Materials Science

http://www.aimspress.com/

AIMS Materials Science, 2(4): 530–545.
DOI: 10.3934/matersci.2015.4.530
Received date 29 June 2015,
Accepted date 23 November 2015,
Published date 02 December 2015

Research article

Computational capacity and energy consumption of complex resistive switch networks

Jens Bürger [1,*], Alireza Goudarzi[2,*], Darko Stefanovic[2], and Christof Teuscher[1]

[1] Department of Electrical and Computer Engineering, Portland State University, Portland, OR, 97211, USA

[2] Department of Computer Science, University of New Mexico, Albuquerque, NM, 87131, USA

[*] **Correspondence:** Email:jb22@pdx.edu, alirezag@cs.unm.edu.

Abstract: Resistive switches are a class of emerging nanoelectronics devices that exhibit a wide variety of switching characteristics closely resembling behaviors of biological synapses. Assembled into random networks, such resistive switches produce emerging behaviors far more complex than that of individual devices. This was previously demonstrated in simulations that exploit information processing within these random networks to solve tasks that require nonlinear computation as well as memory. Physical assemblies of such networks manifest complex spatial structures and basic processing capabilities often related to biologically-inspired computing. We model and simulate random resistive switch networks and analyze their computational capacities. We provide a detailed discussion of the relevant design parameters and establish the link to the physical assemblies by relating the modeling parameters to physical parameters. More globally connected networks and an increased network switching activity are means to increase the computational capacity linearly at the expense of exponentially growing energy consumption. We discuss a new modular approach that exhibits higher computational capacities, and energy consumption growing linearly with the number of networks used. The results show how to optimize the trade-off between computational capacity and energy efficiency and are relevant for the design and fabrication of novel computing architectures that harness random assemblies of emerging nanodevices.

Keywords: memrisive networks; atomic switch networks; network model; reservoir computing

1. Introduction

Emerging nanoscale resistive switches provide possible solutions for creating future computing architectures that are faster, less expensive and more energy-efficient by exploiting their intrinsic switching characteristics [1, 2]. Recent progress in the fabrication of atomic switches [3], as one class of emerging nanodevices, allows the random assembly of these devices into larger networks [4–6]. In [7] it was argued that the complexity of these networks resembles that of biological brains, in which the complex morphology and interactions between heterogeneous network elements are responsible for powerful and energy-efficient information processing [8]. Contrary to designed computation, where each device has a specified role, computation in random resistive switch networks does not rely on specific devices, but is encoded in the collective nonlinear dynamic switching behavior as a result of an applied input signal.

Atomic switches, as well as memristive devices [9, 10], are history-dependent resistive switches. Application of a bias voltage can change the conductance of the device. The nonlinear relationship between an applied input and a resulting conductance change in these devices perform nonlinear transformations of the input, similar to the dynamics found in biological synapses [11–14].

Harnessing the intrinsic nonlinear characteristics of these emerging nanodevices assembled in random structures has been shown for the memristor [15–17] as well as for the atomic switch [4–7]. Simulated random memristor networks have been employed to implement *reservoir computing* (RC) [18, 19]. RC is a computational approach initially inspired by cortical microcircuits, in which computation takes place by translating the intrinsic dynamics of an excited medium, called a reservoir, into a desired output. By observing the nonlinear responses of a random resistive switch network to different input signals, it was possible to perform simple pattern classification [15, 17] as well as computationally more demanding tasks by using multiple independent random assemblies [16]. In contrast to the simulation-based results, work on *atomic switch networks* (ASN) has demonstrated physical random assemblies and discussed fabrication parameters that determine the network morphology [7]. Based on these fundamental device and network characteristics that resemble the complexity of biological brains, it was argued that these networks are viable candidates for the physical realizations of brain-inspired information processing, such as reservoir computing.

Here we present a modeling and simulation framework that enables a detailed analysis of resistive switch network morphologies as determined by nanowire lengths, distributions, and density. The computational capabilities of different network morphologies are analyzed with respect to the compressibility of the measured network signals. We put the computational capabilities into perspective by comparing to corresponding energy-consumption data. This comparison outlines a trade-off between computation and energy consumption. A modular approach is presented that provides a more energy-efficient architecture while achieving higher computational capabilities. Our results demonstrate what constitutes computationally useful networks with respect to network morphology, density, and signal amplitudes. Based on the used modeling parameters, future fabrication of random resistive switch networks can be guided to achieve a desired trade-off between computational capacity and energy consumption.

2. Materials and Methods

2.1. Resistive Switches

Atomic switch networks have so far been the only physical implementation of random resistive switch networks, and were demonstrated in the context of reservoir computing. In this section we will establish the functional similarities between atomic switches, as one type of resistive switch, and other device types, and argue that the methods and results presented here are valid for a broader range of resistive switches. Detailed comparisons of different implementations of resistive switches can be found in [1, 20].

Resistive switches with a *metal-insulator-metal* (MIM) structure, where the insulator is typically a metal-oxide such as TiO_2 or WO_x [1, 10], establish changes in the device conductance by redistributing oxygen vacancies within the metal-oxide. In the absence of an input bias, the oxygen vacancies remain at their positions, which leads to the history-dependent conductance of the device. Volatile behavior was demonstrated for a WO_x device [21]. Here spontaneous diffusion of the oxygen vacancies causes a dissolution of conducting channels in the WO_x thin-film and a return to a low-conductance ground state of the device.

The atomic switch as a gap-type device based on crossing Ag_2S and Pt wires achieves conductance changes by changing the concentration of Ag^+ cations which allows growing metal protrusions of Ag atoms that eventually form a bridge between the two wires. The width of that bridge determines the conductance of the device [20]. In the absence of an input bias, the thermodynamically unstable atomic bridge eventually dissolves and the atomic switch returns to a low conductive equilibrium state [13].

An important functional similarity between MIM and gap-type devices is the nonlinear relation between the applied input and the resulting device state and current. Strukov and Williams presented an exponential ionic drift model that describes how an applied electric field changes the effective activation barrier and velocity for ionic-drift within the metal-oxide [22]. Similarly, Tamura *et al.* observed the exponential dependence of applied bias and switching time and argued this to be caused by a required minimum activation energy necessary to form a metal bridge within the atomic switch [23].

Throughout this paper we will refer to the modeled devices by their general category of resistive switches. This more general discussion will allow our presented concepts to be translated to a broader range of devices that adhere to the discussed qualitative similarities relevant for nonlinear computing.

2.2. Resistive Switch Modeling

In this section we will describe the device model used in our simulations. Our focus is on capturing the fundamental switching characteristics of the discussed devices, not on precisely reproducing empirical data obtained from a specific device. As outlined in the preceding section, resistive switches are characterized by history-dependent nonlinear conductance changes and state decay.

We adopt a memristor model as presented for WO_x devices in [1, 24]. In the original model the conductance as well as the state change are defined as follows:

$$G = \left[(1-w)\,\epsilon\left[1-\exp\left(-\theta V\right)\right] + w\gamma\sinh\left(\delta V\right)\right]\frac{1}{V} \tag{1}$$

$$\frac{dw}{dt} = \lambda\sinh\left(\eta V\right) - \frac{w}{\tau} \tag{2}$$

Here the internal device state is modeled by w, V is the applied input bias, $\epsilon, \theta, \gamma, \delta, \lambda, \eta, \tau$ are the model parameters calculated from the experimental data (detailed experimental data is property of authors of [1, 24] and has not been released to the public). The nonlinear switching, as explained by the exponential ionic drift model [22], is modelled by the sinh term. This model captures well the nonlinear switching as a function of the applied input and the current state.

However, such a first-order model, using only one variable to describe the device conductance, does not capture effects such as Ag^+ cation concentration changes. Such a change affects a device's response to future bias signals, but does not reflect into the actual device conductance. In [20] this difference was described as the memristor using a single variable to model the size of an ion-doped area, while the atomic switch uses two variables, one to model the height of an Ag protrusion, and another to model the width of the atomic bridge that emerges after the Ag protrusion has reached a sufficient height. Recently a second-order memristor model was presented that follows a similar modeling approach to the atomic switch [12]. Here the second variable that functions as an enabler for a subsequent change in conductance is the internal device temperature. Application of an applied bias signal increases the internal device temperature due to Joule heating, which in turn affects drift and diffusion processes described earlier.

To account for effects such as Joule heating or Ag protrusions, we use equation 2 to model an internal state w' that does not directly reflect in the device conductance. Furthermore, we extend the model to implement the different state decays based on the device state. As shown in [21,13], the rate of dissolution of the atomic bridge or the diffusion of ions to an equilibrium state is state-dependent and enables short- and long-term memory within a single device. We adopt equation 2 as well as describe state variable w, which models the device conductance as:

$$\frac{dw'}{dt} = \lambda \sinh(\eta V) - \frac{w'}{\tau}(1 - w') \tag{3}$$

$$w = f(w') \tag{4}$$

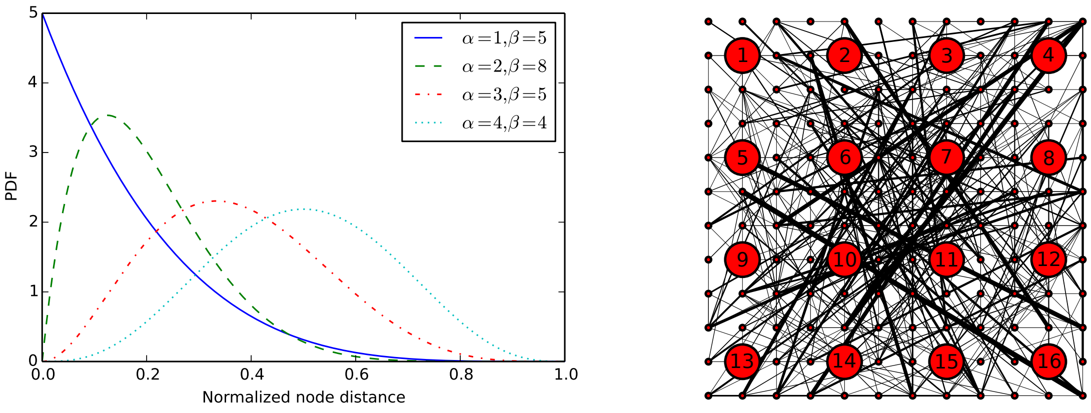
For our simulations we employ a binary switching function f that thresholds w' to create two distinct conductances. Binary behavior of atomic switches was shown in [13] and is also found in some memristive devices [25]. Different levels of sub-surface Ag^+ concentrations are required to establish or dissolve an atomic bridge (length and width of Ag protrusion) [26]. This implies different threshold values for the internal state variable w' to perform device switching. We model this by applying a hysteresis function to w'.

As random assemblies of resistive switch networks will exhibit variations in nanowire diameters, and hence in the resulting device sizes (e.g., gap size of atomic switches), variation is an integral system characteristic [7]. In [5] device parameter ranges for ASN were shown. These parameter ranges were not simply described by a Gaussian distribution around a mean value, but could span several orders of magnitude. For our experiments we apply a simplified model to create device variation where we draw device parameters uniformly from respective parameter ranges.

2.3. Network Modeling

While the application of memristive networks to reservoir computing was simulation-based [15–17], physical assemblies of atomic switch networks were reported in [4–7,27]. A simple modeling approach with focus on localized conductance changes of such networks was shown in the supplementary documentation of [5].

Here we expand the modeling of such networks with the aim to investigate the relation between network morphologies, computational capabilities, and energy consumption. Network morphology is defined by the average length of nanowires. Their density can be controlled by the underlying copper seed posts. In [5] the effects of copper seed posts' shape, size, and pitch on the network's nanowire



(a) Beta Distributions

(b) Network Example

Figure 1. (a) Different beta distributions and the resulting relative nanowire lengths. The x-axis represents the normalized nanowire length and the y-axis the probability of creating a nanowire of that length. (b) Example graph of modelling a random 16_2 memristive/atomic switch network with $\alpha = 2, \beta = 5$, and $\xi = 4$. The large circles represent the interface nodes to the underlying CMOS layer. The small circles create a narrower supporting grid that guides the density and morphology of the network structure. The graph edges represent memristive devices and the edge width is an indicator for the connectivity range, with increasing edge widths representing longer-range connections between nodes.

length and density were described. Smaller seed posts lead to long wires while larger seed posts lead to more fractal local structures. The pitch of the seed posts is a control parameter for the network density.

The connection of the random assembly of nanowires with the size of the seed posts implies that the distributions of the nanowire lengths can be described in terms of some probability density function. Large seed posts would cause more localized connections while small seed posts would result in long-range connections. Hence, we use a *probability density function* (PDF) that can capture different distributions. We use a beta-distribution for our purposes. A beta-distribution B produces values in the interval of $[0, 1]$ and is controlled by the parameters α and β that define the mean value and the skewness around the mean.

$$\begin{aligned} P(x,\alpha,\beta) &= \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1} \quad \text{(PDF)} \\ \mu &= \frac{\alpha}{\alpha+\beta} \quad \text{(Mean value)} \\ S &= \frac{2(\beta-\alpha)\sqrt{\alpha+\beta+1}}{(\alpha+\beta+2)\sqrt{\alpha\beta}} \quad \text{(Skewness)} \end{aligned} \tag{5}$$

$B(\alpha,\beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}dx$ is a normalization factor ensuring that $P(x,\alpha,\beta)$ is a probability measure. We use this PDF to model the distribution of the relative nanowire lengths within a random network. To translate the PDF into nanowire lengths we create a map with normalized euclidean distances ($[0, 1]$) from every seed post within the network. Starting from an initial node, we add a nanowire to the network by drawing a value from $P(x,\alpha,\beta)$ that determines the nanowire length. Beta-distributions

Table 1. Examples of network parameters.

ID	α	β	ξ	Connection length	Density
N1	1	1	2	uniform	sparse
N2	1	10	8	short	dense
N3	10	1	4	long	semi-sparse
N4	10	10	6	medium (small σ)	semi-sparse
N5	5	5	2	medium (medium σ)	sparse

with a focus on local connections ($\alpha < \beta$) will produce more fractal structures as nanowires branch out around the selected starting node for the wire. Distributions favoring long-range connections ($\alpha > \beta$) will connect distant parts of the underlying grid of seed posts. In Fig. 1a we show some examples of $P(x, \alpha, \beta)$ as a function of the control parameters α and β.

For the modeling of the underlying seed post grid we define two post types. Interface seed posts allow the application and reading of network voltages. These posts are established at a more coarse-grained pitch. The second grid is a supporting grid and more fine-grained than the interface grid. This supporting grid improves the formation of fractal structures for localized wire growth [7] and the establishment of more complex structures between the interface nodes; without a supporting grid establishment of multiple devices between two interface nodes will not be favourable, which limits the morphological diversity. The density of a network is defined by the indegree ξ of a node. The parameter ξ defines the average number of devices connected to each grid node. The number of resulting devices, and hence the density, is then given by the total number nodes in the supporting grid multiplied by ξ. The fabrication of networks with different densities was demonstrated in [5]. In Fig. 1b we show an example network model with 16 interface nodes, a supporting grid at a third of the pitch of the interface grid, and a mix of short and long-range connections. For all simulations we have used 16 interface nodes arranged in a 4×4 grid as shown in Fig. 1b. If not mentioned otherwise we have used a supporting grid at half the pitch of the interface grid. Throughout this paper we will refer to this network as of size 16_1, meaning that it has 16 interface nodes and 1 node between two interface nodes (half the pitch).

For a more intuitive understanding of how the defined network parameters inter-relate, we give five examples and describe the resulting network morphology (Table 1). In Fig. 2 and 3, we added the network IDs to the presented results to illustrate the relation between network morphology and information processing capacities.

We simulate the resistive switch networks by treating them as temporarily stationary resistive networks that can be solved efficiently using the *modified nodal analysis* (MNA) algorithm [28]. After calculating one time step using the MNA, we update the memristive devices based on the node voltages present in the network to account for the dynamic state changes of memristors.

3. Results

3.1. Network Morphology and Computational Capacity

As was outlined by Demis *et al.* [7], the structural similarities of ASN and biological brains (i.e., fractal branching similar to dendritic trees) suggests that such complex random assemblies provide hardware platforms for efficient brain-inspired computing. A characteristic feature of cognitive architectures is the nonlinear transformation of an input signal into a high-dimensional representation more suitable for information processing [29,30]. In particular, for $N > M$, the input signal $\mathbf{u} \in \mathcal{R}^M$ is transformed to $\mathbf{x} \in \mathcal{R}^N$ by the dynamics of the cortical microcircuits driven by sensory inputs. In the case of random resistive switch networks this means that the measured signals at the interface nodes are ideally nonlinearly related to the input signal and then provide a platform for brain-inspired computing.

A suitable transformation of the input requires rich dynamics that can preserve relevant distinctions between different input signals in the high-dimensional space [19]. This property is attributed to the dynamics of a system in the critical regime where the distinctions between states do not diverge or converge [31–33]. To provide such a rich dynamics, the activity of the nodes should show the least amount of redundancy. In other words, the dynamics of the nodes should be as uncorrelated as possible. The question is how one could measure that and how this measure would be related to the parameters of the system.

Here we introduce a simple measure that can describe the dynamics of random network signals in a way that is meaningful in the context of information processing. We study the compressibility of the network dynamics as a proxy for its richness. Specifically, we use *principal component analysis* (PCA) to transform the system dynamics into a principal component space [34].

The distribution of variations in the principal component space indicates the amount of redundancy between the different dimensions of the original system. The variation in each principal component is given by the corresponding normalized eigenvalue of the covariance matrix of the system. To calculate these, we record the network dynamics on all interface nodes as a result of an applied network input. All network signals are expressed in the network state matrix X. The covariance matrix of the network dynamics is then given by $C = X^T X$, and the eigenvalues are obtained by diagonalizing, $C = U \Lambda U^{-1}$. The diagonal elements of Λ, $\{\Lambda_1, \Lambda_2, \ldots, \Lambda_N\}$, are the eigenvalues of the corresponding dimension i and can be normalized as $\lambda_i = \frac{\Lambda_i}{\sum_{i=1}^N \Lambda_i}$. Since λ_i are normalized as a probability measure, we can describe their evenness using a single number $\mathcal{H} = -\sum_{i=1}^N \lambda_i \log_2(\lambda_i)$. This is an entropy measure and describes how evenly the λ_i are distributed. In one extreme case, where the system nodes are all maximally correlated, only one eigenvalue will be 1 and the rest will be zero, and the resulting entropy will be $\mathcal{H} = 0$. In the other extreme case, where the nodes are maximally uncorrelated, the eigenvalues will be identical and equal to $\frac{1}{N}$, and the resulting entropy will be $\mathcal{H} = \log_2 N$. In the latter case, every node in the network represents something unique about the properties of the input signal that cannot be described by a combination of the rest of the nodes.

In the following we present entropy measurements as a function of the network morphology. As outlined in section 2.3, the morphology is modeled based on a beta distribution with parameters α and β as well as the indegree ξ that defines the device density. We apply a 5Hz sine wave to the upper left interface node of a 16_1 network and connect the lower right node to ground (0V). Fig. 2 shows the network entropies as a function of α,β, and ξ. Across all densities, the highest entropies,

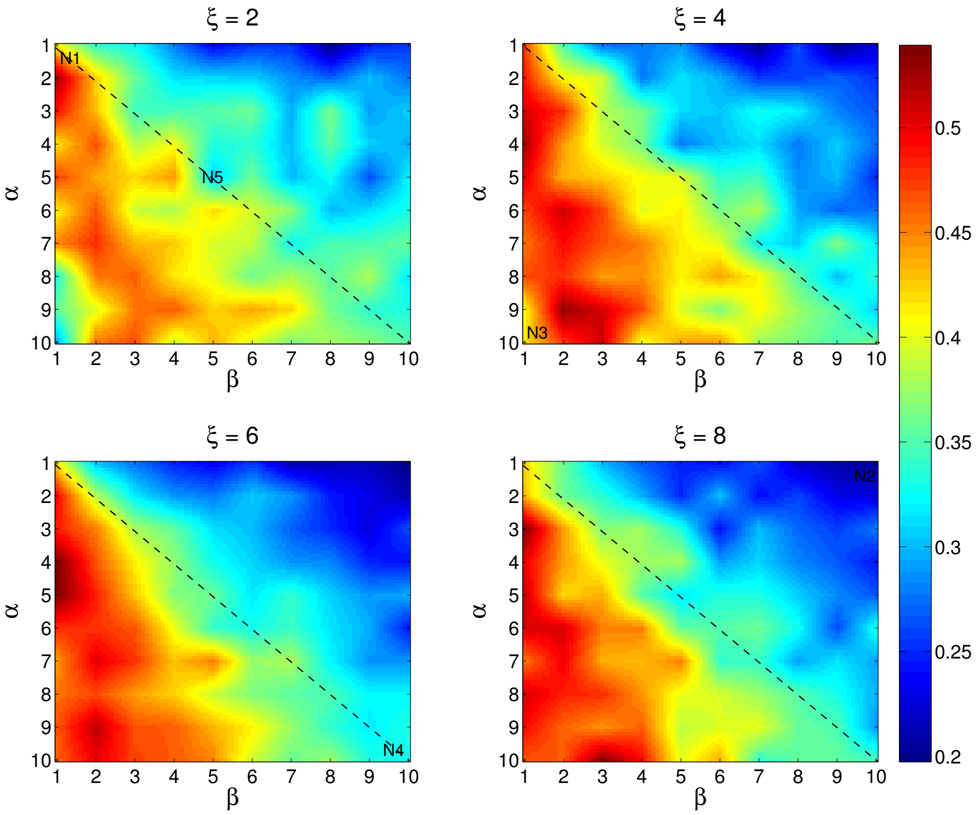


Figure 2. Entropy as a function of the network topology defined by the beta distribution values α,β and the indegree ξ ($v = 8V$). Independent of the density, best entropies are achieved for more globally connected network morphologies described by $\alpha \geq \beta$ (lower triangles of the plots). With respect to the network density an indegree of $\xi = 6$ has resulted in best entropies. Lower ξ values provide fewer signal paths and hence less network activity. Beyond $\xi = 6$, additional connections do not contribute any further to the effective network morphology. Average numbers of network devices were $150, 250, 350, 450$ for $\xi = 2, 4, 6, 8$, respectively. Markers Nx refer to table 1.

and hence the least linearly-dependent network states are achieved with a majority of the nanowires being equal or longer than half the normalized maximum euclidean distance (lower left triangles in Fig. 2 where $\alpha \geq \beta$). Long-range connections spatially distribute the input signal across the network without much voltage drop. Hence, different areas of the network experience a sufficient bias voltage to exhibit switching dynamics useful to information processing. Increasing network densities, which in other words describes the number of devices per area, creates more signal paths and due to device parallelism also higher conductive connections. This also leads to better distribution of the input signal and to an expansion of the morphologies for which larger entropies can be achieved.

Besides the network morphology, the amplitude v of the input signal also greatly affects the network dynamics due to the exponential dependence of applied bias and either activation energy to form an atomic bridge (for atomic switches) or the velocity of ionic drift (for memristive devices). In Fig. 3 we show the entropy as a function of α,β, and v for networks with an indegree $\xi = 6$. It can be seen that the average entropy increases as we increase v. This is related to larger voltages enabling more devices to exhibit switching activity and hence affect the network dynamics. The similarity in the plots as compared to Fig. 2 implies that increased input voltages also allow better spatial distribution of the input and more areas of the network to exhibit switching activity. Note that these plots present qualitative results on the dependence of v and entropy, but the absolute values of v are device-dependent and can change for devices with different threshold behavior.

We also studied network sizes 16_0 and 16_2. The 16_0 networks have shown very similar results to the 16_1 networks presented here. The morphological similarity between the 16_0 and the more globally connected 16_1 networks is that they do not form many local fractal structures. Contrary, the 16_2 networks are characterized by more complex morphologies between the interface nodes. While this might closely resemble complex structures in biological brains, in the context of passive electrical networks, these fine-grained fractal structures with many devices in between interface nodes lead to alleviated individual switching dynamics. While this could be circumvented in simulations by increasing the input amplitude v, in practice this is not a viable approach for reasons of safe operation and energy consumption.

3.2. Energy Consumption

In the previous section we have shown that best computational capabilities are achieved for dense, globally connected networks ($\alpha \geq \beta$) and larger signal amplitudes v. The viability of these results has to be evaluated with respect to the energy that would be consumed by such networks. Based on the application of a 5 Hz sine wave with amplitude v, we calculate the total energy consumption of a network over time T as $E(t) = \int_{i=0}^{T} V_i(t) I_i(t) dt$, with $V(t)$ being the time-dependent signal amplitude and $I(t)$ the current drawn by the network. As the energy consumption is very application-specific, we consider our findings as qualitative measures that highlight the general relations between the network parameters and the consumed energy. Absolute energy numbers presented here are not of relevance, only the information on how drastically energy consumption changes with network parameters. Fig. 4a shows the resulting energy consumption for different α and β, averaged over different ξ and v. The distributions of the energy data across the α and β plane resemble the entropy distributions seen in Fig. 2 and 3, which confirms that high entropies come at the cost of high energy consumption (relative to the consumption at low entropies) caused by high conducting paths in denser networks. This finding is further supported by plotting energy vs. entropy (Fig. 4b). Here we plot the averaged energy

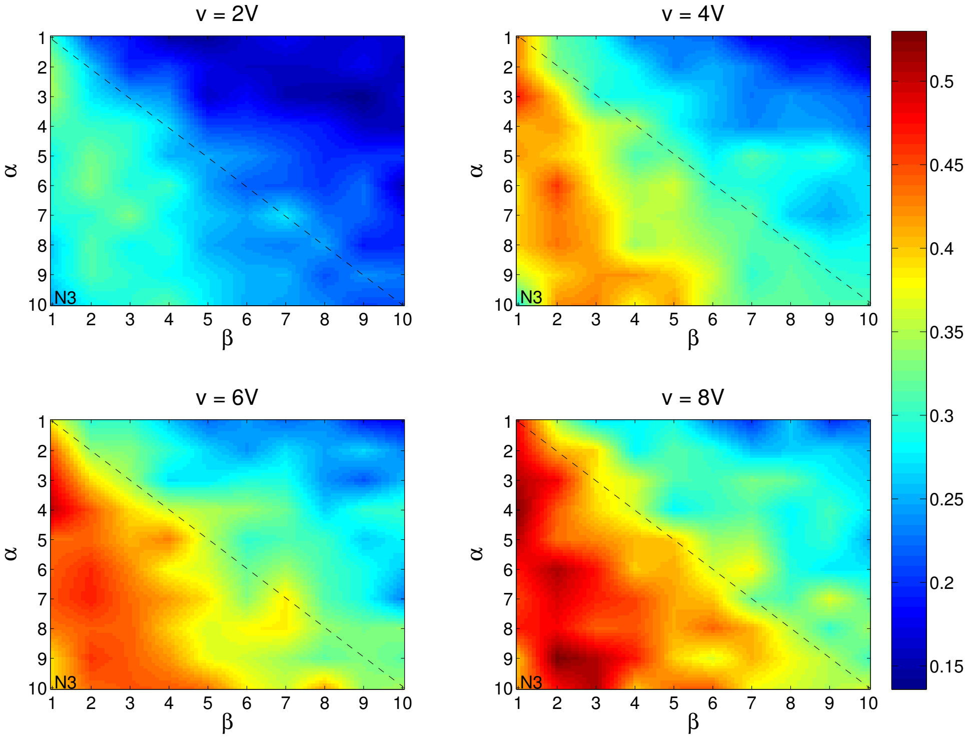
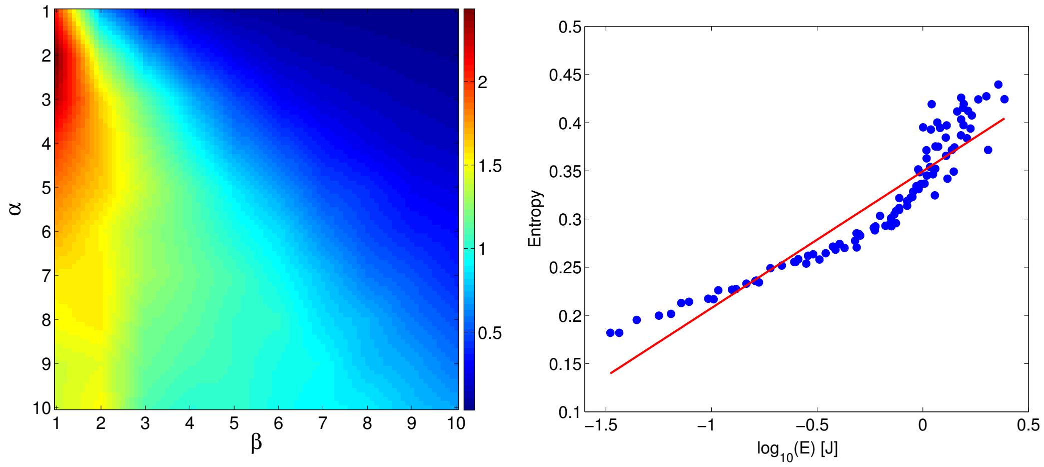


Figure 3. Entropy as a function of the network topology defined by the beta distribution values α, β and the input signal amplitude v ($\xi = 4$). Increasing signal amplitudes v lead to higher bias voltages for individual network devices and results in higher switching activity. Highest entropies are again achieved for more globally connected networks with $\alpha \geq \beta$ (lower triangles of the plots). Markers Nx refer to table 1.



(a) Average Energy Consumption

(b) Energy vs. Entropy

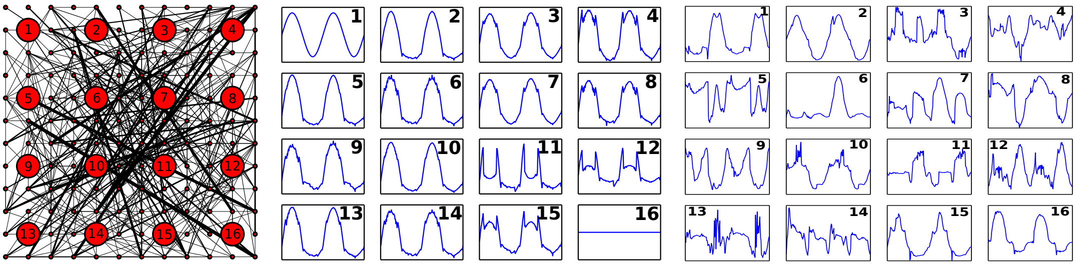
Figure 4. Energy consumption in random resistive switch networks. (a) Averaged energy consumption for different network morphologies expressed as $log_{10}(E)$. Energy grows exponentially when transitioning from locally connected ($\alpha = 1, \beta = 10$) to more globally connected networks ($\alpha \geq \beta$). The link between energy and entropy is shown in (b). Entropy grows linearly with exponential energy increase.

against the averaged entropy for corresponding setups. We can see how entropy grows with energy. However, as the energy grows exponentially, increasing entropy can lead to over-proportional energy consumption.

3.3. Improved Information Processing with Random Resistive Switch Networks

As the computational performance of a single random network is limited by exponentially increasing energy consumption or strong linear dependence at low energies, we will outline an approach to increase entropy with linear growth of energy. In [27] a hierarchical approach of ASN was presented that combined multiple small-world networks on a single chip. Similarly, we have shown a hierarchical approach that embedded independent networks (similar sizes as presented here) in a reservoir computing architecture and showed that a memory and computationally demanding application could be solved [16].

The concept relies on extracting only a subset of signals from each independent network. This allows harnessing the different processing caused by the different random structures of the independent networks. Furthermore, in [16] we have extracted a differential signal from each network. By retrieving a network output as the difference of two network nodes, differences in the spatial and temporal dynamics within networks can be better captured than using signals measured with respect to a common ground. In Fig. 5 we show two examples of 16 network signals. Fig. 5b shows signals obtained from a single random network (qualitatively comparable to signals presented in [7]). Readouts from a single network show some nonlinearities; however, they are mostly characterized by strong linear dependence across the signals (low entropy of 0.37). In contrast, with 16 independent networks exposed to the same input, we can significantly increase the richness of the measured signals. In this example the entropy is 1.79. The total energy consumption of the hierarchical independent networks scales



(a) Network Example

(b) Single Network

(c) Independent Networks

Figure 5. Examples of networks and node signals. (a) Example 16_2 network with $\alpha = 2, \beta = 5$, and $\xi = 4$ indicating the 16 interface nodes. (b) 16 Signals measured from a single 16_1 network with the input applied to node 1 and the ground node connected to node 16. The presented signals have an entropy of 0.37. The numbers correspond to the physical location of the nodes within the network. (c) Signals measured from 16 independent 16_1 networks. The entropy for this example is 1.79. Signals were measured as the difference between interface nodes 2 and 9. The numbers represent the network number. All networks were created with $\alpha = 1, \beta = 5, \xi = 4, v = 2V$

linearly with the number of networks. Considering the gain in entropy, this approach poses a much more viable approach for brain-inspired information processing than relying on single networks with high density and high signal amplitudes. A comparison of how energy and entropy relate in the two presented setups is shown in Fig. 6. The plotted data represents collected data from single networks as well as for the 16 networks approach with $\xi = 4$ and increasing v. The circled areas approximately mark the energy and entropy data obtained with a 2 V input signal. The average energy difference is 16 which corresponds to the number of networks used. However, compared with single networks that exhibit similar energy consumption (around $v = 4..6$) significantly higher entropies can be achieved.

4. Discussion

We have presented a detailed analysis of how the morphology of random resistive switch networks affects their computational capacity and energy consumption. It will be an application-specific choice of how much entropy is required by the individual networks and what costs one is willing to pay in terms of energy. The hierarchical approach and the better entropy per energy ratio is possible as we utilized different independent networks. This implies that cognitive computing is not merely the product of sufficient excitement of network elements but is also rooted in the heterogeneous morphologies across networks, as demonstrated for neural populations in cortical microcircuits [35]. This ability to harness randomness is a strong argument for the future fabrication of nanoscale resistive switch networks. We found that networks with strong fractal structures (i.e., 16_2 networks) performed worse. However, in biological brains such fractal structures (dendrites) are an integral part of the information processing. Biological dendrites are active components with voltage- and calcium-gated ion channels that regulate synaptic responses [36], while the fractal structures in resistive switch networks are purely passive elements. Further investigation of the functional differences might provide better fabrication techniques and methods to utilize fractal resistive switch networks.

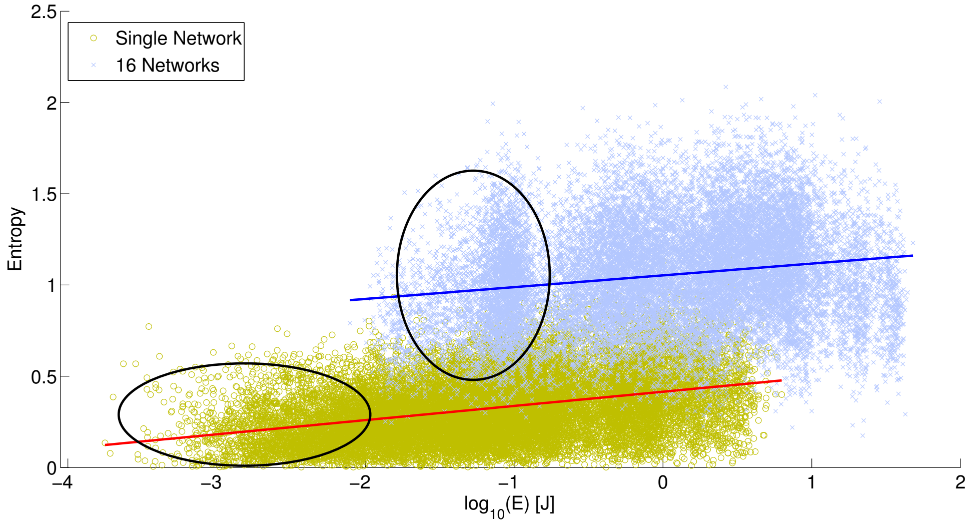


Figure 6. Energy vs. entropy for single networks and a hierarchical approach with 16 independent networks. The circled areas highlight an example of networks with the same network parameters ($v = 2, \xi = 2$). Scaling up the number of independent networks is more energy- as well as computationally-efficient as compared to increasing the entropy of a single network by means of scaling up either density or voltage. Data obtained with $v = 2$ and $\xi = 2 \ldots 8$.

5. Conclusion

The nonlinear dependence of input bias and state transitions in resistive switches allows random assemblies of such devices to exhibit complex behavior useful to computation. The design parameters used to control the simulated network morphologies relate to design parameters for the physical fabrication of the networks. We have shown how network morphologies correlate with computational capacities and energy consumption. A hierarchical approach was presented that allows us to increase computational capacities more energy-efficiently as compared to increasing information processing in single networks. These findings provide insights in the abilities and limitations of random nanoscale resistive switch networks and should serve as a design guide for investigation and fabrication of future nanoscale computing architectures.

Acknowledgments

This work was supported by the National Science Foundation under awards # 1028378, # 1028238, and # 1318833, and by DARPA under award # HR0011-13-2-0015. The views expressed are those of the author(s) and do not reflect the official policy or position of the Department of Defense or the U.S. Government. Approved for Public Release, Distribution Unlimited.

Conflict of Interest

The authors declare no conflict of interest in this paper.

References

1. Chang T, Yang Y, Lu W (2013) Building Neuromorphic Circuits with Memristive Devices. *Circuits, Systems Magazine, IEEE* 13: 56–73.
2. Crutchfield JP, Ditto WL, Sinha S (2010) Introduction to focus issue: intrinsic, designed computation: information processing in dynamical systems–beyond the digital hegemony. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 20: 037101.
3. Hasegawa T, Ohno T, Terabe K, et al. (2010) Learning Abilities Achieved by a Single Solid-State Atomic Switch. *Adv Mater* 22: 1831–1834.
4. Avizienis AV, Sillin HO, Martin-Olmos C, et al. (2012) Neuromorphic atomic switch networks. *PLoS ONE* 7: e42772.
5. Sillin HO, Aguilera R, Shieh HH, et al. (2013) A theoretical, experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology* 24: 384004.
6. Stieg AZ, Avizienis AV, Sillin HO, et al. (2012) Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Adv Mater* 24: 286–293.
7. Demis EC, Aguilera R, Sillin HO, et al. (2015) Atomic switch networks – nanoarchitectonic design of a complex system for natural computing. *Nanotechnology* 26: 204003.
8. Sporns O (2011) *Networks of the Brain*. The MIT Press, Cambridge, MA.
9. Chua L (1971) Memristor - The missing circuit element. *Circuit Theory, IEEE Transactions on* 18: 507–519.
10. Strukov DB, Snider GS, Stewart DR, et al. (2008) The missing memristor found. *Nature* 453: 80–83.
11. Jo SH, Chang T, Ebong I, et al. (2010) Nanoscale Memristor Device as Synapse in Neuromorphic Systems. *Nano Lett* 10: 1297–1301.
12. Kim S, Du C, Sheridan P, et al. (2015) Experimental Demonstration of a Second-Order Memristor, Its Ability to Biorealistically Implement Synaptic Plasticity. *Nano Lett* 15: 2203–2211.
13. Ohno T, Hasegawa T, Tsuruoka T, et al. (2011) Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nat Mater* 10: 591–595.
14. Sah MP, Chua LO (2014) Brains Are Made of Memristors. *IEEE Circuits , Systems Magazine* 14: 12–36.
15. Bürger J, Teuscher C (2013) Variation-tolerant Computing with Memristive Reservoirs. *Nanoscale Architectures (NANOARCH), 2013 IEEE/ACM International Symposium on*, 1–6.

16. Bürger J, Goudarzi A, Stefanovic D, et al. (2015) Hierarchical Composition of Memristive Networks for Real-Time Computing. *Nanoscale Architectures (NANOARCH), 2015 IEEE/ACM International Symposium on*, 33–38.

17. Kulkarni MS, Teuscher C (2012) Memristor-based Reservoir Computing. *Nanoscale Architectures (NANOARCH), 2012 IEEE/ACM International Symposium on*, 226–232.

18. Jaeger H (2001) The ''echo state'' approach to analysing, training recurrent neural networks - with an Erratum note. GMD Report 148, German National Research Center for Information Technology.

19. Maass W, Natschläger T, Markram H (2002) Real-time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Comput* 14: 2531–2560.

20. Hasegawa T, Nayak A, Ohno T, et al. (2011) Memristive operations demonstrated by gap-type atomic switches. *Appl Phys A* 102: 811–815.

21. Chang T, Jo SH, Lu W (2011) Short-term memory to long-term memory transition in a nanoscale memristor. *ACS Nano* 5: 7669–76.

22. Strukov DB, Williams RS (2009) Exponential ionic drift: fast switching and low volatility of thin-film memristors. *Appl Phys A* 94: 515–519.

23. Tamura T, Hasegawa T, Terabe K, et al. (2006) Switching Property of Atomic Switch Controlled by Solid Electrochemical Reaction. *Jpn J Appl Phys* 45: L364–L366.

24. Chang T, Jo SH, Kim KH, et al. (2011) Synaptic behaviors, modeling of a metal oxide memristive device. *Appl Phys A* 102: 857–863.

25. Gaba S, Sheridan P, Zhou J, et al. (2013) Stochastic memristive devices for computing, neuromorphic applications. *Nanoscale* 5: 5872–5878.

26. Ohno T, Hasegawa T, Nayak A, et al. (2011) Sensory, short-term memory formations observed in a Ag2S gap-type atomic switch. *Appl Phys Lett* 99: 203108.

27. Stieg AZ, Avizienis AV, Sillin HO, et al. (2014) Self-organized atomic switch networks. *Jpn J Appl Phys* 53: 0–6.

28. Litovski V, Zwolinski M (1997) *VLSI Circuit Simulation and Optimization*. Chapman & Hall, London, UK.

29. Rabinovich M, Huerta R, Laurent G (2008) Transient Dynamics for Neural Processing. *Science* 321: 48–50.

30. Buonomano DV, Maass W (2009) State-dependent computations: spatiotemporal processing in cortical networks. *Nat Rev Neurosci* 10: 113–125.

31. Bertschinger N, Natschläger T (2004) Real-time computation at the edge of chaos in recurrent neural networks. *Neural Comput* 16: 1413–1436.

32. Langton CG (1990) Computation at the edge of chaos: Phase transitions and emergent computation. *Physica D: Nonlinear Phenomena* 42: 12–37.

33. Snyder D, Goudarzi A, Teuscher C (2013) Computational capabilities of random automata networks for reservoir computing. *Phys Rev E* 87: 042808.

34. Bishop CM (2006) *Pattern Recognition, Machine Learning (Information Science, Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

35. Honey CJ, Thivierge JP, Sporns O (2010) Can structure predict function in the human brain? *NeuroImage* 52: 766–776.

36. Shah MM, Hammond RS, Hoffman DA (2010) Dendritic ion channel trafficking , plasticity. *Trends Neurosci* 33: 307–316.