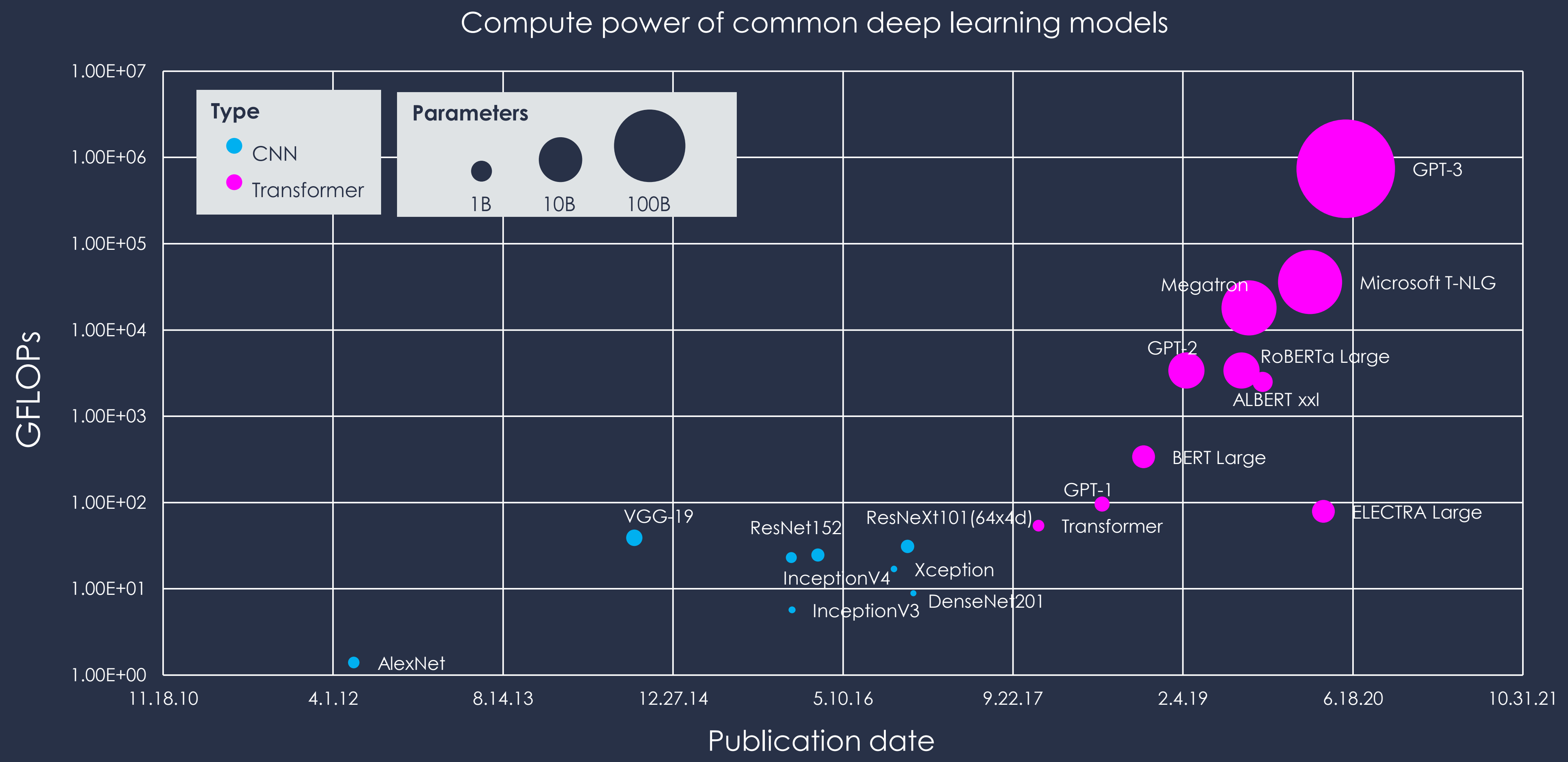
IEEE ICASSP 2022

Optimizing the Consumption of Spiking Neural Networks with Activity Regularization

Simon Narduzzi[1,2] Siavash A. Bigdeli[1] Shih-Chii Liu[2] L. Andrea Dunbar[1]

[1] *CSEM*, Neuchâtel, Switzerland [2] *Institute of Neuroinformatics*, University of Zurich and ETH Zurich, Zurich, Switzerland

The energy consumption problem



Adapted from
https://github.com/amirgholami/ai_and_memory_wall

csem

Edge computing

Advantages

- Rapid decision making
- Efficient pre-processing
- Privacy-preserving applications

1.8B

by 2026*

MAJOR CHALLENGE: Energy consumption

*ABI Research, Artificial intelligence and Machine Learning, 2 QTR 2021

Techniques to reduce consumption

Software

Pruning	Weights / neurons
Quantization	8bits, 4bits, …
Distillation	Teacher – student
Efficient operators	Separable convolutions, etc…
Event-based processing	Spiking neural networks (SNNs)

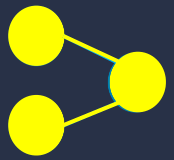
Hardware

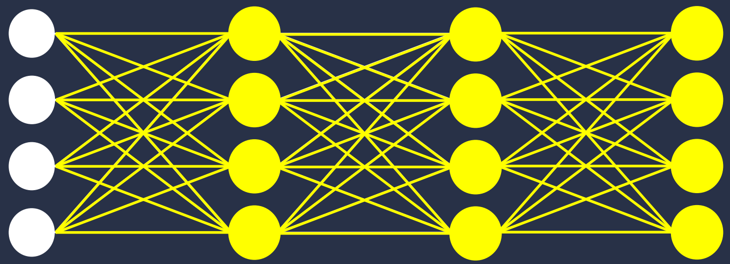
Semi-conductor process tech	FinFET, Fully Depleted Silicon-On-Insulator, etc…
Resource optimization	Power management, flexible accelerators, etc…
Specialized units	Convolution accelerators, zero-skipping, etc…
Event-based processing	Neuromorphic hardware: Intel Loihi, IBM TrueNorth, SpiNNaker, etc…

Artificial vs Spiking Neurons

Artifical Neural Network

$$z = \sigma\left(\sum_{j=1}^{N} W_{ij}x_j + b_i\right)$$

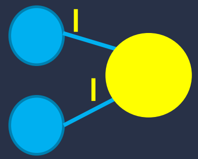


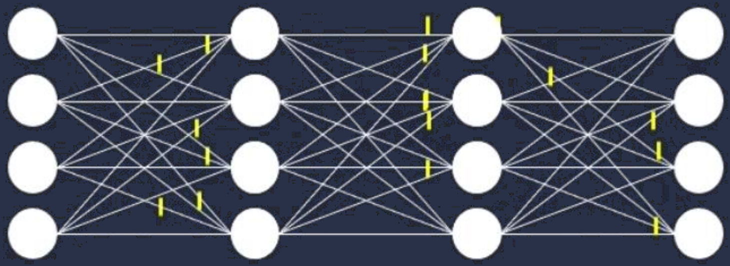


Information processing in
artificial neural networks (ANN)

Spiking Neural Network

$$z = \sigma_{thr}\left(\sum_{j=1}^{N} W_{ij}x_{t,j} + b_i\right)$$





Information processing in
spiking neural networks (SNN)

Metrics

Computation cost for ANN : ‘Effective’ FLOPS

$$EFLOPS = \sum_{l=1}^{L} \phi(W_l) \times \phi(A_{l-1}) + \phi(B_l)$$

$$\phi(x) \coloneqq x \neq 0$$

Computation cost for SNN: SynOps

$$SynOps = \sum_{t=1}^{T} \sum_{l=1}^{L} f_{out,l} \times s_l(t)$$

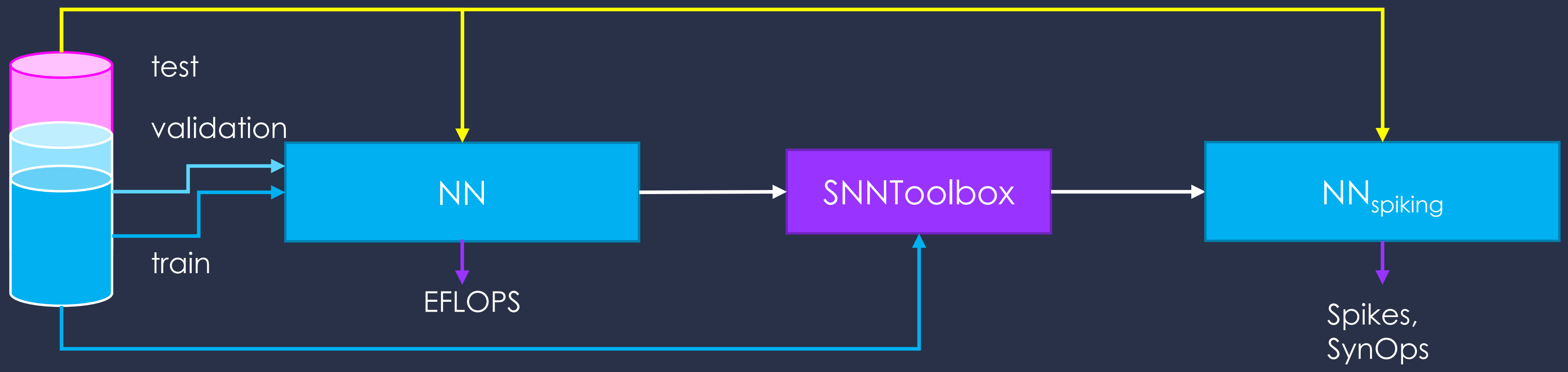
“A million spiking-neuron integrated circuit with a scalable communication network and interface”, *Merolla et al, 2014*

GOAL: Increase sparsity to reduce the computational cost

IDEA: Exploit the natural sparsity of SNNs

PROBLEM: SNNs training is difficult with common back-propagation

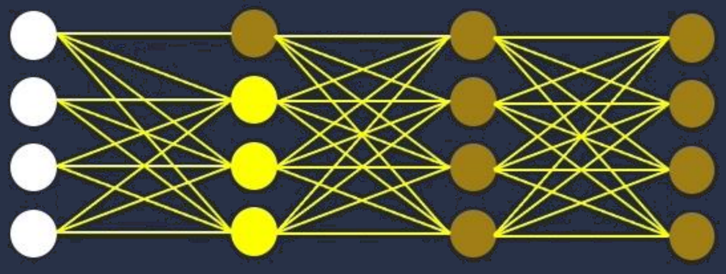
Experimental setup



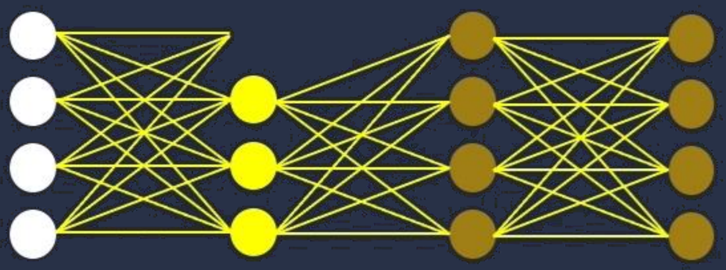
"Conversion of continuous-valued deep networks to efficient event-driven networks for image classification", *Rueckauer et al.*, 2017

Sparsity

- Sparsity reduces computational cost
- Pruning of weights or activation maps



Weight pruning



Neuron (activity) pruning

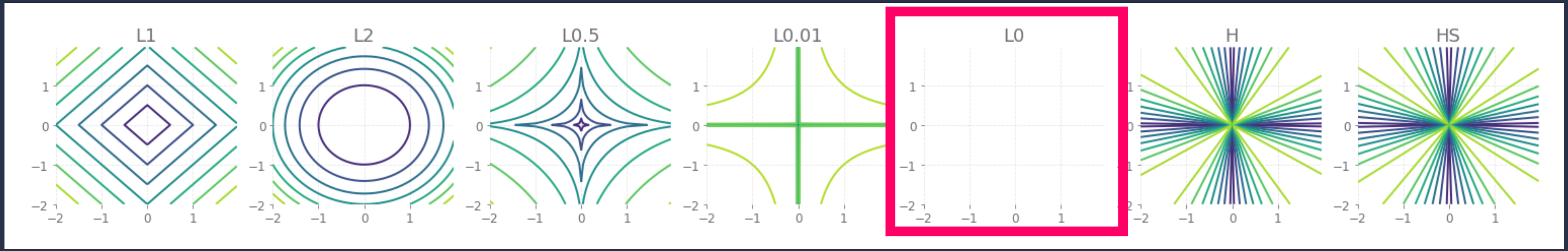
Related work

- *Zhao et al.*, 2021; *Pellegrini et al.*, 2021 – SNN trained from scratch
- *Sorbaro et al.*, 2020 – Optimize SynOps
- *Rückauer et al.*, 2017 – L_1 regularization on weights

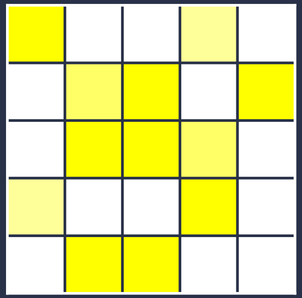
- **Ours**:
  - L_p-regularization and Hoyer,
  - Comparison between ANN and SNN,
  - EFLOPs

Constraint: Regularizers

- Enforce sparsity using regularizers on activity maps



Loss landscape for regularization methods used in our experiments.

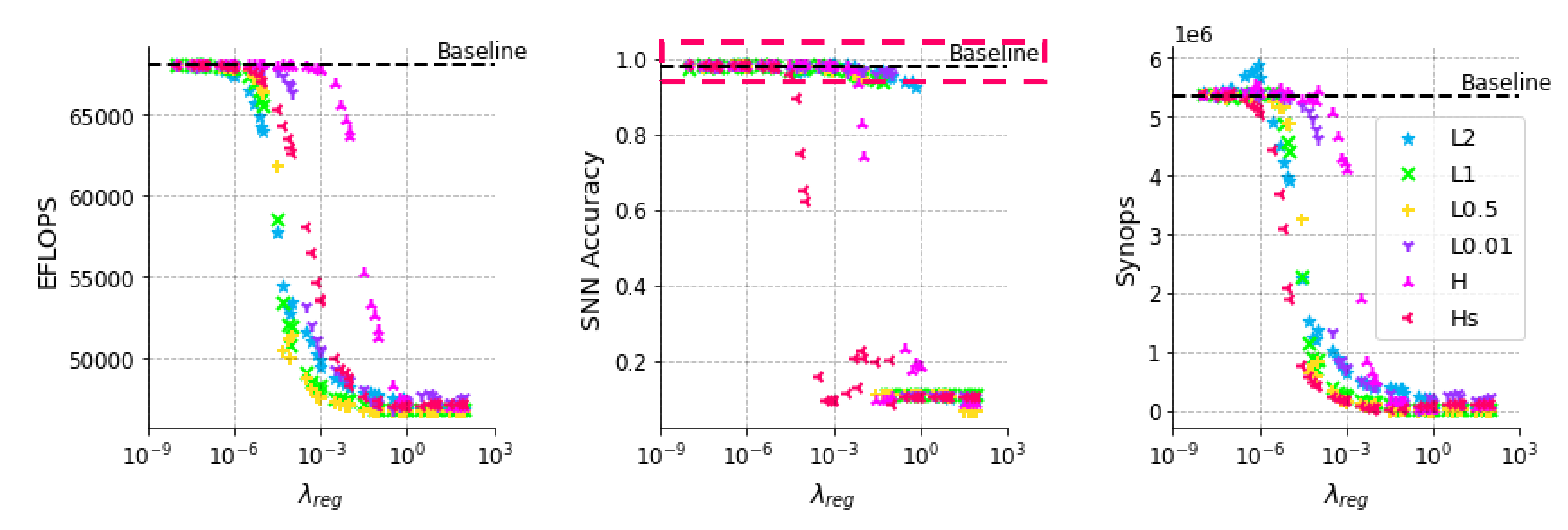


Activity map (X)

Loss function:

$$\mathcal{L} = CE + \lambda_{reg} \sum_{l} \psi(X_l)$$

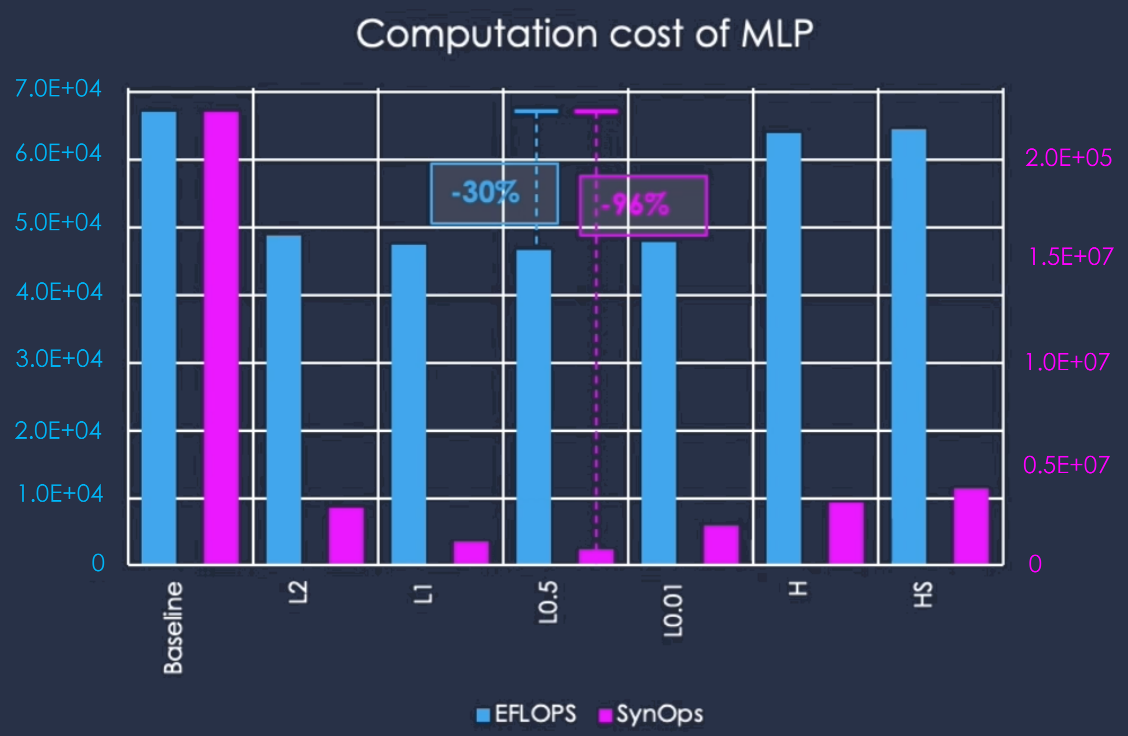
Results on MNIST

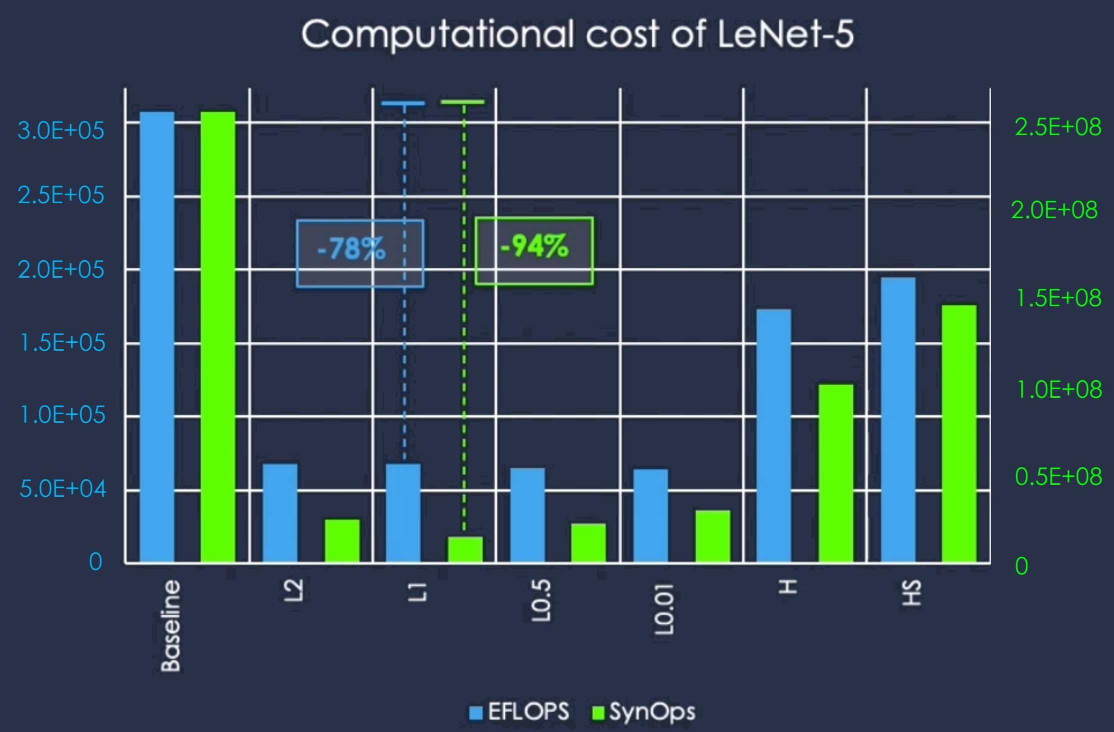


Results of the MLP with respect to λ_{reg}

Activity regularization effect

MNIST

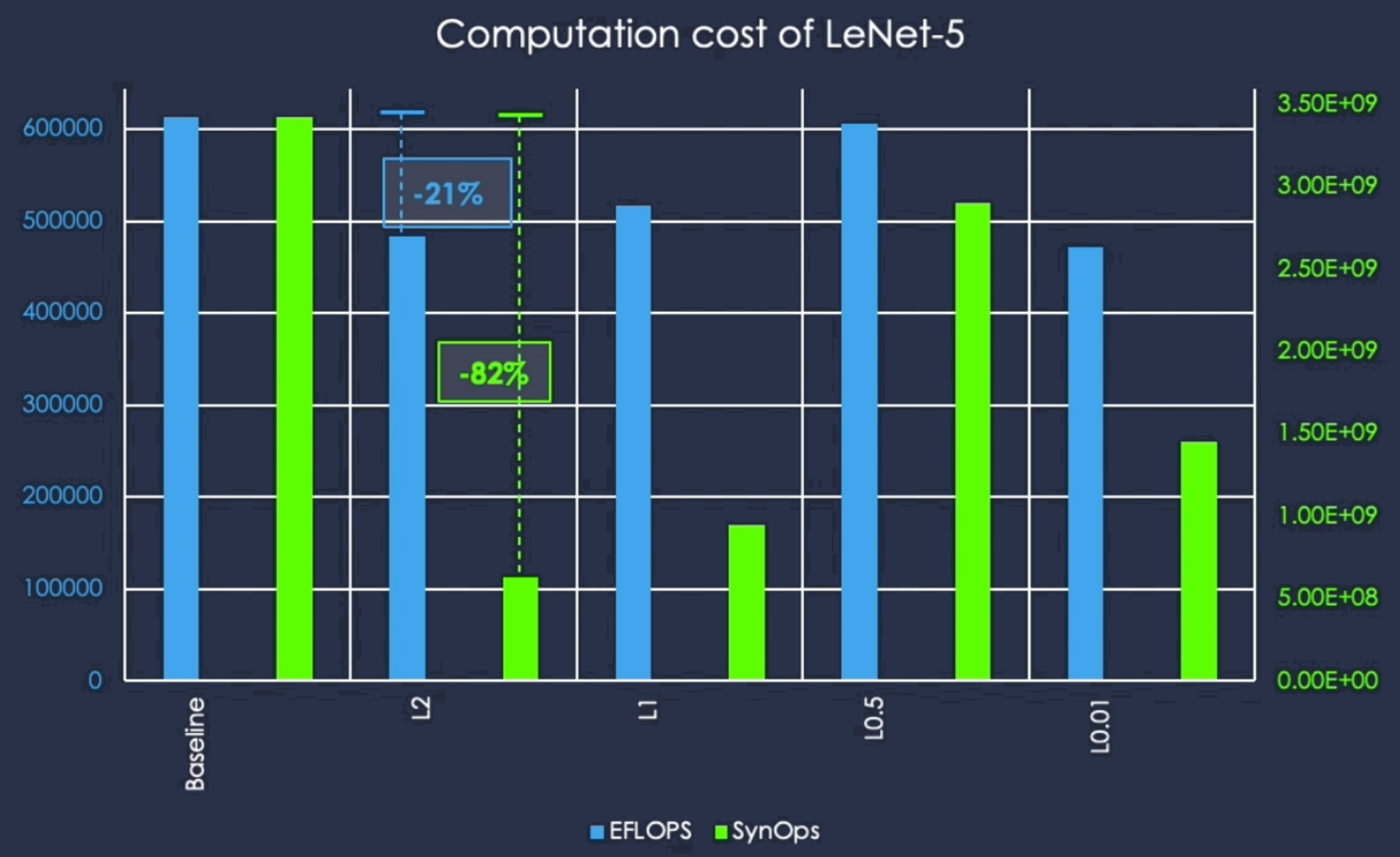




csem

Activity regularization effect

CIFAR-10



csem

Conclusion

- Activity regularization of ANNs is a simple way to reduce the number of SynOps in converted SNNs
- Hoyer regularization has limited effect compared to L_p–regularization
- SynOps and EFLOPs are not correlated, as a reduction in EFLOPs does not necessary result in a similar reduction in SynOps
- Better approximations of L_0 can be found, as $L_{0.01}$ is too aggressive

THANK YOU

Simon Narduzzi
simon.narduzzi@csem.ch