With more generations, we might have achieved better results.

B. Potential for expansion

Our project is a simplified prey-predator ecosystem, but it could be made more complex by adding functionalities such as varying the map, introducing obstacles, providing places to hide, and offering tools for individuals to use. Additionally, we wanted to experiment with different algorithms, allowing agents to select it's behaviour during their evolution along side

with the physical attributes.
We also attempted to create packs where agents could choose to join a pack or operate solo, and considered using Particle Swarm Optimization (PSO) to simulate pack movement. However, due to time constraints and complexity, we postponed this idea.

Another example of potential improvements includes implementing and balancing more defensive and attacking techniques. We tested a few, but balancing them required more time and computation resources.

VI. Conclusion

The primary goal of this project was to simulate an environment and observe the behaviors of individuals across generations. While creating the perfect individual was not our main objective, learning to hunt or escape was essential for our observations. By controlling the fitness function and abilities, we achieved varied results. Changes in individual priorities led to behavior modifications, with some minimizing all attributes to maximize their survival time and giving up on hunting, while others evolved their skills for survival.

Additionally, the behaviors of the species are interdependent. If prey optimize their size or speed, predators will correspondingly adjust their attributes to conserve energy while maintaining their hunting skills.

This project has significant potential for expansion. By incorporating additional abilities and employing diverse algorithms to fine-tune more parameters, the simulation can be further enhanced. Such advancements could be valuable for the video game industry and evolutionary robotics.

References

[1] Yueliang Wu, Aolong Yi, Chengcheng Ma, Ling Chen. Artificial intelligence for video game visualization, advancements, benefits and challenges[J]. Mathematical Biosciences and Engineering, 2023, 20(8): 15345-15373. doi: 10.3934/mbe.2023686
[2] Silver, D., Huang, A., Maddison, C. et al. Mastering the game of Go with deep neural networks and tree search. Nature 529, 484–489 (2016). https://doi.org/10.1038/nature16961
[3] Volodymyr Mnih and Koray Kavukcuoglu and David Silver and Alex Graves and Ioannis Antonoglou and Daan Wierstra and Martin Riedmiller, Playing Atari with Deep Reinforcement Learning, 2013
[4] Vinyals, O., Babuschkin, I., Czarnecki, W.M. et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. Nature 575, 350–354 (2019). https://doi.org/10.1038/s41586-019-1724-z
[5] K.O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies.", 2002
[6] Beyer H-G (1996) Toward a theory of evolution strategies: Self-adaptation. Evolutionary Computation 3(3): 311-347
[7] Cliff D, Miller GF. Visualizing coevolution with CIAO plots. Artif Life. 2006 Spring;12(2):199-202. doi: 10.1162/106454606776073341. PMID: 16539763.