Graduate School Fair 2024

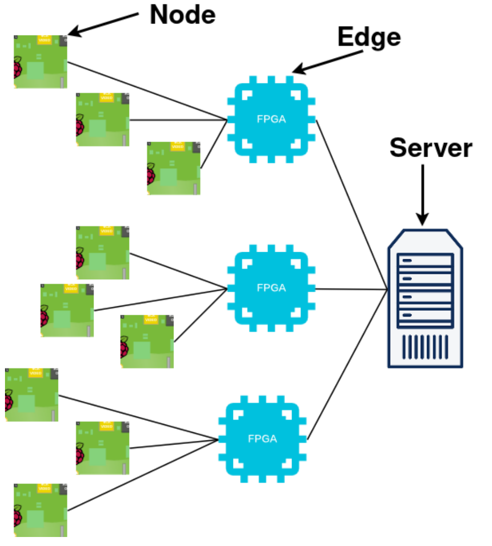
Distributed AI computing: ensemble learning and cross-device inference

Yuga HANYU, Yassine KHEDHER, Supervisor: Khanh N. DANG

Introduction

- **Spiking Neural Networks (SNNs)** utilize biologically-inspired spikes for energy-efficient communication and computation, making them ideal for on-chip learning.
- **Spike Timing Dependent Plasticity (STDP)** is a learning rule where synaptic strengths are adjusted based on the timing of spikes between neurons, mimicking biological learning processes.
- **Ensemble Learning** trains multiple models in parallel on lightweight devices and then merges them.
- **Challenges**:
  - Distributed datasets create heterogeneity among models.
  - Merging models with similar neuron characteristics leads to redundancy.
  - Current methods are not fully STDP-based, limiting efficient on-chip learning.
- **Objective**: This research addresses these challenges to enable efficient on-chip ensemble learning.

Model Architecture



Platforms

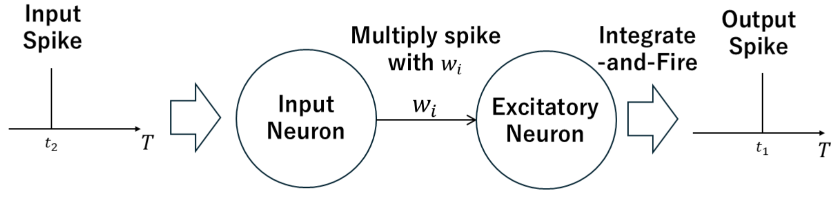
Nodes (IoT devices)
Devices that SNN models are trained with **on chip STDP learning**. Low computational ability but can be embedded in the local environments.

Edges (FPGA)
Gathering the learned weights from nodes and **models merging and compression**, is performed.

Server (PC with a GPU)
After collecting the pre-processed weights from the edges, the server conducts expensive computation such **as optimizing neuron mapping by Genetic Algorithm (GA)** and create the final model.

Spike Timing Dependent Plasticity Learning

Spike Timing Dependent Plasticity
Updates weights based the timings of output and input spikes.



Weight Update

If $\|t_1 - t_2\| > time\ window$: No weight update	Else if $t_1 > t_2$: Increase weight	Else if $t_2 > t_1$: Reduce weight

Model Compression Methods

To remove redundancy of the neurons and keep the final model compact, two compression methods are investigated in this research.

1. Similarity Compression
Remove one of two similar neurons on edges.

Visualized Weights

2. GA Compression
Improve neuron mapping in generations and create the final model on the server.

Run GA to optimize neuron mapping ⇒ **Build the final model based on the GA result**

Preliminary Comparison vs Benchmark

Single model[2] vs Merging 5 sub-models + GA Compression

Model	Single[2]	Merged & Compressed
Neurons	300	300 (5x100-200)
Classification Accuracy	88.87%	88.46%

References

[1] Z.-H. Zhou, “Ensemble methods: foundations and algorithms”, CRC press, 2012.
[2] P. U. Diehl and M. Cook, “Unsupervised learning of digit recognition using spike-timing-dependent plasticity,” Frontiers in computational neuroscience, vol. 9, p. 99, 2015.
[3] Hanyu Yuga and Khanh N. Dang, "EnsembleSTDP: Distributed in-situ Spike Timing Dependent Plasticity Learning in Spiking Neural Networks", 17th IEEE MCSoC 2024 (accepted for publication).