Neuromorphic Computing for High Energy Physics

Rebecca Carney
What defines neuromorphic computing?

Traditional von Neumann architecture

Distributed, parallel architecture

Binary n-bit words

Instruction set

Temporally separated spikes

Connectivity and weight model file
Three neuromorphic hardware projects

2008: NeuFlow @ NYU (J. LeCunn)

2013-18: Neuromorph and Brainstorm @ Stanford

2011: Spikey @ Heidelberg

2016: Spinnaker 0.5M core machine @ HBP/Manchester

Some examples

1. **Spinnaker**: scalable, low-power units
2. **Brainstorm**: NEF and populations of neurons
3. **Spikey**: analogue spikes, neuroplasticity, 10k speedup over real-time

**NeuFlow**: Conv. neural network on-chip. Not really neuromorphic.. but an example of NN hardware!

More neuron-inspired computing at NICE 2016 [link]
IBM's neuromorphic chip

Memory
Controller
Neuron
Scheduler
Router

TrueNorth Chip
64 x 64 cores

Test board

- Low power regime, therefore: 10's of mW, 1kHz throughput
- Bounded asynchrony (to an extent)
- Fully digital
Three steps are performed each iteration in the neuron:

1. **Weighted Input**: 
   - $x_{in0}$ with weight $-2$
   - $x_{in1}$ with weight $+5$

2. **Neuron Potential Calculation**: 
   - $x_{out} = x_{in0} + x_{in1}$

3. **Threshold Comparison**: 
   - If $x_{out} >$ threshold, output a spike.
   - Otherwise, reset the neuron potential.

**Graphical Representation**:
- **Neuron Potential**: A line graph showing the neuron potential changes over time.
- **Input Spikes**: A horizontal line representing the input spikes.
- **Output Spikes**: A vertical line representing the output spikes.
- **Threshold**: A horizontal line indicating the threshold level.

**Diagram**:
- A neuron model with inputs and output connections.
- The neuron potential changes are shown as a function of time, with input spikes and output spikes indicated.
- The threshold level is marked as 25.
Axons, neurons, synapses, and the crossbar

Axons are the xbar input

Weights and threshold applied here

Axons = input + label
Neuron = computation + output
Synapses = connecting axons and neurons
Crossbar = synapse frame
Steady-state KF as linear update

$$x(t + \Delta t) = Ax(t) + By(t)$$

$x = \text{state prediction (position, velocity, accn)}$

$y = \text{measurement (position, velocity, accn)}$
The KF Corelet xbar

Input

Output

$X^{+}_{in}$

$X^{-}_{in}$

$By^{+}$

$By^{-}$

$X^{+}_{out}$ $X^{-}_{out}$

By +1 -1
Quick explanation of plots to come

Kalman filter tracking on TrueNorth

NSCS KF vs numerical KF for \(\sin(x)\) - 1k tick encoding window, 1ms tick

- Amplitude [arb]
- TrueNorth state estimate
- 'Ideal' numerical KF state estimate
- Input state measurement
- Numerical estimate - spiking estimate
- Unscaled residuals [arb]
- Time [ms]

Numerical measurements
NSCS

10/12/16 rcarney@lbl.gov
Sample rate of Kalman filter

Inherent minimum sample rate for working Kalman filter

- Both the spiking and ideal Kalman filters are unable to fully track the sine wave at this sampling rate.
- ~4% error on amplitude estimate when compared to ideal Kalman filter
- ~<1% error on amplitude estimate when compared to ideal Kalman filter
- Increasing the sampling rate improves both the ideal and spiking KF performance.

1D Projectile tracking: 1000 tick encoding, 1e-3 process noise, 1e-1 meas. noise, 1e-3 sampling rate

<table>
<thead>
<tr>
<th>Amplitude [arb]</th>
<th>Time [ms]</th>
<th>Unscaled residuals [arb]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.1</td>
<td>200</td>
<td>0.01</td>
</tr>
<tr>
<td>0.2</td>
<td>400</td>
<td>0.02</td>
</tr>
<tr>
<td>0.3</td>
<td>600</td>
<td>0.03</td>
</tr>
<tr>
<td>0.4</td>
<td>800</td>
<td>0.04</td>
</tr>
<tr>
<td>0.5</td>
<td>1000</td>
<td>0.05</td>
</tr>
</tbody>
</table>

- 200 samples per period
- 2000 samples per period
Rate encoding

In binary:
100 != 010 != 001

In unary:
100 == 010 == 001

x = 1.0
x = 0.4

\( t = 20 \)
\( b = 20 \)

Rate encoding

\( x = 0.4 \)

Spatial encoding

\( x = 1.0 \)

\( t = 20 \)

b = blocks of neurons

\( t = 1 \)

\( t = 1 \)

100
010
001

In binary:

100 != 010 != 001
1D Projectile motion: 3-dim state tracking

NSCS KF vs numerical KF for 1D projectile - 10k tick encoding window, 1ms tick

Input state measurement

‘Ideal’ numerical KF state estimate

TrueNorth state estimate

Quick explanation of more plots
Effect of various encoding windows on TrueNorth Kalman Filter

- A low noise system with a high sampling rate already performs well with a 100 tick encoding window.
- With a 10k tick encoding window the state is better represented and outstanding residuals are reduced.
To represent a trained weight of 0.51273:

8-bit neuron weight: \( s = 64 \)

18-bit neuron threshold: \( \alpha = 125 \)

Using 1 axon, trained weight to 3 s.f.

(15)-bit neuron weight: \( s = 16,801 \)

18-bit neuron threshold: \( \alpha = 32,768 \)

Using 203 axons, trained weight to 5 s.f.
Effect of register size on TrueNorth Kalman Filter

- Encoding trained KF weights in a single axon-neuron connection.
- Encoding trained KF weights using 203 axon-neuron connections.
• A steady-state Kalman filter has been implemented in IBM’s TrueNorth chip for the first time.

• We have adapted CPU’s, GPU’s and FPGA’s for our needs. Could neuromorphic chips feature in our toolkit in 10 years time?

Questions?
Backup: NSCS chip simulation vs chip output

NSCS chip simulation exactly matches NS1e output.
2-state system

\[ Ax = \begin{pmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \end{pmatrix} = \begin{pmatrix} a_{00}x_0 + a_{01}x_1 \\ a_{10}x_0 + a_{11}x_1 \end{pmatrix} \]
2-state system

\[Ax + By = \begin{pmatrix} a_{00}x_0 + a_{01}x_1 \\ a_{10}x_0 + a_{11}x_1 \end{pmatrix} + \begin{pmatrix} (By)_0 \\ (By)_1 \end{pmatrix}\]

Here, the neuron weights and thresholds are both 1, as we just want to use the crossbar to add values, not multiply.
Real-life neurons spike

Real spikes have a finite width, shape, and refractory period. Neuromorphic chips do exist that encode information in the shape of the neuron spikes as well.

TrueNorth decided not to do this and keep their chip fully digital.