eSense 2.0: Modeling Multi-Agent Biomimetic Predation with Multi-layered Reinforcement Learning

D. Michael Franklin
Kennesaw State University, Marietta Campus
Marietta, GA

Derek Martin
Kennesaw State University, Marietta Campus
Marietta, GA

Abstract

Building on the success of the eSense BioMimetic modeling done in (Franklin and Martin 2016), eSense 2.0 expands the modeling to include a stronger predator / prey relationship. eSense provides a powerful yet simplistic reinforcement learning algorithm that employs model-based behavior across multiple learning layers. These independent layers split the learning objectives across multiple layers, avoiding the learning-confusion common in many multi-agent systems. The new eSense 2.0 increases the number of layers and the amount of separation between agents so that the behaviors for each agent can be more highly customized and adds specific additional layers for behavior-only learning. In other words, each agent now has multiple layers to model each aspect of their behavior (e.g., obstacle avoidance, prey observation, prey seeking, etc.). This new abstraction of breaking out the various agent behaviors into multiple levels furthers speeds up the learning and clarifies the objectives the agent is considering. This significantly builds on the general goal of eSense (splitting out multiple agents into their own levels) because now the agent's behaviors are also split out into multiple layers. The learning is now more expressive, faster, and less noisy. This papers seeks to present this new multi-level learning system for multi-agent systems and confirm its performance through experimentation.

Introduction

Real-world artificial intelligence and learning is often made more difficult by the various goals that each agent has and the complex interactions between agents within a system. This is especially true in multi-agent systems where the desired interactions take on a strategic, intelligent meaning. The normal approach of using monolithic policies is rendered ineffective because of having multiple behaviors for each agent, some of which frequently conflict, and exacerbated by the multiple teams of agents which are in direct conflict. In this case, the system being modeled in the simulation is the predator / prey dynamic. To do so, each agent is considered a biomimetic model of a sensing agent. This means that each agent has its own 'personality' - a methodology of movement, a set of goals to seek, other agents to avoid, etc. Further, each agent has a sensing grid - this grid is how the agent sees the world around them. This might be modeled as passive echolocation, active electrolocation, or any of several sensing models previously presented in the original eSense paper (Franklin and Martin 2016). Expanding on that, there are additional sensing layers that are seeking and avoiding goals and other agents, depending on the model in force. Additionally, the new model includes multiple biomimetic models interacting within the environment.

eSense 2.0 takes the single-agent success and builds on it to add multiple layers of perception. This increases the performance of each agent and increases the level of biomimicry. eSense 2.0 also places these much more capable agents into multi-agent scenarios and allows interactions on multiple levels. The experiments will show the increased expressivity, higher fidelity modeling, and the multi-agent interaction capability of the system.

Related Works

In (Hussein 2010) the authors have presented a mathematical analysis on predator / prey relationships and their interactions within shared environments. This interaction can be understood in terms of the pressure each set of animals places on each other from their presence as relates to their distinct populations and amount of shared environment. This work was helpful in modeling these interactions within eSense and understanding a mathematical modeling of these relationships.

The interaction of predators and prey, especially understanding the balance of their populations and level of their interactions within the same ecosystem, is modeled thoroughly in (Freedman and Waltman 1984). This work, though older, is still cited as a reference on the concept of persistence (where persistence is defined as a greater than zero population now and at the limit). This work contributed the idea of balance of populations and the concept of persistence to this research. It is important to note that the referenced paper works in a deterministic environment, so the work proposed herein expands beyond this consideration into both non-deterministic and stochastic space.

One important intuition that was confirmed in (Lima 2002) was that of understanding the difference between isolating predators from prey and analyzing them separately versus studying and understanding them in the proper real-world context. This more accurate context allows for coordinated and reactive strategic behavior from the predators while tracking the prey. There are also two additional works inside this paper that elucidate the complexities of these interactions more clearly. They note first that the multi-trophic games of habitat choice impacts this kind of real world interaction significantly. Secondly, they note that the scale of these interactions matter. While this is not surprising, it is important to have confirmed empirically. This research proposed and confirmed in this paper understands both of these ideas and seeks to model them appropriately (i.e., with the correct level of expressivity and complexity to allow this kind of large-scale strategic interaction).

In (Yi, Wei, and Shi 2009), there is a lengthy treatise on the complexities that arise from multiple populations of predators and prey existing in the same interactive environment. The exhaustive analysis of the Hopf bifurcation and multiple steady-state bifurcations is especially informative in understanding the pressure concepts of multiple agents occupying a competitive space. For the work contained in this research, this paper offered a thorough mathematical analysis of competitive-cycle dynamics of these interactions that are insightful and illuminating. This research builds on the premises and expands them to stochastic and non-deterministic scenarios like those found in the experiments conducted in the simulations to prove the eSense 2.0 expanded models.

There were many models used in the previous work that were now revisited for information about the expanded biomimetic modeling done in eSense 2.0. (Ammari and Garnier 2013) offers a treatise on the modeling of electric fish for experimentation and simulation in multi-agent scenarios. These models were expanded by comparing this previous work with (Boyer 2012). This additional reference provided more detail on the types of models available and insight into passive and active models for both types of target location. Additionally, (Hopkins 2005) offers specific information for modeling passive electrolocation and understanding how it is used in the real world. This work was essential to ground the symbols for each experiment and to ensure experimental veracity. The earlier models and final experimental models were enhanced by using the information from (Shieh 1996). Each of these contributed to the modeling of the fields utilized by both the active and the passive location models.

The integration of the various models and bringing them from real-world, biological models, into simulated entities within a reinforcement learning environment was aided significantly by the work in (Coggan 2008). This work offers some background insight into how others have approached these types of learning environment models. In particular, this work studies exploration and exploitation in reinforcement learning, and, vitally, the balance needed for them both to be effective. This work helped verify the need for utilizing the ϵ-greedy approach utilized by this research. In the proposed work the exploration and exploitation are balanced progressively during the execution of the algorithm. Initially, with ϵ high, the algorithm leans towards utilizing more exploration to explore the state-space more thoroughly. Eventually, over time, as ϵ decreases, the algorithm utilizes more exploitation. While there were no other papers found that have applied a similar model as the one proposed herein, this paper did at least offer insight into the validity of the approach.

The concepts of Reinforcement Learning, of which both Temporal Difference learning and SARSA-λ are examples, were described in (Woergoetter and Porr 2008) and (Taylor and Stone 2006). These works were utilized to confirm the models used for the learning systems for these experiments and to gain insight into common settings for both approaches. While these referenced works propose and define various aspects of reinforcement learning, they do not propose anything similar to the multi-layered dynamic approach described in this research. Further, they allude to why this particular goal, a dynamic reward, is a non-starter for reinforcement learning (that is, convergence is statistically improbable).

Methodology

This new eSense modeling builds on the previous work, referenced in the abstract, and expands on it significantly. In the previous work there were a number of innovations that led to the overall success of eSense, and those will be summarized here for clarity. For complete background, please review (Franklin and Martin 2016).

Originally, eSense innovated the idea of taking the simple-yet-powerful reinforcement learning technique SARSA-λ and utilizing it in creative ways to accomplish complicated learning. In particular, it is well known from (Sutton and Barto 1998) that SARSA-λ is a clean, efficient minimal information learning technique for reinforcement learning, but it does not work if the goal is moving. A moving goal essentially erases all learning in the Q-table because of the history contained within the grid. For clarity, assume, $w.l.o.g.$, that the learning in the SARSA-λ is Q-learning, and this learning uses two distinct tables for tracking the progress of the learning. Standard SARSA expands the typical Q-learning into a Q(s, a) table that stores the values of taking any move from the current state (i.e., Q holds values of each a for and given s). This Q(s, a) table is then consulted any time the agent is preparing to move to select the next action based on one of two options. In standard ϵ-decay technique, the next move is selected pseudo-randomly with ϵ probability and by max value with $(1 - \epsilon)$ probability. This helps the agent to explore more often early on and exploit the learned data more frequently as the learning progresses. The second table utilized in the SARSA-λ variant is the e-table. This second table holds a type of memory of states visited since the epoch has begun and allows for a longer history of updates to the Q(s, a) table to be made each iteration. Typically only the previous states would be updated, but a decaying reward can be effectively propagated back along the entire path of moves by utilizing the e-table date. This method of updating in a typical grid world example means that as long as the goal is stationary, the Q(s, a) table will eventually hold a policy that offers the best move from any given state, thus achieving a minimal pathing from the origin point to the goal. However, as mentioned, if the goal were moved each epoch, not only would the Q(s, a) table no longer lead to the goal, it would actively lead away from it. If, for example, the goal were moved only once, then the learning could eventually repair the table to point towards the new goal location, but it would take much longer than it did the first time because of its bias to the old goal location.

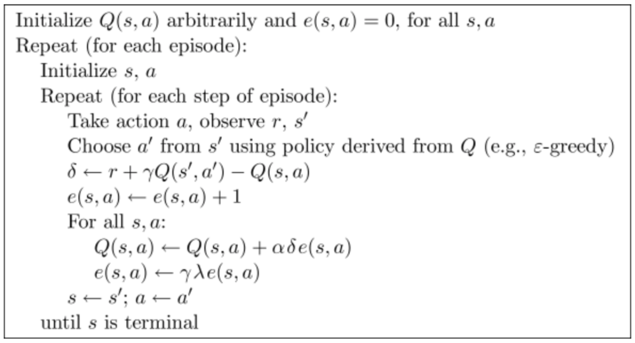
To overcome this, the eSense technique was devised. As presented in the reference paper, this limitation was removed through layering the behaviors into multiple grids. eSense works on multiple levels of learning through the use of a master grid for obstacle detection and avoidance and another layer for sensing (i.e., examining what is around the agent and reacting to that rather than using the master grid). This means that the agent can wander around the master grid and learn obstacle avoidance without worrying about goals. The sensing layer can then be utilized for goal-seeking. The sensing layer is homeostatic, centered around the agent. As the agent searches the grid, using learned data in the master grid to avoid obstacles, the goal eventually moves within the sensing grid. This triggers the learning in the sensing grid to react to the presence of the goal. The key intuition within this technique is that the learning is identical, but the action set is reversed (this can be thought of as moving the goal towards the agent - which is not possible, but it informs the agent's direction of movement). This was a key innovation of the eSense methodology.

Once this technique was proven, eSense went even farther by allowing for a moving target. In the original formulation the goal was stationary, just placed randomly around the grid. In the final formulation, the moving goal became a prey and the agent became a predator. The sensing grids were converted to reflect the various biomimetic models (both passive and active echolocation and electrolocation). This means that the sensing grids had differing sizes and shapes. As the predator searched the master grid the prey would follow a pattern of movement dictated by the program. When the prey entered the predator's sensing grid the predator would react to move towards the prey in an effort to catch it. To be clear, this was not programmed behavior - the predator learned these behaviors from scratch using the simplistic SARSA-λ reinforcement learning without any prior knowledge. This is a significant outcome and the novel contribution of the original eSense paper.

eSense 2.0 expands significantly on this multi-layered learning methodology to include even more layers with additional agents in the system. First, the prey is now an agent. The prey has its own obstacle avoidance master grid that is learning to avoid edges, obstacles, and other obstructions. The prey also has another master grid that is marking locations where food has been found (the food is the prey's goal and can be located at any number of stochastic locations around the grid). Additionally, the prey has two sensing grids. The first sensing grid is designed to detect and react to food. When food appears on this sensing grid, the agent learns through trial and error to seek after the food. The second sensing layer is the predator avoidance layer. This sensing layer detects when predators are within range and learns to avoid them (this is the same learning technique, but with the opposite set of actions). This multi-modal learning is difficult for traditional agents because trying to learn a large, complex monolithic policy is both contradictory (learning to move towards and away from goal objects) and confusing (clouding up the learning with contrary goals and opposing actions) (Franklin 2015). The new multi-layered approach presented in eSense 2.0 allows for less complex learning techniques with single-goal objectives, thus overcoming this learning complexity and confusion. Second, the predator also has the multi-layered approach. As with the prey, the predator has its own obstacle avoidance layer. This could be the same master layer as the prey, but by giving each agent their own obstacle avoidance layer each agent's size can be considered independently. For example, a smaller prey can slip through a smaller opening in the obstacle field that a predator cannot. This individualized behavior is an important part of the eSense methodology. Further, the predator also has an additional master layer to track the most likely places to find prey as well as two sensing layers. The first sensing layer is seeking prey (its food source) while the second is learning to avoid other predators. This configuration allows for an entire hierarchical ecosystem of predators and prey, as well as allowing for multiple agents within each layer.

Each layering the agent model is performing SARSA-λ, though with different ranges and setups. Each layer is learning according to the update function shown in Equation 1. This updates the Q(s, a) table by utilizing the reward r for moving to the next state, the next values provided from taking the chosen next action (a') from the next state (s') (stored as $Q(s', a')$). It is mitigated by the learning rate, α. The algorithm for the updates and the movement tracking history is shown in Figure 1. This shows the step by step updates shown in Equation 2. The update amount, the δ, is calculated in Equation 3. The e-table is incremented for every space that is visited, according to Equation 4. The decaying updates in the e-table are updated according to Equation 4 using the discount rate γ and the decay rate λ. This results in an eligibility trace (a history of decaying rewards based on the previously visited, and, thus, eligible spaces that can receive an update / reward). These traces are similar to those shown in Figure 2.

$$Q(s,a) = Q(s,a) + \alpha(r(s',a') + \gamma Q(s',a') - Q(s,a)) \quad (1)$$



```
Initialize $Q(s,a)$ arbitrarily and $e(s,a) = 0$, for all $s, a$
Repeat (for each episode):
    Initialize $s$, $a$
    Repeat (for each step of episode):
        Take action $a$, observe $r$, $s'$
        Choose $a'$ from $s'$ using policy derived from $Q$ (e.g., $\varepsilon$-greedy)
        $\delta \leftarrow r + \gamma Q(s',a') - Q(s,a)$
        $e(s,a) \leftarrow e(s,a) + 1$
        For all $s, a$:
            $Q(s,a) \leftarrow Q(s,a) + \alpha\delta e(s,a)$
            $e(s,a) \leftarrow \gamma\lambda e(s,a)$
        $s \leftarrow s'$; $a \leftarrow a'$
    until $s$ is terminal
```

Figure 1: SARSA-λ Algorithm (Sutton and Barto 1998)

$$Q(s,a) = Q(s,a) + \alpha\delta e(s,a) \quad (2)$$

$$\delta = r(s', a') + \gamma Q(s', a') - Q(s, a) \tag{3}$$

$$e(s, a) = e(s, a) + 1 \tag{4}$$

$$e(s, a) = \gamma\lambda e(s, a) \tag{5}$$

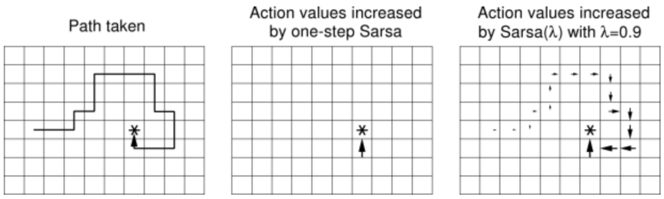

Path taken
Action values increased by one-step Sarsa
Action values increased by Sarsa(λ) with λ=0.9

Figure 2: SARSA-λ Eligibility Traces (Sutton and Barto 1998)

As can be seen from this formulation, each layer described above is actually composed of multiple layers (the Q(s, a) table and the e-table). This means that each layer can be learning on its own independently. Each of these layers is small enough to learn quickly and is focused on only one aspect of the agent's behavior, so the monolithic policy can be avoided and replaced with smaller policies customized for each layer. This arrangement of layers means that there must be one additional layer, the agent layer, that controls the focus of the agent across these multiple layers. The layers are arranged in a hierarchy, as shown in Figure 3. The agent layer sits at the bottom of the hierarchy and organizes the behavior of each agent by receiving the fusion of the sensor layers. For example, the prey agent layer is constantly running the baseline obstacle avoid layer (meaning that it considers all higher actions with respect to the base action of avoiding obstacles). Additionally, it is adding information to its food location layer each time it finds food. Of course, when food is sensed on the food sensing layer (generically, the goal seeking layer), it reacts to pursue that food. Finally, the highest priority layer is the predator avoid layer. This means that the agent is constantly wandering the master grid avoiding obstacles and seeking food. When it finds food, it notes that location and seeks after it. Suppose a predator is sensed - now the agent layer shifts priorities to moving away from the predator, but considers all of the lower actions. In other words, it will move away from the predator, but towards food, all while avoiding obstacles. It also notes what it learns, (e.g., where it sensed predators or food, both stored on their own history layers). Again, to reiterate, this behavior is not programmed in - the agent is given no information ahead of time other than that food receives a positive reward and dying a negative reward. The agents are learning from scratch with no other information than the rewards given. Each layer is able to adapt and learn quickly because the layers are separated as described.

The information from each layer is fused into a best action for the agent in a vector fashion and stored on the agent layer. As can be seen in Figure 4, the input from each layer is laid out on the graph with both a direction and a magnitude.

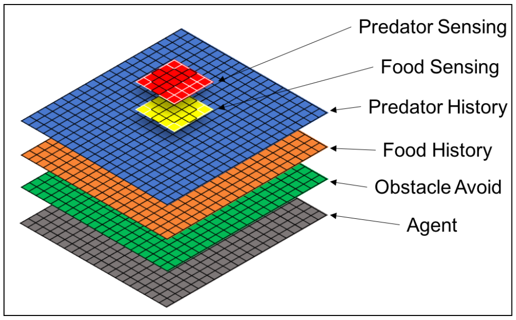


Figure 3: eSense Layers

The magnitude is the weighting for each layer, and this can be provided as a model (in the case of biomimicry) or it can be learned over time by experience. The sum total of each layer's input is the resultant vector on the agent layer, shown in Figure 5. The resultant vector is discretized into the action that most closely matches the intention of the resultant vector and the action is chosen. When the weighting is correct, the optimal performance of layer-fusion is obtained and the behavior maximizes the most important choices while being mindful of all choices. This can also be thought of as maintaining a primary goal (e.g., survival), while operating on the sub-goals (e.g., feeding). This complicated, modeled behavior is being achieved with several simple layers rather than with large, monolithic and unwieldy layers that would take a long time to learn and be difficult to adapt over time.

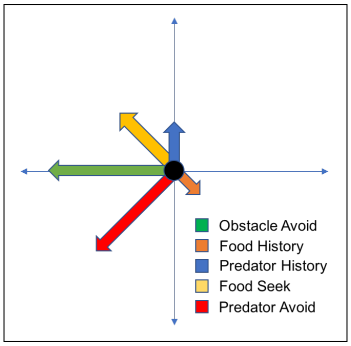


Figure 4: Sensing Layer Fusion

The smaller sensing layers are shaped in accordance to the biomimetic models upon which they are based (e.g., electrolocation or echolocation) as well as the modality of sensing (e.g., active or passive). Two of these shaped sensing layers are shown in Figure 6. The sensing grids are homeostatic, meaning that they stay centered on the agent (whose location is marked within each of these grids). These grids can be shaped to model any reasonable type of sensing array, or, more generally, to resemble any type of goal-seeking apparatus. In any case, the sensing layers are fused to the agent so that all available layers can send their data to the agent layer for processing.

Once a goal condition is encountered within a sensing grid (e.g., a food source, a predator, etc.) the agent layer can

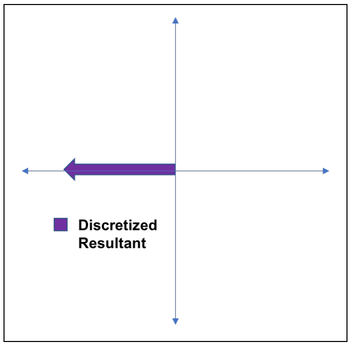


Figure 5: Agent Layer Resultant

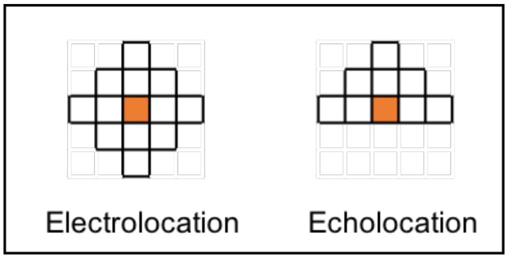


Figure 6: Sensing Grids (Agent location noted)

then process the appropriate action to maneuver the agent towards or away from the goal. As stated previously, the sensing layers work in reverse because the agent cannot move the goal, so the appropriate action is considered as if the goal were movable, then the reciprocal action is taken. For example, if food were detected, the agent would want to move the food towards it, but it cannot. Instead, it takes the inverse action and effectively moves the sensing grid towards the goal. The learned action becomes to move towards food and away from predators, or, more generally, towards or away from goals.

Experiments

In order to test these hypotheses, there were a number of experiments conducted, and they will be noted in this section. The first was to set the prey agent in action to see how well it could learn to: 1) avoid obstacles; 2) find food; 3) learn likely food locations; 4) adjust its wandering pattern in response to likely food locations. This first experiment was successful. The prey agent learned quickly to avoid obstacles efficiently using the prescribed reinforcement learning algorithm (meaning that it started with no information other than the goal rewards, when encountered). Figure 7 shows this for both the prey and the predator. It also learned to locate food and move towards it, though this learning took a bit longer because there are multiple goals (meaning that the agent had to wander enough to discover the other food locations). Figure 8 shows the prey's success at finding food, increasing over time, versus it being caught by the predator, slightly increasing over time. Once this behavior was learned, the wandering pattern of the agent became more centralized near food sources, as was hypothesized.

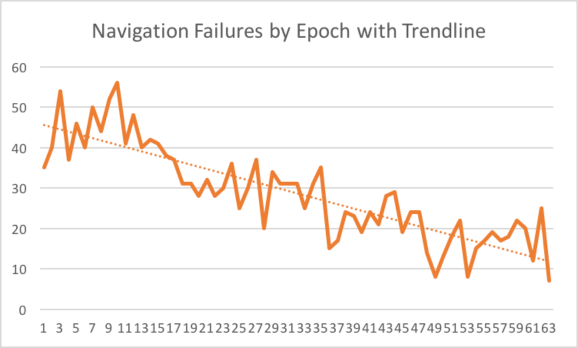


Figure 7: Navigation Failures by Agents

The second experiment was akin to the first, but with the addition of a predator. The predator was simultaneously learning obstacle avoidance, food locations (i.e., the most likely locations of the prey), and predator avoidance. Of course, the introduction of a predator into the environment activated the prey's predator avoidance layer. This resulted in a successful migratory pattern for the predator who learned to localize on the prey's food sources. It also slightly modified the prey's routine to learn to avoid the most likely predator locations, though this learning took longer. Figure 9 shows the average number of moves per epoch for both agents. In the end, the experiments proved successful in modeling a biomimetically accurate predator and prey relationship.

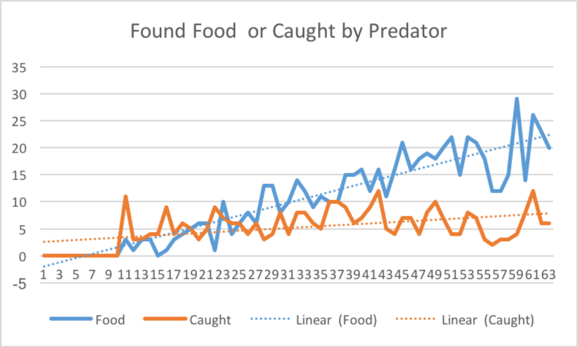


Figure 8: Prey: Found Food vs. Caught by Predator

The third experiment built on the second experiment by introducing multiple prey into the environment. While this still followed the predictable results (the predator now learned a more general migratory pattern to adjust to the multiple locations where prey can be found), it was only a stepping stone to multiple predators and multiple prey. This finalized the progression of the experiments and showed that the prey can learn to distribute themselves across the food sources, predators can spread out to maximize available prey to each, and both prey and predators can avoid their own kind. Figure 10 shows that the addition of multiple agents show did not affect performance, and is thus scalable (the trend line is nearly identical to the single agent average moves graph).

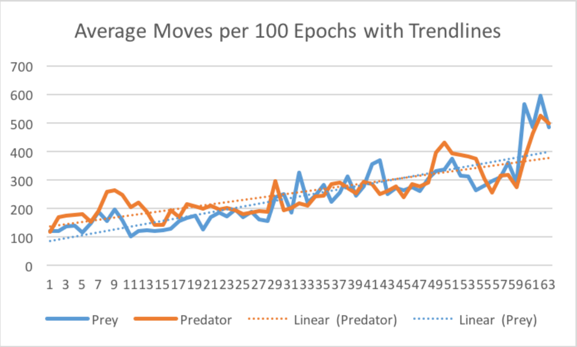


Figure 9: Average Moves without Failure

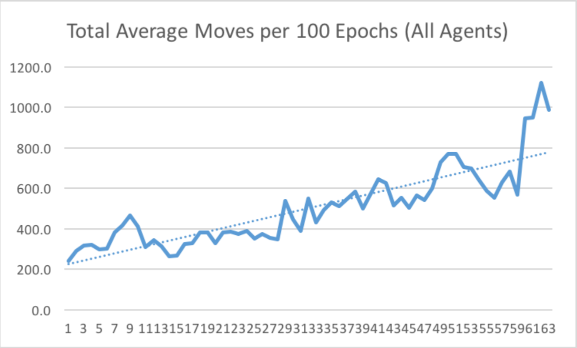


Figure 10: Total Avg Moves without Failure (All Agents)

Analysis

The experiments proved the efficacy and efficiency of the multi-layered predator / prey biomimetic modeling. Further, they verified that complex, intricate, and multi-agent behavior can be learned through even simple reinforcement learning as long as the various behavioral elements are spread across multiple coordinated layers. By keeping each layer simple and focused, the agents were able to learn multiple goals at one time (e.g., finding food sources while avoiding prey and obstacles) without a significant increase in training time. The biomimetic models were further expanded to include multiple behaviors, though this can be expanded in the future. There was a lot of experimentation with the size and shape of the sensing grids and how that impacted the learning, but it was discovered that while these helped demonstrate different models they had no significant impact on learning rates.

Conclusions and Future Work

This work has shown that biomimetic modeling can be realized through simple, multi-layered learning techniques. Additionally, the experiments verified that multi-agent interaction, even with teams of agents, works well without significantly slowing down the learning. It should be noted that the introduction of more prey or more predators once the learning has advanced my cause disruption and instability, but the learning can adapt. This will be tested in greater detail in future work. Finally, in future work the hypotheses will be expanded to include a larger food chain (where predators have predators). Also, there is the hope to introduce group behavior versus lone wolf behavior to see if this can be modeled effectively and, if so, what impact it has on learning.

In conclusion, this work has shown tremendous promise from its simple beginnings and has become more robust through expansion. It is the author's hope that this will continue to be true with further expanded experimentation and more complex modeling.

References

Ammari, Habib, T. B., and Garnier, J. 2013. Modeling active electrolocation in weakly electric fish. *SIAM Journal on Imaging Sciences* 6(1):285–321.

Boyer, Frdric, e. a. 2012. Model for a sensor inspired by electric fish. *IEEE Transactions on Robotics* 28(2):492–505.

Coggan, M. 2008. Exploration and exploitation in reinforcement learning. *CRA-W DMP Project at McGill University, Scholarpedia* 3(3):1448.

Franklin, D. M., and Martin, D. 2016. eSense: BioMimetic Modeling of Echolocation and Electrolocation using Homeostatic Dual-Layered Reinforcement Learning. *Proceedings of the ACM SE 2016*.

Franklin, D. M. 2015. Strategy Inference in Stochastic Games Using Belief Networks Comprised of Probabilistic Graphical Models. *Proceedings of FLAIRS*.

Freedman, H., and Waltman, P. 1984. Persistence in models of three interacting predator-prey populations. *Mathematical Biosciences* 68(2):213–231.

Hopkins, C. D. 2005. *Electroreception: Passive electrolocation and the sensory guidance of oriented behavior*. New York: Springer.

Hussein, S. 2010. Predator-Prey Modeling. *Undergraduate Journal of Mathematical Modeling: One+ Two* 3(1):32.

Lima, S. L. 2002. Putting predators back into behavioral predator–prey interactions. *Trends in Ecology & Evolution* 17(2):70–75.

Shieh, K. T., e. a. 1996. Short-range orientation in electric fish: an experimental study of passive electrolocation. *Journal of Experimental Biology* 199(11):2383–2393.

Sutton, R. S., and Barto, A. G. 1998. Reinforcement Learning: An Introduction. Chp. 4, 5, 8.

Taylor, Matthew E., S. W., and Stone, P. 2006. Comparing evolutionary and temporal difference methods in a reinforcement learning domain. In *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*. ACM.

Woergoetter, F., and Porr, B. 2008. Reinforcement learning. *Scholarpedia* 3(3):1448.

Yi, F.; Wei, J.; and Shi, J. 2009. Bifurcation and spatiotemporal patterns in a homogeneous diffusive predatorprey system. *Journal of Differential Equations* 246(5):1944 – 1977.