Collective behavior in multiagent systems based on Reinforcement Learning

Kabysh A.S. [1), Golovko V.A. [2)
1) BrSTU, Postal address, Anton.Kabysh@gmail.com
2) BrSTU, Postal address, gva@bstu.by

***Abstract:**. In this work would be considered multiagent approach to solve intellectual tasks based on Reinforcement Learning. The multiagent approach involves agent teamwork for the solution of the problem. But classical RL supports for single agent learning. Therefore, the key question is how to modify the reinforcement learning for a group of interacting agents. In this work would be considered a modified algorithm for supporting training for a group of agents. As an example of the problem we will take a coordinated movement in the space group of agents. As a result of training were observed interesting patterns of behavior of groups, such as «leader», «chain of action», «clustering».*

***Keywords*:** Agent, Multiagent System, Reinforcement Learning, Multiagent Reinforcement Learning, Coordinated Movement.

1. MULTIAGENT APPROACH

Multiagent approach - it is an entire paradigm in the development of complex systems consisting of interacting autonomous agents, which are operating with local knowledge and limited capacity, however, can be expected, in general, the behavior of the system. To create a multi-agent system, we should take a description of what the system should do and express it in the behavior of individual agents [1]. This idea shows the beauty of multiagent approach is to create emergent properties through the interaction of the system of agents [3], so a lot of ants form formic colony, the many people across the globe creates a global economy, and the interaction of cells creates a living organism. Main applications of multiagent-approach - it is modeling in the economies of biology and sociology. For multiagent systems are established standards (FIPA) and languages (KQML and ACL) [3], developed new directions, such as hierarchical multi-agent system.

Multiagent approach is used for researching in area of collective behavior, because multiagent system is based on the interaction of agents. Collective behavior can have various manifestations, such as this could be the formation of coalitions in addressing economic problems, or the collective movement of autonomous agents, a joint study of the environment to exchange information. The key to achieving a collective behavior is the organization of interaction between agents [1,2,6]. We show how to organize collective behavior on the basis of reinforcement learning by the example of a collective movement of groups of agents.

2. REINFORCEMENT LEARNING

Reinforcement learning is a popular method of machine learning. They often used for learning autonomous agents [4]. It emerged at the intersection of dynamic programming, machine learning, biology, studies the reflexes and reactions of living organisms [4]. The basic model presented in the work of Sutton and Barto [5], is shown in Fig. 1.

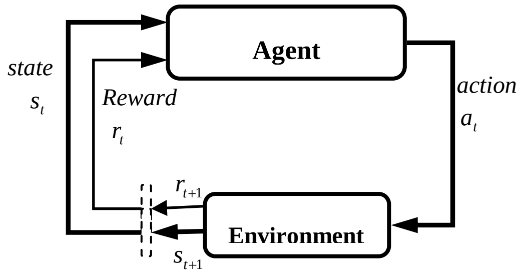

Fig.1 - The classical model of reinforcement learning.

Agent (in the same RL-Agent), or any other entity interacts with the external environment in discrete moments of time $t = 0,1,2,3,....$. Each time, the agent receives some representation of the external environment, the *state* $s_t \in S$, where S - is the set of all possible states. Based on the current state *action* is selected $a_t \in A(s_t)$, where $A(s_t)$ the set of possible actions in the state s_t. One time step later, in part as a consequence of its action, the agent receives a numerical reward $r_{t+1} \in R$, and is moving into a new state s_{t+1}. At each time step, the agent implements a mapping from states to probabilities of selecting each possible action. This mapping is called the agent's *policy* and is denoted π, where is $\pi(s,a)$ the probability that $s_t = s$ if $a_t = a$. Reinforcement learning methods specify how the agent changes its policy as a result of its experience. The agent's goal is to maximize the total amount of reward it receives over the long run.

For each state-action pair agent is trying to determine the value, which is denoted as $Q(s,a)$. Using the temporal difference method [5], we can iteratively update $Q(s,a)$ value as follows:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \gamma Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)] \qquad (1)$$

This update is done after every transition from a nonterminal state s_t. If s_{t+1} is terminal, then $Q(s_{t+1},a_{t+1})$ is defined as zero. This rule uses every element of the quintuple of events $(s_t,a_t,r_{t+1},s_{t+1},a_{t+1})$ that make up a transition from one state-action pair to the next. This quintuple gives rise to the name *SARSA* for the algorithm. Another main learning algorithm *Q-Learning* described below:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t)] \qquad (2)$$

To store the value function we can use tabular view and functional approximation. Function approximation is an instance of *supervised learning*, the primary topic studied in machine learning, artificial neural networks, and pattern recognition. The main goal of function approximation – it is *generalization* of state-action space. In this case, using the neural network with one hidden layer. Number of output neurons equals the number of actions the agent. An error of approximation is temporary difference error. A study only neuron, which action was selected. Neural Network learning rule are discussed in detail in [6].

3. COLLECTIVE REINFORCEMENT LEARNING

Now we present a model of a reinforcement learning generalized to the case of multiagent system [6]. The group of agents operating in a common environment. We need to describe multiagent learning in terms of Reinforcement Learning. The essence of the collective approach in considering the group of agents - as a single entity, while at the same time not forgetting about the local relevance of each agent [6]. Multiagent system reacts with environment as a single organism, providing a distribution of states for the local agents and the collection action from them. The modified model is shown in Figure 3.

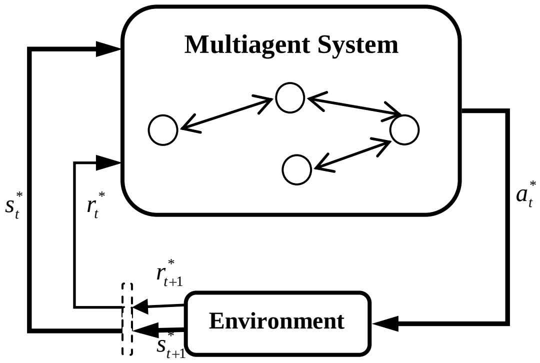

Fig 3. The model of collective reinforcement learning.

Multiagent system of N agents interacts with the external environment in discrete moments of time $t = 0,1,2,3,....$. Each time, the MAS receive some representation of the external environment, the global state s_t^*, which describes the status of each agent in the environment. The *i-th* agent perceives the global state s_t^* locally, in accordance with its sensors (filters) f^i.

$$s_t^i = f^i(s_t^*), \tag{3}$$

Agents are communicated to each other local connections. Such agents are called cooperate agents. The cooperation of agents involves the exchange of information with one another. As information can be a value the current state, past the selected operation and etc. This allows the agent in learning use the information from neighboring agents.

Each agent based on the current state and information from cooperate agents select some action a_t^i. At the current time, the system collects all agents action and creates aggregate action (global action) by combining all local actions.

$$a_t^* = \{a_t^1, a_t^2, ..., a_t^N\}, \tag{4}$$

MAS set and execute global action into environment. One time step later, the multiagent system receives a global numerical reward r_{t+1}^*, and is moving into a new global state s_{t+1}^*. The reward charged to all agents equally, without local perception.

Algorithm collective Reinforcement learning shown below:

1) MAS get a description of the state of the environment.

2) MAS send the state to each of its agent.

3) Each agent perceives the state locally, in accordance with its sensors;

4) Each agent determines the action that he was going to execute.

5) MAS collect all the individual actions in one aggregate action a_t^*, and perform it in the environment.

6) The environment computing a new state s_{t+1}^* and sends it along with the state reinforcement r_{t+1}^*.

7) MAS inform of all agents receive reinforcements, to follow what agents can learn individual by some RL algorithm.

8) Go to step 1.

Agents make set actions and learning locally. But the multiagent system consisting of these agents live as a single organism in environment.

4. TASK

The world has a kind of two-dimensional grid of 100 by 20 with periodic boundary. In the beginning, on the grid randomly located $N = 5$ agents. The agent, which is the right of others on the grid, it is leading for other agents. Agents are fully unlearned.

Agent with neighbors linked by local cooperative connections, as shown in fig. 4. Agents are connected to each other local connection that means cooperating between agents. These links refer to the exchange of information between agents.

The agent can make the transition to any neighboring cell, if it is not occupied by another agent, or can do nothing and remain in current place. If two agents are on one line, the old location remains.

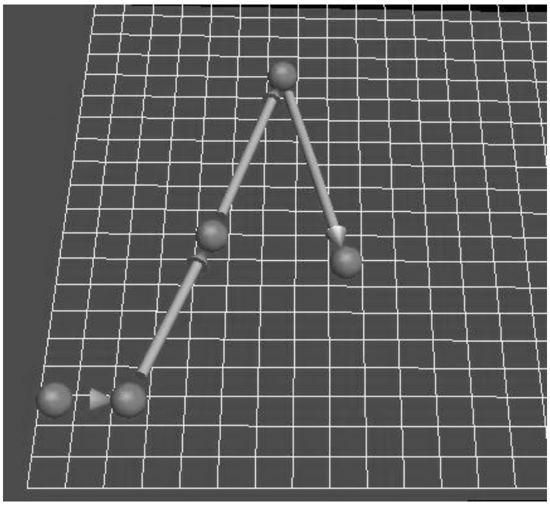

Fig.4 - Initialized model.

The agents make moves in turn from right to left. First – leader (rightmost agent), second - left from leader, third

– left from second and etc. At one time step, all agents choose the one action, and learning by reinforcement learning.

The goal of the experiment - to teach the agents of the collective movement in space. The collective movement means that the agents will move strongly in one direction repeating the steps of each other. The distance between the agents should be minimal. We define the goal function of the distance between agents, which will be minimized. It is also a reward function.

$$r(t)=\sum_{i=0}^{N}(x_i(t)-x_{i-1}(t)+p+abs\,|\,y_i(t)-y_{i-1}(t)\,|), \qquad (5)$$

where $x_i(t)$ – x -coordinate of *i*-th agent on grid at time step *t*, $y_i(t)$ – y -coordinate of *i*-th agent on grid at time, p – offsetting factor is used because on the X-axis distance between the agents will always be negative.

Reward value for agents is calculated as the sum of differences of coordinates consistently associated agents. The meaning of reinforcements that would be it was a maximum when the agents maintain the desired formation.

After initializing, the model began free simulation. The agents were totally free in their environment and did not receive any data point, except state description and reinforcement. The function of reinforcement has been described above, but now considers the state description. It include: The current position in the grid - the coordinates x, y; last action of the lead agent; distance of the x, y coordinates to its lead.

5. RESULT OF EXPERIMENTS

We perform two experiments on this model in order to compare the two methods of RL - tabular and function approximation. In the tabular case, we use a table to store the value of state-action pairs and use a Q-Learning algorithm because process of convergence is guaranteed, and is much faster than other algorithms. Note that a necessary element of convergence in this problem is to use eligibility traces in learning. For functional approximation, we use neural network learned by RL SARSA algorithm. He strongly converges, but the speed of training is unknown in advance. In neural network case we use space tiling for better and faster convergence [5].

Comparison of the dynamics of reward shown in the diagrams of fig. 5.

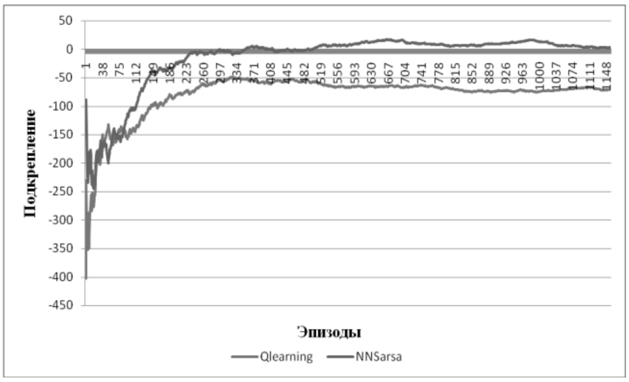

Fig.5 - Dynamics of reward (y-axis) for episodes(x-axis).

NN-SARSA show optimal result then Tabular Q-Learning.

Q-Learning has never been able to achieve optimal behavior, perhaps because of the large state-action space. SARSA method based on neural networks has shown satisfactory results in the simulation blending.

In the first steps of modeling the behavior of agents looks chaotic. However, in future, there is a general tendency for self-organization of optimal location of the agent in the formation. Agent tries to achieve maximum of reinforcement function.

It can be assumed that, depending on the defined rules of the relationship between (type of cooperation) agents and the reinforcement function agents will form a different pattern of behavior. Try to identify these patterns in the current model.

The first pattern - is the optimal behavior of the multiagent system. Agents follow the leader at some distance. There are failures, when the agent drops out of the chain with the subsequent recovery.

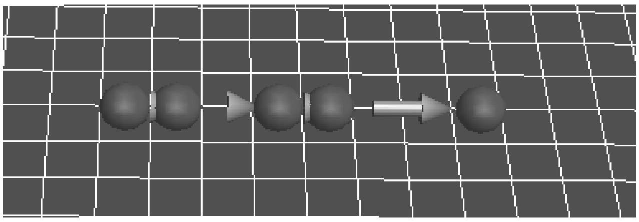

Fig.6 - Optimal behavior of the system.

Second pattern "chain of actions" shown in fig. 7. This is also a normal behavior of the model. Characterized as a chain of similar actions performed by agents at one time step. For example, if an agent commits a leading left, the next it just makes left. There is a consistent echo of reactions agents followed the lead. There is a correlation between value of the reward function and actions selected by agent-leader.

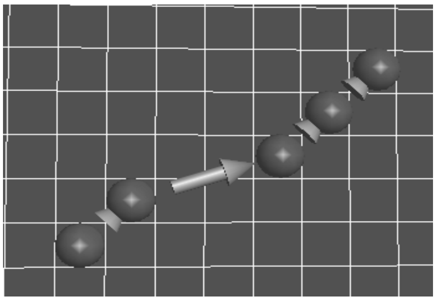

Fig.7 – Chain of actions

The third pattern - this is a bad trained system. The agents gather in a group and do not move. This pattern is possible because it satisfies the reinforcement function. The probability of occurrence of this pattern is highly dependent on the behavior of the agent-leader and his starting position. To avoid this pattern, in reinforcement function was introduced «repulsive factor», reduces the function of reinforcement, if agents were very close to each other.

A large number of parameters of neural networks and reinforcement learning on the one hand an opportunity to settings required by the model results, and on the other hand requires a lot of experiments to determine the optimal configuration.

5. CONCLUSION

The presented model describes the basis for the collective behavior of the MAS on the basis of reinforcement learning. It was shown that, depending on the reward function and description of the state behavior of agents can take a variety of patterns

6. REFERENCES

[1] *Hose M. Vidal. Fundamentals of Multiagent Systems with Net Logo Examples.* (www.multiagent.com)

[2] *Jurgen Schmidhuber. A General Method For Incremental Self-Improvement and Multi-Agent Learning in Unrestricted Environments*, ISDIA, (www.isdia.ch/~juergen)

[3] *On agent-based software engineering.* Jennings NR 2000, Artificial Intelligence.

[4] *From animal to animat.* Редько В.Г. От моделей поведения к искусственному интеллекту. Серия «Науки об искусственном» (под ред. Редько В.Г.). М.: УРСС. 2006.

[5] *Sutton R., Barto A. Reinforcement Learning: An Introduction.* Cambridge: MIT Press. 1998. (http://www.cs.ualberta.ca/~sutton/book/the-book.html)

Kabysh A.S., Golovko V.A. Collective behavior in multiagent systems based on Reinforcement Learning. Neuroinformatics 2009, p. 191-201

[6] *Franchesco de Comite. A Platform for Implementation of Q-Learning Experiment. Reference.*