IJICTR International Journal of Information & Communication Technology Research

Volume 3- Number 4- September 2011 (65-72)

**Technical Note**

# Design of a Cellular Sugarscape Environment to Increase the Learning Speed in a Stochastic Multi-agent Network

Nasim Nourafza
Department of Computer Eng.
Science & Research Branch
Islamic Azad University (IAU)
Tehran, Iran
n_noorafza@yahoo.com

Saeed Setayeshi
Department of Medical Radiation Eng.,
Amirkabir University of Technolog
Tehran, Iran
setayesh@aut.ac.ir

Ahmad Khadem-Zadeh
Iran Telecom Research Center
Tehran, Iran
zadeh@itrc.ac.ir

Received: May 13, 2011- Accepted: July 28, 2011

*Abstract*—**Sugarscape model is a multi-agent environment used for modeling and organizing processes such as social, political and economic processes .The purpose of this study is to assess the learning ability of a learner system in sugarscape and for that reason the Boltzmann machine learning algorithm is evaluated .During the first experience, model number 1 which is a learned multi-agent model and is based on Boltzmann machine algorithm was considered .In this model each agent, was assigned with a parameter which indicates the agent's knowledge .Once the knowledge of agents reached the maximum, the model is converged .The criterion was time of convergence .After that, the second model which is a multi-agent, cellular, and learner based on Boltzmann machine learning algorithm was considered, in which the Boltzmann machine's learning algorithm is implemented in a multi-agent cellular environment . And finally the third model which is a multi-agent, cellular, Boltzmann machine learner model was assessed in sugarscape. And measure the time needed to reach convergence over specific number of agents for each model. After investigating the resulting diagrams it was concluded that the convergence speed of third model is more than the convergence speed of second model .Also, the convergence speed of second model is more than convergence speed of first model .Utilization of cellular automata results in a more speedy convergence for the model and this is due to rules and local interactions and transformation of generalized convergence to localized convergences .Application of sugarscape accelerates convergence speed relative to the case when it is not used .Additionally the maximum number of executable agents in third model is more than second model and also the maximum number of executable agents in second model is much greater than the first model .At last, it was deduced that the learning process of a learned multi-agent system and in sugarscape speeds up this process and does not have the limitations in the number of agents during execution.**

***Keywords- multi-agent system; learning; Boltzmann Machine learning algorithm; cellular automata; sugarscape; convergence***

## I. INTRODUCTION

One important issue in designing multi-agent systems is the difficulty in predicting all possible situations that are encountered by an agent and also to recognize its optimized behavior in such situations. Therefore in such systems where the behavior of other agents is not predefined learning is important [4]. And because of the fact that other agents are capable of learning and so the environment for a learner is not constant, learning in multi-agent is sophisticated [5]. Sugarscape model is also a multi-agent environment which is used for modeling and organizing processes such as social, political and economic processes. This environment is based on simplicity. Epstein and Axtell have proved that with simple rules a space can be realized and with that space the complications of human society can be easily modeled [14, 18-19, 32-36].

One algorithm than may perform in multi-agent environments is Boltzmann machine learning algorithm. Can this algorithm be used for learning in sugarscape? Because sugarscape is a stochastic environment and because most of learning environments are for deterministic environment with linear behavior and in a deterministic environment they have many weak points, such as, no convergence in system, limitations on number of agents or number of their choices and lack of capability for showing learner agent's uncertainty [1, 5, 7, 21-27], so it is well possible that they become nonresponsive in complicated environments where agents act randomly. As a result we see the necessity of using a responsive learning algorithm in stochastic environment. In the previous study it was stated that Boltzmann machine learning algorithm is a suitable learning algorithm for stochastic environment [48]. Thus for assessing learning capability of a learner system in sugarscape which is the goal of this study, Boltzmann machine learning algorithm has been used. In this study multi-agent systems and learning in multi-agent systems is explained first and then for the sake of used methodology, a brief description of Boltzmann machine is given and next an account of cellular automata and sugarscape is given. In part 5, learned multi-agent model based on Boltzmann machine learning algorithm is explained .In this model each agent, was assigned with a parameter which indicates the agent's knowledge .Once the knowledge of agents reached the maximum, the model is converged .The criterion was time of convergence. Then, cellular learned multi-agent model based on Boltzmann learning algorithm and cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugarscape is described. And measure the time needed to reach convergence over specific number of agents for each model. finally, a comparison of the three named models and discussion and conclusion is stated.

## II. MULTI-AGENT SYSTEM

Multi-agent systems are defined as a group of individual creatures that interact in a shared environment which is perceived by their sensors and act by their actuators [22, 28]. This concept is shown in figure number 1. In other words, multi-agent systems include a number of agents that interact with each other by means of connection [2]. Although these interactions may be classified as part of the environment but in such systems the interactions between agents are model separately from the environment. In multi-agent systems, the changes in the environment are dynamic. Because other agents affect the environment.

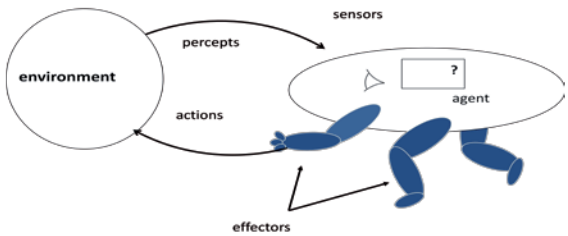



Figure1. Agents interact with environments through sensors and actuators [28].

## III. LEARNING IN MULTI-AGENT SYSTEM

The operational definition for learning in MAS is given as follows: learning is the capability of performing new duties (which were not possible before) and doing previous duties in a better manner as a result of changes that result from learning process [3]. Learning includes acquiring skills and refining skill [20]. Learning is specifically important in multi-agent adjustments where the behavior of other agents is not known beforehand [4]. Also often the structure of environment alters so for the survival of the agent it may be either reprogrammed or a compatibility mechanism is used [6].

In simpler words agents should learn because they don't know everything about their surrounding environment or they are unaware of other agents' behavior. In general learning can be present in a cooperative environment where agents share the knowledge or in an competitive environment where agents tend to become better than others [7]. Even if the agents isn't truly aware of other agents' existence, it perceives them as part of the environment and their behavior will be part of what they are going to learn [8].

Without doubt, online learner algorithms better suits multi-agent systems in which agents require update the information continuously [9].

## IV. METHODOLOGY

The methodology used in this study is based on Boltzmann machine learning algorithm and cellular automata and sugarscape environment which is described in the following:

### *A. Boltzmann machine*

Boltzmann machine was created by Hinton and Sejnowski in 1983, is a neural network designed

based on Boltzmann learning rule .The architecture of Boltzmann machine is illustrated in figure 2 .The output of each node in Boltzmann machine is a stochastic function of inputs rather than a deterministic function [29]. The output of a node is calculated probabilistic rather than with threshold or sigmoid output function [10].
The Boltzmann learning algorithm is based on 3 phase. In the first phase, input patterns and entered and clamp to units. Then units are select randomly and are change their states from $s_j$ to $-s_j$ with this probability:

$$p(s_j \leftarrow -s_j) = \frac{1}{1+\exp\left(\frac{-\Delta E}{T}\right)}$$

The activation phase is continued to state of low energy.
The stochastic nature of Boltzmann machine implies that there couldn't be one single appropriate status for the system .In return, the network can be set in a status until they reach thermal equilibrium in which special neurons change their status but the probability of each status could be calculated .In fact in equilibrium, the status of Boltzmann machine follows Boltzmann distribution [15]. This is the probability of system being in state of $E_\alpha$ energy and T temperature:

$$p(\alpha) = \frac{\exp\left(-\frac{E_\alpha}{T}\right)}{\sum_\beta \exp\left(\frac{-E_\beta}{T}\right)}$$

Note that update can be observed a local act:

$$\Delta E = -\Delta S_j \sum_{i=1}^{N} w_{ji} s_i = -2 \times |V_j|$$

$V_j$ is act of jth unit.
Now, temperature is decreased slowly until system is converged (annealing). In the end temperature, the correlation of pair units is calculated:

$$p_{ij}^{+} = < s_j \quad s_i >^{+}$$

In the second phase that is free run phase, previous calculations are repeated. But this time, the visible units don't clamp after enter the input patterns, all units update. Their status and annealing take place. In the end temperature, the correlation of pair units is calculated:

$$p_{ij}^{-} = < s_j \quad s_i >^{-}$$

In the third phase (learning phase), the Boltzmann machine learning rule is used for update the weights [10-13, 15-17, 29, 30]:

$$\Delta w_{ij} = \eta\left(p_{ij}^{+} - p_{ij}^{-}\right) \quad \forall_{i,j}$$

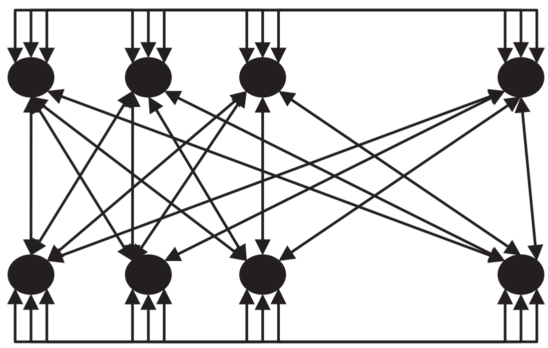
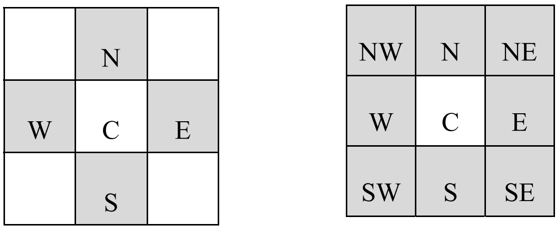


Figure 2. Boltzmann machine architecture includes one layer of visible neurons and one layer of hidden neurons.

*B. Cellular automata*

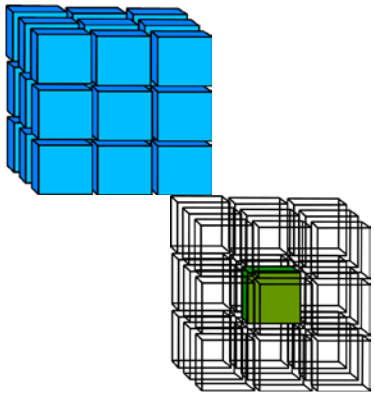
Cellular automata were first presented by John Van Neumann in 1940 [37]. During any stage, each automaton uses a set of rules in the form of a function of its own and its adjacent neighbors' status to derive the next status .Based on this local mutual effect, an initial structure creates a copy of itself after passing some stages .
Then S.Ulam considered a finite two dimensional net for cellular automata which were composed of elements called cells, in a way that these cells locally interact with each other [37]. Cellular automata is a mathematical model that can be employed for calculation and simulation of systems and acts like a simple discrete system that divulge complicated calculations and behavior with simple and local rules.
The characteristics of Cellular automata are given in references [38-40].
For cellular automata, local rule or local transitional function is defined. This transition function may be deterministic or stochastic [38-40] .These rules are defined by a string of bits and each bit indicated the next status corresponding to the bit number [42]. If the radius of neighborhood is increased the number of neighborhood status and therefore the total number of statuses increase [41]. The ratio of 1s in rule, is called Langton parameter [43]. To realizing of cellular automata behavior from the rules is very difficult and requires simulation .This network of cells can have d≥1 dimensions .For example a three dimensional network is shown in figure 3 .
With regard to determining next status according to special rules, cellular automata are divided into deterministic and stochastic groups .Up to this point the cases discussed were deterministic cellular automata, which mean cellular automata transits from one status to one status only in next time stage .In other words, transition rules, takes the status of one cell and its neighbors and precisely describes the next status .If however the status of a cell in the next time stage is described by a stochastic variable and is compared with a predefined countable value that is a function of initial status variable, cellular automata will be stochastic [44] .

Probabilistic cellular automata or its abbreviation (PCA) is studied as a model for determining dynamics of complicated systems [45].




Von Neumann neighborhood in two dimensional network.

Moore neighborhood in two dimensional network.



Showing neighborhoods in three dimensional network

Figure 3. Various important neighborhoods in cellular automata.

## C. *Sugarscape*

Sugarscape was first introduced by Epstein and Axtell [14]. In this environment bottom up method is used for modeling and organizing social and political and economic processes .In other words sugarscape is a multi-agent system which can be used for social modeling and investigation and analyzing social processes .This environment is based on simplicity which means the rules in this environment are simplified so that artificial complications are not enforced to the modeling space. Epstein and Axtell showed that with these simple rules a space can be achieved to model human society's complications in an easy way .

Main elements of sugarscape are:

Agent, rules, environment or space, sugar (resource) which are described in the following .

- *Agent*

The elements present in the sugarscape are called agent .Agents include people or organisms that simulate human behavior [14, 18-19, 32-36, 46-47].

- *Rules*

Rules are for life and continuation of agents in the environment .Different rules cause different behaviors .Execution of rules with different sequences also changes agents' behavior .As a result rules are one of the fundamental and extendable parts in sugarscape and by means of changing their execution sequence certain properties of human society can be modeled .That is to say, by changing and extending these rules the model may be extended [14, 18-19, 32-36, 46-47].

- *Landscape*

No certain topology has been defined for sugarscape but the environment can be regarded as a two dimensional network [14, 18-19, 32-36, 46-47].

- *Sugar (resource or possession)*

Just as sugar is the first supply for life, sugar is also considered as energy in sugarscape .Agents need to consume sugar for survival and sugar resources indicate possession or wealth [14, 18-19, 32-36, 46-47].

The advantage of artificial environment over natural environment lies in the fact that iterative tests may be carried out in artificial environment to assess the reflections of the environment .So far, sugarscape have been used with applications in optimizing wealth distribution in artificial society and also for balancing wealth .In the current study sugarscape model is utilized for application in learning process .

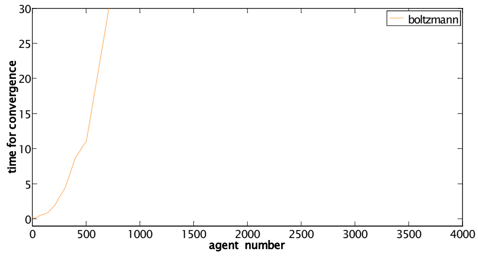
# V. IMPLEMENTATION

Here three learned multi-agent model and cellular learned multi-agent model and cellular learned multi-agent model based on sugarscape are investigated:

## A. *Learned multi-agent model based on Boltzmann machine learning algorithm*

Here a multi-agent environment is design and each agent, has been assigned with a parameter that indicates the knowledge of the agent .The goal is to increase the agent's knowledge until it reaches a definite maximum .For this purpose Boltzmann machine learning algorithm have been used .Initially the location and knowledge of agent was defined in a random manner .Application of Boltzmann machine learning algorithm causes at first, values of agents are clamped and the correlations of pair agents are calculated. Furthermore, in free phase, for each agent, sum of other agents' knowledge multiply their factor of influence is calculated. If this value is greater than the value of a random function introduced, the knowledge of agent updates. This action is repeated for all agent more times. Then the correlations of pair agents are calculated. Furthermore, in learning phase, the Boltzmann machine learning rule is used for updating of weights. This cycle continues until the knowledge of all agents reach maximum value or in other words the system converges .

This model was used for different number of agents 3, 5, 7, 11, 13, 17, 19, 31, 53, 73, 97, 127, 151, 211, 257, 307, 401, 503, 701, 1009, 1601, 2003, 3001, 3989 and iterated five times for each agent and the time spent for convergence was recorded .Graph number 1 shows the average time spent for convergence for a given number of agents .The x axis shows the density of agents and the y axis shows the average time spent for convergence of model .

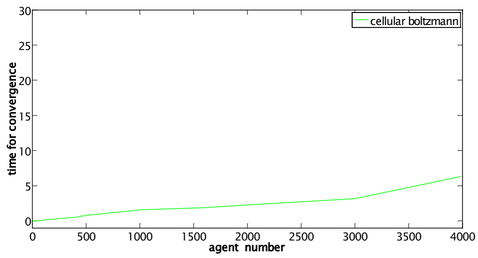



Graph 1. Average time spent to reach convergence for a given number of agents in learned multi-agent model based on Boltzmann machine learning algorithm.

### B. *Cellular learned multi-agent model based on Boltzmann machine learning algorithm*

The difference of this model with learned multi-agent model based on Boltzmann machine learning algorithm is that this model is considered to be cellular .Cellular environment is an n*n matrix and not only the rules of the environment are local; also the interactions of agents are also localized .This means each agent only interact with its neighbors . Here Moore neighborhood model was used .Hence each agent is in contact with eight surrounding neighbors .Therefore in this model, each agent is under the influence of its neighbor's knowledge and their factor of influence and with Boltzmann random function updates its knowledge. And this cycle continues until the knowledge of all agents reach maximum value or in other words the system converges .

This model was also used for different number of agents 3, 5, 7, 11, 13, 17, 19, 31, 53, 73, 97, 127, 151, 211, 257, 307, 401, 503, 701, 1009, 1601, 2003, 3001, 3989 and iterated five times for each agent and the time spent for convergence was recorded .Graph number 2 shows the average time spent for model convergence for a given number of agents .The x axis shows the density of agents and the y axis shows the average time spent for convergence of model .

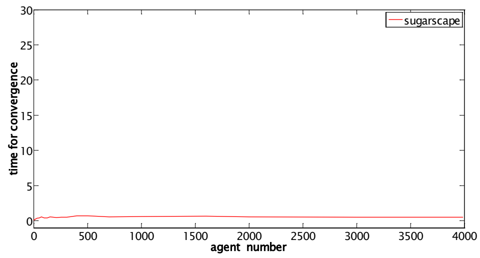



Graph 2. Average time spent to reach convergence for a given number of agents in cellular learned multi-agent model based on Boltzmann machine learning algorithm.

### C. *Cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugar model*

Here a cellular multi-agent model based on sugar model was designed with specific rules .Upon commencement of model execution, the initial location of agents was defined in a random manner . Also the level of sugar in vicinity [0..1], vision scope in vicinity [1..6], metabolism rate in vicinity [0..1], and age of agents in vicinity [60..100] is randomly distributed and the rate of sugar regrowth in the environment is also set as 1 .The environmental rules are localized .Agents in the environment transfer with rules of agents' motion in a way that each agent moves to the nearest location with maximum amount of sugar in sight (in neighborhood of agent) and withdraws the sugar from that location .Agents interact with each other in a manner that the level of sugar in each agent changes with neighboring agents' sugar and their factor of influence and with use of Boltzmann random function .Once the agents present in sugarscape reach sugar peaks the model is converged .Figure 4 illustrates how agents work with an initial population of 100 that randomly placed in sugar model with random vision scope, age, … .When the model run, states and number of remaining agents show in section 2 and 3 until in section 4, the model converged because all of agents reach to sugar peaks. Different colors show that agents are different in properties such as vision scope, age,… .

This model was used for different number of agents 3, 5, 7, 11, 13, 17, 19, 31, 53, 73, 97, 127, 151, 211, 257, 307, 401, 503, 701, 1009, 1601, 2003, 3001, 3989 and iterated five times for each agent and the time spent for convergence was recorded .Graph number 3 shows the average time spent for model convergence for a given number of agents .The x axis shows the density of agents and the y axis shows the average time spent for convergence of model .




Graph 3. Average time spent to reach convergence for a given number of agents in Cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugar model.

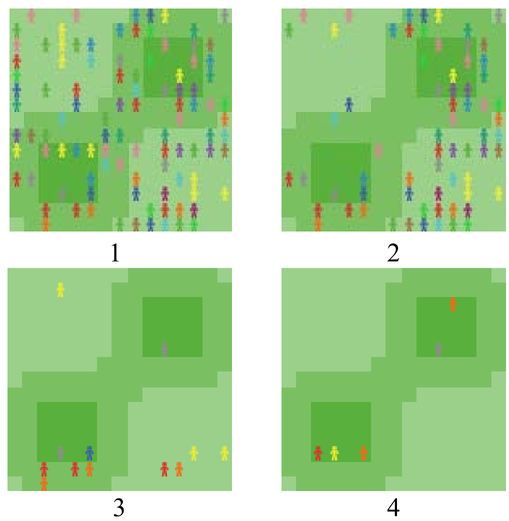



Figure 4. How agents work with use of Boltzmann machine learning algorithm based on sugar model.

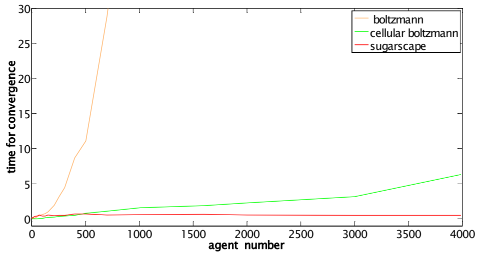
## VI. COMPARISON

Table 1 shows the average execution time to reach convergence over specific number of agents for each model. The average execution time to reach convergence for each model is also shown in the last row of table 1 .Figure 4 compares graph 1 that is the average time spent to reach convergence in model one with graph 2 and 3, which are average time spent to reach convergence in models two and three, respectively .The x-axis shows the density of agents and y-axis shows the average time spent to reach convergence in model .

As seen in table 1 and graph 4, The average execution time to reach convergence in cellular learned multi-agent model based on Boltzmann machine learning algorithm is less than the average execution time to reach convergence in learned multi-agent model based on Boltzmann machine learning algorithm and also the average execution time to reach convergence in cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugar model is much less than the average execution time to reach convergence in cellular learned multi-agent model based on Boltzmann machine learning algorithm .In addition the first model is executable for up to 4000 agents, whereas the maximum for model two is 110000 and for model three is 390000 executable agents .

Table 1. Distribution of average time spent to reach convergence based on model number for different agent numbers

| Agent number | first model | Second model | Third model |
|---|---|---|---|
| 3 | 0.209 | 0.016 | 0.066 |
| 5 | 0.178 | 0.016 | 0.134 |
| 7 | 0.136 | 0.016 | 0.162 |
| 11 | 0.171 | 0.016 | 0.178 |
| 13 | 0.082 | 0.019 | 0.218 |
| 17 | 0.146 | 0.019 | 0.271 |
| 19 | 0.122 | 0.019 | 0.243 |
| 31 | 0.130 | 0.016 | 0.350 |
| 53 | 0.298 | 0.031 | 0.424 |
| 73 | 0.479 | 0.038 | 0.574 |
| 97 | 0.610 | 0.072 | 0.421 |
| 127 | 0.755 | 0.159 | 0.405 |
| 151 | 0.983 | 0.196 | 0.574 |
| 211 | 1.953 | 0.274 | 0.434 |
| 257 | 3.185 | 0.337 | 0.493 |
| 307 | 4.458 | 0.403 | 0.487 |
| 401 | 8.704 | 0.521 | 0.715 |
| 503 | 11.123 | 0.789 | 0.677 |
| 701 | 28.839 | 1.083 | 0.552 |
| 1009 | 70.858 | 1.585 | 0.618 |
| 1601 | 173.603 | 1.869 | 0.630 |
| 2003 | 443.939 | 2.302 | 0.546 |
| 3001 | 1296.425 | 3.145 | 0.521 |
| 3989 | 3141.505 | 6.309 | 0.512 |
| **Average** | **216.204** | **0.802** | **0.425** |




Graph 4. The comparison of the learned multi-agent model based on Boltzmann machine learning algorithm and the cellular learned multi-agent model based on Boltzmann machine learning algorithm and the cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugar model.

## VII. DISCUSSION AND CONCLUSION

In this study firstly the application of Boltzmann machine learning algorithm for learn ability of multi-agent environment was investigated and the average time spent to reach convergence for a given number of agents is shown in graph 1 .It is noteworthy that the considered criterion is the time to reach convergence . The x-axis shows agent density and y axis shows the average time spent to reach convergence .Next the above algorithm was used for learning of cellular multi-agent environment. Graph number 2 which indicates the average time spent to reach convergence is shown .Finally the above algorithm was used in Cellular multi-agent environment based on sugarscape. First a sugar environment was designed and Boltzmann machine learning algorithm was utilized for environment learning .It should be noted that learning in sugar environment takes place by elevation of energy (sugar). Graph 3 shows the average time spent to reach convergence for a given number of agents .As seen in graph number 4 the time to reach convergence in second model is less than time required for convergence in first model . Therefore the second model converges faster .Also the first model is executable for a maximum number of 4000 agents, whereas second model that is cellular can be executed for a maximum number of 110000 agents. Thus not only the second model runs faster than the first model but also a greater number of agents are executable compared with first model .This matter can be justified by the cellular nature of the environment and therefore localized rules and interactions in cellular environment. This is duo to this model is based on localized convergences rather than generalized convergence in model. And also the convergence time in third is much less than convergence time in second model. Also the second model is executable for up to 110000 agents, whereas third model can be executed for a maximum number of 390000 agents. Thus the convergence speed of cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugar model is much more than the convergence speed of cellular learned multi-agent model based on Boltzmann machine learning algorithm. Also the convergence speed of cellular learned multi-agent model based on Boltzmann machine learning algorithm is much more than the convergence speed of learned multi-agent model based on Boltzmann machine learning algorithm. Hence it can be deduced that usage of cellular automata causes faster convergence that is due to local rules and interactions and therefore transformation of generalized convergence to local convergences. The application of sugar model speeds up convergence compared to when it is not applied. Also the maximum number of executable agents in cellular learned multi-agent model with use of Boltzmann machine learning algorithm based on sugar model is much more than cellular learned multi-agent model based on Boltzmann machine learning algorithm and also the maximum number of executable agents in cellular learned multi-agent model based on Boltzmann machine learning algorithm is much more than learned multi-agent model based on Boltzmann machine learning algorithm.

## REFERENCES


[1] Banerjee, B. , Peng, J . Performance Bounded Reinforcement Learning in Strategic Interactions, American Association for Artificial Intelligence, (2004).

[2] Wooldridge, M. An Introduction to multiagent systems, Department of Computer Science, University of Liverpool, UK, (1996), 105-125, 190-220.

[3] Ren, Z., Anumba, C. J. Learning in Multi- Agent Systems: a Case Study of Construction Claims Negotiation, Elsevier, Advance Engineer Informatics, 16, (2003), 265-275.

[4] Alonso, E., d' Inverno, M., kudenko, D., Luck, M., Noble, J. Learning in multi-agent systems, (2001).

[5] Conitzer, V., Sandholm, T. AWESOME: A general multiagent learning algorithm that converges in self – platy and learns a best response against stationary opponents, Springer, Mach Learn, 67, (2006), 23-43.

[6] Miller, J. The Educational spectrum: Orientation to curriculum, New York, Longman Inc, (1993).

[7] Vidal, J. M. Fundamentals of multi-agent systems with NetLogo examples, (2007), 19-38, 61-78.

[8] Nourafza, N., Setayeshi, S., Rahman, A. Identifying models and algorithms learning challenges in multi-agent systems, The first Iranian national conference on software engineering, Islamic azad university roudehen, Iran, (2009).

[9] Ben-David, S. , kushilevitz, E., mansour, Y. Online learning versus offline learning. Machine learning, 29, (1997), 45-63.

[10] Freeman, J. A., Skapura, D. M. Neural Networks Algorithms, Applications, and Programming Techniques, (1991).

[11] Beale, R., Jackson, T. Introduction to Neural Network, (1990).

[12] Haykin, S. Neural Networks a Comprehensive Foundation, (1999).

[13] Krose, B., Smagt, P. An Introduction to Neural Network, (1996).

[14] Epstein, J.M, Axtell R. Growing Artificial Societies: Social Science from the Bottom Up, Brookings Institution Press, Washington DC, (1996).

[15] Fyfe, C. Artificial Neural Networks, (1996).

[16] Anderson, D., McNeill, G. Artificial Neural Networks Technology, (1992).

[17] Fausett, L. Fundamentals of Neural Networks, Architectures, Algorithms, and Application, (1994).

[18] Toma, T. Communication in Artificial Society- Effects of Different Communication Protocols in AN Artificial Environment, Master s thesis, Artificial Intelligence Department of Computer Science, Faculty Of Sciences, Vryje University, Amesterdam, Netherlands , (2003).

[19] Rahman, A., Setayeshi, S., & Zafarghandi M.S. An Analysis to Wealth Distribution Based on Sugarscape Model in an Artificial Society, International Journal of Engineering, Vol 20, Number 3, Tehran, Iran, (2007).

[20] Nourafza, N., Setayeshi, S. Learning based on genetic algorithm and fuzzy genetic, National conference of information Technology, Islamic azad university najafabad, Iran, (2008).

[21] Carmel, D., Markovitch, S.Exploration Strategies for Model-Based Learning in Multi-agent Systems. Autonomous Agents and Multi-Agent Systems, 2, (1999), 141-172.

[22] Busoniu, L., Babuska, R., Schutter, B. D. A Comprehensive Survey of Multi-agent Reinforcement Learning, IEEE Transactions on Systems MAN, AND, Cybernetics- Part c: Applications and Reviews: 38(2), (2008).

[23] Dam, N. Discussion of Learning Algorithms in Stochastic Games. Computer Science Department Iowa State University, Ames, IA 50010

[24] Shoham, Y., Powers, R., Grenager, T. on the agenda(s) of Research on Multi-Agent Learning, Stanford university, (2004).

[25] Chang, Y., Kaelbling, L. P., Playing is believing: the role of beliefs in multi-agent learning, Artificial Intelligence laboratory MIT.

[26] Powers, R., Shoham, Y., Vu, V.A general Criterion and an algorithmic framework for learning in multi–agent systems. Springer, Mach Learn, (2007), 45-67.

[27] Powers, R., Shoham, Y. New Criteria and a New Algorithm for Learning in Multi-Agent Systems, Computer Science Department, Stanford University, Stanford, (2005).

[28] Russell, S. J., Norvig, P. Artificial Intelligence a modern Approach, Prentice Hall, (2003).

[29] Ackley, D. H., Hinton, G. E., Sejnowski, T. J. A learning algorithm for Boltzmann machine, Cognitive Science, 9, (1985), 147-169.

[30] Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. Optimization by simulated annealing, Science, 220, (1983), 671-680.

[31] Nourafza, N., Setayeshi, S., Khadem-zadeh, A. A Novel Approach to Accelerate the Convergence Speed of a Stochastic Multi-agent System using Recurrent Neural Nets, Neural Computing And Applications, Springer, Published online, (2011).

[32] Rahman, A., Setayeshi, S. Implementation of AIDS disease and HIV? virus distribution model in an artificial society as a computation approach to establish the electronic health, In: Proceedings of the first conference on study of IT development approach in Iranian Medical Sciences Universities, Shahid Beheshti University of Medical Science and Health Services, Tehran, Iran, (2007).

[33] Rahman, A., Setayeshi, S. Evolution of social behavior in artificial society, In: Proceedings of the 12th international CSI computer conference, Shahid Beheshti University, Faculty of Electrical and Computer Engineering, Tehran, (2007).

[34] Chen, H. Machine Learning for Information Retrieval: Neural Networks, Symbolic Learning, and Genetic Algorithms, Journal of American Society for Information Science, 43(3), (1995), 194-216.

[35] Rahman, A., Setayeshi, S., Shamsaei, M. Health monitoring in artificial life in the field of contagious diseases and pollution. JHealth Manage, (2008).

[36] Rahman, A., Setayeshi, S. The role of wealth distribution, inheritance and population control in social welfare: simulation of social welfare in artificial society. JSoc Welf. 26, (2007), 183–200.

[37] Jean – Philippe Rennard Ph.d. Introduction to Cellular Automata, (2000).

[38] Allouche, J.P., Courbage, M., Skordev, G. Notes on Cellular Automata , (2001).

[39] Schiff, J.L. Introduction to Cellular Automata, (2005).

[40] Jafarpour, B. Cellular learning automata, Amirkabir University of technology, Computer Engineering and IT department, (2006).

[41] Wolfram, S. A New Kind of Science, (1989).

[42] Wolfram, S. Theory and Application of Cellular Automata. World Scientific, (1986).

[43] Langton, C. G. Computation at the edge of chaos: Phase transition and emergent computation. Phys.D , 42, (1990).

[44] Guinot, V. Modeling Using Stochastic, Finite State Cellular Automata: Rule Interface from Continue Models. Applied Mathematical Modeling, 26, (2002), 701-704.

[45] Carvalho, D., Kelly, C., Tania, T. Probabilistic Cellular Automata Describing a Biological Two Species System. Modern Physics Letters, vol. 18, no. 17, (2004), 873-880.

[46] Rahman, A., Setayeshi, S., & Zafarghandi M.S. Wealth adjustment using a synergy between communication, cooperation, and one-fifth on wealth variables in an artificial society, Springer, 24, (2009), 151-164 .

[47] Rahman, A., Setayeshi, S., & Zafarghandi M.S. Wealth Adjustment in an Artificial Society, Based on a Sugarscape Model Using One Fifth of the Wealth Variable, Iranian journal of electrical and computer engineering, vol.8, no.1, (2009).

**Nasim Nourafza** B.Sc., M.Sc., Ph.D. (Computer software Eng.) is teaching in the Department of Computer Engineering , Islamic Azad University, Najafabad Branch. Nasim Nourafza's research interests are in the areas of Artificial Intelligence, Artificial Life, Multi-Agent Systems, Artificial Society, Knowledge Based Systems, Cellular Automata, Sugarscape Model, E-learning and Dynamics of Complex Systems.

**Saeed Setayeshi**, B.Sc., M.Sc., M.A., M.A.Sc., Ph.D. (Electrical & Computer Eng., TUNS, Canada, 1993) is an Associate Professor and teaching in the Faculty of Nuclear Engineering and Physics, Amirkabir University of Technology (Tehran Polytechnics). He is also presenting some courses in Computer Eng. Department of Res. & Sc. Branch if IAU as an Invited Professor. He has founded AL and Complex System Researches at this branch and has supervised many graduate students in the areas of using CA and LA. He has published more than 100 papers in ISI Journals and Conferences. Dr. Setayeshi's research interests are in the areas of AI & AL, Intelligent Control (Neural-Fuzzy-Expert-GA-CA-LA), Adaptive Signal Processing, Agent Based Modeling, Artificial Society, Social Evolution, Wealth Distribution, Knowledge Based Systems, and Dynamics of Complex Systems.

**Ahmad Khadem-Zadeh** was born in Meshed, Iran, in 1943. He received the B.Sc. degree in applied physics from Ferdowsi University Meshed, Iran, in 1969 and the M.Sc., Ph.D. degrees respectively in Digital Communications and Information Theory & Error Control Coding from the University of Kent, Canterbury, UK.

He is currently the Head of Education and International Scientific Cooperation Department at Iran Telecom Research Center (ITRC). He was the head of Test Engineering Group and the director of Computer and Communication Department at ITRC. He is also a lecturer at Universities in Tehran and a committee member of the Iranian Electrical Engineering Conference Permanent Committee. Dr. Khadem-Zadeh has received four distinguished national and international awards including Kharazmi International Award, and has been selected as the national outstanding researcher of the Iran Ministry of Information and Communication Technology.