LETTER Communicated by Ben Tsud

Flexible Transmitter Network

Shao-Qun Zhang
zhangsq@lamda.nju.edu.cn
Zhi-Hua Zhou
zhouzh@lamda.nju.edu.cn
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China

Current neural networks are mostly built on the MP model, which usually formulates the neuron as executing an activation function on the real-valued weighted aggregation of signals received from other neurons. This letter proposes the flexible transmitter (FT) model, a novel bioplausible neuron model with flexible synaptic plasticity. The FT model employs a pair of parameters to model the neurotransmitters between neurons and puts up a neuron-exclusive variable to record the regulated neurotrophin density. Thus, the FT model can be formulated as a two-variable, two-valued function, taking the commonly used MP neuron model as its particular case. This modeling manner makes the FT model biologically more realistic and capable of handling complicated data, even spatiotemporal data. To exhibit its power and potential, we present the flexible transmitter network (FTNet), which is built on the most common fully connected feedforward architecture taking the FT model as the basic building block. FTNet allows gradient calculation and can be implemented by an improved backpropagation algorithm in the complex-valued domain. Experiments on a broad range of tasks show that FTNet has power and potential in processing spatiotemporal data. This study provides an alternative basic building block in neural networks and exhibits the feasibility of developing artificial neural networks with neuronal plasticity.

1 Introduction

The fundamental computational unit of neural networks is the neuron, corresponding to the cell in biological nervous systems. Though neural networks have been studied for more than half a century and various neural network algorithms and network architectures have been developed, the modeling of neurons is relatively less considered.

The most famous and commonly used formulation of neuron is the MP model (McCulloch & Pitts, 1943), as illustrated in Figure 1. This model formulates the neuron as executing an activation function on the weighted

https://doi.org/10.1162/neco_a_01431

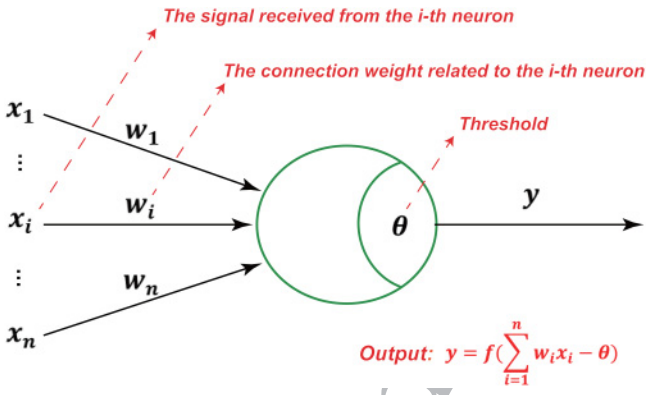

Figure 1: The MP model.

aggregation of signals received from other neurons compared with a threshold, that is,

$$y = f\left(\sum_{i=1}^{n} w_i x_i - \theta\right).$$

In this figure, x_i's are the input signals, w_i's are the connection weights, θ is a neurological threshold, and f denotes the activation function, such as the sigmoid function often used in shallow neural networks and the ReLU function traditionally used in deep ones.

The MP model is very successful, though the formulated cell behavior is quite simple. Actual nervous cells are much more complicated, so exploring other bio-plausible formulations with neuronal plasticity is a fundamental and significant problem. There have been many efforts on modeling the spiking behavior of cells, leading to spiking neuron models (Gerstner & Kistler, 2002; Moon, Wu, Zhu, & Lu, 2021) and pulsed neural networks (Wang, Ma, Cheng, & Yang, 2010). In this work, we consider another interesting aspect and propose a novel type of neuron model.

1.1 Synaptic Plasticity. Neuroscience studies (Lodish et al., 2008; Debanne, Campanac, Bialowas, Carlier, & Alcaraz, 2011) disclose that the communication between neurons relies on the synapse. Signal flows in one direction, from the presynaptic neuron to the postsynaptic neuron via the synapse. The synapse usually forms between the endings (or terminals) of the axon and dendrite, which link to the presynaptic and postsynaptic neurons, respectively. The endings of the axon and dendrite are named *presynapse* and *postsynapse*, respectively. In common synaptic structures, there is a gap (the *synaptic cleft*) of about 20 μm between the presynapse and

postsynapse. The synapse is a combination of the presynapse, synaptic cleft, and postsynapse, as shown in the left half of Figure 2b.

Synapse ensures the one-way communication behavior between two neurons. In detail, when an external signal through the axon arrives at the presynapse, it will be collected by the synaptic vesicles and converted into a chemical substance called a neurotransmitter (Mattson, 1988). With the chemical movement, synaptic vesicles fuse with the presynaptic membrane and open channel-like protein molecules, releasing neurotransmitters into the synaptic cleft. Neurotransmitters diffuse across the synaptic cleft and then bind to the transmitter receptors on the postsynapse. The binding chemical action alters the shape and concentration of the transmitter receptors, leading to the opening or closing of ion channels in the cell membrane. Thus, some ions, such as Ca^{2+}, can pass through the postsynaptic membrane. Some researchers (Schinder & Poo, 2000; Zhong et al., 2009; Park & Poo, 2013) point out that thanks to these binding chemical actions, the target tissue on postsynapse will secrete a class of activated proteins, called neurotrophins. The neurotrophin works in the presence of some given stimulus and creates local effects in the postsynapse, especially the dendritic spine. And the neurotrophin density has been linked to growth in dendritic spine volume and synaptic plasticity, such as the addition of AMPA receptors to the postsynaptic membrane and phosphorylation of ion channels for enhanced permeability. When the transmitter receptors receive inhibition signals, the neurotrophin density reduces and the postsynapse shrinks, and then the shrinking postsynapse inhibits subsequent signal acceptance. If the transmitter receptors receive excitation signals, the neurotrophin density increases and the postsynapse swells up, contributing to subsequent signal acceptance (Bi & Poo, 1998; Zhou, Homma, & Poo, 2004). In summary, regulated by the neurotrophins, the dendritic spine volume will grow or shrink. Simultaneously, the generated stimuli will be persistent, strengthening or weakening based on recent patterns of neurotrophins, a procedure referred to as long-term potentiation or depression (Cooke & Bliss, 2006; Gong et al., 2011). Finally, the generated stimuli are transmitted to the postsynaptic neuron via the dendrite. The one-way communication procedure is shown in Figure 2b.

Based on the neurological knowledge about synaptic plasticity, we review the modeling methods of the classical MP and spiking neuron models. The MP model takes the whole synapse as a connection parameter (the w_i in Figure 1). The signals through this synapse are weighted by w_i. Thus, the MP model is formulated as a real-valued function. The spiking neuron model establishes upon the postsynaptic potential (PSP) assumption that the postsynapse would integrate the membrane potential modified by the neurotransmitters and be activated only if the integrated potential exceeds a threshold. Considering the PSP as the concerned variable, the potential-integrated process is usually simulated by some first-order differential equations (VanRullen, Guyonneau, & Thorpe, 2005). The standard

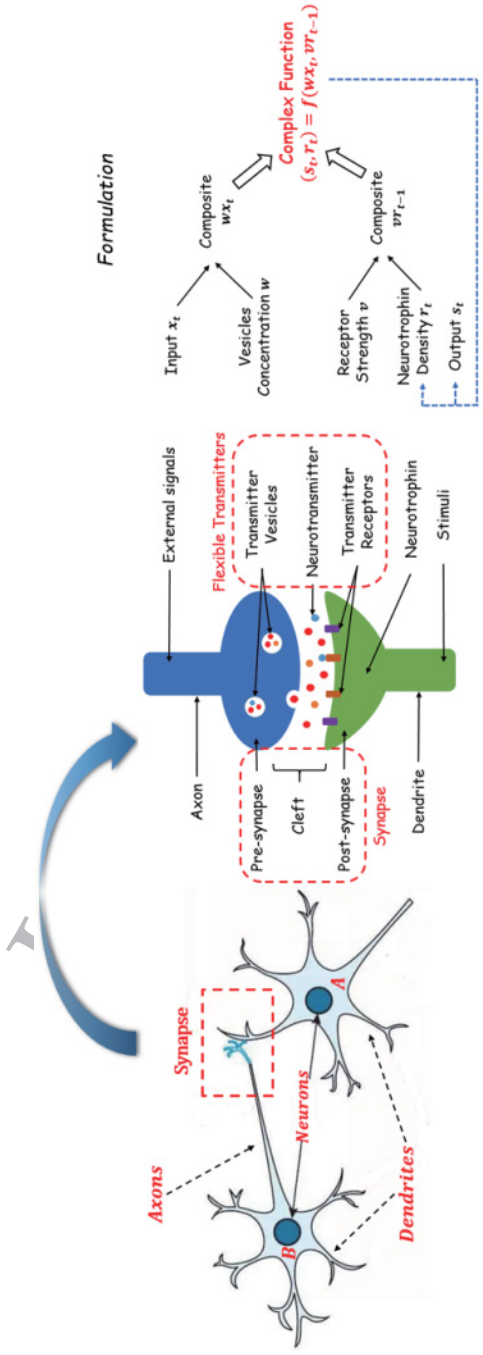

Figure 2: Illustration of (a) biological neurons, (b) synapse with one-way communication, and (c) formulation of the FT model.

and popular spiking neuron models contain the integrated-and-fire model (Stein, 1965) and the Hodgkin-and-Huxley model (Hodgkin & Huxley, 1952).

1.2 Flexible Transmitters. In the one-way neurotransmitter communication procedure, the signal transmitted from a neuron to another has undergone several roles: the external signal until being collected by the synaptic vesicles, the neurotransmitters that diffuse across the whole synapse, and the generated stimuli. And three transformations are carried out. The first is converting the external signal to the neurotransmitters. Synaptic vesicles compound with the external signals, and the vesicle concentration affects the number of neurotransmitters. Second is releasing Ca^{2+} ions through the postsynaptic membrane. The receptor strength controls the number of ions passing through the membrane. The neurotrophin density alters the biological activity of the post-synaptic membrane. Third, a complex chemical action secretes the neurotrophins and generates the stimuli. Notice that synaptic vesicles, receptors, and neurotrophins are three vital elements of the whole communication process. Both the synaptic vesicles and receptors are weighted transmitters, which regulate the magnitude of signals. The neurotrophin density is a circuit variable. On the one hand, the neurotrophin is generated as the neuron model's output; on the other hand, its density affects the passing of the subsequent ions, working as an input.

Here, we call the combination of the synaptic vesicle concentration and receptor strength the *flexible transmitter* (FT), as shown in Figure 2c. It employs a pair of learnable parameters (w, v) to represent them. In addition to the conventional variables, input signal x and output stimulus s, we put up a variable r_t to denote the neurotrophin density at the t time. r_t is not only the output of our model at time t but also the input at the next time, $t + 1$. The compound of w and x records the number of neurotransmitters released in the synaptic cleft by the presynapse. The compound of v and r_t indicates the information that can affect the postsynapse at time t. Finally, we utilize an apposite function f to represent the complex chemical action on the postsynapse. Thereby, a novel neuron model establishes; it consists of two inputs (the external signal x and the neurotrophin density r_{t-1} at the last time), two outputs (the generated stimuli s and the neurotrophin density r_t at the current time), and a pair of learnable parameters (w, v). The FT model has the formation of a two-variable, two-valued function. It's entirely different from the conventional neuron models. In section 2, we introduce the FT model in detail.

Regarding the FT model as the basic building block, various network architectures can be tried; the simplest may be the fully connected feedforward architecture popularly applied with the MP model. In section 3, we present the *flexible transmitter network* (FTNet), a fully connected network constructed by replacing the real-valued MP model with the FT model.

Correspondingly, a practicable and effective backpropagation algorithm for training FTNet is developed. Experiments are conducted on a broad range of spatiotemporal data sets in section 4. The results show that FTNet can get excellent performance with the same setting as the contenders. Section 5 presents our discussion and section 6 our conclusion.

2 Flexible Transmitter Model

The interesting discovery of neuroscience in Figure 2b suggests that the response of neuron A to the external signal from neuron B depends on not only a pair of flexible transmitters in the synapse but also the neurotrophin density of the postsynapse.

Inspired by this recognition, we propose the FT model, as illustrated in Figure 2c. In contrast to the MP model, where the interaction between two neurons is formulated by a single connection weight, in the FT model, the interaction comprises two parts: wx_t, where x_t is the external signal sent to the concerned neuron via the corresponding vesicle concentration w, and vr_{t-1}, where r_{t-1} is the neurotrophin density at the $(t-1)$th time stamp related to the receptor strength v. In brief, the FT model employs a pair of transmitter parameters (w, v) rather than a real-valued weight w_i in the MP model. The output of the FT neuron at the tth time stamp consists of two parts: s_t and r_t, which indicate the generated bio-electric/chemical stimulus and neurotrophin density, respectively. After time t, the stimulus signal s_t is transmitted to the next neuron, while r_t renews the neurotrophin density and participates in the inputs of time $(t+1)$.

In summary, the proposed FT model employs a pair of parameters (w, v) to indicate the transmitters and puts up an exclusive variable r_t to represent the regulated neurotrophin density. Therefore, the FT model intrinsically has a formation of a two-variable, two-valued function f concerning a pair of parameters (w, v):

$$(s_t, r_t) = f(wx_t, vr_{t-1}). \tag{2.1}$$

We call this model the *flexible transmitter*.

The FT model has many benefits. First, paired parameters precisely clarify the roles of the transmitters and provide greater flexibility for synaptic plasticity. From a formulaic perspective, the MP model is a special case of the FT model when ignoring the transmitter parameter v and the neurotrophin density r_{t-1} or forcing these values to 0. Second, the FT model employs an exclusive variable r_t to indicate the neurotrophin density. During the learning process, the neurotrophin density variable frequently achieves self-renewal, thus deriving a local recurrent system. Therefore, the FT model may have the potential to handle more complicated data, even time series signals.

3 Flexible Transmitter Network

3.1 An Implementation of the FT Model. According to equation 2.1, the FT model is dominated by a two-variable, two-valued function f and a pair of parameters (w, v). Both the input and output of the FT model comprise two parts, and their relationship can be complicated since both outputs, s_t and r_t, share common parameters (w, v) and inputs (x_t, r_{t-1}). Existing neuron models often depend on one-valued (or real-valued) functions, so the related technologies are hard to apply to this concern directly. An interesting solution is to resort to a complex-valued formation that represents the input and output of the concerned neurons, respectively, leading to the FT neuron model:

$$s_t + r_t \boldsymbol{i} = f(wx_t + vr_{t-1}\boldsymbol{i}). \tag{3.1}$$

We call equation 3.1 the *complex-valued reaction* of the FT model. According to complex analysis, the real and imaginary parts of the output of a complex-valued function are geminous twins; both s_t and r_t share the common complex-valued function f and parameters (w, v). Thus, if we mastered the value or formulation of s_t, we could easily derive r_t. Further, once some teacher signals supervise the stimulus s_t, the neurotrophin density r_t can still be corrected even if leaving r_t unsupervised.

Notice that the complex-valued reaction is just one approach for implementing the proposed FT model. It may not be the most appropriate one, and there are likely better approaches to be explored in the future.

3.2 A Simple Architecture of FTNet. The FT neuron is a fundamental unit of neural networks. To evaluate its potential, we consider using the simplest fully connected feedforward neural network by replacing the standard MP model with the FT model as its basic building block; thus, we get the FTNet. Based on equation 3.1, we can provide a general vectorized representation for a layer of FT neurons:

$$\boldsymbol{s}_t + \boldsymbol{r}_t \boldsymbol{i} = f(\mathbf{W}\boldsymbol{x}_t + \mathbf{V}\boldsymbol{r}_{t-1}\boldsymbol{i}). \tag{3.2}$$

Notice that given m-dimensional external input signals $\boldsymbol{x}_t$ and n-dimensional output stimuli $\boldsymbol{s}_t$, the transmitter concentration matrices $\mathbf{W} \in \mathbb{R}^{n\times m}$, $\mathbf{V} \in \mathbb{R}^{n\times n}$ and the neurotrophin density vectors $\boldsymbol{r}_t, \boldsymbol{r}_{t-1} \in \mathbb{R}^n$. Reusing the layer-vectorized representation in equation 3.2 layer by layer, we can obtain a multilayer, fully connected feedforward architecture of FTNet.

There remain two unsolved problems: (1) What does the complex-valued function f look like? and (2) How can it be trained? To address these problems, we unfold the complex-valued function f in equation 3.1 as $f = \sigma \circ \tau$ with a conversion function $\tau : \mathbb{C} \to \mathbb{C}$ and an activation function $\sigma : \mathbb{C} \to \mathbb{C}$.

The conversion function τ formulates the complex aggregation. To perform gradient calculation in FTNet, the conversion function must be differentiable, and thus, we restrict the set of possible conversion functions. Nevertheless, there are still various holomorphic functions that can be tried. Ideally, we can get inspiration from bioscience to design the conversion function; here, we have not incorporated bio-knowledge and use the most straightforward linear holomorphic function,

$$\begin{aligned}\tau(\mathbf{W}\boldsymbol{x}_t + \mathbf{V}\boldsymbol{r}_{t-1}) &= (\mathbf{W}\boldsymbol{x}_t + \mathbf{V}\boldsymbol{r}_{t-1}\boldsymbol{i}) \cdot (a + b\boldsymbol{i}) \\ &= (a\mathbf{W}\boldsymbol{x}_t - b\mathbf{V}\boldsymbol{r}_{t-1}) + (b\mathbf{W}\boldsymbol{x}_t + a\mathbf{V}\boldsymbol{r}_{t-1})\boldsymbol{i}, \end{aligned} \tag{3.3}$$

where a and b are constants in $\mathbb{R}$. Then equation 3.2 becomes

$$\boldsymbol{s}_t + \boldsymbol{r}_t\boldsymbol{i} = \sigma\left((a\mathbf{W}\boldsymbol{x}_t - b\mathbf{V}\boldsymbol{r}_{t-1}) + (b\mathbf{W}\boldsymbol{x}_t + a\mathbf{V}\boldsymbol{r}_{t-1})\boldsymbol{i}\right). \tag{3.4}$$

Next, we introduce some activation functions that could be used in FTNet. An intuitive idea is to decompose the activation function σ into two real-valued nonlinear functions, $\sigma = \sigma_{real} + \sigma_{imag}\boldsymbol{i}$, where σ_{real} and σ_{imag} are real-valued activation functions, such as the sigmoid and tanh functions. FTNet also allows complex-valued activations, such as the modReLU (Arjovsky, Shah, & Bengio, 2016) and zReLU (Trabelsi et al., 2017).

Finally, a complete FTNet is established by employing the holomorphic conversion and complex-valued activation functions. For an L-layer FTNet, its feedforward procedure runs as follows:

$$\begin{cases} \boldsymbol{s}_t^0 = \boldsymbol{x}_t, \\ \boldsymbol{s}_t^l + \boldsymbol{r}_t^l\boldsymbol{i} = \sigma(\boldsymbol{\alpha}_t^l + \boldsymbol{\beta}\boldsymbol{i}_t^l), \\ \boldsymbol{y}_t = \boldsymbol{s}_t^L, \end{cases} \quad \text{with} \quad \begin{cases} \boldsymbol{\alpha}_t^l = a\mathbf{W}^l\boldsymbol{s}_t^{l-1} - b\mathbf{V}^l\boldsymbol{r}_{t-1}, \\ \boldsymbol{\beta}_t^l = b\mathbf{W}^l\boldsymbol{s}_t^{l-1} + a\mathbf{V}^l\boldsymbol{r}_{t-1}. \end{cases} \tag{3.5}$$

Throughout this letter, we use the notations FT0 to denote a one-layer FTNet, that is, without any hidden layer, and FT1 to indicate the FTNet with only one hidden layer. The cascade structures of FT0 and FT1 are abbreviated as size$(m, 0, n)$ and size(m, l, n), respectively, where l is the number of hidden neurons.

3.3 Complex Backpropagation. We present the complex backpropagation (CBP) algorithm for training the FTNet. CBP is an extension of the standard backpropagation algorithm in the complex-valued domain. The core idea of CBP is to take the neurotrophin density as an implicit variable so that the desired gradients become a partial derivative function of r_t concerning the connection parameters $\mathbf{W}$ and $\mathbf{V}$. Here, we list the main steps and results of our proposed CBP algorithm. Let $\mathbf{E}(\mathcal{W}, \mathcal{V})$ denote the loss function for FTNet in time interval $[0, T]$,

$$\mathbf{E}(\mathcal{W}, \mathcal{V}) = \frac{1}{2}\int_{t=0}^{T} E_t \, dt = \frac{1}{2}\int_{t=0}^{T} \sum_{i=1}^{n_L} (\hat{\boldsymbol{y}}_t(i) - \boldsymbol{y}_t(i))^2 \, dt,$$

where $\hat{\boldsymbol{y}}_t$ is the supervised signal. The backpropagation gradients of transmitter concentration matrices through time can be calculated by

$$(\nabla_{\mathbf{W}^l}\mathbf{E}, \nabla_{\mathbf{V}^l}\mathbf{E}) = \int_{t=0}^{T} (\nabla_{\mathbf{W}^l}\mathbf{E}_t, \nabla_{\mathbf{V}^l}\mathbf{E}_t)\, dt,$$

where

$$\begin{cases} \nabla_{\mathbf{W}^l}\mathbf{E}_t = \delta \boldsymbol{s}_t^l \dfrac{\partial \boldsymbol{s}_t^l}{\partial \boldsymbol{\alpha}_t^l} \dfrac{\partial \boldsymbol{\alpha}_t^l}{\partial \mathbf{W}^l}, \\ \nabla_{\mathbf{V}^l}\mathbf{E}_t = \delta \boldsymbol{s}_t^l \dfrac{\partial \boldsymbol{s}_t^l}{\partial \boldsymbol{\alpha}_t^l} \dfrac{\partial \boldsymbol{\alpha}_t^l}{\partial \mathbf{V}^l}. \end{cases} \tag{3.6}$$

Equation 3.6 consists of three terms. The first term can be unfolded as

$$\delta \boldsymbol{s}_t^l = a\,(\mathbf{W}^{l+1})^\top \left(\delta \boldsymbol{s}_t^{l+1} \odot \sigma'(\boldsymbol{\alpha}_t^{l+1} + \boldsymbol{\beta}_t^{t+1}\boldsymbol{i})\right),$$

which denotes the backpropagation error correction in the lth layer at time t, where $\odot$ is the point-wise operation. The second term $\partial\, \boldsymbol{s}_t^l / \partial\, \boldsymbol{\alpha}_t^l$ is a diagonal matrix, where its diagonal elements are the point-wise derivatives of activation σ. (3) $\partial\, \boldsymbol{\alpha}_t^l / \partial\, \mathbf{W}^l$ and $\partial\, \boldsymbol{\alpha}_t^l / \partial\, \mathbf{V}^l$ are tensors that belong to $\mathbb{R}^{T \times n_l \times n_{l-1}}$. So equation 3.6 becomes

$$\begin{cases} \nabla_{\mathbf{W}^l}\mathbf{E}_t = \left(\delta \boldsymbol{s}_t^l(1) \dfrac{\partial \boldsymbol{s}_t^l(1)}{\partial \boldsymbol{\alpha}_t^l(1)} \dfrac{\partial \boldsymbol{\alpha}_t^l(1)}{\partial \mathbf{W}^l}, \ldots, \delta \boldsymbol{s}_t^l(n_l) \dfrac{\partial \boldsymbol{s}_t^l(n_l)}{\partial \boldsymbol{\alpha}_t^l(n_l)} \dfrac{\partial \boldsymbol{\alpha}_t^l(n_l)}{\partial \mathbf{W}^l}\right), \\ \nabla_{\mathbf{V}^l}\mathbf{E}_t = \left(\delta \boldsymbol{s}_t^l(1) \dfrac{\partial \boldsymbol{s}_t^l(1)}{\partial \boldsymbol{\alpha}_t^l(1)} \dfrac{\partial \boldsymbol{\alpha}_t^l(1)}{\partial \mathbf{V}^l}, \ldots, \delta \boldsymbol{s}_t^l(n_l) \dfrac{\partial \boldsymbol{s}_t^l(n_l)}{\partial \boldsymbol{\alpha}_t^l(n_l)} \dfrac{\partial \boldsymbol{\alpha}_t^l(n_l)}{\partial \mathbf{V}^l}\right). \end{cases}$$

$\left(\partial\, \boldsymbol{\alpha}_t^l / \partial\, \mathbf{W}^l, \partial\, \boldsymbol{\alpha}_t^l / \partial\, \mathbf{V}^l\right)$ is the core of our CBP algorithm, including two backpropagation pipelines concerning $\mathbf{W}^l$ and $\mathbf{V}^l$. Regarding the neurotrophin density as an implicit variable, these tensors can be calculated as

$$\begin{cases} \dfrac{\partial \boldsymbol{\alpha}_t^l(i)}{\partial \mathbf{W}^l(j,k)} = \begin{cases} a\boldsymbol{s}_t^{l-1}(k) - b\left(\sum_{h=1}^{n_l} \mathbf{V}^l(i,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{W}^l(i,k)}\right) & , \ j=i, \\ -b\left(\sum_{h=1}^{n_l} \mathbf{V}^l(j,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{W}^l(j,k)}\right) & , \ j\neq i, \end{cases} \\ \dfrac{\partial \boldsymbol{\alpha}_t^l(i)}{\partial \mathbf{V}^l(j,h)} = \begin{cases} -b\left(\boldsymbol{r}_{t-1}^l(h) + \sum_{h=1}^{n_l} \mathbf{V}^l(i,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{V}^l(i,h)}\right) & , \ j=i, \\ -b\left(\sum_{h=1}^{n_l} \mathbf{V}^l(j,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{V}^l(j,h)}\right) & . \ j\neq i. \end{cases} \end{cases}$$

In the feedforward process, the FT neurons coupling generate the stimulus signals and neurotrophin densities. In the CBP process, the neurotrophin densities are indirectly regulated by the supervised stimulus signals, so we still need to supply the feedforward errors caused by the neurotrophin densities. The calculation procedure of partial derivatives $\left(\partial \boldsymbol{r}_t^l/\partial \mathbf{W}^l, \partial \boldsymbol{r}_t^l/\partial \mathbf{V}^l\right)$ is similar to that of $\left(\partial \boldsymbol{\alpha}_t^l/\partial \mathbf{W}^l, \partial \boldsymbol{\alpha}_t^l/\partial \mathbf{V}^l\right)$. Here, we directly provide the results:

$$\begin{cases} \dfrac{\partial \boldsymbol{r}_t^l(i)}{\partial \mathbf{W}^l(j,k)} = \dfrac{\partial \boldsymbol{r}_t^l(i)}{\partial \boldsymbol{\beta}^l(i)}\dfrac{\partial \boldsymbol{\beta}_t^l(i)}{\partial \mathbf{W}^l(j,k)}, \\ \dfrac{\partial \boldsymbol{r}_t^l(i)}{\partial \mathbf{V}^l(j,h)} = \dfrac{\partial \boldsymbol{r}_t^l(i)}{\partial \boldsymbol{\beta}^l(i)}\dfrac{\partial \boldsymbol{\beta}_t^l(i)}{\partial \mathbf{V}^l(j,h)}, \end{cases}$$

where

$$\begin{cases} \dfrac{\partial \boldsymbol{\beta}_t^l(i)}{\partial \mathbf{W}^l(j,k)} = \begin{cases} b\boldsymbol{s}_t^{l-1}(k) + a\left(\sum_{h=1}^{n_l} \mathbf{V}^l(i,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{W}^l(i,k)}\right) & , \ j=i, \\ a\left(\sum_{h=1}^{n_l} \mathbf{V}^l(j,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{W}^l(j,k)}\right) & , \ j\neq i, \end{cases} \\ \dfrac{\partial \boldsymbol{\beta}_t^l(i)}{\partial \mathbf{V}^l(j,h)} = \begin{cases} a\left(\boldsymbol{r}_{t-1}^l(h) + \sum_{h=1}^{n_l} \mathbf{V}^l(i,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{V}^l(i,h)}\right) & , \ j=i, \\ a\left(\sum_{h=1}^{n_l} \mathbf{V}^l(j,h)\dfrac{\partial \boldsymbol{r}_{t-1}^l(h)}{\partial \mathbf{V}^l(j,h)}\right) & . \ j\neq i. \end{cases} \end{cases}$$

Then we can obtain the gradients $(\nabla_{\mathbf{W}^l}\mathbf{E}, \nabla_{\mathbf{V}^l}\mathbf{E})$ and correct the concentration parameters according to

$$\begin{cases} \hat{\mathbf{W}}^l = \mathbf{W}^l - \eta \nabla_{\mathbf{W}^l} \mathbf{E}, \\ \hat{\mathbf{V}}^l = \mathbf{V}^l - \eta \nabla_{\mathbf{V}^l} \mathbf{E}, \end{cases}$$

where η is the learning rate. To ensure convergence, we adopt adaptive learning rates compatible with ADAM (Kingma & Ba, 2014) or WADA (Zhong et al., 2020). Before prediction (in the time interval $[0, T]$), we still need to update the imaginary parts (neurotrophin densities $\boldsymbol{r}_0^l, \ldots, \boldsymbol{r}_T^l$) as follows:

$$\begin{cases} \boldsymbol{s}_t^0 = \boldsymbol{x}_t, \\ \boldsymbol{r}_t^l = \sigma\left(b\hat{\mathbf{W}}^l \boldsymbol{s}_t^{l-1} + a\hat{\mathbf{V}}^l \boldsymbol{r}_{t-1}^l\right), \end{cases}$$

and reset the imaginary errors $\left(\partial \boldsymbol{r}_T^l / \partial \mathbf{W}^l, \partial \boldsymbol{r}_T^l / \partial \mathbf{V}^l\right)$ as zeros.

4 Experiments

The goal of experiments is to validate the power of FTNet in handling spatiotemporal data. We compare FTNet with several mature fully connected feedforward neural networks, including FCN with MP neurons and spiking neural networks (SNNs) and some state-of-the-art models on three data sets. We force all contenders to adopt the same setting or parameter magnitudes as possible, except for SNNs, which encourage more neurons to ensure convergence. The entire data set is partitioned into three parts: training, validation, and testing sets, with a partition ratio of around 56%:24%:20%. We employ tanh as the activation function. Other hyperparameters cannot be fixed across tasks; otherwise, the performance may be embarrassingly unsatisfactory. We examine various configurations on the validation set and pick out the best validation performance. Finally, we retrain each model on the combination of training and validation data sets and predict the testing one. The following tables list the final testing performance.

4.1 Image Recognition on Pixel-by-Pixel MNIST. We first conduct the experiments on a benchmark image recognition task to evaluate the ability of FTNet for processing spatial information. Pixel-by-pixel MNIST (Le, Jaitly, & Hinton, 2015) is a famous challenging image recognition data set, a standard benchmark to test the performance of a learning algorithm. We adopt the typical split of 60,000 training and 10,000 testing samples without data augmentation and select 30% from the training samples to constitute the validation set.

Here, we compare our FTNet with another bio-inspired neural network, that is, SNN. Following a similar setup to Pillow, Paninski, Uzzell, Simoncelli, and Chichilnisky (2005) and Zhang, Zhang, and Zhou (2019), each

Table 1: Accuracy of Comparative Models for the Task of Classifying Pixel-Pixel MNIST.

Types	Models	Cascade	Paras	Accuracy (%)
CNNs	CNN-SVM (Niu & Suen, 2012)	–	–	98.79
	LeNet-5 (LeCun, Bottou, Bengio, & Haffner, 1998)	–	6.2×10^4	99.05
MP-based	FCN with MP neurons	size(*,40,10)	3.2×10^4	86.47
	FCN with MP neurons	size(*,150,10)	1.2×10^5	93.82
SNNs	SLAYER	size(*,300,10)	2.4×10^5	94.13
	BSNN	size(*,300,10)	3.2×10^5	96.65
Recurrent Networks	uRNN (Arjovsky et al., 2016)	size(*,150,10)	3.3×10^4	97.28
	CNN-RNN	size(*,150,10)	2.9×10^4	95.21
	CNN-LSTM	size(*,150,10)	8.2×10^4	98.66
Our Work	FT0	size(*,0,10)	4.0×10^2	92.87
	FT1	size(*,150,10)	3.0×10^4	**99.12**

handwritten digit image is converted into a spiking train with a formation of $784 \times T$ binary matrix via Poisson encoding. For classification, we use the spiking counting strategy. During training, we specify a target of 20 spikes for the true neuron and 5 spikes for each false neuron; while testing, the output class is the one that generates the highest spike count. Both SNN models, SLAYER (Shrestha & Orchard, 2018) and BSNN (Zhang et al., 2019), converge within 200 iterations.

All models except SLAYER and BSNN employ the softmax function for classification and are optimized by a cross-entropy loss. These models converge within 800 iterations. The experimental results, set out in Table 1, show that FT1 achieves highly competitive performance.

4.2 Univariate Time Series Forecasting: Yancheng Automobile Registration. We conduct experiments on the Yancheng Automobile Registration Forecasting[1] competition, a real-world univariate time series forecasting task. This competition requires players to use the daily automobile registration records of a certain period (nearly 1000 dates) in the past to predict the number of automobile registrations per day for a period of time in the future. Although the actual competition allows the contestant to develop other data sets or information as an aid freely, we only consider the total number of automobile registrations for five car brands, not including any specific date information. This task is challenging since accurate automobile registration records are a mixture of five car brands, yet there are

[1] https://tianchi.aliyun.com/competition/entrance/231641/information

Table 2: MSE and Settings of Comparative Models for the Task of Forecasting Yancheng Automobile Registration Records.

Types	Models	Settings	Paras	MSE (10^5)
Statistical Models	ARIMA	$(p, d, q) = (6, 1, 3)$	–	84.5129
	MAR (Won & Gray, 2013)	–	–	92.6458
	AGP (Blaauw & Christen, 2011)	–	–	41.0147
	KNNs (Yang, Bukkapatnam, & Barajas, 2011)	$(K, w) = (1, 1)$	–	31.2573
Neural Networks	FCN with MP neurons	size(5,50,1)	200	19.5360
	FCN with MP neurons	size(5,500,1)	3.0×10^3	19.5360
	NARXnet (Guzman, Paz, & Tagert, 2017)	size(5,50,1)	–	20.2631
	RNN (Gers & Schmidhuber, 2000)	size(5,50,1)	2.8×10^3	18.0729
	LSTM (Hochreiter & Schmidhuber, 1997)	size(5,50,1)	8.3×10^3	10.7250
	LSTNet (Lai, Chang, Yang, & Liu, 2018)	size(5,64,1)	9.7×10^3	8.4176
Our Work	FT0	size(5,0,1)	<100	16.4721
	FT1	size(5,50,1)	2.9×10^3	**4.5067**

many missing data and sudden changes caused by holidays or other guiding factors that we cannot know in advance.

We compare our proposed FTNet with several state-of-the-art statistical models and neural networks (Cheng et al., 2015) and evaluate the performance by mean square error (MSE). All neural network models converge within 100 epochs. The experimental results, summarized in Table 2, confirm the superiority of our FTNet to other models.

4.3 Multivariate Time Series Forecasting: Traffic Prediction on HDUK. We also validate FTNet on the Highway Data of United Kingdom (HDUK),[2] a representative multivariate traffic prediction data set. HDUK contains massive average journey time, speed, and traffic flow information for 15-minute periods on all motorways and A-level roads managed by the Highways Agency (known as the Strategic Road Network in England). Journey times and speeds are estimated using various sources, including automatic number plate recognition cameras, in-vehicle global positioning systems, and inductive loops built into the road surface. For convenience, we choose roads with relatively large several traffic flow for study and collect the traffic data of the 12 months in 2011 and partition the first 7 months, the subsequent 3 months, and the last 2 months as training, validation, and testing

[2] http://data.gov.uk/dataset/dft-eng-srn-routes-journey-times.

Table 3: MSE and Confusion Accuracy of Comparative Models for the Task of Forecasting HDUK.

Data Sets	Models & Settings Evaluation(%)	NARXnet □	RNN △	LSTM △	LSTNet △	FT0 ◊	FT1 △
A1	MSE	0.0469	0.1499	0.0262	0.0247	0.1169	**0.0221**
	TPR	97.20	97.20	98.13	98.13	96.20	**99.07**
	TNR	95.29	91.74	96.47	97.41	94.12	**97.65**
A1033	MSE	0.1584	0.1716	0.1397	0.1401	0.1372	**0.1119**
	TPR	88.51	93.10	94.25	94.11	94.11	**96.55**
	TNR	91.43	93.33	92.38	92.25	92.38	**97.14**
A11	MSE	0.1754	0.1770	0.1725	0.1690	0.1755	**0.1651**
	TPR	97.06	96.08	97.06	97.06	97.06	**99.02**
	TNR	95.56	91.11	93.33	95.93	**96.67**	94.44

□Denotes a size(*,0,1) cascade structure and iterates 100 times. △Denotes size(*,100,1) cascade structure and iterates 100 times. ◊Indicates a network configuration with 100 recurrent neurons and 32-dimensional convolution layer and iterates 100 epochs. The best performance on each data set is in bold.

data sets, respectively. For this forecasting task, we input the observation values of (Total Traffic Flow & Travel Time & Fused Average Speed & Link Length) in the previous 8 time intervals and predict one-step ahead value of (Total Traffic Flow).

Besides, "decayed prediction" is a predominant issue in multivariate time series forecasting (see Tschernig & Yang, 2000, for detailed) though one adopts over-parameterized architectures (Zhou, 2021). In this case, it is insufficient to use only one indicator, MSE, to evaluate the forecasting models' performance. To alleviate this issue, we use the "mean-std" normalizer and employ the confusion accuracy, which consists of the true positive rate (TPR) and the true negative rate (TNR), as auxiliary evaluation indicators (Zhang & Zhou, 2020).

Table 3 shows the comparative results of FTNet and other neural networks on a collection of HDUK data sets. FTNet achieves the best performance as other competing neural networks under the same parameter magnitude.

5 Discussion

5.1 About Modeling Synaptic Plasticity. In section 1.1, we introduced the modeling mechanisms of the classical MP and spiking neuron models. At present, modeling discrete action potentials or "pulses" to exchange and transmit information is still the mainstream of realizing machine intelligence with neuromorphic computing (see Roy, Jaiswal, & Panda, 2019). Recently, some researchers have put attention on another kind of synaptic plasticity, short-term synapticplasticity (STP). In neuroscience, STP

describes a millisecond-level phenomenon that presynaptic activity increases the Ca^{2+} concentration in the synaptic cleft but depletes neurotransmitter stores (Zucker & Regehr, 2002; Regehr, 2012). This phenomenon appears on a timescale of tens of milliseconds to a few minutes; thus, it is distinct from our LTP or LTD, which usually lasts from minutes to hours and is more commonly known. In mathematical formulation, the modeling manner of STP is similar to the spiking one; the icon-integrated procedure is often simulated by a first-order equation and reset to zero (rest voltage) once its concentration exceeds a pregiven threshold (Masse, Yang, Song, Wang, & Freedman, 2019; Yang, Song, Newsome, & Wang, 2017).

There is still a lot of seminal work in neuroscience to develop neuron models from modeling communication particulars such as neurotransmitters, neurotrophins, and receptors (Bertram, Smith, & Sherman, 1999; Holcman & Triller, 2006; Trevathan et al., 2017). These bio-plausible neuron models provide the possibility for a better understanding of biological neurons or brains, although most of them have failed in practical applications. For example, only a tiny part of SNNs has made passable progress on small-scale image recognition tasks (Shrestha & Orchard, 2018).

Our work proposes the FT neuron model, starting from a view of "neuroscience-inspired artificial intelligence" (Hassabis, Kumaran, Summerfield, & Botvinick, 2017; Hao, Andolina, Wang, & Zhang, 2021). This study focuses on the development for artificial intelligence technology rather than only providing a better bio-plausible understanding for biological neurons. The experiments in section 4 demonstrate the superiority and effectiveness of our proposed FTNet in handling spatio-temporal data.

5.2 About Complex-Valued Reaction. There has been a great deal of effort on developing neural networks using complex-valued formation. For example, Arjovsky et al. (2016) force the connection weights as a unitary matrix in the complex-valued domain for circumventing the issue of vanishing and exploding gradients in RNNs. Trabelsi et al. (2017) propose a complex-valued connection matrix that works like a convolution building for deep neural networks and later develops this technology into a quaternion-valued formation (Parcollet et al., 2018).

Our proposed FT model is essentially different from previous work. First, the motivations are different. Existing neural networks relative to complex-valued formation are motivated to explore an atomic component for overcoming drawbacks or improving neural networks' representational capacity. However, the FT model is a novel type of neuron model that depicts the neurotransmitter communication mechanism in synaptic plasticity and is formulated by a two-variable, two-valued function. The complex-valued reaction is a valid implementation for the two-variable, two-valued function as well as the FT model. Second, the use of complex-valued formation is different. The connection matrices in Arjovsky et al. (2016) are asked to have complex-valued eigenvalues with an absolute value 1. Thus, the

gradients of recurrent networks are guaranteed to avoid the explosion. The inputs of Trabelsi et al. (2017) and Parcollet et al. (2018) are preprocessed into a complex-valued or quaternion-valued formation for facilitating subsequent convolution-like operation. In this work, we employ a new variable, the neurotrophin density, to model the behavior that the tissue size of synapses would change in the learning process. The neurotrophin densities are regarded as the complex-valued function's imaginary parts for the complex-valued reaction, leading to a local recurrent system in the FT model. The experimental results in section 4 show that the FT model has potential for handing spatiotemporal data.

Finally, we have to note that the complex-valued reaction is just a formulation of the FT model, and many valid implementation approaches are worthy of being tried. Furthermore, numerous holomorphic conversion functions and activations are worthy of being explored; custom-built conversion functions may extract potential and significant adjoint features on some real-world applications. Besides, in this letter, we have provided only the simplest fully connected feedforward network, the FTNet. Various alternative network architectures can be explored in the future.

5.3 Comparison with RNNs. As noted, the proposed FT model derives a local recurrent system. It easily reminds us of the typical recurrent neural networks. The RNN's unit is formalized by a real-valued function $s_t = g(x_t, s_{t-1}; w, v)$, while the FT model is dominated by $(s_t, r_t) = f(x_t, r_{t-1}; w, v)$. Obviously, the RNN's unit is a special case of the FT model; these two models are equivalent to each other when presetting $r_t = s_t$. Therefore, the FT model may have a broader representational capacity than the RNN's unit. Further experiments in section 4 demonstrate the superiority of the FT model.

6 Conclusion

We have proposed the FT model, a new model for the bio-plausible nervous system. In contrast to the traditional MP model regarding the full nervous synapse as a real-valued parameter, the FT model meticulously depicts the neurotransmitter communication mechanism in synaptic plasticity. Specifically, we employ a pair of parameters to model the transmitters and put up a variable to denote the regulated neurotrophin density. The FT model has a formulation of a two-variable, two-valued function, thus taking the MP model and RNN's unit as its special cases. To demonstrate the power and potential of our proposed FT model, we present the FTNet using the most common fully connected feedforward architecture. We employ the holomorphic complex-valued reaction as an implementation paradigm for simplicity and then offer a practicable and effective CBP algorithm for training an FTNet. The experiments conducted on wide-range tasks confirm the effectiveness and superiority of our model.

Acknowledgments

This work was supported by the National Science Foundation of China (61921006) and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

Arjovsky, M., Shah, A., & Bengio, Y. (2016). Unitary evolution recurrent neural networks. In *Proceedings of the 33rd International Conference on Machine Learning* (pp. 1120–1128). PMLR.

Bertram, R., Smith, G. D., & Sherman, A. (1999). Modeling study of the effects of overlapping Ca2+ microdomains on neurotransmitter release. *Biophysical Journal*, *76*(2), 735–750. http://dx.doi.org/10.1016/S0006-3495(99)77240-1, PubMed: 9929478

Bi, G.-Q., & Poo, M.-M. (1998). Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, *18*(24), 10464–10472. http://dx.doi.org/10.1523/JNEUROSCI.18-24-10464.1998, PubMed: 9852584

Blaauw, M., & Christen, J. A. (2011). Flexible paleoclimate age-depth models using an autoregressive gamma process. *Bayesian Analysis*, *6*(3), 457–474. http://dx.doi.org/:10.1214/ba/1339616472

Cheng, C., Sa-Ngasoongsong, A., Beyca, O., Le, T., Yang, H., Kong, Z., & Bukkapatnam, S. T. (2015). Time series forecasting for nonlinear and non-stationary processes: A review and comparative study. *IIE Transactions*, *47*(10), 1053–1071. https://doi.org/10.1080/0740817X.2014.999180

Cooke, S., & Bliss, T. (2006). Plasticity in the human central nervous system. *Brain*, *129*(7), 1659–1673. https://doi.org/10.1093/brain/awl082

Debanne, D., Campanac, E., Bialowas, A., Carlier, E., & Alcaraz, G. (2011). Axon physiology. *Physiological Reviews*, *91*(2), 555–602. https://doi.org/10.1152/physrev.00048.2009, PubMed: 21527732

Gers, F. A., & Schmidhuber, J. (2000). Recurrent nets that time and count. In *Proceedings of the 2000 International Joint Conference on Neural Networks* (3:189–194). Piscataway, NJ: IEEE. http://dx.doi.org/10.1109/IJCNN.2000.861302

Gerstner, W., & Kistler, W. M. (2002). *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9780511815706

Gong, L.-Q., He, L.-J., Dong, Z.-Y., Lu, X.-H., Poo, M.-M., & Zhang, X.-H. (2011). Postinduction requirement of NMDA receptor activation for late-phase long-term potentiation of developing retinotectal synapses in vivo. *Journal of Neuroscience*, *31*(9), 3328–3335. http://dx.doi.org/10.1523/JNEUROSCI.5936-10.2011, PubMed: 21368044

Guzman, S. M., Paz, J. O., & Tagert, M. L. M. (2017). The use of NARX neural networks to forecast daily groundwater levels. *Water Resources Management*, *31*(5), 1591–1603. https://doi.org/10.1007/s11269-017-1598-5

Hao, W., Andolina, I. M., Wang, W., & Zhang, Z. (2021). Biologically inspired visual computing: The state of the art. *Frontiers of Computer Science*, *15*(1), 1–15. https://doi.org/10.1007/s11704-020-9001-8

Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron*, *95*(2), 245–258. http://dx.doi.org/10.1016/j.neuron.2017.06.011, PubMed: 28728020

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hodgkin, A. L., & Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, *117*(4), 500. http://dx.doi.org/10.1113/jphysiol.1952.sp004764, PubMed: 12991237

Holcman, D., & Triller, A. (2006). Modeling synaptic dynamics driven by receptor lateral diffusion. *Biophysical Journal*, *91*(7), 2405–2415. http://dx.doi.org/10.1529/biophysj.106.081935, PubMed: 16844759

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization.* arXiv:1412.6980.

Lai, G.-K., Chang, W.-C., Yang, Y.-M., & Liu, H.-X. (2018). Modeling long- and short-term temporal patterns with deep neural networks. In *Proceedings of the 41st International Conference on Research and& Development in Information Retrieval* (pp. 95–104). New York: ACM. https://doi.org/10.1145/3209978.3210006

Le, Q. V., Jaitly, N., & Hinton, G. E. (2015). *A simple way to initialize recurrent networks of rectified linear units.* arXiv:1504.00941.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. http://dx.doi.org/10.1109/5.726791

Lodish, H., Berk, A., Kaiser, C. A., Krieger, M., Scott, M. P., Bretscher, A., . . . Matsudaira, P. (2008). *Molecular cell biology*. New York: Macmillan.

Masse, N. Y., Yang, G. R., Song, H. F., Wang, X.-J., & Freedman, D. J. (2019). Circuit mechanisms for the maintenance and manipulation of information in working memory. *Nature Neuroscience*, *22*(7), 1159–1167. https://doi.org/10.1101/305714

Mattson, M. P. (1988). Neurotransmitters in the regulation of neuronal cytoarchitecture. *Brain Research Reviews*, *13*(2), 179–212. http://dx.doi.org/10.1016/0165-0173(88)90020-3, PubMed: 2898278

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, *5*(4), 115–133. https://doi.org/10.1007/BF02478259

Moon, J., Wu, Y., Zhu, X., & Lu, W. D. (2021). Neural connectivity inference with spike-timing dependent plasticity network. *Science China Information Sciences*, *64*(6), 1–10. https://doi.org/10.1007/s11432-021-3217-0

Niu, X.-X., & Suen, C. Y. (2012). A novel hybrid CNN-SVM classifier for recognizing handwritten digits. *Pattern Recognition*, *45*(4), 1318–1325. https://doi.org/10.1016/j.patcog.2011.09.021

Parcollet, T., Ravanelli, M., Morchid, M., Linare's, G., Trabelsi, C., De Mori, R., & Bengio, Y. (2018). *Quaternion recurrent neural networks*. arXiv:1806.04418.

Park, H., & Poo, M.-M. (2013). Neurotrophin regulation of neural circuit development and function. *Nature Reviews Neuroscience*, *14*(1), 7–23. http://dx.doi.org/10.1038/nrn3379, PubMed: 23254191

Pillow, J. W., Paninski, L., Uzzell, V. J., Simoncelli, E. P., & Chichilnisky, E. (2005). Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model. *Journal of Neuroscience*, *25*(47), 11003–11013. http://dx.doi.org/10.1523/JNEUROSCI.3305-05.2005, PubMed: 16306413

Regehr, W. G. (2012). Short-term presynaptic plasticity. *Cold Spring Harbor Perspectives in Biology*, *4*(7), a005702. https://doi.org/10.1016/j.conb.2011.02.003

Roy, K., Jaiswal, A., & Panda, P. (2019). Towards spike-based machine intelligence with neuromorphic computing. *Nature*, *575*(7784), 607–617. http://dx.doi.org/10.1038/s41586-019-1677-2, PubMed: 31776490

Schinder, A. F., & Poo, M.-M. (2000). The neurotrophin hypothesis for synaptic plasticity. *Trends in Neurosciences*, *23*(12), 639–645. http://dx.doi.org/10.1016/s0166-2236(00)01672-6, PubMed: 11137155

Shrestha, S. B., & Orchard, G. (2018). Slayer: Spike layer error reassignment in time. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems 31* (pp. 1412–1421). Red Hook, NY: Curran.

Stein, R. B. (1965). A theoretical analysis of neuronal variability. *Biophysical Journal*, *5*(2), 173–194. http://dx.doi.org/10.1016/s0006-3495(65)86709-1, PubMed: 14268952

Trabelsi, C., Bilaniuk, O., Zhang, Y., Serdyuk, D., Subramanian, S., Santos, J. F., . . . Pal, C. J. (2017). *Deep complex networks.* arXiv:1705.09792.

Trevathan, J. K., Yousefi, A., Park, H. O., Bartoletta, J. J., Ludwig, K. A., Lee, K. H., & Lujan, J. L. (2017). Computational modeling of neurotransmitter release evoked by electrical stimulation: Nonlinear approaches to predicting stimulation-evoked dopamine release. *ACS Chemical Neuroscience*, *8*(2), 394–410. https://doi.org/10.1021/acschemneuro.6b00319

Tschernig, R., & Yang, L. (2000). Nonparametric lag selection for time series. *Journal of Time Series Analysis*, *21*(4), 457–487. https://doi.org/10.2307/2291003

VanRullen, R., Guyonneau, R., & Thorpe, S. J. (2005). Spike times make sense. *Trends in Neurosciences*, *28*(1), 1–4. http://dx.doi.org/10.1016/j.tins.2004.10.010, PubMed: 15626490

Wang, Z., Ma, Y., Cheng, F., & Yang, L. (2010). Review of pulse-coupled neural networks. *Image and Vision Computing*, *28*(1), 5–13. https://doi.org/10.1016/j.imavis.2009.06.007

Won, C. S., & Gray, R. M. (2013). *Stochastic image processing*. New York: Springer Science & Business Media.

Yang, G. R., Song, H. F., Newsome, W. T., & Wang, X.-J. (2017). *Clustering and compositionality of task representations in a neural network trained to perform many cognitive tasks.* bioRxiv:183632. http://dx.doi.org/10.1038/s41593-018-0310-2, PubMed: 30643294

Yang, H., Bukkapatnam, S. T., & Barajas, L. G. (2011). Local recurrence based performance prediction and prognostics in the nonlinear and nonstationary systems. *Pattern Recognition*, *44*(8), 1834–1840. http://dx.doi.org/10.1016/j.patcog.2011.01.010

Zhang, S.-Q., Zhang, Z.-Y., & Zhou, Z.-H. (2019). *Bifurcation spiking neural network.* arXiv:1909.08341.

Zhang, S.-Q., & Zhou, Z.-H. (2020). Harmonic recurrent process for time series forecasting. In *Proceedings of the 24th European Conference on Artificial Intelligence* (pp. 1714–1721). Amsterdam: IOS Press. http://dx.doi.org/10.3233/FAIA200284

Zhong, H., Chen, Z., Qin, C., Huang, Z., Zheng, V. W., Xu, T., & Chen, E. (2020). Adam revisited: A weighted past gradients perspective. *Frontiers of Computer Science*, *14*(5), 1–16. http://dx.doi.org/10.1007/s11704-019-8457-x

Zhong, H., Sia, G.-M., Sato, T. R., Gray, N. W., Mao, T., Khuchua, Z., . . . Svoboda, K. (2009). Subcellular dynamics of type II PKA in neurons. *Neuron*, *62*(3), 363–374. http://dx.doi.org/10.1016/j.neuron.2009.03.013, PubMed: 19447092

Zhou, Q., Homma, K. J., & Poo, M.-M. (2004). Shrinkage of dendritic spines associated with long-term depression of hippocampal synapses. *Neuron*, *44*(5), 749–757. http://dx.doi.org/10.1016/j.neuron.2004.11.011, PubMed: 15572107

Zhou, Z.-H. (2021). Why over-parameterization of deep neural networks does not overfit? *Science China Information Sciences*, *64*(1), 1–3. https://doi.org/10.1007/s11432-020-2885-6

Zucker, R. S., & Regehr, W. G. (2002). Short-term synaptic plasticity. *Annual Review of Physiology*, *64*(1), 355–405. http://dx.doi.org/10.1146/annurev.physiol.64.092501.114547, PubMed: 11826273

Received September 28, 2020; accepted May 18, 2021.