Neural Networks 106 (2018) 144–151

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

# Memristive nanowires exhibit small-world connectivity

Ross D. Pantone [a,*], Jack D. Kendall [a], Juan C. Nino [b]

[a] Rain Neuromorphics, Inc., One Embarcadero Center, Suite #1650, San Francisco, CA 94111, United States
[b] Department of Materials Science and Engineering, University of Florida, 166 Rhines Hall, Gainesville, FL 32611, United States

ARTICLE INFO

*Article history:*
Received 14 March 2018
Received in revised form 19 June 2018
Accepted 5 July 2018
Available online 17 July 2018

*Keywords:*
Neuromorphics
Small-world
Network science
Nanowires
Memristors

ABSTRACT

Small-world networks provide an excellent balance of efficiency and robustness that is not available with other network topologies. These characteristics are exhibited in the Memristive Nanowire Neural Network ($MN^3$), a novel neuromorphic hardware architecture. This architecture is composed of an electrode array connected by stochastically deposited core–shell nanowires. We simulate the stochastic behavior of the nanowires by making various assumptions on their paths. First, we assume that the nanowires follow straight paths. Next, we assume that they follow arc paths with varying radii. Last, we assume that they follow paths generated by pink noise. For each of the three methods, we present a method to find whether a nanowire passes over an electrode, allowing us to represent the architecture as a bipartite graph. We find that the small-worldness coefficient increases logarithmically and is consistently greater than one, which is indicative of a small-world network.



## 1. Introduction

Neuromorphic hardware architectures, much like biological neural networks, are subject to constraints that are not present in traditional software-based simulations of artificial neural networks (ANNs) (Yu, Zhang, Chen, & Xie, 2018). For example, the weights of a neural network must be physically linked to the two neurons the weights connect. This introduces the wiring cost of a network, which is a measure of how much wiring is needed to connect all the neurons in the network. In both biological neural networks and neuromorphic hardware architectures, this wiring between the neurons consumes the vast majority of the available space (Raj & Chen, 2011). Therefore, minimizing this wiring cost is extremely important.

In biological neural networks, another important parameter is the global efficiency of the network, which is the inverse of the mean shortest path length between two random neurons (Achard & Bullmore, 2007). This value determines how efficiently the network can process and transmit information. For fully connected networks, the value of this parameter is maximal, while for locally connected networks, it is low. Thus, for example, biological brains are the result of a delicate balance between the competing objectives of creating a well-connected network with a high global efficiency, while simultaneously minimizing the amount of wiring needed to connect the network (Achard & Bullmore, 2007).

Similarly, any scalable neuromorphic architecture must balance these two cost functions. In biological neural networks, as well as other complex networks, the optimization of balancing these two objectives (wiring cost and global efficiency) results in a *small-world network* topology. Small-world networks have mostly local connectivity, with a nontrivial number of random, long-range connections added to the network. These small-world networks have been shown to achieve an efficient balance between wiring cost and global efficiency, allowing small-world networks to scale far more efficiently than fully or locally connected networks (Kleinberg, 2000).

In addition to the small-world property, biological neural networks display many other characteristics of complex networks, such as stochasticity, a diverse degree distribution (scale-free structure), and modularity (Holden, 1983; Martinello, Hidalgo, Maritan, & di Santo, 2017; Rodriguez, Izquierdo, & Ahn, 2017).

Despite the advantages of these complex network topologies, to date, most neuromorphic hardware architectures continue to use fully or locally connected crossbar arrays (Schuman, Potok, & Patton, 2017). This is likely due to the simplicity with which these arrays can be fabricated and integrated into conventional hardware. However, there are merits to exploring alternative methods of fabricating more complex network topologies, given the benefits of small-world connectivity, specifically, with respect to the scalability of the networks.

Here, we show that a novel method for fabricating complex networks based on memristive nanowires developed by Kendall & Nino (2015) has an extremely high density of trainable parameters (~400 million tunable synapses per square centimeter) and more importantly, it exhibits small-world characteristics.

* Corresponding author.
*E-mail addresses:* ross@rain-neuromorphics.com (R.D. Pantone), jack@rain-neuromorphics.com (J.D. Kendall), jnino@mse.ufl.edu (J.C. Nino).

https://doi.org/10.1016/j.neunet.2018.07.002
0893-6080/

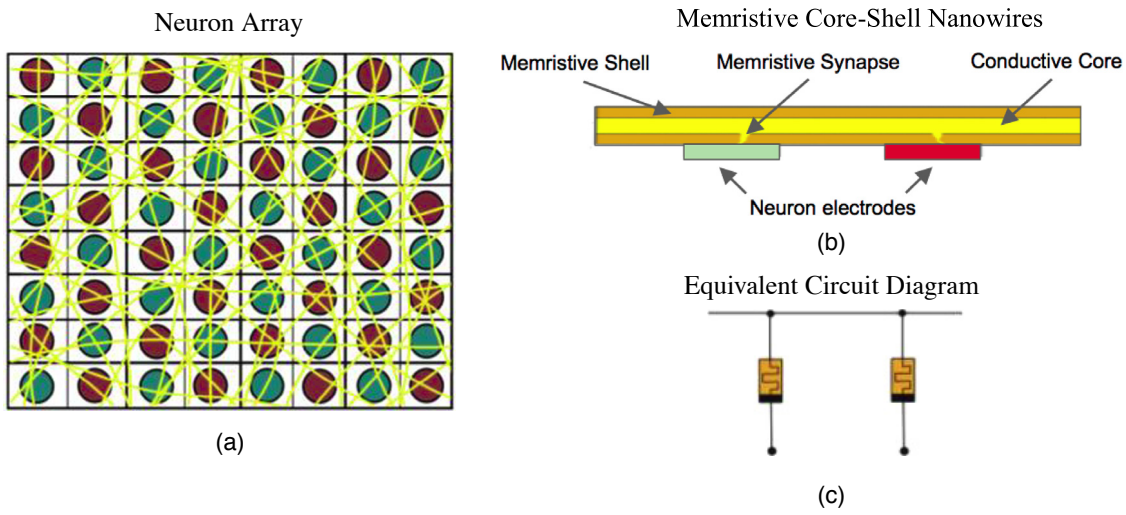


**Fig. 1.** (a) Each red–green pair corresponds to the input and output of a single neuron, tiled across the entire chip. The $MN^3$ is connected with a nanowire mesh overlaid on the neuron grid. (b) Each electrode forms a memristive synapse with the neurons below. (c) The equivalent circuit diagram of Fig. 1b. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**Table 1**
Network specifications.

| Description | Value |
|---|---|
| Nanowire diameter | $d = 100$ nm |
| Neuron width/length (square) | $l = 4\ \mu$m |
| Neuron spacing | $a = 1\ \mu$m |
| Nanowire mat thickness | $s = 1\ \mu$m |
| Wire packing fraction | $p = 0.25$ |

### 1.1. Network architecture

The network we describe is based on a network of core–shell memristive nanowires, or nanofibers, with a conductive core and a memristive shell. The architecture, which we dub the *Memristive Nanowire Neural Network*, or $MN^3$, is shown in Fig. 1. The wires serve as the interconnect layer in an array of CMOS neurons tiled in a square array. Each red–green pair corresponds to a single neuron. It is important to note that unlike conventional architectures, the neurons are not connected in the CMOS layer (Yu, et al., 2018). Instead, a layer of nanowires is deposited on the surface of the silicon, connecting the neurons in a stochastic manner. Metal pillars are grown through the nanowire layer to connect them to the electrodes below. The cores of the nanowires are conductive to allow for signal transmission between neurons, while the shell is made from a memristive material, allowing for the formation of memristive synapses at the interface between each nanowire and neuron.

We can approximate the neuron and synapse densities by using a simple geometric approach. We use the assumptions in Table 1 on the achievable feature sizes in the network. Note that the wire packing fraction is the density of wires compared to that in a close-packed structure (Batch, Cumiskey, & Macosko, 2002).

From $l$ and $a$, we can determine how many neurons $N$ will fit in a 1 $cm^2$ area:

$$N = 1/(l + a)^2 = 1/(0.0004 + 0.0001)^2 = 4 \times 10^6. \quad (1)$$

We can now determine how many wires contact each electrode if the wires are close-packed, $w_c$, and then multiply by the packing fraction to get the average number of wires contacting each electrode, $w$. We have

$$w_c = ls/d^2 = (0.0004)(0.0001)/(0.00001)^2 = 400. \quad (2)$$

Assuming a packing fraction for the wires of 0.25, i.e. only 1/4 of the maximum packing density, we arrive at the wire density per electrode $w$. This value can be increased at the expense of neuron density. We have

$$w = pw_c = (0.25)(400) = 100. \quad (3)$$

Now we can calculate the total number of synapses $S$ in the network. Since a synapse is formed at the intersection of each wire with each electrode, the total number of synapses is equal to the number of electrodes multiplied by the average number of wires per electrode. We have

$$S = Nw \approx (4 \times 10^6)(100) = 4 \times 10^8. \quad (4)$$

The density of neurons in the $MN^3$ ($4 \times 10^6$ neurons per $cm^2$) is several orders of magnitude higher than state-of-the-art values reported in the literature, including Intel's Loihi (218,400 neurons per $cm^2$) (Davies et al., 2018), IBM's TrueNorth (~12,157 neurons per $cm^2$) (Merolla et al., 2014), and Stanford's Neurogrid (~39,620 neurons per $cm^2$) (Benjamin et al., 2014). Two factors contribute to this sharp increase in density: the integration of the synapses and the wiring into a single compound structure, and the offloading of the synapses and the wiring from the surface of the CMOS to a sparsely connected nanowire layer.

By removing the wiring and synapses from the CMOS layer, the neurons can be close-packed as tightly as possible, drastically increasing neuron density. Similarly, the stacked mat of nanowires (connected to the neuron electrodes through vertically grown metal pillars) has a high density of wires, which are capable of connecting neurons across long distances, resulting in a high synapse density. The number of overlap of nanowires is estimated to be 10 (so the nanowires are stacked approximately 10 high). This gives enough spacing so that metal deposition techniques, such as sputtering, can fully penetrate the mesh. Simulations were conducted to verify that the presented wire density is compatible with this fabrication process. Regarding polarities, there are no inhibitory synaptic polarities as all conductances are positive. The neurons (external nodes) are used as hyperbolic tangent units, so they can take both positive and negative values. The memristor polarities themselves are all aligned towards the nanowire nodes (based on the wire core acting as the bottom electrode).

The sparsity of the resulting connection layer is important in reducing the total amount of wiring needed to connect the network.