Gauss-Sigmoid Neural Network

Katsunari SHIBATA and Koji ITO

Tokyo Institute of Technology, Yokohama, JAPAN

shibata@ito.dis.titech.ac.jp

Abstract- Recently RBF(Radial Basis Function)-based networks have been widely used because they can learn a strong non-linear function faster and easily by their local learning characteristics. Among them, Gaussian soft-max networks have generalization ability more than regular RBF networks because of their extrapolation ability. However, since the RBF-based network has no hidden unit which can represent some global information, the internal representation cannot be obtained. Accordingly even if the knowledge which could be obtained through the previous sets of learning is utilized effectively in the present learning, the network has to learn from scratch. While, multi-layered neural networks are able to form the internal representation in the hidden layer through learning.

The present paper proposes a Gauss-Sigmoid neural network for learning with continuous input signals. The input signals are put into a RBF network, and then the outputs of RBF network are put into a sigmoid-based multi-layered neural network. After learning based on back-propagation, the localized signals by the RBF network are integrated and an appropriate space for the given learning is reconstructed in the hidden layer of the sigmoid-based neural network. Once the hidden space is constructed, both the advantage of the local learning and the global generalization ability can exist together.

Keywords- RBF network, Gaussian soft-max network, sigmoid-based neural network, Gauss-sigmoid neural network, localization, internal representation.

1. Introduction

RBF (Radial Basis Function) networks have been widely used because of their local learning characteristics. Local learning means that the learning for an input pattern scarcely damages the already trained input-output relations for the other input patterns which are not close to the present pattern. Therefore they can learn faster in general and can learn a strong non-linear function easily. On the other hand, the sigmoid-based neural network sometimes damages the already trained input-output relations by the learning for the present input pattern. On the other hand, the generalization ability of RBF networks is poor because of their local learning characteristics, while the sigmoid-based neural network has a global generalization ability. There seems to exist a dilemma between the local learning and the global generalization at a glance.

To improve the generalization ability of the RBF network, Gaussian soft-max networks have been utilized recently as an example[1]. Their extrapolation ability is well evaluated. However, since the RBF-based network does not have a hidden unit which can represent some global information as shown in Fig. 1, the reconstructed space cannot be obtained. When we control the robot using neural networks, it is expected to keep the general knowledge in its hidden layer like 3D spatial recognition that is useful for the robot to deal with various tasks. But the Gaussian soft-max network cannot store the general knowledge and the learning on the reconstructed space cannot be realized.

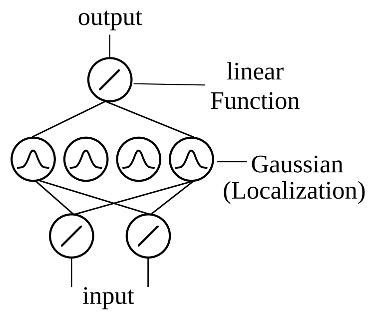


Fig. 1 RBF-based network (RBF network, Gaussian soft-max network). There are no hidden layers which have a unit representing the global information.

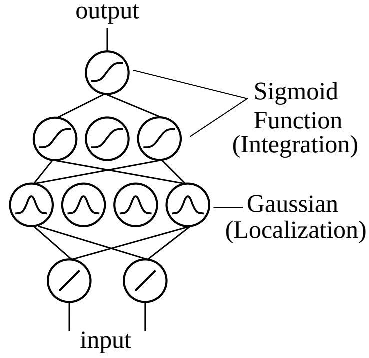


Fig. 2 Gauss-Sigmoid neural network that is proposed in this paper

In this paper, Gauss-Sigmoid neural network (NN) as shown in Fig. 2 is proposed. The activate function of each unit in the lowest hidden layer is Gaussian, which localizes continuous input space. The role of localization is similar to the visual sensor. The activate functions of all the other hidden units and output units are sigmoid functions. It is also proposed to modify the center and size of each Gaussian adaptively together with the connection weights of the sigmoid units according to the error back propagation. The learning performance is compared among the Gauss-Sigmoid NN, RBF network, Gaussian soft-max network and sigmoid NN.

2. RBF(Gauss)-based Network vs. Multi-layered Sigmoid-based Neural Network

2.1 RBF network and Gaussian soft-max network

Two types of the popular RBF(Gauss)-based networks used in this paper are introduced at first. The first one is a regular RBF network. Gaussian is used as a RBF, the output of the RBF network is written as

$$output = \sum_{i=1}^{n} w_i\, g_i(x) + \theta, \quad \text{where} \tag{1}$$

$$g_i(x) = \exp\left(-\sum_{d=1}^{D} \left(\frac{x_d - \mu_{i,d}}{\sigma_{i,d}}\right)^2\right), \tag{2}$$

where $(\mu_{i,1},\ldots, \mu_{i,D})$: the center of the i-th RBF unit, $(\sigma_{i,d},\ldots, \sigma_{i,d})$: the size of the i-th RBF unit, w_i : the connection weight from the i-th RBF unit, θ : bias, n : the number of RBF units, D : the number of the input space. The weights w are trained by learning, and the center μ and the size σ of each RBF unit are often also trained by the error back propagation(BP) learning. In Gaussian soft-max network, the output of each RBF unit is normalized by the sum of all the RBF outputs, and the output of the network is weighted sum of the normalized RBF outputs as

$$output = \sum_{i=1}^{n} w_i\, b_i(x) + \theta, \quad \text{where} \tag{3}$$

$$b_i(x) = g_i(x) \Big/ \sum_{i=1}^{n} g_i(x). \tag{4}$$

2.2 Learning Speed

As written in the introduction, the learning of the RBF-based network is fast, since the learning for an input pattern scarcely damages the already trained input-output relations for the other input patterns which are not close to the present pattern. That is because the output of the RBF unit for the distant input is always close to 0.0. While, the sigmoid-based neural network sometimes gives damages to the already trained input-output relations. This becomes serious when a strong non-linear function is approximated by learning.

However, as more RBF units are allocated, the learning speed becomes slower. While, the sigmoid-based neural network is not affected so much by the number of its hidden units. In consequences, when a function, which requires strong non-linearity, is approximated, RBF network takes an advantage in the learning speed, while when a function with weak non-linearity is approximated, sigmoid-based neural network is better in general.

2.3 Generalization Ability

Different from the regular RBF network, extrapolation works effectively in the Gaussian soft-max network. For example, as shown in Fig. 3, suppose that the input space has two dimensions, and the training signal in the first and third quadrants of the input space is 1.0, and in the second and fourth quadrants, the training signal is –1.0. Suppose also that four RBF units are assigned, and the output for the input pattern α is examined after the learning around the origin. By using regular RBF network, the output becomes close to 0.0 because the input is far from the center of every RBF unit. However, by using Gaussian soft-max network, the output becomes close to –1.0.

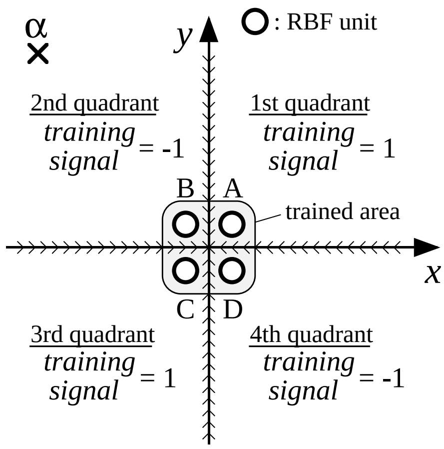


Fig. 3 An example which indicates the extrapolation ability of Gaussian soft-max network in two input space using 4 RBF units.

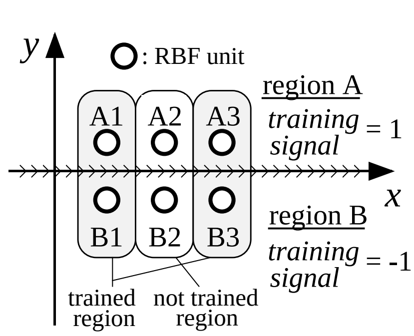


Fig. 4 An example case in which the generalization is not effective in Gaussian soft-max network. In the non-trained region, it cannot approximate correctly, while Sigmoid-based network can do.

By the way, when many RBF units are assigned, interpolation ability is not effective even in Gaussian soft-max network. This property is inherited from the regular RBF network. For example, as shown in Fig. 4, suppose that the training signal for region A is 1.0, and for region B is –1.0, and six RBF units are assigned. Suppose also that the network is trained only around the unit A1, A3, B1, B3. After the training, the output around the unit A2 and B2 will not be close to the expected value 1 and –1 respectively. That is also the reason of slow learning when the number of RBF units is large.

On the other hand, sigmoid-based neural network has more strong generalization ability. That is because sigmoid function is a global function that divide whole the input space smoothly into two regions, while the RBF is a localizing function that picks up a close small region out of the whole input space. So by using the sigmoid-based neural network, the output for the point α in Fig. 3 becomes close to –1.0, and the output around the point A2 in Fig. 4 becomes close to 1.0.

2.4 Internal Representation

In multi-layered neural networks, the internal representation is formed in the hidden layer, while it cannot be formed in the RBF-based networks. Let us see the example of Fig. 4 again. Suppose that the input-output relation has been already trained and the internal representation has been formed. Then another output unit supposes to be assigned to the network with 0.0 hidden-output connection weights. When only one input signal for each region is trained by some training signal, for example, -1.0 for region A, and 1.0 for region B, all the outputs in region A are expected to become close to –1.0, while those in region B are expected to become close to 1.0. However, in the case of the RBF-based network, all the outputs have to be trained to obtain the expected values. The generalization on this internal representation space is sometimes very useful, especially when we utilize the neural network for the robot learning. For example, the spatial recognition ability is useful for the robot to achieve many tasks. If the spatial recognition ability is obtained as the internal representation through the learning of the previous tasks, the robot can learn the next task not on the visual sensory signal space, but on the 3D space that is reconstructed in the hidden layer from the visual sensory signal space. Here this is also called generalization ability in wide meaning.

2.5 Learning of Discontinuous Mapping

The sigmoid-based neural network is not good at the approximation of discontinuous mapping from the input pattern to the output. When the sigmoid-based neural network is trained to approximate such a mapping by the error back propagation(BP) learning, the connection weights from the hidden layer to the output layer become very large as well as those from the input layer to the hidden layer through the learning. For simplicity, suppose that a simple 1-1-1 layered network as shown in Fig. 5 (a) whose output unit has a linear activation function is trained to approximate the discontinuous mapping. The output x_2 of the network is represented as

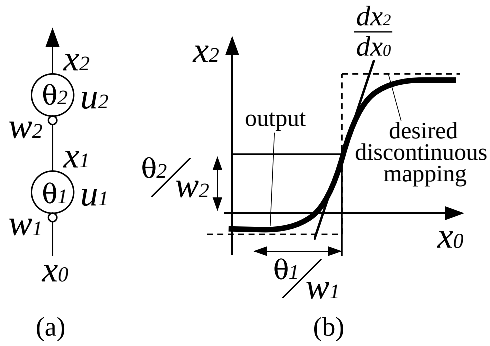


Fig. 5 A simple sigmoid-based neural network. Discontinuous mapping is trained on this network.

$$x_2 = w_2\,[\,f\{w_1\,(x_0 + \frac{\theta_1}{w_1})\} + \frac{\theta_2}{w_2}], \qquad (5)$$

where $f(u)$: the activation function (sigmoid), w_1, w_2 : the input-hidden and hidden-output connection weights, θ_1, θ_2 : the hidden and output bias, and x_0 : one-dimensional continuous input signal. Fig. 5(b) shows the output unit x_2 as a function of the input signal x_0. The derivative of the output of the network x_2 with respect to the input signal x_0 is written as

$$\frac{\partial x_2}{\partial x_0} = w_2\, f'(u_1)\, w_1, \qquad (6)$$

where u_1 : the internal state of the hidden unit. The maximum value of $f'(u)$ is not so large, such as 0.25 when the temperature of the sigmoid function is 1.0 and the value range is from 0.0 to 1.0. Then in order to approximate the discontinuous mapping, w_1 or w_2 has to be very large. However, by the error back propagation learning, both w_1 and w_2 become large. The propagated error to the hidden layer is amplified by the hidden-output connection weight w_2, and the input-hidden connection weight w_1 and the bias of the hidden unit θ_1 suffer sharp fluctuations. The bias change is written as

$$\Delta\theta_1 = error\ w_2\, f'(u_1). \qquad (7)$$

Furthermore, since the boundary of the discontinuous mapping is approximated as θ_1 / w_1 from Eq. (5) as shown in Fig. 5, the approximated boundary moves according to not only the change of the bias θ_1, but also the change of the connection weight w_1. While, in the RBF-based networks, the center of each RBF unit is represented by only one parameter μ. From these reasons, the boundary of the discontinuous mapping approximated by the sigmoid-based neural network changes radically and as a result, the learning becomes unstable.

It is reported that the combination of reinforcement learning and neural network sometimes leads instability[2]. This originates from the fact that a reinforcement learning task sometimes requires a discontinuous mapping like the "mountain car problem" in [2]. We compared the reinforcement learning using a neural network between two types of input signals. The task is that the mobile robot with two visual sensors each of which has one-dimensional array of visual sensory cells, reaches a target object. The first type of input signals is direct visual input signals and the second one is a relative object location which is projected on the visual sensor. The direct visual signal can be considered as the localized signal of the object location. We have shown that the learning is fast and stable in the case of visual input signals from the simulation[3]. We have also shown that the region where a discontinuous mapping is required is magnified in the hidden layer in the neural network after reinforcement learning[4]. From the simple example of supervised learning when the visual sensory signals are the input and the object location is the training signal for the output of the network, the visual sensory space is reconstructed on the hidden layer and the smooth internal representation corresponding to the object location can be obtained through learning[5].

From these results, the localization is useful to learn the discontinuous mapping. The reason can be thought that a special region of the input space can be easily magnified by adjusting the input-hidden connection weights which come from only the corresponding region. And the magnification of the region (large input-hidden connection weight w_1 in Eq. (6)) around the discontinuous mapping makes the learning stable. Moreover, it may be another reason that the localization of the input space is not changed during learning.

3. Architecture of Gauss-Sigmoid Neural Network

From the above discussion, it is the best way to localize the continuous input space into some regions by using Gaussian, in which the location of each Gaussian is represented by only one parameter μ (center), and then the localized signals are dealt with by sigmoid-based multi-layered neural network. Now the Gauss-Sigmoid neural network is proposed

1. to achieve a high learning speed even in approximating a strong non-linear function,
2. to achieve a strong generalization on the internal representation space, and
3. to be stable even in the learning of discontinuous mappings.

The architecture is shown in Fig. 2. When the center and the size of the RBF unit is trained, the learning rate for this parameter has to be very small to keep the stability in the learning.

4. Simulation

In this section, the ability of each network is compared by simulation. Here the input space has two dimensions and the distribution of the training signal is as shown in Fig. 7(a). The training signal on the stripe and circle region is 0.4 and that on the other space is –0.4. The value range of sigmoid function is from –0.5 to 0.5. The learning of the stripe region is easy for the sigmoid-based neural network, while the learning of the circle region is easy for the RBF-based network. The number of the Gaussians of the Gauss-Sigmoid neural network(NN) is 9 or 36, and that of the RBF network or Gaussian soft-max network is 36. The number of parameters in the Gauss-Sigmoid NN with 9 RBF units is almost equal to that of the RBF network or Gaussian soft-max network. The number of the units in each layer of both the Gauss Sigmoid NN and the Sigmoid NN is 2-9(36)-10-4-1. The learning rate for each network, that is chosen through trial and errors to minimize the final error, is as shown in the column (a) in Table 1. The initial hidden-output connection weights are 0.0, and the other connection weights are small random numbers. The initial center of each RBF unit is decided so as that the input space is just covered by all the units and the size σ is decided to be equal to the distance to the closest neighbor units. When the number of RBF units is 36, the center and size of each unit are as in Fig. 8 (a). Input pattern is chosen randomly at each step. The learning rate is reduced to one tenth after 50000 steps. Furthermore, at 80000 steps, all the hidden-output connection weights are reset to 0.0, and the training signal is inverted from –0.4 to 0.4 and from 0.4 to –0.4.

Fig. 6 shows the comparison of the learning curve that is the average of ten trials. The vertical axis indicates the sum of the square of the difference between

Table 1 Learning rate for each network

Network		(a) simullation 1	(b) simullation 2
Gauss-Sigmoid network	sigmoid	0.3	0.3
	gaussian	0.001	0.001
RBF network	linear	0.05	0.05
	gaussian	0.02	0.01
Gaussian Soft Max network	linear	0.05	0.05
	gaussian	0.01	0.0003
Sigmoid-based NN		0.2	0.2

the output of the network and the training signal at 776 input signals on the 26x26 lattice. In the very early stage where we cannot see from the figure, the error was reduced fastest in the RBF network. The Gauss-Sigmoid NN is also better. It is known from the figure that the learning is the slowest in the sigmoid-based NN, and the Gauss-Sigmoid NN with 9 RBF units is the next slowest. But after some steps, the error was not reduced enough in the RBF network and reduced gradually in the sigmoid-based NN and the Gauss-Sigmoid NN with 9 RBF units. When the training signals were inverted, the error of every network becomes large. Then the error was reduced slowly in the RBF network and the Gaussian soft-max network, while fast in the sigmoid-based NN and the Gauss-Sigmoid NN. This indicates that the internal representation in the both networks works effectively. The order of the network with respect to the final error is Gauss-Sigmoid NN with 36 units, Gauss-Sigmoid NN with 9 units, Gaussian soft-max network, sigmoid-based NN and then RBF network. The reason why the sigmoid-based NN is not unstable even if the approximated function includes discontinuity is that the learning rate for the network is small enough, and so the learning speed is very slow.

The next simulation was performed to show the effectiveness of the internal representation,. At first, each network is trained by the same training signal as the previous simulation for 20000 steps. Then the hidden-output connection weights are reset to 0.0 and the training signals are inverted. Each network is trained again for 20000 steps. This set of resetting the weights, inverting the training signals and training for 20000 steps, is repeated 48 times so as that the internal representation can be formed more clearly. Finally the hidden-output connection weights are reset again and the output is trained only for two input patterns. The learning rate is decided by trial and errors as in Table 1 (b). Fig. 7 shows the comparison of the learning and generalization ability in wide meaning. The upper four figures in Fig. 7 show the output distributions after 48 sets of the regular learning. The lower four are the output distributions after the training for only two input patterns whose locations are indicated as the two small circles in each figure. It is clear that the RBF-based network has poor generalization ability, while the Gauss-Sigmoid NN and the sigmoid-based NN has a good generalization ability due to the internal representation in their hidden layers. But this

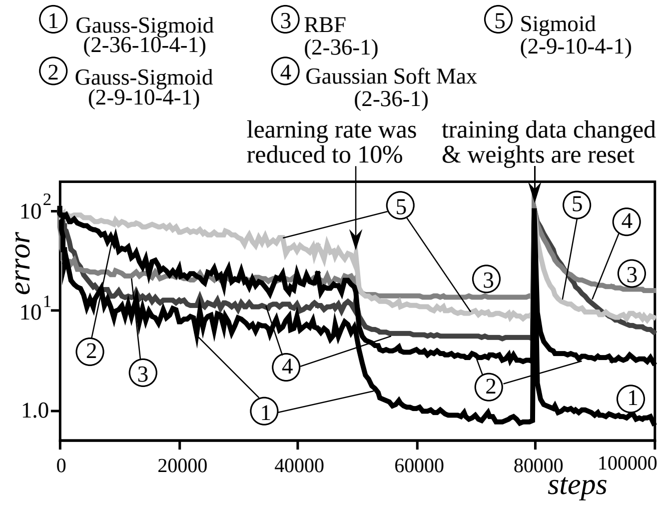


Fig. 6 Comparison of the learning curves between some networks. Gauss-Sigmoid NN with 36 RBF units learns fast and realize the best approximation.

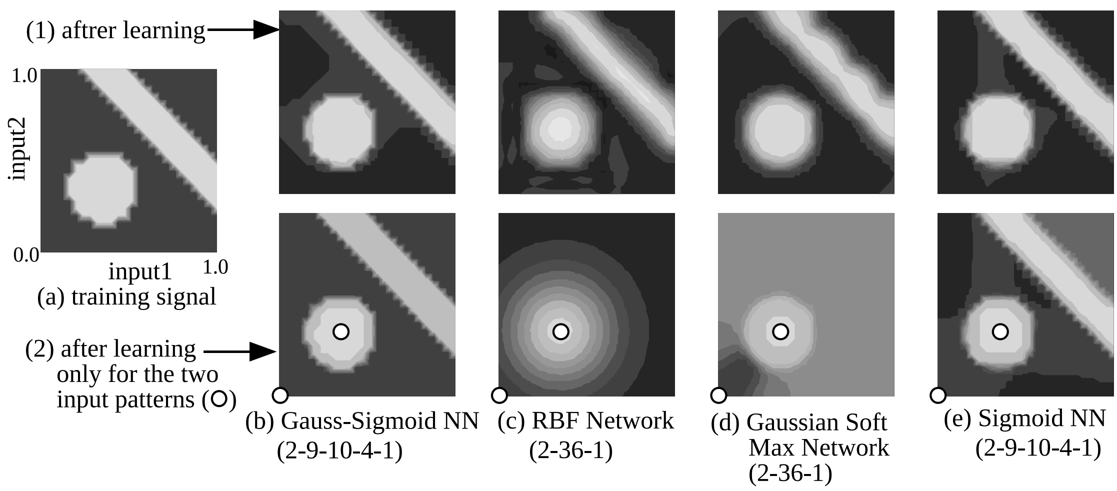
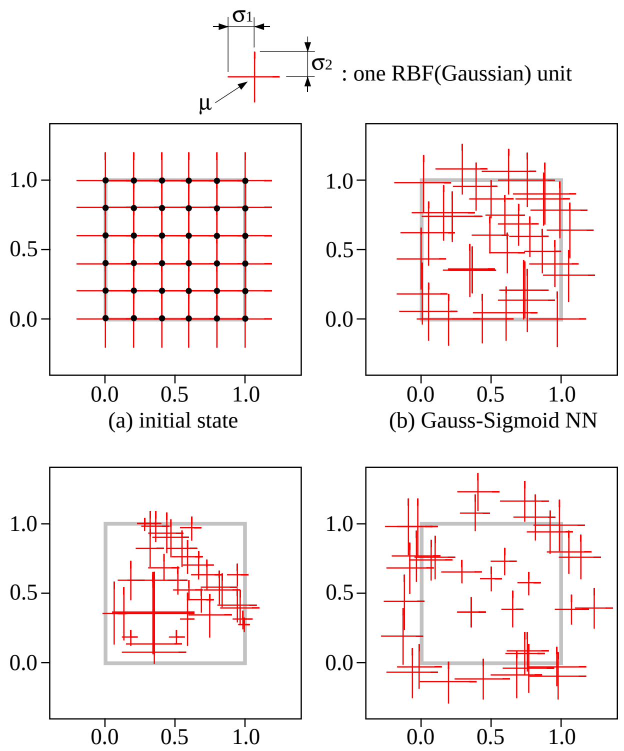


Fig. 7 Comparison of the output distribution after learning and generalization ability in the wide meaning. The RBF-based networks cannot approximate correctly only by the learning of two input patterns because it does not have internal representation.

generalization ability sometimes did not appear due to the initial connection weights between hidden layers. The reason may be that the circle region and stripe region are represented more independently on the top hidden layer. By increasing the number of the layers or decreasing the number of the units in the top hidden layer was effective to avoid these state.

Fig. 8 shows the center and size of each RBF unit for each of the Gauss-Sigmoid NN, the RBF network and the Gaussian soft-max network after the second simulation when the number of the RBF units is 36. It is seen that in the RBF network, the location of each RBF unit is affected strongly by the distribution of the given training signal, and the centers of almost all units are within the range of the input signals that is shown by a gray-framed rectangle in each figure. While, in the Gaussian soft-max network, the RBF units are distributed even out of the input range. In the Gauss-Sigmoid NN, the change of the location of each RBF unit from the initial location is the smallest. When the center and the size of each RBF unit is fixed as the initial state (see Fig. 8 (a)), the learning performance is still good in the Gauss-Sigmoid NN. This suggests that the sigmoid network in the Gauss-Sigmoid NN has an ability to integrate the localized information and reconstruct the input space very flexibly.



(a) initial state (b) Gauss-Sigmoid NN (c) RBF network (d) Gaussian Soft Max Network

Fig. 8 The change of the center and size of each of 36 RBF units by learning for each network.

5. Conclusion

The Gauss-Sigmoid neural network has been proposed, in which the continuous input space is localized into some regions by using RBF (Gaussian) units where the location on the input space is represented by only one parameter, and then the localized signals are dealt with by sigmoid-based multi-layered neural network. It was examined through the simulation that a high learning speed, strong generalization on the internal representation space, and the stability even in the case of discontinuous mapping approximation are simultaneously realized in this network. It can be said that this network has both advantages of RBF-based network and sigmoid-based multi-layer neural network.

The known problem that has to be considered is that when the dimension of the input space is large, we face the curse of dimensions. This is also the problem for RBF-based network.

Acknowledgement

This research was supported by the Scientific Research Foundation of the Ministry of Education, Science, Sports and Culture of Japan (#10450165) and by "The Japan Society for the Promotion of Science" as "Biologically Inspired Adaptive Systems" (JSPS-RFTF96I00105) in "Research for the Future Program".

Reference

[1] Doya, K., “Efficient Nonlinear Control with Actor-Tutor Architecture”, in Advanced in Neural Information Processing Systems, Vol. 9, pp. 1012-1018, The MIT Press (1997)

[2] Boyan, J. A. & Moore, A. W., “Generalization in Reinforcement Learning: safely approximating the value function”, in Advances in Neural Information Processing Systems, Vol. 7, pp. 369-376, The MIT Press (1995)

[3] Shibata, K., Okabe, Y. & Ito, K., “Direct-Vision-Based Reinforcement Learning Using a Neural Network”, material of Emergent System Symposium sponsored by SICE (1998.8) (in Japanese)

[4] Shibata, K., Ito, K. & Okabe, Y., “Direct-Vision-Based Reinforcement Learning in "Going to an Target" Task with an Obstacle and with a Variety of Target Sizes”, Proc. of NEURAP'98, pp. 95-102 (1998)

[5] Shibata, K. & Ito, K., “Reconstruction of Visual Sensory Space on the Hidden Layer in Layered Neural Network”, Proc. of ICONIP’98, pp.405-408 (1999)