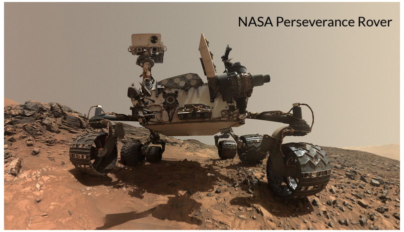
LA-UR-21-21175

The Backpropagation Algorithm Implemented on Loihi

Alpha Renner[1], Forrest Sheldon[2], Anatoly Zlotnik[2], Louis Tao[3], Andrew Sornborger[2]

[1] Institute of Neuroinformatics (INI), University of Zurich and ETH Zurich, Switzerland
[2] Los Alamos National Laboratory, Los Alamos, New Mexico, USA
[3] Center for Quantitative Biology, Peking University, Beijing, China

Motivation



Off-chip training works well now, but
efficient **learning, also after deployment** is especially relevant for exploring **new environments** by agents that need to be autonomous.

But also on earth, on-chip training could reduce ever rising training cost.

Relevant aspects:

- On-chip learning: using Loihi-learning engine ✔
- Online learning: stream of data instead of batches ✔
- Continual learning: learning new data and classes without retraining the whole dataset ✖
- Unsupervised learning: learning without labelled data ✖

Shallow learning will not be enough for state-of-the art deep nets (e.g. transformers).
→ **Goal of this project**: Proof of concept implementation of the (vanilla) backpropagation algorithm fully in the given neural substrate

Spiking backpropagation approaches

New approaches may help to enable a spiking implementation:

- Lee, J. H., Delbruck, T., & Pfeiffer, M., Frontiers in neuroscience 2016

- **Dendritic cortical microcircuits** – Sacramento, J., Costa, R. P., Bengio, Y., & Senn, W., NeurIPS 2018.

- **Eligibility Propagation** – Bellec, G., Scherr, F., Hajek, E., Salaj, D., Legenstein, R., & Maass, W. (TU Graz), on arxiv, Jan 25, 2019.

- **Surrogate Gradient Learning** – Neftci, E. O., Mostafa, H., & Zenke, F., on arxiv, Jan 28, 2019.

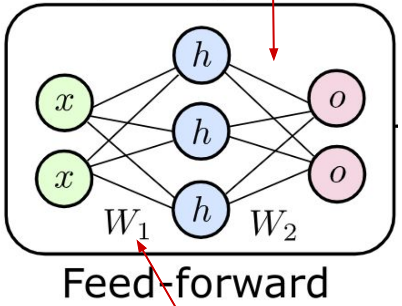
- **Superspike** - Zenke, F., & Ganguli, S.: Neural computation, 2018.

- **Feedback Alignment** - Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J., Nature comm, 2016.
  Arash, S., Lillicrap, T.P., & Tweed, D.B. Neural computation 2017.

Backpropagation network

Let's build a network for backprop...

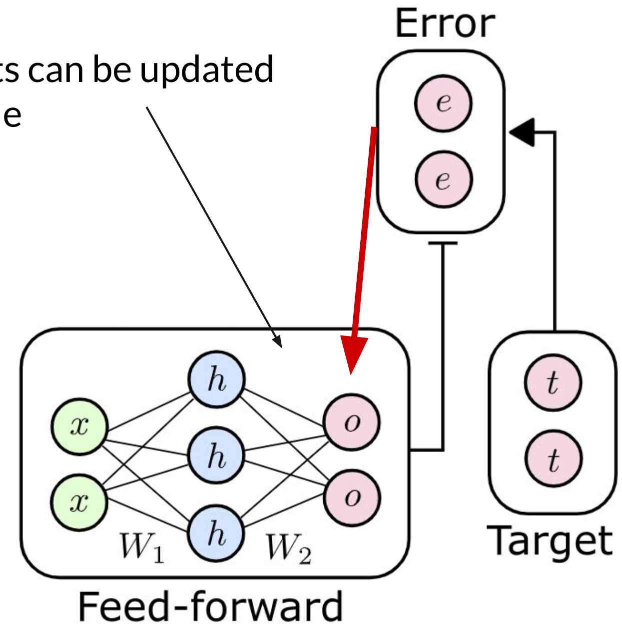
These weights can be updated with delta rule



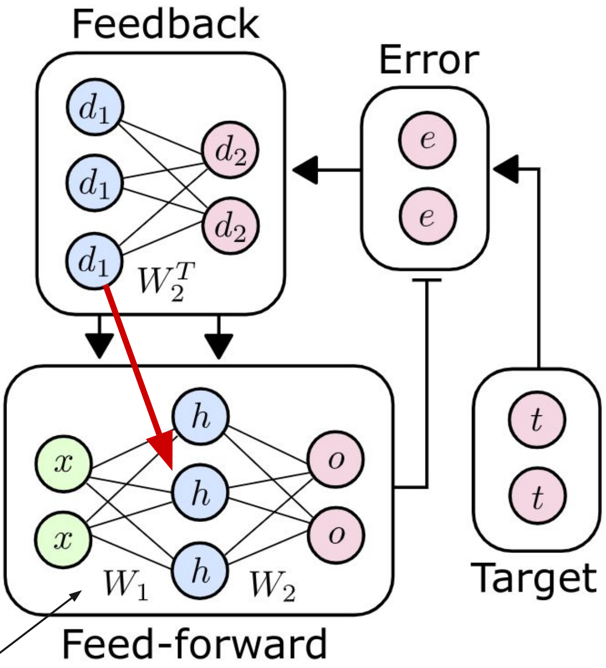
These weights need error backpropagation

Backpropagation network

These weights can be updated
with delta rule



Backpropagation network



These weights need error backpropagation

Backpropagation network

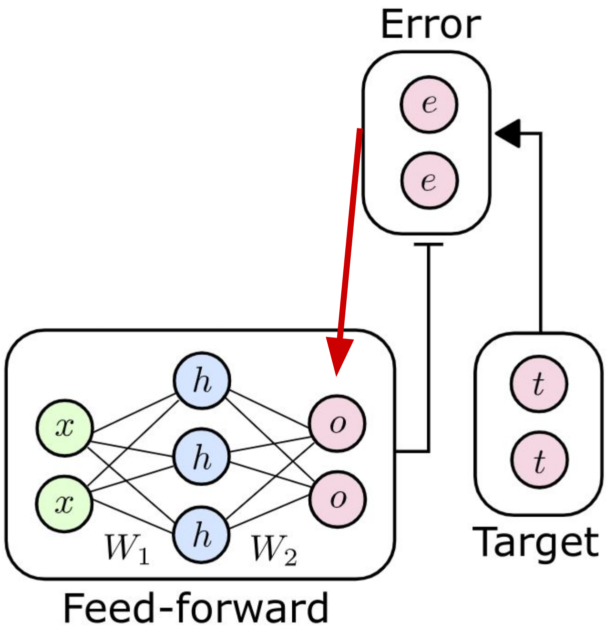
Update of a single weight (last layer):

$$\Delta w_0^{ij} = \delta_a^i \cdot x^j \cdot r'(z^i)$$

"error"

Layer input

Derivative of activation function



Backpropagation network

Update for a single neuron:

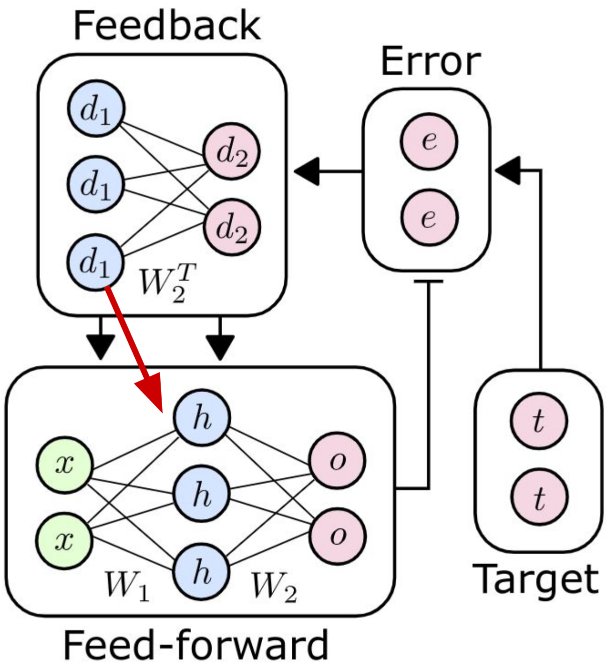
$$\Delta w_0^{ij} = \delta_a^i \cdot x^j \cdot r'(z^i)$$

"error" Layer input Derivative of activation function

Backpropagation to previous layers:

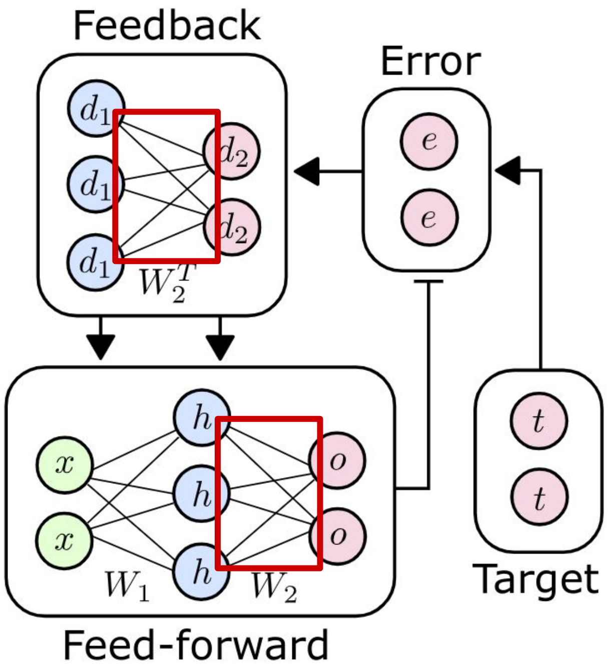
$$\delta_a^i = w_1^{(:,j)T} \cdot \delta_z$$

δ of next layer



Why has backpropagation not been implemented on chip before?

- **Weight transport problem:** For correct credit assignment, feedback weights must be the same as feedforward weights.
- **Activation storage problem:** Forward activations need to be kept in memory for the backward pass.



For a full list of issues see our paper:

Renner, A., Sheldon, F., Zlotnik, A., Tao, L., & Sornborger, A. (2021). The backpropagation algorithm implemented on spiking neuromorphic hardware. arXiv preprint arXiv:2106.07030.

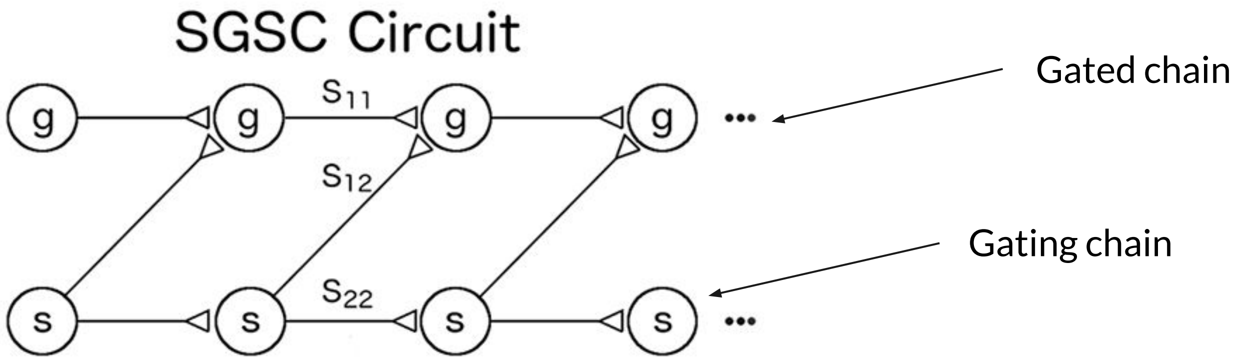
Recommended background:

Liao, Q., Leibo, J., & Poggio, T. (2016). How important is weight symmetry in backpropagation?. In Proceedings of the AAAI Conference on Artificial Intelligence.

Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*.

Routing by gating is vital for neuromorphic algorithms

Synfire-gated SFC route information/spikes through a network

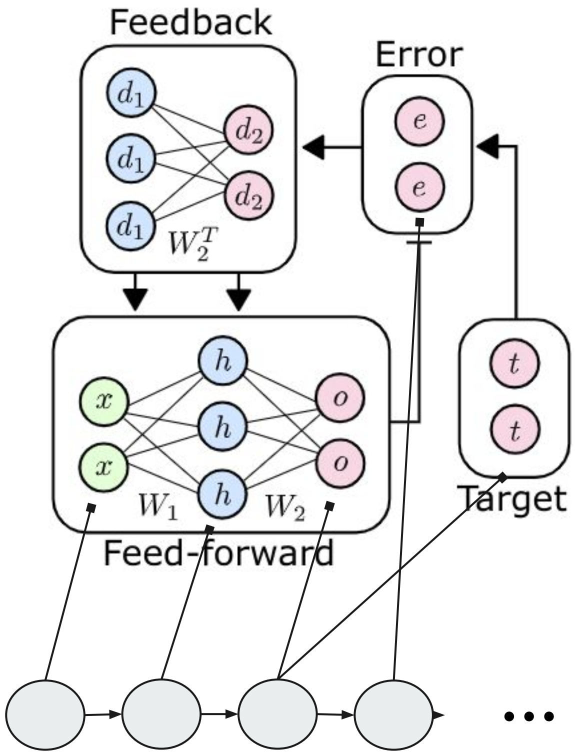


“Taking neurons/layers online based on a schedule”

Wang, Z., Sornborger, A. T., & Tao, L. (2016).
Graded, dynamically routable information processing with synfire-gated synfire chains.
PLoS computational biology

Classical SFC Literature:
Abeles (1982, 1991)
Hertz (1997)
Goedeke and Diesmann (2008)
Diesmann et al. (1999)

Routing helps to solve the main problems



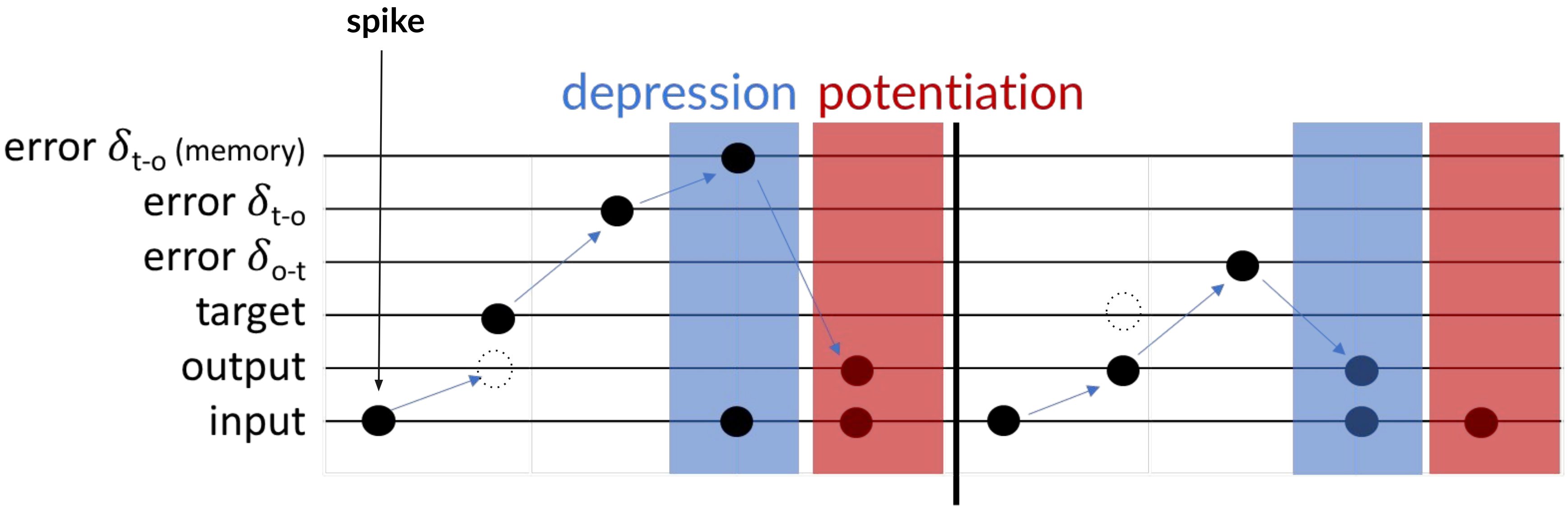
- **Weight transport problem**

  → We can maintain a copy of the weight matrices for backpropagation and route activity for learning back and forth between the forward and backward nets.

- **Activation storage problem**

  → We can maintain a memory layer and route activity back to the relevant layers when needed.

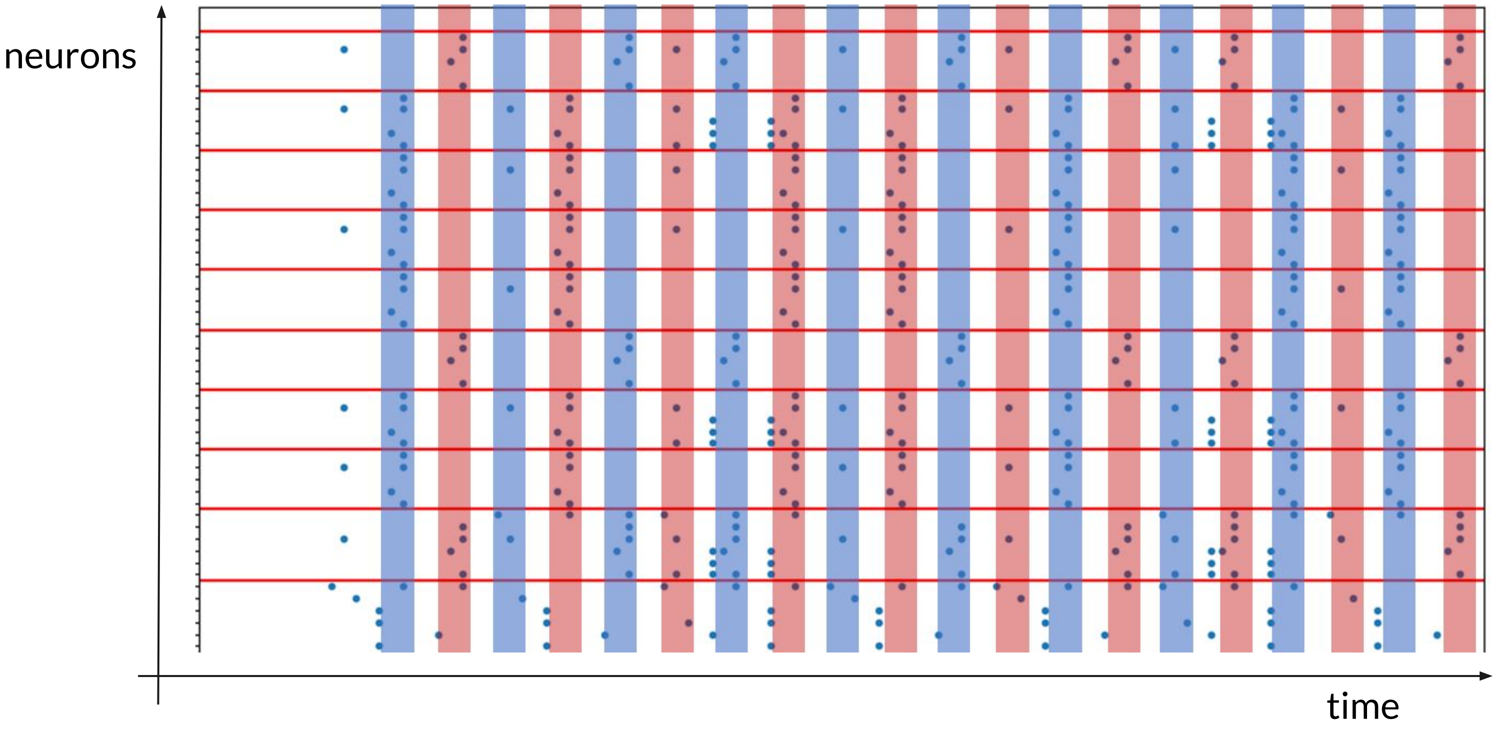
Intuition for the learning mechanism



Target > Output
→ Hebbian learning in potentiation phase

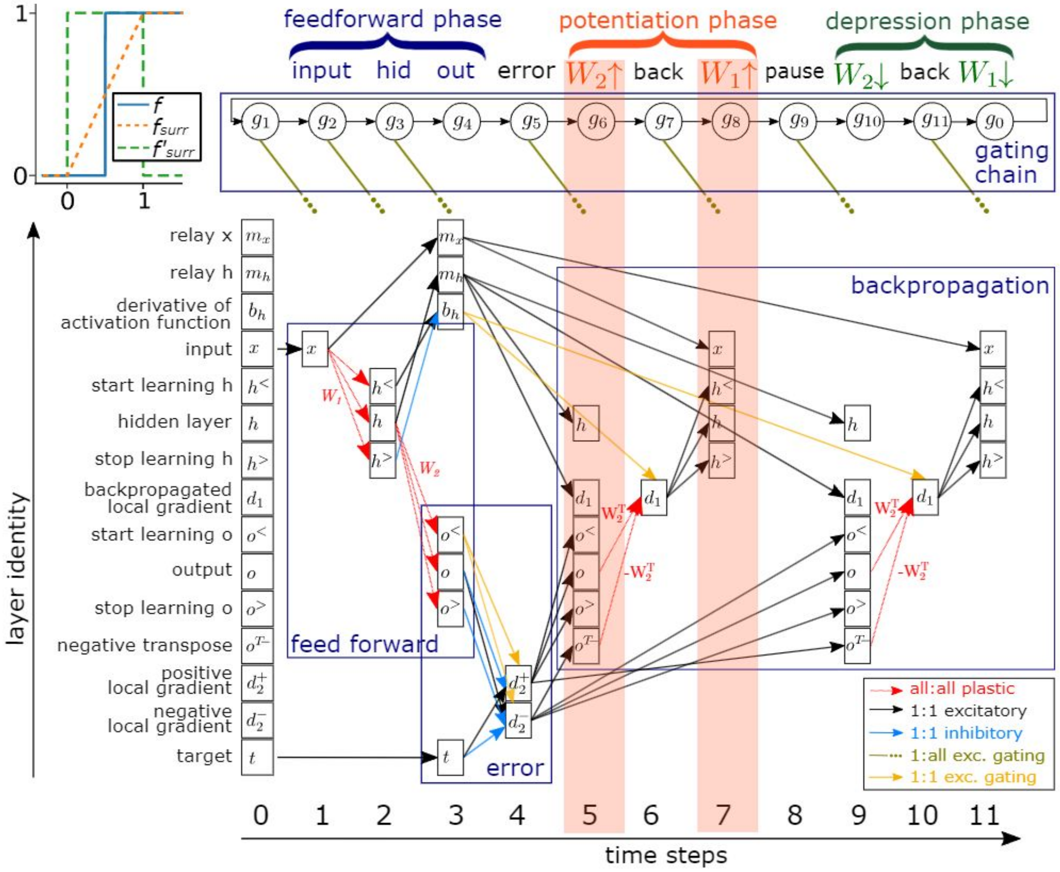
Target < Output
→ Hebbian learning in depression phase

Raster plot during training



→ Binary encoding leads to sparse activation during training and inference

The whole backpropagation network on Loihi



Literature comparison on MNIST

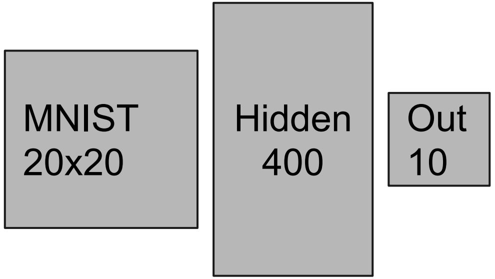
Publication	Hardware	Learning Mode	Network Structure	Energy per Sample (mJ)	Latency per Sample (ms)	Test Accuracy (%)
On-chip backpropagation						
This study	Loihi	on-chip sBP	400-400-10[a]	0.592	1.48	96.2
On-chip single layer training or BP alternatives						
[36] Shrestha et al. (2021)	Loihi	EMSTDP FA/DFA	CNN-CNN-100-10	8.4	20	94.7
[35] Frenkel et al. (2020)	SPOON	DRTP	CNN-10	0.000366[b]	0.12	95.3
[33] Park et al. (2019)	unnamed	mod. SD	784-200-200-10	0.000253[b]	0.01	98.1
[72] Chen et al. (2018)	unnamed	S-STDP	236-20[c]	0.017	0.16	89
[30] Frenkel et al. (2018)	ODIN	SDSP	256-10	0.000015	-	84.5
[73] Lin et al. (2018)	Loihi	S-STDP	1920-10[c]	0.553	-	96.4
[32] Buhler et al. (2017)	unnamed	LCA features	256-10	0.000050	0.001[b]	88
On-chip inference only						
This study	Loihi	inference	400-400-10[a]	0.00249	0.169	96.2
[36] Shrestha et al. (2021)	Loihi	inference	CNN-CNN-100-10	2.47	10	94.7
[35] Frenkel et al. (2020)	SPOON	inference	CNN-10	0.000313	0.12	97.5
[74] Göltz et al. (2019)	BrainScaleS-2	inference	256-246-10	0.0084	0.048	96.9
[73] Lin et al. (2018)	Loihi	inference	1920-10[c]	0.0128[d]	-	96.4
[72] Chen et al. (2018)	unnamed	inference	784-1024-512-10	0.0017	-	97.9
[76] Esser et al. (2015)	True North	inference	CNN (512 neurons)	0.00027	1	92.7
[76] Esser et al. (2015)	True North	inference	CNN (3840 neurons)	0.108	1	99.4
[77] Stromatias et al. (2015)	SpiNNaker	inference	784-500-500-10	3.3	11	95
Neuromorphic sBP in simulated SNN						
[78] Jin et al. (2018)	Simulation	BP	784-800-10	-	-	98.8
[79] Neftci et al. (2017)	Simulation	BP	784-500-10	-	-	97.7
[80] Shrestha et al. (2019)	Simulation	EM-STDP	784-500-10	-	-	97
[81] Tavanaei and Maida (2019)	Simulation	BP-STDP	784-500-150-10	-	-	97.2
[82] Mostafa (2017)	Simulation	BP	784-800-10	-	-	97.55
[83] Lee et al. (2016)	Simulation	BP	784-800-10	-	-	98.64
[84] O'Connor and Welling (2016)	Simulation	BP	784-300-300-10	-	-	96.4
[85] Diehl and Cook (2015)	Simulation	STDP	784-1600-10	-	-	95

[a] 400 (20x20) corresponds to 784 (28x28) after cropping of the empty image margin of 4 pixels
[b] Calculated from given values
[c] Off-chip preprocessing
[d] Dynamic energy reported in the Supplementary Material of [75]

99% on training set

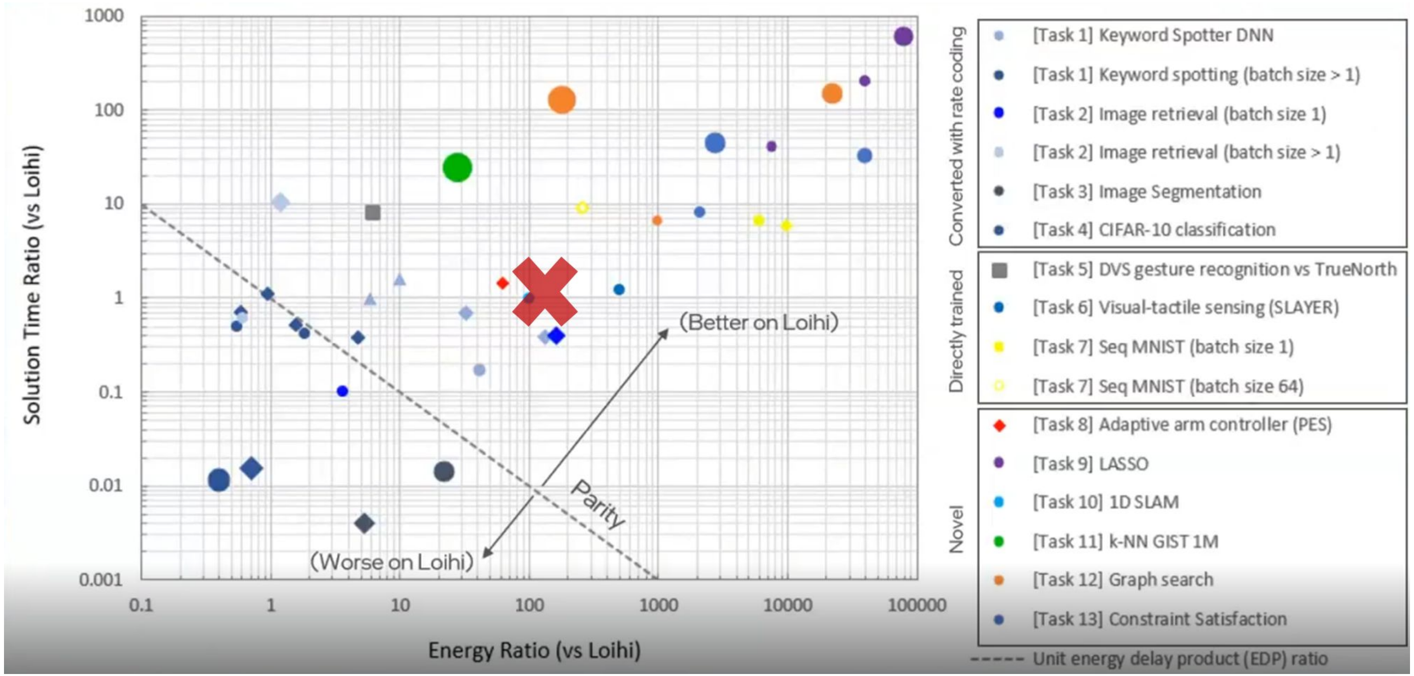
Accuracy is as good as it gets with 400 hidden neurons in an MLP.

Energy and latency are competitive, but not optimized.



Renner, A., Sheldon, F., Zlotnik, A., Tao, L., & Sornborger, A. (2021). The backpropagation algorithm implemented on spiking neuromorphic hardware. arXiv preprint arXiv:2106.07030.

Loihi vs. GPU



Note that interpretation is limited as the network is rather small

Conclusion

- Proof of principle of the **exact backpropagation** algorithm in a spiking network on Loihi.
- Framework of synfire-gated activity allows us to implement operations that are not otherwise suited for SNN (can be used beyond backprop).
- Binary activity encoding leads to high efficiency and sparsity on Loihi (and is likely **compatible with graded spikes** on Loihi 2 allowing for non-binary encoded deeper networks).

Thank you!

More questions?

alpren@ini.uzh.ch

Paper and code: https://arxiv.org/abs/2106.07030

Renner, A., Sheldon, F., Zlotnik, A., Tao, L., & Sornborger, A. (2021). The backpropagation algorithm implemented on spiking neuromorphic hardware. arXiv preprint arXiv:2106.07030.