International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:7, 2008

Using Emotional Learning in Rescue Simulation Environment

Maziar Ahmad Sharbafi, Caro Lucas, Abolfazel Toroghi Haghighat, Omid AmirGhiasvand, and Omid Aghazade

***Abstract*—**RoboCup Rescue simulation as a large-scale Multi agent system (MAS) is one of the challenging environments for keeping coordination between agents to achieve the objectives despite sensing and communication limitations. The dynamicity of the environment and intensive dependency between actions of different kinds of agents make the problem more complex. This point encouraged us to use learning-based methods to adapt our decision making to different situations. Our approach is utilizing reinforcement leaning. Using learning in rescue simulation is one of the current ways which has been the subject of several researches in recent years. In this paper we present an innovative learning method implemented for Police Force (PF) Agent. This method can cope with the main difficulties that exist in other learning approaches. Different methods used in the literature have been examined. Their drawbacks and possible improvements have led us to the method proposed in this paper which is fast and accurate. The Brain Emotional Learning Based Intelligent Controller (BELBIC) is our solution for learning in this environment. BELBIC is a physiologically motivated approach based on a computational model of amygdale and limbic system. The paper presents the results obtained by the proposed approach, showing the power of BELBIC as a decision making tool in complex and dynamic situation.

***Keywords*—**Emotional learning, rescue, simulation environment, RoboCup, multi-agent system.

I. Introduction

UNPREDICTABLE disasters occur frequently in the world such as floods and earthquakes. Crisis management is vital under such circumstances. After occurrence of such events coordination of the rescuers and their optimal decision making can reduce the depth of calamity. In Robocop competitions we should solve this multi-agent problem in a simulated environment of earthquakes. There are several researches in this field [1, 2]. In all of them the researchers tried to extend the artificial intelligence to rescue simulation environment as a multi agent environments.

In this simulation the buildings collapse which ignite some fires, obstruct the roads and injure the people. There are three groups of rescuers. The fire brigades try to put out the fire, ambulances can rescue injured people from damaged buildings and Police Force agents should clear the blocked road and make it passable for others [3].

Solving coordination in distributed multi agents systems is a difficult problem. The dynamicity of the environment and intensive dependency between actions of different kinds of agents make the problem more complex. We believe that the police have the most important role in coordination between them. If the police don't do its duty ideally the other agents can not reach their goals and do their responsibilities. Thus we first tried to solve the decision making problem for PF agents. One of the most helpful methods of solving problems in multi agent systems is learning [4-6].

In this paper we start with partitioning the environment and assigning each police to one partition to decrease the complexity and dependency. Then any polices try to learn the best action in their territory. The usual reinforcement learning method is too slow for this environment. Our aim is to adapt the decision making system with new environment as soon as possible. The Brain Emotional Learning Based Intelligent Controller (BELBIC) which its applications are extended recently [7-9] is our solution for learning in this environment. This method examined before and its performance lead us to us it in this environment [10-12].

This paper is arranged as follows: In the next part we describe the problem and our static approach. In the third section we explain a summary about emotional learning and in the forth section our algorithm will be described. The results are presented in section five. Section six concludes this paper.

II. Police Agent Problem

The main duty of PF agent is clearing the blocked paths and gathering use full information from the environment for other kinds of agents. The results of each kind of agents' actions affect the others work, in other word their actions are highly-dependent to others work. PF agents clear the blocked road; so the ambulances can have access the injured people and FB agents can reach the fiery buildings. Hence they can improve the efficiency of other kind of agents which influence the score directly.

PF agents like other agents need coordination to achieve the best performance. According to ability and characteristics of the PF agents the best method of coordination is task division.

Manuscript received Manuscript received 2006-04-30. This work was supported by the Azad University of Qazvin Mechatronic Research Lab.

M. A. Sharbafi, is a graduate student in Electrical and computer Engineering at University of Tehran, Iran. (Email: m.sharbafi@ece.ut.ac.ir).

C. Lucas, is with the Center for Excellence and Intelligent processing, Department of Computer and Electrical Engineering, University of Tehran, Tehran, Iran. (Email: lucas@ipm.ir).

A. T. Haghighat, is with the Mechatroni Research Lab, Department of Computer and Electrical Engineering, Azad University of Qazvin, Qazvin, Iran. (Haghighat@qazviniau.ac.ir)

O. Amirghiasvand, is the undergraduate student in Computer and Electrical Engineering, Azad University of Qazvin, Qazvin, Iran. (omid.amirghiasvand@gmail.com).

O. Aghazade, is the undergraduate student in Computer and Electrical Engineering, Azad University of Qazvin, Qazvin, Iran. (omid@6mno.com).

It means every agent have a set of paths that he is responsible to clear them. The PF agents clear paths one by one, but the problem is the sequence of choosing paths for clearing. As we say before path selection has intensive influence on other agents. There are lots of factors that influence agents' decision making to select a path for clearing, like blocked agents in path or distance of the path from the refuges and etc. As a result we can say that task division is so complicated in this environment. Now we are going to explain our approach:

We divide the map into several partitions and allocate each PF agent to one partition. Of course we have 2 or 3 free agents for some special purpose with different priority. Every PF agent calculates a value for each path placed in his partition and determines the paths priority using these values. With these priorities they choose the paths to clear. Of course paths value updated with new information that PF agents find or get from other agents. Now we explain how exactly this value calculates for each path. We use our experience to determine some characteristics of paths that make the path so important. These parameters are: These parameters are: Distance of path to self, Distance of path to nearest refuge, Distance of path to nearest fire, Number of locked by blockade or buried Ambulance Team (AT) agents in path, Number of locked by blockade or buried Fire Brigade (FB) agents in path, Number of buried PF agents in path, Number of locked by blockade or buried Civilians in path and number of requests for clearing that path by other agents.

Then we determine a coefficient (α_i) to any of these parameters. The coefficient shows the importance of each characteristic. Each agent must compute a value between 0 and 10 to the parameters of each path according to his knowledge about the world. The value of ith path calculates by (1).

$$V(P_k) = \sum_{i=1}^{n} \alpha_i c_{ik} \quad (1)$$

where α_i is the coefficient of the i[th] characteristic and c_{ik} is the value of that characteristic of the k[th] path. When this value calculates for all paths in PF agent partition, he selects the path with maximum value and goes to clear it. After clearing the selected path he updates the paths value and selects another path again.

This method works well and has relatively good result but as you may guess it has some weaknesses and problems. Rescue Simulation environment is large-scale and partially known environment with sensing and communication limitations so the agent information about the environment is incomplete and even wrong so paths value calculation maybe contain some mistakes. In fact value that we calculate is an approximation of the exact value. Another problem is determining the coefficient of characteristic. We really don't know that making stress on which characteristic cause to get the best result. There is infinite ways to determine characteristics' priorities. Our first answer to this question was using human experience as we used in MRL team for 2005 competitions and made us the forth team in Osaka2005. As we expected this method has good performance and it is clear from its results in previous matches. But using unvarying coefficient in different situations means that the priorities of the paths are constant. It is definitely wrong, because in one map distance to refugee may be more important than fire regarding to the number of refugees and vice versa. So because the environment is complex and non deterministic, human experience is not trustable. Of course it's possible that in this way we hands on a good approximation. To adapt the method with this dynamic space, we tried using learning agents instead of pre-designed ones.

III. BELBIC

As we described before using we need to a fast learning algorithm and chose BELBIC for this reason. In this section we only describe the formulation of this method. You can find more detailed explanation of this algorithm in [10, 11]. A network model has been adopted, developed by Moren and Balkenius [10], as a computational model that mimics amygdala, orbitofrontal cortex, thalamus, sensory input cortex and generally, those parts of the brain thought responsible for processing emotions. In our utilizations of BELBIC the indirect approach is taken, in which the intelligent system only updates the coefficients of the decision making system. In this section first we describe general aspects of BELBIC and next match it to this problem.

The emotional learning occurs mainly in amygdala. The learning rule of amygdala is given in formula (2):

$$\Delta G_a = k_1.\max(0, EC - A) \times SI \quad (2)$$

Where G_a is the gain in amygdala connection, k_1 is the learning step in amygdale, EC, SI and A are the values of emotional cue, Sensory Inputs and amygdala output at each time. Similarly, the learning rule in orbitofrontal cortex is shown in (3). Inhibition of any inappropriate response is the duty of this block, based on the original biological process.

$$\Delta G_o = k_2.(MO - EC) \times SI \quad (3)$$

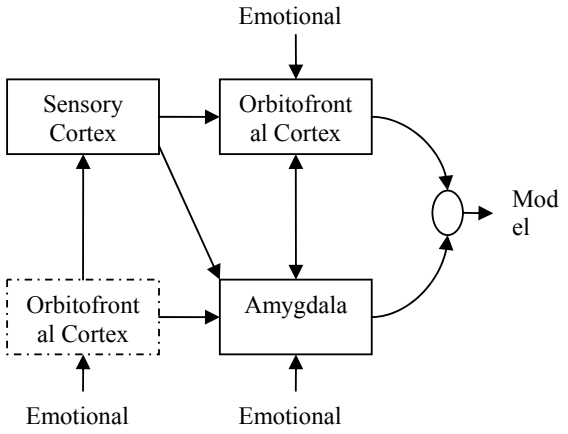

Fig. 1 The abstract structure of the computational model mimicking some parts of mammalian brain

In the above formula, G_o is the gain in orbitofrontal connection, k_2 is the learning step in orbitofrontal cortex and MO is the output of the whole model, where it can be calculated as formula (4), in which, O represents the output of orbitofrontal cortex.

In fact, by receiving the sensory input SI, the model calculates the internal signals of amygdala and orbitofrontal cortex by the relations in (4) and (5) and eventually yields the output MO with (6). Figure 4 shows the structure of emotional system operation introduced in [10]. But as you see in this figure we show the Thalamus with dashed line and it means that we don't use this parameter [8].

$$A = G_a.SI \quad (4)$$

$$O = G_o.SI \quad (5)$$

$$MO = A - O \quad (6)$$

Controllers based on emotional learning have shown very good robustness and uncertainty handling properties [4-6], while being simple and easily implementable. To utilize our version of the Moren-Balkenius model as a controller, it should be noted that it essentially converts two sets of inputs (sensory input and emotional cue) into the decision signal as its output. The emotional cue and sensory input's implemented functions are given in next section.

IV. Second Approach: Learning Method

As it is mentioned before, actually there are two main disadvantages in the previous approach: we can't estimate appropriate coefficients and if we could, it wouldn't be reasonable to use them in every situation. We can solve the first problem with our experience and try and error methods (we can test different values and choose those which give us the best result). In order to solve the second problem we chose learning methods. An example describes these problems more clearly.

Consider a map with these conditions: there are lots of blockades around the fires and around buildings and no agent is locked at the beginning of simulation. We can estimate appropriate coefficients (hereafter we call them α_i s) so PFs can choose and unblock most important paths at the beginning of the simulation. Now consider another map with every AT agent blocked at the beginning of simulation and few fires. Previous Alphas won't be useful in these circumstances. In order to enhance the best results in every map α_i s should be varied according to real time situations during the simulation. In other words PF strategies are determined by α_i s during the simulation which are at first determined by choosing a strategy based on the agent's realities (In RCRSS the data in the world object represent the agent's realities) and are updated based on learning methods.

In our approach we set every α_i to 1 and processed and saved them to the end of each simulation and reused them as initial α_i values for the next simulation. These coefficients are used as the outputs of the BELBIC block. In other word we use the emotional learning to learn the priorities of different parameters of the paths in each map.

As mentioned before, in our approach, at the beginning of each cycle, some parameters are updated for each path and a value is computed from these parameters and path with maximum value is chosen to be cleared. We chose these parameters like the previous approach

The most important part of the learning algorithm is defining the reward function (Emotional cue in BELBIC) that evaluates the performance of our solution. An efficient evaluation method for PF agent should lead to the main goal of simulation (to gain the maximum score) and should be directly related to the PF's activity.

To achieve the first goal we use a function which computes a score that is based on the agent's knowledge which we call Score. For the second goal, we evaluate the Agent's activity with the number of blockades which other agents encounter in a cycle and report them via Messages. Formula (7) evaluates the PF actions.

$$Evaluation\ Parameter\ : Ep = \eta \times Score - \Pr$$
$$\Pr = \beta \times \Pr + Cr \quad (7)$$

In this Formula Cr is the number of reported blockades in current cycle, $\Pr$ is the discounted summation of the reported blockades in previous cycles. This part computes the effect of the previous actions and $\beta < 1$ reduces the role of the prior reports. η determines the portion of the Score in evaluating and tune with the experience of the designer.

The changes in this parameter show the quality of the police's activities. The unrelated events may change Ec and the agent should appreciate if he can improve these changes.

This deduction lead us to use ($\frac{\partial^2}{\partial t^2}$) of the evaluation parameter as shown in (8).

$$Ec = -\gamma \frac{\partial^2 (Ep)}{\partial t^2} \tag{8}$$

The Emotional Cue in this model of the emotion is defined as stress [8]. The increase in stress shows improper result of the agents' action and the minus in (8) is for this. γ is for normalizing this parameter to a reasonable range.

The definition of the Sensory Inputs is obvious. We consider them equal to a vector made of α_i, $\alpha_i C_{ik}$ and the value of the selected path in which k is the index of the selected path (9).

$$SI(i) = [\alpha_i, \alpha_i C_{ik}, V(k)] \tag{9}$$

V. Simulation Results

We examine this method in the map of Virtual City with 2 AT agents, 4 PF agents and 5 FB agents. Fig 2 shows the results in two different runs. The left map is the result of decision making of the agents without learning you can see the score that is 47.9. In this run all of the coefficients are equal to one and this mean that all properties have the same importance. We start with these values of α_i s and the agents tried to learn the best priority, after 5 runs our agents could attain the score of 61.4. In both runs the PF agents could clear most of the roads. But without learning the values of different parameters of the paths are equal and you can see that they didn't open the critical paths that some FB agents are locked in them. Because of their distance to fire sites and refuges their values are less than the others. Table I shows the results during five runs.

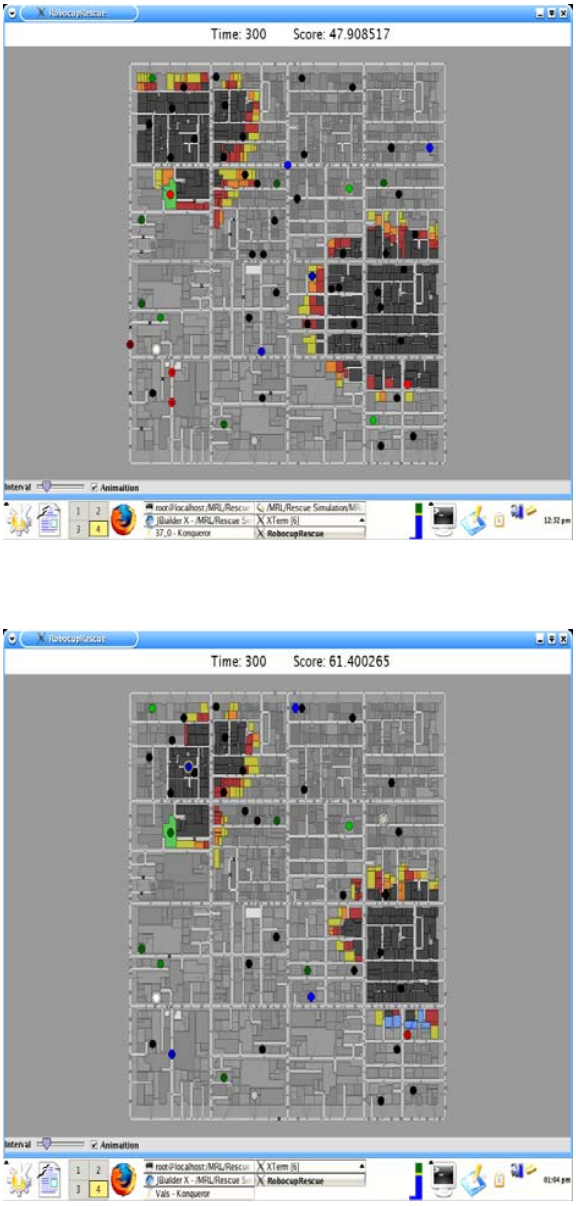

Fig. 2 The upper picture is the result of our first approach with constant coefficients; the other picture is the result of actions of learned agents

TABLE I
The Scores during 10 Runs

Runs	1	2	3	4	5	10
Scores	47.9	49.1	53.5	53.3	61.4	61.9

Although our algorithm for ambulances and fire brigades is consistent their actions may change from run to run. In Table I you can see the results and in forth run the score diminished, but in the next run the increase in the score is considerable and after that the score is about this and after ten runs the score change is not much few. This shows that they can't learn much more.

Fig. 3 shows the coefficients during ten learning run (2 to 11). This result is the average of the coefficients of 4 polices. At the first run which was without learning all coefficients are 1. The invariant coefficient is that of the number of the PF agents, because we didn't have anyone in this map. It is obvious that the importance of requests is the most and number of civilians is the worst. In this map we have many locked rescuers at the beginning and the number of requests should be the most important parameters. Also with only two ambulances after a few preliminary cycles the number of locked civilians decrease, because the ambulances have sufficient injured people for rescue in their civilian sets

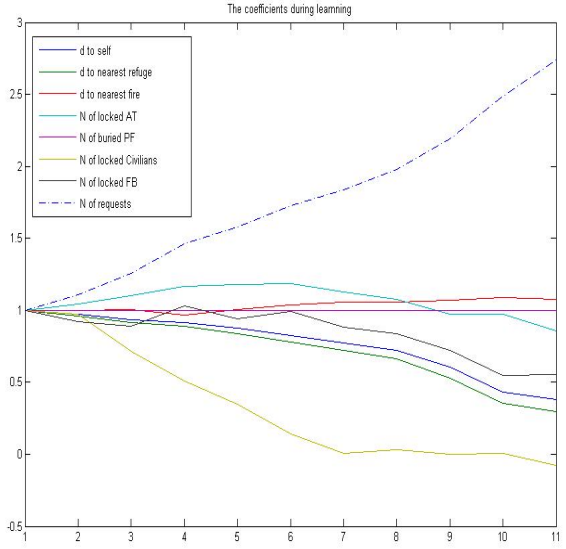

Fig. 3 The coefficients during learning

VI. Conclusion

In this paper the rescue simulation environment was selected as a complex multi agent system. We used a fast and accurate learning method to train our PF agents to choose the best paths to clear. Our emotional learning method gave us good performance. The adaptation power of this method was its most important benefits. Our approach was indirect and it seems that the direct BELBIC controller as a decision maker can be used in this environment too. Usage of similar methods in other agents is suggested.

References

[1] J.Habibi, M.Ahmadi, A.Nouri, M.Sayyadian, M.Nevisi, Utilizing Defferent Multiagent Methods in RobocupRescue Simulation, Robocup2003, 2003.

[2] Kitano, et al.: RoboCup-Rescue: Search and Rescue in Large Scale Disasters as a Domain for Autonomous Agents Research, IEEE Conf on Man, Systems, and Cybernetics (1999).

[3] Robocup-Rescue Simulation Manual, The Robocup Rescue Technical Committee, 2000.

[4] Michael Wooldridge, Nicholas R. Jennings, “Intelligent Agents: Theory and Practice”, The Knowledge Engineering Review, 10:115-152 1995.

[5] Alan H. Bond and Les Gasser. An analysis of problems and research in DAI. In Alan H. Bond and Les Gasser, editors, Readings in Distributed Artificial Intelligence, pages 3–35.Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[6] Peter Stone_ Manuela Veloso, “Multiagent Systems: A Survey from a Machine Learning Perspective”, In Autonomous Robotics volume 8, number 3. July, 2000.

[7] R. M. Milasi, C. Lucas, and B. N. Araabi. Speed Control of an Interior Permanent Magnet Synchronous Motor Using BELBIC (Brain Emotional Learning Based Intelligent Controller). In M. Jamshidi, L. Foulloy, A. Elkamel, and J. S. Jamshidi (eds.), Intelligent Automations and Control- Trends, Principles, and Applications. Albuquerque, NM, USA: TSI Press Series: Proceedings of WAC, 16, M. Jamshidi (series editor), 2004. 280- 286.

[8] R. Mohammadi Milasi, C. Lucas, and B. N. Araabi. A Novel Controller for a Power System Based BELBIC (Brain Emotional Learning Based Intelligent Controller). In M. Jamshidi, L. Foulloy, A. Elkamel, and J. S. Jamshidi (eds.), Intelligent Automations and Control- Trends, Principles, and Applications. Albuquerque, NM, USA: TSI Press Series: Proceedings of WAC, 16, M. Jamshidi (series editor), 2004. 409- 420.

[9] G. Zandesh, J. Moghani, C. Lucas, D. Shahmirzadi, B.N. Araabi O. Namaki, H. Kord. Speed Control of a Switched Reluctance Motor Using BELBIC. WSEAS Transactions on Systems, 3(1), January 2004, 1-7.

[10] J. Moren, C. Balkenius. A Computational Model of Emotional Learning in The Amygdala: From animals to animals. in Proc. 6th International conference on the simulation of adaptive behavior, Cambridge, Mass., The MIT Press, 2000.

[11] C. Lucas, D. Shahmirzadi, N. Sheikholeslami. Introducing BELBIC: Brain Emotional Learning Based Intelligent Controller. International Journal of Intelligent Automation and Soft Computing, Vol. 10, No. 1, 2004, pp. 11-22.

[12] M. Fatourechi, C. Lucas, A. Khaki Sedigh. Emotional Learning as a New Tool for Development of Agent based System. Informatica, 27(2), June 2003, 137-144.