CHAPTER 1

INTRODUCTION

Agent-based modeling has become a commonly-used method of simulation in a wide variety of fields. It has been used to simulate traffic patterns, markets, supply chains, wildlife ecology, and networking. It is used in both the "hard" sciences and in the social sciences, due to its ability to show emergent behaviors, or behaviors that arise from the interaction between the different agents. However, the creation of an agent model can be a difficult task for non-programmers, and the model that is created may need alteration if it does not fit the real-world data as precisely as needed, making the task even more difficult.

Because simulations need to be accurate, the decision processes for agent-based models are frequently derived from sequential observations of behavior. However, the current methods of calculation for an agent’s decision process require a great deal of time, and the calculations can be very complex to do by hand. Those with expert knowledge in the area that is being simulated have to work together with programmers to provide the information that is necessary to filter the data and calculate the appropriate probabilities for the various actions. This required cooperation can create further difficulties, as those that understand the area being simulated do not necessarily understand the limitations imposed by a computational model, and the programmers may not have the expert knowledge needed to correctly interpret the data.

We propose to develop an algorithm that will calculate and output one or more formal representations of an agent’s behavior, similar to a Markov decision process, but with support for cycles, as needed, from the collected information, as well as a tool that will easily allow the alteration of each decision process to meet the requirements of the modeler. This tool will then produce a file in a standard format that can be efficiently read and used in a variety of applications, in order to create initial decision processes for believable behaviors.

CHAPTER 2

RELATED WORK

Agent-based modeling is a useful method to simulate systems in which individual behaviors are varied and complex. It has the ability to show phenomena that arise from the interaction of individual agents that would otherwise be difficult to predict. It is also a flexible system of simulation, allowing the programmer to more easily change the behaviors of individual agents, or to add more agents to the system. This makes it ideal for the simulation of complex, dynamic systems, particularly those that involve human behavior (Bonabeau, 2002).

Because simulation has become such a commonly used tool, it is important that it is both reliable and valid. A simulation can be said to be reliable if it produces the same results when given the same data. While it can be relatively easy to determine if there are consistent results that are obtained with a particular model, determining the validity of that model can be more difficult (Garson, 2009). The validity of a simulation can be defined in more than one way. If a simulation appears to observers to respond in the same way as what is being simulated, it can be said to have “face validity.” This is the easiest type of validity to determine, but it can also be inaccurate. “Event validity”, obtained by providing the simulation with known data and checking the results, is a more accurate method of determining the validity of the simulation. By using known data, the correlation between the results of the simulation and the known results can be determined, giving a good sense of the simulation’s accuracy (Stanislaw, 1986).

2.1 MODELING HUMAN BEHAVIOR WITH AGENT-BASED MODELS

There have been many different decision-making methods used in the attempt to create believable behavior in agent-based models. Some researchers, such as Konolige, have argued in favor of a deductive, first-order logic system of decision-making, basing the agents’ knowledge on a core set of beliefs and the knowledge that can be derived from it (1986). In this type of model, the agent’s behavior is rule-based, using only the knowledge that the agent has about the world and what can be logically determined by using that knowledge. This, however, does not appear to be the way that humans reason, and seems more suitable to a knowledge base than an agent-based model.

To make the model less restrictive, some researchers have used Hintikka’s “possible worlds” approach, allowing the agent a certain amount of belief in any state of the current world that could be possible, based on the facts the agent is currently aware of (1967). Ginsberg and Smith argued for this approach, because the number of changes in the world between time steps is relatively small, so that keeping track of changes in the world, and therefore changes in the agent’s belief state, should be relatively simple (1988). However, when using their approach, if information contradictory to an agent’s current belief state is discovered, the entire knowledge base needs to be reconstructed. It then becomes extremely computationally expensive. Furthermore, scalability becomes an issue, as beliefs about every possible fact in the given world have to be stored and updated for each agent individually.

The BDI model, or Belief, Desire, and Intentions model, is a less computationally complex version of the possible worlds model. It stores the agent’s current beliefs in the state of the world based upon past events of which the agent is aware. Desires are the agent’s current goals, and Intentions are the agent’s plans to achieve those goals. This model is based upon ideas from cognitive psychology about the nature of human thought (Georgeff et al., 1999). By extending intentions to include multi-agent planning, Cohen and Levesque described a method of using BDI models in multi-agent systems (1990). Their agents also include a degree of commitment to their goals, allowing an agent to drop its goals in favor of a new one when interacting with another agent.

2.2 SEQUENTIAL OBSERVATIONS VERSUS COGNITIVE MODELING

A problem with modeling human behavior based upon theories of cognition is that cognitive models are difficult to quantify. Humans do not seem to make decisions based entirely upon mathematical formulae or deductive logic. In contrast to standard machine-learning methodology, humans actually make poorer decisions when given too much information. Instead, humans use heuristics, and these heuristics are not necessarily consistent between each individual, types of decisions, or even each time the same decision is made. They can be derived from individual preference, generalizations, culture, experience, or even learned from others (Gigerenzer and Gaissmeier, 2011).

Emotions also play a large part in human decision-making, as shown in neurological studies related to damage of the prefrontal cortex (Bechara et al., 2000) and studies on the amygdala and Pavlovian responses (Seymour and Dolan, 2008). There have been many attempts to incorporate emotion into agent decision-making, such as the Cathexis model (Velásquez, 1997), EBDI (Jiang et al., 2007), and a Neuro-Fuzzy agent with emotional intelligence (Sharada and Ramanaiah, 2010). However, as shown by Martínez-Miranda and Aldea, while emotion is important in human decision-making, emotional models tend to only perform well in the specific environment for which they were designed (2005). The concept of “emotion” is not well-defined in psychology, making it also difficult to quantify (Sloman, 2001). This makes a standard cognitive-based model for even a simple scenario almost impossible to accurately define and implement.

Time-sequence agent-based models, or agent-based models derived from sequences of observations over time, however, have been implemented with a great deal of success in a variety of areas other than human behavior. For instance, OptorSim uses predictive modeling to optimize the allocation of resources for file sharing, replication, and job execution in a Data Grid based on the sequence of jobs executed over time (Bell et al., 2003). A model of Chilean agriculture designed by Berger predicted economic changes based upon irrigation and adoption of new farming technology (2001). LUCITA, or Land Use Changes In The Amazon, was developed to determine the effects of local farming on Amazonian deforestation (Deadman et al., 2004). Time-sequence traffic data for densely populated regions has been used to predict traffic forecasts in Germany, allowing travelers to select the most appropriate route to their destinations (Wahle and Schreckenberg, 2001)

Likewise, single agents have been successful in learning patterns of human behavior from sequential observations. In an attempt to learn human behavior based on emotion in a smart-home setting, Leon et al. created the iSpace and iDorm, test facilities that used sensors to detect physical changes that are associated with certain emotions (2010). They used an analysis of sequential behavioral patterns along with Autoassociative Neural Networks that use physiological responses to predict the emotional state and likely behavior of the smart-home occupant. This allowed the agent to automate some of the smart-home’s systems.

ILSA, an agent-based smart-home system designed by Guralnik and Haigh to assist the elderly, uses sensor readings to determine sequences of the occupant’s behaviors (2002). It uses the sequential patterns of which sensors fire to determine which times certain activities take place, such as what time a person wakes up, and what time they go to sleep. The researchers that designed ILSA concluded that the order of sensors firing over a time interval was important to learn the behavior patterns of the person living in the home.

MavHome, a smart-home designed by Cook et al., also uses sequential sensor readings for behavior prediction (2003). The goal of MavHome was to adapt to the behavior of its inhabitants by automating processes such as turning up the heat in the morning, or turning on the light and coffee maker after the bedroom alarm goes off. Instead of just using sensors for doors opening and closing, MavHome uses a wide variety of sensors, such as temperature sensors and sensors to monitor the lawn moisture level. Behavior patterns are learned online, and prediction algorithms are used to match patterns in order to determine which devices to operate within the home. Because the learning is online rather than offline, it uses a string compression algorithm, Active LeZi, to compress the behavior sequence and increase the agent’s online learning speed. Active LeZi uses a variable order Markov model to predict the probability of the next behavior in the sequence, reducing computational time (Gopalratnam and Cook, 2004).

2.3 SEQUENTIALLY-BASED MODELS OF HUMAN BEHAVIOR IN MULTI-AGENT SYSTEMS

Because of the relative ease of creating more believable results in video games, the behavior models of human NPC’s, or non-player-character agents, is already frequently based upon sequential observations. In many First-Person Shooters, as well as in games like *Forza Motorsport* and *Black and White*, opposing agent actions mimic the actions of users playing the game. This provides the player with a challenge more suitable to his ability level and style of play, making it more enjoyable. It also makes the opposing agent’s behavior seem more realistic without being too computationally difficult to calculate (van Hoorn et. al, 2009).

More recently, there has been some success in building believable agents in academic competitions. A framework has been developed for the RoboCup simulation league using a combination of observed player behavior and case-based reasoning that makes training the robotic soccer player much more simple (Lam et al., 2006). It learns behaviors from logs of human players’ actions. It then turns these into cases to be used in case-based reasoning. These cases are then weighted automatically using a k-nearest neighbor classifier. This algorithm is run until the agent responds to a situation in a simulation in the same way as the previously stored, “test” agents. After the agent has learned a sufficient amount of behaviors from humans, it can then train other agents (Floyd et al., 2008).

BotPrize is a competition using the game *Unreal Tournament 2004* that uses the Gamebots system (Adobbati et al., 2001). It is held every year, and is a DeathMatch First-Person Shooter type game. Human judges play the game along with the bots, and try to distinguish between human and agent players. The UT^2 bot from the Neural Networks Group at the University of Texas, Austin, originally used recorded human behavior sequences to navigate when the bot became stuck. This bot placed second in 2010 (Karpov et al., 2012). After the 2010 competition, they increased the role of human behavior imitation in the evolution of combat behaviors and in navigation (Schrum et al., 2011). Subsequently, they fooled more than 50% of the judges in the 2012 competition, tying for first place with another bot that also mirrored human behavior.

As agents based upon sequential observations become more believably human, it can clearly be seen that imitation is a valid way of representing human behavior. While it would be computationally difficult to model all possible behaviors in a simulation that is very open-ended, many simulations have a limited number of possible actions available to the agent, making imitation a good way of providing a believable initial decision process (Umarov et al., 2012). Automation of the actual calculation of transitional probabilities between behaviors would allow more time for the modeler to make adjustments as needed, making modeling easier and providing a more accurate, believable end-product.

CHAPTER 3

PROPOSED RESEARCH

THESIS GOAL:

PLAN OF WORK:

APPROXIMATE TIMELINE

s

REFERENCES