ABSTRACT

FRICK, NIKOLAY. Neuromorphic Computing with Self-Assembled Resistive Switching Nanocomposites. (Under the direction of Prof. Thomas LaBean).

Conventional computer architectures approached practical physical limits that cannot match the efficiency of biological brains despite tremendous progress in the last decade. The collective dynamics of neurons, driven by complex electrochemical processes and a constantly evolving network of synapses, makes the brain the most sophisticated computer known to humanity. Thus, unconventional neuromorphic computing with devices that mimic the dynamic functionality of biological brains could serve as a better platform for building intelligent systems that could learn and think as efficiently as humans.

This dissertation aimed to explore theoretically and experimentally the properties of artificial synapse networks made of 3D self-assembled nanoscale silver sulfide nanowire resistive switching composites. These neuromorphic composites exhibited nonlinear electrical properties and some functionality of biological synapses that permit nonlinear threshold switching computation. In addition, unlike isolated resistive switching devices, the networks of memristive devices exhibited negative resistance that extends their potential applications as computing devices.

Thus, the dissertation comprised studies that addressed a range of questions in percolation theory, characterization, and modeling of individual silver-sulfide resistive switches and their networks and provided methods for performing nonlinear computation.

In particular, this work provided a suite of simulation software packages to aid in researching percolation and conductivity phenomena in multicomponent composites. With the help of the simulation software, the analysis of the behavior of the resistive switching phenomena in the composites was performed and used to build physical composites. In addition, the computation method with a network of silver sulfide resistive switches and the experimental and theoretical results were provided. Finally, the limitations of the current studies and guidance for future work are discussed.

Neuromorphic Computing with Self-Assembled Resistive Switching Nanocomposites

by
Nikolay Frick

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Materials Science and Engineering

Raleigh, North Carolina
2021

APPROVED BY:

Prof. Donald Brenner	Prof. Lewis Reynolds
Prof. Daryoosh Vashaee	Prof. Thomas LaBean Chair of Advisory Committee

ACKNOWLEDGEMENTS

I am indebted to the supportive, innovative, and collaborative culture of the NC State University that played a central role in the success of this project.

First and foremost, I am grateful to my advisor Prof. Thomas LaBean for suggesting the project, generous support, and patience during my study. Especially I would like to thank him for providing freedom of exploration that led me to develop original methods and find unconventional solutions.

I want to thank my advisory committee members, professors Lewis Reynolds, Donald Brenner, and Daryoosh Vashaee, for providing invaluable feedback and suggestions that fuelled my research.

Additionally, I would like to thank my colleagues at the LaBean group - Mahshid Hosseini, Damien Guilbaud, Drs. Ming Gao, Abhichart Krissanaprasit, and Jacob Majikes for accompanying me in my experimental work.

My gratitude extends to the Department of Materials Science and Engineering administrative staff, especially Edna Deas, for navigating the formal part of my graduate studies at NC State and George Martell for his exceptional technical and IT support.

I also thank NC State Graduate School staff, especially Dr. Katie Homar, from the Academic and Engineering Writing Support, for her assistance during the dissertation completion grant.

I would like to thank Michael Polyakov and Dr. Julius Goth for inspiring and challenging conversations on a broad range of computer science and artificial intelligence topics.

Finally, I am deeply grateful to my family, Eli, Felix, and Yulia, for their unparalleled support and encouragement while being neglected during multiple years of my study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Organization of the dissertation

Chapter 1 discusses the main objective of the work by introducing the theoretical simulation framework for studying percolation and conductivity in nanoscale circuits formed by randomly distributed nano devices, presented in detail in chapters 2 and 4. The first chapter also introduces the experimental approach used in analyzing physical random networks, discussed in detail in chapter 4. Finally, the language and terminology of the dissertation are established by reviewing related work.

Chapter 2 discusses the simulation framework (CircuitSymphony and Percolator) and its use in analyzing several different composites with realistic parameters. In particular, this chapter will describe the relationship between conductivity and junction resistance between composite fillers for wires with different aspect ratios in 2D and 3D. A few other examples of a particular interest applicable to laboratory setup will be given, including conductivity of a binary filler system and dimensional distributions of spheres and cylinders. Finally, a novel approach in studying systems with various degrees of clustering or agglomeration will be demonstrated.

Chapter 3 introduces a rationale for computation with memristive networks and builds upon CircuitSymphony and Percolator frameworks to analyze the dynamics of the internal circuitry of such systems under external electric potential drive. A comparison between

random and organized memristor networks is discussed. Some unique network-specific effects of memristor networks are studied such as negative differential resistance. A voltage projection approach to computation with random memristor circuits is presented and discussed in computation from n-ary logic gates to image recognition.

Chapter 4 presents experimental results from a single memristor device to a 3D composite with an integrated network of devices. Experimental approaches in materials characterization are presented, such as manufacturing multi-electrode arrays and single wire electronic measurements. Characterization of a single Ag_2S memristor and a stochastic model of its dynamic state is introduced. Ag/Ag_2S PCL and PDMS methods of manufacturing composite with nanoscale fillers are introduced. Finally, computation of XOR gate and image recognition are presented and discussed.

Chapter 5 discusses some intermediate results in the development of thermally coupled nano-neuristors and their complex dynamics.

Chapter 6 concludes with a discussion of possible future research directions of computing with self-assembled chaotic systems.

1.2 The history of modern computing

The principle of modern transistor-based digital computers was first proposed by Alan Turing and is referred to as a Turing machine [Turing, 1937]. A Turing machine consists of a tape and a head, where the head can travel arbitrary distances in any direction along with the tape and read or write data one operation at a time. The moving head is a stateful device containing a set of rules that are executed according to the symbols read from the tape. At the end of the program execution, the head stops completely. Despite its simplicity, the Turing machine's abstract design of a universal computing machine that could perform arbitrary computations was the basis of the modern computing industry for decades.

Eight years after Alan Turing published his work, John von Neumann proposed a computer architecture that, with a few modifications, describes the operation of most modern computers [von Neumann, 1993]. Von Neumann realized an electronic version of Turing's machine with three essential components: a Control Unit (CU), an Arithmetic/Logic Unit (ALU), and a Memory Unit (MU). Typically CU and ALU are combined into one unit called Central Processing Unit (CPU). Neumann's work started a modern era of digital computing that lasted to date.

However, von Neumann's architecture has significant limitations that need to be ad-

dressed. Both a program and data are stored in the same memory module and need to be accessed separately via the same data bus. This creates a bottleneck that constrains the maximum performance of digital computers. Besides the technical limitations of the memory caused by the data bus throughput, there is a physical limit on the maximum practical density of the binary switches, the main components of the CPU. Since 1971 Moore's law, correctly predicted that the packing density of integrated circuits would double every 18 months [Moore, 2006]. However, recently CPU manufacturers started to fall behind the trend. Due to quantum mechanical limits, the practical minimum size of the logical switch gate is above 5 nm [Zhirnov and Cavin, 2015]. Thus modern computers operate at their practical physical limit, and a new computational paradigm is needed. For this, we can turn to biology to mimic how analog information is processed much more efficiently than human-made digital computers.

1.3 Analog computing

Before delving into the most recent advances in analog computing systems, it is important to understand the history of analog computation. The first known analog computing device was developed by Greeks more than two thousand years ago for astronomical calculations and as an aid to navigation at sea [Efstathiou and Efstathiou, 2018]. Since then, analog computers have undergone several steps of evolution at each one relying on mechanical, hydraulic, and, in the 20 century, electronic principles. At each stage of advancement, analog computers enabled humanity to solve more complex problems but still cater to a very narrow set of problems. Further, the invention of a transistor in the 1940s led to the development of integrated circuits, the building blocks of digital computers, that offered a unified approach to problem-solving, displacing the advancement of analog computers as the most versatile tools for business and scientific tasks.

Despite the progress in miniaturization of integrated circuits, we have now arrived at the physical limits of hardware implementation of von Neumann architecture. To overcome this challenge, we need to turn to biological systems that combine both digital and analog computation. A natural model of a combined system is the brain. While DNA is used as a digital store to encode proteins, neurons in the brain form networks that store and process analog information. The primary advantage of distributed analog processing is that the spatial location of the computation region is indistinguishable from the region where data is stored.

Today, there is no unified theory that would describe or predict computation performed by biological neurons. The majority of the neurons in human brain communicate with each other via chemical synapses that encode the shape of electrical depolarization spiking signal that is propagated through an axon to other neurons. Despite the inefficiency in the information content per spike and natural failures, the communication between neurons is quite robust [Koch, 1998]. In this work, we will rely on the base principles that need to be implemented in artificial spiking neural networks in order to harness them for neuromorphic computation: 1) The artificial neuron needs to be able to create new synaptic connections and destroy old ones with neighbor neurons (associative learning) 2) The integration of signals from multiple sources and generating a signal if a threshold is reached (integrate and fire) 3) The ability of a neuron to reset to its original state after excitation (long-term volatility) and, 4) The ability of a neuron to generate a train of spikes of variable frequency with respect to the amplitude of input signal (frequency coding) and finally 5) the ability of a neuron to change state as a function of different input spike frequencies (synaptic plasticity) [Koch, 1998].

The idea of mimicking biological neuron function with an analog device was first proposed in 1960 by Robert Crane [Crane, 1961]. He named that device a "neuristor" as a device that mimics biological "neurons". The biological brain consists of billions of neurons, each of which has dendrites to accept signals from other neurons and a single axon, to send information to other neurons. Together neurons coordinate complex behavior from muscle control to cognitive functioning. A typical neuron has thousands of synaptic connections; every day it creates a few new connections and destroys a few old ones to form long-term memory [Saveliev, 2005]. To mimic this natural system, Crane proposed that any logical computation can be performed with a circuit of slowly burning fuses of various lengths. However, the apparent disadvantage of such a system would be that it can be used only once. With that idea, he developed a concept that allows performing arbitrary computations on fuses with an assumption that these fuses are reusable with some "refractory" period. The two most important properties of the proposal are that the device can store data and generate signals. Despite its simplicity, it took 52 years for a scalable transistor-free memristor-based neuristor model to be described and implemented by Pickett *et al.* in 2012 [Pickett et al., 2012]. We will further talk about neuristors implemented in our laboratory in chapter 4 of this dissertation.

1.4 Biological neurons and artificial neuristors

The engineering of autonomous artificial intelligence [Totschnig, 2020] would be among humanity's apex technological achievements, impacting all aspects of our activity. However, emulating the behavior of biological neurons and their networks requires a comprehensive effort across many disciplines that would require a reliable scaled and autonomous energetic support to hundreds of millions of independent devices. While currently there is no general theory that can explain advanced neurological functions like consciousness in a biophysical framework, we can assume that advanced cognitive functions are an artifact of collective network behavior in the brain and likely can be quantified in the form of network connectivity and the size of the network (citation needed). As engineers, we can at least identify the most critical properties of brain matter at the level of individual neurons and the connected collection of neurons, which certainly needs to be implemented to engineer an artificial bio-inspired autonomous intelligence.

The intelligence phenomenon is a neurological process in a living biological organism that relies on converting nerve impulses from the environment into an "internal state" that results in a series of corresponding complex reactions in order to meet its needs, such as food and procreation (citation needed). This process is autonomous and requires self-organization of groups of neurons and formation of information processing networks sensitive to particular stimuli with much biology behind the mechanism that we still do not know. However, we know that biological neurons in the cortex communicate with each other via transmission of action potential depolarization at appropriate firing rates through chemical synaptic terminals (citation), which, in its' turn, can either activate postsynaptic neuron's depolarization by releasing activator neurotransmitter or oppositely, depress depolarization by releasing inhibitory neurotransmitters [V. Torre, 1978]. The interplay of activating and inhibiting neurotransmitters on a dendritic tree serves as a model of nonlinear operations [K. Koch, 1982]. The strength of the synaptic connections can also change with time and the process is referred to in the literature as synaptic plasticity.

Biological neurons encode information via amplitude and rate of the spike train of action potential sent through axon terminals as a response to stimuli received from preceding neurons via dendrite terminals. In a nutshell, the key factors which influence the amplitude and the rate of firing are dependent on the neuron's membrane ability to repolarize or depolarize concerning the flow of K^+, Cl^- and Na^+ through the membrane's ionic channels. The flow of these ions depends on the external electric or chemical stimulus and whenever it pushes neuron's membrane potential above its depolarization threshold value. In a resting

neuron, the voltage across its membrane is at -60 mV. When this value is pushed above 60 mV, the neuron will start sending spiking signals along the axon until the excitation is removed. Alan Hodgkin and Andrew Huxley developed a model of which describes the mechanism of generation and propagation of action potential in axons [Hodgkin and Huxley, 1952].

Thus in a first approximation, an artificial neuron that is close to its biological counterpart must possess at least three basic properties. First, it must imitate the plasticity of the synapse, which can be approximated by a self-assigned weight or a variable synaptic strength at a particular site relative to the incoming signal from the upstream neurons (either to increase or decrease the value of the particular input channel). Second, an artificial neuron has to be able to generate an output signal for communicating information to the downstream neurons in the form of a rate-coded spike (as in biological neurons) or rate-approximated amplitudes (that are implemented in a vast majority of conventional artificial neural network models). Finally, it must selectively connect to nearby neurons via synapses and destroy old connections to provide means for memorization and forgetting.

All these qualities can be achieved via a combination of devices that can be manufactured in a laboratory. In particular, synaptic plasticity can be emulated with a passive memristive device or synaptor. Synaptors are typically resistive switching devices with positive differential resistance IV characteristics. We will discuss the properties of resistive Ag_2S atomic switches or synaptic devices in chapters 3 and 4. Spiking signal generation can be engineered with volatile, locally active memristive devices [Jin et al., 2018], aka neuristors [Pickett et al., 2012]. We will briefly discuss our preliminary data on autonomous spiking signal generation with VO_2 and NbO_2 trans-metal oxide neuristors in chapter 5. Finally, an experimental proof of synaptic junction destruction, via electric field mediated electromigration of Ag^+ ions in Ag_2S will be discussed in chapter 4 with some evidence towards a possibility of synaptic contact creation.

1.5 Memristor model

First theorized by Leon Chua in 1971, memristors are two-terminal devices in which resistance is equal to the integral of the total charge passed through the device over time [Chua, 1971]. Unlike an active electronic device, such as a transistor, memristors can't immediately act as a gate or an amplifier. In other words, memristors are devices with an analog memory that can store real-valued data in the form of a high resistance state of the device (Roff) or

oppositely low resistance state (Ron) resistance. The first study of a thin-film Transition Metal Oxide (TMO) TiO_{2-x} memristive device was presented in 2008 [Strukov et al., 2008a].

Memristors that retain their resistance state over a long period are called non-volatile memristors. These types of memristors can find applications in generic data storage such as Resistive Random Access Memory (ReRam) or in machine learning applications as units for storing synaptic weights. Sometimes these types of memristors are also referred to as "synaptors". Non-volatile memristors possess a high Ron/Roff ratio and recently were even used for simultaneous matrix-vector multiplications [Hu et al., 2018, Agarwal et al., 2015] and sparse coding [Sheridan et al., 2017].

On the other hand, volatile memristors can either gradually "forget" their states over some period or reset if no external field is applied. Volatility allows the memristor's state to have a complex temporal dependency on the history of the input signal within some time period. Most of the applications of such memristors focus on computing with spike-based programming [Du et al., 2017a]. Some types of memristors, which typically undergo metal-insulator via Mott-Peierls transition, have instant volatility such as VO_2 or NbO_2 and need a constant supply of power to retain their state [Ascoli et al., 2015, Ke et al., 2018].

It is important to distinguish different types of volatility. In particular, volatile memristors can store state for a limited period of time and last up to minutes. For example, the Mott-Peierls transition-based memristors, such as VO_2 and NbO_2, require a constant supply of voltage and lose their "ON" state instantaneously (switch from Ron to Roff). On the other hand, some anionic type oxide devices like $Au/WO_x/W$ have a short state retention lifetime on the order of 100s of milliseconds [Du et al., 2017b]. In this work, as will be discussed in chapter 4, our cationic type $Ag/Ag_2S/Ag$ devices can retain the "ON" state for up to a few minutes.

Most memristors are ionically conductive. There are different types of carriers in the material that result in resistive switching, primarily cationic or anionic. TMO memristors are anionic because the main carriers within the device are positively charged oxygen vacancies - cations, or oxygen anions [Yang et al., 2012]. Cationic memristors are electrochemical cells whose main carriers are positively charged metal anions such as Ag^+ diffusion in Ag_2S-α phase.

This work's main focus is cationic memristive nanowire devices and their configurations, as is shown in Figure 1.1. The configuration consists of 3 layers Metal/Insulator/Metal in this case $Ag/Ag_2S/Pt$. Intermediate phase Ag_2S is an insulator for electrons but can conduct Ag^+ or Cu^+ ions. Since Pt is an inert metal, with an opposite Ag electrode that serves as a reservoir of Ag^+ ions, this device can be switched on and off with an external field of appropriate

polarity. In particular, when the Ag electrode is connected to a positive terminal of the voltage source, Ag^+ ions will be pushed towards the inert Pt electrode and over a short period, will form a conductive pathway via a self-assembled Ag filament, thus switching the device to an ON state. This state has a refractory period from a few seconds to a few minutes and, without the external field, will return to the OFF state, which is a result of a conductive channel or metallic Ag filament. Decomposition is discussed in detail in chapter 4. The OFF state can be reached faster if the polarity of the external power supply is reversed and the Ag electrode is connected to a negative terminal.

The IV curve shown in Figure 1.1 (c) depicts a typical current response of the device to an external 3V sinusoidal excitation and 200 mHz frequency. The shape of the IV curve depends not only on device geometry and the material being used but also on the amplitude and frequency of the input signal. The IV characteristics of the memristive device stimulated with a higher 10 Hz frequency is shown in the inset of the Figure 1.1 (c). A fingerprint of memristive behavior is a pinched hysteresis loop in the IV plot [Chua, 2014]. In a cationic device, the root cause of the hysteresis loop is the migration of Ag^+ ions and expansion or contraction of the conduction channel formed by a metalized Ag filament. However, it might also be caused by other factors such as phase change due to the Joule heating such as in Mott devices, the influence of large magnitudes of the external field, and other secondary phenomena.

1.6 Reservoir computing

This work focuses on unconventional methods in computation with memristive networks. In this section, we will briefly review unconventional computing and discuss how our research relates to it. Unconventional computing denotes any computation architecture that deviates from conventional von Neumann architecture. Typical examples include DNA computing, Quantum computing, neurocomputing, and other approaches that do not require unconventional amounts of resources [Blakey, 2016, Blakey, 2010].

One of the most prominent examples of unconventional computing is reservoir computing which was proposed independently by Herbert Jaeger [Jaeger, 2004] as an echo state network (ESN) and Wolfgang Maas [Maass et al., 2002] as a liquid state machine (LSM). The primary difference between these two models is that LSM is a spiking neural network, whereas ESN approximates spiking via rate-coding. These models are essentially a recurrent neural network (RNN) that can effectively process temporal data. Unlike conventional

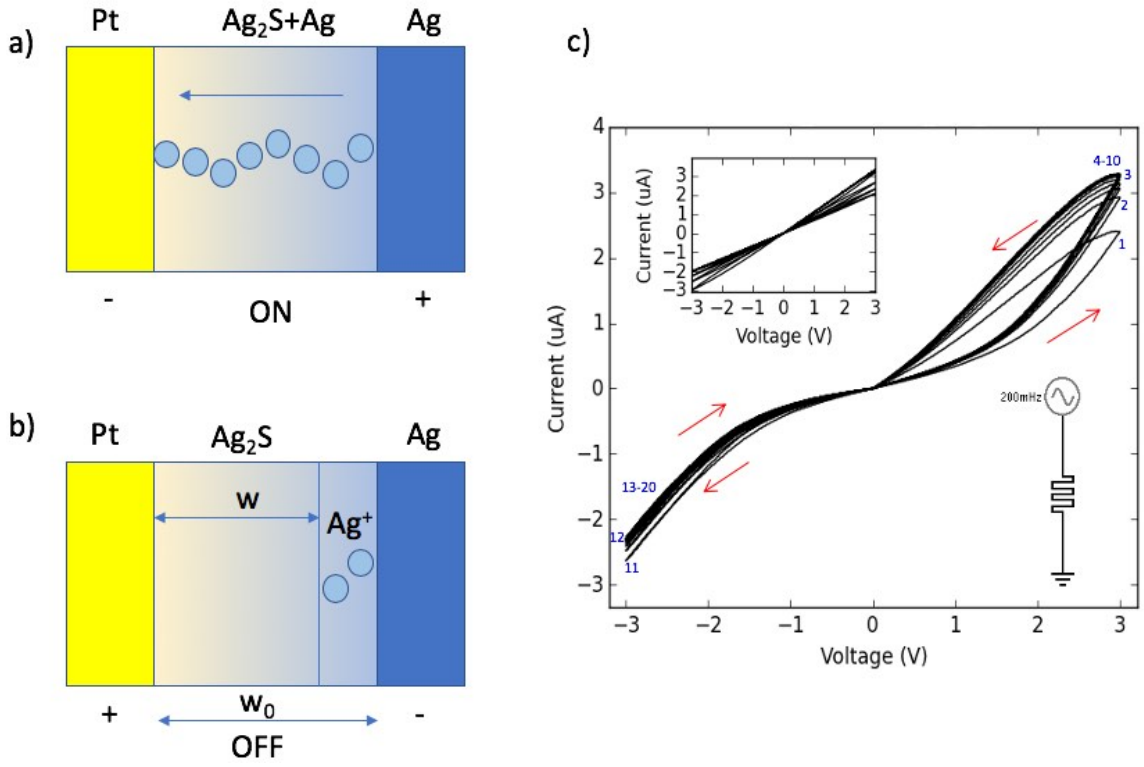


Figure 1.1: An example of a cationic memristor with a characteristic IV plot. The device consists of 3 layers Pt/Ag_2S/Ag. The Ag electrode is a reservoir of Ag ions and the Pt electrode is inert. An insulating chalcogenide phase of Ag_2S serves as a conductor of positively charged Ag+ ions from the Ag electrode under an electric field. a) Device can be turned ON when the positive terminal is connected to the Ag electrode. This causes Ag+ ions to build a conductive bridge between electrodes and reduce the resistance of the device. b) Reversing the polarity of the power supply switches the device back into the highly resistive OFF state. c) Typical IV characteristics of a memristive device stimulated with a 3 V sine wave with 200 mHz frequency. The top-left inset is the IV characteristics of the same device, but stimulated with a higher, 10 Hz frequency sine voltage wave.

RNNs, in reservoir computing, there is no need to modify all the weights within the network during the training; only the output has to be modified. Each of these random models has random recurrent connections between neurons, that enable them to retain the history of the input signal for some period. The states of the neurons and recurrent connections between them are correlated and, as a result, perform complex mathematical operations that can be filtered out with the tuned weights in the output layer.

Computation on reservoirs has shown both theoretically and experimentally promising results in predicting chaotic time-series [Du et al., 2017b, Lukoševičius and Jaeger, 2009, Jaeger, 2004], image recognition [Du et al., 2017b, Schaetti et al., 2016], and financial time-

series [Budhiraja et al., 2021]. One of the most interesting and recent novelties introduced are natural control systems, capable of representing realistic human motion [Jaeger, 2017] . In this report, Jaeger introduced a notion of conceptors, which is essentially a method of clipping the reservoir states in each particular training class in such a way that multiple different patterns can be distinguished with high confidence. This property is critical because, generally neural networks are not good at interpreting cross-domain data and numerous adversary examples show that it is easy to trick a neural network.

As the "no free lunch theorem" states, [Wolpert and Macready, 1997] regarding the supervised algorithms, an algorithm optimized in a particular domain will perform well in its domain but not better than random results obtained from the other domains not related to the training set. Because an algorithm, or reservoir, along with the hyperparameters are chosen to optimize a search with respect to the problem of interest but not the domain of all possible problems. Thus, compared to conventional artificial neural networks, a feed-forward neural network is a universal approximator of static functions whereas reservoir computing is equivalent to universal filter approximators [Maass et al., 2005, Grossberg et al., 2001].

1.7 Memristor based computation

The core hypothesis that motivates this work is that an arbitrary connected memristive circuit with some voltage inputs and current outputs can be configured to perform complex nonlinear computations. The basis for the hypothesis is that a non-volatile memristor, in its most primitive implementation, can perform a basic scalar multiplication operation via Ohm's law. Namely, the current flowing through the memristor encodes the underlying multiplication result (voltage across the memristor) x (memristor's conductivity). The accuracy of the multiplication operation depends on the sensitivity of the measuring equipment, the quality of the memristor and is bounded by the magnitudes of ON and OFF resistance states of the device. However, a simple multiplication operation in a single memristor turns into a more complex matrix operation when several memristors are assembled into a connected circuit. Thus in a random memristive circuit with a vector of n voltage inputs ($V_{in}^{1\times n}$) and a vector of m current outputs ($I_{out}^{1\times m}$) the output currents vector $\mathrm{I_{out}}$ can be expressed through voltage inputs via $n \times m$ matrix Z, namely $I_{out} = Z \times V_{in}$ where $Z^{n\times m}$. The elements of matrix Z are comprised of a set of summation, multiplication, and divisions between the resistances of individual memristors in the circuit. Thus the resulting current vector $\mathrm{I_{out}}$ is a

high dimensional nonlinear mapping that when m>n contains more information than the input V_{in} itself. The above formulation of the memristive reservoir is a projection of voltage input to current output via connecting Z matrix. In chapter 4, we will present our analysis of such memristive circuits and discuss their applications in nonlinear regression and image recognition. Finally, in chapter 5, we will describe physical implementations of randomly assembled 2D and 3D nanoscale memristive networks via randomly distributed memristive nanowires which we successfully used in solving nonlinear problems.

Despite the relatively straightforward manufacturing process of the nanocomposites with random memristive networks that don't require a clean room, research on computation via randomly assembled networks is quite unpopular in the community. One of the primary reasons is the production inconsistency and lack of standardization, as it is impossible to produce two similar distributions of networks in two samples even if they have a similar appearance. Thus below, we will briefly overview and compare two different unconventional computation approaches that we hybridized in our research.

1.7.1 Memristors and crossbar arrays

Crossbar arrays are, among all, the most popular way for utilizing memristors for storage and computation. These arrays constitute a 2D matrix of memristors, each of which can be individually accessed via column and row addressing (Citation). Memristive crossbar arrays have been used to assemble minimalistic networks of individually addressable thin-film memristors and harnessed for complex matrix operations and reservoir computation.

Typically, memristors in crossbar arrays are anionic TMO memristors such as HfO_x, WO_{3-x}, TiO_{2-x}, etc. Besides efficient addressability, this type of configuration can be used to perform complex mathematical operations such as matrix by vector multiplication, data clustering [Choi et al., 2015], solving partial differential equations [Zidan et al., 2018], and instantly computing the inverse of a matrix [Sun et al., 2019]. In other words, crossbar arrays can be considered a universal function approximator whose resolution can be scaled with the size of the I/O equipment's crossbar and the accuracy of the I/O equipment.

Instead of constructing networks via crossbar arrays, our 2D and 3D random memristive nanowire networks comprise an already interconnected system of resistive switches. This massive interconnection between elements produces a "cross-talk" effect between neighbor memristors that is increased with decreased distance between network elements. We will briefly discuss the importance of the "cross-talk" between memristors in theoretical chapter 3 and experimental chapter 4. Here we will only note that the "cross-talk" is an essential

feature that is only observable in a network of memristors that causes a spike-like change in conductivity even in strictly passive non-volatile memristors. Besides electrical coupling, the "cross-talk" can also be induced via Joule heating between neighbor spiking memristors, resulting in thermally coupled memristors states an essential property for building spiking neuristors. For example, Mott-Peierls based memristors with low phase transition temperatures such as VO_2 can be used to synchronize neighbor spiking memristors [Bohaichuk et al., 2019, Velichko et al., 2018, del Valle et al., 2020].

The convenience in deterministic addressability in crossbar arrays makes them less flexible in simulating the collective dynamics of biological neurons. In particular, it is unfeasible to achieve such a crucial property of biological brains as creating new synaptic connections with nearby and distant neurons, since the handling of the switching has to be deterministically controlled by state-of-the-art digital control boards, which are hard to scale.

In previous section 1.5, we introduced passive memristors as volatile/non-volatile systems or synaptors that can be used to store a resistive state, but they cannot be used to generate cyclic signals with DC voltage supply. In other words, signal patterns generated with such networks repeat the shape of the input signal, thereby allowing these systems to act as passive filters.

1.7.2 Evolution-in-materio

The term "evolution-in-materio" was coined by Miller *et al.* [Miller et al., 2014] to describe a method of manipulating the internal organization of a material, usually via a variant of evolutionary algorithms, to achieve some practical computational task. For example, Massey et al. [Massey et al., 2016] developed a classifier with metallic SWCNT nanotubes dispersed in a droplet of liquid crystals on top of a 16 electrode array by feeding voltage through the system. It took several hours to find a minimum of the loss function to solve a simple linear classification task. In another work, Mohid *et al.* used a genetic evolution algorithm to evolve an SWCNT/PMMA composite on a 12 electrode array. They used a mix of metallic and semiconducting SWCNT with which the authors could solve an NP-Hard bin packing problem in 100,000 evolutionary steps in 2 days [Mohid et al., 2014]. Similarly, Bose *et al.* demonstrated a reconfigurable logic gate by evolving a disordered network of gold nanoparticles separated by 1-octanethiols with an eight electrode array setup (two inputs, one output, and five static control terminals) [Bose et al., 2015]. The network was evolved with a genetic algorithm for several days kept under a constant temperature below

15 K. After 100 hours of evolving control voltages on the network, the algorithm found a combination of input voltages that configured the network to represent basic binary logic gates such as XOR or NAND. The configured state of the network decays after several days if the temperature is kept below 15 K or is erased if thermally cycled to room.

As detailed in chapters 3 and 4, the present work relies on a similar approach to find control weights. However, instead of a time-consuming genetic algorithm, we employ a Bayesian optimization algorithm, which is more appropriate for optimizing black-box functions that are expensive to evaluate.

1.8 Experimental approach

The main goal of this work is to understand the electronic properties of random self-assembled memristive nanocomposites and investigate potential computation applications. As a first step, a parallel and distributed simulator framework was designed and implemented. This framework consists of two software pieces, namely, Percolator, [Frick and LaBean, 2021b] and CircuitSymphony [Frick and LaBean, 2021a]. Percolator is a general-purpose particle generator and collision engine. It uses a 3D game engine to identify collision points and construct conductivity graphs that can further be converted into a representative circuit and solved on a circuit simulator. CircuitSymphony is a large-scale general-purpose circuit simulator that can perform a transient analysis of circuits with a few components and more than 106 elements using KLU and NICSLU algorithms [Davis and Natarajan, 2010, Chen et al., 2013b]. Each component works independently and can be scaled from a cluster to a single node. The orchestration can be achieved manually through any environment supporting REST protocol, or with our middle layer helper library, ResSymphony [Frick, 2021], to automate the creation of networks of particles and analyze their electronic performance. Chapter 2 will discuss these frameworks in greater detail.

With this simulation framework, we generated and analyzed arbitrary electronic circuits derived from randomly distributed resistive or memristive devices that perform contact junctions that can also be programmed to have different functional properties. For example, a collision pathway between one pair of components can be routed through a conducting wire, represented in a circuit diagram by a resistor with nominal resistance scaled with respect to the length of the pathway. If the pathway transverses a memristive type of wire, it will be replaced with a memristor element with an appropriately scaled Ron and Roff.

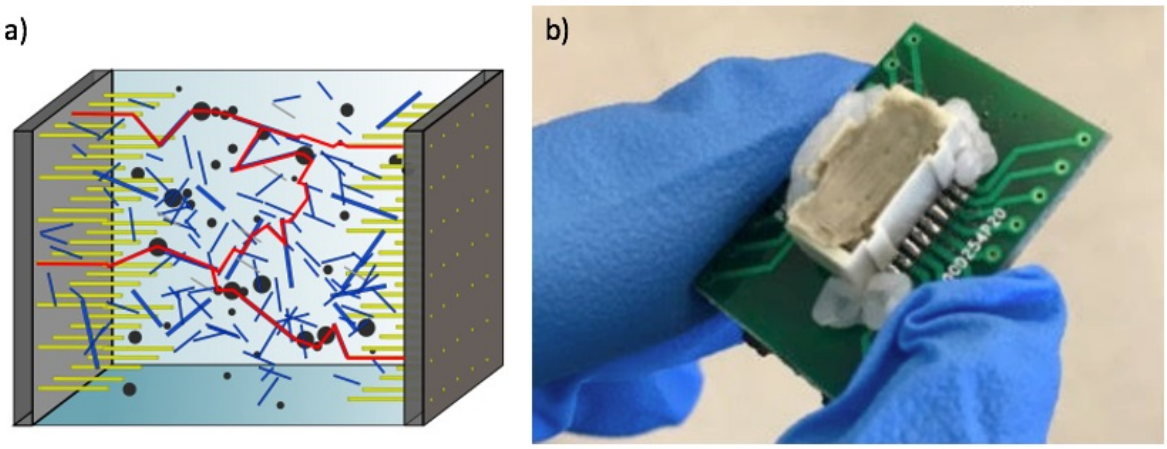


Figure 1.2: Conceptual illustrations of a) hypothetical composite with various kinds of fillers forming a complex percolating network sandwiched by two multi-electrode arrays. A joint connection between 4 randomly chosen array pins is shown with red lines and b) one of the implementation variants of the neuromorphic composite with 2% Ag nanowires by volume dispersed in a PCL matrix, housed in a miniature socket with 20 gold plated electrodes (not shown).

Chapters 3 and 4 will provide more details on how we used these models in our research, and how we can experimentally verify the models with experimental data in the laboratory.

In particular, we will discuss how random memristive networks can be used to compute nonlinear functions with various degrees of complexity in chapter 3. We took exclusive logical sum (XOR) as a nonlinear benchmark function, which is also the smallest binary classification problem that can't be solved with a linear decision boundary.

The summary of our approach in solving XOR classification problems is depicted in Figure 1.3. The simulation starts with configuring a JSON file with a definition of the circuit graph architecture, input, and output channels. Once a JSON file of the circuit is defined, it can be fed into the simulator, either directly into a GUI application or via a REST interface of the headless server. In the latter case, a user-defined program can read currents/voltages at every time step, also via a REST interface, for particular target components and decide whether to continue or stop the simulation. The simulator itself runs either a CPU or a GPU version of the matrix solver, which depends on the size of the circuit matrix (GPU kicks in after matrix size reaches 1000 x 1000 or more for nonlinear elements, i.e., when the matrix needs to be solved at every iteration). The voltages V1 and V2 encode inputs from the XOR truth table as boolean zeroes and ones that need to be mapped to a corresponding class L with boolean output. The algorithm randomly chooses the

corresponding pair of voltages and feeds them into the simulated network (middle panel of Figure 1.3, while simultaneously reading the output currents. The projection of input voltage into the current is a nonlinear mapping of n vector of voltages into m vector of currents where in this case n=2 and m=3. Figure 3 (right panel) shows a nonlinear projection in which dimensionality expansion converts a nonlinear problem into the domain where simple linear classification on the output current values is sufficient to solve the nonlinear XOR problem. This projection of voltages into currents produces a linearly separable solution, but in a higher dimension, that can be solved with a simple linear classification algorithm. We will discuss other more complex applications including projections that do not increase in dimensionality, in chapter 3.

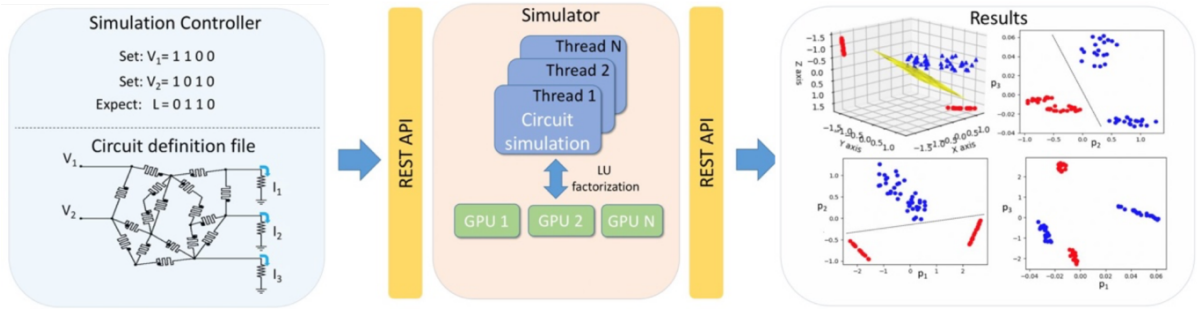


Figure 1.3: The flow diagram of a setup for solving the XOR problem with the random memristor network with two voltages and three current outputs (V1, V2, and I1, I2, I3, respectively). The far-right panel shows the 3-channel current response of the system to the input of XOR truth table values in the form of voltage, labeled in red (logical 0) and blue (logical 1).

Inspired by the simulation results, our experimental work explored the conductivity of single isolated and random networks of Ag and Ag_2S nanowires as in 2D and 3D composites, discussed in detail in chapter 4. Figure 1.4 shows an optical photomicrograph of a single Ag wire contacted by a nanomanipulator microelectrode tip. In the figure, the top end of the nanowire is attached to silver paint while the bottom end is in contact with the Platinum-Iridium microelectrode tip. Of particular interest are the characterization of reversible switching phenomena between Ag nanowires coated with a polyvinylpyrrolidone and single and network assemblies of Ag_2S nanowires. We constructed models of single and network behavior of memristive switches and compared them to biological synapses using these measurements. Chapter 4 will discuss advanced applications of the resistive switching

phenomena from the position of a reservoir computing framework. In the final chapter 5 of the dissertation, we will discuss some investigative results on generating neuron-like spiking signals with negative differential resistance devices.

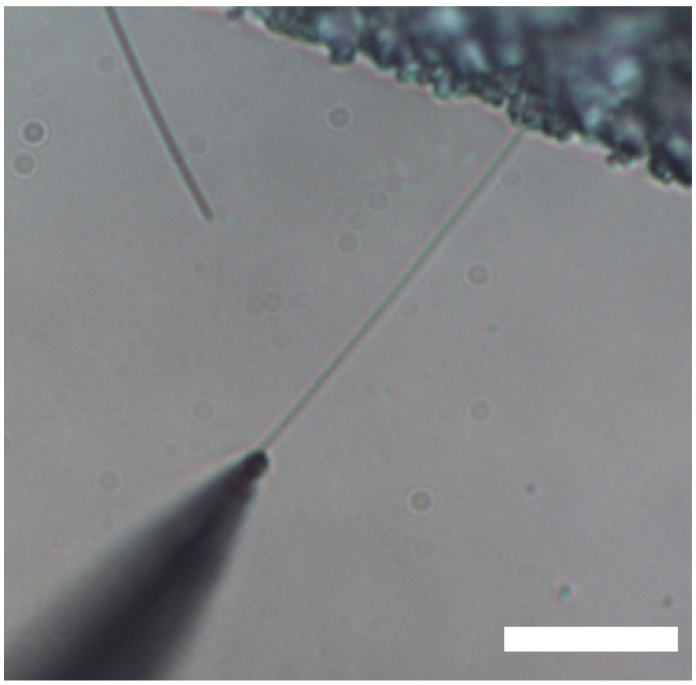


Figure 1.4: Optical microphotography of a single Ag wire during electrical characterization in a 2-probe setup with the help of a nanomanipulator.

CHAPTER 2

Percolation and conductivity studies of composites

2.1 Introduction

The development of functional nanocomposites with conducting nanoscale fillers has attracted considerable attention due to the potential applications in EM absorption [Saini et al., 2009, Lv et al., 2008, Yang et al., 2011], mechanical sensing [Tasaltin and Basarir, 2014, Yao et al., 2017, Hsu et al., 2016, Boland et al., 2016], energy storage [Choudhary et al., 2016, Yuksel et al., 2018], and controlled thermal and electrical conductivity for electronic applications [Chen et al., 2016].

Typical insulating materials used as the matrix phase in composites can be epoxy resin, paint, plastic polymer, photo-curable resin, clays, or gels, etc. Thus, the manufacturing of macroscopic samples with nanoscale filler (i.e., dispersed phase) materials is usually performed by trial and error and requires a significant time investment. Initial conditions may vary from one experiment to another, in which nanocomponents might be unevenly distributed, get agglomerated into isolated clusters, or, with an increase of aspect ratio (AR) of nanofillers, reduce the effective length via increased waviness (i.e., decreased persistence length) [Hicks et al., 2018]. In this chapter, "waviness" refers specifically to the ratio of the total length of the wire to the distance between its ends.

The phenomena of conductivity in composite materials can be modeled with the percolation theory, which describes the phenomena of network connectivity formed by the

placement or removal of conducting pathways. There are two main approaches in modeling percolation, lattices or continuum [Stauffer and Aharony, 2018, Sahini and Sahimi, 1994]. In lattice percolation, systems are modeled with repeating patterns of conducting wires of equal size, the intersection of which forms a junction. This approach is an idealization and lacks the real-world diversity of network conductivity seen in real composites. Continuum percolation, on the other hand, assumes a realistic distribution of objects in the medium. Continuum percolation is used in the current study because it is more suitable for modeling conductivity in composite materials.

In this chapter, our primary interest is the development of a scalable simulation framework for the understanding of electrical conductivity phenomena in materials with different fillers and geometries. To test our approach, we will compare our computation results for metallic nanowire filler-containing composite conductivity with fillers with laboratory results of composites with Ag nanowires and CNTs. Our secondary goal is to use this simulator to analyze the performance of complex polymer composites models with memristive networks for computation, which will be discussed in detail in chapter 3.

The process of manufacturing conductive nanocomposites is complicated due to multiple factors such as uneven distribution, secondary force interaction between particles (resulting in agglomeration), and the use of multiple filler types. Also, frequently during manufacturing, the nanoparticles are covered with an insulating layer, such as polyvinylpyrrolidone (PVP) on Ag nanowires, which increases the percolation threshold due to decreased probability of conducting contacts between elements [Selzer et al., 2016]

To overcome these experimental complications, it is preferable to have a computational model of nanoparticles in an insulating matrix with geometries as close as possible to real conditions, which would account for the following properties: waviness (also known as persistence length), contact (proximity radius), specific electronic properties of each component, and the junction resistance between them. Predicting the electric response of such composites with the help of *in silico* simulations would significantly accelerate the discovery and development of new composite materials. However, simulation of realistic percolation phenomena is challenging because it requires significant computational resources and multiple assumptions about the system, such as degree of agglomeration, waviness, and influence of the host matrix on conductivity. This work demonstrates a new method of studying conductivity and percolation phenomena in composites with fillers having arbitrary geometries using a 3D game engine.

Percolation phenomena in 2D and 3D is a highly active research field in which various approaches are taken based on the particular application. Traditionally, percolation phe-

nomena in composites are studied with filler elements organized in an idealized square regular lattice network, as it offers a convenient way to generate and quantize conductive clusters [Bunde and Dieterich, 2000, Shi et al., 2013, Shi et al., 2014, Kirkpatrick, 1973]. Lattice networks are useful for finding percolation threshold, but not so much for calculating conductivities of composites with realistic filler properties where more computationally demanding continuum percolations models considering many-body interactions of randomly distributed particles are used. An example to consider is curved nanowires with different degrees of alignment. For example, Hicks *et al.* studied conductivity of 2D distributions of curved nanowires where waviness, or curl ratio λ, defined as a ratio of total wire length $l_w$to distance between ends (effective length) l_{eff}. Eq 2.1, was approximated with Bezier curves which exhibited power law dependency of resistivity on the wires curl ratio [Hicks et al., 2018]. Recently percolation and conductivity of random distribution of binary systems such as wires and disks in 2D and 3D composites were studied with different fillers' shapes [Ni et al., 2018, Ni et al., 2019, Rahatekar et al., 2010].

$$\lambda = \frac{l_w}{l_{eff}} \tag{2.1}$$

However, most simulations focus on one type of filler with 1D geometry, and there is a lack of information about the simulation method. Additionally, these simulations are often performed in a rule-based fashion where it is non-trivial to apply the simulation framework to a more advanced setup with more complex geometries.

To our knowledge, there are no reports on percolation studies for composite materials where different components with various geometric shapes mediate conductivity. The primary difficulty in performing such simulations with 3D objects is in deriving the connectivity graph from the underlying distribution of components. It requires estimation of collisions of each element with its neighbors, which requires up to N^2 number of operations, where N is the total number of components.

Our novel approach is geometry-agnostic and is achieved via two necessary steps: a) each particle has a mesh volume and can be constructed from a set of primitive particles such as cylinder, disk, and sphere, and b) the resulting circuit is constructed through collision analysis between individual parts of the volume.

In this work, our main contribution is software that can precisely generate and map the arbitrary distribution of nanoscale configurations of conductors in a material into an electric circuit. In particular, integration with the physics engine allows the software to locate the 3D coordinates of surface collisions between particles and assemblies despite

their complex geometries. The simulator's overall possibilities go beyond the scope of this report, and we will only focus on a small subset of experiments that can be performed with this platform.

Recently, a similar graph-based approach for estimating conduction with complex networks was reported by Yao *et al.* [Yao et al., 2020]. The work doesn't rely on a 3D physics engine for collision estimation. However, each reported experimental setup was also represented by a network of links, which means that each object has zero volume. Also, their implementation relied on solving circuit equations via Cholesky decomposition through a third-party Circuitscape package [R. Anantharaman, 2020]. Although Cholesky decomposition is sufficient for solving linear systems, it is not suitable for cases with active elements or reactive energy storage that typically produce asymmetric circuit matrices [Pillage et al., 1994]. In contrast, our simulator employs sparse solvers tailored for solving circuit matrices such as NICSLU [Chen et al., 2013b], a multithreaded modification of KLU solver [Davis and Natarajan, 2010] capable of handling both symmetric and asymmetric circuit matrices.

Our simulation approach has a set of useful features applicable in large-scale simulations, such as the additive generation of particles, fine control over particle distribution, and application of periodic boundaries for continuum simulations. In particular, we will show that the ratio of the filler's length to the volume of the simulated composite is a source of a significant error in conductivity estimation which can be eliminated with the application of periodic boundary conditions.

In the results sections 2.8 and 2.9, we will discuss 2D and 3D percolation and conductivity with various length wire fillers with different waviness, junction resistance, and other simulation parameters summarized in Table 2.1. We will further discuss and relate obtained simulation data with our laboratory results in section 2.10.

Section 2.12 discusses a study on percolation and conductivity analysis of binary composite with a wire and a sphere in 2D and 3D. These findings can aid in navigating experiments for the manufacturing of sensors and other smart devices whose behavior relies on the synergistic functionality of different nanomaterials.

We will conclude with section 2.13, where we will discuss the effect of agglomeration on conductivity in 3D systems. Studying agglomeration (i.e., clustering) of nanofillers with high aspect ratios is important because achieving well-mixed, uniform distribution of dispersed phases poses a significant obstacle in developing such nanomaterials and studying realistic distributions with simulation tools.

To summarize, the goal of this chapter is to introduce Percolator [Frick and LaBean, 2021b] and CircuitSymphony [Frick and LaBean, 2021a] software packages and show how

they can be used to analyze conductivity in arbitrary 2D and 3D composites and reinforce these observations with experimental data.

2.2 Percolator and CircuitSymphony software

To predict and analyze the electrical performance of arbitrary distributions of particles, the custom software tools Percolator and CircuitSymphony were developed. Percolator allows the creation of virtual composites with arbitrary geometries and filler concentrations and their visualization in 2D or 3D with a convenient user interface, shown in Figure 2.1. The framework also consists of server and client components, enabling it to be used in standalone or distributed cluster environments. The server component is written in Java programming language, which permits efficient memory management and speed of calculations comparable to implementations in C++ programming language.

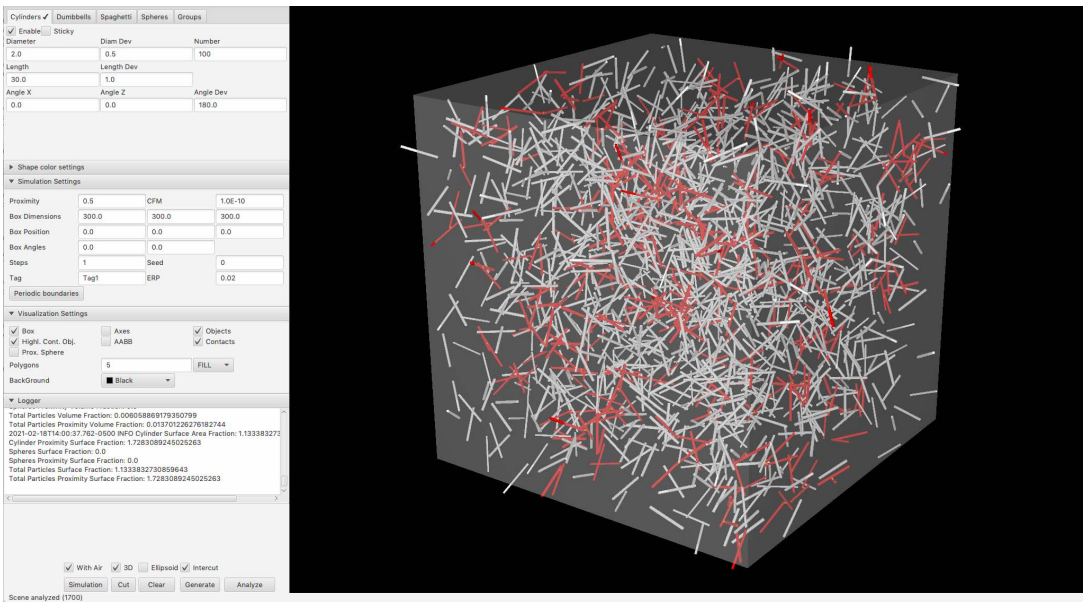


Figure 2.1: Snapshot of UI interface showing control panel on the left and sample 3D distribution of cylindrical rods on the right panel. Red-colored rods denote collision.

After generating virtual composites, Percolator's primary function is to estimate a con-

ductivity network as a set of pathways through various objects by precisely identifying surface contact points of collisions between them. Analysis is presented as a graph with node and edge attributes holding the coordinates of collision and the type of objects connecting two nodes. In Percolator, contact point estimation is performed via Java port of an Open Dynamic Engine for Java (ODE4j) 3D game engine [tza,]. With this framework's help, the first collision between two objects is identified via the axis-aligned bounding boxes method (AABB) [van den Bergen, 1997]. Since in the ODE4j engine, each object has a virtual bounding box. The intersection of the bounding box of one 3D object with that of another can be simply calculated with a logic comparison test. If a collision is identified as intersecting volumes, it will be further analyzed to locate the exact coordinates of the contact points on each object's surface. An example of contact points marked with red diamond objects is shown in Figures 2.3 and 2.4, representing a graph that can be further converted into a circuit.

The contact distance between conducting objects is typically small, at its lowest estimated value being on the order of a fraction of a nanometer due to van der Waals separation [Luo et al., 2014] . At its highest estimated value, this distance is limited by the electron tunneling range, constraining it to a few nanometers. In the simulator, to identify nearby contacting objects, each object is equipped with a proximity shell which is used for intersection analysis. Any two objects are considered to be in contact only when their proximity shells intersect.

Even a small system of objects can produce an exponentially large number of contact points. Therefore, a circuit with thousands of conducting elements needs to be analyzed to estimate conductivity. Simulation of such a large circuit's electrical response is complicated, especially in systems with passive, active, or reactive elements. SPICE (Simulation Program with Integrated Circuit Emphasis) simulation is proven to be the most effective approach in solving such problems. To the best of our knowledge, at the moment of writing, there are no commercial or open-source packages capable of tackling a problem of such magnitude in a scalable and automated manner. The absence of such capability led us to develop CircuitSymphony, a SPICE-like simulator with a graphic user interface and an API for remote communication [Frick and LaBean, 2021a].

The CircuitSymphony simulator is based on Paul Falstad's original Java code (`https://www.falstad.com/`) with major algorithms implemented by Pillage *et al.* [Pillage et al., 1994]. Besides a substantial code refactoring, we added a handful of modifications to the original code: enhanced support for various netlist formats, solving circuits with sparse solvers on CPU or dense solvers on both CPU and GPU, added a headless API communica-

tion layer, improved GUI for visualizing large circuits, and added support for containerization suitable for scalable, parallel simulations. CircuitSymphony can solve circuits comprised of hundreds of thousands of elements through adopting efficient single-threaded KLU [Davis and Natarajan, 2010] and multithreaded NICSLU [Chen et al., 2013b] direct sparse solvers.

Unlike other SPICE simulators, CircuitSymphony permits an on-the-fly modification, addition, and removal of circuit elements as long as the change does not result in a singular circuit matrix (such as open current sources or short circuit). At the moment of writing, CircuitSymphony's remote interface permits construction of circuits with bipolar passive or active elements. Thus, a circuit can be as well considered as a graph where components can be spontaneously added and removed without resubmitting the entire circuit definition file. To accomplish this, we implemented a token-based service layer that allows multiple clients to perform simulations isolated from other users. Each user can create, modify and destroy circuits where the service layer resolves resource allocation and administers tasks. This property is advantageous in simulating dynamic networks with randomly appearing and disappearing connections, such as in nanoscale composites.

In order to orchestrate communication between Percolator and CircuitSymphony, a ResSymphony Python library was created [Frick, 2021]. Through the Python language interface, a user can request Percolator to assemble and analyze the random distribution of objects with arbitrary geometries and perform one-step or transient conductivity analysis with the CircuitSymphony library. Besides interfacing functionality, the ResSymphony library also can estimate the shortest distances between conducting neighbors and construct conductivity graphs concerning user-specified collision distance parameters. More advanced functionality and the results obtained with this library will be further discussed in chapter 3.

Thus the simulator suite consists of three independent modules: 3D object generator and collision estimator along with the network connectivity analyzer (Percolator), [Frick and LaBean, 2021b], utility library with user-defined functions for circuit creation, and an interface to the cluster (ResSymphony) [Frick, 2021] and a circuit simulator (CircuitSymphony) [Frick and LaBean, 2021a] which are summarized in Figure 2.2. Given the distributed nature of the simulator components, a cluster can be constructed, where each node processes requests from an arbitrary number of users and from any programming language supporting REST API.

In a typical setup, the user provides a list of tasks that are fed into the cluster via the cluster manager and ResSymphony utilities. Each task carries specifications of the parameters

of objects that need to be generated via Percolator, such as length of wires, orientation, the geometry of the environment such as the size of the box, etc. If the task is set to compute conductivity, the resulting collision network graph will be also fed to CircuitSymphony for conductivity estimations. Once the cluster manger consumes the list of tasks, it distributes them appropriately across the computing nodes and waits until each step in the task is complete, after which it aggregates the results into user-specified format, saving the information to the database.

Since all server modules are designed to work in a distributed publisher-subscriber pattern, each module's performance is restricted by the resources of the individual cluster node. Thus, when the analyzer module builds a conductivity network, it occupies all CPUs on the dedicated node while staying isolated from other computing nodes' resources. Similar constraints are applied to network builder and network circuit simulator.

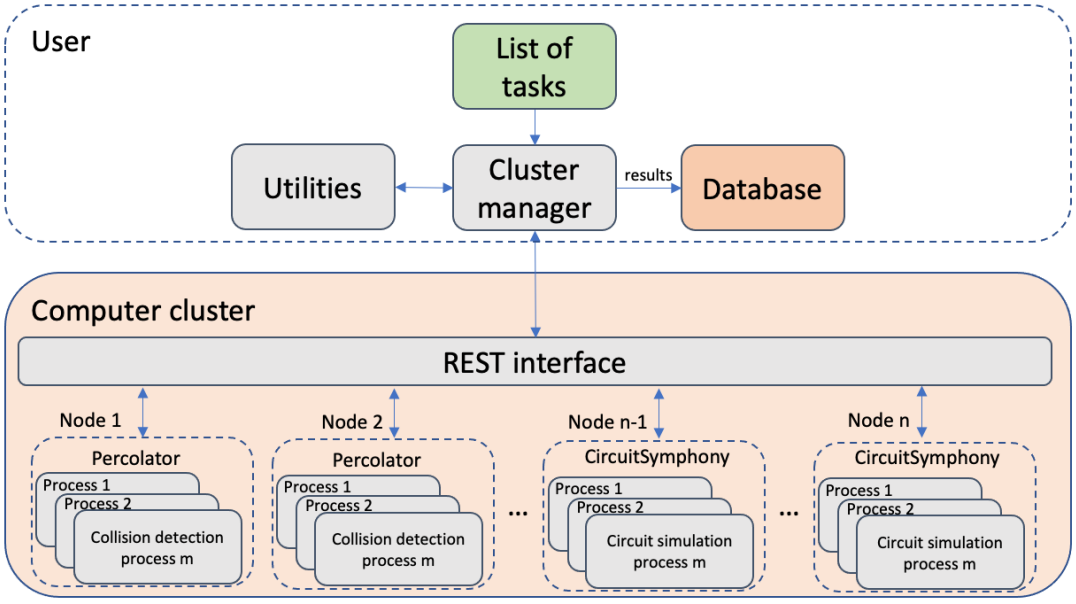


Figure 2.2: An example organization of the computing cluster for percolation conductivity studies with Percolator and CircuitSymphony in parallel mode. A user submits a list of tasks to the cluster manager, which controls the flow of the tasks and stores results in the database. The engine of the cluster manager also performs chaining of the tasks. An example of chaining is the instance when the result from the first computation step is converted into an input for the second computation step. For example, results from the collision calculation (Percolator) are converted into a circuit by the cluster manager via user-provided utility package and further simulated with the circuit simulator (Circuit Symphony). The results of each completed task are stored in the database.

By default objects in Percolator are permeable and can be well described by the "soft-core" percolation model [Balberg, 2020]. However, in real composites, filler particles don't interpenetrate which in the literature is referred to as the "hard-core" or "core-shell" model [Balberg, 2020]. For simulating this regime of percolation, Percolator includes a force-field assisted repulsion of the objects. This regime might cause performance degradation. As we will discuss further in section 2.3, the impermeability of the objects has only a minor contribution to the quality of percolation estimations.

2.3 Procedure for generating distributions of particles

This section describes procedures taken to generate particle volumes and distributions by continuum percolation. First, before generating particle distribution, a volume (3D) or area (2D) experiment is created with a characteristic side length L scaled with respect to the largest feature of the desired elements (i.e., components). The typical scaling factor is 5 for segmented or 10 for straight wires and depends on the overall number of objects generated in the system. We compared conductivities and percolation probabilities for periodic vs. nonperiodic boundaries. We found that the small square area with a 5xL side and periodic boundary has an indistinguishable magnitude of the conductivity from a square area with a 15xL side without periodic boundaries. We also found that there is no significant difference in conductivity for square areas above 10xL. We will discuss this in greater detail in section 2.9

Regardless of geometry, each filler type is represented by a sphere, 3D cylinder, or set of cylinders. The sphere and cylinder are the basis for the construction of other objects. For example, a high AR cylinder can represent a wire, whereas a low AR cylinder would represent a disk. In addition, assembling cylinders within a group can construct different shapes like a cube or a pyramid.

Fillers can be generated simultaneously for each area or volume at the beginning of the simulation or sequentially, for example, to simulate agglomeration. At sufficiently high concentrations of particles, some particles collide and even intersect. However, given the high AR of the fillers studied in this work, accounting for intersection is unnecessary [White et al., 2010].

There are two approaches in treating the volume of particles in continuum percolation, soft-core and core-shell. In the soft-core model, particles are allowed to interpenetrate [Bug

et al., 1985, Bug et al., 1986, Balberg et al., 1984]; thus percolation threshold is typically underestimated. In a core-shell model [Balberg and Binenbaum, 1987], particles cannot interpenetrate and are usually repelled to an equilibrium distance. The hard-core model is primarily used to model systems with prevailing tunneling conductivity [Berhan and Sastry, 2007] and also is useful to model emulsions and liquids [Bug et al., 1985]. However, in systems with mixed conductivity, according to White *et al.* [White et al., 2010], the soft-core model provides closer results to experimental data, if compared with more realistic core-shell models. Thus without losing accuracy, in the present work, we permit interpenetration of particles in both 2D and 3D experiments due to their relatively high AR.

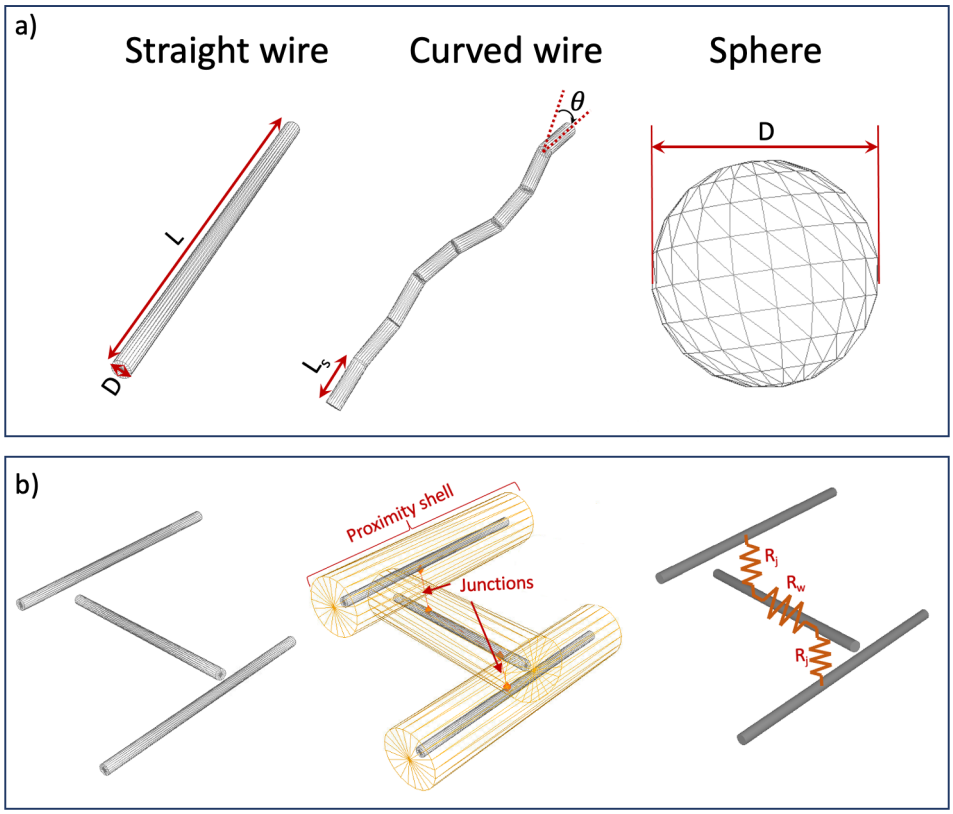


Figure 2.3: The types of 3D objects available in the Percolator and an example of assembling a circuit from conducting rods. a) Three kinds of objects with which Percolator. A straight rod (or wire) with length L and diameter D. A curved wire made of an arbitrary number of sequentially attached identical straight rods of length Ls, diameter D, and a maximum angle of attachment θ. A sphere of diameter D. Each object can also have a set of custom attributes or identifiers that can be used to construct structures from them. b) An example of constructing a circuit from identical straight conducting wires separated from each other. Each object in the Percolator can have a predetermined proximity shell that defines a permissible region of connectivity with other objects. The intersection of proximity shells can be treated as a junction resistance R_j, whereas the resistance of the wire R_w. There can be multiple types of connections that depend on the type of object and its attributes.

Thus, the circuit representation of a composite can be constructed from the shortest pathways between the contacts on the object's surface of the object and between the objects through the matrix. In this case, the matrix is represented either by the insulating medium between the wires such as epoxy or even polymer coating on the surface of the wire. A node in the graph and the circuit represents a contact between two objects. For example, a contact between a wire and a sphere objects. Matrix resistance is introduced in order

to simulate tunneling effects and can be changed by varying R_j. In particular when the shortest distance between two objects falls below a predefined tunneling cutoff distance value d_c an edge with two nodes is created. A typical value for d_c is set between 1.4 nm to 1.8 nm [Li et al., 2007, Bao et al., 2012] for ΔE taken between 1 to 5 eV [Hu et al., 2008a, Li et al., 2007] according to Eq. 2.2. As is shown in Figure 2.3 (b), the gap between objects is an edge with the distance d that can be constrained by $0 \leq d \leq d_c$. A more advanced model would assume van der Waals interaction distance and improve the existing model. For CNTs, this cutoff distance typically is 1.4 nm. In particular, there are two types of edges in the graph: the first type represents a straight pathway connecting objects through a polymer matrix, and the second type represents a conduction path through an object when it has more than one contacting neighbor, as is shown in Figure 2.4 (d).

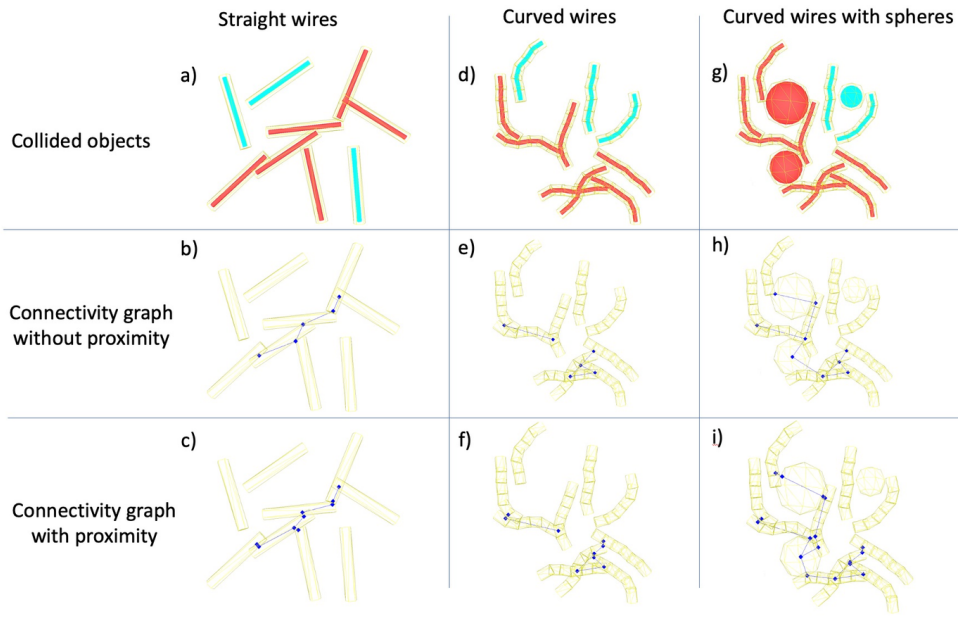


Figure 2.4: Procedure for estimating conductivity pathways between straight, curved and mixed types of colliding objects. a) collision between straight cylindrical wires in 2D with collided wires highlighted in red and b) resulting connectivity graph obtained from connecting points of contact (blue dots) between the cylinders' proximity shell (yellow cage) and shortest path (blue lines) between the contacts routed through the body of the cylinder without proximity gap (tunneling distance) and c) with tunneling distance d, e, f) collision analysis similar to a, b and c) but for curved wires and g, h, i) collision analysis similar to d, e, f) but for curved nanowires contacting spheres of different diameters where by convention all contacts on the surface of the sphere are routed through the center.

$$d_c = \frac{\hbar}{\sqrt{8 m_e \Delta E}} \tag{2.2}$$

For cylindrical objects, including disks, which contact several neighboring objects, the graph edges are constructed as the shortest path through all contacts. On the other hand, for spherical objects, all pathways with neighbor objects are routed through the sphere's center. In the contact graph, each node and an edge connecting these collision points contain additional attributes such as the coordinates of the node where the contact was identified and the property of the edge, which typically encodes the name and property of the conducting material.

To perform electrical analysis of the underlying system of connections, we assume that opposite sides of the surface for 2D or volume for a 3D environment where the objects are generated are covered with conducting electrodes. The resulting contact graph is analyzed with the Dulmage-Mendelsohn algorithm to identify a giant component graph that connects electrodes in the setup [Pothen et al., 1990]. The contact nodes within the electrode volume are tagged with a specific flag defining them as an electrode type node. Further, these nodes are interconnected with a bare wire to emulate the conductivity of a physical electrode.

In more complex instances, heterogenous geometries and structures must be accounted for. According to our observations, up to 80% of connections in such random networks are made of dangling, or free-ended edges (i.e., leaves) and loops, which do not contribute to the conductivity and can be removed to improve the efficiency of calculations significantly. Elimination of these components is especially beneficial in simulating random circuits with a large number of elements by optimizing the utilization of computational resources. To eliminate these dangling elements, we perform a two-step network preconditioning procedure during which only those elements that participate in charge transport remain. An algorithm iterates over all nodes during the first step and truncates leaf nodes with only one neighbor connection. However, this algorithm cannot remove dangling loops as each node in the loop has at least two neighbors. Truncation of dangling loops is achieved by applying a voltage to the electrodes and eliminating all edges with zero current.

2.4 Treatment of intersecting objects in 2D experiments

Intersecting objects add another layer of complexity to simulation data and require more in-depth consideration. Percolation experiments frequently exhibit inconsistencies with simulation data, which can arise from difficulties in calculating the actual volume or surface of intersecting particles. These discrepancies lead to incorrect estimations of percolation threshold ϕ_0. A typical origin of this difference may be the use of surface density instead of a surface fraction. In a typical 2D percolation experiment, the particles' overlapped area can be counted only once via a microscope or other projection-based technique. Thus, these techniques provide an estimation of the surface fraction. However, in the simulation setup, the overall surface area is equal to the sum of all section areas of each particle, as shown in Figure 2.5 (a) providing results in the units of surface density. Here we will show that for small concentrations of particles, this effect is negligible. However, the effect becomes more significant for large particle concentrations, Figure 2.5 (c), where the difference between actual and visible surface fractions increases with the increase of surface density of the nanowires. Thus particular attention needs to be paid to 2D distributions of short wires or rods as the increased percolation threshold will result in a more significant error in visible surface mass estimation. In 3D systems, on the other hand, in Figure 2.5 (d), the concentration of fillers is generally expressed as mass or volume fraction; thus, this analysis was not performed.

For 2D simulations, we perform a correction of the surface fraction magnitude for the area of wires intersection being counted only once. In particular, we calculated the surface fraction of the wires by estimating the average pixel intensity of digitally processed microphotographs of sliver nanowires distributions and comparing them to the images of simulated distributions with a known number of objects. The fitted relationship from this analysis is shown in Eq. 2.3.

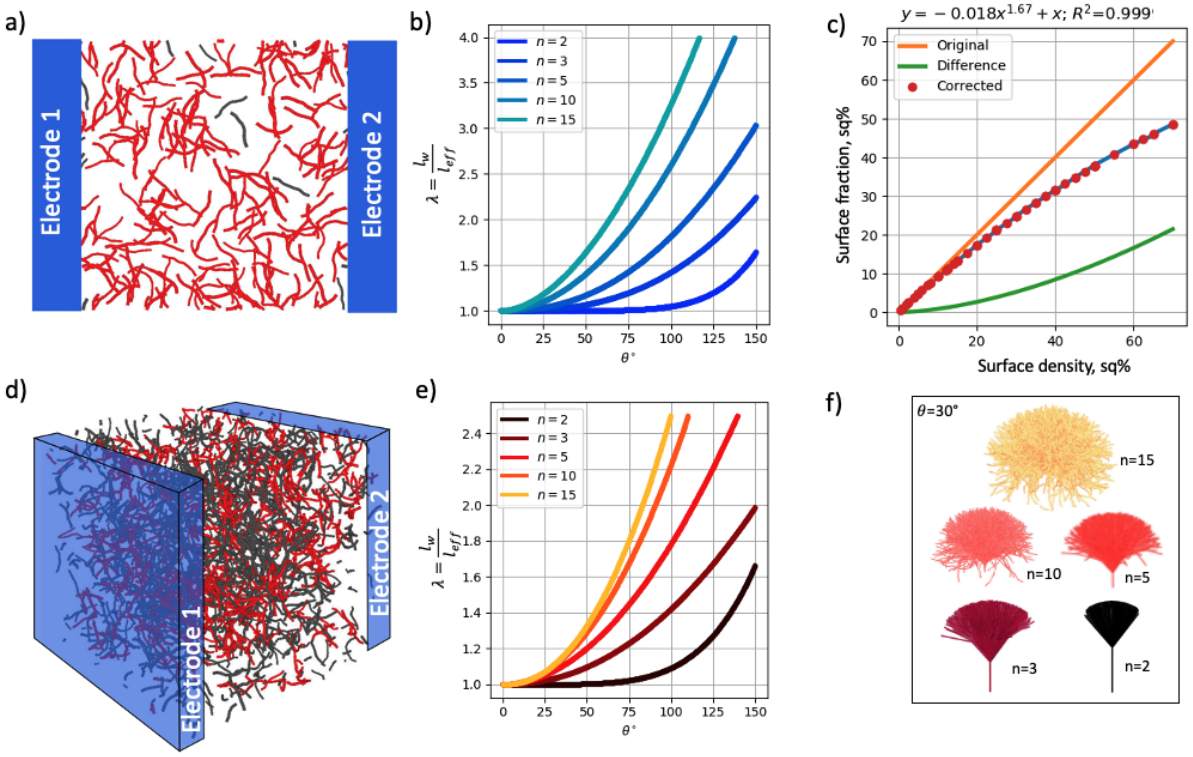


Figure 2.5: Estimation of waviness for curved wires (gray wires are isolated and red ones have contact with other wires). a) Sample 2D distributions of curved wires and b) waviness coefficient as a function of angle attachment. c) Surface fraction as a function of surface density where, orange line depicts the total surface fraction of generated nanowires without over-counting overlapping areas, red dots and blue line describes corrected surface fraction where the only area covered by nanowires is counted, green line depicts the difference between actual and corrected and suggests large deviations at high surface fractions. d) Sample 3D distribution of curved nanowires (gray wires are isolated and red ones have contact with other wires) and e) waviness coefficient λ as a function of attachment angle θ where the number of segments was fitted with $\lambda(\theta) = a\theta^b + 1$ relation that implies that in a constrained random walk approximation of a curved nanowire, waviness correlates with the number of segments in the wire. f) Visualization of the relationship between waviness and the number of segments n in a non-straight wire with a maximum angle of attachment between the segments $\theta \leq 30°$.

$$y = -0.018x^{1.67} + x \tag{2.3}$$

In this work, all 2D computational experiments have corrected surface fractions according to Eq. 2.3. In particular, in each generated random distribution of nanowires, the surface density is converted with Eq. 2.3. where x is surface density and y is the resulting surface

fraction. As a result fillers with low AR have a high percolation threshold ϕ_0 and are subject to the area fraction overestimation.

2.5 Periodic boundary

In the literature, boundaries are inconsistently defined and considered for analysis, leading to discrepancies between studies. Many computational percolation studies do not mention whether boundaries of the simulated volumes or areas are considered [Ni et al., 2018, Ni et al., 2019, Takeda et al., 2011, Park et al., 2019, Román et al., 2018]. In many cases, periodic boundaries (PB) do not play a significant role in percolation calculations, especially when the overall system's size is relatively large compared to the size of individual elements. However, computational resources required to simulate percolation systems are proportional to the problem's dimensionality and might be prohibitively large for simulating bulk systems.

In this research, we adopted periodic boundary conditions similar to Bao *et al.* [Bao et al., 2013] for most of the systems in 3D and 2D geometries and these PB conditions are shown in Figure 5. For single segment straight wires, the periodic boundary is applied in the following manner: wires intersecting the boundary of the area or volume are split into two parts. The portion outside of the boundary is translated to the opposite inner side of the region. On the other hand, segmented wires are split at the nearest junction location close to the intersection of the segments with the boundary, in which case the length of each segment stays intact.

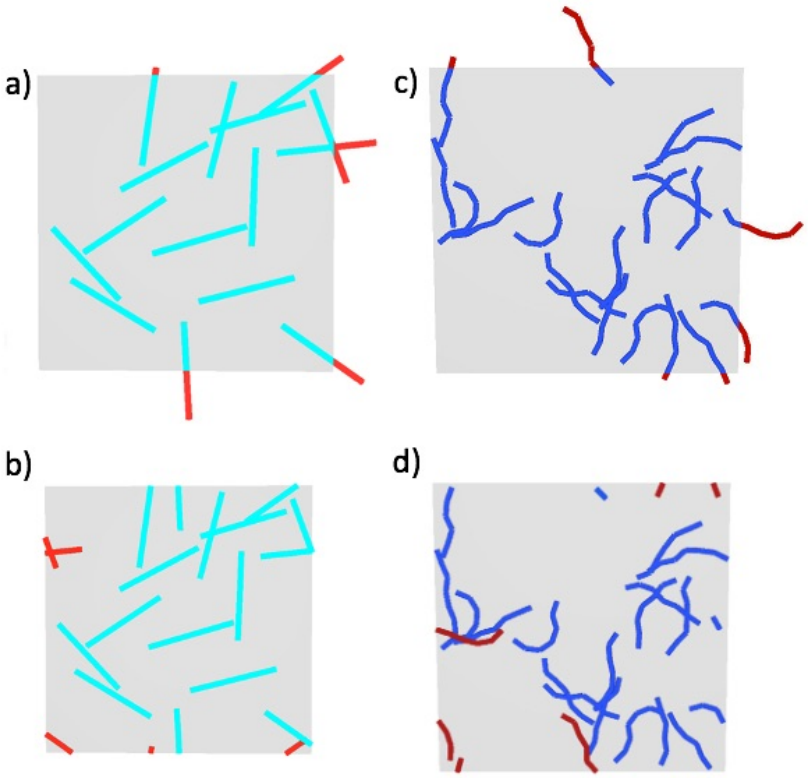


Figure 2.6: Periodic boundary conditions. An example of the application of PB to objects with a single segment (straight) and multisegment (curved) objects a) before PB and b) after PB was applied. The portion of cylinders that go outside the square area are painted red and are transferred to the opposite side of the area. c) and d) illustrate the same for curved wires.

In this work, periodic boundary conditions are applied to find an optimal compromise between the required computational resources and the highest accuracy in each simulation. According to our estimation shown in Figure 2.7, the size of the side of the square area in 2D or box volume in 3D in percolation experiments can be between 5 to 10xthe length of the longest feature of the filler. It was also estimated, that when when the box's side is 10x or more of the longest filler feature, the standard error of conductivity estimation without periodic boundary is equivalent to the standard error obtained with periodic boundaries. In some studies of distributed particles or agglomerated segregated structures, the volume of the environment smaller than 5x of the largest feature can be used but only if periodic boundary conditions are applied [Bao et al., 2013, Hu et al., 2008b].

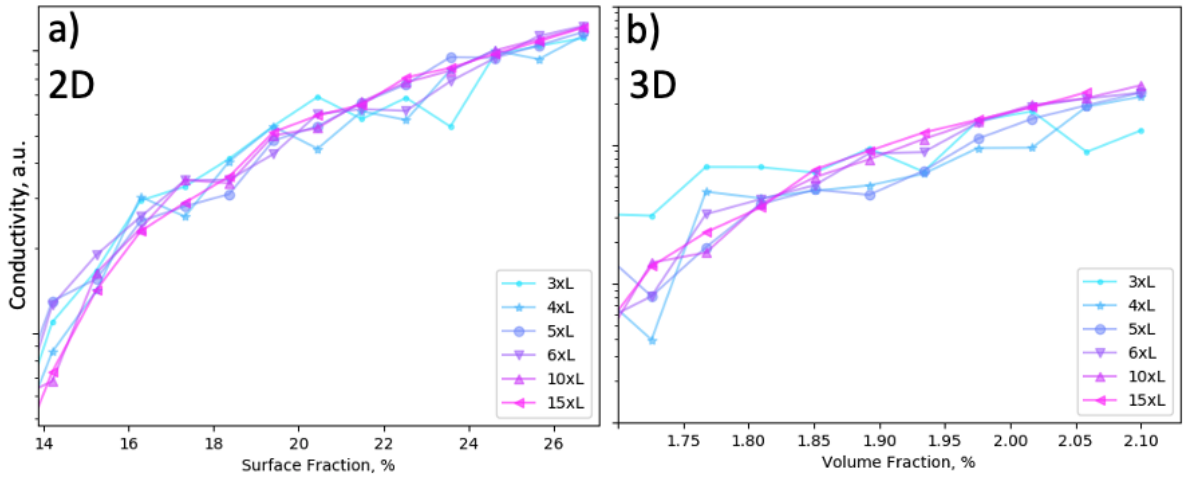


Figure 2.7: The effect of application of periodic boundary condition on the variability of conductivity of randomly distributed wires a) on a square surface and b) in a cube volume which size scaled proportionally to the length of the wire. In the graphs, each point is averaged from 50 distinct random distributions of wires with length L=2 and diameter D=0.1 all in a.u. It is clear that above 10x scale the variability of conductivity becomes indistinguishable from experiments with smaller scaling in both 2D and 3D.

2.6 Power law in percolating systems

The goal of this section is to demonstrate the validity of our simulation method to analyze percolation phenomena in complex composites. In laboratory experiments we are typically interested in conducting composites in their supercritical regime for which the concentration of conducting fillers is above particular threshold value ϕ_0. This threshold value depends on the geometry of the fillers, their microscopic organization, and the system's geometry at the macro scale. Several experimental and theoretical studies of supercritical phenomena in percolating nanorod distributions show that most percolating quantities such as electrical and thermal conductivity obey the power law scaling above the percolation threshold ϕ_0 as is shown in Eq. 2.4 which was first proposed by Kirkpatrick *et al.* [Kirkpatrick, 1973]. In particular Eq 2.4 relates conductivity to the concentration of the fillers and also microscopic properties of them via exponent t. The conductivity σ is the measured conductivity of the system, σ_0 is the constant related to the conductivity of the fillers [Weber and Kamal, 1993], although the real meaning of this term is still under dispute. The difference between the actual concentration ϕ and percolation threshold ϕ_0 of the

fillers are raised to the power of t. The exponent t is a complex function that depends on multiple factors like waviness, effective volume fraction, and other parameters depending on the experiment's nature [Zare and Rhee, 2020]. The scaling exponent t was assumed universal for a fixed geometry and independent from the microscopic structure but, vary with the dimensionality of the system [Sahini and Sahimi, 1994]. However, recent studies suggest that the exponent t is related to microscopic and macroscopic properties of the composite and includes a unitless network efficiency factor, introduced by Zare *et al.* [Zare and Rhee, 2020]. This complex dependency of t on the micro scale is also supported by our simulations that show that there is a dependency of the exponent t on the resistivity of the matrix.

Thus, in a first approximation, an adequate assessment of the quality of computational results can be made by comparing the fitted magnitude of exponent factor t in Eq 2.4 to our laboratory experiments with similar geometry of the wires and identical results obtained by other groups. In addition to that, power law fitting allows us to make other useful observations about the system's geometry, which will be discussed further in section 2.8.

However, the applicability of the power law to percolating systems is controversial, which has led to the proposal of other, non-power law, models [Clingerman et al., 2001, Li and Östling, 2015]. One of the biggest disputes highlights that the power law scaling does not hold for the whole region of filler concentrations and is invalid only below critical concentration ϕ_0 [Li and Östling, 2015].

$$\sigma = \sigma_0 \left(\phi - \phi_0\right)^t \tag{2.4}$$

In this work, power law fittings were made for the conductivity for simulated random distributions of the wires in 2D and 3D environments with the Percolator and CircuitSymphony software. In particular, in our simulation approach, we varied waviness (as an angle of segment attachment), junction resistance, and AR of the wires. In order to make each experiment equivalent, the concentrations for each case of waviness and AR were chosen so that the percolation profile is symmetric in all cases and centered around ϕ_0 (i.e. to make measurements for all experiments equivalent). Further, a set of 60 unique distributions of nanowires for each value of concentration, AR, waviness, and R_j was generated and the resulting network connectivity was analyzed. Finally, based on the obtained network in each particular experiment and its connectivity analysis, the conductivity and other auxiliary parameters such as mean shortest path (discussed further) were estimated in a 2

electrode setup. Obtained values were saved in a database with each row dedicated to a particular simulation and a column dedicated to the parameters and metadata describing how the experiment was generated for reproducibility purposes.

In addition to conductivity, we also studied the average short path between opposing electrodes. This profile provides useful insights regarding the dimensionality of the system and the microarchitecture of the network which will be discussed further in the results section 2.8.

2.7 Simulation procedure and parameters used in this work

Many parameters must be considered for each simulation in order to gather accurate data. Below, we summarize the parameters used to generate and analyze our simulated systems of percolating networks in Table 2.1. Wire length L and diameter D were chosen to reflect the actual dimensions of Ag nanowires produced in our laboratory. Waviness angle θ is chosen to cover straight for $\theta = 0°$ and severely curved wires when $\theta = 90°$. The number of wire segments n_s defines the number of cylindrical components which will be sequentially attached to each other constrained by an imaginary cone of a fixed angle θ coaxial to the parent wire segment. In a sense, this procedure of constructing a pathway from segments is referred to as a correlated random walk and has been used for insect movement analysis [Kareiva and Shigesada, 1983]. Without losing accuracy, the size of the square area or cubic volume side's size varies from 10xL for 2D to 5xL for 3D cases. For example, in a 2D experiment with the length of the wire where L=10 µm, the square area's side will be equal to A=100x100 μm^2 whereas for the 3D case, the volume of the box V=50x50x50 μm^3. Intrinsic resistivity $\rho_w = 5\ \frac{\Omega}{\mu m}$ is the magnitude of the resistance of a straight segment between two nearest contact points on the material. It is chosen for silver concerning a single wire measurement, discussed in detail in chapter 4 [Selzer et al., 2016]. Thus for a wire with L=10 µm and contacts on both ends, the resulting resistance is $R_w = 50\ \Omega$. The contact resistance R_j is also chosen according to the values obtained in our laboratory in a single measurement between two overlapping wires. Finally, the contact cutoff distance d_c is determined to be equal to 1.4 nm according to the tunneling distance considerations mentioned in the previous section and Equation 2.2.

Parameter	**Symbol**	**Values**
Length of the wire, μm	L	2, 4, 10, 20, 50
Wire diameter, nm	D	100
Number of wire segments	n_s	1 (2D and 3D), 5 (3D), 10 (2D)
Segment attachment angle 2D	θ	0°, 5°, 15°, 30°, 60°, 90°
Segment attachment angle 3D	θ	0°, 12°, 24°, 55°, 92°, 150°
Box or square side size, μm	b	10xL (2D) and 5xL (3D)
Linear wire resistivity, ($\frac{\Omega}{\mu m}$)	ρ_w	5
Junction resistance, (Ω)	R_j	10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3, 10^4, 10^5
Contact cutoff distance, (nm)	d_c	1.4

Table 2.1: Input parameters used in the simulations for both 2D and 3D models.

Each simulation was performed within different surface and volume fractions which depend on the length of the wire. These values are summarized in Table 2.2.

	2D		3D	
L, [μm]	**Sf_{min},** [%]	**Sf_{max},** [%]	**Vf_{min},** [%]	**Vf_{max},** [%]
2	0.110	0.420	0.0272	0.0453
4	0.069	0.222	0.0131	0.0210
10	0.033	0.089	0.0051	0.0074
20	0.015	0.047	0.0024	0.0035
50	0.006	0.020	0.0009	0.0013

Table 2.2: Surface and volume fractions, [%] used in 2D and 3D experiments.

To produce results with high statistical significance > 95%, each simulation setup is comprised of a linear range of 50 points between $\phi_{\min}$ and $\phi_{\max}$ of the surface fractions in 2D experiments and volume fractions for 3D experiments within which percolation threshold is contained and centered concerning Eqn 2.5 in which the parameter K satisfied $0.25 \leq K \leq 0.5$. For each surface or volume fraction point, 60 distinct distributions of particles were generated with different seed values magnitudes while keeping other parameters from Table 2.2 fixed.

$$\log(\frac{\phi_{max}}{\phi_0}) = \log(\frac{\phi_0}{\phi_{min}}) = K \tag{2.5}$$

Thus for given input parameters, multiple particle distributions of a system with the same parameters but a different placement of particles are generated. However, during the analysis, some objects' distributions might produce ill-defined circuits (e.g., negative conductance) or circuits with singular matrices. Such circuits are discarded from the study and replaced with a well-posed newly generated distribution.

2.8 Straight and curved nanowires in 2D

This section discusses simulation results for 2D and 3D distributions of Ag nanowires. First, 2D systems were simulated and analyzed. In these systems, straight and curved wires were generated on a square surface with sides b scaled with 10x of the lengths L of the wire with applied periodic boundary conditions. We generated wires with five different values for each variable-length L, contact resistance R_j and waviness expressed by θ according to the parameters in Table 2.1 while keeping the diameter of the wires constant D=0.1 μm. The number of segments for each wire was set to n=1 in all cases when $\theta = 0°$ and n=10 for any $\theta > 0°$.

For each sample distribution of wires, collision points were estimated, and the representative circuit was constructed and comprised of the resistors R_w representing intrinsic resistance of the material's body (in this case Ag) and R_j representing junctions between any two distinct objects. We adopted two different strategies in treating junction resistance between two objects. In the first approach, each segment of the wire between two collision points was complemented with an additional value of the junction resistance so that that the total resistance of the wires' segment R_s had the relationship $R_s = R_w + R_j$, thus making the resulting circuit network consist only of resistive wire's segments. In the second approach, a new element with junction resistance R_j was added between two colliding objects which makes the resulting circuit graph having two independent elements, one related to intrinsic conductivity of the wire R_w and another related only to the resistance of the junction R_j.

In the experiments with variable junction resistance, instead of generating a new circuit for each R_j, multiple copies of the circuit were created with different junction resistance values, while other parameters were left unchanged.

As shown above in Figure 2.5 (b), the waviness λ of a cylindrically shaped wire in 2D is

related to the angle of attachment of the segments and the total number of segments in the wire. In this work for a 2D case, $n = 10$ and $\theta = 90°$ which yields $\lambda_{\theta=90°} = 2.5$.

Electrical characterization of 2D samples of straight wire networks of different lengths and junction resistances is shown in Figure 2.8 (a), where simulated conductivity is presented as a function of surface fraction for different junction resistances R_j (for straight wires). In the figure for each sample of the network with distinct straight wire lengths, the junction resistance R_j was varied between 0 10^{-5} Ω to 10^5 Ω, where in the analysis the value of R_j was varied with one order of the magnitude step. For each sample the conductivity was calculated through stimulation of the network circuit with an input DC voltage difference of U=1 V across the opposite edges of the square surface. After recording the current, for each sample, the surface conductivity $[S]$ was obtained. For convenience, each sample set of points was fitted with a power law, Eq. (2.4), and is depicted with a solid line that extends for an additional length for increased visibility.

As can be further inferred from Figure 2.8 (a), increasing the wires' length pushes the percolation threshold (vertical dash line) to smaller surface fractions. Also, high values of the junction resistance R_j set the bottom limit for the conductivity at the percolation threshold for all lengths of the wires. Thus for $R_j = 10^5$ Ω, all conductivity lines start at around 3x10^{-7} S. In contrast, at lower values of junction resistance $R_j = 10^{-5}$ Ω the conductivity at the percolation threshold for wires with different lengths has a much wider gap. This difference is showed in a greater detail in Figure 2.10 (b) as conductivity at the percolation threshold converge to a single point ≈ 3x10^{-7} S. This difference in the values at percolating threshold are caused by two different conductivity regimes, namely, junction resistance dominated conductivity and wire resistance dominated conductivity regimes. In the junction resistance dominated regime, the conductivity of the network is defined by the junction resistances that are much larger than the resistance of the wires. Thus at high junction resistances, the resulting conductivity of the whole network is not influenced by the length of the wires as much as it depends on the junction resistance. Oppositely, in a wire resistance-dominated regime for which $R_j \approx 0$ Ω, the conductivity of the network can be noticed to change primarily as a function of the length of the wires.

Figure 2.8 (b) shows the dependency of conductivity on the attachment angle θ for curved wires ($n = 10$ segments) of various lengths and zero junction resistance ($R_j = 0$ Ω). With the waviness taking one of three values $\theta = (0°,\ 60°,\ 90°)$ the conductivity pattern also follows power scale law. From the figure, it can be observed that while the percolation threshold depends on the value of the attachment angle, the conductivity at the percolation threshold does not depend on waviness. This suggests that conductivity at the percolation

threshold is primarily a function of junction and wire resistances R_j and R_w.

In Figure 2.8 (a), we notice that the magnitude of the exponent t slightly oscillates as the length of the wire is changed. In fact, the noise depends on the junction resistance R_j, waviness or attachment angle θ , and less on the AR of the wire. The dependency on R_j is quite noticeable as in all cases, the fitted exponent's magnitude decreases with R_j becoming greater than 0. However, the trend reverses for the waviness as the exponent tends to increase along with the increase of waviness. This difference is slightly larger when R_j is greater than zero.

Like straight wires, the fitted exponent's magnitude decreases with increasing R_j and becomes even more pronounced with a decrease in waviness. Wires with the highest R_j exhibit the lowest conductivity at the lowest surface fraction, whereas low AR wires have the highest conductivity and require a higher surface fraction to conduct.

In order to show the influence of the junction resistance on the power law conductivity model Eq. (2.4), the fittings to the conductivity curves for straight wires in Figure 2.8 (a) are displayed in Figure 2.9 (a).

Surprisingly increasing junction resistance between conductors tends to increase the exponent of the power law fittings from approximately 1.2 at low R_j to below 1 at high R_j, as is shown in Figure 2.9 (a). These results are similar to previously reported results for power law modeling of conductivities of networks formed by idealized thin wires with constant wire resistance [Hicks et al., 2018, Li and Östling, 2015, Keblinski and Cleri, 2004, Forró et al., 2018]. In these works, the magnitude of the exponent exhibited an increase from nearly 1 to approximately 2 when the magnitude of the junction resistance was increased relative to the resistance of the wires. In particular, in our simulation experiments, the wires are with true volume and each contact is located on the surface of the wire which slightly increases the length of the line connecting two contact points making the resistance of the wire slightly larger than a straight coaxial line.

However, this phenomenon needs to be studied in greater detail since we did not observe similar patterns in a 3D system, which will be discussed further in section 2.9. Also, for the exponent t, derived fitted exponent values for straight wires with zero junction resistance correlate well with the literature's values that show 1.1-1.3 for the 2D case [Li and Zhang, 2010, Stauffer and Aharony, 1985].

The conductivity as a function of surface fraction for different values of attachment angle θ in Figure 2.8 (b) was obtained similar to Figure 2.8 (a) except the junction resistance R_j was set to zero and the displayed curves had the attachment angle θ which, for enhanced visibility, was set to 0°, 30° and 60°.

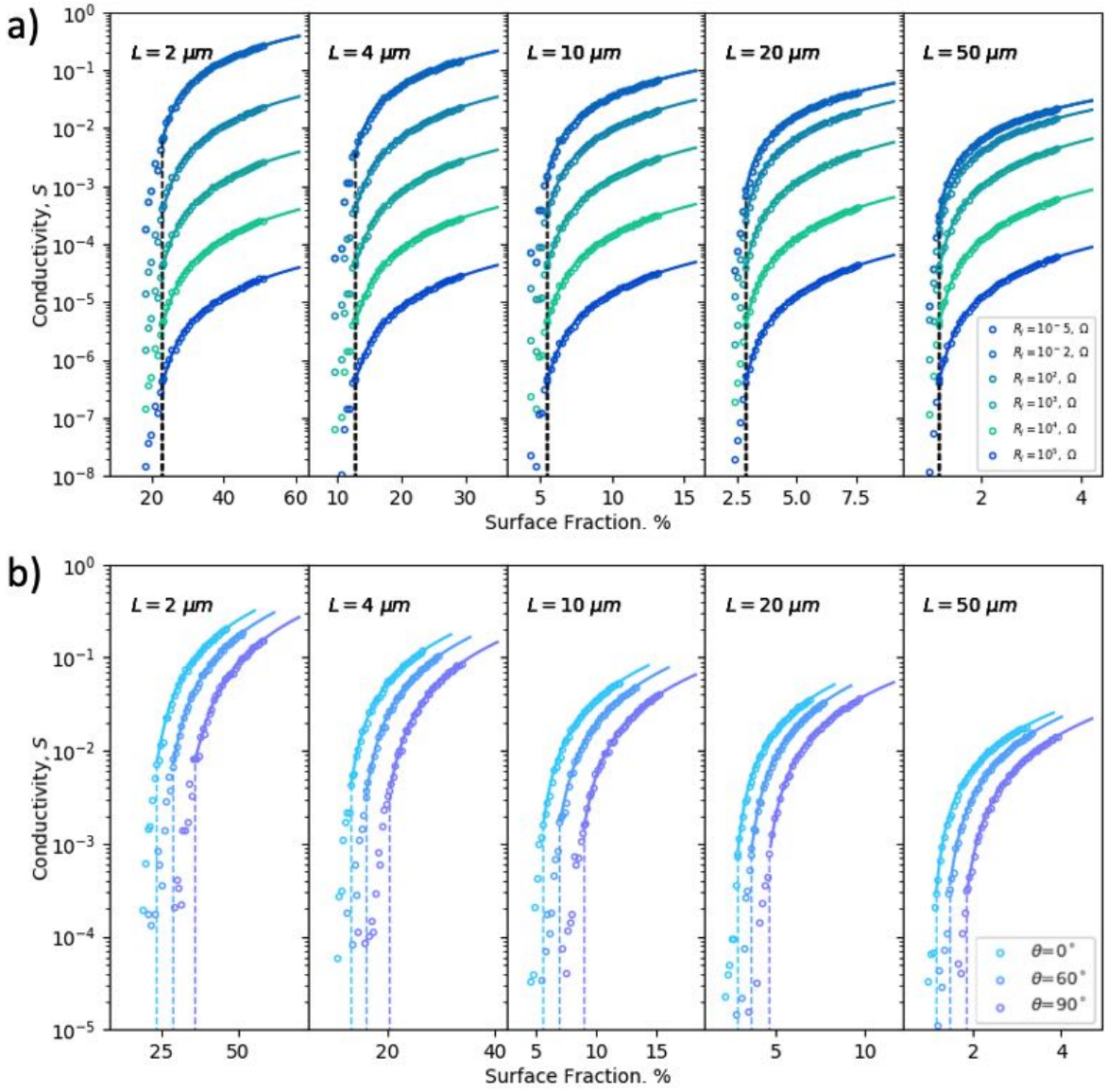


Figure 2.8: Conductivity as a function of surface fraction for different junction resistances R_j (for straight wires) (a) and (b) attachment angles θ for $R_j = 0$ and the number of segments $n = 10$. In all cases, the side of the square area on which the network was generated was scaled to 10xL μm for which the diameter of all wires was held constant D=0.1 μm. In all subplots fitted power law lines (solid color) are extended to an additional 20% length beyond the simulated data for improved visibility. Dotted vertical lines denote percolation thresholds at the 50% percolation probability. In (a) there is only one percolation threshold line for different values of junction resistances R_j as the network architecture remains the same and only R_j magnitude is being replaced. On the contrary, in (b), the percolation threshold increases with the increase of the attachment angle θ, also denoted by vertical dashed lines.

The exponents of the fitted power law to conductivity curves in Figure 2.8 (b) are also functions of the angle of attachment θ in Figure 2.9 (b). For long wires, in particular L=50 μm and L=20 μm the increase of the angle of attachment increases the exponent from t=1.2 for straight wires to up to t=1.35 for curved wires with $\theta = 90°$. As can be noted from the figure, smaller wires suffer from a larger noise impact, which seems to be inversely proportional to the length of the wire.

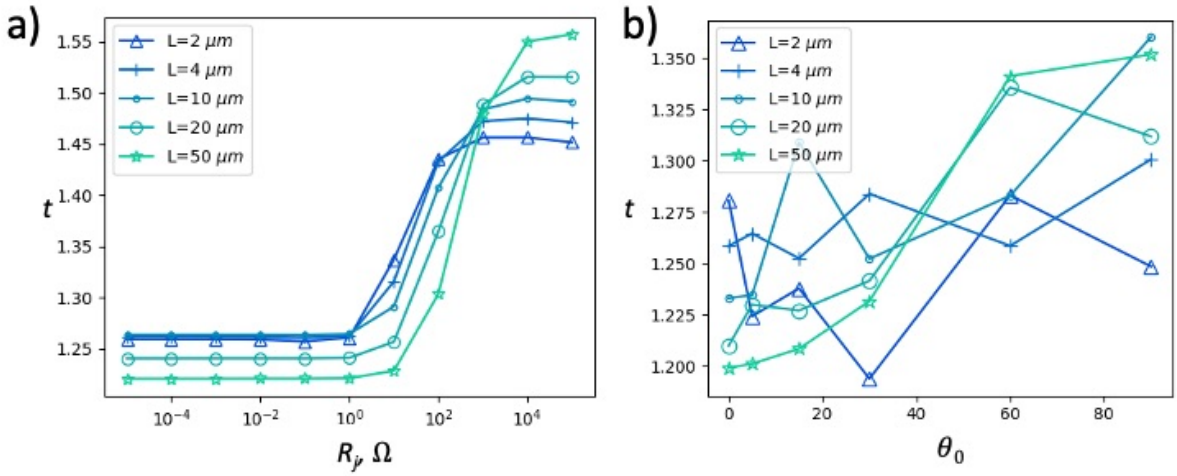


Figure 2.9: Scatter plot matrix for 2D wire distributions summarizing the influence of variable parameters L, θ, and R_j on the magnitude of the power law exponent t, where the color of each point represents the magnitude of t. This plot shows the positive correlation of the exponent t with θ and the negative correlation with L and R_j.

In a laboratory sample preparation, it was frequently desired to estimate conductivity of the composite right in the critical conductivity regime at the percolation threshold since conductivity at this point is expected to possess the most interesting behavior due to the sparse connection of the fillers. The percolation threshold's conductivity can be estimated via percolation simulation, assuming that wires are well distributed and deagglomerated. Figure 2.10 (a) relates surface fraction at percolation threshold as a function of wire waviness (or angle of wire segment attachment) for different wires' lengths. The surface fraction at the percolation threshold has an exponential dependency on the length of the wires. It increases rapidly with the increase of the curl ratio θ following the pattern described by Eq 2.5 and can be noticed in Figure 2.8 (b). Since the percolation threshold doesn't depend

on the junction resistance, all data in the plots are taken with $R_j = 0\,\Omega$.

Figure 2.10 (b) shows conductivities at percolation threshold for different wire lengths L and junction resistances R_j. Thus, as expected, the increase of junction resistance R_j between fillers decreases the conductivity of the composite. However, regardless of the length of the wire, all values converge almost to the same values of around 3×10^{-7} S.

Our applied analysis of the percolation phenomena in 2D composites would not be complete without summarizing the dependency of the surface fraction at the threshold as a function of attachment angle θ and wire length L which is summarized in Figure 2.10 (c). The relationship shows that the primary direction of growth of surface fraction at the percolation threshold is when both wire length is decreasing and the attachment angle is increasing, with the highest peak situated in the upper left corner. On the contrary increase in the length of the wire and a decrease in the attachment, angle causes a decrease of the percolation threshold with the lowest point being in the right bottom corner. While this piece of information seems trivial, it reinforces our computational approach in analyzing percolating systems with the developed software.

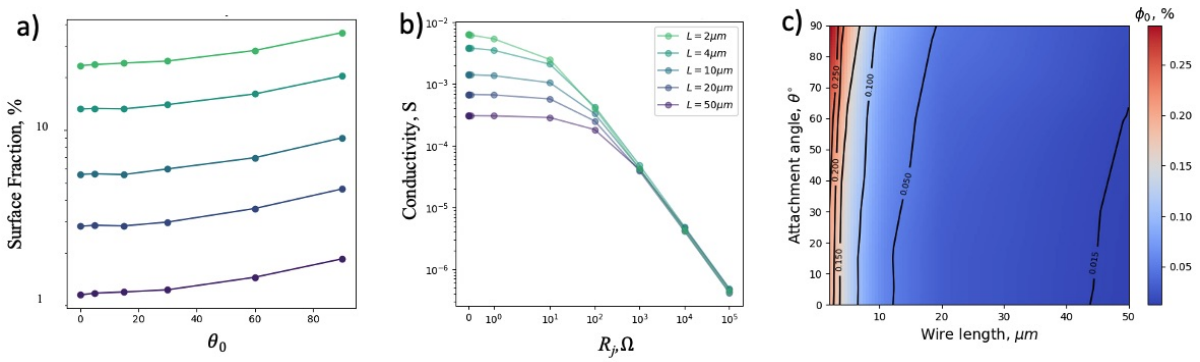


Figure 2.10: a) Surface fraction [sq %] at the percolation threshold for different wire lengths as a function of the segment attachment angle θ. b) Conductivity at percolation threshold as a function of R_j. c) Surface fraction at percolation threshold ϕ_0 as a function of attachment angle θ and wire length L.

We also studied the mean shortest path (number of segments) between opposite electrodes as a function of the surface fraction shown at the bottom of Figure 2.11. Here

shortest path describes the smallest number of segment paths through the network that connects two opposing electrodes. In a sense, this is a simplified approximation of the betweenness centrality of the network but just between two opposite electrodes [Freeman, 1977]. The obtained curves can be thought to consist of two portions. The first one followed a sigmoidal shape resembling a scaled percolation threshold pattern with the exact percolation threshold point situated in the middle of the curve and marked with a dashed vertical line. The second component starts at the top of the sigmoidal curve's end and spawns multiple straight lines, each having a different slope that depends on the angle of attachment θ. Essentially these curves show that the shortest path between two electrodes in a conducting 2D network grows explosively fast until it reaches the critical point (at the top of the first segment), after which the growth is slightly decreased and reaches a plateau for highly curved wires ($60° < \theta < 90°$), or diverges for straight wires ($\theta < 60°$). Highly curved wires behave more like particles. An increase in the number of the particles does not guarantee neither an increase nor a decrease in the number of segments connecting opposing electrodes. Therefore, we see plateau patterns in these cases. On the contrary, the divergence of the shortest path line in the cured wire sample results from the exponential increase in the number of contacts, caused by overlapping wires and causing an increase in the number of segments connecting opposite electrodes.

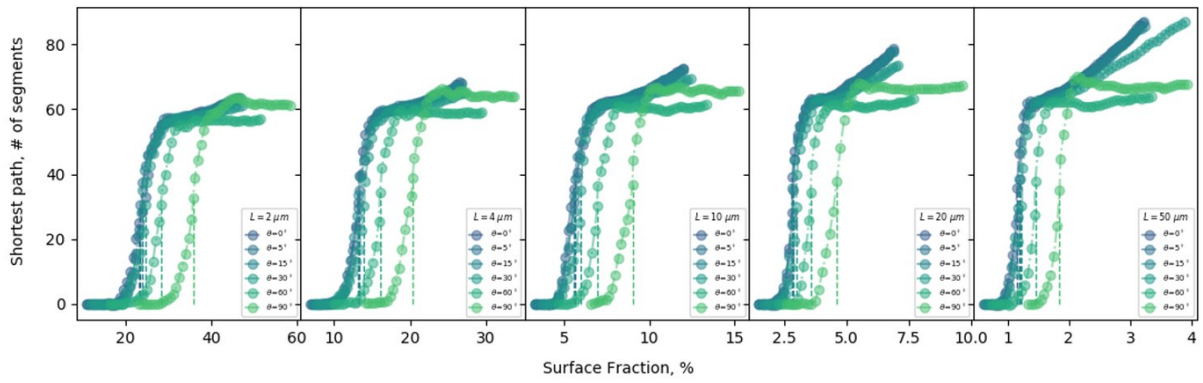


Figure 2.11: Conductivity at percolation threshold for different R_j as a function of waviness. Bottom: Mean shortest path (number of segments) between opposing electrodes. Vertical lines denote concentration at the percolation threshold.

It is interesting that in all cases, the shape of the first portion of the curve is similar,

for all lengths of wires, whereas the rate of change in the second portion of the graph is more pronounced for straighter (low value of θ). A straight wire can connect to a larger number of neighbors, and thus we can see that the rate of growth of the mean shortest path is much larger for longer wires. This means that straight wires are more prone to increasing the number of junctions than shorter wires and the high value of the attachment angleθ boosts this influence.

In the next section, we will return to these results and compare them to nanowires' 3D distributions.

2.9 Straight and curved nanowire distributions in 3D

After thoroughly considering 2D circumstances, we delved into the modeling of Ag NWs in 3D environments. Studying 3D distributions of nanowires is complicated by a much larger number of collision events, which need to be analyzed compared to 2D. As a result, the demand for computational resources increases dramatically, thus before simulation, the system's size needs to be reduced. However, in our treatment here we discuss a few approaches with such systems that can be analyzed without loss of precision.

An example of a 3D distribution of curved nanowires is shown in Figure 2.5 (d). A typical pitfall is the visual perception of percolation in 3D due to overlapping particles in the 2D projection creating a false illusion of connectivity. Despite a visual perception of percolation, in reality, the system can be quite far away from being conductive.

Figure 2.5 relates the degree of waviness (the angle of attachment θ) to the maximum angle of segment attachment for curved nanowires with a varying number of segments, which are further depicted in Figure 2.5 (f). Unlike in the 2D case, a wire with $n = 10$ segments and $\theta = 90°$ in the 3D case yields a lower degree of curl ratio from Eq. 2.1, $\lambda_{\theta=90°} \approx 1.7$.

As will be explained in more detail later in this section, conductivities of distributions of curved nanowires with a large number of segments can be successfully approximated with nanowires with a smaller number of components but the larger magnitude of the maximum angle of attachment. In other words, waviness for a wire with 15 segments at $\theta_{max} = 75°$ is functionally equivalent to a wire with five segments at $\theta_{max} = 100°$ which can be inferred from Figure 2.5 (e) and Figure 2.12. This substitution increases computational efficiency without penalty altering the resulting calculated conductivities and percolation thresholds. As shown in Figure 2.12, there is an indistinguishable overlap between θ=60

at n=10 and θ=114 at n=3 which suggests that a smaller number of segments can represent wires with continuous waviness.

With 3D percolation, our goal is to provide a framework for simulating conductivity in bulk materials. The total amount of constituents needed to meet the percolation threshold grows proportionally with the number of degrees of freedom. Thus for fillers with AR=20, where for example L=2 μm, D=0.1 μm and area side S=Lx10 μm^2 in 2D, the percolation threshold will be reached with approximately 20000 wires; however, for similar conditions in 3D, it will require around 200,000 wires. In addition, curved nanowires need to be segmented into smaller pieces to represent desired curl ratio λ, which contributes to the increase of the total number of 3D objects in the volume that require collision analysis.

In this work, our goal is to analyze large 3D volumes filled with nanowires using fewer computational resources by reducing the number of wire segments. We will set the upper maximum for waviness to λ_{max}= 2.5 and the number of segments n=10 and the angle of attachment $\theta = 90°$. To reduce the number of elements needed to simulate such a system without performance loss, we can reduce the number of segments in the wire. Thus a wire with n=5 and $\theta = 130°$ angle of segment attachment is equivalent to a wire with n=2 number of segments and $\theta = 180°$ angles of attachment as is shown in Figure 2.12.

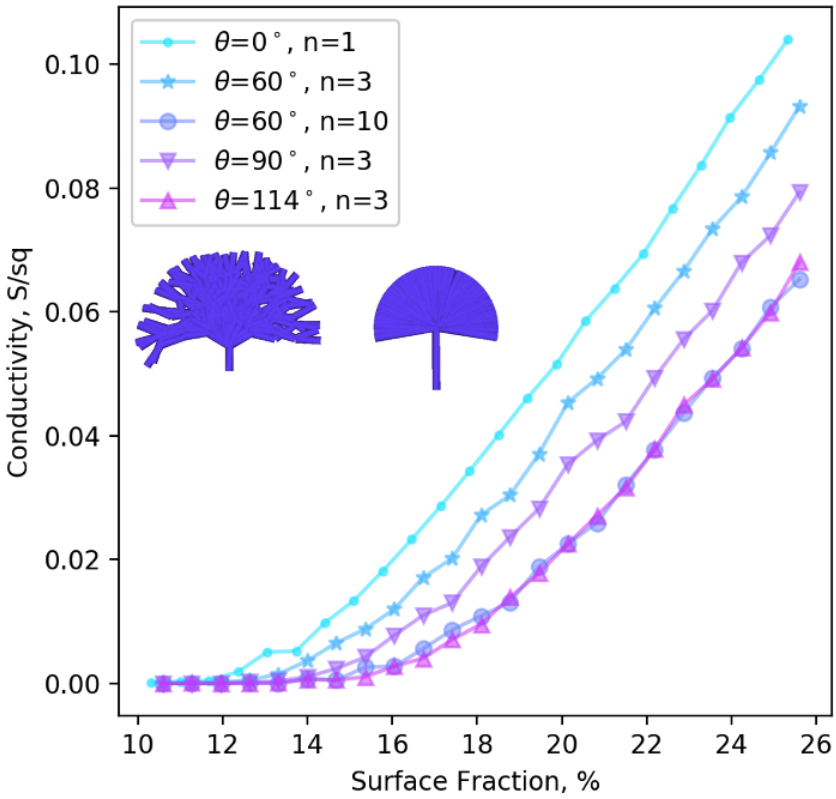


Figure 2.12: Waviness equivalence for wires with different numbers of segments. There is indistinguishable overlap between $\theta = 60^\circ$ at $n = 10$ and $\theta = 114^\circ$ and $n = 3$ suggesting that a smaller number of segments can be used to represent wires with continuous waviness. Inset shows the difference between these configurations.

The conductivity of 3D networks made of curved nanowires also follows the power law Eq 2.4. However, the magnitude of the exponent is larger than the one for 2D distribution and in some reports is considered not to exceed 2 [Katsura et al., 1985b, Gennes, 1976]. Some analyses report values above theoretical estimations such as 3.1 for carbon fibers in an epoxy matrix [Carmona et al., 1980] or up to 2.4 for stainless steel fibers in a polypropylene matrix [Katsura et al., 1985a, Sun et al., 1993].

Similar to 2D cases, in 3D, percolation threshold reduces with the increase of the AR of the wires. However, the bottom line of the conductivity at the percolation threshold for the highest R_j values does not settle at a plateau as we observed in the 2D case. Instead, the conductivity value decreases with the increase in the wire length, also reflected in Figure 2.15 (b).

Unlike 2D cases, in 3D the exponents of the power law fittings follow the opposite trend from Figure 2.15 (a). Instead of climbing, with the increase of the junction resistance R_j, the values drop monotonously from t=2.16 for L=50 μm at $R_j = 10^{-5}$ Ω to t=1.45 at $R_j = 10^5$ Ω. This interesting observation can be explained by the difference in network organization in 2D and 3D. In 2D, any new wire added to the network that contacts other wires always splits them into smaller segments thus increasing the shortest path or betweenness centrality of the resulting conductivity graph. Thus in a 2D network increase of filler concentration pushes the network into an R_j dominated conductivity regime. On the contrary, the 3D case addition of new wires in the network makes it less dependent on R_j and keeps it in a R_w-dominated regime.

One possible explanation of such a difference in conductivity regimes for curved nanowires in 2D and 3D concerns the number of neighbors a wire has access to. In 2D increasing the waviness of the wire increases the probability of connection to their neighbor much more than in 3D. As a result, the patterns for the shortest paths are different in Figure 2.11 and Figure 2.16. In 2D, the shortest path rapidly increases before reaching the percolation threshold, after which it continues to grow but with a much slower pace at a plateau near the maximum. On the contrary, in 3D, the shortest path rapidly decreases after reaching the percolation threshold toward a plateau below the peak.

We observed that the exponent is high when the highest value for large AR nanowires and tends to settle around 1.8 for short nanowires, as shown in Figure 2.13 (a).

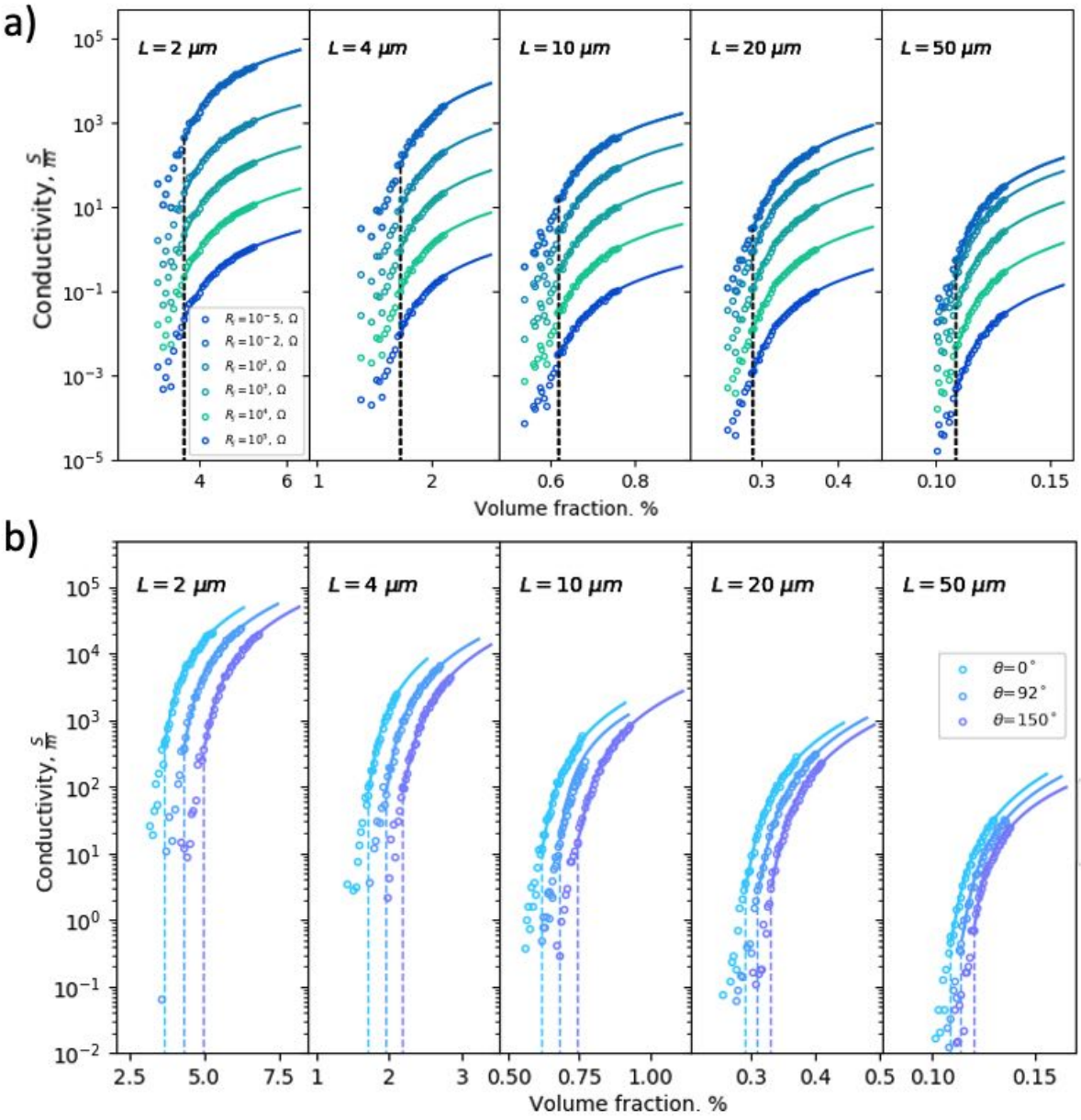


Figure 2.13: Conductivity as a function of attachment angle θ for different L and R_j in 3D. At high AR, the exponent is evaluated at 2 and tends to fall with the decrease in the nanowire length. Thus, there is less sensitivity to the R_j. The difference between max and min exponents is also the largest for the wires with high AR. Fitted power law lines are extended to an additional 50% length.

Figure 2.13 depicts the conductivity relationship on the waviness of the wires (the angle of attachment θ). Again, the pattern of this relationship looks similar for both 2D

and 3D cases. However, as shown in Figure 2.14 (b) the magnitudes of fitted exponents, show different behavior than 2D. In particular, the exponent magnitude in 3D experiments decreases with the increase of waviness, which is contrary to what was observed in 2D experiments where the magnitude of the exponent increased with increased waviness.

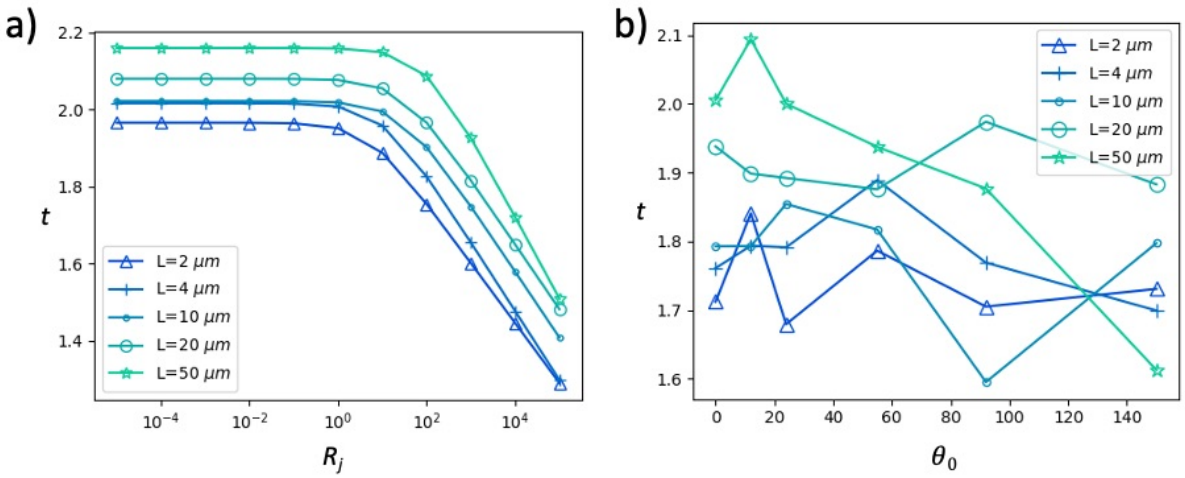


Figure 2.14: Scatter plot matrix for 3D wire distributions summarizing the influence of variable parameters L, θ and R_j on the magnitude of the power law exponent t where the color of each point represents the magnitude of t. This plot shows the strong dependency of the exponent t on the cumulative change of R_j and θ or L and θ but in contrast, t has a smaller dependency on the change of R_j and L.

Figure 2.15 (a) relates the volume fraction at percolation threshold as a function of waviness (or angle of wire segment attachment θ) for different lengths of wires in a 3D system. The volume fraction at the percolation threshold has an exponential dependency on the wires' length and increases rapidly with the increase of the curl ratio θ also follows the pattern described by Eq 2.5.

However, the conductivity at the percolation threshold for various values of R_j shown in Figure 2.15 (b) demonstrates different behavior compared to the 2D system. While a 2D case such as Figure 2.10 (b) conductivities merge at a single value of around $\sigma = 10^{-5}$ S in a 3D case, the pattern is linear for all cases, and each conductivity line is distinct. This behavior also can be explained by the difference in the mean shortest path at the bottom of

Figures 2.11 and 2.16. This phenomenon requires further investigation in theoretical and experimental setups. In the 2D case, the minimum and maximum range of the shortest path is between 55 and 85 segments. However, this ratio is slightly larger for 3D systems, as will be discussed in the next section.

As in the 2D case, the dependency of volume fraction ϕ_0 at the percolation threshold as a function of attachment angle θ and wire length L is summarized in Figure 2.15 (c). The relationship shows that the primary direction of growth of surface fraction at the percolation threshold is when both wire length decreases and the attachment angle increases, with the highest peak situated in the upper left corner. On the contrary, an increase in the length of wire and decrease of the attachment angle causes a reduction of the percolation threshold with the lowest point being in the right bottom corner. While this piece of information seems trivial, it reinforces our computational approach in analyzing percolating systems with the developed software.

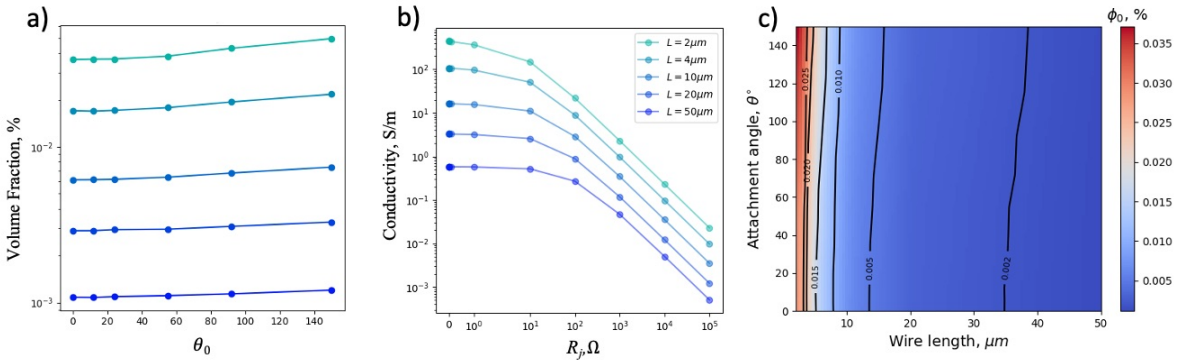


Figure 2.15: a) Volume fraction at percolation threshold for different wire lengths as a function of the segment attachment angle θ. b) Conductivity at percolation threshold as a function of R_j.

As we noticed before there is a significant difference between the network properties of 2D and 3D systems. In particular, we observed opposing behavior in the conductivities at the percolation threshold and power law fittings as functions of junction resistance R_j and angle of attachment θ as we moved from a lower dimensional setup to a higher dimensional one. These differences can be explained by the difference in the shortest path patterns for 2D Figure 2.11 and 3D Figure 2.16, as is described below.

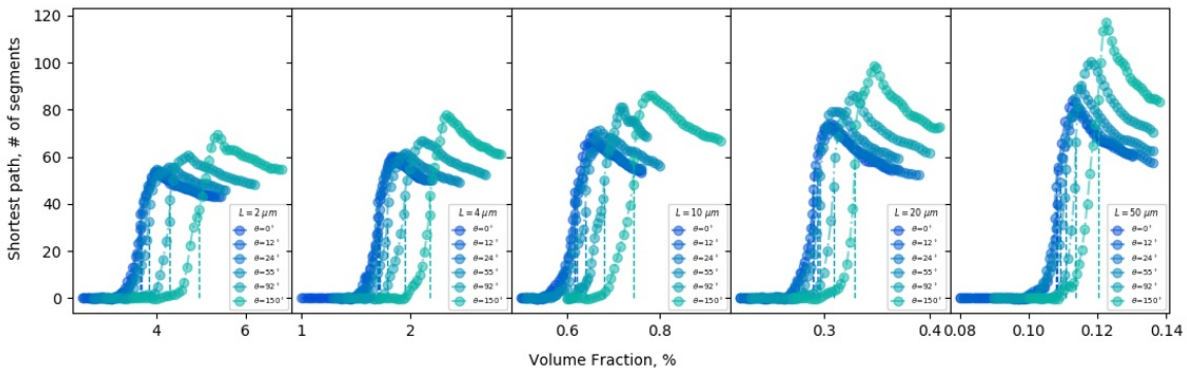


Figure 2.16: Average conductivity at percolation threshold in 3D for different lengths of nanowires and different R_j values (top). The shortest path between opposing electrodes for the 3D system for other AR and waviness of the nanowires (bottom).

To explain this inverse dependence behavior, we need to pair the bottom of Figure 2.11 for the 2D case with the bottom row of Figure 2.16 for the 3D scenario. Unlike in the 2D case, in the 3D system, the average shortest path tends to decrease with the increase of fillers' concentration after reaching the saturation point at the end of the curve's sigmoidal portion. This is marked by a progressive decline in the number of shortest paths between two electrodes. In other words, adding new wires tends to create new alternative shorter paths rather than splitting a path into segments connected by junctions. Thus we observe an inverse relationship of the exponent with the curl ratio parameter θ.

2.10 Experiment vs. simulations in 2D

In this section, we compare our simulation results with experimental results for 2D scenarios. Typical AR for Ag NWs produced in our laboratory is around 500, which is more than sufficient for transparent applications in sensors or wearable electronics, where the practical bottom limit of AR is above 100 [Mutiso et al., 2013].

In this work Ag nanowires are synthesized via the polyol process [Andrés et al., 2015] with the average diameter D = 124 nm and average length L=42 μm measured by TEM (JEOL 2010) [Gao, 2021]. The probability density of the distribution of length and diameter is shown in the inset of Figure 2.17 (a). The Ag NW suspension was triple washed with acetone and ethanol (ETOH) to remove ethylene glycol, and the final product was stored in 97% ethanol.

The surface of the nanowires is covered with residues of PVP, which is used in the polyol process as a capping agent to confine the growth of Ag nanoparticle seeds to a single direction that results in the formation of long nanowires. However, the thickness of this insulating layer is typically around a few nanometers [Milano et al., 2019] and poses a significant obstacle in creating highly conductive networks. While complete removal of the PVP is challenging to achieve due to strong bonding with the Ag NWs surface [Ge et al., 2017], in this work, partial removal of PVP was reached via ethanol [Wang et al., 2014] and acetone [Vafaei et al., 2014, Tokuno et al., 2011] washing of the final product. Comparison of electrical properties of individual Ag NWs with and without PVP layer will be discussed with more details in chapter 3.

Microscope slides for Ag nanowire coatings were pre-cut into 25x25 mm^2 pieces and thoroughly washed with deionized water (DIW), ethanol, and acetone to remove dust and organic contamination. Further, the slides were cleaned with O_2 plasma for 30 seconds to activate the surface.

In this experiment, eight different suspensions of Ag nanowires at concentrations of 0.12, 0.24, 0.36, 0.48, 0.6, 0.72, 0.82 and 0.94 mg/ml were prepared in a 70% ETOH solution with further addition of 0.05 % wt of hydroxypropyl methylcellulose as a dispersing and thickening agent. Then for each concentration, 40 µL of the Ag NW suspension was drop-casted onto a separate clean glass slide allowing it to evenly distribute the wires and create a thin wet film followed by heating of the glass slides on a hot plate at 45 °C to accelerate the evaporation of the solvent for 10 minutes before electrical characterization.

Four-point probe surface conductivity was performed with a Source Measurement Unit (SMU) (Keithley 2636B). Transmission VIS spectroscopy was completed using a broad spectral range spectrometer LR1 (Canada) paired with a 15 W tungsten halogen lamp. Separate measurements were performed on the non-annealed pristine sample then, on the thermally annealed sample as shown in Figure 2.17 (a). The increase in conductivity in the annealed sample results from decreased contact resistance and wire contact welding caused by the thermal diffusion of atomic silver Ag to reduce surface energy.

Various annealing strategies exist for achieving the lowest sheet resistance. For example, Atkinson *et al.* suggest annealing at 180 °C for 30 minutes [Atkinson and Goldthorpe, 2020] while Lagrange *et al.* [Lagrange et al., 2015] experimentally showed that the lowest sheet resistance could be achieved by raising the temperature at a ramp rate of 2 °C/min from room temperature (RT) to 255 °C after which the samples were allowed to cool down back to the RT. In this work, thermal annealing of the Ag NW coated microscope slides was performed on a hot plate at a constant temperature of 150 °C for 30 min.

To compare our results to simulations, the transmittance from Figure 2.17 (a) was converted into a surface fraction with the relationship $T(\phi) = 100 - 1.42\ \phi$ (where T is transmittance at 600 nm wavelength and ϕ is a surface fraction) obtained from fitting the data given by [Khanarian et al., 2013, Lagrange et al., 2015] shown in the inset of Figure 2.17 (b). Notably in both studies, wire diameter played the leading factor affecting the fitted line's slope, in which increasing wire diameter results in a steeper slope of the $T(\phi)$ function.

The power law fittings of the annealed and non-annealed samples in Figure 2.17 (b) show exponent factors $t = 1.48$ for non-annealed and $t = 1.42$ for annealed films, which are in good agreement with the simulation data reported in Figures 2.8 (a) and 2.9 (a). In addition, the experimental values of the conductivity are consistent when compared to the conductivity obtained from simulations where junction resistance R_j between individual wires is between $10^2 - 10^3\ \Omega$ (despite the removal of insulating PVP layer), as is confirmed by the individual wire junctions measurements discussed further in chapter 4. This difference can be attributed to the conservative estimation of the linear nanowire resistivity of $\rho = 5\ \frac{\Omega}{\mu m}$ in Table 2.1.

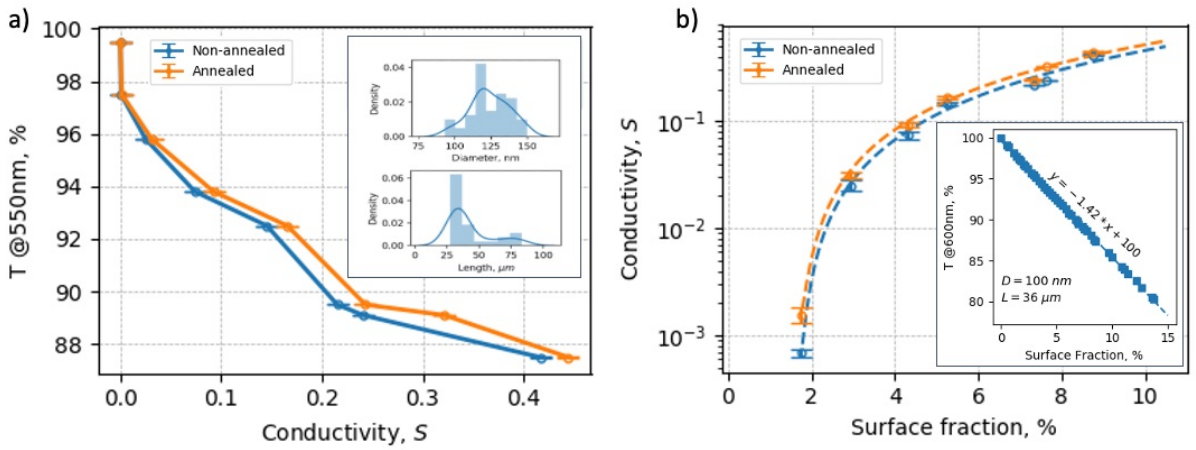


Figure 2.17: Optical and electrical characterization of Ag NW films. a) Transmittance at 550 nm vs. conductivity for non-annealed (blue) and annealed (orange) nanowire films. Insets show the distribution of diameters (top) and lengths (bottom) of the physical nanowires, for which the average diameter of 124 nm and the average length of 42 μm was obtained with TEM (JEOL 2010). b) Corresponding logarithmic conductivity vs. surface fraction for non-annealed and annealed samples. The inset shows the relationship between transmittance and a surface fraction [Khanarian et al., 2013, Lagrange et al., 2015] for wires with D=100 nm and L=36 μm. The dashed line represents best power law fits with exponent factors $t = 1.48$ for non-annealed (blue) and $t = 1.42$ for annealed (orange) samples.

2.11 Experiment vs. simulations in 3D

In this section, we compare the electronic characterization of 3D samples with simulation results. Compared to 2D films, the time and cost to produce 3D samples are significantly higher, especially when using a hard plastic matrix such as polycaprolactone. Therefore, we prototyped a new 3D sample fabrication method using Thinking Putty® as a soft matrix. Electrodes could be repositioned and electrical measurements could be repeated on the same sample several times as needed to accumulate statistics thanks to the material's viscoelastic properties.

Thinking Putty® is a popular sticky gum toy that appears to be produced by a boric acid-induced crosslinking of silicone oil [Boland et al., 2016]. For this work's experimental section, a clear variant of Thinking Putty® was obtained from Crazy Aaron Enterprises, Inc.

Putty samples were manufactured with 0.5%, 1%, 1.5%, 2%, 2.5% and 4% vol of Ag NWs with the average length L=42 μm and average diameter D = 124 nm. First, 500 μL of the putty was swelled in 2 mL of chloroform with suspended Ag NWs and kept at 50 °C under magnetic stirring and simultaneously ultrasonically agitated with a horn sonicator at 5W power for 5 hours until the solvent is evaporated. The resulting sticky gum with a characteristic pearl grey color was then left under a fume hood for another 10 hours for complete solvent evaporation.

In a two probe conductivity measurement setup, each sample was cast into a cylindrical shape with 5 mm diameter and 6 mm height and slightly compressed with copper pistons placed inside a screening cage connected to the SMU (Keithley 2636B).

The resulting conductivity profile exhibits two distinct conductivity regimes shown in Figure 2.18. The first regime is denoted with blue points, and the dashed fitting line represents the subcritical regime. We hypothesize that in this regime, the conductivity is dominated by tunneling through the insulating matrix of the polydimethylsiloxane (PDMS) and PVP coating of the NWs due to the separation of the Ag NWs led by the surface tension forces of the viscoelastic putty. However, it has been noticed that the subcritical regime's conductivity falls within tens of nS/m range and increases proportionally to volume fraction as the best-fit slope is equal to 1. The trend continues until it reaches a critical point at 2% vol. that marks the intersection of the low and high conductivity modes. At this critical point, the concentration of the Ag NW in the volume reaches the critical value, and the surface tension of the viscoelastic putty cannot separate the wires, which designates the supercritical conduction regime. In this regime, the Ag NWs predominantly have ohmic junctions, and conductivity is significantly increased to the region with fractions of S/m.

The exponent of the power law fitted to the supercritical region in Figure 2.18 was determined to be $t = 2.4$, which is in good agreement with the experimental results reported in the literature [Sun et al., 1993, Katsura et al., 1985a, Carmona et al., 1982] and the results of our simulations in Figure 2.13. Although the volume fractions in our simulations are below the experimental results, this reduction can be explained by the subcritical and supercritical conductivity regimes, which require further investigation.

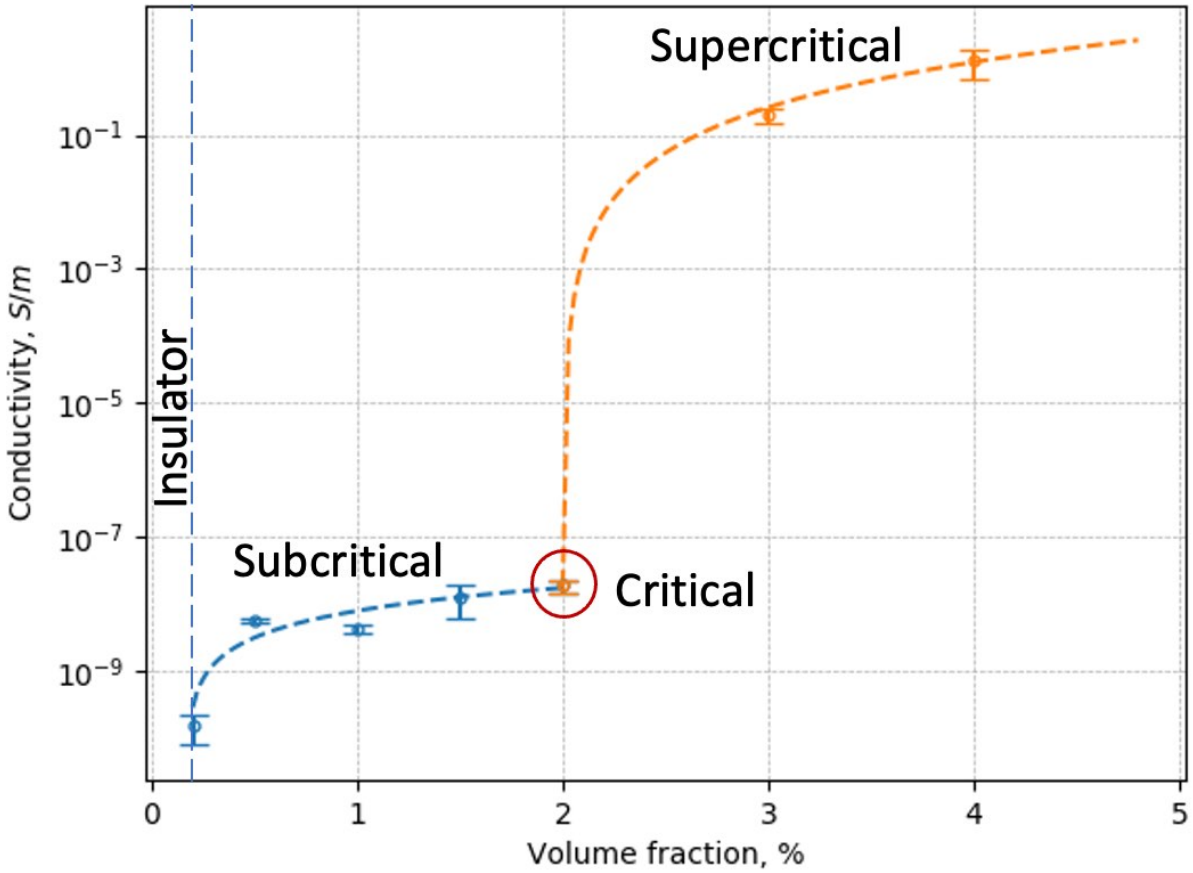


Figure 2.18: Conductivity profile of the Thinking Putty® filled with Ag NWs at various volume fractions exhibiting linear subcritical tunneling junction conductivity regime (blue) and supercritical ohmic contact conductivity regime (orange). (Measurements performed in collaboration with Mahshid Hosseini.)

2.12 Binary filler composite

For this report, binary filler composites are defined as composites of two kinds with different geometries, namely wires and spheres. In this section, we show that our simulation suite can be used to provide insights on electrical properties of more complex component composites, in particular, comprised of a combination of spherical and cylindrical filters. Such composites could describe various systems. For example, a composite composed of highly conductive silver nanowires and solid electrolyte silver sulfide microspheres serves as a neuromorphic atomic switching system [Avizienis et al., 2012] and which will be discussed in detail in chapters 3 and 4. The vast majority of chemical biosensor composites also rely on hybrid nanomaterials' synergistic action [Lu et al., 2020, Xiao et al., 2020]. In addition to biosensor applications, multicomponent composites can be harnessed in a dual-purpose application. For example, carbon black and CNT dispersed in a PDMS matrix were used to increase mechanical strength and provide an ad-hoc strain sensing capability of the composite [Yin et al., 2017].

As a sample system in this section, we explore a ternary composite setup, i.e., two types of filler components, namely wires and spheres, with junction resistance R_j being equal to 0 Ω and 100 Ω to emulate resistivity introduced by the matrix. In particular, we will look at the surface and volume fractions of the fillers in 2 and 3D at different ratios of concentrations of the filler types. We assume that cylindrical wires and spheres are randomly and uniformly scattered on a square surface or in a 3D volume with a uniform orientation probability for the cylinders. The cylinders and spheres have the following dimensions. Length and diameter for the cylinders: L=50 μm and $D = 200$ nm, whereas the spheres' diameter is set to D=5 μm.

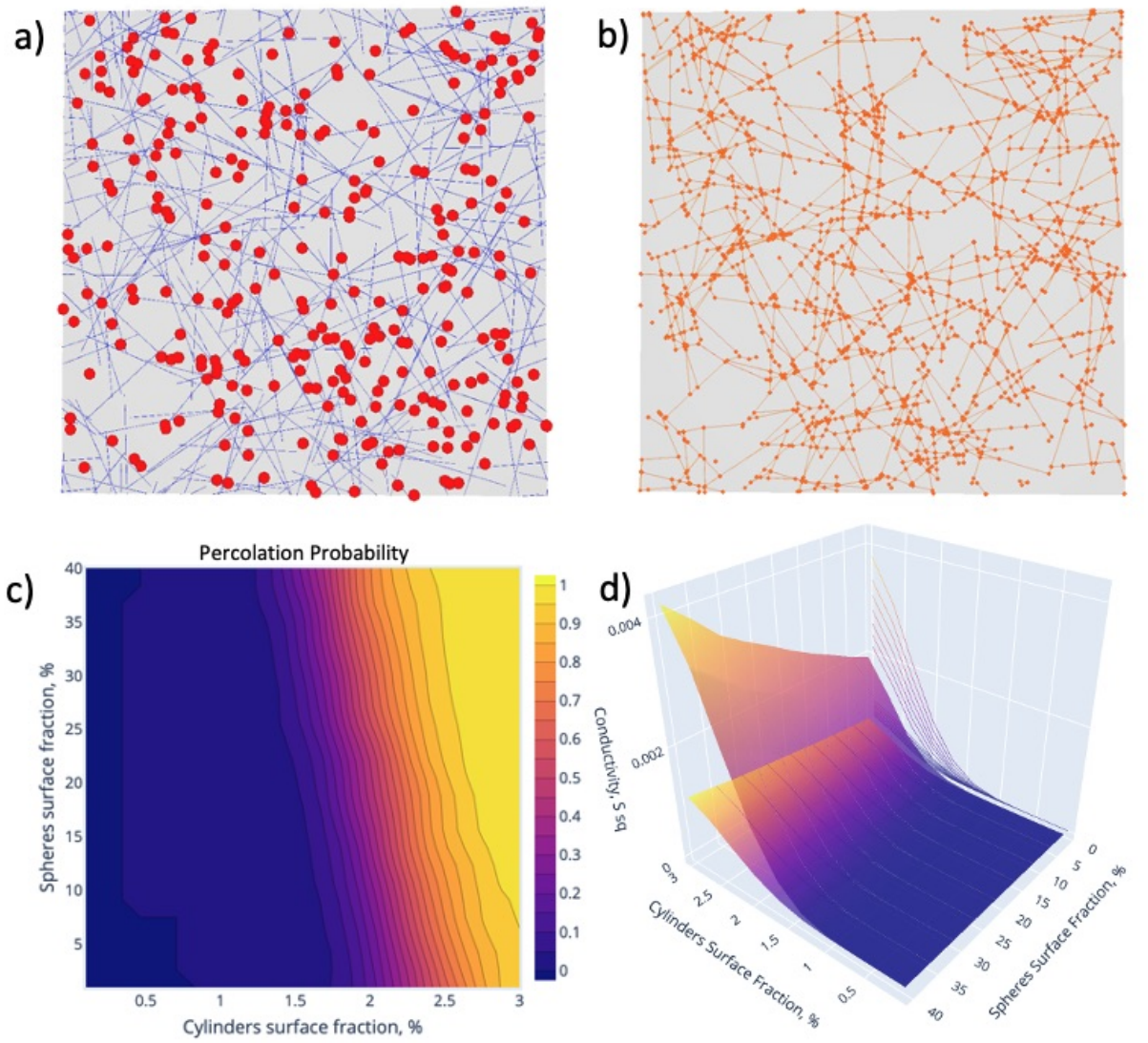


Figure 2.19: Electrical conductivity analysis for the 2D binary system of spheres (D=5 μm and cylinders (L=50 μm, D=200 nm). a) sample distribution of spheres (red) and cylinders (blue) on a surface with periodic boundary conditions. b) derived effective conductivity pathways c) percolation probability for various concentrations of components d) conductivity profile for a network with R_j=0 Ω (top surface) and R_j=100 Ω (bottom surface) where both components made of silver. Projected lines show conductivities at constant surface fractions of spheres, namely 5, 10, 15, 20, 25, 30, and 35 %.

The figures below show examples of simulated percolation analysis in 2D Figure 2.19 and 3D Figure 2.20 systems. In both Figures 2.19 and 2.20 images (a) shows the distribution of cylindrical wires (blue) mixed with spherical particles (red), and image (b) shows a contact

point connectivity graph formed by the collision analysis discussed in section 2.4. Also, Figures 2.19 and 2.20 (c) shows a heatmap of percolation probability as a function of each component's concentrations. The map shows that the concentration of cylinders has a dominant influence on the percolation compared to spheres. Image (d) in Figures 2.19 and 2.20 shows the conductivity profile for a network with R_j=0 (top surface) and R_j=100 (bottom surface) where both components are made of silver. Projected lines show conductivities for constant surface fractions of spheres.

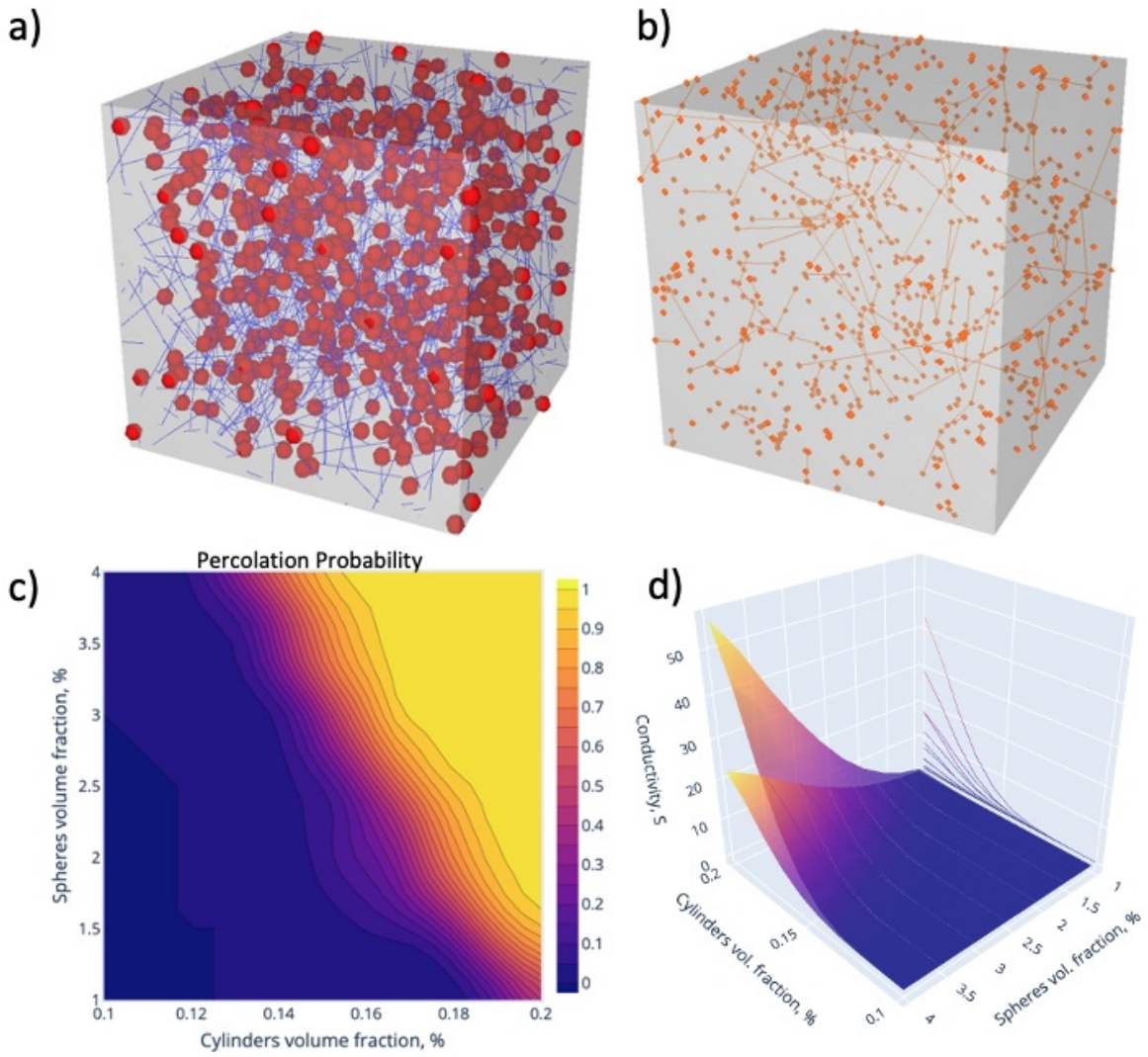


Figure 2.20: Electrical conductivity analysis for a 3D binary system of spheres (D=5 μm) and cylinders (L=50 μm, D=200 nm). a) sample distribution of spheres (red) and cylinders (blue) on a surface with periodic boundary conditions. b) derived effective conductivity pathways c) percolation probability for various concentrations of components show an increased probability of conductivity when concentrations of both components are high but low probability when either of components' concentration is low d) conductivity profile for a network with $R_j = 0$ (top surface) and $R_j = 100$ (bottom surface) where both components are made of silver. Projected lines show conductivities for constant surface fractions of spheres, namely 1.5, 2, 2.5, 3, 3.5, and 4 %.

2.13 Agglomeration

Agglomeration, or clustering, of wires, can significantly affect the conductive profiles of these materials. To conclude this chapter, we will explore another important application of the simulator suite to understand the effect of the agglomeration of cylindrical wires on the percolation and conductivity profiles. In particular, we show that agglomeration can significantly increase the percolation probability in systems where the initial volume fraction of cylindrical fillers with Ag conductivity was below the percolation threshold.

Uniform dispersion of metallic nanowires is a complex task. When dispersed in a medium, the wires are typically agglomerated and cause significant problems with conductivity if not properly deagglomerated without degradation of the nanowire's geometry [Ma et al., 2010].

Unlike other studies where agglomerated particles have sizes comparable with the size of the system [Román et al., 2018, Bao et al., 2013], here we study a larger system with multiple clusters. In particular, our strategy is to introduce a coefficient of clustering which controls the fraction of material being in the cluster outside of it. In order to achieve it, the fillers are populated in two regimes, first a sparse (below percolation threshold) background population of the wires is generated, after which at a fixed number of random locations (typically n=10) and smaller spherical subregions (with diameter D=1/5*x where x is the box's side length) are filled with a much denser (above percolation threshold) population of wires. The sum of volume fractions in the sparse and dense phases is equal to the initial value. Taking a ratio of volume fraction of the wires in the dense phase to the volume fraction of the sparse phase would yield a clustering coefficient that in this experiment varied from 0 to 0.05%.

The setup of the model is shown in Figure 2.21 (a) for the wires and (b) the resulting conductivity networks. Interestingly the plots in Figure 2.21 where percolation probability (c) and conductivities for various junction resistances (d-e) are shown. The green and white arrows in Figure 2.21 show the direction of increase of clustering while keeping constant the total volume fraction of the wires. This plot suggests, that increase of the clustering once the system has reached the percolation threshold tends to decrease percolation probability and hence conductivity (Figure 2.21 (c) green arrow). However, those samples which have not reached the percolation threshold tend to increase percolation probability up to a certain point and become insulating at an even higher clustering ratio (Figure 2.21 (c) white arrow).

This novel way of analyzing percolating systems can be handy during the quantitative analysis of conductivity in 3D systems from confocal microscopy or X-ray CT tomography.

In all of the experiments described above, particles were generated in two steps: first clustered ones, then second uniformly distributed background ones. In the end, the periodic boundary was applied.

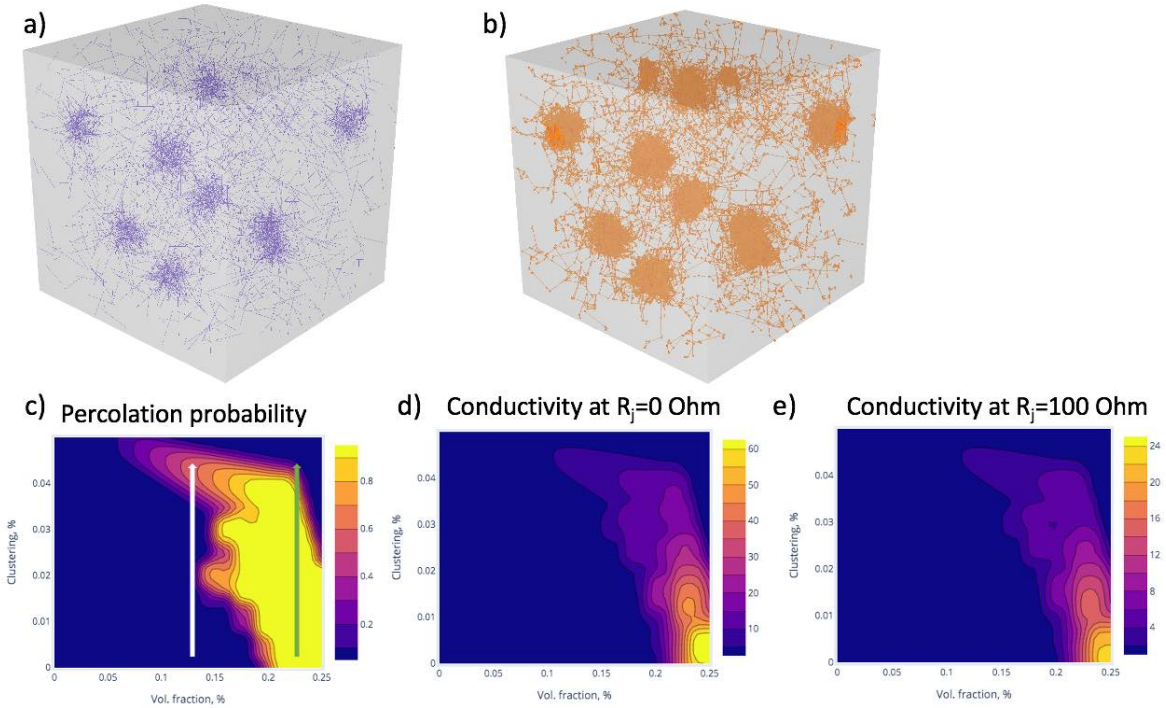


Figure 2.21: Percolation and conductivity analysis of a system with variable concentrations of localized dense and background sparse populations of randomly distributed wires. a) example model of nanowires showing the sparse distribution of wires in the background and 10 dense clusters. b) connectivity graph of the system c) The clustering coefficient denotes the ratio of wires in the clusters to the total number of wires outside the clusters. The increase of the clustering coefficient is accomplished by relocating randomly chosen wires from the background distribution and adding them to the clusters. c) Percolation probability as a function of variable clustering coefficient and volume fraction. For example, at a fixed low volume fraction of 0.13 %, the increase of the clustering coefficient, marked by the white arrow, transforms the non-conducting system into a conducting one when the clustering coefficient is set between 0.04 and 0.045. This temporary increase in percolation probability and conductivity is attributed to the balanced equilibrium of connectivity created by the clusters and background wires. The calculated conductivities for the system at R_j=0 Ω and R_j=100 Ω are shown in d) and e) correspondingly.

2.14 Conclusion

This work shows a novel 3D graphics engine-powered approach to analyzing percolation in arbitrary composite systems. With this software, we performed an in-depth analysis of 2D and 3D systems composed of straight and curved Ag NW fillers as single-type fillers and as composite or agglomerated phases.

In particular, we analyzed the influence of wires' AR, waviness (via angle of segment attachment), and contact resistance showed direct for 2D and inverse for 3D correlation between the critical exponent and the waviness of the wires.

The percolation and conductivity profiles of binary composites in 2D and 3D showed how the composite's conductivity could be studied with our simulation suite. Importantly the simulation suite allows one to estimate without running complex experiments the approximate magnitudes of concentrations of the filler materials to achieve desired conductivities. More complex composites (containing more than two components) are possible to study and can be easily implemented with our software suite.

In addition to binary composite studies with our simulation suite, we looked at nanowire composites' agglomeration phenomena. In particular, we derived percolation probability and conductivity profiles as a function of the degree of agglomeration of nanowires in the volume. From these profiles, it was found that agglomeration might help reach the percolation threshold even at constant concentrations of the wires. This study may prove helpful in multiple applications where agglomeration is difficult to avoid or when it can be specifically controlled and programmed.

Further, we showed that our theoretical studies of percolation phenomena agree with other theoretical reports. In contrast, the experimental data we obtained exhibits complex behavior that is corroborated by the critical exponent fittings' magnitudes in simulations studies. In all the cases, the fillers' concentrations in theoretical models are lower than the ones observed in the experimental models. This suggests that in real composites, the conductivity phenomena is largely mediated by not just ohmic regimes but, rather a mixture of ohmic and tunneling effects that require an in-depth study of each particular scenario. Further improvement of the theoretical model will benefit from implementing a "hard-core" model and an implementation of junction distance mediated interplay between tunneling and ohmic conductivity in the contact region of the fillers.

To summarize, the theoretical methods described in this study enable a researcher to quickly estimate percolation conditions and estimate concentrations of materials needed for each particular case despite the complexity of the geometry of individual components

and geometry of the system.

CHAPTER 3

Simulation of nonlinear computing with resistive switching networks

3.1 Introduction

The development of autonomous intelligent systems with conventional top-down processes has several disadvantages. Unlike biological systems, the modern VLSI architectures require careful engineering, making them particularly vulnerable to any faults within their architecture. Further, in conventional VLSI systems, storage is separated from processors resulting in severe performance limitations that are frequently referred to as the "von Neumann bottleneck" [Petrenko, 2018, Backus, 1978]. Thus, a new engineering approach reliant on self-assembly and an in-place computation is required.

The memristor is one of the most promising candidates for constructing bio-inspired computers as it simultaneously embodies both storage and computation units [Stieg et al., 2011]. Furthermore, the computational power of memristors grows with the number of devices and connectivity between them, as we will argue in section 3.3. In particular, memristor crossbar arrays can be used to perform real-time matrix dot-product operations [Alibart et al., 2012], real-time data clustering [Choi et al., 2015], and solving matrices [Sun et al., 2019] and partial differential equations [Zidan et al., 2018].

While the crossbar array approach eliminates the "von Neumann bottleneck," it requires a top-down manufacturing process that is highly sensitive to production flaws. Moreover,

unlike crossbar arrays, reservoir computer architecture [Lukoševičius and Jaeger, 2009, Jaeger, 2004] is entirely random and comprises a network of artificial neurons sparsely interconnected via synapses that can be self-assembled at nanoscale.

Thus, due to the explainability of the results, most reservoir computing research focuses on the application of crossbar oxide memristors in such tasks like image recognition [Du et al., 2017b] and temporal signal processing [Zhong et al., 2021]. However, due to the issues with repeatability, there are only a handful of reports on the physical implementation and analysis of self-assembled nano-scale reservoirs. For example, self-assembled silver sulfide nanowires were used for a patterned electrical signal generation [Stieg et al., 2014, Sillin et al., 2013] or a network of gold nanoparticles were configured with a heuristic search algorithm into a logic gate [Bose et al., 2015].

In this computational work, we study the collective behavior of random memristive networks at various degrees of organization. In section 3.2 we simulate Ag_2S nanowire networks assembled in regular lattices or at random in 2D and 3D. In these studies, we found that despite the organization of the network, under similar stimuli conditions all networks expose a degree of current-controlled negative differential resistance (NDR) characterized by the nonlinear current response, disproportionate to the voltage input. In this particular case, NDR is a network effect that requires deeper investigation; however, we argue that it enhances the mapping representation in classification problems. Finally, we conclude the section with a visual explanation of the phenomena with the help of a Wheatstone bridge-like memristor circuit configuration.

In section 3.3 we explore the nonlinear mapping of the input voltage into output current with the memristor circuits. Our observations unveil the complexity of the possible states of the network and suggest that these states can be harnessed for computation.

To reinforce our observations and create a foundation for more complex experiments, section 3.4 illustrates the dynamics of nonlinear mapping of the input voltage into output current at different input hold times and different control voltage magnitudes. The outcome supports our hypothesis that, indeed, output representation can be altered with the help of static control voltage.

Further, in section 3.4.2, several results are presented in linearizing binary and ternary XOR classification problems with random memristive reservoirs using static control voltage. Additionally, an application of a random memristive reservoir to image recognition is explored.

In previous results sections, networks were constructed with identical memristors with the same geometry and physical properties. In section 3.5 we modeled randomly

distributed Ag_2S atomic-switch nanowires in 2D with different lengths that exhibited useful computation capabilities.

Finally, section 3.6 shows results for image recognition with random memristive networks and compares them to an equivalent cross-bar array implementation published by other groups. We conclude the section with a detailed comparison of the classification accuracy between random and deterministic (cross-bar) networks, where random networks showed a noticeable increase in accuracy compared to those designed with cross-bar arrays.

3.2 Ordered and random memristive networks

This section aims to compare the response of ordered and random networks to periodic voltage stimuli. We begin by showing the response of a simple 2D lattice network comprised of non-volatile memristors, as described below.

Equations 3.1 and 3.2 summarize the non-volatile metal-oxide memristor model [Strukov et al., 2008b] with normalized length of the doped (Ag rich) region ω. The resistance value of R_{ON} and R_{OFF} represent ON and OFF states when ω=1 and $\omega = 0$ correspondingly. The mobility μ and thickness of the thin film device L (or length for wires). Equation 3.1 represents Ohm's law for the device, whereas the ordinary differential equation 3.2 relates the rate of change of the width of the doped region ω to the direction and the magnitude of the current passing through the device.

$$V = \big(R_{on}\omega + R_{off}(1-\omega)\big)\, I \tag{3.1}$$

$$\frac{d\omega}{dt} = \mu \frac{R_{on}}{L} I \tag{3.2}$$

A lattice network constructed of memristive wires of equal length intersecting at 90° angles to each other in a 2D surface is shown in the right pane inset of Figure 3.1. The input triangle shape voltage signal is fed into the system, whereas the resulting current is harvested from the grounded end. The left pane of Figure 3.1 shows the result of such a simulation. The memristors used in this model are non-volatile. The misbalance of current induces volatility in the circuit, which will be discussed in detail further. A train of input

positive and negative triangular voltage patterns through a 2D lattice of memristor elements produces a nonlinear response. The network behavior has a nonlinear switching pattern with apparent differential resistance of the changing sign during the sweep of the same polarity. This is an emergent property of memristive networks, which results in responses not observable in isolated memristors.

In this section, each of the memristors has equal dimensions and parameters with mobility and resistances scaled to the size of each device taken for Ag/Ag_2S. We study the response to similar stimuli for a 2D and 3D network of randomly distributed Ag/Ag_2S memristors. In particular, mobility $\mu = 20\ \frac{\mu m^2}{Vs}$ and $R_{on} = 1\ \frac{\Omega}{\text{nm}}$ and $R_{off} = 1\ \frac{k\Omega}{\text{nm}}$ all initialized to their intermediate state i.e. $R = \frac{R_{on}+R_{off}}{2}$. The network is comprised of 18x18 randomly oriented memristors where each device's polarity is undetermined.

Input voltage and corresponding current going through the network are shown on the top and middle left, respectively. The computed resistance of the device is shown on the bottom left. The I-V characteristics of the network (right) display an NDR pattern. This is mainly expressed during the positive sweep on the right of the plot. The loop starts with the current at around 2 μA and moves to the bottom and then to the top, a phenomenon that cannot be observed in a single non-volatile bipolar memristive device. However, NDR behavior is observed in memristors with Mott switching mechanisms in such materials as VO_2 [del Valle et al., 2020] and NbO_2 [Pickett et al., 2012].

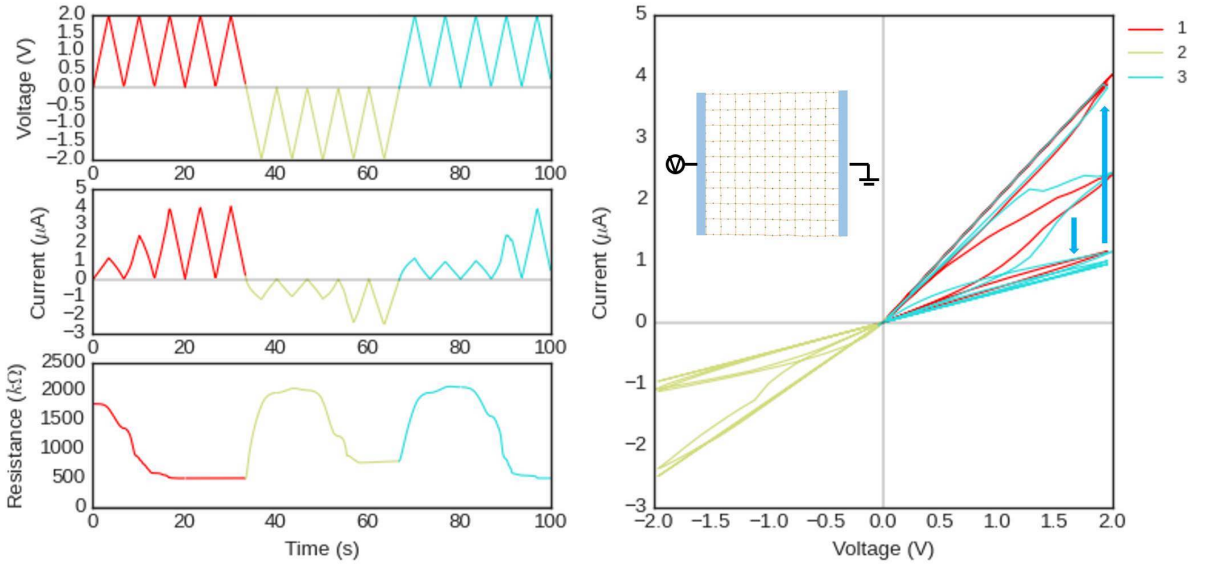


Figure 3.1: Triangular voltage sweeps through a 2D lattice of memristor elements. Input voltage and corresponding current going through the network are shown on the top and middle left respectively. The computed resistance of the device is shown on the bottom left. The I-V characteristics of the network (right) display an NDR pattern. This is mainly expressed during the positive sweep on the right of the plot. The loop starts with the current at around 2 μA and moves to the bottom at 1 μA first and then all the way to the top at 4 μA.

A similar resistivity and IV relationship was noticed in a 3D periodic lattice, as shown in Figure 3.2. Interestingly, despite the input signal pattern (-2 V, 2V) and 100 s duration being identical in this and the 2D sample from Figure 3.1, the resistance spike was shorter, with around 5 seconds duration (vs. 10 seconds in Figure 3.1).

A similar study was performed for a random 2D network of 879 randomly distributed memristive nanowire devices, as shown in Figure 3.3 (right pane). Each device is randomly placed in the network in this random configuration, similar to the ordered lattice of memristors shown in Figure 3.1. In particular, there is clear evidence of the short-term current-controlled NDR. During the second wavetrain, depicted in yellow, the network's resistance starts from 600 $k\Omega$, gradually dropping to 800 $k\Omega$ after reaching a peak value of 1 $M\Omega$.

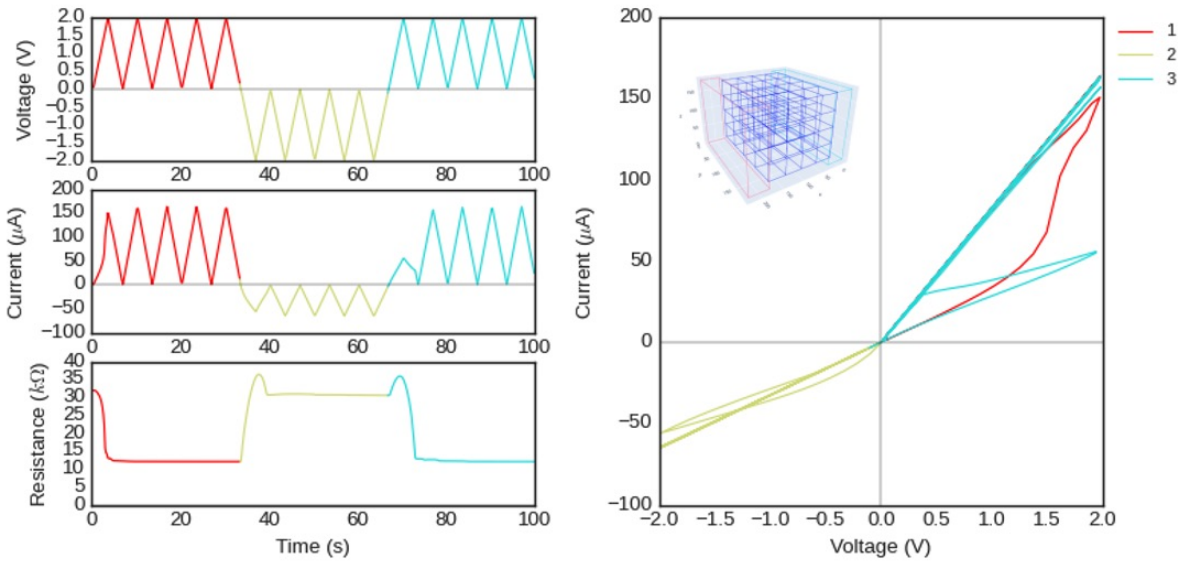


Figure 3.2: Triangular voltage sweeps through a 3D lattice of memristor elements. Input voltage and corresponding current going through the network are shown on the top and middle left, respectively. The computed resistance of the device is shown on the bottom left. The I-V characteristics of the network (right) also display an NDR pattern. This is mainly expressed during the positive sweep on the right of the plot. The loop starts with the current at around 2 μA and moves to the bottom at 1 μA first and then to the top at 4 μA.

The behavior of a random 3D network exposes more diversity in local connections and a higher probability when creating distant connections. Despite that current looks similar to the voltage input, the network's resistance undergoes abrupt changes of different scales as seen in Figure 3.4. In particular, the loops on the I-V plot have a higher level of distortion, yet we also witness the abrupt change in the resistance. Unlike the 2D case, the distribution of the device lengths in 3D has a more linear descent and can be approximated by Poisson or Gamma distributions.

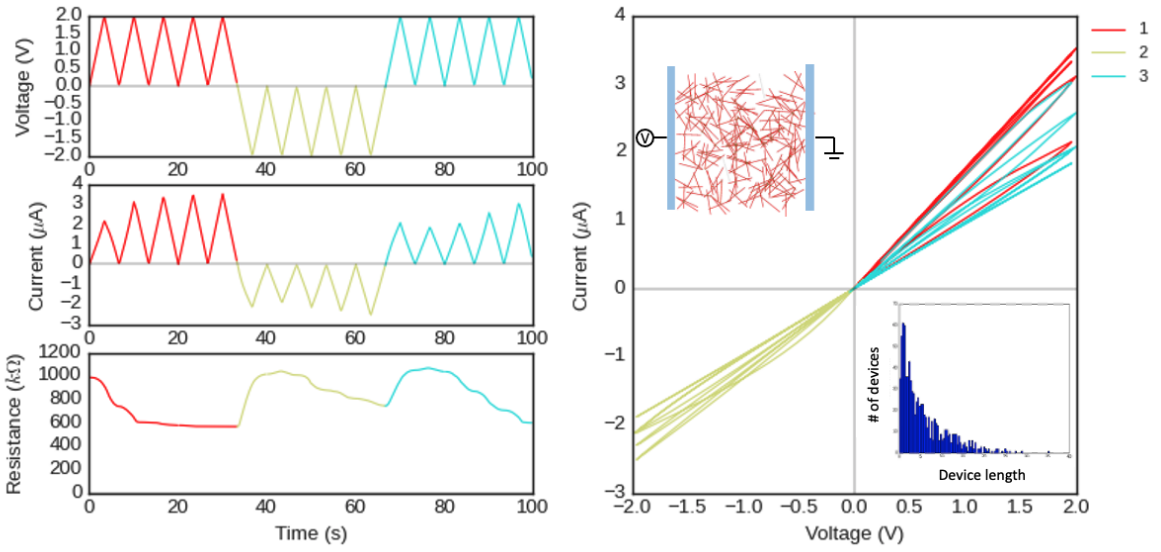


Figure 3.3: Triangular voltage sweeps through a network of 879 randomly distributed memristors. During the second wavetrain (yellow) the grid's resistance starts at 600 $k\Omega$ then gradually drops to 800 $k\Omega$ after reaching a peak value of 1 $M\Omega$. The top-left inset depicts the actual network placed on a 200x200 μm^2 surface. The inset on the bottom right shows the distribution lengths of the memristor wires in the network.

In all the above-simulated studies of simplified systems, i.e., 2D and 3D lattice and random networks, we have observed two common behavior patterns that make these systems different from a single memristor device's behavior. First, the network's behavior becomes more complex when (more concise) dimensionality of the distribution of the constituents of the system increases, such as in the 3D case. In particular, a simple 2D grid network exposes globally rich properties, i.e., the nonlinear features on the scale of the experiment timespan. On the contrary, due to the monotonic distribution of the devices' lengths, lattice networks do not have as rich local irregularities as random networks which have nonuniform device length distribution. Second, each of the patterns above exposes NDR, which lasts for a short time and is recoverable only after resetting the device.

Both of these observations can be explained by two properties of the memristive networks: the distribution of the sizes of the devices and orientation, and the availability of conductive pathways. Thus in a lattice network, the key contributor to nonlinearity is the orientation of the poles of each device, whereas in a random network distribution of sizes starts to play a more important role.

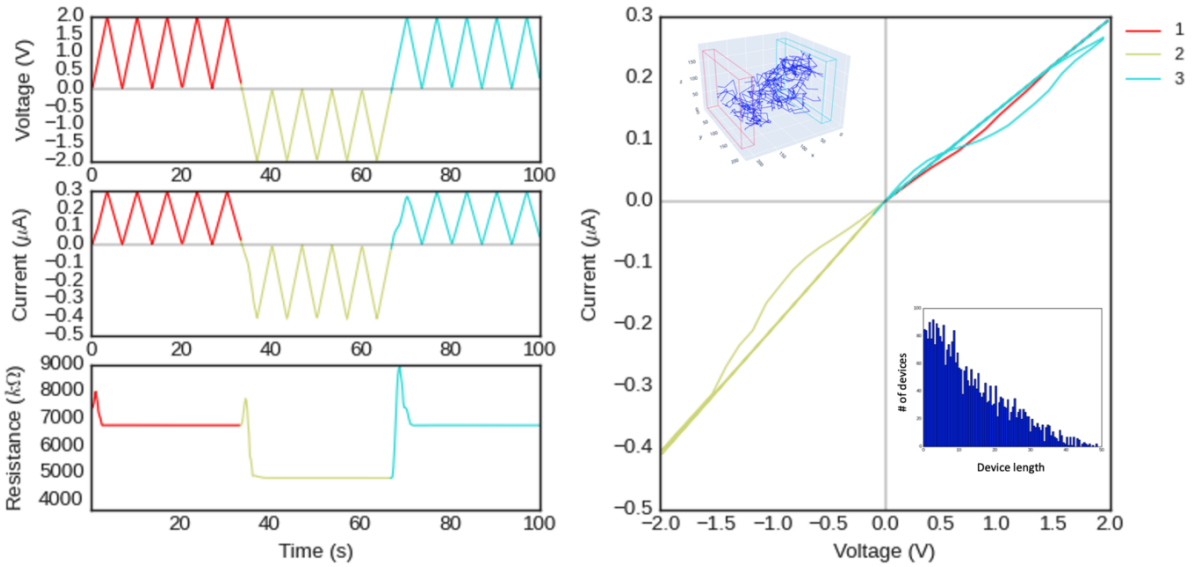


Figure 3.4: Triangular voltage sweeps through a 3D network of randomly distributed Ag_2S memristors. During the negative portion of the triangle wavetrain, depicted with yellow color, the grid's resistance suddenly spikes from 7 $k\Omega$ to 8 $k\Omega$ but then drops to 5 $k\Omega$ values. The inset on the I-V plot depicts the actual network in a 200x200x200 μm^3 volume. The inset on the bottom right of the I-V plot shows the network's length distribution of the memristor devices.

The NDR effect in the network of memristors can be explained by the cumulative action of series and parallel connections as shown by a simplified diagram of a memristor Wheatstone bridge on the right side of Figure 3.5. The pathway of the charge flow is determined by the sign of the input voltage signal. Both pathways are constrained by the middle memristor, which, for this particular configuration, always settles in the ON state (or OFF state if its orientation is reversed), regardless of the polarity of the input state. However, during the switch of the polarity, it does undergo an abrupt increase in resistance, which is responsible for the bump on all the diagrams in Figures 3.1-3.4. Remarkably, the voltage on the memristor resembles a depolarization-like pattern if the bridge is stimulated with a rectangular signal, as is shown in Figure 3.5.

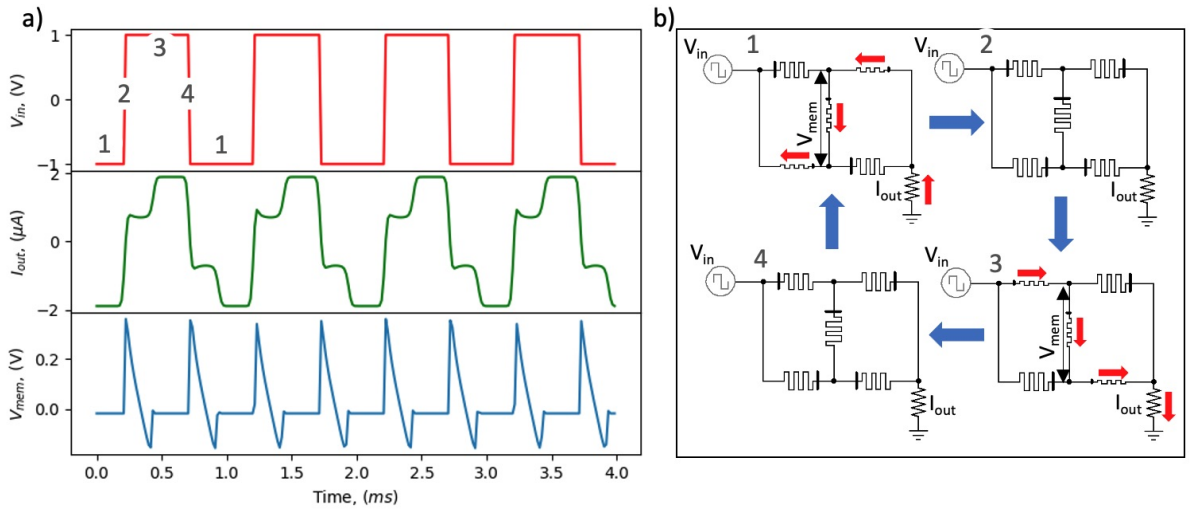


Figure 3.5: Negative differential resistance in Wheatstone bridge memristive circuit. a) Input square voltage (red), output current (green), and the voltage spikes (V_{mem}) through the bridge's central memristor, which surprisingly looks like biological cell depolarization voltage. b) The system's four different states during the inputs' voltage cycle: -1,0,1,0,-1,. The direction of the major current is marked with red arrows.

3.3 Memristor circuit as a nonlinear map

The complex self-assembled networks of synaptic logical gates in vertebrate brains enable them to selectively respond to desired physiological stimuli. In 1965, Barlow and Levick [Barlow and Levick, 1965] described a mechanism of preferential direction determination of the light source in the rabbit retina. It was later determined that the non-linear interplay of excitatory and inhibitory synapses on the passive dendritic tree constructs complex logical expressions which can result in various kinds of responses [K. Koch, 1982].

Memristors are ideal candidates for computation due to their ability to change conductivity with respect to the amount and direction of the passed charge. This serves as an approximation of synaptic function that can also be used to construct complex logical gates. As was shown in previous sections, memristive circuits act as a reservoir with current-controlled NDR. This section aims to show that memristive networks represent a reservoir of mathematical operations on the input that can be harnessed for computation where the state of the circuit depends not only on the stimuli at the time of measurement but also on the history of previous states [Jaeger, 2007]. This property of reservoir computers is useful for expanding the dimensionality of input data and highlighting correlations

between the input features, making them linearly separable at the output [Du et al., 2017b]. In the next sections, we will describe the series of *in silico* experiments in which we tested memristive circuits' ability to solve the XOR problem, a building block to perform arbitrary computation, and basic image recognition tasks.

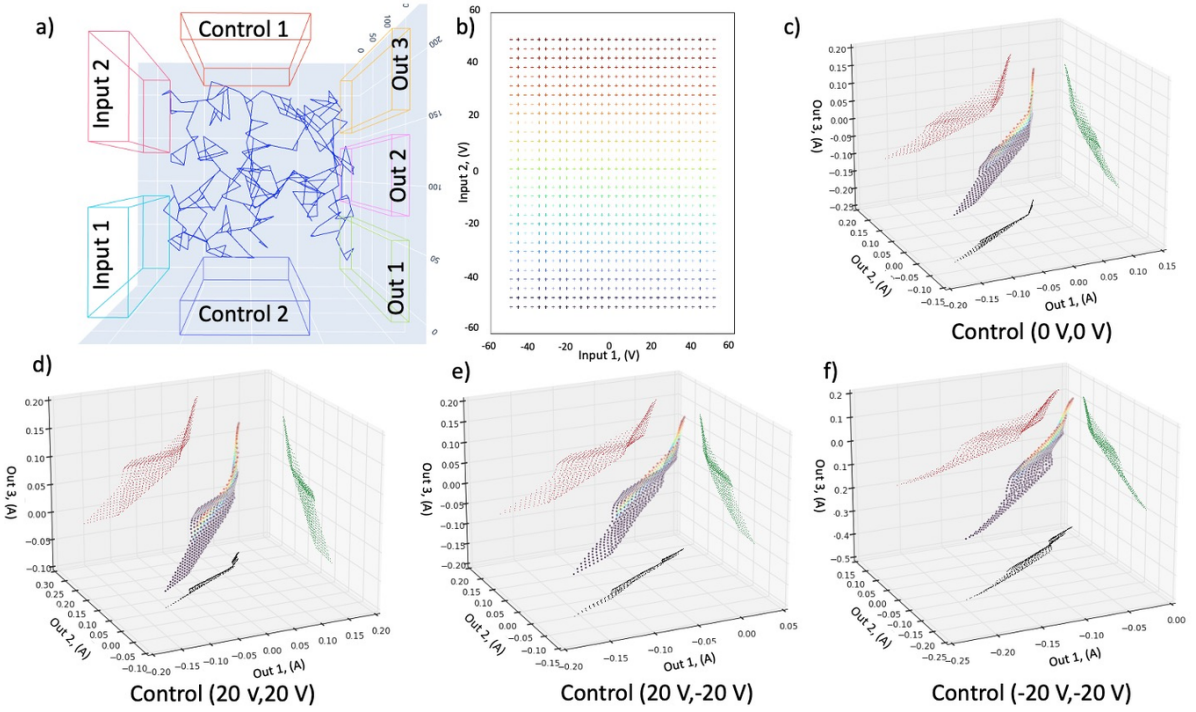


Figure 3.6: Mapping input voltage into output currents with a random memristive network with 2 input, 3 output, and 2 control electrodes. a) The network, modeled of Ag/Ag_2S system b) the map of input voltages where Input 1 corresponds to voltage V1. The network was first reset to the initial state before a value from the map is drawn and is set as an input for the duration of 1ms. c) The resulting 3D representation of output current with zero voltage on control electrodes in which each axis is a current measured through output resistors connected to each output terminal. For example, Out1 corresponds to current measured through the R1 resistor. d) During the same scan, the input voltage on control terminals V1=20V and V2=20V deforms the shape of the resulting mapping surface. e) Similar to d) but with control voltages equal to V1=20V and V2=-20V. f) Similar to d) but with control voltages equal to V1=-20V and V2=-20V.

An experimental setup in Figure 3.6 (a) shows a small random network modeled with an ideal non-volatile rod-shaped Ag/Ag_2S memristors scattered on a 2D surface whose inputs and control voltages are connected to voltage sources whereas each output terminal

is grounded through a 100 Ω resistor. First, a voltage mesh grid was created in Figure 3.6 (b) for each input terminal where each color-coded point in the grid was picked sequentially and fed into a freshly initialized network for the duration of 1 ms under control voltages which varied in amplitude. After that, currents on the output terminal resistors (R1, R2, and R3) were measured and analyzed via scatter-plot, resulting in the 3D surface figures shown in Figure 3.6 (c-f). First, the voltages on the control terminals were set to 0, resulting in a complex surface representation as shown in Figure 3.6 (c). The color-coding on the resulting surface suggests that the output current vector is the result of a series of affine transformations on the input voltage vector such as scaling, translation, rotation, and shear, which is impossible to represent with basic linear passive elements such as a resistor.

$$I_j(t) = \sum_{i=0}^{N} \int_0^T Z_{ij}(t)\, V_i(t)\, dt \tag{3.3}$$

The nonzero voltage on the control terminals distorts the resulting mapping and is shown in Figure 3.6 (d-f). These results suggest that the random memristor network acts as a two-port voltage to current map Eq 3.3, where $Z(t)$ is an MxN time-dependent transfer matrix, equivalent to conductance matrix, where M is the number of outputs and N is the number of inputs. I_j and V_i represent current on the j'th output and voltage set on the i'th input correspondingly. When control voltage is used, V_i can be extended with control voltages and the size of Z_{ij} will be increased to MxN+C, where C is the total number of control inputs.

The $Z_{ij}(t)$ matrix embodies complex relationships that depend on the architecture and topology of the underlying circuit. However, since the size of the Z matrix can be non-square if the number of inputs is not equal to the number of outputs, we cannot directly solve (invert), as this system of equations can be under or over-defined. Further, this section, will show how to square Z matrices can be constructed using only graph theory and reduce it to conventional modified nodal analysis (MNA) [Ogrodzki, 2018] circuit matrix.

In order to develop intuition about the matrix operator Z_{ij}, we will construct it with an assumption that any circuit can be represented with a non-directed graph with edges serving as resistive elements between nodes. In particular, any circuit can be represented with a non directed graph $G = G(V, W)$, where the voltages v_i on each node are represented by vector $V = \{v_1 .... v_m\}$ and the conductance, or equivalently inverse of resistance of each edge e_{ij} connecting nodes i and j, is represented by the adjacency weight matrix $W = w_{ij} = \frac{1}{R_{ij}} = \sigma_{ij}$. Thus the graph of such a circuit can be described with a square symmetric Laplacian matrix $L = D - W$ where D is a diagonal degree matrix $D = diag(d_1 ... d_m)$ specifying the sum

of conductances of the nodes' neighbors $d_i = \sum_{j=1}^{m} w_{ij}$ or equivalently a sum of rows or columns in the matrix W.

Generic Laplacian formulation of the Ohm's law can be obtained as is shown in the vector and matrix forms in Eq 3.4 where V and I are the column vectors describing the voltages and the injected currents in each node:

$$LV = I \qquad \begin{pmatrix} d_1 & \omega_{12} & \omega_{13} & \cdots & \omega_{1n} \\ \omega_{21} & d_2 & \omega_{23} & \cdots & \omega_{2n} \\ \omega_{31} & \omega_{32} & d_3 & \cdots & \omega_{3n} \\ \cdots & \cdots & \cdots & \ddots & \omega_{4n} \\ \omega_{\mathrm{n1}} & \omega_{\mathrm{n2}} & \omega_{\mathrm{n3}} & \omega_{\mathrm{n4}} & d_n \end{pmatrix} \begin{pmatrix} v_0 \\ v_1 \\ v_2 \\ \cdots \\ v_n \end{pmatrix} = \begin{pmatrix} i_1 \\ i_2 \\ i_3 \\ \cdots \\ i_n \end{pmatrix} \tag{3.4}$$

The V vector of node voltages in the circuit is unknown, except the nodes connected to the voltage source. Similarly, the elements of the current vector I are all zero except the ones which are connected to the power supply i_{in} and the ground i_{out} and where $i_{out} = -i_{in}$. One can rearrange these equations to move all unknown nodal potentials and input currents to the left side while the known input voltages are defined by the vector on the right side. Then the columns and rows of the matrix that correspond to the zero ground node with zero potential can be removed [Kagan, 2015]. As a result of this, each power supply will have one of the poles connected to the ground. The resulting set of equations that are shown in Eq 3.3 will be equivalent to the ones obtained with the MNA [Ogrodzki, 2018]. The MNA is an industrial standard for simulating circuits and is easy to implement programmatically is an integral part of the major SPICE (Simulation Program with Integrated Circuit Emphasis) circuit simulator software.

$$\Theta = \tilde{V} => \begin{pmatrix} d_1 & \omega_{1,2} & \cdots & \omega_{1,n-m} & 1 & \cdots & 0 \\ \omega_{2,1} & d_2 & \cdots & \omega_{2,n-m} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \omega_{n-m,1} & \omega_{n-m,2} & \cdots & \ddots & \vdots & \cdots & 1 \\ 1 & 0 & \cdots & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_{\text{n-m}} \\ i_1 \\ \vdots \\ i_m \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ v_{\text{in1}} \\ \vdots \\ v_{\text{inm}} \end{pmatrix} \tag{3.5}$$

Since in Eq 3.3 we reduced the problem to a system of linear equations, the vector with unknown nodal voltages Θ and currents can be found by multiplying the inverted circuit matrix $\tilde{L}$ by the vector of known input voltages $\tilde{V}$ as is shown in Eq 3.6 where we replace the symbol of the inverse of matrix $\tilde{L}$ with Z:

$$\Theta = \tilde{L}^{-1}\tilde{V} = Z\tilde{V} \tag{3.6}$$

This simple linear operation has several important outcomes. First the voltage vector V, where all values are zero except a few input voltages, is multiplied by an inverse of the circuit matrix Z, which results in a vector Θ with nearly all elements being non-zero each reflecting magnitudes for node potentials and voltage source currents. This operation of small input to large output can be treated as expansion of dimensionality of the input vector V. Second since each nonzero element in the circuit matrix Z is a function of edge's nonlinear memconductance or the sum of memconductances (d elements along the diagonal), the projection of $\tilde{V}$ into Θ is accompanied by a series of rotation, reflection and scaling operations which can be obtained from a singular value decomposition (SVD) Eq 3.7 [Eckart and Young, 1936]. In the SVD the columns of U and Λ matrices form an orthogonal basis of the matrix M and the projection of a vector on them results in a rotation operation whereas diagonal matrix Σ represents scaling.

$$Z = U\Sigma\Lambda \tag{3.7}$$

Thus every input voltage vector V assigned to the input ports of the circuit is transformed linearly first by Λ rotation, followed by the Σ scaling and finalized by another rotation U. Finally, the values of Z matrix contain expressions formed by summation, multiplication and division operations between the magnitudes of the conductivities of memristive elements with memristance $M_{i.}$

$$\tilde{L} = \begin{pmatrix} \frac{1}{M_4}+\frac{1}{M_1} & -\frac{1}{M_1} & 0 & -\frac{1}{M_4} & 1 & 0 \\ -\frac{1}{M_1} & \frac{1}{M_3}+\frac{1}{M_1}+\frac{1}{R_1} & -\frac{1}{M_3} & 0 & 0 & 0 \\ 0 & -\frac{1}{M_3} & \frac{1}{M_3}+\frac{1}{M_2} & -\frac{1}{M_2} & 0 & 1 \\ -\frac{1}{M_4} & 0 & -\frac{1}{M_2} & \frac{1}{M_4}+\frac{1}{M_2}+\frac{1}{R_2} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{3.8}$$

Besides the fantastic complexity of arithmetic operations found in the Z matrix that are caused by the diversity of connections in the circuit, it also exhibits temporal nonlinearity of memristive elements which change resistance according to the magnitude, direction and previous history of applied voltages. The representation of the $\tilde{L}$ and it's inverse Z matrix for the circuit with six elements in Figure 3.7 (a) is shown in Eq 3.8 and Eq 3.9 correspondingly. Each non-zero element in the Z matrix from Eq 3.9 is a result of complex operations which depend on a subset of resistance (memristance) states of several elements in the system and thus constitute a reservoir of operations that can be harnessed for computation if an appropriate readout layer is applied, which assigns weight to particular output as it will be discussed further in sections 3.5 and 3.6. The size and connectivity of the network directly influence the diversity and complexity of the reservoir of these operations.

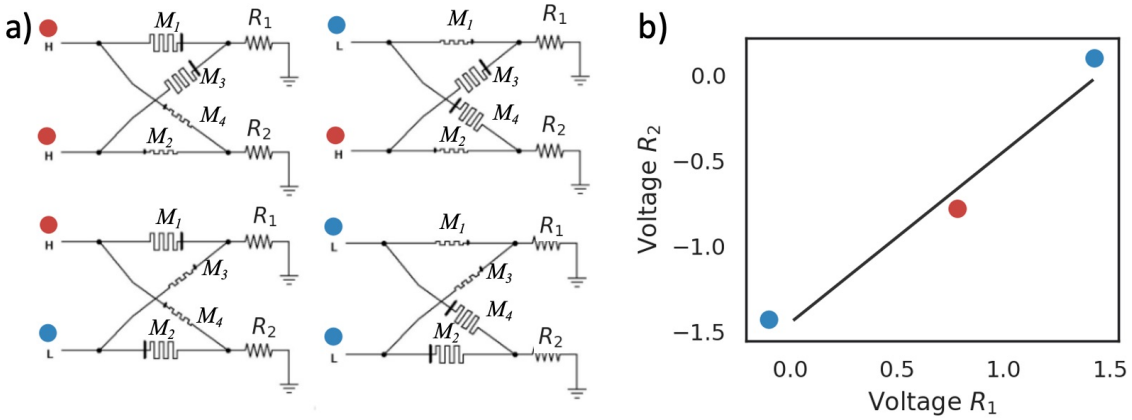


Figure 3.7: a) Simple network that can convert XOR problem into a linearly separable problem. Strukov's memristor model was used in this example with Ron/Roff=100/16000. L and H values represent to some + and - voltage values, respectively. Thick memristors represent high resistivity, whereas thin memristors represent high conductivity. b) The resulting mapping of the XOR table as a voltage reading on output resistors from the circuit

In the following chapter, we will discuss physical implementations of these networks as materials with embedded memristive nanowires. Our efforts to determine the architecture of embedded circuitry in these materials with micro and nano computed tomography will also be discussed. Despite the algorithmic simplicity of finding the circuit matrix operator in theoretical setups, for real macroscopic samples of materials with thousands (or even millions) of randomly dispersed nanowires, it is not possible to even approximate circuit

matrix. The primary difficulty in determining a circuit matrix is the lack of knowledge of all the network and device parameters, which affects the collective switching process. Thus the analysis provided in this section serves as convenient approach to understanding, rather than predicting, the switching phenomena in a particular memristive network.

$$Z=\tilde{L}^{-1}=\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & \frac{M_1M_3R_1}{M_3R_1+M_1(M_3+R_1)} & 0 & 0 & \frac{M_3R_1}{M_3R_1+M_1(M_3+R_1)} & \frac{M_1R_1}{M_3R_1+M_1(M_3+R_1)} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{M_2M_4R_2}{M_4R_2+M_2(M_4+R_2)} & \frac{M_2R_2}{M_4R_2+M_2(M_4+R_2)} & \frac{M_4R_2}{M_4R_2+M_2(M_4+R_2)} \\ 1 & \frac{M_3R_1}{M_3R_1+M_1(M_3+R_1)} & 0 & \frac{M_2R_2}{M_4R_2+M_2(M_4+R_2)} & -\frac{R_2(M_4R_1+M_3(M_4+R_1))+M_1(M_3+R_1)(M_2+R_2)+M_2(R_1(M_4+R_2)+M_3(M_4+R_1+R_2))}{(M_3R_1+M_1(M_3+R_1))(M_4R_2+M_2(M_4+R_2))} & \frac{R_2((M_3+M_4)R_1+M_1(M_3+R_1))+M_2R_1(M_4+R_2)}{(M_3R_1+M_1(M_3+R_1))(M_4R_2+M_2(M_4+R_2))} \\ 0 & \frac{M_1R_1}{M_3R_1+M_1(M_3+R_1)} & 1 & \frac{M_4R_2}{M_4R_2+M_2(M_4+R_2)} & \frac{R_2((M_3+M_4)R_1+M_1(M_3+R_1))+M_2R_1(M_4+R_2)}{(M_3R_1+M_1(M_3+R_1))(M_4R_2+M_2(M_4+R_2))} & -\frac{R_1(M_4R_2+M_2(M_4+R_2)+M_3(M_4+R_2))+M_1(M_4R_1+M_4R_2+M_2(M_4+R_2)+M_3(M_4+R_2)+R_2R_1)}{(M_3R_1+M_1(M_3+R_1))(M_4R_2+M_2(M_4+R_2))} \end{pmatrix} \tag{3.9}$$

3.4 Computing with memristive reservoirs

In order to show the computational utility of the memristive circuit projections discussed in the previous section, we will discuss several approaches in solving Boolean problems with these networks. We will define a class of binary classification problems and look at how these types of problems are solved in deep artificial neural networks.

The mechanism of most artificial feed-forward deep neural networks, in classification tasks, can be summarized as follows. The aim of neurons in each layer in an artificial feed-forward neural network is to collectively identify and extract significant features via nonlinear transformation and obtain a simpler representation for the downstream layers. This process continues while data is transformed and propagated from one layer of artificial neurons to another. The role of the final layer in the network is to apply linear decision boundaries to separate classes from each other.

Our study will start exploring computation tasks by learning how to replace primitive nonlinear logical gate with the memristive network. Out of all binary logic gates, the output of the XOR gate is linearly inseparable but can be constructed with four NAND, five NOR, or three mixed logic gates [Fletcher, 1980]. Moreover, the XOR gate also can be represented with at least three perceptrons or non-linear artificial neurons in a two-layer perceptron or a neural network configuration [Haykin, 2008].

To make it a bit more challenging we will also use multivariable XOR gates. The multivariable XOR gates have a truth table that can be constructed through consequent application of $\oplus$ (XOR) operator such as: $y = x_1 \oplus x_2 \oplus ... \oplus x_n$. The minimum number of neurons in a single layer neural network required to solve n variable XOR problem scales exponentially with the number of variables. However, deep neural networks are more efficient than single layer ones since the number of required neurons to solve n variable XOR problem scales linearly with the number of inputs [Haykin, 2008].

Thus, in this work, as an estimate of a computational capacity of a memristive network we take a network's ability to approximate logical functions (XOR gate) through linear separability of the output current from mapped input voltage representation of the truth table.

The smallest memristor networks that can linearize the XOR problem is shown in Figure 3.7 (a) where each input is mapped to each output via memristors Mn with opposite polarity. The boolean values in each row of the XOR truth table were converted into voltages, namely red dots correspond to the logical True and are represented by +5 V voltage at the input terminal. Conversely, blue dots represent logical False with the low -5 V voltage.

The system was simulated until each memristor reached its terminal state (non changing state). For simplicity of the interpretation, the memristors are plotted with thick and thin bodies to denote high resistance and low resistance states, respectively.

Figure 3.7 (b) shows the resulting mapping plot tells us that this simple memristive circuit network is equivalent to a single-layer neural network that simplified the linearly inseparable XOR problem into linearly separable output which can be further classified via a single linear perceptron. The surprising ability of such a small network to linearize nonlinear problems is solely due to the property of memristive elements that cannot be attained by other kinds of passive devices such as diodes, resistors or capacitors.

Mathematically the circuit in Figure 3.7 (a) converts input voltages into Hamming distance representation where the resulting output distance between different input voltage values is greater than zero and is equal to zero when inputs are equal to each other.

To understand the transformation caused by the simple circuit, in Figure 3.8 we provide visual proof of how the nonlinear XOR problem is converted to a linear one with the circuit from Figure 3.7 (a). In the example we use the following parameters for memristors M_n. For each memristor, ON and OFF state resistances are R_{on}=100 Ω, R_{off}=36 $k\Omega$, mobility μ and thickness of the device D are set to $\mu = 10^{-10}\,\frac{m^2}{sV}$ and $D = 10$ nm. All memristors were initialized with R_{off} state.

Each point in the grid of input voltages in Figure 3.8 (a) is color-coded to match the readout voltages shown in Figure 3.8 (b-h). In the experiment, each pair of input voltages were fed into the circuit and held for different durations, after which the voltage on the output resistors R_1 and R_2 were recorded and displayed in Figure 3.8 (b-h). Thus, the input and output voltage grids have red and blue circles that mark a variant of the XOR problem to show the dynamics of transformation of input voltages into the output currents. In other words, we take red circle as boolean True when the pair of input voltages are different, namely V_1=2 V, V_2=-2 V, and V_1=-2 V, V_2=2 V. Conversely, boolean False is marked with blue circles when input voltages are equal, such as V_1=2 V, V_2=2 V, and V_1=-2 V, V_2=-2 V. This configuration is one of the many ways an XOR problem can be represented.

The resulting mappings for the incremental input voltage hold times Figure 3.8 (h) start with 1 μs of holding input voltages. As can be seen from the plot, the projected voltage map is an affine transformation (shear and rotation) as memristors in the circuit behave like resistors since the hold time was insufficient for them to switch between OFF and ON states, hence the output is still linearly inseparable. As the input hold time increases to 1 ms, the transformation becomes nonlinear which is designated by bending the left side of the input Figure 3.8 (c). In the plot, the red (True) circle from the top left corner moved down relative

to other circles, yet insufficiently to draw a linear decision boundary. The complexity of the transformation is increased dramatically as the holding durations of input voltage is varied from 1 ms to 8.03 ms, as shown in Figures 3.8 (d-f), bringing the problem closer to linearly separable one. Finally in Figure 3.8 (g and h) red circle from the upper left corner (of the input voltage grid) moved sufficiently close to the other red circle (in the output voltage representation) that makes the XOR problem linearly separable. A similar approach was studied by Pershin *et al.*, where authors studied the effect of variability of memristor parameters on the resulting output voltage mappings, however, no applications to solving nonlinear problems were discussed [Pershin and Shevchenko, 2017].

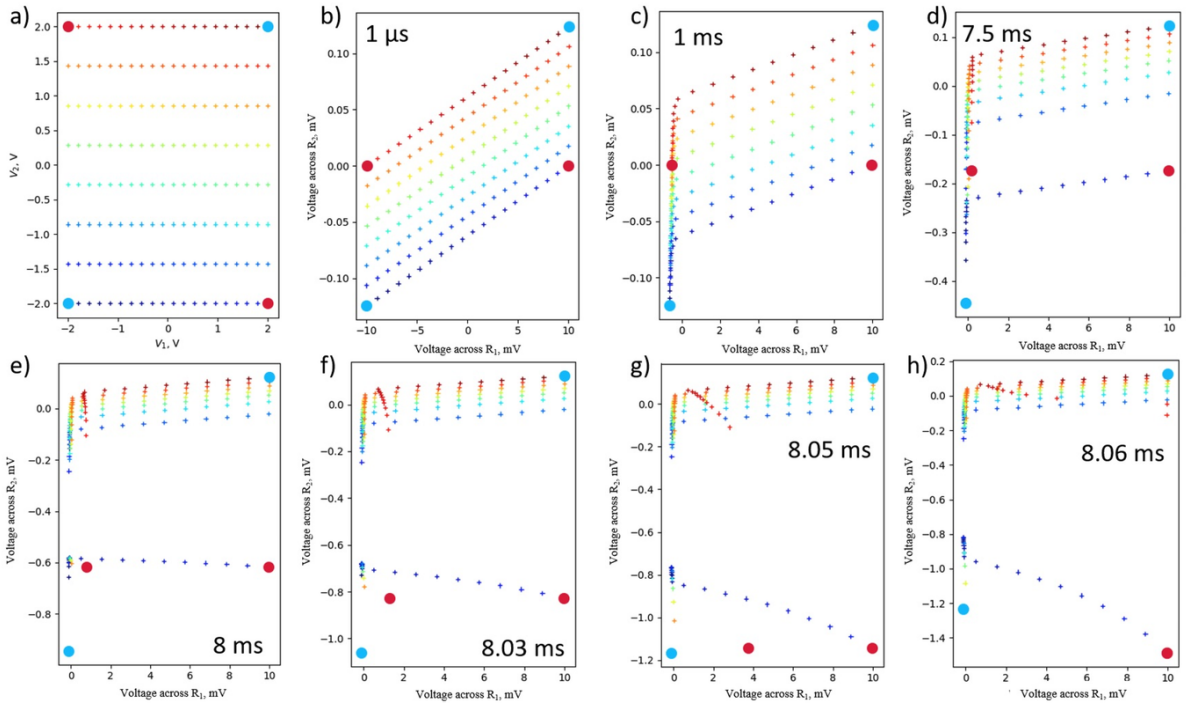


Figure 3.8: Observation of transformation of input voltages into output currents with a circuit from Figure 3.7. a) Grid of input voltages with red and blue circles marking a variant of linearly inseparable XOR problem. (b-h) Resulting in current readouts through the output resistors R_1 and R_2 at different input voltage hold time durations. The problem becomes linearly separable above 8.03 ms of input voltage hold duration.

3.4.1 Solving XOR with a large random network

Not every randomly generated circuit can be used to simplify complex problems. However, there are a few network parameters that can influence the success of conversion. In order to show how the size of the network and the number of memristors in it influence the mapping characteristics, in the next example we provide a mapping of a random small-world network generated with Watts-Strogatz algorithm [Watts and Strogatz, 1998] where each edge was replaced with a memristor with two randomly assigned pairs of inputs and outputs. Biological processes favor small-world networks because of the balance of short and long-range random connectivity that provides more efficient connectivity utilization than complete or random networks [Barabasi, 2016]. In the example, the network was generated with the following parameters: number of nodes n=50, probability of edge rewiring p=0.2 and the number of closest neighbors for each node k=4. The resulting graph was converted into a circuit, as is shown in Figure 3.9 (a), where each edge is represented by an HP memristor [Strukov et al., 2008b] configured similarly to the previous example in section 3.4, yielding 100 memristors in total. The mappings of voltages V_1 and V_2 were performed in a similar fashion as was described in previous example except the input voltage magnitudes were bounded between -1 and 1 V. The characteristics of the output current projections is slightly more diverse compared to a smaller network depicted in Figures 3.7 (with mappings presented in Figure 3.8). The projection of input voltages in Figure 3.9 (b) undergoes a nonlinear transformation as the duration of the holding voltages is increased from 1 us to 100 μs Figure 3.9 (c-e). The final plot in Figure 3.9 (e) shows that red markers moved sufficiently close to each other and far away from the line connecting blue markers, that permits drawing a line that can separate blue and red markers. In our experiments, we noticed that both small and large networks could be tuned to convert nonlinear problems into linear ones. In particular, circuits made of complete graphs or graphs with a large number of neighbors have a higher probability of providing nonlinear mapping circuits that can be used for linearizing nonlinear problems.

The most common projection pattern formed by any random memristive network, when one input is set to constant voltage and another is allowed to change, is a set of intersecting bent straight lines with one or several turning points. For example, in a 2D representation of voltages measured on R_1 and R_2 resistors, the interior angle formed by the turning point is always greater than 90° which a vast majority of randomly generated circuits (more than 80%) is indeed sufficient for converting nonlinear classification problem into a linear one without static control inputs.

As a side project in this study, we varied different parameters of the random circuit architecture, spanning from periodic lattice to small-world graphs. Based on these studies, no conclusions regarding the effect of network organization parameters effects on projection patterns can be drawn with a high degree of confidence. However, below we summarize qualitatively a few distinctive patterns that were observed to be the most influential in most network configurations we studied.

Networks with a prevailing number of local connectivities than long-range ones, such as small-world graphs or networks formed from complete graphs, expose the most diverse set of nonlinearity patterns and provide slightly more intersecting projection lines. On the other hand, circuits formed from a regular 2D lattice grid tend to create scattered lines of short bent projection lines.

Certainly the size of the network plays an important role in the diversity of output projections, although not the most significant role. As we showed large network with 100 memristors provide projections of similar nonlinearity by the small four memristor network, this matter needs to be studied in greater detail. Nevertheless, we conclude that the nonlinearity of projections depend on the relative placement of input and output channels with respect to each other whereas the secondary role is played by the size and topology of the network.

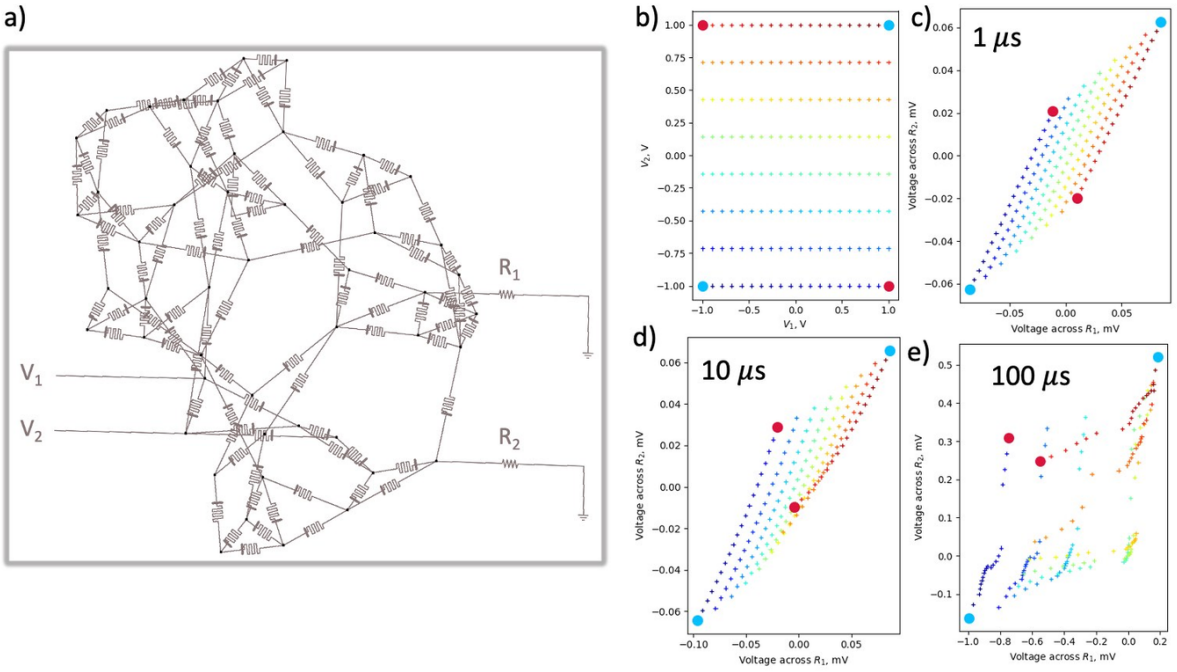


Figure 3.9: Voltage projections formed by a large random small-world circuit. a) the circuit constructed with Watts-Strogatz algorithm where the probability of rewiring p=0.2 and the number of local connections k=4, n=50 nodes (100 memristors, 2 inputs, and 2 outputs). b) input voltage map c) projected voltage map measured as the voltage on output resistors R_1 and R_2 after 1 us of holding. d) Projection after 10 us of holding input voltages e) linearly separable projection after 100 μs of holding input voltages.

In the following section, we will discuss a variant of evolutionary optimization study. We introduce static control voltages supplied to the circuit during the projection stage and the values for the main input to tune the circuit and provide the desired output. We will argue that, indeed, within a reasonable range it is possible to tune an output of an arbitrary random network using evolutionary algorithms.

3.4.2 Training random memristor reservoir to solve XOR problem

As we mentioned in the previous section, not every random circuit can be used to linearize complex classification problems. In this section, we show that using a set of static control (or "teacher") voltages on the circuit along with varying input voltages severely increases the diversity of the mapping function that can be used to simplify complex tasks. For example example, Figure 3.10 (a) shows a random small-world circuit (n=50, k=4 and p=0.2) with

two voltage inputs V_1, and V_2, two outputs R_1 and R_2 and six control voltages C_1, C_2, C_3, C_4, C_5, and C_6.

At the beginning of each iteration step, the circuit is initialized and a set of control voltages C_n suggested by an optimization algorithm in Optuna library [Akiba et al., 2019] are drawn. Next, the control voltages are set on the circuit while the input voltages V_1 and V_2 are pulled from a truth table where True and False inputs are represented by 10 and -10 V. To simulate the effect of input noise and improve stability, the classification table was populated with 8 copies of XOR truth table, 32 rows total, where also a 10% of uniformly distributed noise was added to each input value. After 1 ms of holding input and control voltages, the resulting voltages on the output resistors are read and appended to an output table. The circuit is then reset and a new pair of input voltage values are drawn from the truth table while the control voltages remain similar to the ones obtained at the beginning of the iteration. The process continues until all 32 values of the input voltages are drawn, and the output table is populated with 32 rows.

The score of the iteration is calculated with logistic regression as a fraction of correct classifications. In other words, if all 8 truth tables were classified correctly, the resulting score would be 100%. However, the score cannot be less than 50% because even in the worst configuration, it is always possible to find a decision boundary that can separate two classes evenly.

The results of the first 30 iterations are shown in Figure 3.10 (b) where the top bar chart depicts the relative magnitudes of the control voltages. The bottom chart shows the classification score obtained at each iteration step. As can be deduced from Figure 3.10 (b) first 9 iterations Optuna suggestion algorithm tries a variety of random values that yield scores of 50%. At iteration 10, the score is increased to 75% perhaps due to all negative values except C3 and C4. Thus, the scores in all iterations are around 50 % until iteration 19 when the resulting score 100%. Also, in the top image of Figure 3.10 (b), Optuna's suggestions do not change significantly after the 100 % score was reached at the 19th iteration.

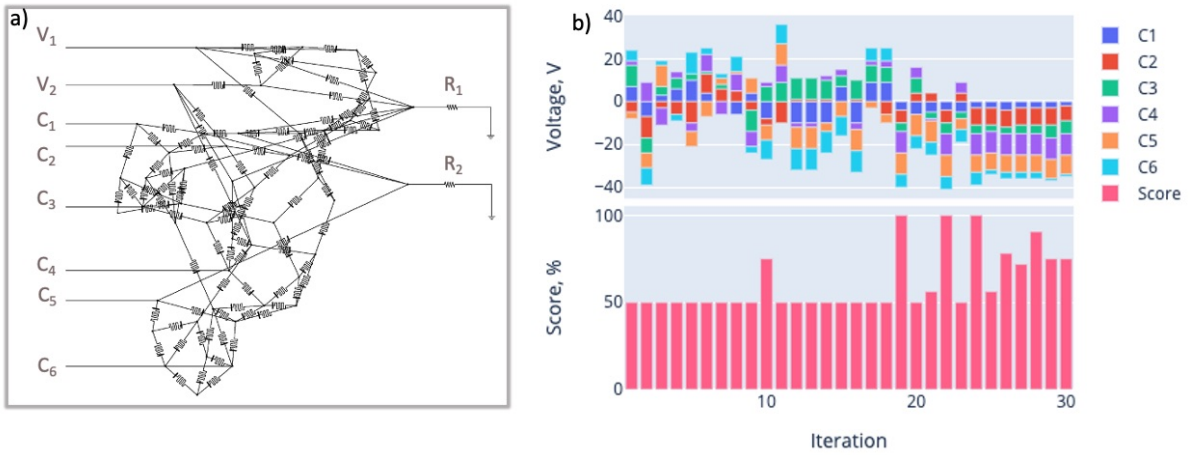


Figure 3.10: Evolving random memristive circuit with control voltages to linearize XOR problem. a) Random small-world circuit with 2 inputs and outputs and 6 control voltages. b) (top) Relative bar chart of the evolution of the magnitudes of control voltages. (bottom) Classification scores correspond to the configuration of control voltages at each iteration step.

Figure 3.11 shows the influence of the input control voltage on the resulting mappings. In Figure 3.11 (a), the input voltage grid is similar to the examples in section 3.4, while (b-d) shows a snapshot of projections for the input voltages and control voltages according to the specified iteration. The inset in Figures 3.11 (b-d) results from the XOR experiment, where a PCA projection and a sample logistic decision boundary separate red circles from blue ones. Clearly, at iteration 19 not only the grid of voltages is modified that visually makes this configuration applicable to linearizing XOR problem, but also, according to the inset plot where 10% noise was added to the input, the decision boundary completely separates the classes (note 10% noise is added to each pair of voltage inputs, but not the control voltages).

Analogous measurements were performed on multiple circuits with similar configurations all showing similar successful results. Importantly, in none of the other cases, we noticed an abrupt transition of the mapped points. This observation suggests, that the mapping function itself is differentiable, and one could potentially apply conventional gradient descent optimization techniques for finding optimal solutions [Press, 1992].

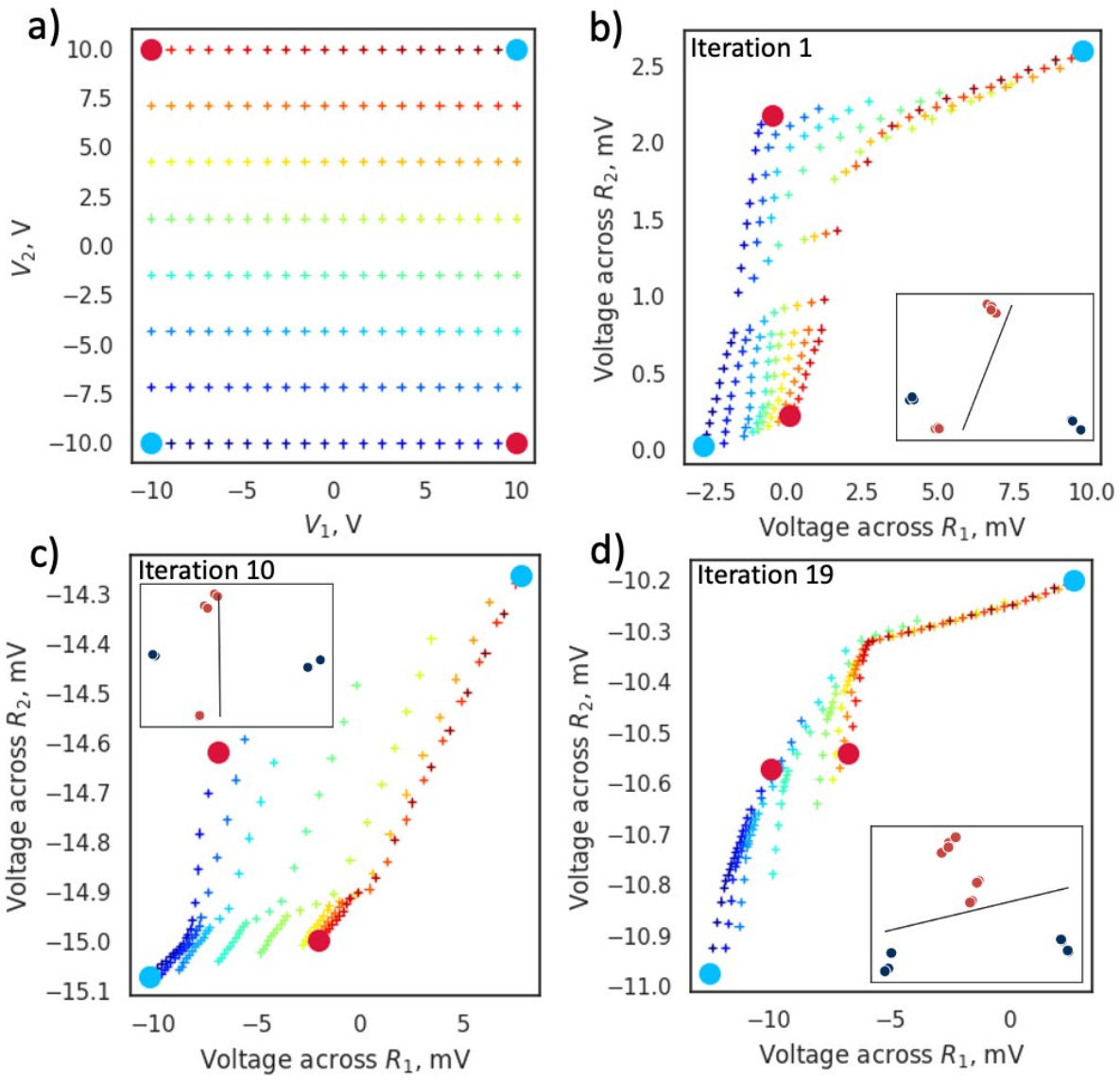


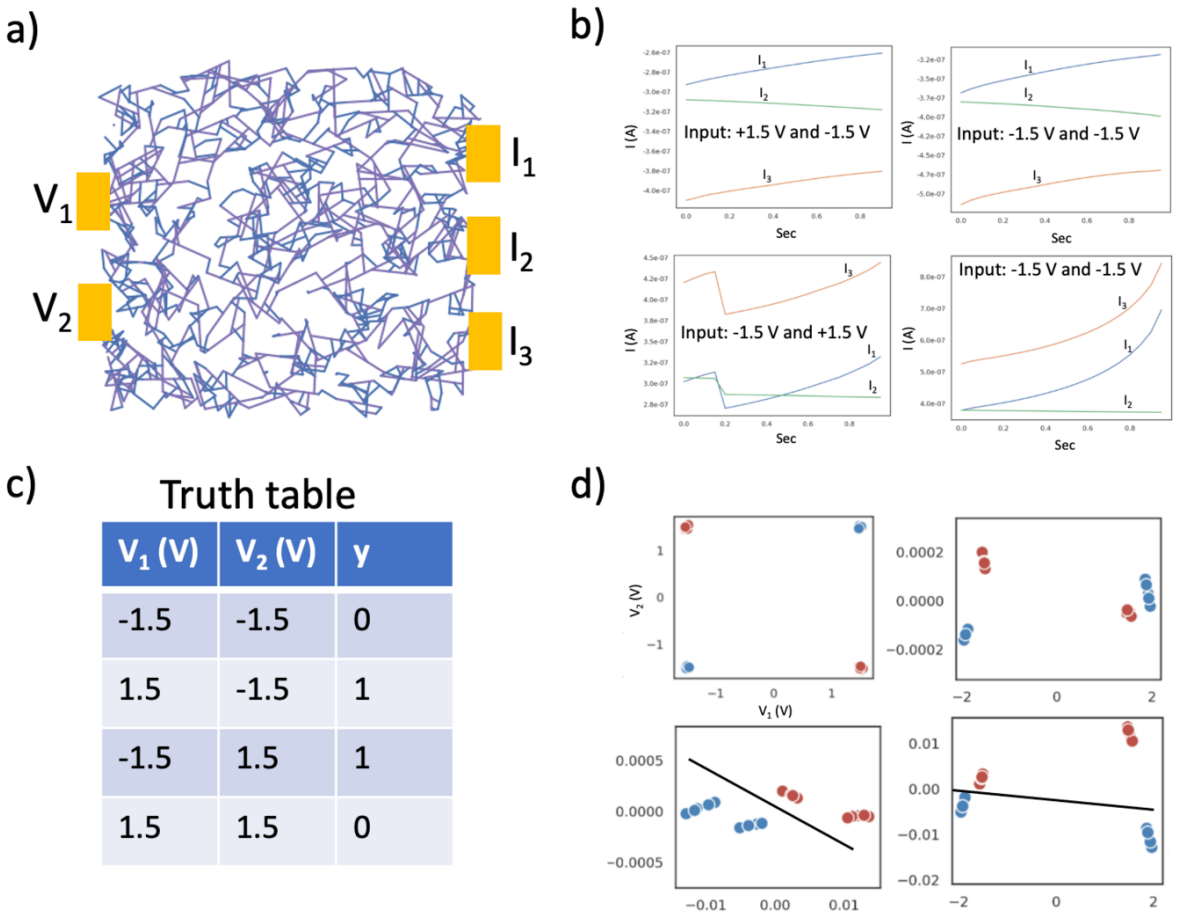
Figure 3.11: Evolution of input voltage projections for different control voltages configurations in a random circuit from Figure 3.10. a) Grid of input voltages between -10 to 10 V. Red and blue circles denote ends of the square formed by the corners of the grid to visually aid the process of mapping b) representation of projected voltages at the first iteration (top and bottom of Figure 3.10 (b)). Inset shows the ability of the circuit to linearize the XOR problem (repeated 8 times with 10 % noise added to input voltages). c) similar for iteration 10 d) 100% score at iteration 19

This section showed the impressive properties of memristive random networks and discussed approaches in computation applications. While in all examples above we used a standard HP model, where each memristor had the same characteristics. Since our goal

is to evaluate this approach in composite systems, formed by high aspect ratio resistive switching wires (that will be discussed in Chapter 4), such as Ag_2S, we cannot assume that each memristor can be described by identical parameters. Thus in the following section we will make the next step in our study of random networks by evaluating more realistic composites in which memristive circuits are constructed from the percolation of randomly distributed Ag_2S nanowires on a surface and in 3D.

3.5 Simulating memristive nanodevice composites

The previous section demonstrated that random circuits made of similar memristors have enormous capabilities for changing the representation of the feature space of the input signal via projecting it to a continuous collective state. We are adding control channels that provide flexibility of evolving the networks and allow tuning of the output representation. Since our primary goal is to develop composites for non-conventional computation, in this section, we continue this exploration but with memristive circuits and elements assembled from randomly distributed Ag and Ag_2S switching nanowires. To simplify the search of optimal configuration, we will use an increased number of output channels as is suggested by Cover's theorem, which is defined as follows: "A complex pattern-classification problem, cast in a high-dimensional space nonlinearly, is more likely to be linearly separable than in a low-dimensional space, provided that the space is not densely populated." [Cover, 1965].



V_1 (V)	V_2 (V)	y
-1.5	-1.5	0
1.5	-1.5	1
-1.5	1.5	1
1.5	1.5	0

Figure 3.12: An example random network system (800 memristors total) with 2 input electrodes and 3 output electrodes for solving the XOR problem. Ron_pnm=1, Roff_pnm=1000, mobility=2.56e-9, nw_res_per_nm=0.002, t_step="1e-5". b) stimulation of the network with logical input values from the XOR table converted into +/- 1.5 V where 1 corresponds to 1.5 V and 0 corresponds to -1.5 V. Each line corresponds to one out of 3 output channels. c) XOR truth table used in the simulation where for the input voltages Truth and False are replaced with 1.5 V and -1.5 V correspondingly d) 2D representation of XOR problem (top left) where blue dots and red dots represent 0's and 1's respectively. (top right, bottom left and bottom right) are the projections of the current representation of the input data. Two out of three of these projections are linearly separable.

In order to solve a simple XOR problem with a random network made of switching nanowires, first, a random network with two types of components (Ag and Ag_2S nanowires) is generated with the Percolator software [Frick and LaBean, 2021b] discussed in Chapter 2 and is shown on Figure 3.12 (a). In this work we are modeling a device made of scattered Ag_2S (memristive) and Ag (resistive with linear resistivity (linear resistivity $\sigma = 5\ \frac{\Omega}{\text{nm}}$) devices on

a 2D surface. Thus a network comprised of 800 memristive wires of 30 μm length, modeled from HP memristor [Strukov et al., 2008b] and described by equations 3.1 and 3.2. And 400 plain resistive devices of 60 μm length with linear resistivity of 5 $\frac{\Omega}{\mu m}$. The architecture and dimensionality of the memristor wires are taken similar to the ones we produced in our laboratory, which will be discussed further in chapter 4.

In the simulator, the diameter and distance cutoff (proximity radius) for both fillers was set to 0.1 μm and 5 nm. The term proximity radius was discussed in chapter 2 and denotes an imaginary shell around a wire used to determine connectivity between any object that falls within the range of this radius.

The network in Figure 3.12 (a) provides connectivity to all 5 electrodes depicted, out of which 2 serve as voltage input (V_1 and V_2) and 3 for current output (I_1, I_2 and I_3). Further, 80% of the edges of the network are converted into memristors, whereas the remaining 20% of the edges are resistive wires. The resistance and memristance magnitudes are defined by the length of the wire and simulation parameters. The following parameters are used for the resistive filler: Rr=2 mΩ/nm, and for the memristor: Ron=1 Ω/nm, Roff=1 kΩ/nm with Ag^+ ion mobility $\mu = 1x10^3 \frac{\mu m^2}{sV}$. Linear junction resistivity between wires R_j is used in these experiments to express real conditions and is set to R_j=500 Ω/nm. At the beginning of the experiment, each memristor's state is initialized to an intermediate value $\omega = \frac{L}{2}$.

The response of this system to sample DC voltage is shown in Figure 3.12 (b). Due to the memristive nature of the network and polarities of devices, the current response is nonlinear in time and has different directions of slopes along with final amplitudes. The difference in magnitudes of amplitudes at the end of these responses enables these random networks to project complex 2D problems into simpler 3D problems.

Figure 3.12 (d) in the top left plot shows input voltage of the V_1 and V_2 columns from XOR table Figure 3.12 (c) where input "0" (red) was converted into -1.5 V and input "1" (blue) is transformed into 1.5 V. As in the previous section, 10% normalized noise was added and each pair of inputs was fed repeatedly 8 times that results in slight deviation of the response current during each iteration. The remaining three plots in Figure 3.12 (d) show PCA projections of the 3 current output channels through grounded 100 Ω resistors (not shown) recorded after 1 ms of holding each pair of inputs. Remarkably, two out of three PCA projection plots show clear linear separability, which supports Cover's theorem that increasing dimensionality of the nonlinear transformation increases the probability of simplifying complex problems.

3.5.1 Simulating nano-device composites with static control voltages

In the previous section we showed an example how two variable XOR problem can be solved with a random memristive network with two inputs and three outputs. In this section, we increase the complexity of the problem to three variable XOR to be solved on a random network with three inputs and outputs and with iteratively updated static control voltages. Figure 3.13 shows an example experiment where three variable XOR problem was solved with 100% accuracy on a planar grid network with randomly oriented memristors. Similar to previous examples, we use the non-volatile HP memristor model [Strukov et al., 2008b] from Eqs 3.1 and 3.2. In this experiment, the planar random network was surrounded by 12 electrodes where 3 were used for voltage input, the other 3 for current readout, and 6 for control bias. A total of 100 independent experiments were performed where voltage representation of 3D XOR truth table variables $\vec{V} = (v_1, v_2, v_3)$ were fed into the network line by line 5 times consecutively to harvest statistics, and corresponding current response $\vec{I} = (i_1, i_2, i_3)$ was read for each iteration as a new feature dataset yielding a total of 5 (repetitions)x8 (rows in the truth table)=40 data points. A logistic regression classifier was trained to linearly separate each class in the new dataset where each $\vec{V}$ row is replaced with corresponding $\vec{I}$ response with respect to y column in the original 3-variable XOR truth table. The resulting classification score was recorded after each experiment for further optimization in the control vector $\vec{V}c = (V\,c_1, V\,c_2, V\,c_3, V\,c_4, V\,c_5, V\,c_6)$. Bayesian optimization library Optuna [Akiba et al., 2019] was used to set static control voltages vector $\vec{V}c$ to maximize XOR classification accuracy score Figure 3.13 (c). Figure 3.13 (d) and (e) shows PCA projections of the output current without and with calculated control bias applied correspondingly, displaying the system's sensitivity to static control perturbations.

a)

V_1 (V)	V_2 (V)	V_3 (V)	y
-1.5	-1.5	-1.5	0
-1.5	-1.5	1.5	1
-1.5	1	-1.5	1
-1.5	1	1	0
1	-1.5	-1.5	1
1	-1.5	1	0
1	1	-1.5	0
1	1	1.5	1

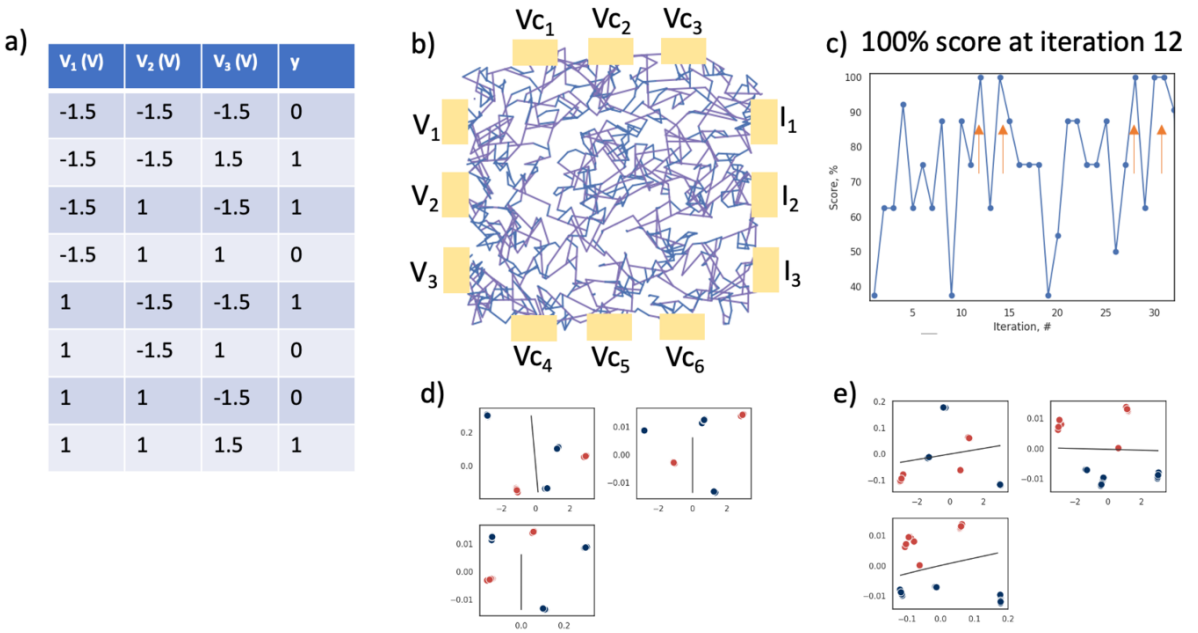


Figure 3.13: Solving 3D XOR problem through mapping voltages into currents with memristive network. a) 3 variable XOR truth table used in the experiment where each value is converted into voltage. b) grid network with resistors (purple), memristors (short blue segments) and 12 electrodes for input, output and control. output and cont. c) The trend of classification scores during optimization process where for each experiment, a vector of control voltages $\vec{c}$ is chosen with a sequential model-based algorithm with an objective to increase accuracy of subsequent experiments. d) Top 3 high variance PCA projections of resulting output currents where control voltage vector $\vec{c} = \vec{0}$ and e) where is optimal and $\vec{c} \neq \vec{0}$. Red dots correspond to $y = 1$ and blue ones to $y = 0$ labels.

Following this experiment, we discovered that not only is this network capable of solving 3 input to 3 output XOR problems, but also it is capable of solving more complex configurations, namely 3 input and 2 output XOR. In this configuration, only I_1 and I_2 outputs were used and I_3 is left disconnected. Similarly, we were able to find bias configuration for 2, 3 and 4 bit XOR problems with 2,3, and 4 dimensional outputs for each of the configurations with a 100% score. This surprising result means that the memristive network's internal states are sufficiently rich to reduce the complexity of the problem into lower dimensions during the optimization stage. In other words, temporal dynamics of the memristive networks and it's elements behave like a tunable nonlinear filter.

Also, the success rate depends on the choice of the proper signal and the duration of the signal being held at the input terminals. The degree to which these networks can be tuned depends on multiple parameters, the R_{on}/R_{off} ratio of memristive elements, the topology of

the network and the location of the input, output, and control electrodes. Accounting for the mobility and size of the system, in all of the above experiments, the input and control signal's time varied between 1 ms and up to 1 s for voltage signals between 1 and 10 V. Ideally one needs to find optimal parameters by trial and error, or by using Bayesian optimization methods such as Optuna [Akiba et al., 2019] or Hyperopt [Bergstra et al., 2015].

3.6 Handwritten digit recognition

The primary objective of this section is to investigate how random memristive networks can be tailored to increase the accuracy in handwritten digit recognition tasks. The secondary objective is to explore various memristive network architectures with different degrees of randomness and, in the first approximation provide comparative analysis in their performance in handwritten digit recognition.

Recently Du *et al.* conducted a study on the classification of MNIST handwritten digits with a reservoir of 88 isolated WO_x memristors selected from a 32x32 cross-bar array [Du et al., 2017b]. The WO_x memristors produced bytheir group had volatile or short-term memory behavior that resulted in gradual resetting of the state over a short period of time (on the order of a few seconds). This kind of short-term memory is a valuable property of memristors that is important in finding temporal correlation in the input signal [Moon et al., 2019].

In their experiment, 60000 greyscale MNIST images, 28x28 pixels each, were binarized via threshold filtering and rescaled to 22x20 pixel images to eliminate non-informative pixel boundaries. The resulting downscaled images were fed into 88 memristor reservoir at two different rates where each memristor received values from only five adjacent pixels (i.e., 22 rows 4 sections per row and 5 pixels per section, and 2 different rates). The final state of each memristor was fed into a software-implemented logistic classifier with 176x10 parameters that were trained on the representation of 2000 images. This procedure of splitting memristors into zones, each receiving different portions of the image as a signal allowed them to obtain a classification accuracy of 88.1%. When only one rate was used (88x10 parameters), the accuracy fell to 85.6%. This remarkable result shows that even a small reservoir is can reduce the complexity of a complex task that can further be solved, with reasonable accuracy, by a relatively small neural network of 880 parameters.

In another exciting study, a volatile variant of a reservoir consisting of Ag-doped SiO_x memristors implemented on a cross-bar array was used to achieve 83% accuracy in MNIST

handwritten digit recognition [Midya et al., 2019]. In the work, each image was similarly converted into binary representation and cropped to 22x20 size, after which it was vertically re-stacked to form a 110x4 image that was fed into 110 diffusive memristors selected from the cross-bar array. The resulting final state of each of the memristors was obtained as a current response to the read voltage pulse fed to the readout layer. Instead of using a software-based readout layer, the authors constructed a hardware implemented readout layer, made of a 110x10 memristors each paired with a transistor to allow tuning of resistance of each memristor.

In order to appreciate the benefit of the reservoir in handwritten digit recognition in the above cases, here we will perform similar classification tests on a logistic readout layer classifier without the reservoir. In particular, a simple logistic classifier network from Scikit Learn library [Buitinck et al., 2013] with no hidden layers and default solver [Morales and Nocedal, 2011] was trained on MNIST handwritten digits dataset with accuracy similar or better than the above results. In each particular case MNIST digit was binarized, cropped and resized to a 20x20 image. Since the role of the reservoir is to compress sequences of data with minimum loss, the logistic readout layer has reduced the number of the inputs and therefore cannot accept all 400 pixels of the image. Instead, the images were restacked similar to the above reports into three different configurations, 400x1, 100x4, and 80x5 matrix so that only the last column was used as an input into a corresponding logistic classifier, with either 400x1, 80x10, or 100x10 weights. The results of the experiment are summarized in Table 3.1.

As is shown in Table 3.1, a 400x10 classifier network trained on the whole portion of the preprocessed 400 pixel image reached 88% accuracy when trained on 60000 and tested with 10000 images (60k/10k train/test). The accuracy drops 1% each time the amount of training and testing data is reduced either from 60k/10k to 40k/5k or from 40k/5k to 10k/2k train/test size. However, if only every 4th pixel is fed into the classifier with 800x10 weights, the accuracy drops to 81% on a 60k/10k train/test handwritten digit datasets or 76% on a 10k/2k train/test dataset. This notable difference in the accuracy is attributed to the tremendous loss in the informative portion of the data as only a quarter of the input data was used to train and test the classifier. Even lower accuracy was obtained on an 80x10 classifier when only every 5th pixel is being in each image resulting in the highest 75% accuracy for 60k/10k train/test and the lowest 69% accuracy in 10k/2k train/test.

The increasing difference between the adjacent accuracies for each pixel step at different train/test dataset sizes highlights the ability of each network to generalize. Thus in the 1 pixel step, despite the high value of the accuracy, the difference between minimum and

	10k/2k	40k/5k	60k/10k
Acc. @1 px step	86%	87%	88%
Acc. @4 px step	76%	78%	81%
Acc. @5 px step	69%	71%	75%

Table 3.1: Accuracy for logistic classifiers trained on pre-processed (cropped and binarized) MNIST digits with different amount of train/test data (columns) and sparsity of data (rows). Accuracy at @1 pixel step represents performance with pristine data and 4000 parameters learned in the logistic classifier. Accuracy @5 pixel step means that only every 5th pixel was used from each image and only 800 parameters were trained in the readout layer

maximum is only 2% and the difference between adjacent accuracies at different train/test sizes is 1%. This means that the classifier network has an adequate basis to generalize even with a smaller train dataset. On the other hand, in 4 and 5 pixel step configuration, the difference between the minimum and maximum accuracies is 5% and 6% correspondingly. In both cases the increased accuracy climbs 2% between 10k/2k and 40k/5k train/test datasets, and 3% between 40k/5k and 60k/10k train/test datasets which means that in these cases the logistic classifier needed more data in order to generalize and score higher.

Thus with only 800 parameters, and every fifth pixel is taken from the preprocessed image, the network gives around 75% accuracy (with 10000 training samples and 1000 test samples, default (LBFGS)[Morales and Nocedal, 2011] solver and standard scaling of the inputs. A network with 100x10 parameters where only each 4th pixel from the 20x20 handwritten digit is used provides an accuracy of around 80%. An additional 10% of accuracy can be added if one uses a 400x10 network yielding maximum accuracy of around 90% for the total of 4000 parameters. Thus the real benefit of using an isolated memristor reservoir in an image classification problems might be a lower than CMOS power consumption rather than higher accuracy, and a more comprehensive statistical study is needed to compare side by side conventional methods with memristor reservoir.

Unlike the above approaches where the reservoir was constructed from isolated memristors selected from a cross-bar array, our approach relies on a network of randomly connected memristors of different network architectures in order to potentially provide a much larger space of possible representation states due to correlation between neighbor inputs.

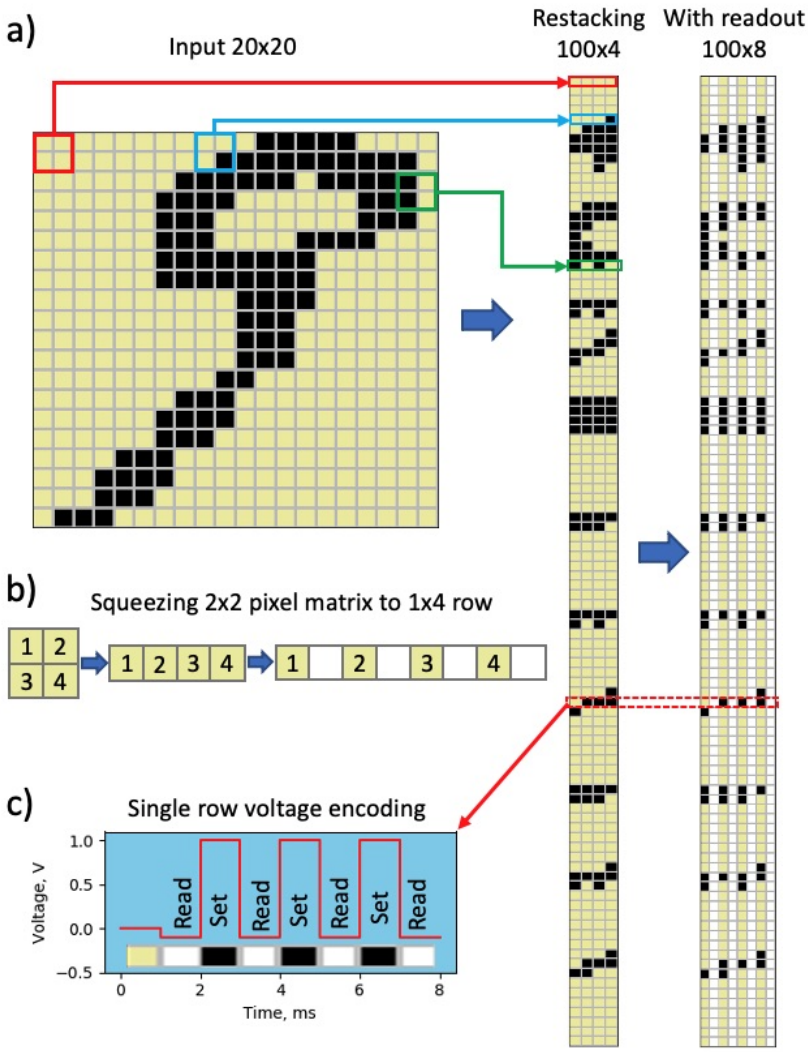


Figure 3.14: Block-coding a preprocessed MNIST digit and sending it to a reservoir as a train of voltage spikes. a) A 20x20 digit from a preprocessed MNIST dataset digit is broken down to 2x2 squares and according to the visual rule in b) are further flattened in to a 1x4 array resulting in a 100x4 matrix. Readout columns (white pixels) are inserted after each column to produce a 100x8 matrix that represents the digit. c) The, yellow, black and white pixels in the input matrix are further encoded into 0, 1 and -0.1 V while being fed into a reservoir with a 1 ms period.

In this comparison, we looked at the classification performance of networks with the different organizational architecture of connectivity that could represent various natural phenomena. However, we will only present results for only two kinds of network configurations, random and ordered, that we identified as the most relevant while others did

not display any unique behavior, at least with respect to particular tasks. As a first network architecture for the reservoir, we used a random network, square lattice and a linear array of isolated memristor architectures. The random sparse network reservoir had 97% sparsity and was constructed from a complete graph of 300 nodes where 97% of edges were removed Figure 3.15 (a). The reason for choosing this kind of network was inspired by Echo State (ES) networks first studied by H. Jaeger [Jaeger, 2004] . In a typical ES reservoir, neurons are connected by a sparse random network, with sparsity being set to 97%. The sparsity is required to facilitate loosely coupled correlation between states of the reservoir.

Interestingly, the sparse random network has some similarities with the small-world networks, which itself is a hybrid between ordered lattice and random networks where long-range connectivity is combined with dense local connections to neighbors [Watts and Strogatz, 1998]. Biological brains resemble small-world connectivity, as well as this architecture, allow efficient communication, that allows the signal to be propagated with the least number of steps compared to completely random or lattice networks [Kawai et al., 2019]. In particular, in 2006 Bassett *et al.* performed a magnetoencephalographic study on 22 healthy volunteers and showed that during simple motor tasks the neural networks of their brains exhibited both local and long-range connectivity [Bassett et al., 2006]. The similarity between these particular configurations of random sparse and small world networks lies primarily in the same average shortest path. In particular average shortest path in a 300 node small-world graph with the number of neighbors k=4 and probability of rewiring p = 0.1, and sparse random networks with sparsity of 97% is ~7 and ~2.3 correspondingly. While the average shortest path is somewhat different, our preliminary studies showed that sparse and small world networks have similar performance and hence going forward we gave our preference to a random sparse networks configuration.

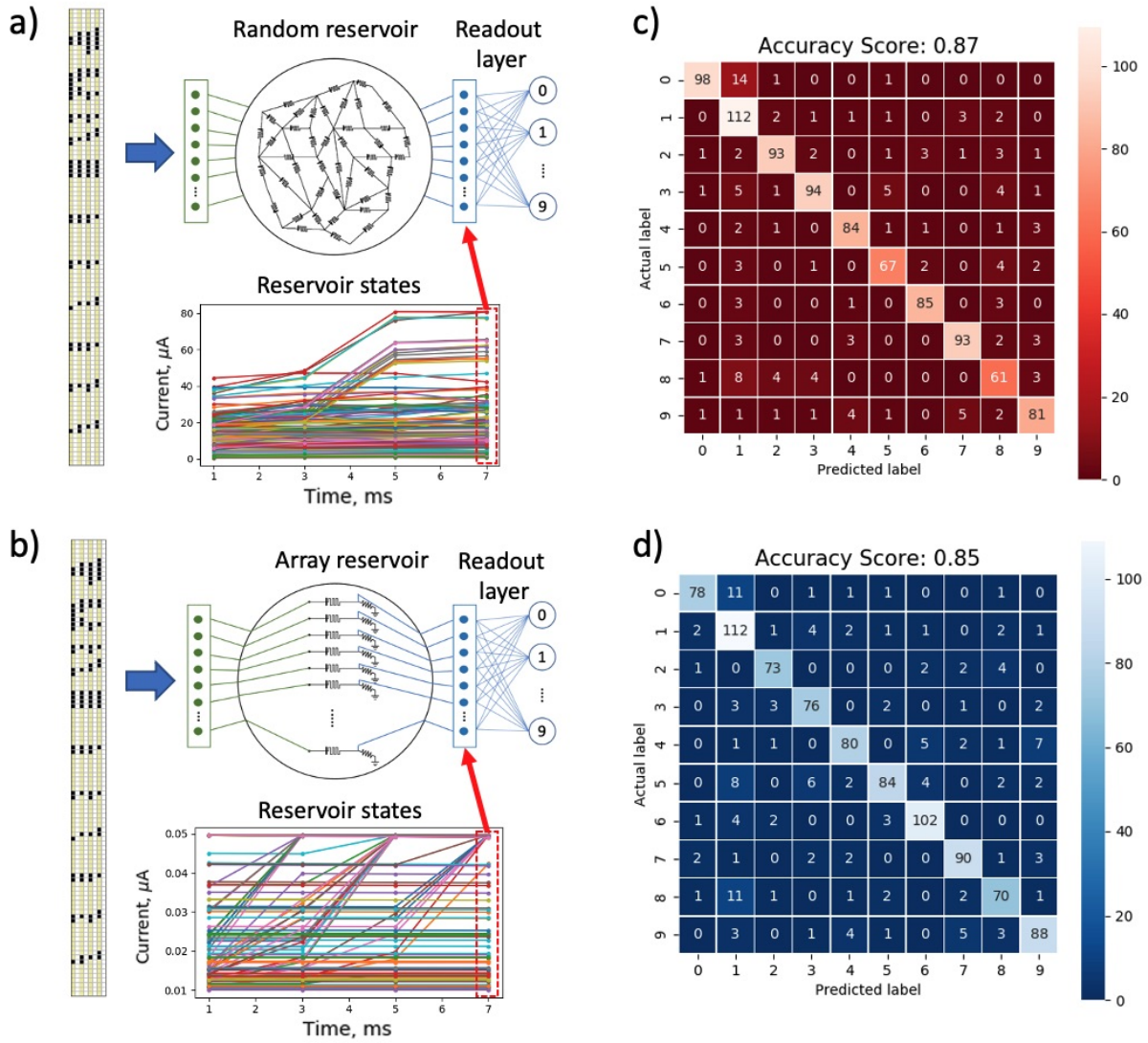


Figure 3.15: The procedure of performing handwritten digit recognition with reservoirs. A representation of a digit (see Figure 3.14) is fed into a) a random sparse reservoir with 97% sparsity) and b) a deterministic reservoir with 100 similar memristors each providing 100 inputs and 100 outputs to match the number of rows in the input matrix. Each column from the dataset is fed with 1 ms interval, making it a total 8 steps including readout signals. The resulting reservoir states in a) and b) represent the current values obtained during the readout portion of the input matrix, i.e. current values obtained during -0.1 V (white pixel) input. The obtained current mappings from each reservoir are rescaled and fed into a readout layer (logistic regression classifier) with 100x10 parameters that is trained to classify each digit. Confusion matrix displaying the accuracy of obtained classification results for each reservoir are shown in c) and d) with the random reservoir scoring a few percent higher than an array reservoir.

Besides random sparse and small-world networks, we also looked at square lattice networks. The square lattice networks are interesting because they represent an approximation of equal-sized rod-like memristors distributed on a 2D surface. However, it is quite challenging to implement a self-assembled lattice networks at the nanoscale. In our preliminary study, a square lattice network was constructed via repeated copying of a 2D square pattern resulting in a 30x30 node graph with 1740 edges. The average shortest path in such a network with 900 is L=20 which is much larger than random-sparse networks and means that on average there are at least 20 edges separating any two randomly chosen nodes. However in our preliminary study, we found no noticeable difference in the performance in handwritten digit recognition task of a square lattice networks compared to a small-world and random sparse memristive networks.

Finally, as a primitive yet powerful benchmark network configuration, in Figure 3.15 (b) we used a primitive array of equivalent isolated memristors similar to the ones reported by Du *et al.* and Midya *et al.* [Du et al., 2017b, Midya et al., 2019].

The procedure of producing a circuit from a graph is described as follows. From each of the generated network configurations a circuit was constructed by replacing each edge in the network with an HP [Strukov et al., 2008b] memristor model with Ron=20 $k\Omega$ and Roff=100 $k\Omega$ device thickness d=30 nm and width of the doped region set at midpoint w=15 nm. The mobility μ in each memristor was set to $\mu = 10^{-10}\ \frac{m^2}{sV}$. Once the circuit is constructed, the nodes for inputs and outputs are randomly selected (except linear array configuration where each memristor serves as input and output).

The MNIST handwritten digit dataset consisting of 60000 digits was preprocessed by clipping unused boundaries and resizing to a 20x20 pixel images. Randomly selected 10000 images were used for training the readout network while other 1000 randomly drawn images were left for testing. Finally each of the preprocessed image was broken down to a 2x2 square matrices that were flattened and stacked on top of each other forming a 100x4 matrix. Further, the stacked matrix was augmented with additional readout columns that followed each of the columns producing a 100x8 matrix as is shown in Figure 3.14. Since HP memristors are non-volatile, the inverse polarity of the readout columns was used to induce volatility, or forgetting of the previous state of the memristor. Importantly only the current magnitude obtained from the last white column is registered as the reservoir state while preceding readout columns serve as a volatility inducing signal. It is worth noting that alternation of data and readout signals can be used to significantly accelerate the rate of computation on memristors that naturally do not have volatility like $TiO_{2\text{-}x}$.

Figure 3.15 shows the resulting current readout states obtained during the last white

column in the input matrix is fed into one of the reservoirs. The states in the random reservoir, Figure 3.15 (a), are more diverse than those of the ordered array reservoir Figure 3.15 (b). As a result, the accuracy of the random reservoir is slightly higher than the one obtained with an ordered reservoir Figure 3.15 (c) vs Figure 3.15 (d). The cause of the three orders of magnitude difference in measured current in Figure 3.15 (a) and (b) is that in the random network typically has a lot of parallel connections that increase cumulative current compared to a single memristor.

While the results reported in Figure 3.15 require an in-depth statistical analysis, our exploration shows that in the given handwritten digit recognition task, with an equivalent testing conditions, random reservoir performs at least as good as a linear array, namely namely 87% vs 85%. These results are supported by previous results reported independently by Du *et al.* (max accuracy 88% with software 176x10 readout layer) and Midya *et al.* (max accuracy 83% with hardware-implemented 110x10 readout layer) [Du et al., 2017b, Midya et al., 2019]. In the vast majority of our explorations, random reservoir performed better than a linear memristive array and in some cases accuracy was equal.

While our theoretical observations show that cross-bar array based reservoir is not better than a random memristive reservoir, another interesting difference between these two reservoir configurations is in the diversity of the output states. In the random reservoir, high connectivity makes some outputs less sensitive to the inputs while in the linear memristive array each memristor is "listening" to the data from a single row in the input matrix. The "cross-talk" between inputs in the random array provides a way for a lossy compression mechanism. In particular, based on this property we found that by removing 50% of the reservoir states with the lowest variance in the random reservoir only reduced the accuracy to 85%. Thus at the expense of a small accuracy reduction a random reservoir with 100 input channels can have only 50 output channels (selected based on large variance in the state) and with only a 50x10 readout network perform just as good as a 100 memristor array with 100x10 readout network. This is quite an interesting observation that might be useful in low energy mobile sensing and computing applications.

Compared to linear array reservoirs another benefit of random reservoirs is the ability to input auxiliary training signals such as control signals described in section 3.4.2. In our control input optimization, test with random memristive networks we chose a smaller dataset with 2000 training images and 500 testing images. The images were block-coded (Figure 3.14) and fed into the random reservoir with 100 inputs and 40 control inputs that were optimized via the Optuna library each iteration. Within the first 100 steps of the optimization the accuracy increased by 5%, and reached up to 90% with 100x10 logistic

regression classifier readout layer after. With the increased number of training samples, the accuracy tends to increase, thus with full a 60000 train and 10000 test images, we expect around 3-5% improvement of accuracy.

3.7 Conclusion

In this chapter, we computationally explored properties of memristive networks with different interfaces (input and output), and structure (randomness) and various applications. In section 3.2, we started with exploring I-V characteristics of lattice and random memristive networks with periodic input voltage and found that networks of non-volatile passive memristors exhibit local activity represented by the decrease in resistivity or NDR [CHUA, 2005] as emergence of complex behavior in memristive systems. We further argued that the local activity is sole property of memristive networks that contain loops and is caused by different switching periods between groups of memristors with larger and smaller voltage drop across them. For example, in a periodic lattice networks, memristors aligned perpendicular to electrodes switch faster than those aligned in parallel according to the KVL and KCL. We provided a small Wheatstone bridge memristive network model that reproduced the NDR behavior to visually reinforce this observation. This particular property could be observed in Figure 3.5.

Without strict mathematical proves we showed that non-volatile memristive reservoir is comprised of various mathematical expressions on the input voltage and

In section 3.3 we provided the mathematical formulation of a generic circuit matrix operator with n voltage inputs and m current outputs. An accent was made on the intuitive derivation of some aspects of the modified nodal analysis starting from the circuit graph Laplacian. The resulting matrix equation representation Eq. 3.9 of a simple circuit with only 4 memristors and 1 resistor from Figure 3.7 exposed full spectrum of mathematical operations (summation, subtraction, division and multiplication) that can be viewed as an artificial neural network, or a reservoir that we develop further in section 3.4.

In section 3.4 we analyzed mappings of a voltage representation of the XOR table, produced by various random memristive networks with the goal to produce linearly separable red (true) and blue (false) circles on the current mapping. In particular linearly separable mapping in Figure 3.8 was produced with the simple bridge circuit from Figure 3.7 and has fewer features than linearly separable mapping produced with a larger circuit shown in Figure 3.9. An increased number of features in the large circuit mapping entails a higher

diversity of operations in the underlying circuit matrix of the circuit in Figure 3.9 that reinforces our discussion from section 3.3.

Further in section 3.4.2, we explored a static control voltage tuning technique by introducing control channels on top of the input channels. In this, our goal was to increase probability of successful mapping by varying voltage on the control channels with a Bayesian optimization algorithm (Optuna). Our findings showed that indeed tuning control voltages greatly increased probability of obtaining linearly separable XOR mapping. In particular, Figure 3.10 showed a random network with two inputs and 6 static control channels that were evolved to solve XOR problem. In only 19 iterations, the optimization algorithm was able to find an optimal configuration that resulted in linearly separable XOR (red and blue) problem depicted in Figure 3.11.

In section 3.5 we modelled random networks from a realistic memristive nanowires that were manufactured in our laboratory (and are further discussed in Chapter 4) and successfully applied them to solve XOR problem without and with static control voltages. We didn't observe much benefit of using realistic memristive nanowires except some differences in timing and voltage magnitudes and thus further we reverted back to memristive networks made of an equivalent memristive HP devices.

In section 3.6 we showed that random memristive networks are useful in information classification tasks. In particular we compared results of handwritten digit recognition obtained with random and linear array reservoirs (which were in agreement with results reported by other groups) which concluded that random reservoir has some computational advantages not only in the accuracy but also in the ability to reduce the dimensionality of the output (without much sacrifice in the accuracy) and also provides ability for tuning via static control voltage inputs.

CHAPTER 4

Manufacturing and computing with neuromorphic composites

4.1 Introduction

This chapter presents our results in the development of nanoscale resistive-switching composites with embedded neuromorphic architectures and demonstrates applications in complex logic classification with a discussion of opportunities for use in more advanced cases. In all of the composites, neuromorphic synaptic networks were constructed with Ag/Ag_2S nanowires that served as a resistive switching (RS) element.

Silver (Ag) is the third metal, metal, after gold and copper, known to humanity. Its first use has been dated to 4000 BC [Hill and Pillsbury, 1939]. The high ductility of this metal allowed ancients to use it in the production of fashion jewelry and food utensils [Alexander, 2009]. However, one of the most important applications of Ag was in medicine as ancients discovered its remarkable antimicrobial properties. While being a non-toxic metal to the human body, if contacted with water, or bodily fluids, Ag releases (Ag+) ions that interact with bacteria and fungal membranes, eventually destroying them [Lansdown, 2006].

Exposure to sulfur, naturally present in the air, especially in the form of hydrogen disulfide (H_2S), transforms Ag metal into silver sulfide (Ag_2S) that is characterized by the change of the surface of the exposed metal to a dark grey color [Storme et al., 2015]. There are multiple polymorphs of Ag_2S, and the two discussed in this work exist between 273 K and 860 K. Acanthite Ag_2S-α, is the low-temperature nonconducting polymorph with

a monoclinic crystal structure that is stable up to ~450K. Above 450K and up to ~860K Ag_2S is an ionically conductive argentite Ag_2S-β with the bcc crystal structure [Simonnin et al., 2020, Xu et al., 2010, Banerjee et al., 2007]. Besides temperature, the transformation between acanthite and argentite can also be induced by an external electric field [Xu et al., 2010], making Ag_2S a resistive switch. This is an essential property that has been used in the research of neuromorphic computing applications, which will be the main topic of this chapter [Wang et al., 2020].

In this work, we use a mix of Ag and Ag_2S NWs in 2D and 3D as the basis material for producing nanoscale networks with resistive switching connections, that mimic connectivity of neurons in the biological brain. While Ag wires serve as reliable conductors with relatively low constant resistivity (5 Ω/μm), the Ag_2S NWs represent synaptic resistive switch that together with Ag-NWs can be used for neuromorphic information processing applications.

The chapter is organized as follows:

We will begin with revisiting the theory of memristor devices that were previously introduced in chapter 3. The first topic we would like to address in section 4.2 is the modeling of resistive switching in individual Ag/Ag_2S nanowire (NWs) devices. With the help of in-house-made custom nanomanipulator probes, we measured the IV characteristics with which we built a stochastic memristor model with parameters that can be used to simulate individual Ag/Ag_2S nanowire resistive switches and networks. Further, in this section we also will present results on resistive switching of Ag NWs covered with polyvinylpyrrolidone (PVP), a polymer used in the manufacturing of Ag NWs. Finally, the section will be concluded by observing the electromigration-induced deformation of a small Ag_2S network and its IV characterization after deformation.

Further, in section 4.6 we provide a method for the fabrication of polycaprolactone (PCL) neuromorphic nanoelectronic composites (NNC) with embedded Ag/Ag_2S NWs and describe a method for manufacturing microelectrode arrays used in electronic characterization. Further in the section, we present some of the key electronic characterization results obtained with NNC devices such as resistive switching characteristics, time-resolved state forgetting, and sliding threshold plasticity, a property observed in biological neurons.

In section 4.17 we explore the multichannel characterization of the NNC device that exhibited several benefits that were not observed in other materials, such as negative differential resistance (NDR), naturally appearing noisy synaptic behavior, and complex forgetting time constants. This property of the material was applied in linearization tasks of a basic XOR gate and further explored a 3-variable XOR gate via optimization of static control voltages that supported theoretical observations from chapter 3.

4.2 Electronic characterization of a single Ag_2S nanowire

This section describes the manufacturing of pristine Ag NWs, Ag NWs with insulating polyvinyl pyrrolidone (PVP) coating, and Ag_2S NWs and the results of electronic characterization of the individual wires and junctions between them.

Ag NWs with average diameter D=124 nm and average length L=42 μm were manufactured in our lab using a modified polyol procedure initially described by Andres *et al.* [Andrés et al., 2015] with ethylene glycol (EG) as a solvent and polyvinyl pyrrolidone (PVP) as a capping agent to restrict the growth of nanowires to one dimension. At the end of the reaction, the obtained wires suspended in EG were split into two separate vials. The wires in the first vial were washed and centrifuged sequentially three times in ethanol and acetone to remove the EG solvent and reduce the insulating PVP residue on the surface of the wires [Wang et al., 2014, Vafaei et al., 2014, Tokuno et al., 2011]. The contents of the second vial were washed only with ethanol in order to retain insulating PVP coating residue on the wires. At the end of the washing and centrifugation ethanol was added to each container, producing a pearl-colored Ag NW suspension. The SEM image of obtained Ag NWs is shown in Figure 4.1 (a).

UV-Vis analysis was also performed on the synthesized nanowires with Thermo Scientific NanoDrop 2000c Spectrophotometer. Ag NWs dispersed in ethanol in a 1 cm path length quartz cuvette exhibited absorption spectra shown in Figure 4.1 (b). The prominent broad peak is confined between 390 nm and 410 nm agrees with the previously reported UV-Vis data for Ag NWs [Andrés et al., 2015]. Additionally, the thickness of the major peak correlates with the diameter of the obtained nanowires.

The length and diameter of the resulting nanowires varied from batch to batch and were found to depend on the molecular weight (Mw) of the capping agent (PVP) along with the temperature and duration of the reaction. However, the average value of the length and diameter was measured by TEM and, was found to be around 60 μm and 110 nm respectively as depicted in Figure 4.1 (c and d).

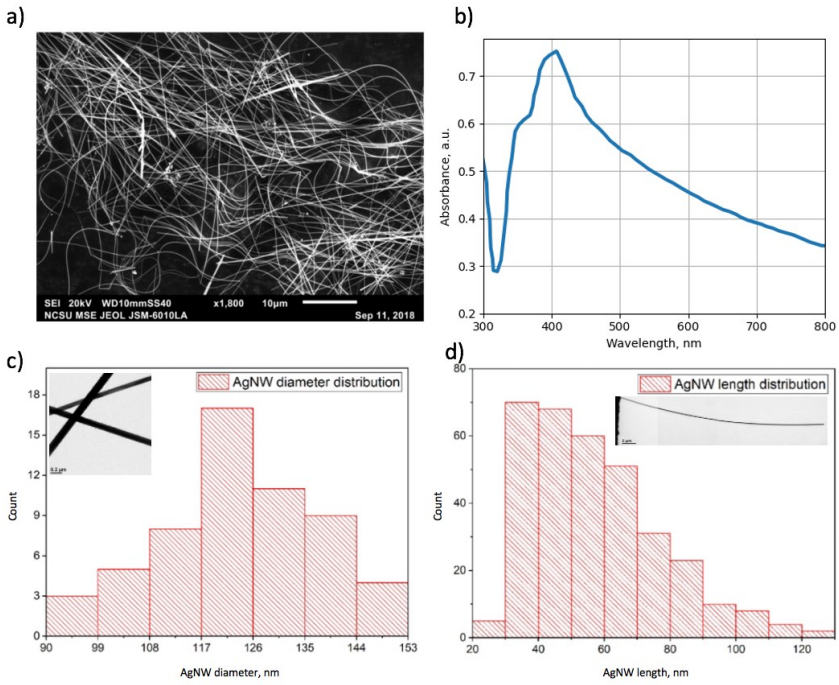


Figure 4.1: Characterization of obtained Ag NWs: a) SEM image of the Ag NWs with that exhibiting a large degree of curviness b) UV-VIS spectra of the obtained nanowires with broad absorption peak between 390 and 410 nm c) diameter distribution and an inset with a high magnification TEM (JEOL JEM-2000FX)) image used for diameter estimation d) length distribution of the NWs with an inset TEM snapshot of a single long Ag NW. The SEM/TEM characterization and size distribution measurements were performed by Ming Gao [Gao, 2021].

To prepare Ag_2S NWs, the Ag NWs without PVP layer (pre-washed with ethanol and acetone several times) were sulfurized, as was previously reported by Xiong *et al.[Xiong et al., 2016]*. Briefly, a stoichiometric amount of sulfur powder (Sigma-Aldrich) was ultrasonically dispersed in a vial with ethanol and mixed with an appropriate amount of the Ag NWs to create a dilute suspension. Then, the mixed suspension of Ag NWs and sulfur in ethanol was periodically agitated with a vortex mixer at room temperature to prevent precipitation of Ag_2S NWs and subsequent agglomeration. Since the complete conversion of Ag NWs in Figure 4.2 (a) to Ag_2S NWs in Figure 4.2 (b-c) is undesirable due to high resistivity, the reaction time was restricted to 5, 7, and 10 min when the color of the suspension changed from pearl grey to light brown, dark brown and black, correspondingly as is shown in Figure 4.2 (e), also shown with corresponding XRD spectrum. As can be visually confirmed by Figure 4.2 (a, b and d), the sulfurization leads to a reduction of the average length of the

wires by half and a slight increase in its diameter, with the measured dimensions of the Ag_2S NWs wire found D to be around 135 nm and L to be 25 μm length.

It was previously reported [Levard et al., 2012] [Belov et al., 2014] that high concentrations of sulfur the Ag particles get converted into Ag_2S via solid-fluid reaction characterized by direct formation of acanthite Ag_2S nanocrystal or nano bridges inclusions. In this work, we used this observation to model resistive switching of the incompletely converted Ag_2S NW with the percolation theory of scattered nanoclusters of conductive Ag within acanthite Ag_2S NW.

A schematic of the formation process of nanocrystals, along with the supporting EDS and SEM images, is shown in Figure 4.2 (d). Schematic drawings illustrating the state of conversion of Ag NWs, were constructed from the EDS spectra data, which show the quantitative elemental composition of the wire and visual appearance of SEM images shown on the right side of Figure 4.2 (d). The top frame shows an Ag NW that spent less than 5 min in the sulfur environment, showing that its surface is partially covered with sulfur with some spots being converted to Ag_2S. The EDS spectra show a much larger fraction of sulfur (84.1 %wt.) than silver (15.9 %wt.), meaning that the wires are covered with sulfur but not wholly converted to Ag_2 because the suspension is characterized by light brown color. Additionally, surface conductivity measurements (as discussed in chapter 2) showed metallic-like behavior similar to pristine Ag NWs.

At 5 to 10 min exposure time to sulfur, the color of the suspension changed from light-brown to dark brown. According to the EDS spectra, this observation denotes the reduction of sulfur concentration on the surface of the wire due to migration inside of the wire. As a result, its increases namely the amount of Ag (Ag 29.4 %wt.) and (S 70.6 %wt.) as shown in the middle frame of Figure 4.2 (d). Furthermore, at this point surface resistivity of the wires is noticeably increased due to a thin Ag_2S layer formed on the surface.

Above 10 minutes of exposure of Ag NWs to sulfur, the color of the suspension turns from dark brown to black as shown in the bottom image of Figure 4.2 (d). At this point, large amounts of sulfur diffused inside of the bulk of the Ag NW and therefore a significant portion of the wire is converted into Ag_2S. According to EDS spectra, the atomic composition is equilibrated and for both elements is 50 wt.%. Comparing to the weight fraction at the beginning of the reaction, sulfur reduced by approximately 30% while the amount of silver increased 3 fold. When the color of the suspension was dark brown the wires almost completely converted to Ag_2S with the fragmented distribution of Ag and Ag_2S nano-islands in the bulk volume of the wire. The conductivity of these wires is considerably lower than a light brown sample. When the wires were exposed to a sulfur environment longer than 10

min, then most of the Ag islands were converted into Ag_2S, and the wires behaved as an insulator.

The XRD spectra in Figure 4.2 (e) exhibits similar patterns as reported in other research [Motte and Urban, 2005, Sadovnikov and Gusev, 2017, Xu et al., 2010, Wang et al., 2016].

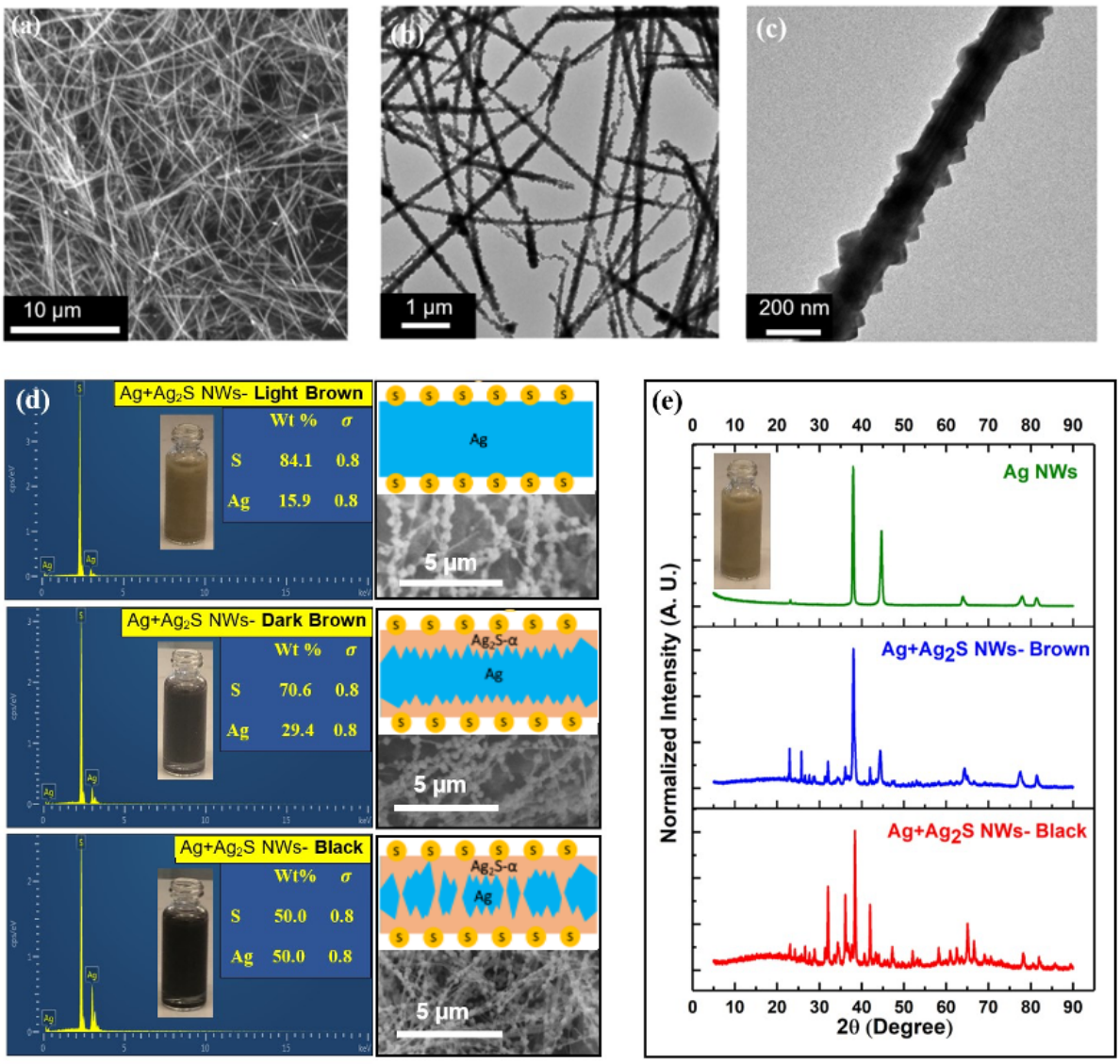


Figure 4.2: Characterization of produced Ag and Ag_2S NWs a) Scanning electron micrograph (SEM) of Ag_2S NWs (when the suspension was dark brown). b) Transmission electron microscopy (TEM) of a network of Ag_2S NWs. c) TEM of a single Ag_2S NW. d) Energy Dispersive X-Ray Spectroscopy (EDS) spectra and quantitative elemental data from Ag NWs at different durations of exposure of Ag NW to sulfur. At the end of the first 5 minutes of exposure to S, the Ag NW suspension turned light brown (top). After 7 mins suspension turned dark brown (middle). Finally, with increased amounts of S diffused into the volume of the wire from the surface, the suspension turned black (bottom). On the right side, SEM images and schematic drawings of NWs correspond to each of the timesteps of sulfurization. Once Ag NWs were placed in the sulfur environment, the proportions of Ag nano-clusters in the Ag_2S matrix varied proportionally to the duration of exposure to sulfur. e) X-ray Diffraction (XRD) spectra of Ag NWs before sulfurization, the inset shows Ag NWs suspension color (top), Ag NWs with partial Ag_2S inclusions (brown; middle), and black Ag_2S NWs (bottom).(Characterization by Mahshid Hosseini).

A single Ag NW and Ag_2S NW were characterized via a two-point contact created between a sharp tip of a nanomanipulator (Kleindiek, GmbH) probe and silver paint, as shown in Figure 4.3 (a). The setup was housed on a Nikon Optiphot 150 microscope stage with long working distance objectives (up to 4 mm) to provide sufficient space for the movement nanoprobe and a source measurement unit (SMU) Keithley 2636B.

To perform the measurement, a diluted suspension of Ag or Ag_2S NWs in ethanol was drop-casted with a pipette at the center of an ultrasonically cleaned microscope glass slide where a thin network layer of NWs was formed after ethanol evaporation. Further, the copper tape was deposited near the area of NWs, and a thin layer of fast-drying silver paint (Fisher Scientific) was applied to cover the copper tape and the deposited NWs partially, as shown in Figure 4.3 (a). After Ag paint was dried, the area near the boundary of the silver paint exposed many NWs free-ended Ag NWs with one end submerged under the Ag paint. A two probe measurement was performed on a single wire and a network of wires with the "low" terminal of an SMU connected to the copper tape and the nanomanipulator's tip connected to the SMU's "high" terminal. However, not every electrode material is suitable for establishing a reliable low resistance contact with the Ag or Ag_2S NWs.

Commercially available tungsten tip electrodes did not provide reliable contact with the NWs first due to the high relative hardness of tungsten, which broke the relatively soft Ag NWs. The second problem was caused by the formation of oxide film (WOx) on the surface of the probe after a few hours of exposure to the ambient environment. Thus custom-made Pt/Ir electrodes were manufactured with a tip radius of 100 nm, as was suggested by other reports [So/rensen et al., 1999, Kupper, 2012, Zhang and Lian, 2016].

In the procedure, Platinum-Iridium (PtIr) wire of 127 μm diameter composed of 20 %wt. Ir (Alfa Aesar) was ultrasonically cleaned in ethanol and acetone. A saturated $CaCl_2$ solution was prepared (20 %wt.) and transferred to a 50 ml graphite crucible. About 1 cm of PtIr wire was soldered to a 24 gauge copper wire and mounted on a micropositioning XYZ stage (Newport) above the crucible with $CaCl_2$ solution to allow controlled removal of the tip from the etching solution to create a sharp profile.

The tip of the wire was partially submerged (~2 mm) in the solution, and an AC power supply was connected between the crucible and the PtIr wire. The XYZ stage was set to pull the wire up from the solution at the speed of 100 μm/min while the power supply was delivering 40 V at 40 Hz between the graphite crucible and the PtIr tip. During the etching, the initial current of 2 A gradually fell to 0 A as the wire was pulled out of the solution, and the thickness of the wire was reduced to hundreds of nm. Controlling the

speed of wire removal from the solution, with the help of the XYZ stage and the magnitude of AC voltage, allowed us to control the geometry of the tip and achieve a high degree of sharpness [Zhang and Lian, 2016]. We found that PtIr wires with less than 20% wt. Ir is too soft for single wire measurement since the tip easily bends upon contact with hard surfaces such as microscope slide glass. Thus it is preferable to use 25% or even 30% wt. Ir wires for single NW measurements with a nanomanipulator.

A few different lengths of the same wire were stimulated with one period of bipolar triangular voltage as shown in Figure 4.3 (b) and (c) correspondingly. First, the 16 μm long Ag_2S NW was chosen, and the contact with the nanomanipulator's tip was established for two-probe IV characterization Figure 4.3 (b). Then the characteristics were measured with the same wire which length was mechanically reduced to 8 μm, Figure 4.3 (c).

There are several noticeable features in the IV characteristic of Figure 4.3 (b). The 1st quadrant of the IV characteristics in Figure 4.3 (b) shows a negligibly small current until a sharp peak at 2.4 nA slightly above 4 V (1st red circle), followed by a subsequent reduction of the current to 2 nA at 5 V (2nd red circle). As voltage started to decrease during the reverse loop, the current decreased to 1 nA until after passing the 3.5 V mark, the current instantly increased to just below 8 nA (4th red circle), designating devices ON state. As the polarity of the voltage sweep crossed negative value, the current in the 3rd quadrant of Figure 4.3 (b) still showed the device ON state reaching the peak near -4.5 V and -6.5 nA (5th red circle) and rapidly dropping to 0.5 nA at the end of the cycle at -5 V mark (6th red circle).

In Figure 4.3 (c), the same Ag_2S wire's length was reduced to L=10 μm and stimulated with the same triangular voltage bias as in Figure 4.3 (b), which exhibited a different RS pattern. As with longer wire in Figure 4.3 (b), hard resistive switching took place at 4 V, the maximum current increased by nearly 75%. In other experiments, the ON-state resistance was proportional to the length of the wire and was consistent with a wide range of lengths of the wires with an approximate value of 10 k/nm as discussed further. Thus the resistive switching process is not restricted to a single location within the wire, instead, it is distributed across its entire length.

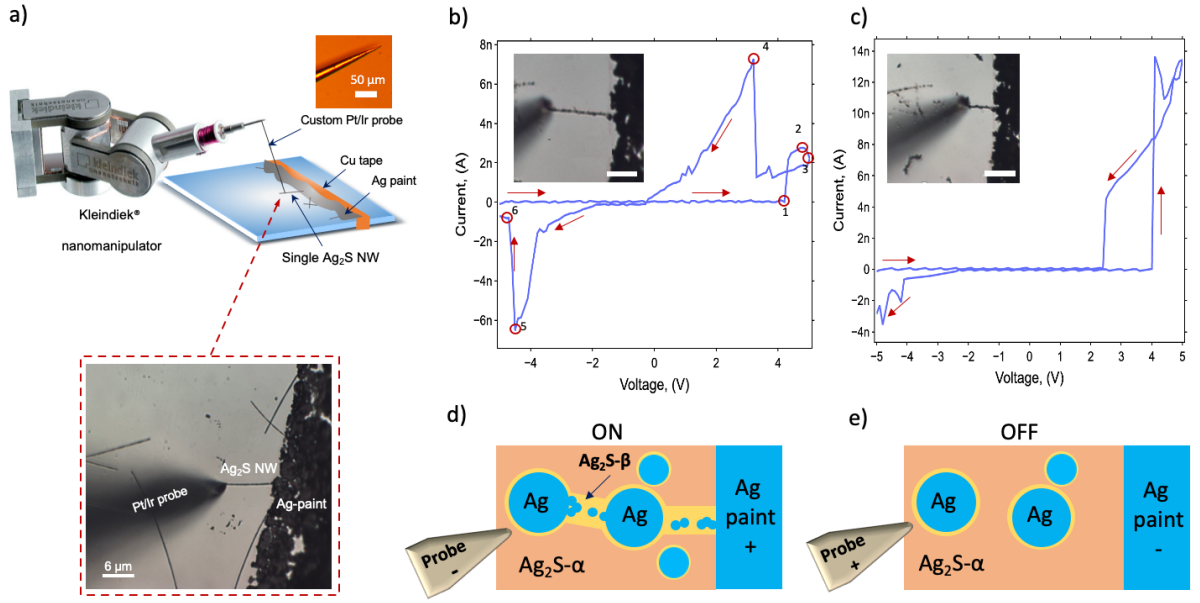


Figure 4.3: Resistive switching of a single Ag_2S NW. a) Schematics of a nanomanipulator (Kleindiek Nanotechnik, GmbH) with a custom-made Pt/Ir 80-20 electrode contacting a tip of a single Ag NW exposed on a microscope slide with another tip submerged under Ag paint (Image by Mahshid Hosseini). b) A microscopic image and IV characteristic of a single 20 µm long Ag_2S NW under a negative-positive-negative triangular-shaped voltage cycle. c) IV characteristics of the same Ag_2S NW as in b) after it was shortened to 10 µm. d) Conducting (ON) state of the model of resistive switching in a single Ag_2S NW with scattered Ag nanocrystal inclusions and Ag_2S-β nanobridges in Ag_2S-α phase (adapted from Xu *et al.* [Xu et al., 2010]). e) Non conducting (OFF) state of the model.

The pinched hysteresis of the switching shown in Figure 4.3 (b) and (c) can be well described by the model of RS in a thin film TiO_{2-x} memristor [Strukov et al., 2008b]. Although this model reproduces the key features of memristive behavior, such as pinched hysteresis, it does not describe all RS regimes, particularly the stochastic reversal of the loop near the maximum voltage we observed in Figure 4.3 (b) point 2 and 3 and Figure 4.3 (c). The twisting of the loop is spontaneous and is likely caused by Joule heating that leads to partial breakdown of Ag filaments, formed within the argentite Ag_2S-β phase as first described by Liao *et al.* [Liao et al., 2010].

Thermal and crystalline vibrations [Sánta et al., 2019, Sánta et al., 2021] induce distortions in the conducting filaments and nano-islands that can break down and be measured in the conductivity instabilities. However, it is likely that these instabilities are partially compensated by the scattered nano-islands and more developed filaments, as is supported

by the smoother current curve during the reversal of the bias voltage displayed in Figure 4.3 (b) and (c).

4.3 Stochastic redox switching model of a single Ag_2S NW

To describe the stochastic breakdown shown in Figure 4.3 (d) and (e) the equations 4.1 and 4.2, of the standard memristor model, can be augmented with additional constraints presented in equations 4.3 and 4.4. The Ag_2S-β pathways within the Ag_2S-α phase not only allow for Ag^+ ions to migrate under the influence of an electric field but also provide an environment for the formation of scattered Ag nanocrystals or filaments [Xu et al., 2010] Figure 4.3 (b). Given the large dimensions of our wires with diameters of hundreds of nm and lengths of tens of microns, the resistivity of the Ag_2S wire will depend on the concentration and distribution of randomly scattered argentite phase inclusions and therefore spontaneously formed Ag clusters in them under the influence of electric field bias.

Thus here, we introduce the "stochastic drift memristor model" in which resistivity of the device is controlled by two variables, namely δ and ω instead of just one parameter ω.

The stochastic parameter δ governs the magnitude of Ron and specifies the volume fraction of nano-islands and conductive filaments in the Ag_2S matrix. Due to the fluctuations caused by Ag^+ ion redox exchange and metastable atomic positions, the conductive channel's thickness will also fluctuate unpredictably [Sánta et al., 2019, Gubicza et al., 2016], and hence can be modeled with a stochastic process described by Equations 4.3 and 4.4 with normally distributed noise with standard deviation parameter σ. Metallic Ag nano-clusters are formed and reorganized spontaneously within the volume of the Ag_2S phase due to the heat and electric field-induced relocation of Ag^+ ions in 3D volume [Hebb, 1952, Simonnin et al., 2020]. Therefore, percolation theory is used to connect the conductivity in the ON state via the R_{on} variable with the volume fraction of Ag nano-islands in the mixture of Ag_2S-α and Ag_2S-β phases and thus is described by the power-law in Equation 4.3. Determination of percolation threshold is a mundane task. Thus in the simulator, we set δ=δ-δ_0 where δ can take only positive values or zero.

Thus in the model high concentration of scattered nano-islands will produce a low Ron value and a low effective length L of the wire for fixed mobility μ found in the literature [Allen and Moore, 1959, Simonnin et al., 2020]. On the other hand, a low concentration of scattered islands will result in a high magnitude of Ron and effective length of the wire L close to observed.

$$V = \left(R_{on}(\delta)\omega + R_{off}(1-\omega)\right) I, \text{where } R_{off} = L\rho_{off}, \text{ and } R_{on} = L\rho_{on} \tag{4.1}$$

$$\frac{d\omega}{dt} = \mu \frac{R_{on}}{L} I F(\omega) \tag{4.2}$$

$$R_{on}(\delta) = L\rho_{on}(\delta - \delta_0)^{-\beta} \tag{4.3}$$

$$\delta = \delta_{min} + \mathcal{N}(0, \sigma^2), \text{ where, } \delta_0 \leq \delta_{min} = R_{ONmax}^{-\frac{1}{\beta}} + \delta_0 \leq \delta \leq \delta_{max} = R_{ONmin}^{-\frac{1}{\beta}} + \delta_0 \tag{4.4}$$

Scattered Ag nano-islands in a volume and on a surface of ionic conductors were reported in various experimental configurations. In particular, Ag clusters were formed on the surface of ZnO nanowire during cycles of resistive switching in a Ag/ZnO/Pt system, [Milano et al., 2018]. In another study, Wang *et al.* showed an in situ formation of Ag nanoclusters with HRTEM in a planar system of Au/SiOxNy:Ag/Au [Wang et al., 2016]. Finally, a detailed study of resistive switching of Ag_2S ionic conductors performed with HRTEM suggested the mechanism of ionic transport [Xu et al., 2010].

The evidence of the Ag nano-crystal formation in Ag_2S NW is also supported by comparing switching times in thin films and long wires. In an acanthite, Ag_2S-α wire model with an effective length of 16 μm, switching between ON/OFF states would take approximately 50 seconds at 3 V driving voltage potential due to relatively low mobility of Ag^+ ions in the phase, namely μ=.5 $\mu m^2/(sV)$ [Allen and Moore, 1959, Simonnin et al., 2020]. However, in the experiment and the best-fitted parameters of the model to the experimental data of a single Ag_2S NW (Figure 4.4 (d) and Table 4.1), switching occurs within tens of seconds. The best-fitting was found only when the wire had a shorter effective length between L_e=8 μm and L_e=12 μm, compared to actual L=16 μm, and supports the hypothesis of scattered nano-islands in the volume of a single Ag_2S NW device. Thus the area between electrodes is filled with a mixture of high mobility argentite, low mobility acanthite, and islands of Ag inclusions. Since the model only exhibits stochastic current pattern in ON state, there is no stochastic pattern in the OFF state as can be noticed comparing currents during negative sweep in Figures 4.4 (a) and (d).

To test our hypothesis we simulated a memristive nanowire described by equations 4.1 to 4.4 with the CircuitSymphony, a circuit simulator described in chapter 2 [Frick and LaBean, 2021a] with the parameters of the following simulations. In equations 4.1 to 4.4,

the magnitudes of R_{ON} and R_{OFF} are defined to be proportional to the length of the Ag_2S NW with the multiplier ρ_{off} or ρ_{on}. Instead of explicitly defining the minimum and maximum values of the Ag^+ ions drift δ_{min} and δ_{max} and to simulate the stochastic memristors model from Equations 4.1-4.4 with memristive NWs of arbitrary lengths, a new set of auxiliary parameters were introduced, namely R_{ONmin} and R_{ONmax}. The values of R_{ONmin} and R_{ONmax} are used as multipliers to the fixed value of resistivity ρ_{on} in order to define the magnitude of R_{ON} and satisfy the following inequality $R_{ONmin} < R_{ONmax}$ that defines the boundary of drift of R_{ON}. Thus at the percolation threshold when the volume fraction of Ag filaments δ is at its minimum, namely $\delta=\delta_{min}>\delta_0$, the R_{ON} will be at its maximum value, $R_{ON}=\rho_{on}$ $L(\delta_{min}-\delta_0)^{-\beta}=R_{ONmax}$ $L\rho_{on}$ and when the volume fraction of Ag nanocrystals is above percolation threshold at some maximum value $\delta=\delta_{max}$, $R_{ON}=\rho_{on}$ $L(\delta_{max}-\delta_0)^{-\beta}=R_{ONmin}$ $L\rho_{on}$, will correspond to the minimum value of R_{ON}. In the simulation, the dynamics of δ follow a random walk process, Equation 4.4 and at each iteration step, the boundaries of R_{ON} are checked in the simulator to prevent an unconstrained drift. If at some iteration step, the new value of δ produces R_{ON} below R_{ONmin} $L\rho_{on}$, then the value of δ will be set to be equal to the highest value of δ_{max}. On the other hand, if the new randomly assigned value of δ causes R_{ON} to grow above $R_{ONmax}L\rho_{on}$ the value of δ will be replaced with the smallest magnitude at the lower boundary δ_{min}.

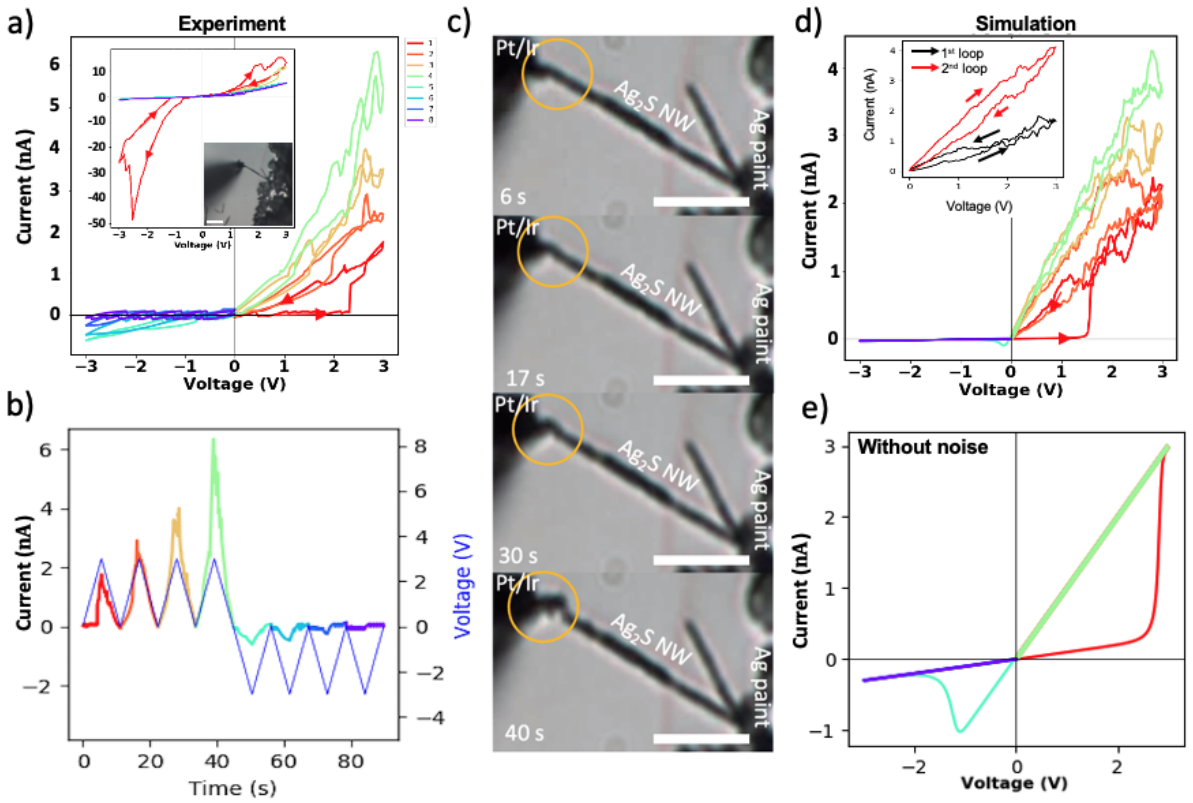


Figure 4.4: Measurement and modeling of stochastic resistive switching of Ag_2S NW memristor. a) IV characteristics of a 16 μm long Ag_2S NW stimulated first with four positive triangular pulses followed by four negative pulses (10 s period) showing OFF state (high resistance) at the negative bias and gradually reducing resistance in the ON state of the device. The inset IV plot showed spontaneous loop reversal when the wire was stimulated with alternating negative and positive triangular pulses with a 10 s period. The micrograph shows a single Ag_2S NW contacted by the nanomanipulator. b) Current response to triangular voltage stimulation of the Ag_2S NW. c) Time-lapse of the deformation caused by electromigration of Ag^+ ions in the Ag_2S NW during the first 40 s of the triangular voltage stimulation. d) Suboptimal simulation of the stochastic switching that replicates the behavior of the experimental data, with the proposed memristor model in equations 4.1 to 4.4 and the parameters from table 1. Inset in d) shows spontaneous loop reversal. e) Simulation of the resistive switching with default parameters in table 4.1 with zeroed noise parameter. All scale bars are 10 μm long.

The experimental simulation results are shown in Figure 4.4 (a), where 16 μm long Ag_2S NW was stimulated with four positive and four negative triangular pulses with 10 s period. The memristor was modeled with the parameters listed in Table 4.1. The fitting was performed with Bayesian optimization provided by Optuna library [Akiba et al., 2019], over a set of hand-picked discrete values of the model parameters. The loss was calculated as a

Parameter	Value
ρ_{on} (linear resistivity in ON state)	4.75×10^3 $\frac{\Omega}{\text{nm}}$
ρ_{off} (linear resistivity in OFF state)	4.75×10^6 $\frac{\Omega}{\mu\text{m}}$
ω (length of Ag rich region)	0 nm
L_e (effective length)	8 to 12 μm
μ (Ag^+ ion mobility)	1×10^3 to 8×10^3 $\frac{\mu\text{m}^2}{\text{sV}}$
σ (noise factor)	20 to 50
R_{ONmin} (minimum resistance factor)	0.5
R_{ONmax} (maximum resistance factor)	5
β (percolation exponent)	1.1
Simulator Integration Step	0.5 μs

Table 4.1: Simulation parameters for stochastic memristor model in Figure 3.

Euclidean distance, or root-mean-squared distance, between the laboratory data points and model output at a particular time step for the whole duration of the stimulation. Thus, each iteration was comprised of 50 independent measurements, after which the mean loss was the metric of performance of the model for the iteration.

Interestingly, fitted values were found close to the actual values reported in the literature. In particular, the mobility μ was found between 1x10^3 to 8x10^3 $\mu m^2/(sV)$ and was only slightly below the mobility experimentally observed for the cubic Ag_2S-β phase μ=15x10^3 $\mu m^2/(sV)$ at T=450 K (for comparison the Ag^+ mobility in the monoclinic phase of Ag_2S-α, is μ~1 $\mu m^2/(sV)$ at T=300 K) [Allen and Moore, 1959, Simonnin et al., 2020]. The difference could be attributed to the presence of different charge carriers, namely electrons and Ag^+ ions, and a mixture of acanthite and argentite phases [Hebb, 1952].

This subsection briefly described our approach in modeling stochastic resistive switching in a single Ag_2S NW based on percolation conductivity in between Ag nanoclusters. Further refinement of the model with additional data from an *in situ* TEM during the resistive switching is required. The robustness of the model might also be increased by taking into account thermodynamic parameters of the NW such as modeling Joule heating and Newton's cooling. The stochastic nature of the conductive filament creation and destruction in the Ag_2S NW is an interesting property that mimics biological synapses. In particular biological synapses exhibit stochastic openings of ionic channels that cause unreliability of signal propagation through the synapse that are thought to play an important role in the brain's energy conservation and enhances learning [Zador, 1998, Rusakov et al., 2020, Harris et al., 2012].

4.4 Ag_2S NW electromigration plasticity

Interesting microscopic phenomena can be observed if a small network of Ag_2S NWs is stimulated with a constant voltage. In Figure 4.5 (a), 5 V DC voltage bias was applied to the Y-shaped Ag_2S network for 20 s between the microprobe tip (- terminal) and the Ag paint (+ terminal) while the structural changes in the network were recorded on a camera. Under the influence of the electric field, some parts of Ag_2S NWs convert to argentite Ag_2S-β phase that permits migration of Ag^+ ions migrating from the Ag pain shore to the tip of the PtIr microprobe tip. This migration of the Ag^+ ions and expansion of the argentite Ag_2S-β phase pathways deformed the shape of the network first near the microprobe tip [Xu et al., 2010]. As can be noticed from the figure, expansion of the wire leads to multiple effects such as 3D bending (marked with green (planar) and yellow (lift-off from the plane of the microscope slide)) and also detachment of one of the pathways (red arrow). Despite significant deformation, the Ag_2S NW network retained the resistive switching property shown in Figure 4.5 (b), where it was stimulated with a single 20 s long triangular pulse.

The ability of biological neural networks to learn is attributed not only to the creation of synaptic connections but also their destruction [Saveliev, 2005]. The density of synaptic connections in humans was estimated to be the highest in toddlers' age and is subsequently reduced by half in adulthood through the natural process known as "synaptic pruning" [R., 1979, Feinberg, 1982]. In addition, it was discovered, that astrocytes play a significant role in synapse formation, function, and elimination [Chung et al., 2015]. Thus, since most energy in the brain is concentrated at synapses, removal of unnecessary synapses can be treated as an optimization process [Harris et al., 2012] that makes neural networks more efficient in remembering only crucial information while discarding and unnecessary information [Saveliev, 2005] [Sakai, 2020].

A similar concept is frequently used in artificial neural networks during training and is referred to as "dropout" regularization [Srivastava et al., 2014]. In a feed-forward artificial neural network architecture, the dropout mechanism disconnects a fraction (typically 20%-50%) of connections between adjacent layers at each iteration. This leads the remaining connections to be more sensitive to the critical information sent from the upstream layers and increases the information contained in each connection. As a result, networks trained with the "dropout" technique can better generalize on the domain of trained data due to lesser overfitting.

The plasticity and relative reliability of Ag_2S NWs could add an important feature in neuromorphic computing applications that aim to mimic biological synapse behavior with

"synaptic pruning" or "dropout" mechanisms. Furthermore, in denser Ag_2S NW networks, especially in 3D, the movement of the NW under electric bias could create new connections with nearby wires.

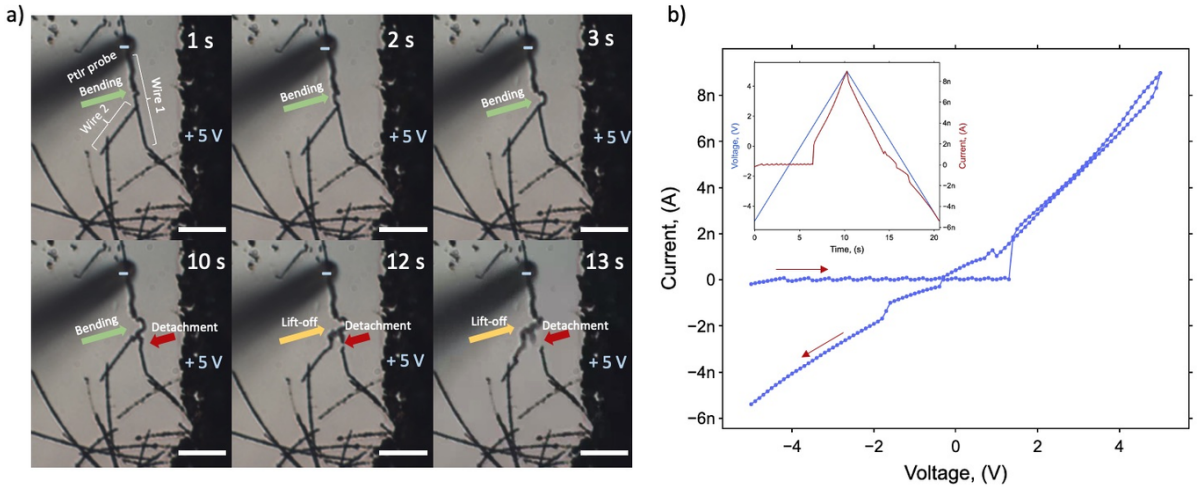


Figure 4.5: Deformation of a small Ag_2S NW network under influence of electric field and subsequent measurement of the IV characteristics a) time-lapse of the deformation of the Ag_2S NW network under constant DC bias of +5 V on the Ag paint shore. Over the period of 13 seconds, the Y-shaped structure, formed by two wires, connected to PtIr tip and Ag paint, evolved into a new structure. Due to electromigration, wire 1 broke into two parts near the intersection with one part being fused with wire 2 and undergoing bending near the intersection (green arrow), which was followed by a detachment from the lower part of wire 2 marked with a red arrow at 10 s. Finally, the junction of wire 1 and half of wire 2 starts to bend upward away from the surface of the microscope slide. b) Despite severe deformation, the remaining structure preserved resistive switching behavior, which was confirmed with an IV curve obtained with a single 20 s long triangular pulse of 5 V.

4.5 PVP coated nanowire junction

In this subsection, we present resistive switching of a junction of two PVP-coated Ag NWs. PVP is a capping agent used in the polyol manufacturing method of Ag NWs and is conveniently present on the surface of the wires if not intentionally removed. The experimental setup was similar to the above single NW characterization, except the wires used were not washed with acetone and thus retain PVP on the surface [Vafaei et al., 2014, Tokuno et al., 2011] with 1-2 nm thickness [Manning et al., 2018]. Several nanowire junctions were visually located through an optical microscope with the tip of the first Ag NW being under the grounded Ag paint, whereas the second NW, crossing with the first one, has a dangling end that can be approached with nanomanipulator's PtIr tip and close the circuit as is shown in Figure 4.6. A series of triangular pulses with 7 s period were fed through the junction, exposing instantaneous volatile switching above 0.5 V. The uniform PVP coating on the Ag NW [Manning et al., 2018] that mediates Ag filament growth under the force of external electric field makes hysteresis loops look self-similar as shown in Figure 4.6 (b) in which a 100 Ω resistor was used in series. In particular, each subsequent loop is nearly identical to the previous and is symmetric concerning the polarity of the input voltage. The resistive switching through PVP junction has volatile binary nature, meaning that it hard switches between insulator to conductor states and nearly instantaneously loses conducting state upon reversal of the voltage polarity. In contrast, Ag_2S NW resistive switching is more complex with a continuous increase or decrease of conductivity if stimulated with voltage bias of the same polarity. This difference is evidence supporting our percolation theory-centered stochastic model of Ag nanocluster inclusions discussed in the previous subsection.

The top inset of Figure 4.6 (b) shows the temporal variation of the bias voltage and corresponding current. In almost all examples, switching occurred around 0.5 V and rarely exceeded 1 V, whereas, in Ag_2S NWs, the range of hard switching varied between 1 V to 4 V. The bottom inset of Figure 4.6 (b) shows a different presentation of the plot where negative differential resistance (NDR) can be observed when X-axis is an actual voltage on the junction (i.e., input voltage minus voltage drop on 100 Ω series resistor) and Y-axis represents current. Unfortunately, we found that PVP junctions are unreliable and typically break down after a few dozens of cycles; thus in the rest of the chapter, we will focus on Ag_2S NWs.

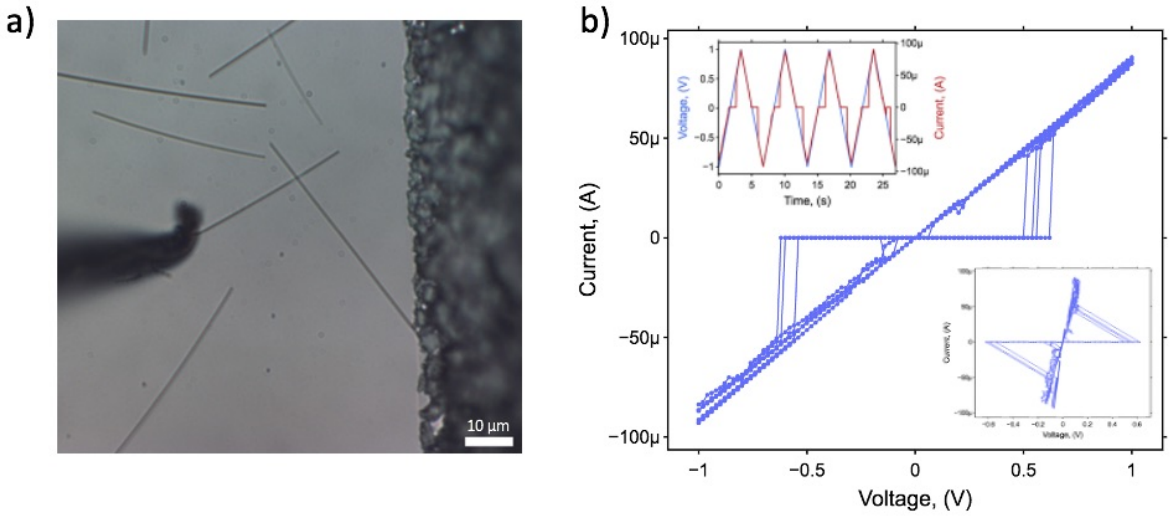


Figure 4.6: IV characteristics of a junction between two PVP coated Ag nanowires. a) microphotograph of two overlapping wires near Ag paint shore. b) I-V characteristics showing reversible abrupt metallization slightly above 0.5V 10 kΩ series resistor is used. (top inset) time progression of input voltage and resulting current. (bottom inset) IV characteristics where voltage is taken for the junction

This section presented experimental results on the diverse switching behavior of isolated Ag_2S NWs, their networks, and junctions of PVP-coated Ag NWs. In the next section, we will present results on the production of 3D composites with embedded Ag and Ag_2S NWs along with their electronic characterization and a method for solving an XOR task.

4.6 Neuromorphic nanocomposites

This chapter demonstrates how to make scalable the manufacturing of 3D neuromorphic multicomponent composites with ionic flux-based atomic switching networks. It will be shown that these composites can be manufactured, characterized, and employed for computing applications. We will discuss their advantages and disadvantages and compare some of their properties to conventional computation methods.

Our experimental results are the continuation of simulations presented in chapter 3 that showed that random resistive switching networks could solve simple boolean logic problems and perform more complex classification tasks such as handwritten digit recog-

nition.

Our model of memristive networks relies on the empirical conductivity study of isolated Ag/Ag_2S NWs presented in the previous section. In particular, we show that memristive networks embedded into a polycaprolactone (PCL) polymer matrix form an all synapse circuit that can change internal connectivity pathways through the application of an external electric field and display similar characteristics as simulated studies from chapter 3.

4.7 Overview of neuromorphic materials and our approach

The research in neuromorphic computing is performed in several directions that can be categorized by the type of materials used, the degree of materials organization, and computation methods.

4.8 Neuromorphic materials

Typical materials with resistive switching properties are electrochemical metallization cells (EMC) [Valov and Kozicki, 2013, Hasegawa et al., 2001], where resistive switching is achieved by growth and destruction of Ag or Cu filaments in an electrolyte such as Ag_2S [Ohno et al., 2011, Xu et al., 2010], used in this work, or an insulating matrix with sufficient ionic conductivity such as SiO_2 [Yang et al., 2012, Li et al., 2016] or ZnO [Milano et al., 2018, Chen et al., 2013a]. The prototype of EMC switching was reported in 2005 by Infineon Technologies that showed a prototype of a conductive bridge memory device [Kund et al., 2005] based on previous work of Hasegawa *et al.* [Hasegawa et al., 2001] and Hirose *et al.* [Hirose and Hirose, 1976]. Since this work is focused on Ag_2S NW and devices made of them, we will be mostly discussing research related to these kinds of memristors.

Other types of memristive materials are also available based on the valence change mechanism and typically belong to the family transition metal oxide devices. In particular, TiO_x [Choi et al., 2005] or WO_x [Biju et al., 2011], where the highly conductive state is created by the formation of conducting channels due to migration of oxygen vacancies or oxygen anions in the oxygen-poor phase of the material, whereas a highly resistive state is achieved by the formation of insulating oxygen-rich metal oxide phases [Goldfarb et al., 2012].

Besides thermochemically driven [Ielmini et al., 2011] and phase change driven [Wong et al., 2010, Sarwat, 2017] resistive switching mechanisms, particular attention is attracted to Mott-Peierls transition devices that are capable of generating biological neuron-like

action potential spikes, and thus are called neuristors. A few examples of Mott transition spiking neuristor devices that have been recently reported are based on VO_2 [del Valle et al., 2020] and NbO_2 [Pickett et al., 2012]. Recently, Yi *et al.* experimentally demonstrated 23 biological neuron spiking behaviors with a single VO_2 active neuristor [Yi et al., 2018], making this material even more attractive for neuromorphic computation.

4.9 Degree of organization and computing methods

The degree of organization, or the architecture of the resistive switching NNC material, is related to the desired method of computation. Crossbar arrays, introduced in chapter 1, are the most conventional method for harnessing memristive devices for computation. In the crossbar array approach, memristors, typically TMO such as TiO_x or WO_x, are sandwiched between intersecting conducting nanoscale bars allowing each particular memristor to be addressed in a column and row fashion independently from other memristors located in different rows and columns. Due to the top-down manufacturing processes, the crossbar arrays provide addressability and uniformity in memristive device characteristics, which allow the creation of reproducible structures of memristive networks. Some of the notable applications include matrix operations such as matrix-vector dot-product [Choi et al., 2015], principal component analysis [Choi et al., 2015], pattern classification [Truong et al., 2014, Alibart et al., 2013], one step matrix inversion [Sun et al., 2019], reservoir computing based image recognition [Du et al., 2017b] and even partial differential equation solvers [Zidan et al., 2018].

Despite remarkable results, the memristive networks constructed with crossbar arrays heavily rely on the external electronics that configure the memristive network and manage data flows. Moreover, the number of resistive switches in a single crossbar array is restricted by the employed technological process. And while crossbar arrays allow constructing artificial neural networks with feed-forward architectures that perform well in classification tasks [Li et al., 2018, Midya et al., 2019], they still rely on conventional deep learning optimization techniques and, unlike humans, require massive amounts of training data. Also, conventional artificial neural networks perform poorly with adversarial examples or cross-domain data [Waldrop, 2019, Papernot et al., 2016].

The randomly assembled memristive networks, especially in 3D, aim to create brain-inspired architectures that can perform regression and classification tasks similar to cross-

bar arrays without the need for complex VLSI manufacturing processes. Instead, such systems rely on evolutionary or other optimization techniques that reconfigure randomly assembled structures to perform required tasks, frequently referred to as *evolution-in-materio* [Miller et al., 2014, Massey et al., 2015]. Also, reservoir computing approaches based on randomly connected artificial recurrent neural networks aka Echo State Networks (ESN) [Jaeger, 2004, Jaeger, 2002] or Liquid State Machines (LSM) [Maass et al., 2002] spiking equivalent [Tanaka et al., 2019, Nakajima, 2020].

Some of the notable achievements in the evolutionary reconfiguration of the circuitry have shown impressive results. For example, Massey *et al.* were able to solve a 2 class linear classification problem by evolving carbon nanotubes suspended in a drop of liquid crystals on a 4x4 planar electrode array, with 2 inputs 2 outputs, and optimization of 8 voltage potentials on it, essentially performing the functionality of a single linear perceptron [Massey et al., 2016]. Likewise, Bose *et al.* demonstrated a reconfigurable logic gate (AND, XOR and NAND) by evolving a disordered network of gold nanoparticles separated by 1-octanethiols with an 8 electrode array setup (2 inputs, 1 output, and 5 control voltages) [Bose et al., 2015]. In 2020, a similar system was used by the authors of the original work with slight modifications to solve the MNIST [Lecun et al., 1998] handwritten image recognition problem obtaining an impressive 96% accuracy [Chen et al., 2020].

While evolutionary computing mentioned above relies on reconfiguring the underlying circuitry via electric field calculated with an optimization algorithm, reservoir computing relies on different computing approaches. In reservoir computing, the signal is applied to the randomly connected network of neurons (typically with connectivity sparsity above 95%) [Lukoševičius and Jaeger, 2009, Jaeger, 2004], and only the weights in the readout layer are tuned to perform a particular task.

James Gimzewski and colleagues at UCLA have pioneered research in this direction, in particular with self-organized Ag_2S [Stieg et al., 2011] and Cu_2S [Nayak et al., 2012] atomic switching networks and demonstrated applications in different computational tasks, including reservoir computing [Sillin et al., 2013]. In particular, they successfully solved waveform generation with Ag_2S dendrites [Sillin et al., 2013] grown on a 2D microelectrode array with copper contacts. The researchers trained a software-implemented output layer with static weights connected to the device with which they generated signals of triangular or rectangular shape by feeding in only sinusoidal shape signal. This remarkable result shows that Ag_2S networks with only "synaptic" connections can behave like a recurrent neural network, i.e., a network whose that final state depends on the history of states and that can generate new harmonics that can be combined to generate a novel signal that

the network has not seen before. Recently their group showed a method of solving the spoken word recognition [Lilak et al., 2021] task with Ag_2S and AgI dendrites grown on Cu posts of a microelectrode array. The spoken words were converted into 13 mel-frequency timeseries [Furui,] fed as patterns into the system, and the resulting current was harvested with which the output layer was trained.

The devices in the examples presented above were implemented in 2D and as we showed in chapter 2 of this dissertation, planar geometry requires less material to achieve connectivity but reduces the diversity of available connections and networks. 3D integration, on the other hand, requires more material for connectivity but offers more diversity in local connectivity and a higher probability in distant connections that could play an important role in neuromorphic computing. Another problem in the mentioned implementations above is the brittleness and short longevity of the devices. The performance experiments executed in our lab following the suggested procedure of manufacturing dendritic Ag_2S nanowires with Cu nanoparticle seeds [Sillin et al., 2014] has shown that the performance of these networks irreversibly fades after several days, perhaps due to the brittleness of Ag branches with different crystallographic orientations [Bahadori et al., 2020], making them unusable in long-term neuromorphic applications. Thus 3D polymer composites developed in this work with Ag_2S NWs could play an important role in the future implementation of neuromorphic devices, especially with physically reconfigurable connections, as shown in Figure 4.5.

4.10 Manufacturing 3D neuromorphic composites

The first step to mimicking brain connectivity requires 3D integration. This section presents our approach in the fabrication of scalable composite for the *evolution-in-materio* with reservoir computing in randomly connected Ag/Ag_2S NWs in 3D memristive for computing tasks.

Knowing the average size of the Ag and Ag_2S NWs produced in our laboratory, we identified the optimal volume fractions of the material needed to develop 3D composites. First, the material we would like to fabricate should have some long-range conductors such as Ag NW and shorter-range connectivity realized by Ag_2S NWs. The average diameter of Ag NWs prepared and used in this work was estimated at D=124 nm and the average length L=42 μm. The average diameter of Ag_2S NW was D~160 nm, and the average length

was L~16 μm (shorter due to the material's brittleness). Second, the material is required to be able to undergo resistive switching. Thus primary conductivity needs to be mediated by Ag_2S NWs. Adding Ag NWs below the percolation threshold typically helps to reduce the overall amount of the required switching material. Thus the most optimal volume fraction of Ag NWs in the composite was estimated with the help of the percolator software introduced in chapter 2 [Frick and LaBean, 2021b] for 3D composites at 0.5 vol % (simulated with 10 segmented cylindrical wire with average curviness $\lambda = 1.2$, see chapter 2) and 1 vol % of Ag_2S NW (simulated with a straight cylindrical wire).

Our group fabricated nanocomposites with component concentrations as calculated above and different types of polymeric matrixes, including thermoplastic polymers (i.e., acrylic plastic casting, polydimethylsiloxane (PDMS)), light-cure acrylic encapsulant (Loctite 3922), thermosetting polymers (i.e., Ecoflex^{TM}, epoxy, EMbed 812), and biopolymers (i.e., agarose hydrogels and aerogels, DNA hydrogels). Among all types of tested polymer matrices, the author of the present work determined that PCL along with agarose aerogel with Ag/Ag_2S NW networks showed the best performance in overall electrical behavior. However, in this work, we will focus on PCL composites due to their excellent mechanical properties. Besides simple fabrication, PCL composites provide superior mechanical stability, excellent retention of resistive switching properties over long periods of time (>2 years) that can be reshaped or press-formed into the desired shapes.

PCL is a biodegradable polymer with a low melting point of about 338 K with a density of 1.145 g/cm^3 and excellent dielectric properties [Hirai et al., 2007], making this encapsulant suitable for laboratory use. In addition, PCL can be easily dissolved or swelled in several organic solvents with the best swelling achievable by chloroform or tetrahydrofuran with lesser swelling achievable in dimethylformamide [Bordes et al., 2010].

Conductive PCL films with Ag NWs were reported by Moreno *et al. [Moreno et al., 2012]*, and electrical properties of PCL/CuNP 3D composite were also reported by Sain *et al.* [Sain et al., 2013]. However, the fabrication of composites in the above cases produced only thin films, and in the last report, the thin films were crushed into pellets that were further pressed into a 3D form [Sain et al., 2013]. For neuromorphic applications, crushing a composite into powder or pellet form and further reshaping is undesirable since the process could severely damage the embedded network architecture. Thus in this work, our goal was to produce a 3D composite in one step, as described in the following section.

4.11 Preparation of PCL Ag/Ag_2S NW composite

The PCL Ag/Ag_2S neuromorphic nanoscale composite (NNC) can be manufactured in a few simple steps. First, PCL pellets (>99% InstaMorph^{TM}) with average Mw~120,000 (Mn~69,000) and melting temperature of ~70 °C were dissolved in chloroform (ACS grade, 0.75% ethanol, Fisher Scientific) overnight with 1:6 ratio by volume at 35 °C under magnetic stirring. Next, Ag NWs and Ag_2S NWs suspensions of appropriate concentrations were prepared in separate containers in ethanol (see details on sulfurization in section 4.2). Further, the Ag 0.5 vol% and Ag_2S NWs 5 vol% were combined in the same glass, and NWs were allowed to precipitate at the bottom of the vial, after which the supernatant ethanol was carefully replaced with chloroform leaving insignificant traces of ethanol. Finally, the mixture of NWs was added to the vial with PCL dissolved in chloroform. The resulting mixture with characteristic dark color was magnetically stirred at 100 rpm in an open glass container to allow the solvent to evaporate slowly for 6 hrs at 35 °C. Slow evaporation of the solvent along with benign agitation of magnetic stirring caused an increase in viscosity and resulted in an even distribution of NWs in the final solid phase of the product. Thus, at relatively small amounts of the composite material, compared to the volume of the vial and the size of the magnetic stirring bar, a single spherically shaped NNC particle can be formed once chloroform is evaporated, as is shown in Figure 4.7 (a).

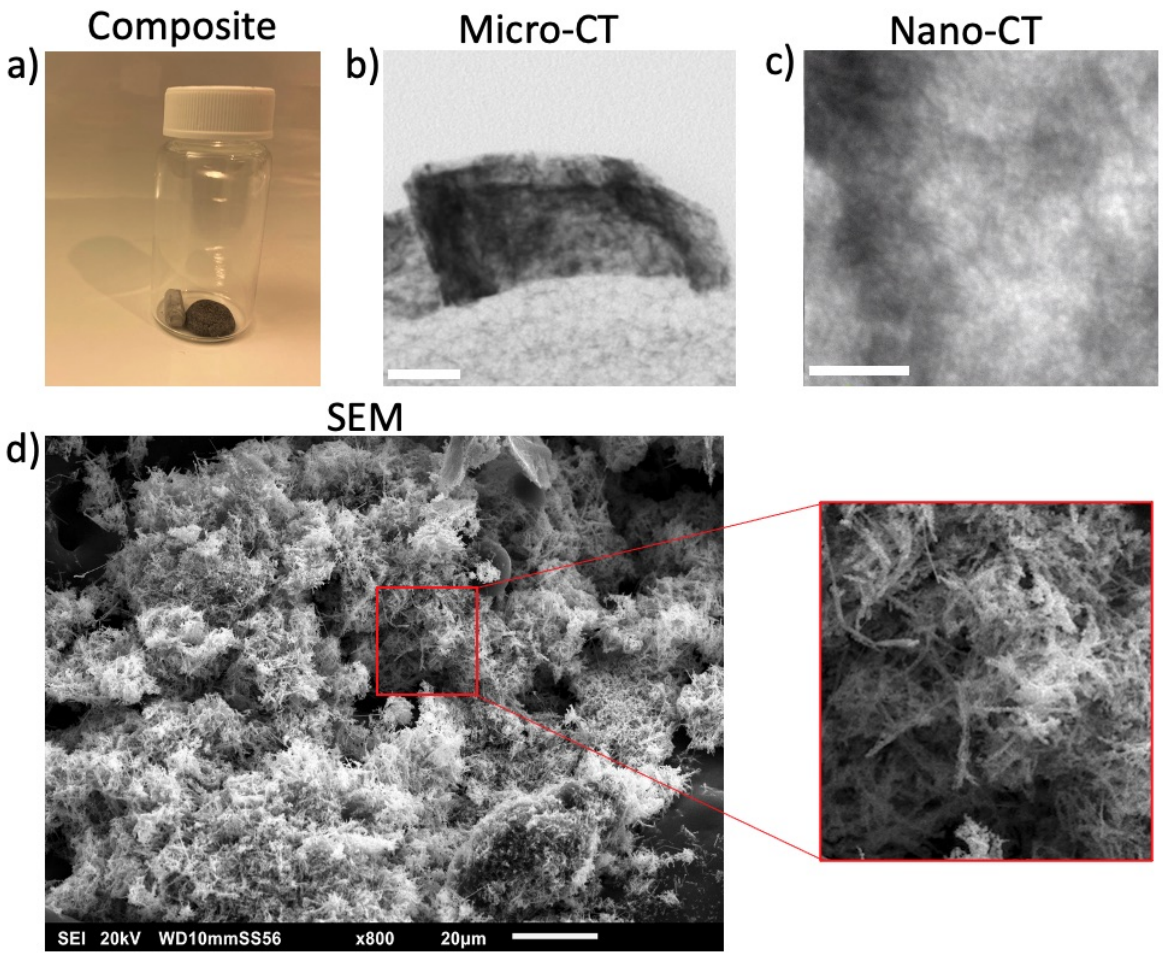


Figure 4.7: Characterization of NNC a) NNC composite at the bottom of a glass vial after chloroform has been evaporated (with magnetic bar) b) X-ray scan of a piece of the NNC composite (dark portion in the middle of the image) residing on a polyurethane foam (lower portion of the image) made with Micro-CT (Bruker SkyScan 1174) and 0.5 mm scale bar c) image of a portion of a 1 mm^3 piece of the NNC composite made with ZEISS Xradia 510 Versa X-ray microscope showing a projection of a cloud of Ag/Ag_2S NWs network (35 µm scale bar) d) SEM image of a fractured surface of the NNC showing dense network of randomly assembled NWs (Characterization by Ming Gao).

The resulting composite was cut into pieces via a razor blade for X-ray microscopy or mechanically fractured for SEM analysis. Unreconstructed images obtained with micro and nano X-ray microscopes are shown in Figure 4.7 (b) and (c) correspondingly. Despite insufficient resolution of the X-ray microscope to resolve individual wires, both images taken at different magnifications, exhibit projections of metallic Ag NWs and insulating Ag_2S NWs appearing as highly dense portions designated by blackening of the image and

resembling a 3D conducting network held together by the PCL. Figure 4.7 (d) shows an SEM image of the surface of the same sample obtained by fracturing the sample into two pieces showing a dense 3D network of NWs.

4.12 Preparation of microelectrode arrays

Due to the 3D structure of the NNC material, we were inspired by the microelectrode arrays technique of measuring brain activity in animals [Fan et al., 2011, Musk, 2019] and humans [Nurmikko et al., 2010a]. Therefore, we developed a simple method of in-house manufacturing of multielectrode microarrays. Microelectrode arrays are manufactured with 120 μm diameter and 25 mm long surgical steel acupuncture needles (Millennia ®) fixed with a dielectric epoxy. At the same time, the tips are aligned with a laser-drilled steel plate grid, as is shown in Figures 4.8 (a) and (b). The resulting microelectrode array possesses sufficient rigidity to be manually inserted into the NNC material to form a device.

The manufacturing process is performed in a few steps. The surgical stainless steel needles are first ultrasonically cleaned for 30 minutes to remove surface contaminants and improve binding with the epoxy. Further, they are aligned with two custom-made steel plates with laser drilled 6x6, 140 μm diameter holes. Two kinds of plates were produced with pitch equal to 100 μm and 200 μm to create arrays with different spacing. Once the tips of the needles are aligned and held by the static friction force that occurs between the needles and the holes of the steel plates, the needles are covered (excluding both ends) with an insulating two component epoxy (J-B Weld) and left overnight to solidify.

This simple method permits control of various array configuration parameters, such as the working length of the tips and the height of individual needles concerning the desired application. Figure 4.8 (c) shows a photomicrograph of a sample 4x4 array profile produced with this method, with the spacing between tips being equal to 200 μm. A custom-made adaptor board with zero insertion force (ZIF) sockets was also designed for connecting the microelectrode array to measurement electronics and is shown in Figure 4.8 (d). Finally, the sample can be installed on the microelectrode array, as shown in Figure 4.8 (e).

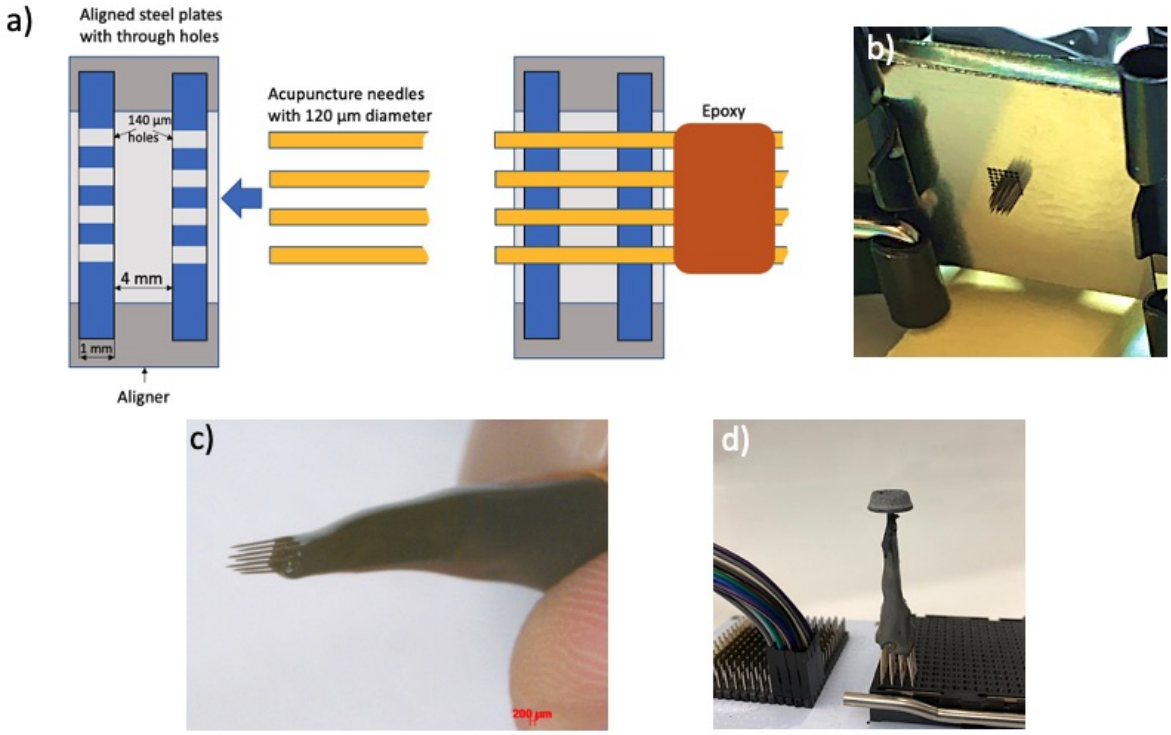


Figure 4.8: Microelectrode arrays made of acupuncture needles. a) Manufacturing process of the array with 120 µm diameter needles. b) Optical micrograph of the 100 µm pitch aligner plates holding acupuncture needles before application of the epoxy. c) Example of 200 µm pitch 3x3 array magnified under microscope. e) Microelectrode array with installed NNC material plugged into a ZIF socket adaptor.

4.13 Electronic characterization of the neuromorphic materials

The self-assembled NW network in the developed NNC exhibits interesting resistive switching properties that are not typically observed in other devices. This section explores the properties of the composite materials installed on the microelectrode array connected to an analog input and output system (National Instruments, cDAQ 9714, NI 9205, and NI 9264).

Compared to conventional thin-film memristor devices where each device is made with nearly identical characteristics, our devices (i.e., memristive composite installed on the

microelectrode array) have a network of NW memristors between each pair of the inserted microelectrode array. Each pin of the microelectrode array is connected to another one via a shared conductive network of NWs. Therefore, the memristive network creates a crosstalk effect, where voltage signal set to one pin will be observed through other pins with slightly different output signal characteristics. The difference of the IV characteristics between each pair of pins is attributed to the random nature of the internal organization of the conductive network but also due to the invasive and partially locally destructive effect of insertion of the microelectrode array into the material.

Despite the effect of the microelectrode array insertion, a high yield of success rate was achieved where every pin of the microelectrode array is connected to another one and forms a stable memristive connection. Typically nearby pins form low R_{ON} memristive connections while long-distance connections form high R_{ON} devices. However, in our research, we observed opposite configurations as well.

4.14 IV characteristics at different timescales

As simulations from previous chapter 3 showed, one of the signatures of a memristive network is negative differential resistance that is typically expressed in the non-monotonous decrease in the current while the device is switching into the OFF state. The primary cause of this effect is delayed action of memristive devices connected in series and in parallel in the network. If a voltage is supplied between two distant, randomly chosen points on a memristive network, the devices residing parallel to the lines of the applied electric field will experience the largest potential drop across them and thus will start to switch ON/OFF more quickly. Memristive devices that are oriented perpendicular to the applied field will typically have a smaller drop of potential difference and, therefore, will take longer to transition between ON/OFF states.

To test this, NNC with 0.5% wt. Ag and 2% wt. Ag_2S NWs were installed on a 3x3 microelectrode array with 200 µm pitch and two adjacent pins randomly chosen. The device was first initialized to create conductive Ag filament pathways with a 20 Hz sinusoidal and amplitude varying from 0 to 4 V in 20 s until a distinct hysteresis can be observed. Once initialized, a train of 4 negatives and 4 positive triangular voltage signals (4 V, 200 ms period) are sent while the current was measured through a 100 kΩ resistor as shown in Figure 4.9 (a). The hysteresis plot, where each loop is color-coded that matches the inset I-t and V-t plot, shows how the device switches to low resistivity R_{ON} state during the first 4 pulses and

switches to high resistivity R_{OFF} state during the next 4 pulses, each lasting 1 s. Interestingly, during the last 4 pulses, the device started to switch OFF starting with pulse 5 but then suddenly switched ON at pulse 8. Thus, the hysteresis shows that the tips of loops 5 and 8 are twisted by rapidly increasing current and could be either an effect of the introduction of slowly switching memristive devices oriented perpendicular to the field but also could be caused by Joule-Heating phenomena.

Similarly to above, right after the first 8 stimulating pulses, an increased number of pulses (9 negative and 9 positives) are sent with the same period and amplitude (Figure 4.9 (b)). Thus, the first half of the pulses induce the device to switch ON. However, a similar pattern is observed in Figure 4.9 (a), when the polarity of the input voltage is reversed. The current starts to fall during the first 3 pulses, denoting the device's switching into the OFF state. However, during the 4th positive pulse (pulse 13), the current jumped slightly up, after which the current started to drop again. While, as we already mentioned, this could be attributed to the network effect or Joule-Heating, a more detailed study would help to identify the exact reason for sudden and more so predictable changes in conductivity during the switching OFF of the device.

A typical ON/OFF procedure with pulses of various widths is shown in Figure 4.9 (c), where a similar material was used on the same electrode array as above, except a different pair of pins was selected. The device is first initialized, as mentioned previously, after which it is stimulated with +4 V 3 ms wide and 10 ms period rectangular pulses that switch the device into an ON state. Next, an application of the inverse polarity signal with -2 V amplitude and the same width and period switches the device OFF momentarily. Finally, the device is further switched back ON with the same train of pulses as before, but now the switching into the OFF state is performed with narrower pulses of 1.5 s width during the device takes longer to switch OFF. Interestingly, despite some variability in the shape of the ON pattern in all 3 cases in Figure 4.9 (c), the ON state is consistently in the last two instances. In contrast, the first one has a slightly smaller amplitude, perhaps due to the smaller amount of produced conductive Ag filaments in the NNC.

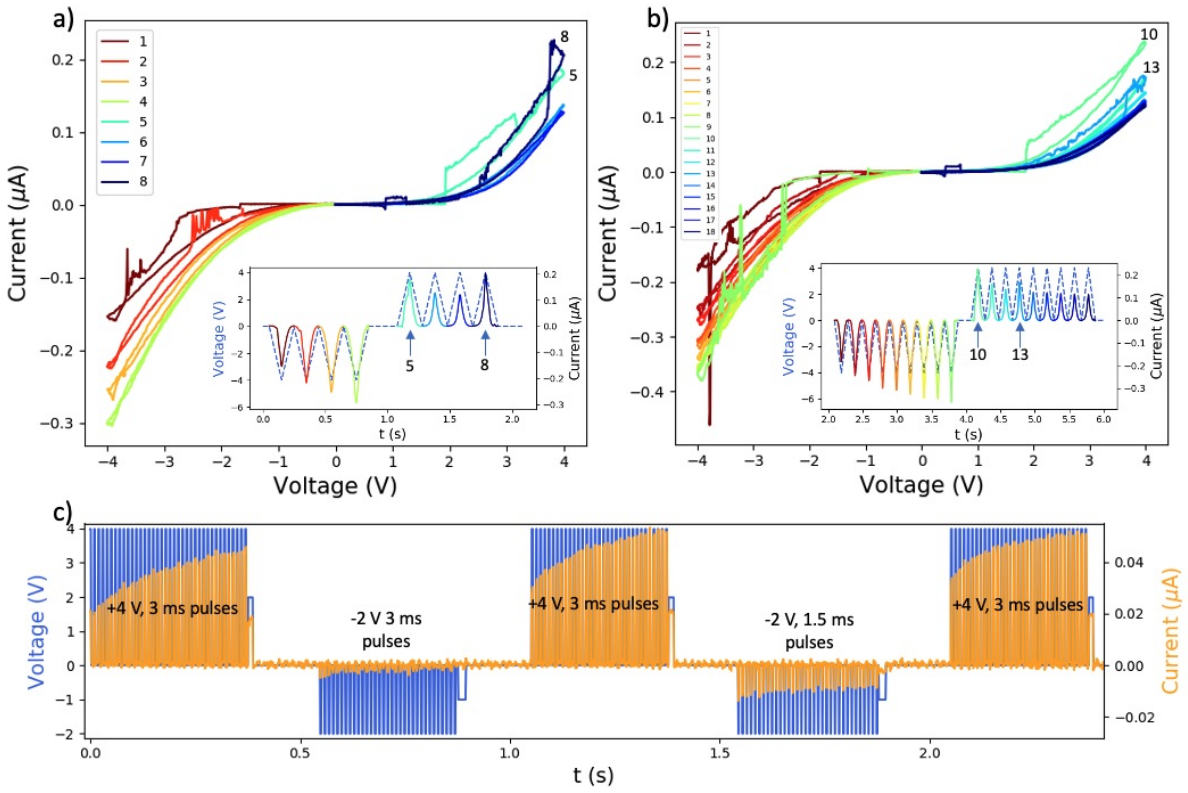


Figure 4.9: IV characterization of the NNC with 0.5% wt. Ag and 2% wt. Ag_2S NWs installed on a 3x3 microelectrode array with two randomly chosen pins a) Color-coded 4 negative and 4 positive 4 V triangular pulses and corresponding current measured from a 100 kΩ series resistor. The inset shows negative voltage turns the device ON, while positive pulses switch it OFF but not monotonously. Peaks 5, 6, and 7 show current decline until a sudden jump at peak 8. b) The second part of the stimulation followed right after the previous one but with additional pulses (9 negatives, 9 positive). The device gradually switches ON during a negative voltage sweep. In contrast, positive voltage sweep caused a similar pattern of decline of the current during the first 3 pulses (pulses 10-12) and followed by a rapid jump during pulse 13. c) stimulation of the device via different pairs of contacts on the electrode array with short pulses of different widths. The device switches OFF much faster with 3 ms pulses compared to 1.5 ms ones.

Despite the inertness of the stainless steel pins of the microelectrode array, the majority of devices exhibit bipolar switching mechanisms. It can be noticed, from Figure 4.9, that the resistive switching in the NNC device is bipolar, since in Figure 4.9 a) and b) device switches ON only during negative sweeps while it switches OFF during negative sweeps. Hence, each pair of stainless steel pins in the microelectrode array inserted into NNC represent a memristive device with a dedicated cathode and anode polarity. However, due

to the self-assembled nature of the material, it is impossible to predict the polarity of the pins, since the polarity of the memristive device formed between them is defined by the comparative abundance of Ag NWs near one electrode. The pin that has more Ag NWs in the neighborhood than another one can be taken as an anode since it will produce more Ag^+ ions if a positive voltage is applied, while the opposite electrode can be counted as a cathode.

4.15 State relaxation in NNC device

Like the biological synapse, NNC also exhibits an ability to forget its state by gradually reducing conductivity over time. However, the rate of forgetting appeared to be different for different batches of material. It was also found to depend on the previous history of stimulation. To measure the forgetting rate, two adjacent pins were chosen on the multi-electrode array Figure 4.10 (a), through which the device was set ON with a train of short triangular pulses of 4 V amplitude and 200 ms period while the current was measured with 1 ms wide rectangular pulses of 1 V amplitude and 2 s period and is shown in the inset of Figure 4.10 (b).

The law that governs the relaxation curve in Figure 4.10 (b) is a widely known stretched exponential function that was first proposed by Rudolf Kohlrausch to describe the discharge of a capacitor [Kohlrausch, 1854] and is presented by Equation 4.5. Equation 4.5 is a general form of exponential law where A, τ and β are constants and t is a variable that can change between 0 to $+\infty$. The stretched exponential is a phenomenological law that can be reduced to other well-known exponential distributions by changing β. For example, when β=2, it turns into Gaussian law distribution, or when β=1, it becomes a usual exponential distribution. In physics and engineering, a stretched exponential function describes relaxation phenomena in disordered systems [Elton, 2018].

$$f_\beta(t) = Ae^{-(\frac{t}{\tau})^\beta} \tag{4.5}$$

Fitting the data from Figure 4.10 (b) with Equation 4.5 yields τ=11.6 s and β=0.89 that can be translated as that the magnitude of current drops e times every 11.6 s, while the drop in the current is quite sharp, due to large β, it approaches 0 A in about 1 min. To test the influence of the history of stimulation on the relaxation rate, after the previous measurement, the NNC was stimulated with positive, negative, and again positive 4 V

square pulses, sequentially switching the device through ON/OFF/ON cycle after which the current was measured with 1V, 1 ms square pulses with 2 s interval as shown in the inset in Figure 4.10 (c). As a result, the relaxation time constant increased to τ=50.8 s, while the distribution became less sharp due to β=0.46.

The above results can be compared to similar fittings performed with a W/WO_x/Pd thin film TMO memristor stimulated with 1.3 V, 400 μs pulses, and 60 ms period. Simultaneously, resistive states were read by 0.5 V, 3 ms pulses at 1-sec interval shown in Figure 4.10 (d) adopted from Chang *et al.* [Chang et al., 2011]. The probability of recalling human memory, or equivalently the probability that a person can recall an event after some time in the future, also exhibits stretched exponential relaxation law and is shown in Figure 4.10 (e), also adopted from Chang *et al.* [Chang et al., 2011].

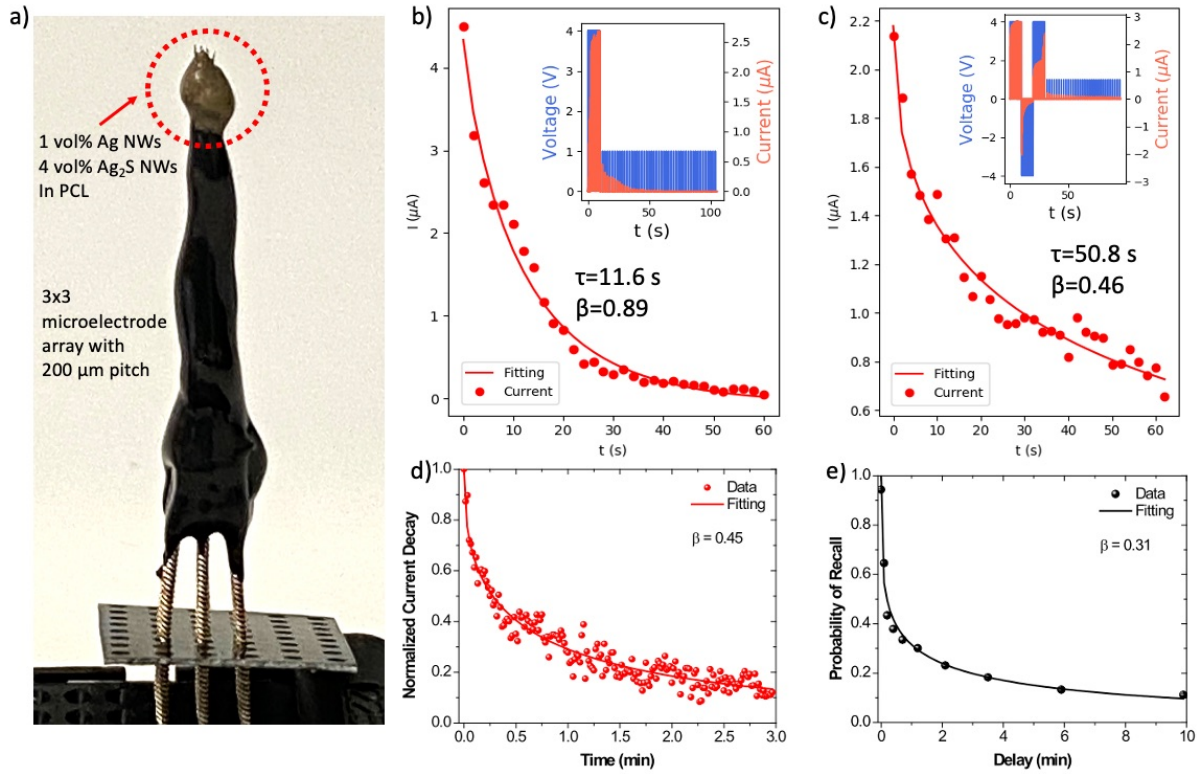


Figure 4.10: Relaxation of high conductivity state in the NNC and comparison to mechanism in WO_3 memristors and human memory. a) NNC installed on a microelectrode array b) NNC stimulated with 4 V positive 200 ms triangular pulses and current read with 1 V and 1 ms wide rectangular pulses with 2 s period. The fitting of stretched exponential function line yields fast relaxation with decay constant τ=11.6 s and β=0.89 c)The same pair of microelectrode tips is stimulated with a series of positive, negative, and positive 4 V triangular pulses of 200 ms width after which current is measured again with 1V, 1 ms wide rectangular pulses at 2 s interval. After this stimulation, the fitting of the stretched exponent exhibits a much longer and less steep decay rate of τ=50.8 s and β=0.46 d) similar experiment reported by Chang *et al.* [Chang et al., 2011] where thin-film W/WO_x/Pd memristor was set ON with a train of 1.3 V rectangular pulses, 400 µs wide and 60 ms interval, while the current was collected by stimulating the device with 0.5 V pulses of 3 ms width at 1 s interval. The fitting of the stretched exponential function shows comparable characteristics to our measurements with τ~50 s and β=0.45 e) Forgetting curve of human memory is reproduced from Chang *et al.* [Chang et al., 2011].

While in artificial neural networks, the strength of the synaptic connection is represented by a single scalar value in biological synapses, the strength of the synaptic connection (or equivalently synaptic efficiency) R is described by the product of three variables, as follows: $R = npq$. In this relationship, n and p are presynaptic variables describing the

number of neurotransmitter release sites and probability of synaptic release per site. On the other hand, q is a postsynaptic variable attributed to the change in postsynaptic membrane conductance [Koch, 1998]. While the probability of synaptic activation p has been identified, for single release sites it is found to be around or less than 30% [Auger and Marty, 2000]. The change in the probability p is largely responsible for so-called short-term synaptic plasticity that can last from milliseconds to minutes. On the other hand, the change in the number of synaptic release sites n and postsynaptic response q are mostly responsible for the long-term plasticity that typically lasts 10 s of minutes to hours [Yang and Calakos, 2013].

NNC is a network of synapse-like connections formed through ionic channels of individual Ag_2S NWs and it exhibits similar properties of synaptic efficiency as biological synapses. One can imagine the density of Ag nanoclusters formed in the Ag_2S-β phase during stimulation typically increases and hence induces longer relaxation and typically higher conductivity. Thus the density of Ag nanoclusters that mediate conductivity and the averaged inverse distance between them is equivalent to n and p from the above relationship. Thus the high concentration of Ag is equivalent to a large number of release sites, but this number is not sufficient to define the synaptic efficiency of the Ag_2S NW network. Thus the average inverse distance between the clusters p that defines how close the Ag nanoclusters are distributed and along with the concentration n serves as a good description of synaptic efficiency.

4.16 Rate dependent plasticity

According to Hebb's original rule, the change in synaptic weight depends on the pre-synaptic and post-synaptic activity and should increase when these activities are correlated [Sejnowski, 1977]. In biological neurons, and in particular, the visual cortex, the strength of the synaptic connection between i'th (presynaptic) and j'th (postsynaptic) neuron follows a modified Hebbian learning rule as described by Equation 4.6 [Bienenstock et al., 1982] [Koch, 1998]. The change in scalar weight w_{ij} depends on the product of presynaptic activity V_i (such as spiking frequency) and a function ϕ of postsynaptic activity V_j (such as membrane conductivity or membrane potential) with some variable threshold θ_m.

$$\Delta w_{ij} = V_i \phi \left(V_j, \theta_m \right) \tag{4.6}$$

The function ϕ is depicted in Figure 4.11. The magnitude of θ_m sets the threshold

for synaptic modification and results in increased or decreased synaptic modification, or equivalently weight w_{ij} based on the immediate history of stimulation [Bienenstock et al., 1982]. In particular, Dudek *et al.* showed experimentally that synaptic strength depends on the frequency of presynaptic stimulation, where synaptic efficiency w_{ij} decreases after the periods of low frequency and oppositely, increases after high stimulating frequency [Dudek and Bear, 1992].

However, the threshold θ_m depends on the postsynaptic activity V_j and defines zero-crossing of the function ϕ. Thus even during the high activity of presynaptic neuron Vi the weight w_{ij} can decrease due to the higher activity of postsynaptic neuron V_j according to the relationship $\theta_m = \langle V_i^2 \rangle$ moving to $\theta_{m'}$ oppositely increase during the loser activity of the postsynaptic neuron $\theta_{m''}$ as shown in Figure 4.11.

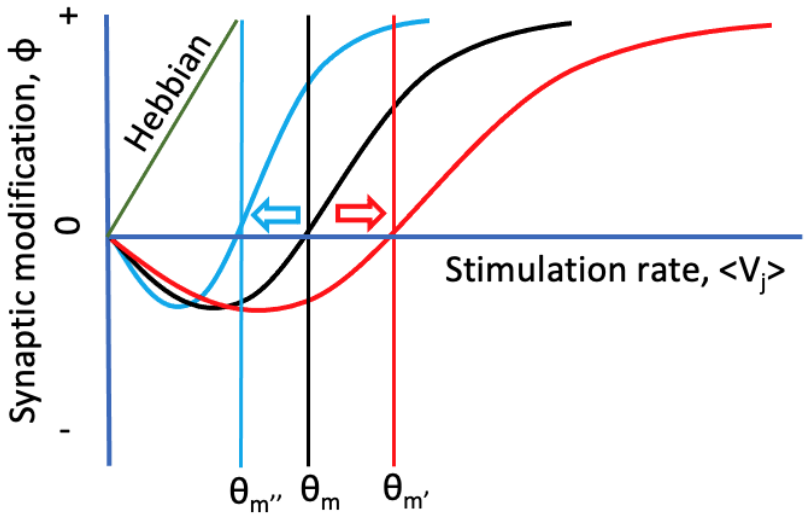


Figure 4.11: The sliding threshold in Hebbian for biological neurons. When the activity of the presynaptic neuron is high, the threshold θ_m slides right $\theta_{m'}$, inducing a decrease in synaptic weight. Oppositely, when activity is low, threshold slides left ($\theta_{m''}$) inducing an increase in synaptic weight. The sliding threshold changes proportionally to a time-averaged supralinear activity of presynaptic neuron $\theta_m = \langle V_i^2 \rangle$ [Koch, 1998].

Thus unlike in pure Hebbian learning, where the weight growth is unbounded (Figure 4.11 green line), the sliding threshold allows the neuron to stabilizes its activity.

A similar phenomenon was observed in the NNC material developed in our laboratory and other memristive devices. Below we tested the sliding threshold behavior in the 1 vol%

Ag NW 4 vol% Ag_2S NW PCL NNC device installed on a 3x3 electrode array in Figure 4.12, repeating the experiment reported by Chang *et al.* with WO_x memristor[Chang, 2012]. In the experiment, two adjacent pins of the microelectrode array in Figure 4.12 (b) were chosen and connected to a programmable function generator (Siglent, SDG1032x) and an analog voltage measurement device (National Instruments, NI 9205). The device was stimulated with a train of 4 V and 1 ms rectangular pulses of different frequencies (40 Hz, 10 Hz, 2 Hz, 25 Hz, 1 Hz, and 10 Hz) with results shown in Figure 4.12 (a). At first, the device was stimulated with fast 40 Hz pulses, resulting in a rapid increase in conductivity. Further, during 10 Hz stimulation, the current started to decline noticeably from ~4.8 μA to ~4 μA, obviously switching the device into an OFF state. After that, the current continued to decline during the 2 Hz stimulation but underwent a rapid increase during the 25 Hz stimulation that moved the maximum of current slightly above 4 μA. After the decline during the 1 Hz stimulation, the current rapidly increased after 10 Hz stimulation. The pattern of decrease of the current, or equivalently, the synaptic weight, after rapid stimulation followed by the decrease of the current, resembles behavior as biological synapses and is referred to as sliding threshold long term potentiation or long term depression. After high-frequency stimulation, synaptic connection undergoes reconfiguration that engages decreased synaptic weight during low-frequency stimulation [Lee and Kirkwood, 2019].

A similar experiment was performed on a commercially available Ag_2Se chalcogenide memristor (Knowm) shown in Figure 4.12 (c) and (d). Due to the technological limitations, the device was stimulated with 1.5 V pulses while the duration and period of the stimulation pulses remained the same as above. As shown in Figure 4.12 (c), the sliding threshold effect in the Ag_2Se chalcogenide device is much smaller than in the NNC device. This difference could be attributed to the specifics of the thin film device's architecture depicted in Figure 4.12 (d).

The original experiment performed with a film WO_x device exhibits a nearly identical sliding threshold effect and is shown in Figure 4.12 (e) with the device depicted in (f) [Du et al., 2015] [Chang, 2012]. While our NNC device closely reproduces the same behavior the WO_x memristor, except the higher noise in the NNC device due to lower current that can be explained by a much smaller overall cross-sectional area of the device compared to the WO_x and by the difference in the properties of the materials, such as the mobilities of Ag^+ and V_{ox} (oxygen vacancy). NNC device also showed a faster increase in current during the second 10 Hz stimulation.

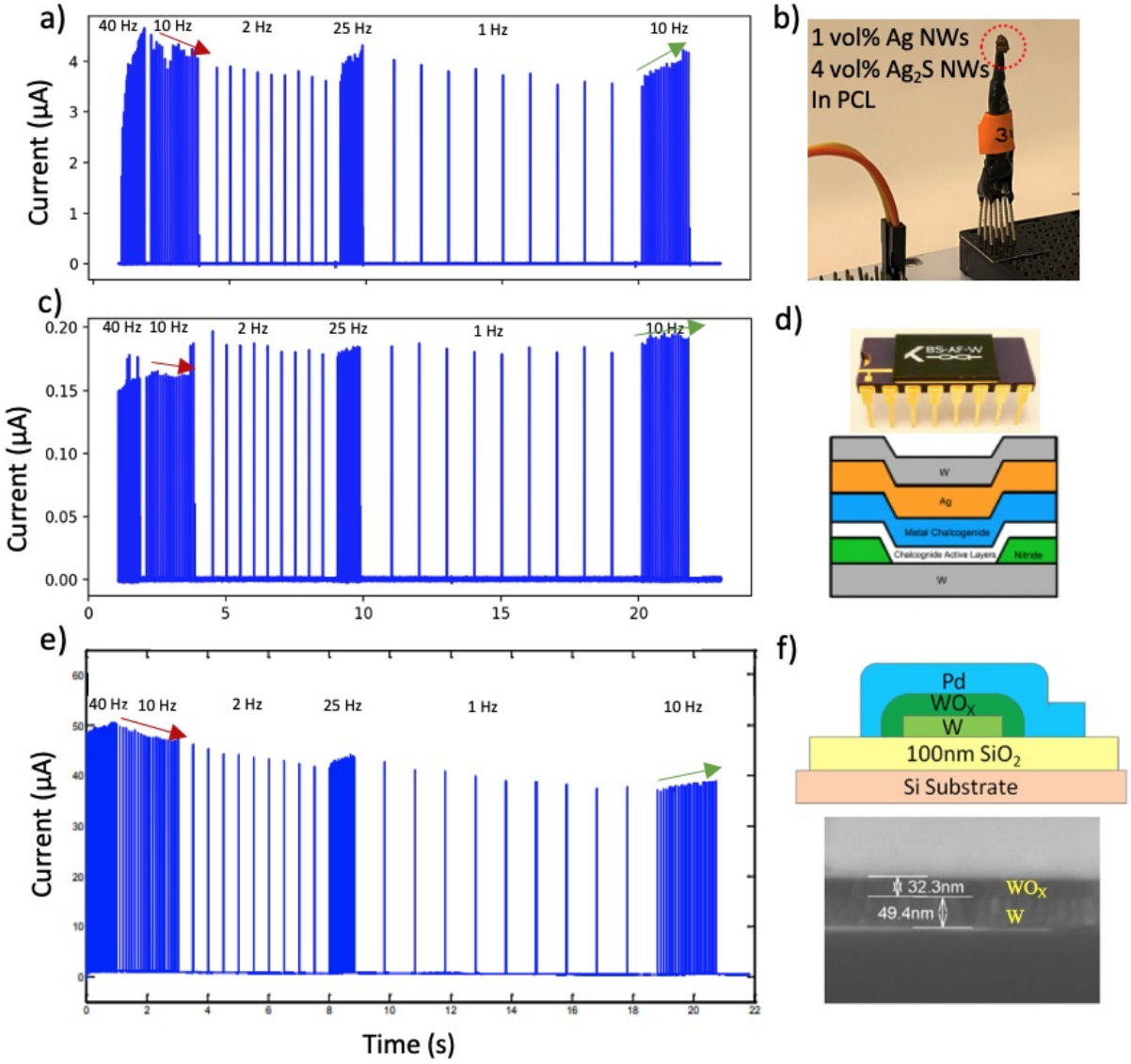


Figure 4.12: Rate dependent resistive switching in various memristive devices a) NNC device stimulated with 1 ms 4 V pulses at various frequencies exhibits reduction in conductivity at 10 Hz stimulation immediately after high rate stimulation of 40 Hz and experiences an increase in conductivity after 2 Hz and 1 Hz b) actual NNC device installed on 3x3 microelectrode array with 200 μm pitch c) similar response obtained from commercially available Ag chalcogenide memristor (Knowm) with d) a photo of the device and depiction of layered structure (e) an original experiment performed with WO_x thin film memristor of 200x200 μm size and (f) its structure both adopted from Chang *et al.* [Chang, 2012]

To explain the sliding threshold phenomenon in thin-film WO_x memristors, Du *et al.* de-

veloped a “second-order” dynamic two-state variable memristor model with one variable representing the effective area of conducting region and another representing the effective mobility of oxygen vacancies V_o [Du et al., 2015]. The dynamic equations are constructed to capture the effect of voltage stimulation and effective decay, each with its own time constants that describe the long and short-term effects in V_o mobilities and effective channel width similar to the sliding threshold.

4.17 Multichannel stimulation of the NNC device

The previous section showed the rich properties of the nonlinear dynamic response of the NNC device performed in a single voltage input and single current output configuration. However, since the nanoscale memristive network embedded into the PCL matrix connects all parts of the material randomly, the cross-talk between inputs and outputs in a multi-input and output configuration will produce an even more complex dynamic response. Thus, this section presents the electronic characterization of the NNC device in such multi-electrode configurations and discusses applications in linearizing complex nonlinear problems such as XOR.

4.18 Response of NNC to voltage spikes

To show the diversity of nonlinear response of the 3D nanoscale memristive network, the NNC sample was stimulated with square voltage pulses of two different durations while the current was measured. The sample was installed on a 4x4 multi-electrode array (100 μm pitch) with 3 voltage inputs and 3 current outputs. At first, packets of ten 250 ms wide square pulses of absolute 6 V amplitude were sent with alternating polarity and changing sequence through inputs 0 and 1 as shown in Figure 4.13 (a) (upper time axis) and corresponding current response in (b). During the first 3 packets in the sequence, the polarity of the packets on inputs 1 and 2 had opposing signs. During the 4th packet, only input 0 was set to negative polarity while input 0 was turned off. Further input 0 was turned off while input 1 was stimulated with four packets of pulses of alternating polarity.

During the second run, the sequence and the amplitude of pulse packets remained the same, but the width of pulses was shortened to 50 ms as shown in Figure 4.13 (a) (bottom time axis) and the corresponding current response in Figure 4.13 (c).

The current response observed in 3 different outputs in Figure 4.13 (b) and (c) expose a few notable features resembling its reservoir nature. In particular, two current patterns (b) and (c) look nearly identical, except for differences in current amplitudes caused by the difference in the overall action of the electric field in the samples. Second, there is an active property of convex current patterns that causes negative differential resistance in the output 2 during the 5th and 7th packet that is depicted by an immediate increase of the current at the beginning of the packet followed by its decrease and then increase (Figure 4.13 (b)). This phenomenon was not observed in (c) due to the narrower bias peaks in the latter case. Third, besides local activity of the system [CHUA, 2005] expressed in the negative differential resistance resembled by convex current pattern, various rates of current change observed during stimulation with single and two channels speak about the different conductivity paths with different memristive characteristics that were established between input and output electrodes. Finally, the switch in the polarity of the current response at the beginning of packet 3 in (b) and (c) emphasizes the influence of the network structure of the conducting pathways formed between the inputs and outputs. Namely, reversing the polarities of input signals between packets 2 and 3 causes the device to reset and redirect the current's direction in all output channels as if it would be performed in an electronic circuit with an active device such as a transistor.

An analogy of the device's behavior in Figure 4.13 could be depicted as similar to natural phenomena. The internal memristive networks inside of the device can be imagined to be similar to rivers at the bottom of the valleys in a mountainous area. The flow and direction of the water in the rivers are governed by the location of the artificial dams (input channels), the location of the ocean (output channels), and the widening/narrowing of conducting channels during the system's evolution. The value of the electrical potential on the input channels specifies the altitude of the dam in the mountains. In contrast, the width of the channels is dictated by the initial architecture of the network and its evolution.

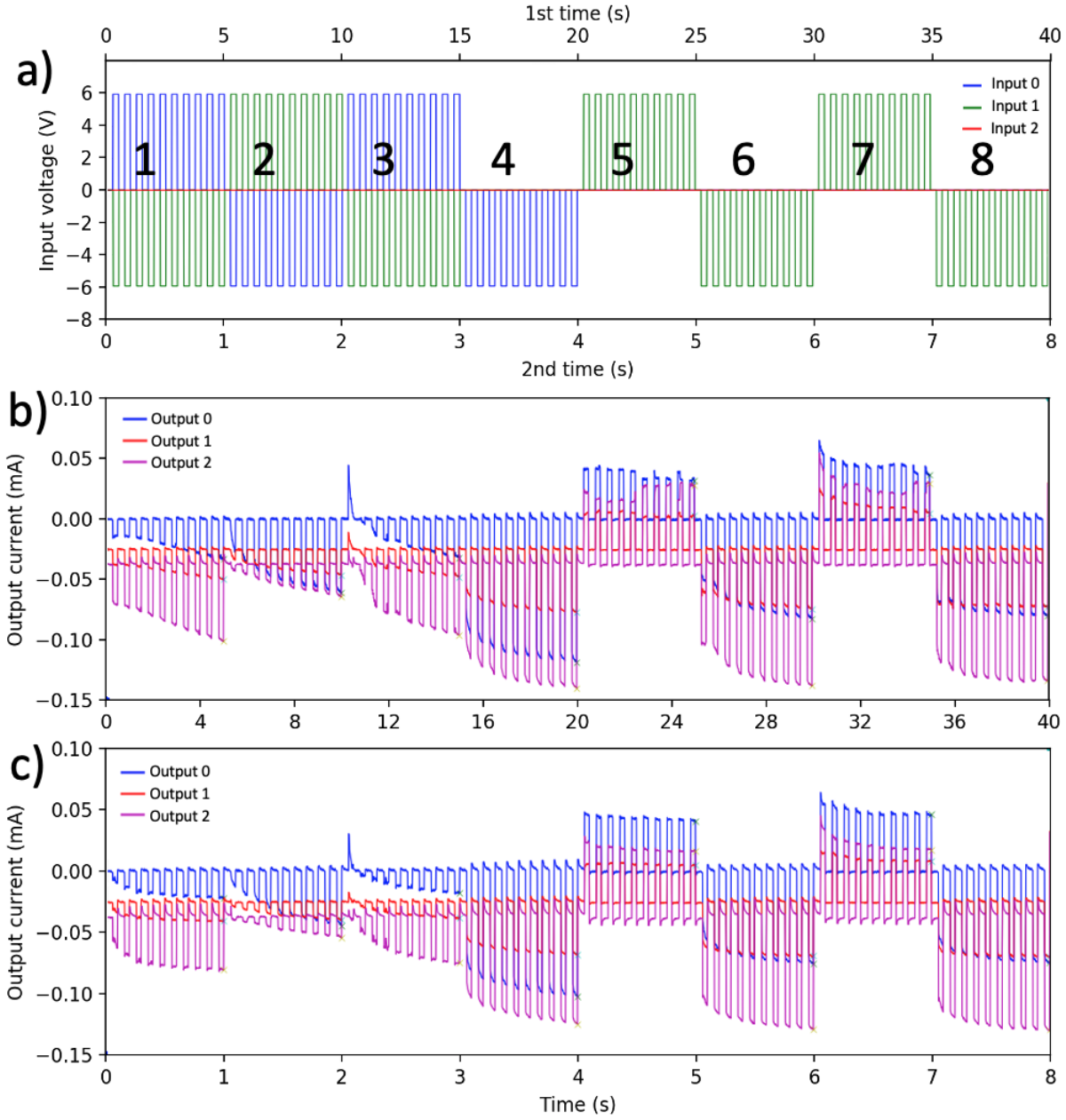


Figure 4.13: Typical transient characteristics of the NNC device with two inputs and three outputs. a) two input voltage bias patterns at different time scales with 250 ms pulse width (upper axis) followed by 50 ms pulse width (lower time axis). b) Corresponding current read from 3 separate channels during 250 ms pulse width stimulation and c) during 50 ms pulse width stimulation.

To complement our results, we performed a simulation of an equivalent Ag_2S/Ag NW network and captured its temporal evolution, shown in Figure 4.14. The network is stimu-

lated with a DC bias where input-0 is set to 1 V, and input-1 is kept at 0 V for 50 s, and all memristors were set to intermediate magnitude of memristance, i.e., $\frac{R_{on}+R_{off}}{2}$. The circuit was simplified in that the switching occurs only within Ag_2S NW without the need for the source of Ag ions from Ag NW, which was observed experimentally.

Here we introduce circuit entropy to aid the visual interpretation of the process equivalent to the amount of information needed to describe the distribution of currents in the circuit. There are multiple ways that the information content of a system can be defined. For example, algorithmic complexity is defined as the length of the shortest program, written in any predefined programming language, that completely describes the distribution of the currents in a circuit [Kolmogorov, 1998]. However, this approach is computationally inefficient since it requires a guarantee that there is no other, more efficient algorithm.

Here we use a more friendly mathematical formulation, the graph or a circuit entropy S. It is defined similarly to von Neumann entropy, a variant of Shannon entropy [Shannon, 1948] shown in Equation 4.7. The value of ρ_{ij} is defined between a pair of nodes n_i and n_j connected by an edge e_{ij} in the graph and is proportional to the absolute value of current going through the edge divided by the sum of absolute values of currents going through each edge. If the edge does not exist, then the value of ρ_{ij} is set to 1. Mathematically, ρ_{ij} is the density matrix of the graph and is defined as $\rho_{ij} = \frac{|I_{ij}|}{\sum_{i,j}|I_{ij}|}$ where $\rho_{ij} = 1$ when either $I_{ij} = 0$ or $\sum_{i,j} |I_{ij}| = 0$.

$$S = -\sum_{i,j}^{n} \rho_{ij} \log(\rho_{ij}) \tag{4.7}$$

Initially, the current in the network is small and roughly equal in each wire of the network. Thus, the amount of information needed to describe such a system of evenly distributed current among the small components of the network is high; hence the thickness of all connections is relatively similar in Figure 4.14 at time 0 s.

As time progresses, memristors in the network are switching and new, better-defined paths are created within the network, requiring less information to describe such a system exhibited by the entropy plot. For example, at the time step of 36 s, a conducting pathway connecting input-0 and both output-0 and output-1 start to be visible. While the current in both channels instantly increased, the entropy underwent an instant decrease.

Finally approaching 46 s, the memristor pathway turns into a major path of least resistance, and the current rapidly jumped in both output channels. However, immediately after that, the current in the output-1 cannel dropped, while the current in the output channel-1

continued to increase. This phenomenon is similar to the current drop observed in the NNC device shown in Figure 4.13 (b) during the 5th and 7th packets. Notice the thickness of the upper channel in Figure 4.14 at 46 s slightly decreased when the lower channel appeared slightly after 46 s.

Interestingly, the switching behavior in the circuit takes place in abrupt falls followed by periods of plateau that is depicted by the circuit entropy plot. At close observation, the plateau is not straight. There is a slight increase to the right of the center of the plateau before the next drop. This increase has been attributed to an increase in current distribution among other network elements after the main conducting channel was created. The behavior in this NW electronic circuit initiated by constant DC electric field stimulation is quite complex. As we will show in the next section, this complexity can be harnessed for neuromorphic computation.

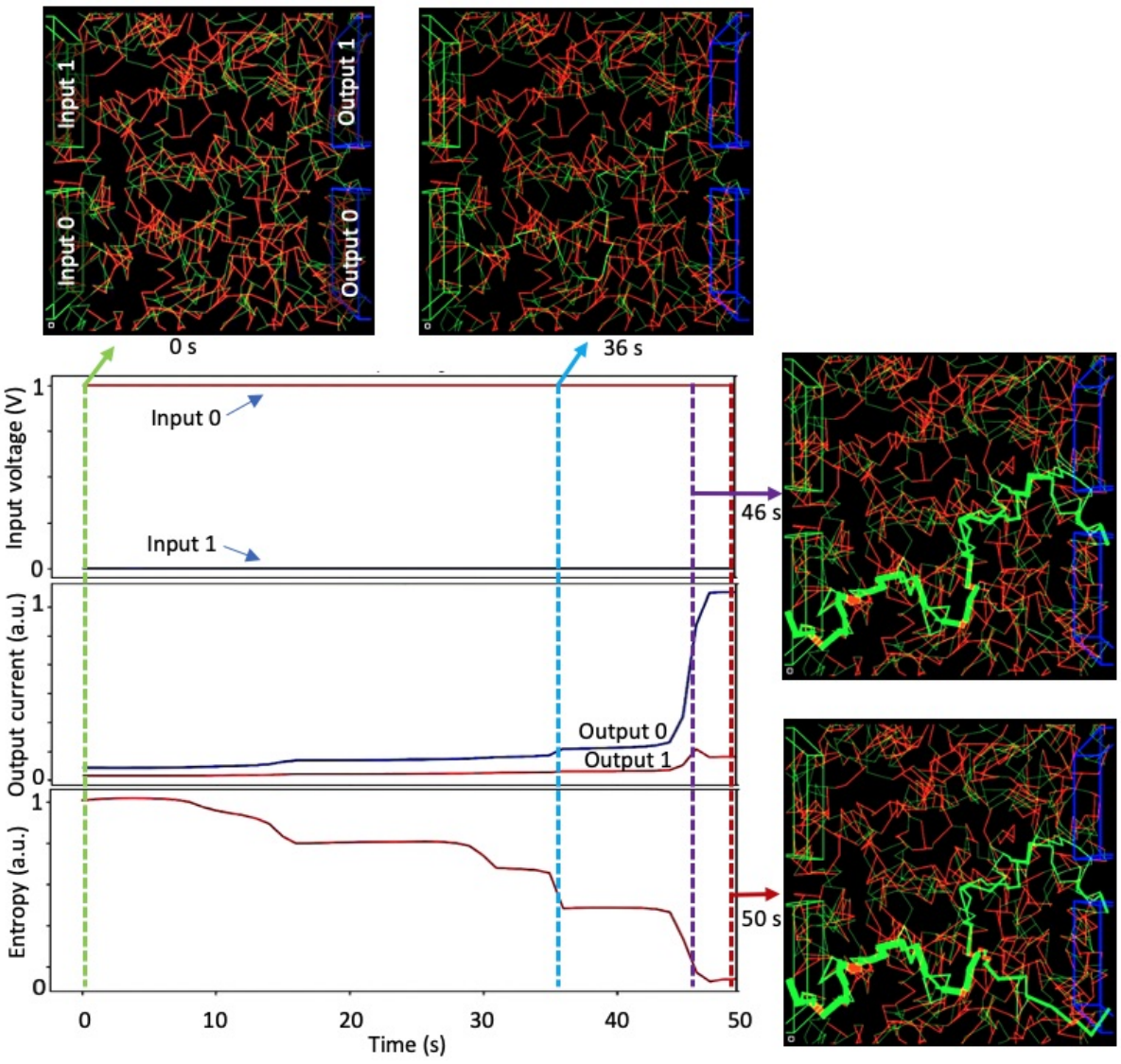


Figure 4.14: Simulation of the temporal evolution of a memristive nanowire network with 2 voltage inputs marked with green boxes and current outputs, marked with blue boxes. DC input voltage was applied to input 0 at 1 V while input 1 was kept to 0 V. Green lines denote memristive Ag_2S NWs, red lines denote resistive Ag NWs and thickness of the wire is proportional to the magnitude of current going through a wire. Output currents and circuit entropy, based on conductivities of each element are measured.

4.19 Linearizing XOR problem with NNC

Inspired by the results from multichannel experiments, we introduce an approach to harnessing NNC to linearize the XOR problem. Following the methods in theoretical chapter 3, linearization of a complex problem is achieved by projecting logical representation of the problem into the high dimensional space of the memristive reservoir. In a nut shell, this approach is equivalent to binary threshold switching logic where the output I of a neuron with n inputs V_i weights w_i is calculated with the following relationship Eq 4.8 where f is an activation function such as sigmoid, tanh or Heaviside [Rosenblatt, 1958].

$$I = f\left(\sum_{i=1}^{n} V_i \omega_i\right) \tag{4.8}$$

The binary state of the output I is "true" when its value is greater than some predefined threshold T or "false" if it is less. Thus, based on this approach, one can build arbitrary logic gates or even networks with complex behavior [Borresen and Lynch, 2012].

Thus, the NNC material is equivalent to the first layer in a typical two-layer feed-forward neural network (2 input and 1 output neurons) used for solving the XOR problem. The proposed approach only makes the problem linear, meaning that one still needs a linear artificial neuron to convert the result into logical 0 and 1 or "true" and "false" notation. In our experiments, we applied software logistic regression classification in the last layer to account for the identification of the threshold [Murtagh, 1991]. Furthermore, to aid the visual presentation of the multidimensional output from the reservoir, the data is processed with principal component analysis (PCA) that shows 2D projections of the data on the axes with the highest variance [Pearson, 1901].

Here the experiment was conducted on the NNC reservoir material (1% wt Ag NWs and 4% wt Ag_2S NWs in PCL) where the 2 variable XOR problem was linearized through finding a set of static control voltages with Hyperopt optimization algorithm [Bergstra et al., 2015] which loss function was constructed to optimize the linearization of the problem as is shown in Figure 4.15 (e).

The XOR table used in these experiments was converted into a table of voltages (logical "true"=6 V and logical "false"=-6 V) that was fed into the reservoir as triangle-shaped signals with 50 ms width through the 4x4 microelectrode array with two static voltages. At the same time, the current was read through 4 output terminals. Thus, a single sample consisted of feeding each row in the truth table repeatedly 4 times and then repeating it once again, yielding a total of 24 measurements to collect statistics for calculating decision boundaries.

The resulting currents from 4 output terminals were labeled and rescaled, after which the score of the decision boundary was calculated with a logistic regression algorithm [Haykin, 2009].

Due to technical limitations in our measurement equipment, we were restricted to a maximum of 10 V input. Still, at higher voltages, the dynamics of state change in these systems are significantly higher. Thus according to our measurements, at 15 V, the state of the memristor can change a few orders of magnitudes faster than it was observed at 5V.

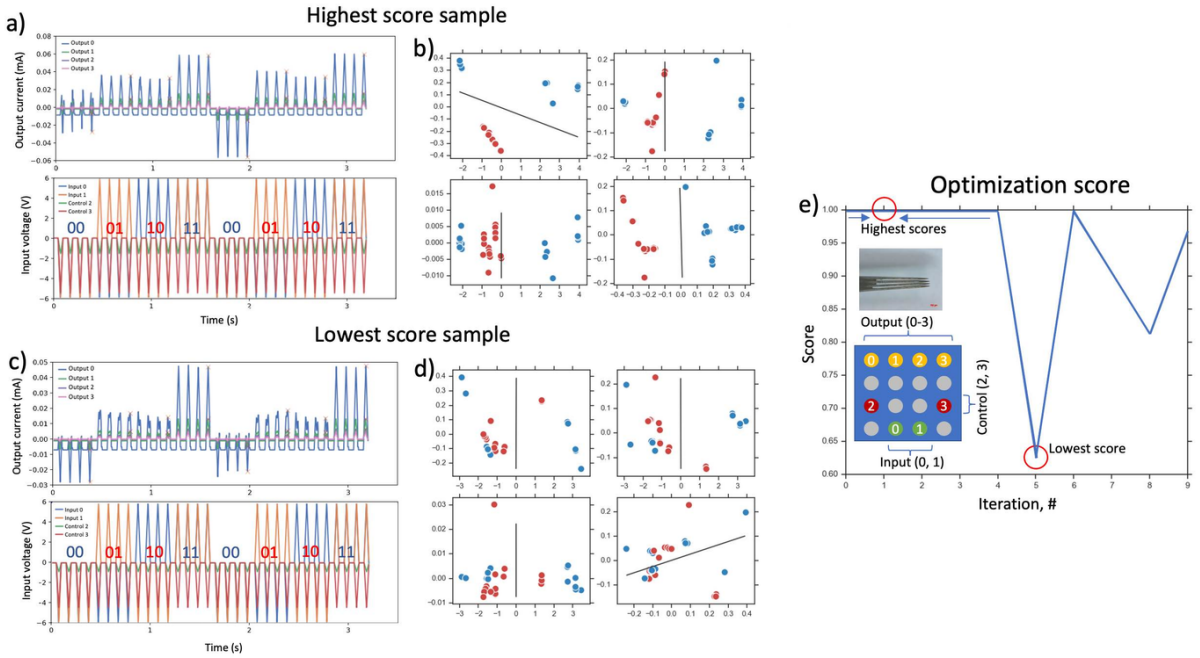


Figure 4.15: Linearization of a two-variable XOR problem with the NNC device. a) The output currents from 4 channels (top) and corresponding input voltages along with the control voltages (bottom) for one of the highest score samples. The color-coded sequence of double digits represents the XOR truth table where the blue color corresponds to the “false” state and the red color corresponds to the “true” state. c) is similar to a) except for the lowest score sample. b) PCA projections from the top 4 combinations of the output currents showing linear separability in two projections (top left and bottom right) with decision boundary determined with logistic regression. d) PCA projections showing that the network was configured to a state that did not linearize the problem. e) The progress of optimization showing that the network linearized the XOR problem during the first 5 iterations. The top inset shows the photo of the 4x4 micro-electrode array used in the configuration and the bottom inset shows the configuration of the pins.

As shown in Figure 4.15 (e), the search optimization algorithm was able to locate the

optimal configuration of the two static control voltages with which the system linearized the XOR problem, as is shown in Figure 4.15 (a) and (b). However, we noticed that the material typically drifted after around 100 steps of optimization if it was stimulated with voltages above 3 V, making it difficult to reuse the state with previously trained static control voltages. In other words, optimal control voltages that gave a 100% accuracy score at one point exhibit stable high score performance for a limited number of iterations, with a gradual decline of the score and eventually falling to 50%.

On the other hand, lower voltages did not induce a dramatic change in the internal network configuration and showed more robust performance over a longer period of time. Thus we assume a need to either identify a state reset protocol that would set the network into an equivalent state each time before the new batch of experimental data is fed. On the other hand, such instability of states, in our opinion, makes it a perfect candidate for true reservoir computations where very long-term memory relationships increase the total dimensionality of the state space of the system.

Hyperopt uses a combination of Bayesian optimization and Parzen tree estimators [Bergstra et al., 2015]. Its objective has long and short-term goals, where even if it could locate local minimum, it persists in exploring randomly other areas of unexplored state space. In addition, we still are unaware of how rough the surface of the underlying optimization function is, which, together with the previous statement, might explain the roughness of the optimization trend in Figure 4.16 (e) and necessary conditions for finding a plateau with the highest score.

4.20 Linearizing two-layer XOR gate with NNC

Following the linearization of the two variable XOR problem, we repeated the experiment with the 3 variable XOR (3XOR) problem that is shown in Figure 4.16. The 3XOR problem is constructed by applying the XOR operator between two adjacent variables whose truth table is shown as a color-coded sequence of triple digits in the inset of Figure 4.16 (a) and Figure 4.16 (b). In this figure, the black triplet represents logical “false” while the red triplet represents logical “true”. Strictly speaking, this particular network is a two-layer XOR gate circuit.

In this setup, 4 static control variables were used to optimize the solution in a similar fashion as in the previous XOR setup. The logical values of the variables were converted to 6 V voltage, and the optimization was carried by the Hyperopt algorithm. The progress

of the optimization is shown in Figure 4.16 (e), with the inset depicting the configuration of the pins on the microelectrode array. Due to the increased number of control voltages and the complexity of the problem, it took the algorithm nearly 45 steps to find an optimal solution.

The problem is quite complex, and to be solved would require at least 4 artificial neurons. Thus the NNC device here represents 3 artificial neurons from the first layer. The PCA projections in Figure 4.16 for the best sample (c) and the sample with the lowest score (d) exhibit a similar pattern as in the previous example. In particular, according to the highest score, the NNC device successfully linearized the problem in 2 out of 4 projections while the results of the lowest score sample turned linearly inseparable.

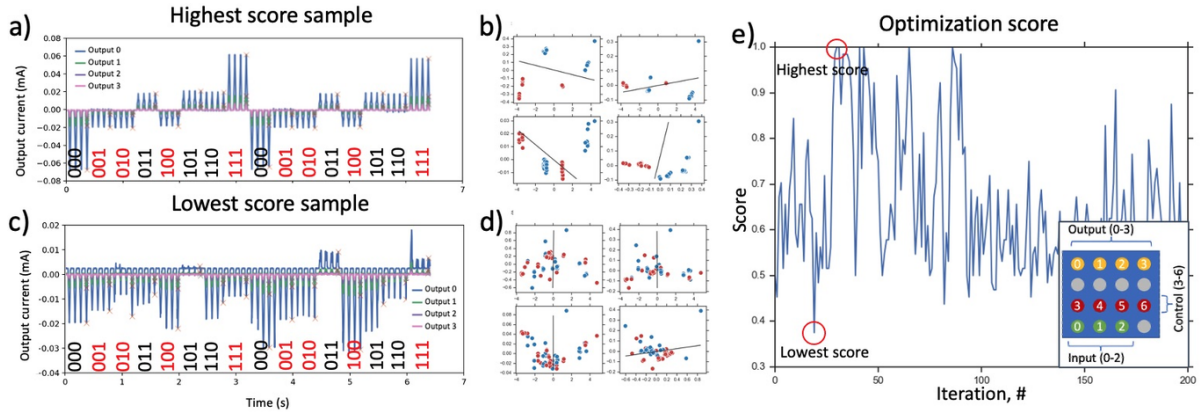


Figure 4.16: Linearization of three variable XOR problem with the NNC device. a) The output currents from 4 channels for one of the highest score samples. The color-coded sequence of double digits represents the XOR truth table where the black color corresponds to the “false” state, and the red color corresponds to the “true” state. b) PCA projections from the top 4 combinations of the output currents showing linear separability in two projections (top left and bottom right) with decision boundary found with logistic regression. c) The output currents from 4 channels for the lowest score sample and d) corresponding PCA projections. e) progression of the optimization algorithm. Due to the increased number of control voltages and the complexity of the problem, it took the algorithm nearly 45 steps to find an optimal solution. The inset shows the pinout of the connection to the NNC device

As can be noticed from Figure 4.16 (e), the evolution of the static control voltages reached the highest sore (linear separability) multiple times, with the last successful score noticed at step 84, after which the highest score could not be achieved. This could be attributed

to the fatigue of the device's complete reorganization of the internal structure, making it harder to find an optimal state.

4.21 Conclusion

This chapter presented various manufacturing and characterization techniques of neuromorphic materials across a broad range of scales, from individual resistive switching Ag_2S NWs to composites with large 3D neuromorphic networks.

Characterization of a single Ag_2S NW and PVP coated Ag NW junctions was successfully performed with the help of a nanomanipulator paired with an optical microscope under ambient conditions. With the obtained data and observations from other research regarding the scattered Ag nanocluster inclusions in Ag^+ conducting solid electrolytes, we developed a phenomenological stochastic resistive switching model for a single Ag2S NW. The model was fit to the experimentally obtained IV curves obtained from a single Ag_2S NW. After analyzing the fitted values of mobility and effective length of the wire and comparing them to the values reported in the literature, we concluded that the obtained results supported our hypothesis of a fragmented nanocluster Ag inclusions in Ag_2S NWs. In particular, the model exhibited spontaneous loop reversal and twisting that we observed in our experimental data. However, the work on improving the model is ongoing to consider Joule's heating Newton's cooling processes.

Further, we showed the first time, to our knowledge, manufacturing, and characterization of a 3D neuromorphic device with nanoscale resistive switching networks NNC. The device consisted of a network of carefully calculated and experimentally verified Ag_2S and Ag NWs at appropriate concentrations embedded into a polycaprolactone thermoplastic polymer. The micro and nano X-ray microscopy images, along with the cryo-SEM results, showed that the obtained composites host a quasi-uniform network of NWs. Electronic characterization of the material, performed with an in-house manufactured microelectrode array, showed remarkable properties that cannot be observed in conventional crossbar array devices. For example, reduction in conductivity during unipolar voltage stimulation characterized by a convex profile of the current is an active property of the device. We also referred to it as a temporary negative differential resistance. Another example of unique behavior involves a cross-talk effect, which is responsible for all outputs in the device being able to react to some degree to stimuli at every input.

We also briefly discussed some inherent properties of biological neurons, such as forget-

ting past events and a modification of synaptic weight via sliding threshold. For example, we showed that the NNC device possesses a complex forgetting mechanism that depends on the previous stimulation history. Moreover, the forgetting parameters are comparable to the data published by other groups and human memory. As another example, we showed a sliding threshold phenomenon that is also observed in biological neurons and other memristive devices. In biological neurons, the sliding threshold is a modified mechanism of synaptic plasticity. A synaptic weight is controlled not only by the activity of the presynaptic neuron but also by the activity of the postsynaptic one. In memristive devices, a similar sliding threshold phenomenon was observed immediately after the change in stimulation rate. Comparing our results to results obtained in other groups, we believe that the sliding threshold occurs due to two controlling variables that define the width and length of conducting channel. A further investigation into understanding this phenomenon is under development.

To reinforce the active reduction of current and cross-talk observed in the experimental results, during the transient analysis of the 3D NNC, we performed a simulation of an equivalent 2D circuit made of Ag/Ag_2S NWs with two inputs and two outputs. The device stimulation with a DC voltage showed the current response on every output pin and visually explained the underlying switching processes. In particular, the simulation showed that the equivalent device possessed a few conducting channels that changed the amount of current that passed through it and where the current was redistributed (active property), resulting in a reduction of the current in one channel. Thus we were able to explain the experimental phenomenon with the help of a simulation.

Finally, we showed how we used these materials to linearize complex problems such as XOR. In the last section, we presented results on the real experiment where the NNC device was installed on a microelectrode array and stimulated with voltages representing an XOR table and a set of calculated static control voltages. The obtained labeled currents, taken from 4 output electrodes at each iteration, were analyzed with a logistic regression classifier that served as a cost function. With the help of the optimization algorithm, sets of control voltages were found with which we NNC linearized 2 and 3 variable XOR tables. Further, we argued that the NNC device serves as the first layer of a two-layer perceptron model based on the results.

Despite inspiring results and the long shelf life of the material, the device can be improved in several ways. In particular, we found that the material exhibits volatility since its behavior changes over time even with the same set of static control voltages. This change is attributed to the internal modification of the network and could be potentially solved by

reducing the operating voltage and/or the ambient temperature. To address this, the work on other kinds of 3D NNC devices with transition metal oxide materials such as WO_x as switching elements instead of Ag_2S is ongoing in our laboratory. Also, we are developing experimental setups for solving image and speech recognition problems with the NNC devices.

In conclusion, we showed that neuromorphic composites with Ag_2S NW resistive switches exhibit rich reservoir-like behavior that can be used for solving nonlinear tasks. However, more research is required on understanding the properties of these materials and the kinds of tasks that can be solved with them.

CHAPTER 5

Conclusions

Summary

This dissertation presented several novel theoretical and experimental results and methods that contributed to the field of neuromorphic computing with composites near the percolation threshold, introduced in chapter 1. The core deliverable goal of the dissertation was to design, produce and characterize self-assembled neuromorphic 2D and 3D composites and, as a proof of concept, perform nonlinear computation. Thus the dissertation was split into three major chapters that covered theoretical and experimental aspects of percolation phenomena and computation with neuromorphic composites.

Thus after introducing the domain of research in the introductory chapter, a significant portion of the first core chapter was dedicated to describing the results obtained with a custom-made software suite for studying percolation and conductivity in complex multicomponent systems. In particular, I developed a series of Java and Python software packages (CircuitSymphony [Frick and LaBean, 2021a] and Percolator [Frick and LaBean, 2021b]) and an interface to them (ResSymphony [Frick, 2021]) with some examples of capabilities and comparison to the literature presented in chapter 2. I showed the relationships of the power law exponent as a function of the waviness and aspect ratio and resistivity of the nanowires that very well supported the relationships reported in the literature. Using the simulators, I also studied the predicted conductivity and percolation probability as a

function of the concentration of two-component composites in 2D and 3D systems and as a function of agglomeration of nanowires in 3D systems that my colleagues and I used as guidelines in manufacturing of the materials in the laboratory.

A particular effort was made to make this software platform-agnostic and scalable with minimum knowledge required to set it up on a personal computer or a computing cluster using modern technologies and a versatile user interface. The flexibility of the software allowed creating and analyzing networks formed by arbitrary 3D geometries on the fly and analyzing percolation via Monte-Carlo simulations, or brute-force and estimating conductivity by running SPICE circuit simulation. For example, the software enables studies of the conductivity or percolation of a system composed of multiple classes of components each having different geometry and conductivity. In addition to that, via a user interface or programmatically, one can create clusters of conducting objects and produce cavities to simulate a huge range of possible materials morphologies. My colleagues and I have successfully used this software to estimate percolation thresholds and conductivity in 2D and 3D multicomponent composites with various fillers such as Ag and Ag_2S NWs in polycaprolactone, carbon nanotubes and graphene in agarose aerogels, and silicone oil-based matrix and resistive switching composite of WO_x nanoplatelets with Ag NWs. The comparative results showed that these software packages offer a convenient way to conduct large-scale simulations of 2D and 3D networks and understand the emergence of complex phenomena in percolating resistive switching systems.

Inspired by successful results in percolation studies, my secondary goal was to support experimental observations of resistive switching phenomena in 2D and 3D networks with the *in silico* simulations of identical systems. In the first sections of chapter 3, I presented results of the exploration of memristive networks at various degrees of architectural organization by examining the current responses to triangular voltage stimuli packets of alternating polarity. Despite no significant difference in nonlinearity in the responses between ordered, disordered 2D and 3D networks, each configuration exhibited a complex nonlinear current response when stimulated with a train of triangular voltage pulses of different polarities. As it was further explored, this change was attributed to an uneven distribution of polarities of resistive switches and their sizes in the network. That resulted in a circuit having fast switching memristive devices oriented parallel to the electric field lines and slowly switching memristive devices oriented perpendicular to the electric field lines. The complex reorganization of the network architecture could be used for signal amplification and therefore can be considered an active phenomenon.

The dynamic reorganization of the memristive network under electric potential is also

a variant of analog computers, which I also discussed in chapter 3. At first, I presented the smallest circuit made with four memristors connected in a Wheatstone bridge with two inputs and two outputs that linearized the XOR problem. The ability to solve or linearize XOR is a universal benchmark for any universal computing system and is the key to solving problems of arbitrary complexity. To prove the validity of the solution, I showed the mappings of the input voltages, representing the XOR table and output currents with different timings. The voltages measured on the output terminals converged to a linearly separable form only when the duration of stimulation voltage peaks was reduced below a particular value. A random memristive network circuit observed a similar ability to solve the XOR problem in a two-input, two-output fashion. Hence, boolean logic computation on random memristive network systems is plausible, but only with optimal input voltage parameters such as amplitude and duration.

To provide a formal mathematical description of the underlying process, I provided a simple matrix formulation of an arbitrary memristive circuit derived from graph theory. I argued that a memristive circuit is a dynamic matrix operator that acts on a vector of input voltages that returns a vector of output currents (or voltages). The values of the circuit matrix constructed for a simple 4 memristor and 2 resistor circuit contained complex arithmetic relationships that depended on the circuit's internal organization and previous history of stimulation. Thus the output values of the current in such networks represent the voltage-weighted sum of matrix elements and therefore dictate the magnitude of the output. Furthermore, the complexity of mathematical relationships representing matrix elements increases dramatically with larger networks. Thus the probability of linearization of complex problems can be increased by increasing the size of the circuit and the number of outputs one is trying to analyze.

In the latter sections of chapter 3, I used the nonlinear property of the material to linearize the XOR problem with random memristive networks. The calculated static voltages supplied to additional input terminals showed that almost any circuit can be tuned to successfully solve a simple XOR and even a more complex two-layer XOR problem. Moreover, the result proved that optimizing static control voltages is a powerful trick in converting an arbitrary memristive circuit into a binary threshold logic gate.

Inspired by these results, I presented an approach for MNIST [Lecun et al., 1998] handwritten image recognition with random memristive networks. With a set of preprocessed 10000 training and 1000 testing images mapped into a random memristive network, I achieved an accuracy of 87%, comparable to the results reported by other groups and is around 10% above a two-layer perceptron network. Surprisingly, a random memristive

network performed slightly better than a linear memristor array (cross-bar array) in all the tests. Thus, a random memristive network has a slight advantage over an array of individual memristors selected from a cross-bar array in the complex classification task.

Chapter 4 presented an excerpt of experimental work that I built on top of all the previous chapters. I started with exploring the resistive switching behavior of individual memristive wires, the fundamental building blocks of the resistive switching networks. My colleagues and I performed XRD, TEM, and SEM analysis of individual wires and developed a stochastic resistive switching model of individual Ag_2S NW which parameters were fit to experimental IV data. The model very well described experimental measurements and replicated important features such as spontaneous loop twisting and reversal that can not be observed with the conventional memristor model.

One of the most exciting observations was on a network of Ag_2S NWs stimulated with a single voltage source delivered by a nanomanipulator. Due to the electromigration of Ag ions, the network underwent a series of structural changes that resulted in the change of thickness of some wires and rupture followed by the relocation of one branch of the network, yet retaining consistent resistive switching through remaining channels. This phenomenon is exciting as it resembles synaptic detachment observed in biological brains [Chen, 1978, Blinzinger and Kreutzberg, 1968].

Under ambient conditions, the shelf life of Ag/Ag_2S NWs switching is severely impacted by oxidation, resulting in the loss of resistive switching property after a few days. Moreover, as we showed in chapter 3, more complex relationships can be formed in a larger circuit matrix, especially from networks with higher degrees of freedom, due to the synergistic interplay of multiple components. Thus further in chapter 4, I presented my efforts regarding the development of novel 3D resistive switching composites based on appropriate concentrations of Ag/Ag_2S NWs in thermoplastic PCL matrix. The concentrations, usually estimated with the Percolator software, depend on the average geometry of the wires. The resulting solid Neuromorphic Nanoscale Composites (NNC) were analyzed with micro and nano X-Ray microscopes which showed a distributed network of nanowires within the volume of the material. A two probe electronic assessment of the composite exhibited resistive switching, which “off” and “on” state resistances depended strongly on the concentration of the components and relative position of the electronic probes attesting connectedness of the material components. Sparse electric connectedness in the material makes it an ideal candidate for multielectrode array measurements, similar to those performed on biological brains.

Microelectrode arrays are one of the most popular invasive methods of measuring the

local electric activity of neurons in biological brains *in vivo* [Musk, 2019, Nurmikko et al., 2010b]. Typical microelectrode arrays consist of several thin metallic electrodes, each of which can have an additional set of contacts [Musk, 2019]. However, most commercially available microelectrode arrays have a high price and require costly interfaces and software. Thus I designed a simple microelectrode array with stainless steel acupuncture needles installable on major kinds of zero insertion force (ZIF) sockets. My colleagues and I used these microelectrode arrays for multichannel characterization of the NNC devices. In particular, I used the microelectrode array to estimate the relaxation rate of the high conductivity state, the sliding threshold-like phenomenon, and XOR gates, also presented in chapter 4.

Some of the most remarkable neuromorphic properties of the NNC that resemble behavior in biological synapses is the relaxation of high conductivity state and sliding threshold. The process of forgetting plays an important evolutionary role that allows biological organisms to save energy by recalling the essential information while adapting to rapidly changing environments. The natural mechanisms of forgetting are caused by loss of synaptic connection strength due to lack of stimulation. Similarly, the highly conductive state of a volatile Ag_2S memristive device gets lost over some time, depending on the history of previous stimulation. In particular, I estimated the relaxation time of the NCC device by fitting a stretched-exponential function to the data. Despite the nanowire-based random architecture of the NNC device with large distances between electrodes, the forgetting rate was comparable to the rates reported in the literature for a thin film WO_3 device and also, coincidentally, for the forgetting rate observed in human memory studies.

I also observed the sliding threshold phenomenon in the NNC device and compared it to previously reported observation for the WO_3 device [Du et al., 2015]. The sliding threshold is a modification of the Hebbian learning rule, where synaptic strength is regulated not only by the activity of a pre-synaptic neuron but also by the activity of the post-synaptic one and is important in long-term developmental processes [Bienenstock et al., 1982].

Linearization of the XOR problems was performed similarly to chapter 3, where input variables were converted into a train of voltage spikes transmitted via input pins to the NNC and calculated static control voltages. The projections of the voltage drop measured on the output resistors at the end of each iteration showed that thanks to the static control voltages, the NNC was configured to a binary threshold gate capable of representing a single and double layer XOR gate.

Future research directions

The overall motivation of this work was to establish an approach towards the development of Autonomous Artificial Intelligence Machines (AAIM). In the framework of this work, AAIM is an engineered spiking neural network system that closely mimics the functionality and connectivity of neurons of the biological brain that potentially can be capable of learning as efficiently as humans. In computer science literature such level of inteligence is also abstractly referred to as Artificial General Intelligence (AGI) [Yampolskiy and Fox, 2012]. However, there is much criticism on whether such systems are possible on conventional CMOS computing architectures. Therefore, more researcher interest continues to grow in non-conventional neuromorphic implementations that mimic biological brains [rep,]. Biological brains have many properties that can not be replicated or even efficiently simulated on conventional computers. First, unlike conventional binary computer hardware, computation with biological neurons is performed via electrochemical oscillations and is non-Boolean. Second, neurons are interconnected via complex self-adapting networks of synapses that constantly evolve, eliminating old and creating new connections [Chung et al., 2015]. Finally, even at a relatively small frequency of operation, around 500 Hz [Wang et al., 2016], biological neurons have a massive associative memory capacity that empowers abstract thinking and the ability to solve previously unseen problems with a much smaller set of training examples than modern neural networks [Oleinik, 2019]. Perhaps these are among the main factors that granted biological computers immense adaptability and massive evolutionary success. Therefore, an AAIM has to be built similarly to its biological analog.

Coupled oscillators and integration with existing synaptic networks

We already showed that memristive NNC devices exhibit synaptic properties and can be used for computation, but to compute it requires complex electronics and therefore is not sufficient to build AAIM. Thus another building block is required, a neuristor (aka artificial neuron), a generator of oscillating signals which, if paired with synaptors (aka artificial synapses) that store the weights and provide coupling for phase or frequency synchronization of oscillations generated by different neuristors could be a potential pathway to creating AAIM [Fries, 2005, Fell and Axmacher, 2011, Schnitzler and Gross, 2005]. Depending on the amplitude of the received signal, the neuristor can generate a train of oscillating electrical signals that other neuristors will receive via synaptors.

Similar to the sensory and cognitive processes in the biological neural tissue being processed by a network of spiking oscillators (neurons) connected by a network of synapses, at the core of engineered AAIM lays a system of artificial signal generating oscillators (neuristors) also interconnected by a network of weights (synaptors) that facilitate the phase and frequency synchronization between the oscillators [Strogatz, 2018]. Such an interconnected system of nonlinear oscillators would evolve chaotically around some attractor points as a response to an interplay between external stimuli, current internal state, the coupling strength between oscillators, and the previous history of stimulation. The network of synapses that sparsely connects these oscillators will also evolve due to internal states. Similar to the biological brain, some weight values will strengthen while others become weaker. The autonomy of such a system is a property necessary for evolutionary processes and of goal-setting in reinforcement learning environments [Sutton and Barto, 2018].

This dissertation only explored the synaptic (Ag/Ag_2S NWs) component of the proposed AAIM polymer composite and showed that these dynamic networks are tunable to perform complex computation tasks. In particular, spiking signals were used in the presented computation experiments to mimic oscillations of generated signals, and static optimal control voltages represented tuning signals from other oscillators received through synaptic weights. Thus, further development of the project is aimed to replace static control voltages with self-adapting signals received as outputs from coupled nonlinear oscillators.

Typically in electronics, relaxation oscillators circuits are built with capacitance which is hard to implement at the nanoscale. However, recent implementations of VO_2 and NbO_2 neuristor devices exhibited robust relaxation oscillations closely mimicking depolarization of biological neuron membrane discussed in chapters 1 and 4 [Pickett et al., 2012, Yi et al., 2018]. Furthermore, compared to NbO_2, VO_2 has a low temperature of metal-insulator transition of around 345K, which permits the use of thermal capacitance instead of electrical capacitance to sustain relaxation oscillations [Bohaichuk et al., 2019, del Valle et al., 2020]. While classical oscillating circuits rely on electrical capacitance to shift the phase of the current ahead of the voltage, inherent thermal capacitance of the VO_2, or other Mott memristor device, can be used to store the energy in form of the heat and reduce the barrier of the metal-insulator transformation. In this fashion, multiple individual oscillators can be thermally coupled in a miniature nanoscale circuit without the need for an external capacitor [Velichko et al., 2018, Velichko and Belyaev, 2019]. Thermal coupling is the key to this project's next step, creating an autonomous network of evolvable spiking oscillators.

As a proof of concept, I developed an oscillatory device, as shown in Figure 5.1 (a), with

VO_2 NPs, that doesn't require thin layer deposition as was used in a device by Pergament *et al.* [Pergament et al., 2018]. Instead of sputtering a thin layer of VO_2 onto a sapphire substrate, the device was made with a 500 nm thick polypropylene film with a 200 μm hole filled with VO_2 NPs (99.9%, 100-200 nm, NanoArmor) and sandwiched between two glass slides covered with thin strips of copper tape. The device showed insulator-metal transition when driven by a steady triangle voltage source Figure 5.1 (c-d). It also exhibited a negative differential resistance pattern Figure 5.1 (e-f) if driven with a triangular current pattern (with parallel 100 kΩ resistor). The stable oscillations of the system shown in an oscilloscope screenshot Figure 5.1 (b) were ignited with a constant voltage range between 5-10 V. Interestingly, the frequency of oscillations did depend on the pressure applied to the glass slides, as it changed the magnitude of the capacitance created by the copper tapes and the effective surface area of contact between the VO_2 NPs and copper tape.

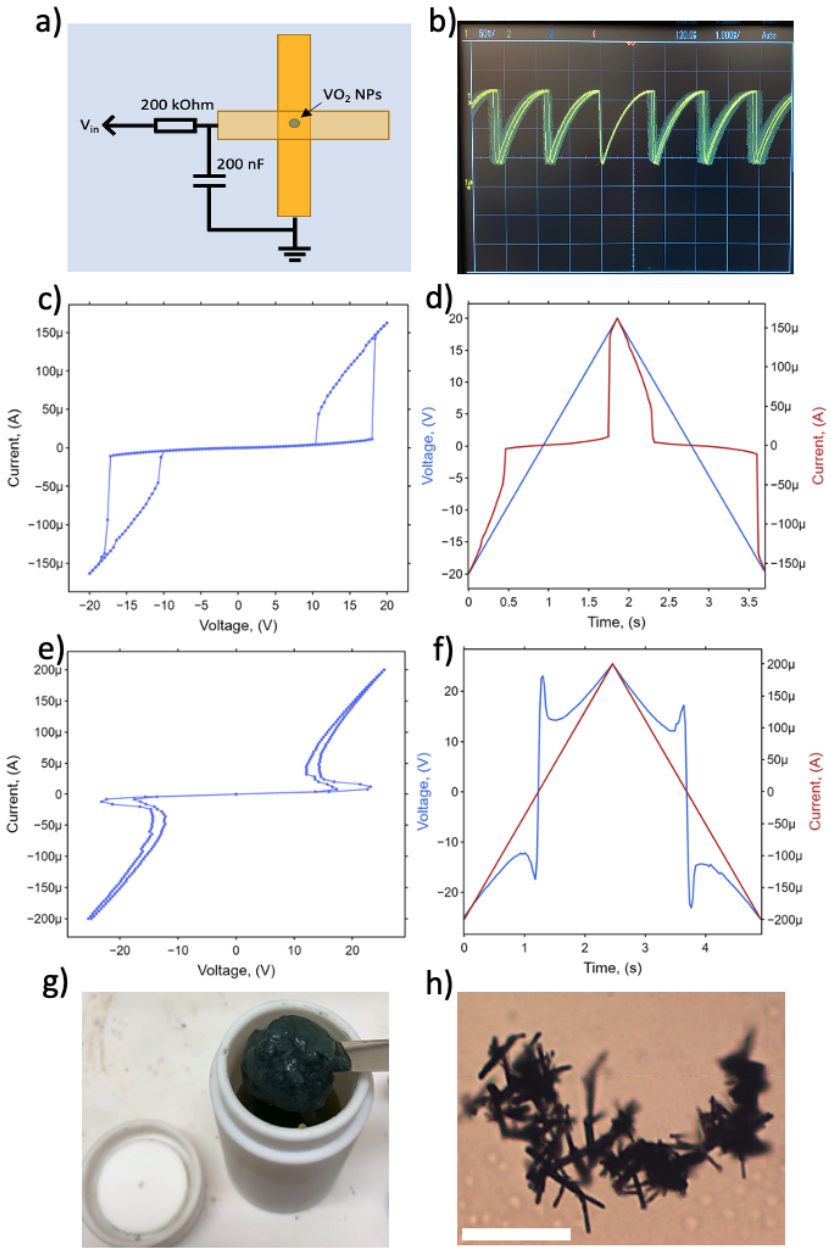


Figure 5.1: IV characteristics and relaxation oscillations of a simple VO_2 NP device. a) Schematics of the implemented oscillating device. b) Voltage-driven relaxation oscillations of 430 Hz with the VO_2 NP device. Different patterns of relaxation oscillations can be achieved in such devices driven with voltage and current-driven sources see Pergament *et al.* [Pergament et al., 2018]. c) voltage-driven IV characteristics showing an abrupt insulator-metal transition at around ±17 V and metal-insulator transition at around ±10 V. d) shows time voltage and current as a function of time e) current-driven IV characteristics of the device exhibiting soft transition, compared to voltage-driven one. As the current going through the system drops below 50 µA, the voltage rapidly, but continuously increases due to metal-insulator transition. f) time voltage and current as a function of time for current-driven stimulation. g) hydrothermally grown VO_2 NWs with an average 15 µm length and 500 nm diameter produced in our laboratory based on the recipe by Horrocks *et al.* [Horrocks et al., 2014] h) an optical micrograph of the VO_2 NWs with 50 µm scale bar.

While nanoparticles are convenient for the creation of a single device due to their small aspect ratio, currently work is in progress on producing 2D and 3D devices with thermally coupled high aspect ratio VO_2 NWs [Horrocks et al., 2014] and shown in Figure 5.1 (g). The VO_2 NWs also exhibited identical IV characteristics as VO_2 NPs.

Composites with transition metal oxide-based switches

The stochastic behavior of Ag/Ag_2S NW based composite showed remarkable switching behavior but a short stability lifetime. The estimated static control voltages worked only for up to 200 steps in the XOR problem. Primarily, instability is caused by significant structural changes in the wires induced by the electromigration of Ag^+ ions. While stability could be improved with a more careful adjustment of the magnitudes of static control voltages and their duration, more stable materials that do not undergo phase transformation could be used instead of Ag/Ag_2S. One of such materials being studied by colleagues and myself is an allotrope of WO_3. In preliminary studies, WO_3 nanosheets sandwiched between Ag electrodes showed superior switching behavior and state retention than Ag/Ag_2S system. Although the switching mechanism (cationic or anionic) remains unknown, the device exhibited smaller mobility than Ag_2S and required larger magnitudes of switching voltages.

Development of 3D dendritic structures

The nanowires loosely mimic the shape and connectivity of biological neurons. To make structures that better represent biological neurons, my colleagues and I performed an explorative study where, instead of high aspect ratio NWs, 3D dendritic structures have created that look nearly identical to biological neurons in an optical microscope 5.2.

I grew dendritic structures in agarose hydrogel with Cu NPs serving as seeds with diffusion-limited aggregation process (DLA) [Witten and Sander, 1981]. DLA is based on the propagation of waves of ion concentrations that produce sites of preferential attachment [Langer, 1980, Oikawa and Kurita, 2016]. The 3% agarose gel diluted in DI water was loaded with 100 nm diameter Cu NPs, heated to 70 and then cooled to gel, then placed in a concentrated $AgNO_3$ solution. However, four days later, I observed increased size of Cu NPs and growth of dendritic branching from them having a diameter between up to 50 μm shown in Figure 5.2 (b-d) that under a microscope looked similar to neurons, and glial cells,

shown in Figure 5.2 (e-f) reproduced from Saveliev *et al.* [Saveliev, 2005]. Interestingly, the objects did not exhibit dense connection, which could result from obstructions caused by the crosslinked polymer of the agarose hydrogel that prevented the propagation of the concentration waves from building connected dendritic structures, as can be observed in 2D [Demis et al., 2015]. Thus in the next iteration of the study, we will investigate the creation of densely connected dendritic 3D structures by carefully controlling the temperature of the solidification of the agarose gel to prevent the Cu NPs and dendritic structures from precipitating and thereby enhancing the formation of long-range connectivity by eliminating obstructions to the waves of ion concentration.

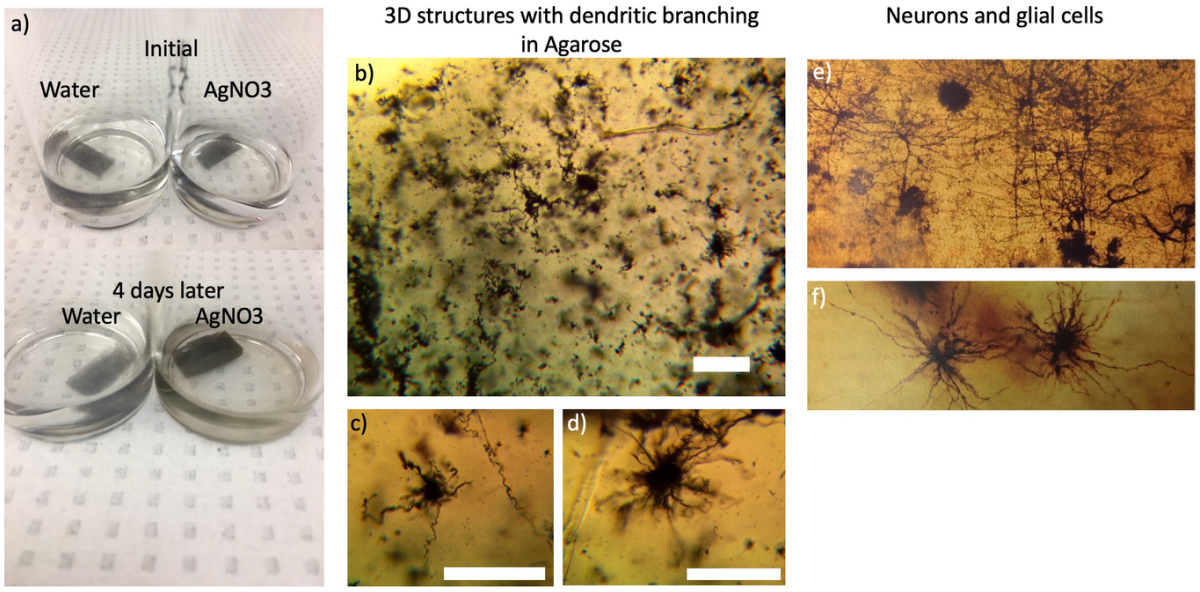


Figure 5.2: 3D Ag structures with dendritic branching grown in Agarose gel. a) the process of growth of dendritic structures in agarose gel with embedded Cu NPs seeds. In the beginning, gels look similar with characteristic dark inclusions of Cu NPs (top). Four days later (bottom), gels suspended in $AgNO_3$ solution appeared much darker than those placed in DI water. b) and d) show snapshots of obtained structures in Agarose evaluated under an optical microscope with 100 μm long scale bars. e) neurons observed in an optical micrograph f) optical micrograph of glial cells. Both e) and f) reproduced from Saveliev *et al.* [Saveliev, 2005].

Improving computer interface

One of the largest obstacles in this work was the lack of an appropriate interface between a PC and the material. Working with 3D composites and microelectrode arrays required a large number of input and output channels that are not typically available in conventional scientific electronic equipment. Therefore, many groups that work in the field of nonconventional neuromorphic computing design and manufacture their own electronic interfaces to the materials with various degrees of sophistication [Zidan et al., 2018, Mohid et al., 2014, Miller et al., 2014].

Thus, I designed and developed a universal high-speed switching matrix. The switching matrix comprises three-way 16x16=256 pins, each having an individual switch for a row, column, and node pins. Unlike conventional cross-bar arrays, such configuration can cater to collision-free connectivity between multiple nodes simultaneously (while keeping unneeded nodes isolated) and is useful for interacting with the NNC device via multielectrode array and a small number of high precision source meters.

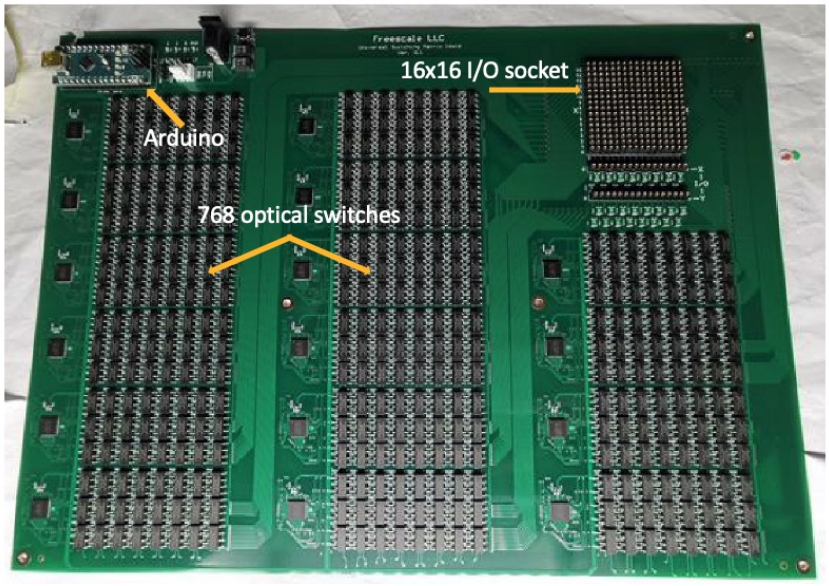


Figure 5.3: Universal 16x16x3 switching matrix.

The switching matrix allows addressing of any electrode pairs in order to probe the connecting path between specified electrode contacts. It will enable my colleagues and me

to train the NNC, or its variants to perform classification or regression on more advanced problems. For example, we plan to tackle such problems as image and voice recognition, reinforcement learning and investigation of the plausibility of the proposed AAIM systems with coupled nonlinear oscillators.

REFERENCES

[rep,] THE INTERNATIONAL TECHNOLOGY ROADMAP FOR SEMICONDUCTORS 2.0: 2015. https://www.semiconductors.org/resources/2015-international-technology-roadmap-for-semiconductors-itrs/.

[tza,] tzaeschke/ode4j. https://github.com/tzaeschke/ode4j. Accessed on Wed, December 30, 2020.

[Agarwal et al., 2015] Agarwal, S., Quach, T., Parekh, O., Hsia, A., DeBenedictis, E., James, C., Marinella, M., and Aimone, J. (2015). Energy Scaling Advantages of Resistive Memory Crossbar Based Computation and Its Application to Sparse Coding. *Front Neurosci*, 9:484.

[Akiba et al., 2019] Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A Next-generation Hyperparameter Optimization Framework.

[Alexander, 2009] Alexander, J. W. (2009). History of the Medical Use of Silver. *Surgical Infections*, 10(3):289–292.

[Alibart et al., 2012] Alibart, F., Gao, L., Hoskins, B. D., and Strukov, D. B. (2012). High precision tuning of state for memristive devices by adaptable variation-tolerant algorithm. *Nanotechnology*, 23(7):075201.

[Alibart et al., 2013] Alibart, F., Zamanidoost, E., and Strukov, D. B. (2013). Pattern classification by memristive crossbar circuits using ex situ and in situ training. *Nature Communications*, 4(1).

[Allen and Moore, 1959] Allen, R. L. and Moore, W. J. (1959). Diffusion of Silver in Silver Sulfide. *The Journal of Physical Chemistry*, 63(2):223–226.

[Andrés et al., 2015] Andrés, L. J., Menéndez, M. F., Gómez, D., Martínez, A. L., Bristow, N., Kettle, J. P., Menéndez, A., and Ruiz, B. (2015). Rapid synthesis of ultra-long silver nanowires for tailor-made transparent conductive electrodes: proof of concept in organic solar cells. *Nanotechnology*, 26(26):265201.

[Ascoli et al., 2015] Ascoli, A., Slesazeck, S., Mahne, H., Tetzlaff, R., and Mikolajick, T. (2015). Nonlinear Dynamics of a Locally-Active Memristor. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 62(4):1165–1174.

[Atkinson and Goldthorpe, 2020] Atkinson, J. and Goldthorpe, I. A. (2020). Near-infrared properties of silver nanowire networks. *Nanotechnology*, 31(36):365201.

[Auger and Marty, 2000] Auger, C. and Marty, A. (2000). Quantal currents at single-site central synapses. *The Journal of Physiology*, 526(1):3–11.

[Avizienis et al., 2012] Avizienis, A. V., Sillin, H. O., Martin-Olmos, C., Shieh, H. H., Aono, M., Stieg, A. Z., and Gimzewski, J. K. (2012). Neuromorphic Atomic Switch Networks. *PLoS ONE*, 7(8):e42772.

[Backus, 1978] Backus, J. (1978). Can programming be liberated from the von Neumann style? *Communications of the ACM*, 21(8):613–641.

[Bahadori et al., 2020] Bahadori, S. R., Mei, L., Athavale, A., jen Chiu, Y., Pickering, C. S., and Hao, Y. (2020). New Insight into Single-Crystal Silver Dendrite Formation and Growth Mechanisms. *Crystal Growth & Design*, 20(11):7291–7299.

[Balberg, 2020] Balberg, I. (2020). Principles of the Theory of Continuum Percolation. In *Encyclopedia of Complexity and Systems Science*, pages 1–61. Springer Berlin Heidelberg.

[Balberg and Binenbaum, 1987] Balberg, I. and Binenbaum, N. (1987). Invariant properties of the percolation thresholds in the soft-core–hard-core transition. *Physical Review A*, 35(12):5174–5177.

[Balberg et al., 1984] Balberg, I., Binenbaum, N., and Wagner, N. (1984). Percolation Thresholds in the Three-Dimensional Sticks System. *Physical Review Letters*, 52(17):1465–1468.

[Banerjee et al., 2007] Banerjee, S., Bhattacharya, S., and Chakravorty, D. (2007). Resistivity Hysteresis of Ag2S Nanocomposites. *The Journal of Physical Chemistry C*, 111(36):13410–13413.

[Bao et al., 2013] Bao, W. S., Meguid, S. A., Zhu, Z. H., Pan, Y., and Weng, G. J. (2013). Effect of carbon nanotube geometry upon tunneling assisted electrical network in nanocomposites. *Journal of Applied Physics*, 113(23):234313.

[Bao et al., 2012] Bao, W. S., Meguid, S. A., Zhu, Z. H., and Weng, G. J. (2012). Tunneling resistance and its effect on the electrical conductivity of carbon nanotube nanocomposites. *Journal of Applied Physics*, 111(9):093726.

[Barabasi, 2016] Barabasi, A. (2016). *Network Science*. Cambridge University Press.

[Barlow and Levick, 1965] Barlow, H. B. and Levick, W. R. (1965). The mechanism of directionally selective units in rabbit's retina. *The Journal of Physiology*, 178(3):477–504.

[Bassett et al., 2006] Bassett, D. S., Meyer-Lindenberg, A., Achard, S., Duke, T., and Bullmore, E. (2006). Adaptive reconfiguration of fractal small-world human brain functional networks. *Proceedings of the National Academy of Sciences*, 103(51):19518–19523.

[Belov et al., 2014] Belov, A. N., Pyatilova, O. V., and Vorobiev, M. I. (2014). Synthesis of Ag/Ag&ltsub&gt2&lt/sub&gtS Nanoclusters Resistive Switches for Memory Cells. *Advances in Nanoparticles*, 03(01):1–4.

[Bergstra et al., 2015] Bergstra, J., Komer, B., Eliasmith, C., Yamins, D., and Cox, D. D. (2015). Hyperopt: a Python library for model selection and hyperparameter optimization. *Computational Science & Discovery*, 8(1):014008.

[Berhan and Sastry, 2007] Berhan, L. and Sastry, A. M. (2007). Modeling percolation in high-aspect-ratio fiber systems. I. Soft-core versus hard-core models. *Physical Review E*, 75(4).

[Bienenstock et al., 1982] Bienenstock, E., Cooper, L., and Munro, P. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *The Journal of Neuroscience*, 2(1):32–48.

[Biju et al., 2011] Biju, K. P., Liu, X., Siddik, M., Kim, S., Shin, J., Kim, I., Ignatiev, A., and Hwang, H. (2011). Resistive switching characteristics and mechanism of thermally grown WOx thin films. *Journal of Applied Physics*, 110(6):064505.

[Blakey, 2010] Blakey, E. (2010). Unconventional complexity measures for unconventional computers. *Natural Computing*, 10(4):1245–1259.

[Blakey, 2016] Blakey, E. (2016). Unconventional Computers and Unconventional Complexity Measures. In *Emergence Complexity and Computation*, pages 165–182. Springer International Publishing.

[Blinzinger and Kreutzberg, 1968] Blinzinger, K. and Kreutzberg, G. (1968). Displacement of synaptic terminals from regenerating motoneurons by microglial cells. *Zeitschrift fr Zellforschung und Mikroskopische Anatomie*, 85(2):145–157.

[Bohaichuk et al., 2019] Bohaichuk, S. M., Kumar, S., Pitner, G., McClellan, C. J., Jeong, J., Samant, M. G., Wong, H.-S. P., Parkin, S. S. P., Williams, R. S., and Pop, E. (2019). Fast Spiking of a Mott VO2–Carbon Nanotube Composite Device. *Nano Letters*, 19(10):6751–6755.

[Boland et al., 2016] Boland, C. S., Khan, U., Ryan, G., Barwich, S., Charifou, R., Harvey, A., Backes, C., Li, Z., Ferreira, M. S., Mobius, M. E., Young, R. J., and Coleman, J. N. (2016). Sensitive electromechanical sensors using viscoelastic graphene-polymer nanocomposites. *Science*, 354(6317):1257–1260.

[Bordes et al., 2010] Bordes, C., Fréville, V., Ruffin, E., Marote, P., Gauvrit, J., Briançon, S., and Lantéri, P. (2010). Determination of poly(ε-caprolactone) solubility parameters: Application to solvent substitution in a microencapsulation process. *International Journal of Pharmaceutics*, 383(1-2):236–243.

[Borresen and Lynch, 2012] Borresen, J. and Lynch, S. (2012). Oscillatory Threshold Logic. *PLoS ONE*, 7(11):e48498.

[Bose et al., 2015] Bose, S. K., Lawrence, C. P., Liu, Z., Makarenko, K. S., van Damme, R. M. J., Broersma, H. J., and van der Wiel, W. G. (2015). Evolution of a designless nanoparticle network into reconfigurable Boolean logic. *Nature Nanotechnology*, 10(12):1048–1052.

[Budhiraja et al., 2021] Budhiraja, R., Kumar, M., Das, M. K., Bafila, A. S., and Singh, S. (2021). A reservoir computing approach for forecasting and regenerating both dynamical and time-delay controlled financial system behavior. *PLOS ONE*, 16(2):1–24.

[Bug et al., 1985] Bug, A. L. R., Safran, S. A., Grest, G. S., and Webman, I. (1985). Do Interactions Raise or Lower a Percolation Threshold? *Physical Review Letters*, 55(18):1896–1899.

[Bug et al., 1986] Bug, A. L. R., Safran, S. A., and Webman, I. (1986). Continuum percolation of permeable objects. *Physical Review B*, 33(7):4716–4724.

[Buitinck et al., 2013] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). Api design for machine learning software: experiences from the scikit-learn project.

[Bunde and Dieterich, 2000] Bunde, A. and Dieterich, W. (2000). Percolation in Composites. *Journal of Electroceramics*, 5(2):81–92.

[Carmona et al., 1982] Carmona, F., Barreau, F., and Delhaes, P. (1982). 225. Anisotropic percolation in carbon fiber polymer composites. *Carbon*, 20(2):149.

[Carmona et al., 1980] Carmona, F., Barreau, F., Delhaes, P., and Canet, R. (1980). An experimental model for studying the effect of anisotropy on percolative conduction. *Journal de Physique Lettres*, 41(22):531–533.

[Chang, 2012] Chang, T. (2012). *Tungsten Oxide Memristive Devices for Neuromorphic Applications*. PhD thesis, University of Michigan.

[Chang et al., 2011] Chang, T., Jo, S.-H., and Lu, W. (2011). Short-Term Memory to Long-Term Memory Transition in a Nanoscale Memristor. *ACS Nano*, 5(9):7669–7676.

[Chen et al., 2016] Chen, C., Wang, H., Xue, Y., Xue, Z., Liu, H., Xie, X., and Mai, Y.-W. (2016). Structure rheological, thermal conductive and electrical insulating properties of high-performance hybrid epoxy/nanosilica/AgNWs nanocomposites. *Composites Science and Technology*, 128:207–214.

[Chen, 1978] Chen, D. H. (1978). Qualitative and quantitative study of synaptic displacement in chromatolyzed spinal motoneurons of the cat. *The Journal of Comparative Neurology*, 177(4):635–663.

[Chen et al., 2013a] Chen, J.-Y., Hsin, C.-L., Huang, C.-W., Chiu, C.-H., Huang, Y.-T., Lin, S.-J., Wu, W.-W., and Chen, L.-J. (2013a). Dynamic Evolution of Conducting Nanofilament in Resistive Switching Memories. *Nano Letters*, 13(8):3671–3677.

[Chen et al., 2020] Chen, T., van Gelder, J., van de Ven, B., Amitonov, S. V., de Wilde, B., Euler, H.-C. R., Broersma, H., Bobbert, P. A., Zwanenburg, F. A., and van der Wiel, W. G. (2020). Classification with a disordered dopant-atom network in silicon. *Nature*, 577(7790):341–345.

[Chen et al., 2013b] Chen, X., Wang, Y., and Yang, H. (2013b). NICSLU: An Adaptive Sparse Matrix Solver for Parallel Circuit Simulation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(2):261–274.

[Choi et al., 2005] Choi, B. J., Jeong, D. S., Kim, S. K., Rohde, C., Choi, S., Oh, J. H., Kim, H. J., Hwang, C. S., Szot, K., Waser, R., Reichenberg, B., and Tiedke, S. (2005). Resistive switching mechanism of TiO2 thin films grown by atomic-layer deposition. *Journal of Applied Physics*, 98(3):033715.

[Choi et al., 2015] Choi, S., Sheridan, P., and Lu, W. D. (2015). Data Clustering using Memristor Networks. *Scientific Reports*, 5(1).

[Choudhary et al., 2016] Choudhary, N., Li, C., Chung, H.-S., Moore, J., Thomas, J., and Jung, Y. (2016). High-Performance One-Body Core/Shell Nanowire Supercapacitor Enabled by Conformal Growth of Capacitive 2D WS2 Layers. *ACS Nano*, 10(12):10726–10735.

[Chua, 1971] Chua, L. (1971). Memristor-The missing circuit element. *IEEE Transactions on Circuit Theory*, 18(5):507–519.

[Chua, 2014] Chua, L. (2014). If it's pinched it's a memristor. *Semiconductor Science and Technology*, 29(10):104001.

[CHUA, 2005] CHUA, L. O. (2005). LOCAL ACTIVITY IS THE ORIGIN OF COMPLEXITY. *International Journal of Bifurcation and Chaos*, 15(11):3435–3456.

[Chung et al., 2015] Chung, W.-S., Allen, N. J., and Eroglu, C. (2015). Astrocytes Control Synapse Formation Function, and Elimination. *Cold Spring Harbor Perspectives in Biology*, 7(9):a020370.

[Clingerman et al., 2001] Clingerman, M. L., King, J. A., Schulz, K. H., and Meyers, J. D. (2001). Evaluation of electrical conductivity models for conductive polymer composites. *Journal of Applied Polymer Science*, 83(6):1341–1356.

[Cover, 1965] Cover, T. M. (1965). Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. *IEEE Transactions on Electronic Computers*, EC-14(3):326–334.

[Crane, 1961] Crane, H. (1961). The neuristor. In *1961 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*. Institute of Electrical and Electronics Engineers.

[Davis and Natarajan, 2010] Davis, T. A. and Natarajan, E. P. (2010). Algorithm 907. *ACM Transactions on Mathematical Software*, 37(3):1–17.

[del Valle et al., 2020] del Valle, J., Salev, P., Kalcheim, Y., and Schuller, I. K. (2020). A caloritronics-based Mott neuristor. *Scientific Reports*, 10(1).

[Demis et al., 2015] Demis, E. C., Aguilera, R., Sillin, H. O., Scharnhorst, K., Sandouk, E. J., Aono, M., Stieg, A. Z., and Gimzewski, J. K. (2015). Atomic switch networks—nanoarchitectonic design of a complex system for natural computing. *Nanotechnology*, 26(20):204003.

[Du et al., 2017a] Du, C., Cai, F., Zidan, M., Ma, W., Lee, S., and Lu, W. (2017a). Reservoir computing using dynamic memristors for temporal information processing. *Nat Commun*, 8:2204.

[Du et al., 2017b] Du, C., Cai, F., Zidan, M. A., Ma, W., Lee, S. H., and Lu, W. D. (2017b). Reservoir computing using dynamic memristors for temporal information processing. *Nature Communications*, 8(1).

[Du et al., 2015] Du, C., Ma, W., Chang, T., Sheridan, P., and Lu, W. D. (2015). Biorealistic Implementation of Synaptic Functions with Oxide Memristors through Internal Ionic Dynamics. *Advanced Functional Materials*, 25(27):4290–4299.

[Dudek and Bear, 1992] Dudek, S. M. and Bear, M. F. (1992). Homosynaptic long-term depression in area CA1 of hippocampus and effects of N-methyl-D-aspartate receptor blockade. *Proceedings of the National Academy of Sciences*, 89(10):4363–4367.

[Eckart and Young, 1936] Eckart, C. and Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

[Efstathiou and Efstathiou, 2018] Efstathiou, K. and Efstathiou, M. (2018). Celestial Gearbox. *Mechanical Engineering*, 140(09):31–35.

[Elton, 2018] Elton, D. C. (2018). Stretched Exponential Relaxation.

[Fan et al., 2011] Fan, D., Rich, D., Holtzman, T., Ruther, P., Dalley, J., Lopez, A., Rossi, M., Barter, J., Salas-Meza, D., Herwik, S., Holzhammer, T., Morizio, J., and Yin, H. (2011). A wireless multi-channel recording system for freely behaving mice and rats. *PLoS One*, 6:e22033.

[Feinberg, 1982] Feinberg, I. (1982). Schizophrenia: Caused by a fault in programmed synaptic elimination during adolescence? *Journal of Psychiatric Research*, 17(4):319–334.

[Fell and Axmacher, 2011] Fell, J. and Axmacher, N. (2011). The role of phase synchronization in memory processes. *Nature Reviews Neuroscience*, 12(2):105–118.

[Fletcher, 1980] Fletcher, W. (1980). *An engineering approach to digital design*. Englewood Cliffs, N.J. : Prentice-Hall.

[Forró et al., 2018] Forró, C., Demkó, L., Weydert, S., Vörös, J., and Tybrandt, K. (2018). Predictive Model for the Electrical Transport within Nanowire Networks. *ACS Nano*, 12(11):11080–11087.

[Freeman, 1977] Freeman, L. C. (1977). A Set of Measures of Centrality Based on Betweenness. *Sociometry*, 40(1):35.

[Frick, 2021] Frick, N. (2021). nfrik/ResSymphony: rel-py-v5.1e.

[Frick and LaBean, 2021a] Frick, N. and LaBean, T. (2021a). nfrik/CircuitSymphony: rel-osx-v9.0.1.

[Frick and LaBean, 2021b] Frick, N. and LaBean, T. (2021b). nfrik/Percolator: rel-percolator-ub-vnc-v5.1.1.

[Fries, 2005] Fries, P. (2005). A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. *Trends in Cognitive Sciences*, 9(10):474–480.

[Furui,] Furui, S. Speaker-independent isolated word recognition based on emphasized spectral dynamics. In *ICASSP '86. IEEE International Conference on Acoustics Speech, and Signal Processing*. Institute of Electrical and Electronics Engineers.

[Gao, 2021] Gao, M. (2021). *Self-Assembled Three-Dimensional Nanoelectronics Systems with Neuromorphic Network Architectures*. PhD thesis, NC State University.

[Ge et al., 2017] Ge, Y., Duan, X., Zhang, M., Mei, L., Hu, J., Hu, W., and Duan, X. (2017). Direct Room Temperature Welding and Chemical Protection of Silver Nanowire Thin Films for High Performance Transparent Conductors. *Journal of the American Chemical Society*, 140(1):193–199.

[Gennes, 1976] Gennes, P. D. (1976). On a relation between percolation theory and the elasticity of gels. *Journal de Physique Lettres*, 37(1):1–2.

[Goldfarb et al., 2012] Goldfarb, I., Miao, F., Yang, J. J., Yi, W., Strachan, J. P., Zhang, M.-X., Pickett, M. D., Medeiros-Ribeiro, G., and Williams, R. S. (2012). Electronic structure and transport measurements of amorphous transition-metal oxides: observation of Fermi glass behavior. *Applied Physics A*, 107(1):1–11.

[Grossberg et al., 2001] Grossberg, S., Maass, W., and Markram, H. (2001). Introduction. *Neural Networks*, 14(6-7):587.

[Gubicza et al., 2016] Gubicza, A., Manrique, D. Z., Pósa, L., Lambert, C. J., Mihály, G., Csontos, M., and Halbritter, A. (2016). Asymmetry-induced resistive switching in Ag-Ag2S-Ag memristors enabling a simplified atomic-scale memory design. *Scientific Reports*, 6(1).

[Harris et al., 2012] Harris, J. J., Jolivet, R., and Attwell, D. (2012). Synaptic Energy Use and Supply. *Neuron*, 75(5):762–777.

[Hasegawa et al., 2001] Hasegawa, T., Terabe, K., Nakayama, T., and Aono, M. (2001). Quantum Point Contact Switch using Solid Electrochemical Reaction. In *Extended Abstracts of the 2001 International Conference on Solid State Devices and Materials*. The Japan Society of Applied Physics.

[Haykin, 2008] Haykin, S. (2008). *Neural Networks and Learning Machines*. Pearson Prentice Hall New Jersey.

[Haykin, 2009] Haykin, S. S. (2009). *Neural networks and learning machines*. Pearson Education, Upper Saddle River, NJ, third edition.

[Hebb, 1952] Hebb, M. H. (1952). Electrical Conductivity of Silver Sulfide. *The Journal of Chemical Physics*, 20(1):185–190.

[Hicks et al., 2018] Hicks, J., Li, J., Ying, C., and Ural, A. (2018). Effect of nanowire curviness on the percolation resistivity of transparent conductive metal nanowire networks. *Journal of Applied Physics*, 123(20):204309.

[Hill and Pillsbury, 1939] Hill, W. R. and Pillsbury, D. M. (1939). *Argyria: the pharmacology of silver*. Williams & Wilkins.

[Hirai et al., 2007] Hirai, N., Ishikawa, H., and Ohki, Y. (2007). Electrical conduction properties of several biodegradable polymers. In *2007 Annual Report - Conference on Electrical Insulation and Dielectric Phenomena*. IEEE.

[Hirose and Hirose, 1976] Hirose, Y. and Hirose, H. (1976). Polarity-dependent memory switching and behavior of Ag dendrite in Ag-photodoped amorphous As2S3films. *Journal of Applied Physics*, 47(6):2767–2772.

[Hodgkin and Huxley, 1952] Hodgkin, A. L. and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4):500–544.

[Horrocks et al., 2014] Horrocks, G. A., Singh, S., Likely, M. F., Sambandamurthy, G., and Banerjee, S. (2014). Scalable Hydrothermal Synthesis of Free-Standing VO2 Nanowires in the M1 Phase. *ACS Applied Materials & Interfaces*, 6(18):15726–15732.

[Hsu et al., 2016] Hsu, C.-L., Tsai, J.-Y., and Hsueh, T.-J. (2016). Ethanol gas and humidity sensors of CuO/Cu2O composite nanowires based on a Cu through-silicon via approach. *Sensors and Actuators B: Chemical*, 224:95–102.

[Hu et al., 2018] Hu, M., Graves, C., Li, C., Li, Y., Ge, N., Montgomery, E., Davila, N., Jiang, H., Williams, R., Yang, J., Xia, Q., and Strachan, J. (2018). Memristor-Based Analog Computation and Neural Network Classification with a Dot Product Engine. *Adv Mater*, 30.

[Hu et al., 2008a] Hu, N., Karube, Y., Yan, C., Masuda, Z., and Fukunaga, H. (2008a). Tunneling effect in a polymer/carbon nanotube nanocomposite strain sensor. *Acta Materialia*, 56(13):2929–2936.

[Hu et al., 2008b] Hu, N., Masuda, Z., Yan, C., Yamamoto, G., Fukunaga, H., and Hashida, T. (2008b). The electrical properties of polymer nanocomposites with carbon nanotube fillers. *Nanotechnology*, 19(21):215701.

[Ielmini et al., 2011] Ielmini, D., Bruchhaus, R., and Waser, R. (2011). Thermochemical resistive switching: materials mechanisms, and scaling projections. *Phase Transitions*, 84(7):570–602.

[Jaeger, 2002] Jaeger, H. (2002). *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the" echo state network" approach*, volume 5. GMD-Forschungszentrum Informationstechnik Bonn.

[Jaeger, 2004] Jaeger, H. (2004). Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication. *Science*, 304(5667):78–80.

[Jaeger, 2007] Jaeger, H. (2007). Echo state network. *Scholarpedia*, 2(9):2330.

[Jaeger, 2017] Jaeger, H. (2017). Controlling Recurrent Neural Networks by Conceptors.

[Jin et al., 2018] Jin, P., Wang, G., Iu, H. H.-C., and Fernando, T. (2018). A Locally Active Memristor and Its Application in a Chaotic Circuit. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 65(2):246–250.

[K. Koch, 1982] K. Koch, Poggio, T. (1982). Retinal ganglion cells: a functional interpretation of dendritic morphology. *Philosophical Transactions of the Royal Society of London. B Biological Sciences*, 298(1090):227–263.

[Kagan, 2015] Kagan, M. (2015). On equivalent resistance of electrical circuits. *American Journal of Physics*, 83(1):53–63.

[Kareiva and Shigesada, 1983] Kareiva, P. M. and Shigesada, N. (1983). Analyzing insect movement as a correlated random walk. *Oecologia*, 56(2-3):234–238.

[Katsura et al., 1985a] Katsura, T., Kamal, M. R., and Utracki, L. A. (1985a). Electrical and thermal properties of polypropylene filled with steel fibers. *Advances in Polymer Technology*, 5(3):193–202.

[Katsura et al., 1985b] Katsura, T., Kamal, M. R., and Utracki, L. A. (1985b). Some properties of polypropylene filled with metal fibers. *Polymer Composites*, 6(4):282–295.

[Kawai et al., 2019] Kawai, Y., Park, J., and Asada, M. (2019). A small-world topology enhances the echo state property and signal propagation in reservoir computing. *Neural Networks*, 112:15–23.

[Ke et al., 2018] Ke, Y., Wang, S., Liu, G., Li, M., White, T. J., and Long, Y. (2018). Vanadium Dioxide: The Multistimuli Responsive Material and Its Applications. *Small*, 14(39):1802025.

[Keblinski and Cleri, 2004] Keblinski, P. and Cleri, F. (2004). Contact resistance in percolating networks. *Physical Review B*, 69(18).

[Khanarian et al., 2013] Khanarian, G., Joo, J., Liu, X.-Q., Eastman, P., Werner, D., O'Connell, K., and Trefonas, P. (2013). The optical and electrical properties of silver nanowire mesh films. *Journal of Applied Physics*, 114(2):024302.

[Kirkpatrick, 1973] Kirkpatrick, S. (1973). Percolation and Conduction. *Reviews of Modern Physics*, 45(4):574–588.

[Koch, 1998] Koch, C. (1998). *Biophysics of Computation*. Oxford University Press.

[Kohlrausch, 1854] Kohlrausch, R. (1854). Theorie des elektrischen Rückstandes in der Leidener Flasche. *Annalen der Physik und Chemie*, 167(2):179–214.

[Kolmogorov, 1998] Kolmogorov, A. (1998). On tables of random numbers. *Theoretical Computer Science*, 207(2):387–395.

[Kund et al., 2005] Kund, M., Beitel, G., Pinnow, C.-U., Rohr, T., Schumann, J., Symanczyk, R., Ufert, K., and Muller, G. (2005). Conductive bridging RAM (CBRAM): an emerging non-volatile memory technology scalable to sub 20nm. In *IEEE InternationalElectron Devices Meeting 2005. IEDM Technical Digest.* IEEE.

[Kupper, 2012] Kupper, M. (2012). Very Sharp Platinum Tips by Electrochemical Etching. Master's thesis, Graz University of Technology.

[Lagrange et al., 2015] Lagrange, M., Langley, D. P., Giusti, G., Jiménez, C., Bréchet, Y., and Bellet, D. (2015). Optimization of silver nanowire-based transparent electrodes: effects of density size and thermal annealing. *Nanoscale*, 7(41):17410–17423.

[Langer, 1980] Langer, J. S. (1980). Instabilities and pattern formation in crystal growth. *Reviews of Modern Physics*, 52(1):1–28.

[Lansdown, 2006] Lansdown, A. B. (2006). Silver in Health Care: Antimicrobial Effects and Safety in Use. In *Biofunctional Textiles and the Skin*, pages 17–34. KARGER.

[Lecun et al., 1998] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[Lee and Kirkwood, 2019] Lee, H.-K. and Kirkwood, A. (2019). Mechanisms of Homeostatic Synaptic Plasticity in vivo. *Frontiers in Cellular Neuroscience*, 13.

[Levard et al., 2012] Levard, C., Hotze, E. M., Lowry, G. V., and Brown, G. E. (2012). Environmental Transformations of Silver Nanoparticles: Impact on Stability and Toxicity. *Environmental Science & Technology*, 46(13):6900–6914.

[Li et al., 2018] Li, C., Belkin, D., Li, Y., Yan, P., Hu, M., Ge, N., Jiang, H., Montgomery, E., Lin, P., Wang, Z., Song, W., Strachan, J. P., Barnell, M., Wu, Q., Williams, R. S., Yang, J. J., and Xia, Q. (2018). Efficient and self-adaptive in-situ learning in multilayer memristor neural networks. *Nature Communications*, 9(1).

[Li et al., 2007] Li, C., Thostenson, E. T., and Chou, T.-W. (2007). Dominant role of tunneling resistance in the electrical conductivity of carbon nanotube–based composites. *Applied Physics Letters*, 91(22):223114.

[Li and Zhang, 2010] Li, J. and Zhang, S.-L. (2010). Conductivity exponents in stick percolation. *Physical Review E*, 81(2).

[Li and Östling, 2015] Li, J. and Östling, M. (2015). Conductivity scaling in supercritical percolation of nanoparticles – not a power law. *Nanoscale*, 7(8):3424–3428.

[Li et al., 2016] Li, W., Liu, X., Wang, Y., Dai, Z., Wu, W., Cheng, L., Zhang, Y., Liu, Q., Xiao, X., and Jiang, C. (2016). Design of high-performance memristor cell using W-implanted SiO2 films. *Applied Physics Letters*, 108(15):153501.

[Liao et al., 2010] Liao, Z.-M., Hou, C., Zhang, H.-Z., Wang, D.-S., and Yu, D.-P. (2010). Evolution of resistive switching over bias duration of single Ag2S nanowires. *Applied Physics Letters*, 96(20):203109.

[Lilak et al., 2021] Lilak, S., Woods, W., Scharnhorst, K., Dunham, C., Teuscher, C., Stieg, A. Z., and Gimzewski, J. K. (2021). Spoken Digit Classification by In-Materio Reservoir Computing With Neuromorphic Atomic Switch Networks. *Frontiers in Nanotechnology*, 3.

[Lu et al., 2020] Lu, L., Hu, X., Zhu, Z., Li, D., Tian, S., and Chen, Z. (2020). Review—Electrochemical Sensors and Biosensors Modified with Binary Nanocomposite for Food Safety. *Journal of The Electrochemical Society*, 167(3):037512.

[Lukoševičius and Jaeger, 2009] Lukoševičius, M. and Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149.

[Luo et al., 2014] Luo, Y., Zhao, R., and Pendry, J. B. (2014). van der Waals interactions at the nanoscale: The effects of nonlocality. *Proceedings of the National Academy of Sciences*, 111(52):18422–18427.

[Lv et al., 2008] Lv, R., Kang, F., Gu, J., Gui, X., Wei, J., Wang, K., and Wu, D. (2008). Carbon nanotubes filled with ferromagnetic alloy nanowires: Lightweight and wide-band microwave absorber. *Applied Physics Letters*, 93(22):223105.

[Ma et al., 2010] Ma, P.-C., Mo, S.-Y., Tang, B.-Z., and Kim, J.-K. (2010). Dispersion interfacial interaction and re-agglomeration of functionalized carbon nanotubes in epoxy composites. *Carbon*, 48(6):1824–1834.

[Maass et al., 2005] Maass, W., Joshi, P., and Sontag, E. (2005). Computational aspects of feedback in neural circuits. *PLoS Computational Biology*, preprint(2006):e165.

[Maass et al., 2002] Maass, W., Natschläger, T., and Markram, H. (2002). Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Computation*, 14(11):2531–2560.

[Manning et al., 2018] Manning, H. G., Niosi, F., da Rocha, C. G., Bellew, A. T., O'Callaghan, C., Biswas, S., Flowers, P. F., Wiley, B. J., Holmes, J. D., Ferreira, M. S., and Boland, J. J. (2018). Emergence of winner-takes-all connectivity paths in random nanowire networks. *Nature Communications*, 9(1).

[Massey et al., 2015] Massey, M. K., Kotsialos, A., Qaiser, F., Zeze, D. A., Pearson, C., Volpati, D., Bowen, L., and Petty, M. C. (2015). Computing with carbon nanotubes: Optimization of threshold logic gates using disordered nanotube/polymer composites. *Journal of Applied Physics*, 117(13):134903.

[Massey et al., 2016] Massey, M. K., Kotsialos, A., Volpati, D., Vissol-Gaudin, E., Pearson, C., Bowen, L., Obara, B., Zeze, D. A., Groves, C., and Petty, M. C. (2016). Evolution of Electronic Circuits using Carbon Nanotube Composites. *Scientific Reports*, 6(1).

[Midya et al., 2019] Midya, R., Wang, Z., Asapu, S., Zhang, X., Rao, M., Song, W., Zhuo, Y., Upadhyay, N., Xia, Q., and Yang, J. J. (2019). Reservoir Computing Using Diffusive Memristors. *Advanced Intelligent Systems*, 1(7):1900084.

[Milano et al., 2018] Milano, G., Luebben, M., Ma, Z., Dunin-Borkowski, R., Boarino, L., Pirri, C. F., Waser, R., Ricciardi, C., and Valov, I. (2018). Self-limited single nanowire systems combining all-in-one memristive and neuromorphic functionalities. *Nature Communications*, 9(1).

[Milano et al., 2019] Milano, G., Pedretti, G., Fretto, M., Boarino, L., Benfenati, F., Ielmini, D., Valov, I., and Ricciardi, C. (2019). Self-organizing memristive nanowire networks with structural plasticity emulate biological neuronal circuits. *arXiv preprint arXiv:1909.02438*.

[Miller et al., 2014] Miller, J. F., Harding, S. L., and Tufte, G. (2014). Evolution-in-materio: evolving computation in materials. *Evolutionary Intelligence*, 7(1):49–67.

[Mohid et al., 2014] Mohid, M., Miller, J. F., Harding, S. L., Tufte, G., Lykkebo, O. R., Massey, M. K., and Petty, M. C. (2014). Evolution-in-materio: Solving bin packing problems using materials. In *2014 IEEE International Conference on Evolvable Systems*. IEEE.

[Moon et al., 2019] Moon, J., Ma, W., Shin, J. H., Cai, F., Du, C., Lee, S. H., and Lu, W. D. (2019). Temporal data classification and forecasting using a memristor-based reservoir computing system. *Nature Electronics*, 2(10):480–487.

[Moore, 2006] Moore, G. E. (2006). Cramming more components onto integrated circuits Reprinted from Electronics, volume 38, number 8, April 19, 1965, pp.114 ff. *IEEE Solid-State Circuits Society Newsletter*, 11(3):33–35.

[Morales and Nocedal, 2011] Morales, J. L. and Nocedal, J. (2011). Remark on "algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization". *ACM Transactions on Mathematical Software*, 38(1):1–4.

[Moreno et al., 2012] Moreno, I., Navascues, N., Irusta, S., and Santamaría, J. (2012). Silver nanowires/polycarbonate composites for conductive films. *IOP Conference Series: Materials Science and Engineering*, 40:012001.

[Motte and Urban, 2005] Motte, L. and Urban, J. (2005). Silver Clusters on Silver Sulfide Nanocrystals: Synthesis and Behavior after Electron Beam Irradiation. *The Journal of Physical Chemistry B*, 109(46):21499–21501.

[Murtagh, 1991] Murtagh, F. (1991). Multilayer perceptrons for classification and regression. *Neurocomputing*, 2(5-6):183–197.

[Musk, 2019] Musk, E. (2019). An Integrated Brain-Machine Interface Platform With Thousands of Channels. *Journal of Medical Internet Research*, 21(10):e16194.

[Mutiso et al., 2013] Mutiso, R., Sherrott, M., Rathmell, A., Wiley, B., and Winey, K. (2013). Integrating simulations and experiments to predict sheet resistance and optical transmittance in nanowire films for transparent conductors. *ACS Nano*, 7:7654–63.

[Nakajima, 2020] Nakajima, K. (2020). Physical reservoir computing—an introductory perspective. *Japanese Journal of Applied Physics*, 59(6):060501.

[Nayak et al., 2012] Nayak, A., Ohno, T., Tsuruoka, T., Terabe, K., Hasegawa, T., Gimzewski, J. K., and Aono, M. (2012). Controlling the Synaptic Plasticity of a Cu2S Gap-Type Atomic Switch. *Advanced Functional Materials*, 22(17):3606–3613.

[Ni et al., 2019] Ni, X., Hui, C., Su, N., Cutler, R., and Liu, F. (2019). A 3D percolation model for multicomponent nanocarbon composites: the critical role of nematic transition. *Nanotechnology*, 30(18):185302.

[Ni et al., 2018] Ni, X., Hui, C., Su, N., Jiang, W., and Liu, F. (2018). Monte Carlo simulations of electrical percolation in multicomponent thin films with nanofillers. *Nanotechnology*, 29(7):075401.

[Nurmikko et al., 2010a] Nurmikko, A., Donoghue, J., Hochberg, L., Patterson, W., Song, Y., Bull, C., Borton, D., Laiwalla, F., Park, S., Ming, Y., and Aceros, J. (2010a). Listening to Brain Microcircuits for Interfacing With External World-Progress in Wireless Implantable Microelectronic Neuroengineering Devices: Experimental systems are described for electrical recording in the brain using multiple microelectrodes and short range implantable or wearable broadcasting units. *Proc IEEE Inst Electr Electron Eng*, 98:375–388.

[Nurmikko et al., 2010b] Nurmikko, A. V., Donoghue, J. P., Hochberg, L. R., Patterson, W. R., Song, Y.-K., Bull, C. W., Borton, D. A., Laiwalla, F., Park, S., Ming, Y., and Aceros, J. (2010b). Listening to Brain Microcircuits for Interfacing With External World—Progress in Wireless Implantable Microelectronic Neuroengineering Devices. *Proceedings of the IEEE*, 98(3):375–388.

[Ogrodzki, 2018] Ogrodzki, J. (2018). *Circuit Simulation Methods and Algorithms*. CRC Press.

[Ohno et al., 2011] Ohno, T., Hasegawa, T., Tsuruoka, T., Terabe, K., Gimzewski, J. K., and Aono, M. (2011). Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials*, 10(8):591–595.

[Oikawa and Kurita, 2016] Oikawa, N. and Kurita, R. (2016). A new mechanism for dendritic pattern formation in dense systems. *Scientific Reports*, 6(1).

[Oleinik, 2019] Oleinik, A. (2019). What are neural networks not good at? On artificial creativity. *Big Data & Society*, 6(1):205395171983943.

[Papernot et al., 2016] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. (2016). The Limitations of Deep Learning in Adversarial Settings. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE.

[Park et al., 2019] Park, S.-H., Hwang, J., Park, G.-S., Ha, J.-H., Zhang, M., Kim, D., Yun, D.-J., Lee, S., and Lee, S. H. (2019). Modeling the electrical resistivity of polymer composites with segregated structures. *Nature Communications*, 10(1).

[Pearson, 1901] Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

[Pergament et al., 2018] Pergament, A., Velichko, A., Belyaev, M., and Putrolaynen, V. (2018). Electrical switching and oscillations in vanadium dioxide. *Physica B: Condensed Matter*, 536:239–248.

[Pershin and Shevchenko, 2017] Pershin, Y. V. and Shevchenko, S. N. (2017). Computing with volatile memristors: an application of non-pinched hysteresis. *Nanotechnology*, 28(7):075204.

[Petrenko, 2018] Petrenko, S. (2018). Limitations of Von Neumann Architecture. In *Big Data Technologies for Monitoring of Computer Security: A Case Study of the Russian Federation*, pages 115–173. Springer International Publishing.

[Pickett et al., 2012] Pickett, M. D., Medeiros-Ribeiro, G., and Williams, R. S. (2012). A scalable neuristor built with Mott memristors. *Nature Materials*, 12(2):114–117.

[Pillage et al., 1994] Pillage, L., Rohrer, R., and Visweswariah, C. (1994). *Electronic Circuit and System Simulation Methods*. Mcgraw-Hill.

[Pothen et al., 1990] Pothen, A., Simon, H. D., and Liou, K.-P. (1990). Partitioning Sparse Matrices with Eigenvectors of Graphs. *SIAM Journal on Matrix Analysis and Applications*, 11(3):430–452.

[Press, 1992] Press, W. H.; Teukolsky, S. A. V. W. T. F. B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*. New York: Cambridge University Press.

[R., 1979] R., H. P. (1979). Synaptic density in human frontal cortex — Developmental changes and effects of aging. *Brain Research*, 163(2):195–205.

[R. Anantharaman, 2020] R. Anantharaman, K. Hall, V. S. A. E. (2020). Circuitscape in Julia: High Performance Connectivity Modelling to Support Conservation Decisions. *JuliaCon Proceedings*, 1(1):58.

[Rahatekar et al., 2010] Rahatekar, S. S., Shaffer, M. S., and Elliott, J. A. (2010). Modelling percolation in fibre and sphere mixtures: Routes to more efficient network formation. *Composites Science and Technology*, 70(2):356–362.

[Román et al., 2018] Román, S., Lund, F., Bustos, J., and Palza, H. (2018). About the relevance of waviness agglomeration, and strain on the electrical behavior of polymer composites filled with carbon nanotubes evaluated by a Monte-Carlo simulation. *Materials Research Express*, 5(1):015044.

[Rosenblatt, 1958] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

[Rusakov et al., 2020] Rusakov, D. A., Savtchenko, L. P., and Latham, P. E. (2020). Noisy Synaptic Conductance: Bug or a Feature? *Trends in Neurosciences*, 43(6):363–372.

[Sadovnikov and Gusev, 2017] Sadovnikov, S. I. and Gusev, A. I. (2017). Recent progress in nanostructured silver sulfide: from synthesis and nonstoichiometry to properties. *Journal of Materials Chemistry A*, 5(34):17676–17704.

[Sahini and Sahimi, 1994] Sahini, M. and Sahimi, M. (1994). *Applications Of Percolation Theory*. CRC Press.

[Sain et al., 2013] Sain, P. K., Goyal, R. K., Bhargava, A. K., and Prasad, Y. V. S. S. (2013). Thermal and electronic behaviour of polycarbonate–copper nanocomposite system. *Journal of Physics D: Applied Physics*, 46(45):455501.

[Saini et al., 2009] Saini, P., Choudhary, V., Singh, B., Mathur, R., and Dhawan, S. (2009). Polyaniline–MWCNT nanocomposites for microwave absorption and EMI shielding. *Materials Chemistry and Physics*, 113(2-3):919–926.

[Sakai, 2020] Sakai, J. (2020). Core Concept: How synaptic pruning shapes neural wiring during development and possibly, in disease. *Proceedings of the National Academy of Sciences*, 117(28):16096–16099.

[Sánta et al., 2019] Sánta, B., Balogh, Z., Gubicza, A., Pósa, L., Krisztián, D., Mihály, G., Csontos, M., and Halbritter, A. (2019). Universal 1/f type current noise of Ag filaments in redox-based memristive nanojunctions. *Nanoscale*, 11(11):4719–4725.

[Sánta et al., 2021] Sánta, B., Balogh, Z., Pósa, L., Krisztián, D., Török, T. N., Molnár, D., Sinkó, C., Hauert, R., Csontos, M., and Halbritter, A. (2021). Noise Tailoring in Memristive Filaments. *ACS Applied Materials & Interfaces*, 13(6):7453–7460.

[Sarwat, 2017] Sarwat, S. G. (2017). Materials science and engineering of phase change random access memory. *Materials Science and Technology*, 33(16):1890–1906.

[Saveliev, 2005] Saveliev, S. V. (2005). *Proizhozhdenie mozga [The origin of the brain]*. Vedi.

[Schaetti et al., 2016] Schaetti, N., Salomon, M., and Couturier, R. (2016). Echo State Networks-Based Reservoir Computing for MNIST Handwritten Digits Recognition. In *2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES)*. IEEE.

[Schnitzler and Gross, 2005] Schnitzler, A. and Gross, J. (2005). Normal and pathological oscillatory communication in the brain. *Nature Reviews Neuroscience*, 6(4):285–296.

[Sejnowski, 1977] Sejnowski, T. J. (1977). Storing covariance with nonlinearly interacting neurons. *Journal of Mathematical Biology*, 4(4):303–321.

[Selzer et al., 2016] Selzer, F., Floresca, C., Kneppe, D., Bormann, L., Sachse, C., Weiß, N., Eychmüller, A., Amassian, A., Müller-Meskamp, L., and Leo, K. (2016). Electrical limit of silver nanowire electrodes: Direct measurement of the nanowire junction resistance. *Applied Physics Letters*, 108(16):163302.

[Shannon, 1948] Shannon, C. E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(4):623–656.

[Sheridan et al., 2017] Sheridan, P. M., Cai, F., Du, C., Ma, W., Zhang, Z., and Lu, W. D. (2017). Sparse coding with memristor networks. *Nature Nanotechnology*, 12(8):784–789.

[Shi et al., 2013] Shi, F., Wang, S., Forest, M. G., and Mucha, P. J. (2013). Percolation-Induced Exponential Scaling in the Large Current Tails of Random Resistor Networks. *Multiscale Modeling & Simulation*, 11(4):1298–1310.

[Shi et al., 2014] Shi, F., Wang, S., Forest, M. G., Mucha, P. J., and Zhou, R. (2014). Network-Based Assessments of Percolation-Induced Current Distributions in Sheared Rod Macromolecular Dispersions. *Multiscale Modeling & Simulation*, 12(1):249–264.

[Sillin et al., 2013] Sillin, H. O., Aguilera, R., Shieh, H.-H., Avizienis, A. V., Aono, M., Stieg, A. Z., and Gimzewski, J. K. (2013). A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology*, 24(38):384004.

[Sillin et al., 2014] Sillin, H. O., Sandouk, E. J., Avizienis, A. V., Aono, M., Stieg, A. Z., and Gimzewski, J. K. (2014). Benchtop Fabrication of Memristive Atomic Switch Networks. *Journal of Nanoscience and Nanotechnology*, 14(4):2792–2798.

[Simonnin et al., 2020] Simonnin, P., Sassi, M., Gilbert, B., Charlet, L., and Rosso, K. M. (2020). Phase Transition and Liquid-like Superionic Conduction in Ag2S. *The Journal of Physical Chemistry C*, 124(18):10150–10158.

[So/rensen et al., 1999] So/rensen, A. H., Hvid, U., Mortensen, M. W., and Mo/rch, K. A. (1999). Preparation of platinum/iridium scanning probe microscopy tips. *Review of Scientific Instruments*, 70(7):3059–3067.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

[Stauffer and Aharony, 1985] Stauffer, D. and Aharony, A. (1985). *Introduction to Percolation Theory*. Taylor & Francis.

[Stauffer and Aharony, 2018] Stauffer, D. and Aharony, A. (2018). *Introduction To Percolation Theory*. Taylor & Francis.

[Stieg et al., 2014] Stieg, A. Z., Avizienis, A. V., Sillin, H. O., Aguilera, R., Shieh, H.-H., Martin-Olmos, C., Sandouk, E. J., Aono, M., and Gimzewski, J. K. (2014). Self-organization and Emergence of Dynamical Structures in Neuromorphic Atomic Switch Networks. In *Memristor Networks*, pages 173–209. Springer International Publishing.

[Stieg et al., 2011] Stieg, A. Z., Avizienis, A. V., Sillin, H. O., Martin-Olmos, C., Aono, M., and Gimzewski, J. K. (2011). Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials*, 24(2):286–293.

[Storme et al., 2015] Storme, P., Schalm, O., and Wiesinger, R. (2015). The sulfidation process of sterling silver in different corrosive environments: impact of the process on the surface films formed and consequences for the conservation-restoration community. *Heritage Science*, 3(1).

[Strogatz, 2018] Strogatz, S. H. (2018). *Nonlinear Dynamics and Chaos*. CRC Press.

[Strukov et al., 2008a] Strukov, D., Snider, G., Stewart, D., and Williams, R. (2008a). The missing memristor found. *Nature*, 453:80–3.

[Strukov et al., 2008b] Strukov, D. B., Snider, G. S., Stewart, D. R., and Williams, R. S. (2008b). The missing memristor found. *Nature*, 453(7191):80–83.

[Sun et al., 1993] Sun, J. S., Gokturk, H. S., and Kalyon, D. M. (1993). Volume and surface resistivity of low-density polyethylene filled with stainless steel fibres. *Journal of Materials Science*, 28(2):364–366.

[Sun et al., 2019] Sun, Z., Pedretti, G., Ambrosi, E., Bricalli, A., Wang, W., and Ielmini, D. (2019). Solving matrix equations in one step with cross-point resistive arrays. *Proceedings of the National Academy of Sciences*, 116(10):4123–4128.

[Sutton and Barto, 2018] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. The MIT Press, second edition.

[Takeda et al., 2011] Takeda, T., Shindo, Y., Kuronuma, Y., and Narita, F. (2011). Modeling and characterization of the electrical conductivity of carbon nanotube-based polymer composites. *Polymer*, 52(17):3852–3856.

[Tanaka et al., 2019] Tanaka, G., Yamane, T., Héroux, J. B., Nakane, R., Kanazawa, N., Takeda, S., Numata, H., Nakano, D., and Hirose, A. (2019). Recent advances in physical reservoir computing: A review. *Neural Networks*, 115:100–123.

[Tasaltin and Basarir, 2014] Tasaltin, C. and Basarir, F. (2014). Preparation of flexible VOC sensor based on carbon nanotubes and gold nanoparticles. *Sensors and Actuators B: Chemical*, 194:173–179.

[Tokuno et al., 2011] Tokuno, T., Nogi, M., Karakawa, M., Jiu, J., Nge, T. T., Aso, Y., and Suganuma, K. (2011). Fabrication of silver nanowire transparent electrodes at room temperature. *Nano Research*, 4(12):1215–1222.

[Totschnig, 2020] Totschnig, W. (2020). Fully Autonomous AI. *Science and Engineering Ethics*, 26(5):2473–2485.

[Truong et al., 2014] Truong, S., Ham, S.-J., and Min, K.-S. (2014). Neuromorphic crossbar circuit with nanoscale filamentary-switching binary memristors for speech recognition. *Nanoscale Research Letters*, 9(1):629.

[Turing, 1937] Turing, A. M. (1937). On Computable Numbers with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265.

[V. Torre, 1978] V. Torre, T. P. (1978). A synaptic mechanism possibly underlying directional selectivity to motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 202(1148):409–416.

[Vafaei et al., 2014] Vafaei, A., Hu, A., and Goldthorpe, I. A. (2014). Joining of Individual Silver Nanowires via Electrical Current. *Nano-Micro Letters*, 6(4):293–300.

[Valov and Kozicki, 2013] Valov, I. and Kozicki, M. N. (2013). Cation-based resistance change memory. *Journal of Physics D: Applied Physics*, 46(7):074005.

[van den Bergen, 1997] van den Bergen, G. (1997). Efficient Collision Detection of Complex Deformable Models using AABB Trees. *Journal of Graphics Tools*, 2(4):1–13.

[Velichko et al., 2018] Velichko, A., Belyaev, M., Putrolaynen, V., Perminov, V., and Pergament, A. (2018). Thermal coupling and effect of subharmonic synchronization in a system of two VO 2 based oscillators. *Solid-State Electronics*, 141:40–49.

[Velichko and Belyaev, 2019] Velichko, A. A. and Belyaev, M. A. (2019). An Investigation of the Effect of the Thermal Coupling Time Delay on the Synchronization of VO2-Oscillators. *Technical Physics Letters*, 45(2):61–64.

[von Neumann, 1993] von Neumann, J. (1993). First draft of a report on the edvac. *IEEE Annals of the History of Computing*, 15(4):27–75.

[Waldrop, 2019] Waldrop, M. M. (2019). News Feature: What are the limits of deep learning? *Proceedings of the National Academy of Sciences*, 116(4):1074–1077.

[Wang et al., 2014] Wang, J., Jiu, J., Araki, T., Nogi, M., Sugahara, T., Nagao, S., Koga, H., He, P., and Suganuma, K. (2014). Silver Nanowire Electrodes: Conductivity Improvement Without Post-treatment and Application in Capacitive Pressure Sensors. *Nano-Micro Letters*, 7(1):51–58.

[Wang et al., 2016] Wang, Z., Joshi, S., Savel'ev, S. E., Jiang, H., Midya, R., Lin, P., Hu, M., Ge, N., Strachan, J. P., Li, Z., Wu, Q., Barnell, M., Li, G.-L., Xin, H. L., Williams, R. S., Xia, Q., and Yang, J. J. (2016). Memristors with diffusive dynamics as synaptic emulators for neuromorphic computing. *Nature Materials*, 16(1):101–108.

[Wang et al., 2020] Wang, Z., Wu, H., Burr, G. W., Hwang, C. S., Wang, K. L., Xia, Q., and Yang, J. J. (2020). Resistive switching materials for information processing. *Nature Reviews Materials*, 5(3):173–195.

[Watts and Strogatz, 1998] Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442.

[Weber and Kamal, 1993] Weber, M. E. and Kamal, M. (1993). Volume Resistivity and Mechanical Properties of Electrically Conductive Long Fiber Composites. *Journal of Reinforced Plastics and Composites*, 12(8):844–853.

[White et al., 2010] White, S. I., Vora, P. M., Kikkawa, J. M., and Winey, K. I. (2010). Resistive Switching in Bulk Silver Nanowire-Polystyrene Composites. *Advanced Functional Materials*, 21(2):233–240.

[Witten and Sander, 1981] Witten, T. A. and Sander, L. M. (1981). Diffusion-Limited Aggregation a Kinetic Critical Phenomenon. *Physical Review Letters*, 47(19):1400–1403.

[Wolpert and Macready, 1997] Wolpert, D. and Macready, W. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.

[Wong et al., 2010] Wong, H.-S. P., Raoux, S., Kim, S., Liang, J., Reifenberg, J. P., Rajendran, B., Asheghi, M., and Goodson, K. E. (2010). Phase Change Memory. *Proceedings of the IEEE*, 98(12):2201–2227.

[Xiao et al., 2020] Xiao, T., Huang, J., Wang, D., Meng, T., and Yang, X. (2020). Au and Au-Based nanomaterials: Synthesis and recent progress in electrochemical sensor applications. *Talanta*, 206:120210.

[Xiong et al., 2016] Xiong, J., Han, C., Li, W., Sun, Q., Chen, J., Chou, S., Li, Z., and Dou, S. (2016). Ambient synthesis of a multifunctional 1D/2D hierarchical Ag–Ag2S nanowire/nanosheet heterostructure with diverse applications. *CrystEngComm*, 18(6):930–937.

[Xu et al., 2010] Xu, Z., Bando, Y., Wang, W., Bai, X., and Golberg, D. (2010). Real-Time In Situ HRTEM-Resolved Resistance Switching of Ag2S Nanoscale Ionic Conductor. *ACS Nano*, 4(5):2515–2522.

[Yampolskiy and Fox, 2012] Yampolskiy, R. V. and Fox, J. (2012). Artificial General Intelligence and the Human Mental Model. In *The Frontiers Collection*, pages 129–145. Springer Berlin Heidelberg.

[Yang et al., 2012] Yang, J. J., Strukov, D. B., and Stewart, D. R. (2012). Memristive devices for computing. *Nature Nanotechnology*, 8(1):13–24.

[Yang et al., 2011] Yang, R.-B., Liang, W.-F., Lin, W.-S., Lin, H.-M., Tsay, C.-Y., and Lin, C.-K. (2011). Microwave absorbing properties of iron nanowire at x-band frequencies. *Journal of Applied Physics*, 109(7):07B527.

[Yang and Calakos, 2013] Yang, Y. and Calakos, N. (2013). Presynaptic long-term plasticity. *Frontiers in Synaptic Neuroscience*, 5.

[Yao et al., 2020] Yao, H., Hsieh, Y., Kong, J., and Hofmann, M. (2020). Modelling electrical conduction in nanostructure assemblies through complex networks. *Nat Mater*.

[Yao et al., 2017] Yao, S., Swetha, P., and Zhu, Y. (2017). Nanomaterial-Enabled Wearable Sensors for Healthcare. *Advanced Healthcare Materials*, 7(1):1700889.

[Yi et al., 2018] Yi, W., Tsang, K. K., Lam, S. K., Bai, X., Crowell, J. A., and Flores, E. A. (2018). Biological plausibility and stochasticity in scalable VO2 active memristor neurons. *Nature Communications*, 9(1).

[Yin et al., 2017] Yin, F., Ye, D., Zhu, C., Qiu, L., and Huang, Y. (2017). Stretchable Highly Durable Ternary Nanocomposite Strain Sensor for Structural Health Monitoring of Flexible Aircraft. *Sensors*, 17(11):2677.

[Yuksel et al., 2018] Yuksel, R., Alpugan, E., and Unalan, H. E. (2018). Coaxial silver nanowire/polypyrrole nanocomposite supercapacitors. *Organic Electronics*, 52:272–280.

[Zador, 1998] Zador, A. (1998). Impact of Synaptic Unreliability on the Information Transmitted by Spiking Neurons. *Journal of Neurophysiology*, 79(3):1219–1229.

[Zare and Rhee, 2020] Zare, Y. and Rhee, K. Y. (2020). Definition of "b" exponent and development of power-law model for electrical conductivity of polymer carbon nanotubes nanocomposites. *Results in Physics*, 16:102945.

[Zhang and Lian, 2016] Zhang, M. and Lian, X. (2016). Rapid Fabrication of High-Aspect-Ratio Platinum Microprobes by Electrochemical Discharge Etching. *Materials*, 9(4):233.

[Zhirnov and Cavin, 2015] Zhirnov, V. V. and Cavin, R. K. (2015). Fundamental limits for logic and memory. In *Microsystems for Bioelectronics*, pages 101–147. Elsevier.

[Zhong et al., 2021] Zhong, Y., Tang, J., Li, X., Gao, B., Qian, H., and Wu, H. (2021). Dynamic memristor-based reservoir computing for high-efficiency temporal signal processing. *Nature Communications*, 12(1).

[Zidan et al., 2018] Zidan, M. A., Jeong, Y., Lee, J., Chen, B., Huang, S., Kushner, M. J., and Lu, W. D. (2018). A general memristor-based partial differential equation solver. *Nature Electronics*, 1(7):411–420.

APPENDIX

APPENDIX A

Acronyms

A summary of all acronyms is documented in Table A.1.

Table A.1: A summary of acronyms used in alphabetical order.

Acronym	Abbreviation
AAIM	Autonomous Artificial Intelligence Machines
AI	Artificial Intelligence
API	Application Programming Interface
CNT	Carbon Nano Tube
DMF	Dimethylformamide
DNA	Deoxyribonucleic Acid
ESN	Echo State Network
GPU	Graphics Processing Unit
HRTEM	High-Resolution Transmission Electron Microscopy
LSM	Liquid State Machine
MWCNT	Multi-Wall Carbon Nano Tube
NICSLU	Nano-scale Integrated Circuits and Systems
NNC	Neuromorphic Nanoscale Composites
PCL	Polycaprolactone
PMMA	Poly(methyl methacrylate)

Table A.1 (Continued)

REST	Representational State Transfer
RNN	Recurrent Neural Network
SEM	Scanning Electron Microscopy
SPICE	Simulation Program with Integrated Circuit Emphasis
TEM	Transmission Electron Microscopy
TMO	Transition Metal Oxide
VLSI	Very Large-Scale Integration