

控制与决策

Control and Decision

脉冲神经网络研究进展综述

胡一凡, 李国齐, 吴郁杰, 邓磊

引用本文:

胡一凡, 李国齐, 吴郁杰, 等. 脉冲神经网络研究进展综述[J]. *控制与决策*, 2021, 36(1): 1–26.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2020.1006>

您可能感兴趣的其他文章

Articles you may be interested in

[基于强化学习的倒立摆分数阶梯度下降RBF控制](#)

Reinforcement learning based fractional gradient descent RBF neural network control of inverted pendulum
控制与决策. 2021, 36(1): 125–134 <https://doi.org/10.13195/j.kzyjc.2019.0816>

[Actor–Critic框架下一种基于改进DDPG的多智能体强化学习算法](#)

A multi-agent reinforcement learning algorithm based on improved DDPG in Actor–Critic framework
控制与决策. 2021, 36(1): 75–82 <https://doi.org/10.13195/j.kzyjc.2019.0787>

[MADDPG算法经验优先抽取机制](#)

Multi-agent deep deterministic policy gradient algorithm via prioritized experience selected method
控制与决策. 2021, 36(1): 68–74 <https://doi.org/10.13195/j.kzyjc.2019.0834>

[结合注意力机制的循环神经网络复述识别模型](#)

Recurrent neural networks based paraphrase identification model combined with attention mechanism
控制与决策. 2021, 36(1): 152–158 <https://doi.org/10.13195/j.kzyjc.2019.0638>

[基于改进堆叠自动编码器的循环冷却水系统工艺介质温度预测控制方法](#)

Predictive control method of process medium temperature in circulating cooling water system based on improved stacked auto encoders
控制与决策. 2020, 35(12): 2835–2844 <https://doi.org/10.13195/j.kzyjc.2019.0694>

脉冲神经网络研究进展综述

胡一凡¹, 李国齐^{1†}, 吴郁杰¹, 邓磊²

(1. 清华大学类脑计算研究中心, 北京 100084; 2. 加州大学
圣塔芭芭拉分校 电子与计算机工程系, 美国 CA 93106)

摘要: 近年来,起源于计算神经科学的脉冲神经网络因其具有丰富的时空动力学特征、多样的编码机制、契合硬件的事件驱动特性等优势,在神经形态工程和类脑计算领域已得到广泛的关注. 脉冲神经网络与当前计算机科学导向的以深度卷积网络为代表的人工神经网络的交叉融合被认为是发展人工通用智能的有力途径. 对此,回顾了脉冲神经网络的发展历程,将其划分为神经元模型、训练算法、编程框架、数据集以及硬件芯片等 5 个重点方向,全方位介绍脉冲神经网络的最新进展和内涵,讨论并分析了脉冲神经网络领域各个重点方向的发展机遇和挑战. 希望本综述能够吸引不同学科的研究者,通过跨学科的思想交流与合作研究,推动脉冲神经网络领域的发展.

关键词: 脉冲神经网络; 神经网络; 类脑计算; 神经形态计算; 深度学习; 人工通用智能

中图分类号: TP301.6

文献标志码: A

DOI: 10.13195/j.kzyjc.2020.1006

开放科学(资源服务)标识码(OSID):



引用格式: 胡一凡,李国齐,吴郁杰,等. 脉冲神经网络研究进展综述[J]. 控制与决策, 2021, 36(1): 1-26.

Spiking neural networks: A survey on recent advances and new directions

HU Yi-fan¹, LI Guo-qi^{1†}, WU Yu-jie¹, DENG Lei²

(1. Center for Brain-Inspired Computing Research, Tsinghua University, Beijing 100084, China; 2. Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106, USA)

Abstract: Over the past few years, spiking neural networks (SNNs) originated from computational neuroscience have received extensive attention in the field of neuromorphic engineering and brain inspired computing due to their rich spatio-temporal dynamics, diverse coding schemes, and event-driven characteristics that naturally fit the neuromorphic hardware. Combining neuroscience-oriented spiking neural networks with computer-science-oriented artificial neural networks such as deep convolutional neural networks, is considered as a promising approach to artificial general intelligence. In this survey, we review the recent advances of spiking neural networks and focus on five major research areas, which we define as spiking neuron models, SNN algorithms, programming frameworks, datasets and neuromorphic hardware, and conclude with a broad discussion on the opportunities and challenges. The goals of this work are to provide an exhaustive review of spiking neural networks, attract researchers in different disciplines and motivate new works in this field through interdisciplinary exchange of ideas and collaborative research.

Keywords: spiking neural networks; neural networks; brain-inspired computing; neuromorphic computing; deep learning; artificial general intelligence

0 引言

神经网络被认为是当前人工智能发展的主要驱动力,其经历了几个发展阶段. 第 1 个阶段是感知机. 1958 年,美国神经学家 Rosenblatt^[1] 提出了可以模拟人类感知能力的机器,并称之为“感知机”,随后成功地在 IBM704 机上完成了感知机的仿真,并于 1960 年基于感知机实现了能够识别一些英文字母的神经计算机^[2]. 第 2 个阶段是基于联结主义的多层人工神经网络 (artificial neural network, ANN),其兴起于 20

世纪 80 年代中期^[3],并在 2006 年以后以深度卷积神经网络^[4] 引领了近十几年人工智能的发展. 然而,ANN 在生物学上是不精确的,缺少神经内部的动力学机制,不能较准确地模仿生物大脑神经元的运作机制. 近年来,起源于脑科学的脉冲神经网络 (spiking neural network, SNN) 被誉为新一代的神经网络^[5],以其丰富的时空领域的神经动力学特性、多样的编码机制、事件驱动的优势引起了学者的关注^[6]. 与此同时,脉冲神经网络的发展使得当前以脑科学成果为基础和启

收稿日期: 2020-07-21; 修回日期: 2020-08-31.

基金项目: 科技部重点研发项目 (2018YEF0200200, 2018AAA0102600); 国家自然科学基金项目 (61876215, 61836004); 清华大学国强研究院支持项目; 广州市重点研发计划项目 (20200703006).

†通讯作者. E-mail: liguoqi@mail.tsinghua.edu.cn.

发,以“人工通用智能”(图1)以及高效终(云)端智能解决方案为目标的一种新型计算范式——类脑计算(brain-inspired computing, BIC)^[7]呈星火燎原之势。

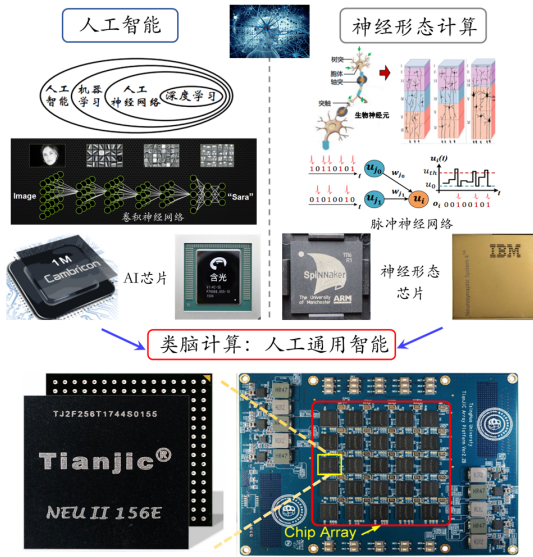


图1 目标“人工通用智能”的类脑计算呈星火燎原之势

2014~2020年是国内外类脑计算与人工智能系统发展大爆炸的几年:谷歌旗下人工智能公司DeepMind在Nature期刊连续报道了诸多成果^[8-10]; Science报道了以IBM TrueNorth^[11]为代表实现脉冲神经网络的神经形态芯片以及海德堡大学的SpiNNaker芯片^[12]与英特尔Loihi芯片^[13]。这些研究已经慢慢模糊了类脑计算与人工智能的严格分界线,促进了人工智能、神经形态系统和类脑计算领域的深入交叉、相互借鉴、扬长避短,共同促进智能系统的长足发展。2019年,清华大学研制的ANN/SNN异构融合天机芯登上Nature封面^[14],指出计算机科学导向的深度学习和神经科学导向的脉冲神经网络的交叉融合将是人工通用智能的发展方向。

当前,丰富的任务和数据集、友好的编程工具(如TensorFlow^[15]和Pytorch^[16])、以误差反向传播(back-propagation, BP)为代表的训练算法以及高效的训练平台(GPU)共同推进了ANN在各个深度学习领域的繁荣(如计算机视觉和自然语言处理),也推进了支持ANN的各种深度学习加速器的研究,如:中国科学院计算所的“寒武纪”系列芯片^[17-19]、Google公司的TPU芯片^[20]、清华大学的Thinker芯片^[21]以及美国麻省理工学院Eyeriss芯片等^[22]。与ANN领域的繁荣相比,SNN领域的研究仍然处于快速发展的早期阶段。当前,SNN领域的研究主要围绕神经元模型、训练算法、编程框架、数据集以及硬件芯片加速五大方向进行。

1) 神经元模型.

在神经元模型领域,积分泄漏发放(leaky

integrate-and-fire, LIF)模型及其变体^[23-26]、Izhikevich模型^[27]、Hodgkin-Huxley^[28]等模型都引起了学者的关注,并且已经有基于LIF神经元模型和Izhikevich模型的硬件电路出现^[29-30]。寻找兼具高生物可实现性、良好的网络学习能力以及超大规模集成电路构建能力的神经元模型是目前需要研究的问题。

2) 训练算法.

当前训练算法主要分为以脉冲时间依赖可塑性(spike-timing dependent plasticity, STDP)为代表的非监督训练算法^[31]和以ANN转SNN^[32]和时间空间维度的误差反传^[33]为代表的有监督训练算法。尽管训练算法众多,当前SNN在超大超深层的网络训练算法上,仍然面临脉冲信号如何编码、梯度消失、训练资源开销大甚至算法收敛性的问题。因此,研究超大超深层SNN网络的编码方式和匹配的训练算法是需要研究的问题。

3) 编程框架.

当前支持ANN训练的编程框架种类繁多,常见的有Theano^[34]、TensorFlow^[15]、Caffe^[35]、Pytorch^[16]、MXNet^[36]和Keras^[37]等。用户友好的编程接口、统一的数据流处理方式,使得一个初学者也能方便地搭建和训练ANN网络,这大大推动了ANN领域的发展。但是在SNN领域,目前仅有BindsNET^[38]、Spyketch^[39]等少数平台可以支持大规模SNN的构建与训练。搭建一个大的SNN网络仍需要编程者具有非常好的编程功底。因此,开发用户友好的编程工具以有效地部署大规模SNN对该领域的发展至关重要。

4) 数据集.

在数据集方向,SNN和ANN应该面向不同的数据集和任务。当前,ANN领域有种类繁多的数据集,包括MNIST^[40]、CIFAR-10^[41]、ImageNet^[42]等。而适合SNN的数据集由时空事件流构成,目前仅有Neuromorphic-MNIST、N-Caltech101^[43]、DVS-CIFAR10^[44]、DVS-Gesture^[45]等比较小的数据集。发展更大规模且更加契合SNN时空处理能力的数据集将成为SNN发展的驱动力之一。

5) 硬件芯片加速.

在硬件芯片加速方向,正因为SNN在硬件电路上具有超低能耗实现的优势,最近10年以曼彻斯特大学的SpiNNaker芯片^[46]、美国斯坦福大学Neurogrid芯片^[47]、IBM TrueNorth芯片^[11]、英特尔Loihi芯片^[13]、浙江大学达尔文芯片^[48]为代表,支持SNN硬件实现的神经形态芯片异军突起。2019年,清华大学研制出的ANN/SNN异构融合类脑天机芯

片^[14]更是能够同时支持以MLP、CNN和LSTM为代表的深度学习常用ANN和从计算神经科学发展而来的SNN.未来可能基于新型神经形态器件的发展成熟以及现有优势技术的交叉融合这两个方向,进一步探索以超低功耗模拟人脑的复杂计算系统的构建.

综上所述,SNN基于神经动力学的事件驱动机制在高效处理复杂、稀疏和嘈杂的时空信息方面取得了卓越的性能.事件驱动的通信在启用具有内存计算功能的高能效AI系统方面特别有吸引力,该系统将在普遍存在的智能领域中发挥重要作用.当前,SNN的研究仍在进行中,其神经元模型、训练算法、基准框架、编程工具和高效的硬件有望取得更大的进步.但是,由于相关的论文很多,出版速度极快,报道的方法大相径庭且缺乏系统的梳理,极大地妨碍了初学者获得正确的视野并以正确的方式入门.因此,本文希望通过对SNN领域的各个重点方向、发展的机遇和挑战进行全面系统的介绍梳理和分析比较,吸引不同学科的研究者,通过跨学科的思想交流和合作研究,推动SNN领域的快速发展.

1 脉冲神经网络的基本要素与生物学背景

在本节中,将介绍脉冲神经网络的基本要素及其相关的生物学背景,包括作为基本处理单元的神经元模型、作为学习与记忆基础的突触可塑性理论、神经元通信中脉冲序列的编码方式以及网络层面上各基本层的拓扑结构.以上几种要素共同构成了脉冲神经网络,并使其拥有区别于人工神经网络的特质.

1.1 神经元模型

神经元的典型结构主要包括树突(dendrite)、胞体(soma)以及轴突(axon)三个部分.其中:树突的功能是收集来自其他神经元的输入信号并将其传递给胞体;胞体起到中央处理器的作用,当接受的传入电流积累导致神经元膜电位超过一定阈值时产生神经脉冲(即动作电位);脉冲沿轴突无衰减地传播并通过位于轴突末端的突触(synapse)结构将信号传递给下一个神经元.

针对神经元工作时电位的动态特性,神经生理学家建立了许多模型,它们是构成脉冲神经网络的基本单元,决定了网络的基础动力学特性.其中影响较大的主要有H-H(Hodgkin-Huxley)模型、LIF(leaky integrate-and-fire)模型、Izhikevich模型和脉冲响应模型(spike response model, SRM),以下分别对其进行介绍.

1) Hodgkin-Huxley模型.1952年,Hodgkin和Huxley^[28]对乌贼巨轴突的电位数据进行研究,提出

了神经元电活动机制的理论数学模型(H-H模型)以及对应的电路模拟,在动作电位的产生及传播机制方面作出了奠基性贡献,并由此获得了1963年的诺贝尔生理学奖.H-H模型可表示为以下4个方程:

$$I = C_M \frac{dV}{dt} + \bar{g}_K n^4 (V - V_K) + \bar{g}_l (V - V_l) + \bar{g}_{Na} m^3 h (V - V_{Na}), \quad (1)$$

$$\frac{dn}{dt} = \alpha_n(V)(1 - n) - \beta_n(V)n, \quad (2)$$

$$\frac{dm}{dt} = \alpha_m(V)(1 - m) - \beta_m(V)m, \quad (3)$$

$$\frac{dh}{dt} = \alpha_h(V)(1 - h) - \beta_h(V)h. \quad (4)$$

其中: I 表示总膜电流密度; V 表示膜电位; C_M 表示单位面积膜电容; $g_K = \bar{g}_K n^4$, $g_{Na} = \bar{g}_{Na} m^3 h$, \bar{g}_l 分别表示钾离子、钠离子和其他离子的电导密度; V_K 、 V_{Na} 、 V_l 表示几种离子通道的反转电位; n 、 m 、 h 被假设为某几种与离子转运相关的粒子浓度,其对应的 α 和 β 象征该粒子向膜内、膜外移动的速率.在H-H模型中, Na^+ 、 K^+ 等离子通道与动作电位之间的产生机制被清晰地建模,并对生物神经组织实验记录的电位变化曲线有精确的近似.

2) LIF模型.早在1907年Lapicque就提出了Integrate-and-fire (I&F)模型,由于当时对动作电位的产生机理知之甚少,动作电位的过程被简化描述为:“当膜电位达到阈值 V_{th} 时神经元将激发脉冲,而膜电位回落至静息值 V_{reset} ”.模型则针对阈下电位的变化规律进行描述,其中最为简单且常见的是LIF模型^[23],即

$$\tau_m \frac{dV}{dt} = V_{rest} - V + R_m I. \quad (5)$$

其中: τ_m 表示膜时间常数, V_{rest} 表示静息电位, R_m 、 I 分别表示细胞膜的阻抗与输入电流.

LIF模型极大简化了动作电位过程,但保留了实际神经元膜电位的泄露、积累以及阈值激发这3个关键特征.在此基础上存在系列变体,例如二阶LIF模型^[24]、指数LIF模型^[25]、自适应指数LIF模型^[26]等,这些变体模型注重对神经元脉冲活动细节的描述,以一定的额外实现代价进一步补充并增强了LIF模型的生物可信度.

3) Izhikevich模型. Izhikevich模型^[27]希望能在具有生物合理性的H-H模型与计算高效的LIF模型之间得到折衷.在Izhikevich模型中采用了与LIF模型类似的动作电位描述方式,对于电势则简化H-H模型为如下两个微分方程:

$$\frac{dV}{dt} = 0.04V^2 + 5V + 140 - u + I, \quad (6)$$

$$\frac{du}{dt} = a(bV - u). \quad (7)$$

其中: u 为一膜恢复变量, 用于总体描述离子电流行为; a 、 b 分别用于调整 u 的时间尺度和关于膜电位 V 的敏感度. 通过参数的选择, Izhikevich 模型可以展示大脑皮层几乎所有已知神经元的放电模式, 但相比于 H-H 模型, 仿真所需要的计算开销约为百分之一^[49], 这也使得该模型在 2004 年能够支持 10 万个神经元规模网络的模拟, 结合突触可塑性规则进行神经元自组织特性的研究^[29].

4) SRM 模型. 脉冲响应模型是基于 LIF 模型提出的一种更具通用性的描述模型, 它也采用了 LIF 模型中动作电位的简化描述, 而与 LIF 模型相比, SRM 模型包含了对于不应期 (refractory period) 的模拟, 且采用了滤波器而非微分方程的形式描述电位变化^[50], 即

$$u(t) = \eta(t - \hat{t}) + \int_{-\infty}^{+\infty} \kappa(t - \hat{t}, s) I(t - s) ds; \quad (8)$$

$$\hat{t} \leftarrow t, u(t) \geq \theta \ \& \ \dot{u}(t) > 0; \quad (9)$$

$$\theta(t - \hat{t}) = \begin{cases} +\infty, & t - \hat{t} \leq \gamma_{\text{ref}}; \\ \theta_0 + \theta_1 e^{-\frac{t-\hat{t}}{\tau_\theta}}, & t - \hat{t} > \gamma_{\text{ref}}. \end{cases} \quad (10)$$

其中: η 描述神经元自身在 \hat{t} 时刻激发动作电位及随后连续的电位变化情形, κ 是对输入脉冲 I 的线性滤波器, γ_{ref} 保证了一个绝对不应期的实现, θ_i 与 τ_θ 则根据实际生理数据进行优化调整. 模型名称中“脉冲响应”这一点来源于 κ 与 η , 分别描述了神经元对于输入脉冲以及自身激发脉冲这两个行为的响应. 由于 κ 的选择可以较为广泛, SRM 相较于传统 LIF 神经元具有一定的通用性.

5) ANN 神经元模型. 作为对比, 本文在此也对人工神经网络中的基本神经元结构进行简单介绍. ANN 神经元模型保留了生物神经元多输入单输出的信息处理功能, 但对其阈值特性以及动作电位机制作了进一步的抽象简化, 其建模如下:

$$y_i^l = \varphi \left(\sum_{j=0}^{n^{l-1}} w_{ij}^{l-1} x_j^{l-1} \right). \quad (11)$$

其中位于 l 层的第 i 个神经元输出值 y_i^l 由前层 n^{l-1} 个神经元的输出 x_j^{l-1} 加权求和后, 经过非线性激活函数 $\varphi(\cdot)$ 计算得到. 相较于 SNN, ANN 神经元使用高精度的连续激活函数值而非离散脉冲序列进行通信, 舍弃了在时间域上的运算而仅保留逐层计算的空间域结构. SNN 虽然表达精度较低, 但保有更为丰富的神经元动态, 当前状态除了接受空间域中的输入外, 还天然地受到过去历史时刻的影响, 因此, SNN 可能具有更强的时空数据处理潜力. 此外, 由于阈值特性的

存在, SNN 的脉冲信号通常具有稀疏性, 且计算由事件驱动 (仅当脉冲到达时执行), 结合 0/1 的脉冲信号表达形式可以避免 ANN 中高昂的乘法计算代价, 表现出超低功耗的特性^[51], 这也促成了诸多神经形态硬件的诞生.

整体而言, 对生物神经元的建模在仿生度与计算代价之间存在着多个不同抽象层次的选择 (见图 2), 能够满足从单个神经元到网络层级的建模需要. SNN 神经元所具有的脉冲通信方式与动力学特征构成了与 ANN 之间最基本的差异, 并赋予其进行超低功耗计算和时序任务处理的潜力. 在大规模神经网络的层面上, 对单个神经元允许投入的算力较为有限, 这使得 H-H 模型等使用多变量、多组微分方程进行精确活动描述的复杂模型无法应用, 因此, 简化模型进而加速在计算机中的模拟与仿真过程是不可或缺的. 当前, 在脉冲神经网络中较为广泛地采用 LIF 模型, 正是因为其简洁的数学表达可以保证较低的实现代价, 然而, 较早的提出年代限制了 LIF 模型对神经元的了解, 致使它在生物可信度上有所欠缺. 因此, 在保证大规模集成电路构建能力的基础上, 寻找兼具良好学习能力与高生物可信度的神经元模型, 仍然是目前需要研究的问题.

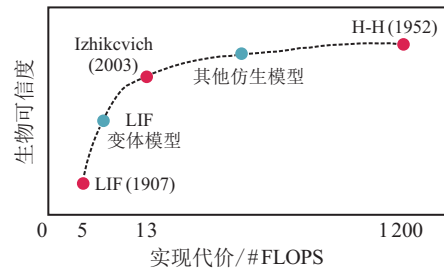


图 2 不同神经元模型生物合理性与实现代价^[49]

1.2 突触可塑性的生物学基础

大脑是以何种方式在事件的因果性上建立起联想与回忆的? 这是长久以来存在于哲学、心理学以及神经科学领域的一个核心问题, 也是近年来计算神经科学研究的一个重点. 亚里士多德首先在《论灵魂》(De Anima) 中介绍大脑为一块白板, 这与他的导师柏拉图形成鲜明对比, 后者在《斐多篇》(Phaedo) 中称人的心灵为先天形成, 随后加入身体. 在近代, 对大脑的全新观点通常被认为是 17 世纪英国哲学家 Locke 提出的^[52], 他认为, 出生时人并没有任何天生的想法, 经验完全塑造了大脑, 后天的学习塑造了人. 在知晓突触的存在之前, Bain^[53] 就指出细胞间的物理连接是支持学习和记忆进行的基础. 随后神经生物学的系列发现让这些猜想逐渐落到实处. 20 世纪上半叶见证了许多里程碑式的研究, 这些研究让人们对化

学突触和神经递质的存在有了更为清晰的认识,对神经元信息流的理解以及学习、记忆和行为的看法产生了重大影响^[54]. 到20世纪中期,人们已经清楚地知道信号在突触中的方向性流动,所有输入经由胞体整合,一旦达到阈值,信号便以动作电位的形式沿轴突开始传播(如1.1节所述).

1949年,Hebb(赫布)^[55]在《The Organization of Behavior》一书中总结了早期突触可塑性的相关研究结果,形成了一个对后续研究产生巨大影响力的神经记忆与学习机制假设:当A细胞重复或持续地参与激活B细胞时,两个细胞间会发生某种生长过程或代谢变化,使得A细胞激活B细胞的效率得到提高.换言之,同时激活的细胞将彼此相连^[56]. 随后一系列关于海马体中长时间增强作用(long-term potentiation, LTP)与长时间抑制作用(long-term depression, LTD)两种可塑性的实验发现为赫布假说提供了支撑. 其中, Malinow^[57]在海马体切片中分离出4对单突触连接的CA3-CA1神经元,随后通过同时激发突触前后神经元成功诱导了LTP产生,将LTP现象由神经元群的协同表现缩小定位至单个突触结构中. 这是第一项真正针对突触结构进行的可塑性研究,也构成了对赫布假说的严格检验.

由于记忆与联想存在时间属性,神经元对其学习显然需要引入时间的概念,但早期的实验研究很少有直接考察时间在可塑性中的作用. Markram等^[58-59]首先针对单突触连接中前后两神经元发放脉冲的相对时间进行了研究. 这项研究揭示了在10ms的时间差异下,突触前-突触后的神经元发放顺序会引起LTP,而突触后-突触前的神经元发放顺序将导致LTD. “同时”不一定导致连接增强,时间上存在的逻辑将会决定突触改变的方向和幅度^[54]. 这种现象后来被称作脉冲时间相关的突触可塑性(spike-timing dependent plasticity, STDP)^[60]. 随后,STDP的相关规则得到广泛发现并进一步丰富,其中Bell等^[61]发现了倒置的STDP现象. 多项研究发现,在大脑皮层部分区域存在LTD窗宽度相较于LTP窗大得多的不对称现象^[62-64],覆盖整个STDP时间窗的详细突触变化曲线也得到绘制^[65-66]. 此外,除了在切片培养神经元中进行的STDP研究,完整大脑中的实验也提供了STDP存在的证据,包括在鼠、猫以及人类大脑中观察到的结果^[67-71].

赫布学习与STDP规则成功揭示了突触结构内权重的修改过程,但它们不足以解释突触个体的改变如何协调以实现神经系统的整体目标. 学习不能是短视、盲目的,因此如果要理解大脑的学习,则需要

进一步揭示在整个网络中协调可塑性的原则. 神经调质(neuromodulator)作为突触可塑性中除了前后神经元活动外的第3种影响因素,它的一系列发现丰富了原有的赫布规则. 目前,人们已经知晓一些神经调质与奖励信号之间存在密切的联系,例如多巴胺神经元能够预估、检测奖励并相应地发出报警或激励信号^[72],乙酰胆碱与去甲肾上腺素在处理新奇事物时分泌会有所增加^[73]. 实验也证实了这些神经调质能够调整它们所到之处的突触可塑性^[74],作用形式可能是作为使能的门控信号^[75],或是对于STDP窗口的形状和极性进行调整,同时多种神经调质在组合下也可发挥不同的作用^[76],这在在大脑不同区域实现各异的可塑性提供了生理基础.

总体来看,神经调质作为突触前后神经元活动外的第3种影响因素存在,可将其^[74]描述为

$$\dot{w} = M \times H(\text{pre}, \text{post}). \quad (12)$$

其中: M 表示神经调质的影响,pre表示突触前神经元的输出脉冲(序列),post表示突触后神经元的脉冲响应或电位响应, H 表示突触活动在赫布规则下造成的预期影响或称为资格痕迹(eligibility trace)的可塑性变化暂态标记.

赫布规则与STDP规则具有局部学习的特征,是脉冲神经网络模型中无监督算法模拟的重要对象,而神经调质的引入为微观上的突触结构变化与宏观上生物体适应性行为间建立了联系的桥梁,突触可塑性的变化因而具有了目的和方向. 在整个神经网络功能优化的全局角度上,这样的三因子规则(three-factor learning rule)相较于原始赫布规则引入了监督性学习的特点,有力地推动了大脑深层网络的高效学习.

1.3 编码方式

生物对周围环境的持续感知、对自身内部环境稳态的平衡调节以及感官刺激下的行为反应,最终都通过神经元短暂的脉冲序列实现信息的传递. 尽管信号通过动作电位与突触结构在神经元之间传播,但信息以何种形式蕴藏在具有一定时空模式的脉冲序列中尚不明了,这也是神经编码(neural coding)这一领域期待解决的问题.

目前较为常见的神经编码方式主要包括频率编码(rate coding)、时间编码(temporal coding)、bursting编码和群编码(population coding)等(见图3). 具体的脉冲在持续时间、振幅或形状上都可能有所不同,但在神经编码研究中,它们通常被视为相同的定型事件.

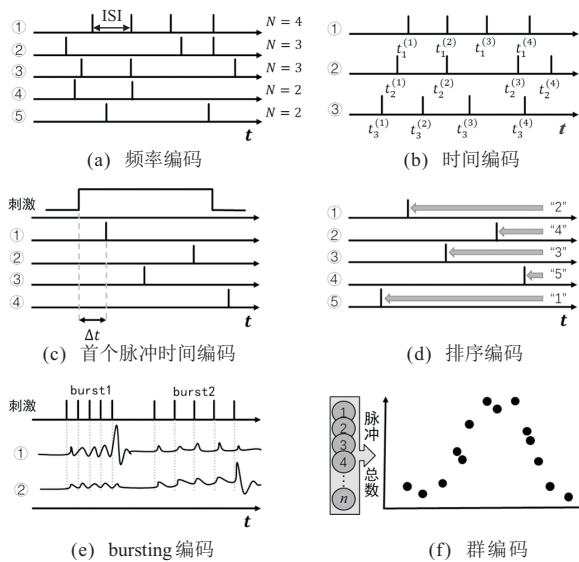


图3 编码方式示意图

频率编码主要考察脉冲发放率 (firing rate), 即神经元发放的脉冲数量在其所对应记录时间上的平均值. 刺激的强弱程度可由神经元发放脉冲的频率反映, 强烈的刺激将导致更为高频的脉冲序列^[77], 而序列内部存在的时间结构, 例如内部脉冲间隔 (interspike interval, ISI) 不被考虑. 频率编码被视为是对神经元输出的一种量化衡量, 在脉冲神经网络的深度学习领域, 特别是在经由训练好的 ANN 网络向 SNN 转换的相关工作中, 可将脉冲发放率与 ANN 中连续的输出值等价, 这使得频率编码得到了大量的使用. 相对于频率编码, 时间编码更加注重在时间结构上的差异, 除了完整的脉冲时间模式 (temporal pattern) 之外, 从接受刺激到发放首个脉冲的时间以及脉冲之间的时序逻辑都被认为存在编码重要信息的可能, 其中前者被称为首个脉冲时间编码 (time-to-first-spike coding)^[78], 后者被称为排序编码 (rank order coding)^[79], 如图 3(c)、图 3(d) 所示. 频率编码的概念在早期被广泛接受, 但一些研究表明, 其忽略时间结构仅计算脉冲数量的方式不利于信息的高效传输, 相较于时间编码可能需要更长的记录时间或更多的神经元参与^[80-81].

Bursting 是广泛存在于大脑中的一种神经元活动模式, 指神经元在某段时间内密集、快速地激发脉冲, 随后进入较长时间静息的行为. 早期人们通常认为 bursting 包含多个脉冲仅仅是为了增加信息传递的可靠性, 但 2003 年, Izhikevich 等^[82] 的发现揭示了突触后神经元存在的阈下膜电位共振现象, 这导致不同神经元针对 bursting 频率产生特异性选择, 并可能为神经元间进行选择通信提供了有效的编码机制.

与更多关注神经元个体的频率编码、时间编码等不同, 群编码认为刺激产生的神经元信息是由多个神经元的联合活动表征的. 其中较为著名的是由 Georgopoulos 等^[83] 提出的群体向量模型 (population vector model). 在该模型中, 尽管灵长类大脑运动皮层中手臂区域的单个神经元仅对三维空间中的某一方表现出特异反应, 但实际的精确运动方向能够以各个神经元的活动强度为依据, 对特异方向向量进行加权求和得到. 稀疏编码是群编码中较为特殊的一种, 强调信息仅由较大群体中共同活跃的小部分神经元表达, 且神经元数量可能远远大于输入信息的维度. 这种过完备 (overcomplete) 的表达方式具有记忆容量与能耗上的理论优势和哺乳动物大脑皮层的实验观测证据^[84], 其相关理念也在计算机视觉等机器学习领域得到广泛应用^[85].

当前对于神经编码的具体方案仍未盖棺定论, 而实际上在大脑中采用不同编码的神经元群也可能同时存在并互相配合, 从而为时间信息的充分感知提供神经基础^[86]. 神经编码方案可能在不同情景和大脑的不同区域中表现不同, 现有的编码方案则对其提供了不同角度的描述. 相较于目前 SNN 中通常预设单一编码方案的受限情形, 更为理想、通用的 SNN 应能支持不同编码的混合运用, 并灵活地借助不同编码的各自优势取得在任务表现、时延与功耗上的进一步优化.

1.4 拓扑结构

在灵长类动物中, 腹侧视觉通路 (ventral visual pathway) 被认为是涉及对象视觉识别的大脑区域. 前期区域 (如 V1 区) 的神经元对视网膜图像相对简单的局部空间特征作出反应, 而后期区域 (如 V4 区和下颞叶皮层) 对更大的视觉区域上更为复杂的特征作出反应. 沿着腹侧视觉通路前进, 神经元逐渐以更为鲁棒的方式来表征视觉物体, 而不受限于确定视角与光照条件下的固定外观^[87]. 目前, 先进的人工神经网络就是受到人脑层级启发的深度结构, 使用多层结构对潜在特征进行提取与表征.

用于构建神经网络的基本层拓扑结构 (见图 4) 主要包括全连接层、循环层以及卷积层. 它们对应形成的神经网络分别是多层感知机 (multilayer perceptron, MLP)、循环神经网络 (recurrent neural network, RNN) 和卷积神经网络 (convolutional neural network, CNN). RNN 和 MLP 分别只包括具有或不具有层内循环连接的堆叠全连接层. 与 MLP 和 RNN 中的一维特征不同, 卷积层使得 CNN 能够完成面向二维特征的处理. 卷积层中的每个神经元只接收来自

前一层特征映射(feature map)局部感受野(receptive field)的输入,并重复使用卷积核权重进行局部的二维卷积计算.此外,CNN也使用池化层调整特征映射的尺寸,并使用全连接层构建最终的分层器.

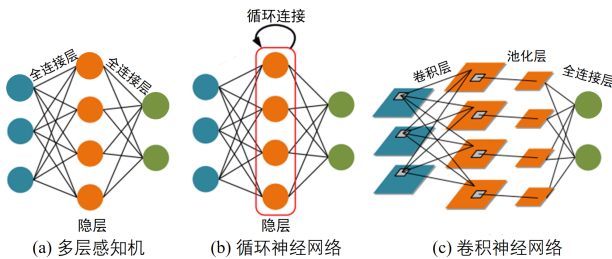


图4 神经网络基本层拓扑结构

2 神经形态数据集

数据集的发展对推动神经网络技术进步发挥了至关重要的作用.在传统人工神经网络领域,图像、语音等领域数据集的不断扩充以及任务场景的复杂化,对ANN的模型性能提出了挑战,也从另一方面推动着ANN技术的发展.这一特性对脉冲神经网络也是相同的.由于脉冲神经网络的研究落后于ANN,发展适合于脉冲神经网络的数据集是未来的发展方向.其中,受神经形态视觉传感器成像机制启发所产生的数据集被认为是目前最适合脉冲神经网络应用的一类数据集.

神经形态视觉传感器是受生物视觉处理机制启发,捕捉视野中的光强变化并产生异步时间流的一类传感器.具有代表性的神经形态视觉传感器有动态视觉传感器(DVS)及动态主动成像传感器(DAVIS)^[88]等.该类传感器捕捉并记录视野中的变化信息,根据信息变化的方向不同(增加或减少),记录正负两种变化方向的脉冲串信息.此类动态传感器主要关注视野中的变化特征而自动忽略背景中静态无关的信息,这也使得该类传感器具有低延迟、异步、稀疏的相应特性,进而在诸多领域具有非常广阔的前景,例如光流估计、目标跟踪、动作识别等.受此启发所产生的数据集被称为神经形态数据集,数据一般由四维向量组成 (x, y, t, p) .其中: (x, y) 记录成像的拓扑坐标, t 记录脉冲产生的时间信息(精确到 μs), p 记录脉冲的极性(由正或负方向生成).

神经形态数据集的以下特征使其适合用于脉冲神经网络的基准测试:1)脉冲神经网络可以自然地处理异步、事件驱动信息,使其与神经形态数据集的数据特征非常契合;2)神经形态数据集中内嵌的时序特征(比如精确的发放时间及帧之间的时序相关性)为展现脉冲神经元利用时空动力学特征处理信息的能力提供了很好的平台.

按照数据集构建方式的不同,目前神经形态数据

集可分为以下3类.

第1类是实地场景采集而得的数据集,其主要通过神经形态传感器直接捕捉而生成无标签的数据.这类数据集生成简单,贴近实际应用场景,例如可用于追踪和检测的数据集^[89-90](见图5)、用于光流估计的神经形态数据集^[90-91]、用于3D场景重构的数据集^[92-93]、用于手势识别的数据集^[45]等.得益于神经形态视觉传感器快速、高动态频率的特性,这类数据集对发展特定神经形态传感器的应用有着重要帮助.

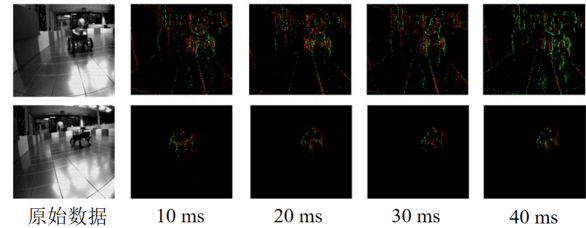


图5 PRED-18数据集示意

第2类数据集是转换数据集,它主要由带标签的静态图像数据集通过神经形态传感器实拍生成.不同于第1类直接对实际场景采集而成的数据集,此类数据集主要由已被广泛研究的、用于传统非脉冲任务的数据集转换得到(见图6),如N-MNIST、CIFAR10-DVS.为了生成此类数据集,研究者一般先固定一张静态的图片(如一张打印好的手写数字体图案),再用动态传感器沿指定方向平移产生相应数据的脉冲事件流版本.由于转换原始图片的特征及标签是已知的,使得研究者可以较为轻松地得到该类数据集的标签信息.转换数据集与原始广泛采用的数据集具有一定的特征相似性,此类转换数据集更易于使用和易于评估,因此,这类转换数据集也是目前脉冲神经网络领域使用最为广泛的数据集.

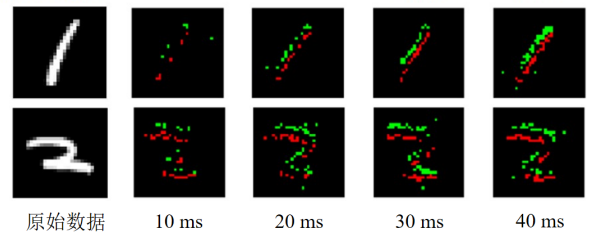


图6 N-MNIST数据集示意

第3类是生成数据集,它主要利用带标签数据,通过算法模拟动态视觉传感器特性,进而生成得到.由于动态传感器主要捕捉视频流中的动态信息,这一过程可以间接地利用相邻帧的差分等信息得到.因此,第3类数据集直接从已有的视频流(或者图片)信息中,通过特定的差分算法^[94]或基于相邻帧的生成算法,生成神经形态数据集的版本^[95-96].如Yang等^[96]训练一个用于数据生成的卷积神经网络,通过输入相邻帧和目标精确时刻的信息,生成脉冲事件流

版本的特定帧信息. 这一做法可以避免数据采集所需要的大量工程实验, 可以快速地生成研究所需要的含标签的特定场景信息. 然而, 由于传统视频流帧数有限, 模拟得到的数据集帧数无法达到实际生成的数据集要求, 同时也无法高分辨地模拟数据的事件驱动特性.

尽管上述3类数据集的研究仍在持续发展中, 但这3类数据集存在各自的局限性. 例如, 由于使用者对第1类数据集的预处理方式不统一(时间分辨率、图片压缩尺度等), 所报道的结果目前很难被公平地比较; 第2、第3类数据集主要是由原始数据二次转换生成, 其数据很难表达丰富的时序信息, 因此, 无法充分利用脉冲神经网络的时空处理特性等. 可以说目前对神经形态数据集的研究还尚在起步阶段, 脉冲神经网络领域仍缺乏公认的、基准性的测试集. 因此, 发展规模更大、功能更契合(利用脉冲神经网络时空处理能力和数据事件驱动特性)的数据集是今后的一大发展方向.

3 脉冲神经网络学习算法

人工神经网络的学习是以数据为基础, 面向特定任务进行网络关键参数的调整与优化的过程, 学习算法无疑在其中扮演着至关重要的角色. 与误差反向传播相结合的梯度下降(gradient descent)算法是目前人工神经网络优化理论的核心, 其系列变体由朴素SGD^[97]逐渐演化到ADAM^[98]、AMSGrad^[99]等算法. 此外, 由于批归一化(batch normalization, BN)^[100]和分布式训练(distributed training)^[101]等手段的加入, 使得大规模、高性能的人工神经网络得以实现, 并广

泛应用于人工智能领域的实际场景中.

相比之下, 当前脉冲神经网络领域尚不存在公认的核心训练算法与技术. 在生物合理性与任务表现间存在不同的侧重度, 以及网络采用的神经元模型和编码方式各异, 均造成了训练算法的多样化. 依据训练过程中是否使用标签信息, 可以将其概括性地划分为无监督学习与有监督学习两类. 其中: 无监督学习主要包括基于第1.2节中Hebb与STDP两种突触可塑性规则的仿生学习算法; 有监督学习则可以进一步划分为初期有监督学习算法和深度有监督学习算法. 有监督学习面临的主要困难在于BP算法在脉冲神经网络中的应用, 它的不应症主要来源于两个方面. 一方面, BP算法自身缺乏生物合理性, 突触信息传递的方向性使得前向传递和可能存在的反馈路径在生理上是分离的, 而目前并不存在已知的方式来协调二者以实现反向传播中对前传权重的获取, 这被称为weight transport问题^[102]; 另一方面, 脉冲神经网络中传递的信号为不可微的离散二值信号, 脉冲形式的激活函数给基于梯度的优化算法的直接应用造成了困难. 因此在21世纪初期, 适应脉冲形式计算的有监督算法得到了广泛研究并发展得相当丰富. 随后, 以2012年作为一个标志性的分水岭, 该年AlexNet在大规模视觉识别挑战赛ILSVRC中拔得头筹, 使得深度学习方法在传统机器学习领域脱颖而出. 脉冲神经网络的发展亦受到深度学习思想的影响, 在训练算法、网络规模以及应用场景的追求上都与前期产生了一定的差异.

脉冲神经网络学习算法概览如图7所示.

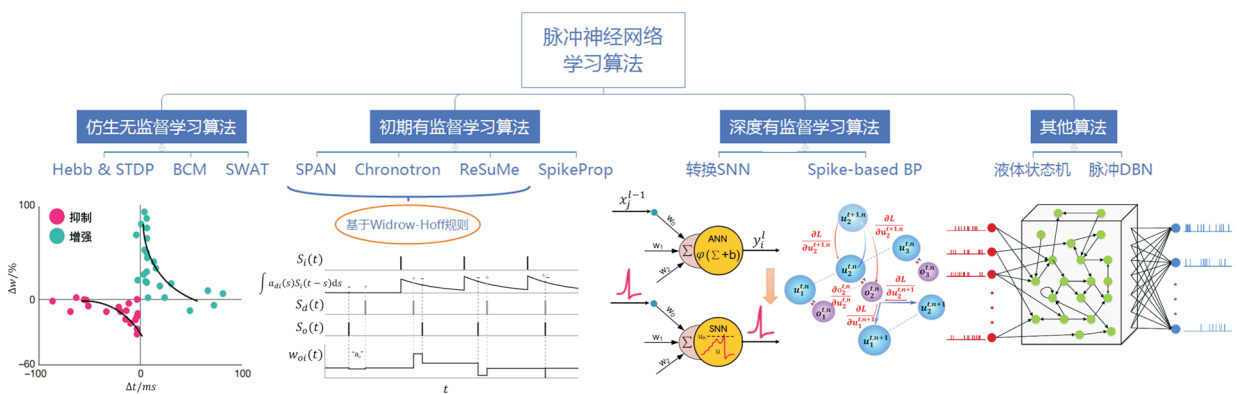


图7 算法概览

3.1 仿生无监督学习方法

本小节将介绍具有生物神经网络结构与突触可塑性规则的脉冲神经网络工作进展. 如前所述, STDP作为一种突触可塑性规则在脉冲神经网络训练中得到了广泛的应用, 常见的描述公式^[31]如下式所示:

$$\Delta w_{ij} = \begin{cases} a^+ \exp\left(\frac{t_j - t_i}{\tau^+}\right), & t_j - t_i > 0; \\ a^- \exp\left(\frac{t_i - t_j}{\tau^-}\right), & t_j - t_i \leq 0. \end{cases} \quad (13)$$

其中: Δw_{ij} 表示权重改变量, t_i 与 t_j 分别表示突触前后神经元的脉冲发放时间, τ^+ 与 τ^- 是影响STDP时

间窗尺度的时间常数, a^+ 与 a^- 分别对应了 LTP 与 LTD 的两种权重变化方向. 在网络层级的建模时, Δw_{ij} 随时间呈指数衰减的特性可能会得到进一步的简化, 以实现计算开销与仿真程度的折中. 式(13)为实验^[103]中观察到的权重更新规则提供了合理的数学近似.

网络层级学习功能的实现是每个神经元局部动态互相结合的成果. 在单个神经元层面, Guyonneau 等^[104]发现在 STDP 规则下的突触仿真中, 传入神经元群激发的固定脉冲序列能够在单个突触后神经元中引起快速且有选择性的识别与响应. Diehl 等^[105]展示了一个仅由单层兴奋性神经元以及与其一一对应的抑制性神经元层构成的双层脉冲神经网络, 由输入到兴奋神经层之间的连接由 STDP 训练, 抑制神经层的存在可以保证侧向抑制以及神经元间的彼此竞争. 完成无监督训练后, 兴奋性神经元具有了对于输入特征选择性响应的能力, 在 MNIST 数据集上获得 95% 的准确率. Masquelier 等^[31,106]利用 STDP 规则设计了一个模仿大脑中腹侧视觉通路的前馈型脉冲神经网络, 当反复为网络呈现图像时, 神经元将逐渐形成对于同类图像中共同特征的选择性并逐渐缩短后层神经元激发所需的脉冲时延, 最终包含图像重要特征信息的脉冲将会快速发放并可用于分类任务.

在实际视觉场景中, 人对物体的观察会受到视角、大小、变形、光照条件、物体遮挡以及不同背景等诸多因素的影响, 但视觉系统在绝大多数情况下仍能快速准确地进行识别, 随后的工作^[107]证实 STDP 规则构建的无监督学习网络在多尺寸、多视角的数据集 3D-Object^[108] 以及 ETH-80^[109] 上表现出优于 HMAX^[110] 以及深度卷积网络 AlexNet^[4] 的性能. 此外, 受到深度卷积网络在计算机视觉领域取得的系列成果的启发, 在脉冲卷积网络上应用 STDP 规则的研究也见诸报道^[111], 与之前模型大多仅包含单层 STDP 训练权重不同, 该模型中包含多层适用 STDP 规则的卷积层用于特征提取, 最后一层的神经元不激发脉冲, 而将最终的膜电位作为输出接入支持向量机 (support vector machine, SVM) 中进行分类器有监督训练, 在 MNIST 数据集上达到 98.4% 的准确率水平.

除了赫布规则与 STDP 规则之外, BCM 规则 (bienenstock-cooper-munro rule)^[112] 是 1982 年依据视觉皮层实验提出的另外一种突触可塑性规则. 原始的赫布规则由于并不包含突触连接的衰减机制或是增强阈值, 所构建的模型并不具有稳定性. BCM 规则在原始赫布规则基础上假定神经元具有决定突触权重改变方向的阈值, 且阈值动态适应于神经元的历史活动, 使得连接权值最终能够达到稳态. 其中

神经元活动主要由脉冲发射率衡量, 因而在 BCM 模型中多使用频率编码. 随后的 SWAT (synaptic weight association training)^[113] 结合了 BCM 规则中可变阈值的特点, 为 STDP 可塑性窗口的形状施加了负反馈调节, 增强了脉冲神经网络训练过程中的稳定性. 在利用 STDP 和 BCM 训练输出层的同时, 利用网络中的隐藏层作为频率滤波器, 从输入脉冲时间序列中提取特征用于分类任务.

3.2 初期有监督学习算法

有监督学习算法通常针对带有标签的数据, 学习从输入到标签的映射函数关系. 在脉冲神经网络中, 这种标签被编码为具有时序特征的目标脉冲序列, 算法需要学习网络中的连接权重以实现针对不同输入的特异性识别, 并转换输出目标脉冲序列.

Bohte 等^[114] 在 2002 年首次提出了使用误差反向传播训练 SNN 学习目标脉冲的有监督算法 SpikeProp. SpikeProp 使用时间编码和 SRM 模型, 通过线性处理神经元接收的输入和由此产生的脉冲发放时间的关系, 绕过在阈值处产生的不连续性问题, 最小化由目标脉冲与实际脉冲时间之差定义的误差函数, 最终训练得到的网络表明能够对非线性可分数数据集进行正确分类. 然而, 最初的 Spikeprop 也存在着如神经元长时间静息后突触权重无法保持以及单次仿真过程中每个神经元仅允许输出一个脉冲的限制, 后者实际上明显降低了神经元信息表达可能具有的丰富性. 随后的工作在此基础上进行了改进, 提出了 Multi-SpikeProp 算法, 将神经元扩展至使用多个脉冲传递信息的形式^[115], 并在 iris 和 EEG 两个分类问题上相较原始 SpikeProp 取得了较大优势.

ReSuMe^[116] 代表了一种与经典的 Widrow-Hoff 算法^[117] 相类似的方法, 该算法使用基于速率的神经元模型, 在不需要显式梯度计算的情况下, 最小化输出与目标信号之间的差距. ReSuMe 学习的目标是在神经网络上构建所需的输入-输出特性, 即网络接受给定的输入能产生符合预期的脉冲序列. ReSuMe 结合了赫布规则以及远程监控 (remote supervision) 两个概念, 其中远程监控指目标信号并不直接传递给受训练的神经元, 而是与突触前神经元共同决定突触权重的变化. 在 Widrow-Hoff 规则中针对输入 x_i , 期望输入 y_d 以及实际输出 y_o , 权重更新表达为

$$\Delta w_{oi} = \alpha x_i (y_d - y_o). \quad (14)$$

考虑引入时间维度以及脉冲形式后, ReSuMe 的学习规则可重写为

$$\frac{dw_{oi}(t)}{dt} = [S_d(t) - S_o(t)] \left[a_d + \int_0^\infty a_{di}(s) S_i(t-s) ds \right]. \quad (15)$$

其中: $S_i(t)$ 、 $S_o(t)$ 、 $S_d(t)$ 分别表示输入、输出以及期望脉冲序列, a_d 用于调节权重变化的平均强度, 而起到实际精确调节脉冲时间功能的是包含 $a_{di}(s)$ 的赫布规则项, s 表示产生输出时相对输入的时间间隔. 该学习规则的作用机理可由图8直观表现, 期望信号主要与输入相互作用而与实际输出相对独立, 不对输出神经元有直接影响, 由此体现出 ReSuMe 中“远程监督”的特性. ReSuMe 已被证明能够有效地在特定输入下学习产生预期信号, 并且输出信号可用于分类任务.

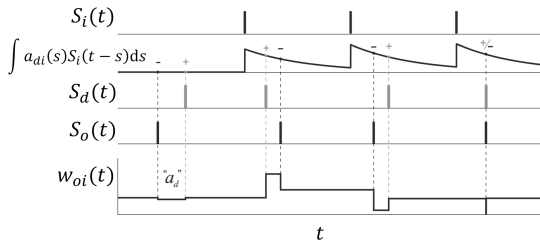


图8 ReSuMe 规则示意^[116]

与 ReSuMe 的目标一致, *chronotron*^[118] 作为一种神经网络可以产生在时间上精确符合预期的脉冲序列的算法. *tempotron*^[119] 作为其前身受限于单个脉冲输出的限制, 无法编码足够的信息在多个 *tempotron* 形成的网络之间传递. *chronotron* 针对脉冲序列的差异, 采用 VP 距离 (vector-purpura distance) 作为损失函数的构建依据, 并提出了 E-learning 以及 I-learning 两种学习规则, 其中前者基于梯度下降方法, 后者则相对更具有生物合理性. *chronotron* 中损失函数定义如下:

$$E = \sum_{t^f \in \mathcal{F}^*} u(t^f) + \sum_{\tilde{t}^f \in \tilde{\mathcal{F}}^*} (\theta - u(\tilde{t}^f)) + \frac{\gamma_d}{2\tau_q^2} \sum_{\substack{t^f \in \mathcal{F} - \mathcal{F}^* \\ \tilde{t}^g \in \tilde{\mathcal{F}} - \tilde{\mathcal{F}}^*}} (t^f - \tilde{t}^g)^2. \quad (16)$$

其中: \mathcal{F} 和 $\tilde{\mathcal{F}}$ 分别表示实际脉冲序列和目标脉冲序列, 序列中第 f 个脉冲分别以时间 t^f 、 \tilde{t}^f 表示, \mathcal{F}^* 和 $\tilde{\mathcal{F}}^*$ 分别表示实际脉冲需要删除的以及目标脉冲需要加入的脉冲序列, u 衡量膜电位, γ_d 和 τ_q 为两个正参数. E-learning 权重更新规则如下:

$$\Delta w_j = \gamma \left[\sum_{\tilde{t}^f \in \tilde{\mathcal{F}}^*} \lambda_j(\tilde{t}^f) - \sum_{t^f \in \mathcal{F}^*} \lambda_j(t^f) + \frac{\gamma_r}{\tau_q^2} \sum_{\substack{t^f \in \mathcal{F} - \mathcal{F}^* \\ \tilde{t}^g \in \tilde{\mathcal{F}} - \tilde{\mathcal{F}}^*}} (t^f - \tilde{t}^g) \lambda_j(t^f) \right]. \quad (17)$$

其中: γ 表示学习率, λ 表示由该突触输入导致的突触后电位变化. 由于脉冲的增删实际上仍为一个离散过程, E-learning 采用分段梯度下降对损失函数进行最小化. I-learning 在此基础上参照 ReSuMe, 调整权

重规则为局部可计算的突触电流形式, 即

$$\Delta w_j = \gamma \text{sign}(w_j) \left[\sum_{\tilde{t}^f \in \tilde{\mathcal{F}}} I_j(\tilde{t}^f) - \sum_{t^f \in \mathcal{F}} I_j(t^f) \right]. \quad (18)$$

chronotron 表现出在 200 ms 时长的不同输入序列下以亚毫秒级的时间误差学习目标脉冲序列的能力.

SPAN (spike pattern association neuron)^[120] 也是以有监督的方式训练神经元学习所需输入-输出脉冲映射关系的一种算法. 与 ReSuMe 不同, SPAN 将脉冲信号与特定的核函数进行卷积, 转换为模拟信号, 使得 Widrow-Hoff 规则可以在转换后的信号上直接应用以调整突触权重. 文中采用的核函数如下所示:

$$\tilde{s}(t) = \sum_{t^f} \kappa(t - t^f) = \sum_{t^f} e\tau^{-1}(t - t^f) e^{-\frac{t-t^f}{\tau}} H(t - t^f). \quad (19)$$

其形式多用于突触后电位变化的模拟, 具有的效果实际上是藉由兴奋性突触后电位的连续变化替代单次脉冲进行计算.

修改后的 Widrow-Hoff 规则可写为

$$\Delta w_i = \lambda \int_0^\infty \tilde{x}_i(t) (\tilde{y}_d(t) - \tilde{y}_{out}(t)) dt, \quad (20)$$

其中式(14)的各个变量转换为核函数下的连续信号 \tilde{x}_i 、 \tilde{y}_d 、 \tilde{y}_{out} .

3.3 深度有监督脉冲神经网络

本节集中讨论具有深度脉冲神经网络实现能力的有监督学习算法, 主要包含以转换 SNN 为代表的间接监督性学习算法和以时空反向传播为代表的直接监督性算法. 其中“间接监督性”体现在有监督信号仅在向 SNN 转换前的原始模型中进行训练这一点上, 直接监督性学习则指在 SNN 结构中直接适用的有监督学习算法.

转换 SNN (ANN-converted SNN) 是在神经形态硬件崭露头角后, 为了在已发展出的深度学习成果上进一步利用事件驱动特性的低能耗优势, 从 ANN 的视角出发为 SNN 的训练提供一种替代做法. 转换 SNN 的基本理念是在使用 ReLU 函数的 ANN 网络中, 连续的激活函数值可以由 SNN 中频率编码下的平均脉冲发放率近似^[121-122], 在完成原始 ANN 训练后, 再通过特定手段将其转换为 SNN. 实质上, 转换 SNN 的训练依赖的仍是在 ANN 中进行的反向传播算法, 因此它避免了对 SNN 进行直接训练所面临的困难. 就性能表现而言, 转换 SNN 保持着与 ANN 发展进程最小的差距, 并具有在大规模的网络结构与数据集上实现的能力, 例如 Rueckauer 等^[123] 实现了 SNN 中的 VGG-16 与 GoogLeNet 模型; Sengupta^[32] 报告了 VGG-16 模型以 0.56% 的转换精度损失在完整

ImageNet数据集得到69.96%准确率;Hu等^[124]更是使用ResNet-50的深层结构取得72.75%准确率的分类成果。

为方便转换,多数工作对原始的ANN模型作出了一定的约束,例如将偏置限制为零、无法使用批归一化方法、必须采用平均池化(average pooling)而非最大池化(max pooling)等。这会造成原始模型性能的部分下降。Rueckauer等^[123]针对以上约束讨论了适合SNN结构的实现方式,但这也使得转换后SNN的调整复杂化并潜在地造成更大的性能损失。此外,在转换方法中格外关键的一点是增加I&F神经元的阈值项后,需要对阈值和权重进行重新平衡。过低的阈值使得神经元易于激发而丧失特异性,反之则会使得脉冲较难激发,深层网络的仿真步数大幅增加。

2019年Severa等^[125]提出了一种称为“磨刀石”(whetstone)的新颖SNN转换方式:在ANN网络中采用有界ReLU函数作为激活函数,在完成最初网络训练之后,逐层地进行有界ReLU函数向阶跃函数的渐进式转换,并在观察到一定性能下降时重启对网络的训练,完整训练流程结束后即可得到对应的SNN网络。相比于针对完整网络进行单次转换,这种渐进的方式能够减缓转换带来的性能损失,但目前该方法得到的SNN仅使用单步仿真时长,在性能表现上相对不佳,且SNN本身在时域上可能的动态特性以及处理能力无法得到进一步的发掘与利用。

整体而言,转换SNN可以较为快速地将ANN领域的突破转化应用至SNN领域,但这种方法也有其内在的局限性。除了对原始ANN施加约束造成的性能下降外,转换SNN完成一次前向推理通常需要几百至几千时间步的长时间模拟,与SNN直接训练算法差距颇大,这导致了与原始目的相悖的额外延迟和能耗。近期也有工作尝试结合转换SNN与直接训练算法,将转换SNN作为一种权重初始化手段并接受直接算法的后续训练,以期得到低时间步下的高性能表现^[126]。除此之外,令人遗憾的一点是转换SNN的视野大多专注于发展新结构的转换方法和缩小与ANN之间的性能差异,而不在于对SNN特质的探究,对SNN发展的直接推动较为有限。

近年来,由于误差反向传播算法在深度神经网络中获得了广泛的成功,不少工作也开始重新考虑将其用于SNN端对端直接训练的可能性,而且在解决第3节之初提及的BP算法不应症上亦取得了一些进展。关于weight transport问题,Lillicrap等^[127]与Baldi等^[128]发现,反向连接对前传权重的精确获取,对于BP算法的有效实施可能并非不可或缺,通过将误差信号乘以随机权重实行的RBP(random

backpropagation)并不会显著地影响学习表现。这对于误差信号在大脑中可能的利用形式提出了新的见解,适应脉冲形式的RBP也见诸报道^[129]。另一个在BP过程中突出面临的问题仍是脉冲函数可导性质的缺乏,目前较为常见的解决方案是采用与之相近的连续函数对脉冲函数或其导数进行替代,进而产生基于脉冲的反向传播算法(spike-based BP),相关综述可见Neftci等^[130]的工作。在具体函数形式选择与算法实现上,直接训练算法表现得较为多样化。例如:在Lee等^[131]的工作中,梯度求导主要围绕膜电势展开,脉冲输入采用低通滤波下的形式对膜电势产生连续影响,而膜电势存在的突变被视为噪声忽略处理;Jin等^[132]提出了HM2-BP算法(hybrid macro/micro level backpropagation),将SNN中的误差反向传播解构为微观上脉冲序列引起的突触后电位改变,以及宏观上由频率编码定义的损失函数反向传播这两个过程;S4NN方法^[133]采用首个脉冲时间编码,通过近似误差梯度得出了基于脉冲时延的BP算法,但网络内神经元受限仅能进行单次脉冲发放;Wu等^[134]基于近似脉冲函数导数的方式,提出了一种在SNN中兼顾空间域和时间域的反向传播算法STBP(spatio-temporal backpropagation, STBP),首先报道了在CIFAR-10数据集上实现高性能深层SNN直接训练的成果^[133],随后的工作借助STBP进一步探索了SNN相较于ANN所具有的优势^[51],在包含时空信息的数据流(例如N-MNIST与DVS-CIFAR10)处理上,SNN表现出以较低的计算开销获得比ANN更高任务性能的能力;Lee等^[135]将LIF神经元的导数处理为两个部分,首先采用不包含电位泄露特性的IF神经元的近似导数,再添加相应的泄露校正进行补偿,实现了在VGG以及ResNet等结构上对SNN的直接训练。

应当指出的是,现有的直接训练算法在深层结构上的应用仍有待探索,与转换SNN或者ANN的发展现状之间依然存在一定的差距。这一方面是因为当前编程框架下SNN额外的时间维度将造成数倍于同规模ANN的训练显存需要,并且由于阈值激发特性与脉冲的稀疏性,当SNN趋于深层时可以预见的是仿真周期的延长,这将进一步提升对显存的需求;另一方面,深层网络训练的难点(例如梯度消失)在SNN结构中同样存在,而部分利于深度神经网络训练的技巧在SNN上进行简单移植时将破坏其保有的优势特征(例如批归一化手段可能造成SNN通信的脉冲形式无法得到保证)。这两点造成了深度训练算法的缺失,使得SNN无法在大规模数据集上获得足够令

表1 SNN算法在不同数据集上的表现

作者/时间	采用算法	网络结构	数据集	准确率/%
Diehl(2015) ^[1105]	指数STDP	二层网络	MNIST	95.00
Kheradpisheh(2018) ^[1111]	STDP	脉冲卷积网络	MNIST	98.40
Zhao(2015) ^[136]	Tempotron	脉冲卷积网络	MNIST	91.29
Tavanaei(2019) ^[137]	BP-STDP	多层感知机	MNIST	97.20
Lee(2016) ^[131]	Spike-based BP	多层感知机	MNIST	98.77
Jin(2018) ^[132]	HM2-BP	脉冲卷积网络	MNIST	99.49
Lee(2020) ^[135]	Spike-based BP	脉冲卷积网络	MNIST	99.59
Wu(2018) ^[134]	STBP	多层感知机	MNIST	98.89
Severa(2019) ^[125]	Whetstone	脉冲卷积网络	MNIST	99.53
Diehl(2015) ^[122]	转换方法	脉冲卷积网络	MNIST	99.10
Rueckauer(2017) ^[123]	转换方法	脉冲卷积网络	MNIST	99.44
Neftci(2013) ^[138]	对比散度	脉冲RBM	MNIST	92.60
O'Connor(2013) ^[139]	转换方法	脉冲DBN	MNIST	94.09
Stromatias(2015) ^[140]	转换方法	脉冲DBN	MNIST	94.94
Wu(2019) ^[33]	STBP	脉冲卷积网络	CIFAR-10	90.53
Lee(2020) ^[135]	Spike-based BP	脉冲卷积网络	CIFAR-10	90.95
Severa(2019) ^[125]	Whetstone	脉冲卷积网络	CIFAR-10	84.67
Rueckauer(2017) ^[123]	转换方法	脉冲卷积网络	CIFAR-10	90.85
Sengupta(2019) ^[32]	转换方法	脉冲卷积网络	CIFAR-10	91.55
Rathi(2020) ^[126]	转换方法+Spike-based BP	脉冲卷积网络	CIFAR-10	92.02
Rathi(2020) ^[126]	转换方法+Spike-based BP	脉冲卷积网络	ImageNet	65.19
Sengupta(2019) ^[32]	转换方法	脉冲卷积网络	ImageNet	69.96
Hu(2018) ^[124]	转换方法	脉冲卷积网络	ImageNet	72.75
Lee(2016) ^[131]	Spike-based BP	多层感知机	N-MNIST	98.74
Jin(2018) ^[132]	HM2-BP	多层感知机	N-MNIST	98.88
Lee(2020) ^[135]	Spike-based BP	脉冲卷积网络	N-MNIST	99.09
Wu(2019) ^[33]	STBP	脉冲卷积网络	N-MNIST	99.53
Wu(2019) ^[33]	STBP	脉冲卷积网络	DVS-CIFAR10	60.50
Sironi(2018) ^[141]	HATS	非神经网络	DVS-CIFAR10	52.40
Lagorce(2017) ^[142]	HOTS	非神经网络	DVS-CIFAR10	27.10

人信服的证据表现. 因此未来一段时间内, 高效深层SNN训练算法仍将是一块能够真正叩开深度学习社区大门的敲门砖. 表1对SNN算法在不同数据集上的表现进行了总结.

3.4 其他脉冲神经网络学习算法

生物神经网络建模中的一个关键问题是, 如何依靠典型神经电路构成的网络对多模态的连续时空输入进行实时处理. 在2002年, Maass等^[143]提出的液体状态机(liquid state machine, LSM)提供了一种可能的计算模型, 其结构和功能受到哺乳动物中枢神经系统中新皮质的启发. 液体状态机主要包括3个部分: 输入层、液体层和输出层. 输入层接受模拟输入或离散的脉冲输入, 并与所有液体层的神经元相连. 液体层由大量LIF神经元构成的循环结构组成, 连接的方向和权重均随机指定, 它们接受输入并将其最终映射至该层所有神经元的放电状态. 循环连接的存在使网络具有了记忆性质, 放电状态包含了当前与历史输入的信息. 输出神经元经受训练, 学习从循环网络的高维状态中提取关于当前与历史输入的信息并用于多种任务. 液体状态机的两个宏观性

质构成了它具有实时计算能力的充要条件: 分离特性(separation property, SP)与逼近特性(approximation property). 分离特性描述了两个不同输入流引起的系统内部状态轨迹之间距离的大小; 逼近特性描述输出机制的分辨和重编码能力, 能够区分液体的不同内部状态并将其转换为给定的目标输出. 与此前的图灵机相比, 液体状态机在数学上同样具有理论的通用计算能力, 但是不需要在离散内部状态之间进行顺序转换, 且更适应具有生物现实性的时变输入情景. LSM具有的记忆与动态输入流实时处理能力令其在语音识别^[144-145]、轨迹预测^[146]等方面获得了应用.

深度信念网络(deep belief nets, DBN)是Hinton等^[147]于2006年提出的一种概率生成式模型. DBN由多层受限玻尔兹曼机(restricted Boltzmann machine, RBM)堆叠组成, 采用对比散度算法(contrastive divergence, CD)高效地近似最大似然法(maximum likelihood learning)进行逐层无监督训练. 在完成无监督预训练之后, 可以在DBN上增加一层包含监督信号的网络, 利用BP算法将误差信息自顶向下传播

至每一层RBM,对无监督过程中初始化的权重进行进一步微调. DBN克服了当时深层BP网络因随机初始化权值而易陷入局部最优和训练耗时长的缺点,首先提出了真正行之有效的深度结构训练方法,并在MNIST上取得了优于支持向量机的结果,使得深度结构更为广泛地进入研究者的视野^[148]. 近年来,具有事件驱动与低功耗特点的神经形态计算架构快速发展,让DBN这一传统网络模型在SNN领域中的拓展应用受到一定的关注. Neftci等^[138]提出了脉冲形式下的对比散度算法,针对LIF模型构成的RBM进行训练;文献^[139,149]则采取了针对LIF模型的近似手段,将预先训练好的DBN转换为适合硬件实现的SNN形式,并在SpiNNaker平台上证实脉冲DBN能

以较低的精度损失得到高效、低延时的MNIST分类结果. 但是, DBN由于相对于深度学习方法在精度上表现欠佳,总体而言不再更多地受到研究者的青睐,脉冲DBN的优势主要继承于神经形态硬件,而其本身对脉冲特性的契合相比之下并不见长.

4 脉冲神经网络编程工具

脉冲神经网络编程工具是用于帮助脉冲神经网络实现快速仿真、网络建模及学习训练的软件平台. 由于研究目标以及实现手段的差异,现阶段存在多种脉冲神经网络编程平台(见表2). 不同的平台对脉冲神经元的生物特性的描述粒度、网络的功能支持及网络的模拟计算效率有很大差异.

表2 脉冲神经网络编程平台总结

平台	编程语言	主要功能	风格	仿生学习	梯度学习	并行度
Gensis(2.4)	C	神经元仿生模型	GUI交互或脚本运行	N/A	N/A	N/A
Neuron(7.4)	C, C++, Fortran, Python	神经元仿生模型	GUI交互或脚本运行	N/A	N/A	MPI
Nest	C++, Python	神经元及网络仿生模型	脚本运行为主	STDP规则等	N/A	分布式MPI
Brian(2.0)	Python	神经元及网络仿生模型	脚本运行	短期可塑性	N/A	分布式GPU
Nengo(2.1)	Python	神经元及网络仿生模型	GUI交互或脚本运行	BCM,Oja规则等	支持	分布式MPI,GPU
SpyKetorch	Python	网络学习模型	脚本运行	STDP规则等	N/A	分布式GPU
BindsNet	Python	网络学习模型	脚本运行	STDP规则等	支持	分布式GPU

Neuron^[150]和Genesis^[151]是一类通用的用于模拟脉冲神经元及神经网络生物学特性的仿真平台. 这类平台提供了用户友好的图像化接口及丰富的功能模块来实现高仿真特性的行为及神经元模型模拟,例如具有不同的形态学细节及功能的多房室神经元模型. 此类平台主要基于C++语言并支持导入多种外接功能包以实现不同的任务需求. 然而,目前此类平台主要支持同类型的大规模神经网络模拟,较难实现同时具有多类不同神经元的大规模网络模拟.

Nest^[152]和Brian^[153]是一类高度灵活、可拓展的脉冲神经网络编程平台. 它可以灵活地支持多房室的突触连接模型及多种不同类型的神经元模型(如LIF模型、Hodgkin-Huxley模型等),以及同时存在不同类神经元的大规模网络模拟. 此外,此类平台还支持部分脉冲神经网络的仿生学习,例如Hebbian、STDP学习. 该类平台支持将模拟结果写入文件,以便可以使用Matlab、Mathematica等工具交互,同时也支持利用多处理器和计算机群集来增加可用内存或加快仿真速度.

NeuCube^[154]和Nengo^[155]是专注于脉冲神经网络高级行为模拟的编程平台. 它们也可以很好地支持基于多种不同神经元所构建的大型神经网络,同时还支持对Matlab及Java等的交互,常被作为实现神经工程的基本框架以进行3D大脑脑区的功能及行为

模拟. 其中,Nengo是基于Python编写的开源项目,提供了TensorFlow等深度学习加速平台的接口,以提高仿真速度及提供部分机器学习方法的使用.

BindsNet^[38]和Spyketorch^[39]是主要面向脉冲神经网络学习算法及应用的编程平台. 该类平台为脉冲神经网络在仿生学习、监督学习以及强化学习任务中的构建提供便利. 由于此类平台主要基于Pytorch/Tensorflow等深度学习加速平台编写而成,可直接利用其优化技术进行大规模加速模拟,同时也可以利用自动梯度求导机制对脉冲神经网络进行学习和训练.

5 神经形态计算平台

神经形态的概念最早由加州理工学院的Mead^[156]提出,通过采用专用集成电路实现神经科学计算. 本文中的神经形态计算平台主要指面向脉冲神经网络的专用处理平台,如图9所示.

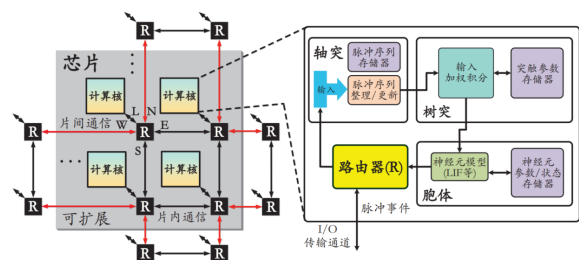


图9 典型的神经形态芯片架构图(以网格状路由为例)

言,芯片之间采用2D网格路由拓扑,而片内的计算核之间采用树形结构路由拓扑(每个父节点有4个子节点),计算核内采用多播和标签(tag)匹配的方式进行路由包传输,匹配过程由内容寻址存储器(CAM)完成,这种路由结构能够结合树形拓扑的低延迟和网格拓扑的低带宽需求的优势.实际制造的系统配置为,单板含9块芯片,单芯片含4个计算核(如图11所示),单核含256个神经元,单神经元拥有64扇入和约4000扇出能力.原文演示了一个基于4层卷积神经网络的扑克识别应用.

除了利用晶体管模拟电特性的数模混合电路,还有一类数模混合神经形态计算平台则采用新型纳米忆阻器件.这类平台利用忆阻器阵列的存算一体特性完成模拟域的突触存储和树突积分^[159],采用传统RC电路^[160-161]或基于忆阻器的RC电路实现神经元计算^[162-163],而其他部分仍为数字电路.忆阻器(memristor)是2008年发明的一种两端纳米电子器件,能够保持过去的历史电导状态,用于表示突触权重,并在输入的电压或电流信号作用下逐渐被调制到新的状态^[164].忆阻器阵列具有高密度特性,能够将计算与存储融为一体,可以大大提高矩阵向量乘法操作的运算速度和能效.如图12所示,典型的忆阻器神经形态计算平台主要由忆阻器阵列及其读写驱动电路、神经元电路、额外缓存与路由网络等构成.但由于存在阵列的漏电与可控性问题,目前集成大规模的忆阻器阵列还比较困难,实际中流片的忆阻器系统规模都比较小^[160-161],中大规模的忆阻器神经形态计算平台仍以仿真为主^[165-166].

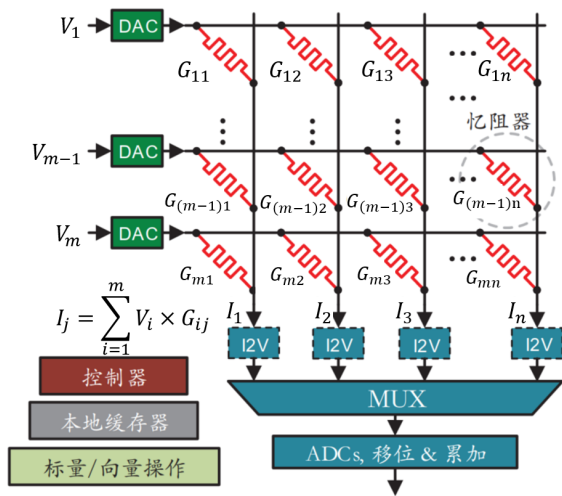


图 12 基于忆阻器件的神经形态计算系统

5.1.2 全数字电路

虽然采用模拟电路可以相对容易地实现较为复杂的神经元动力学,但模拟电路具有稳定性不足、可

编程性较弱、难以仿真等缺陷.在这种背景下,基于全数字电路的神经形态计算平台更加受到产业界的青睐,其中离线学习的先驱和代表性工作当数 IBM 的 TrueNorth^[167]. TrueNorth 单芯片含有 4 096 个计算核(见图 13),计算核通过 2D 网格的路由进行连接,可多片扩展.每个计算核包括 256 个神经元和 256×256 尺寸的突触阵列,支持基本 LIF 神经元模型及其诸多变体,每个神经元共享 4 种强度的突触权重.由于采用事件驱动的异步同步电路混合设计方案,仅在脉冲事件输入的情况下才启动计算,使得其在模拟百万神经元规模的脉冲神经网络时功耗也仅有几十、上百毫瓦^[11,168].与硬件系统配套, TrueNorth 团队还开发了与芯片一一对应的硬件仿真器 Compass^[169],并基于面向对象的“封装类”设计了神经形态网络设计语言 Corelet^[170],应用于多目标识别追踪、图像识别、语音识别、机器人、事件预测、避障决策等诸多领域^[168,171-172].

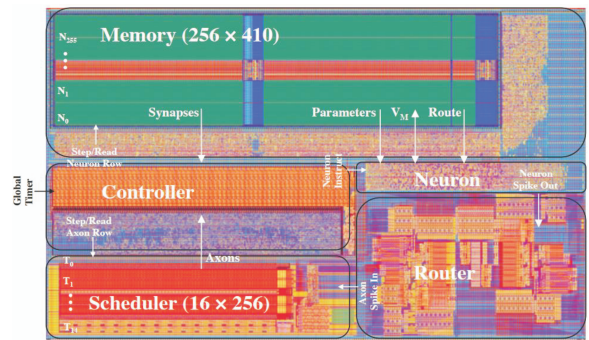


图 13 TrueNorth^[167] 计算核

传统神经形态计算平台的核间通信仅能以二值脉冲的格式进行传输,这导致了在高扇入情况下的精度损失问题,进而损害整体的应用性能. Shenjing^[173]通过设计支持两种数据格式的路由器克服了这个问题:除去普通的支持脉冲数据的路由器,额外增加了支持高精度部分和数据的路由器,用以传输中间累加的膜电位.例如,通过结合两个扇入能力均为 256 的计算核,将其中一个计算核的中间膜电位结果发送至另一个计算核,在路由器中与本地的中间膜电位进行累加,可以增大神经元扇入能力至 512 而不损害应用性能.

脉动阵列是计算矩阵乘法的高效率方式,随着谷歌 TPU^[20] 机器学习加速器的成功而再次引起关注.一个脉动阵列通常含有一个乘法器阵列.输入矩阵、权重矩阵和输出矩阵中有一组数据固定在乘法器阵列的局部缓存中,而其他两组数据沿两个不同的方向在阵列中流动,以实现数据的最大复用,从而减少对远处缓存的访问以降低能耗.这种架构最近也被应用至脉冲神经网络中^[174],并且根据脉冲神经网络

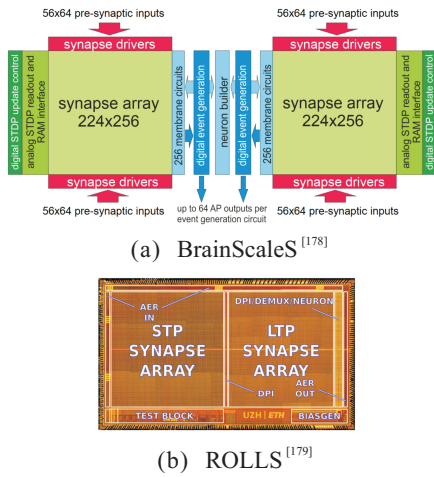


图 16 基于数模混合电路的在线学习神经形态计算平台

比 Neurogrid 和 BrainScaleS 晚一些的 ROLLS^[179] (图 16), 采用亚阈值模拟电路实现神经元和突触动力学, 不追求网络规模(单芯片仅有 256 神经元、 256×256 短时程可塑性突触和 256×256 长时程可塑性突触), 专用于研究仿生突触学习机制. 长时程突触利用脉冲驱动突触可塑性 (spike-driven synaptic plasticity, SDSP) 进行学习, 具体公式如下:

$$\begin{cases} w^+ = \Delta w^+, u_{t_{pre}} > \theta_u \ \& \ \theta_1 < Ca_{t_{pre}} < \theta_3; \\ w^- = \Delta w^-, u_{t_{pre}} < \theta_u \ \& \ \theta_1 < Ca_{t_{pre}} < \theta_2. \end{cases} \quad (21)$$

其中: $u_{t_{pre}}$ 是输入脉冲到达时后端神经元的膜电位, θ_u 是决定权重增减的膜电位阈值, $Ca_{t_{pre}}$ 表示输入脉冲到达时后端神经元的钙浓度, θ_1 、 θ_2 、 θ_3 为决定权重增减的 3 个钙浓度阈值. 短时程突触则允许用户进行突触权重编程而非逐渐学习, 具体而言, 每个突触具有一个可编程锁存器用于权重选择. 此外, 突触中有额外电路用于模拟短时程抑制现象, 使得突触后响应强度在连续脉冲信号注入期间逐渐减弱. ROLLS 原文演示了小规模吸引子网络和二分类三层感知器.

如前所述, 基于纳米忆阻器件的数模混合神经形态计算平台也得到广泛研究. 因其具有存算一体的特性, 忆阻器阵列既作为突触权重存储介质又作为计算模块, 故忆阻器系统的在线学习也需要在阵列本地完成. 忆阻器被发明后的几年间, 存在许多单个器件级别的突触学习规则研究, 主要以 STDP 规则的实现为主^[180-181]. 随后, STDP 在线学习被推广到阵列级别或小规模网络级别^[160, 165, 182], 但受限于忆阻器系统本身的集成难度和仿生突触学习规则的限制性, 尚未见到基于忆阻器的在线学习神经形态平台应用于大规模的脉冲神经网络. 此外, 在线学习与离线学习不同,

需要对突触器件进行反复写入操作, 因此对忆阻器件的写寿命有较高的需求, 这也是忆阻器件用于在线学习神经形态计算平台的一项挑战.

5.2.2 全数字电路

SpiNNaker 是早期可以支持在线学习的数字电路神经形态计算平台^[12, 46]. 单个芯片含有 18 个 ARM 处理器核和 128 兆字节的片外 DRAM 存储器, 每个 ARM 核可以仿真近 1000 个神经元, DRAM 存储器用于贮存突触参数. 由于 ARM 核的高度编程灵活性, SpiNNaker 可以支持多种神经元动力学模型, 包括 LIF、Izhikevich 与 Hodgkin-Huxley 模型, 甚至支持突触学习功能^[183]. 基于该芯片架构, SpiNNaker 团队进一步开发了含 48 个芯片的 PCB 板, 预计实现人脑百分之一规模需要约 1200 块该电路板. 神经元间采用六角形拓扑结构互连以构成庞大的神经形态网络^[184], 并基于此设计了路由容错机制^[185]. 由于每个芯片的所有 ARM 核共享一个片外 DRAM 存储器, 只有在前端脉冲输入时才会加载片外突触参数到局部核内. 对此, 通过将前后端脉冲信号均可触发的 STDP 规则修改为仅有前端脉冲信号可以触发的 STDP 规则, 进一步利用延迟触发的方案解决前端脉冲信号到达时后端脉冲信号还未发生的问题, 实现了在线学习功能^[186].

ODIN^[187] 可以认为是前面所述 ROLLS 的全数字电路版本. 整个芯片仅有单个计算核, 包含 256 个神经元与 256×256 个突触, 所采用的学习规则与 ROLLS 的长时程突触一样, 即如式 (21) 所示的 SDSP. ODIN 支持标准的 LIF 神经元模型或自定义可呈现 20 种 Izhikevich 响应行为的神经元模型, 演示了一个两层全连接神经网络, 并证明推理阶段使用脉冲顺序编码比脉冲频率编码功耗更低.

MorphIC^[188] 是来自相同团队的 ODIN 升级版, 主要改变有两个地方. 首先, 权重从 4 比特缩减为 1 比特, 连续的 SDSP 突触学习规则变为随机脉冲驱动突触可塑性 (stochastic spike-driven synaptic plasticity, S-SDSP), 具体如下:

$$\begin{cases} w_b^+ = \zeta^+(1-w_b), u_{t_{pre}} > \theta_u \ \& \ \theta_1 < Ca_{t_{pre}} < \theta_3; \\ w_b^- = \zeta^-w_b, u_{t_{pre}} < \theta_u \ \& \ \theta_1 < Ca_{t_{pre}} < \theta_2. \end{cases} \quad (22)$$

其中: w_b 为二值权重, ζ^\pm 为二值随机变量, 其他变量与式 (21) 相同. 其次, 网络规模相较于 ODIN 明显增加, 单片含有 4 个计算核, 单核含有 512 个神经元、 512×32 个 L2 突触、 512×512 个 L1 突触和 512×512 个 L0 突触. L0、L1、L2 分别对应于类似 DYNAPs 的三级异构路由拓扑: L0 为计算核内路由、L1 为片内核间

树形路由、L2为片间网格路由. 其中:L0和L1路由会将脉冲信号分别传送至目标核中L0和L1某一行的所有突触从而影响该核所有神经元,而L2路由只将信息传送给L2突触中的某一个突触. 原文演示了基于离线学习3层感知器的手写数字体识别和基于在线学习4层卷积神经网络的八分类模式识别,但后者只有最后一层全连接层采用了S-SDSP规则进行突触学习.

Loihi^[13]是英特尔发布的在线学习神经形态计算芯片,主要针对仿生突触学习的研究,如多种形式的STDP. 单个芯片包含128个计算核,单核包含最多1024个神经元和16兆字节突触容量(突触精度可于1比特至9比特之间灵活调整),整个芯片集成了33兆字节的片上SRAM存储空间. Loihi的灵活性体现在多处地方,如支持可配置突触延迟、状态变量随机噪声、树突计算、发放阈值自适应、稀疏连接模型、核对核多播、神经元群分级连接等. 原文演示了基于脉冲神经网络求解LASSO问题,Loihi可获得比CPU高5000倍的能效.

以上4种基于数字电路的在线学习神经形态计算平台如图17所示.

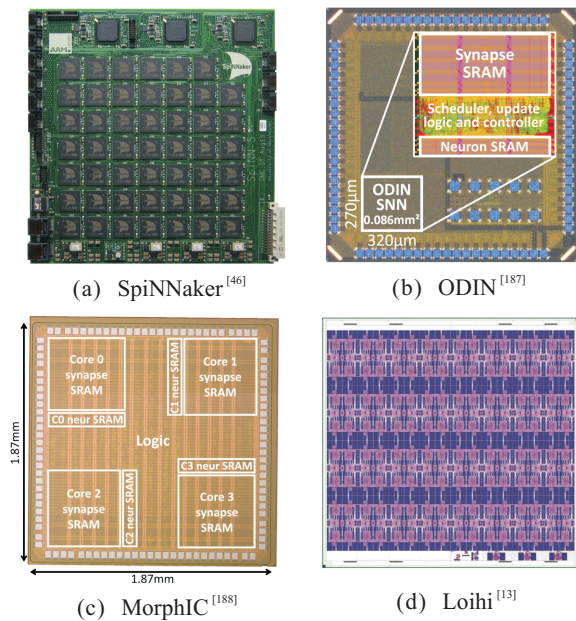


图17 基于数字电路的在线学习神经形态计算平台

前述的在线学习神经形态计算平台所支持的学习规则比较单一或有限,对此,FlexLearn^[189](见图18)设计了一个在线学习引擎,能够支持17种具有代表性的突触学习规则,包括长时程可塑性、短时程可塑性和内稳态等三大类别. FlexLearn通过为每种学习规则设计专用的变量轨迹追踪通路,并分析各类通路的依赖性,最大程度地进行资源复用以提高效率;同时,将上述三大类别学习规则的数据通路拆分为细粒度的子通路,进而可支持并行执行和流水线优化. 进一步,与支持多种神经元模型的Flexon^[190]模块集成,可完整地实现高度灵活的在线学习神经形态计算核. 128个这样的计算核通过网格路由拓扑进行连接,便可构成单个在线学习神经形态计算芯片.

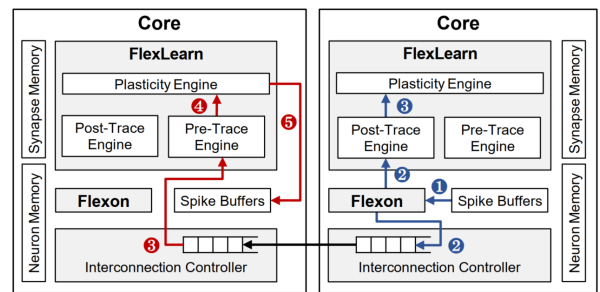


图18 FlexLearn^[189]架构示意

目前,绝大多数在线学习神经形态计算平台均采用无监督学习规则,主要由于其具有局部性良好、在众核去中心化架构上容易实现的特性. 也存在少数平台采用改进后的轻量化有监督学习规则,如韩国首尔国立大学最近的一项工作^[191]. 为了消除误差反向传播算法的层间串行依赖性,最后一层的输出脉冲序列和目标脉冲训练被直接传递至中间各隐层以并行地产生相应的权重修改值;其次,当输出脉冲序列与目标脉冲序列一致时,屏蔽权重改变量的产生和权重修改模块的执行以降低能耗;最后,各层实行对连续输入样本的流水线执行和权重的跨样本乱序更新,从而提高整体的并行性. 但该文演示的规模很小,学习方法和硬件架构的可扩展性还需进一步验证.

表4对在线学习神经形态计算平台进行了总结.

表4 在线学习神经形态计算平台总结

平台	BrainScaleS ^[178]	ROLLS ^[179]	SpiNNaker ^[46]	Loihi ^[13]
类型	数模混合	数模混合	全数字	全数字
制程工艺 / nm	180	180	130	14
面积 / mm ²	50	51.44	102	60
核数	352	1	18	128
神经元/核	8~512	256	~1K	1024
突触/核	~130K	128K	~1M	~1M
单次突触操作功耗 / pJ	174 ^[192]	N/A	4K ^[177]	23.6

6 总结与展望

大脑始终是人工智能的指向标与目的地. 本篇综述的目的是让读者能够全面了解在脉冲神经网络领域中围绕神经元模型、训练算法、编程框架、数据集以及硬件芯片五大方面已经取得的历史成果以及未来可能的发展方向,并期待通过跨学科的思想交流与合作研究,加快迈向人工通用智能目标的步伐.

在本篇综述中,首先介绍了脉冲神经网络的构成要素及其生物学渊源. SNN对于神经元脉冲特征及相应动力学的保留构成了它与ANN的基本差异,面向不同目的的仿真研究能够有针对性地使用不同精细度的建模层次. 脉冲序列是SNN中信息传递的载体,在其上神经元拥有丰富的编码方式. 但是当前相当数量的SNN算法仅对频率编码给予关注,而这一编码方式无视于序列内部的时间结构,很可能无法发挥SNN在时域信息处理上的优势. 因此,适应于高信息密度时间编码的算法设计是目前有待探索的方向.

在算法部分,将现有的SNN训练算法划分为两个主要方向. 以STDP为代表的仿生无监督算法较好地遵循了大脑中神经元连接的强度调整规律,能够通过局部学习有效地进行特征提取. 有监督训练算法针对脉冲函数的不可导性质提出了不同的解决方案,主要通过Widrow-Hoff规则或近似BP算法两种方式实现. 相较于无监督算法,它们在生物合理性上有所缺失,BP算法更是面临weight transport问题,但引入作为监督信号的标签信息使得在相同数据集上的表现得到提升并更加支持网络向深层结构进行扩展. 现有的算法在生物合理性与性能表现、模型容量之间始终存在矛盾,对此,Tavanaei等^[137]提出了BP-STDP算法,希望获得兼具可塑性规则的生物合理性与梯度下降方法最优性训练能力的综合算法. 但整体而言,BP-STDP并不具有STDP局部学习与时间依赖的特点,其内核仍是梯度下降方法. 对生物合理性与性能表现、模型容量两方面的有机结合依然会是SNN算法领域不懈的追求目标. 另外,相较深度学习发展现状而言,当前SNN在真正大规模的深层网络训练算法上仍有亟待填补的空白. 例如STBP^[33]等直接训练算法在仿真周期、模型性能以及网络规模上综合表现较优,可能能够作为未来SNN训练算法持续发展的途径,但是,在深度神经网络训练中存在的梯度消失、资源开销大甚至算法不收敛等问题都有待于进一步确认与探讨.

转换SNN是针对时新深度学习成果进行SNN转化的一种快捷方式,在网络性能上拥有接近原始

ANN网络的表现,但较长的仿真周期以及原始模型的额外约束使其受到一定的掣肘. 从模型压缩的角度而言,SNN化的过程也是一种针对激活值进行的极端量化,在ANN中存在二值网络(binary neural network, BNN)^[193]等具有相似理念的工作,而对于二者之间的联系与差异以及额外时间维度可能造成的影响,目前还没有较为明确的阐述. SNN神经元所具有的阈值发放特性可能使其对模型压缩算法具有更高的接受度,因此,与权重量化、剪枝等压缩算法的结合也有待探索,以使SNN计算效率优势得到进一步发扬与利用.

在ANN的发展历程中,数据集与编程工具都是领域发展与社区建设中不可或缺的驱动力,然而,目前与SNN相配套的二者仍处于较为初级的发展阶段. 在ANN领域的静态图片数据集上,SNN的表现通常不如ANN,但有研究表明一味地在该类数据集上以单标准衡量SNN是不明智的,在包含更多动态时间信息且天然具有脉冲信号形式的数据集中,SNN完全可以在性能以及计算开销上取得更优的成绩^[51]. 通过神经形态视觉传感器获得的数据集是当下SNN数据集的主流,但并不排除可能有其他更合适的数据来源仍待探索. 在图像识别任务之外,希望能够发展出面向多样化任务的时空事件流数据集,最终为SNN提供广泛且公允的测试对比基准,并进一步挖掘SNN的潜在优势与可能的应用场景. 编程工具则极大地关系到领域准入门槛的有效降低以及大规模SNN工程的高效开发,为SNN的科学研究与工业部署提供实质性的支撑.

神经形态硬件作为遭遇冯·诺依曼瓶颈与摩尔定律失效的情况下对传统数字电路的一种替代计算范式,成为研究热点已经有十余年的历史,并取得了丰硕的成果. 通过从大脑的结构和功能中汲取灵感,神经形态硬件为SNN中基于脉冲事件驱动的计算提供了高效的解决方案,实现了大量并行和超低功耗等重要特性. 本文认为,在神经形态器件自身进行架构创新、继续追求更高效的基础上,针对现有硬件平台探索各种优势技术的结合,是另一个需要深入钻研的重要方向. 这种交叉融合可能体现在多个角度,例如低精度忆阻器与高精度数字电路的混合精度计算、ANN与SNN两类范式异构融合提升总体性能、高效率低准确率的无监督局部学习与高准确率低效率的有监督全局学习相结合的全能训练平台等等. 此外,也同样期待神经形态计算平台能够像GPU之于机器学习一样,为SNN的推理与训练提供灵活高效的加

速支撑,进而促进对SNN模型的探索,催化新兴SNN算法的诞生.

参考文献(References)

- [1] Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain[J]. *Psychological Review*, 1958, 65(6): 386-408.
- [2] Rosenblatt F. Perceptron simulation experiments[J]. *Proceedings of the IRE*, 1960, 48(3): 301-309.
- [3] Hopfield J J. Neural networks and physical systems with emergent collective computational abilities[J]. *Proceedings of the National Academy of Sciences*, 1982, 79(8): 2554-2558.
- [4] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. *Commun ACM*, 2017, 60(6): 84-90.
- [5] Maass W. Networks of spiking neurons: The third generation of neural network models[J]. *Neural Networks*, 1997, 10(9): 1659-1671.
- [6] Roy K, Jaiswal A, Panda P. Towards spike-based machine intelligence with neuromorphic computing[J]. *Nature*, 2019, 575(7784): 607-617.
- [7] Zhang B, Shi L, Song S. Creating more intelligent robots through brain-inspired computing[J]. *Science*, 2016, 354(6318): 4-9.
- [8] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. *Nature*, 2015, 518(7540): 529-533.
- [9] Silver D, Huang A, Maddison C J, et al. Mastering the game of go with deep neural networks and tree search[J]. *Nature*, 2016, 529(7587): 484-489.
- [10] Senior A W, Evans R, Jumper J, et al. Improved protein structure prediction using potentials from deep learning[J]. *Nature*, 2020, 577(7792): 706-710.
- [11] Akopyan F, Sawada J, Cassidy A, et al. TrueNorth: Design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015, 34(10): 1537-1557.
- [12] Painkras E, Plana L A, Garside J, et al. SpiNNaker: A 1-W 18-core system-on-chip for massively-parallel neural network simulation[J]. *IEEE Journal of Solid-State Circuits*, 2013, 48(8): 1943-1953.
- [13] Davies M, Srinivasa N, Lin T H, et al. Loihi: A neuromorphic manycore processor with on-chip learning[J]. *IEEE Micro*, 2018, 38(1): 82-99.
- [14] Pei J, Deng L, Song S, et al. Towards artificial general intelligence with hybrid tianjic chip architecture[J]. *Nature*, 2019, 572(7767): 106-111.
- [15] Abadi M, Barham P, Chen J, et al. TensorFlow: A system for large-scale machine learning[C]. *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*. Savannah: USENIX Association, 2016: 265-283.
- [16] Paszke A, Gross S, Massa F, et al. PyTorch: An imperative style, high-performance deep learning library[M]. *Advances in Neural Information Processing Systems 32*. Red Hook: Curran Associates Inc., 2019: 8026-8037.
- [17] Chen Y J, Luo T, Liu S L, et al. DaDianNao: A machine-learning supercomputer[C]. *The 47th Annual IEEE/ACM International Symposium on Microarchitecture*. Piscataway: IEEE, 2014: 609-622.
- [18] Du Z D, Fasthuber R, Chen T S, et al. ShiDianNao: Shifting vision processing closer to the sensor[C]. *Proceedings of the 42nd Annual International Symposium on Computer Architecture*. Piscataway: IEEE, 2015: 92-104.
- [19] Zhang S J, Du Z D, Zhang L, et al. Cambricon-X: An accelerator for sparse neural networks[C]. *The 49th Annual IEEE/ACM International Symposium on Microarchitecture*. Piscataway: IEEE, 2016: 1-12.
- [20] Jouppi N P, Young C, Patil N, et al. In-datacenter performance analysis of a tensor processing unit[C]. *Proceedings of the 44th Annual International Symposium on Computer Architecture*. New York: Association for Computing Machinery, 2017: 1-12.
- [21] Yin S Y, Ouyang P, Tang S B, et al. A high energy efficient reconfigurable hybrid neural network processor for deep learning applications[J]. *IEEE Journal of Solid-State Circuits*, 2018, 53(4): 968-982.
- [22] Chen Y H, Krishna T, Emer J S, et al. Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks[J]. *IEEE Journal of Solid-State Circuits*, 2017, 52(1):127-138.
- [23] Dayan P, Abbott L. *Computational neuroscience: Theoretical neuroscience: Computational and mathematical modeling of neural systems*[M]. Cambridge: MIT Press, 2001: 162-166.
- [24] Brunel N, Latham P E. Firing rate of the noisy quadratic integrate-and-fire neuron[J]. *Neural Computation*, 2003, 15(10): 2281-2306.
- [25] Fourcaud-Trocmé N, Hansel D, van Vreeswijk C, et al. How spike generation mechanisms determine the neuronal response to fluctuating inputs[J]. *The Journal of Neuroscience*, 2003, 23(37): 11628-11640.
- [26] Brette R, Gerstner W. Adaptive exponential integrate-and-fire model as an effective description of neuronal activity[J]. *Journal of Neurophysiology*, 2005, 94(5): 3637-3642.
- [27] Izhikevich E M. Simple model of spiking neurons[J]. *IEEE Transactions on Neural Networks*, 2003, 14(6): 1569-1572.
- [28] Hodgkin A L, 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.
- [29] Izhikevich E M, Gally J A, Edelman G M. Spike-timing dynamics of neuronal groups[J]. *Cerebral Cortex*, 2004,

- 14(8): 933-944.
- [30] Kornijcuk V, Lim H, Seok J Y, et al. Leaky integrate-and-fire neuron circuit based on floating-gate integrator[J]. *Frontiers in Neuroscience*, 2016, 10: 212.
- [31] Masquelier T, Thorpe S J. Learning to recognize objects using waves of spikes and spike timing-dependent plasticity[C]. *The 2010 International Joint Conference on Neural Networks*. Piscataway: IEEE, 2010: 1-8.
- [32] Sengupta A, Ye Y T, Wang R, et al. Going deeper in spiking neural networks: VGG and residual architectures[EB/OL]. 2018, arXiv: 1802.02627.
- [33] Wu Y J, Deng L, Li G Q, et al. Direct training for spiking neural networks: Faster, larger, better[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, 33: 1311-1318.
- [34] Team T T D, Al-Rfou R, Alain G, et al. Theano: A python framework for fast computation of mathematical expressions[J]. 2016, arXiv:1605.02688.
- [35] Jia Y Q, Shelhamer E, Donahue J, et al. Caffe: Convolutional architecture for fast feature embedding[C]. *Proceedings of the 22nd ACM International Conference on Multimedia*. New York: ACM Press, 2014: 675-678.
- [36] Chen T Q, Li M, Li Y T, et al. MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems[J]. 2015, arXiv:1512.01274.
- [37] Chollet F, et al. Keras[EB/OL]. 2015. <https://keras.io>.
- [38] Hazan H, Saunders D J, Khan H, et al. BindsNET: A machine learning-oriented spiking neural networks library in python[J]. 2018, arXiv: 1806.01423.
- [39] Mozafari M, Ganjtabesh M, Nowzari-Dalini A, et al. SpykeTorch: Efficient simulation of convolutional spiking neural networks with at most one spike per neuron[J]. *Frontiers in Neuroscience*, 2019, 13: 625.
- [40] LeCun Y, Cortes C, Burges C J. The mnist database of handwritten digits, 1998[J]. URL <http://yann.lecun.com/exdb/mnist>, 1998, 10: 34.
- [41] Krizhevsky A. Learning multiple layers of features from tiny images[D]. Toronto: Department of Computer Science, University of Toronto, 2009: 1-60.
- [42] Deng J, Dong W, Socher R, et al. Imagenet: A large-scale hierarchical image database[C]. *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*. Los Alamitos: IEEE Computer Society, 2009: 248-255.
- [43] 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.
- [44] Li H M, Liu H C, Ji X Y, et al. CIFAR10-DVS: An event-stream dataset for object classification[J]. *Frontiers in Neuroscience*, 2017, 11: 309.
- [45] Amir A, Taba B, Berg D, et al. A low power, fully event-based gesture recognition system[C]. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway: IEEE, 2017: 7388-7397.
- [46] Furber S B, Galluppi F, Temple S, et al. The SpiNNaker project[J]. *Proceedings of the IEEE*, 2014, 102(5): 652-665.
- [47] Benjamin B V, Gao P, McQuinn E, et al. Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations[J]. *Proceedings of the IEEE*, 2014, 102(5): 699-716.
- [48] Shen J C, Ma D, Gu Z H, et al. Darwin: A neuromorphic hardware co-processor based on spiking neural networks[J]. *Science China Information Sciences*, 2016, 59(2): 1-5.
- [49] Izhikevich E M. Which model to use for cortical spiking neurons?[J]. *IEEE Transactions on Neural Networks*, 2004, 15(5): 1063-1070.
- [50] Jolivet R, Timothy J, Gerstner W. The spike response model: A framework to predict neuronal spike trains[C]. *Proceedings of the 2003 Joint International Conference on Artificial Neural Networks and Neural Information Processing*. Berlin: Springer, 2003: 846-853.
- [51] Deng L, Wu Y J, Hu X, et al. Rethinking the performance comparison between SNNs and ANNs[J]. *Neural Networks*, 2020, 121: 294-307.
- [52] Shand J. Central works of philosophy: Central works of philosophy, Volume 2: Seventeenth and eighteenth century[M]. Montreal: McGill-Queen's University Press, 2005: 115-120.
- [53] Bain A. Mind and body: The theories of their relation[M]. New York: D. Appleton and Company, 1874: 91.
- [54] Markram H, Gerstner W, Sjöström P J. A history of spike-timing-dependent plasticity[J]. *Frontiers in Synaptic Neuroscience*, 2011, 3: 4.
- [55] Hebb D O. The organization of behavior: A neuropsychological theory[M]. New York: John Wiley and Sons, 1949: 62.
- [56] Shatz C J. The developing brain[J]. *Scientific American*, 1992, 267(3): 60-67.
- [57] Malinow R. Transmission between pairs of hippocampal slice neurons: quantal levels, oscillations, and LTP[J]. *Science*, 1991, 252(5006): 722-724.
- [58] Markram H, Sakmann B. Action potentials propagating back into dendrites trigger changes in efficacy of single-axon synapses between layer V pyramidal neurons[C]. *The 25th Annual Meeting of Society for Neuroscience Abstracts*. Washington DC: Society for Neuroscience, 1995: 2007.
- [59] Markram H, Lübke J, Frotscher M, et al. Regulation of synaptic efficacy by coincidence of postsynaptic apses and epsps[J]. *Science (New York)*, 1997, 275(5297): 213-215.
- [60] Song S, Miller K D, Abbott L F. Competitive hebbian learning through spike-timing-dependent synaptic plasticity[J]. *Nature Neuroscience*, 2000, 3(9): 919-926.
- [61] Bell C C, Han V Z, Sugawara Y, et al. Synaptic plasticity

- in a cerebellum-like structure depends on temporal order[J]. *Nature*, 1997, 387(6630): 278-281.
- [62] Debanne D, Gähwiler B H, Thompson S M. Long-term synaptic plasticity between pairs of individual CA3 pyramidal cells in rat hippocampal slice cultures[J]. *The Journal of Physiology*, 1998, 507(1): 237-247.
- [63] Feldman D E. Timing-based ltp and ltd at vertical inputs to layer II/III pyramidal cells in rat barrel cortex [J]. *Neuron*, 2000, 27(1): 45-56.
- [64] Sjöström P J, Turrigiano G G, Nelson S B. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity[J]. *Neuron*, 2001, 32(6): 1149-1164.
- [65] Zhang L I, Tao H W, Holt C E, et al. A critical window for cooperation and competition among developing retinotectal synapses[J]. *Nature*, 1998, 395(6697): 37-44.
- [66] Bi G Q, Poo M M. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type[J]. *The Journal of Neuroscience*, 1998, 18(24): 10464-10472.
- [67] Yao H, Dan Y. Stimulus timing-dependent plasticity in cortical processing of orientation[J]. *Neuron*, 2001, 32(2): 315-323.
- [68] Schuett S, Bonhoeffer T, Hübener M. Pairing-induced changes of orientation maps in cat visual cortex[J]. *Neuron*, 2001, 32(2): 325-337.
- [69] Meliza C D, Dan Y. Receptive-field modification in rat visual cortex induced by paired visual stimulation and single-cell spiking[J]. *Neuron*, 2006, 49(2): 183-189.
- [70] Jacob V, Brasier D J, Erchova I, et al. Spike timing-dependent synaptic depression in the in vivo barrel cortex of the rat[J]. *Journal of Neuroscience*, 2007, 27(6): 1271-1284.
- [71] Müller-Dahlhaus F, Ziemann U, Classen J. Plasticity resembling spike-timing dependent synaptic plasticity: The evidence in human cortex[J]. *Frontiers in Synaptic Neuroscience*, 2010, 2: 34.
- [72] Schultz W. Predictive reward signal of dopamine neurons[J]. *Journal of Neurophysiology*, 1998, 80(1): 1-27.
- [73] Ranganath C, Rainer G. Neural mechanisms for detecting and remembering novel events[J]. *Nature Reviews Neuroscience*, 2003, 4(3): 193-202.
- [74] Frémaux N, Gerstner W. Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules[J]. *Frontiers in Neural Circuits*, 2015, 9: 85.
- [75] Pawlak V, Kerr J N D. Dopamine receptor activation is required for corticostriatal spike-timing-dependent plasticity[J]. *Journal of Neuroscience*, 2008, 28(10): 2435-2446.
- [76] Seol G H, Ziburkus J, Huang S Y, et al. Neuromodulators control the polarity of spike-timing-dependent synaptic plasticity[J]. *Neuron*, 2007, 55(6): 919-929.
- [77] Adrian E D, Zotterman Y. The impulses produced by sensory nerve-endings, Part II: The response of a single end-organ[J]. *The Journal of Physiology*, 1926, 61(2): 151-171.
- [78] VanRullen R, Guyonneau R, Thorpe S J. Spike times make sense[J]. *Trends in Neurosciences*, 2005, 28(1): 1-4.
- [79] Thorpe S, Gautrais J. Rank order coding[C]. *Computational Neuroscience*. Boston: Springer, 1998: 113-118.
- [80] Gautrais J, Thorpe S. Rate coding versus temporal order coding: A theoretical approach[J]. *Biosystems*, 1998, 48 (1/2/3): 57-65.
- [81] Rullen R V, Thorpe S J. Rate coding versus temporal order coding: What the retinal ganglion cells tell the visual cortex[J]. *Neural Computation*, 2001, 13(6): 1255-1283.
- [82] Izhikevich E M, Desai N S, Walcott E C, et al. Bursts as a unit of neural information: Selective communication via resonance[J]. *Trends in Neurosciences*, 2003, 26 (3): 161-167.
- [83] Georgopoulos A P, Schwartz A B, Kettner R E. Neuronal population coding of movement direction[J]. *Science*, 1986, 233(4771): 1416-1419.
- [84] Olshausen B A, Field D J. Sparse coding of sensory inputs[J]. *Current Opinion in Neurobiology*, 2004, 14 (4): 481-487.
- [85] Wang Z, Yang J, Zhang H, et al. Sparse coding and its applications in computer vision[M]. New Jersey: World Scientific, 2015: 3-6.
- [86] Lu T, Liang L, Wang X Q. Temporal and rate representations of time-varying signals in the auditory cortex of awake primates[J]. *Nature Neuroscience*, 2001, 4(11): 1131-1138.
- [87] Cox D D, Dean T. Neural networks and neuroscience-inspired computer vision[J]. *Current Biology*, 2014, 24(18): R921-R929.
- [88] Brandli C, Berner R, Yang M, et al. A 240×180 130 dB 3μs latency global shutter spatiotemporal vision sensor[J]. *IEEE Journal of Solid-State Circuits*, 2014, 49(10): 2333-2341.
- [89] Moeys D P, Neil D, Corradi F, et al. Pred18: Dataset and further experiments with davis event camerain predator-preyrobot chasing[J]. 2018, arXiv: 1807.03128.
- [90] Bardow P, Davison A J, Leutenegger S. Simultaneous optical flow and intensity estimation from an event camera[C]. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway: IEEE, 2016: 884-892.
- [91] Benosman R, Clercq C, Lagorce X, et al. Event-based visual flow[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2013, 25(2): 407-417.
- [92] Carneiro J, Ieng S H, Posch C, et al. Event-based 3d reconstruction from neuromorphic retinas[J]. *Neural Networks*, 2013, 45: 27-38.
- [93] Kim H, Leutenegger S, Davison A J. Real-time 3D reconstruction and 6-DoF tracking with an event camera

- [C]. Computer Vision—ECCV 2016. Cham: Springer International Publishing, 2016: 349-364.
- [94] Bi Y, Andreopoulos Y. PIX2NVS: Parameterized conversion of pixel-domain video frames to neuromorphic vision streams[C]. 2017 IEEE International Conference on Image Processing. Piscataway: IEEE, 2017: 1990-1994.
- [95] Gehrig D, Gehrig M, Hidalgo-Carrió J, et al. Video to events: Recycling video datasets for event cameras[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 3586-3595.
- [96] Yang Z, Wu Y, Wang G, et al. Dashnet: A hybrid artificial and spiking neural network for high-speed object tracking[J]. 2019, arXiv: 1909.12942.
- [97] Bottou L. Stochastic gradient descent tricks[M]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012: 421-436.
- [98] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. 2014, arXiv: 1412.6980.
- [99] Reddi S J, Kale S, Kumar S. On the convergence of adam and beyond[J]. 2019, arXiv: 1904.09237.
- [100] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[J]. 2015, arXiv: 1502.03167.
- [101] Dean J, Corrado G, Monga R, et al. Large scale distributed deep networks[C]. Advances in Neural Information Processing Systems 25. Red Hook: Curran Associates Inc., 2012: 1223-1231.
- [102] Grossberg S. Competitive learning: From interactive activation to adaptive resonance[J]. Cognitive Science, 1987, 11(1): 23-63.
- [103] Bi G Q, Poo M M. Synaptic modification by correlated activity: Hebb's postulate revisited[J]. Annual Review of Neuroscience, 2001, 24(1): 139-166.
- [104] Guyonneau R, VanRullen R, Thorpe S J. Neurons tune to the earliest spikes through stdp[J]. Neural Computation, 2005, 17(4): 859-879.
- [105] Diehl P U, Cook M. Unsupervised learning of digit recognition using spike-timing-dependent plasticity[J]. Frontiers in Computational Neuroscience, 2015, 9: 99.
- [106] Masquelier T, Thorpe S J. Unsupervised learning of visual features through spike timing dependent plasticity[J]. PLoS Computational Biology, 2007, 3(2): e31.
- [107] Kheradpisheh S R, Ganjtabesh M, Masquelier T. Bio-inspired unsupervised learning of visual features leads to robust invariant object recognition[J]. Neurocomputing, 2016, 205: 382-392.
- [108] Savarese S, Fei-Fei L. 3d generic object categorization, localization and pose estimation[C]. 2007 IEEE 11th International Conference on Computer Vision. Rio de Janeiro: IEEE, 2007: 1-8.
- [109] Leibe B, Schiele B. Analyzing appearance and contour based methods for object categorization[C]. Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Madison: IEEE, 2003: II-409.
- [110] Serre T, Wolf L, Bileschi S, et al. Robust object recognition with cortex-like mechanisms[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(3): 411-426.
- [111] Kheradpisheh S R, Ganjtabesh M, Thorpe S J, et al. Stdp-based spiking deep convolutional neural networks for object recognition[J]. Neural Networks, 2018, 99: 56-67.
- [112] Jedlicka P. Synaptic plasticity, metaplasticity and bcm theory[J]. Bratislavské Lekárske Listy, 2002, 103(4/5): 137-143.
- [113] Wade J J, McDaid L J, Santos J A, et al. SWAT: A spiking neural network training algorithm for classification problems[J]. IEEE Transactions on Neural Networks, 2010, 21(11): 1817-1830.
- [114] Bohte S M, Kok J N, La Poutré H. Error-backpropagation in temporally encoded networks of spiking neurons[J]. Neurocomputing, 2002, 48(1/2/3/4): 17-37.
- [115] Ghosh-Dastidar S, Adeli H. A new supervised learning algorithm for multiple spiking neural networks with application in epilepsy and seizure detection[J]. Neural Networks, 2009, 22(10): 1419-1431.
- [116] Ponulak F, Kasiski A. Supervised learning in spiking neural networks with ReSUMe: Sequence learning, classification, and spike shifting[J]. Neural Computation, 2010, 22(2): 467-510.
- [117] Widrow B, Hoff M E. Adaptive switching circuits[R]. Stanford: Electronics Laboratory, Stanford University, 1960: 7-22.
- [118] Florian R V. The chronotron: A neuron that learns to fire temporally precise spike patterns[J]. PloS One, 2012, 7(8): 1-27.
- [119] Gütig R, Sompolinsky H. The tempotron: A neuron that learns spike timing-based decisions[J]. Nature Neuroscience, 2006, 9(3): 420-428.
- [120] Mohemmed A, Schliebs S, Matsuda S, et al. Span: Spike pattern association neuron for learning spatio-temporal spike patterns[J]. International Journal of Neural Systems, 2012, 22(4): 1250012.
- [121] Cao Y, Chen Y, Khosla D. Spiking deep convolutional neural networks for energy-efficient object recognition [J]. International Journal of Computer Vision, 2015, 113(1): 54-66.
- [122] Diehl P U, Neil D, Binas J, et al. Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing[C]. 2015 International Joint Conference on Neural Networks. Piscataway: IEEE, 2015: 1-8.
- [123] Rueckauer B, Lungu I A, Hu Y H, et al. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification[J]. Frontiers in

- Neuroscience, 2017, 11: 682.
- [124] Hu Y, Tang H, Wang Y, et al. Spiking deep residual network[J]. 2018, arXiv: 1805.01352.
- [125] Severa W, Vineyard C M, Dellana R, et al. Training deep neural networks for binary communication with the whetstone method[J]. Nature Machine Intelligence, 2019, 1(2): 86-94.
- [126] Rathi N, Srinivasan G, Panda P, et al. Enabling deep spiking neural networks with hybrid conversion and spike timing dependent backpropagation[J]. 2020, arXiv: 2005.01807.
- [127] Lillicrap T P, Cownden D, Tweed D B, et al. Random synaptic feedback weights support error backpropagation for deep learning[J]. Nature Communications, 2016, 7(1): 1-10.
- [128] Baldi P, Sadowski P, Lu Z Q. Learning in the machine: Random backpropagation and the deep learning channel [J]. Artificial Intelligence, 2018, 260: 1-35.
- [129] Neftci E O, Augustine C, Paul S, et al. Event-driven random back-propagation: Enabling neuromorphic deep learning machines[J]. Frontiers in Neuroscience, 2017, 11: 324.
- [130] Neftci E O, Mostafa H, Zenke F. Surrogate gradient learning in spiking neural networks[J]. IEEE Signal Processing Magazine, 2019, 36(6): 51-63.
- [131] Lee J H, Delbruck T, Pfeiffer M. Training deep spiking neural networks using backpropagation[J]. Frontiers in Neuroscience, 2016, 10: 508.
- [132] Jin Y, Zhang W R, Li P. Hybrid macro/micro level backpropagation for training deep spiking neural networks[C]. Advances in Neural Information Processing Systems 31. Red Hook: Curran Associates Inc., 2018: 7005-7015.
- [133] Kheradpisheh S R, Masquelier T. S4nn: Temporal backpropagation for spiking neural networks with one spike per neuron[J]. 2019, arXiv: 1910.09495.
- [134] Wu Y, Deng L, Li G, et al. Spatio-temporal backpropagation for training high-performance spiking neural networks[J]. Frontiers in Neuroscience, 2018, 12: 331.
- [135] Lee C, Sarwar S S, Panda P, et al. Enabling spike-based backpropagation for training deep neural network architectures[J]. Frontiers in Neuroscience, 2020, 14: 119.
- [136] Zhao B, Ding R X, Chen S S, et al. Feedforward categorization on aer motion events using cortex-like features in a spiking neural network[J]. IEEE Transactions on Neural Networks and Learning Systems, 2015, 26(9): 1963-1978.
- [137] Tavanaei A, Maida A. BP-STDP: Approximating backpropagation using spike timing dependent plasticity[J]. Neurocomputing, 2019, 330: 39-47.
- [138] Neftci E, Das S, Pedroni B, et al. Event-driven contrastive divergence for spiking neuromorphic systems[J]. Frontiers in Neuroscience, 2013, 7: 272.
- [139] O'Connor P, Neil D, Liu S C, et al. Real-time classification and sensor fusion with a spiking deep belief network[J]. Frontiers in Neuroscience, 2013, 7: 178.
- [140] Stomatias E, Neil D, Galluppi F, et al. Scalable energy-efficient, low-latency implementations of trained spiking deep belief networks on SpiNNaker[C]. 2015 International Joint Conference on Neural Networks. Piscataway: IEEE, 2015: 1-8.
- [141] Sironi A, Brambilla M, Bourdis N, et al. HATS: Histograms of averaged time surfaces for robust event-based object classification[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2018: 1731-1740.
- [142] Lagorce X, Orchard G, Galluppi F, et al. HOTS: A hierarchy of event-based time-surfaces for pattern recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(7): 1346-1359.
- [143] Maass W, Natschläger 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.
- [144] Zhang Y, Li P, Jin Y, et al. A digital liquid state machine with biologically inspired learning and its application to speech recognition[J]. IEEE Transactions on Neural Networks and Learning Systems, 2015, 26(11): 2635-2649.
- [145] Schrauwen B, D'Haene M, Verstraeten D, et al. Compact hardware liquid state machines on fpga for real-time speech recognition[J]. Neural Networks, 2008, 21(2/3): 511-523.
- [146] Burgsteiner H, Kröll M, Leopold A, et al. Movement prediction from real-world images using a liquid state machine[J]. Applied Intelligence, 2007, 26(2): 99-109.
- [147] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets[J]. Neural Computation, 2006, 18(7): 1527-1554.
- [148] Bengio Y. Learning deep architectures for AI[J]. Foundations and Trends in Machine Learning, 2009, 2(1): 1-27.
- [149] Moradi S, Qiao N, Stefanini F, et al. A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (DYNAPS)[J]. IEEE Transactions on Biomedical Circuits and Systems, 2018, 12(1): 106-122.
- [150] Carnevale N, Hines M. The neuron book[M]. Cambridge: Cambridge University Press, 2006: 1-31.
- [151] Kobayashi C, Jung J, Matsunaga Y, et al. GENESIS 1.1: A hybrid-parallel molecular dynamics simulator with enhanced sampling algorithms on multiple computational platforms[J]. Journal of Computational Chemistry, 2017, 38(25): 2193-2206.
- [152] Gewaltig M O, Diesmann M. NEST (Neural simulation tool)[J]. Scholarpedia, 2007, 2(4): 1430.

- [153] Goodman D F, Brette R. Brian: A simulator for spiking neural networks in python[J]. *Frontiers in Neuroinformatics*, 2008, 2: 5.
- [154] Kasabov N K. NeuCube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data[J]. *Neural Networks*, 2014, 52: 62-76.
- [155] Bekolay T, Bergstra J, Hunsberger E, et al. Nengo: A python tool for building large-scale functional brain models[J]. *Frontiers in Neuroinformatics*, 2014, 7: 48.
- [156] Mead C. Neuromorphic electronic systems[J]. *Proceedings of the IEEE*, 1990, 78(10): 1629-1636.
- [157] Arthur J V, Boahen K A. Silicon-neuron design: A dynamical systems approach[J]. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2011, 58(5): 1034-1043.
- [158] Merolla P, Arthur J, Alvarez R, et al. A multicast tree router for multichip neuromorphic systems[J]. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2014, 61(3): 820-833.
- [159] Prezioso M, Merrih-Bayat F, Hoskins B, et al. Training and operation of an integrated neuromorphic network based on metal-oxide memristors[J]. *Nature*, 2015, 521(7550): 61-64.
- [160] Chu M, Kim B, Park S, et al. Neuromorphic hardware system for visual pattern recognition with memristor array and CMOS neuron[J]. *IEEE Transactions on Industrial Electronics*, 2014, 62(4): 2410-2419.
- [161] Park S, Chu M, Kim J, et al. Electronic system with memristive synapses for pattern recognition[J]. *Scientific Reports*, 2015, 5: 10123.
- [162] Pickett M D, Medeiros-Ribeiro G, Williams R S. A scalable neuristor built with mott memristors[J]. *Nature Materials*, 2013, 12(2): 114-117.
- [163] Wang Z R, Joshi S, Savel'ev S, et al. Fully memristive neural networks for pattern classification with unsupervised learning[J]. *Nature Electronics*, 2018, 1(2): 137-145.
- [164] StrukovDB, SniderGS, StewartDR, et al. The missing memristor found[J]. *Nature*, 2008, 453(7191): 80-83.
- [165] Garbin D, Bichler O, Vianello E, et al. Variability-tolerant convolutional neural network for pattern recognition applications based on OxRAM synapses[C]. 2014 IEEE International Electron Devices Meeting. Piscataway: IEEE, 2014: 28.4.1-28.4.4.
- [166] Ji Y, Zhang Y, Xie X, et al. Fpsa: A full system stack solution for reconfigurable rram-based nn accelerator architecture[C]. Proceedings of the 24th International Conference on Architectural Support for Programming Languages and Operating Systems. New York: Association for Computing Machinery, 2019: 733-747.
- [167] Merolla P A, Arthur J V, Alvarez-Icaza R, et al. A million spiking-neuron integrated circuit with a scalable communication network and interface[J]. *Science*, 2014, 345(6197): 668-673.
- [168] Esser S K, Merolla P A, Arthur J V, et al. Convolutional networks for fast, energy-efficient neuromorphic computing[J]. *Proceedings of the National Academy of Sciences*, 2016, 113(41): 11441-11446.
- [169] Preissl R, Wong T M, Datta P, et al. Compass: A scalable simulator for an architecture for cognitive computing[C]. SC'12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. Piscataway: IEEE, 2012: 1-11.
- [170] Amir A, Datta P, Risk W P, et al. Cognitive computing programming paradigm: A corelet language for composing networks of neurosynaptic cores[C]. The 2013 International Joint Conference on Neural Networks. Piscataway: IEEE, 2013: 1-10.
- [171] Esser S K, Andreopoulos A, Appuswamy R, et al. Cognitive computing systems: Algorithms and applications for networks of neurosynaptic cores[C]. The 2013 International Joint Conference on Neural Networks. Piscataway: IEEE, 2013: 1-10.
- [172] Hwu T, Isbell J, Oros N, et al. A self-driving robot using deep convolutional neural networks on neuromorphic hardware[C]. 2017 International Joint Conference on Neural Networks. Piscataway: IEEE, 2017: 635-641.
- [173] Wang B, Zhou J, Wong W F, et al. Shenjing: A low power reconfigurable neuromorphic accelerator with partial-sum and spike networks-on-chip[J]. 2019, arXiv: 1911.10741.
- [174] Guo S S, Wang L, Wang S Q, et al. A systolic SNN inference accelerator and its co-optimized software framework[C]. Proceedings of the 2019 on Great Lakes Symposium on VLSI. New York: ACM, 2019: 63-68.
- [175] Ju X, Fang B, Yan R, et al. An fpga implementation of deep spiking neural networks for low-power and fast classification[J]. *Neural Computation*, 2020, 32(1): 182-204.
- [176] Deng L, Wang G, Li G, et al. Tianjic: A unified and scalable chip bridging spike-based and continuous neural computation[J]. *IEEE Journal of Solid-State Circuits*, 2020, 55(8): 2228-2246.
- [177] Chen G K, Kumar R, Sumbul H E, et al. A 4096-neuron 1M-synapse 3.8-PJ/SOP spiking neural network with on-chip STDP learning and sparse weights in 10-nm finFET COMS[J]. *IEEE Journal of Solid-State Circuits*, 2019, 54(4): 992-1002.
- [178] Schemmel J, Brüderle D, Gröbl A, et al. A wafer-scale neuromorphic hardware system for large-scale neural modeling[C]. 2010 IEEE International Symposium on Circuits and Systems. Paris: IEEE, 2010: 1947-1950.
- [179] Qiao N, Mostafa H, Corradi F, et al. A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128 K synapses[J]. *Frontiers in Neuroscience*, 2015, 9: 141.

- [180] Yu S, Wu Y, Jeyasingh R, et al. An electronic synapse device based on metaloxide resistive switching memory for neuromorphic computation[J]. *IEEE Transactions on Electron Devices*, 2011, 58(8): 2729-2737.
- [181] Kuzum D, Jeyasingh R G D, Lee B, et al. Nanoelectronic programmable synapses based on phase change materials for brain-inspired computing[J]. *Nano Letters*, 2012, 12(5): 2179-2186.
- [182] Bichler O, Suri M N, Querlioz D, et al. Visual pattern extraction using energy-efficient “2-PCM synapse” neuromorphic architecture[J]. *IEEE Transactions on Electron Devices*, 2012, 59(8): 2206-2214.
- [183] Rast A D, Galluppi F, Jin X, et al. The leaky integrate-and-fire neuron: A platform for synaptic model exploration on the SpiNNaker chip[C]. *The 2010 International Joint Conference on Neural Networks*. Barcelona: IEEE, 2010: 1-8, DOI: 10.1109/IJCNN.2010.5596364.
- [184] Furber S B, Lester D R, Plana L A, et al. Overview of the SpiNNaker system architecture[J]. *IEEE Transactions on Computers*, 2012, 62(12): 2454-2467.
- [185] Navaridas J, Furber S, Garside J, et al. SpiNNaker: Fault tolerance in a power-and area-constrained large-scale neuromimetic architecture[J]. *Parallel Computing*, 2013, 39(11): 693-708.
- [186] Jin X, Rast A, Galluppi F, et al. Implementing spike-timing-dependent plasticity on SpiNNaker neuromorphic hardware[C]. *The 2010 International Joint Conference on Neural Networks*. Barcelona: IEEE, 2010: 1-8, DOI: 10.1109/IJCNN.2010.5596372.
- [187] Frenkel C, Lefebvre M, Legat J D, et al. A 0.086-mm²12.7-pJ/sop 64k-synapse 256-neuron online-learning digital spiking neuromorphic processor in 28-nm cmos[J]. *IEEE Transactions on Biomedical Circuits and Systems*, 2018, 13(1): 145-158.
- [188] Frenkel C, Legat J D, Bol D. MorphIC: A 65-nm 738k-synapse/mm² quad-core binary-weight digital neuromorphic processor with stochastic spike-driven online learning[J]. *IEEE Transactions on Biomedical Circuits and Systems*, 2019, 13(5): 999-1010.
- [189] Baek E, Lee H, Kim Y, et al. FlexLearn: Fast and highly efficient brain simulations using flexible on-chip learning[C]. *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*. New York: ACM, 2019: 304-318.
- [190] Lee D, Lee G, Kwon D, et al. Flexon: A flexible digital neuron for efficient spiking neural network simulations[C]. *2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture*. Piscataway: IEEE, 2018: 275-288.
- [191] Park J, Lee J, Jeon D. A 65nm 236.5 nJ/classification neuromorphic processor with 7.5/% energy overhead on-chip learning using direct spike-only feedback [C]. *2019 IEEE International Solid-State Circuits Conference*. Piscataway: IEEE, 2019: 140-142.
- [192] Bouvier M, Valentian A, Mesquida T, et al. Spiking neural networks hardware implementations and challenges: A survey[J]. *ACM Journal on Emerging Technologies in Computing Systems*, 2019, 15 (2): 1-35.
- [193] Rastegari M, Ordonez V, Redmon J, et al. XNOR-net: ImageNet classification using binary convolutional neural networks[C]. *Computer Vision—ECCV 2016*. Cham: Springer International Publishing, 2016: 525-542.

作者简介

胡一凡(1997—), 男, 博士生, 从事神经形态计算、机器学习理论与脉冲神经网络算法的研究, E-mail: huyf19@mails.tsinghua.edu.cn;

李国齐(1982—), 男, 副教授, 博士生导师, 从事类脑计算架构与类脑智能等研究, E-mail: liguqi@mail.tsinghua.edu.cn;

吴郁杰(1993—), 男, 博士生, 从事神经形态计算、机器学习理论与脉冲神经网络算法的研究, E-mail: wuyj16@mails.tsinghua.edu.cn;

邓磊(1990—), 男, 博士, 从事类脑计算、机器学习与芯片架构的研究, E-mail: leideng@ucsb.edu.

科研团队简介

清华大学李国齐副教授科研团队围绕类脑芯片, 研究匹配的芯片理论、计算架构和训练算法, 在 *Nature*、*Proceedings of the IEEE*、*IEEE TPAMI*、*IEEE TNNLS* 等顶级期刊, 以及 *ICLR*、*NeurIPS*、*AAAI* 等人工智能领域重要会议上发表论文 110 余篇, 出版类脑计算领域专著一部。团队成员是清华大学类脑计算中心研制的世界首款异构融合类脑芯片“天机芯”的骨干成员, 其中邓磊博士曾入选 2019 年《麻省理工科技评论》35 岁以下科技创新 35 人中国榜单、2017 年清华大学博士毕业生紫荆学者。“天机芯”荣获 2019 世界互联网 15 项领先科技成果奖, 并入选 2019 年中国科学十大进展。李国齐科研团队承担并参与了科技部 2030 新一代人工智能重点研发项目课题、国家自然科学基金面上项目、重点项目、科技部战略性新兴产业国际科技创新合作重点专项等科研项目。

(责任编辑: 李君玲)