Optimal Learning Conditions for Hunting Strategies and Foraging Behaviors in Evolutionary Robotics

by

Toan Nguyen

Supervised by Dr. Chris Marriott

A senior thesis submitted in partial fulfillment of the departmental honors requirements for the degree of

Bachelor of Science

Computer Science & Systems

University of Washington Tacoma

October 2023

Faculty advisor: C Marriott Date:10/18/2023

CSS Program Chair: Thomas Capaul Date:10/19/2023

Presentation of work given on: **10/16/2023**

Abstract

This study focuses on simulating hunting and foraging behaviors in evolved robots. The experiments sought to investigate the influence of role awareness, overall agent speed, speed advantage and gradual consumption on hunting and foraging behaviors. Our goal was to observe the conditions which foster the predator's success in capturing prey along with prey's in consuming food and evading predators. Our agents use artificial neural networks trained using the NeuroEvolution of Augmenting Topologies (NEAT) algorithm. This approach allows our agents to evolve unique behaviors involving evasive movement patterns, cornering prey, staying out of bound and circling stationary food. The experiment was evaluated through aggregated data analysis on the total stationary food consumed by prey, the total prey hunted per generation and the total ticks out of bound. The experiments revealed that role awareness had a minimal impact on agent performance, with enabling it resulting in a slight performance decrease. Our findings suggest that predator agents are most successful at capturing prey when maintaining equal speeds between these two roles of agents emerged, as any speed advantage granted to one type of agent enhanced their respective roles but disrupted the balance between them. When it comes to the overall speed, the fastest option is not the most optimal choice. Rather, a moderate value proves to be more effective. High speed causes both agents to go out of bound more frequently, while a low speed leads to poor food consumption. Additionally, changing the ticks for full food consumption mainly benefits prey with them eating more. While for predators, the performance does not show improvement despite the extended presence of prey in the area where food is spawned.

1 Intro

Evolutionary robotics is a growing field that harnesses the power of natural selection to develop complex behaviors. In this paper, the simulation focuses on demonstrating the effectiveness of evolutionary robotics in developing hunting and foraging behaviors within a simplified and controlled environment. To achieve this, we used the Neural Evolution of Augmenting Topologies (NEAT) algorithm to develop neural network topologies and fine-tune their weights.

The selection of appropriate parameters is an important aspect of evolutionary robotics. Well chosen settings can result in the production of high quality agents. This study focuses on the significance of agent and environmental parameter choices in shaping the outcomes of evolutionary robotics experiments. Specifically, our experiment aims to investigate the influence on agent performance by key factors, including role awareness, overall agent speed, speed differentials between two roles of predator and prey agents along with partial consumption of stationary food. For prey agents, performance is evaluated based on their ability to efficiently consume food while evading predators, while the effectiveness of predators is measured by their hunting capabilities. Through systematic exploration of these parameters, we aim to uncover optimal configurations that yield desirable outcomes in evolutionary robotics. The findings of this study will provide valuable insights for designing and fine-tuning future robotic systems.

2 Model (Methodology)

2.1 Agent

An agent is represented by a circular shape with a radius r = 7.5 pixels, and its direction of movement is determined by an angle α (measured in radians). Each agent is equipped with two wheels on either side, each having a radius $R_w = 1$. The movement of the agent is governed by the kinematic physics model designed for differential drive robots. The power p provided to the wheels is directly controlled by the output nodes of the artificial neural network (ANN), ranging from -5 to 5. The power values for the left and right wheels are denoted as p_l and p_r, respectively. The resulting motion of the agent is determined by the vector velocity $V = \left[V_x, V_y\right]$ generated by these wheel powers.

$$V_x = \frac{R_w}{2} * (p_l + p_r) * cos(\alpha) \qquad [1]$$

$$V_y = \frac{R_w}{2} * (p_l + p_r) * sin(\alpha) \qquad [2]$$

$$Angular\ velocity : V_\alpha = \frac{R_w}{r} * (p_r - p_l) \qquad [3]$$

2.1.1 Predator and Prey Role Assignment:

Agents are categorized into two roles: predator and prey, which significantly impact their energy acquisition methodology. Prey agents are solely capable of consuming stationary food sources to replenish their energy levels. In contrast, predators cannot derive energy from stationary food and instead rely on collisions with prey agents to obtain sustenance. The successful consumption of prey by predators results in the transfer of energy to the predator agents.

2.1.2 Sensors

In order to perceive their environment, the agents employ a raycasting system to simulate vision capabilities. This system involves the emission of multiple rays from an eye, which is positioned at distance r from the center of the agent in the direction α it is facing. The rays are uniformly distributed within the agent's vision cone at an angle β, which is experimentally set as default at 180 degrees. In our experiments, we utilize 13 rays with an unlimited range to capture information from the agent's surroundings.

2.1.3 Calories/metabolism

Energy that the agent uses is symbolized as calories. Predators possess an unlimited amount of energy, whereas Prey agents start with a finite energy value of 250 calories. The energy of each agent gradually decreases over time as they perform various actions, such as moving. This energy depletion is determined by the equation:

$Energy = Energy - \lambda \frac{|p_l| + |p_r|}{10} - 0.1$ **[4]**. Since the maximum magnitude of each power value is 5, the sum of $|p_l| + |p_r|$ can reach a maximum value of 10. The division by 10 in the formula normalizes the powers, ensuring that the energy loss from movement falls within the range of [0, λ]. In our experiments, the default value of λ is set to 1, effectively limiting the energy loss from movement to a maximum of 1 calories per tick. When an agent's energy is depleted, it becomes deactivated, rendering it unable to move or consume food. Both predator and prey agents are subject to the same deactivation protocol when they are beyond the bounds of the environment. Food is available in the environment and can be consumed by prey to replenish a fixed amount of energy (50 calories).

2.2 NN (Artificial Neural Network):

The agent's vision system serves as the input for the artificial neural network. Each ray in the vision system corresponds to two input nodes, representing distance and hue information. The distance input informs the agent about the distance to an object in its surroundings, while the hue input aids in identifying the nature of the object. Hue values span the range of [0, 360], but for the purpose of feeding them as inputs to the ANN, they are normalized to the range of [0, 1]. The hues perceived by the agent are relative to their role within the environment, providing role-specific color information for effective perception and decision-making processes.

	Hue	**Normalized Hue**
Value for Wall	30 (Orange ■)	0.08333

Value for Full Stationary Food	330 (Pink ■)	0.91667
Value for Predator Agent	40 (Orange ■)	0.11111

Table 2.2.1.1 Hue values for prey vision system

	Hue	**Normalized Hue**
Value for Wall	30 (Orange ■)	0.08333
Value for Full Stationary Food	180 (Cyan ■)	0.5
Value for Prey Agent	330 (Pink ■)	0.91667

Table 2.2.1.1 Hue values for predator vision system

The hue of stationary food in the agent's vision system is determined through linear interpolation between values representing the depleted level and the full level, based on the percentage of total calories available in the stationary food. The chosen hue values are designed to reflect the desirability of the entity, with 0 representing undesirability, 180 as neutral, and 360 as desirable. Following this principle, the prey's vision system assigns similar hue values to both predator agents and walls, enabling them to perceive both as potentially dangerous entities. For predators, since stationary food is no longer their target, the hue value for prey is assigned the same value as stationary food, while the hue value for stationary food is reassigned to 180 to indicate its non-threatening and non-desirable nature. In addition to the hue value inputs, the distance from the start of each ray to the collision point is inverted and normalized between the range of [0, 1]. The normalization is performed by dividing this distance by the maximum possible distance, which is calculated as the length between opposing corners of the environment minus the agent's diameter. This normalization ensures that the maximum length of a ray fits within the [0, 1] domain, with an input of 1 representing a distance of 0 and an input of 0 representing an object located at the maximum distance. Lastly, the agent's vision system incorporates two outputs for the wheels, corresponding to the left and right wheels. These outputs determine the power that drives the wheels, as mentioned previously.

2.3 Genome

In the NEAT algorithm, genomes serve as the encoding mechanism for constructing artificial neural networks. These genomes, as described by Stanley and Miikkulainen (2001), are specifically designed to facilitate the alignment of corresponding genes during crossovers. The encoded information includes both the connections between nodes and the corresponding weights associated with these connections. In the original NEAT paper by Stanley (2001), it was observed that frequent mutations led to more successful results. This tolerance to frequent mutations is attributed to the protective effect of speciation within the system. As they stated, “The system is tolerant to frequent mutations because of the protection speciation provides”(Stanley 2001) Taking inspiration from these findings, our project adopted a strategy of frequent yet small mutations for the numeric mutations. In line with the work of Stanley and

Miikkulainen (2001), we incorporated an 80% chance of mutation for each weight, as it was shown to be effective in promoting variation within the neural network structures.

2.3.1 Numeric Mutations

We use a modified sigmoid function for our logistic activation function:

$$\frac{2}{1+e^{-k(x-m)}} - 1 \qquad [5]$$

Along with the weights, each non-input node, including the output nodes, is associated with k and m value, which can be subjected to mutation with an 80% probability, mirroring the mutation chance for weights. The default value for k is set at 0.75, while the default value for m is 0. The mutation of k allows the agent to adapt its overall speed by modifying the shape of the sigmoid function. A larger k value results in a steeper function, influencing the agent's speed adjustments accordingly. On the other hand, mutating m enables the agent to prioritize specific output ranges by shifting the sigmoid function along the x-axis. This allows the agent to favor certain output values and fine-tune its behavior based on environmental conditions and objectives.

For all numeric mutations there are overlapping mutation types:

Mutation Operation	Definition	Occurrence Chance	K Value Mutation Range	M Value Mutation Range	Weight Value Mutation Range
Re-Roll	Rescales the mutations by up to a large amount, but with the mean being to scale by either -1 or 1	0.1%	[-2, 2]	[-2, 2]	[-2, 2]
Shift	Shift up or down by a small amount with the uniform distribution centered around 0	49.9%	[-0.05, 0.05]	[-0.3, 0.3]	[-0.5, 0.5]
Scale	Multiply the current number by either a uniform random number or the inverse of said number. The uniform distribution is centered around multiplying by 1.	50%	Re-scale up to 0.25 or down to 0.2	0.25 or down to 0.2	Re-scale up to 0.5 or down to 0.(3)

Table 2.3.1.1 Mutation Operations

2.4 Environment

The environment is defined as a world that ensures the isolation of agents from each other. The distribution process entails pairing agents into separate worlds, where they will compete as predator and prey which is assigned to them randomly. In every world, exactly one agent assumes the role of prey, while the other takes on the role of predator. To ensure a comprehensive assessment of agent performance, a second simulation run is conducted, where the roles from the initial run are reversed. This approach allows for a more balanced evaluation where every agent can perform as both predator and prey.

2.4.1 Stationary Food

In the simulation, the stationary food items resemble a circle with a radius of r = 9 pixels and are centered on specific coordinates. To introduce variability in their placement, a method called food pod is utilized for spawning food. These food pods act as zones where the food can randomly appear. Each food pod is represented by a circular region with a radius of 200 pixels. In total, the environment consists of four food pods positioned at the North, East, South, and West regions. The main purpose of employing food pods is to introduce randomness in the initial spawning locations of the food items during each simulation run. This ensures that the distribution of food is not fixed and allows for dynamic and unpredictable scenarios. To acquire energy, the prey agents must overlap with the stationary food for a specific duration measured in ticks. The calories rewarded is calculated as $\overline{Amount\ of\ ticks^{50}\ overlapping}\ calories$. For the instant consumption method, this duration is set to 1 tick. As for gradual (partial) consumption, it is set at 50 ticks as default in our experiment. It is noted that the prey agents do not need to continuously overlap with the food to perform consumption. The ticks counter will reset after 100 ticks. Only when a prey fully consumes the food, it will be regenerated and spawned in another food pod within the environment. The ratio of spawning food is set to 2 per 1 prey agent. This ratio ensures a balanced food supply while also offering an alternative option when the predator is circling within the vicinity of a food.

2.4.2 Wall / Border

To encourage the agents to stay within the boundary of a world, we use walls constructed in the shape of a square with the size of 1000 by 1000 pixels. Each wall is represented by the coordinates of its two endpoints, forming a line. Walls deactivate agents upon contact to prevent them from exiting the designated area. Four walls are positioned along the North, East, South, and West borders, effectively enclosing the environment.

3 Experimental Setup (Design and Architecture)

3.1 Agent Evaluation

The evaluation of agent performance at the end of each NEAT generation is conducted using a fitness function, which plays a major role in agent elimination and crossover processes. The fitness function takes into account various factors, including the total calories consumed (C), the number of ticks the agents spent out of bounds (T), whether the hunt was successful (H) and the time when the hunt concluded (Hc) representing a bonus to prey when it survives longer or

predator when it successfully captures the prey faster. These components contribute to the overall fitness score, which determines the agent's performance in the simulation.

$$Fitness = w_{f0} * C + w_{f1} * T + w_{f2} * H + w_{f3} * Hc \quad \textbf{[6]}$$

Fitness Attribute	**Weight value**
Calories eaten (C)	$w_{f0} = 10$
Ticks out of bounds (T)	$w_{f1} = -2$
Successful Hunt (H)	$w_{f2} = 50$ (Predator) $w_{f2} = -50$ (Prey)

Hunt Conclusion Bonus ( ) Hc	$w_{f3} = 50$

Table 3.1 Fitness Function Weights

In specific, each of these factors are defined as follows. The calories consumed (C) awarded to each agent are proportional to their calories consumed. For prey, it is from their food consumption. While for predators, it is from eating the prey. The amount of calories a predator can gain from prey depends on the calories prey obtained from stationary food during the run. This aims to incentivize predators to allow the prey to eat up. Additionally, the total number of ticks spent out of bounds (T) is one of the punishment metrics for agents. Agents are immediately deactivated when they move out of bounds, accumulating a higher value for T. As for the successful hunt H, it is the reward $w_{f2} = 50$ given to the predator when it successfully captures the prey. On the other hand, prey receive this penalty for being caught, denoted as $w_{f2} = -50$. Lastly, the hunt conclusion bonus is given based on the percentage of ticks the prey was alive during the run, while for the predator agent, it represents the percentage of ticks remaining in the run after capturing the prey.

3.2 Speciation

3.2.1 Species Selection

Speciation is a crucial component of NEAT aimed at preserving innovations that may not yield immediate short-term improvements but have the potential for long-term performance enhancements. Agents with similar genome structures are grouped into species. Throughout the simulation, these species compete with one another. Fitness sharing is implemented among agents within the same species to prevent domination of the population. During reproduction, the method to produce the next generation is by breeding random individuals from each species, based on their average fitness. This approach encourages diversity and allows promising genome material to persist and evolve over time.

3.2.2 Compatibility measuring

Compatibility measuring, denoted as δ, is a method employed in NEAT to assess the similarity between agents, enabling their classification into distinct species. The calculation of compatibility follows a similar approach to the original NEAT compatibility distance formula, with a crucial modification. Given that our genetic mutations involve k and m values, they must be taken into account during compatibility assessment.

$$\delta = c_1 \bar{E} + c_2 \bar{D} + c_3 \bar{W} + c_4 K + c_5 M$$

The coefficients $c_1 c_2, c_3, c_4$ and c_5 determine the importance of the factors E (Excess Genes), D

D (Disjoint Genes), W (Average weight difference), K (Average K value difference) and M (Average M value difference).

For our settings, we chose:

Coefficients	**Value**
c_1	1
c_2	1
c_3	0.3
c_4	0.4
c_5	0.4

Table 3.2 Species Compatibility Weights

3.3 Parameter Varying in Experiments

Four different parameters were varied during the study: role awareness, overall agent speed, speed differentials between two roles of predator and prey agents along with gradual consumption of stationary food.

3.3.1 Role Awareness

During the course of this experiment, two types of distinct tests were conducted. In the first type, the agents were provided with the knowledge of their assigned role as either a prey or a predator. We convey this information to the artificial neural network (ANN) by pushing to the input a value of 1 if the agent was a predator and 0 if it was a prey. In contrast, the other withheld this information from the agents, thereby making the agents unaware of their specific role. To further examine role awareness's impact on the agent's performance, we performed these two types of tests on gradual consumption (with ticks to fully consume set at 50, calories per food at 250), and instant consumption, where prey agents only need to overlap their body with the food's to achieve full consumption. This will be measured in calories instead of food counts.

3.3.2 Speed advantages

During the course of this experiment, the speed advantages are given to prey and predator as:

	Prey Agent Max Speed	Predator Agent Max Speed
Prey Advantage: 125% faster	6.25	5
Prey Advantage: 150% faster	7.5	5
Predator Advantage: 125% faster	5	6.25
Predator Advantage: 150% faster	5	7.5

Table 3.3 Speed Advantage Setup

3.3.3 Overall speed

During the trials in this experiment, both predator and prey agents shared the same maximum speed. The values of the shared speeds tested are as follows: 1 pixels per tick (25% Default Speed), 2.5 pixels per tick (50% Default Speed), 3.75 pixels per tick (75% Default Speed), 5 pixels per tick (Default Speed), 7.5 pixels per tick (150% Default Speed), 10 pixels per tick (200% Default Speed).

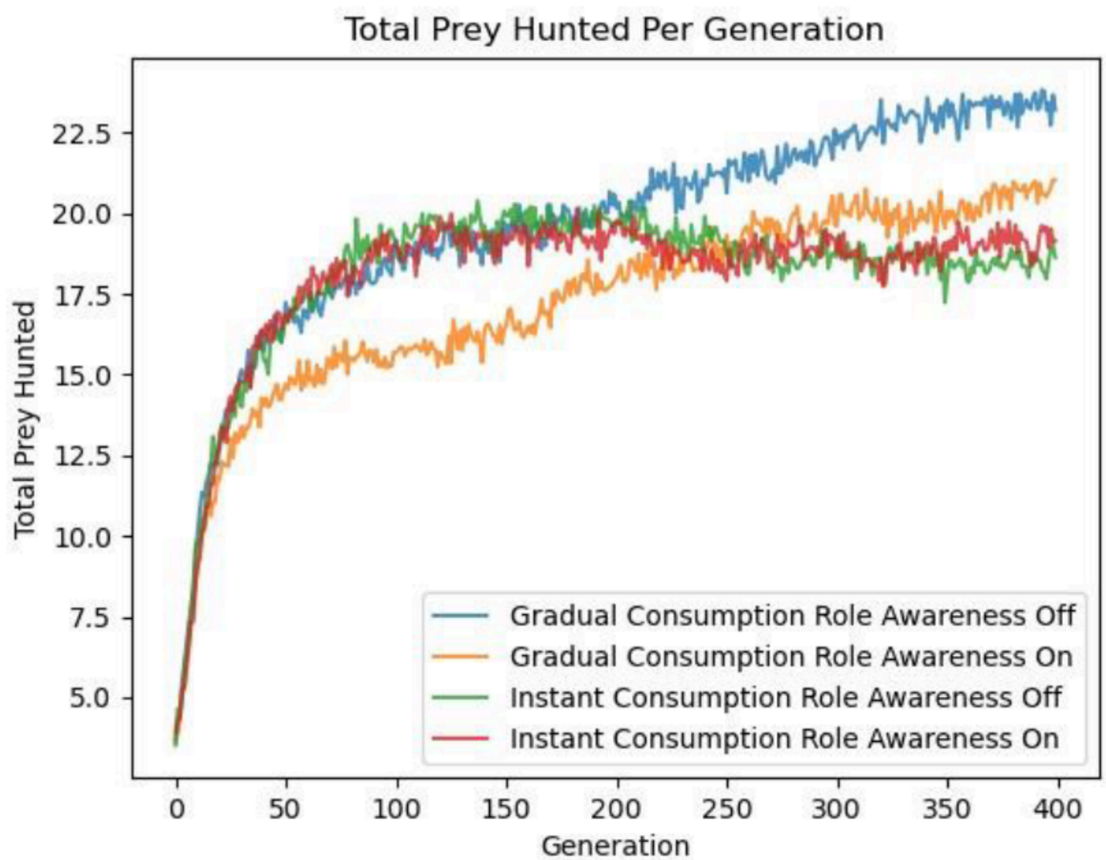
3.3.4 Gradual Consumption

In this experiment, we tested the impact of the ticks to achieve full consumption with the calories reward remaining fixed at 10 calories per tick overlapped. This will affect how often the agent will need to relocate to another area where the new food is spawned. This will be measured in calories instead of food counts. Unlike food counts, this evaluation is carried out in terms of calories. This shift provides a more comprehensive understanding of the agents' adeptness at obtaining sustenance. Details are as follows:

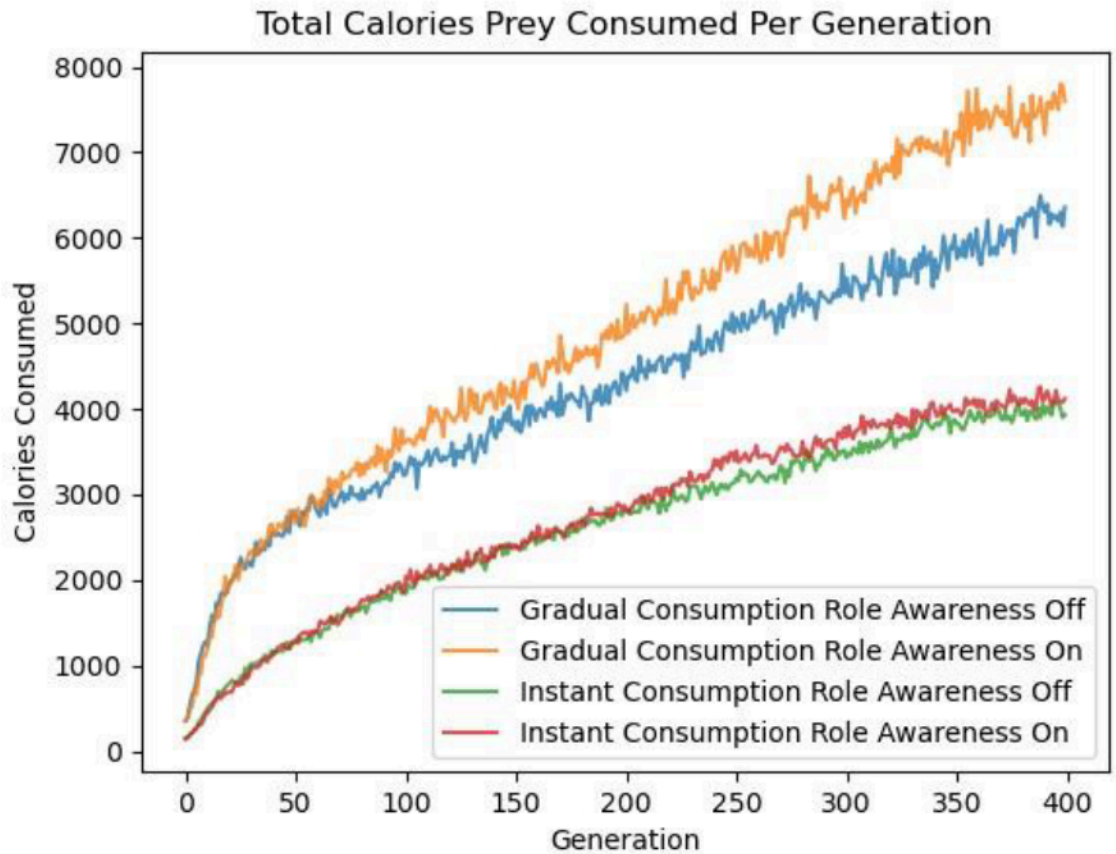
Tick to full consumption (Ticks)	Total calories per food (Calories)
1 (Instant)	10
25	250
50	500
100	1000
150	1500

4 Discussion

4.1 Role Awareness



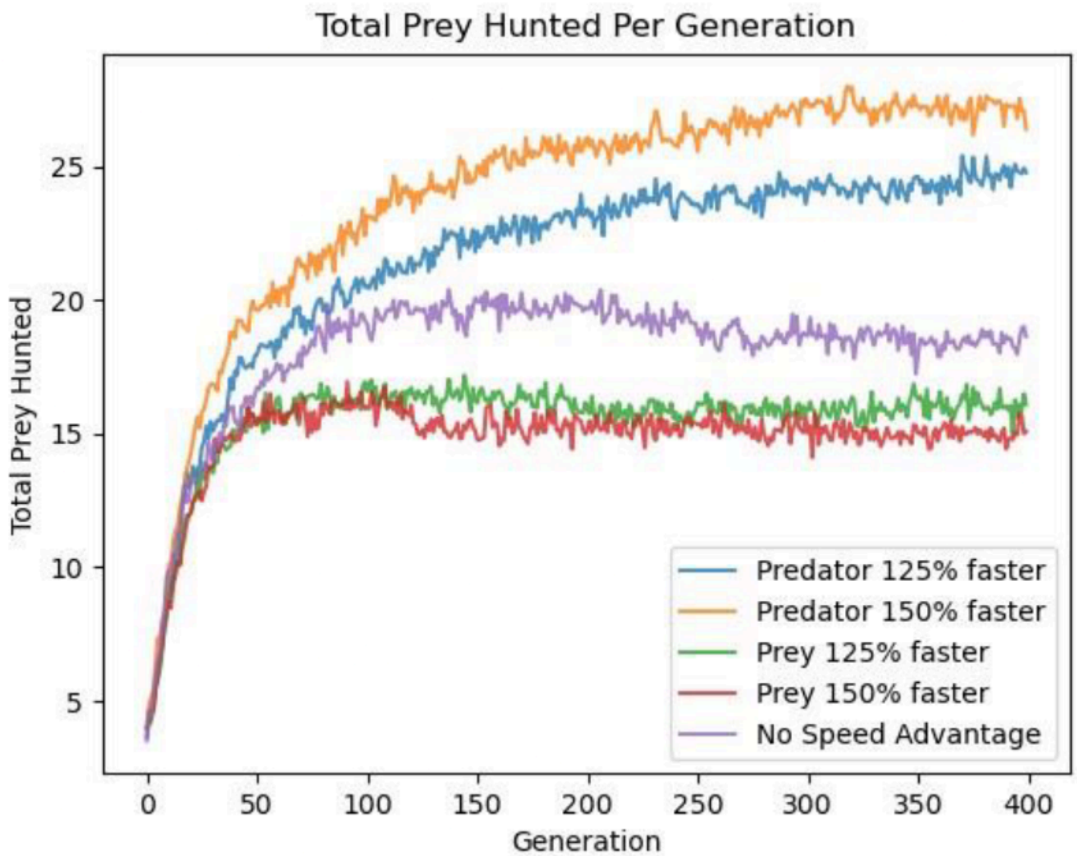
4.1.1 Role Awareness: Prey Hunted Per Generation Chart For the number of prey captured, role awareness has a negative effect for the gradual consumption profiles. While having a similar upward trend to when it was deactivated, role awareness had a consistently lower count of 5 preys per generation. While for instant consumption, role awareness initially resulted in a slightly lower count during the early and middle generations, but it was larger at the end. Introducing changes in how prey consumes food increased the difference between the agent's roles. With gradual consumption, the prey agent not only has to touch the food but remain in the vicinity while watching out for the predator. In contrast, for instant consumption, the prey's task involves simply identifying the food, moving towards it, and promptly proceeding to the next one even while being chased by the predator. It's worth noting that the role of the predator remains consistent across both profiles, which involves identifying the prey and pursuing it. It is almost identical to the prey's role in instant consumption. We theorized that the learning experience acting as prey did not translate well when the agent became a predator in gradual consumption profiles. As evidence in the total calories consumed by prey, the difference between the profiles starts to appear around generation 50. While for the number of prey hunted, it is sooner, around generation 25, showing that prey becomes better at identifying and consuming food than predator in recognizing and catching prey. It is noted that the way prey see food is identical to predator seeing prey. Overall, this phenomenon warrants a deeper investigation into the agent's neural network, particularly in the structure and firing sequences when they are either predator or prey. For instant consumption, due to the similarity in their tasks, having to specify which roles they play is redundant. Hence, role awareness offers no significant performance improvement to predators.



4.1.2 Role Awareness: Food Consumption Calories Per Generation Chart

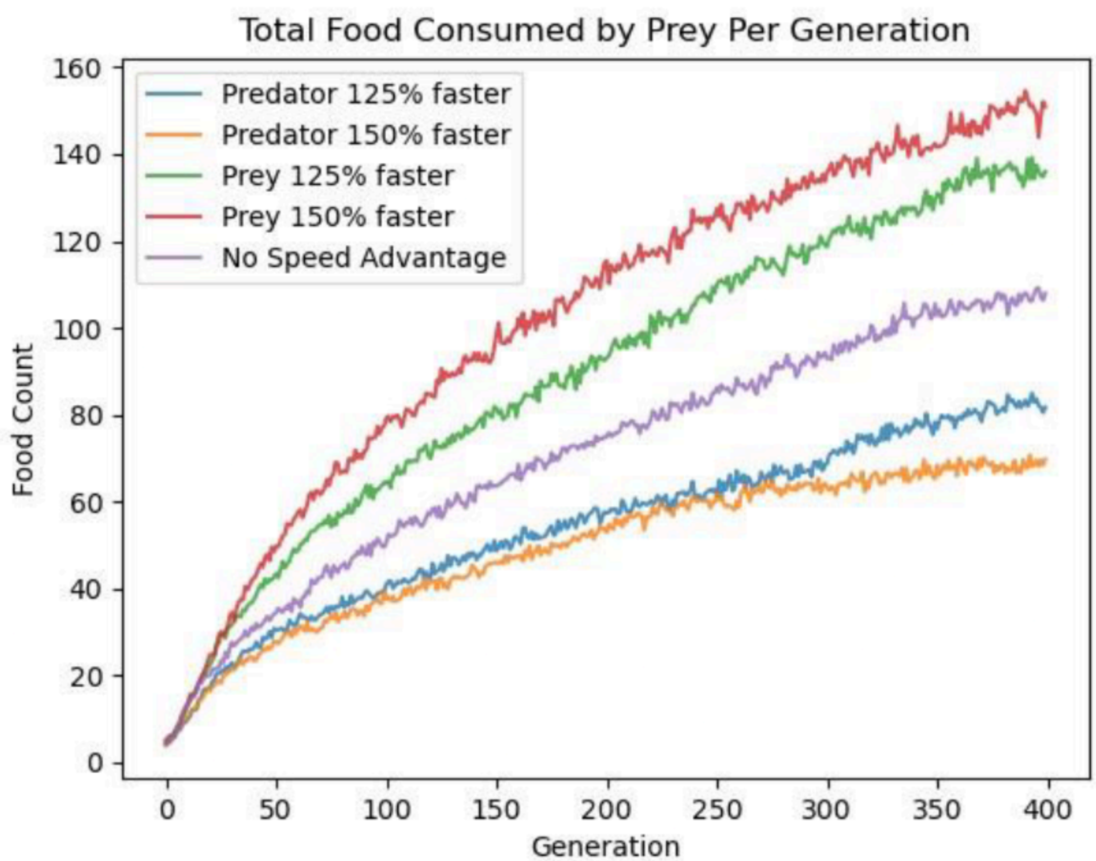
For food consumption, role awareness had a positive effect. This is particularly in the case for gradual consumption where it showed a steady increase after generation 50 compared to role awareness being disabled. We expect this result, as the presence of a role indicator helps the agent in learning the threats from predators more quickly, facilitating the evolution of countermeasures in their neural networks. This also explained the fewer number of prey being captured in the above section. For instant consumption, with the previously discussed small difference in functionality between predators and prey, role awareness exhibited a small positive impact on the agent's ability to gather food. The reason this effect is not as significant as gradual consumption might also lie in the way the agent consumes food. In instant consumption, prey must relocate frequently because food respawns in a different location upon contact. Thus, they will spend more time traveling and being more exposed to predators.

4.2 Speed Advantages



4.2.1 Speed Advantages: Prey Hunted Per Generation Chart

The speed advantage significantly influences the performance of agents, irrespective of their assigned roles. For example, the predator experiences substantial benefits when it possesses a higher speed, as demonstrated by the pronounced increase in the number of successful hunts depicted in the graphs within Section 4.2.1.2. The magnitude of this speed advantage directly correlates with the predator's proficiency in capturing prey. This is expected, due to prey lacking sufficient maneuverability to evade predators. This is further illustrated when prey became faster. Predators had less successful hunts depending on how big the advantage prey had. It's worth noting that predators display the ability to adapt to the speed disadvantage, with the number of successful hunts fluctuating in the early generations and exhibiting a small decline in the later ones.

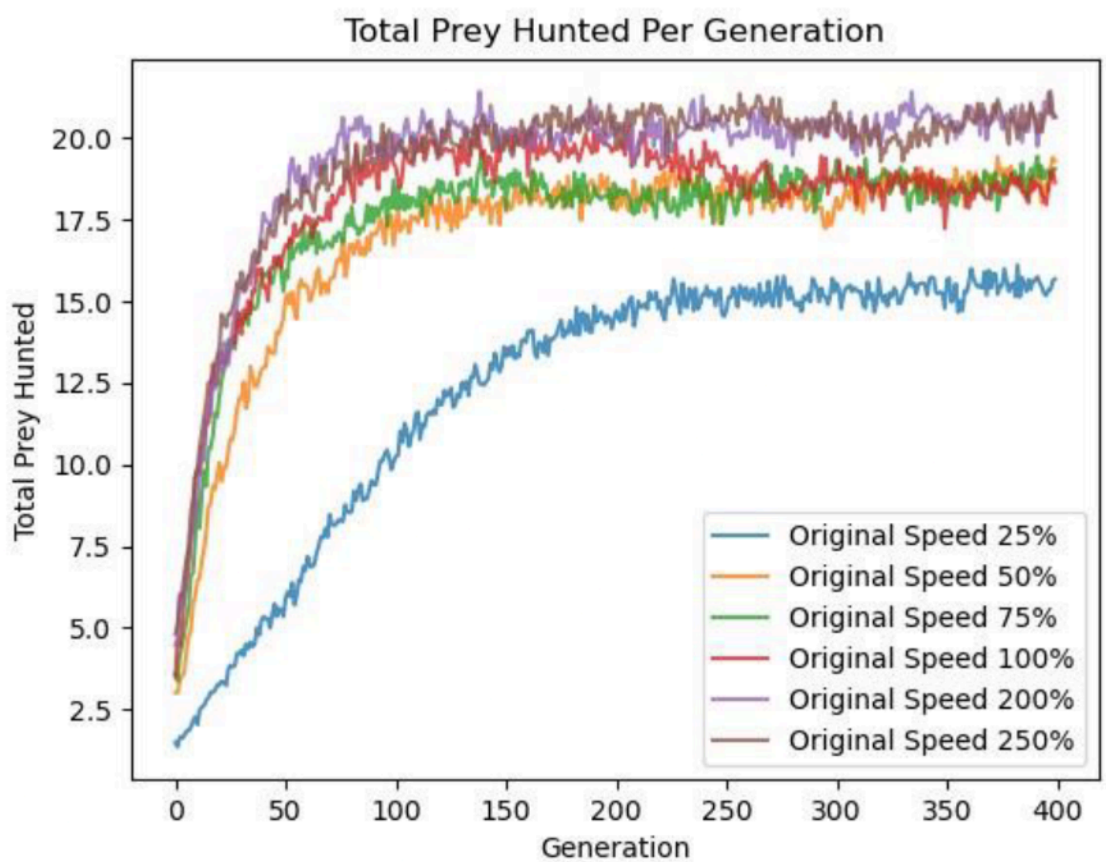


4.2.2 Speed Advantages: Food Consumption Count Per Generation Chart

In terms of the food amount acquired by prey, it is similar to predator where prey's speed advantage allows them to consume more food. This efficiency grows correspondingly to the magnitude of the speed advantage they hold over predators and vice versa, when predators were faster than prey. However, prey being faster does not benefit them fully. This can lead to some unfavorable behaviors. For example, as prey can simply outrun the predator, they tend to underestimate the threat posed by predators. They exhibited dangerous behaviors such as running directly in front of predators without attempting to evade. This leads to the number of prey being captured remains relatively high despite having the advantage.

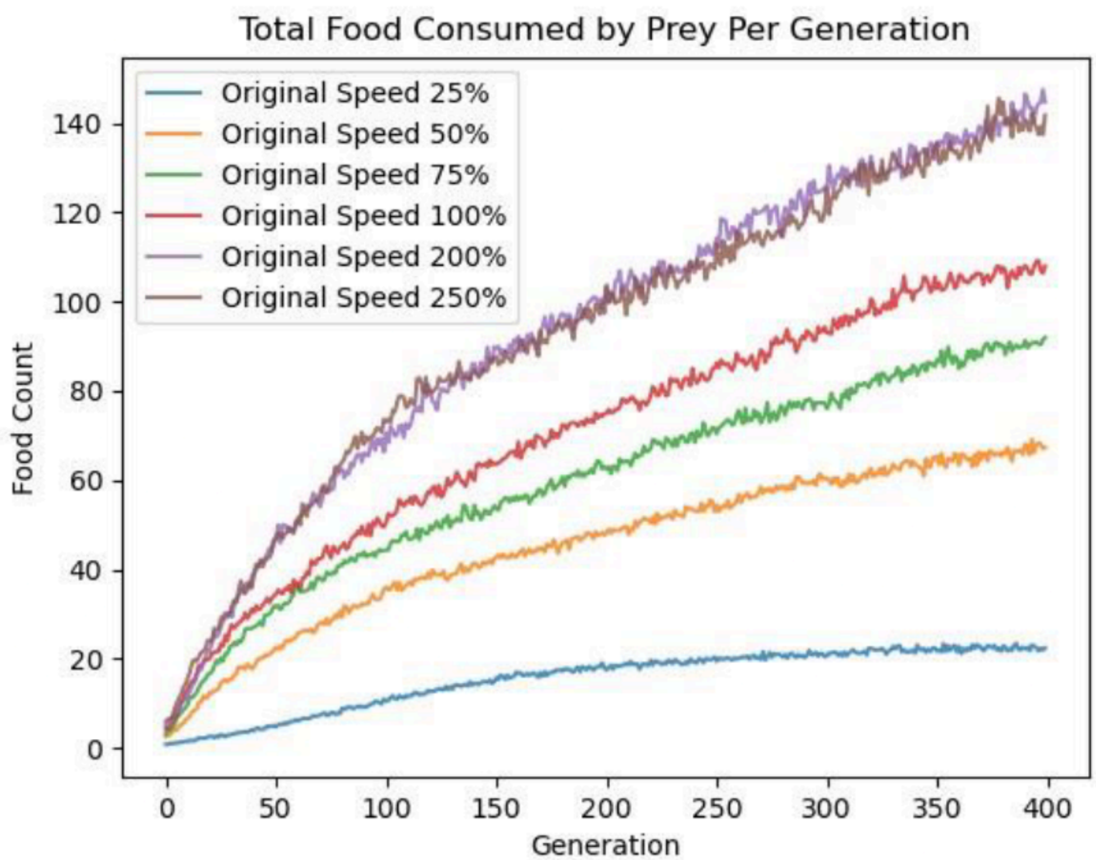
In general, we achieved the best optimal result by having no speed advantage.While favoring either predator or prey does indeed lead to improved performance, it also hinders the other entity with a speed disadvantage. As a result, maintaining equilibrium between the capabilities of predators and prey becomes crucial for achieving the best outcome over an extended duration. This equilibrium ensures that predators excel in capturing prey while prey exhibit prowess in evading and efficiently consuming food.

4.3 Overall speed



4.3.1 Overall speed: Prey Hunted Per Generation Chart

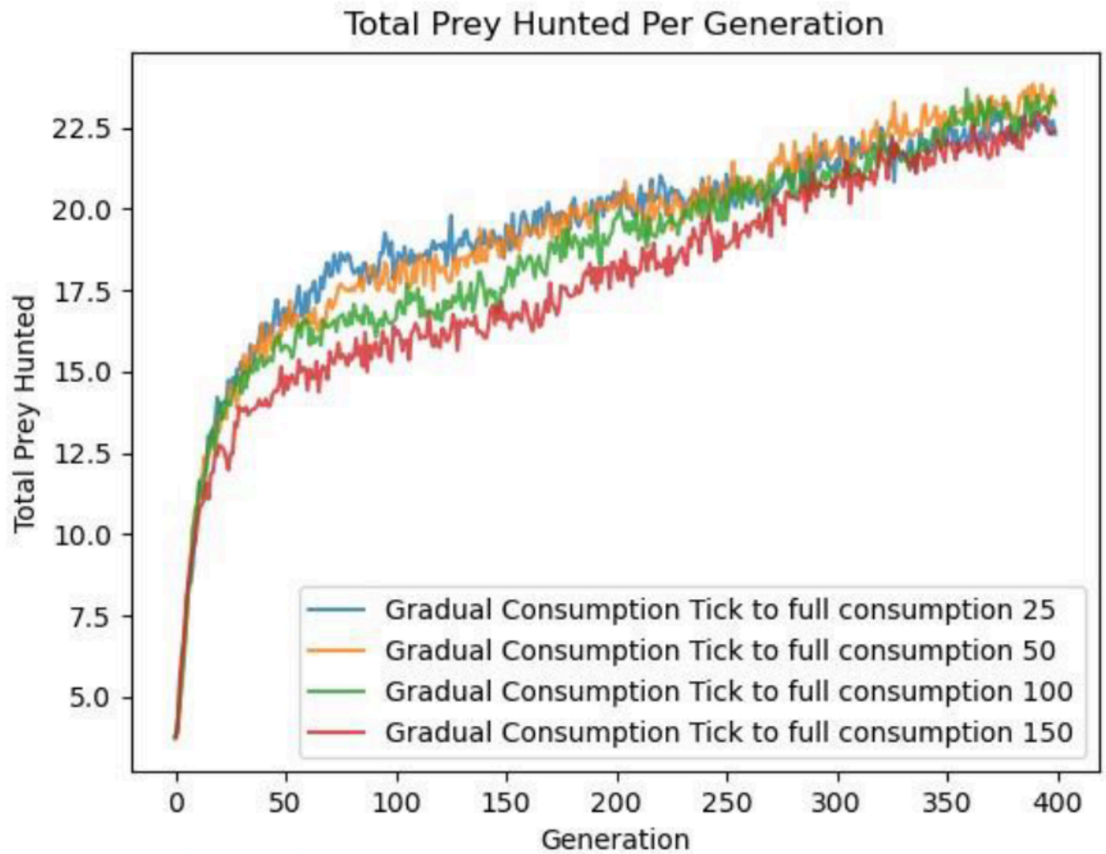
From the chart above, increasing the speed does not significantly affect the predator performance in catching prey. Modifying the agent's overall speed is similar to changing the size of the world and the relative space between each entity. Therefore, variations in speed, whether increased or decreased, have negligible impact on the predator's performance. This is due to the relative distance between the predator and its prey being consistent for every variation. An exception arises when the speed is excessively lowered, particularly when it is reduced to a quarter of the default setting. This led to more generations where the predator failed to reach the prey within the designated time frame.



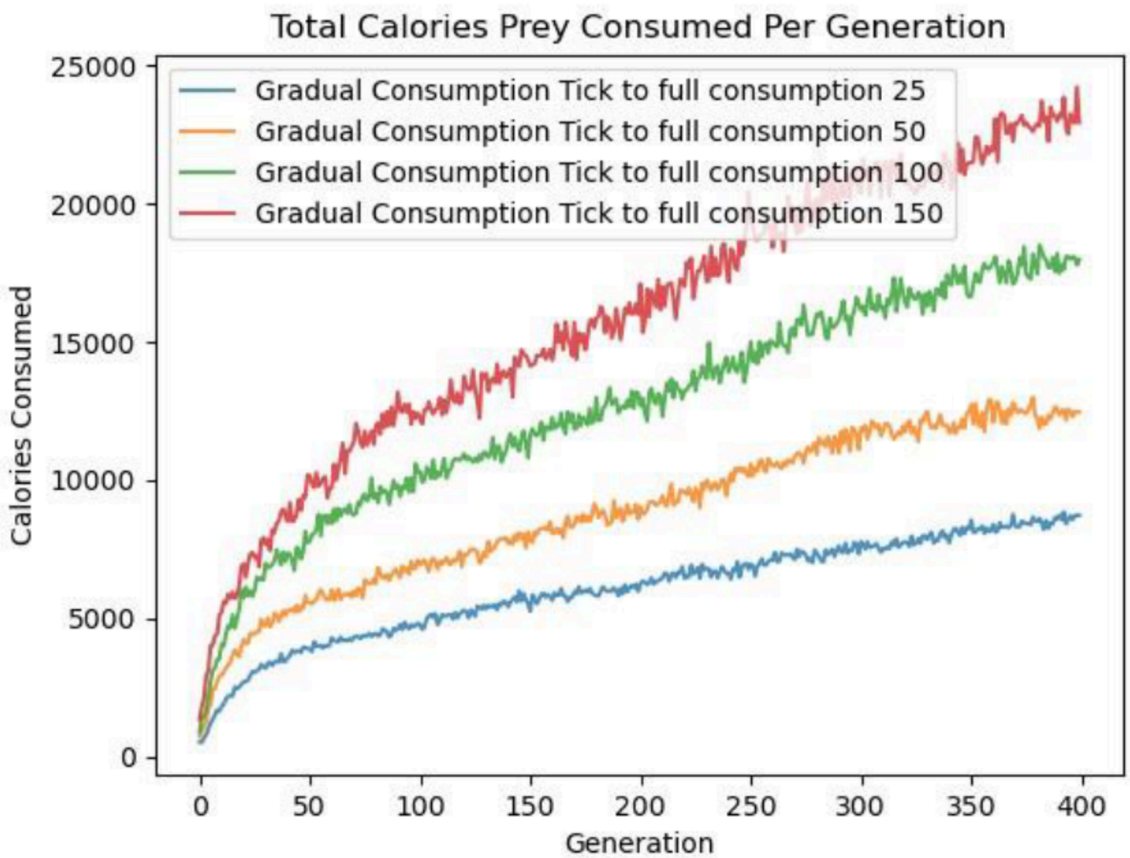
4.3.2 Overall Speed: Food Consumption Count Per Generation Chart

Same as the number of prey being captured, the amount of food consumed by prey is also correlated to how fast the agent moves. As speed decreases, the time the agent requires to reach the food becomes longer. Thus, we could see that prey consumed more food as the speed increased. However, this growth of performance slowed down when the speed increased to two times the default amount. We speculate that this is due to the agents going out of bound more in the later generations. As the agent's speed increases, its maneuverability decreases, making it more prone to crossing the boundary and getting deactivated as a result. This warrants an in depth investigation into the reason for the stagnation phenomenon. Overall, a moderate speed is the best option as exceeding the previously mentioned threshold becomes unproductive with both performance metrics no longer showing a stable increase.

4.4 Gradual Consumption



4.4.1 Gradual Consumption: Prey Hunted Per Generation Chart As observed in section 4.4.1, increasing the ticks required for full consumption does not significantly impact the predator's performance in catching prey in the long run. This increment in the requirement to fully consume the food causes prey to remain in a particular area longer. This should make them more susceptible to ambush and being cornered by the predator. However, we witnessed the opposite effect in which more prey is being captured the lower the ticks to full consumption is between generation 50 to 250. This effect wanes over the generations, failing to bring about a substantial change in the number of prey captured in the long run. This implies that predators learn quicklier when prey relocates frequently with their preference toward catching prey while it is on the move. This is because prey has picked up a behavior to hide behind the food. Thus, the longer the tick for full consumption is, the longer they can hide.



4.4.2 Gradual Consumption: Calories Consumption Per Generation Chart

For the calories consumed, it increases along with the amount of ticks for full consumption. It is worth noting that the amount of calories gained per tick remains constant. Thus the increase in the amount of calories consumed is attributed to the fact that the agent does not need to relocate as often. They can remain in an area longer until the food sources are depleted. This increment in calories consumption aligns with the common sense hypothesis that when a food source is both more abundant and accessible, it would naturally lead to a higher consumption. This further highlights the prey's performance as calorie consumption steadily increased, while the number of prey captured did not show a significant rise.

5 Conclusion

Overall, the NEAT algorithm consistently demonstrated its efficacy throughout the simulations in evolving hunting and foraging behaviors within two distinct agent roles: predator and prey, while using a shared neural network. This adaptability was observed across a number of diverse environmental settings, showcasing the agents' ability to develop target recognition, sufficient maneuvering and evading abilities.

Role awareness has a more apparent effect where there is a huge disparity between the roles. For our experiment, it has a positive impact on the prey agents' performance, while having a slightly negative effect on predators. Due to the small difference in their task for instant consumption profile, this effect is not as apparent as in gradual consumption profiles. Our hypothesis is that having a clear indicator of their roles helps prey in identifying threats coming from predators sooner. Thus, this helps them evolve countermeasures in their neural network early. While for predators, we speculated that the negative impact is due to experience gained as prey, such as target identification, do not carry over effectively. Overall, this opens a new opportunity for a thorough analysis into the agent's neural network, primarily in the structure and firing sequence in their corresponding role.

Regarding speed advantages, our study indicates that maintaining equal speeds for both predator and prey agents results in a superior overall outcome compared to scenarios where we favor one type of agent over the other. The data indicates that if either prey or predator is granted a speed advantage, the faster agent performs better in their respective roles of hunting or evading and eating food. Therefore, when evolving agents with multifunctional roles such as hunting and foraging, it's crucial to ensure a balance in terms of speed between them.

For overall speed, our simulation showed that it mostly affects the prey's performance. The relative speed of agents to each other does not change unlike in the previous experiments, so predators are not benefiting from the increase of overall speed. In fact, their performance tends to decline as both agents are more likely to go out of bounds at higher speeds. For prey, we noted a rise in food consumption as speed increases. However, this performance increase starts to level off when the speed surpasses the default value of 5 pixels per tick. Therefore, to optimize agent performance in both predator and prey roles, it's essential to select a moderate speed value that strikes a balance between avoiding going out of bounds and maximizing food consumption.

Finally, our experiments exploring the impact of adjusting the ticks required for full food consumption indicate that this change primarily benefits prey agents with the predator's performance not increasing despite having prey in a certain area longer. When the number of ticks needed for full consumption remains low, predators demonstrate improved hunting performance in the early generations due to prey having fewer time hiding behind food. However, this advantage levels off in later generations, suggesting that prey agents also adapt to evade predators over time using other means. The prey's performance in terms of consuming stationary food increases the higher the amount of food is. This confirms the common sense hypothesis as preys do not have to relocate frequently.

6 References

Bryan, Gabe and Toan Nguyen "Gabe-Bryan/NEAT-Hunting-and-Gathering." GitHub, 17 June. 2023 https://github.com/Gabe-Bryan/NEAT-Hunting-and-Gathering

Potafiy, Artem and Parker Rosengreen. “Apotafiy/NEAT-Research-Tcss499.” GitHub, 28 Sept. 2022, https://github.com/apotafiy/NEAT-research-tcss499

Yong, C. H. and R. Miikkulainen. “Coevolution of Role-Based Cooperation in Multiagent Systems.” IEEE Transactions on Autonomous Mental Development, vol. 1, no. 3, Oct. 2009, pp. 170–86, https://doi.org/10.1109/tamd.2009.2037732.

Stanley, K. O., and R. Miikkulainen. “Competitive Coevolution through Evolutionary Complexification.” Journal of Artificial Intelligence Research, vol. 21, Feb. 2004, pp. 63–100, https://doi.org/10.1613/jair.1338.

Papavasileiou, Evgenia, et al. “A Systematic Literature Review of the Successors of “NeuroEvolution of Augmenting Topologies.”” Evolutionary Computation, 5 Nov. 2020, pp. 1–73, 10.1162/evco_a_00282. Accessed 20 Dec. 2020.

Stanley, K., & Miikkulainen, R. (2002). Evolving Neural Networks Through Augmenting Topologies. Evolutionary Computation, 10(2), 99-127.