Acquisition of Deterministic Exploration Behavior by Reinforcement Learning

Katsunari Shibata
Dept. of Electrical and Electronic Engineering, Oita University, 700 Dannoharu, Oita 870-1192
shibata@cc.oita-u.ac.jp

Abstract
Exploration is an important factor that influences the performance in reinforcement learning, and random factors are usually used to realize it. However, the exploration that real lives are doing does not seem just random actions, but seems a kind of deterministic and intelligent actions using their knowledge and considering the context. In this paper, the author tries to explain such explorations as a deterministic behavior and propounds a novel approach that effective exploration is acquired by reinforcement learning. It is shown that an agent with a recurrent neural network trained based on reinforcement learning becomes to explore effectively to some extent in some simple problems.
Keywords: exploration, reinforcement learning, recurrent neural network, context

1. Introduction

Reinforcement learning is an autonomous and purposive learning based on reward and punishment by trial and error. The trial and error is usually called "exploration" and it is an important factor that has a large influence on the performance of reinforcement learning. Usually, it is realized by stochastic action selections using random numbers, such as ε-greedy or Boltzmann selection[1]. However it is difficult to decide the ratio of the random factors in the action selection due to the "exploration-exploitation dilemma".

This research was triggered by the simple questions as "Do we take stochastic actions?" and "Is there a random number generator in our brain?" For example, if we are put in an unknown maze, we try not to pass the same way as we have already passed, and try to imagine a two-dimensional map in our brain. In other words, we explore the maze using the knowledge we have and considering the context. When we are at a branch, we don't move our fingers, jump, or go to the midway of the two paths, and choose one of the two paths. From this fact, our "exploration" is not just random actions, but seems very intelligent actions. When we don't have any idea about which way we should go, we sometimes shoot a dice. However, even in such a situation, the decision itself can be considered to be deterministic.

Against the above interpretation, the following discussions can be accepted. One of them is that that is the result of a stochastic action selection with a high probability for the action with a high action value after representing appropriate action values using knowledge and/or context. Another is that the exploration is not done at each actuator level, but is the result of the stochastic action selection on higher abstracted space. Actually, abstraction of action space has been discussed as temporal abstraction[2].

It is certainly difficult to deny the possibility of the existence of stochastic action selection. However, there seems no rational reason why stochastic action selection must be employed except for "the concern that an agent cannot explore all the unknown states without stochastic factors" or "easiness of statistical analysis". When our "exploration" mentioned above is considered, it is thought that exploration can be realized without stochastic factors. If exploration is a result of deterministic decision making, it is considered as one of the actions in a wide meaning. Then it is expected that appropriate explorations can be learned through experiences as well as the other functions. The author has been thinking that the reinforcement learning is useful not only for the learning of actions but for the learning of a variety of functions including recognition and memory, and plays an important role in the emergence of our intelligence as living creatures[3].

According to the above discussion, in this paper, based on the hypothesis that "exploration" by real lives is not realized by stochastic action selections, but a kind of deterministic ones, it is propounded that the actions that can be interpreted as exploration are acquired by reinforcement learning. However, since the learning without using random number generator is another ongoing research subject, this paper is focused only on the acquisition of non-random exploration and random numbers are used for learning. Since context is useful to realize effective "exploration" as mentioned above, a recurrent neural network is used[4]. Two kinds of very simple tasks are picked up, and the acquisition of exploration behavior is examined. In one of them, the goal cannot be known explicitly, but effective exploration using knowledge and context is required. The other task is mainly focused on the acquisition of knowledge by explorations.

2. Learning

Here, Elman-type recurrent neural network whose hidden outputs are fed back to the input layer at the next time step is employed as the most general recurrent neural network. The present observation signals s_t are the input of the network. The number of output neurons is the same as the number of actions, and each output is used as Q value for the corresponding action. After forward computation of the output for the previous input signals, the training signal $Q_{s,a_{t-1}}$ for the Q value of the previous action $Q_{a_{t-1}}(s_{t-1})$ is generated autonomously based on Sarsa algorithm[1] as

$$Q_{s,a_{t-1}} = r_t + \gamma Q_{a_t}(s_t) \tag{1}$$

where r is a given reward and γ is a discount factor.

The training signal is given only to the corresponding output, and the network is trained by BPTT (Back Propagation Through Time)[5]. The output function used in the hidden and output layer is the sigmoid function whose value ranges from -0.5 to 0.5, so the output of the network after adding 0.4 is used as Q value, and the training signal is used after subtracting 0.4 from the signal generated by Eq. (1).

The learning is done independently for each episode (trial). Before each episode, all the hidden outputs are reset to 0.0, and all the Q values after reaching a goal are 0.0. When the agent reaches the goal, it can get a reward $r = 0.8$, and otherwise $r = 0.0$ always.

3. Simulation

3.1 Exploration on branch situation

At first, the branch situation is simulated in which only one of two marked states among many states is the real goal, but there are no information about which is the real one between the two marked states. As shown in Fig. 1, an agent is located at the center of the 5x5 grid world, and two landmarks are put randomly on two states on the four sides of the world. The real goal is chosen randomly from the two marked states, but the agent does not know that. The two points that should be observed are whether the agent can go to one of the marked states by its intention and also whether the agent can change its moving direction after reaching the state and finding that it is not the real goal.

The agent can choose one of the four actions, moving up, right, down, and left, and the state transition is deterministic. If the agent hits against a wall, it stays at the same state. The agent can observe whether the state is marked or not for each of 9x9 grid world whose center is fixed at the agent location. This enables the agent to catch the both landmarks wherever the agent exists. The input signals to the network are 9x9=81 binary signals. The neural network has three layers, and the number of hidden neurons is 20. The maximum time steps traced back through time for BPTT is 30. ε-greedy is used as stochastic action selection during learning. ε is fixed at 0.1 and discount factor γ is 0.92. The initial weight for each hidden-output connection is 0.0 and that for each non-feedback input-hidden connection is chosen randomly from -0.5 to 0.5. For the self-feedback connections, the initial weight is 4.0, while that for the other feedback connections is 0.0. The learning constant for BPTT is 0.2.

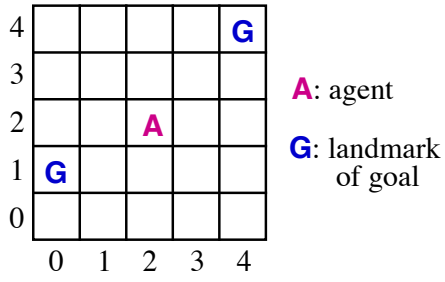


Fig. 1 The task in which there are two landmarks of goal, but no information about which is the real goal.

At the early phase of the learning, even though the agent reached one marked state, it was difficult to go toward the other marked state. Sometimes it took several thousands of steps to reach the real goal, but the number decreased as the learning progressed. After learning of 100000 episodes, the agent still failed to reach the goal for 5 combinations of real and fake goal locations, but it successfully reached the real goal 251 combinations among the total of 256 combinations.

Successful sample behaviors for 4 combinations are shown in Fig. 2. It is seen that the agent heads to one of the marked states at first, and then after finding the state is not the real goal, it changes its moving direction to the other marked state even though one redundant action is seen just after reaching the first marked state in the case of Fig. 2 (a) (b). In these cases, the distance from the center to each marked state is the same with each

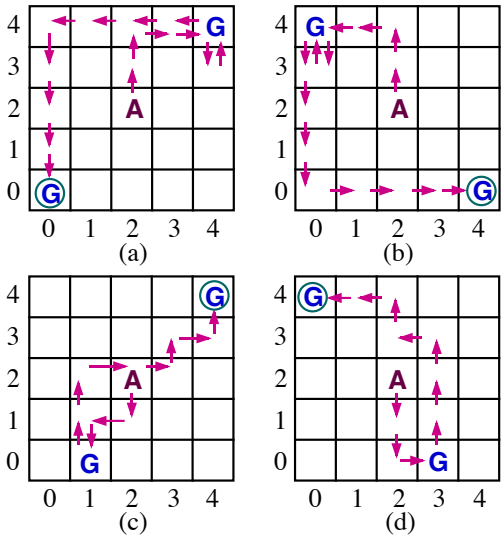


Fig. 2 The behaviors after learning. 'G' indicates the landmark of goal and circled one indicates the real goal.

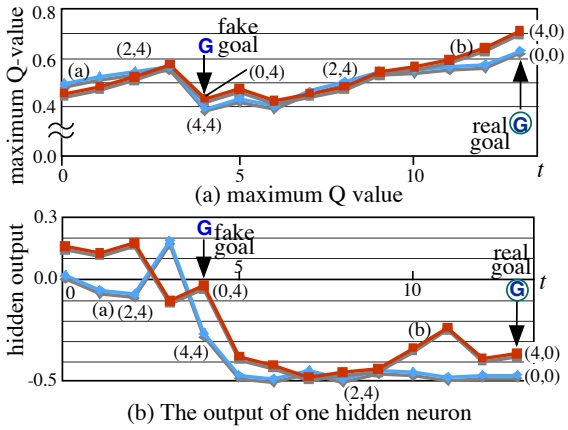


Fig.3 The change of the maximum Q value and one hidden neuron's output in one trial for the case of Fig.2 (a) and (b). 'G' indicates the landmark of goal and circled one indicates the real goal.

other, but when one of the marked state was moved closer to the center, it visited to the closer marked place at first as shown in Fig. 2 (c) (d) with some exceptions.

Fig. 3 shows the change of the maximum Q value and the output of one hidden neuron in one trial for the both cases of Fig. 2 (a) and (b). It is seen that the maximum Q value increased as the agent approached to the marked place, but once it found the marked state was not the real goal, the Q value went down. However, it increased again as the agent approached to the other marked state. In the case of Fig. 2 (a), it passed on the state (2,4) twice, and the observation is completely the same between before and after reaching the fake goal at the state (4,4). Nevertheless, the Q value on the state (2,4) is 0.54 for the right move and 0.45 for the left move before reaching the fake goal, while it changes to 0.42 for the right move and 0.50 for the left move. This means that the recurrent network extracted and kept the information that the first marked state was the fake goal, and the network reflected the information to the Q value. The hidden neuron in Fig. 3 (b) took the value around 0.0 at first, but once it reached the fake goal, the value went down around -0.4. For the other combination of the marked states, the same tendency can be seen in this neuron. It can be considered that the neuron is representing the passing of the fake goal.

3.2 Exploration without information of goal location

A mouse in a maze explores effectively without any information about food location. There appears to be a problem whether such exploration behavior can be acquired by reinforcement learning. Then this section is focused on the exploration without information of goal location, and it is examined whether actions that can be called as exploration can be acquired through learning.

As shown in Fig. 4, five 2x2 mazes are prepared. Four of them have one wall inside, and the other has no walls. One of the five is chosen randomly at every trial, and the agent is located randomly at one of four states. The agent can take one of the five actions each of which is move to one of the four directions or stay at the same state. The state transition is deterministic and if the agent chooses the action to move against a wall, it stays at the same state. A total of 17 observation signals can be divided into four kinds. The first four signals indicate whether a wall exists in one of the four directions, and other eight signals indicate whether the goal exists at each of eight neighbor states. The other 5 signals indicate the previous action that is one of the five actions. Each signal is binary signal and is put into the recurrent network. At the beginning of a trial, all the observation signals representing previous actions are 0. Action selection method is ε-greedy and ε is 0.1 at the beginning of learning, and is gradually decreased to 0.0 by a constant value until the end of the learning. This means that the agent selects its action greedy at the end of the learning. The number of layers in the network is four and each of which has 17 (except for the feedback input), 20, 10, 5 neurons respectively from the input layer to the output layer. All the outputs of the

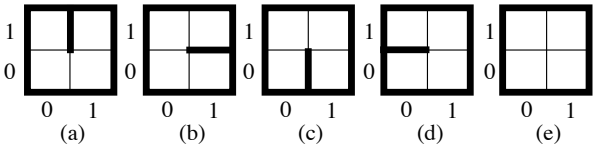


Fig. 4 Five 2x2 mazes used in the simulation

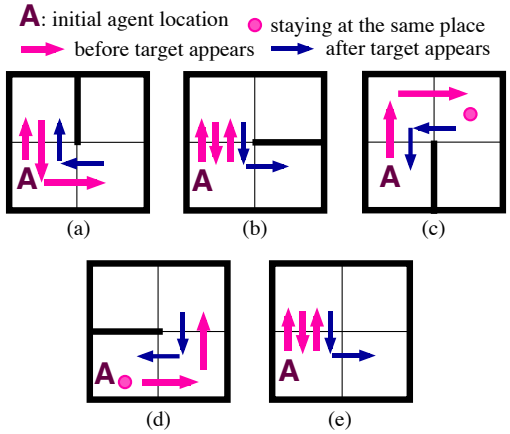


Fig. 5 Acquired exploration behaviors before and after the goal appears.

upper hidden neurons are also the input at the next time step together with the 17 external input signals. The network is trained by BPTT with the training signal as Eq. (1). The number of time steps traced back to the past is 10 here. The initial weight value is 0.0 for all the connections from the upper hidden layer to the output layer, and that for the other connections is chosen randomly from -1.0 to 1.0. The discount factor γ is 0.9 and learning rate for BPTT is 0.2.

The goal appears randomly one of the three states where the agent does not exist after three state transitions by the agent action. For example, if the agent is located initially at (0, 0) and no wall is detected, it cannot identify whether the maze is (a), (b) or (e). Accordingly, unless it moves, it cannot know whether it should go up or right when the goal appears on the diagonal state (1, 1). However, if the agent moves beforehand and identifies the maze, it can go to the correct direction even though the goal appears on the diagonal state. Furthermore, if the agent exists at the next of the inner wall when the goal appears, it has to move three times to reach the goal if the goal was put behind the inner wall. Accordingly, it should be on a state that is not next to the inner wall just before the appearance of the goal and should remember where is the inner wall.

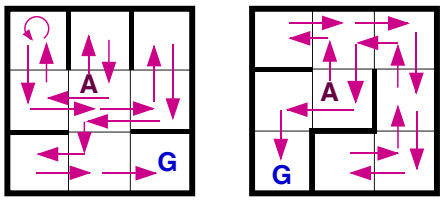
The average time steps from the appearance of the goal for 1000 trials after 100000 trials of learning is 1.35 for the four-layer Elman network case, 1.55 for three-layer Elman network case, and 1.57 for four-layer regular network case. The optimal number of expected steps is 1.33. The behavior of the agent using four-layer Elman network is shown in Fig. 5. In this figure, the initial agent location is (0, 0), and a thick arrow or point

indicates the action before the appearance of the goal, and a thin line indicates the action after the goal appeared on the diagonal state of the agent. In all the cases, the agent existed on the state that is not next to the inner wall after some actions before the appearance of the goal, and when the goal appeared on the diagonal state, it could go to the way that the inner wall did not block. Comparing with (a) and (b), the agent went up to (0, 1) at first, and then returned to (0, 0) in the both cases, but the third action is to go right in the case of (a), while to go up in the case of (b) even though the observation signals are the same between them. This means that the recurrent network memorized whether the inner wall existed or not on the state (0, 1), and the memorized information was reflected to the third action.

Finally, in order to examine whether the state value increases by knowing the maze shape and it promotes the exploration behaviors, the maze shape is made to be more complicated and the sight of the goal is completely deprived. Here, the shape of 3x3 maze such as shown in Fig. 6 is decided randomly at every trial, and the agent is put on the center of the maze. The goal is also located randomly, but the agent cannot know the location. The agent's observation has eight signals. Four of them represent the wall presence in each of 4 directions, and the other four represent the previous action. The Elman network has three layers, and the number of hidden neurons is 30. The initial weight values are the same as in the previous simulation. The learning rate for BPTT is 0.01, and the number of time steps traced back to the past is truncated at 30 steps. ε in the ε-greedy is 0.1, and the discount factor γ is 0.9.

Fig. 6 shows the learning result. Even though the maze shape changes randomly at every trial, the exploration behavior to go everywhere effectively can be obtained as shown in the figure with some exceptions. In this case, since there is certainly a goal in the maze, if exploration progresses and the number of non-visited states becomes small, the possibility to reach the goal at the next time step becomes large when it goes to a non-visited state. Accordingly, it is expected that the recurrent network learns the fact, and the maximum Q value that is the state value increases gradually according to the progress of exploration. However, the increase of the Q value could not be observed.

Then the goal location was limited at one of four corners. The learning of exploration behavior could not be obtained. For example, in the case of Fig. 6 (a), if the goal is located on one random state on the four sides, there is a possibility to reach the goal whatever direction the agent goes to. However, if the goal is located only on one of the four corners, there is no possibility to reach the goal at the next time step from the center of the maze. That might be the reason that the agent could not learn an appropriate exploration behavior such that the agent goes to the left since it has been to the right before. If the recurrent network memorizes the information about all the visited states in the maze, a proper state value function might be obtained. The recurrent network is difficult to learn the function of a



(a) (b)

Fig. 6 Exploration behavior after learning in the case of no sight of the goal.

counter or multiple state transitions. The problem might be originated from the learning ability of the recurrent network. However, hereafter, more detailed analysis is necessary to know the cause correctly.

4. Conclusion

It is propounded that by not treating "exploration" in reinforcement learning as simple stochastic actions but treating it as a kind of intended actions, effective "exploration" is acquired by reinforcement learning. This idea was introduced in the task that the goal information is not clear and also in the task in which "exploration" is required with no information about the goal location, and effective "exploration" can be obtained to some extent by reinforcement learning.

It is thought that this idea is deeply relating to the temporal and spatial "abstraction" and also to the "curiosity" that has a possibility to make actions emerge without reward. It is desired to construct a system to explain these items consistently.

Acknowledgement

This research was supported by JSPS Grant in Aid in Scientific Research (B) #14350227and #15300064, and also Overseas Advanced Education and Research Application Assistance Program. I would like to say thank to Prof. R. S. Sutton at Univ. of Alberta, Canada for useful comments.

Reference

[1] Sutton, R.S. and Barto, A.G., "Reinforcement Learning: An Introduction", MIT Press, Cambridge, MA (1998)

[2] Sutton, R.S., Precup, D., and Singh, S., "Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning". *Artificial Intelligence 112*:181-211 (1999).

[3] Shibata, S., "Reinforcement Learning and Intelligence of Robots", Proc. of the 16th Annual Conf. of JSAI, 2A1-05 (2002) (in Japanese)

[4] Shibata, K. and Sugisaka, M., "Dynamics of a Recurrent Neural Network Acquired through the Learning of a Context-based Attention Task", *Artificial Life and Robotics*, Vol. 7, pp. 145-150, (2004)

[5] Rumelhart, D.E, Hinton, G.E., and Williams, R.J., "Learning Internal Representations by Error Propagation". *Parallel Distributed Processing,* The MIT Press, pp. 318-362 (1986).