Artificial Intelligence in an Agent-Based Model

John Walsh
Thomas Jefferson High School for Science and Technology
Burke, Virginia

June 2, 2009

Abstract

Agent-based modeling is an extremely diverse field of research, and much analysis and research into the effectiveness of agent-based modeling has been done. Agent-based modeling is valued for its ability to model scenarios in a level of detail that would be prohibitively complicated in an equation based model. I will investigate the possible value of detailing agent behavior beyond simple rules, to the level of basic artificial intelligence for each agent. This project will yield a deeper understanding of multi-agent modeling. A modeling of a simple predator-prey interaction with implementation of advanced decision-making mechanisms for agents should yield different results depending on the level of intelligence each agent has.

Keywords: modeling, agents, artificial intelligence

1 Introduction

1.1 Project Scope

Clearly, implementing a scenario that investigates all artificial intelligence in agent-based modeling is beyond the scope of the resources available. As a result, this project will attempt to analyze the results of implementing varying levels of artificial intelligence for agents in a simple simulation between predator and prey. Multiple cases of the scenario will be set up, and the rules for each agent will be varied between cases. The cases will be compared to yield results concerning the effect of the artificial intelligence.

1.2 Problem Goal

Ideally I will gather information about the impact of modeling a scenario with artificial intelligence in contrast with simpler setups. The existing research into the effectiveness of artificial intelligence in agents is considerably sparser than the investigation into the rest of this area, and this project will hopefully provide a starting point for further research.

2 Background

Research has been done on the effectiveness of agent-based modeling versus equation-based modeling. A specific example compared traditional equation-based models of interactions to agent-based modeling, and attempted to determine whether agent-based modeling provided advantages over equation-based modeling. The subject under examination was the interrelation of wealth versus education over the generations, and the comparison yielded similar results in the increasing disparity of wealth over generations due to the education of the respective children. However, the agent-based modeling yielded more information than the equations, showing that the classes which a family runs in do change over the generations, a result that the equation-based model was unable to show.

In addition, another experiment investigated the necessary level of detail for rules that govern agents in a model, and determined that clearly a basic level of modeling agents is not sufficient when the results do not match results found in the actual scenario being modeled. An initial decision was to avoid grid-based setups. Although much easier to create, I believe grids are unreasonably unnatural and constricting. Also, movement is less natural in grid-based setups. However, I learned that there is a good reason most agent-based models are grid-based: agent interaction is very difficult to handle otherwise.

3 Development

First I created a basic, non-grid based environment with agents that moved randomly. Next, I defined the interaction between predator and prey. Subsequently, I implemented the ability to seed the program with a given value, to allow for reproduction of previous runs. Then I edited the code to output the populations over time, before implementing agent reproduction. Due to the issue of difficult collision detection inherent in my non-grid based scenario, I had to take a considerable chunk of time implementing a faster form of collision detection. Finally, I began work on the basic and advanced intelligence scenarios.

4 Program Setup

The code can be seeded by the compiler, allowing simulations to be repeated, but this is set inside the code, meaning that the user cannot adjust it without recompiling. However, the user *can* adjust the sliders to set the initial values for each kind of agent, while the reset button resets the simulation using the current seed. The new sim button allows the user to get a new seed for the simulation, using the current parameters defined in the user interface.

5 Basic Intelligence Scenario

Each agent moves and interacts according to methods that are defined separately. Agents are allowed to reproduce within a species, and predators are capable of consuming the prey. In additon, the populations of each type of agent over time are displayed on screen, allowing the user to view the value of each population as a function of time. Also, the program is now capable of outputting the populations over time to a file for later graphing.

Figure 1: Early program during runtime

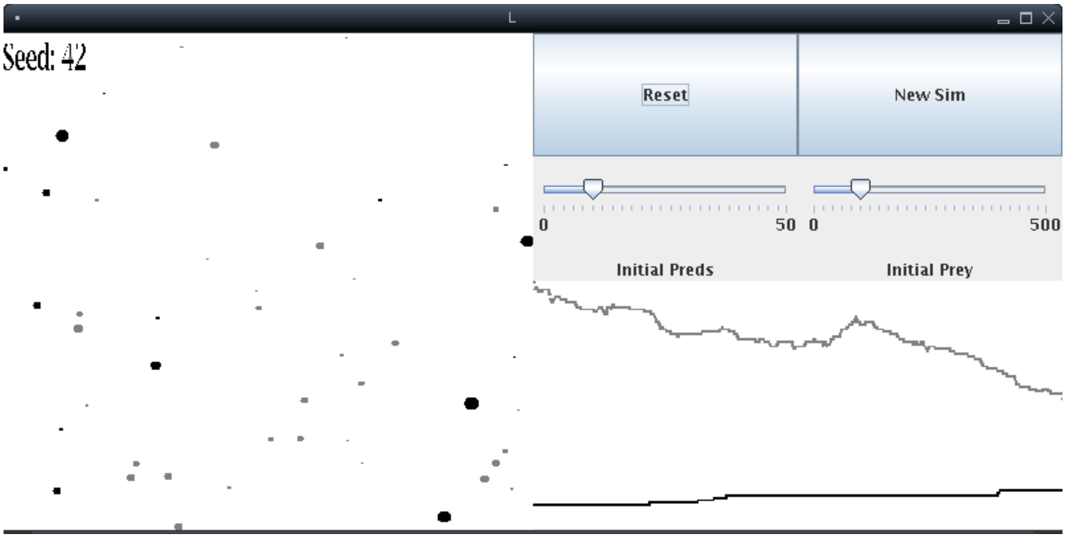
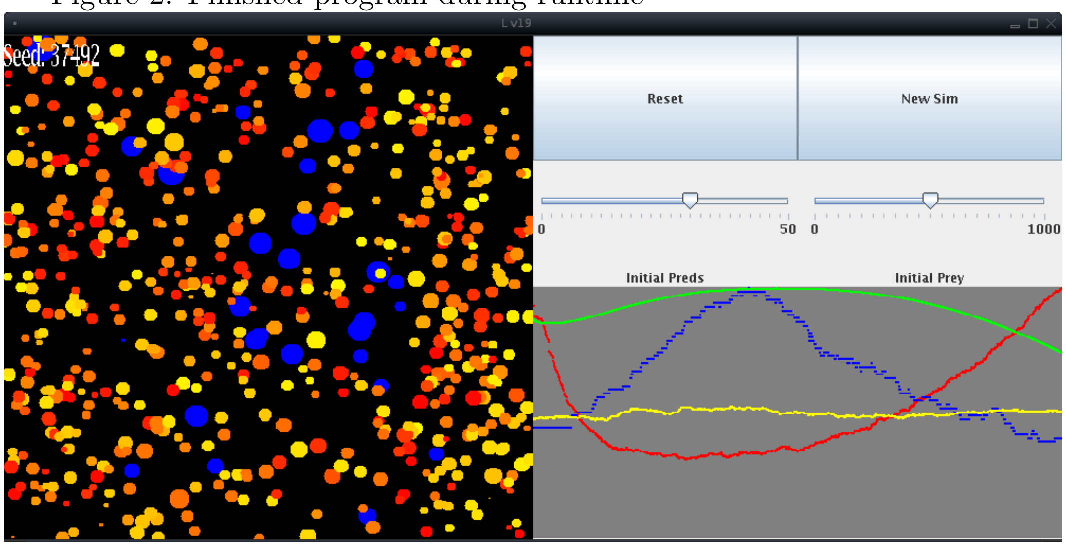


Figure 2: Finished program during runtime



An important feature was food for the prey, because otherwise they simply died out and the predators quickly followed. After much tweaking, I was finally able to determine a setup that allowed for a relatively stable population of prey. After this, I began tweaking parameters in the operation of the predators to setup a scenario in which the population of the predators was also stable. Along the way, the most surprising thing I learned was how difficult it was to set the parameters (speed, efficiency of mass absorbtion from the prey, minimum time between meals, etc) for the simulation such that the resulting feedback loops created relatively stable populations.

6 Advanced Intelligence Scenario

I have implemented a non-random intelligence for the prey. Each prey is given a value from 0 to 1 determining how likely it will be to head toward the nearest food above a certain threshold each time it picks a direction. When prey reproduce, the child receives the average of the parents' intelligence plus a value that varies randomly between -.25 and .25. A lower value for intelligence indicates that the agent's movements are more random. Unfortunately, due to In addition, I graphed the average intelligence over time of the agents, and found that it does in fact increase, as the less intelligent agents are weeded out.

7 Results and Discussion

Using the data output I implemented, I looked at the data generated by the simulation. I will have the program output its results to a file, based on the seed used for the program, and graph that data separately for presentation. The results could be useful to someone considering modeling a population of agents in a more detailed way, incorporating more specific aspects of a given scenario's agents. So far I have reached the conclusion that it is much more difficult to implement a good non grid-based agent model than one would expect. Continually one runs into the issue of processing time. It takes orders of magnitude longer to deal with agent interactions when there is not a simple grid to check the agents against. I implemented a collision detection system that breaks the area into blocks and then places each agent into each block it intersects before checking each agent within a block against every other agent in that block. This helps shorten the time required to check each agent against every other agent, but still does require significant processing power.

I attempted to create AI for the prey to allow them to flee the closest predator. However, I ran into issues in which the prey would run away from the closest predator it "saw", crossed the border, and "see" a predator that was actually much closer. The end result was that the "intelligent" agents would oscillate on the edges of the map, depleting the food in the area, and starving to death. I attempted to get the predators to look across edges, but time constraints and difficulty level prevented success, so I had to strip out the code and return to the previous scenario.

Figure 3: Graph of early program results

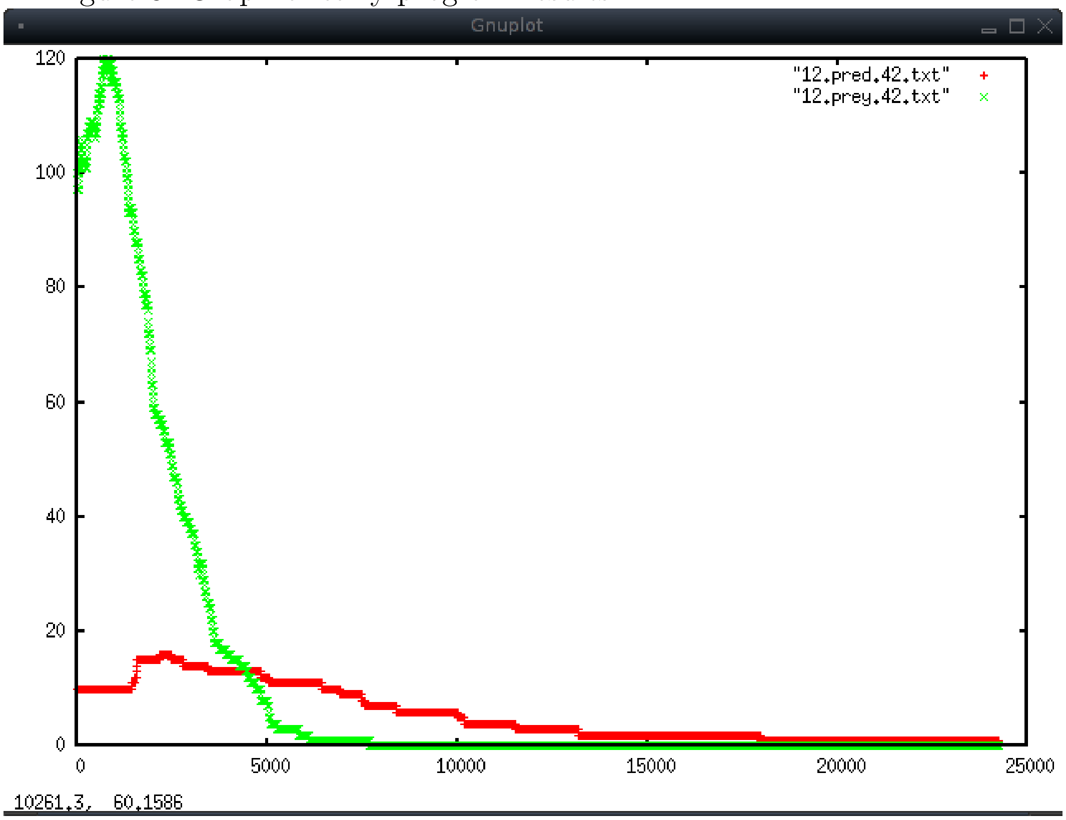
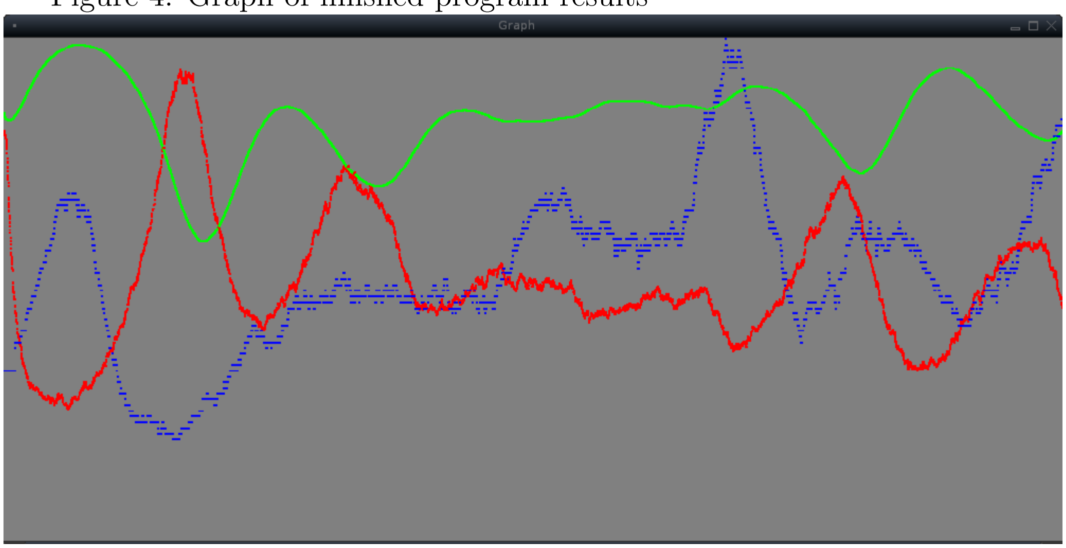


Figure 4: Graph of finished program results



References

[1] Barstow, D. 1987. Artificial intelligence and software engineering. In *Proceedings of the 9th international Conference on Software Engineering* (Monterey, California, United States). International Conference on Software Engineering. IEEE Computer Society Press, Los Alamitos, CA, 200-211.

[2] Chopinaud, C., Fallah Seghrouchni, A. E., and Taillibert, P. 2005. Automatic Generation of Self-controlled Autonomous Agents. In *Proceedings of the IEEE/WIC/ACM international Conference on intelligent Agent Technology* (September 19 - 22, 2005). IAT. IEEE Computer Society, Washington, DC, 755-758. DOI= http://dx.doi.org/10.1109/IAT.2005.46

[3] Guo, Z. and Tay, J. C. 2008. Granularity and the validation of agent-based models. In *Proceedings of the 2008 Spring Simulation Multiconference* (Ottawa, Canada, April 14 - 17, 2008). SpringSim '08. ACM, New York, NY, 153-161. DOI= http://doi.acm.org/10.1145/1400549.1400568

[4] Macal, C. M. and North, M. J. 2005. Tutorial on agent-based modeling and simulation. In *Proceedings of the 37th Conference on Winter Simulation* (Orlando, Florida, December 04 - 07, 2005). Winter Simulation Conference. Winter Simulation Conference, 2-15.

[5] Ribeiro, C. H. 1998. Embedding a Priori Knowledge in Reinforcement Learning. *J. Intell. Robotics Syst.* 21, 1 (Jan. 1998), 51-71. DOI= http://dx.doi.org/10.1023/A:1007968115863

[6] Tang, Y., Parsons, S., and Sklar, E. 2006. Agent-based modeling of human education data. In *Proceedings of the Fifth international Joint Conference on Autonomous Agents and Multiagent Systems* (Hakodate, Japan, May 08 - 12, 2006). AAMAS '06. ACM, New York, NY, 129-131. DOI= http://doi.acm.org/10.1145/1160633.1160654

[7] Xu, D., Volz, R., Ioerger, T., and Yen, J. 2002. Modeling and verifying multi-agent behaviors using predicate/transition nets. In *Proceedings of the 14th international Conference on Software Engineering and Knowledge Engineering* (Ischia, Italy, July 15 - 19,

2002). SEKE '02, vol. 27. ACM, New York, NY, 193-200. DOI= http://doi.acm.org/10.1145/568760.568794