# ****引言****

## 研究背景

为了开创认知计算的新时代，我们正在开发TrueNorth（图1），这是一个非冯·诺依曼的、模块化、并行、分布式、事件驱动的、可扩展的架构——其灵感来源于生物大脑的功能、低功耗和小体积。TrueNorth是一个多功能的基质，用于集成多模态、符号前、传感器-执行器系统的时空、实时认知算法。TrueNorth由可配置的神经突触核心的可扩展网络组成。每个核心将记忆（“突触”）、处理器（“神经元”）和通信（“轴突”）紧密地结合在一起，核心间的通信通过基于消息传递的全有或全无的尖峰事件进行。

最近，我们取得了一些重要的里程碑：首先，我们在45纳米硅片上展示了256个神经元、64k/256k个突触的神经突触核心，这在2011年12月的《科学》封面上得到展示；其次，我们展示了多个实时应用；第三，我们开发了一个名为Compass的TrueNorth架构模拟器，模拟了超过20亿个神经突触核心，超过了10^14个突触；第四，我们可视化了猕猴大脑的长距离连接——映射到TrueNorth架构上，这在《科学》和《计算机协会通讯》的封面上得到展示。

我们在这三篇论文中揭示了一系列相互关联的创新。在本文中，我们提出了一个多功能且高效的数字尖峰神经元模型，它是TrueNorth架构的构建模块。在两篇伴随的论文中，我们介绍了一种用于分层组合和配置认知系统的编程范式，这对程序员来说是有效的，对TrueNorth架构来说也是高效的[9]，同时我们还展示了一套算法和应用，证明了TrueNorth架构的潜力和编程范式的价值[10]。

## 研究动机

在19世纪末，海因里希·冯·瓦尔德耶-哈尔茨（Heinrich von Waldeyer-Hartz）将大脑中的生物细胞命名为“神经元”。在过去的一百年中，这些神经元已经在不同抽象层次上被建模（见表I为简明时间线）。这些模型从现象学的，目标是使用简单的数学抽象来捕捉神经元的输入输出行为（例如，McCulloch-Pitts [11]），到生物物理学的，目标是模拟神经元膜的电学生理特性（例如，Hodgkin-Huxley [12]）。在TrueNorth系统架构的背景下，我们寻求一个神经元模型，它能够平衡能力（从计算的角度）和成本（从实施的角度）的双重目标，这两者都是神经元模型潜在复杂性的函数。神经元的能力应该足以支持有用和有趣的认知算法[10]，而成本应该不超过必要的功率，面积和速度。

## 主要贡献

我们的主要贡献，是在能力与成本之间谨慎地取得平衡，从数字编码、数学和现象学的角度，开发了一个简单、可重配置、多功能的神经元模型。这个模型对于广泛的应用具有明确的输入输出行为，具有异质性和在群体和时间上的可变性，同时消耗少量晶体管，适用于在密集的CMOS工艺中实现，并且可以可重复和可预测地模拟，实现硬件和仿真之间的一对一等效模拟。我们总结了所提出的神经元模型的构建、能力、组合性和成本。

文章结构：在第二节中，我们从经典的LIF神经元开始，为模型的输入、状态和输出增加了可配置和可复制的随机性。具体来说，我们引入了随机突触输入、漏电和阈值，使得在群体和时间上能够展现出丰富的动态。接着，我们引入了四种漏电模式，这些模式会影响内部状态的动态，使得神经元对相同输入可以有截然不同的反应。具体而言，我们允许漏电从膜电位中减去或者加上，同时允许膜电位偏离或趋向于一个静息电位。此外，我们引入了两种阈值模式，允许确定性或随机性阈值，因此即使在相同的累积膜电位下，神经元也可以有不同的发射方式。最后，我们引入了六种重置模式，这些模式决定了发射后膜电位的值，从而实现丰富的有限状态转移行为。

功能及构成：在第三节中，我们利用神经元模型的参数化方法，展示了计算功能的广泛多样性；例如算术、控制、数据生成、逻辑、记忆、经典神经元行为、信号处理和概率计算。模型可以支持多种神经编码，包括速率、群体、二进制和尖峰时间编码，从而允许神经元间通信的丰富语言。为了便于使用这些神经元，我们创建了一个参数化和特征化的神经元功能库，包含50多个元素，包含基本构建块。受到精简指令集（RISC）的启发，通过组合多个神经元，我们可以从简单的库元素合成极其丰富和多样化的复杂计算和行为。

虽然我们的设计目标并不是要精确复制神经科学观察到的神经元行为，但在第四节中，我们非常意外地展示了我们能够使用少量的基本神经元定性地复制Izhikevich动态神经元模型[24]的20种行为。

成本：根据设计，我们的神经元模型仅使用简单的加法和多路选择算术/逻辑单元，避免了使用复杂的函数单元，如乘法、除法和指数运算。它可以使用定点算术实现，避免了复杂的浮点电路。因此，当映射到ASIC标准单元库中，在最先进的硅工艺中制造时，神经元模型只需1272个门（924个门用于模型计算，348个门用于随机数生成器）即可实现。从操作角度来看，加法和多路选择电路本质上比更复杂的算术函数电路功耗低。值得注意的是，与硅的速度相比，神经元在实时操作中的发射频率相对缓慢，使我们有可能通过三种方式降低成本。首先，我们可以重用物理算术电路，大幅减少总的实现面积。其次，我们可以门控功率，当这些电路处于静止状态时关闭电源，减少总功耗。第三，神经元可以以事件驱动的方式实现，使其活跃功耗基于它必须处理的尖峰事件数量（例如，见[29]）。

# II. 神经元详述

我们的神经元模型基于LIF神经模型，我们以几种方式对其进行了增强。我们首先简要回顾经典的LIF神经模型，然后深入描述我们的神经元模型。

## A. LIF神经元

LIF神经元模型的操作由五个基本操作描述：1. 突触积分，2. 漏电积分，3. 阈值，4. 尖峰发射，以及 5. 重置。LIF神经元模型在图2中的方程(1)-(6)中总结了一般情况。对于第j个神经元在第t个时间步，膜电位Vj(t)是前一个时间步的膜电位Vj(t − 1)和突触输入的总和。对于每个突触，突触输入是当前时间步的尖峰输入xi(t)与有符号突触权重si的乘积之和。积分后，LIF神经元模型从膜电位中减去漏电值λj。在线性漏电中，这个常数在每个时间步被减去，不考虑膜电位或突触活动。这个操作作为对神经动态的恒定偏置。然后，LIF神经元模型将当前时间步的膜电位Vj(t)与神经元阈值αj进行比较。如果膜电位大于或等于阈值电压，神经元就会发射一个尖峰并重置其膜电位。在典型情况下，重置电压Rj为零。这个基本的神经计算模型，如图3所示，可以用来生成多种功能和行为。

## B. 完整神经元模型

我们神经元模型的完整规范在图4中的方程(10)-(19)中给出。符号在表II中总结。表的下半部分的符号是用户可配置的参数，而表的上半部分的符号是系统变量。规范使用了以下子函数：

signum函数：

sgn(x) =

如果 x < 0，则为 -1；

如果 x = 0，则为 0；

如果 x > 0，则为 1

(7)

条件随机评估的二元比较操作（见第二节-D）：

F(s, ρ) = (

如果 |s| ≥ ρ，则为 1；

否则为 0

) (8)

二进制位与操作：&

狄拉克δ函数：

δ(x) = (

如果 x = 0，则为 1；

否则为 0

) (9)

神经元操作的细节将在以下子节中介绍。

## C. 突触积分：交叉开关

神经突触核心包括一个突触交叉开关，用于连接轴突和神经元（图1）。传入的尖峰信号到达，目标是某个轴突，然后激活该轴突上所有的活跃突触。突触积分方程(1)变为方程(10)，其中Ai(t)是第i个轴突在时间t的活动，wi,j是突触交叉开关矩阵。轴突活动Ai(t)在当前（第t个）时间步有尖峰存在时为1，否则为0。突触交叉开关矩阵中的条目wi,j，如果第i个轴突和第j个神经元之间存在突触连接则为1，否则为0。

为了在使用二进制突触交叉开关的同时实现多值突触，我们为每个轴突分配了一个类型Gi。然后每个神经元对该轴突类型都有一个单独的有符号整数权重。在当前实现中，存在四种可能的轴突类型{Gi ∈ 0, 1, 2, 3}，每个神经元都有四个与这四种轴突类型Gi相关的有符号权重sGi j。当任何Gi类型的轴突上到达一个尖峰时，具有该轴突上突触的神经元将整合其Gi类型的有符号权重sGi j，从而得到突触积分求和：∑255 i=0 Ai(t) wi,j sGi j。（这个方程假设轴突类型的突触配置位设置为确定性模式，bGi j = 0。）

## D. 随机突触和漏电积分

对于每个神经元，每个突触权重和漏电都有一个配置位，分别为bGi j和cλ j，其中将位设置为0选择确定性模式，设置为1选择随机模式。对于随机突触和漏电积分，操作如下：每次发生有效的突触或漏电事件时，神经元抽取一个均匀分布的随机数ρj。如果突触权重sGi j或漏电权重λj大于或等于抽取的随机数ρj，则神经元整合{−1, +1}，否则不整合。这种行为可以用以下数学方式描述。整合的值{−1, +1}由相应突触的符号：sgn(sGi j)或漏电：sgn(λj)决定。随机操作使用二元比较运算符F(s, ρ)表示，其中s是标量值，ρ是随机数。如果标量值大于或等于随机数，则F(s, ρ)返回1，否则返回0，根据方程(8)。结合确定性模式和随机模式，突触积分的完整方程见方程(10)，漏电积分的完整方程见方程(12)。

## E. 漏电模式

我们扩展了线性漏电，使其可以为正或为负，并引入了一种“漏电反转”模式。在标准的漏电操作中，有符号的漏电总是被直接整合，而不考虑膜电位的值。漏电总是正的或负的，导致产生一个单调增加或减少膜电位的偏置。在漏电反转模式下，当膜电位高于零时，有符号的漏电被直接整合，而当膜电位低于零时，其符号被反转。（在零电位时，膜电位不会发生漏电，保持为零。）这种机制创造了两种漏电反转模式：一种是收敛漏电，其中神经元膜电位从上方（Vj > 0）和下方（Vj < 0）都向零泄露；另一种是发散漏电，其中神经元膜电位从零向上方向正阈值（+αj）和向下方向负阈值（−βj）泄露。这些漏电模式在图5中进行了总结。我们使用漏电反转参数ϵj来在选择的模式之间进行切换。漏电还支持随机模式，如前一节II-D所描述的。随机模式可以与四种漏电模式中的任何一种结合使用。通过使用参数cλj选择确定性或随机模式，从而得出漏电积分的完整方程(11)和(12)。

## F. 阈值

除了我们为漏电积分-发射神经元（3）描述的正阈值外，我们还引入了一个负阈值−βj，用于神经元膜电位积分低于零的情况。这个负阈值在被越过时有两种不同的行为。在第一种情况下，负阈值是一个地板，所以当膜电位Vj穿过−βj时，Vj保持在−βj。在第二种情况下，我们定义了当膜电位Vj穿过−βj时发生“反弹”。在这种情况下，Vj被重置为负重置值−Rj。当我们穿过−βj以下时，不会生成输出尖峰。这个特性允许一对神经元保持同步，参数反转（见图6）。例如，我们可以定义一个开关神经元对，其中一个神经元在接收到开启刺激时发射，而关闭神经元在刺激消失时发射。带有“反弹”的负阈值确保两个神经元的膜电位将是彼此的精确倒影副本。我们通过设置κj = 1来选择地板行为，通过设置κj = 0来选择“反弹”行为，如(17)所定义。

## G. 随机阈值

神经元模型支持一个随机阈值ηj，该阈值被加到确定性阈值αj和βj上。随机阈值值ηj由Mj和随机数生成器值ρTj的按位与操作形成，如方程(13)所示。掩码Mj是一个可配置宽度的全1掩码，从最不重要位开始。这个掩码缩放了添加到阈值的随机数ηj的范围。例如，十六进制掩码值为0x000F产生从0到15的均匀随机数，而值为0x01FF产生从0到511的均匀随机数。掩码为零时，随机值总是零，有效地禁用了阈值值的随机部分。正阈值检查与随机阈值的方程是(14)，而负阈值检查(17)只在“反弹”模式（κj = 0）下包括随机阈值。

## H. 重置模式

接下来我们介绍三种控制神经元重置行为的模式：正常模式（γj = 0）、线性模式（γj = 1）和非重置模式（γj = 2）。在正常模式（γj = 0）下，行为遵循标准的积分-发射模型，即当穿过正阈值并发射一个尖峰后，膜电位Vj被设置为重置电压Rj（14）。高于阈值的任何残余电位都被丢弃。例如，如果αj = 100且Vj在单个时间步内整合到110，神经元发射，重置为Rj，而高于阈值的10个单位电位被忽略。Rj可以被赋值为正、负或零值。在线性模式（γj = 1）下，高于阈值的残余电位不被丢弃。例如，如果αj = 100且Vj在单个时间步内整合到110，神经元发射，并重置为高于阈值的电位：Vj = 10。这种模式下不使用Rj。非重置模式（γj = 2）是一种特殊模式，旨在与随机阈值（在第II-G节中描述）一起使用，创建了一种随机神经元类型。当穿过随机阈值（αj + ηj）时，神经元不会重置其膜电位。在这种模式下，必须使用突触或漏电积分将膜电位拉低到阈值以下。使用克罗内克δ函数表示法（9），正重置方程是(16)，负重置方程是(18)。配置模式在表III中总结。

公式推导

连续时间的LIF模型：

连续时间的LIF神经元模型通常由以下微分方程描述：

手机屏幕截图

描述已自动生成

离散化：

我们将时间离散化为等间隔的步长 Δt，利用差分逼近

手机屏幕截图

描述已自动生成

数值解：

通过重新排列方程，我们可以解出下一个时间步的膜电位

这是数字型LIF神经手机屏幕截图

描述已自动生成元模型的更新方程，它描述了神经元在每个时间步的膜电位变化。

阈值检测和重置：

在模拟神经元的动作电位时，我们检查膜电位是否超过了阈值。如果超过了阈值，就发生了动作电位，然后将膜电位重置为初始值。

HDL仿真

