Fractal memory structure in the spatiotemporal learning rule

Takemori Orima[1]*, Ichiro Tsuda[2], Minoru Tsukada[3], Hiromichi Tsukada[4] and Yoshihiko Horio[1]

[1]Advanced Comprehensive Research Organization, Teikyo University, Itabashi, Japan, [2]AIT Center, Sapporo City University, Sapporo, Japan, [3]Brain Science Institute, Tamagawa University, Machida, Japan, [4]Center for Mathematical Science and Artificial Intelligence, Chubu University, Kasugai, Japan

The spatiotemporal learning rule (STLR) can reproduce synaptic plasticity in the hippocampus. Analysis of the synaptic weights in the network with the STLR is challenging. Consequently, our previous research only focused on the network's outputs. However, a detailed analysis of the STLR requires focusing on the synaptic weights themselves. To address this issue, we mapped the synaptic weights to a distance space and analyzed the characteristics of the STLR. The results indicate that the synaptic weights form a fractal-like structure in Euclidean distance space. Furthermore, three analytical approaches—multi-dimensional scaling, estimating fractal dimension, and modeling with an iterated function system—demonstrate that the STLR forms a fractal structure in the synaptic weights through fractal coding. These findings contribute to clarifying the learning mechanisms in the hippocampus.

1 Introduction

In the brain, the hippocampus plays a crucial role in forming episodic memories by storing spatiotemporal contexts (Kovcs, 2020). This process requires real-time, online learning, where patterns must be processed sequentially as they arrive, rather than being stored in a buffer for offline batch training (Melchior et al., 2024). However, current artificial neural networks often suffer from catastrophic forgetting, making online learning significantly more challenging than offline learning (McCloskey and Cohen, 1989). One reason for this limitation is their common reliance on Hebbian-type learning rule (Hebb, 1949). Hebbian-type learning, based on classical conditioning, updates synaptic weights according to the correlation between inputs and outputs. In contrast, the spatiotemporal learning rule (STLR) (Tsukada et al., 1996), discovered through physiological experiments in the hippocampus, differs fundamentally from Hebbian-type learning. The STLR updates synaptic weights based on correlations among inputs.

The STLR was proposed to model the synaptic plasticity in the CA1 region (Tsukada et al., 1996) as a specific learning rule for the hippocampal neurons. The CA1 region, as the output area of the hippocampus, encodes inputs from CA3 (Guardamagna et al., 2023) to set up associatively learned backprojections to the neocortex, allowing subsequent retrieval of information to the neocortex (Rolls, 2010). In this paper, the single-layer feedforward neural network with STLR synapses models this CA1 region.

Furthermore, it has been reported that networks utilizing the STLR demonstrate superior separation performance in tasks that are difficult for Hebbian learning rules (Tsukada and Pan, 2005). In this network, the synaptic weight values are updated using the coincidence among the inputs to the neuron connected through the STLR synapses.

The network learns slight differences in similar spatiotemporal patterns inputted from CA3 to CA1. These spatiotemporal patterns are time series consisting of binary spatial vectors with small Hamming distances (Tsukada and Tsukada, 2021a). After learning them, a histogram of the synaptic weight values shows a multimodal distribution reflecting the temporal order of the input spatial vectors (Tsukada and Tsukada, 2021a). In other words, the network learned slight differences in the temporal order of the constituent spatial patterns of the spatiotemporal inputs, which is not possible with the Hebbian learning rule. A histogram of the Hamming distances between the outputs also shows a multimodal distribution obtained by applying a specific readout pattern to the network (Tsuji et al., 2023a). Additionally, an order-nested structure is observed in the output space using a two-dimensional distance map created according to the temporal order of the input spatial vectors, and this structure has been analyzed in detail in our previous study (Tsuji et al., 2023a). The order-nested structure in the output space strongly suggests that the STLR formed a fractal structure in the synaptic weight space.

A fractal structure in the CA1 region by fractal coding has been previously discussed in the literature (Fukushima et al., 2007). The hippocampus uses fractal properties such as self-similarity and scale-invariance (Husain et al., 2022) to facilitate efficient information compression and processing (Bieberich, 2002; Werner, 2010; Tsuda, 2001; Kuroda et al., 2009). For example, fractal coding embeds high-dimensional spatiotemporal information into a low-dimensional synaptic weight space. Consequently, the hippocampus can efficiently use limited resources, such as the number of neurons and synaptic connections. This feature is also useful for engineering applications, such as edge devices (Orima et al., 2023). One example of fractal coding is a contraction-mapping system (Tsuda, 2001), which memorizes information as an attractor, driven by chaotic signals. Another example is a feedforward neural network (Yamaguti et al., 2011), which learns and recalls patterns generated by a Cantor set using Hebbian learning rule (Hebb, 1949) synapses. However, in these models, learning itself was not the primary mechanism for generating the fractal structure.

In this study, we confirm that the STLR can form a fractal structure solely through learning. Furthermore, we demonstrated that the order-nested structure in the output space is a reliable and practical indicator of the fractal structure in the synaptic weight space. The following three analysis methods were employed, as it was difficult to theoretically prove the formation of a fractal structure: The first method involved multi-dimensional scaling (MDS) (Torgerson, 1952) to qualitatively confirm the fractal structure in the synaptic weight space. Second, the box-counting (Mandelbrot, 1982; Sarkar and Chaudhuri, 1994), lacunarity (Mandelbrot, 1994; Smith et al., 1996), and mass-radius (Landini and Rippin, 1993) methods were used to quantitatively evaluate the fractal dimension and structure in the synaptic weight space. Third, modeling with an iterated function system (IFS) was employed for mathematical analysis. The following section briefly describes the STLR, neurons, network configuration, and initialization processes used in this study.

2 Materials and methods

2.1 Synapse with STLR

A single-layer feedforward neural network with the STLR synapses that models the connection from CA3 to CA1 (Tsukada and Pan, 2005) is shown in Figure 1. In this figure, large circles and small triangles represent neurons and synapses, respectively; $\mathbf{x}(t)$ and $\mathbf{y}(t)$ are the input and output column vectors of the network at discrete time t, respectively; $x_j(t)$ and $y_i(t)$ are the jth element of $\mathbf{x}(t)$ and ith element of $\mathbf{y}(t)$, respectively; M and N are the number of inputs and neurons, respectively; and $w_{ij}(t)$ is the synaptic weight connected from the jth input to the ith neuron.

The value of $w_{ij}(t)$ is updated based on the STLR (Tsukada et al., 1996) using the following equations:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t), \tag{1}$$

$$\Delta w_{ij}(t) = \begin{cases} \eta & (q_{ij}(t) > \theta_{\mathrm{LTP}}), \\ 0 & (\theta_{\mathrm{LTD}} \leq q_{ij}(t) \leq \theta_{\mathrm{LTP}}), \\ -\eta & (q_{ij}(t) < \theta_{\mathrm{LTD}}), \end{cases} \tag{2}$$

where η, θ_{LTP}, θ_{LTD}, and $q_{ij}(t)$ are the learning constant, long-term potentiation (LTP) threshold, long-term depression (LTD) threshold, and time history of the coincidence among the inputs to the ith neuron from the jth synapse (hereafter referred to as the input coincidence), respectively. The time history $q_{ij}(t)$ is defined as follows:

$$q_{ij}(t) = \sum_{t'=0}^{t} c_{ij}(t') \exp\left(-\frac{t-t'}{\tau_Q}\right), \tag{3}$$

$$c_{ij}(t) = w_{ij}(t)x_j(t) \sum_{j'=1, j'\neq j}^{M} w_{ij'}(t)x_{j'}(t),$$
$$= w_{ij}(t)x_j(t)(p_i(t) - w_{ij}(t)x_j(t)), \tag{4}$$

where τ_Q, $p_i(t)$, and $c_{ij}(t)$ are the time constants, the internal state of the ith neuron, and the input coincidence of the ijth synapse, respectively. The network learns the spatial and temporal structures of the input spatiotemporal patterns through $c_{ij}(t)$ and $q_{ij}(t)$, respectively.

2.2 Neuron

For simplicity, we employ an analog neuron model in the network shown in Figure 1 (Tsukada and Tsukada, 2021a). The internal state and output of the ith neuron in the network are expressed as

$$p_i(t) = \sum_{j=1}^{M} w_{ij}(t)x_j(t), \tag{5}$$

$$y_i(t) = f\left(p_i(t)\right) = \begin{cases} 1 & (p_i(t) \geq \theta), \\ 0 & (p_i(t) < \theta), \end{cases} \tag{6}$$

where $p_i(t)$ and $y_i(t)$ denote the internal state and output of the ith neuron, respectively; $f(\cdot)$ is the output function of the neuron (step function); and θ is the firing threshold of the neuron.
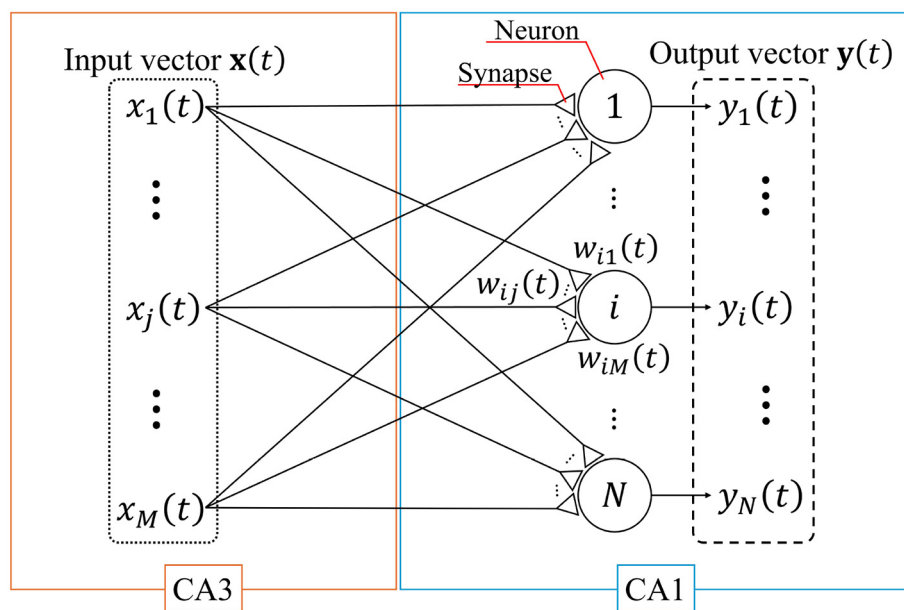
FIGURE 1
Single-layer feedforward neural network. This network models a connection from CA3 to CA1. The large circles and small triangles represent neurons and synapses, respectively. Synapses are updated based on the STLR. The input vector x(t) and output vector y(t) consist of M elements (enclosed by the dotted line) and N elements (enclosed by the dashed line), respectively.

2.3 Network configuration

The network shown in Figure 1 consists of the synapses and neurons, as described in Sections 2.1 and 2.2. The network learns the input spatiotemporal patterns that are slightly different (Section 2.4). We initialize the synaptic weights to the same initial values for all of the input spatiotemporal patterns. After learning each spatiotemporal pattern, the synaptic weights of the network are fixed, and a common readout pattern is input (Section 2.5).

In the following sections, we describe the procedure for generating spatiotemporal patterns and the details of the numerical simulation experiments.

2.4 Input spatiotemporal patterns

Input spatiotemporal patterns, each composed of a time series of binary spatial vectors with small Hamming distances from one another, are used to facilitate detailed analyses. The procedure for generating a spatiotemporal pattern $\mathbf{X}$ is as follows:

$$\mathbf{X} = \big(\mathbf{x}(1), \cdots, \mathbf{x}(t), \cdots, \mathbf{x}(T)\big), \tag{7}$$

$$\mathbf{x}(t) = [x_1(t), \cdots, x_j(t), \cdots, x_M(t)]^\top, \tag{8}$$

where $(\cdot)$ and $[\cdot]$ denote a vector time-series and vector, respectively, $\top$ denotes transposition, and T is the length of $\mathbf{X}$ (in this study, $T = 5$).

At time t, one of the spatial vectors is input as

$$\mathbf{x}(t) \in \{\mathbf{z}^1, \cdots, \mathbf{z}^k, \cdots, \mathbf{z}^K\}, \tag{9}$$

where $\{\cdot\}$, K, $\mathbf{z}^k$, and k denote a set, the number of spatial vectors, the spatial vector, and its index of the spatial vector, respectively.

In the present study, the three different spatial vectors ($K = 3$) are randomly generated under the following constraints:

$$\mathbf{z}^k = [z_1^k, \cdots, z_j^k, \cdots, z_M^k]^\top, \ 1 \le k \le 3, \tag{10}$$

$$z_j^k \in \{0, 1\}, \sum_{j=1}^M z_j^k = \frac{M}{2}, \tag{11}$$

$$\mathbf{z}^1 \ne \mathbf{z}^2, \mathbf{z}^2 \ne \mathbf{z}^3, \mathbf{z}^3 \ne \mathbf{z}^1. \tag{12}$$

Some elements of $\mathbf{z}^k$ are bit-flipped such that the Hamming distance between them is 10 (Tsukada and Tsukada, 2021a).

$$HD(\mathbf{z}^1, \mathbf{z}^2) = HD(\mathbf{z}^2, \mathbf{z}^3) = HD(\mathbf{z}^3, \mathbf{z}^1) = 10, \tag{13}$$

where $HD(\mathbf{x}, \mathbf{y})$ is the Hamming distance between the vectors $\mathbf{x}$ and $\mathbf{y}$. The total number L of spatiotemporal patterns is $K^T = 3^5 = 243$, which are generated using $\mathbf{z}^1$, $\mathbf{z}^2$, and $\mathbf{z}^3$ as follows:

$$\mathbf{X} \in \big\{(\mathbf{z}^1, \mathbf{z}^1, \cdots, \mathbf{z}^1), (\mathbf{z}^1, \mathbf{z}^1, \cdots, \mathbf{z}^2), \cdots, (\mathbf{z}^3, \mathbf{z}^3, \cdots, \mathbf{z}^2),$$
$$(\mathbf{z}^3, \mathbf{z}^3, \cdots, \mathbf{z}^3)\big\}. \tag{14}$$

Finally, the generated spatiotemporal patterns are labeled using k and arranged in increasing order. The labels of the spatiotemporal patterns are shown in Table 1. In this table, k_t^l denotes the index of the spatial vector at t in the lth spatiotemporal pattern.

2.5 Numerical simulation experiment

A numerical simulation experiment, consisting of a learning phase, a reading phase, and an evaluation, is subsequently conducted.

TABLE 1 Labels of the spatiotemporal patterns.

Label	Spatiotemporal pattern
1	$(\mathbf{z}^1, \mathbf{z}^1, \cdots, \mathbf{z}^1)$
2	$(\mathbf{z}^1, \mathbf{z}^1, \cdots, \mathbf{z}^2)$
$\vdots$	$\vdots$
l	$(\mathbf{z}^{k_1^l}, \cdots, \mathbf{z}^{k_t^l}, \cdots, \mathbf{z}^{k_5^l})$
$\vdots$	$\vdots$
242	$(\mathbf{z}^3, \mathbf{z}^3, \cdots, \mathbf{z}^2)$
243	$(\mathbf{z}^3, \mathbf{z}^3, \cdots, \mathbf{z}^3)$

Each pattern consists of a sequence of three spatial vectors. The notation k_t^l represents the index (1, 2, or 3) of the spatial vector $\mathbf{z}^k$ at a time t in the lth spatiotemporal pattern. The patterns are labeled in increasing order. These labels correspond to the arrangement of the horizontal and vertical axes in the distance maps shown in Figure 2.

First, for the learning phase, we set the initial values of the network and define a synaptic weight matrix. Additionally, we conduct experiments where white noise is applied to the input. The amplitude of the white noise is based on the standard deviation of the internal state of a neuron in the noise-free condition (for detailed statistics, such as the standard deviation of the internal state without noise, see Supplementary Section 1).

Second, for the reading phase, we define an output vector obtained by inputting the readout signal. The firing threshold of neurons is adjusted so that information embedded in the synaptic weights is exposed in the output.

Finally, for the evaluation, we create two-dimensional distance matrices using synaptic weight matrices and output vectors obtained in the learning and reading phases. To investigate the relationship between the synaptic weight and output structures, we calculate the similarity between the two-dimensional distance matrices. We further confirm the fractal structure of the synaptic weights using the MDS method and estimate the fractal dimensions using the box-counting, lacunarity, and mass-radius methods. We also analyzed the synaptic weight structure using Isomap (Tenenbaum et al., 2000) and UMAP (McInnes et al., 2020), which are common dimensionality reduction methods (for detailed results, see Supplementary Section 6). Additionally, based on the results of the MDS analysis, we mathematically demonstrate that a fractal structure is formed in the synaptic weight space through modeling with an IFS.

2.5.1 Learning phase

The network learns the L spatiotemporal patterns generated in Section 2.4. First, the initial values of the synaptic weights are set as follows:

$$w_{ij}(0) \sim U(-1, 1), \qquad (15)$$

where $U(a, b)$ denotes a uniform random number ranging from a to b. The other variables are initialized to zero, as $\Delta w_{ij}(0) = q_{ij}(0) = c_{ij}(0) = x_j(0) = p_i(0) = y_i(0) = 0$.

Second, the network learns one lth of the 243 spatiotemporal patterns. At $t = 1$, $\mathbf{x}(1) = \mathbf{z}^{k_1^l}$ is the input to the network, and the states of the neurons are updated using Equations 5 and 6.

The synaptic weight values are then updated using Equations 1–4. Similarly, $\mathbf{z}^{k_2^l}, \mathbf{z}^{k_3^l}, \cdots, \mathbf{z}^{k_T^l}$ are input to the network at $t = 2, 3, \cdots, T$ and the neuron states and synaptic weight values are updated at each time. At $t = T+1$, the update of the synaptic weight values is stopped and the synaptic weight matrix is defined as

$$\mathbf{W}^l(T+1) = \begin{bmatrix} w_{11}^l(T+1) & \cdots & w_{1j}^l(T+1) & \cdots & w_{1M}^l(T+1) \\ \vdots & \ddots & & & \vdots \\ w_{i1}^l(T+1) & & w_{ij}^l(T+1) & & w_{iM}^l(T+1) \\ \vdots & & & \ddots & \vdots \\ w_{N1}^l(T+1) & \cdots & w_{Nj}^l(T+1) & \cdots & w_{NM}^l(T+1) \end{bmatrix}. \qquad (16)$$

Numerical simulations are also performed when noise is applied to the input. In such cases, the noise amplitude is varied relative to the standard deviation of the internal state σ_p at $t = 1$ under noise-free conditions. For details on statistics of the network at $t = 1$, such as σ_p, see Supplementary Section 1.

2.5.2 Reading phase

To read out the information memorized in the synaptic weights, the readout pattern $\mathbf{z}^{\text{read}}$ is input to the network with fixed synaptic weights, where $\mathbf{z}^{\text{read}}$ satisfies the following equation:

$$\mathbf{z}^{\text{read}} \neq \mathbf{z}^1, \mathbf{z}^{\text{read}} \neq \mathbf{z}^2, \mathbf{z}^{\text{read}} \neq \mathbf{z}^3, \qquad (17)$$

$$HD(\mathbf{z}^{\text{read}}, \mathbf{z}^1) = HD(\mathbf{z}^{\text{read}}, \mathbf{z}^2) = HD(\mathbf{z}^{\text{read}}, \mathbf{z}^3) = 10. \qquad (18)$$

The synaptic weight values of the network are fixed to $\mathbf{W}^l(T+1)$, $\mathbf{z}^{\text{read}}$ is input to the network (e.g., $\mathbf{x}(T+1) = \mathbf{z}^{\text{read}}$), and the states of the neurons are updated using Equations 5 and 6.

The output vector is obtained as follows:

$$\mathbf{Y}^l(T+1) = [y_1^l(T+1), \cdots, y_i^l(T+1), \cdots, y_N^l(T+1)]^\top, \qquad (19)$$

where θ in Equation 6 is adjusted to ensure that the $N/2$ neurons are fired.

2.5.3 Evaluation

First, we investigate the relationship between $\mathbf{W}^l(t)$ and $\mathbf{Y}^l(t)$ in distance space. To analyze the structure of $\mathbf{W}^l(t)$, the Euclidean distances between the synaptic weight matrices are calculated using the following equation:

$$ED\left(\mathbf{W}^l(t), \mathbf{W}^{l'}(t)\right) = \sqrt{\sum_{i=1}^{N}\sum_{j=1}^{M}\left(w_{ij}^l(t) - w_{ij}^{l'}(t)\right)^2}. \qquad (20)$$

The distance matrix in the synaptic weight space $\mathbf{D}_{\text{w}}(t)$ is defined using $ED(\mathbf{W}^l(t), \mathbf{W}^{l'}(t))$ by arranging the rows and columns in

increasing order using l as follows:

$$
\mathbf{D}_{\mathrm{w}}(t) =
$$
$$
\begin{bmatrix}
ED\left(\mathbf{W}^1(t),\mathbf{W}^1(t)\right) & \cdots & ED\left(\mathbf{W}^1(t),\mathbf{W}^{l'}(t)\right) & \cdots & ED\left(\mathbf{W}^1(t),\mathbf{W}^L(t)\right) \\
\vdots & \ddots & & & \vdots \\
ED\left(\mathbf{W}^l(t),\mathbf{W}^1(t)\right) & & ED\left(\mathbf{W}^l(t),\mathbf{W}^{l'}(t)\right) & & ED\left(\mathbf{W}^l(t),\mathbf{W}^L(t)\right) \\
\vdots & & & \ddots & \vdots \\
ED\left(\mathbf{W}^L(t),\mathbf{W}^1(t)\right) & \cdots & ED\left(\mathbf{W}^L(t),\mathbf{W}^{l'}(t)\right) & \cdots & ED\left(\mathbf{W}^L(t),\mathbf{W}^L(t)\right)
\end{bmatrix}.
$$
$$(21)$$

In contrast, the distance matrix in the output space $\mathbf{D}_{\mathrm{y}}(t)$ obtained from the Hamming distance $HD(\mathbf{Y}^l(t),\mathbf{Y}^{l'}(t))$ between the output vectors is defined by

$$
\mathbf{D}_{\mathrm{y}}(t) =
$$
$$
\begin{bmatrix}
HD\left(\mathbf{Y}^1(t),\mathbf{Y}^1(t)\right) & \cdots & HD\left(\mathbf{Y}^1(t),\mathbf{Y}^{l'}(t)\right) & \cdots & HD\left(\mathbf{Y}^1(t),\mathbf{Y}^L(t)\right) \\
\vdots & \ddots & & & \vdots \\
HD\left(\mathbf{Y}^l(t),\mathbf{Y}^1(t)\right) & & HD\left(\mathbf{Y}^l(t),\mathbf{Y}^{l'}(t)\right) & & HD\left(\mathbf{Y}^l(t),\mathbf{Y}^L(t)\right) \\
\vdots & & & \ddots & \vdots \\
HD\left(\mathbf{Y}^L(t),\mathbf{Y}^1(t)\right) & \cdots & HD\left(\mathbf{Y}^L(t),\mathbf{Y}^{l'}(t)\right) & \cdots & HD\left(\mathbf{Y}^L(t),\mathbf{Y}^L(t)\right)
\end{bmatrix}.
$$
$$(22)$$

We further create two-dimensional distance maps using $\mathbf{D}_{\mathrm{w}}(t)$ and $\mathbf{D}_{\mathrm{y}}(t)$ and qualitatively confirm the structural similarity between the synaptic weight space and the output space. Furthermore, the cosine similarity is calculated to quantitatively analyze the structural similarity between $\mathbf{D}_{\mathrm{w}}(t)$ and $\mathbf{D}_{\mathrm{y}}(t)$ as follows:

$$CS(t) = \frac{\mathbf{D}_{\mathrm{w}}(t) \cdot \mathbf{D}_{\mathrm{y}}(t)}{\|\mathbf{D}_{\mathrm{w}}(t)\| \, \|\mathbf{D}_{\mathrm{y}}(t)\|}, \tag{23}$$

where $\cdot$ and $\|\mathbf{D}\|$ are the sum of the Hadamard product and Frobenius norm of matrix $\mathbf{D}$, respectively, and are calculated as

$$\mathbf{D}_{\mathrm{w}}(t) \cdot \mathbf{D}_{\mathrm{y}}(t) = \sum_{l=1}^{L}\sum_{l'=1}^{L} ED\left(\mathbf{W}^l(t),\mathbf{W}^{l'}(t)\right)$$
$$HD\left(\mathbf{Y}^l(t),\mathbf{Y}^{l'}(t)\right), \tag{24}$$

$$\|\mathbf{D}_{\mathrm{w}}(t)\| = \sqrt{\sum_{l=1}^{L}\sum_{l'=1}^{L} ED\left(\mathbf{W}^l(t),\mathbf{W}^{l'}(t)\right)^2}, \tag{25}$$

$$\|\mathbf{D}_{\mathrm{y}}(t)\| = \sqrt{\sum_{l=1}^{L}\sum_{l'=1}^{L} HD\left(\mathbf{Y}^l(t),\mathbf{Y}^{l'}(t)\right)^2}. \tag{26}$$

Second, the MDS method is used to clearly visualize the structure of $\mathbf{D}_{\mathrm{w}}(t)$ as it is an appropriate method for expressing the distances between the synaptic weight matrices in low dimensions. A reasonable compressed dimension n ($1 \leq n \leq L$) is estimated to reduce the dimensions to a lower-dimensional space without losing as much high-dimensional information as $\mathbf{D}_{\mathrm{w}}(t)$. The dimension-compressed distance matrix in the synaptic weight space $\mathbf{G}_{\mathrm{w}}(t)$ is defined as

$$\mathbf{G}_{\mathrm{w}}(t) = [\mathbf{g}^1(t),\cdots,\mathbf{g}^l(t),\cdots,\mathbf{g}^L(t)]^\top \in \mathbb{R}^{L\times n}, \tag{27}$$

$$\mathbf{g}^l(t) = [g_1^l(t),\cdots,g_m^l(t),\cdots,g_n^l(t)] \in \mathbb{R}^n, \tag{28}$$

where $\mathbf{g}^l(t)$ ($1 \leq l \leq L$) is a coordinate in the distance space compressed into n-dimensions, and $g_m^l(t)$ is the mth element in $\mathbf{g}^l(t)$. To estimate n, double centering is used to calculate a symmetric matrix $\mathbf{G}_{\mathrm{w}}(t)\mathbf{G}_{\mathrm{w}}(t)^\top$ as follows:

$$\mathbf{G}_{\mathrm{w}}(t)\mathbf{G}_{\mathrm{w}}(t)^\top = -\frac{1}{2}\left(\mathbf{I}-\frac{1}{L}\mathbf{J}\right)\mathbf{D}_{\mathrm{w}}(t)\left(\mathbf{I}-\frac{1}{L}\mathbf{J}\right), \tag{29}$$

where $\mathbf{I}$ and $\mathbf{J} \in \mathbb{R}^{L\times L}$ are defined as

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \quad \mathbf{J} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}. \tag{30}$$

The right-hand side of Equation 29 contains L eigenvalues. The sum of L eigenvalues $\Lambda_L = \sum_{m=1}^{L}\lambda_m$ is used as a criterion for estimating n, where λ_m is the mth eigenvalue and m is an index when the eigenvalues are sorted in descending order. In the left-hand side of Equation 29, the sum of the eigenvalues $\Lambda_n = \sum_{m=1}^{n}\lambda_m$ of $\mathbf{G}_{\mathrm{w}}(t)\mathbf{G}_{\mathrm{w}}(t)^\top$ is calculated using n eigenvectors obtained from the right-hand side of Equation 29. If $\Lambda_n \geq 0.5\Lambda_L$, n is a reasonably compressed dimension. The dimension-compressed position $\mathbf{g}^l(t)$, which is compressed into n-dimensions, is labeled r according to the temporal order in $\mathbf{X}$ and is classified into Cluster r. By applying the MDS method to each Cluster r, the structure within the Cluster r was analyzed in greater detail. This operation is repeated $T-1$.

Third, we evaluate the fractal dimension and structure of $\mathbf{G}_{\mathrm{w}}(t)$ using the box-counting (Mandelbrot, 1982), lacunarity (Mandelbrot, 1994), and mass-radius (Smith et al., 1996) methods. However, it is difficult to apply these methods to $\mathbf{G}_{\mathrm{w}}(t)$ because $\mathbf{G}_{\mathrm{w}}(t)$ loses the high-dimensional information of $\mathbf{D}_{\mathrm{w}}(t)$. Therefore, $\mathbf{G}_{\mathrm{w}}(t)$ is adjusted to compensate for the lost information of $\mathbf{D}_{\mathrm{w}}(t)$, and an adjusted distance matrix in the synaptic weight space $\mathbf{G}_{\mathrm{w}}^{\mathrm{adj}}(t)$ is derived. As an adjustment method, the center of the Cluster r is defined as CC_r and adjusted using the eigenvalues obtained from Equation 29. This adjustment compensates for the lost high-dimensional information, such as the distance between the clusters. The center CC_r of the Cluster r is adjusted to

$$CC_r^{\mathrm{adj}} = \alpha_n CC_r, \tag{31}$$

$$\rho_n = \frac{\Lambda_n}{\Lambda_L}, \tag{32}$$

$$\alpha_n = \frac{1}{\rho_n}, \tag{33}$$

where CC_r^{adj}, ρ_n, and α_n are the adjusted CC_r, the cumulative contribution ratio, and correction coefficient, respectively. The information loss rate is represented by $1-\rho_n$. For example, when $\rho_n = 0.8$, 20% of the high-dimensional information is lost. Therefore, in $\mathbf{D}_{\mathrm{w}}(t)$, the distances between the clusters are larger. To compensate for the lost information, CC_r is multiplied by α_n. After adjusting the centers of all clusters, the box-counting, lacunarity, and mass-radius methods are applied to $\mathbf{G}_{\mathrm{w}}^{\mathrm{adj}}(t)$ to estimate the fractal dimension.

Finally, based on the results obtained from numerical simulation experiments, we reproduce the fractal structure formed

in the synaptic weight space using an IFS. In numerical experiments, the input patterns increase exponentially with the number of spatial vectors K and their length T; thus, it is difficult to simulate and observe the detailed structure of the distance between the synaptic weights. Therefore, an IFS is used to reproduce the time evolution of distances between the synaptic weights obtained from the numerical simulation experiments, enabling observation of fine-scale structures and confirming that the STLR forms a fractal structure. We describe an IFS that operates on $\mathbf{g}(t)$, the two-dimensional representation of the synaptic weights $\mathbf{W}(t)$ obtained through the MDS method as

$$\mathbf{W}(t) \in \mathbb{R}^{N \times M} \xrightarrow{\text{MDS}} \mathbf{g}(t) \in \mathbb{R}^2, \tag{34}$$

$$\mathbf{g}(t+1) = \bigcup_k^K F_k(\mathbf{g}(t)), \tag{35}$$

$$F_k(\mathbf{g}(t)) = \mathbf{g}(t) + \gamma_k(t)[0 \ 1] \begin{bmatrix} \cos(\phi_k(t)) & \sin(\phi_k(t)) \\ -\sin(\phi_k(t)) & \cos(\phi_k(t)) \end{bmatrix}, \tag{36}$$

where F_k is a mapping function when $\mathbf{z}^k$ is input. $\gamma_k(t)$ and $\phi_k(t)$ is an expansion coefficient and a phase angle based on $\mathbf{g}(t)$ at $k = 1$, $\phi_1(t) = 0$, respectively. Furthermore, the vector $[0 \ 1]$ is multiplied to keep the mapping symmetric with respect to the y-axis. The IFS is constructed such that the geometric relationships in the low-dimensional space $\mathbf{g}(t)$ reflect the distance in the high-dimensional synaptic weights space $\mathbf{W}(t)$. For instance, if mapping functions F_1 and F_2 transform $\mathbf{g}(t)$, the resulting Euclidean distance between $\mathbf{g}^1(t+1)$ and $\mathbf{g}^2(t+1)$ corresponds to the Euclidean distance between the updated weights $\mathbf{W}^1(t+1)$ and $\mathbf{W}^2(t+1)$.

3 Results

In the following numerical simulations, the parameter values are set as $N = 120$, $M = 120$, $\eta = 2$, $\theta_{\text{LTP}} = 4$, $\theta_{\text{LTD}} = -2$, and $\tau_Q = 2.23$. These parameter values are the best learning performance in the previous studies (Tsuji et al., 2023a). Furthermore, these parameter values satisfy the necessary conditions for the network's synaptic weights to be updated (for details, see Supplementary Sections 1, 2, and 9).

3.1 Two-dimensional distance maps and cosine similarity analysis

The two-dimensional distance maps created using $\mathbf{D}_w(T+1)$ and $\mathbf{D}_y(T+1)$ are shown in Figures 2a and 2b, respectively. In Figures 2a and 2b, the vertical and horizontal axes represent the labels of the input spatiotemporal patterns.

In Figure 2a, the color bar represents $ED(\mathbf{W}^l(T+1), \mathbf{W}^{l'}(T+1))$. The higher values indicate the higher dissimilarity. When the network learns the same spatiotemporal pattern (i.e., $l = l'$), $ED(\mathbf{W}^l(t), \mathbf{W}^{l'}(t)) = 0$. By contrast, in Figure 2b, the color bar represents $HD(\mathbf{Y}^l(T+1), \mathbf{Y}^{l'}(T+1))$.

In Figure 2a, a yellow area appears at $ED(\mathbf{W}^l(T+1), \mathbf{W}^{l'}(T+1)) > 400$ when the network learns the different first spatial vectors

in the input spatiotemporal patterns (i.e., when $k_1^l \neq k_1^{l'}$). In contrast, three green rectangles appear at $ED(\mathbf{W}^l(T+1), \mathbf{W}^{l'}(T+1)) \approx 350$ when the network learns the same first spatial vectors in the input spatiotemporal patterns (i.e., when $k_1^l = k_1^{l'}$). Moreover, when the spatial vectors up to the second are the same (i.e., when $k_1^l = k_1^{l'}$ and $k_2^l = k_2^{l'}$), $3^2 = 9$ light-blue rectangles appear at $ED(\mathbf{W}^l(T+1), \mathbf{W}^{l'}(T+1)) \approx 200$. Consequently, a fractal-like structure is confirmed in the distance space of the synaptic weights, which reflects the temporal order of the spatial vectors in the input spatiotemporal patterns.

Figure 2b further shows an order-nested structure similar to that observed in Figure 2a. This finding suggests that the output space structure represents that of the synaptic weight space. In other words, the outputs of the network can be used to evaluate the synaptic weights.

In Figure 2, the cosine similarity in Equation 23 is used to quantitatively analyze the similarity between the structures of the synaptic weights and outputs in the distance space, resulting in $CS(T+1) = 0.959$. The average and variance of the cosine similarity are 0.959 and 3.73×10^{-5}, respectively, when the initial values of the synaptic weights are changed 1,000 times (Supplementary Figure S1), suggesting that the synaptic weights and outputs have similar structures in the distance space. Furthermore, even when the network parameter values are changed, the similarity between the synaptic weights and outputs in the distance space is high (for details, see Supplementary Section 3).

3.2 MDS analysis of fractal structures

The MDS method (Torgerson, 1952) is applied to visualize $\mathbf{D}_w(t)$ in a low dimension. A reasonable compression dimension $n = 2$ is obtained when $\Lambda_n \approx 0.5\Lambda_L$ (Supplementary Figure S5). Figure 3 shows the time evolution of $\mathbf{G}_w(t)$ obtained by compressing $\mathbf{D}_w(t)$ into two dimensions.

In Figure 3, the two horizontal and vertical axes represent the distances of $\mathbf{G}_w(t)$ and the discrete time t, respectively. In the horizontal axes, ED_{dim1} and ED_{dim2} represent the first and second distances in the dimension-compressed distance space, respectively. At time $t = 0$, red $\times$ indicates the initial positions of the synaptic weights in the dimension-compressed distance space. For the 243 spatiotemporal patterns, the initial values of the synaptic weights are set to the same values, and all positions of $\mathbf{G}_w(0)$ are $(ED_{\text{dim1}}, ED_{\text{dim2}}) = (0, 0)$. Subsequently, at time $t = 1$, the red $\times$ branches to the three blue points in the figure because three different spatial vectors exist in the spatiotemporal pattern. Furthermore, branching occurs at $t \geq 2$. Assuming that this branching occurs repeatedly, this suggests that a self-similar structure is formed in the distance space. In other words, we propose that a fractal structure is formed in the synaptic weight space.

We further analyze the structure shown in Figure 3 in detail using the MDS method, repeatedly applying it according to the temporal order of the input spatiotemporal patterns. Figure 4a shows $\mathbf{G}_w(t)$ at $t = 5$ from Figure 3.

In the figure, $\mathbf{g}^l(t)$ is labeled as r according to k_1^l. In other words, Clusters $r = 1, 2,$ and 3 when the network learns $\mathbf{X} = (\mathbf{z}^1, \cdots)$,
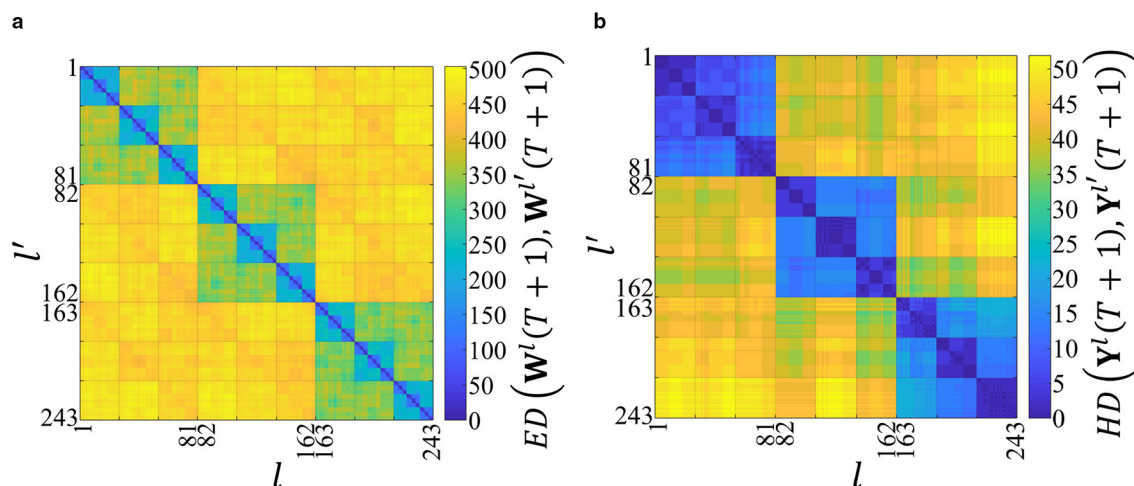
FIGURE 2
Structures in the distance space of the synaptic weights and the outputs after the network has learned 243 spatiotemporal patterns. The horizontal and vertical axes represent the labels of the input spatiotemporal patterns, and the color bars represent the distances. **(a)** Two-dimensional distance map using $D_w(T+1)$. The color of each point (l, l') represents the Euclidean distance $ED(W^l(T+1), W^{l'}(T+1))$ between the final weight matrices learned from patterns l and l'. The color scale indicates that warmer colors (yellow) represent greater distances. The distinct squares along the diagonal reveal a clear hierarchical and self-similar clustering, which we identify as a fractal-like structure. **(b)** Two-dimensional distance map using $D_y(T+1)$. The color represents the Hamming distance $HD(Y^l(T+1), Y^{l'}(T+1))$ between the corresponding output vectors. This map exhibits a similar hierarchical clustering, which we term an order-nested structure, reflecting the temporal order of the input patterns. The high structural similarity between **(a)** and **(b)** suggests that the output space reflects the fractal-like structure of the synaptic weight space.
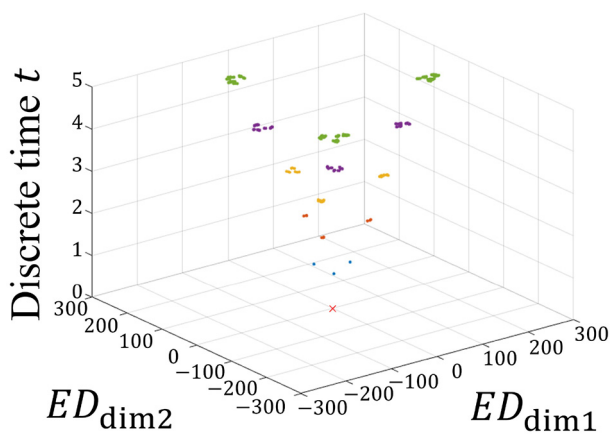

FIGURE 3
Time evolution of $G_w(t)$. ED_{dim1} and ED_{dim2} represent the distances in the dimension-compressed distance space. The vertical axis represents the discrete time t. The red × represents the initial state of $G_w(0) = 0$ because all $W^l(0)$ are the same initial value. At times $t = 1, 2, 3, 4$, and 5, $G_w(t)$ are represented by blue, orange, yellow, purple, and green, respectively. As the network learns the spatiotemporal patterns, these states branch out sequentially at each time step, forming a hierarchical structure. This branching process visually demonstrates how the STLR progressively separates patterns, leading to the formation of a self-similar structure.

$(z^2, \cdots)$, and $(z^3, \cdots)$, respectively. In Figure 4a, ○, ◇, and □ indicate the centers of each cluster, where $(ED_{dim1}, ED_{dim2}) = (-171, -100)$, $(0, 196)$, and $(171, -96)$, respectively. We apply the MDS method to Clusters 1, 2, and 3. The results are shown in

Figures 4b–d. In Figure 4b, ○ represents the center of Cluster 1 in Figure 4a. Similarly, in Figures 4c and 4d, ◇ and □ represent the centers of Clusters 2 and 3 in Figure 4a, respectively. In Figures 4b–d, $g^l(t)$ is labeled according to k_2^l. For example, Cluster $r = 13$ in Figure 4b represents the case when the network learns $X = (z^1, z^3, \cdots)$.

Based on the above results, the spatiotemporal pattern of the input is memorized as a self-similar structure in the synaptic weight space. Additionally, scale-invariance can be confirmed because the triangular structure shown in Figure 4a also appears in Figures 4b–d. As a result, we qualitatively confirm that the STLR forms a fractal structure on the synaptic weight space.

Additionally, we analyzed the structure in the distance space between synaptic weights using the Isomap (Tenenbaum et al., 2000) and UMAP (McInnes et al., 2020) methods. The results are shown in Supplementary Figures S6 and S7, respectively. The Isomap and UMAP methods map high-dimensional information to a low-dimensional space while preserving as much of the original information as possible. These methods utilize k_{NN}-nearest neighbors, and by adjusting the value of k_{NN}, it is possible to investigate structures from local to global. However, even using these methods, it was not possible to reveal in detail the local and global structures shown in Figure 6. This is because of the significant difference in distance scales between local and global clusters.

Figure 5 shows the time evolution and the structure of $G_w(t)$ when a white noise signal of amplitude σ_p is applied to the input (for results when the amplitude is changed to values other than σ_p, see Supplementary Section 7). Figures 5a, c show the time evolution and the resulting structure at $t = 5$, respectively, when $\theta_{LTP} = 4$. Similarly, Figures 5b, d show the results when $\theta_{LTP} = 2$. When
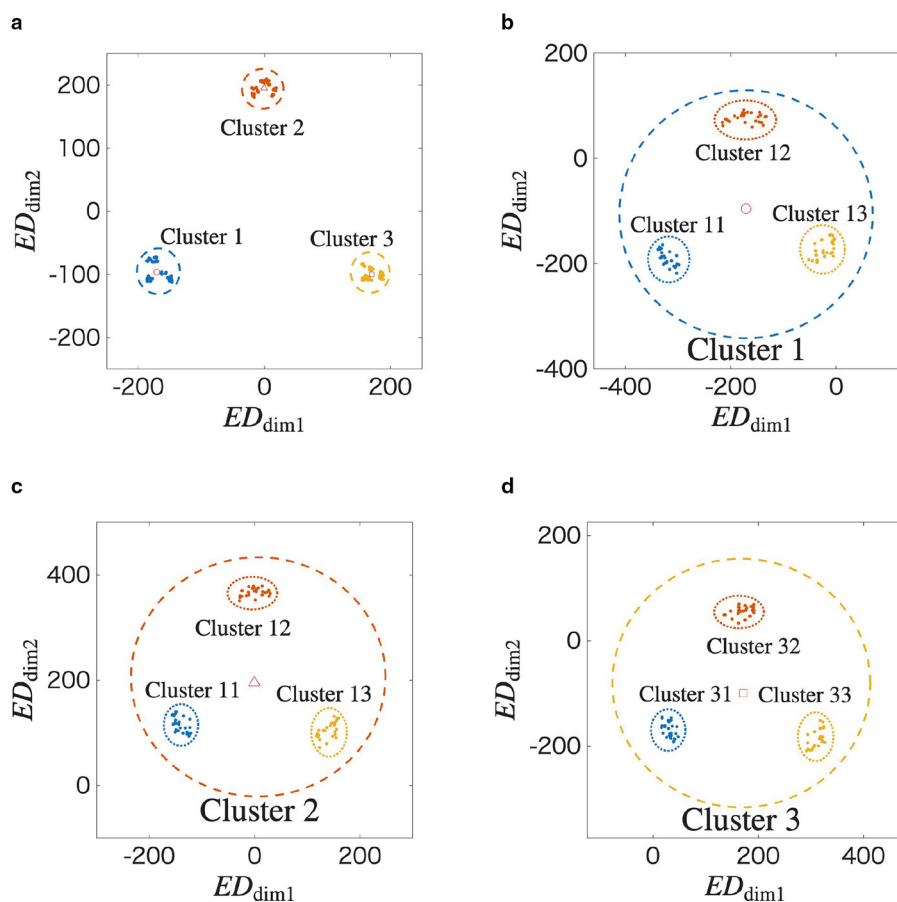
FIGURE 4
Recursive analysis of the final learned structure at $t = 5$, revealing its self-similar and scale-invariant properties. **(a)** A two-dimensional view of the final state from Figure 3. The data points are grouped into three clusters based on the first spatial vector in their respective input sequences (Cluster 1, 2, and 3). The center of each cluster is marked. **(b−d)** are applied the MDS to the Cluster 1, 2, and 3 in Figure 4a, respectively. Figures 4b−d reveal that the internal structure of a cluster is itself composed of three smaller sub-clusters, formed in a similar triangle. These sub-clusters are formed based on the second spatial vector in the input sequences. The repetition of the same geometric pattern at a smaller scale is a clear demonstration of the fractal properties of self-similarity and scale-invariance.

noise is high, decreasing θ_{LTP} enhances the network's ability to handle noisy inputs, improving the pattern separation accuracy of the clusters, which mitigates the cluster overlap seen in Figure 5c, as shown in Figure 5d. A decrease in θ_{LTP} caused an increase in the number of updated synapses. As a result, a noisy spatial vector is complemented. Therefore, it can be suggested that the ability to separate spatiotemporal patterns with noise is improved. The relationship between θ_{LTP} and the number of updated synapses is detailed in Supplementary Section 2.

3.3 Estimation of fractal dimension

To estimate the fractal dimension in Figure 4, $\mathbf{G}_{\text{w}}(t)$ is adjusted using ρ_n. The centers of each Cluster r are adjusted using Equation 33. For example, in Figure 4a, $CC_r^{\text{adj}} = 2.2CC_r$, where $n = 2$; $r = 1, 2, 3$; $\rho_n = 0.46$; and $\alpha_n = 2.2$. That is, CC_1^{adj}, CC_2^{adj}, and CC_3^{adj} are $(-374, -220)$, $(0, 430)$, and $(374, -210)$, respectively. $\mathbf{G}_{\text{w}}(t)$ is adjusted to $\mathbf{G}_{\text{w}}^{\text{adj}}(t)$ using α_n as shown in Table 2.

Figure 6 shows $\mathbf{G}_{\text{w}}^{\text{adj}}(t)$ at $t = 5$. In Figure 6, a fractal structure similar to a Sierpinski gasket is confirmed, which could not be observed in Figure 4. The fractal dimension is determined to be 1.415 by applying the box-counting method to $\mathbf{G}_{\text{w}}^{\text{adj}}(t)$ shown in Figure 6 (Supplementary Figure S14). Similarly, the average of the coefficient of variation and the averag of local fractal dimension are 0.6048 and 1.321 by applying the lacnarity and mass-radius methods, respectively (Supplementary Figures S15 and S17). These results quantitatively confirm that a fractal structure is formed in the synaptic weight space through the STLR.

3.4 Reproduction of fractal structure using IFS

We utilize the IFS to reproduce the fractal structure in the distance space between the synaptic weights. The initial values and parameter values for the IFS in Equations 35 are as follows: $\mathbf{g}(0) = [0 \ 0]$, $\gamma_k(0) = 1200$, $\gamma_k(t + 1) = 0.424\gamma_k(t)$, $\phi_1(t) = 0$, $\phi_2(t) = 2\pi/3$, and $\phi_3(t) = -2\pi/3$, respectively. These parameter
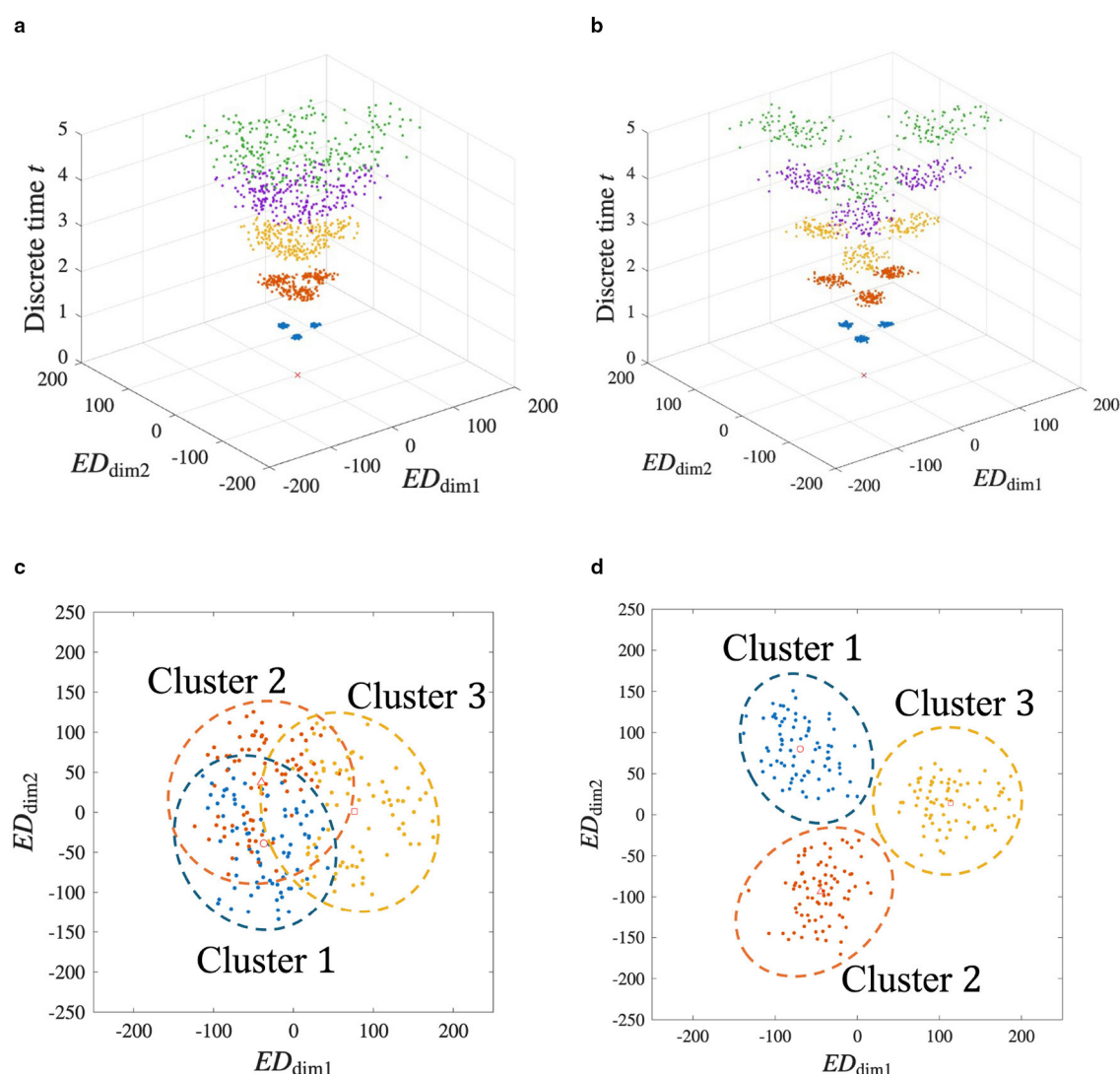
FIGURE 5
MDS analysis of the synaptic weight space under noisy input conditions. This figure demonstrates the network's robustness to noise and the role of the LTP threshold. White noise with an amplitude of σ_p is added to the inputs during learning. **(a)** and **(c)** Results for $\theta_{LTP} = 4$. Figure 5a and 5c show the time evolution of the structure and the final state at $t = 5$, respectively. Figure 5c indicates the significant overlap between clusters. **(b)** and **(d)** Results for $\theta_{LTP} = 2$. The time evolution **(b)** and final state **(d)** show that lowering the threshold results in better-defined, more clearly separated clusters. This suggests that a lower θ_{LTP} enhances the network's ability to handle noisy inputs, allowing it to improve the overall separation accuracy.

values are obtained from the branching behavior observed in the network simulation as shown in Figure 3 (for details, see Supplementary Section 10). Functions F_1, F_2, and F_3 are selected with a 1/3 probability using uniform random numbers. Equation 35 is repeated for $0 \leq t \leq 5$. Figure 7 shows the results of $\mathbf{g}(t)$ at $t = 5$ after repeating the above procedure 2,000 iterations. The resulting structure's striking resemblance to the fractal structure shown in Figure 6 provides strong evidence that the learning process of the STLR can form a fractal structure. This provides a mathematical basis for the fractal coding being performed through the STLR.

In conclusion, a single-layer feedforward neural network with the STLR synapses memorizes the input spatiotemporal patterns as the fractal structure in the synaptic weight space.

4 Discussion

4.1 Fractal structure in the STLR

In Section 3.3, we analyzed the fractal structure in the distance space between the synaptic weights using the box-counting, lacunarity, and mass-radius methods (Supplementary Figures S14–S17) by the MDS method, as shown in Figure 4. The coefficient of variation and the distribution of the local fractal dimension shown in Supplementary Figures S15 and S17, respectively, suggest that the fractal structure is inhomogeneous.

However, the rigorous multifractal formalism established by Halsey et al. (1986) analyzes the scaling properties of a measure defined on that set, rather than the geometric distribution of the

TABLE 2 Correction coefficients used for adjusting cluster coordinates in the MDS visualization.

Cluster label r	α_n
1, 2, 3	2.2
11, 12, $\cdots$, 33	1.6
111, 112, $\cdots$, 333	1.4
1111, 1112, $\cdots$, 3333	1.3
11111, 11112, $\cdots$, 33333	1

The correction coefficients applied to adjust the cluster center coordinates for the visualization presented in Figure 6. These coefficients are calculated as the inverse of the cumulative contribution ratio of the eigenvalues obtained from the MDS analysis at each cluster. The "Cluster label r" denotes the hierarchical depth, with "1, 2, 3" representing the top-level clusters and "11, 12, ..., 33" representing the sub-clusters at the next level.

set itself. Our analysis focused only on geometric information, and a strict multifractal analysis has not yet been performed. This points to a critical issue for future research.

By deriving the generalized dimensions and the singularity spectrum within the rigorous multifractal formalism, it is possible to characterize the STLR's learning properties. The value of the generalized dimension allows for the analysis of different parts of a measure. A crucial future task is to quantitatively discuss how the memory structure formed by the STLR balances these conflicting functions of pattern completion and separation, as shown in Figure 5, by analyzing the shape of its multifractal spectrum.

4.2 Fractal coding through the STLR

Fractal coding encodes information into hierarchical self-similar structures as shown in Figure 4. The spatiotemporal patterns are encoded into a static synaptic weight space through the STLR and read out in the output space by inputting a specific signal in a single shot. Therefore, fractal coding through the STLR enables highly efficient compression and processing of information. On the other hand, for the Cantor coding proposed in the literature (Tsuda, 2001; Yamaguti et al., 2011), the spatiotemporal patterns are dynamically encoded into the state space of a network and read out as a time-series by inputting a specific signal. However, fractal coding using the STLR also dynamically encodes and reads out information by introducing recurrent connections into the network (Tsukada and Tsukada, 2021b).

.

The pattern separation ability of the STLR significantly contributes to the fractal coding. The STLR can learn slight differences in the similar spatiotemporal patterns. Therefore, clusters and hierarchical structures were formed based on the spatial and temporal differences in the spatiotemporal patterns using the STLR, respectively. Consequently, the spatiotemporal patterns were memorized as a fractal structure.

For the pattern separation ability of the STLR, we suggest that the learning rate η, the firing rate of the input spatial vectors (the ratio of 0s to 1s), the time constant τ_Q, LTP threshold θ_{LTP}, and LTD threshold θ_{LTD} are important parameters from preliminary experiments. In the present study, these parameter
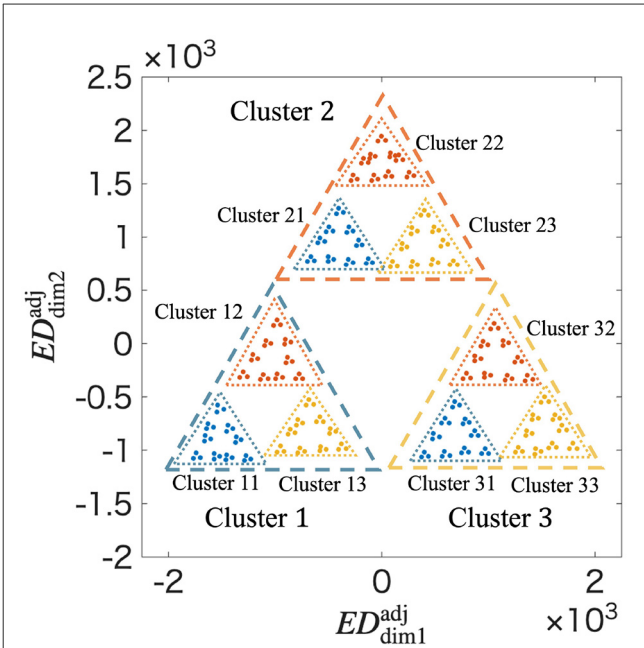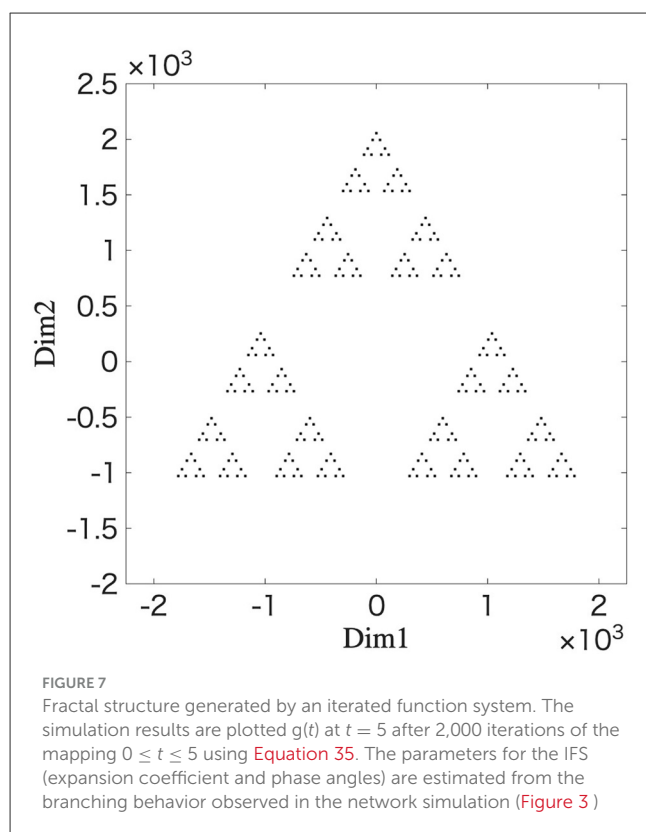

FIGURE 6
Fractal structure in $G_w^{adj}(t)$ at $t = 5$ using ρ_n obtained from the MDS. Clusters are classified according to the temporal order of spatial vectors up to the second in the input spatiotemporal patterns. ED_{dim1}^{adj} and ED_{dim2}^{adj} represent the distances in the adjusted distance space. The adjustment geometrically arranges the clusters according to their true distances in the high-dimensional space. The resulting visualization reveals a clear and detailed fractal structure, closely resembling a Sierpinski gasket. This adjusted representation allows for a more accurate estimation of the fractal dimension and confirms the hierarchical self-similarity of the memory structure.

values were obtained from (Tsuji et al., 2023a), where the order-nested nested structure in the output is clearly observable through an exhaustive search. However, these parameter values should be determined in accordance with physiological findings (Tsukada et al., 1996) to reproduce physiological experiments. In particular, θ_{LTP} and θ_{LTD} are important for learning the spatial structure of the spatiotemporal patterns.

From a physiological perspective, the balance between these thresholds is hypothesized to be related to the concentration of neuromodulators such as acetylcholine. The relationship between the thresholds controls the trade-off between pattern completion and pattern separation. Under the condition $\theta_{LTD} \ll \theta_{LTP} \approx \mu_q$, the learning of common features within an input sequence becomes dominant, strengthening the characteristics of pattern completion and contributing to the convergence to an attractor through fractal structuring. Conversely, under the condition $\theta_{LTD} \approx \mu_q \ll \theta_{LTP}$, the learning of common elements is canceled out, and the system becomes dominated by the learning of orthogonal elements, thereby strengthening pattern separation. This suggests that the hippocampus forms memories based on an appropriate balance between these two abilities, a balance that may be modulated by acetylcholine. Furthermore, this suggestion aligns with the results of the noise-added experiments in Section 3.2. This perspective is important as it has implications for physiological experimental research.

FIGURE 7
Fractal structure generated by an iterated function system. The simulation results are plotted g(t) at $t = 5$ after 2,000 iterations of the mapping $0 \leq t \leq 5$ using Equation 35. The parameters for the IFS (expansion coefficient and phase angles) are estimated from the branching behavior observed in the network simulation (Figure 3)

Based on the conditions derived in Supplementary Section 2, if θ_{LTP} is too large and θ_{LTD} is too small, the synaptic weight values are hardly updated (i.e., the network does not learn the spatiotemporal patterns) and clusters are not formed. Additionally, τ_Q significantly affects the learning of the temporal structure of the spatiotemporal patterns. If τ_Q is too small, learning the temporal structure is difficult owing to the influence of the initial values of the synaptic weights. Therefore, it is necessary to determine these parameter values based on the results of physiological experiments. Numerical simulations using these parameter values enabled analyses that are difficult to perform in physiological experiments. Furthermore, a theoretical approach is required to elucidate the learning mechanisms of the STLR in detail.

4.3 Interrelationship between the structures in the synaptic weight and output spaces

The relationship between the fractal structure in the synaptic weight space and the order-nested structure in the output space is crucial, as the output space provides a physiological link to an organism's behavior. We propose that the sensory memory structure of a recognition system is encoded as a fractal structure in the synaptic weight space, and this structure can be expressed in the output space as needed to inform behavior.

In a previous study, we identified the order-nested structures formed in the distance space of the network's outputs with the STLR synapses (Tsuji et al., 2023a). Figure 2 confirms that the

distance map created from synaptic weight values has a fractal structure. To clarify the interrelationship between these structures, we performed numerical simulations by changing the initial values of the synaptic weights 1,000 times when $T = 3$ (see Supplementary Figure S2). Additionally, even when the network parameters are changed, the similarity between the synaptic weights and outputs in the distance space is high (for details, see Supplementary Section 3).

Histograms and distance maps of the synaptic weights, internal states, and outputs were created, respectively, for each initial value (Supplementary Figure S4). Consequently, a fractal structure was observed in the synaptic weight space for all the initial conditions. By contrast, the order-nested structures were not always observed in the internal state and output spaces. In other words, the fractal structure in the weight space does not always appear in the output space. There are two possible reasons for this. The first is the dimensional compression from the input to the internal state. The inputs are multiplied by the weights and spatially summed and averaged in the internal state. Consequently, the information embedded in the synaptic weight space is compressed from $\mathbb{R}^{M \times N}$ to $\mathbb{R}^N$. The second is the binarization of the internal state at the output through thresholding. This further compresses information in the internal state (i.e., $\mathbb{R}^N \to \{0, 1\}^N$). If the firing threshold is not properly adjusted, synaptic weight information is further reduced. Conversely, our simulations showed that when an order-nested structure appears in the output space, a fractal structure is always found in the synaptic weight space. This is because synaptic weights always form a fractal structure (Supplementary Section 9) when the parameter values satisfy the conditions derived in Supplementary Section 2. Consequently, we concluded that the order-nested structure in the output space guarantees a fractal structure in the synaptic weight space. However, the fundamental reason why the order-nested structure does not emerge in the output space remains unexplained. We suggest that the initial values of the synaptic weights have a significant influence on this issue. As mentioned in Supplementary Section 10, this network exhibits chaotic behavior, and this is due to its characteristic sensitivity to initial conditions. Therefore, future research will analyze the chaotic behavior of synaptic weights and the conditions under which the order-nested structure appears in the output.

4.4 Hardware implementation

Recently, neural-network-based artificial intelligence (AI) has been applied to a variety of fields (Maslej et al., 2024). However, several problems have arisen in this regard. For example, the large-language models consume a large amount of power and data for learning, causing serious social problems (Team, 2025). This indicates that current AI models are not suitable for applications where energy, area, and training data are restricted, such as edge-devices.

In contrast, a neural network based on the STLR can learn contexts embedded in the spatiotemporal sequences with a small number of examples. Furthermore, the STLR uses only

local information around each synapse; thus, a small area is sufficient for hardware implementation. Additionally, because the STLR is inherently asynchronous (Orima et al., 2023), it is suitable for spiking neural networks (Tsuji et al., 2024), which drastically reduces power consumption. From a hardware perspective, fractal coding achieves hierarchical spatiotemporal information compression, resulting in highly efficient memory usage. Therefore, the STLR is a good candidate for the integrated circuit implementation of small-sized, efficient, and high-performance spiking neural networks with one-shot learning capability (Orima et al., 2023). Furthermore, we investigated circuit precision requirements for hardware implementation and found that the original STLR model can be reproduced even with approximately 8-bit circuit implementation (Tsuji et al., 2023b). Based on this result, we are developing a dedicated hardware emulator for designing analog integrated circuits (Tsuji et al., 2024). Using this emulator, we are investigating parameter values for the implementation of analog integrated circuits and verifying the robustness of the system, considering device variations.

However, it is hard to measure the synaptic weight values from an integrated circuit because the state of each synaptic circuit/device is usually inaccessible, and if possible, the number of synapses is unpractically large. Therefore, evaluating the learning characteristics based on the synaptic weight space is difficult. That is, the fractal structure in the synaptic weight space resulting from the STLR, which was evaluated using simulations, could not be experimentally observed from the integrated circuit measurements. However, as discussed in Section 4.3, an order-nested structure in the output space provides a good measure of the fractal structure in the synaptic weight space. Consequently, the outputs of the network, which are easily measurable, can be used to evaluate fractal coding through the STLR in integrated circuit systems.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

TO: Writing – original draft, Writing – review & editing. IT: Writing – review & editing. MT: Writing – review & editing. HT: Writing – review & editing. YH: Writing – original draft, Writing – review & editing.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Gen AI was used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fncom. 2025.1641519/full#supplementary-material

References

Bieberich, E. (2002). Recurrent fractal neural networks: a strategy for the exchange of local and global information processing in the brain. *BioSystems* 66, 145–164. doi: 10.1016/S0303-2647(02)00040-0

Fukushima, Y., Tsukada, M., Tsuda, I., Yamaguti, Y., and Kuroda, S. (2007). Spatial clustering property and its self-similarity in membrane potentials of hippocampal CA1 pyramidal neurons for a spatio-temporal input sequence. *Cogn. Neurodyn.* 1, 305–316. doi: 10.1007/s11571-007-9026-9

Guardamagna, M., Stella, F., and Battaglia, F. P. (2023). Heterogeneity of network and coding states in mouse CA1 place cells. *Cell Rep.* 42:112022. doi: 10.1016/j.celrep.2023.112022

Halsey, T. C., Jensen, M. H., Kadanoff, L. P., Procaccia, I., and Shraiman, B. I. (1986). Fractal measures and their singularities: the characterization of strange sets. *Phys. Rev. A* 33, 1141–1151. doi: 10.1103/PhysRevA.33.1141

Hebb, D. O. (1949). *The Organization of Behavior*. New York: Wiley.

Husain, A., Nanda, M. N., Chowdary, M. S., and Sajid, M. (2022). Fractals: an eclectic survey, part-I. *Fractal Fract.* 6:89. doi: 10.3390/fractalfract6020089

Kovcs, K. A. (2020). Episodic memories: how do the hippocampus and the entorhinal ring attractors cooperate to create them? *Front. Syst. Neurosci.* 14:559186. doi: 10.3389/fnsys.2020.559186

Kuroda, S., Fukushima, Y., Yamaguti, Y., Tsukada, M., and Tsuda, I. (2009). Iterated function systems in the hippocampal CA1. *Cogn. Neurodyn.* 3, 205–222. doi: 10.1007/s11571-009-9086-0

Landini, G., and Rippin, J. W. (1993). Notes on the implementation of the mass – radius method of fractal dimension estimation. *Bioinformatics* 9, 547–550. doi: 10.1093/bioinformatics/9.5.547

Mandelbrot, B. B. (1982). *The Fractal Geometry of Nature*. San Francisco, CA: Freeman.

Mandelbrot, B. B. (1994). *A Fractal's Lacunarity, and How It Can Be Tuned and Measured*. Basel: Birkhäuser Basel.

Maslej, N., Fattorini, L., Perrault, R., Parli, V., Reuel, A., Brynjolfsson, E., et al. (2024). *Artificial Intelligence Index Report 2024*. Available online at: https://arxiv.org/abs/2405.19522

McCloskey, M., and Cohen, N. J. (1989). Catastrophic interference in connectionist networks: the sequential learning problem. *Psychol. Learn. Motiv.* 24, 109–165. doi: 10.1016/S0079-7421(08)60536-8

McInnes, L., Healy, J., and Melville, J. (2020). *UMAP: Uniform Manifold Approximation – and Projection for Dimension Reduction*. Available online at: https://arxiv.org/abs/1802.03426

Melchior, J., Altamimi, A., Bayati, M., Cheng, S., and Wiskott, L. (2024). Correction: a neural network model for online one-shot storage of pattern sequences. *PLoS ONE* 19:e0313130. doi: 10.1371/journal.pone.0313130

Orima, T., Tsuji, T., and Horio, Y. (2023). An extended spatiotemporal contextual learning and memory network model for hardware implementation. *Procedia Comp. Sci.* 222, 478–487. doi: 10.1016/j.procs.2023.08.186

Rolls, E. T. (2010). A computational theory of episodic memory formation in the hippocampus. *Behav. Brain Res.* 215, 180–196. doi: 10.1016/j.bbr.2010.03.027

Sarkar, N., and Chaudhuri, B. (1994). An efficient differential box-counting approach to compute fractal dimension of image. *IEEE Trans. Syst. Man Cybern.* 24, 115–120. doi: 10.1109/21.259692

Smith, T., Lange, G., and Marks, W. (1996). Fractal methods and results in cellular morphology – dimensions, lacunarity and multifractals. *J. Neurosci. Methods* 69, 123–136. doi: 10.1016/S0165-0270(96)00080-5

Team ICS (2025). *AI and Resource Consumption: Examining the Environmental Impact*. Available online at: https://www.computer.org/publications/tech-news/research/ai-and-the-environment

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 2319–2323. doi: 10.1126/science.290.5500.2319

Torgerson, W. S. (1952). Multidimensional scaling: I. theory and method. *Psychometrika* 17, 401–419. doi: 10.1007/BF02288916

Tsuda, I. (2001). Toward an interpretation of dynamic neural activity in terms of chaotic dynamical systems. *Behav. Brain Sci.* 24, 793–848. doi: 10.1017/S0140525X01000097

Tsuji, T., Orima, T., and Horio, Y. (2023a). "Detailed evaluation of spatiotemporal learning rule based on hamming distances among output vectors," in *The 2023 International Symposium on Nonlinear Theory and Its Applications (NOLTA)* [The Institute of Electronics, Information and Communication Engineers (IEICE)], 415–418.

Tsuji, T., Orima, T., and Horio, Y. (2023b). "A study on circuit accuracy for hardware implementations of spatio-temporal learning rule," in *Proc. IEICE Gen. Conf. (in Japanese)* [The Institute of Electronics, Information and Communication Engineers (IEICE)].

Tsuji, T., Orima, T., and Horio, Y. (2024). "An event-driven mixed analog/digital spiking neural network circuit model for hippocampal spatiotemporal context learning and memory," in *2024 International Joint Conference on Neural Networks (IJCNN)* [The Institute of Electronics, Information and Communication Engineers (IEICE)], 1–8.

Tsukada, H., and Tsukada, M. (2021a). 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

Tsukada, H., and Tsukada, M. (2021b). "Context-dependent learning and memory based on spatio-temporal learning rule," in *Advances in Cognitive Neurodynamics*, eds. A. Lintas, P. Enrico, X. Pan, R. Wang, and A. Villa (Singapore: Springer Singapore), 89–94.

Tsukada, M., Aihara, T., Saito, H., 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

Werner, G. (2010). Fractals in the nervous system: Conceptual implications for theoretical neuroscience. *Front. Physiol.* 1:15. doi: 10.3389/fphys.2010.00015

Yamaguti, Y., Kuroda, S., Fukushima, Y., Tsukada, M., and Tsuda, I. (2011). A mathematical model for cantor coding in the hippocampus. *Neural Netw.* 24, 43–53. doi: 10.1016/j.neunet.2010.08.006