A Diagonal State Space Model on Loihi 2 for Streaming Sequence Processing

Svea Marie Meyer, Philipp Weidel, Philipp Plank, Leobardo Campos-Macias, Sumit Bam Shrestha, Philipp Stratmann, Jonathan Timcheck, Mathis Richter

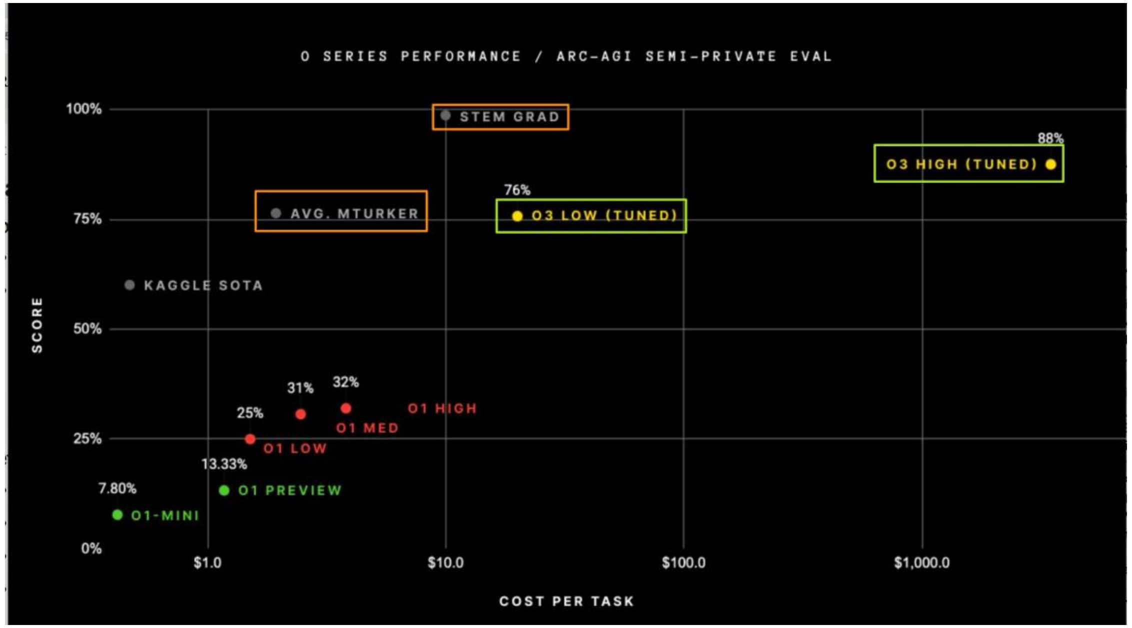
intel labs

NICE 2025 – March 27, 2025

Introduction

- Current AI solutions are extremely compute intense
- Transformers resource requirements grow quadratically with context length in their vanilla form

  => i.e. linear attention
- Alternative architectures and hardware is needed



https://en.softonic.com/articles/openai-breaks-barriers-with-the-o3-model-is-general-artificial-intelligence-near

State-space models as alternative to Transformers

- SSMs outperform Transformers in many tasks
- Compute scales linearly with context length
- S4 basis of large-scale language models such as Mamba

Model (Input length)	sMNIST (784)	psMNIST (784)	sCIFAR (1024)
Transformer (Vaswani et al., 2017; Trinh et al., 2018)	98.9	97.9	62.2
CCNN (Romero et al., 2022)	**99.72**	**98.84**	**93.08**
LipschitzRNN (Erichson et al., 2020)	99.4	96.3	64.2
LSSL (Gu et al., 2021b)	99.53	98.76	84.65
S4 (Gu et al., 2021a; 2022)	99.63	98.70	91.80
S4D-LegS (Gu et al., 2022)	-	-	89.92
Liquid-S4 (Hasani et al., 2022)	-	-	92.02
S5 (Smith et al., 2022)	99.65	98.67	90.10
Q-S5 (8 bit precision PTQ) (Abreu et al., 2024)	96.27	-	44.83
Q-S5 (8 bit precision QAFT) (Abreu et al., 2024)	99.54	-	86.95
AHP SNN on Loihi 1 (Rao et al., 2022)	96.00	-	-

Long-Range Arena

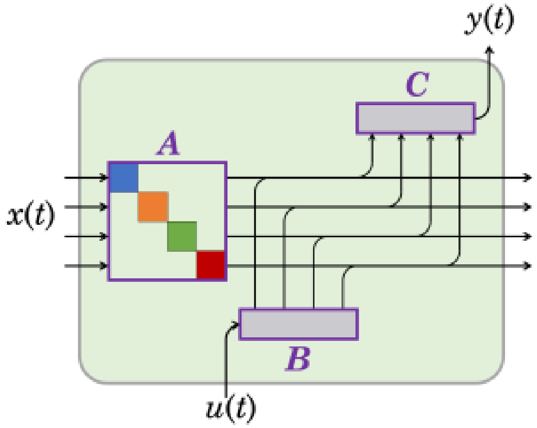
Model	ListOps	Text	Retrieval	Image	Pathfinder	Path-X	Avg
Transformer	36.37	64.27	57.46	42.44	71.40	✗	53.66
Reformer	37.27	56.10	53.40	38.07	68.50	✗	50.56
BigBird	36.05	64.02	59.29	40.83	74.87	✗	54.17
Linear Trans.	16.13	65.90	53.09	42.34	75.30	✗	50.46
Performer	18.01	65.40	53.82	42.77	77.05	✗	51.18
FNet	35.33	65.11	59.61	38.67	77.80	✗	54.42
Nyströmformer	37.15	65.52	79.56	41.58	70.94	✗	57.46
Luna-256	37.25	64.57	79.29	47.38	77.72	✗	59.37
S4	**59.60**	**86.82**	**90.90**	**88.65**	**94.20**	**96.35**	**86.09**

WikiText-103 language modeling

Model	Params	Test ppl.	Tokens / sec
Transformer	247M	**20.51**	0.8K (1×)
GLU CNN	229M	37.2	-
AWD-QRNN	151M	33.0	-
LSTM + Hebb.	-	29.2	-
TrellisNet	180M	29.19	-
Dynamic Conv.	255M	25.0	-
TaLK Conv.	240M	23.3	-
S4	249M	**20.95**	**48K** (**60**×)

Gu, Albert, Karan Goel, and Christopher Ré. "Efficiently modeling long sequences with structured state spaces." *arXiv preprint arXiv:2111.00396* (2021).

SSMs: Neuromorphic-friendly alternative to transformers



S4D Recurrent View

Gu, Albert, et al. "On the parameterization and initialization of diagonal state space models." Advances in Neural Information Processing Systems 35 (2022): 35971-35983.

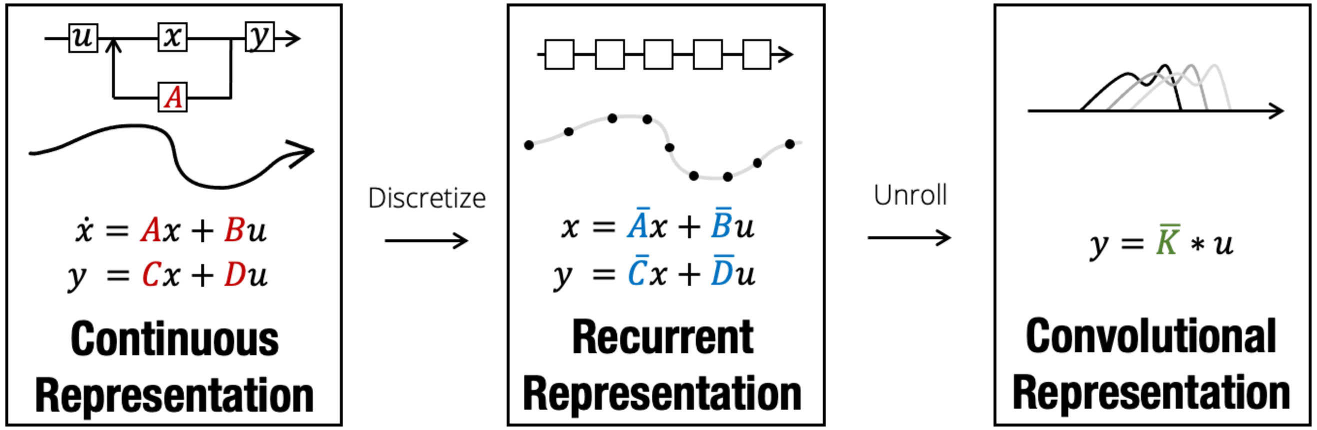
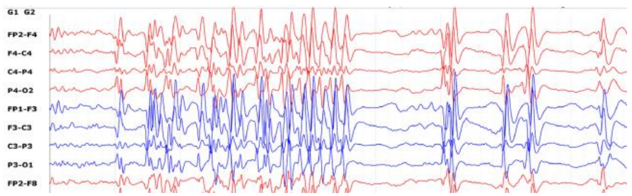


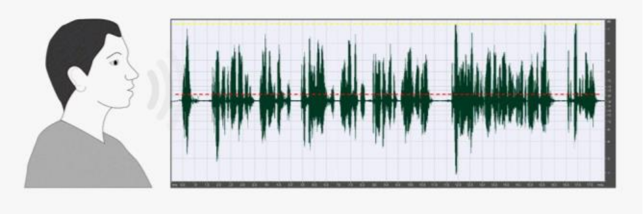
Image from blog post [Structured State Spaces: Combining Continuous-Time, Recurrent, and Convolutional Models] by Albert GU et al. (2022)

Application examples

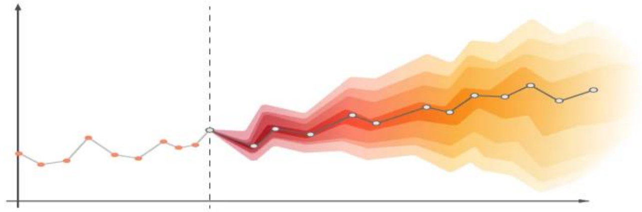
EEG



Speech



Time Series Forecasting

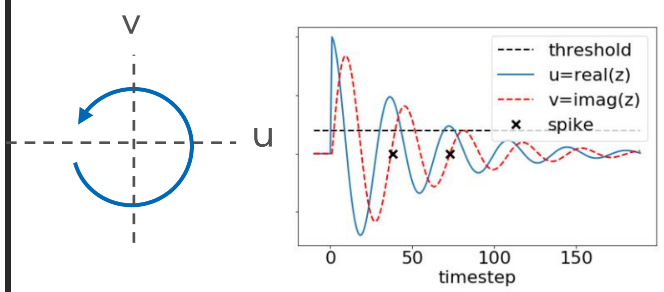


S4D neuron dynamics match Resonate-And-Fire neurons

Sanja Karilanova

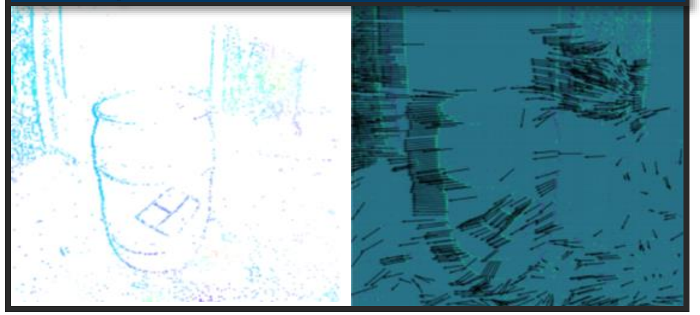
Resonate and Fire neurons compute optical flow for event-cameras with higher accuracy and 90x fewer ops than leading DNN solution

Resonate & Fire Neuron Behavior



$$z_k[t] = \lambda e^{i\omega\Delta t} z_k[t-1] + a_k[t]$$

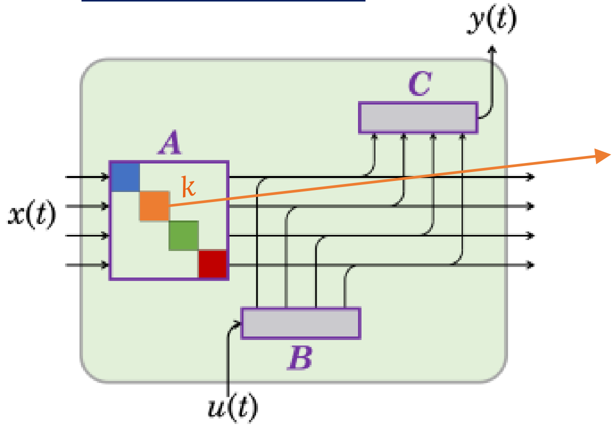
Optical Flow for Event Cameras



State Space Model

$$x_{t+1} = Ax_t + Bu_t$$

$$y_{t+1} = Cx_{t+1}$$



S4D Recurrent View

$$x_{t+1}^{(k)} = a_k e^{i\omega_k \Delta t} x_t^{(k)} + Bu_t$$

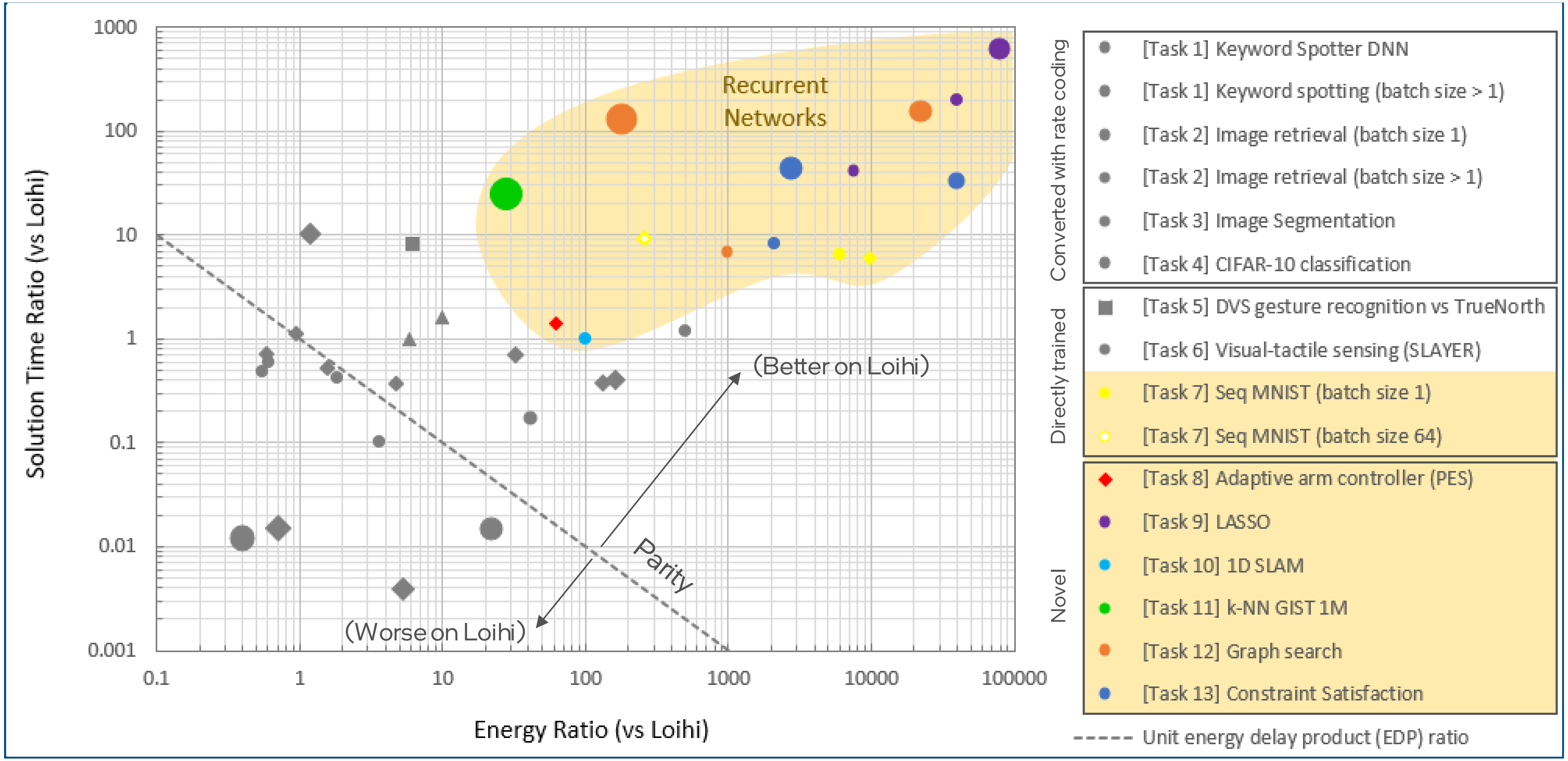
Resonator "neuron model"
Fully supported by Loihi 2

G. Orchard et al, "Efficient Neuromorphic Signal Processing with Loihi 2" IEEE International Workshop on Signal Processing Systems, Coimbra, Portugal, Oct 2021
S. Shrestha et al, "Efficient Video and Audio Processing with Loihi 2" ICASSP 2024

Recurrent networks give the best gains on Loihi

Reference architecture

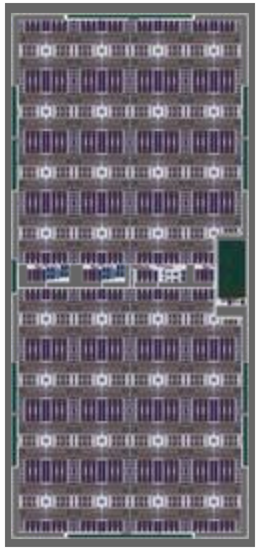
- CPU (Intel Core/Xeon)
- GPU (Nvidia)
- Movidius (NCS)
- TrueNorth

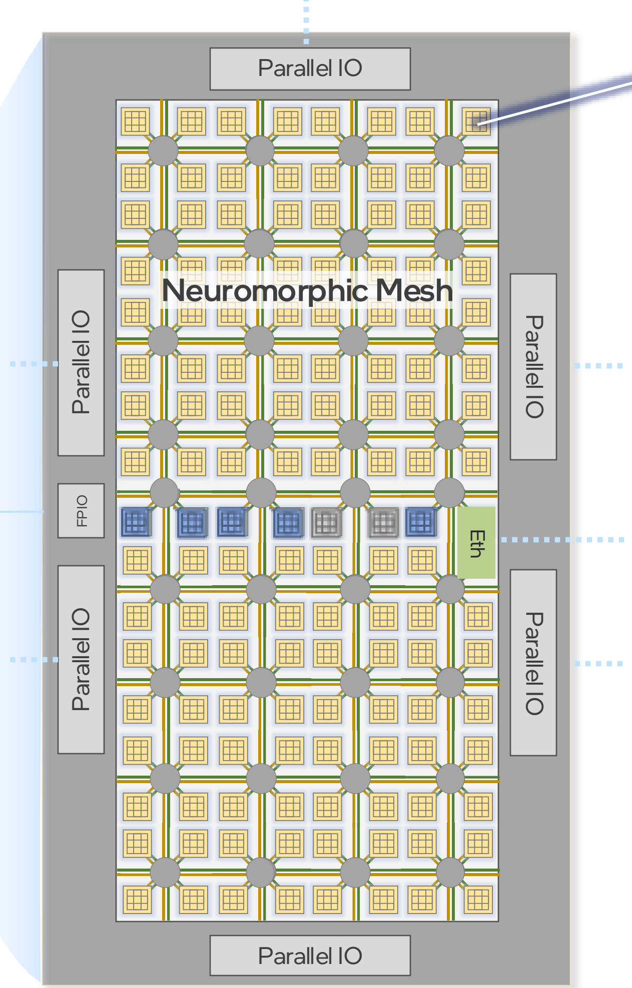


M. Davies et al, "Advancing Neuromorphic Computing With Loihi: A Survey of Results and Outlook," Proc. IEEE, 2021. Results may vary.

Loihi 2 architecture

Technology:	Intel 4
	31 mm^2
Neuro cores:	126
CPU cores:	5
Max # neurons:	1 M
Max # synapses:	123 M
Transistors:	2.3 B
Memory:	24 MB

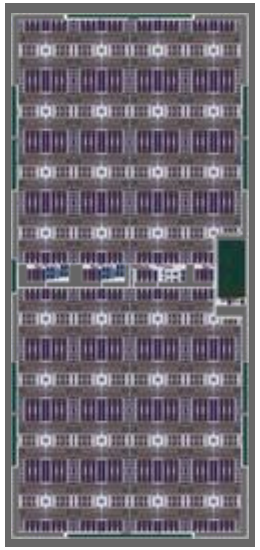


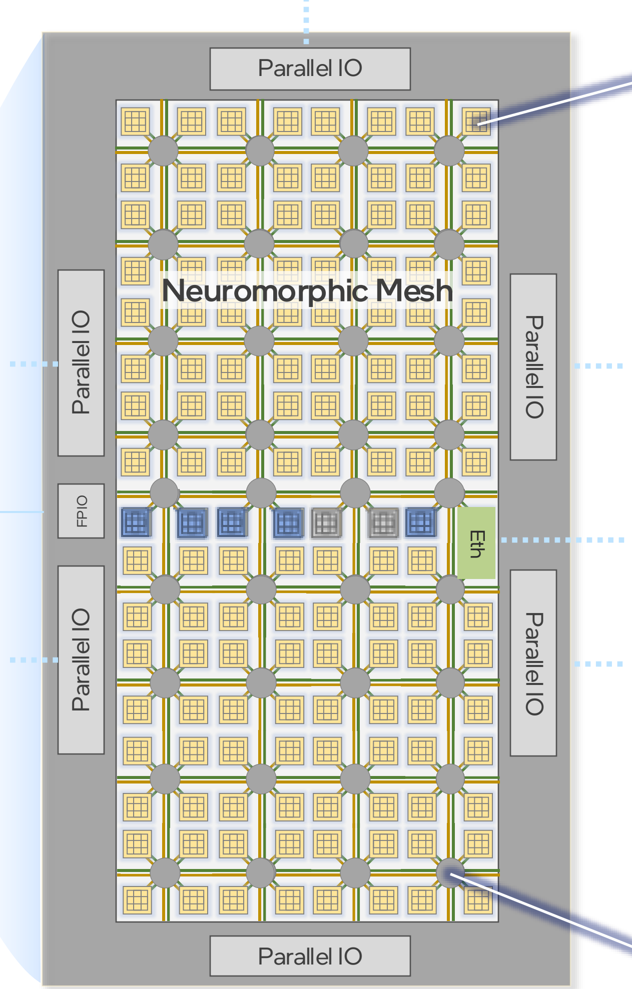


Neuromorphic cores (126)
Programmable neuron models
Programmable learning
Up to 8192 neurons per core
Communication with graded spikes

Loihi 2 architecture

Technology:	Intel 4
	31 mm^2
Neuro cores:	126
CPU cores:	5
Max # neurons:	1 M
Max # synapses:	123 M
Transistors:	2.3 B
Memory:	24 MB



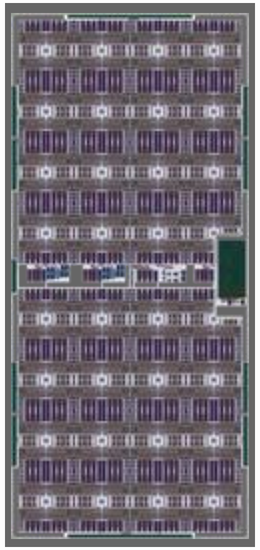


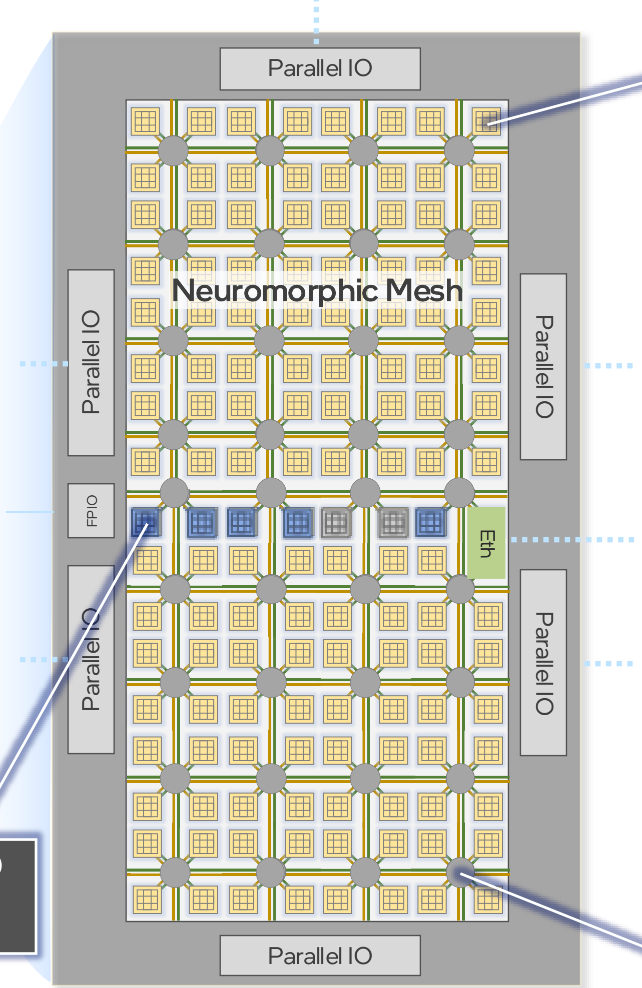
Neuromorphic cores (126)
Programmable neuron models
Programmable learning
Up to 8192 neurons per core
Communication with graded spikes

Optimized Async NoC
<200ns min timesteps possible[6] with accelerated barrier synchronization

Loihi 2 architecture

Technology:	Intel 4
	31 mm^2
Neuro cores:	126
CPU cores:	5
Max # neurons:	1 M
Max # synapses:	123 M
Transistors:	2.3 B
Memory:	24 MB





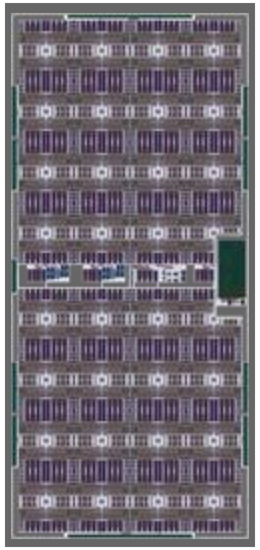
Microprocessor cores (5)
Synchronous x86 (2)
Asynchronous RISC-V (3)

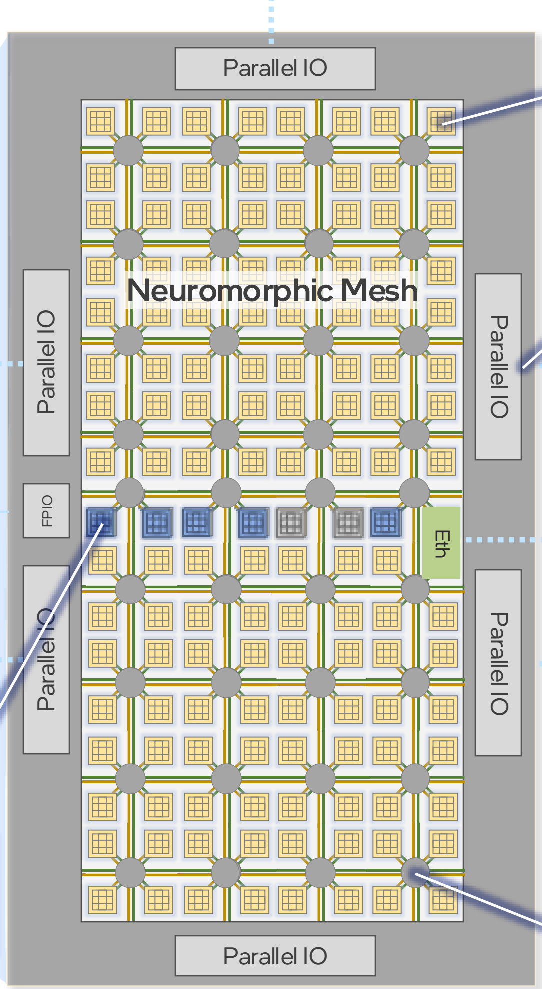
Neuromorphic cores (126)
Programmable neuron models
Programmable learning
Up to 8192 neurons per core
Communication with graded spikes

Optimized Async NoC
<200ns min timesteps possible[6] with accelerated barrier synchronization

Loihi 2 architecture

Technology:	Intel 4
	31 mm^2
Neuro cores:	126
CPU cores:	5
Max # neurons:	1 M
Max # synapses:	123 M
Transistors:	2.3 B
Memory:	24 MB



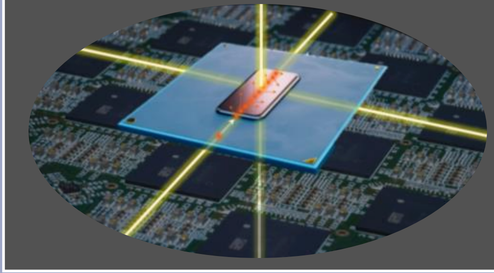


Microprocessor cores (5)
Synchronous x86 (2)
Asynchronous RISC-V (3)

Neuromorphic cores (126)
Programmable neuron models
Programmable learning
Up to 8192 neurons per core
Communication with graded spikes

Parallel off-chip interfaces (6)
Async wave pipelined at 10 Gb/s
with multicast compression

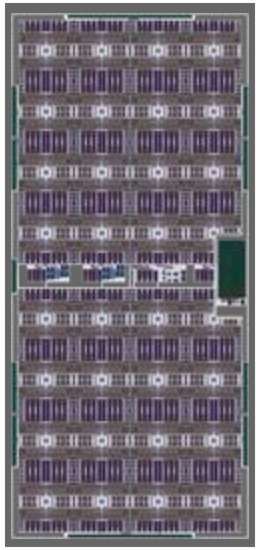
3D scaling support

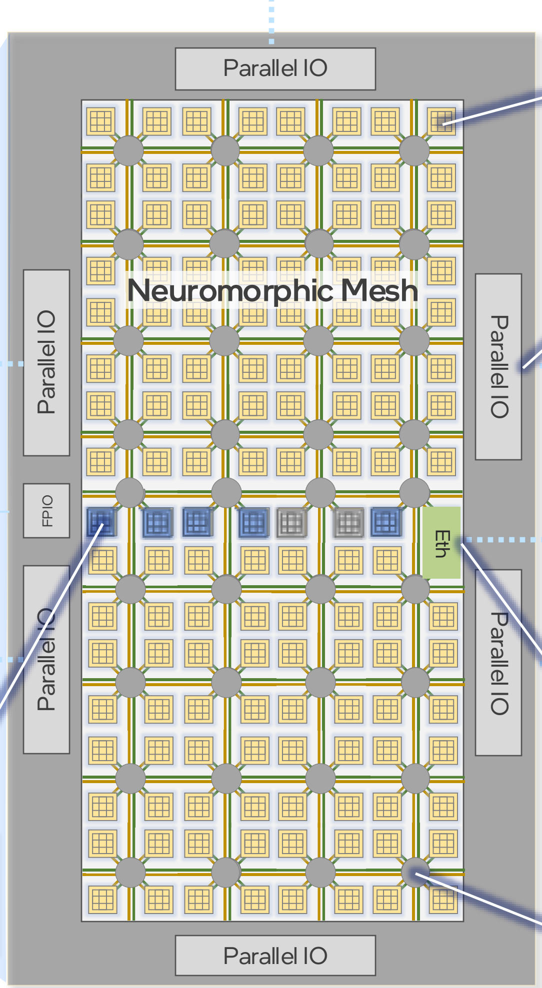


Optimized Async NoC
<200ns min timesteps possible[6] with
accelerated barrier synchronization

Loihi 2 architecture

Technology:	Intel 4
	31 mm^2
Neuro cores:	126
CPU cores:	5
Max # neurons:	1 M
Max # synapses:	123 M
Transistors:	2.3 B
Memory:	24 MB



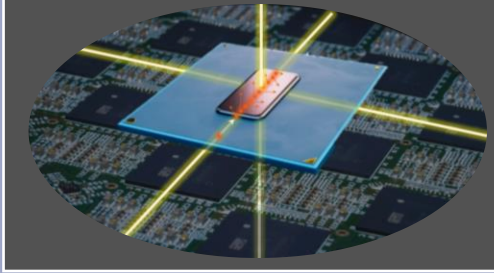


Microprocessor cores (5)
Synchronous x86 (2)
Asynchronous RISC-V (3)

Neuromorphic cores (126)
Programmable neuron models
Programmable learning
Up to 8192 neurons per core
Communication with graded spikes

Parallel off-chip interfaces (6)
Async wave pipelined at 10 Gb/s
with multicast compression

3D scaling support

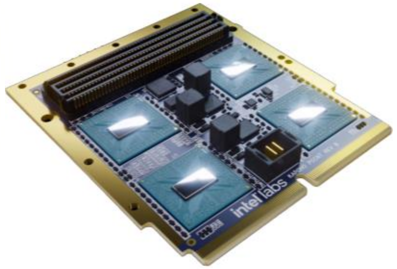


10G/25G Ethernet
Accelerated host message + spike I/O

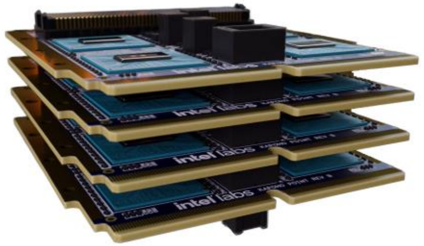
Optimized Async NoC
<200ns min timesteps possible[6] with
accelerated barrier synchronization

Loihi 2 Systems

Kapoho Point
1 chip



Kapoho Point
8 chips



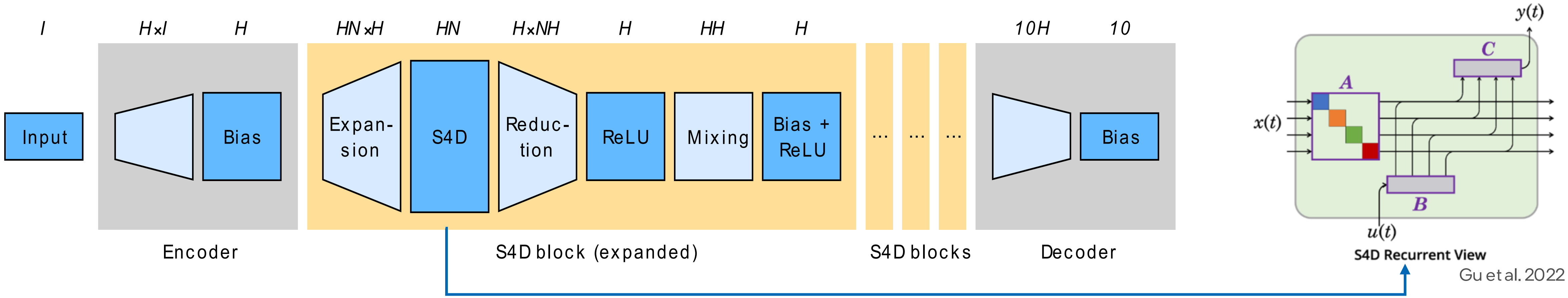
KP Stack
32 chips

Alia Point
128 chips
Datacenter

Hala Point
1152 chips
High Performance Computing

N-S4D: bringing S4D to Loihi 2

Model architecture



Neuromorphic adaptations to S4D

- Only ReLU activations for sparseification
- No normalization layers

Neuron dynamics

- In n-S4D, all matrices are diagonal -> no non-local information is needed
- Implement the complete neuron dynamics as programmable neuron
- Higher bit-precision (24bits instead of 8bits)
- Less on-chip communication

Two model sizes for different tasks

(p)sMNIST:

- Small model (H=64, N=32) using 31 cores (67k parameters)

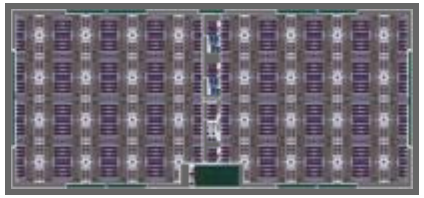
SCIFAR:

- Large model (H=128, N=64) using 111 cores (265k parameters)

Training and deployment pipeline

Sirine Arfa

Full-precision Training (convolutional representation) → **Post Training Quantization (recurrent representation)** → **Quantization Aware Fine Tuning (recurrent representation)** →



Full-precision Training (convolutional representation)

- Fast convolutional pre-training in full-precision
- Architecture search and baseline

Post Training Quantization (recurrent representation)

- Apply quantization (de)scaling directly without retraining
- Loss of precision
- Performance might drop

Quantization Aware Fine Tuning (recurrent representation)

- Apply fake-quantization
- Keep parameters in floating-point precision
- Mimic loss of precision
- Allows retraining

- Results of QAFT and on Loihi 2 match exactly

Results

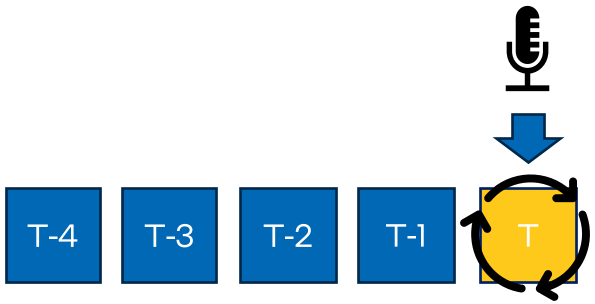
- n-S4D reaches similar performance as the original S4D model in full-precision and outperforms transformers
- PTQ can lead to a significant drop in accuracy
- The accuracy can mostly be recovered by QAFT
- SOTA is CCNN (2M parameters vs 256k for n-S4D)

Model (Input length)	**sMNIST** (784)	**psMNIST** (784)	**sCIFAR** (1024)
Transformer (Vaswani et al., 2017; Trinh et al., 2018)	98.9	97.9	62.2
CCNN (Romero et al., 2022)	**99.72**	**98.84**	**93.08**
LipschitzRNN (Erichson et al., 2020)	99.4	96.3	64.2
LSSL (Gu et al., 2021b)	99.53	98.76	84.65
S4 (Gu et al., 2021a; 2022)	99.63	98.70	91.80
S4D-LegS (Gu et al., 2022)	-	-	89.92
Liquid-S4 (Hasani et al., 2022)	-	-	92.02
S5 (Smith et al., 2022)	99.65	98.67	90.10
Q-S5 (8 bit precision PTQ) (Abreu et al., 2024)	96.27	-	44.83
Q-S5 (8 bit precision QAFT) (Abreu et al., 2024)	99.54	-	86.95
AHP SNN on Loihi 1 (Rao et al., 2022)	96.00	-	-
n-S4D, full precision **(Ours)**	99.51	97.53	86.53
n-S4D, after PTQ **(Ours)**	99.20	92.45	71.74
n-S4D, on Loihi 2 after QAFT (Ours)	99.20	96.16	84.13

Streaming vs batched processing

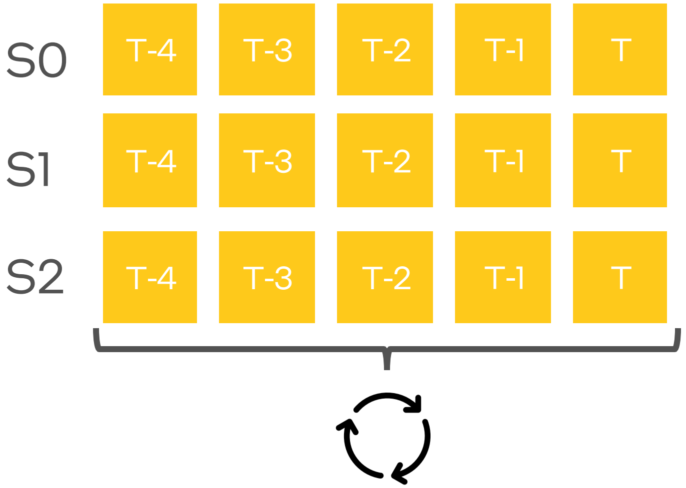
Streaming (token-by-token)

- Datapoints / tokens arrive in real time from a source with a (high) sampling-rate
- Difficulty: keep up with the sampling rate -> minimize latency



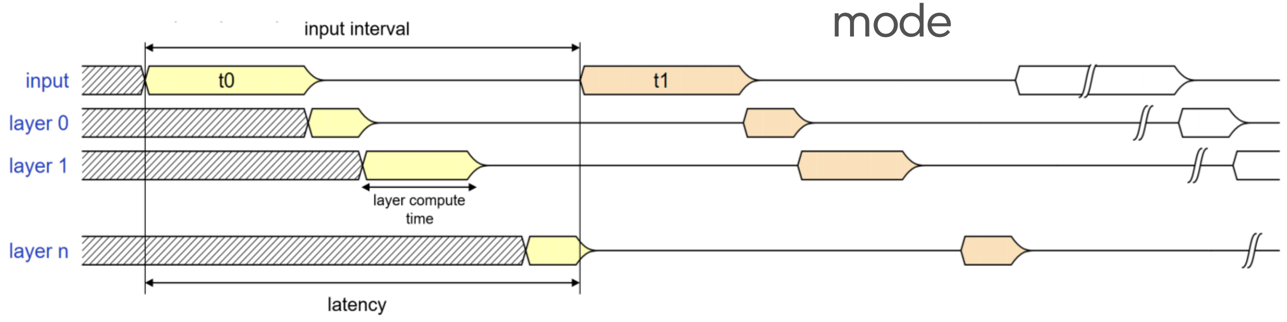
Batched (sample-by-sample)

- Data is readily available
- Batches of samples can be processed simultaneously
- Difficulty: maximize throughput



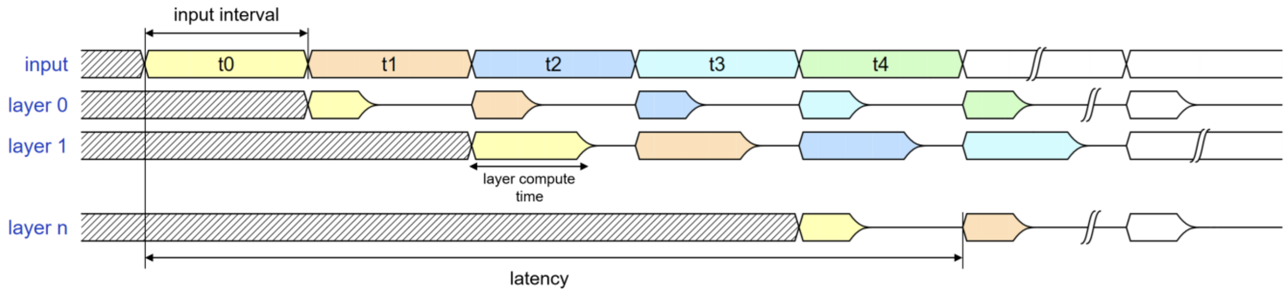
Optimizing latency and throughput on Loihi 2

Fall-through mode



- Optimizes latency
- Process one sample as fast as possible

Pipelined mode



- Optimizes throughput
- Insert data each timestep
- Speed determined by slowest core / layer

Streaming diagonal SSMs run extremely well on Loihi 2 compared to an edge GPU (Jetson Orin Nano)

	Parameters	sMNIST	psMNIST	sCIFAR
CCNN [leader]	2M	99.72	98.84	93.08
S4D on Loihi 2	256k	99.20	96.16	84.13

Loihi 2 massively outperforms versus GPU when processing streaming data with S4D (token-by-token inference)

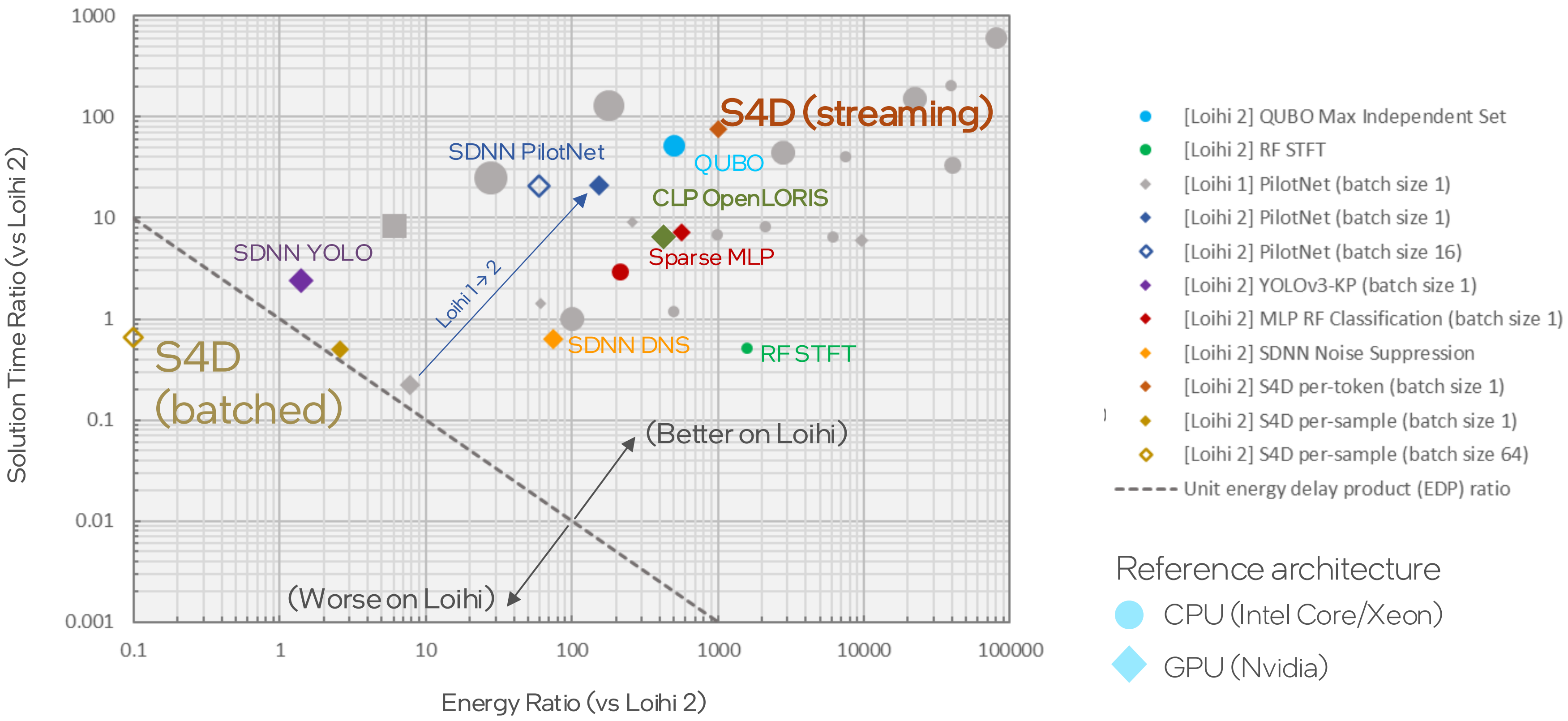
Token-by-token processing	Acc.	Energy (mJ)	Latency (ms)	Throughput
Loihi 2 (fall-through)	84.13	0.016	0.066	15,260
Orin Nano GPU (recurrent)	86.53	16.11	4.98	201
Loihi advantage	-2.4%	1,006x	75.5x	75.9x

Convolutional SSM evaluation on GPU outperforms when all tokens per sample can be processed as a batch

Sample-by-sample processing	Acc.	Energy (mJ)	Latency (ms)	Throughput
Loihi 2 (pipelined)	84.13	10.36	12.74	80
Orin Nano GPU (conv. batch 1)	86.53	26.89	6.33	158
Orin Nano GPU (conv. batch 64)	86.53	0.961	8.476	7,550
GPU advantage	2.4%	0.39x / 11x	2x	2x / 94x

Loihi 2 workloads were characterized on an Oheo Gulch system with N3C2-revision Loihi 2 chips running on NxCore 2.5.8 and alpha version of the NxKernel API with on-chip IO unthrottled sequencing of input tokens. † GPU workloads were characterized on an NVIDIA Jetson Orin Nano 8GB 15W TDP running Jetpack 5.1.2, TensorRT 8.6.1, Torch-TensorRT 1.3.0. Energy values include CPU GPU CV and SOC components as reported by jtop. ‡ Performance results are based on testing as of September 2024 and may not reflect all publicly available security updates. Results may vary.

Recurrent networks with streaming tasks work well on Loihi 2



M. Davies et al, "Advancing Neuromorphic Computing With Loihi: A Survey of Results and Outlook," Proc. IEEE, 2021. Results may vary.

Outlook

- S4D is the foundation for SOTA state-space models such as Mamba, Vision Mamba, etc
- Use SSMs in real-world applications on neuromorphic hardware
- Advance to more modern versions of SSMs such as S5:

Pierro, Alessandro, et al. "Accelerating Linear Recurrent Neural Networks for the Edge with Unstructured Sparsity." arXiv preprint arXiv:2502.01330 (2025).

- Go alternative routes such as MatMul-Free LLMs

Abreu, Steven, et al. "Neuromorphic Principles for Efficient Large Language Models on Intel Loihi 2." First Workshop on Scalable Optimization for Efficient and Adaptive Foundation Models.

Team

Svea Marie Meyer

Philipp Plank

Leobardo Campos-Macias

Sumit Bam Shrestha

Philipp Stratmann

Jonathan Timcheck

Mathis Richter

Legal Information

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Your costs and results may vary.

Results have been estimated or simulated.

Intel technologies may require enabled hardware, software or service activation.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Thank You!

Email inrc_interest@intel.com for more information