



Comparison of Pattern Discrimination Mechanisms of Hebbian and Spatiotemporal Learning Rules in Self-Organization

Hikomichi Tsukada^{1,2*} and Minoru Tsukada³

¹ College of Engineering, Chubu University, Kasugai, Japan, ² Neural Computation Unit, Okinawa Institute of Science and Technology Graduate University, Okinawa, Japan, ³ Brain Science Institute, Tamagawa University, Tokyo, Japan

The spatiotemporal learning rule (STLR) proposed based on hippocampal neurophysiological experiments is essentially different from the Hebbian learning rule (HEBLR) in terms of the self-organization mechanism. The difference is the self-organization of information from the external world by firing (HEBLR) or not firing (STLR) output neurons. Here, we describe the differences of the self-organization mechanism between the two learning rules by simulating neural network models trained on relatively similar spatiotemporal context information. Comparing the weight distributions after training, the HEBLR shows a unimodal distribution near the training vector, whereas the STLR shows a multimodal distribution. We analyzed the shape of the weight distribution in response to temporal changes in contextual information and found that the HEBLR does not change the shape of the weight distribution for time-varying spatiotemporal contextual information, whereas the STLR is sensitive to slight differences in spatiotemporal contexts and produces a multimodal distribution. These results suggest a critical difference in the dynamic change of synaptic weight distributions between the HEBLR and STLR in contextual learning. They also capture the characteristics of the pattern completion in the HEBLR and the pattern discrimination in the STLR, which adequately explain the self-organization mechanism of contextual information learning.

Keywords: spatiotemporal learning rule, STLR, contextual learning, context-dependent memory, self-organization, pattern discrimination, pattern completion, Hebbian learning rule

OPEN ACCESS

Edited by:

Alessandro E. P. Villa,
Neuro-Heuristic Research Group,
Switzerland

Reviewed by:

Izabella Silkis,
Institute of Higher Nervous Activity
and Neurophysiology (RAS), Russia
Yoshio Sakurai,
Doshisha University, Japan

*Correspondence:

Hikomichi Tsukada
tsukada@isc.chubu.ac.jp

Received: 31 October 2020

Accepted: 26 January 2021

Published: 29 March 2021

Citation:

Tsukada H and Tsukada M (2021)
Comparison of Pattern Discrimination
Mechanisms of Hebbian and
Spatiotemporal Learning Rules in
Self-Organization.
Front. Syst. Neurosci. 15:624353.
doi: 10.3389/fnsys.2021.624353

1. INTRODUCTION

Learning is the embedding of information from the outside world into the changes in the connections between neurons in a neural network (changes in synaptic weights) based on the correlation of neural activity. The representation of information in neural networks through learning is called self-organization. In general, learning a neural network requires setting all synaptic weights in the network to initial values before training. The synaptic weight is generally initialized with a random number. Therefore, when the synapse weight vector of each neuron is normalized, it is converted into a unit vector in an n-dimensional space and uniformly distributed near the hypersphere. In contrast, the input vectors in the neural network are usually not evenly distributed

and are grouped into relatively small sections of the hypersphere surface. Considering the case where the input series are dynamically input from the external environment, each input vector in the series will be similar to each other. These input vectors are discriminated into different categories or integrated into one category in the information processing of memory in the brain. Therefore, these two contradictory functions are working in the memory system of the brain. Here we discuss the possibility that the spatiotemporal learning rule (STLR) and Hebbian learning rule (HEBLR) have such a functional role in memory and learning processing, and we clarify their characteristics.

The HEBLR (Hebb, 1949) modifies neural network connections according to the instantaneous external information when an output neuron fires. That is, if the output neuron does not fire, nothing occurs. This learning rule is widely used in unsupervised learning in neural networks (Widrow, 1959; Rosenblatt, 1962; Grossberg, 1974; Kohonen, 1988). In contrast, the STLR proposed by Tsukada et al. enables learning according to the information structure of the external environment without firing the output neurons (Tsukada et al., 1996, 2007; Tsukada and Pan, 2005). This enables a significantly flexible representation of information. The difference between the two learning rules was clarified using a simulation model of a neural network trained with relatively similar spatiotemporal pattern sequences (Tsukada and Pan, 2005). The HEBLR has the property of completing multiple similar input sequences into a single output pattern, whereas the STLR can discriminate multiple similar input sequences into separate output patterns. That is, the HEBLR has a pattern completion function and STLR has a pattern discrimination function for similar context input pattern sequences.

This study explains the difference between the two rules based on the self-organization mechanisms of the training vector and the synaptic weight vector. The clarification of the mechanism helps us to understand the use of the learning rule based on physiological experiments for the representation of information in learning and memory networks and also contributes to the applications in brain-inspired artificial intelligence.

2. METHODS

2.1. Neuron Model

Figure 1 shows a neuron used as the basic unit of a learning neural network. The set of inputs, $X = (x_1, \dots, x_N)$, from an external network or existing networks is applied to a post-synaptic neuron. Each x is multiplied by the synaptic weight, w , and summed. Then, the postsynaptic membrane potential, s , is given by the following equation:

$$s = w_1x_1 + w_2x_2 + \dots + w_Nx_N. \tag{1}$$

Here, the output y is given by the following equation with the output function, F , and the threshold value, η :

$$y = F(s - \eta), \tag{2}$$

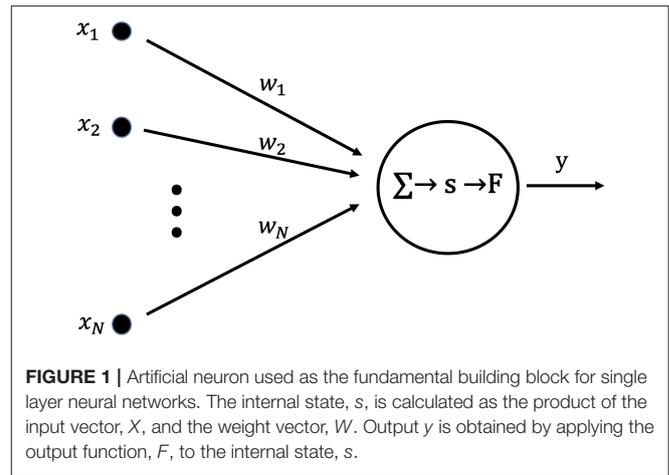


FIGURE 1 | Artificial neuron used as the fundamental building block for single layer neural networks. The internal state, s , is calculated as the product of the input vector, X , and the weight vector, W . Output y is obtained by applying the output function, F , to the internal state, s .

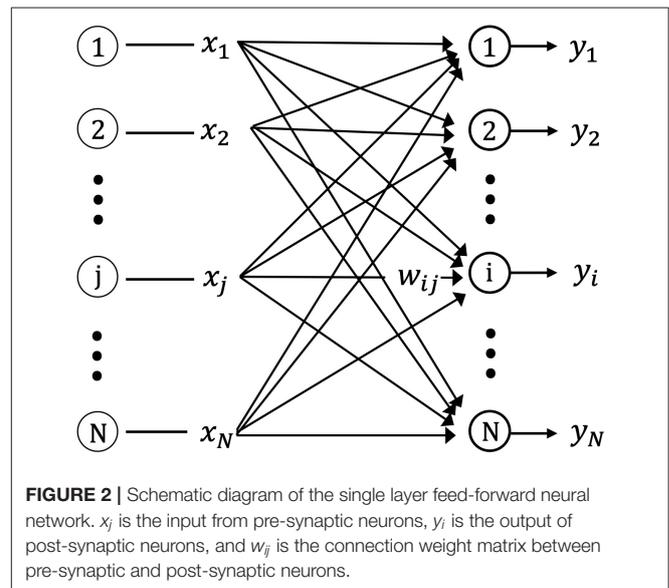


FIGURE 2 | Schematic diagram of the single layer feed-forward neural network. x_j is the input from pre-synaptic neurons, y_i is the output of post-synaptic neurons, and w_{ij} is the connection weight matrix between pre-synaptic and post-synaptic neurons.

$$F(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0. \end{cases} \tag{3}$$

2.2. Network Architecture

Figure 2 shows the one-layer neural network used in this study for storing the spatiotemporal pattern (sequences of input vectors) using unsupervised learning. The network is composed of N neurons with feed-forward excitatory connections. Each neuron is connected to the source (input) neurons via excitatory synapses. The input neuron, j ($j = 1, 2, \dots, N$), connects the output neuron, i ($i = 1, 2, \dots, N$), via the synaptic weight, w_{ij} .

The input signals at time t_k , $X(t_k)$, consist of a spatial pattern with the n dimensional binary elements, (x_1, x_2, \dots, x_N) . For example, a neuron, i , with a synapse weight vector, $W_i(t_k)$, for an input vector, $X(t_k)$, at a certain time, t_k , is shown in Figure 2. Neuron i has the sequence of synapse weight vectors $\{W_i(t_1), W_i(t_2), \dots, W_i(t_k)\}$ for the input vector

sequence, $\{X(t_1), X(t_2), \dots, X(t_k)\}$. $s_i(t_k)$ is expressed by the following equation for the internal state of neuron i at time t_k :

$$s_i(t_k) = \sum_{j=1}^N w_{ij}(t_k) x_j(t_k) = W_i(t_k) X(t_k). \quad (4)$$

This value is calculated for each neuron, i , in the network. Following the calculation for $s_i(t_k)$, the output function, F , and the threshold, η , is used to yield the output, $y(t_k)$, using the following equations:

$$y_i(t_k) = F(s_i(t_k) - \eta), \quad (5)$$

$$F(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0. \end{cases} \quad (6)$$

2.3. Input Spatiotemporal Pattern

The spatiotemporal pattern used in this simulation is a sequence of five spatial patterns $\{A_1, A_2, A_3, A_4, A_5\}$ with a Hamming distance ($HD = 10$) between them, comprising input vector sequences $\{X(t_1), X(t_2), \dots, X(t_5)\}$. The vectors consist of n elements ($N = 120$), with each element randomly selected as “1” or “0” and the total number of “1” s is constant over different spatial patterns. This value is constant (for this simulation, half of the elements of one vector are “1” and half are “0”). To evaluate the learning of the input information learned in the weight space owing to the change of the series, two types of input series vectors are considered as training sets: one is each sequence having the same spatial patterns, $\{X(t_1) = X(t_2) = X(t_3) = X(t_4) = X(t_5) = A_i | i \in (1, \dots, 5)\}$, and the other is the sequences having the different spatial patterns, which consist of four randomly chosen elements from five spatial patterns $\{A_1, A_2, A_3, A_4, A_5\}$ and replaced the first four input vectors $\{X(t_1), X(t_2), X(t_3), X(t_4)\}$ in each sequence (Figure 3). The two types of sequences are applied to the network as a training set, and the weight vector distributions for the HEBLR and STLR are compared.

2.4. Learning Rules: HEBLR and STLR

2.4.1. HEBLR Algorithm

Hebb (1949) proposed that a synaptic connection between the input and output neurons is modified whenever both neurons fire. This is considered to strengthen a synapse according to the correlation between the neuron excitation levels of both neurons. This is expressed by the following equation:

$$w_{ij}(t_{k+1}) = w_{ij}(t_k) + \alpha y_i(t_k) x_j(t_k). \quad (7)$$

The synaptic weight change of HEBLR is given by

$$w_{ij}(t_{k+1}) = \begin{cases} w_{ij}(t_k) + \delta w & s_i(t_k) \geq \eta \\ w_{ij}(t_k) & s_i(t_k) < \eta. \end{cases} \quad (8)$$

Here, $w_{ij}(t_k)$ is the synaptic weight from neuron j to neuron i at time t_k , and $w_{ij}(t_{k+1})$ is the weight at time t_{k+1} . $x_j(t_k)$ is the

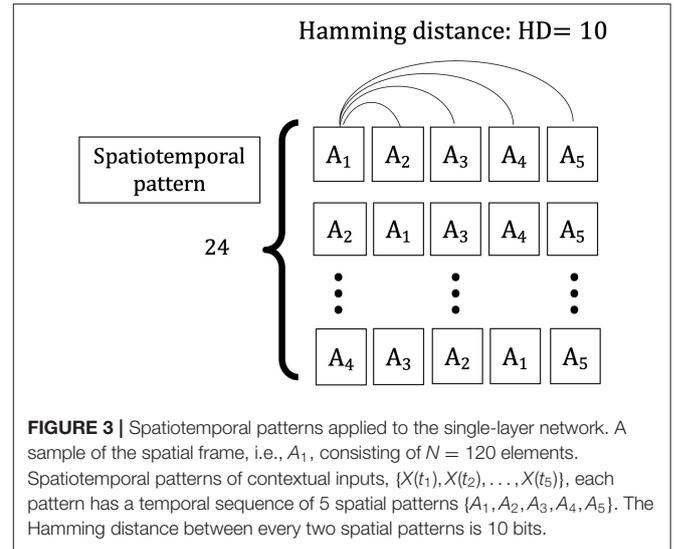


FIGURE 3 | Spatiotemporal patterns applied to the single-layer network. A sample of the spatial frame, i.e., A_1 , consisting of $N = 120$ elements. Spatiotemporal patterns of contextual inputs, $\{X(t_1), X(t_2), \dots, X(t_5)\}$, each pattern has a temporal sequence of 5 spatial patterns $\{A_1, A_2, A_3, A_4, A_5\}$. The Hamming distance between every two spatial patterns is 10 bits.

activity of the j th input neuron at time t_k , $y_i(t_k)$ is the activity of the i th output neuron at time t_k , and α is the training rate coefficient. $s_i(t_k)$ is the internal state of neuron i at time t_k . The weight change, $w_{ij}(t_{k+1}) - w_{ij}(t_k) = \delta w$ (constant), occurs under the condition of $y_i(t_k) = 1$ and $x_j(t_k) = 1$. $y_i(t_k) = 1$ is given by Equation (5) in the case $s_i(t_k) \geq \eta$.

2.4.2. STLR Algorithm

Changes in the synaptic weights in the STLR depend on “timing coherence” and “temporal history” of the input neuronal activity without firing of output neurons (Tsukada et al., 1994, 1996, 2007). The time history factor was omitted here for simplicity. The coincidence coefficient (Fujii et al., 1996), $I_{ij}(t_k)$, depends on the strength of the cooperative activity of the other input neurons when the spike of input neuron j reaches output neuron i at time t_k and is defined as follows:

$$I_{ij}(t_k) = w_{ij}(t_k) x_j(t_k) \sum_{m=1, m \neq j}^N w_{im}(t_k) x_m(t_k). \quad (9)$$

Here, under conditions where the network size is large and the firing of the incoming inputs is not sparse, the summation term is given by

$$\sum_{m=1, m \neq j}^N w_{im}(t_k) x_m(t_k) = s_i(t_k) - w_{ij}(t_k) x_j(t_k) \simeq s_i(t_k), \quad (10)$$

$$\therefore s_i(t_k) \gg w_{ij}(t_k) x_j(t_k).$$

Therefore, $I_{ij}(t_k)$ is approximated as follows:

$$I_{ij}(t_k) = w_{ij}(t_k) x_j(t_k) s_i(t_k). \quad (11)$$

The algorithm for changing synaptic weights by the STLR is given by

$$w_{ij}(t_{k+1}) = \begin{cases} w_{ij}(t_k) + \delta w & I_{ij}(t_k) \geq \theta_1 \\ w_{ij}(t_k) & \theta_1 > I_{ij}(t_k) > \theta_2 \\ w_{ij}(t_k) - \delta w & I_{ij}(t_k) \leq \theta_2. \end{cases} \quad (12)$$

Here, θ_1 and θ_2 are determined by the change in the intracellular Ca^{2+} concentration due to its influx through the N-Methyl-D-aspartate (NMDA) channel and release from the intracellular stores with high density inducing long-term potentiation (LTP) and low density inducing long-term depression (LTD); this is called the Bienenstock-Cooper-Munro (BCM) synapse modification rule (Bienenstock et al., 1982; Lacaille and Schwartzkroin, 1988).

3. COMPARISON OF LEARNING ALGORITHMS: HEBLR AND STLR

3.1. HEBLR Algorithm

The synaptic weight change $\Delta \hat{W}_i(t_k)$ of neuron i depends on the number of the increasing weight $N_i(t_k)$ when $y_i(t_k) = 1$ in the HEBLR. The equation is given by

$$\Delta \hat{W}_i(t_k) = \{y_i(t_k) \sum_j x_j(t_k)\} \delta w = N_i(t_k) \delta w. \quad (13)$$

Neurons with weight vector $W_i(t_k)$ that are significantly correlated with the input vector $X(t_k)$ are trained to reduce the difference between their weight vector and the input vector.

If the element vectors of the input vector series are identical $\{X(t_1) = X(t_2) = X(t_3) = X(t_4) = X(t_5) = A_i | i \in (1, \dots, 5)\}$, the group of neurons that satisfy $s_i(t_k) \geq \eta$ is the “winner” and the weights associated with the winning neuron are enhanced with training to yield the same output vector. Each distribution of the synapse weight vector has a one-peak distribution to approach the input vector, and the distribution sharpens as the training progresses. In contrast, for an input vector sequence whose differences between the input vectors have $HD = 10$, the common elements among the input vectors are strengthened, whereas the different elements among the input vectors are subdued as training progresses. Thereafter, the distribution of the synaptic weight vector converges to a one-peak distribution based on the common elements between the input vectors.

3.2. STLR Algorithm

The important point of STLR is that the amount of change in w_{ij} differs depending on the classification of I_{ij} by two different thresholds, θ_1 and θ_2 , which enables various learning. In relation to the input, x_j , STLR learning proceeds only when $x_j = 1$ (see Equation (11)). Then, the equation is given by

$$\hat{I}_{ij}(t_k) = w_{ij}(t_k) s_i(t_k). \quad (14)$$

This simplification helps us to understand how w_{ij} changes with learning, depending on the threshold. The two quantities mentioned above (I_{ij} and w_{ij}) are quite difficult for us to understand intuitively because the changes in their quantities due to learning are diverse (in position, orientation, and magnitude) in high-dimensional space. Therefore, we try to capture the whole picture of these quantities by considering a transformation with a one-to-one correspondence that projects these quantities into the interior of the unit sphere in high-dimensional space. This would

allow us to visualize the variation of w_{ij} and capture the important aspects of the change due to learning. According to this line of thinking, we normalize the input vector by its norm as follows:

$$\tilde{x}_j = \frac{x_j}{\|X\|}, \quad (15)$$

where

$$X = (x_1, x_2, \dots, x_j, \dots, x_N). \quad (16)$$

Similarly, normalization of the synaptic weight vector is

$$\tilde{w}_{ij} = \frac{w_{ij}}{\|W_i\|}, \quad (17)$$

where

$$W_i = (w_{i1}, w_{i2}, \dots, w_{ij}, \dots, w_{iN}). \quad (18)$$

Then, normalization of the internal state of neuron i is given by

$$\tilde{s}_i = \sum_{j=1}^N \tilde{w}_{ij} \tilde{x}_j. \quad (19)$$

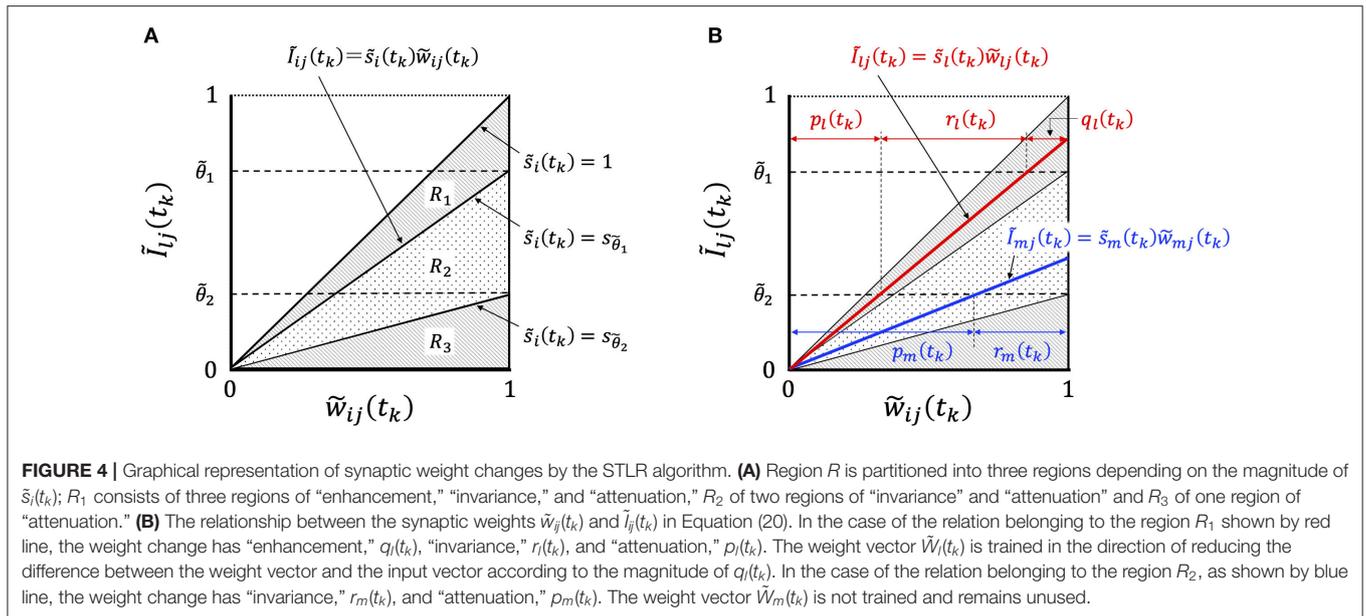
By the above normalization, $\hat{I}_{ij}(t_k)$ in Equation (14) is transformed into the interior of the unit sphere and on its surface, and is given by the following equation with two different thresholds, $\tilde{\theta}_1$ and $\tilde{\theta}_2$:

$$\tilde{I}_{ij}(t_k) = \tilde{w}_{ij}(t_k) \tilde{s}_i(t_k). \quad (20)$$

In order to consider the essential features of STLR, we focused on the relationship between $\tilde{I}_{ij}(t_k)$ and $\tilde{w}_{ij}(t_k)$, and showed the change in their relationship with learning in **Figure 4**.

The relationship between the synaptic weights $\tilde{w}_{ij}(t_k)$ and $\tilde{I}_{ij}(t_k)$ in Equation (20) is represented by straight lines passing through the origin of the coordinate axes with the gradient $\tilde{s}_i(t_k)$. The synaptic weights $\tilde{w}_{ij}(t_{k+1})$ can be classified into three types (R_1, R_2 , and R_3) of weight changes by the magnitude of $\tilde{s}_i(t_k)$: R_1 , which satisfies the condition $\tilde{s}_i(t_k) \geq \tilde{\theta}_1$, has the three components “enhancement,” “invariance,” and “attenuation”; R_2 ($\tilde{\theta}_1 > \tilde{s}_i(t_k) > \tilde{\theta}_2$) has the two components “invariance” and “attenuation”; and R_3 ($\tilde{s}_i(t_k) \leq \tilde{\theta}_2$) has a component “attenuation” (**Figure 4A**). We defined the group of weight changes, “enhancement,” q_i , “invariance,” r_i , and “attenuation,” p_i , which are normalized by the norm of weights for neuron i . When the normalized values at time t_k are $q_i(t_k)$, $r_i(t_k)$, and $p_i(t_k)$, respectively, then $q_i(t_k) + r_i(t_k) + p_i(t_k) = 1$. We explain the mechanisms of synaptic weight changes in **Figure 4B**.

When a series of input vectors are in the same context, $\{X(t_1) = X(t_2) = X(t_3) = X(t_4) = X(t_5) = A_i | i \in (1, \dots, 5)\}$, then the distribution of the synaptic weights changes dynamically as the training progresses. The synaptic weights of neurons are enhanced by the same inputs each time, yielding the same output vector. The synapse weight vectors are rotated in the direction of the input vector and produce a one-peak distribution around the



input vector. In this case, the characteristics are similar to those of the HEBLR.

In the case of the different input vector series (different context) whose differences among the input vectors have $HD = 10$, the synapse weight vectors are trained by the different inputs; the distribution is dependent on each input vector because each synaptic weight vector is trained by different input vectors. Therefore, the distribution becomes multimodal depending on the different input vectors, which is completely different from the distribution in the HEBLR.

To conclude, we assume that the distribution of synaptic weight vectors related to the output has similar characteristics to the synaptic weight vector distribution of HEBLR for learning the same context. However, the synaptic weight vectors are trained near the axis of different input vectors in learning different contexts. Thereafter, the distribution of their weight vectors is developed near each input vector. Therefore, the distribution of synaptic weight vectors is multi-modal, which is completely different from the distribution of synaptic weight vectors in HEBLR.

4. SIMULATION RESULTS

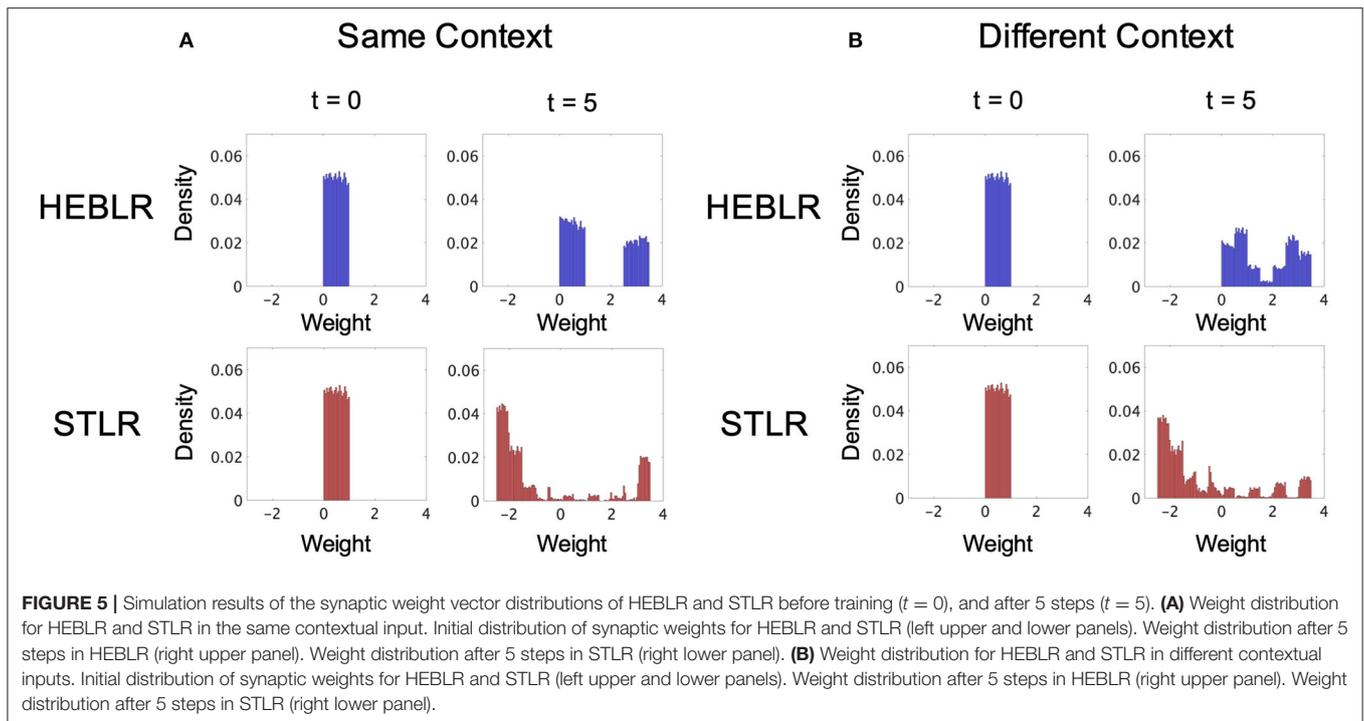
To investigate the difference between the HEBLR and STLR for a context input, we put the input series generated by combining spatial patterns (see section 2.3) into the network. We set the initial synaptic weights to be a uniform distribution, and compared the distributions after learning using each learning rule. To examine the effect of context learning, we considered two types of inputs: the same contextual input series $\{X(t_1) = X(t_2) = X(t_3) = X(t_4) = X(t_5) = A_1\}$ and different contextual input series $\{X(t_1) = A_1, X(t_2) = A_2, X(t_3) = A_3, X(t_4) = A_4, X(t_5) = A_5\}$.

4.1. Learning With Same Contextual Input Sequences

Figure 5A shows the synaptic weight distributions using the HEBLR (upper panel) and STLR (lower panel) for the same contextual input sequences.

The distribution of synaptic weights in the HEBLR represents a uniform distribution in the initial step ($t = 0$; upper left panel in Figure 5A) and a bimodal distribution after five steps ($t = 5$; upper right panel in Figure 5A). The simulation result is consistent with the results of the HEBLR algorithm described in section 3.1. The “enhancement” subset of synaptic weight vectors changes in the direction of the input vector after training and exhibits a unimodal distribution. The “invariant” subset remains untrained and appears as another unimodal distribution. The synaptic weights that were initially associated with output firing continue to be enhanced, whereas the other synapse weights remain unchanged. Therefore, the output vector of the network after five steps is similar to the vector enhanced by the initial firing.

In the STLR training, the distribution of synaptic weights represents a uniform distribution in the initial step ($t = 0$; lower left panel in Figure 5A) and a bimodal distribution after five steps ($t = 5$; lower right panel in Figure 5A). The bimodal distributions are mainly created by the “enhancement” subset trained in the direction of the input vectors and the “attenuation” subset trained the opposite direction of the input vectors. These two distributions become larger in the case of same contextual input sequences because the same input vector trains the same synaptic weight many times. Our simulation results also show a small distribution created by the “invariant” subset in between the two distributions (Figure 5A, lower right panel).



4.2. Learning With Different Contextual Input Sequences

Figure 5B shows the synaptic weight distributions using the HEBLR (upper panel) and STLR (lower panel) for the different contextual input sequences.

The distribution of synaptic weights in the HEBLR represents a uniform distribution in the initial step ($t = 0$; upper left panel in **Figure 5B**) and a bimodal distribution after five steps ($t = 5$; upper right panel in **Figure 5B**). The bimodal distribution represented the synaptic weights trained by the “enhancement” and “invariant” subsets, as discussed in section 3.1. Therefore, the weight distribution is similar to the weight distribution trained by the same context input (upper right panel in **Figure 5A**).

The distribution of synaptic weights in the STLR represents the multimodal distribution after five steps ($t = 5$; lower right panel in **Figure 5B**). This distribution differs from the distribution in the same contextual input sequences. There are mainly two large peaks in the positive and negative regions in the same contextual input sequences (lower right panel in **Figure 5A**), whereas there are the multiple peaks in the case of the different contextual input sequences (lower right panel in **Figure 5B**). These multiple peaks can be interpreted as the effect of learning by the different five inputs vectors. When different contextual input sequences are trained, the “enhancement” subset creates new distributions around each of the five different input vectors. Therefore, new synaptic weights corresponding to each input are enhanced and form multiple peaks. The two large peaks (positive and negative regions) corresponding to the “enhancement” and “attenuation” subsets created by the same input sequences decrease in size in the different contextual input sequences.

4.3. Comparison Between the HEBLR and STLR in the Same and Different Contextual Input Sequences

We compared the synaptic weight distributions trained by the HEBLR in the same and different contextual input sequences (upper right panel in **Figures 5A,B**). The synaptic weights associated with output firing initially continue to be enhanced (“enhancement” subset), whereas the other synapse weights remain unchanged (“invariant” subset) for both identical and different context input sequences. This characteristic in the HEBLR produces the bimodal distribution after five steps in both conditions. These results are consistent with the HEBLR algorithm described in section 3.1.

We also compared the synaptic weight distributions trained by the STLR in the same and different contextual input sequences (**Figures 5A,B**). The distribution of synaptic weights represents a bimodal distribution after five steps in the same contextual input sequences (lower right panel in **Figure 5A**) and a multimodal distribution after five steps in the different contextual input sequences (lower right panel in **Figure 5B**). The bimodal distributions in the same contextual input sequences are mainly created by the “enhancement” subset trained in the direction of the input vectors and the “attenuation” subset trained the opposite direction of the input vectors. These two distributions become larger because the same input vector trained the same synaptic weight several times. The distribution created by the “invariant” subset is also in between the two distributions; however, the size is quite small. In contrast, the multimodal distributions in the different contextual input sequences are also trained by the “enhancement” and “attenuation” subsets. However, the difference from the distribution in the same

contextual input sequences is in that the “enhancement” subset creates new distributions around each of the five different input vectors. This mechanism produces the multiple peaks. Therefore, the two large peaks (positive and negative regions) corresponding to the “enhancement” and “attenuation” subsets created by the same input sequences decreased in size, and the synaptic weights form the multimodal distribution in the different contextual input sequences.

Theoretically, the synaptic weights are trained by three types of conditions (“enhancement,” “invariant,” and “attenuation”) in the STLR, and our simulation results were consistent with those of the algorithm described in section 3.2.

5. DISCUSSIONS

Life improves itself based on the structure of the outside world, or rather, it would be better to say “creates”. This is self-organization and the basic principle of life. How the brain achieves this is a basic secret of the brain. Here, we focused on two learning rules (HEBLR and STLR), that have been confirmed in physiological experiments, in excitatory feed-forward neural networks, and clarified the differences of these two learning rules in the self-organization mechanisms for learning dynamic spatiotemporal information from the external world.

Hebb’s rule modifies the synaptic connections according to the instantaneous external information when output neurons are fired. Therefore, if they do not fire, the synaptic connections do not change anything. In contrast, the STLR enables modifying the synaptic connections according to the external environment without firing the output neurons and transforming the spatio-temporal input information into the synaptic weight space. That is, the STLR enables adaption to the information structure of the external world without requiring the self-ignition of neurons. This function develops a significantly flexible memory structure. Both learning rules are working in the hippocampus brain area (Tsukada et al., 1996, 2005, 2007).

We assumed the case that events in the natural world are related to each other in time. For example, we have to process information from the external environment dynamically when driving a car. If we made an error in the order of pressing the brake and the accelerator, this error can cause a serious accident. Therefore, it is important to consider how we can learn similar spatiotemporal contextual information that changes dynamically over time. To answer this question, we investigated the learning algorithms of HEBLR and STLR to identify their self-organization mechanism using relatively similar spatiotemporal context information, and we clarified the differences between them through our simulations. Here, we discuss the self-organization mechanisms of the two learning rules (HEBLR and STLR) from both the algorithm and simulation aspects.

5.1. Learning Algorithms

In the HEBLR, a group of cells fires when the inner product of the input vector and the weight vector exceeds the threshold, η ; then, the weights associated with the fired cells are strengthened. This means that the group of cells with weights similar to those

of the input vector is selected and further strengthened with learning each time. The weight vector changes in direction to decrease the distance from the input vector. That is, the weight vector is rotated in the direction of the input vector without significant changes in its size. In the HEBLR, cells with a weight vector that has a low correlation with the input vector do not fire. Consequently, these cells do not produce any output and do not learn.

The STLR learns input information, synchronization, and the degree of coincidence between input vectors. The coincidence coefficient depends on the inner product of the input vector and the synapse weight vector (Equation 9). Then, the threshold value, θ_1 and θ_2 , determines the increase or decrease of the synaptic weights (Equation 11). The synapse weight vector is uniformly and randomly distributed before learning and develops its weight distribution depending on the input vector sequence. The synaptic weight vectors in the STLR are created in mainly three distributions: a distribution with a peak around the input vector, one with a peak around the inversion vector of the input vector, and one with a peak in between them.

If the element vectors of the input vector series are identical $\{X(t_1) = X(t_2) = X(t_3) = X(t_4) = X(t_5) = A_i | i \in (1, \dots, 5)\}$, the group of neurons that satisfy $s_i(t_k) \geq \eta$ is the “winner,” and the weights associated with the winning neuron are enhanced via training to yield the same output vector as in the HEBLR. Each distribution of the synapse weight vector has a one-peak distribution to approach the input vector, and the distribution sharpens as the training progresses. In contrast, the distribution in the same contextual input sequences in the STLR has a peak in the distribution of positive regions, which is similar to the HEBLR. Both distributions are mainly created by the “enhancement” subset, which train the same synaptic weight many times in the direction of the input vectors.

For an input vector sequence whose differences between the input vectors have $HD = 10$, the common elements among the input vectors are strengthened, whereas the different elements among the input vectors are subdued as training progresses in the HEBLR. Thereafter, the distribution of the synaptic weight vector converges to a one-peak distribution depending on the common elements between the input vectors. In contrast, in the STLR, there are multiple peaks in the distribution of positive regions. These multiple peaks can be interpreted as the effect of learning by the different input vectors. When different contextual input sequences are trained, the “enhancement” subset creates new distributions around each of the input vectors. Therefore, new synaptic weights corresponding to each input are enhanced and form multiple peaks. This mechanism is the crucial difference between the HEBLR and STLR, and as well as the reason for why the STLR can discriminate similar input sequences and also have excellent pattern discrimination capability.

5.2. Verification Through Simulation

We applied input sequences with spatially similar patterns as in natural events to a feed-forward neural network and characterized the learning rules of the HEBLR and STLR by simulation. We set up the input vectors to have a difference of 10 bits between each other ($HD = 10$), where the cosine

distance between each input pattern is approximately 0.93. The synaptic weight vectors are trained to approach this new input axis when the input vector arrives. New synaptic weight vector groups near the input vectors are added by the learning rules, and the synaptic weight vector groups that are far from the input vectors deviate.

The HEBLR algorithm reinforces the synaptic weight vectors in a manner that reinforces the common vector elements of the five input vectors. Therefore, the common synaptic weight vectors are organized. The STLR training for different contextual inputs forms a multimodal distribution of synaptic weight vectors (Figure 5B, lower right panel). This result indicates that the STLR has features that are highly sensitive to different elements among the input vectors. The STLR training algorithms develop the synaptic weight vector distribution in the opposite direction of the input vector. The STLR can also preserve unlearned weight vectors to learn the differences between the next incoming inputs.

5.3. STLR and Synaptic LTP/LTD With the Change in Ca^{2+}

STLR is a learning rule based on the results of physiological (in vitro) experiments in which the coincidence of neural activity between input cells was controlled (Tsukada et al., 1996, 2007). Furthermore, STLR learns independently of posterior cell firing. This learning rule changes the induction of LTP/LTD depending on the degree of coincidence of neural activity between inputs. Therefore, STLR is closely related to LTP/LTD in physiological experiments.

It is well-known that NMDA receptors, a type of glutamate, play an important role in the induction of LTP in the CA1 area of the hippocampus and in the visual cortex; NMDA receptors are blocked by Mg^{2+} when the postsynaptic membrane potential is at rest, but are depolarized by high-frequency or associative inputs for prolonged periods of time. The NMDA receptor is blocked by Mg^{2+} when the postsynaptic membrane potential is quiescent, but when depolarization is prolonged by a high-frequency or associative input, the Mg^{2+} block is removed and large amounts of Ca^{2+} flow into the post-synapse from NMDA receptor-intrinsic channels. This increase in Ca^{2+} triggers a cascade of biochemical changes that cause long-term changes in transmission efficiency.

It has been clarified that Ca^{2+} is also involved in the induction of LTD. According to the experiment in which a Ca^{2+} chelator was injected into layer II/III neurons to chelate Ca^{2+} at the postsynaptic site in a rat visual cortex section specimen, Ca^{2+} chelated neurons that should cause LTP by the intensity and frequency of tetanic input induced LTD (Kimura et al., 1990). This explanation of switching between LTP/LTD demonstrates the validity of the Ca^{2+} binding hypothesis of Ca^{2+} /calmodulin-dependent protein kinase II (CaMKII) (Lisman, 1989) and protein dephosphorylating enzyme (Goto et al., 1985; Funauchi et al., 1992).

The retrograde messengers that cause transmitter release from the posterior to anterior synapses can induce LTP/LTD depending on the activity of the presynaptic terminal (Bliss

and Collingridge, 1993). The synaptic-induced increase in Ca^{2+} activates various protein kinases and causes the synthesis of retrograde messengers. After entering the synaptic cleft, these synthesized retrograde messengers act on the presynaptic terminal to induce LTP when the terminal is activated, or LTD when it is not. These physiological experiments suggest that the change in Ca^{2+} in the postsynapse triggers LTP/LTD.

It is assumed that in STLR, LTP/LTD is switched by changes in Ca^{2+} . The size of the coincidence coefficient (I_{ij}) between inputs changes the triggering of LTP/LTD (Equation 11). This learning rule is consistent with the results of physiological experiments (in vitro) in which phase congruency between input cells was controlled (Tsukada et al., 1996, 2007), and is also consistent with the results that LTP/LTD changes with the “common source” (Silkis, 1996). In addition, although omitted for simplicity, the STLR shows that LTP/LTD changes in response to past input history (Tsukada and Pan, 2005). In other words, the STLR produces LTP/LTD depending on the Ca^{2+} change caused by “common source” and “time history.” This is consistent with physiological experiments wherein the LTP and LTD are not dependent on the absolute size of Ca^{2+} but on the relative one (Abraham and Bear, 1996; Grassi et al., 1996; Silkis, 1996). From this point of view, the BCM model of floating thresholds (relative) by Bienenstock and colleagues, the unitary Hebbian modification rules by Silkis (1998), the learning rules by Frégnac (1991) and Tamura et al. (1992), and the hypothesis of Mayford et al. (1995) are also consistent with our learning rule.

In this study, we compared the characteristics of the HEBLR and STLR through algorithmic examination and simulation. The results suggested that the HEBLR and STLR have contradictory features of self-organization for series of inputs: pattern completion for the HEBLR and pattern discrimination for the STLR. The fact that both learning rules coexist in the hippocampus allows us to understand the remarkable mechanism of self-organization of biological memory neural networks. There is an interesting result that a single-layer neural circuit with feedforward-feedback excitatory connections can instantly learn the dynamic contextual information by integrating these two learning rules (HEBLR and STLR), and learn the information via one-shot learning with self-similar information compression in the memory system (Tsukada et al., 2016, 2018). This result might be one reason why our brains have adopted these two contradictory learning rules during the course of evolution. Our series of studies will contribute to the design of artificial intelligence and can be a promising method for information processing principles such as automatic driving systems that must learn in an external environment in a short period.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

The conception of the study was conducted by HT and MT. HT and MT participated in the design and modeling of the study, drafted and revised the manuscript critically for important intellectual content, and read and approved the manuscript. HT simulated and analyzed the data. MT interpreted and supervised the data. All authors contributed to the article and approved the submitted version.

REFERENCES

- Abraham, W. C., and Bear, M. F. (1996). Metaplasticity: the plasticity of synaptic plasticity. *Trends Neurosci.* 19, 126–130. doi: 10.1016/S0166-2236(96)80018-X
- Bienenstock, E., Cooper, L., and Munro, P. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *J. Neurosci.* 2, 32–48. doi: 10.1523/JNEUROSCI.02-01-00032.1982
- Bliss, T. V. P., and Collingridge, G. L. (1993). A synaptic model of memory: long-term potentiation in the hippocampus. *Nature* 361, 31–39. doi: 10.1038/361031a0
- Frégnac, Y. (1991). Computational approaches to network processing and plasticity. In: *Long-term Potentiation: A Debate of Current Issues*, eds M. Baudry, and J. L. Davis, Cambridge: MIT Press, 425–435.
- Fujii, H., Ito, H., Aihara, K., Ichinose, N., and Tsukada, M. (1996). Dynamical cell assembly hypothesis-theoretical possibility of spatio-temporal coding in the cortex. *Neural Netw.* 9, 1303–1350. doi: 10.1016/S0893-6080(96)00054-8
- Funahashi, M., Tsumoto, T., Nishigori, A., Yoshimura, Y., and Hidaka, H. (1992). Long-term depression is induced in Ca^{2+} /calmodulin kinase-inhibited visual cortex neurons. *NeuroReport* 3, 173–176. doi: 10.1097/00001756-199202000-00013
- Goto, S., Yamamoto, H., Fukunaga, K., Iwasa, T., Matsukado, Y., and Miyamoto, E. (1985). Dephosphorylation of microtubule-associated protein 2, π factor, and tubulin by calcineurin. *J. Neurochem.* 45, 276–283. doi: 10.1111/j.1471-4159.1985.tb05504.x
- Grassi, S., Pettorossi, V. E., and Zampolini, M. (1996). Low-frequency stimulation cancels the high-frequency-induced long-lasting effects in the rat medial vestibular nuclei. *J. Neurosci.* 16, 3373–3380. doi: 10.1523/JNEUROSCI.16-10-03373.1996
- Grossberg, S. (1974). Classical and instrumental learning by neural networks. *Prog. Theor. Biol.* 3, 51–141. doi: 10.1016/B978-0-12-543103-3.50009-2
- Hebb, D. O. (1949). *The Organization of Behavior; A Neuropsychological Theory*. Oxford: Wiley.
- Kimura, F., Tsumoto, T., Nishigori, A., and Yoshimura, Y. (1990). Long-term depression but not potentiation is induced in Ca^{2+} -chelated visual cortex neurons. *NeuroReport* 1, 65–68. doi: 10.1097/00001756-199009000-00018
- Kohonen, T. (1988). Optical Associative Memories. In: *Self-Organization and Associative Memory*. Springer Series in Information Sciences, Berlin; Heidelberg: Springer, 8. doi: 10.1007/978-3-662-00784-6_10
- Lacaille, J., and Schwartzkroin, P. (1988). Stratum lacunosum-moleculare interneurons of hippocampal CA1 region. II. Intracellular and intradendritic recordings of local circuit synaptic interactions. *J. Neurosci.* 8, 1411–1424. doi: 10.1523/JNEUROSCI.08-04-01411.1988
- Lisman, J. (1989). A mechanism for the Hebb and the anti-Hebb processes underlying learning and memory. *Proc. Natl. Acad. Sci. U.S.A.* 86, 9574–9578. doi: 10.1073/pnas.86.23.9574
- Mayford, M., Wang, J., Kandel, E. R., and O'Dell, T. J. (1995). Camkii regulates the frequency-response function of hippocampal synapses for the production of both LTP and LTD. *Cell* 81, 891–904. doi: 10.1016/0092-8674(95)90009-8
- Rosenblatt, F. (1962). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Cornell Aeronautical Laboratory. Report no. VG-1196-G-8. Spartan Books. doi: 10.21236/AD0256582
- Silkis, I. G. (1996). Long-term changes, induced by microstimulation of the neocortex, in the efficiency of excitatory postsynaptic transmission in the thalamocortical networks. *Neurosci. Behav. Physiol.* 26, 301–312. doi: 10.1007/BF02359032
- Silkis, I. G. (1998). The unitary modification rules for neural networks with excitatory and inhibitory synaptic plasticity. *Biosystems* 48, 205–213. doi: 10.1016/S0303-2647(98)00067-7
- Tamura, H., Tsumoto, T., and Hata, Y. (1992). Activity-dependent potentiation and depression of visual cortical responses to optic nerve stimulation in kittens. *J. Neurophysiol.* 68, 1603–1612. doi: 10.1152/jn.1992.68.5.1603
- Tsukada, H., Tsukada, M., and Isomura, Y. (2018). “A structure and function of hippocampal memory networks in consolidating spatiotemporal contexts,” in *Advances in Cognitive Neurodynamics (VI)*, eds J. M. Delgado-García, X. Pan, R. Sánchez-Campusano, and R. Wang (Singapore: Springer Singapore), 103–108. doi: 10.1007/978-981-10-8854-4_13
- Tsukada, H., Tsukada, M., and Tsuda, I. (2016). “A concept of spatiotemporal attractors,” in *Advances in Cognitive Neurodynamics (V)*, eds R. Wang and X. Pan (Singapore: Springer Singapore), 749–753. doi: 10.1007/978-981-10-0207-6_101
- Tsukada, M., Aihara, T., Kobayashi, Y., and Shimazaki, H. (2005). Spatial analysis of spike-timing-dependent LTP and LTD in the CA1 area of hippocampal slices using optical imaging. *Hippocampus* 15, 104–109. doi: 10.1002/hipo.20035
- Tsukada, M., Aihara, T., Mizuno, M., Kato, H., and Ito, K. (1994). Temporal pattern sensitivity of long-term potentiation in hippocampal CA1 neurons. *Biol. Cybern.* 70, 495–503. doi: 10.1007/BF00198802
- Tsukada, M., Aihara, T., Saito, H.-A., and Kato, H. (1996). Hippocampal LTP depends on spatial and temporal correlation of inputs. *Neural Netw.* 9, 1357–1365. doi: 10.1016/S0893-6080(96)00047-0
- Tsukada, M., and Pan, X. (2005). The spatiotemporal learning rule and its efficiency in separating spatiotemporal patterns. *Biol. Cybern.* 92, 139–146. doi: 10.1007/s00422-004-0523-1
- Tsukada, M., Yamazaki, Y., and Kojima, H. (2007). Interaction between the spatiotemporal learning rule (STLR) and Hebb type (HEBB) in single pyramidal cells in the hippocampal CA1 area. *Cogn. Neurodyn.* 1, 157–167. doi: 10.1007/s11571-006-9014-5
- Widrow, B. (1959). Adaptive sampled-data systems—A statistical theory of adaptation. *Wescon Conv. Rec., Inst. Radio Engrs, N. Y. Pt 4*, 74–85.

FUNDING

This work was supported by JSPS KAKENHI Grant Numbers JP17K00322, JP20H04246, and Brain/MINDS from AMED under Grant Number JP20dm0207088.

ACKNOWLEDGMENTS

The authors would like to thank Professors Shigetoshi Nara and Ichiro Tsuda for helpful discussions.