Demonstration of Learned Helplessness with Fuzzy Reinforcement Learning

Vali Derhami[1] Zahra Youhannaei[2]

[1]Electrical and Computer Department, Yazd University, Yazd, Iran.
Phone: (98351) 8122610, Email:*vderhami@yazduni.ac.ir*
[2] Razy Clinic, Karaj, Iran, Email: *darya1112@yahoo.co*

Abstract
This paper demonstrates a kind of learned helplessness in human being that is appeared in Fuzzy Reinforcement Learning (FRL) algorithm. At the beginning of learning, when an agent continuously performs actions that cause sequential punishments, afterwards it does not usually behave well and often selects actions that evoke punishments. This faulty learning is similar to what is called learned helplessness in social sciences. Here, this problem is demonstrated in training an agent by FRL algorithm. By analyzing this problem, a new reinforcement function for prevention is presented.

Keywords: Learned helplessness, Fuzzy systems, Reinforcement learning.

1. Introduction

Fuzzy Reinforcement Learning (FRL) algorithms are a combination of fuzzy system and Reinforcement Learning (RL) [1-3]. Fuzzy systems imitate human decision making capability and RL is a powerful interactive learning methodology that has been inspired from human beings' and animals' learning.

Here, we focus on Fuzzy Sarsa learning (FSL) that is the first critic-only FRL with mathematical analysis [4]. This algorithm tunes the parameters of conclusion parts of the fuzzy system rules online.

During training agents using FRL algorithms such as FSL, the learning is not successful in some runs. Our studies on learning details showed that if the agent received sequential punishments in the first episodes of learning, the punishments would cause the agent not to learn, although sometimes agent has reached to goal and received some positive reinforcement.

The above situation is similar to learned helplessness situation. Helplessness is seen in social and psychological life, when a person receives some sequential and uncontrolled punishments, so he/she may behave unsuitable in the future. Seligman showed that learned helplessness not only depends on undesired experiments, but also depends on the disability or imaginative disability about something that he/she can not do anything about it. So the helplessness generalizes to the other situations and as a result the organism will be passive [5,6].

The helplessness phenomenon can be established in many kinds, such as human being, by harmful and undesired unconditional stimulus. The equivalent for the word "helplessness" are: apathy in doing anything to avoid punishment, being generally passive, withdrawal, fear, depression, and accepting whatever happens [5,6].

In this paper, we first demonstrate learned helplessness in training an agent by FSL in boat problem [7]. The reasons

of learned helplessness are analyzed and discussed. Then a new reinforcement function to prevent learned helplessness is presented. By relating the new reinforcement function to the inverse of "fuzzy visit value" of the current state, when agent sequentially makes a mistake, the amount of punishment is decreased as well as the helpless agent receives an incremental reward upon it selects a good action. Based on our knowledge this is the first work in this category.

The organization of this paper is as follows: Fuzzy Sarsa learning is presented in Section 2. In Section 3, the learned helplessness is demonstrated. Adaptive reinforcement function to prevent the learned helplessness is presented in Section 4. Finally, the conclusion is given in Section 5.

2. Fuzzy Sarsa Learning (FSL)

FSL is an extension of Sarsa learning (a well known RL algorithm) [8] for continuous state and action spaces using a zero order Takagi-Sugeno (T-S) fuzzy system [9] as function approximator. In this section, we describe FSL briefly; readers can find the comprehensive information about FSL in [4].

Sarsa method estimates the value of action a in state s denoted by $Q(s,a)$ for the current policy according to the following update formula [8]:

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

where α is the learning rate, γ is the discount factor, and r_{t+1} is the immediate reward received from the environment after applying action a_t in state s_t.

Consider an *n*-input one-output zero-order T-S fuzzy system with *R* rules of the following form [4]:

R_i : If x_1 is L_{i1} and … and x_n is L_{in}, then (a_{i1} with value w^{i1}) or … or (a_{im} with value w^{im})

where $s = x_1 \times \cdots \times x_n$ is the vector of n-dimensional input state, $L_i = L_{i1} \times \cdots \times L_{in}$ is the n-dimensional strictly convex and normal fuzzy set of the *i*-th rule with a unique center, m is the number of possible discrete actions for each rule, a_{ij} and w^{ij} are the *j*-th candidate action and the approximate value of the *j*-th action in the *i*-th rule, respectively. The goal of FSL is to adapt w^{ij} on-line to be used to obtain the best policy.

The action selection probability of the *i*-th candidate action in the *i*-th rule in state s_t is computed based on the following modified Softmax policy [4]:

$$p(a_{ij}) = \frac{\exp(\mu_i(s_t) w^{ij} / T)}{\sum_{k=1}^{m} \exp(\mu_i(s_t) w^{ik} / T)} \quad (2)$$

where $\mu_i(s_t)$ is the normalized firing strength of the *i*-th rule for state s_t, and $T > 0$ is the temperature factor.

Notice that to calculate the overall action, first an action is selected for each rule from among the candidate actions of that rule. Denoting the selected action in *i*-th rule and its corresponding value by a_{ii^+} and w^{ii^+}, respectively, the system output (i.e., the overall continuous action) and its corresponding approximate Action Value Function (AVF) are computed as follows [4,7]:

$$a_t(s_t) = \sum_{i=1}^{R} \mu_i(s_t) a_{ii^+} \quad (3)$$

$$\hat{Q}_t(s_t,a_t) = \sum_{i=1}^{R} \mu_i(s_t) w_t^{ii^+} \quad (4)$$

Thus, the final continuous action is the weighted sum of the selected discrete actions of the rules.

Applying action a_t, the environment goes to the next state s_{t+1}, and the agent receives reinforcement signal r_{t+1}. The next final action a_{t+1} is chosen based on the present weight w_t. Then, the weight parameters of the *i*-th rule are updated by [4]:

$$\Delta w_{t+1}^{ij} = \begin{cases} \alpha_{t+1} \times \Delta \hat{Q}_t(s_t, a_t) \times \mu_i(s_t) \ if \ \ j = i^+ \\ 0 \qquad\qquad\qquad\qquad\quad otherwise \end{cases} \tag{5}$$

where $\Delta \hat{Q}$ is the approximate action value error determined by:

$$\Delta \hat{Q}_t(s_t, a_t) = r_{t+1} + \gamma \hat{Q}_t(s_{t+1}, a_{t+1}) - \hat{Q}_t(s_t, a_t) \tag{6}$$

3. Demonstration of learned helplessness

In this section, we show the learned helplessness can happen during training an agent by FRL algorithms. Our experiments show that at the beginning of learning, when an agent continuously performs actions that cause sequential punishment, afterwards it does not usually behave well and often selects actions that evoke punishments. The reason for this behavior is that in FRL algorithms AVF is approximated by fuzzy system. Continuous punishments (negative reinforcements) tend all weight parameters of approximator toward these negative amounts. Hence even for the not visited states, there is not a neutral approximation and the amount of AVF in all of learning space is negative.

Consequently, agent cannot select suitable actions and receives punishments again due to its bad selections. This cycle is repeated continuously and the learning performance is dropped significantly.

Here, we demonstrate this learned helplessness phenomenon in training agent as a driver in Boat Problem. Consider an agent in Boat Problem as a driver [7] that has to learn to drive a boat from the left Bank to the right bank quay in a river with a strong nonlinear current. The goal is to reach the quay from any position on the left bank (see Fig. 1). FSL algorithm is used to tune fuzzy controller.

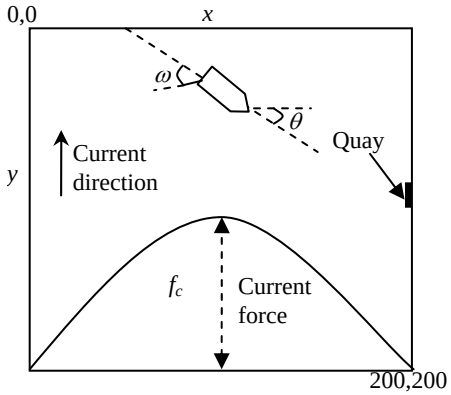

Fig. 1: Boat problem [7].

3.1. System dynamics

This problem has two continuous state variables, namely, *x* and *y* position of the boat's bow ranging from 0 to 200. The quay center is located at (200,100) and has a width of five. The fuzzy control command determines the rudder angle of boat. The detail of the system dynamics can be found in [7].

3.2. Learning

As shown in Fig. 2, five fuzzy sets are used to partition each input variable resulting in 25 rules [7]. The discrete candidate action set for each consequence of the fuzzy rules is made up of 12 directions: $A = \left\{ \begin{matrix} -100, -90, -75, -60, -45, \\ -35, -15, 0, 15, 45, 75, 90 \end{matrix} \right\}$ as used in [7]. The controller generates continuous actions using Eq. (3).

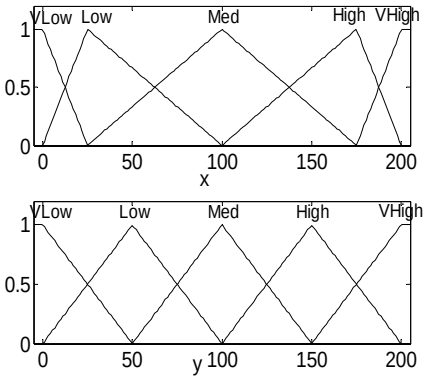

Fig. 2: Input membership functions.

The reinforcement function is zero during the traverse of boat, and is a non-zero function of boat position in *y* direction after reaching any of the following three zones in the right bank: The success zone Z_s ($x = 200$, $y \in [97.5, 102.5]$), the viability zone Z_v ($x = 200$, $y \in [92.5, 97.5] \cup [102.5, 107.5]$), and the failure zone Z_f , which includes all the other points of the bank. The reinforcement function is defined by [7]:

$$r(x,y) = \begin{cases} +1 & (x,y) \in Z_s \\ D(x,y) & (x,y) \in Z_v \\ -1 & (x,y) \in Z_f \\ 0 & \text{otherwise} \end{cases} \qquad (7)$$

where function $D(x,y)$ decreases linearly from 1 to -1 based on the distance from the success and failure zones.

100 runs are accomplished. Every run includes a "learning phase" and a "testing phase". For the learning phase, 100 sets of random positions are generated. The learning phase ends when either 40 successive non-failure zones are hit, or the number of episodes exceeds 5000. We call the episode number at the end of learning phase as the *Learning Duration Index* (LDI), which is a measure of learning time-period. The testing phase is made of 40 episodes. Let *d* be the distance error of the reached bank position relative to the quay center given by [6]:

$$d(x,y) = \begin{cases} |y - 100| & \text{if right bank reached} \\ 100 + (200 - x) & \text{otherwise} \end{cases} \qquad (8)$$

Then, Distance Error Index (DEI) can be defined as the average of *d* over the 40 episodes in the testing phase, which is a measure of action quality.

Fig. 3 shows the histogram of LDI's in 100 runs. As seen, although agent has learned in less than 2000 LDI's in the most runs, but LDI's are greater than 2000 in some runs and LDI's has reached to upper bound (5000) in four runs. In these runs, agent has not learned well and DEI in the testing phase is very bad.

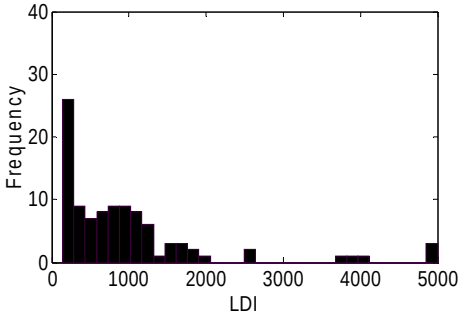

Fig. 3: LDI histogram for FSL with conventional reinforcement function.

Fig. 4 shows the received rewards during an unsuccessful run. As seen, the agent has received many punishments (negative reinforcements) at the beginning of learning. These punishments have caused the high error in AVF approximation.

Fig. 5 shows the highest action values, i.e. $\max_a(\hat{Q}(s,a))$, for the learning space in an unsuccessful run. As seen, the values of the all of state space have been strongly influenced by the negative reinforcements, and consequently tended to -1. All values are less than zero even for

the area near the quay, whose optimal values should be close to one. Hence, the agent has been helpless in learning.

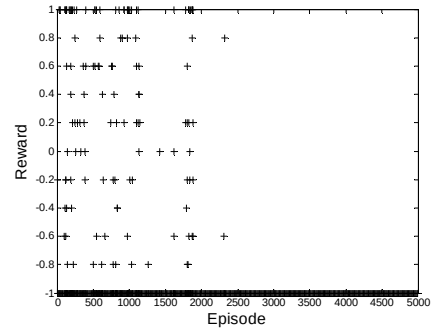

Fig. 4: Rewards in an unsuccessful run.

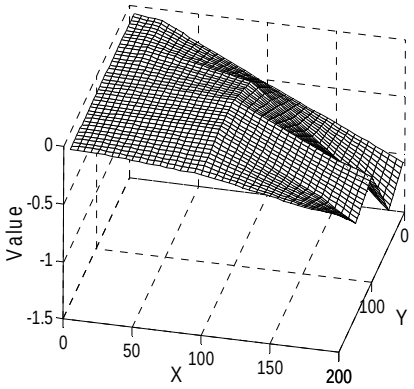

Fig. 5: Highest action values in an unsuccessful run.

4. Adaptive reinforcement function

As said and demonstrated in section 3, the main reason of learned helplessness is receiving sequential negative reinforcements that tend the all weights of AVF approximator to negative amount.

A new reinforcement function to prevent learned helplessness is proposed in this section. This reinforcement function is adaptive and depends on the number of visit of the state as well as the distance time of visit. In the following the proposed solution is described.

Let N be an $R\times 1$ vector whose i-th element is sum of the past firing strengths of the i-th fuzzy rule. Then, N can be written in the following recursive form:

$$N_t = N_{t-1} + \mu(s_t) \qquad (9)$$

where $\mu(s_t)=[\mu_1(s_t),\cdots,\mu_R(s_t)]^T$ is the vector of the normalized firing strength of the rules. The above formula shows that the i-th element of N increases when agent visits the i-th fuzzy rule patch (the fuzzy area defined by the antecedent of the i-th rule) more, or equivalently, when the i-th rule fires more. We also define the fuzzy visit value of state s_t as:

$$FV(s_t) = \frac{\mu(s_t)^T \cdot N_{t-1}}{\sum_{j=1}^{R} n_{t-1}^{j}} \qquad (10)$$

where n^j is the j-th element of vector N. It can be easily shown that $0 \le FV(s_t) \le 1$. The adaptive reinforcement function is defined as:

$$r_a(t) = \min\left(\frac{|r_t|\cdot R}{FV(s_t)}, r_{\max}\right)\times sign(r_t) \qquad (11)$$

In this formula, r_t is a conventional reinforcement function for the problem, $sign(.)$ is sign function, and $r_{\max}$ is the upper bound of reinforcement function.

According to the Eq. (11), the reinforcement function will be small for the parts of space that the agent visits frequently and will be large for other parts. This adaptive reinforcement function gives opportunity to agent that does good action and upon a depressed agent does a good action, agent receives a incremental reward.

To assess the proposed reinforcement function, we apply it in FSL in the boat problem. Fig. 6 shows the histogram of LDI's for this experiment. As seen, ap-

plying the adaptive reinforcement function has prevented learned helplessness. All of LDI's are less than 3500. Moreover, the average of LDI's has improved more than 44% (it has been decreased from 980 (for FSL with conventional reinforcement function) to 540).

Moreover, the average of EDI in test phase is 4.1 cm in contrast 6.27 cm in FSL with the conventional reinforcement function.

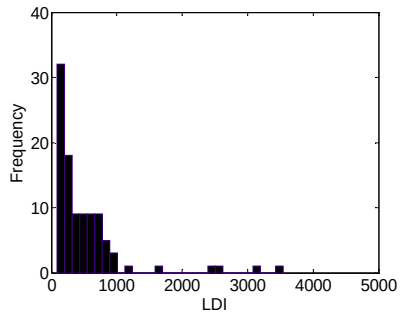

Fig. 6: LDI histogram for FSL with the adaptive reinforcement function.

5. Conclusion

This paper demonstrated a kind of learned helplessness in FRL. It was shown, how the sequential punishments can cause faulty learning that it is similar to the learned helplessness or depression in human beings. The adaptive reinforcement function was presented that prevents the sequential punishments in such situations. Applying this adaptive reinforcement function in FSL in Boat problem showed significant improvement. This solution can be extended to our social and psychological life. When we face with person that he/she makes sequentially mistakes, we can not continue punishments but also we have to look forward to opportunity that he/she does a good work and we get him/her a high reward.

6. References

[1] H. R. Berenji, D. Vengerov, "A convergent actor-critic-based FRL algorithm with application to power management of wireless transmitters," *IEEE Trans. On Fuzzy Systems*, vol. no.4, pp.478 – 485, 2003.

[2] M. J. Er, C. Deng, "Online tuning of fuzzy inference systems using dynamic fuzzy Q-learning", *IEEE Trans. Syst., Man, Cybern. B*, vol. 34, no, 3, pp. 1478 – 1489, 2004.

[3] P. Ritthipravat, T. Maneewarn, D. Laowattana, J. Wyatt, "A modified approach to fuzzy Q learning for mobile robots", *IEEE Int. Conf. Syst., Man, Cybern.*, pp. 2350-2356, 2004.

[4] V. Derhami, V.J. Majd, M. Nili Ahamadabadi, "Fuzzy Sarsa learning and the proof of existence of its stationary points", *Asian Journal of Control*, vol. 10, no.5, pp. 535-549, 2008.

[5] M.E.P., Seligman, "On the generality of the laws of learning" *Psychological Review*, vol. 77, pp. 406-418, 1970.

[6] M.E.P. Seligman, *Helplessness*. SAN Francisco, Freeman, 1975.

[7] L. Jouffe, "Fuzzy inference system learning by reinforcement methods," *IEEE Trans. Syst., Man, Cybern. C*, vil.28, no.3, pp.338-355, 1998.

[8] R.S. Sutton, A. G. Barto, *Reinforcement learning: An introduction*, Cambridge, MIT Press, 1998.

[9] J. S. R. Jang, C. T. Sun, and E. Mizutani, *"Neuro-Fuzzy and soft computing,"* Prentice-Hall, 1997.