基于脉冲序列标识的深度脉冲神经网络时空反向传播算法

王子华[①] 叶 莹[①④] 刘洪运[②③] 许 燕[①] 樊瑜波[①] 王卫东*[①②③]

[①](北京航空航天大学生物与医学工程学院 北京 100191)

[②](中国人民解放军总医院医学创新研究部生物工程研究中心 北京 100853)

[③](工业和信息化部生物医学工程与转化医学重点实验室 北京 100853)

[④](华北计算技术研究所 北京 100083)

摘 要：尖峰放电的脉冲神经网络(SNN)具有接近大脑皮层的信号处理模式，被认为是实现大脑启发计算的重要途径。但是，目前对于深度脉冲神经网络的学习仍缺乏有效的监督学习算法。受尖峰放电速率标识的时空反向传播算法的启发，该文提出一种面向深度脉冲神经网络训练的基于时间脉冲序列标识的监督学习算法，通过定义突触后电位和膜电位反传迭代因子分别分析脉冲神经元的空间和时间依赖关系，使用替代梯度的方法解决反传过程中不连续可微的问题。不同于现有基于尖峰放电速率标识的学习算法，该算法能够充分反映了脉冲神经网络输出的时间脉冲序列的动态特性。因此，所提算法非常适合应用于需要较长时间序列标识的计算任务，例如行为的时间脉冲序列控制。该文在静态图像数据集CIFAR10和神经形态数据集NMNIST上验证了所提算法的有效性，在所有这些数据集上都显示出良好的性能，这有助于进一步研究基于时间脉冲序列应用的大脑启发计算。

关键词：脉冲神经网络；监督学习；误差反向传播；时间脉冲序列标识；替代梯度

中图分类号：TP18 **文献标识码**：A **文章编号**：1009-5896(2024)00-0001-09

DOI: 10.11999/JEIT230705

Spiking Sequence Label-Based Spatio-Temporal Back-Propagation Algorithm for Training Deep Spiking Neural Networks

WANG Zihua[①] YE Ying[①④] LIU Hongyun[②③] XU Yan[①]
FAN Yubo[①] WANG Weidong[①②③]

[①](*School of Biological and Medical Engineering, Beihang University, Beijing* 100191, *China*)

[②](*Research Center for Biomedical Engineering, Medical Innovation & Research Division, Chinese PLA General Hospital, Beijing* 100853, *China*)

[③](*Key Laboratory of Biomedical Engineering and Translational Medicine, Ministry of Industry and Information Technology, Beijing* 100853, *China*)

[④](*North China Institute of Computing Technology, Beijing* 100083, *China*)

Abstract: Spiking Neural Networks (SNN) have a signal processing mode close to the cerebral cortex, which is considered to be an important approach to realize brain-inspired computing. However, the lack of effective supervised learning algorithms for deep spiking neural networks has been a great challenge for spiking sequence label-based brain-inspired computing tasks. A supervised learning algorithm for training deep spiking neural network is proposed in this paper. It is an error backpropagation algorithm that uses surrogate gradient to solve the problem of non-differentiable spike generation function, and define the postsynaptic potential and membrane potential reversal iteration factors represent the spatial and temporal dependencies of pulsed neurons, respectively. It differs from existing learning algorithms based on firing rate encoding, fully reflects analytically the temporal dynamic properties of the spiking neuron. Therefore, the algorithm proposed in this paper is well-suited for application to tasks that require longer time sequences rather than spiking firing rates,

收稿日期：2023-07-15；改回日期：2024-03-12；网络出版：2024-04-09

*通信作者： 王卫东 wangwd301@126.com

基金项目：科技创新——2030“新一代人工智能”重大项目(2020AAA0105800)

Foundation Item: The Scientific and Technological Innovation 2030 - “New Generation Artificial Intelligence” Major Project (2020AAA0105800)

such as behavior control. Proposed algorithm is validated on the static image datasets CIFAR10, and the neuromorphic dataset NMNIST. It shows good performance on all these datasets, which helps to further investigate spike-based brain-inspired computation.

Key words: Spiking Neural Networks (SNN); Supervised learning; Error backpropagation; Temporal spike sequence labeling; Surrogate gradient

1 引言

脉冲神经网络(Spiking Neural Networks, SNN)是在结构形态和时间动态方面更加接近生物神经网络特性的新一代神经网络模型[1]，被称为第3代人工神经网络，具有时间稀疏、计算异步、事件驱动的特点，可在神经形态芯片上[2–4]实现低功耗运行，能够执行复杂时间序列的智能计算与控制，在类脑智能领域受到广泛关注[5]。但是，与传统的神经网络——模拟神经网络(Analogic Neural Networks, ANN)不同，由非线性微方程或方程组描述[6]的SNN的输出是不连续的脉冲信号，具有不可微和时空动态的复杂性，给SNN的训练带来了极大的困难。

SNN监督学习算法主要有两类：第1类是ANN到SNN的训练迁移法，第2类是标识数据集的直接训练法。训练迁移法是利用成熟的经典误差反向传播(Back Propagation, BP)算法训练ANN，再将训练所得的归一化权重迁移到具有相同网络结构的SNN，以避免SNN的脉冲不可微问题带来的训练困难，且提高了训练效率[7–11]。此类方法只适用于放电速率标识的训练集，且仅当ANN神经元的激活函数与SNN神经元的激励-放电速率归一化一致的情况下，同时训练窗口足够大时才能有较好的转换性能。

直接训练法是将BP算法直接推广到SNN的训练，根据标识码类型的不同，可分为两类：基于尖峰放电速率标识监督学习算法[12,13]和基于脉冲时间序列标识监督学习算法。放电速率标识的损失函数是SNN输出的放电速率与放电速率标识值之间的误差的均方值，这类算法多数采用时间反向传播(Backpropagation Through Time, BPTT)的方法实现网络输出误差在隐含层之间的时间递归传播。由Wu等[12]人率先提出的时空反向传播算法(Spatio-Temporal Back Propagation, STBP)就是一个基于尖峰放电速率的高效高性能监督学习算法。STBP反向传播的迭代形式简单，使用替代梯度解决脉冲发放函数带来的不可微分问题，考虑了时间边界条件的复杂性，同时需要将SNN输出的时间脉冲序列(Temporal Spiking Sequence, TSS)转换为放电速率，从而使得SNN失去了时间结构信息的学习能力。脉冲序列标识的损失函数是SNN输出的时间脉冲序列与时间脉冲序列标识值之间的Hamming距离或van Rossum[14]距离的均方值，这类方法使得SNN获得了学习时间结构信息的能力，受到业内的普遍重视。但是，这种计算时间序列误差的方式使算法变得更加复杂，阻碍了脉冲时序标识指导的学习算法的发展。

受BP算法的启发，Bohte等人[15]提出了首个基于单个脉冲放电时间的SNN监督学习的BP算法——SpikeProp，由于该算法及其变体要求每个神经元只能发放一个脉冲，其它脉冲固定不变，也就限制了该算法在实际任务中的应用[12, 15–17]。目前，研究时间脉冲序列标识的SNN学习算法相对较少，Zenke等人[18]提出的SuperSpike是较早采用脉冲时间序列描述损失函数的算法，但该算法对神经元模型进行了简化，从而忽略了膜电位时间的依赖关系，没有考虑反向传播的时间迭代，难以应用于深度SNN的训练。Zhang等人[19]提出了一种针对脉冲时间序列标识样本的监督学习算法——脉冲时间序列反向传播(Temporal Spike Sequence Learning via Back Propagation, TSSL-BP)算法。TSSL-BP使用van Rossum距离评价输出时间脉冲序列和标识时间脉冲序列之间的误差作为网络损失，通过分析神经元间和神经元内的依赖性来分解误差进行反向传播，该算法充分考虑了误差在时间和空间结构上的传播，并在通用数据集上取得了较好的学习效果。但其梯度的计算依赖于神经元放电时间，该算法由于它的分段枚举性，使得当实际输出中第一个脉冲产生时间晚于目标序列的首个脉冲放电时间时，将导致此部分误差无法进行计算回传，使得在长时程脉冲时间序列标识样本的情况下训练变得困难。

受尖峰放电速率标识的时空反向传播算法的启发，本文将时空反向传播算法推广到脉冲序列标识的深度SNN的训练中，既保持了时空反向传播算法具有的简单高效时间递归的空间反向传播形式，又保留了SNN具备的学习时间动态变化能力。该算法还适用于深度SNN训练，如多层感知器和卷积神经网络等。在CIFAR10[20]等通用静态图像数据集以及神经形态数据集NMNIST[21]上进行了该算法的测试，均取得较好的学习效果，验证了本文提出的SNN学习算法的有效性和良好性能。

2　模型预算法

2.1　脉冲神经元模型

本文选择漏电积分放电(Leaky Integrate-and-Fire, LIF)脉冲神经元模型作为基本模型，其膜电位$u(t)$满足下列方程

$$\tau_{\mathrm{m}}\frac{\partial u}{\partial t}=-u+R_{\mathrm{m}}I_0(t),\ t\in[0,T] \tag{1}$$

这里τ_{m}为膜时间常量，R_{m}为膜电阻，$I_0(t)$为刺激电流。将微分方程(1)的解离散化，得到时间序列$u(t_n),\ t_n=n\Delta t,\ n\in[0,N_{\mathrm{T}}-1]$，其中$\Delta t=\dfrac{T}{N_{\mathrm{T}}-1}$为采样周期，$N_{\mathrm{T}}$为膜电位样本数量。根据欧拉迭代法，$u(t_n)$近似满足

$$u(t_n)=\tau u(t_{n-1})+v(t_n) \tag{2}$$

其中$\tau=\left(1-\frac{\Delta t}{\tau_{\mathrm{m}}}\right),\ v(t_n)=\dfrac{\Delta t}{\tau}R_{\mathrm{m}}I_0(n\Delta t)$。

考虑到神经元尖峰放电、不应期和复位过程，根据差分方程(2)，离散化形式的脉冲神经元模型可以写成下式，见图1。式中*代表卷积操作，式(3a)中的求和对时间进行，求和范围是整个时间窗口[0,T]。

$$v(t_n)=\sum w_k(\varepsilon(t_n)*o_k(t_n)) \tag{3a}$$

$$u(t_n)=(\tau u(t_{n-1})+v(t_n))(1-o(t_{n-1}))+V_{\mathrm{rest}}o(t_{n-1}) \tag{3b}$$

$$o(t_n)=g(u(t_n)-V_{th}) \tag{3c}$$

其中，$o_k(t_n)$和w_k分别是其它神经元输入的零幺脉冲序列和其他神经元与该神经元的连接权重，V_{rest}是神经元静息电位，V_{th}是神经元阈值电位，此处th代表threshold，与时间无关，$g(*)$是阶跃函数。突触后电位$v(t_n)$(PostSynaptic Potential, PSP)在模型中起到了关键作用，其中卷积核定义为

$$\epsilon(t_n)=\begin{cases}\dfrac{t_n}{\tau_{\mathrm{s}}}\mathrm{e}^{1-\frac{t_n}{\tau_{\mathrm{s}}}}, & t_n>0\\ 0, & \text{其他}\end{cases}$$

$\varepsilon'(t_n)$是$\varepsilon(t_n)$的微分，

$$\varepsilon'(t_n)=\begin{cases}\left(\dfrac{1}{\tau_{\mathrm{s}}}-\dfrac{t_n}{\tau_{\mathrm{s}}^2}\right)\mathrm{e}^{1-\frac{t_n}{\tau_{\mathrm{s}}}}, & t_n>0\\ 0, & \text{其他}\end{cases}$$

其τ_{s}是PSP响应时间常量。

2.2　脉冲网络模型

考虑多层SNN结构，如图1所示。记该网络中第l层中第i个神经元在t_n时刻的输出为$o_i^l(t_n)$，w_{ij}^{l+1}表示第l层中第i个神经元与第l+1层中第j个神经元之间的连接权重，第l层神经元个数为N_l。则第l层第i个神经元$u_i^l(t_n)$可以表示为

$$u_i^l(t_n)=\left(\tau u_i^l(t_{n-1})+\sum_{h=1}^{N_{l-1}}w_{hi}^l\varepsilon(t_n)*o_h^{l-1}(t_n)\right)\cdot\left(1-o_i^l(t_{n-1})\right)+V_{\mathrm{rest}}o_i^l(t_{n-1}) \tag{5}$$

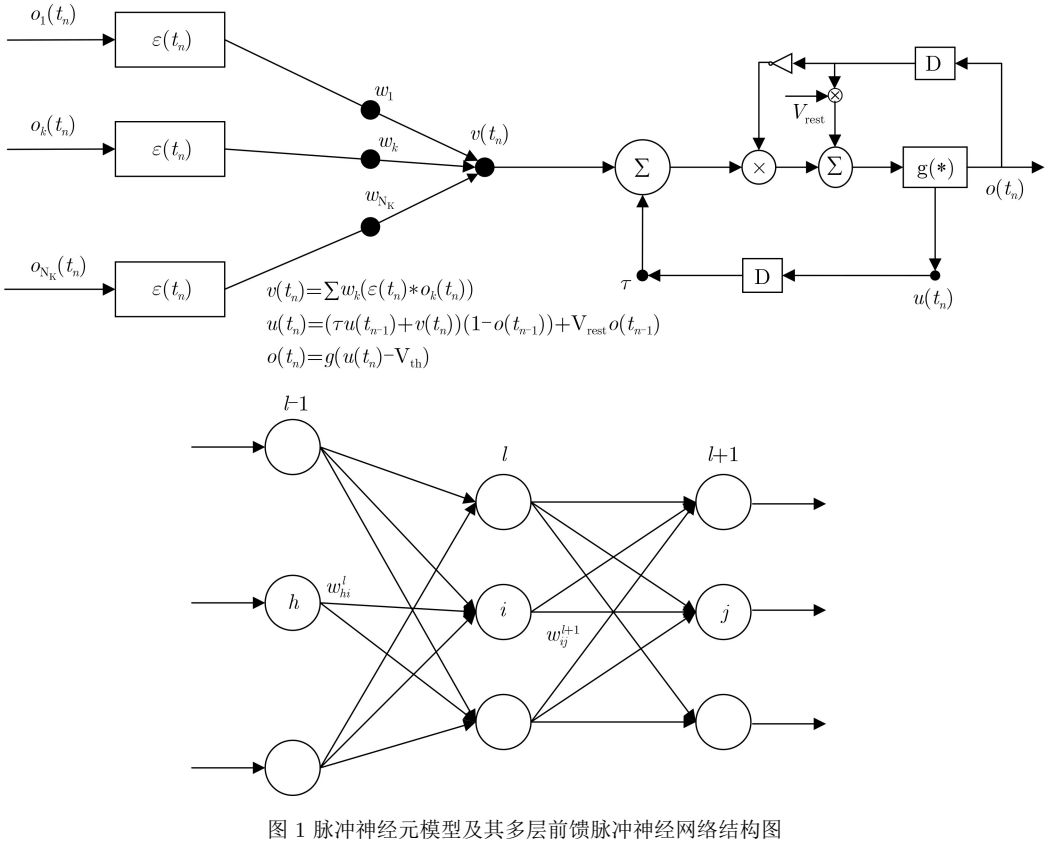

图 1 脉冲神经元模型及其多层前馈脉冲神经网络结构图

其中 $o_h^{l-1}(t_n)=g(u_h^{l-1}(t_n)-V_{\text{th}})$，$o_i^l(t_{n-1})=g(u_i^l(t_{n-1})-V_{\text{th}})$。特别的，对于第1层输入神经元，它的脉冲序列是经过泊松编码产生的一串零幺脉冲序列，是一个有限维的向量，长度与时间窗口大小一致，为T。

不失一般性，设神经网络共有$l+1$层，第$l+1$层中第j个神经元在t_n时刻的目标输出为$d_j^{l+1}(t_n)$。采用监督学习策略，$l+1$层神经元输出$o_j^{l+1}(t_n)$和目标输出$d_j^{l+1}(t_n)$之间的van Rossum距离损失函数为

$$L_j(t_n,s)=\left(\varepsilon(t_n)*d_j^{l+1}(t_n)-\varepsilon(t_n)*o_j^{l+1}(t_n)\right)_s^2 \tag{5}$$

则$l+1$层所有神经元的损失函数之和L为

$$L=\frac{1}{2N_{\text{T}}N_s}\sum_{s=1}^{N_s}\sum_{j=1}^{N_{l+1}}\sum_{n=0}^{N_{\text{T}}}-1L_j(t_n,s) \tag{6}$$

这里N_s为训练样本批处理数量，s表示批处理中的第s个样本。

为推导方便，记$a_i^l(t_n)=\varepsilon(t_n)*o_i^l(t_n)=(\varepsilon*o_i^l)(t_n)$，$b_i^l(t_n)=\varepsilon'(t_n)*o_i^l(t_n)=(\varepsilon'*o_i^l)(t_n)$，$\bar{o}_i^l(t_n)=1-o_i^l(t_n)$。

图2描述了使用2.1节基本神经元模型搭建的SNN，信息在各变量中的流动关系。从中看到，网络中各层级间的空间依赖关系主要发生在PSP，时间依赖关系主要来自膜电位。

2.3 时空反向传播

反向传播的关键，是计算总损失L关于网络内连接权重w_{ij}^k的梯度。由式(5)和式(6)得到

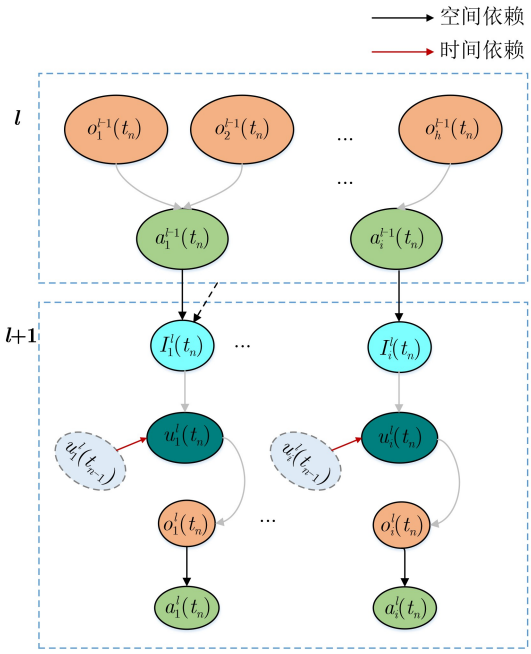

图 2 前馈结构信息流动关系

$$\frac{\partial L}{\partial w_{ij}^k}=\frac{1}{2N_{\text{T}}N_s}\frac{\partial}{\partial w_{ij}^k}\sum_{s=1}^{N_s}\sum_{j=1}^{N_k}\sum_{n=0}^{N_{\text{T}}-1}\left(\varepsilon(t_n)*d_j^k(t_n)-\varepsilon(t_n)*o_j^k(t_n)\right)_s^2 \tag{7}$$

由式(4)得到

$$\frac{\partial o_j^k(t_n)}{\partial w_{ij}^k}=g'(u_j^k(t_n)-V_{th})\frac{\partial u_j^k(t_n)}{\partial w_{ij}^k}$$

由图2可知，非时空边界情形下，$u_j^k(t_n)$和$u_j^k(t_{n+1})$都和w_{ij}^k相关，因此分别定义PSP反传迭代因子和膜电位反传迭代因子来表示空间依赖性和时间依赖性。定义神经元膜电位的反向梯度为

$$\tilde{\delta}_i^k(t_n)\triangleq\frac{\partial L}{\partial u_i^k(t_n)} \tag{8}$$

定义神经元PSP的反向梯度为

$$\hat{\delta}_i^l(t_n)\triangleq\frac{\partial L}{\partial a_i^l(t_n)}=\frac{\partial L}{\partial(\varepsilon*o_i^l)(t_n)} \tag{9}$$

由此得到如下递推关系

$$\tilde{\delta}_i^l(t_n)=\hat{\delta}_i^l(t_n)\left(\varepsilon*g'(u_i^l-V_{th})\right)(t_n)+\tilde{\delta}_i^l(t_{n+1})\tau\bar{o}_i^l(t_n) \tag{10}$$

其中$\hat{\delta}_i^l(t_n)$计算为

$$\begin{aligned}\hat{\delta}_i^k(t_n)=&\sum_{j=1}^{N_{l+1}}\hat{\delta}_j^{k+1}(t_n)w_{ij}^{k+1}\left(1-o_j^{k+1}(t_{n-1})\right)\\&\cdot\left(\varepsilon(t_n)*\frac{\partial g}{\partial u_j^{k+1}(t_n)}\right)+\hat{\delta}_i^k(t_{n+1})\\&\cdot\frac{\left(\varepsilon(t_{n+1})*\dfrac{\partial g}{\partial u_i^k(t_{n+1})}\right)}{\left(\varepsilon(t_n)*\dfrac{\partial g}{\partial u_i^k(t_n)}\right)}\tau\bar{o}_i^k(t_n)\end{aligned} \tag{11}$$

其中，$\dfrac{\partial g}{\partial x}=f(x-V_{\text{th}})$。式(10)和式(11)给出了$\hat{\delta}_i^k(t_n)$和$\tilde{\delta}_i^k(t_n)$的时空反向传播关系。

下面处理边界情形，这里存在3种边界状态：处于输出层和时间终点，处于隐含层和时间终点以及处于输出层和非时间终点。

(1)输出层$l+1$和时间终点$N_{\text{T}}-1$的膜电位的传播误差的边界值

$$\begin{aligned}\tilde{\delta}_j^{l+1}(t_{N_{\text{T}}-1})=&\hat{\delta}_j^{l+1}(t_{N_{\text{T}}}-1)\left(\varepsilon*f(u_j^{l+1}-V_{\text{th}})\right)\\&\cdot(t_{N_{\text{T}}}-1)\end{aligned} \tag{12}$$

其中

$$\begin{aligned}\hat{\delta}_j^{l+1}(t_{N_{\text{T}}}-1)=&-\frac{1}{N_{\text{T}}N_{\text{s}}}\sum_{s=1}^{N_s}\left((\varepsilon(t_{N_{\text{T}}}-1)\right.\\&\left.*d_j^{l+1}(t_{N_{\text{T}}}-1))-a_j^{l+1}(t_{N_{\text{T}}-1})\right)\end{aligned} \tag{13}$$

(2)隐含层l的时间终点$N_{\text{T}}-1$的PSP的传播误差的边界值

$$\tilde{\delta}_i^k(t_{N_\mathrm{T}-1}) = \hat{\delta}_i^k(t_{N_\mathrm{T}-1})\left(\varepsilon * f(u_i^k - V_{th})\right)(t_{N_\mathrm{T}-1}) \tag{14}$$

其中

$$\hat{\delta}_i^k(t_{N_\mathrm{T}-1}) = \sum_{j=1}^{N_{l+1}} \hat{\delta}_j^{k+1}(t_{N_\mathrm{T}-1}) w_{ij}^{k+1} \cdot \left(\varepsilon * f(u_j^{k+1} - V_{th})\right)(t_{N_\mathrm{T}-1}) \bar{o}_j^{k+1}(t_{N_\mathrm{T}-2}) \tag{15}$$

(3)时间在t_n时刻的输出层l+1的膜电位的传播误差的边界值

$$\tilde{\delta}_j^{l+1}(t_n) = \hat{\delta}_j^{l+1}(t_n)\frac{\partial g}{\partial u_j^{l+1}(t_n)} + \tilde{\delta}_j^{l+1}(t_{n+1})\tau\bar{o}_j^{l+1}(t_n) \tag{16}$$

其中

$$\hat{\delta}_j^{l+1}(t_n) = \hat{\delta}_j^{l}(t_{n+1})\tau\bar{o}_j^{l}(t_{n+1})\frac{\left(\varepsilon(t_{n+1}) * \frac{\partial g}{\partial u_j^{l+1}(t_{n+1})}\right)}{\left(\varepsilon(t_n) * \frac{\partial g}{\partial u_j^{l+1}(t_n)}\right)} \tag{17}$$

最终，第l层中第i个神经元与第l+1层第j个神经元之间连接权w_{ij}^{l+1}的梯度为

$$\frac{\partial L}{\partial w_{ij}^{l+1}} = \sum_{n=0}^{N_\mathrm{T}-1} a_i^l(t_n)\tilde{\delta}_j^{l+1}(t_n) \tag{18}$$

最终权重的更新公式为$w_{ij}^{l+1}|_{e=m+1} = w_{ij}^{l+1}|_{e=m} - \eta\frac{\partial L}{\partial w_{ij}^{l+1}|_{e=m}}$，其中$w_{ij}^{l+1}|_{e=m+1}$表示第$m$+1轮训练中第$l$层中第$i$个神经元与第$l$+1层中第$j$个神经元之间连接权，$\eta$为学习率。

至此，推导出了以van Rossum距离为损失函数的基于脉冲序列标识的深度SNN的时空递归反向传播算法。同样，可以把该算法推广到以Hamming距离和时间脉冲序列与其膜电位之间距离为损失函数的情况，这里仅仅是边界条件(1)发生改变，其它计算都不会发生改变。

令t_n时刻的l+1输出层中的第j个神经元输出的时间脉冲序列的实际和期望的差值平方值为

$$L_j(t_n, s) = \left(d_j^{l+1}(t_n) - o_j^{l+1}(t_n)\right)_s^2 \tag{19}$$

则与van Rossum距离不同，基于时间脉冲序列的Hamming距离的梯度下降学习算法中的输出层l+1和时间终点N_T−1的膜电位的传播误差的边界值

$$\tilde{\delta}_j^{l+1}(t_{N_\mathrm{T}-1}) = \hat{\delta}_j^{l+1}(t_{N_\mathrm{T}-1}) f\left(u_j^{l+1}(t_{N_\mathrm{T}-1}) - V_\mathrm{th}\right) \tag{20}$$

其中

$$\hat{\delta}_j^{l+1}(t_{N_\mathrm{T}-1}) = -\frac{1}{N_\mathrm{T}N_s}N\sum_{s=1}^{N_s}\left(d_j^{l+1}(t_n) - o_j^{l+1}(t_n)\right)_s \tag{21}$$

其它边界值的计算方程不发生改变。

依据上述推导两种不同损失函数的替代梯度下降的时空递归误差反向传播算法伪代码如算法1所示。现有研究表明替代函数的选择对结果并没有显著的影响[22,23]，常用的替代函数有矩形窗函数、多项式函数、sigmoid函数和高斯函数等，在本文中使用的是分段分式函数

$$f(u) = (a|u - V_\mathrm{th}| + 1)^{-2} \tag{22}$$

值得强调的是，尽管本文的所有推导都基于全连接网络进行，但本文提出算法同样可应用于卷积神经网络中，本文所做实验也都是基于卷积神经网络完成的。算法对CNN和全连接网络并无本质区别。这是因为在深度学习中，卷积操作是通过卷积核在原始数据(以图像为例)上遍历，对应元素相乘后再求和完成的。当图像的格式为一个3维张量：行(M)*列(N)*通道数(C)时，此时卷积核是一个4维张量：卷积核个数(T)*卷积核尺寸(K)*卷积核尺寸(K)*通道数(C)。每次卷积时，$K * K * C$大小的卷积核与图像上同样大小的空间做乘法并求和，得到一个输出结果，这就得到了一个通道上一个位置的输出结果，接着遍历整张图，得到一个通道的全部输出结果，T个卷积核进行这样的操作就会得到了T个通道的输出结果，由此完成1次卷积操作。通过上述过程的分析，可将卷积操作视为输出神经元为T，输入神经元数量为C之间的“全连接”，而这里的“全连接”其实就是K*K的卷积核在M*N的图像上进行的2维卷积，此时对应的权重w也都是2维的矩阵。因此下面分析一下2维卷积权重在反向传播中如何处理。

令$w_{(x,y)}^{l+1}$表示第l–1与l层连接权重矩阵中第(x,y)位置的值，$o_{(x',y')}(t_n)$表示输出矩阵中第(x',y')位置的像素点t_n时刻的输出脉冲结果，$d_{(x',y')}(t_n)$表示输出矩阵中第(x',y')位置的像素点t_n时刻的目标脉冲结果。此时梯度为：

$$\frac{\partial L}{\partial w_{(x,y)}^{l+1}} = \frac{1}{2N_\mathrm{T}}\frac{\partial}{\partial w_{(x,y)}^{l+1}}\sum_{s=1}^{N_S}\sum_{x'=1}^{M}\sum_{y'=1}^{N}\sum_{n=0}^{N_\mathrm{T}-1}(\varepsilon(t_n) * d_{(x',y')}(t_n) - \varepsilon(t_n) * o_{(x',y')}(t_n))_s^2 \tag{23}$$

可见只是与w^{l+1}相关的神经元由$\sum_{j=1}^{N_K}$变成了$\sum_{s=1}^{N_S}\sum_{x'=1}^{M}$，仅发生求和范围的变化，而对于膜电位反传迭代因子和突触后电位反传迭代因子的

算法 1 伪代码

input : Network inputs X, class label Y, initial weight vector$\{W^l\}_{l=1}^{L}$, membrane decay factor τ_m, synaptic decay factor τ_s, threshold potential V_{th}, simulation windows T

output : Output Spike Sequence O^L

1 Forward Pipeline :

2 $v_{0:T-1}^{1} \leftarrow$ *repeat input X through T*

3 target $\leftarrow$ $encode\_repeat(Y)$

4 for $l \leftarrow 2$ **to** L **do**

5 **for** $t \leftarrow 0$ **to** $T \leftarrow 1$ **do**

6 $u_t^l,\ o_t^l \leftarrow Update\_neuron\_state(w^l, v_t^{l-1}, u_{t-1}^l, \tau_m, V_{th})$ /Eq.(4)-(5)

7 $v_t^l \leftarrow Compute\_psp(o_t^l)$ /Eq.(3a)

8 **end**

9 end

10 $Loss \leftarrow loss\_function(O^L, \text{target})$

11 Backward Pipeline :

12 for $l \leftarrow L$ **to** 1 **do**

13 **for** $t \leftarrow T-1$ **to** 0 **do**

14 **if** $t = T$ and $l = L$ **then**

15 $\hat{\delta}_{T-1}^{L} \leftarrow Initial\_psp\_iteration\_factor\left(\dfrac{\partial L}{\partial a_{T-1}^{L}}\right)$ /Eq.(13)

16 $\tilde{\delta}_{T-1}^{L} \leftarrow Initial\_mem\_iteration\_factor\left(\hat{\delta}_{T-1}^{L}, \dfrac{\partial a_{T-1}^{L}}{\partial u_{T-1}^{L}}\right)$ /Eq.(12)

17 **end**

18 **else if** $t = T-1$ and $l! = L$ **then**

19 $\hat{\delta}_{T-1}^{l} \leftarrow Update\_psp\_iteration\_factor(\hat{\delta}_{T-1}^{l+1}, w^{l+1}, u_{T-1}^{l+1}, o_{T-2}^{l+1})$ /Eq.(15)

20 $\tilde{\delta}_{T-1}^{l} \leftarrow Update\_mem\_iteration\_factor(\hat{\delta}_{T-1}^{l}, u_{T-1}^{l})$ /Eq.(14)

21 **end**

22 **else if** $t! = T-1$ and $l = L$ **then**

23 $\hat{\delta}_{t}^{L} \leftarrow Update\_psp\_iteration\_factor(\hat{\delta}_{T+1}^{L}, o_{T+1}^{L}, u_{T+1}^{L}, u_t^L)$ /Eq.(17)

24 $\tilde{\delta}_{t}^{L} \leftarrow Update\_mem\_iteration\_factor(\hat{\delta}_{t}^{L}, u_t^L, \tilde{\delta}_{T+1}^{L}, o_t^L)$ /Eq.(16)

25 **end**

26 **else**

27 $\hat{\delta}_{t}^{l} \leftarrow Update\_psp\_iteration\_factor(\hat{\delta}_{t}^{l+1}, o_{T-1}^{l+1}, u_t^{l+1}, \hat{\delta}_{T+1}^{l}, u_t^l, u_{T+1}^l, o_t^l)$ /Eq.(11)

28 $\tilde{\delta}_{t}^{l} \leftarrow Update\_mem\_iteration\_factor(\hat{\delta}_{t}^{l}, u_t^l, \tilde{\delta}_{T+1}^{l}, o_t^l)$ /Eq.(10)

29 **end**

30 **end**

31 end

32 *Update* W based on: $\Delta W^l = \sum_{T=0}^{T-1} a_t^{l-1} \tilde{\delta}_t^l$ /Eq.(18)

计算，则同样只需将覆盖的神经元范围有$\sum_{j=1}^{N_K}$变成$\sum_{s=1}^{N_S}\sum_{x'=1}^{M}$即可，这里的$(x', y')$代表前一层与之相连的x′, y′位置处的神经元。由此可见算法对CNN和全连接网络并无本质区别，公式推导只需更改为标量形式即可，本文选择以全连接网络为例，通过标量方式推导，是为了更方便理解。

3 数据与测试结果

为验证文中所提出的算法性能，本文使用公开静态图像数据集CIFAR10和神经形态数据集NMNIST在NVIDIA RTX 3090 GPU上进行了新算法

的有效性验证和性能评价。对于静态图像在处理时，直接将像素值作为网络输入而没有进行编码处理，对于神经形态数据集，使用时间驱动的脉冲串作为网络输入。本文所提出的新算法在多个数据集和不同网络结构下都取得了良好的性能，表明了算法的有效性和泛化能力。在以下的测试中，没有采用批归一化处理，初始学习率为0.000 3，τ_m=5，τ_s=5，迭代步长采用AdamW优化器。

CIFAR10是一个彩色图像分类数据集，共有50000张训练图像和10 000张测试图像，每幅图像大小为32×32。本文在AlexNet网络上对算法性能进行简单验证。在CIFAR10公开数据集上的基于van Rossum距离和Hamming距离算法的结果如表1和3所示。使用Accuracy作为评价指标时，本文提出的两种算法在CIFAR10上分别取得了88.47%和89.10%的结果，这个结果超越STBP算法的同类研究，证明了本文算法具备处理较为复杂数据集的能力。此外，本文算法性能优于基于转换的方法和ANN，这意味着直接使用时间脉冲时序训练SNN的方式有着很好的发展前景。

NMINST是将静态图像数据集MNIST通过扫描得到的脉冲神经形态数据集，每个图像的空间大小为34×34×2，持续时间为300 ms，所以网络的输入不是模拟的像素值而是时间脉冲序列，富含更多的时间信息，也更适合检验SNN学习算法的性能。在NMNIST数据集上基于van Rossum距离和Hamming距离算法的测试结果如表2和4所示。本文算法在NMNIST上超越了同类研究，大大减少了计算的时间复杂度，节省了计算资源，同时也证明本文算法在神经形态数据集上依然具备较好的性能。

本文算法使用van Rossum和Hamming距离作为损失函数，在静态和神经形态数据集上都获得了优于STBP的结果。值得一提的是本文算法在实验中使用更少的时间步长，减少了推理时间，节省了训练资源的同时，降低了对训练平台的要求。此外在CIFAR10数据集上，本文算法的结果也优于基于转换方法的SNN，这表明作为时间序列编码的学习算法相较基于脉冲频率编码的方法具有更好的潜力。

通过观察发现，对于CIFAR10和NMNIST数据集，基于Hamming距离的实验结果，在同样的网络结构和神经元参数设置的前提下，均优于基于van Rossum距离的实验结果。这可能是由于Hamming距离严格反映输出脉冲序列和目标脉冲序列的时间结构差异，而van Rossum距离中通过对脉冲串求卷积即计算两PSP的方式，虽然在脉冲少时保证了网络的训练，增加了容错性，但当放电比较充分时也带来了损失计算的误差，使得学习效果较基于Hamming距离的方法略有下降。

4　讨论

与基于尖峰放电速率标识的时空反向传播算法[12]不同，本文提出的时间脉冲序列标识的时空递归误差反向传播算法，能够充分发挥SNN的时间动态特点，保留时间结构信息，有利于SNN在控制领域的应用。同时，与时间脉冲序列学习的反向传播算法[19]相比，本文算法无需分析神经元内部时间依赖关系所使用的复杂分段枚举方法，表达形式更加简单直观、容易理解，也不存在长时程标识中分段枚举带来的复杂性，避免了求导依赖放电时间导致的当神经元本该发放脉冲却未发放时带来的该时刻误差无法回传的问题。

本文分别在公开数据集上进行了验证实验，在相同网络结构下，本文算法使用更短的时间步长取得了与同类研究[12]可比或超越的性能，在CIFAR-10和NMNIST上可达到89.10%、98.67%的准确率。值得指出的是，在CIFAR-10上的结果超越了基于ANN转换的结果，这表明使用脉冲时间序列标识的学习算法具有更好的学习潜力，这对加速脉冲序列指导

表 1　CIFAR10的测试结果(van Rossum距离)

方法	网络	时间步长	周期	准确率(%)
Converted SNN[24]	AlexNet①	80		83.52
STBP[12]	AlexNet	8	150	85.24
本文算法	AlexNet	5	100	**88.47**

①：96C3-256C3-P2-384C3-P2-384C3-256C3-1024-1024

表 2　NMNIST的测试结果(van Rossum距离)

方法	网络	时间步长	周期	准确率(%)
STBP[12]	12C5-P2-64C5-P2	50	150	98.19
本文算法	12C5-P2-64C5-P2	25	100	**98.61**

表 3　CIFAR10的测试结果(Hamming距离)

方法	网络	时间步长	周期	准确率(%)
Converted SNN[24]	AlexNet①	80		83.52
STBP[12]	AlexNet	8	150	85.24
本文算法	AlexNet	10	100	**89.10**

①：96C3-256C3-P2-384C3-P2-384C3-256C3-1024-1024

表 4　NMNIST的测试结果(Hamming距离)

方法	网络	时间步长	周期	准确率(%)
STBP[12]	12C5-P2-64C5-P2	50	150	98.19
本文算法	12C5-P2-64C5-P2	25	100	**98.67**

SNN直接监督学习算法的应用和发展带来了信心和支持。同时本文也对比了提出算法基于Hamming距离和van Rossum距离的结果，发现基于Hamming距离的方法略优于van Rossum距离的结果，这可能是由于van Rossum距离在增强训练容错性的同时，也引入了部分误差，使得结果略低于精确衡量输出和目标差距的Hamming距离方法。但本文认为van Rossum会在放电较少的情况下，避免了损失的计算值很小从而导致梯度消失的问题，保证了网络的训练。因此，当在选择合适的卷积宽度时控制误差量，van Rossum距离会得到较好的学习效果。

5 结论

SNN作为一种受生物启发的神经计算模型，具有更好的生物可解释性和时间动态特性，被认为是实现类脑计算的重要方式之一。但脉冲神经元脉冲放电的形式在带来网络稀疏、低功耗优势的同时，其脉冲生成函数不可微分的问题也给SNN的学习带来了巨大困难。现有SNN学习算法可大致分为基于ANN转换的间接学习方法和受BP启发的直接监督学习算法两大类。其中基于转换的方法，得益于ANN良好的性能，一般体现出较好的准确性，但这类方法存在的问题是需要较长的推理时间来保证转换精度不下降，直接增大了计算需求和能耗，使得与使用SNN的初衷背道而驰。受BP启发的SNN监督学习算法根据训练标识的不同，也可分为基于放电频率和基于脉冲时间序列两种。其中基于放电频率的方法与基于转换的间接学习存在同样的问题，很依赖于对长推理时间的需求，同时对时间维度的压缩也使得网络失去了学习时间动态变化的能力。而基于脉冲时间序列指导的学习算法，可以充分考虑脉冲神经元的时间动态特性，学习脉冲序列的时间结构，适合应用于行为控制等领域。但同时计算损失在时间和空间的传播，也大大增加了此类算法的推导和训练难度。

本文提出一种基于脉冲时间序列标识的新型时空递归误差反向传播算法，不同于现有的基于放电脉冲速率标识的时空反向传播算法，它通过计算两脉冲串之间的Hamming距离或van Rossum距离来评估学习损失，分析脉冲神经元的膜电位和PSP上的空间和时间依赖性，完成误差的反向传播，引入替代梯度在解决脉冲生成函数不可微分问题的同时，也避免了长时程脉冲序列标识计算复杂和当脉冲神经元不发放时误差无法反向传播的问题。

本文算法分别在静态和神经形态数据集上进行了验证，均表现出良好的性能，同时也缩短了对时间步长的需求，大大地降低了计算量和能耗。从应用角度出发，对于一些分类等基础的图像处理任务，使用脉冲频率标识的学习算法，同样可以获得不错的效果。这是由于此类任务往往只需给出一个像素强度或类别概率的模拟值就可以达到目的，需要脉冲表达的信息是简单的，处理成放电频率后就可以代表。但对于行为控制这类复杂问题来说，每个行为都需要唯一的编码，此时简单的放电频率就不足以表示多样的行为信息，而使用脉冲序列标识的方法，依赖少量的神经元就可以传递出多样复杂的指令，恰恰体现出SNN的优势。因此作为脉冲序列指导的学习算法在将来也会有更重要且广阔的应用前景，而本文算法也为此类算法的研究发展助力。

致谢 感谢科技创新2030—“新一代人工智能”重大项目(2020AAA0105800)对该研究工作的支持，感谢清华大学张颢老师对论文提出的宝贵意见。

参 考 文 献

[1] YAMAZAKI K, VO-HO V K, BULSARA D, *et al*. Spiking neural networks and their applications: A review[J]. *Brain Sciences*, 2022, 12(7): 863. doi: 10.3390/brainsci12070863.

[2] MAKAROV V A, LOBOV S A, SHCHANIKOV S, *et al*. Toward reflective spiking neural networks exploiting memristive devices[J]. *Frontiers in Computational Neuroscience*, 2022, 16: 859874. doi: 10.3389/fncom.2022.859874.

[3] BÜCHEL J, ZENDRIKOV D, SOLINAS S, *et al*. Supervised training of spiking neural networks for robust deployment on mixed-signal neuromorphic processors[J]. *Scientific Reports*, 2021, 11(1): 23376. doi: 10.1038/s41598-021-02779-x.

[4] ASGHAR M S, ARSLAN S, and KIM H. A low-power spiking neural network chip based on a compact LIF neuron and binary exponential charge injector synapse circuits[J]. *Sensors*, 2021, 21(13): 4462. doi: 10.3390/s21134462.

[5] ROY K, JAISWAL A, and PANDA P. Towards spike-based machine intelligence with neuromorphic computing[J]. *Nature*, 2019, 575(7784): 607–617. doi: 10.1038/s41586-019-1677-2.

[6] HODGKIN A L and HUXLEY A F. A quantitative description of membrane current and its application to conduction and excitation in nerve[J]. *The Journal of Physiology*, 1952, 117(4): 500–544. doi: 10.1113/jphysiol.1952.sp004764.

[7] ZHAN Qiugang, LIU Guisong, XIE Xiurui, *et al*. Effective transfer learning algorithm in spiking neural networks[J]. *IEEE Transactions on Cybernetics*, 2022, 52(12): 13323–13335. doi: 10.1109/TCYB.2021.3079097.

[8] LUO Xiaoling, QU Hong, WANG Yuchen, *et al*. Supervised learning in multilayer spiking neural networks with spike

temporal error backpropagation[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2023, 34(12): 10141–10153. doi: 10.1109/TNNLS.2022.3164930.

[9] JIANG Runhao, ZHANG Jie, YAN Rui, *et al*. Few-shot learning in spiking neural networks by multi-timescale optimization[J]. *Neural Computation*, 2021, 33(9): 2439–2472. doi: 10.1162/neco_a_01423.

[10] XIE Xiurui, YU Bei, LIU Guisong, *et al*. Effective active learning method for spiking neural networks[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2024: 1–10. doi: 10.1109/TNNLS.2023.3257333.

[11] SENGUPTA A, YE Yuting, WANG R, *et al*. Going deeper in spiking neural networks: VGG and residual architectures[J]. *Frontiers in Neuroscience*, 2019, 13: 95. doi: 10.3389/fnins.2019.00095.

[12] WU Yujie, DENG Lei, LI Guoji, *et al*. Spatio-temporal backpropagation for training high-performance spiking neural networks[J]. *Frontiers in Neuroscience*, 2018, 12: 331. doi: 10.3389/fnins.2018.00331.

[13] WU Yujie, DENG Lei, LI Guoji *et al*. Direct training for spiking neural networks: Faster, larger, better[C]. The 33th AAAI Conference on Artificial Intelligence, California, USA, 2019: 1311–1318. doi: 10.1609/aaai.v33i01.33011311.

[14] VAN ROSSUM M C. A novel spike distance[J]. *Neural Computation*, 2001, 13(4): 751–763. doi: 10.1162/089976601300014321.

[15] BOHTE S M, KOK J N, and LA POUTRÉ H. Error-backpropagation in temporally encoded networks of spiking neurons[J]. *Neurocomputing*, 2002, 48(1/4): 17–37. doi: 10.1016/S0925-2312(1)00658-0.

[16] XIAO Mingqing, MENG Qingyan, ZHANG Zongpeng, *et al*. SPIDE: A purely spike-based method for training feedback spiking neural networks[J]. *Neural Networks*, 2023, 161: 9–24. doi: 10.1016/j.neunet.2023.01.026.

[17] GUO Yufei, HUANG Xuhui, and MA Zhe. Direct learning-based deep spiking neural networks: A review[J]. *Frontiers in Neuroscience*, 2023, 17: 1209795. doi: 10.3389/fnins.2023.1209795.

[18] ZENKE F and GANGULI S. SuperSpike: Supervised learning in multilayer spiking neural networks[J]. *Neural Computation*, 2018, 30(6): 1514–1541. doi: 10.1162/neco_a_01086.

[19] ZHANG Wenrui and LI Peng. Temporal spike sequence learning via backpropagation for deep spiking neural networks[C]. 34th International Conference on Neural Information Processing Systems, Vancouver, Canada, 2020: 1008.

[20] KRIZHEVSKY A, NAIR V, and HINTON G. The CIFAR-10 dataset[EB/OL]. https://www.cs.toronto.edu/~kriz/cifar.html, 2009.

[21] ORCHARD G, JAYAWANT A, COHEN G K, *et al*. Converting static image datasets to spiking neuromorphic datasets using saccades[J]. *Frontiers in Neuroscience*, 2015, 9: 437. doi: 10.3389/fnins.2015.00437.

[22] ZENKE F and VOGELS T P. The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks[J]. *Neural Computation*, 2021, 33(4): 899–925. doi: 10.1162/neco_a_01367.

[23] TAVANAEI A, GHODRATI M, KHERADPISHEH S R, *et al*. Deep learning in spiking neural networks[J]. *Neural Networks*, 2019, 111: 47–63. doi: 10.1016/j.neunet.2018.12.002.

[24] HUNSBERGER E and ELIASMITH C. Training spiking deep networks for neuromorphic hardware[EB/OL]. https://arxiv.org/abs/1611.05141, 2016.

王子华：男，博士生，研究方向为计算机视觉在医学图像中的应用.
叶　莹：女，硕士生，研究方向为脉冲神经网络算法.
许　燕：女，教授，研究方向为医学人工智能.
王卫东：男，研究员，研究方向为医学影像物理与工程、信号与信息处理、生物计算理论、脉冲神经网络算法等.

责任编辑：马秀强