Discretization of Analog Communication Signals by Noise Addition in Communication Learning

Katsunari Shibata(shibata@cc.oita-u.ac.jp) Masanobu Nakanishi

Dept. of Electrical & Electronic Engineering, Oita University, 700 Dannoharu, Oita 870-1192, Japan.

Abstract

Using symbols, our humans can communicate complicated information cleverly with each other. Thinking about the "Symbol Grounding Problem" and the brain structure of the living things, the authors believe that it is the best solution for generating communication to use a neural network that is trained based on reinforcement learning. However, it has been said that neural networks are not good at symbol processing. As the first step of the research of symbol emergence using neural network, this work focused on the necessity of "elimination of noise effect", and it was examined that analog communication signals are binarized in some degree by adding some noise in reinforcement learning-based communication acquisition. It was also observed that when the noise ratio became larger, the degree of the binarization became larger.

1 Introduction

Our humans can communicate complicated information cleverly with each other using symbols. It has been thought that artificial neural networks(ANNs) are good at continuous nonlinear approximation, but are not good at symbol handling or logical processing. In our living things, the functional difference has been pointed out between the left brain and the right brain[1]. Especially, the Broca's area and Wernicke's area that relate deeply to language are located in the left brain[2].

Based on this common knowledge, the idea of the specialization that the ANN corresponding to the right brain is used for pattern processing, and a digital computer corresponding to the left brain is used for logical processing has been accepted generally. However, there is no general idea about what signals should be transferred between the ANN and the computer, and that causes the "symbol grounding problem". Furthermore, it seems strange that the left brain and the right brain looks almost the same in the real brain compared with the difference between the ANN and the digital computer.

We think that this "symbol grounding problem" is very serious. We believe that in order to solve the problem, the pattern processing and the logical processing should not be distinguished, and that is essential to realize intelligence in robots. Accordingly we expect the ANN to do both processing without any discrimination. In our brain also, a natural neural network must process symbols. For this reason, it is very significant to show that the ANN has the ability to discretize analog signals only by applying reinforcement learning. Then, there appears a question "is it true that symbols do not emerge from the ANN through learning?"

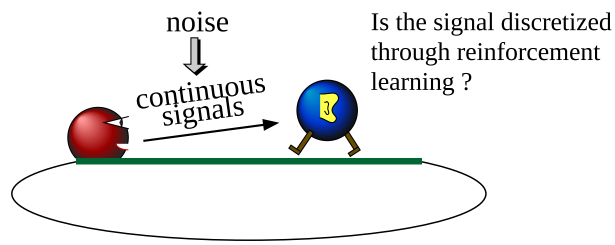

Figure 1: Discretization of communication signals by the addition of noise.

Here, for simplicity, symbols are considered as discretized signals. Then, why do we discretize the communication signals? Considering from necessity, either logical thinking or eliminating noise effect can be one of the reasons. Considering from the structure, associative memory, in other words, fixed-point dynamics can be a solution to realize the discretization. As the first step of the research of symbol emergence, this work focuses on the necessity of "elimination of noise effect", and it is examined whether analog communication signals are discretized or not by adding some noise in reinforcement learning-based communication acquisition.

2 Learning and Task

As a simple communication environment, two agents are assumed. Referring to [3], one of them can transmit some communication signals to the other. They are put on a very simple one-dimensional space as shown in Fig. 1. When the both agents touch together, they get some reward. The transmitting agent cannot move, but can observe the relative location of the opponent, and generates its communication signals by its own neural network. The receiving agent interprets the communication signals and generates its motion command also by its own neural network. It can move according to the motion command. It can-

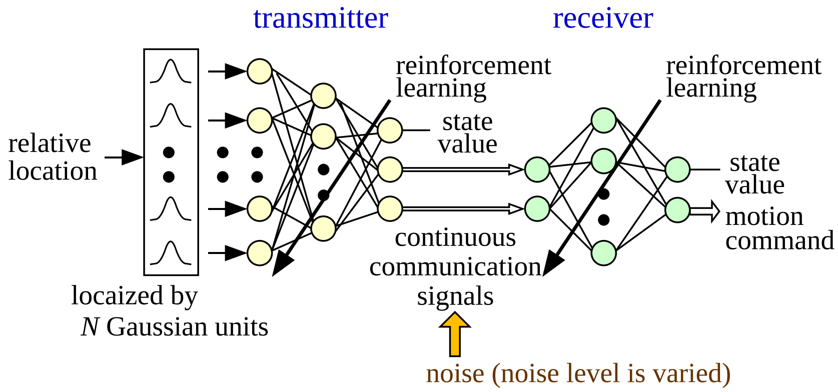

Figure 2: Architecture of each agent and signal flow.

not observe anything except for the communication signals, and cannot transmit anything. Both agents are trained based on reinforcement learning independently. The transmitting agent deals with the communication signals as its actions, while the receiving agent deals with the communication signals as its states.

The transmitting agent is fixed at the left edge on a one-dimensional ring where the left edge is linked to the right edge. The length of the ring is 1.0. The receiving agent is located randomly at every trial. The distance moved is proportional to the motion command that is the sum of the output of the receiver's neural network and a random number as a trial and error factor. The motion command can be negative. When the command is positive, it goes to the right, and when it is negative, it goes to the left. When the distance between the transmitter and receiver is less than some value, they can touch each other and get a reward. However, if the motion command is large though the receiver is close to the transmitting agent, it goes past the transmitter, and they cannot get the reward. Accordingly the receiver's motion should be in a range, and the range is gradually sifted according to the relative receiver's location.

Fig. 2 shows the architecture of each agent and the signal flow. The transmitting agent observes the relative receiver's location, and then the information is localized by N Gaussian units. This helps the neural network to learn a strong nonlinear transformation. The center of each Gaussian is arranged between 0.0 and 1.0 with a constant interval. The size of each Gaussian σ is $1.0/(N-1)$, where N is the number of Gaussian units. The output is described as

$$GS_i(dist) = exp\left(-\frac{1}{2}\left(dist - \frac{i}{N-1}\right)^2\right), \quad (1)$$

where i is the suffix of the Gaussian unit number ($i = 0, 1, 2, .., N-1$), $dist$ is the relative receiver's distance from the transmitter. Here, $N = 30$.

Two or three of the outputs of the transmitting agent are used to decide the communication signals. Each signal is the sum of the corresponding output and a random number as a trial and error factor that is added for reinforcement learning. A noise factor is also added to the signal.

As a reinforcement learning architecture, actor-critic is employed for each agent. One of the outputs of the network is used as critic, and the others are used as actor. The hidden neurons are used in common by both types of outputs. The training signals are computed based on reinforcement learning, and the network is trained based on Back Propagation. TD error $\hat{r}$ is calculated as

$$\hat{r}_t = r_t + \gamma P_t - P_{t-1} \quad (2)$$

where r is the reward, P_t is the critic output, and γ is a discount factor. The critic output is trained by the training signal as

$$P_{s,t-1} = P_{t-1} + \hat{r}_t = r_t + \gamma P_t. \quad (3)$$

The actual motion vector $\mathbf{M}$ is calculated as

$$\mathbf{M}_t = \alpha(2.5\mathbf{A}_t + \mathbf{rnd}_t + \mathbf{n}_t) \quad (4)$$

where $\mathbf{A}$ is the actor output vector, $\mathbf{rnd}$ is the random number vector for the trial and error factor, and $\mathbf{n}$ is the noise vector that is not added in the case of the receiver's motion, but is added in the case of the communication signals. α is a constant. The actor output is trained by the training signal as

$$\mathbf{A}_{s,t-1} = \mathbf{A}_{t-1} + \beta\hat{r}_t\mathbf{rnd}_{t-1} \quad (5)$$

where β is a constant, and it is 0.5 here.

The output function of each hidden or output neuron is a sigmoid function that ranges from -0.5 to 0.5. The upper limit for all the training signals is 0.4, and the lower limit is -0.4 to avoid the saturation area of the sigmoid function. In Eq. (4), by multiplying 2.5 to each actor output, the range becomes from -1.0 to 1.0, and after that, the trial and error factors and noises are added. Here, the trial and error factor is cubed uniform random number whose level, in other words, whose amplitude is ± 0.4 or ± 0.6. The noise factor is a uniform random number whose level is varied from ± 0.0 to ± 1.6 with the interval of 0.2 in the following simulations. When the value becomes larger than 1.0 or less than -1.0, it is returned to 1.0 or -1.0 respectively. Even in the case that the noise factor is always zero, the random number for the trial and error factor is received as a noise for the receiver.

For the critic computation based on TD (Temporal Difference) learning in Eq. (2) and (3), 0.5 is added to the output for the critic actually. The reward that is given to the both agents is 0.9. To generate the communication signal, α in Eq. 4 is 1.0 in the transmitting agent. For the motion command, α is 0.4 or 0.43 in the receiving agent so as that the receiver can touch the transmitter in one step from any locations by an appropriate motion. The number of layers is 3, and the number of neurons in the hidden layer is 10 for both agents.

Two tasks are introduced in this paper. In the first one, the both agents can touch with each other when the distance is less than 0.11, and α in Eq. (4) is 0.4. If the receiver's motion is discretized, no less than 4 levels of output is required. The number of communication signals is two. In the other task, the both agents can touch when the distance is less than 0.08, and α is 0.43. If the receiver's motion is discretized, no less than 6 levels of output is required. The number of communication signals is three.

3 Result

In this communication learning, it was observed whether the transmitted signals became discrete when the noises were added to the communication signals during the learning. The communication signals after learning with no noise are as shown in Fig. 3(a) and those with some noise (level=0.8) are as shown in Fig. 3(b). Fig. 3(c) shows the signals after learning with some noises (level=0.8) in the case of the task with 3 communication signals. Fig. 4 shows the receiver's motion after learning with noise for each of two and three communication signals cases. The sloping lines in Fig. 4 indicate the maximum and minimum limit values of the motion for the receiver to touch the transmitter by the motion. After learning with noise, each communication signal was almost binarized, and only around the boundary of the binary values, the signal took a medium value. However, it is clear that the degree of binarization is larger than in the case of no noise. The receiver's motion is discretized into four levels by the combination of the two binary communication signals in the range between the maximum and the minimum values. The motion is more clearly discretized than the communication signals. The reason might be that the receiver learned to binarize the received signal utilizing non-linear transformation of the neural network. When the number of communication signals is three, the information to generate the motion command is allotted well among three signals.

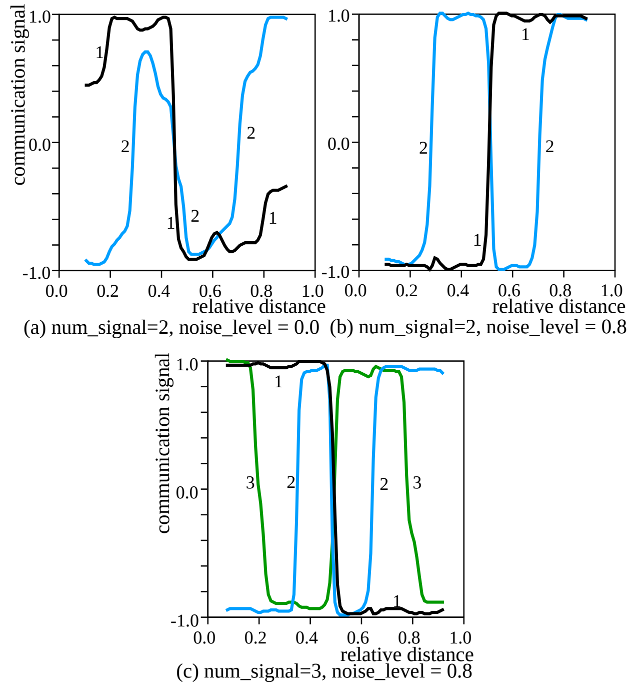

Figure 3: The communication signals as a function of the relative distance. The random number level is 0.4.

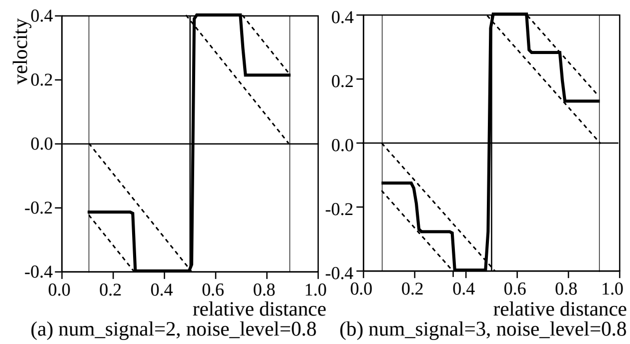

Figure 4: The receiver's motor command as a function of the relative distance. The random number level is 0.4 and the noise level is 0.8.

The relation between the noise range and the discretization was also observed. The degree of binarization that means how the signal is close to 1.0 or -1.0 is defined as

$$bin = \sum_{i}^{Nc} \sum_{j}^{Nd} |com_{i,j}| / (Nc \cdot Nd) \tag{6}$$

where Nc is the number of communication signal, Nd is the number of sampled relative receiver's locations, and com is the communication signal without the ran-

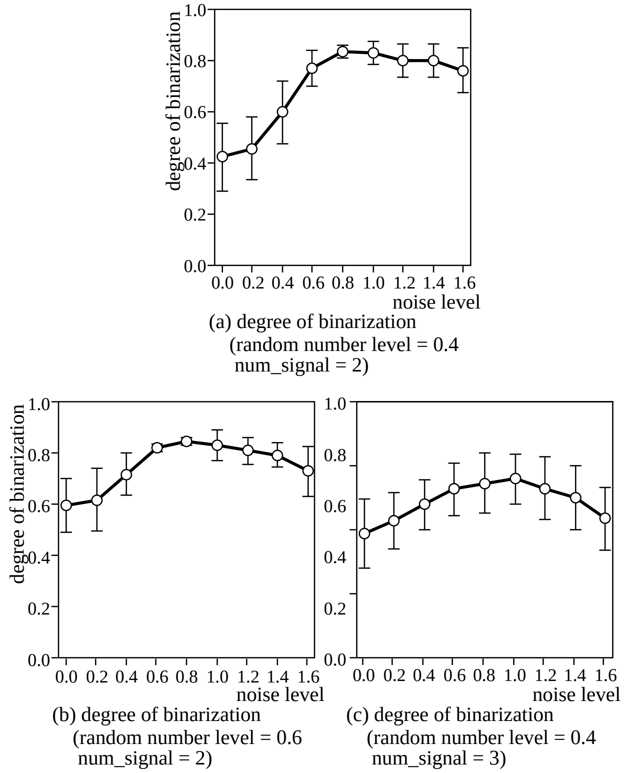

Figure 5: The degree of binarization according to the noise level in the learning phase.

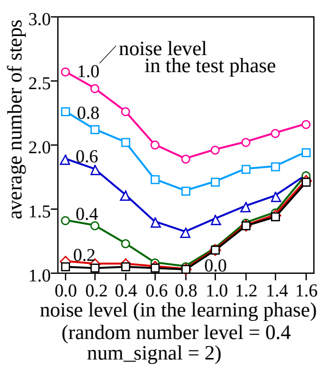

Figure 6: Noise tolerance according to the noise level in the learning phase.

dom number and noise. If the communication signal is always -1.0 or 1.0, the degree becomes the maximum value of 1.0, while if the signal is always 0.0, it becomes the minimum value of 0.0. The degree of binarization according to the noise range is shown in Fig. 5(a). Each small circle shows the average over 50 simulations, and the vertical line shows the standard deviation. It can be seen that when the noise level becomes larger, the degree of binarization becomes larger and the deviation becomes smaller. However, when the noise level becomes larger than 0.8, the degree decreases slightly according to the noise level.

The noise tolerance was also examined. Fig. 6 shows the average steps to the goal as a function of the noise level in the learning phase for each noise level in the test phase after learning. It can be seen that if some noise is added in the test phase, the performance is the best when the noise level in the learning phase is 0.8. It is interesting that under the same condition, the degree of binarization is the maximum. However, when the noise level in the learning phase is larger than 0.8, the performance becomes worse even if the noise level in the test phase is 0.0. This means that the learning itself did not progress by the large noise.

In this simulation, the random number is added to each communication signal other than the noise, but for the receiver, the random number also works as the noise. In order to see the effect of the random number, Fig. 5(b) shows the degree of binarization when the level of the random number is 0.6 on behalf of 0.4. Comparing with Fig. 5(a), the degree is larger when the random number level is larger, and the noise level in the case of the maximum degree of binarization is shifted slightly to be small. It is known that the random number for reinforcement learning also promoting the binarization of the signals.

Fig. 5(c) shows the degree when the number of communication signals is three. It can be also seen that the degree of binarization becomes large according to the noise level and have the maximum value.

4 Conclusion

It was shown that unless the noise level is too large for learning to progress, the communication signal is binarized more, and is tolerant of noise more according to the noise level when the communication signal is generated using a neural network, and the network is trained based on reinforcement learning. The authors think it very significant to show that the signal generated by a neural network is binarized only by reinforcement learning. It is thought that by using a recurrent network, the signal is binarized more clearly.

Acknowledgment

This research was partially supported by the Japan Society for the Promotion of Science, Grants-in-Aid for Scientific Research, #14350227 and #15300064.

References

[1] Sperry, R.W., “Hemisphere Deconnection and Unity in Conscious Awareness”, *American Psychologist*, **23**, pp. 723-733, 1968.

[2] Geschwind, N., “Specialization of the Human Brain”, In *the Brain*, W.H. Freeman and Company, San Francisco, 1979.

[3] Ono, N, Ohira, T. and Rahmani, A.T., “Emergent Organization of Interspecies Communication in Q-Learning Artificial Organs”, *Advances in Artificial Life*, pp.396–405, 1995.