Proceedings of the Twenty-Seventh International Florida Artificial Intelligence Research Society Conference

An Approach for Constructing Reliable Social Agent Based Systems Considering Dynamic Environment and Other Factors Affecting Their Progress

Inderjeet Singh Dogra and Ziad Kobti

School of Computer Science
University of Windsor,
Windsor, Ontario, N9B 3P4, Canada.
{dogra, kobti}@uwindsor.ca

Abstract

Construction of agent based model systems is often difficult considering the dynamic and complex nature of real world problems and various implicit factors affecting their behavior. This presents a problem in building accurate and valid systems for use as decision support tools. In this paper, we present an approach for handling some of these factors, specifically acceptance rate, retention rate and social influence, and enable simulated social agent models to evolve in a dynamic environment. The objective is to achieve a more reliable behavior and outcome for the agents in the simulation, despite unpredictable environmental changes. The agents' knowledge, in terms of their observed responses to the environment and corresponding outcomes, is captured in a semantic tree. A metric is used to detect changes in the environment and the threshold of response of the agent, thereby triggering the agent to adapt its decision tree to maintain a reasonable response beyond its historical knowledge. The results reveal the ability of agents to detect changes in the environment more quickly and with better accuracy, using a case study, and as a result learn to adapt by modifying their decision tree under the influence of considered factors.

1 Introduction

The acceptability of an Agent Based Model (ABM) strongly depends upon the validation and reliability of its results with respect to the real world problem being modeled. Considering this, (Remondino and Correndo 2006) describes three classes of validation, i.e. *Empirical Validation*, *Predictive Validation* and *Structural Validation*. *Empirical Validation* intends to validate an ABM by comparing its simulation results with those observed in real world system. This gives an idea of the correctness of model for some given situations. *Predictive Validation* focuses on validating an ABM, in terms of producing more reliable outcomes for unseen situations, the ones not directly observable in real world. It helps study the behavior of real world system for non-repeatable situations. Lastly, *Structural Validation* ensures that processes used by an ABM to generate outcome comply with those used in the real world system.

Of the three validation types, in recent work by (Dogra and Kobti 2013) was an attempt towards handling the *Predictive Validation* in ABMs working under the influence of dynamic environment. Here, the overall validation or reliability of ABM was measured in terms of average prediction accuracy. This measure was shown to increase using the proposed approach despite the initialization of agents with a prior knowledge different from the one reflected by the simulation environment. However, the experiments were conducted in a controlled environment. It was assumed that the data, used for training the agents and thereby representing the simulation environment, was available all at once.

In this work, we incorporate three main factors namely acceptance rate, retention rate and social influence (Kobti et al. 2006) into the approach outlined in (Dogra and Kobti 2013) and present a detailed study of their effects on the overall progress of ABM, aiming specifically at the learning rate of agents and synchronization level attained by them. Further, the training data set is modeled as a continuous stream of data, causing the knowledge represented by it to change with every time step during the simulation. In the next section, we present an overview of our previous work, followed by a detailed description of each of the considered factors. The case study is then presented along with the experimental setup and the results obtained are discussed in the section following it. In final section, we present our conclusion and the scope of future work in this research.

2 Related Work

An integrated approach was introduced in (Dogra and Kobti 2013), for improving the overall reliability of an ABM working under dynamic environment, by integrating the features of learning and adaptability in it. These features ensured the consistent synchronization of ABM with the dynamically changing environment, thus maintaining the validity of its results at all times. The reliability of ABM was measured in terms of average prediction accuracy (Olson and Delen 2008) to determine its capability of predicting correct outcomes. Two aspects were identified towards achieving the outlined goal: the first one dealt with incorporating learning into agents to make them capable of learning from their past experiences and absorb environmental changes. The second one focused on computing the similarity of knowledge, reflected by the decision trees, contained by an agent and that represented by the dynamic simulation environment. This similarity measure was also used by an agent to detect a change in the environment and thereby trigger its learning process. In addition, it was also used as a fitness function by them to determine their level of synchronization with the simulation environment and accordingly monitor their learning behavior.

3 Defining Factors/Parameters

This section describes some of the real world factors/parameters that tend to affect the learning of new information. Three factors/parameters have been considered: acceptance rate, retention rate and social influence; which tend to mimic the accuracy with which people learn new information, their chances of sticking to existing information and not learning the new one, and the role of social interactions in knowledge sharing amongst them, respectively (Kobti et al. 2006). Here, the new information corresponds to the change in knowledge represented by environment, i.e. global knowledge.

Learning Rate: The learning rate of an agent gives an idea of its capability to adapt to environmental changes. In other words, it is defined as the rate at which an agent detects and absorbs new information or a change in it, from the simulation environment. In this paper, the average learning rate of agents is considered, depicted by the slope of average similarity measure plot, shown in Figure 1.

Synchronization Level: The synchronization level attained by an agent refers to the measure of similarity between its own knowledge and the global knowledge. It gives an idea of the correctness of agent's knowledge with respect to the global knowledge. In this paper, the average synchronization level of agents is considered, reflected by the y-axis of the average similarity measure plot, shown in Figure 1.

4 Case Study and Experimental Approach

We use the same dataset of *Golf Play* referred from the University of Regina. The experiment was setup around the problem described in (Emele et al. 2012)'s work. In the experimental setup, the global agent was considered as the supervisor of golf ground, with the prime motive of granting agents the access to ground based on current weather conditions. The global knowledge, possessed by the global agent, represents the rules/policies for allocating ground under different weather conditions. This knowledge can be accessed by the simulation agents, referred to as the seeker agents, through interactions with the global agent. In the beginning of the simulation, each agent was assigned a prior knowledge different from that of the global knowledge. This was done to model the worst case scenario, where the global knowledge (i.e. rules/policies) has recently changed and all agents are out of synch with it. A synthetic data generator was used to extend the original *Golf Play* dataset. This extended dataset was then used as a training dataset for the agents and the decision tree induced from it represented the global knowledge. Further, a decision tree editor was used to generate different variants of the original decision tree (obtained over *Golf Play* dataset), which were randomly assigned to the seeker agents (Dogra and Kobti 2013). With this initialization, the aim of all seeker agents was to build their respective prediction models that mapped the global knowledge as closely as possible. This would help agents know, beforehand, if the global agent will grant access to golf ground or not, for the current weather conditions. This would help them considerably reduce the number of requests to the global agent, by only requesting for the ground when they are highly certain of getting access to it. Therefore, each seeker agent intends to build more accurate prediction model, by continuously learning and adapting the changes occurring in the global knowledge.

5 Results and Discussions

For all the results presented in this section, a population of 20 agents was used except for the social influence factor where a population of 50 agents was used instead. Further, 1000 synthetic data records were generated from the original Golf Play dataset. Thus, each simulation ran for 1000 time steps and every time step a record from this dataset was transmitted to all the seeker agents. Moreover, a memory length of 120 records was used for all the seeker agents and the corresponding simulation was referred to as ABM-120. For all the results obtained, the average similarity measure plot, shown in Figure 1, was used as the standard to study the average learning rate of agents (represented by the slope of graph) and their average synchronization level (represented by average similarity measure, expressed in %). In addition, SMT was set to 80% in all the experiments.

The training dataset, obtained using synthetic data generator, was modeled as a continuous stream of data. In other words, instead of using all the records in this dataset for generating a decision tree for the global knowledge, this dataset was initialized with only 120 records at the beginning and with every time step one record was added to it. This caused the global knowledge to update incrementally with every time step. Figure 1 presents a comparison of the results obtained by using incrementing and non-incrementing datasets. These results are referred to as ID-Enabled (Incrementing Dataset) and ID-Disabled, respectively, where ID-Disabled is referred from (Dogra and Kobti 2013).

Effect of Retention Rate

ABM-120 was run for three different values of retention rate, specifically 70, 80 and 90. The corresponding results obtained, referred to as RR-70, RR-80 and RR-90 respectively, are shown together in Figure 2 for an easy comparison. For all three values of retention rate, the acceptance rate was kept constant at 100%.

Higher retention rate should make an agent more reluctant to accept the new information, sticking firmly to its existing knowledge, and vice-versa. The results shown in Figure 2 portray similar behavior, where the average similarity measure for agents in RR-70 is the earliest to cross the SMT value and for RR-90 it never reaches the SMT value. On the other hand, the performance of average similarity measure for RR-80 lies approximately in between the two. Following this, it can be observed that the agents initialized

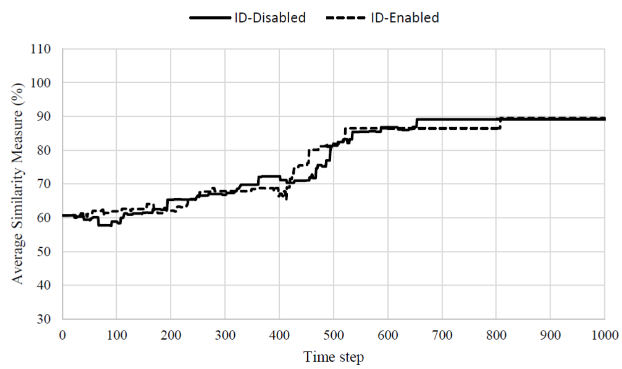

Figure 1: Average similarity measure of agents with and without incrementing training dataset for ABM-120 and SMT 80%

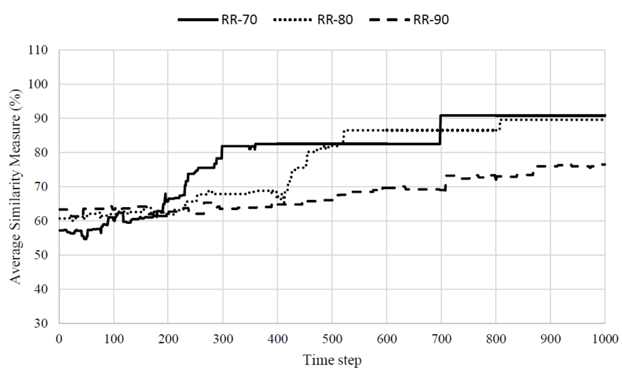

Figure 2: Average similarity measure of agents for different values of retention rate for ABM-120 and SMT 80%

with smaller value of retention rate require lesser number of records to reach the SMT. Specifically, RR-70 required only 300 records to achieve desired synchronization level, RR-80 required approximately 455 records and RR-90 never reached the SMT. Thus, the results go well with the hypothesis indicating that the agents with higher retention rate are more likely to ignore the new information as compared to their counterparts. In addition, the average learning rate reflected by the slope of each plot also follows the similar trend, where it is highest for agents in RR-70, lowest for agents in RR-90 and follows a decent rate in RR-80, again lying somewhere in between the previous two.

Effect of Acceptance Rate

Three different values for acceptance rate, i.e. 60, 80 and 100, were used to run ABM-120. The corresponding results obtained are referred to as AR-60, AR-80 and AR-100, respectively, and are shown in Figure 3. Here, for all three values of acceptance rate, the retention rate was opted as 80% considering it to be a trade-off between its other two values.

Following the discussion on acceptance rate, it is expected that an agent with a higher value for this parameter will learn the new information with least possible error and vice-versa. The results shown in Figure 3 follow a similar trend,

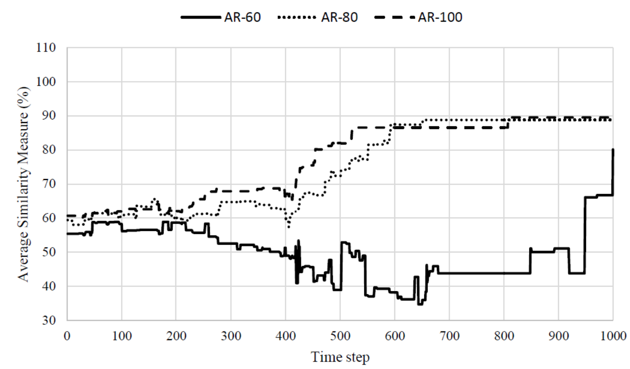

Figure 3: Average similarity measure of agents for different values of acceptance rate for ABM-120 and SMT 80%

where AR-100 agents are the fastest and AR-60 agents are the slowest to reach the SMT. The performance of AR-80 agents, though lies in between the two, but is closer to AR-100, as depicted by the results. Although the results for AR-80 and AR-100 closely follow each other, however it can still be concluded that the agents with high acceptance rate tend to achieve a higher level of synchronization with the environment.

The result for AR-60 shows deterioration in the average similarity measure for the first 600 records. The reason for such a behavior is the difference in the agents' knowledge and the global knowledge in the beginning, due to which despite updating their knowledge in every time step, the agents end up learning incorrect information in 40% of the cases. However, after going through sufficient number of records, the agents start making an improvement towards their individual synchronization level and consequently the overall average similarity measure also starts improving. Thus, the results justify the hypothesis that higher the value of acceptance rate in agents, the more accurately they tend to learn the new information and require less time or data records to achieve desired level of synchronization. Further, the slopes of each plot in Figure 3 give a clear indication of highest learning rate shown by AR-100 agents, closely followed by the learning rate of AR-80 agents and finally a poor learning rate is displayed by AR-60 agents.

Effect of Social Influence

Using this parameter, we intend to study the impact of knowledge sharing among the agents, taking into account the parameters of retention rate and acceptance rate. The value acceptance rate and retention rate was set to 80%, as it depicts the trade-off between their other values. Also, this value tends to provide better and more realistic results as shown in their respective figures, i.e. Figure 2 and Figure 3. In this case, a population of 50 agents was considered since the number 20 was too small to setup any neighborhood network amongst the agents. Each agent was linked to its 3 closest neighbors based on their respective locations. The ABM-120 was run twice, with and without the social influence and the results obtained are referred to as SI-Enabled

(SI-E) and SI-Disabled (SI-D), respectively.

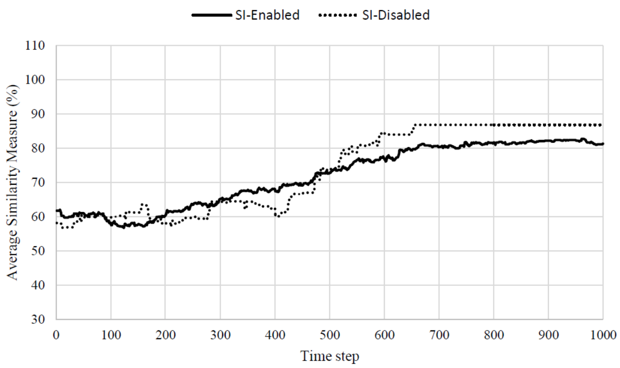

Figure 4: Average similarity measure of agents with and without social influence for ABM-120 and SMT 80%

Social influence and hence the social interactions were considered to enable knowledge sharing among the agents, with the expectation that the agents will learn faster requiring lesser number of records to attain the desired synchronization level, as compared to when they are not socially active. However, the results shown in Figure 4 contradict with this hypothesis. In the results, the agents achieve the SMT value faster in the case when no social interactions exist, pointing towards the negative effect of social influence on the agents' learning. This indicates that the knowledge shared by the agents in a social network may not always be correct. The reason for such a behavior is that the knowledge of agents in the beginning was different from the global knowledge, and under such situation allowing them to share knowledge would lead to the spread of incorrect knowledge. Due to this, it requires more time/data records for an agent to acquire enough information regarding the change. Thus, despite learning from the global agent and through social interactions, the overall time taken by the average similarity measure to reach SMT is increased noticeably. Therefore, it can be inferred that social influence will have a positive effect on the working of agents, only when considerable number of agents possess the correct knowledge. The agents involved in social interactions, though tend to learn slowly, but are steadier with their progress in comparison to their counterpart.

6 Conclusions and Future Work

The experimental results show that for a lower value of retention rate and a higher value of acceptance rate, the agents exhibit a higher learning rate achieving a higher level of synchronization with the environment. However, for a value too high and too low for retention rate and acceptance rate, the agents require more records to reach SMT and show a significant deterioration in their learning rate, respectively. Thus, the results justify the hypothesis that lower retention and higher acceptance enable an agent to accept new information with more accuracy. The results for social influence highlight the negative impact knowledge sharing can have on the overall learning rate and synchronization level of an ABM, when majority of agents possess incorrect knowledge. Instead of increasing the learning rate and achieving higher level of synchronization, as expected, the results portray an opposite behavior. Therefore, it can be inferred that for social influence to work effectively, a considerable number of agents should possess the correct knowledge. Moreover, the comparative analysis done for incrementing dataset highlights that, agents tend to absorb nearly all environmental changes and thus attain better synchronization with environment, when trained using a continuous stream of training dataset. For future work we plan to test our approach on a bigger dataset that complies with the characteristics exhibited by the *Golf Play* dataset.

References

Bache, K., and Lichman, M. 2013. UCI machine learning repository.

Baqueiro, O.; Wang, Y. J.; McBurney, P.; and Coenen, F. 2009. Integrating data mining and agent based modeling and simulation. In *Advances in Data Mining. Applications and Theoretical Aspects*. Springer. 220–231.

Dogra, I. S., and Kobti, Z. 2013. Improving prediction accuracy in agent based modeling systems under dynamic environment. In *IEEE Congress on Evolutionary Computation (CEC), 2013*, 2114–2121. IEEE.

Emele, C. D.; Norman, T. J.; Şensoy, M.; and Parsons, S. 2012. Exploiting domain knowledge in making delegation decisions. In *Agents and Data Mining Interaction*. Springer Berlin Heidelberg. 117–131.

Kobti, Z.; Snowdon, A. W.; Rahaman, S.; Dunlop, T.; and Kent, R. D. 2006. Modeling the effects of social influence on driver behavior in applying child vehicle safety restraint. In *IEEE International Conference on Systems, Man and Cybernetics, 2006. SMC'06.*, volume 1, 145–150. IEEE.

Olson, D. L., and Delen, D. 2008. *Advanced data mining techniques*. Springer Berlin Heidelberg.

Quinlan, J. R. 1993. *C4.5: programs for machine learning*, volume 1. San Francisco: Morgan kaufmann.

Remondino, M., and Correndo, G. 2005. Data mining applied to agent based simulation. In *Proceedings of the 19th European Conference on Modelling and Simulation, Riga, Latvia*.

Remondino, M., and Correndo, G. 2006. Mabs validation through repeated execution and data mining analisys. *International Journal of Simulation Systems, Science & Technology* 7(6):10–21.

Song, A.; Padgham, L.; and Cavendon, L. 2007. Prediction in dynamic environment: Robocup rescue exploration.

Tadokoro, S.; Kitano, H.; Takahashi, T.; Noda, I.; Matsubara, H.; Shinjoh, A.; Koto, T.; Takeuchi, I.; Takahashi, H.; Matsuno, F.; et al. 2000. The robocup-rescue project: A robotic approach to the disaster mitigation problem. In *Proceedings of the IEEE International Conference on Robotics and Automation, 2000. ICRA'00.*, volume 4, 4089–4094. IEEE.