

脉冲神经网络:模型、学习算法与应用

程 龙[†], 刘 洋

- (1. 中国科学院自动化研究所 复杂系统管理与控制国家重点实验室, 北京 100190;
2. 中国科学院大学, 北京 100049)

摘 要: 脉冲神经网络是目前最具有生物解释性的人工神经网络,是类脑智能领域的核心组成部分. 首先介绍各类常用的脉冲神经元模型以及前馈和循环型脉冲神经网络结构;然后介绍脉冲神经网络的时间编码方式,在此基础上,系统地介绍脉冲神经网络的学习算法,包括无监督学习和监督学习算法,其中监督学习算法按照梯度下降算法、结合STDP规则的算法和基于脉冲序列卷积核的算法3大类别分别展开详细介绍和总结;接着列举脉冲神经网络在控制领域、模式识别领域和类脑智能研究领域的应用,并在此基础上介绍各国脑计划中,脉冲神经网络与神经形态处理器相结合的案例;最后分析脉冲神经网络目前所存在的困难和挑战.

关键词: 脉冲神经网络; 脉冲神经元模型; 学习算法; STDP规则; 类脑智能; 神经形态处理器

中图分类号: TP273

文献标志码: A

Spiking neural networks: Model, learning algorithms and applications

CHENG Long[†], LIU Yang

- (1. State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China; 2. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Spiking neural network is the most biologically plausible neural network model, which is the core component of brain-inspired intelligence. Firstly, the most commonly used spiking neuron models and the structures of feedforward and recurrent spiking neural networks are introduced, and the temporal coding method of spiking neural networks is introduced. Then, the learning algorithms of spiking neural networks systematically, including the unsupervised learning and supervised learning algorithms are introduced. The supervised learning algorithm follows the gradient descent algorithm, the algorithm based on STDP rules and the algorithm based on spike trains convolution rules are introduced and summarized in details. Subsequently, this survey gives application examples of spiking neural networks in the fields of control, pattern recognition and brain-inspired intelligence. Some case studies of spiking neural networks and neuromorphic processor in various national brain initiatives are presented. Finally, this survey discusses the current difficulties and challenges of spiking neural networks.

Keywords: spiking neural network; spiking neuron model; learning algorithm; STDP rule; brain-inspired intelligence; neuromorphic processor

0 引 言

人工神经网络是对生物神经网络的模拟,传统的人工神经网络,其输入和输出均为模拟量,这些模拟量从生物学角度可以解释为:在一定时间内神经元释放脉冲的频率,即脉冲频率编码^[1]. 然而,研究表明:生物能够快速响应外界的刺激是因为生物神经网络的信息传递依赖于具体的脉冲时刻^[2]. 相比于缓慢的频率编码,脉冲时间是更为高效的编码方式^[3]. 在此背景下,Maass^[4-5]提出了基于脉冲神经元的新型神

经网络,即脉冲神经网络. 与基于频率编码的传统神经网络相比,脉冲神经网络拥有更强大的计算能力,可以任意逼近非线性函数和模拟生物神经网络信号,是处理复杂时空信息的有效工具.

近年来,深度学习的兴起引发了新一轮人工智能的研究热潮,然而,由于深度学习自身的局限性,研究人员发现深度学习难以实现真正的强人工智能. 作为自然界最优秀的智能体,生物大脑是进行人工智能研究最重要的参考. 因此,研究大脑机制,开发受脑启

收稿日期: 2017-10-30; 修回日期: 2017-12-21.

基金项目: 国家自然科学基金项目(61422310, 61633016, 61370032); 北京市自然科学基金项目(4162066).

作者简介: 程龙(1982—), 男, 研究员, 博士, 从事智能控制与机器人等研究; 刘洋(1993—), 男, 硕士生, 从事计算智能的研究.

[†]通讯作者. E-mail: long.cheng@ia.ac.cn

发的人工智能技术,是未来智能化时代的核心领域,是新一轮科技革命的重心.在此背景下,美国于2013年首先启动脑科学计划(BRAIN initiative),随后,欧盟和日本先后启动了欧洲脑计划(The human brain project)和日本脑计划(Brain/Minds project).为紧紧把握新一轮科技革命的浪潮,我国于2016年提出了中国脑计划,即“脑科学与类脑科学研究”.该计划主要有两个研究方向:以探索大脑秘密、攻克大脑疾病为导向的脑科学研究以及以建立和发展人工智能技术为导向的类脑研究.其中类脑科学的研究,是通过类人脑神经网络模型和计算方法的建立以及通过类脑计算、处理以及存储设备技术的研究,开发新一代人工智能机器以及类脑机器人等.脉冲神经网络是最具有生物意义、运行机制最类似大脑的神经网络模型.在类脑科学研究中,脉冲神经网络占据核心地位,其低功耗、高性能的特点也是实现人工智能技术的新突破点.随着脑科学计划的进行,脉冲神经网络日益成为研究的焦点.

随着脉冲神经网络的发展,基于脉冲神经网络的学习算法也随之出现.最早出现的是非监督学习算法,通常是基于Hebb规则的学习算法.随后,更具有生物学特性的STDP(Spike-timing-dependent-plasticity)学习规则成为研究热点^[6].传统人工神经网络的监督学习算法,是指根据神经网络的实际输出值与目标值相比较,按照得到的误差调整神经网络的权重.尽管有相关实验表明,在生物神经网络,尤其是运动感知网络中存在着监督学习策略^[7],但是生物神经网络如何实现这一机制仍不明确.脉冲神经网络是目前最接近生物神经网络的模型,其传递的信息是以脉冲时间序列表示的,神经元内部的状态和误差函数不满足连续可微的条件.因此,传统神经网络的监督学习算法,如BP算法等,无法直接使用.这一问题吸引了大量研究者的关注,并提出了诸多面向脉冲神经网络的监督学习算法^[8].根据权重调整规则的不同,大致可分为梯度下降学习算法、监督STDP学习算法和基于脉冲序列卷积的学习算法.

近年来,由于脉冲神经元本身强大的计算能力和脉冲神经网络学习算法的不断完善,脉冲神经网络也越来越多地应用在实际工程中.根据应用方式不同,大致可分为如下3类:

1) 结合无监督学习策略,训练脉冲神经网络无监督的提取数据特征,感知外界环境刺激或者进行聚类分析^[9].

2) 与传统神经网络类似,使用监督学习方法对数

据进行分类或进行时间序列预测^[10].

3) 相比传统神经元模型,脉冲神经元模型具有更低的功耗,更加适合并行计算.因此设计脉冲神经网络芯片可加速计算,最典型的代表是IBM公司开发的TrueNorth芯片^[11].

本文首先介绍几类重要的脉冲神经元模型;然后给出脉冲神经元的编码方式和网络结构,并重点介绍脉冲神经网络的监督学习算法,进而介绍脉冲神经网络的最新应用和研究情况;最后对脉冲神经网络的研究热点及发展趋势进行预测.

1 神经元模型

在脉冲神经网络中,神经元的状态由膜电势和激活阈值决定.神经元的膜电势由来自上一层神经元的突触后电势决定,突触后电势分为激励型和抑制型,分别记为EPSP和IPSP,其中,EPSP会提高神经元的膜电势,IPSP会降低神经元的膜电势.当神经元的膜电势升高到激活阈值时,神经元会产生一个脉冲(spike),这个脉冲通过神经元的轴突传递到下一神经元中.脉冲沿着突触传递的过程需要一定的时间,这个时间被称为突触延迟^[12],脉冲神经元激活过程如图1所示.

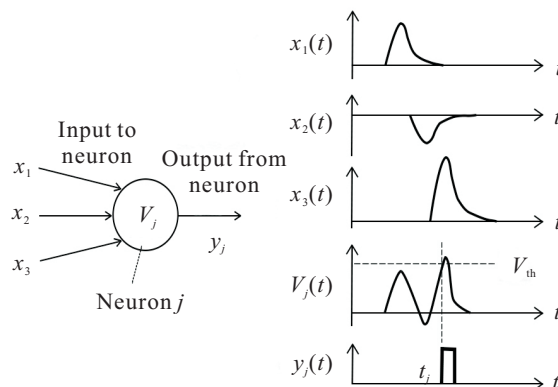


图1 脉冲神经元激活过程

脉冲神经元从模型机理上可分为4类,分别是基于电流输入输出的膜电势模型、自然信息输入模型、药理刺激模型和层级即时记忆模型.目前关于脉冲神经元模型的研究已经非常详尽,本节主要介绍最具代表性的4类神经元模型,分别是Hodgkin-Huxley模型、积分点火模型(IAF)、脉冲响应模型(SRM)和Izhikevich模型.各类脉冲神经元如表1所示.

1.1 Hodgkin-Huxley模型

1952年,Hodgkin等^[14]利用电压钳测得乌贼轴突的电位数据,并推导得出了一组非线性微分方程,提出了Hodgkin-Huxley模型.该模型可以准确详细地描述各类离子通道动态变化引起的神经元膜电

表 1 脉冲神经元模型

类型	名称
	IAF 模型 ^[13]
	Hodgkin-Huxley 模型 ^[14]
	LIF 模型 ^[12]
	SRM 模型 ^[12]
	分数阶 LIF 模型 ^[15]
	Galves-Löcherbach 模型 ^[16]
膜电势模型	指数 IAF 模型 ^[17]
	FizHugh-Nagumo 模型 ^[18]
	Morris-Lecar 模型 ^[19]
	Hindmarsh-Rose 模型 ^[20]
	Compartmental 模型 ^[21]
	Thorpe 模型 ^[22]
	Izhikevich 模型 ^[23]
	非齐次泊松过程模型 ^[24]
自然信息输入模型	两状态马尔可夫模型 ^[25]
	非马尔可夫模型 ^[26]
药理刺激模型	突触传递模型 ^[27]
层级即时记忆模型	HTM 模型 ^[28]

势变化, 量化描述神经元膜电压和电流的变化过程. Hodgkin-Huxley 模型的具体形式由以下微分方程组表示:

$$\begin{cases} I_{\text{ext}} = C_m \frac{dV_m}{dt} + I_{Na} + I_K + I_L, \\ I_{Na} = \bar{g}_{Na} m^3 h (V_m - V_{Na}), \\ I_K = \bar{g}_K n^4 (V_m - V_K), \\ I_L = \bar{g}_L (V_m - V_L). \end{cases} \quad (1)$$

其中: V_{Na} 、 V_K 和 V_L 分别为钠离子通道电流、钾离子通道电流和漏电流的平衡电位; \bar{g}_{Na} 、 \bar{g}_K 和 \bar{g}_L 分别为各电流对应的最大电导^[29]; m 、 n 、 h 分别为钠离子通道激活变量、钾离子通道激活变量和钠离子通道抑制变量, 具体形式为

$$\begin{cases} \frac{dm}{dt} = \alpha_m(V_m)(1 - m) - \beta_m(V_m)m, \\ \frac{dn}{dt} = \alpha_n(V_m)(1 - n) - \beta_n(V_m)n, \\ \frac{dh}{dt} = \alpha_h(V_m)(1 - h) - \beta_h(V_m)h; \end{cases} \quad (2)$$

$\alpha_*(V_m)$ 和 $\beta_*(V_m)$ 分别为各离子通道打开和关闭的转换率, 与膜电势有关, 具体形式为

$$\begin{cases} \alpha_m(V_m) = \frac{0.1(25 - V_m)}{\exp((25 - V_m)/10) - 1}, \\ \alpha_n(V_m) = \frac{0.01(10 - V_m)}{\exp((10 - V_m)/10) - 1}, \\ \alpha_h(V_m) = 0.07 \exp(-V_m/20), \\ \beta_m(V_m) = 4 \exp(-V_m/18), \\ \beta_n(V_m) = 0.125 \exp(-V_m/80), \\ \beta_h(V_m) = \frac{1}{\exp((80 - V_m)/10) + 1}. \end{cases} \quad (3)$$

在经典的 Hodgkin-Huxley 模型中, 各参数值如表 2 所示.

表 2 Hodgkin-Huxley 模型

参数	参数值	单位
C_m	1.0	$\mu\text{F}/\text{cm}^2$
\bar{g}_{Na}	120	ms/cm^2
\bar{g}_K	36	ms/cm^2
\bar{g}_L	0.3	ms/cm^2
V_{Na}	115	mV
V_K	-12	mV
V_L	10.6	mV

1.2 积分点火模型

积分点火模型是最早的脉冲神经元模型, 其本质是将神经元抽象为一个 RC 电路. 神经元接收到输入电流后, 膜电势会升高, 直至达到激活阈值, 释放脉冲, 这个过程称为“积分点火”. 脉冲释放后, 神经元膜电势立刻恢复至静息电位^[13]. 膜电势变化的动态过程可由如下微分方程表示:

$$I(t) = C_m \frac{dV_m(t)}{dt}. \quad (4)$$

其中: V_m 为神经元的膜电势, C_m 为神经元膜电容, $I(t)$ 为外界输入电流.

积分点火神经元的缺陷在于其具有记忆效应, 即无输入时, 神经元膜电势会一直维持在当前状态, 这与真实的神经元相差甚远, 因此, 引入漏电流项, 使得神经元在无输入时, 膜电势能够恢复至静息电位, 这样能够较好地反映出真实生物神经元中的离子扩散效应^[30]. 此时神经元修正为

$$I(t) - \frac{V_m}{R_m} = C_m \frac{dV_m(t)}{dt}. \quad (5)$$

在积分点火模型中, 外部输入电流 $I(t)$ 通常来自突触前神经元的脉冲信号, 因此, 神经元 i 收到来自第 j 个神经元的脉冲电流可以表示为

$$I(t) = \sum_j w_{ij} \sum_f \alpha(t - t_j^{(f)}). \quad (6)$$

其中: w_{ij} 为突触的权重值; $t_j^{(f)}$ 为神经元 j 发出第 f

个脉冲的时间; $\alpha(*)$ 没有固定的形式, 最简单的形式是用Dirac函数表示^[31].

1.3 脉冲响应模型

脉冲响应模型(Spike response model)是积分点火模型的推广. 该模型中, 神经元的状态仅由膜电势 V_m 描述, 并且运用了3个不同的核函数来表示外界输入和自身激活状态对膜电势的影响. 当膜电势 V_m 升高到激活阈值 ϑ , 即 $V_m(t) = \vartheta$, $dV_m(t)/dt > 0$ 时, 神经元发出脉冲. 令 $V_m(t)$ 表示神经元 i 的膜电势, 用 \hat{t}_i 表示该神经元上一次激活的时刻, 则神经元膜电势在 t 时刻的状态为

$$V_m(t) = \eta(t - \hat{t}_i) + \sum_j w_{ij} \sum_f \epsilon_{ij}(t - \hat{t}_i, t - t_j^{(f)}) + \int_0^\infty \kappa(t - \hat{y}_i, s) I_{\text{ext}}(t - s) ds. \quad (7)$$

其中: $t_j^{(f)}$ 为突触前神经元 j 第 f 次发出脉冲的时刻, w_{ij} 为突触连接权重. 核函数 $\eta(*)$ 为神经元激活后的不响应期, 描述了神经元发出脉冲后到恢复至静息电位的动态过程; 核函数 $\epsilon(*)$ 为神经元接收到的脉冲信号对膜电势的影响; 核函数 $\Theta(*)$ 表示外部输入电流 I_{ext} 对膜电势的影响^[32]. 式(7)中3个核函数形式分别为

$$\eta(t - \hat{t}_i) = -\eta_0 \exp\left(-\frac{t - \hat{t}_i}{\tau_{\text{refr}}}\right) \Theta(t - \hat{t}_i), \quad (8)$$

$$\epsilon(t - \hat{t}_i, t - t_j^{(f)}) = \left(\exp\left(-\frac{t - t_j^{(f)}}{\tau_m}\right) - \exp\left(-\frac{t - \hat{t}_i}{\tau_{\text{rec}}}\right)\right) \Theta(t - t_j^{(f)}), \quad (9)$$

$$\kappa(t - \hat{t}_i, s) = \frac{R}{\tau_m} \left[1 - \exp\left(-\frac{t - \hat{t}_i}{\tau_{\text{rec}}}\right)\right] - \exp\left(-\frac{s}{\tau_m}\right) \Theta(s) \Theta(t - \hat{t}_i - s). \quad (10)$$

其中: τ_m 为神经元膜电势常数, τ_{refr} 为不响应期时间常数, τ_{rec} 为响应恢复时间常数, $\Theta(*)$ 为阶跃函数.

经典的脉冲响应模型比较复杂, 若假设外界输入电流 I_{ext} 非常弱, 膜电势的升高主要由突触前神经元发出的脉冲信号引起, 则可以忽略 $\kappa(*)$ 项^[33]. 假设输入到神经元 i 的脉冲信号对神经元膜电势的影响与该神经元上一次发出脉冲的时间无关, 则可以忽略掉 $\epsilon(t - \hat{t}_i, t - t_j^{(f)})$ 中的 $t - \hat{t}_i$ 项. 简化的脉冲响应模型可以记为

$$V_m(t) = \eta(t - \hat{t}_i) + \sum_j w_{ij} \sum_f \epsilon(t - t_j^{(f)}). \quad (11)$$

1.4 Izhikevich模型

Izhikevich 简化了 Hodgkin-Huxley 模型, 基于动态系统理论, 利用如下两个微分方程描述了神经元的动态过程:

$$\begin{cases} \frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I, \\ \frac{du}{dt} = a(bv - u). \end{cases} \quad (12)$$

其中: v 为神经元的膜电势, u 为神经元膜电势的恢复变量. 从生物学角度讲, u 代表了钾离子通道激活变量和钠离子通道抑制变量, 并且为膜电势 v 提供了负反馈^[23].

在 Izhikevich 模型中, 神经元激活阈值为 $\vartheta = 30 \text{ mV}$, 若膜电势 v 超过激活阈值 ϑ , 则神经元发出脉冲, 并且 v 和 u 重置为

$$\begin{cases} v \leftarrow c, \\ u \leftarrow u + d. \end{cases} \quad (13)$$

式(12)和(13)中: 参数 a 表示恢复变量 u 的时间度量, a 越小, 膜电势的恢复速度越慢; 参数 b 表示 u 对膜电势 v 波动的敏感程度; 参数 c 表示发出脉冲后膜电势的重置值, 与快速高阈值钾离子电导(Fast high-threshold K^+ conductances)有关; 参数 d 表示发出脉冲后恢复变量 v 的重置值, 受慢高阈值钠离子和钾离子电导控制(Slow high-threshold Na^+ and K^+ conductances)^[34].

2 脉冲编码与网络结构

2.1 脉冲编码

脉冲神经元接收的信息是脉冲时间序列, 因此原始的模拟数据需要编码为脉冲时间序列才能输入到脉冲神经网络中.

2.1.1 一维脉冲编码

实验表明, 在生物视觉神经网络中, 神经元对信息的编码与首次脉冲发生的时刻有关^[35], 因此引入一维编码方式^[36]: 若实数 $x \in [\min, \max]$, 编码后时间序列 $T \in [t_1, t_2]$, 则编码规则为

$$T(x) = \frac{t_2 - t_1}{\max - \min} \cdot x + \frac{t_1 \cdot \max - t_2 \cdot \min}{\max - \min}. \quad (14)$$

2.1.2 稀疏脉冲编码

稀疏编码(Sparse coding)是另一类常用的编码方式, 其编码来源于生物神经元的感受野(Receptive field)^[37-41]. 一个实数值通过一组互相重叠的感受野函数, 函数的输出值代表这组神经元的激活度, 函数值越大, 激活时间越早, 所编码的时间值越小. 通常使用高斯函数代表感受野函数, 设有 k 个神经元进行编

码,则 k 个高斯函数的中心和宽度为

$$C_i = \min + \frac{\max - \min}{k - 2} \cdot i, \quad (15)$$

$$\sigma = \beta \cdot \frac{\max - \min}{k - 2}. \quad (16)$$

其中: 待编码实数值 $x \in [\min, \max]$, β 的经验取值为 1.5. 这种编码方式又称为群体编码 (Population coding), 即生物神经网络的某些部分将外界刺激通过一组神经元编码^[42], 如图2所示.

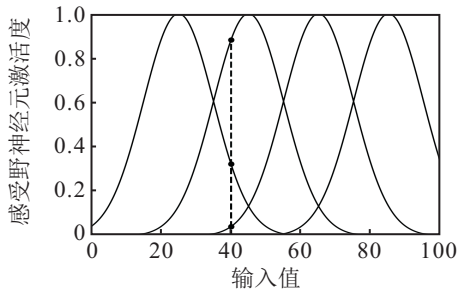


图2 稀疏编码

2.1.3 AER脉冲编码

对于连续采样的信号, AER 编码 (Address event representation) 是一类简单有效的编码方式^[22]. 具体而言, 对于连续采样信号, 根据相邻两次采样数值之差确定该时刻是否有脉冲产生, 具体表达如下:

$$s(t) = \begin{cases} 1, & u(t) - u(t - 1) \geq \text{Threshold}; \\ 0, & \text{Otherwise.} \end{cases} \quad (17)$$

其中: $s(t)$ 表示编码神经元在 t 时刻的状态, 即是否发出脉冲, 其值为 1 时, 表示该时刻产生一个脉冲, 其值为 0 时表示没有脉冲发出; $u(t)$ 和 $u(t - 1)$ 表示 t 和 $t - 1$ 时刻的采样值; Threshold 表示神经元激活阈值.

除以上 3 类编码方式外, 还有相位编码^[43]、延迟编码^[44]、BSA (Bens spike algorithm) 编码^[45] 等时间编码方式, 在此不一一赘述.

2.2 网络结构

脉冲神经网络的结构与传统神经网络类似, 可分为静态结构和动态结构. 静态结构是指神经元数量和层数不变, 只改变权重等参数, 包括多层前馈网络结构和循环网络结构; 动态结构是指网络中神经元数量和连接均可以动态调整, 典型的代表是进化脉冲神经网络 (Evolving spiking neural networks).

2.2.1 多层前馈网络结构

对于多层前馈脉冲神经网络, 其输入层为编码层, 隐含层采用脉冲神经元, 输出层根据实际需要, 既可以采用脉冲神经元, 也可使用线性单元等传统神经网络神经元. 为保证有足够多的脉冲神经元能够被激活, 通常神经元之间的连接为多连接, 称为

多突触连接, 其中每条突触上设定不同的延迟以保证脉冲神经网络的稳定性^[46]. 多层前馈神经网络结构如图3所示.

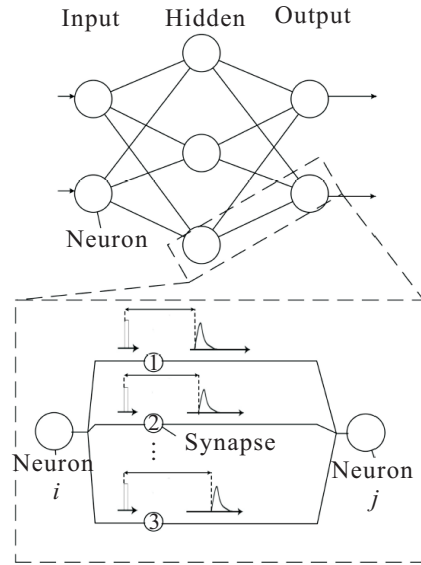


图3 前馈脉冲神经网络

2.2.2 循环网络结构

循环结构的脉冲神经网络的典型代表是液体状态机 (Liquid state machine)^[47]. 液体状态机依然可以分为输入层、中间层和输出层 3 层结构. 输入层负责将脉冲时间序列输入至中间层, 通常为 1 层编码神经元. 中间层又称为储备池, 是神经元数量庞大的点阵状结构, 内部连接随机确定, 且一旦确定就不再改变, 即中间层不需要进行训练, 是一个稀疏结构的循环神经网络^[48]. 中间层任意两神经元之间是否有连接可按照下式确定:

$$P(i, j) = C \cdot \exp\left(-\frac{D(i, j)}{\lambda^2}\right). \quad (18)$$

其中: $P(i, j)$ 为两个神经元是否有连接的概率, $D(i, j)$ 为两个神经元之间的欧氏距离, C 和 λ 为控制变量.

液体状态机具有对任意序列进行计算的能力, 其核心就在于中间层的循环结构, 它的功能是将输入序列 $u(\cdot)$ 映射至高维状态 $x^M(t)$, 如图4所示. 中间层输出为

$$x^M(t) = (L^M u)(t). \quad (19)$$

液体状态机的输出层没有固定的形式, 只要满足任务需要, 任意机器学习算法均可^[49]. 输出层的输出可表示为

$$y(t) = f^M(x^M(t)). \quad (20)$$

液体状态机具有以下优点:

- 1) 非常适合处理时序信号;

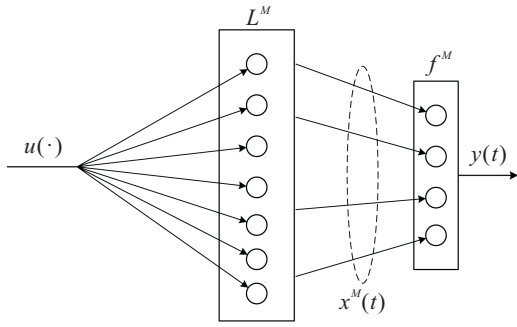


图4 液体状态机

2) 同一个网络结构可以处理任意时间范围的时间序列信号^[50];

3) 中间层可视为一个滤波器^[51],将数据从低维映射至高维,更易发现数据隐含的特征.

2.2.3 进化脉冲神经网络

进化脉冲神经网络的思想来自于连接进化系统,这类系统可以以自适应、自组织、在线连续的方式动态变化系统结构和功能^[52].进化脉冲神经网络结构如图5所示.每个样本经过多个高斯神经元编码后送入进化脉冲神经网络,根据样本特征,进化脉冲神经网络可以动态地生成新的神经元,并将其添加到相应类别的神经元储备池中^[22].在进化脉冲神经网络中,采用次序规则学习和表示输出类别,即最早激活的神经元代表相应类别.

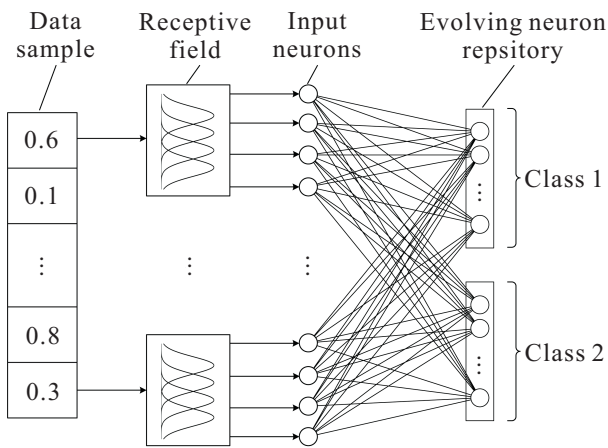


图5 进化脉冲神经网络

3 脉冲神经网络学习算法

脉冲神经网络的学习算法分为无监督学习算法和监督学习算法.无监督学习算法通过模拟生物神经网络内部的突触重塑规则来训练神经网络实现聚类等任务;监督学习算法分为梯度下降学习算法、监督STDP学习算法、基于脉冲序列卷积学习算法和针对进化脉冲神经网络的次序学习算法^[53].本节将选取最具代表性的学习算法进行详细介绍.

3.1 无监督学习算法

无监督学习算法起源于Hebb学习规则,即权重调整仅依赖于当前神经元发出脉冲时间和接收脉冲时间.Hebb对于生物神经网络权重调整的猜想由蒲慕明院士验证,Markram在此基础上发现了STDP学习规则^[54-55],在STDP规则中权重的调整量与前后两个神经元激活的时间差呈函数关系,该函数称为学习窗函数,典型的学习窗函数如下所示:

$$W(s) = \begin{cases} A_p \exp\left(\frac{-s}{\tau_p}\right), & s \geq 0; \\ -A_d \exp\left(\frac{s}{\tau_d}\right), & s < 0. \end{cases} \quad (21)$$

其中: s 为神经元发出脉冲时间与接收到脉冲时间之差, τ_p 和 τ_d 分别为权重增强和减弱的时间常数, A_p 和 A_d 分别为权重增强与减弱的增益^[56].典型的STDP学习窗函数如图6所示.

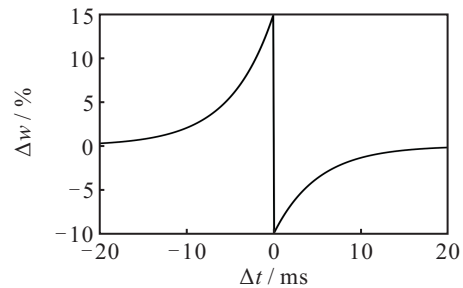


图6 STDP学习窗函数

3.2 监督学习算法

3.2.1 梯度下降学习算法

荷兰国家数学与计算机科学研究中心(Centrum Wiskunde & Informatica)的Bohte等^[57]最先提出了可应用于多层前馈脉冲神经网络的梯度下降学习算法,称为SpikeProp学习算法.该算法使用简化的脉冲响应模型,限制每个神经元仅发放一次脉冲,并且解决了由于脉冲神经元激活导致的不连续性问题.在简化的脉冲响应模型中,神经元 j 的膜电势为

$$V_j(t) = \sum_{i \in \Gamma_j} \sum_k w_{ij}^k \epsilon(t - t_i^{\text{out}} - d_{ij}^k). \quad (22)$$

其中: Γ_j 为所有与神经元 j 连接的神经元集合, w_{ij}^k 为神经元 i 与 j 之间的第 k 条连接突触权重, t_i^{out} 为神经元 i 发出脉冲的时间, d_{ij}^k 为当前连接的延迟.式(22)可简记为

$$V_j(t) = \sum_{i \in \Gamma_j} \sum_k w_{ij}^k y_i^k(t). \quad (23)$$

该算法定义平方误差函数为

$$E = \frac{1}{2} \sum_j (t_j^{\text{out}} - t_j^d)^2. \quad (24)$$

根据误差反向传播算法, 隐含层到输出层权重调整为

$$\frac{\partial E}{\partial w_{ij}^k} = \frac{\partial E}{\partial t_j^{\text{out}}} \frac{\partial t_j^{\text{out}}}{\partial w_{ij}^k} = \frac{\partial E}{\partial t_j^{\text{out}}} \frac{\partial t_j^{\text{out}}}{\partial V_j} \frac{\partial V_j}{\partial w_{ij}^k}. \quad (25)$$

Bohte给出假设: 在 $t = t_j^{\text{out}}$ 的非常小的邻域内, V_j 可以近似视为是 t 的线性函数, 因此有

$$\frac{\partial t_j^{\text{out}}}{\partial V_j} = \frac{-1}{\frac{\partial V_j}{\partial t_j^{\text{out}}}} = \frac{-1}{\sum_{i \in \Gamma_j} \sum_k w_{ij}^k \left(\frac{\partial y_j^k(t_j^{\text{out}})}{\partial t_j^{\text{out}}} \right)}. \quad (26)$$

进而, 隐含层到输出层的权重梯度可记为

$$\frac{\partial E}{\partial w_{ij}^k} = y_j^k(t_j^{\text{out}}) \delta_j, \quad (27)$$

其中

$$\delta_j = \frac{\partial E}{\partial t_j^{\text{out}}} \frac{\partial t_j^{\text{out}}}{\partial V_j} = \frac{-(t_j^{\text{out}} - t_j^d)}{\sum_{i \in \Gamma_j} \sum_k w_{ij}^k \frac{\partial y_j^k(t_j^{\text{out}})}{\partial t_j^{\text{out}}}}. \quad (28)$$

对于输入层到隐含层, 有

$$\delta_j = \frac{\sum_{i \in \Gamma_j} \delta_i \sum_k w_{ij}^k \frac{\partial y_j^k(t_j^{\text{out}})}{\partial t_j^{\text{out}}}}{\sum_{h \in \Gamma_i} \sum_l w_{hi}^l \frac{\partial y_h^l(t_i^{\text{out}})}{\partial t_i^{\text{out}}}}. \quad (29)$$

最终脉冲神经网络权重调整为

$$\Delta w_{ij}^k = -\eta y_j^k(t_j^{\text{out}}) \delta_j. \quad (30)$$

近年来, 众多研究者对 SpikeProp 算法进行了改

进, Xin 等^[58] 通过增加动量项, 加速了算法的收敛. Matsuda^[59] 提出了针对突触延迟、激活阈值等多参数的梯度下降法. Kennoch 等^[60] 提出了加速收敛的 RProp 和 QuickProp 算法. Shrestha 等^[61] 提出了基于权重收敛性分析的学习律自适应算法. Booi 等^[62] 将 SpikeProp 算法进行扩展, 解除了隐含层只发放一次脉冲的限制, 提出了 Multi-SpikeProp 算法. Xu 等^[63] 在此基础上, 提出了新的梯度下降法, 该方法不限制任何层神经元发放的脉冲数量. 实验表明, 对于大量的输出脉冲, 该算法具有较高的精度. 随后, Xu 等^[64] 又将该算法进行推广, 提出了 OnMSGDB 算法, 使之能够实现在线学习. Tiño 等^[65] 将 SpikeProp 算法扩展到循环脉冲神经网络中, 实现了脉冲序列模式的学习.

除了上述基于 SpikeProp 的梯度下降学习算法以外, Gütig 等^[66] 提出了 Tempotron 学习算法, 目的是将目标输出膜电势和实际输出膜电势之差最小化, 以实现调整突触权重, 但是 Tempotron 算法仅适用于单个神经元. Florian^[67] 提出了 Chrontron E-learning 学习算法, 可以针对完整的脉冲序列调整权重. Tempotron 和 Chrontron 算法均从传统人工神经网络出发, 将感知器的概念扩展到脉冲神经元. 表 3 总结了各类脉冲神经网络梯度下降学习算法的特点.

表 3 典型梯度下降监督学习算法

编号	算法	脉冲数量约束	网络结构
1	SpikeProp ^[57]	所有神经元仅发放一次脉冲	多层前馈结构
2	RProp 和 QuickProp ^[60]	所有神经元仅发放一次脉冲	多层前馈结构
3	SpikeProp for all parameters ^[59]	神经元发放脉冲序列	多层前馈结构
4	Multi-SpikeProp ^[62]	仅限制输出层发放单个脉冲	多层前馈结构
5	SpikeProp ^[63]	神经元发放脉冲序列	多层前馈结构
6	SpikeProp ^[63]	神经元发放脉冲序列	循环网络结构
7	OnMSGDB ^[64]	神经元发放脉冲序列	多层前馈结构
8	Tempotron ^[66]	单个神经元发放或者不发放脉冲	单神经元
9	Chrontron E-Learning ^[67]	神经元发放脉冲序列	单层神经网络

3.2.2 监督 STDP 学习算法

STDP 规则是生物神经网络内部的一种无监督学习活动, 通过突触权重的无监督调整, 可以增强求解某些具体问题的能力. 结合 STDP 规则的监督学习算法是目前脉冲神经网络学习算法研究的热点, 此类算法具有更好的生物解释性. 典型的监督 STDP 学习算法是 Ponulak 等^[68] 提出的远程监督学习算法, 称为 ReSuMe (Remote supervised method), 该算法具有在线学习能力, 并且能够处理脉冲时间序列. ReSuMe 算

法中权重调整为

$$\frac{dw(t)}{dt} = [S^d(t) - S^l(t)] \left[a + \int_0^\infty W(s) S^{\text{in}}(t-s) ds \right]. \quad (31)$$

其中: $S^d(t)$ 和 $S^l(t)$ 分别为目标脉冲序列和实际输出脉冲序列, $S^{\text{in}}(t)$ 为输入脉冲序列; 参数 a 为 non-Hebbian 项, 用于加速收敛; $W(s)$ 为 STDP 规则中的学习窗函数 (如图 6 所示).

ReSuMe 算法仅适用于单层神经网络, 通常该算法与液体状态机结合, 训练液体状态机的输出层权

重. Taherkhani^[69]提出了DL-ReSuMe,该算法不仅可以训练权重,还可以修正每条连接突触上的延迟. Sporea等^[70]结合SpikeProp算法和ReSuMe算法,成功训练了多层前馈神经网络.

除ReSuMe外,还有众多基于STDP规则的监督学习算法. Wade等^[71]提出了SWAT学习算法,该算法输出层包含多个输出神经元和一个训练神经元.学习过程中,先对训练神经元的突触权重进行训练,然后将其映射到对应的输出神经元,该算法可以处理

脉冲序列,且可应用于多层前馈神经网络. Paugam-Moisy等^[72]结合STDP规则和Polychronization规则,提出了基于储备池输出层的监督学习算法. Brea等^[73]将监督STDP学习算法推广到循环神经网络中. Xu等^[74]提出了PBSNLR学习规则,利用突触前神经元膜电势结合STDP规则进行学习. Xie等^[75]根据膜电势在目标时刻和实际脉冲时刻的电势差,提出ASA学习算法,经实验验证,该算法具有较好的分类性能. 各类监督STDP规则学习算法特点如表4所示.

表4 典型监督STDP规则学习算法

编号	算法	脉冲数量约束	网络结构
1	ReSuMe ^[58]	神经元发放脉冲序列	单神经元或单层神经网络
2	DL-ReSuMe ^[69]	神经元发放脉冲序列	单神经元或单层神经网络
3	ReSuMe & SpikeProp ^[70]	神经元发放脉冲序列	多层前馈结构
4	SWAT ^[71]	神经元发放脉冲序列	多层前馈结构
5	Reservoir-Based STDP ^[72]	输出层发放单个脉冲	储备池结构
6	Recurrent STDP ^[73]	神经元发放脉冲序列	循环网络结构
7	PBSNLR ^[74]	神经元发放脉冲序列	单层神经网络
8	ASA ^[75]	神经元发放脉冲序列	多层前馈结构

3.2.3 基于脉冲序列卷积的学习算法

为了方便分析和计算神经元所发放的脉冲序列,选择特定的核函数 $\kappa(t)$,将离散的脉冲时间序列转换为连续的函数,这便是基于脉冲序列卷积的学习算法的核心思想,即

$$\tilde{S}(t) = S(t)\kappa(t) = \sum_f \kappa(t - t^f). \quad (32)$$

基于脉冲序列卷积的学习算法的代表是Mohammed等^[76]提出的SPAN(Spike pattern association neuron)算法. 首先将脉冲时间序列与核函数卷积,即

$$\begin{cases} x_i(t) = \sum_{t_i^f \in F_{in}} \kappa(t - t_i^f), \\ y_d(t) = \sum_{t_d^g \in F_d} \kappa(t - t_d^g), \\ y_a(t) = \sum_{t_a^h \in F_a} \kappa(t - t_a^h), \end{cases} \quad (33)$$

其中 F_{in} 、 F_d 和 F_a 分别为输入、目标和实际输出脉冲时间序列. SPAN算法受Widrow-Hoff学习规则启发,权重调整为

$$\Delta w_i(t) = \lambda x_i(t)(y_d(t) - y_a(t)). \quad (34)$$

式(34)表明: SPAN学习算法是一个实时学习规则,突触权重随着时间不断调整. 因此在整个时间域上,权重调整为

$$\Delta w_i = \lambda \int_0^\infty x_i(t)(y_d(t) - y_a(t))dt, \quad (35)$$

脉冲序列卷积核函数常取为 $e\tau^{-1}te^{-t/\tau}H(t)$, $H(t)$ 为阶跃函数. 卷积后脉冲序列为

$$\tilde{S}(t) = \sum_f \kappa(t - t^f) = \sum_f e\tau^{-1}(t - t^f)e^{-(t-t^f)/\tau}H(t - t^f). \quad (36)$$

将式(36)代入(35)可得

$$\begin{aligned} \Delta w_i &= \lambda \int_0^\infty \Delta w_i(t)dt = \\ &= \lambda \left(\frac{e}{2}\right)^2 \left[\sum_g \sum_f (|t_i^f - t_d^g| + \tau)e^{-\frac{|t_i^f - t_d^g|}{\tau}} - \right. \\ &\quad \left. \sum_h \sum_f (|t_i^f - t_a^h| + \tau)e^{-\frac{|t_i^f - t_a^h|}{\tau}} \right], \end{aligned} \quad (37)$$

其中 t_i^f 、 t_d^g 和 t_a^h 分别为输入、目标和实际输出脉冲序列对应的脉冲发放时间.

SPAN算法也有其局限性: 该算法比较复杂,且仅适用于单层神经网络结构,无法像SpikeProp算法一样对多层前馈网络进行训练. 通常SPAN算法与液体状态机结合,用来训练液体状态机的输出层. 此外, Carnell等^[77]应用线性代数的方法进行学习,首先将脉冲序列表示为加权脉冲时间序列的形式 $S(t) = \sum w_i s(t_i)$,并且定义两个脉冲时间序列的内积为

$$\langle S_i(t), S_j(t) \rangle = \sum w_i w_j \exp\left(-\frac{t_i - t_j}{\delta}\right). \quad (38)$$

神经网络的误差定义为 $E = S_d(t) - S_o(t)$,应用Gram

Schmidt过程定义的投影公式计算权重的变化

$$\Delta w_i = \frac{\text{Proj}_{S_i(t)} E}{\text{norm}(S_i(t))}. \quad (39)$$

Yu等^[78]扩展了SPAN算法,提出了PSD(Precise-spike-driven)学习算法.与SPAN不同的是,PSD算法仅将输入脉冲时间序列通过核函数卷积为连续的函数,该算法在手写数字识别等任务中取得了较好的效果^[79].表5是几类典型的脉冲序列卷积学习算法.

表5 基于脉冲序列卷积的学习算法

编号	算法	脉冲数量约束	网络结构
1	SPAN ^[76]	神经元发放脉冲序列	单层神经网络
2	线性代数内积方法 ^[77]	神经元发放脉冲序列	单层神经网络
3	PSD ^[78]	神经元发放脉冲序列	单层神经网络

3.2.4 次序学习算法

次序学习算法由Thorpe提出,并由Kasabov成功应用于进化脉冲神经网络^[80].进化脉冲神经网络首先对样本类别 l 生成空的神经元储备池 R_l (如图5所示),对任意属于类别 l 的样本 X_i ,生成新的神经元 i ,放入相应类别的神经元储备池,该神经元各项参数为

$$w_{ij} = (m_l)^{\text{order}(j)}, \quad (40)$$

$$u_{\max}^i = \sum_j w_{ij} (m_l)^{\text{order}(j)}, \quad (41)$$

$$\vartheta_i = c_l u_{\max}^i. \quad (42)$$

其中: w_{ij} 为新生成神经元与前一层输入神经元的连接权重; u_{\max}^i 为该神经元最大膜电压; ϑ_i 为该神经元的激活阈值; $\text{order}(j)$ 为从神经元 j 的输入到新生成神经元 i 的脉冲时间序列的次序,如果神经元 j 发出的脉冲第一个到达神经元 i ,则 $\text{order}(j) = 0$; m_l 和 c_l 均为常数.

新生成的神经元 i 与该类别神经元储备池中所有的神经元比较权重的欧氏距离,若与神经元 k 的权重欧氏距离小于阈值 s_l ,即

$$\min(d(w_{ij}, w_{kj})) \leq s_l, \quad (43)$$

则更新神经元 k 的权重和激活阈值分别为

$$w_{kj} = \frac{w_{ij} + N w_{kj}}{1 + N}, \quad (44)$$

$$\vartheta_k = \frac{\vartheta_i + N \vartheta_k}{1 + N}, \quad (45)$$

其中 N 为神经元 k 在此之前更新过的次数.权重和激活阈值更新结束后,新生成的神经元 i 将从储备池 R_l 中丢弃.该过程实际上是判断新生成神经元 i 与现有神经元 k 的相似程度,如果两个神经元之间的差异在某个容许范围内,则通过略微修改现有神经元 k 的

参数,以便其更好地表达样本特征.事实上,随着样本不断送入神经网络,该更新过程趋近于激活阈值和权重的平均化过程^[81].

如果新生成的神经元 i 与储备池中任意神经元的权重欧氏距离均超过了阈值 s_l ,则认定神经元 i 包含新的特征,因此会保留在储备池中.

次序学习算法的优点在于不需要多次学习样本,仅需将所有训练样本一次导入,便可以达到极佳的学习效果,且训练速度较快^[82].

4 脉冲神经网络的应用

4.1 脉冲神经网络在机器人控制中的应用

Cao等^[83]将脉冲神经网络成功应用于机器人控制中.环境信息由编码器、CCD摄像机和超声波传感器获取,并将编码的脉冲时间序列送入3层脉冲神经网络.其中输入层神经元分为3部分:目标检测神经元、传感器神经元和避障神经元.避障神经元的权重固定,其他神经元的权重均可根据简单的Hebb规则自适应调整.输入层神经元使用频率编码,对于神经元 j ,输入为

$$N_j = \text{round}\left(\frac{I_j}{S_{\text{unit}}}\right), \quad (46)$$

即求单位时间 S_{unit} 内收到的脉冲数量,并取整.随后将计算得到的脉冲频率在一固定时间窗口 t_{time} 内平均分布,脉冲序列为

$$O_j(t) = \sum_{i=1}^{N_j} \delta\left(t - \text{round}\left(\frac{t_{\text{time}}}{N_j}(i-1) + 1\right)\right). \quad (47)$$

避障神经元有3个脉冲神经元,其中神经元 O_1 和 O_3 采用WTA规则^[84],即 $N_1 \geq N_3$ 时,神经元 O_1 输出;反之,则神经元 O_3 输出.隐含层避障神经元的输出为

$$\begin{cases} O_{\text{hide},1}(t) = O_1(t) \wedge O_2(t), \\ O_{\text{hide},2}(t) = O_3(t) \wedge O_2(t), \end{cases} \quad (48)$$

其中 \wedge 表示逻辑“与”运算,只有当两个神经元同时发放脉冲时,对应的隐含层神经元才发放脉冲^[85].输出层使用了脉冲响应模型,用来进行运动控制.运动控制神经元分成两组,分别控制左右两侧的电机,并通过控制电机正传和反转来控制机器人前进或后退.由第1.3节的脉冲响应模型可知,当神经元膜电势 $V_m \geq \vartheta$ 时,神经元发放脉冲.记运动控制神经元的输出为

$$O_i(t) \begin{cases} 1, & t \in \Gamma_i^f; \\ 0, & \text{Otherwise.} \end{cases} \quad (49)$$

其中 Γ_i^f 为神经元 i 在时间窗口 t_{time} 内发出的脉冲时间序列.因此每个运动神经元发出脉冲频率为

$$n_i = \sum_{t=1}^{l_{\text{time}}} O_i(t), \quad i = lf, rf, lb, rb. \quad (50)$$

同一侧的电机显然不可能同时正转和反转,因此引入竞争机制^[86],即输出为

$$\begin{cases} \bar{n}_{lo} = \text{sign}(n_{lf} - n_{lb}) \max(n_{lf}, n_{lb}), \\ \bar{n}_{ro} = \text{sign}(n_{rf} - n_{rb}) \max(n_{rf}, n_{rb}). \end{cases} \quad (51)$$

经约束滤波后,电机转速为

$$\begin{cases} v_l = p \cdot n_{lo}, \\ v_r = p \cdot n_{ro}. \end{cases} \quad (52)$$

此外,Mahadevuni等^[87]将脉冲神经网络与强化学习结合,成功实现了机器人的路径规划任务.Takase等^[88]利用进化脉冲神经网络实现了机器人的运动控制任务.Hulea等^[89]成功将脉冲神经网络应用于人工肌肉的控制任务.Botzheim等^[90]结合情感模型,成功实现了基于脉冲神经网络的机器人伴侣.de Azambuja等^[91]利用液体状态机实现仿人机器人的控制.

4.2 脉冲神经网络在类脑研究中的应用

生物大脑可以视为一个处理复杂时空信息的结构,通过对大脑施加某种特定的外界刺激,测定大脑的时间和空间响应信息,获得大脑结构、功能区块划分、分子和基因信息等,是研究大脑的最重要方式之一.通过研究大脑,进一步可以启发人工智能和其他机器学习及控制领域研究.现有的传统机器学习方法对复杂的脑信息(如脑电波(EEG)、功能性磁共振成像(fMRI)、弥散张量成像(DTI)等信息^[92])处理效果并不理想.脉冲神经网络与大脑类似,均使用脉冲时间编码,在处理脑信息具有得天独厚的优势.Kasabov^[93]提出用脉冲神经网络处理脑电波信息,并取得了较好的效果,设计了基于储备池的脉冲神经网络,并开发了NeuCube工具箱.采集到的脑电信息经编码器编码为脉冲序列,编码方式为式(17)所示的AER编码.编码后的脉冲时间序列送入NeuCube中,NeuCube是一个特殊的液体状态机,内部神经元之间的连接权重是可以动态无监督调整的,调整规则使用了STDP规则.NeuCube的空间结构是仿照大脑结构设计的,内部神经元按照功能区连接,同一功能区内部的神经元空间距离很近,且有大量的连接,而功能区之间的连接比较稀疏,功能区的划分均是由DTI、EEG和MRI等方式测量得到的^[94].NeuCube的结构也是可以动态变化的,根据任务需求,可以动态地生成新的连接或新的神经元.

NeuCube将输出的状态向量,即脉冲时间序列送

入分类器.分类器形式并不唯一,Kasabov等^[95]给出的方式是使用动态进化脉冲神经网络进行分类,或结合SPAN学习规则,使用单层脉冲神经网络进行分类.

除了分析脑电信号外,Tu等^[96]利用NeuCube工具箱提高了处理时空信息的精度;Maliavko等^[97]基于人脑结构建立了具有自学习和自修正的脉冲神经网络;Zhang等^[98]受大脑海马体启发建立了具有记忆效应的脉冲神经网络;Kasap等^[99]受上丘脑启发,模拟了生物扫视运动的生理过程;Pedretti等^[100]结合STDP规则建立了在线学习和目标追踪的神经网络模型;Li等^[101]结合STDP规则和液体状态机,成功实现了无监督聚类任务.

4.3 脉冲神经网络在模式识别中的应用

随着半导体材料和生物技术的发展,仿生设备也随之得到了长足进步,电子眼、电子鼻等仿生器官逐渐在医疗康复领域推广.脉冲神经网络是最具有生物特性的神经网络,非常适合处理仿生设备的信息^[102].Sarkar等^[103]提出利用脉冲神经网络,结合仿生电子鼻,对气味进行分类.Martinelli等^[104]的实验表明:在嗅觉系统中,绝大多数信息均由首次脉冲时间传递.因此,Sarkar在文章中指出:对电子鼻的信号编码仅考虑首次脉冲时间,并且结合稀疏编码,使用5个高斯神经元进行编码.电子鼻中每个传感器的输出不是一个固定的值,而是时变的函数.传感器输出函数与每个编码高斯函数第一次相交的时间,就是编码时间.为保证编码成功,需要将电子鼻传感器信号进行归一化处理,设第*i*个传感器得到的时间序列为 S_i ,则归一化后为

$$\hat{S}_i = \frac{S_i - \min(S_i)}{\max(S_i) - \min(S_i)}. \quad (53)$$

电子鼻传感器输出函数与第*k*个编码高斯神经元($rf^{(k)}(t)$)的交点为

$$I_i^{(k)} = \{t : rf^{(k)}(t) = \hat{S}_i(t), 0 \leq t \leq T_{\text{sample}}\}. \quad (54)$$

由于仅需考虑首次脉冲时间,最终编码后得到的脉冲时间序列可以表示为

$$\mathbf{T}^{(k)} = \{I_i^{(k)}(0) : 1 \leq i \leq n_{\text{sensors}}\}. \quad (55)$$

在仅考虑首次脉冲时间的条件下,采样时间窗口仅需要10ms左右,提高了神经网络的运行效率.编码后的脉冲时间序列送入一个多层前馈脉冲神经网络中,并使用SpikeProp学习算法进行训练学习,经训练后,该神经网络可有效区分气味,并且效果优于传统的多层前馈神经网络.

此外脉冲神经网络还在语音识别^[105-108]、图像识别^[109-111]和目标检测^[112]等领域有诸多应用,并取得

了良好的效果.

5 脉冲神经网络的硬件实现

随着各国脑计划的推进,类脑智能研究逐渐成为热点.为实现生物大脑的高性能、低功耗、并行计算的运行机制,传统的冯诺依曼架构计算机已经难以满足要求,设计神经网络芯片成为类脑智能研究中的一个重点方向.由于诸多脉冲神经元模型是由微分方程或连续核函数描述的,这类脉冲神经元构成的网络可以直接表达为模拟电路,通过模拟电路搭建的脉冲神经网络可以连续地更新神经网络状态,克服了传统计算机离散化为数字电路所产生的量化误差^[113].模拟神经网络芯片最重要的代表是BrainScaleS计划,其核心是HICANN神经网络芯片,单个芯片内部含有512个LIF神经元,共计有131 072条连接突触,多个HICANN芯片可以组合使用,可进行并行计算^[114].然而,相比大脑结构,BrainScaleS计划的芯片仍然非常原始简陋,Benjamin等^[115]设计了Neurogrid脉冲神经网络芯片,该芯片内部包含16个神经网络核心,每个核心内部有65 536个脉冲神经元,整个芯片内部含有60亿个连接突触,在实现大规模模拟大脑方面展现出优秀的效果.美国DARPA发起SyNAPSE计划,由HRL实验室承接该计划,设计脉冲神经网络芯片,成功地将深度卷积神经网络映射到脉冲神经网络芯片上,在图像处理方面展现出了强大能力^[116].此外,Qiao等^[117]设计ROLLS脉冲神经网络芯片,实现了网络的实时学习机制,并在功耗控制方面表现突出,典型应用下的功耗仅有4 mW.

模拟电路在描述脉冲神经元动态性能方面展现了优秀的性能,但是模拟电路本身受噪声、温度等外界影响较大,且编程配置方面较为困难.为克服以上的缺陷,研究者设计了基于数字电路的脉冲神经网络,典型的代表是IBM公司设计的TrueNorth芯片^[11],芯片内部有4 096个核心,每个核心包含256个脉冲神经元,每个神经元有256条连接权重.与上文介绍的脉冲神经网络芯片不同,TrueNorth芯片的目的不是模拟大脑的运行机制,而是实现高效的实际应用.上述脉冲神经网络芯片的共同缺陷是灵活性较低,在不同任务中不能够实现内部结构的大规模调整,应用局限比较大.针对以上问题,在欧洲脑计划中,Furber等^[118]开发了SpiNNaker芯片,该芯片内部包含18个ARM处理器,每个处理器可以模拟大约1 000个脉冲神经元,每个神经元可以包含1 000条连接突触,由于SpiNNaker是基于ARM处理器开发的,使用中可以调用ARM指令集,在实际应用时,灵活性

大大提高.Ma等^[119]设计了达尔文芯片,该芯片共可同时模拟2 048个脉冲神经元,并且能生成超过400万条突触连接,这些连接权重均可改变,达尔文芯片现已成功应用于类脑研究和模式识别等任务中.

6 总结与展望

本文综合介绍了脉冲神经网络的神经元模型、结构、学习算法、应用及硬件实现等基本内容,给出了近年来脉冲神经网络研究的整体情况,并简要分析了脉冲神经网络各类学习算法和应用中优缺点.通过对当前脉冲神经网络研究的情况和各国脑计划的研究进展的分析,可以看出脉冲神经网络是未来人工智能的发展方向,是实现智能化必不可少的一部分.当前脉冲神经网络仍存在众多问题亟待解决,脉冲神经网络学习算法繁多,但未发现类似于传统神经网络BP算法这样高效的学习算法.此外,目前脉冲神经元模型仍只关注于神经元膜电压的变化情况,尚不能完整描述生物神经元内部复杂的生理机制;神经元之间突触的生长、消亡等复杂机理仍不明确;大脑神经网络如何针对信息进行编码等问题,仍然困扰着研究者,这些问题是未来类脑智能研究的热点和重点.另外,人工智能的热潮也将推动脉冲神经网络的蓬勃发展,高性能、低功耗的脉冲神经网络芯片必将是人工智能的重要实现平台.总体而言,脉冲神经网络是人工智能领域较为前沿的领域,在类脑智能研究中占据重要地位,是人工智能与生物智能之间的重要桥梁,有待研究人员开展长期持续的科研工作.

参考文献(References)

- [1] Haykin S S. Neural networks and learning machines[M]. Upper Saddle River: Pearson Education, 2009: 115-128.
- [2] Quiroga R Q, Panzeri S. Principles of neural coding[M]. Boca Raton, FL: CRC Press, 2013: 5-30.
- [3] Dayan P, Abbott L F. Theoretical neuroscience: Computational and mathematical modeling of neural systems[M]. New York: MIT Press, 2001: 101-108.
- [4] Maass W. Networks of spiking neurons: The third generation of neural network models[J]. Neural Networks, 1997, 10(9): 1659-1671.
- [5] Maass W, Bishop C M. Pulsed neural networks[M]. Massachusetts: MIT Press, 1999: 55-81.
- [6] Hebb D O. The organization of behavior: A neuropsychological theory[M]. New York: Wiley-Interscience, 1949: 9-70.
- [7] Knudsen E I. Supervised learning in the brain[J]. J of Neuroscience, 1994, 14(7): 3985-3997.
- [8] Kasinski A, Ponulak F. Comparison of supervised learning methods for spike time coding in spiking neural

- networks[J]. *Int J of Applied Mathematics and Computer Science*, 2006, 16(1): 101-113.
- [9] Rekabdar B, Nicolescu M. Using patterns of firing neurons in spiking neural networks for learning and early recognition of spatio-temporal patterns[J]. *Neural Computing and Application*, 2017, 25(5): 881-897.
- [10] Mohemmed A, Schliebs S, Matsuda S. Training spiking neural networks to associate spatio-temporal input-output spike patterns[J]. *Neurocomputing*, 2013, 107(4): 3-10.
- [11] Akopyan F, Sawada J, Cassidy A, et al. True North: Design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip[J]. *IEEE Trans on Computer-Aided Design of Integrated Circuits and Systems*, 2015, 34(10): 1537-1557.
- [12] Gerstner W, Kistler W M. *Spiking neuron models: Single neurons, populations, plasticity*[M]. New York: Cambridge University Press, 2002: 20-25.
- [13] Abbott L F. Lapique's introduction of the integrate-and-fire model neuron(1907)[J]. *Brain Research Bulletin*, 1999, 50(5): 303-304.
- [14] Hodgkin A L, Huxley A F. A quantitative description of membrane current and its application to conduction and excitation in nerve[J]. *Physiol*, 1952, 117(4): 500-544.
- [15] Teka W W, Upadhyay R K, Mondal A. Fractional-order leaky integrate-and-fire model with longterm memory and power law dynamics[J]. *Neural Networks*, 2017, 93: 110-125.
- [16] Galves A, Löcherbach E. Infinite systems of interaction chains with memory of variable length-A stochastic model for biological neural nets[J]. *J of Statistical Physics*, 2013, 151(5): 896-921.
- [17] Fourcaud T N. How spike generation mechanisms determine the neuronal response to fluctuating inputs[J]. *The J of Neuroscience*, 2003, 23(37): 11628-11640.
- [18] Li X, Ning L J. Stochastic resonance in FitzHugh-Nagumo model driven by multiplicative signal and non-Gaussian noise[J]. *Indian J of Physics*, 2015, 89(2): 189-194.
- [19] Behdad R, Binczak S, Dmitrichev A, et al. Artificial electrical Morris-Lecar neuron[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2015, 26(9): 1875-1884.
- [20] Lakshmanan S, Lim C P, Nahavandi S, et al. Dynamical analysis of the Hindmarsh-Rose neuron with time delays[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2017, 28(8): 1953-1958.
- [21] Henley B C, Shin D C, Zhang R, et al. Compartmental and data-based modeling of cerebral hemodynamics: Nonlinear analysis[J]. *IEEE Trans on Biomedical Engineering*, 2017, 64(5): 1078-1088.
- [22] Kasabov N K, Dhoble K, Nuntalid N, et al. Dynamic evolving spiking neural networks for on-line spatio and spectro-temporal pattern recognition[J]. *Neural Networks*, 2013, 41(5): 188-201.
- [23] Izhikevich E M. Simple model of spiking neurons[J]. *IEEE Trans on Neural Networks*, 2003, 14(6): 1569-1572.
- [24] Castro L, Aguiar P. Phase precession through acceleration of local theta rhythm: A biophysical model for the interaction between place cells and local inhibitory neurons[J]. *J of Computational Neuroscience*, 2012, 33(1): 141-150.
- [25] Nossenson N, Megal N, Messer H. Detection of stimuli from multi-neuron activity: Empirical study and theoretical implications[J]. *Neurocomputing*, 2016, 174: 822-837.
- [26] Elliot T. The mean time to express synaptic plasticity in integrate-and-express, stochastic models of synaptic plasticity introduction[J]. *Neural Computation*, 2011, 23(1): 124-159.
- [27] Costa R P, Padamsey Z, D'Amour J A, et al. Synaptic transmission optimization predicts expression loci of long-term plasticity[J]. *Neuron*, 2017, 96(1): 177-189.
- [28] Fan D, Sharad M, Sengupta A, et al. Hierarchical temporal memory based on spin-neurons and resistive memory for energy-efficient brain-inspired computing[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2017, 27(9): 1907-1919.
- [29] Forrest M D. Can the thermodynamic Hodgkin-Huxley model of voltage-dependent conductance extrapolate for temperature[J]. *Computation*, 2014, 2(2): 47-60.
- [30] Koch C, Segev I. *Methods in neuronal modeling: from ions to networks*[M]. 2nd ed. Cambridge: MIT Press, 1999: 55-90.
- [31] Burkitt A N. A review of the integrate-and-fire neuron model: I. Homogeneous synaptic input[J]. *Biological Cybernetics*, 2006, 95(1): 1-19.
- [32] Masquelier T, Guyonneau R, Thorpe S J. Competitive STDP-based spike pattern learning[J]. *Neural Computation*, 2009, 21(5): 1259-1276.
- [33] Muntean I L, Peter R I, Joldos M. Dynamics analysis of parallel simulations of the spike response model[C]. *The 14th Int Symposium on Symbolic and Numeric Algorithms for Scientific Computing*. Timisoara: IEEE, 2012: 481-486.
- [34] Izhikevich E M, Edelman G M. Large-scale model of mammalian thalamocortical systems[J]. *Proc of the National Academy of Sciences of the United States of America*, 2008, 105(9): 3593-3598.
- [35] Gollisch T, Meister M. Rapid neural coding in the retina with relative spike latencies[J]. *Science*, 2008, 319(5866): 1108-1111.
- [36] Kostal L, Lansky P, Rospars J. Neuronal coding and spiking randomness[J]. *European J of Neuroscience*, 2007, 26(10): 2693-2701.
- [37] Rehn M, Sommer F T. A network that uses few active neurons to code visual input predicts the diverse

- shapes of cortical receptive fields[J]. *J of Computational Neuroscience*, 2007, 22(2): 135-146.
- [38] Hromadka T, Deweese M R, Zador A M. Sparse representation of sounds in the unanesthetized auditory cortex[J]. *PLoS Biology*, 2008, 6(1): 124-137.
- [39] Crochet S, Poulet J, Kremer Y, et al. Synaptic mechanisms underlying sparse coding of active touch[J]. *Neuron*, 2011, 69(6): 1160-1175.
- [40] Ito I, Ong R C, Raman B, et al. Sparse odor representation and olfactory learning[J]. *Nature Neuroscience*, 2008, 11(10): 1177-1184.
- [41] Lin A C, Bygrave A, Calignon A D, et al. Sparse, decorrelated odor coding in the mushroom body enhances learned odor discrimination[J]. *Nature Neuroscience*, 2014, 17(4): 559-568.
- [42] Mathis A, Herz A V, Stemmler M B, et al. Resolution of nested neuronal representations can be exponential in the number of neurons[J]. *Physical Review Letters*, 2012, 109(1): 018103.
- [43] Wang Z, Guo L, Adjouadi M. Wavelet decomposition and phase encoding of temporal signals using spiking neurons[J]. *Neurocomputing*, 2016, 173: 1203-1210.
- [44] Popovych O V, Yanchuk S, Tass P A. Delay and coupling-induced firing patterns in oscillatory neural loops[J]. *Physical Review Letters*, 2011, 107(22): 228102.
- [45] Schrauwen B, Campenhout J V. BSA, a fast and accurate spike train encoding scheme[C]. *Int Joint Conf on Neural Networks*. Portland: IEEE, 2003: 2825-2830.
- [46] Rowcliffe P, Feng J. Training spiking neuronal networks with applications in engineering tasks[J]. *IEEE Trans on Neural Networks*, 2008, 19(9): 1626-1640.
- [47] Maass W, Natschlager T, Markram H. Real-time computing without stable states: A new framework for neural computation based on perturbations[J]. *Neural Computation*, 2002, 14(11): 2531-2560.
- [48] Yamazaki T, Tanaka S. The cerebellum as a liquid state machine[J]. *Neural Networks*, 2007, 20(3): 290-297.
- [49] Maass W, Markram H. On the computational power of recurrent circuits of spiking neurons[J]. *Neural Computation*, 2004, 16(4): 593-616.
- [50] Hananel H, Larry M M. Topological constraints and robustness in liquid state machines[J]. *Expert Systems with Applications*, 2012, 39(2): 1597-1606.
- [51] Jin Y, Li P. Performance and robustness of bio-inspired digital liquid state machines: A case study of speech recognition[J]. *Neurocomputing*, 2017, 226: 145-160.
- [52] Watts M J. Sleep learning and max-min aggregation of evolving connectionist systems[C]. *2016 Int Joint Conf on Neural Networks*. Vancouver: IEEE, 2016: 4325-4330.
- [53] Gavrilov A, Panchenko K. Method of learning for spiking neural networks[C]. *The 13th Int Scientific-Technical Conf on Actual Problems of Electronics Instrument Engineering*. Novosibirsk: IEEE, 2016: 455-460.
- [54] Roy S, Basu A. An online unsupervised structural plasticity algorithm for spiking neural networks[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2017, 28(4): 900-910.
- [55] Higgins I, Stringer S, Schnupp J. Unsupervised learning of temporal features for word categorization in spiking neural network model of the auditory brain[J]. *PLoS One*, 2017, 12(8): e0180174.
- [56] Erfanian N E, Blamey P, Anthony N, et al. Learning pitch with STDP: A computational model of place and temporal pitch perception using spiking neural networks[J]. *PLoS Computational Biology*, 2016, 12(4): e1004860.
- [57] Bohte S, Kok J, Poutre H. Error-backpropagation in temporally encoded networks of spiking neurons[J]. *Neurocomputing*, 2002, 48(1): 17-37.
- [58] Xin J, Embrechts M J. Supervised learning with spiking neuron networks[C]. *Int Joint Conf on Neural Networks*. Washington DC: IEEE, 2001: 1772-1777.
- [59] Matsuda S. BPSpike: A backpropagation learning for all parameters in spiking neural networks with multiple layers and multiple spikes[C]. *Int Joint Conf on Neural Networks*. Vancouver: IEEE, 2016: 293-298.
- [60] Kennoch S, Liu D, Bushwell L G. Fast modifications of the SpikeProp algorithm[C]. *Int Joint Conf on Neural Networks*. Vancouver: IEEE, 2006: 3970-3977.
- [61] Shrestha S B, Spng Q. Adaptive learning rate of SpikeProp based on weight convergence analysis[J]. *Neural Networks*, 2015, 63(63): 185-198.
- [62] Booi O, Nguyen T H. A gradient descent rule for spiking neurons emitting multiple spikes[J]. *Information Processing Letters*, 2005, 95(6): 552-558.
- [63] Xu Y, Zeng X, Han L, et al. A supervised multi-spike learning algorithm based on gradient descent for spiking neural networks[J]. *Neural Networks*, 2013, 43(4): 99-113.
- [64] Xu Y, Yang J, Zhong S. An online supervised learning method based on gradient descent for spiking neurons[J]. *Neural Networks*, 2017, 93: 7-20.
- [65] Tiño P, Mills A J S. Learning beyond finite memory in recurrent networks of spiking neurons[J]. *Neural Computation*, 2006, 18(3): 591-613.
- [66] Gütig R, Sompolinsky H. The tempotron: A neuron that learns spike timing-based decisions[J]. *Nature Neuroscience*, 2006, 9(3): 420-428.
- [67] Florian R V. The Chronotron: A neuron that learns to fire temporally precise spike patterns[J]. *PLoS One*, 2012, 7(8): e40233.
- [68] Ponulak F, Kasinski A. A supervised learning in spiking neural networks with ReSuMe: Sequence learning, classification, and spike shifting[J]. *Neural Computation*, 2010, 22(2): 467-510.
- [69] Tahekhani A, Belatreche A, Li Y H, et al. DL-ReSuMe:

- A delay learning-based Remote supervised method for spiking neurons[J]. *IEEE Trans on Neural Networks*, 2015, 26(12): 3137-3149.
- [70] Sporea I, Grüning A. Supervised learning in multilayer spiking neural networks[J]. *Neural Computation*, 2013, 25(2): 473-509.
- [71] Wade J J, McDaid L L, Santos J A, et al. SWAT: A spiking neural network training algorithm for classification problems[J]. *IEEE Trans on Neural Networks*, 2010, 21(11): 1817-1830.
- [72] Paugam-Moisy, Martinez R, Bengio S. Delay learning and polychronization for reservoir computing[J]. *Neurocomputing*, 2008, 71(7): 1143-1158.
- [73] Brea J, Senn W, Pfister J P. Matching recall and storage in sequence learning with spiking neural networks[J]. *The J of Neuroscience*, 2013, 33(23): 9565-9575.
- [74] Xu Y, Zeng X, Zhong S. A new supervised learning algorithm for spiking neurons[J]. *Neural Computation*, 2013, 25(6): 1472-1511.
- [75] Xie X, Qu H, Zhang Y, et al. Efficient training of supervised spiking neural network via accurate synaptic-efficiency adjustment method[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2017, 28(6): 1411-1424.
- [76] Mohemmed A, Schliebs S, Matsuda S, et al. Training spiking neural networks to associate spatio-temporal input-output spike patterns[J]. *Neurocomputing*, 2013, 107: 3-10.
- [77] Carnell A, Richardson D. Linear algebra for times series of spikes[C]. *The 13th European Symposium on Artificial Neural Networks*. Evere: d-Side, 2005: 363-368.
- [78] Yu Q, Tang H, Tan K, et al. Precise-spike-driven synaptic plasticity: Learning hetero-association of spatiotemporal spike pattern[J]. *PLoS One*, 2013, 8(11): e78318.
- [79] Yu Q, Tang H, Tan K, et al. A brain-inspired spiking neural network model with temporal encoding and learning[J]. *Neurocomputing*, 2014, 138: 3-13.
- [80] Capecci E, Doborjeh Z G, Mammone N, et al. Longitudinal study of alzheimer's disease degeneration through EEG data analysis with a NeuCube spiking neural network model[C]. *2016 Int Joint Conf on Neural Networks*. Vancouver: IEEE, 2016: 1360-1366.
- [81] Saleh A Y, Shamsuddin S M, Hamed H A. An integrated harmony search algorithm-based multiobjective differential evolution of evolving spiking neural network[J]. *Int J of Intelligent Systems Technologies and Applications*, 2016, 15(3): 192-202.
- [82] Dora S, Subramanian K, Suresh S, et al. Development of a self-regulating evolving spiking neural network for classification problem[J]. *Neurocomputing*, 2016, 171: 1216-1229.
- [83] Cao Z, Cheng L, Zhou C, et al. Spiking neural network-based target tracking control for autonomous mobile robots[J]. *Neural Computing & Applications*, 2015, 26(8): 1839-1847.
- [84] Yesim O, Okyay K. Control of a direct drive robot using fuzzy spiking neural networks with variable structure systems-based learning algorithm[J]. *Neurocomputing*, 2015, 149: 690-699.
- [85] Wang Xu, Cao Zhi-qiang, Zhang Wen-wen, et al. Formation control for multiple mobile robots based on the spiking neural network[C]. *IEEE Conf on Robotics, Automation and Mechatronics*. Singapore: IEEE, 2010: 447-452.
- [86] Fischl K D, Tognetti G, Mendat D R, et al. Neuromorphic self-driving robot with retinomorphic vision and spike-based processing/closed-loop control[C]. *The 51st Annual Conf on Information Sciences and Systems*. Baltimore: IEEE, 2017: 17-23.
- [87] Mahadevuni A, Peng L. Navigating mobile robots to target in near shortest time using reinforcement learning with spiking neural networks[C]. *2017 Int Joint Conf on Neural Networks*. Anchorage: IEEE, 2017: 2243-2250.
- [88] Takase N, Botzheim J, Kubota N. Evolving spiking neural network for robot locomotion generation[C]. *2015 IEEE Congress on Evolutionary Computation*. Sendai: IEEE, 2015: 558-565.
- [89] Hulea M, Caruntu C F. Spiking neural network for controlling the artificial muscles of a humanoid robotic arm[C]. *The 18th Int Conf on System Theory, Control and Computing*. Sinaia: IEEE, 2014: 163-168.
- [90] Botzheim J, Kubota N. Spiking neural network based emotional model for robot partner[C]. *2014 IEEE Symposium on Robotic Intelligence in Informationally Structured Space*. Orlando: IEEE, 2014: 14-20.
- [91] de Azambuja R, Cangelosi A, Adams S V. Diverse, noisy and parallel: A new spiking neural network approach for humanoid robot control[C]. *2016 Int Joint Conf on Neural Networks*. Vancouver: IEEE, 2016: 1134-1142.
- [92] Kasabov N K, Doborjeh M G, Doborjeh Z G. Mapping, learning, visualization, classification and understanding of fMRI Data in the NeuCube evolving spatiotemporal data machine of spiking neural networks[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2017, 28(4): 887-899.
- [93] Kasabov N K. NeuCube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data[J]. *Neural Networks*, 2014, 52(4): 62-76.
- [94] Doborjeh M G, Wang G Y, Kasabov N K. A spiking neural network methodology and system for learning and comparative analysis of EEG data from healthy versus addiction treated versus addiction not treated subjects[J]. *IEEE Trans on Biomedical Engineering*, 2016, 63(9): 1830-1841.
- [95] Kasabov N K, Scott N M, Tu E, et al. Evolving

- spatio-temporal data machines based on the NeuCube neuromorphic framework: Design methodology and selected applications[J]. *Neural Networks*, 2016, 78: 1-14.
- [96] Tu E, Kasabov N K, Yang J. Mapping temporal variables into the NeuCube for improved pattern recognition, predictive modeling, and understanding of stream data[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2017, 28(6): 1305-1317.
- [97] Maliavko A A, Gavrilo V. Towards development of self-learning and self-modification spiking neural network as model of brain[C]. 2016 13th Int Scientific-Technical Conf on Actual Problems of Electronics Instrument Engineering. Novosibirsk: IEEE, 2016: 461-463.
- [98] Zhang T L, Zeng Y, Zhao D C, et al. HMSNN: Hippocampus inspired memory spiking neural network[C]. 2016 IEEE Int Conf on Systems, Man, and Cybernetics. Budapest: IEEE, 2016: 2301-2306.
- [99] Kasap B, van Opstal A J. A spiking neural network model of the midbrain superior colliculus that generates saccadic motor commands[J]. *Biological Cybernetics*, 2017, 111(3): 249-268.
- [100] Pedretti G, Milo V, Ambrogio S, et al. Memristive neural network for online learning and tracking with brain-inspired spike timing dependent plasticity[J]. *Scientific Reports*. 2017, 7(1): 52-88.
- [101] Li X, Liu H, Xue F. Liquid computing of spiking neural network with multi-clustered and active-neuron-dominant structure[J]. *Neurocomputing*, 2017, 243: 155-165.
- [102] Matsubara T, Torikai H. An asynchronous recurrent network of cellular automation-based neurons and its reproduction of spiking neural network activities[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2016, 27(4): 836-852.
- [103] Sarkar S T, Bhondekar A P, Macas M, et al. Towards biological plausibility of electronic noses: A spiking neural network based approach for tea odour classification[J]. *Neural Networks*, 2015, 71: 142-149.
- [104] Martinelli E, Polese D, Dini F, et al. An investigation on the role of spike latency in an artificial olfactory system[J]. *Frontiers in Neuroengineering*, 2011, 4(20): 16.
- [105] Morri A L, Gomez F, Jimenez F Z. A spiking neural network for real-time spanish vowel phonemes recognition[J]. *Neurocomputing*, 2017, 226: 249-261.
- [106] Silva M, Vellasco M, Cataldo E. Evolving spiking neural networks for recognition of aged voices[J]. *J of Voice*, 2017, 31(1): 24-33.
- [107] Dominguez M J, Jimenez A, Rois N A, et al. Multilayer spiking neural network for audio samples classification using SpiNNaker[C]. The 25th Int Conf on Artificial Neural Networks. Barcelona: IEEE, 2016: 45-53.
- [108] Cerezuola E, Jimenez A, Paz R, et al. Sound recognition systems using spiking and MLP neural networks[C]. The 25th Int Conf on Artificial Neural Networks. Barcelona: IEEE, 2016: 363-371.
- [109] Stomatias E, Soto M, Serrano G. An event-driven classifier for spiking neural networks fed with synthetic or dynamic vision sensor data[J]. *Frontiers in Neuroscience*, 2017, 11(28): 350.
- [110] Sun Q, Wu Q, Wang X, et al. A spiking neural network for extraction of features in colour opponent visual pathways and FPGA implementation[J]. *Neurocomputing*, 2017, 228: 119-132.
- [111] Ltaief M, Bezine H, Alimi A M. Training a spiking neural network to generate online handwriting movements[C]. The 16th Int Conf on Intelligent Systems Design and Applications. Porto: ISEP, 2016: 289-298.
- [112] Matsubara T, Torikai H. An asynchronous recurrent network of cellular automation-based neurons and its reproduction of spiking neural network activities[J]. *IEEE Trans on Neural Networks and Learning Systems*, 2016, 27(4): 836-852.
- [113] Cruz-Albrecht J M, Derosier T, Srinivasa N. A scalable neural chip with synaptic electronics using CMOS integrated memristors[J]. *Nanotechnology*, 2013, 24(38): 384011.
- [114] Aamir S A, Mueller P, Hartel A. A highly tunable 65-nm CMOS LIF neuron for a large scale neuromorphic system[C]. The 46th European Solid-State Device Research Conf. Lausanne: IEEE, 2016: 63-66.
- [115] Benjamin B, Gao P, McQuinn E, et al. Neurogrid: A mixed-analog-digital multichip system for large scale neural simulations[J]. *Proc of the IEEE*, 2014, 102(5): 699-716.
- [116] Cao Y, Chen Y, Khosla D. Spiking deep convolutional neural networks for energy efficient object recognition[J]. *Int J of Computer Vision*, 2015, 113(1): 54-66.
- [117] Qiao N, Mostafa H, Corradi F, et al. A reconfigurable online learning spiking neuromorphic processor comprising 256 neurons and 128 K synapses[J]. *Frontiers in Neuroscience*, 2015, 9(14): 141.
- [118] Furber S B, Galluppi F, Temple S, et al. The SpiNNaker project[J]. *Proc of the IEEE*, 2014, 102(5): 652-665.
- [119] Ma D, Shen J, Gu Z, et al. Darwin: A neuromorphic hardware co processor based on spiking neural networks[J]. *J of Systems Architecture*, 2017, 77(2): 51-53.