November 19th, 2025
To: Jim York
From: Jackson Mowry
Subject: Information and Recommendation for New Areas of Neuromorphic Research

As we begin to move into the Winter season I put together a report for new areas of research that I or our lab should potentially explore during the coming months. The purpose of this report is to provide you with three exciting new applications of Neuromorphic Computing so that we can learn what we can contribute to the field. You may already be familiar with machine learning and artificial intelligence, which is a great start into understanding the field of Neuromorphic Computing. Luckily many of those same concepts will transfer over into the field. This report will contain summaries of three different Neuromorphic applications, and then a recommendation for where our communications research can fit in toimprove the state of the art.

NAP: Neuromorphic Artificial Pancreas

This research explores using small spiking neural networks (SNNs) to control insulin delivery for people with Type 1 diabetes, aiming to develop a low-power, embedded artificial pancreas system. The work titled “NAP: Neuromorphic Artificial Pancreas" written by Rizzo et. al. explores networks which have been trained to decide how much insulin to give every 3 minutes based on continuous glucose monitor (CGM) data. SNNs are a new wave of computing in the post Moore's law era, representing computations as biological networks of neurons and synapses that store information in the weights and delays between connections.

Key points

The researchers trained many SNNs using evolutionary optimization and reinforcement learning. They decided to use previous research to guide their decisions around network hyperparameters, leaving more time for exploration of application specific details.

The insulin doses controlled by the networks could only change in fixed steps of 0.15 Units per output spike, which limited fine control. This sometimes caused the network to give too much insulin ("overbolus") or made it hard to maintain a steady basal insulin level, especially for younger patients.

Blood glucose data from both adults and children showed that the networks were generally effective at keeping blood sugar in the healthy range (called the "euglycemic" range), but children's glucose levels were more variable. The CGM sensor data is noisy and sometimes inaccurate. The networks were able to handle this noise fairly well without overreacting too much and causing dangerous blood sugar drops.

Compared to other reinforcement learning methods tested on similar diabetes simulations, the small spiking neural networks performed as well or better, while being much smaller and simpler. This makes them ideal for use on tiny, low-power devices. This is likely due to the fact that SNNs can store much more information that a traditional "neuron" in an ANN. SNNs are able to store both a weight and a delay value, allowing the networks to take advantage of the temporal dimension in their computation, a fact that led to the success of these networks.

The researchers deployed the best adult network on a Raspberry Pi Pico microcontroller and measured its power use. The network processed a full day's worth of glucose readings very quickly and used very little energy overall—about 267 joules per day. This suggests such a system could run efficiently on a small embedded device.

The small size (only 5 neurons and 5 connections) also makes the network easier to understand and analyze compared to large neural networks.

Key Takeaways

Tiny spiking neural networks can effectively control insulin dosing in real time, and they are efficient enough to run on low-power embedded hardware. This supports the idea that such networks could be used to build a neuromorphic artificial pancreas—an implantable or wearable device that automatically regulates blood sugar for people with diabetes.

Embracing the Hairball: An Investigation of Recurrence in Spiking Neural Networks for Control

The study focuses on leveraging recurrence—a mechanism where outputs from neurons are fed back as inputs—to enhance performance in tasks requiring memory and temporal processing. “Embracing the Hairball: An Investigation of Recurrence in Spiking Neural Networks for Control” was written by Scuman et. al, the head of the fantastic TENNLab at University of Tennessee Knoxville. They showed how memory plays a crucial role in control tasks by enabling the network to retain information about past states and use it for future decision-making. If an agent is able to remember where it came from in the past, these memories can help guide the decision making process, allowing for long term goals that aid in completing complex tasks.

Why Recurrent Spiking Neural Networks

Traditional neural networks often require a large number of neurons to handle complex tasks, leading to higher power consumption. Recurrent SNNs can achieve comparable or superior results with fewer neurons, making them more suitable for low-power applications such as neuromorphic computing. Recurrence enables these networks to process temporal information and maintain memory, which is essential for control tasks like balancing or navigation.

Challenges with Recurrence

Training recurrent networks is inherently more challenging than training feed-forward ones due to the need for specialized algorithms capable of handling feedback loops. Neuromorphic hardware has historically been optimized for feed-forward architectures, which has limited the exploration and implementation of recurrence-based designs.

Methods Used

The study employed evolutionary algorithms to train recurrent SNNs: LEAP which optimizes parameters for feed-forward networks, and EONS which evolves both the structure and parameters of recurrent networks. These methods were tested on three control tasks. Pole Balancing (with and without velocity inputs) a classic control task where a pole must be balanced on a moving cart. Pole Balancing without velocity inputs is a specific task that requires memory in order to find an optimal solution Bipedal Walker (OpenAI Gym) A simulated robot navigating uneven terrain.

Results

Recurrent SNNs consistently outperformed feed-forward networks across all tasks, particularly in scenarios requiring memory (e.g., pole balancing without velocity inputs). These networks demonstrated better efficiency by achieving high performance with fewer neurons and synapses, highlighting their suitability for resource-constrained environments. Visualizations of the best-performing networks revealed highly interconnected “hairball-like” structures that were difficult to interpret but proved effective in solving complex tasks.

Future Directions

The researchers hope to extend the application of recurrence-based SNNs to other domains such as anomaly detection and spatiotemporal classification. They also strive to develop methods to improve the explainability of recurrent connections, facilitating better understanding of their contributions to decision-making processes.

Key Takeaways

Recurrent spiking neural networks show great promise for control tasks by reducing neuron count while improving performance. Their ability to efficiently process temporal information makes them ideal for power-constrained applications like robotics and neuromorphic computing. However, challenges remain in training these networks due to their complex structures and limited support from current hardware. Further research into explainability and broader applications could unlock their full potential.

If you’re unfamiliar with neural networks or machine learning, this research can be understood as an effort to design “brain-like” systems that solve problems more efficiently than traditional computer programs by mimicking how our brains process information over time.

The RISP Neuroprocessor - Open Source Support for Embedded Neuromorphic Computing

The paper titled “The RISP Neuroprocessor - Open Source Support for Embedded Neuromorphic Computing” by authors James S. Plank, Keegan E. M. Dent, Bryson Gullett, Charles P. Rizzo, and Catherine D. Schuman introduced the RISP neuroprocessor as an open-source tool designed to advance research and development in neuromorphic computing.

The RISP Neuroprocessor

A simplified neuromorphic processor that uses “integrate-and-fire” neurons and synapses with discrete delays. It enables researchers to simulate and implement neuromorphic networks on hardware like Field-Programmable Gate Arrays (FPGAs).

Applications

Examples include solving problems like balancing a cart-pole system, computing mathematical functions like sine, recognizing patterns (bars-and-stripes), and clustering data using DBSCAN algorithms. These applications demonstrate how neuromorphic computing can handle complex tasks efficiently.

Advantages of RISP

Open-source tools make it accessible for researchers without requiring expensive hardware or proprietary solutions. The approach focuses on embedding specific networks directly into FPGA hardware, optimizing resource usage compared to general-purpose neuroprocessors.

Challenges

Communication between the FPGA and host computer is currently a bottleneck due to limitations in UART communication protocols. Current communication protocols require interaction at every neuroprocessor timestep, even when output spikes are sparse or unnecessary.

In summary, this paper introduces the RISP neuroprocessor as a practical tool for exploring neuromorphic computing in embedded systems. By leveraging open-source software and hardware implementations, researchers can create energy-efficient systems capable of handling complex tasks while addressing current challenges in communication efficiency.

Analysis of the Field

Based on the articles discussed I believe that the field of Neuromorphic Computing will be moving towards both low power and high bandwidth devices. This is clear to see when reading the first two articles which showcase how relatively small, yet dense networks can perform well on complex tasks. However, getting information from the real world into the networks is often a bottleneck as was discussed in the third article. The key paradigm shift is from “big data” to “local data”, by training small specialized networks to perform key tasks we can hopefully apply Neuromophic Computing to many different business needs.

The field as a whole has been hindered by their lack of high speed communication, obviously shown in the third article. The researchers showed that no matter how fast your compute layer is, if we can’t feed it information fast enough it will be starved, and run suboptimally. There is a huge opportunity for someone new to enter the Neuromorphic Computing space and solve this problem once and for all.

Recommendation Based on Analysis

After covering a broad range of topics from the field of Neuromorphic Computing I found two common opinions that we should use to guide our future investments and explorations into the technology. Those two are a focus on small dense networks, and an approach can lessen the burden of communication. The first is nearly a solved problem as it just acts as a constraint on networks themselves, with little other caution needed to adhere to it. However, the second requires a fundamental change in the way Neuromorphic Computing is approached. We should focus our research efforts on exploring all currently viable high-bandwidth communication solutions, and applying them directly to Neuromorphic Computing to see what works.

As a first step we should document all currently used communication strategies in Neuromorphic Computing, and then compare those to communication strategies used elsewhere in computing. Then we can begin to explore the best possible candidate options which we can apply to our own internal problems, allowing us to both explore Neuromorphic Networks, and the potential new communications simultaneously.