Neural Computing for Scientific Computing Applications: More than Just Machine Learning

Neuromorphic Computing Workshop, Knoxville TN, 7/17/17
Brad Aimone (jbaimon@sandia.gov), Ojas Parekh, William Severa
Sandia National Laboratories
Hardware Acceleration of Adaptive Neural Algorithms (HAANA) 2014-2017
Neural inspired computing lacks theoretical foundation to translate between fields

Von Neumann computing

Quantum computing

Classic Algorithms

Materials Science & Device Physics

Quantum Algorithms

Quantum physics
Neural inspired computing lacks theoretical foundation to translate between fields

What is the brain as inspiration?
Established conventional wisdom: neural-inspired computing is bad at math

Why?

- It is a challenge to separate *brains* (cognitive capability) from *neurons* (low-energy mechanism)

- Belief that neurons are noisy

- Moore’s Law – It has always been easier to wait for faster processors than to re-invent numerical computing on specialized parallel architecture
Theoretical models of the brain do not need to capture everything

- Shallow Depth Inference
- Rapid, Stable Learning
- Context Modulated Decisions
- Memory Capacity
- Power Efficient
- Distributed Representations
- Not Consistently Logical
- Bad at Math

= Implicit in model
= Not implicit in model
Spiking neurons are a more powerful version of classic logic gates.

Spiking threshold gates provide high degree of parallelism at very low power.

High fan-in

Compute more powerful logic functions

Incorporate time into logic
Are threshold gates and spiking neurons equivalent?
HAANA has produced a number of spiking numerical algorithms

- Cross-correlation
  - Severa et al., ICRC 2016

- SpikeSort
  - Verzi et al., submitted
  - SpikeMin
  - SpikeMax

- SpikeOptimization
  - Verzi et al., IJCNN 2017

- Sub-cubic (i.e., Strassen) constant depth matrix multiplication
  - Parekh et al., submitted
A Velocimetry Application

- A motivating application is the determination of the local velocity in a flow field
- The maximal cross-correlation between two sample images provides a velocity estimate
- SNN algorithms are straightforward; exemplify core concepts
  - Highly parallel
  - Different neural representations
  - Modular, precise connectivity
  - Time/Neuron tradeoff
Time Multiplexed Cross Correlation

- Time-coded Inputs: Temporal Coding
- Feature Detectors: Rate Coding
- Integrators: Latency Coding

Temporal Coding: $O(n)$ neurons; $O(n)$ runtime

Parallelize inputs and corresponding timesteps to achieve $O(n^2)$ neurons; $O(1)$ runtime

Fires regularly; forces integrator to fire

Severa et al., ICRC 2016
Cross-Correlation Exhibits Time/Neuron Tradeoff

- Exchange Time Cost $\leftrightarrow$ Neuron Cost
- Complexity is unchanged
- **Neurons:** $O(n^2) \leftrightarrow O(n)$
- **Time:** $O(1) \leftrightarrow O(n)$

Severa et al., ICRC 2016
“Neural” network for matrix multiplication

Standard:
8Ms, 4As → $O(N^3)$

Strassen:
7Ms, 18A/Ss → $O(N^{2+\varepsilon})$

Strassen formulation of matrix multiply enables less than $O(N^3)$ neurons – resulting in less power consumption

Parekh et al., submitted
Strassen multiplication in neural hardware may show powerful advantages.

<table>
<thead>
<tr>
<th>Method</th>
<th>Depth</th>
<th># Gates</th>
<th>Value of ( \varepsilon )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>3</td>
<td>( O(N^3) )</td>
<td>–</td>
</tr>
<tr>
<td>“Direct” Strassen</td>
<td>( d )</td>
<td>( O(N^{\omega+\varepsilon}) )</td>
<td>( 1/d )</td>
</tr>
<tr>
<td>Refined Strassen</td>
<td>( d )</td>
<td>( O(N^{\omega+\varepsilon}) )</td>
<td>( O(1/c^d) )</td>
</tr>
<tr>
<td>Non-constant Depth</td>
<td>( O(\log \log N) )</td>
<td>( O(N^{\omega}) )</td>
<td>–</td>
</tr>
</tbody>
</table>

Example: Triangle Counting in Graphs

- **Input:** adjacency matrix of a graph with entries either 0 or 1
- **Output:** does the graph have \( \geq T \) triangles? Applications to social network analysis

Point at which Strassen method becomes useful

Conventional

Strassen-TG

Parekh et al., submitted
Theoretical models of the brain do not need to capture everything

- Shallow Depth Inference
- Rapid, Stable Learning
- Context Modulated Decisions
- Memory Capacity
- Power Efficient
- Distributed Representations
- Not Consistently Logical
- Bad at Math

- = Implicit in model
- = Not implicit in model

Spiking Threshold Gates

Neuroscience Systems Model
How do we take advantage of neuroscience?

Primate visual cortex
Felleman and Van Essen, 1991

Hippocampus
View of brain as computing system
Cortex – hippocampus interaction can extend AI to more complete computing system

- Cortex learns to process sensory information at different levels of abstraction
  - Similar to deep learning, though more sophisticated in biology
- Hippocampus would be a content addressable memory
  - Provide context and retrieval cues to guide cortical processing
A robust hippocampus abstraction can bring a complete neural system to AI

- Desired functions
  - Learn associations between cortical modalities
  - Encoding of temporal, contextual, and spatial information into associations
  - Ability for “one-shot” learning
  - Cue-based retrieval of information

- Desired properties
  - Compatible with spiking representations
  - Network must be stable with adaptation
  - Capacity should scale nicely
  - Biologically plausible in context of extensive hippocampus literature
  - Ability to formally quantify costs and performance
Formalizing CAM function one hippocampus layer at a time

- Constraining EC inputs to have “grid cell” structure sets DG size to biological level of expansion (~10:1)
- Mixed code of broad-tuned (immature) neurons and narrow tuned (mature) neurons confirms predicted ability to encode novel information

William Severa, NICE 2016
Severa et al., Neural Computation, 2017
Brain uses a different approach to processing in memory
Questions?

Thanks to Sandia’s LDRD HAANA Grand Challenge and the DOE NNSA Advanced Simulation and Computing program