ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9.
Available from http://www.i6doc.com/en/.

Energy-efficient detection of a spike sequence

Louis Le Cœur, Nicholas J. Riedman, Saarthak Sarup, Kwabena Boahen *

Stanford University · Departments of Electrical Engineering and Bioengineering
450 Jane Stanford Way, Stanford, CA 94305-2004, USA

Abstract. We present a novel 3D spike sorting network (3DSS) that detects a spike sequence efficiently and memorizes it upon a single presentation without configuration. We analyze the wiring and switches of alternatives and show that 3DSS reduces energy per spike quadratically compared to existing 2D networks. Applications include large-scale document retrieval and self-configuring hardware.

1 Detecting a spike sequence

The brain represents information as a sequence of spikes, and memory recall reinstantiates a unique spike order [1]. Thus, accurate detection of this spatiotemporal pattern is required to discriminate one item from another.

Previous work suggests that a stretch of dendrite can detect a specific spike sequence [2]. The *temporal* order of this sequence is memorized in the *spatial* order of axons along the dendrite stretch (Figure 1). This permutation can be realized by forming a dedicated path from each input pin to its output pin with a circuit-switched interconnection network. Developing one that consumes minimal energy would serve numerous applications, such as large-scale document retrieval.

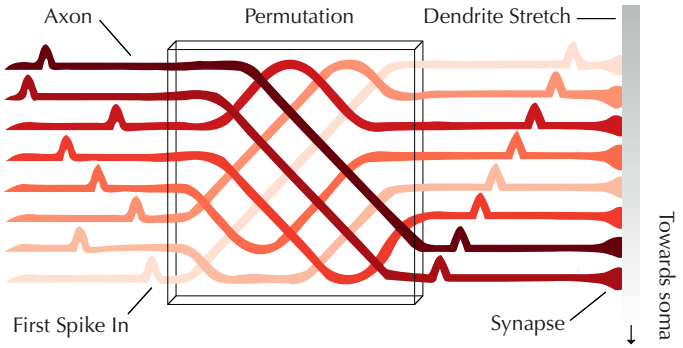


Fig. 1: **How to detect a spike sequence.** Axons are permuted in space such that the first spike in the sequence is routed to the top of the dendrite stretch, followed by the second spike, etc. These consecutive spikes propel a plateau potential along the stretch towards the soma.

*This work was supported by US National Science Foundation (EFRI/BRAID Program, Grant 2223827) and Stanford Institute for Human-Centered Artificial Intelligence (Hoffman-Yee Program).

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

The goal is for the permutation network to memorize a sequence in one shot, by configuring itself appropriately upon a single presentation of the spikes.

2 Minimizing wiring in permutation networks

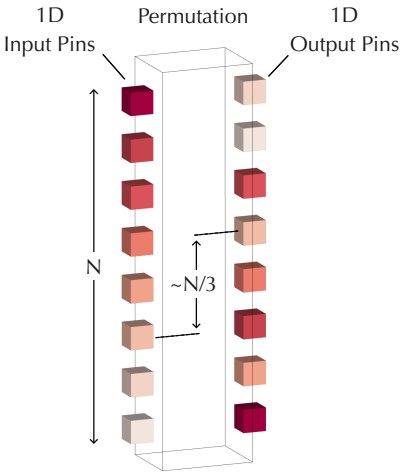
As the energy a spike uses is proportional to the distance it travels, reducing wiring minimizes energy consumption. Given N input and output pins and an arbitrary permutation, what is the minimum wiring needed to connect all inputs to their respective outputs?

Theorem 1. *With pins arranged in 1D, the minimum wiring is of order* N^2.

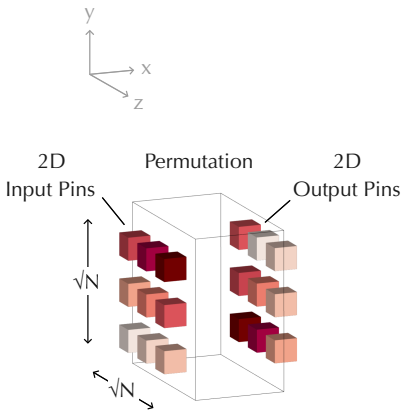
The average distance [3] along y between an input pin and its output pin is $1/3\ (N - 1/N) \sim N/3$, that is, of order N, even before considering displacement along x (Figure 2a). Since the routes between input and output pins are not shared, N distinct paths are required. That yields the N^2 wiring lower bound.

Theorem 2. *With pins arranged in 2D, the minimum wiring is of order* $N^{3/2}$.

By arraying pins in 2D, the average Manhattan distance along y and z between an input pin and its output pin is now $\sim 2/3\ \sqrt{N}$, that is, of order $\sqrt{N}$. This represents a quadratic improvement over the 1D case (Figure 2b). Since N distinct paths are still required, we get the $N^{3/2}$ wiring lower bound. By considering the cross-section that the N paths must traverse, it is apparent that this $3/2$ exponent cannot be achieved by a 2D layout but requires a 3D layout.



(a) **1D pins. Minimum wiring of order** N^2. The average distance along y between an input pin and its output pin is $\propto N$. There are N such paths.



(b) **2D pins. Minimum wiring of order** $N^{3/2}$. The average distance along y and z between an input pin and its output pin is $\propto \sqrt{N}$. N such paths.

Fig. 2: **Minimum wiring as a function of pin layout**

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

3 Memorizing a sequence

By adjusting the state of its switches, a circuit-switched network can implement different permutations. How can we set switches to detect a specific spike sequence? We can *configure* or *sort*, depending on the network topology:

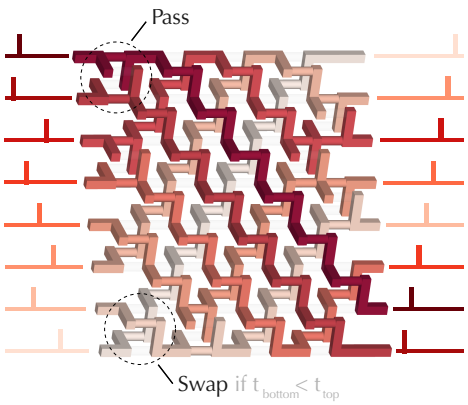
- *Configure:* topologies that minimize switches like the Beneš network [4] use as few as $N \log_2 N$ switches, but solve a global optimization problem and thus configure these switches only after the entire permutation is known.
- *Sort:* sorting networks require more switches, at least $\propto N \log_2^2 N$ [5], but each switch state is determined locally, allowing the incremental memorization of a permutation, as shown for an odd-even sorter (Figure 3a).

The tradeoff between switches and wiring makes our goal of minimizing wiring for energy efficiency incompatible with minimizing switches (Table 1). Consequently, the crossbar is the most energy-efficient 2D permutation network.

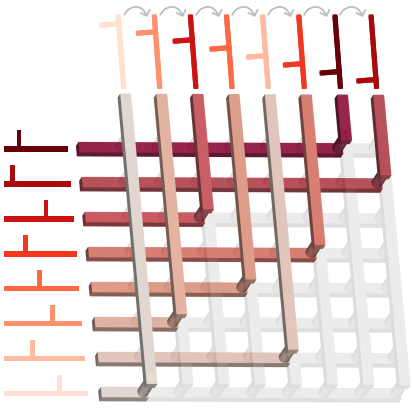
A crossbar can also memorize a spike sequence incrementally using a simple rule (Figure 3b), and its N^2 crosspoints can fan-out inputs to several outputs.

Table 1: 2D Network Costs

Network	Switches (2×2)	Wires	Incremental memorization
Beneš network	$N \log_2 N$	$4\ N^2$	No
Bitonic sorter	$1/2\ N \log_2^2 N$	$8\ N^2$	Yes
Odd-even sorter	$1/2\ N^2$	$4\ N^2$	Yes
Crossbar	$1/4\ N^2$	$3\ N^2$	Yes



(a) **Odd-even sorter.** *Rule:* each 2×2 switch swaps inputs if its 1st spike came from the bottom. Otherwise, it passes.



(b) **Crossbar.** *Rule:* the i^{th} spike sets the switch at its arrival row × column i, shifted with each new spike memorized.

Fig. 3: **Memorizing a sequence incrementally with 2D networks**

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

4 Constructing an efficient 3D spike sorting network

We can reduce wiring from N^2 to $N^{3/2}$ by arranging N input and output pins in $\sqrt{N} \times \sqrt{N}$ arrays, and connecting them via $\sqrt{N}$ stacked crossbars, each with $\sqrt{N}$ inputs and outputs. But a crossbar can only permute pins along one direction. Adding a second cube with crossbars arranged in the orthogonal direction is still not enough to handle all possible permutations.

Adding a third cube yields a 3D topology equivalent to a Clos network, which can route all possible permutations [6], but this structure needs switches to be *configured*: it cannot *sort* a spike sequence incrementally. Because sorting is fundamentally one-dimensional, sorting a 2D array using only row- and column-wise sorts is not trivial [7]. At least six cubes are required (Figure 4).

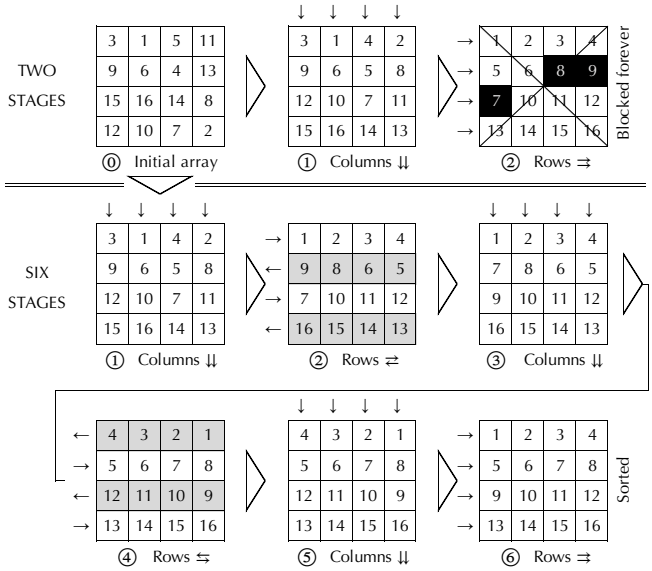


Fig. 4: **Multidimensional sorting.** Sorting a 2D array using only row- and column-wise sorts requires six stages and non-trivial reversals.

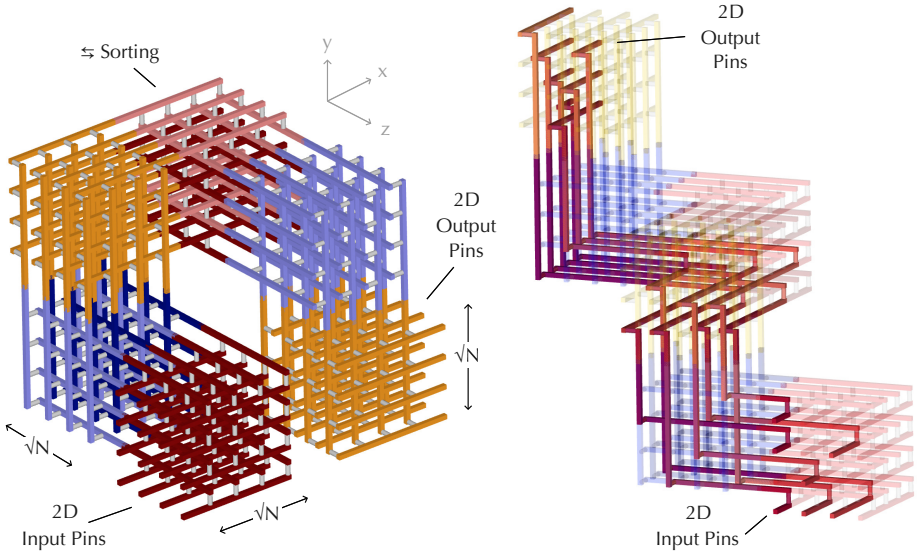
Our final structure is a novel 3D spike sorting (3DSS) network composed of six cubes of stacked crossbars (Figure 5). This network can memorize a spike sequence upon its first presentation, and detect its subsequent occurrences efficiently. Crossbars need a pitch of 2 to align pins properly between cubes, so the wiring per crossbar is $5(\sqrt{N})^2 = 5\ N$ and the total wiring 30 $N^{3/2}$ (Table 2).

It is possible to decrease the wiring prefactor by merging the six cubes into a denser layout, not shown here. This is the next focus of our research, in conjunction with further optimizations for nanofabrication and thermal dissipation.

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

Table 2: 3D Network Costs

Network	Switches (2×2)	Wires	Incremental memorization
Clos network	$^3/_4\ N^{3/2}$	$15\ N^{3/2}$	No
3DSS network	$^3/_2\ N^{3/2}$	$30\ N^{3/2}$	Yes



(a) **Folded 3DSS.** Composed of six cubes made of stacked crossbars, it can sort a spike sequence in one shot.

(b) **Real-time 3DSS simulation.** Memorization of an incoming spike sequence. Unfolded for visual clarity.

Fig. 5: **3D spike sorting network (3DSS)**

5 Conclusion

By minimizing wiring rather than switches of permutation networks, we developed a novel 3D structure than can memorize a spike sequence upon a single presentation, and then detect this same sequence energy-efficiently. Its $O(N^{3/2})$ wiring reduces energy per spike quadratically compared to existing 2D networks.

Beyond strict permutations, our structure can fan-out spikes to multiple destinations, or fan-in multiple spikes—using relevant operators—to the same destination. We plan to further explore this capability in future work.

The stacked layout of our structure is in line with current developments in 3D chip manufacturing, suggesting that it could be implemented using technologies like ReRAM, and associated with nanoscale electronic devices emulating a dendrite segment [2].

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

References

[1] A. P. Vaz, J. H. Wittig Jr, S. K. Inati, and K. A. Zaghloul, "Replay of cortical spiking sequences during human memory retrieval," *Science*, vol. 367, no. 6482, pp. 1131–1134, 2020.

[2] K. Boahen, "Dendrocentric learning for synthetic intelligence," *Nature*, vol. 612, no. 7938, pp. 43–50, 2022.

[3] D. E. Knuth, *The Art of Computer Programming, Volume 3: Sorting and Searching*, 2nd ed. Addison-Wesley, 1998, solution 28, p. 597.

[4] V. E. Beneš, "Optimal rearrangeable multistage connecting networks," *The Bell System Technical Journal*, vol. 43, no. 4, pp. 1641–1656, 1964.

[5] K. E. Batcher, "Sorting networks and their applications," in *Proceedings of the Spring Joint Computer Conference*. Association for Computing Machinery, 1968, pp. 307–314.

[6] C. Clos, "A study of non-blocking switching networks," *Bell System Technical Journal*, vol. 32, no. 2, pp. 406–424, 1953.

[7] A. Norollah, D. Derafshi, H. Beitollahi, and M. Fazeli, "RTHS: A low-cost high-performance real-time hardware sorter, using a multidimensional sorting algorithm," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 7, pp. 1601–1613, 2019.