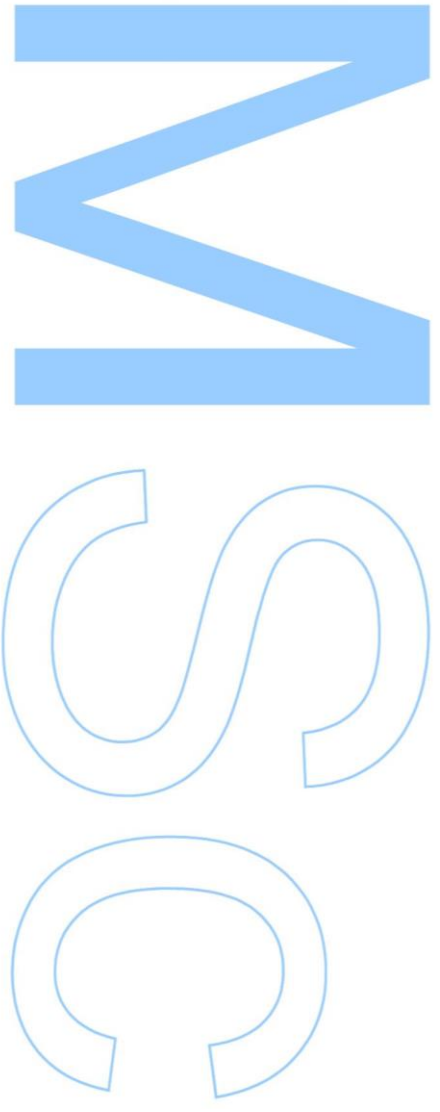
Simulation and analysis of neuro-memristive hybrid circuits

João Alexandre da Silva Pereira Reis
Mestrado em Física
Departamento de Física e Astronomia
2016

Orientador
Paulo de Castro Aguiar, Investigador Auxiliar do Instituto de Investigação e Inovação em Saúde da Universidade do Porto

Coorientador
João Oliveira Ventura, Investigador Auxiliar do Departamento de Física e Astronomia da Universidade do Porto



U.PORTO

FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

Todas as correções determinadas pelo júri, e só essas, foram efetuadas.

O Presidente do Júri,

Porto, _____/_____/_____

University of Porto

Master's degree

Physics

Simulation and analysis of neuro-memristive hybrid circuits

Author:
João Alexandre Reis

Advisor:
Dr. Paulo Aguiar
Co-Advisor:
Dr. João Ventura

A dissertation submitted in partial fulfilment of the requirements
for the degree of Master of Science

at

Department of Physics and Astronomy
Faculty of Science of University of Porto

Acknowledgments

First and foremost, I need to thank my dissertation advisors Dr. Paulo Aguiar and Dr. João Ventura for their constant counsel, however basic my doubts were or which wall I ran into. Regardless of my stubbornness to stick to my way to research and write, they were always there for me.

Of great importance, because of our shared goals, Catarina Dias and Mónica Cerquido helped me have a fixed and practical outlook to my research.

During the this dissertation, I attended MemoCIS, a training school of memristors, which helped me have a more concrete perspective on state of the art research on technical details, modeling considerations and concrete proposed and realized applications. For that I need to acknowledge COST (European Cooperation in Science and Technology) organization for financing my enrollment on the school and costs for the duration of the same.

I also need to acknowledge the organizations that permitted my attendances at the Institute of Nanoscience and Nanotechnology Workshop, the 5th i3s Annual Meeting and the 7th International Conference on Advanced Nanomaterials which enabled me to acquire valuable work experience.

Last but not least, my family was an unfaltering pillar during my years of study and throughout this dissertation's research and writing. Even my erratic schedules and non-availability did not stop their support, and for that, I heartily thank my family.

João Alexandre Reis

Resumo

O objectivo desta dissertação é a de realizar um estudo preparatório sobre a viabilidade de sistemas híbridos de redes de neurónios e memristors.

A investigação em memristors ou dispositivos de comutação de resistência tem recentemente sido frutífera na produção à escala nanométrica de dispositivos duráveis. Este desenvolvimento recente juntamente com técnicas de produção em massa possibilitou a investigação de aplicações em outras áreas. Nomeadamente, a capacidade para mudar a resistência de acordo com o historial do sinal de entrada está a ser usada para desenvolver memórias eletrónicas comerciais e ao nível de investigação cientifica, desenvolver algoritmos e *hardware* que copia certas propriedades do sistema nervoso. Investigação de ponta em bioengenharia a nível celular está a possibilitar aplicações de controlo da atividade elétrica de redes de neurónios biológicas e no futuro próximo, estas áreas de investigação vão estar desenvolvidas o suficiente para testar experimentalmente estes sistemas híbridos hipotéticos. Entre as muitas aplicações biomédicas destes sistemas híbridos, é esperado um papel fundamental no fornecimento de tecnologia para novos elétrodos terapêuticos adaptativos e implantáveis.

É introduzida a literatura relevante sobre a atividade elétrica de neurónios, redes de neurónios e das propriedade gerais de dispositivos memristivos e com a base estabelecida, nesta dissertação, analisa-se numericamente o uso de dispositivos memristivos como sinapses com modelos analíticos e numéricos, procedendo com a proposta de propriedade memristivas que necessitam de ser caracterizadas para qualquer trabalho nesta área. Porque os modelos presentemente disponíveis não são detalhados o suficiente para retirar conclusões preditivas, também são propostos sistemas híbridos genéricos, que aproveitam as capacidades de adaptação e sincronização.

Palavras-chave: neurónio, redes de neurónios biológicos, memristor, adaptação, aprendizagem, sistemas híbridos

Abstract

The aim of this dissertation is to conduct a preparatory study on the feasibility of hybrid systems of biological neural networks and memristive devices.

Research in memristive or resistive switching devices has recently been fruitful in producing nanometer scale and durable devices. This recent development coupled with mass producing techniques has enabled to research their applications on other areas. Most notably, the ability to change their resistance according to the input's history is currently being used to develop commercially feasible electronic memories and on research level, develop algorithms and hardware that mimics certain characteristics of the nervous system. State of the art bioengineering research at the cellular level is paving the way for applications that can fine control the electrical activity of biological neural networks and in the near future, both these areas of research will be developed enough to experimentally test these hypothetical hybrid systems. Among the many biomedical applications of these hybrid systems, it is expected that they will play a fundamental role in providing the technology for novel implantable and adaptive therapeutic electrodes.

The relevant literature is introduced concerning the electrical activity of neurons, neural networks and the general properties of memristive devices. With the basis established, in this dissertation, it is numerically analyzed the uses of memristive devices as synapses with analytical and numerical models and subsequently, propose memristive properties that need to be characterized for any further work in this subject. Because the available models are not detailed enough for many predictive conclusions, generic hybrid systems are proposed, that take advantage of adaption and synchronization.

Keywords: neuron, biological neuronal network, memristor, adaption, learning, hybrid systems

Contents

List of Figures

Nomenclature

CMOS	Complementary Metal-Oxide-Semiconductor
GUI	Graphical interface
HH	Hodgkin and Huxley
HOC	High Order Calculator
IF	Integrate and Fire
LTD	Long Term Depression
LTP	Long Term Potentiation
$N_i M$	Generic definition of hybrid circuit's node
NMODL	Neuron MOdel Description Language
PSR	Post-Synaptic Responses
RAM	Random Access Memory
RRAM	Resistive Random Access Memory
STDP	Spike Timing Dependent Plasticty
VLSI	Very-Large-Scale Integration

Chapter 1

Introduction

Over the last few decades, the accelerated development of computers has shed light on their inherent limitations to process information efficiently over large data sets and to adapt to less than ideal inputs or working conditions. In opposition, humans (and other living beings) have an apparently disorganized structure, the nervous system, that excels at processing environmental data and simultaneously control the rest of the body, on about the same energy consumption as a light bulb. This efficiency coupled with a tremendous adaption capability, results from an architecture that melds processing and memory. These types of architectures are not unique to biological systems, because artificial neural networks are a concept that has been used to model the nervous system and in computer science, as a generic frame to develop algorithms/software that mimic certain aspects of the brain.

Recently, the development of nanotechnology has enabled to translate this computational concept into neural network chips and with state of the art biotechnology, the integration of artificial and biological neural networks is imminent. The difficulties arise from the ubiquitous use of transistors in modern electronics, that do not necessarily result in dynamics compatible with the nervous system, either for typical magnitude of variables or for how the learning/adaption occurs. However, the memristor is a two-terminal nanoscale device, that since its discovery, has been postulated to have the adequate characteristics, including resistance that adapts to the input's history, scalability and low power consumption.

To understand how both these systems can be interfaced, the biological and electrical portions need to be individually studied, before they can be integrated.

The goal of this dissertation is to explore and assess the necessary conditions to effectively bridge activity and information between biological neurons and memristive devices. This is achieved through simulations and analysis of detailed biophysical neuronal models combined with mathematical models of memristor dynamics.

The nervous system is a complex multiscale biological structure, requiring detailed knowledge and integration of biological, chemical and physical processes. To understand how it works, it is

useful to compare it with von Neumann architectures. In typical von Neumann architectures, processors and memory are critical; any action requires data transmission from memory and processor and back. But while in von Neumann architectures memory and processing are separate, in the nervous system both are supported by the same substrate.

This multiscale property of the brain has strong implications in our understanding of how it works. At sub-micrometer (down to nano) scale, ion pumps/transporters regulate the gradient of concentrations between the intra- and extra-cellular media while being tied to the electrical activity of the cellular membrane; molecular neuroscience examines the interaction of ions and other "small" structures in this non-uniform medium. In practice the nervous system cannot be completely understood using such basic components, it becomes necessary to enlarge the scale.

Taking the other extreme, cognitive neuroscience studies how psychological functions like perception and emotions emerge from relatively simple building blocks and how external social or environmental stimulus influence its subsystems. This research field relies on an array of different techniques, including psychology, behavioral science, brain scanning and mapping, direct and indirect electric stimulation.

This dissertation focuses on cellular and systems neuroscience, areas of research that analyze neurons and synapses and how they couple to form subsystems of the brain. Neurons and synapses are the base units of the nervous system architecture; it basically comes down to very simple ideas: neurons are cells that emit voltage spikes and are connected with synapses. When these spikes travel through the synapses, they change the synapses' state and are transmitted with some modification. In von Neumann vocabulary, processing and memory write/read cycles occur simultaneously and cannot be separated.

On the scale of neurons and neural networks, the intrinsic chemical reactions induced by neural dynamics are not important for the overall dynamics, meaning that, at this size scale the relevant portions of neural systems are related to signal generation and transmission. Given that for this context what matters is the electrical signals, the mentioned voltage spikes form the basis of most results and discussions in this dissertation; they are localized in time and space and are self-propagating waves that disperse little energy, akin to an event. It is this property that permits to use an event-driven approach to analyze neural networks. By the same perspective, biologically detailed simulations are impractical for large networks, due to the sheer number of variables that need to be accounted for. With neural networks, cells can be reduced to a dimensionless structure and replaced by a set of differential equations; in practical terms, the cell is replaced by an electric circuit that mimics the electric activity and produces spikes with the same characteristics.

As mentioned before, the main goal of this dissertation is to find out how can these types of biological systems interact with the recently developed memristors or memristive devices, resistor-like devices that switch their value according to the input's history. These devices where theoretically introduced in 1971, using group theory arguments and recognized experimentally in 2008. While

there are a few established and/or accurate analytic models, they are developed to describe physical devices on fairly limited ranges, producing non physical results outside those ranges; I will start by analyzing analytic models and at a later point use treated experimental data.

Current complementary metal-oxide-semiconductor (CMOS) and very-large-scale integration (VLSI) technologies have already made advances on mimicking neural dynamics. Memristive devices can be produced at nanometer scale and require little power, making them a prime candidate for an adaptable electrical element, but biological neural networks and memristive devices do not function within the same magnitude ranges of voltage, currents and time. Memristive models need to be correctly parameterized to optimally interface with neurons and, with that information, some possible schemes of memristive synapses are built and analyzed how they influence neuronal dynamics.

Because of the variability of devices, specific hybrid circuit applications need to be constructed with specific types of memristors in mind. Different tests are developed/reiterated to identify several crucial characteristics, like thresholds, sensitivity, type of device, imperfections... With those basic elements in place, schemes of hybrid circuits are rendered: 2d arrays, 3d arrays, artificial memory banks, adaptive rectifying circuits; basically taking established electrical circuits and imprinting memristive properties.

Currently, it is possible to interface the brain and electrical/electronic systems, being a common occurrence for research purposes. Except for the simplest applications (like recording the electrical activity), this interface is done via non-portable systems, at the expense of scalability and energy efficiency. Without considering how neuronal signals could be read and transmitted to a computer, the intended applications with *in silico* models should require at least a mid range CPU/GPU with 10^2 to 10^3 W power consumption, while the human body wattage averages at about 100 W; an ideal system should be implantable (biocompatible) and require no external power supply, effectively drawing the necessary power from the nervous system.

Eventually, these systems could be used to treat various pathologies in the nervous system: replace missing nerves/neural pathways from accidents or malformations, regulate excessive electrical activity from epilepsy or neurodegenerative diseases. Current research level RRAM (Resistive Random Access Memory), RAM that uses memristors, can, in principle, also be adapted to extend our own memory and cognitive skills, with biocompatible artificial memory banks.

Chapter 2 reviews the relevant properties of the nervous system, focusing on certain aspects of cellular membrane modeling and neuronal cable theory, following up into the biological mechanisms of neural adaption. Establishing the relevant working principles of neural networks, history of memristor devices is reviewed, generic models and uses for computer memory and neuromorphic applications.

Chapter 3 expands on the methods used in this dissertation, namely on the simulation environment NEURON, a combination of different programming tools/languages that is optimized for

neurons and neural network modeling. A small revision of numerical tools and typical workflow in this framework is done.

Finally, chapter 4 consists of the majority of the obtained results. It starts with a preamble section, gathering some results that are not completely original and that are developed for this context. With the basis of results to work with, possible memristive synapses are proposed and given a scheme to implement on NEURON, ending this section with numerical results that arise from connecting neurons and memristors.

With several characteristics and problems identified, chapter 5 expands on workarounds to difficulties (when found), how those characteristics can affect the dynamics of hybrid circuits/systems and how they can be identified. Hybrid systems that were not implemented and therefore not analyzed numerically, are proposed for less generic uses.

Chapter 6 gathers relevant results and conclusions, and lays down the work plan for a project in this field.

Chapter 2

Literature Review

Contents

2.1 Nervous system and the brain

There are many possible pathologies in the nervous system: severe neurological diseases caused by mutated ion pumps [2, 3], neurons with incorrect morphology or wrongly positioned neural circuits [4, 5], diseases on a system level magnitude such as depression, obsessive-compulsive disorder, Alzheimer or Parkinson. These pathologies can all affect the electrical activity of the nervous system and undo its general operating architecture.

The brain is the center of the nervous system and it manages many aspects of the body, containing up to 10^{12} neurons (and about as much nonneuronal cells [6]) with, for example, 10^{10} neurons in the neocortex [7] or 10^{5} neurons in the hippocampus [8]; and about 10^{14} synapses in the human neocortex [9], containing about 75% of the brain's volume [10]. Understanding this system, from the standpoint of electrical activity and bioelectrical circuits, is fundamental for the intended applications with memristive devices or other electrical elements.

2.1.1 Biophysical models

In neuroscience, there are several ways to study the brain. Molecular and cellular neuroscience characterize the chemistry and physics of neurons, synapses and propagation of signals; systems

neuroscience already ignores the finer details of neurons, but studies how neural circuits are formed and react; finally, at a larger scope, cognitive neuroscience studies the interactions of neural circuits and the environment that result in psychological functions.

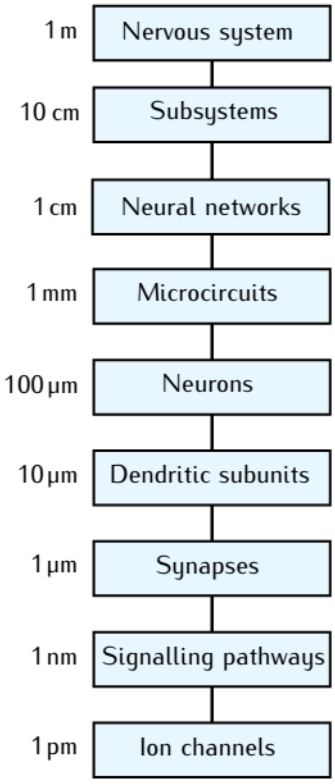


Figure 2.1: Complexity levels of the nervous system [1]

Neurons are a necessary building block to understand the nervous system, but in figure 2.1 they are at the middle of the scale; 10^{-4} smaller than the entire system and 10^{-8} larger than ion channels/pumps. Neurons are a middle ground in the size scale, providing an effective element of the nervous system for neural networks applications. A neuron is a cell that reacts to electrical and chemical inputs; its cellular body connects to other neurons through synapses, where electrochemical signals travel (see figure 2.2). The electrical signal is controlled by ionic pumps in the cellular membrane, the chemical portion of the dynamics that controls concentration gradients between extra- and intra-cellular mediums.

Action potentials occur when membrane potential exhibits fast changes, within a few milliseconds and typically crossing zero voltage value; if connected to other neurons, such events can induce more action potentials. A model of these signals depends on the morphology and composition of cells, ionic gates, propagation in non-linear media (includes diffusion, drift and wave equations) and their interactions.

Even the simplest model of a full nervous system, in the perspective of system neuroscience, requires a set of 10^{14} variables (neurons + synapses). Considering that the involved differential equations can be nonlinear, convergence issues involves iterative solutions at each instant, increasing the actual number of calculations per time step.

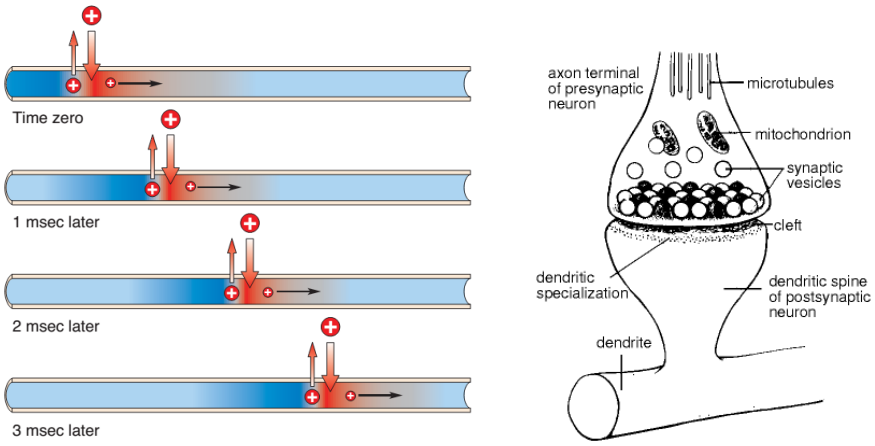


Figure 2.2: Propagation of signal within a axon [11] and from axon to dendrite [12]

At such a high number of variables a detailed analytical analysis is impossible but physics has since long employed statistical tools (statistical mechanics, thermodynamics, among others) to evaluate the characteristics of large systems. But there are actually two reasons why it fails in this case: system size and homogeneity, and precision of spike time (temporal coding). To correctly describe systems with statistical tools, the brain would need to meet certain requirements of size, homogeneity and equilibrium. The timing of each spike needs to be known to window of a few milliseconds. Take for example a set of experiments published in 1996, where subjects need to recognize a given property in an image shown for 20 ms; a clear sign of recognition happens after 150 ms [13]. Figure 2.3 shows the general features of a visual system and with 18 layers (plus feedback), there is less than 10 ms for spikes to pass from one layer to the other; typical firing rates have an upper value of around 100 Hz, meaning that each layer can fire once in that time window. This shows that the information content of a single spike, largely encoded in its precise spike time, is very large.

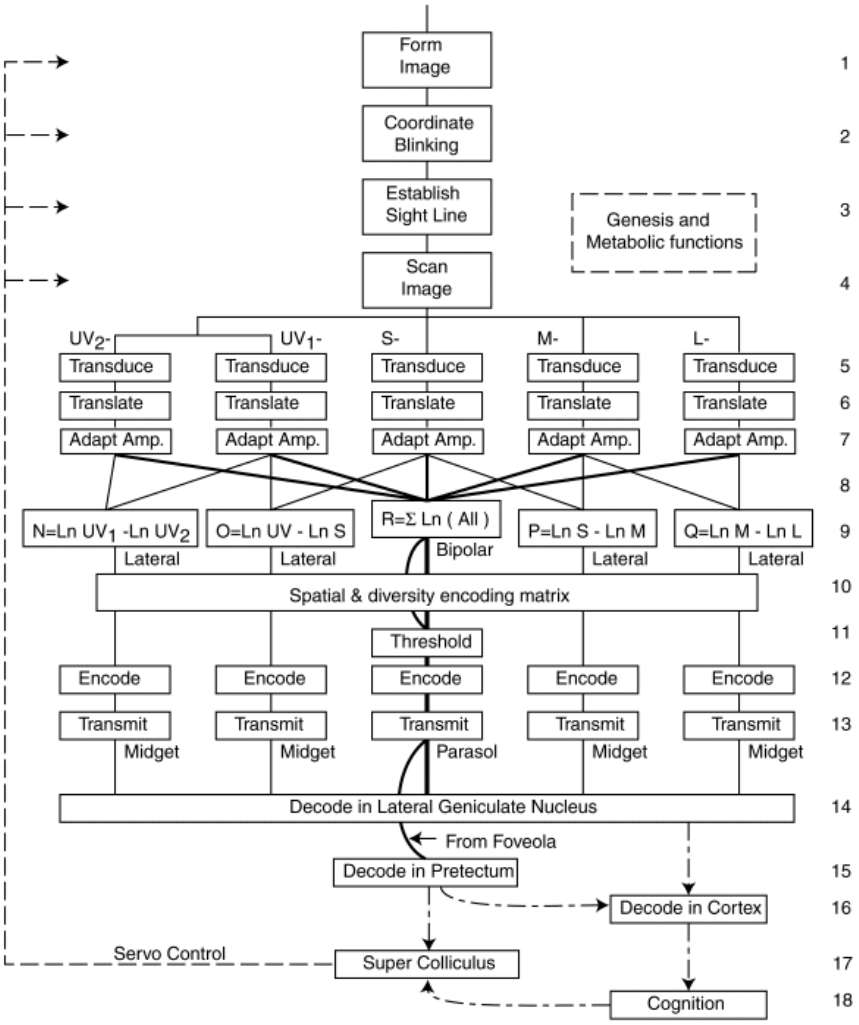


Figure 2.3: Generic visual system ([14], Section 11.6.3)

Taking direct inspiration from statistical physics, the thermodynamic limit requires a system that can be extended to infinity by approximation or by use of its geometry and symmetry and that is sufficiently homogeneous. But the nervous system is separated into many different subsystems, for example, to see, to regulate variables like concentrations or temperature, to make muscles react and more; the system as a whole performs these functions simultaneously, therefore it is an heterogeneous system.

2.2 Neurons

The basic description of a neuron includes a soma, the neuronal membrane, axons, synapses and dendrites ([11], pp. 29-46 and pp. 111-114).

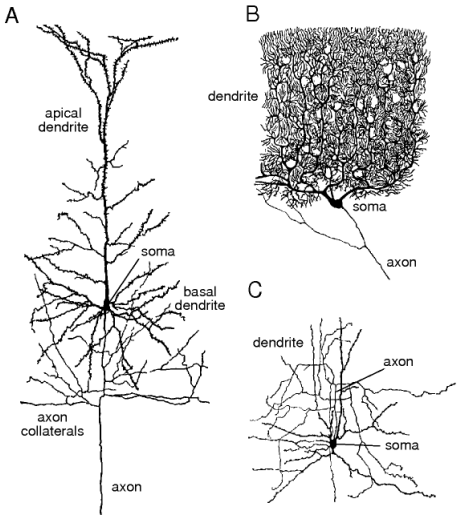


Figure 2.4: Possible configurations of neurons [12]

The soma (or cellular body) is an approximately spherical structure with a radius on the magnitude of micrometers; in terms of organelles, the contents of the soma are similar to other cells. While the cellular body plays an important role in maintaining the homeostasis of the cell, the neuronal membrane has most of the properties that allow the typical spiking activity of neurons, through pumps and ion channels in its surface which control the intra-cellular ionic concentrations and the ionic currents across the cell membrane; they are distributed non-uniformly and make the propagation of action potentials possible.

To best understand axons, synapses and dendrites, in this context, they can be reduced to their ability to transmit signals between cells. Axons extend outwards from the soma, generally many orders of magnitude above the typical soma size, up to a meter, and are the longest structure of neurons, akin to a wire; they can branch out to multiple axons, with each propagating the original signal and along axons, the membrane properties can change, assuming a bulgy disk shape that maximizes the contact with other structures. The point of contact is a synapse, which contains a high concentration of small structures, the most relevant of those structures to neural transmission are vesicles that release neurotransmitters (chemical messengers) to dendrites; synapses can be chemical or electrical (or gap junctions). Just like axons, dendrites extend outwards from the soma but are normally shorter; the dendritic membrane contains receptors that detect the neurotransmitters, which allows to transmit the signal outwards to the soma. These signals can be back

propagated [15] and annihilate if they collide with other signals [16].

The generic neuron contains these four elements, but the signal can be spatially directed, like schemes A or B in figure 2.4, or have an amorphous structure, akin scheme C.

The signals that characterize the electrical activity of the nervous systems are called voltage spikes or action potentials, and are transverse waves where the ionic current is between the intra- and extra-cellular medium. By the properties of membrane's depolarization, when an action potential travels in a patch of membrane, it does not recover immediately, which drives the spike into areas of the membrane that have not been depolarized within a specific time window, the refractory period; it is this transient characteristic of the membrane that turns the current orthogonal to a net current parallel to the membrane.

Except in very basic situations, this model for signal transmission is too complex to handle analytically and too numerically intensive for network simulations. At this point, any model needs to be simpler, with neuronal cable theory becoming very useful.

2.2.1 Hodgkin-Huxley model

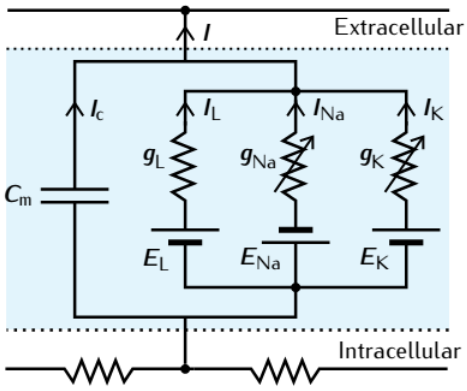


Figure 2.5: Patch of membrane that follows the Hodgkin-Huxley model ([1], p.50)

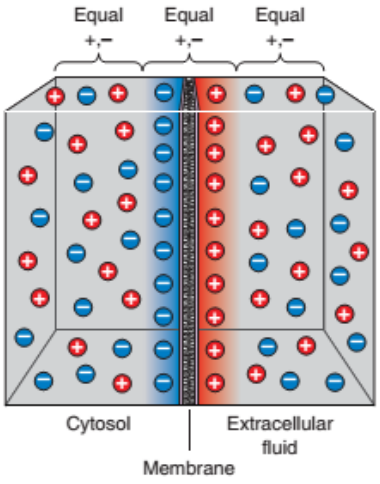


Figure 2.6: Ions near the membrane ([11], p.68)

Because the thickness of the neuronal membrane is on the order of a few nanometers, the electrostatic interaction between intra- and extra-cellular ions attract each other to this layer (figure 2.6). Storing electric charge like this allows to identify the capacitance of the membrane, C_m; the membrane tries to maintain a chemical equilibrium through a reversal potential (E_L) with ions flowing through it against a membrane resistance (R_m). The reversal potential is a value of voltage at which a particular type of ion or set of ions have no net current across the membrane.

In 1952, Hodgkin and Huxley published a model describing the electrical activity of giant squid fibers [17], modeling the membrane with three resistances: one leakage resistance ($R_m \rightarrow {}^{1}/_{g_L}$) with a reversal potential (E_L), a sodium resistance ($g_{Na} = \overline{g_{Na}} m^3 h$) with another reversal potential (E_{Na}) for Na^+ and a similar resistance for potassium, K^- ($g_K = \overline{g_K} n^4$ and E_K). This can also be referred as the HH model, with the circuit exemplified in figure 2.5. Summing all currents, including an electrode (I_e, not in figure 2.5), results in:

$$C_m \frac{dV}{dt} = -g_L (V - E_L) - \overline{g_{Na}} m^3 h (V - E_{Na}) - \overline{g_K} n^4 (V - E_K) + I_e(t) \tag{2.1}$$

with

$$\begin{aligned} C_m &= 1\,\mu F\,cm^{-2} & & \\ E_{Na} &= 50\,mV & \overline{g_{Na}} &= 120\,mS\,cm^{-2} \\ E_K &= -77\,mV & \overline{g_K} &= 36\,mS\,cm^{-2} \\ E_L &= -54.4\,mV & g_L &= 0.3\,mS\,cm^{-2} \end{aligned}$$

The variables m, h and n are probabilities of the ionic pumps opening and are adimensional state variables between 0 and 1, referred as gating variables and are voltage controlled by the rate constants α and β, with the following differential equations:

$$\frac{d\mathbf{A}}{dt} = \alpha_{\mathbf{A}}(1 - \mathbf{A}) - \beta_{\mathbf{A}}\mathbf{A} \qquad \mathbf{A} = m, h, n \tag{2.2}$$

For sodium, one has:

$$\begin{aligned} \alpha_m &= 0.1 \frac{V+40}{1-\exp(-(V+40)/10)} & \alpha_h &= 0.07 \exp\left(-(V+65)/20\right) \\ \beta_m &= 4 \exp\left(-(V+65)/18\right) & \beta_h &= \frac{1}{1+\exp(-(V+35)/10)} \end{aligned}$$

While for potassium:

$$\alpha_n = 0.01 \frac{V+55}{1-\exp(-(V+55)/10)} \qquad \beta_n = 0.125 \exp\left(-(V+65)/80\right)$$

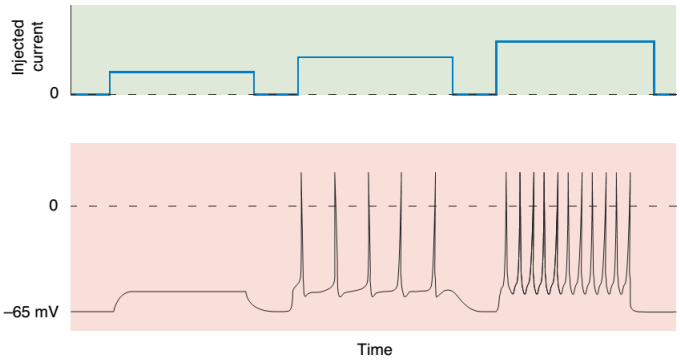


Figure 2.7: Typical electrical activity of a realistic neuron ([11], p.85)

The shape of the spike is dependent on the gating variables, causing the sudden decrease in voltage when above 0. Figure 2.7 exemplifies the dependency of firing rate of spikes on I_e, where the capacitive properties are magnified when below a set threshold value and the resistive properties are magnified when above the threshold.

This set of equations, four ordinary differential equations, can be numerically too taxing to incorporate in a neural network, requiring major simplifications. The most used of these simplifications is the **integrate and fire** (IF) model and variations, used in computational cellular neural networks research [18, 19], taking into account only the passive properties (leakage current and capacitance) and a numerical approximation:

$$\tau_m \frac{dV}{dt} = E_m - V + R_m I_e \text{ and } \tau_m = R_m C_m. \tag{2.3}$$

The numeric approximation includes a threshold and reset voltage with the following condition: if $V(t) = V_{threshold} \rightarrow V(t + \Delta t) = V_{reset}$.

The quantity $\tau_m = R_m C_m$ has time units; this is the typical decay time of the voltage spike, when the neuron is in its refractory period. While it does not have many biophysical details, in some applications, this model is acceptable.

The previous models describe patches of neuronal membrane, but are not enough to explain the spatial characteristics of neural networks. A problem of particular importance in the spatial domain is understanding the spatial range of locally produced voltage perturbations, as well as understanding signal propagation. Neuronal cable theory was created for this purpose, but it does not describe the chemical processes that happen in the membrane, it models, instead, an approximate circuit that mimics the passive electrical properties of a cylindrical portion of membrane. The resulting equation is a 1D diffusion partial differential equation (same as an heat equation), similar to Eq.2.4. In addition to the membrane time constant τ, this equation introduces also the membrane space constant λ which defines the spatial scale for how far a local perturbation can reach in the membrane. In simulations, the continuous cable equation is replaced by a spatial discretization of the neuronal fibers, either axon or dendrites (figure 2.8).

$$\tau \frac{dV}{dt} = \begin{bmatrix} \text{Model specific} \\ \text{current channels} \end{bmatrix} + \lambda^2 \frac{\partial^2 V}{\partial x^2} \tag{2.4}$$

Figure 2.8: Discretized axon ([1], p.36)

In each of the smaller segments, an electrical circuit is associated, akin figure 2.9, where these segments of membrane only have the most basic of features that allow signal propagation.

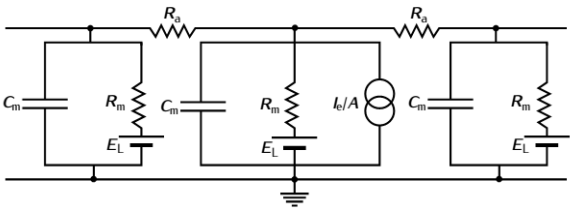


Figure 2.9: Equivalent circuit of discretized axon ([1], p.36)

Notice that neural circuits do not have solely electrical properties, but also need to be considered in the medium of the nervous system. Each equivalent circuit (soma, axon, dendrite or patch of cellular membrane) is grounded because structures in the nervous system are not electrically isolated from the rest, effectively grounding everything to the same reference.

2.2.2 Synapses

There are two important features that distinguishes electrical and chemical synapses: speed of transmission and effect on the postsynaptic neuron. With respect to the transmission speed, chemical synapses are slower than electrical synapses; depending on the type of chemical synapse, neurotransmitters can induce post-synaptic membrane responses (PSR) within milliseconds while electrical synapses are “nearly instantaneous” [20].

The reason for this difference comes from the biophysical mechanisms in place; in gap junctions the synaptic cleft is about 2-4 nanometers, allowing direct transmission of the ions that produce PSR; while in chemical synapses, synaptic vesicles are stimulated by action potentials and release neurotransmitters that traverse a 20-40 nanometer gap [21].

The effects of the different synapses can be classified in three categories [20]:

- Electrical synapses are two-way connections that synchronize neurons (also regulate inner cell processes);
- Chemical synapses are one-way electric connections and can be inhibitory or excitatory:
  - excitatory synapses stimulate the postsynaptic neuron to produce depolarizing PSR;
  - inhibitory synapses inhibit possible action potentials in the postsynaptic neuron by producing hyperpolarizing PSR.

Chemical synapses can modify the amplitude of the signal (synaptic strength) allowing mechanisms of cooperation/competition between neurons and, very importantly, allowing mechanisms of adaptation and learning, also know as plasticity.

2.2.3 Plasticity

Barring cellular death, neuronal activity is susceptible to homeostasis, where neurons and synapses are actively regulated. This regulation of activity is an umbrella term for a variety of biological mechanisms, that on the scale of neural networks, adjusts the firing rate to a target value.

Synaptic plasticity

For the context of this dissertation, synapses form an important basis for comparison and inspiration terms for some of the results that will be shown later. Like other structures in the nervous system, they are regulated by plasticity mechanisms, in this case, to adjust the effect on the post-synaptic neuron.

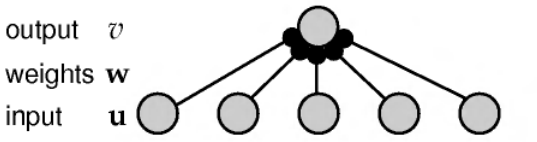


Figure 2.10: Simple model of neural network [12]

For the output (post-synaptic neuron) in figure 2.10, the differential equation is

$$C\frac{dv}{dt} = [\text{Model dependent current}] + f\left(v, \boldsymbol{u}, \mathbf{w}\right),$$

where v and $\boldsymbol{u}$ are the output and input activities and **w** is the synaptic weight of the connection, and it is the dynamic behavior of **w** that is believed to represent the formation of memories and learning [22, 23]. The function f (an unspecified current source) and the equation that rules the evolution of **w** reflect the fact different types of plasticity occur in the nervous system [24, 25, 26].

The simplest plasticity type is referred as Hebbian theory [12]:

$$\frac{d\mathbf{w}}{dt} = v\boldsymbol{\alpha} \cdot \boldsymbol{u}. \tag{2.5}$$

According to this theory, synchronous events (that is $v \cdot u_i \neq 0$) change the strength of the connection. The signal of $\boldsymbol{\alpha}$ determines whether the synapse is strengthened or weakened. If α_i is positive, long-term potentiation (LTP) increases the strength of the synapse; for negative α_i, long-term depression (LTD) decreases the strength of the synapse. However, synapses do not have an infinite retention time, that in this perspective, means that the value of **w** is not constant, even if $v \cdot \boldsymbol{u} = 0$. Adding an exponential term, $-\boldsymbol{\beta} \cdot \mathbf{w}$, to the right side of Eq.(2.5) simulates volatility.

Additive/Subtractive learning allows for LTD and LTP within the same model:

$$\frac{d\mathbf{w}}{dt} = \boldsymbol{\alpha}\left(v - \mathbf{u}\right). \tag{2.6}$$

The relative difference of electrical activity also influences the synaptic process. This is an important learning mechanism, because in electrical systems, an adaptive element takes into consideration potential difference between its terminals, just like in Eq.(2.6).

More complex models take into consideration timing between spikes, producing significant strength change only if within a certain time window. For example, experimental data (figure 2.11) shows spike timing dependent plasticity (STDP) within a window of 100 milliseconds, using rat hippocampal neurons [27].

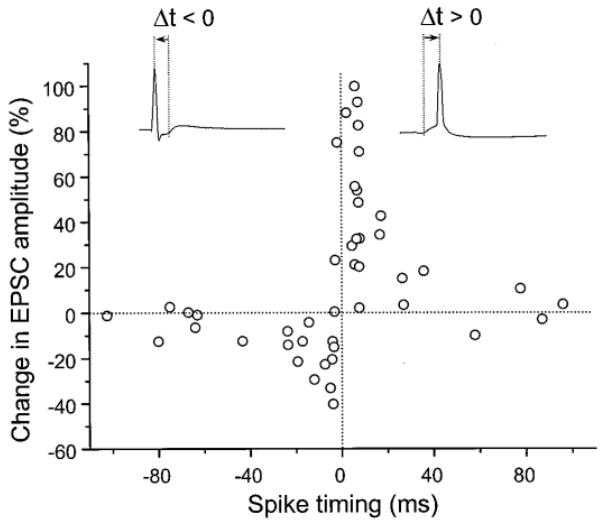


Figure 2.11: Synaptic strength typically increases if the post-synaptic spike occurs after the pre-synaptic spike, decreases the timing is reversed [27]

These different types of plasticity support different adaption mechanisms and in the context of synapses, it is how the neural network learns. If one of the input neuron fires and within a certain time window the output also fires the strength should change. This learning process changes the set of parameters in the neural network, but what matters are the two possible outcomes:

- if the synaptic strength increases, the synapse is **potentiated**;
- if the synaptic strength decreases, the synapse is **depressed**.

Hebbian learning refers to the simplest learning scheme and can be stated as “neurons that fire together, wire together” [12]. In a more realistically manner and using more complex models, neurons that fire within a specific time window, connect in a different way. And in a system where in average each neuron connects to other 10^2-10^3 neurons, the neural pathways that have been potentiated matter the most and depressed pathways are less relevant. These competition based mechanisms affects the development of neural circuitry [28] and after growth [29]; correlation-based or covariance rules enable to map these changes to simple changes in synaptic strength, useful for artificial neural networks [30, 31] but do not have enough biophysical details to exhibit true effects of competition/cooperation in biological systems [32].

Homeostatic plasticity

While synaptic plasticity can explain neuronal adaptation at short term and partially at long term, the nervous system has other mechanisms in place. Hebbian type mechanisms are not enough to explain plasticity because they tend to destabilize neuronal networks [33], saturating the synapses and deviating from target firing rates [34].

Global synaptic scaling adjusts the strength of all synapses for activity rates to meet some target firing rate [34]. This mechanism causes a multiplicative change (α) in synaptic weights: at reduced activity, $\alpha > 1$ and with increased activity, $\alpha < 1$. Structural synaptic plasticity refers to the formation and elimination of synapses, changing the structural models of networks in the visual and motor systems [35, 36].

Homeostatic plasticity plays a relevant role on neuronal dynamics and should be mentioned to show that synapses are not the only necessary element, but it will not be discussed in this dissertation.

2.3 Memristors

Memristors or resistive switching devices are a novel concept in material science [37, 38, 39, 40], neuromorphic engineering [41, 42] and RAM (Random Access Memory) applications [43, 44, 45].

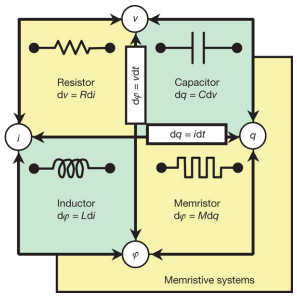


Figure 2.12: Graphical demonstration of memristor existence [37]

Memristors were theoretically introduced by Chua in 1971 [46], with figure 2.12 graphically completing the group of passive electrical elements with the memristor, by group theory arguments, and first experimentally recognized in 2008 on metal-insulator-metal devices [37]. It is a two-terminal electric component, similar to a resistor with a variable resistance that is controlled by one or more state variables, exemplified by figure 2.13 with two different models when subject to sinusoidal inputs. Since the first reference of memristors by Chua, the definition has been changed many times to accommodate for different experimental results and presently the more inclusive models are generic memristors [47]:

$$\begin{aligned} y(t) &= g(\mathbf{x}, u, t)\, u(t) \\ \frac{d\mathbf{x}}{dt} &= f(\mathbf{x}, u, t), \end{aligned} \tag{2.7}$$

where $y(t)$ and $u(t)$ are all the possible fundamental circuit variables (current, charge, voltage or flux), g the function that connects the two circuit variables, for example, with $g =$ resistance if $y =$ voltage and $u =$ current, and $\mathbf{x}$ a vector of internal state variables. In this general form, Eq.(2.7) describe many fundamental electronic elements: sources, resistors, inductors, capacitors...

To specifically represent memristors, extended models have two extra conditions:

$$\begin{aligned} y(t) &= g(\mathbf{x},\, u,\, t)\, u(t) \\ g(\mathbf{x},\, 0,\, t) &\neq \infty \\ \frac{d\mathbf{x}}{dt} &= f(\mathbf{x},\, u,\, t) \end{aligned} \tag{2.8}$$

and all hysteresis loops are pinched at the origin, $(y, u) = (0, 0)$.

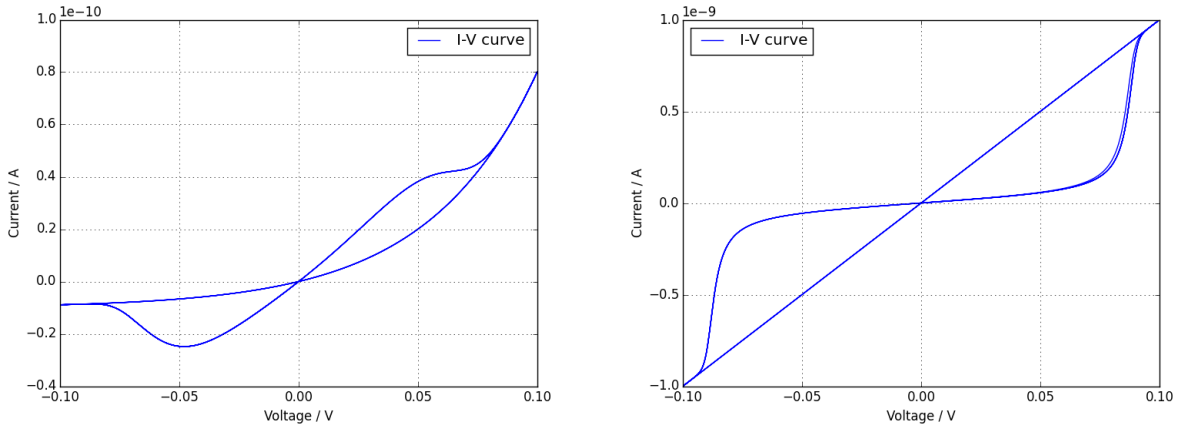


Figure 2.13: Examples of current-voltage curves of simple analytical memristive models

These devices are interesting because the set of Eqs. (2.2) and (2.7) share a few similarities, namely on the gating and state variable. Currently, the research is still centered on improving performance requirements [48] and due to limitations, there are three main areas of research on memristive devices: material science, resistive random access memory and neuromorphic applications.

Material science

Experimental results since 2008 suggest that the memristor has been found [37], but are still objections about the possibility to obtain the originally proposed device [49]. Those objections come partially from divergences between the original formulation and currently available devices, partially from electrodynamical principles and from fabrication issues that cause device variability [50], constraining the applications [51].

More recently, it has been discussed whether memristors can only exist as binary [52, 53] or multilevel/continuous devices [54, 55].

Resistive random access memory or RRAM

At a more commercial level, there are structures called crossbar arrays which store 0's or 1's (bits), acting as a memory/data storage [56, 48]. The interest lies on the reading and writing times of these structures, in the scale of nanoseconds [57], allowing to create faster memory devices.

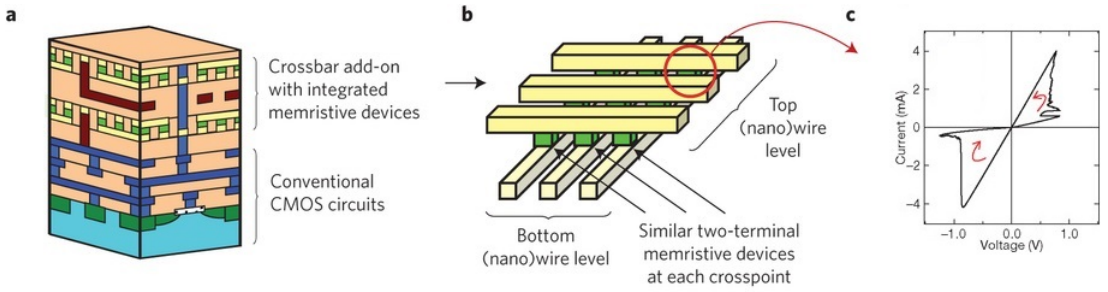


Figure 2.14: **a** - Cross section of crossbar memristor arrays; **b** - Schematical representation of the same; can be very dense; **c** - On and off state, each crossing point is a storage bit; adapted from [48] and [37]

Neumorphic applications

In this area there are many possibilities: computing that differs from the usual von Neuman architecture [56]; fuzzy logic in contrast to Boolean logic [58, 59]; hardware visual systems with fast detection times [60, 61], neural networks and neural computing [62, 63] (as in circuitry that mimics the highly parallel, highly redundant architecture of the nervous system) and neuro-inspired content-addressable memory systems [64].

Besides all of the above, memristors can be fabricated at nanometer scale [65] and very low power (hundreds of femto to tens of pJ [66] or 10^{-1} to 10^{-3} W during switching process), compared to its digital counterparts.

In another section it will show how ionic pumps and memristive devices are similar, focusing on the capacity to retain information and plasticity, with short term memory for ionic pumps and long term memory for memristors.

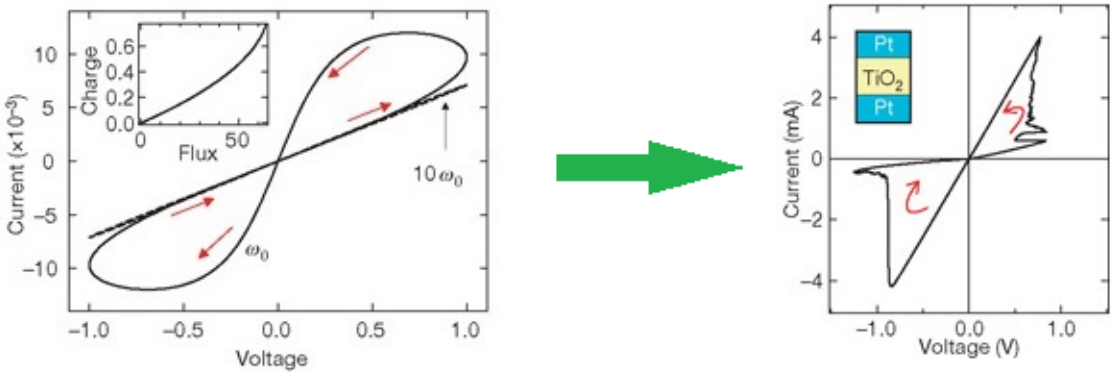


Figure 2.15: Comparison of theoretical model (left) with experimental data (right) [37]

Correctly speaking, memristors were not developed experimentally in 2008, they were just not recognized as such until then [37]. Typically, memristive properties emerge from the electrical properties of stacks of materials [67, 68, 69, 70]: electrodes between some insulating material.

2.3.1 Memristive models

From a physical standpoint, a detailed model depends heavily on the type of device, making the development of generic models difficult. However, for circuit applications there are simplified models that exhibit resistance switching characteristics, without considering the finer details (see figure 2.15).

For RRAM applications, because resistive switching devices are used as a storage bit, the models for these memories do not need to take into account all physical details and primarily emphasize the binary values of resistance.

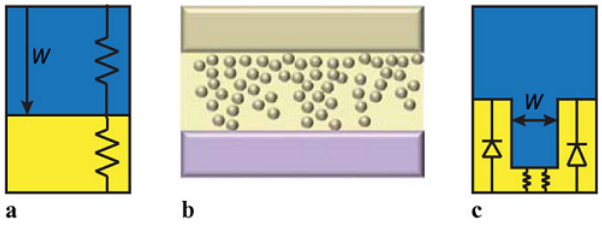


Figure 2.16: Possible physical representation of resistance switching [71]

In figure 2.16, model **a** represents a moving wall between the blue region (low resistance volume) and the yellow region (high resistance), with the following state equations:

$$\begin{array}{rcl} V(t) & = & M(x,I,t)I(t) \\ M(x,I,t) & = & R_{ON}x + R_{OFF}(1-x) \\ \frac{dx}{dt} & = & a \cdot f\left(x\right) \cdot I(t) \end{array} \tag{2.9}$$

Eqs. (2.9) represent one of the simpler models of resistance switching, with the linear transformation from R_{on} to R_{off}. This is referred as the linear memristor, acting as variable resistance or memristance (M) and relating voltage (V) and current (I). The state variable (x) linearly controls the memristance between a high (R_{off}) and low (R_{on}) value. The entire dynamical information is in the window function: $f\left(x\right) = 1$ for the originally proposed device, $f\left(x\right)$ some function for simple models (figure 2.17) and a given parametrized function for real devices.

Model	Current–voltage relation	State variable derivative
Linear ion drift	$v(t)=\left(R_{on}\frac{w(t)}{D}+R_{off}(1-\frac{w(t)}{D})\right)i(t)$	$\frac{dw(t)}{dt}=\frac{\mu_v R_{on}}{D}i(t)$
Nonlinear ion drift	$i(t) = w^n(t)\beta sinh(\alpha v(t))+\chi[exp(\gamma v(t))-1]$	$\frac{dw(t)}{dt}=av^m(t)f(w)$
Simmons tunneling barrier	$i(t) = \tilde{A}(x,v_g)\phi_1(v_g,x)\times exp\left(-B(v_g,x)\cdot\phi_1^{0.5}(v_g,x)\right)-\tilde{A}(x,v_g)\left(\phi_1(v_g,x)+e\lvert v_g\rvert\right)\times exp(B(v_g,x)(\phi_1(v_g,x)+ev_g)^{0.5})$ $v_g = v-i(t)R_s$	$\frac{dx(t)}{dt}=$ $\begin{cases} c_{off}sinh(\frac{i}{i_{off}})exp[-exp(\frac{x-a_{off}}{w_c}-\frac{\lvert i\rvert}{b})-\frac{x}{w_c}] & i>0 \\ c_{on}sinh(\frac{i}{i_{on}})exp[-exp(\frac{x-a_{on}}{w_c}-\frac{\lvert i\rvert}{b})-\frac{x}{w_c}] & i<0 \end{cases}$
TEAM	$v(t)=[R_{on}+\frac{R_{OFF}-R_{ON}}{x_{off}-x_{on}}(x-x_{on})].i(t)$ or $v(t)=R_{ON}.exp(\frac{\lambda}{x_{off}-x_{on}}(x-x_{on})).i(t)$	$\frac{dx(t)}{dt}=\begin{cases} k_{off}(\frac{i(t)}{i_{off}}-1)^{\alpha_{off}}.f_{off}(x) & 0<i_{off}<i \\ 0 & i_{on}<i<i_{off} \\ k_{on}(\frac{i(t)}{i_{on}}-1)^{\alpha_{on}}.f_{on}(x) & i<i_{on}<0 \end{cases}$

Figure 2.17: Examples of memristor models with the same moving wall logic [72]

To correctly parametrize memristive devices, more physically detailed models are required. Experimental results indicate that most memristive devices exhibit resistance switching with ion migration, forming filaments of material with different conductivity [48, 73, 74]; less common results also identify interfacial and bulk transitions in certain devices [75, 76].

Work developed on filamentary memristor models [77, 78, 79, 80, 81] relies on ion migration considerations (figure 2.18), having a more accurate representation on schemes **b** and **c**.

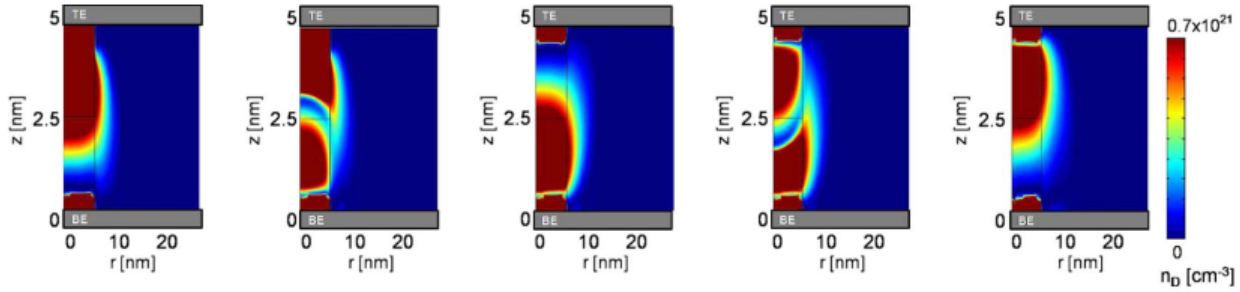


Figure 2.18: Ion migration between the electrodes (TE and BE) when applied differently signed voltages [77]

As is shown further, the plasticity of memristors is very sensitive to the models that are being used; fine tuning and sensitivity makes the study of possible applications difficult. In that perspective, a temporary shift from voltage-current models is necessary, which are easy to implement in circuit based simulations (ie SPICE), to flux-charge ($\varphi - q$) models. These models are the equivalent of voltage-current models in the integral form, but have a simpler form. The following model shows exemplifies the differences:

$$\begin{aligned} V &= MI \\ M &= R_{on}x + R_{off}(1-x) \\ \dot{x} &= \alpha I \end{aligned}$$

Not including a window function simplifies the math, and integrating the $V - I$ state equation, the model results in:

$$\begin{aligned} \varphi &= \int V\, dt \\ q &= \int I\, dt \\ \varphi &= R_{off} q + \frac{R_{on} - R_{off}}{2} \alpha q^2 + \varphi_0 \end{aligned}$$

But when considering window functions or more complex $V - I$ models, it is difficult to present the model on an analytic closed form. Even though the linear model (without window functions) is represented by an injective relationship in the $\varphi - q$ plane, realistic devices generally cannot. However, in this plane, it is easier to match experimental data to a numerical function or table.

2.4 Neuromorphic applications

So far, it has been presented the nervous system and memristors with separate mathematical representations, briefly touching upon the similarities. Namely, in terms of formalism, Eqs. (2.1) and (2.2), except capacitive effects, fall under the type described by Eq. (2.7). However, because real devices do not spontaneously emit spikes without the use of other electrical elements, the generic memristor is not used, alone, to describe neurons. It is nevertheless, an extraordinary neuromorphic match for voltage-gated ion channels and synapses.

The similarities are revealed by the electrical equivalence of a neuronal excitable membrane, because it introduces variable resistors controlled by membrane potential's history. The gating and state variables, of neurons and memristors, are the clear parallel between these two systems.

2.4.1 Memristive plasticity

It is useful to now focus on how memristors imitate biological properties. While gating and state variables have similar mathematical models, state variables and synaptic weights are akin in terms of plasticity.

Simulation results takes into account a simple moving wall model and input different signals (spikes) at both terminals with a time difference ΔT [82]. In figure 2.19 it is clear that memristors allow for distinct plasticity rules, like the nervous system also has [26]. The main point that needs to be stressed is that, in the nervous system, different plasticity rules are tied to different synaptic models, that rely on different biophysical foundations, while the signal shape is not reliable enough to select one of the rules. I will show later that this causes a fine tuning issue that constraints the intended applications.

2.4.2 Artificial cells

Using computational and numerical tools it is possible to demonstrate that memristive devices can manifest adaption mechanisms that mimic biological processes. Neuromorphic engineering

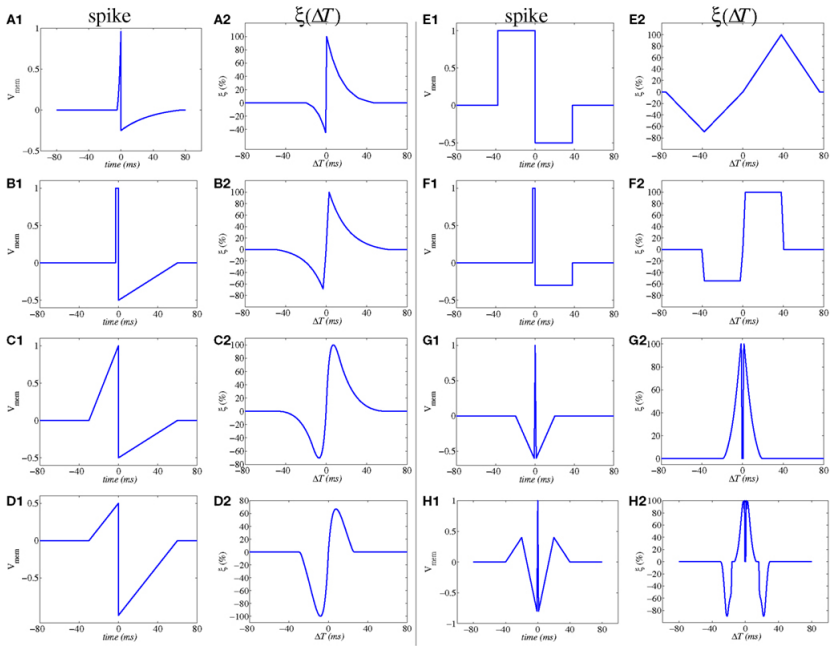


Figure 2.19: $\xi\,(\Delta T)$ is a measure of state/resistance variability after the pre- and postsynaptic signal [82]

is an umbrella term for the intersection of several areas of research, including biology, electrical engineering (with focus on electronics), chemistry and others. This area results in designs that mimic the nervous system and neural networks, whether it being circuits that model an neuron (IF [83], HH [84] or others), whole subsystems of the nervous system, like the visual or auditory system [85] and eventually a brain on a chip.

Starting from simple building blocks, resistance switching devices provide a basis for IF neurons, because both integrate the input that they receive. For example, a neuristor [86] is a memristor based artificial neuron [84] that mimics the dynamics of a HH neuron.

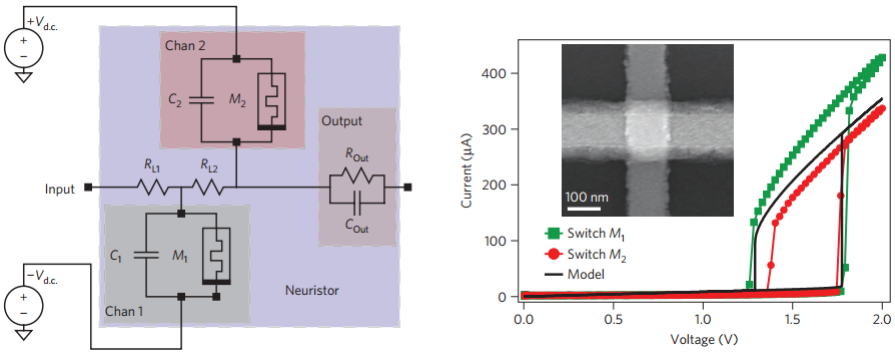


Figure 2.20: **Left** - memristor based neuristor; **Right** - Voltage-current curves of the memristors [84]

Notice on figure 2.20 that this circuit, excluding the capacitors (that introduce latency), is a variation of a memristive voltage divider. While $V_{d.c.}$ is a constant input (after being turned on) to set the state of each memristor, the input is characterized as a solitary square wave that destabilizes both

switching devices for a limited time window, creating a spike profile similar to an action potential.

Other cells can be copied, like sensory cells, to mimic adaption to repeated input and for example, amoeba-like cells can adapt to periodic inputs, implying that these cells have memory [87]. A circuit involving a resistor, inductor, capacitor and memristor exhibits the same type of memory [88], where the inductor and capacitor regulate the total change the memristor's state.

2.5 Cellular neural networks

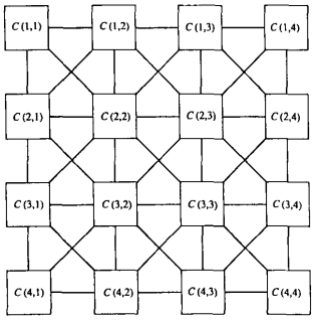


Figure 2.21: A 4 by 4 two dimensional cellular neural network, with $C(i,j)$ acting as cells connect with all its neighbors [89]

On a larger scope, biological neural network and electrical circuits can be tied together in the context of cellular neural network [89, 90], a type of artificial neural networks. Introduced by Chua and Yang in 1988, it is a large scale analog circuit composed of dynamical systems connected locally and can be used for image processing/analysis, like edge or pattern detection.

The hybrid circuits that are introduced in this dissertation are a type of cellular neural networks, exemplified by figure 2.21, with the cells modeled as realistic HH neurons and the connections as memristive devices.

Chapter 3

Numerical methods

Contents

Due to their complexity, bioelectrical systems cannot be addressed analytically in general, and due to current experimental limitations it is hard to set up long term hybrid circuits. Numerical simulations provide a good middle ground to low cost *in silico* neuron/memristor systems, but biologically detailed neuron models can still be too numerically taxing and to compensate for that, a specialized simulation environment called NEURON is used, to simulate biologically detailed neurons and neural networks.

To construct such models the user performs two different jobs at once: modeler and programmer. The first is obvious, because you need to parametrize the system by some set of dynamical variables or decide which models are more adequate for the intended applications. As a programmer, you need to go from the mathematical model to the numeric model and more often than not, it requires simplifying and/or modifying equations so that they are numerically stable.

3.1 NEURON

From a physics and engineering view, the most intuitive way to understand the nervous system is to start with neurons, that can be modeled in a simple way but still exhibit some complex dynamics. However, detailed analytical models are hard to obtain in neuroscience, meaning that simulations provide the best solution.

NEURON is a simulation environment that incorporates some C derived languages, with C's speed but allowing to be written in a higher level syntax; it is a simulation tool to model individual

neurons and networks of neurons. NEURON is a well established tool in the field of in silico/computational neuroscience, have been used in hundreds of scientific publications, and is the core simulation engine of the Human Brain Project (HPB). The best way to understand how it works is by modeling a simple cell.

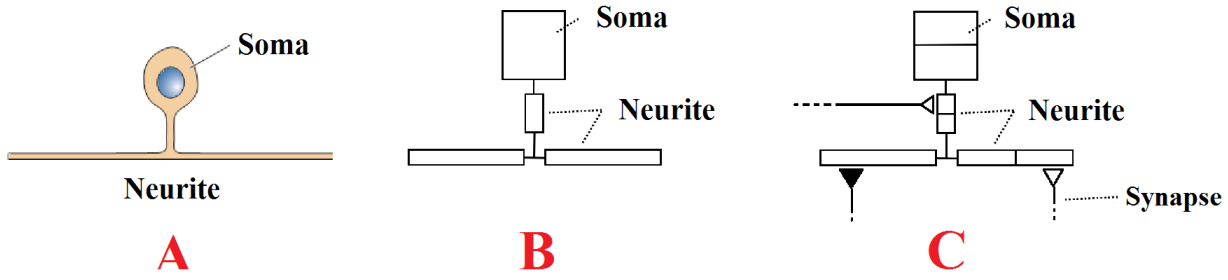


Figure 3.1: From “real” neuron ([11], p.48) to computational model

Like other computational methods, modeling a system as a set of numerical equations and conditions requires special care, with the modeling often being a collection of guidelines. Figure 3.1 shows the typical progression from realistic systems to computational systems, when considering biological neural networks or individual neurons. Scheme **A** represents an unipolar neuron. From **A** to **B**, the user chooses how accurate the model needs to be. The simplest model needs only one compartment (one box) representing the soma (or cellular body), some equation dictates when and how it spikes. Scheme **B** is a multi compartment model and the neurite (projection from the cellular body) is separated from the soma, with each box having an indexed set of equations. Each individual does not need to have identical equations. From **B** to **C**, how small the compartments should be depends on what needs to be studied, but like any other simulation method, increasing the resolution can limit the numeric error, if the integration is robust, at expense of time and possible convergence issues. Finally, scheme **C** shows synapses (the triangles) that are added to the model, in order to integrate the cell in a network.

Besides modeling cells, NEURON allows user-defined compartments, point compartments (e.g. memristors) and has some basic circuit simulation capabilities.

Due to its specific purpose, NEURON is not written/executed like generic programming languages. Designed to test models, most of its basic numeric tools are hidden from the end user, like the integration methods and model projection to matrix form; in a basic program in NEURON, the user only setups the model’s equations and possible interactions, letting the simulation environment take care of linking all equations and their dynamics. Or in another perspective, the programmer’s job is separated from the modeler’s job and maintained by the numerical engine. NEURON is also two languages at the same time: NMODL (Neuron MOdel Description Language) refers to compiled .mod files describing model’s equations, and like C it has a somewhat rigid syntax; HOC (High Order Calculator) is an interpreted language and a more fluid syntax. In addition, there is a graphical interface (GUI) that allows to construct simple models without writing code.

3.2 Building the system of equations

So far there has been a focus on the electrical aspects, that requires continuous connections between all elements. As a biological system, the nervous system can also be simulated with event-based methods, where each node can advance independently in some global (or independent) time, and only synchronize when they interact; this paradigm comes from how chemical synapses can be modeled.

Simple models describe synapses as a current source (same as an ion channel) with a time varying weight:

$$\frac{dV_i}{dt} = [\text{Current sources}] + \sum_j w_{ij} \left(E_i^{syn} - V_i\right),$$

$$\frac{dw_{ij}}{dt} = \alpha_{ij}\delta(t - t_{ij}).$$

The subscripts i and j refer to the individual index of the neurons: V the membrane potential, E^{syn} the synaptic driving potential, w the synapse's conductivity, α is the synapse's strength and t_{ij} is the spike time from neuron j to neuron i. The previous portion shows that on the point of view of neuron i, other neurons are not relevant; in fact, apart from t_{ij} these neurons are isolated systems; chemical synapses connecting to neuron i only require the time (or temporal distribution) of spike from spike j. It should be noted that not all models of synapses contribute to this paradigm, but excluding gap junctions and more detailed models, neurons can be considered *mostly* independent.

What does this mean to the numerical model? If node i only depends on node i's history, then the line of the matrix reserved to node i is only occupied in the $i-th$ column. Considering only the voltage dynamics with chemical synapses, then the differential equation becomes: $\frac{d}{dt}\overrightarrow{V} = F\left[\overrightarrow{V}\right]$, where F is a completely diagonal matrix; the matrix of weights w remains to be determined at all times.

Even when considering gap junctions, certain applications do not destroy the matrix sparsity, because in many situations neural circuits can be considered feed-forward networks, meaning that there are no loops. This allows to construct the matrix from some generic point in the net and gap junctions are the other non-zero elements in the row. The following situations involving two neurons connected electrically shows the partial destruction of sparcity: $\overrightarrow{V} = [...V_i...V_j...]$ and conductivity $g_{ij} = g_{ji} = g$:

$$\begin{cases} \frac{dV_i}{dt} = g_{ij}\left(V_j - V_i\right) \\ \frac{dV_j}{dt} = g_{ji}\left(V_i - V_j\right) \end{cases} \Leftrightarrow \frac{d\overrightarrow{V}}{dt} = \begin{bmatrix} & & \cdots & & \\ \cdots & -g_{ij} & \cdots & g_{ij} & \cdots \\ & & \cdots & & \\ \cdots & g_{ji} & \cdots & -g_{ji} & \\ & & \cdots & & \end{bmatrix} \overrightarrow{V} \Leftrightarrow \frac{d\overrightarrow{V}}{dt} = -g\overrightarrow{V} + \begin{bmatrix} \text{off diagonal} \\ \text{elements} \end{bmatrix}.$$

Sparcity is important because matrix operations can be very efficient when most of the elements

are on or near the diagonal. The previous cases are the best case scenarios, with no or few gap junctions, where the model can be easily solved. As is presented later, user defined mechanism or processes can alter this generic construction.

3.3 Numerical integrators

With the numerical model built, the system can advance in time and, from an end user perspective, there are several possibilities: fixed global, adaptive global and adaptive local time step methods.

Fixed time step methods

Like the name implies, the integration occurs at predetermined instants, multiples of the time step (dt), without considering error propagation; while they are not unconditionally stable, some options can be enabled to stagger the method and increase accuracy. In that case, this Crank-Nicolson method staggers the time step so that voltage is solved at $t + dt$, ionic concentrations at $t + \frac{dt}{2}$ and ionic currents at $t - \frac{dt}{2}$. It is suitable for individual neurons or small networks.

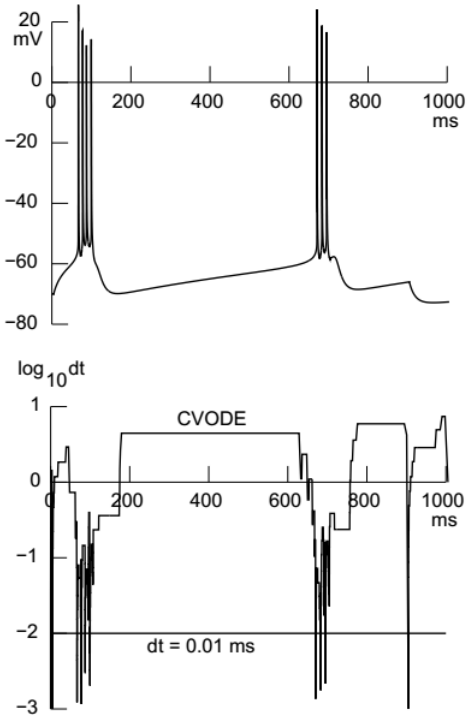


Figure 3.2: To stay within tolerance, the step size only decreases around voltage spikes.

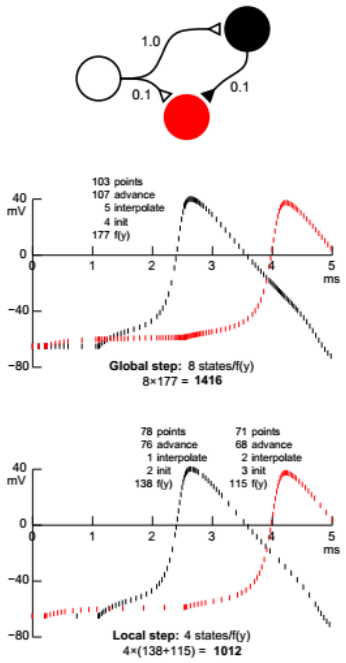


Figure 3.3: On the top, the white neuron delivers events to the red and black neuron.

Adaptive methods

In this case, the user does not control the time step, but the tolerance criteria (figure 3.2); the typical dynamics of neurons (idle time, interspersed with intense activity periods) does not need to be fitted to an arbitrarily small step. There are two possible modes, global and local simulation time; both modes use variable time step Backwards Differentiation Formula methods.

When NEURON is set to global time step, the entire network/cell setup has a global simulation time. When the derivative of variables increases, typically at action potentials, the time step decreases enough to stay within tolerance. But in a big enough network, there will be few periods of time without action potentials, meaning that the time step needs to be always small. With local time step, it is important to go back to section 3.2 where it is explained that neurons are *mostly* independent, because each cell do not directly interact at all times, meaning that they can be advanced in time at different time rates, and only synchronized at interaction times.

In figure 3.3, the event based process becomes clearer. Only at the height of action potentials (or spike time) both neurons really interact. Furthermore, because they do not interface electrically and the shape of the spike does not change when receiving another event. In this case, the interaction happens from the black to the red neuron, at the spike times and if they are both outside their refractory periods.

3.4 Standard run system

From the programmer point of view, most applications only require knowledge of the “standard run system”, a specific order of commands/procedures that the simulation environment executes in NEURON:

1. run()
2. stdinit()
3. init()
4. finitialize()
5. continuerun() *or* steprun()
6. step()
7. advance()
8. fadvance()

When the user types in HOC run(), many automatic processes occur. From steps 2 to 4 all variables, sections or pointers are being initialized to their default or user defined values, where step 3 (init) has a simple enough structure to allow user additions; these are executed once per run. Steps 5

to 8 are executed at each time step, where step 7 (advance) also has a simple structure for user manipulation. Steps 3 and 7 are implemented as programming hooks.

3.5 NMODL

NMODL is an object-oriented modeling language that standardizes model construction to be translated to computer readable code, in this case C code. Each object is a self contained module separated into specific blocks: declaring variables, setting up equations, etc. Abstracting the process from modeling concerns, an NMODL object workflow will typically follow the example given by figure 3.4. NMODL is used to extend the repertoire of mechanisms available in NEURON. It was therefore the modeling language used to encode the memristor dynamics which were then used in circuits of HH neurons.

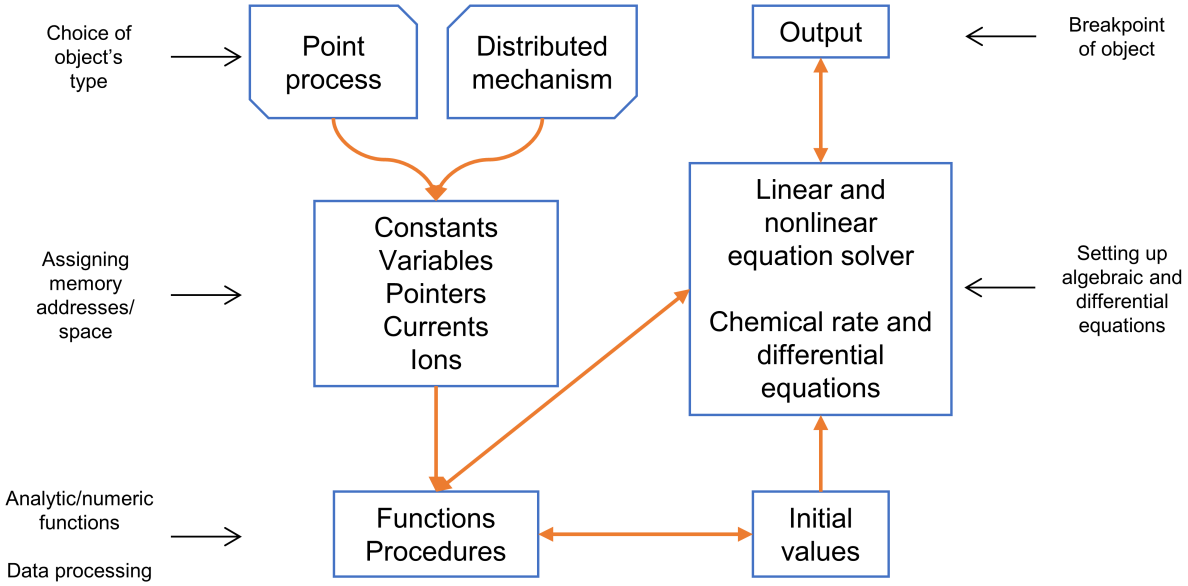


Figure 3.4: Typical workflow on a NEURON module

Chapter 4

Results

Contents

4.1 Preamble to results

Before reviewing results on hybrid circuits, details that are useful in this context, not usually explained in the literature, are examined.

4.1.1 Biologically detailed voltage spikes

The archetypal voltage spike has a duration of a few milliseconds and depolarizes the membrane. In figure 4.1, the spike is assumed to be restricted to a duration of 2-3 milliseconds, an amplitude of $\sim$ 100 mV, almost symmetric rising and falling phase and a undershoot period of about 1 millisecond. Using NEURON, figure 4.2 is the action potential generated in a single cylindrical compartment of 20 μm diameter and height with Hodgkin-Huxley properties. Unless otherwise stated, the parameters used to model the HH neuron (including compartment size), for the rest of

this dissertation, are:

$$\begin{aligned} E_L &= -54.3\,mV & g_L &= 0.3\,mS\,cm^{-2} \\ E_K &= -77\,mV & \overline{g_K} &= 36\,mS\,cm^{-2} \\ E_{Na} &= 50\,mV & \overline{g_{Na}} &= 12\,mS\,cm^{-2} \end{aligned}$$

$$C_m = 1\,\mu F\,cm^{-2}$$

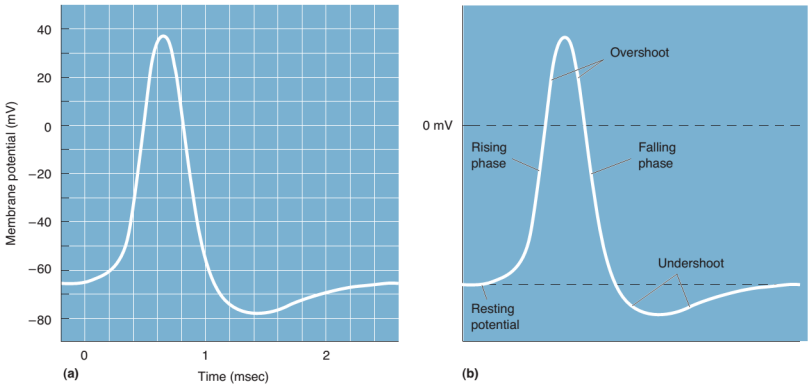


Figure 4.1: Basic features of a biological voltage spike ([11], p.84)

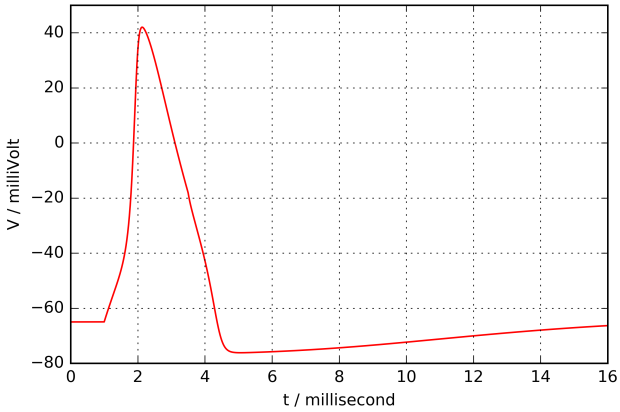


Figure 4.2: Hodgkin-Huxley action potential

While the refractory period is a fixed membrane property, the undershoot period depends on the injected current and the rising/falling phase are asymmetric, and contributes for some nontrivial effects on memristive plasticity with well synchronized spikes.

4.1.2 Neuronal current range

Another important feature of biologically detailed neurons is the range of currents that can be applied. In the IF model the input is integrated, meaning that the higher the current, the larger the

firing rate; while in more realistic HH neurons the firing rate has an upper ceiling. Increasing the current causes the neuron's voltage to stabilize at a value different from the resting potential, losing its active properties. With a neuron with similar properties as the one used in the previous section, figures 4.3 and 4.4 display under and over saturation, as a function of input current.

Barring transient behavior at points of discontinuity of current, the membrane voltage tends to stabilize around some value if the injected current is outside the cell activation range.

4.1.3 Inhibitory/excitatory synapses

It has been previously stated that chemical synapses are one way connections (page 12), stimulating or inhibiting the post-synaptic neuron. Previously, they have been presented as generic current sources, but for this dissertation purpose, can be modeled as current sources that drive the membrane potential to a given value; furthermore, to simplify this model, spikes are detected in the instant at which voltage crosses 0 from negative to positive values.

The IF model with such synapses results in:

$$\frac{dV_i}{dt} = \frac{E_i^L - V_i + V_i^{input} + \sum_j w_{ij}\left(E_i^{syn} - V_i\right)}{\tau_i} \tag{4.1}$$

and

$$\frac{dw_{ij}}{dt} = \alpha_{ij}\delta(t - t_{ij}) \tag{4.2}$$

In Eqs. (4.1) and (4.2), a single subscript refers to the index of the neuron: V is the voltage, E^L is the reversal potential, $\tau = R_m C_m$, with R_m is the resistance and C_m the capacitance of the membrane, V^{input} is the voltage induced by the electrode and E^{syn} the reversal potential of the synapse. The interaction occurs through the matrix w, representing the strength of connection between neurons i and j, α is a fixed matrix and t_{ij} represents the spike time between the *i*-th and *j*-th neuron. In this formalism, w is not necessarily symmetric or anti-symmetric, imposing in this model a behavior typical of chemical synapses. In contrast, the equivalent matrix for gap junctions needs to obey $w = -w^T$.

From Eq. (4.2) both neurons interact from the action of t_{ij}, but neuron A is inhibited only if neuron B is active; this means that an inhibitory synapse is useless by itself. Homeostatic plasticity mechanisms should induce inactive neurons to activity, but in this neural network level and inhibitory/excitatory perspective, chemical synapses need to be coupled with direct stimulation mechanisms and/or reciprocal synapses (direct or indirectly).

What classifies a synapse as excitatory or inhibitory, at this complexity level, is the synaptic reversal potential:

- $E^{syn} > V_{reset} \longrightarrow$ inhibitory synapse;
- $E^{syn} < V_{reset} \longrightarrow$ excitatory synapse.

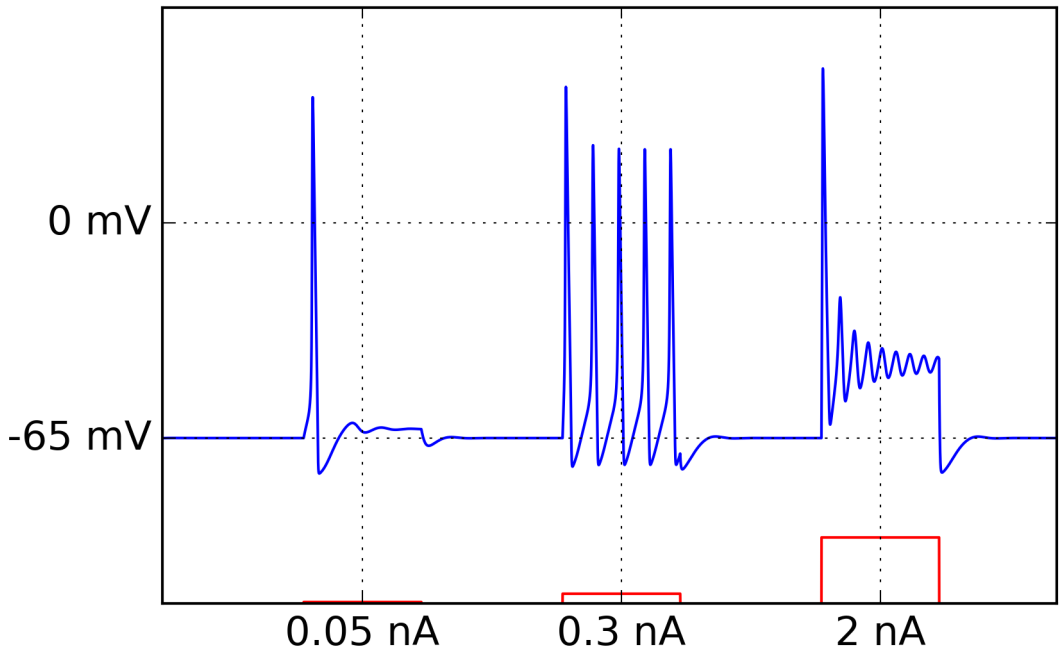


Figure 4.3: Under and over saturation of a HH neurons, simulated in NEURON

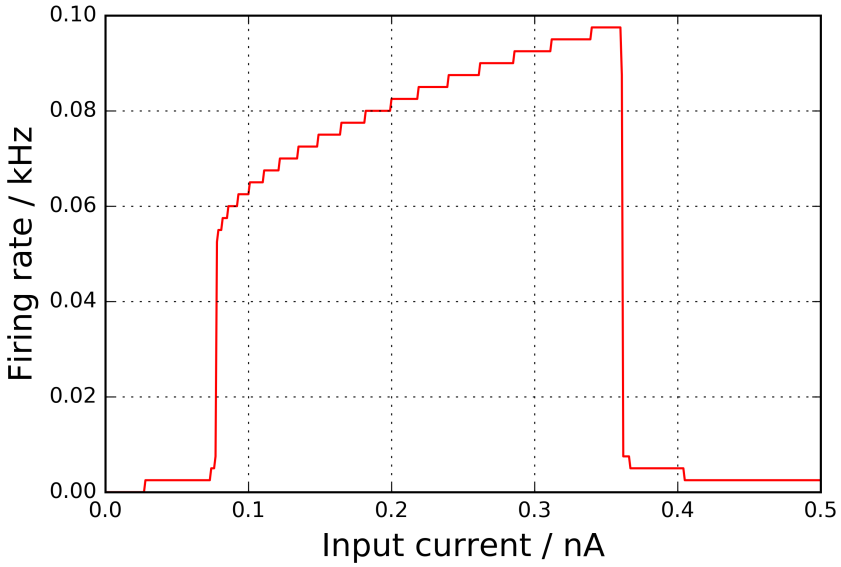


Figure 4.4: Saturation is reached at ~0.4 nA (step like behavior comes from analyzing discrete events on a limited continuous domain, in this case, time)

In this setup, there are two IF neurons (A and B). Neuron A is connected to neuron B with an inhibitory synapse, is stimulated with a given amplitude of V_i^{input} (in mV) and neuron B is connected to neuron A with an excitatory synapse.

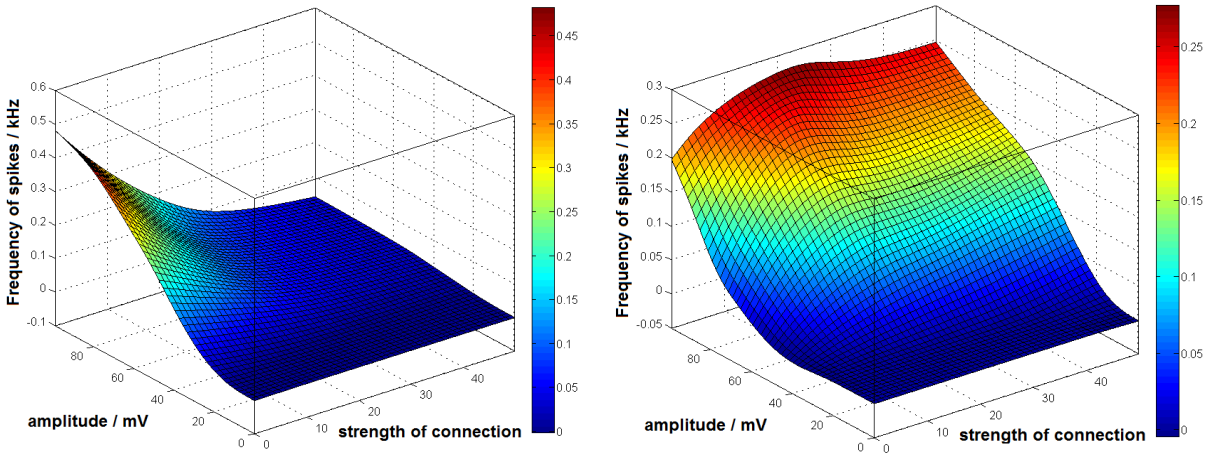


Figure 4.5: Neuron A on the left (inhibed) and neuron B on the right (excited); results are smoothed.

Figure (4.5) displays the need for coupled sets of inhibitory and excitatory synapses. For null strength of connection, akin to no synapses, neuron B is never active (smoothing of numerical data hides the behavior near the axis) meaning that neuron A is not inhibited; while a stronger connection (greater α_{ij}) *transfers* the electrical activity (voltage spikes) from neuron A to B. Note that the total electrical activity of the pair is not completely inhibited, but reduced, with a firing rate about 0.5 kHz with no synapses and 0.25 kHz when $\alpha_{ij} \gg 0$.

4.1.4 Tightening of memristive curves

While current and voltages can be rescaled in hardware applications (at the cost of scalability), neurons and memristive devices should have time scales in the same magnitude. For these types of one variable models, the state dynamic is generally dictated by $\dot{x} = a \cdot f(x, V, I)$ and $|f(x, V, I)|_{max} \sim 1$, meaning that a determines how much should the device react to any given input. For periodic inputs, as a increases, the total state change per period decreases; and for non periodic inputs, decreases the elapsed time to saturation.

Under the a sinusoidal input, figure (4.6) exemplifies the dynamics of a linear (or linear ion drift) memristor with a Prodromakis window function [91] (one type of window function that is well behaved under NEURON numerical engine), with the following set of equations and parameters:

$$\begin{aligned} V &= [R_{on}x + R_{off}(1-x)]\, I \\ \frac{dx}{dt} &= \alpha \left[1 - \left((x-0.5)^2 + 0.75\right)^p\right] I \end{aligned}$$

and

$$\begin{aligned} R_{on} &= 40\,\Omega \\ R_{off} &= 500\,\Omega \\ \alpha &= 1\,A^{-1}s^{-1} \\ p &= 1 \end{aligned}$$

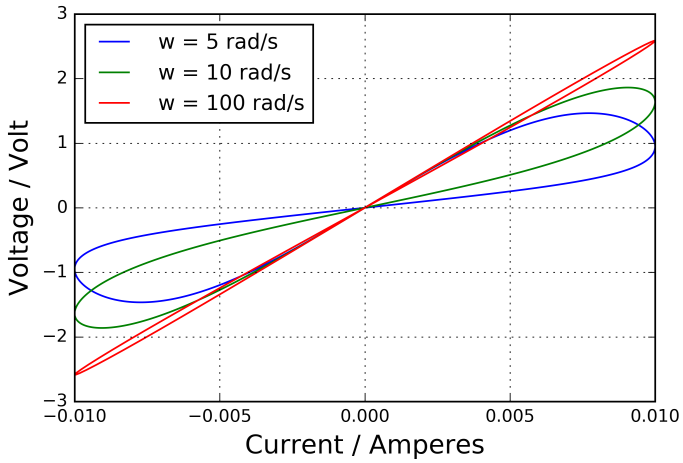


Figure 4.6: Tightening of hysteresis cycle at higher frequencies

4.1.5 Energy associated with neuronal transmission

Generally, voltage and current magnitudes of neuronal signals are insufficient to affect memristive devices and memristors' switching energy consumption can be on the order of hundreds of fJ [66] or lower. In the nervous system it is not as easy to measure the energy consumption of neurons, but it is possible to estimate the ATP cost per bit of information transmission:

- 1 bit of information transmission in a synapse costs $\sim 10^4$ ATP molecules [92, 93];
- the Gibbs free energy released in ATP hydrolysis is about 30 kJ / mol [94];
- 1 bit of information costs around 10^{-16} J to transmit.

Given that memristors can store at least a bit (high or low resistance), the ratio between bit transmission in synapses and bit storage in memristors is $r \lesssim \frac{10^{-13}}{10^{-16}} = 10^3$. For devices that require more energy, the ratio is even larger.

4.1.6 Voltage divider as spike generator

Taking inspiration from the concept of neuristor [84] and stripping the proposed circuit to its most basic ideas, what remains is a simple memristive voltage divider. In an attempt to introduce

non-linearities to the simple linear memristor and more specifically, non-linear effects into the state variable at the $I - V$ relation, the nonlinear ion drift model is used (see figure 2.17 on page 19).

The nonlinear ion drift model is described by the following equations:

$$I = x^n \beta \sinh(\alpha V) + \chi [\exp(\gamma V) - 1] \tag{4.3}$$

and

$$\frac{dx}{dt} = a \cdot f(x) \cdot V^m \tag{4.4}$$

Eq.(4.3) can be linearized to $V \to 0$:

$$\begin{aligned} I &\sim x^n \alpha\beta V + \gamma\chi V \\ &= (x^n \alpha\beta + \gamma\chi) V \\ &\equiv \left[x^n \left(R_{on}^{-1} - R_{off}^{-1}\right) + R_{off}^{-1}\right] V \\ &\equiv M^{-1} V \end{aligned}$$

While this altered model does not have the original asymmetric conductance, the exponent in the state variable (greater than 1 and integer) causes the device to be in the off state unless $x \to 1$, meaning that the resistance transition is not linear.

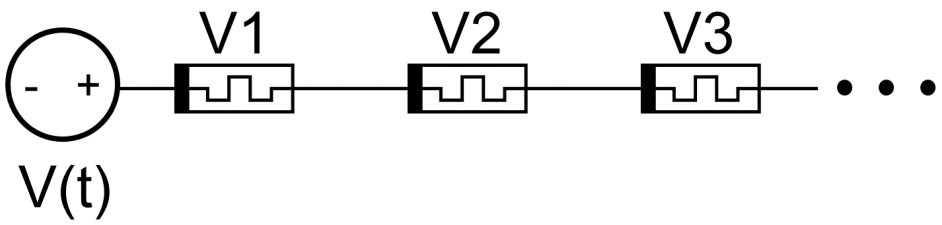


Figure 4.7: Generic memristive voltage divider circuit

Like the normal voltage divider, the voltage in each device is:

$$V_i = V(t) \frac{M_i^{-1}}{\sum_i M_i^{-1}} .$$

With the following parameters:

$$\begin{aligned} n &= 14 \\ R_{on}^{-1} &= 18\,\Omega^{-1} \\ R_{off}^{-1} &= 0.04\,\Omega^{-1} \\ m &= 1 \\ a &= 4\,V^{-1}s^{-1} \end{aligned}$$

and the Joglekar window function [39]: $f(x) = 1 - (2x - 1)^{2p}$ with $p = 1$ (another type of window function that is well behaved under NEURON numerical engine).

To generate biphasic spikes the voltage source needs to introduce sharp discontinuities, in this case a square wave, causing the sudden change in memristance; both memristors need to have the same polarity and the initial states/resistances tunes how distributed the spike is (figure (4.8)). For equal initial states, both memristors spike in an equal way and for $|\mathrm{x1}(t=0)\,\text{-}\,\mathrm{x2}(t=0)| \to 1$, the spike is concentrated in one of the devices.

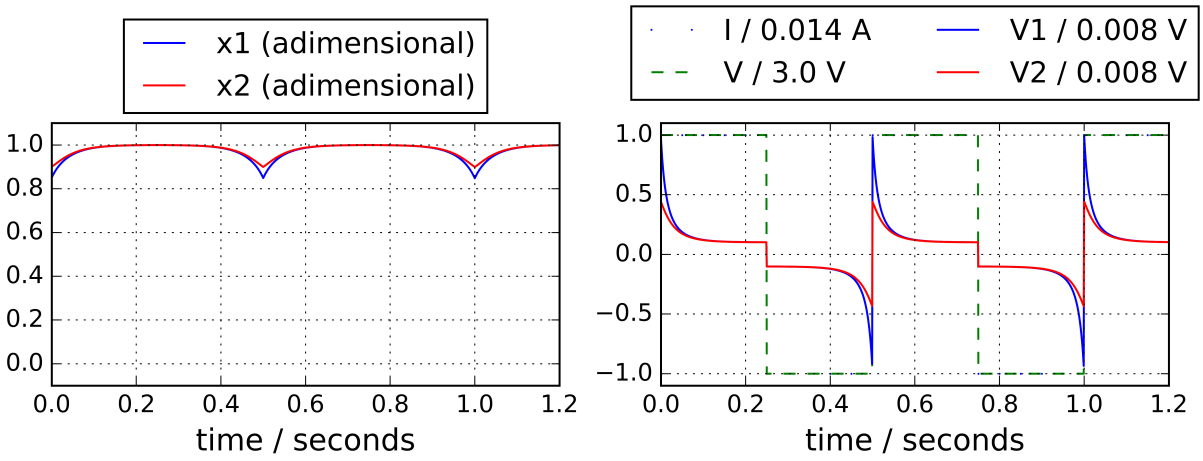


Figure 4.8: Dynamic of memristive voltage divider

Those spikes also occur for any series ensemble of N memristors, with the initial memristive state dictating the spread the spike amplitude on the devices. In particular, for $N = 3$ and 4 with $\Delta \mathrm{x} = 0.05$ between neighbors (with equal parameters), all the devices display biphasic spike with variable amplitude.

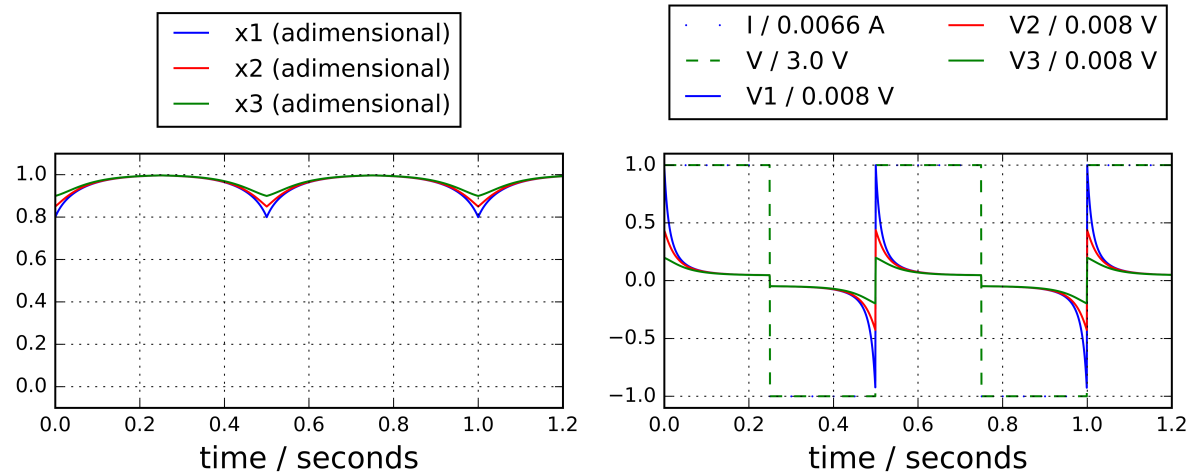


Figure 4.9: Biphasic spike generation for series ensemble for 3 memristors

Generalizing for any N, the devices need only be current controlled. Figure 4.9 and 4.10 exemplify the spike distribution of these generic configurations, where the initial memristance config-

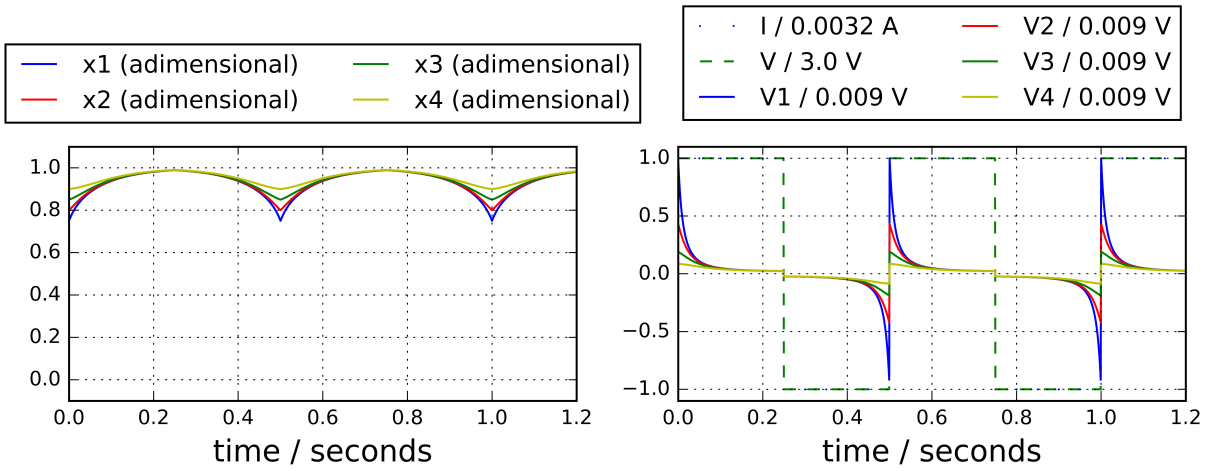


Figure 4.10: Biphasic spike generation for series ensemble for 4 memristors

uration rules the amplitude distribution[1].

4.2 Memristive synapses schemes

Towards the goal of designing, simulating and analyzing simple hybrid systems, different configurations of neuron-memristor circuits will be addressed.

one memristor connections

This is simplest and easiest configuration. Because of the current status of memristor fabrication techniques, variability and non reliability makes most of the discussed features impractical for mass applications.

multi memristor connections

Still within the context of connecting two neurons, ensembles of multiple devices can be reduced to simple equivalent devices; in linear models, for example, there are equivalence relations for composite memristive circuits: series $\rightarrow M_{eq} = \sum_{all\, i} M_i$; parallel $\rightarrow M_{eq}^{-1} = \sum_{all\, i} M_i^{-1}$. But there are ensembles that enhance or take advantage of certain features.

Because current research focuses on RRAM/crossbar arrays, memristive devices are mostly developed for fast transitions (nanoseconds), making them binary resistive devices in the context of neural networks, but multi level or continuous devices would replicate synaptic behavior more closely.

[1]The numerical results of the memristive voltage divider are obtained with a Python script, in page 81. These results can be verified on NEURON simulation environment, but due to time constraints, are left as part of future work.

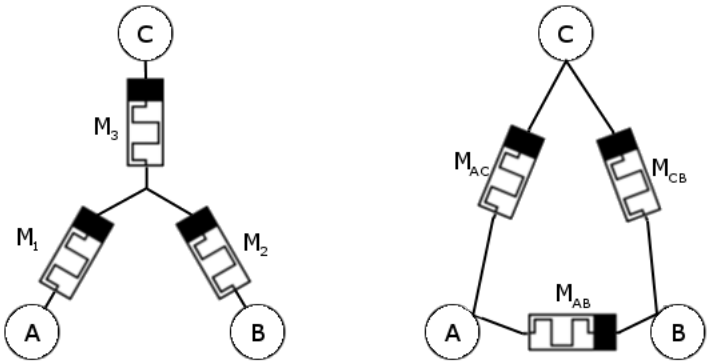


Figure 4.11: **Left** - Y configuration; **Right** - Equivalent Δ configuration

Taking inspiration from crossbar arrays (memory implementations) and focusing in lines or columns, the obvious simplification is series or parallel ensemble controlled by one input and one output neuron. Assuming that memristors can be this fine tuned in the a parameter (reaction strength) or in R_{on}/R_{off}, ensembles of series or parallel devices can be used to create multi-level resistance devices, from binary devices.

The series ensemble takes advantage of voltage divider effect, spreading the voltage drop unevenly: $V_i = V_{input}\frac{M_i}{\sum_{j=1}^{N} M_j}$, $i =$ index of all elements in the ensemble. For voltage controlled devices, if their switching characteristics and R_{on}/R_{off} are different enough then state transitions (high to low or vice-versa) do not occur simultaneously. And for the parallel ensemble: $I_i = \frac{V}{M_i}$; current controlled devices are ideal for this scheme. Just as crossbar arrays form $N \times M$ bit memories, these series/parallel lines form N/M bit memories, but more importantly produce an equivalent multi-level resistance device or a continuous device, in the limit of N or $M \to \infty$.

This is not necessarily a problem free scheme, because it does not solve the problem of voltage/current scaling and linearly increases the energy requirements of these ensembles. Other publications [95] suggest that anti-parallel and anti-serial arrangements screens the devices to small amplitude signals, requiring even larger gains to neuronal signals. But these connections schemes rely on the basis of device variability, which is compatible with current research standards.

multi memristor with multi neuron

Taking a further step on generalizing these artificial synapses schemes, M neurons can be connected with N devices. One possible arrangement are crossbar arrays, but to illustrate the problems in employing circuit analysis for hybrid circuits the connection of three neurons with three memristors is studied.

The left circuit can be analyzed with Kirchoff's circuit laws, given that memristors are variable resistors it is possible to use the Y $-$ $\triangle$ transformation to compute the resistance between the three pairs of neurons:

$$M_{AC} = \frac{M_1M_2+M_2M_3+M_3M_1}{M_2}$$
$$M_{AB} = \frac{M_1M_2+M_2M_3+M_3M_1}{M_3}$$
$$M_{BC} = \frac{M_1M_2+M_2M_3+M_3M_1}{M_1}.$$

But the dynamics of the three devices' state variables remain to be determined. While in this case it is still possible to infer the signal at all memristors, because the signal in the neuron A, B, C is equal to the signal at device 1, 2 and 3, inferring from combinations of the currents I_{AC}, I_{AB} and I_{BC}, but for more complex circuits this is not necessarily true. It is important to remember that hybrid circuits can not be easily analyzed by circuit laws and that each particular synaptic scheme require different algorithms/tools[2]. It is of note that, in general, neurons as circuit nodes do not conserve current, when considered as multi-compartment structures.

It is important to note that this Y $-$ $\triangle$ configuration is one of the simpler example of memristive crossbar arrays, with obvious applications for information storage but signal transmission guarantees the potentiation/depression of current paths in this setup.

one memristor/one other electrical element

- Inductor + Memristor

An inductor (L) in series with a memristor results in the following differential equation: $\frac{dI}{dt} = \frac{V(t)-MI}{L}$ + state dynamic equation, where $V(t)$ is the potential between terminals and $\frac{M}{L} = \tau$ a characteristic time scale of signal spreading; τ changes with signal input. Typical RL circuits are characterized by exponential decay, smoothing the current output, but in this context, it increases the causality window, with the increased overlap of pre- and post-synaptic signals.

- Diode + Memristor

The addition of a diode results in a memristive synapse that behaves like a chemical synapse, that is, it becomes a device where current only travels in one way.

- Source + Switch + Memristor

A problem of using memristors as a substitute of biological synapses is that electrical elements always transmit current from one neuron to another, exciting/inhibiting in pairs. Including diodes result in a device that resembles an excitatory synapse, but a purely inhibitory synapse is harder to achieve.

[2] In this case, using the same type of implementation explained in section 4.3, requires that at each time step of the simulation various transformations to occur. $\{M_1, M_2, M_3\}$ needs to be transformed to $\{M_{AC}, M_{AB}, M_{BC}\}$ in order to determine the currents $\{I_{AC}, I_{AB}, I_{BC}\}$, which can be used to determine the currents that are being applied to cells 1,2 and 3 and finally determine the changes in each of the state variables. But more importantly, generalizing this algorithm structure in NMODL code is impractical, because it requires the usage of several generic matricial transformations (not implemented in NMODL) with variable number of parameters, which would be written as C code, without special numerical libraries, on top of all NMODL memristive modules. This would vastly increase development time, even then it is unknown if the numerical engine could scale these types of hypothetical modules to a large network and would remain as an *ad hoc* implementation.

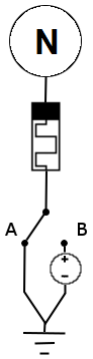


Figure 4.12: “Self” synapse

Figure 4.12 shows how to create a self-excitatory/inhibitory devices with voltage controlled switches. In both cases $I = {}^{V}/_{M}$ is small enough to not produce spikes, so that it can act as a driving force to saturate the memristor in its low or high resistance state, and the switch is on A or B, depending on some activity threshold.

1. The memristor is set up so that if connected to B it saturates in its low resistance state, in a self-excitatory device;

2. To create a self-inhibitory device, the memristor saturates in high resistance state when connected to B.

In case 1, when the neuron (N) is spiking from the action of other synapses (on A), the self-synapse transits to high resistance and becomes irrelevant to the dynamics; after a given period of time, the neuron stops spiking and the switch connects to B, decreasing M and inputting enough current to produce spikes. In case 2, when the neuron is above threshold it connects to A, decreasing M, creating the least resistance path for current and slowly inhibiting spikes.

4.3 Possible implementations of memristors in NEURON

A good implementation of memristors in NEURON would behave like any electrical element in circuit simulation environments, but there are several reasons why it cannot be done in a simple way; there are default equations and quantities that cannot be changed in the environment. For example, a section in NEURON created with the statement “create section”, has by default the IF equation and assigned variables that cannot be overwritten (like voltage), making any implementations of electrical elements harder.

Gap junctions (or electrical synapses) provide a guide to implement memristors. In NEURON, gap junctions can be implemented with point processes or linear mechanisms, both having advantages and disadvantages.

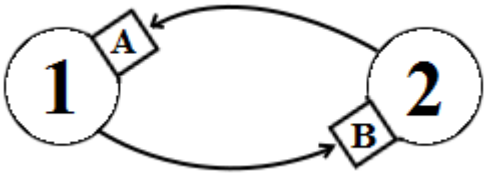


Figure 4.13: Double point processes (squares) implementation of gap junction connecting neurons (circles)

Point processes are not distributed in space and cannot affect sections other than the one it was placed (at least without C verbatim code); this simple implementation doubles the amount of calculations that need to be done, while causing convergence issues. It is very flexible, because it permits many types of calculations within the same block.

But this approach is not adequate because point processes can only be placed in sections, interfacing only with that section. In figure (4.13), point process A connects to neuron 2 with a pointer and point process B connects to neuron 1; if the elements are replaced by memristors all equations need to be repeated, including the *state differential equation*, with no guarantee that both equation evolve in the same way.

On the other hand, linear mechanisms correctly add elements to equation matrix, but only when it can be expressed in following way: $c\frac{dy}{dt} + gy = b$, y and b are vectors and c and g are matrices, and to describe a gap junction, g represents the conductance of the junction and y the voltage, with all other elements null.

$$(current\,balance\,equation) + \begin{bmatrix} G & -G \\ -G & G \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = 0$$

$$\Leftrightarrow$$

$$(current\,balance\,equation) + G \begin{bmatrix} V_1 - V_2 \\ V_2 - V_1 \end{bmatrix} = 0$$

As it as been mentioned by the developers [96], this is the proper way to add gap junctions to models; the problem is that it does not permit state dependent elements.

Because neither are particularly useful for the intended use, ideas are taken from both and used to create a mixed version.

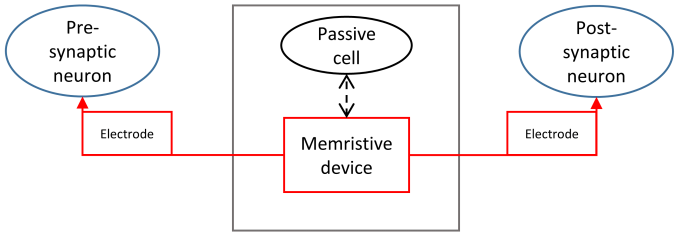


Figure 4.14: NEURON implementation of memristive devices

In figure 4.14, the box is more a convenience feature than a necessity, but it acts as an isolated object that boxes up a placeholder neuron and a point process that contains all memristive equations, anchoring it to the neuron. From limitations of the language all sections have passive properties and all point processes need to be associated to a section, therefore a “free” memristor that can be placed in the simulation environment is not possible; but using a passive cell, adding little computational complexity to the model, and *anchoring* a point process to it achieves the same purpose; the result from the equations is then pointed to perfect electrodes at either terminal of

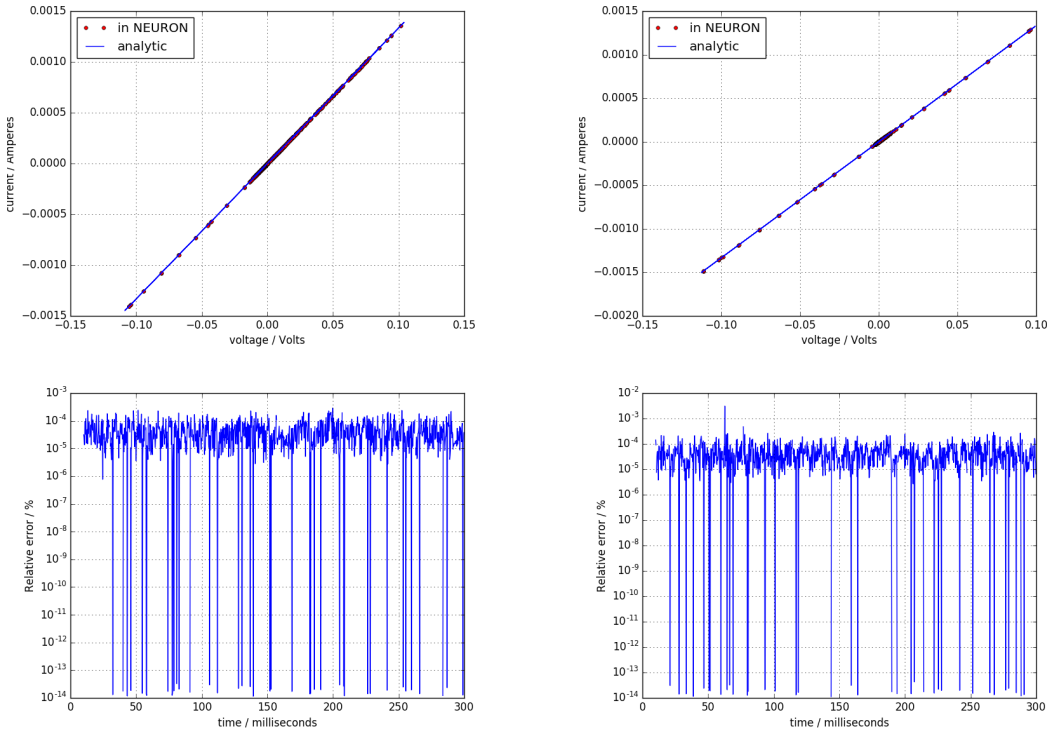


Figure 4.15: **Above -** Comparison between numerical and analytical current-voltage curves; **Below -** Relative error introduced in numerical algorithm

this object. The entire implementation is wrapped in a template, NEURON's version of objects (pseudo-code implementation on page 76).

With this implementation, the set objectives of the language are being tested (in this case NMODL), potentially adding unchecked error to the full equations. The following setup is done on NEURON: 3 HH neurons connected with flux-charge memristors, in a chain like manner, and one end of the chain stimulated; two cases are tested:

- $q(\varphi) = 0$

This case is trivial and NEURON outputs no current from this connection.

- $q(\varphi) = \alpha\varphi \Rightarrow \frac{d}{dt}[q(\varphi) = \alpha\varphi] \Rightarrow I = \alpha V$

In this case it reduces to a simple gap junction, providing a good testing ground.

To understand the importance of this result, the algorithm that is used in the .mod file needs to be explained. All calculations are done on a specific block called DERIVATIVE, which is useful to integrate variables but no block in NMODL can automatically compute derivatives. The steps in the algorithm are:

1. $\varphi = \int V(t)\, dt$, in NEURON expressed as $\varphi' = V$

- Because the DERIVATIVE block is specialized to make these calculations, the main part of the error should not come from here

2. $q = q(\varphi)$, can be a function or a linearly interpolated table

   - Ideally the assignment of charge can be done with a closed form expression, resulting in minimal error; otherwise the resolution of the numerical table needs to be adequately chosen, usually decreased to minimize error

3. Creation of four variables: t_0, $t_0 - dt$, $q(t_0)$ and $q(t_0 - dt)$

   (a) Each time the block is executed $t_0 =$ simulation time and $q(t_0) =$ charge at simulation time

   (b) With the purpose of using first order derivatives: $t_0 - dt = t_0$ and $q(t_0 - dt) = q(t_0)$, two queues of two elements

   (c) $I(t_0) = \frac{q(t_0) - q(t_0 - dt)}{(t_0) - (t_0 - dt)}$

   - Of all steps, first order approximations can be problematic, introducing first order error

In the second case, for both memristors in the model, the error remains within the same range 10^{-5} to 10^{-4}%; an error source that makes some results harder to validate. Let me analyze how HH equations propagates errors:

$$\frac{dV}{dt} \propto .... + I \text{ and if } I \to I + \Delta I \Rightarrow \frac{dV}{dt} \propto .... + I + \Delta I.$$

Ignoring the gating variables, integrating this equation results in $V(t) \propto ... + (I + \Delta I) \cdot t$, which means that with two identical HH neurons injected with I and $I + \Delta I$, the temporal mismatch between signals should increase linearly with time $(\Delta V = \Delta I \cdot t)$.

Figure 4.16 simulates the activity of two HH neurons, single section with 20x$20\,\mu m$; one of them is injected with 1 nA and the other with $\left(1 + 10^{-6}\right)$ nA. After some initial transient behavior, the relative error between voltages exemplifies the mismatch that increases linearly with time at spike times; the difference from the predicted result comes from the action of the gating variables. Each maximum is actually a double maximum, where the time difference between those maximum is a measure of spike time difference; after 27 spikes the mismatch is still sub millisecond, but the linear increase ensures that the mismatch becomes more relevant with the increase of simulation time. Just like there is an initial transient behavior, the undershoot is a transient behavior in the action potential (see figures (4.1) and (4.2)) that also causes an accumulation of error and after the period of stimulation produces a final spike of relative error, of lesser amplitude.

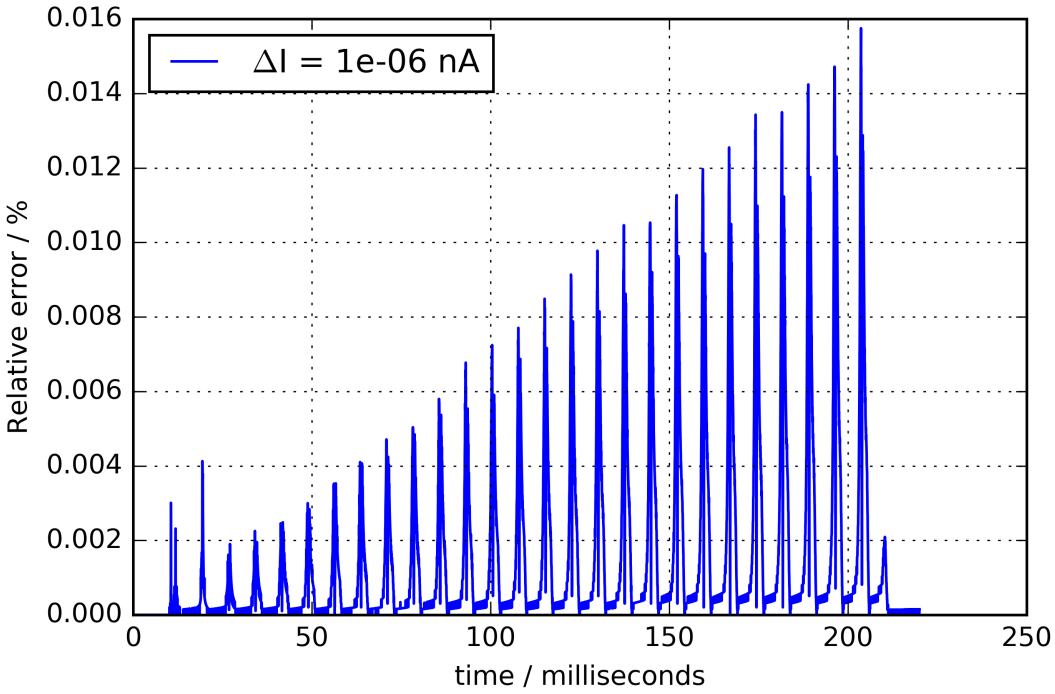


Figure 4.16: Local maxima point to mismatch between spikes

4.4 Memristive plasticity

According to the literature, synaptic plasticity of memristive devices can mimic biological plasticity. To study plasticity on these devices, the input signal is

$$f\left(t;\Delta T\right)=spike\left(t-\frac{\Delta T}{2}\right)-spike\left(t+\frac{\Delta T}{2}\right);$$

for a simple integration model with $\dot{x}\propto V\rightarrow x\propto\varphi_f+\varphi_0$, all possible values of ΔT the state variation is always null, because $\varphi_f=\int\left[spike\left(t-\frac{\Delta T}{2}\right)-spike\left(t+\frac{\Delta T}{2}\right)\right]dt=\varphi_\Delta-\varphi_{-\Delta}=\varphi_\Delta-\varphi_\Delta=0$; meaning that for the $V-I$ presented models the entire dynamical properties are on the window function. It needs to be stressed that these windows do not have physical significance and are parametrized to more closely resemble real devices. At software/hardware level the spike profile is tunable, but when considering biological neuron networks it is not an easily accessible parameter. For the intention of a simple estimation, spikes have 100 mV amplitude in a time window of 2-3 ms, therefore $\varphi_{spike}\sim 100\,mV\,ms=100\,\mu Wb$; a subsequent numerical integration of a typical signal results in $\varphi_{spike}\simeq 113\,\mu Wb$.

If the spike profile is not tunable, then the plasticity can only be changed if the memristor's model is changed; three cases are analyzed, all with $\mathrm{x}\left(t=0\right)=0.5$:

1. Control case: $\dot{\mathrm{x}}=a\cdot V$

2. Voltage controlled case: $\dot{\mathrm{x}} = a \cdot window\,(\mathrm{x}) \cdot V$

3. Linear ion drift case: $\dot{\mathrm{x}} = a \cdot window\,(\mathrm{x}) \cdot I = a \cdot window\,(\mathrm{x}) \cdot \frac{V}{\mathrm{x} + \frac{R_{off}}{R_{on}}(1-\mathrm{x})}$

Case 1

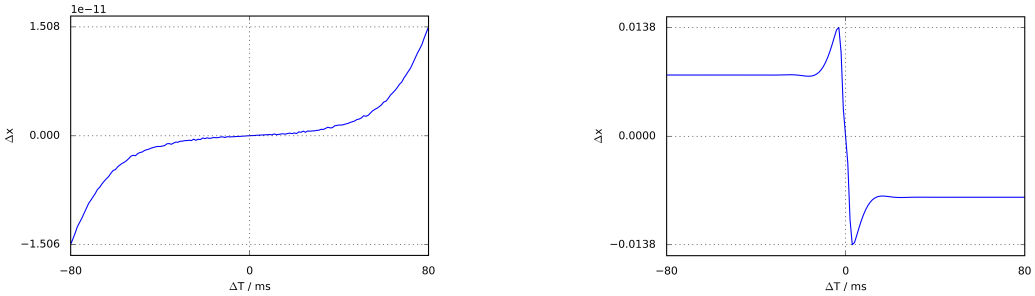


Figure 4.17: State change with $a = 5 \cdot 10^1\, V^{-1} s^{-1}$; for the control case on the left and the voltage controlled case on the right

In this simple model, a memristor is a voltage integrator, it is possible to obtain the analytical solution as $\mathrm{x}\,(t) = \varphi\,(t) + \varphi_0$; but for symmetry reasons for all $f\,(t; \Delta T)$ the state change should always be zero. As a numerical engine, NEURON cannot maintain null error and grows non-linearly (left side of figure (4.17)).

Case 2

The introduction of a symmetric window function (Jogleakar window function, centered on $\mathrm{x} = 0.5$) changes how the memristive devices detect the input, where the window function acts as a time variable reparameterization of a. While it exhibits STDP with a "causality window", even two opposing non causal spikes cause the device to change; this model scales not linearly with $|\Delta \mathrm{x}|_{max} \sim 2|\Delta \mathrm{x}\,(\Delta T \gg 0)\,|$.

With $a = 5 \cdot 10^1\, V^{-1} s^{-1}$, the maximum in the control case is $\sim 10^{-11}$, which gives an estimated error of the order of 10^{-7}%.

Case 3

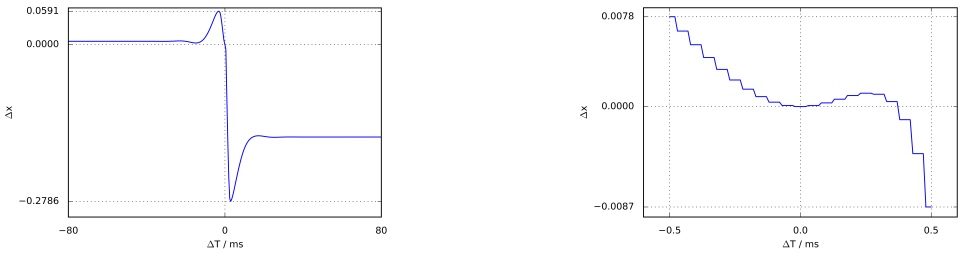


Figure 4.18: State change with $a = 5 \cdot 10^1\, V^{-1} s^{-1}$ and $\frac{R_{off}}{R_{on}} = 1 \cdot 10^1$; the graph on the right is a zoom in to $\Delta \mathrm{T} \to 0$ from the graph on the left

In this current controlled model, the linear ion drift memristor, introducing memristance into the dynamic breaks the time symmetry and generates a version of LTD (figure (4.18)). For this model,

a controls the symmetry around $\Delta T = 0$, with $\frac{d}{da}|\frac{\Delta x(-\Delta T)}{\Delta x(\Delta T)}| > 0$. Figure 4.19 represents the phase space of all the memristor's final configurations, when plotted to a range of ΔT and a values, which permits to visualize the symmetry around $\Delta T = 0$. Note that for $a \to 30\, V^{-1}s^{-1}$ case 3 resembles case 2.

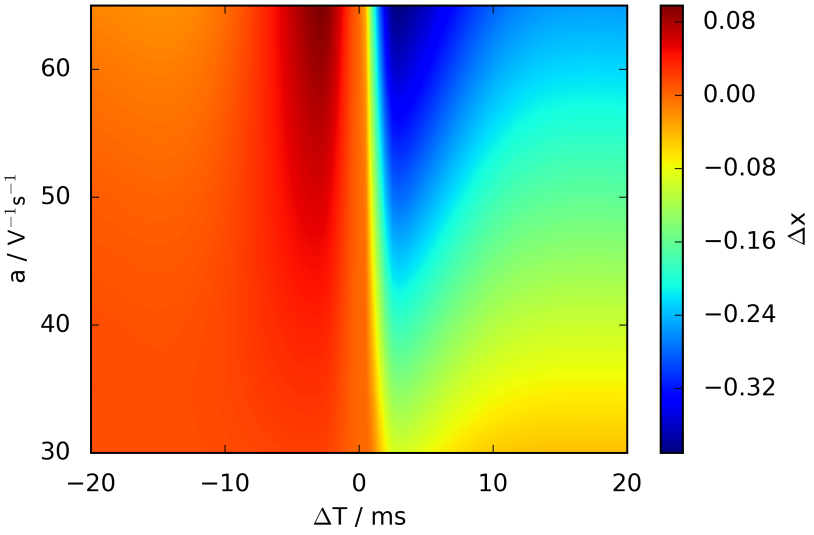


Figure 4.19: Phase space of state change for different values of a and ΔT

4.5 Memristive dynamics

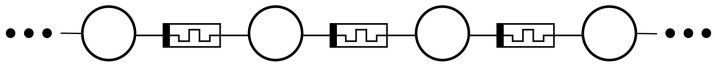


Figure 4.20: Generic hybrid chain (with each circle representing a neuron)

So far I have exemplified that memristive devices are plastic enough to neuronal signals if current experimental devices characteristics can be relaxed, but have not described in detail any hybrid circuit. The simpler non trivial circuit is a single memristive device connecting two neurons (figure 4.20).

There are two very clear phases to synapse: low current and high current. If the state of the synapse is at high resistance, then current output is insufficient to provoke and action potential in the post synaptic neuron; in figure 4.21 until 200 ms the device is effectively just reading the voltage and does not contribute with above threshold current to the overall dynamic of the neural network; on the other extreme, if the resistance is low enough to provoke action potentials, then the connection synchronizes neurons. Also note the orange box, showing inhibition through saturation.

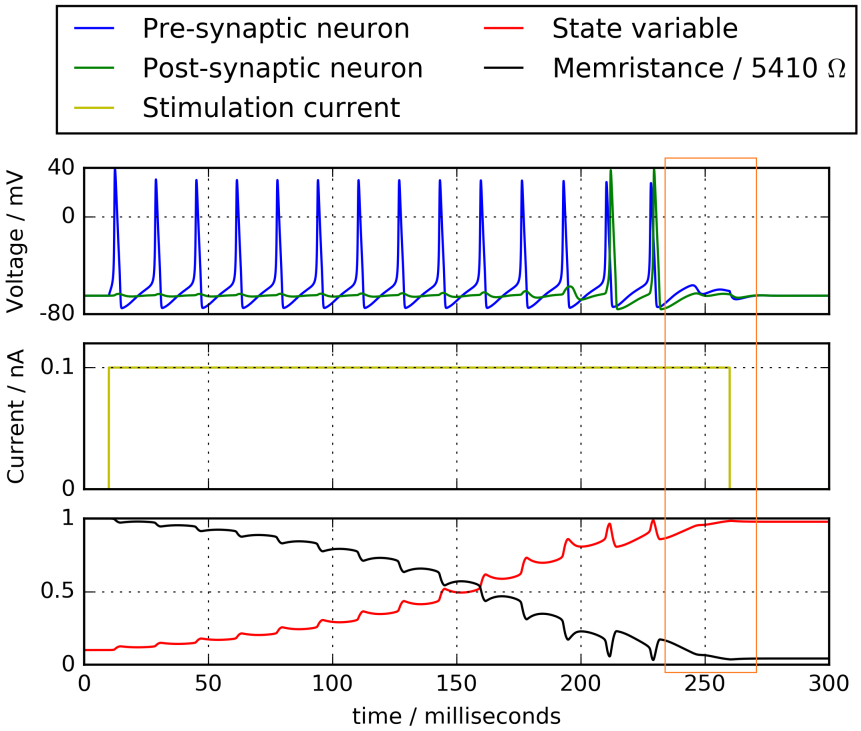


Figure 4.21: Time evolution of memristive synapse (note the saturation after approximately 230 ms)

4.6 Range of interaction

From a standpoint of plasticity and transmission of signals, memristive devices provide a straightforward framework for electrical synapses to combine resistive and chemical characteristics; but the transmission of signals is not guaranteed for all hybrid circuit topologies. To test this characteristic, it is implemented in NEURON an one dimensional chain of neurons connected with realistic flux-charge memristive devices, characterized by a numerical table of flux and current values, for two stimulation schemes with one end of the chain stimulated with 0.5 nA for 200 ms (figure (4.22)):

- the total stimulation time is continuous;
- the total stimulation time is broken into 100 ms chunks, intermittent with 0 nA and 0.5 nA.

Extending the previous scheme to a total stimulation time of 5000 ms, a chain of 10 neurons and 9 memristive devices. In figure 4.23 the mean and standard deviation measure how far the initial stimulus reaches in the chain:

$$< r > (t) = \sum_{i=neuron\ index} \frac{i \cdot s_i(t)}{s_i(t)} \qquad \sigma_r(t) = \sqrt{< r^2 > (t) - < r >^2 (t)}$$

and $s_i(t)$ counts how many spikes have generated in neuron i, until time t.

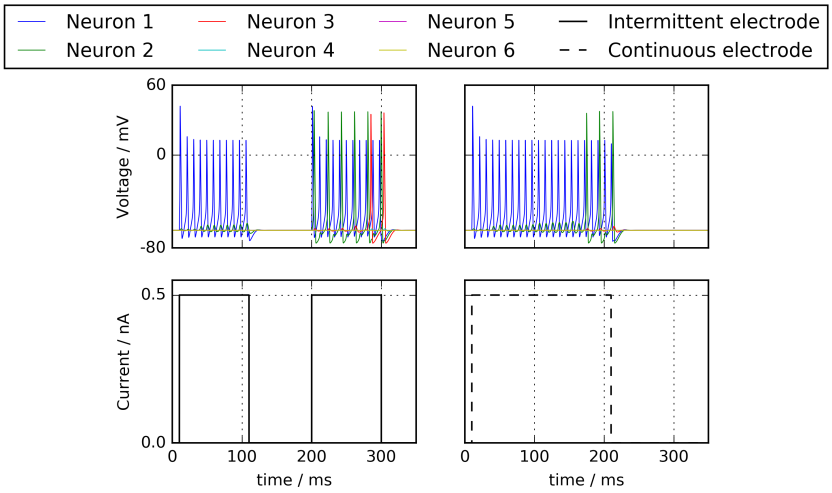


Figure 4.22: Differences between intermittent and continuous stimulation to memristive dynamics

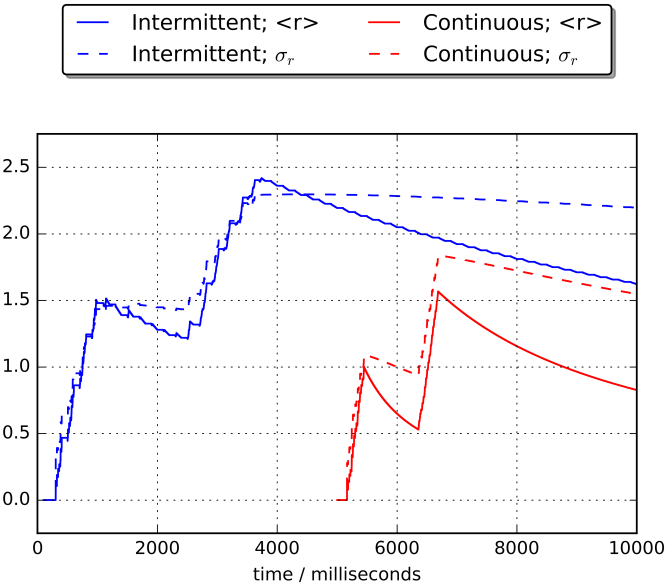


Figure 4.23: Mean and standard deviation of interaction range

Figure 4.23 shows how saturation can be used to inhibit neuronal activity. The next section demonstrates the reason for the local maximum of interaction range, that after approximately 2 seconds of stimulation, the first memristor in the chain reaches a value of resistance low enough to conduct saturating current.

4.7 Memristive learning and timing in generic hybrid circuits

From figure 4.23, the local maximum of interaction range at about 500 ms of stimulation remains to be determined. The most relevant difference between real and simplified devices is the resis-

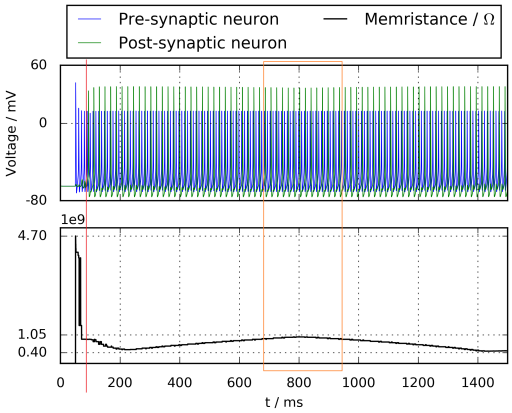


Figure 4.24: Realistic memristive device connecting two HH neurons; the pre-synaptic neuron is continuously stimulated with 0.5 nA (red line and orange box highlight important features)

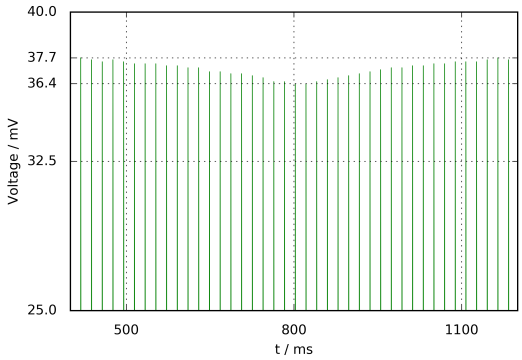


Figure 4.25: Decrease of activity on post-synaptic neuron, with about 1 mV difference between the largest and smallest relative maximum (zoom to orange box region in figure 4.24)

tance trajectory, when subjected to neuronal activity; figure 4.21 shows a linear memristor steadily decreasing its resistance, but figure 4.24 shows how the low resistance state is not necessarily a stable configuration.

Before the red line, in figure 4.24, the post-synaptic neuron initiates a series of failed action potentials due to the particular memristor state and timing of the post- and pre-synaptic spikes. Around 800 ms (orange box), the memristance increases to levels that previously have produced failed action potentials, and the timing of spikes is favorable. However, figure 4.25 demonstrates that the height of the post-synaptic spike decreases roughly 1%, when the low resistance state reaches its maximum. For the chain of the previous section, the local maximum of $< r >$ is caused by a particular temporal sequence of spikes, causing the second neuron in the chain to produce failed action potentials for a given period of time and effectively stop the propagation of the initial stimulus.

Chapter 5

Discussion

Contents

5.1 Signal shape and synaptic plasticity

So far I have assumed that the electrodes are perfect. In electrophysiology the electrical properties of cells can be recorded with intra-cellular or extra-cellular methods, resulting in different signals.

Intra-cellular techniques pierce the membrane and record directly the potential; these types of techniques are unsuitable for long-term applications, because the damage eventually causes cellular death. On the other hand, extra-cellular techniques rely on the placement of electrodes on the surface or near the membrane, measuring other physical properties of the nervous system and noise. Typically for extra-cellular recordings the action potential (figure 4.1 on page 30) is a biphasic signal.

Depending on the location of the electrode the action potential's flux changes, approaching zero when closer to the cell, which causes $\Delta x\,(|\Delta T| \to 0) \to 0$ (in figures 4.17 and 4.18). And any noise causes problems on the cases of synchronous signals, because $\Delta x\,(\Delta T \to 0) \neq 0$ due to the noise's net flux; considering all the variables of signal amplification, shape and memristor model, the noise's net flux (on average is null) can cause the artificial synapse to transition from high to low resistance or vice versa.

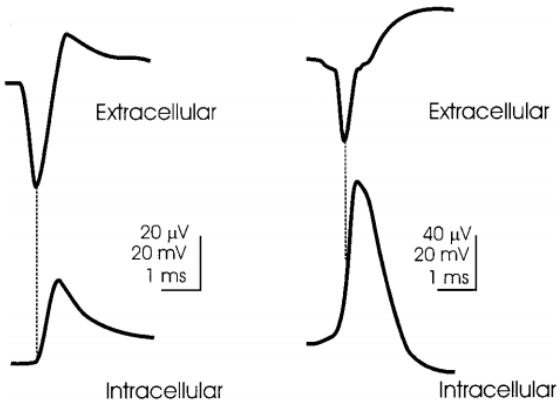


Figure 5.1: Intra-cellular vs extra-cellular recording (scale of extra-cellular signal in μV; intra-cellular signal in mV and time in ms) [97]

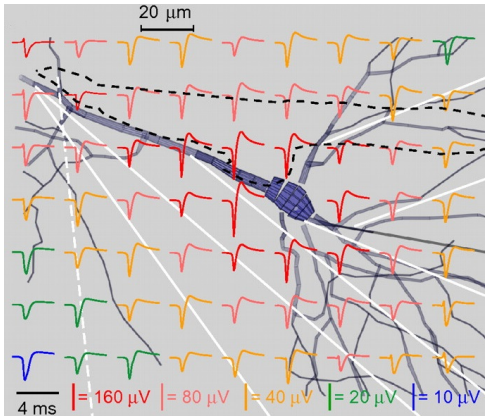


Figure 5.2: Extra-cellular signal as a function of distance (simulation data from [98])

5.2 Modeling devices

For the illustrative purposes of this section I analyze the dynamical behavior for two devices, composed by several layers of different composition and thickness in nanometers (e.g. composition (thickness)).

- SiO/Si/SiO/Ti (25)/Pt (150)/Si (25)/TiW (100); the $V - I$ curve is the left graph on figure 5.3
- SiO/Si/SiO/Ti (25)/Pt (150)/MgO (30)/Ta (20)/Ru (5); the $V - I$ curve is the right graph on figure 5.3

It should be clear by now that hybrid circuits are sensitive to the models of neurons and memristors, and it should have a set of desirable characteristics to be integrated into hybrid circuits. Instead of analyzing models, some stereotypical experimental data is shown and with that important features are extracted. Any data presented is not representative of memristive dynamical behavior, but instead is a particular type of detail that should be more closely examined.

There are many details to analyze in figure 5.3; in the following list, each item is increasingly speculative:

- Endurance;
- Stochasticity;
- Resistance transitions;
- Voltage and/or flux threshold;
- Asymmetric switching;
- Window function.

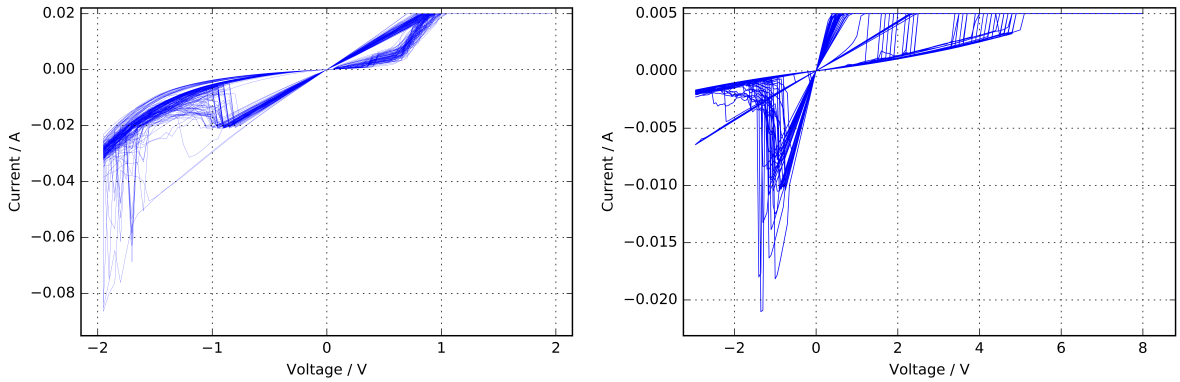


Figure 5.3: Voltage-current curves for two memristive devices, for several cycles of triangular induced voltage

It needs to be noted that the previous items were not studied in detail, but gleaned from models and experimental data, with increasing sensibility to simulation results of hybrid neuro-memristive circuits.

Endurance and stochasticity

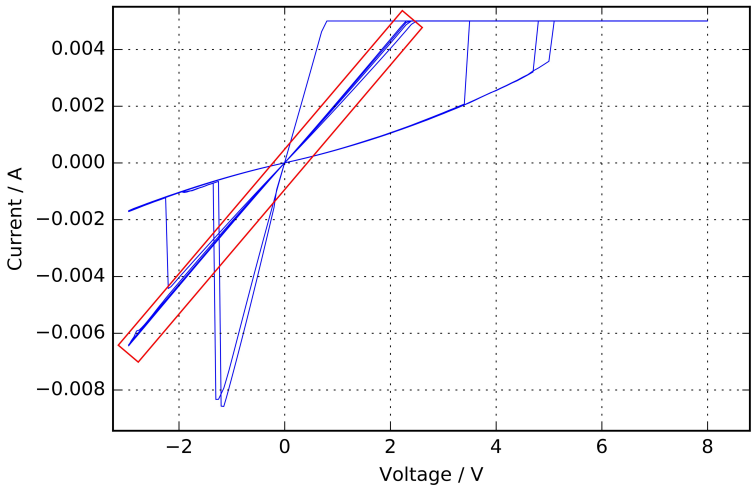


Figure 5.4: Transition from loss of adaption capability, after about 100 cycles (in the red box)

In figure 5.3 the last few cycles show that memristive devices can have very limited endurance. The transition that comes from the loss of adaption capability and from the numerical standpoint of the previously mentioned models, the transition occurs in the state variable differential equation, with

$$\frac{dx}{dt} \neq 0 \longrightarrow \frac{dx}{dt} = 0.$$

Endurance varies immensely from a few hundred to 10^{12} cycles [99, 100], but is generally analyzed from the standpoint of high/low resistance states, ignoring the finer details that are needed for multi-level/continuous applications. Endurance is also tied to the number of cycles, and because

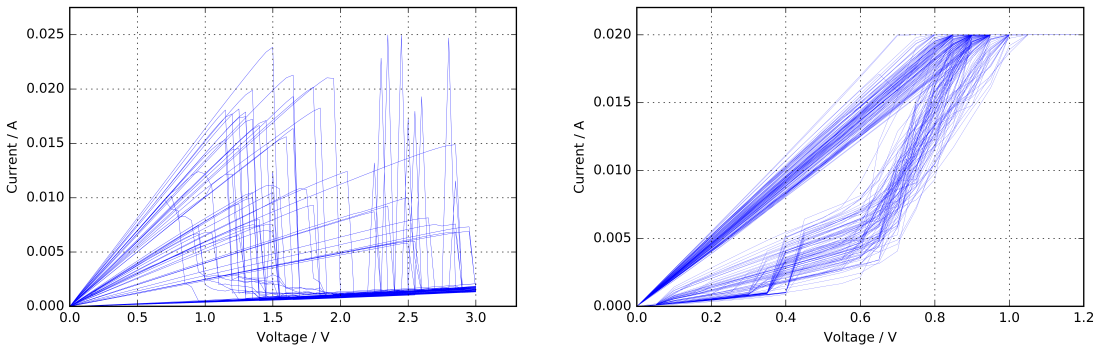


Figure 5.5: Jumps in resistance for two different devices

neuronal dynamics is generally noisy with many zero voltage crossings, it could make current memristive devices unsuitable for the intended applications.

Generally, high and low resistance states are not fixed values and vary stochastically within some device specific range. For example, in figure 5.3, within a cycle the left graph exhibits continuous changes in resistance, making the identification of high and low resistance states hard, while the right graph clearly shows fast transitions (1st and 3rd quadrants), at a variable threshold. The slope of the curve (resistance) after these transitions, in both devices, is also not constant, which introduces further stochastic properties into what is high and low resistance range of states.

Resistance transitions, voltage and flux thresholds

Contrary to the shown models, memristive devices can also exhibit sudden jumps in resistance.

Figure 5.5 zooms into the positive portion of cycles of two different devices, showing the two extremes of possible dynamics; on the left side, there is typically a large resistance jump from low to high with a fast transition (binary memristor) and on the right the transition is spread out and/or composed of smaller jumps. But what is relevant for this section are how those jumps can occur, explained in the following paragraphs.

Thresholds

In figure 5.6, there is clearly some stochastic threshold between resistive and memristive behavior. There are two possibilities:

- Transition occurs between $[0.05;\ 0.40]$ V;
- Transition occurs between $[12;\ 300]$ mWb.

The total flux of a generic voltage spike is about 100 μWb and has a maximum amplitude of 100 mV, meaning that in many cases (with this device) a single spike does not change the memristance, due to threshold properties.

These transitions can be numerically implemented in a few ways: different types of threshold can be introduced into the state variable dynamic or by introducing a new dummy state variable.

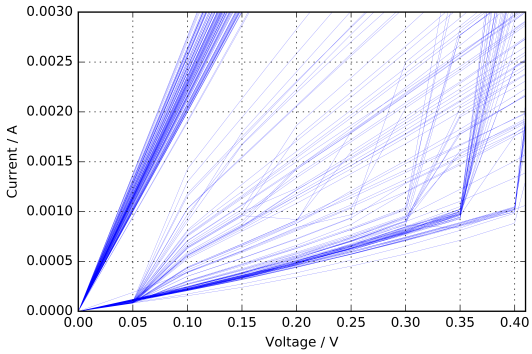


Figure 5.6: Zoom in to positive portion of cycle for continuous memristor

Threshold can be hard or soft, that is, the transition can occur sharply between two domains or there is an intermediate region between the two domains. For hard thresholds, $\dot{x} \longrightarrow \dot{x} \cdot f(y)$ where $f(y) = \begin{cases} 1 & \text{outside range} \\ 0 & \text{inside range} \end{cases}$, with $y =$ voltage or flux; and for soft thresholds, where $f(y)$ is at least a C^0 function.

However, whether the threshold is hard or soft, these transitions can introduce unchecked error that makes the system diverge from its analytical solution. The two variable model is an approximation where a second variable is added to the model in the following way:

$$\begin{aligned} V(t) &= M(z)\, I(t) \\ z &= h(x) \\ \frac{dx}{dt} &= f(x,\, u,\, t) \end{aligned}$$

and $V =$ voltage, $I =$ current, $M =$ memristance.

Like the originally presented models, x is a bounded variable that changes under non-null input, but in this case, $f(x,\, u,\, t)$ can absorb the window function, a time dependent function and a dependency on voltage, flux, current or charge (u). The new variable is introduced (z) to copy the original state under certain conditions:

$$\begin{cases} z = x & if\ y^{-}_{threshold} > y > y^{+}_{threshold} \\ \dot{z} = 0 & else \end{cases}.$$

The new variable z is the state variable that affects the memristance, where any discontinuity (transition) occurs in the variable z but the variable x changes continuously.

Asymmetric switching

If the electrodes of the device are not equal, filament formation is not symmetric, translating into asymmetric switching. Assuming that the electrodes are ohmic ($I = {}^{V}/_{R}$), then the asymmetry is easy to translate into a numerical approximation:

$$V \to \begin{cases} \frac{V}{V^{-}_{rescale}} & if\, V < 0 \\ \frac{V}{V^{+}_{rescale}} & if\, V > 0 \end{cases}$$

To the analyzed one state models, so far, it makes the device change faster or slower depending on the signal of applied voltage and can adjusted from the voltage rescale values ($V_{rescale}$).

Device example

It has been mentioned that window functions were one of the first theoretical approaches to real devices, but are now regarded as an over simplification, while still useful for circuit or large scale applications; the advantage of using window functions comes from the fact that most numerical approximations can be hidden in this artificial property.

The following analytical device exemplifies different characteristics that were explained so far:

- linear device;
- voltage threshold ($= th$), screening the device from small amplitude random noise;
- unequal electrodes;
- two variable model, flux (φ) controlled, to implement resistance transitions without numerical problems from discontinuities.

$$V = \left[R_{on} z + R_{off}\left(1 - z\right)\right] I$$

$$f\left(V\right) = \begin{cases} \frac{V - V^{+}_{th}}{V^{+}_{rescale}} & if\, V > V^{+}_{th} \\ \frac{V + V^{-}_{th}}{V^{-}_{rescale}} & if\, V < V^{-}_{th} \\ 0 & else \end{cases} \qquad \dot{x} = \alpha \cdot window\left(x\right) \cdot f\left(V\right) \qquad \begin{cases} z = x & if\, \varphi^{-}_{th} > \varphi > \varphi^{+}_{th} \\ \dot{z} = 0 & else \end{cases}$$

No simulation data of this device is shown, because the addition of voltage thresholds causes numerical instability. More specifically, numerical stability in NEURON after $V = V^{+}_{th}$ or V^{-}_{th} is not guaranteed.

5.3 Memristive tests

The intended application restricts what characteristics are desirable: binary, multi-level or continuous memristive synapses result in different hybrid dynamics. Current experimental memristors

can function under distinct principles (section 2.3), with each type adequate for certain applications. For any hybrid circuit application, tests need to be performed to determine what are the device's characteristics.

Endurance

This is already a standard test in the area, being obtainable through a periodic signal of large enough amplitude; the resistance transitions eventually destroy the flexibility of the device, but helps determine the endurance cycle of the device.

Threshold

To identify the type or types of threshold in the devices there are some battery of tests that can be employed.

- Linear (or otherwise) voltage/current sweeps

Changing the time scale of these sweeps it should be possible to identify the type of threshold; at the start of each sweep the device should always be on the same state (high or low). If the data shows transitions between resistive and memristive behavior at a consistent range of voltages/currents then there is a voltage/current threshold and if the transition range increases with decreasing slope (or vice versa) then there is a flux/charge threshold.

The problem with this set of measures comes from transient dynamics of these devices, which can hide the transition.

- Beat

General measures of memristors use periodic signals to test high/low resistance states, endurance, etc; using the scheme with a sum of two signals of approximate frequency it should possible to identify the type of threshold. The smaller frequency ensures that the flux/charge is zero periodically, and the larger frequency makes the state transits periodically but with different values of flux/charge.

The phase between null flux/charge and null voltage/current (after the transient period) should enable to identify the type of threshold: if at each zero voltage/current crossing the device becomes resistive then there is voltage/current threshold, but if the transition between resistive and memristive (and vice versa) occurs once per period of the beat's envelope then there is flux/charge threshold.

Imbalance of electrodes and other characteristics

To analyze how both electrodes compare, it is not as simple as interchanging the sign of the input signal. While it is possible to saturate the memristor in the high or low resistance state, it is not clear that an halfway symmetric state (x_{half}) exists in all devices; this state needs to obey

the following: $window\left(x_{half}+\varepsilon\right) = window\left(x_{half}-\varepsilon\right)$ for $\varepsilon \to 0$. In practical terms, this state is impossible to detect because any number of factors, for example, minute changes in the window function, can hide this from the experimental point of view.

5.4 N$_i$M nodes

Going back to section 4.1.2, HH neurons can show signs of being under or overstimulated, both resulting in a relative lack of electrical activity. However, under any stimulation, the neuron does not remain in its resting potential, which causes voltage differences between nodes on the system, and therefore, causes the electrical connections to pass some current. In figure 4.4 a HH neuron with 20 μm radius becomes saturated at less than 0.4 nA and requires more than 0.08 nA to spike.

So far, only the effect of one memristor on one neuron has been analyzed, but apart from experimental limitations, it should be possible to connect several devices to the same neuron. But in the same perspective, a neuron connected to a memristor has a different dynamic from a neuron connected to several memristors.

It is useful to conceptualize hybrid circuits as networks where nodes are neurons and the edges are the artificial synapses (in this case edges are dynamic elements); at each node the dynamic is determined by the number of edges that are connected, being distinguished by the term N$_i$M $=$ (# edges that are depressed from node's action)$_{\text{node index}}$(# edges that are potentiated from node's action).

Given that voltage spikes have an amplitude of around 100 mV, an estimate of a spike at half height should be able to induce action potentials in the post-synaptic neuron, with the following memristance range (R):

$$\begin{aligned} 0.4\,nA &\gtrsim \tfrac{50\,mV}{R} &\Rightarrow R \gtrsim\ & 10^8\,\Omega &= R_1 \\ 0.08\,nA &\lesssim \tfrac{50\,mV}{R} &\Rightarrow R \lesssim\ & 6\cdot 10^9\,\Omega &= R_2 \end{aligned}$$

The actual range of memristance depends on the application, but to activate node i if P ($\leq$ N+M) pre-synaptic spikes are needed, on average, the memristance should be on the following range: $R \in P \cdot [R_1; R_2]$. If node i emits a voltage spike, it is distributed to all P artificial synapses.

Regardless of the resistance of all memristors in the N$_i$M node, the sum of the input signals has a cumulative effect that even if no action potentials reach this node, like noise or uninitiated voltage spikes, can still induce neuron (or node) i to become active for a given period time.

5.5 Workaround for threshold and energy requirements

In section 4.1.5 there is a mention that real memristors have a minimum energy requirement for full resistance switching, on the order of magnitude of femtoJoules, and voltage/current thresholds

screen devices for small signal amplitudes; these characteristics are not favorable for resistance switching controlled by neuronal dynamics. Amplifying the voltage of a cell is capable of solving these problems, but at the expense of further noise and with a single action potential switching the resistance, whether it is a spurious spike or within a train spike. Signal-to-noise ratio can be controlled (to a point) by the quality of components and by the experimental conditions, but controlling spurious electrical activity is not feasible for a large scale systems. However, summing the membrane voltage of several neurons can solve the problem of spurious activity, by taking advantage of synchronous neural populations, where at any time there is electrical activity, but only when several neurons are active does it represent a definite event.

5.6 Applications

Throughout this dissertation, some tentative attempts to link structures of the nervous system and memristive devices have been proposed and to some extent, other electrical elements. Section 2.2 elaborates on the electrochemical principles of spike generation and transmission, that results in an event-driven dynamics, while electrical circuits depend on electromagnetic fields; this means that equating the synapses/neurons and memristors comes with a few caveats. The difference between synaptic and memristive plasticity is one of the most visible: synaptic plasticity is based on homeostatic principles that neurons will fire within some range of frequency, without saturation or stop generating spikes, correlating the causality of the same spikes, while memristive plasticity is related to the timing of pre- and post-synaptic signals (not necessarily spikes) and the net flux/charge, both of them dictating the resistance change. Another difference between hybrid systems and electrical circuits is on the nodes, that comes is three varieties: N_iM nodes, electrical nodes and neuronal nodes. Electrical nodes are analyzed by Kirchoff's laws, neuronal nodes are the nodes of neuronal networks, ruled by biological principles and N_iM nodes are a combination of the previous that can be expanded to equivalent circuits, but it is not feasible for detailed neurons, either for morphological or electrochemical details.

There is, however, common ground between electrical and neuronal networks. Gap junctions are useful to synchronize neurons, their electrical behavior enables to identify neural populations that show closely correlated firing and fast propagation, enabling "knee-jerk" reactions. The generalization of these information/spikes "highways" are synfire chains, structures of neuronal populations connected with feed forward excitatory and few recurrent connections, that synchronize neurons. Memristive devices take advantage of these characteristics, with adaption and non-symmetry in input sign, and while it is not explicitly feed forward, back propagation inhibition can be engineered into the device, through device parameters or by the use of other electrical elements. Taking these characteristics/applications in mind, the primary element of hybrid circuits are hybrid chains (see section 4.6); each node of this chain is stimulated by the previous node, or in a way, each node copies the electrical activity to its neighbors, propagating the initial stimulus. The obvious generalization is N interconnected chains, in 2d or 3d setups, connecting different neuronal populations

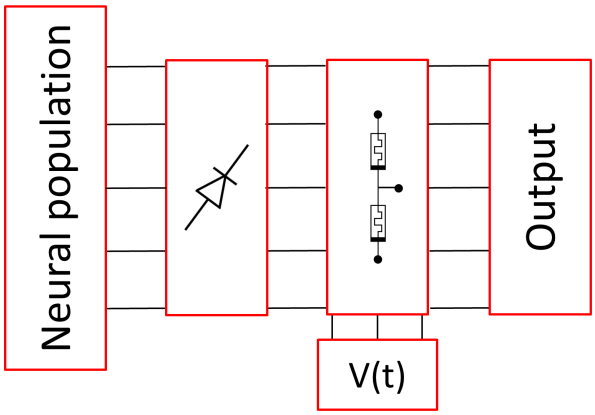


Figure 5.7: Generic scheme for a artificial biphasic neuron

and creating an adaptive conductive bridge, with or without asymmetric conductance. Besides the obvious electrical conductance use, organizing these chains in this meld of electrical and biological could, hypothetically, be used to create hybrid memory "banks". These memories are the hybrid equivalent of memristive crossbar arrays.

These hybrid chains and bridges can be used to construct adaptive control systems and conceptualizing these systems on the basis of input and output layers, the input layer signals can be spatially directed into different output layers, while also interacting with other analog/digital systems. A hybrid system using these elements can be further conceptualized as a control system, or part of one, processing the input information, dynamically changing its parameters and relaying the processed information. If these types of systems can be made biocompatible, the interaction between biological tissue and electric circuits is possible, paving a way for biorobotics.

However, this is an ambitious objective and requires the careful analysis of simpler applications. It has been exemplified, so far, of memristive devices as artificial surrogates to synapses because it is the most obvious use and enables to identify most of the possible problems and generic dynamical properties. Neuromorphic research has produced several theoretical and experimental artificial neurons, most using transistors but also memristors [84], and depending on the use, resulting in variable accuracy of spike profile. It is now proposed a generic setup to mimic a spatially distributed neuron recorded by extra-cellular electrodes (section 5.1), using the simple concept of voltage dividers (section 4.1.6).

In figure 5.7, it is represented an artificial neuron, as a projection of a biological neural network, by way of memristance dynamic. From left to right: the neuronal population is a set of live/recorded signals, being used to control memristance values in the setup; the diode symbol indicates that the connection between neurons and memristor is asymmetric, with the forward direction from neuronal population to memristive array; the memristive voltage divider is a general array of series memristors, controlled by the input signal V(t)= square wave, generating biphasic spikes at each device once per period of the input; the output layer organizes the signal spatially.

The concept of hybrid chains is still retained on this artificial neuron, because the connection of the neuronal population and the memristive array can be engineered to change certain properties of the input signal. This connection takes a distributed set of signals from the neural population and cellular neural networks, a general type of network that includes hybrid circuits, can be used for edge detection in image processing, for example, and, with some generalization, to detect spatial patterns in neuronal activity, via electrical spikes. Other types of neural networks, like convolution [101] or recurrent [102] neural networks, can be used to reduce noise in input signal.

Chapter 6

Conclusions

The integration of biological and electrical systems is an emerging field with important applications in many areas of medicine and neuroscience. The potential of this field in the construction of innovative therapeutic strategies is vast, and so is the demand for producing real word applications which need to be scalable, lightweight and efficient. However, like other emerging fields, the long term effects of these applications are still unknown, and for the nervous system arguably the most important portion of the human body this becomes an enormous barrier for any potential. This means that any candidate application needs to be thoroughly explored by a combination of theoretical, computational and experimental work. Because models are still not developed enough to explain many emerging characteristics of the nervous system, interfacing this gray box (combination of black and white box systems) with other types of systems can lead to many unknown side effects and endanger any potential application. The work here presented is an important step towards the understanding and assessment of the conditions to effectively bridge memristive devices and real neurons.

On the larger scope of scientific literature, hybrid systems fall into the broad category of cellular neural networks, which so far, have proven useful in image processing, pattern recognition and analysis, modeling of certain physical processes or even as sensors. Historically, the cells of these networks have interesting biology-like properties (e.g. neuromorphic) but technological constraints have directed these systems into entirely electronic applications, that while using the same ideas as the nervous system, turns into a different system. And replacing the cells by realistic neurons brings a whole ensemble of design constraints and unexpected effects.

The main concern that needs to be considered in any further study comes on model sensitivity. Hybrid systems are a combination of biological networks (event-driven, timing sensitive) and electrical circuits (dynamic, sensitive to voltage differences), not necessarily producing compatible dynamics. For example, a N_iM node with $N+M \neq 0$ will at any instant receive a variety of signals that do not qualify as action potentials, whether it is uninitiated action potentials or just noise, and the sum of all inputs can cross the current threshold that the neuron needs to fire, cause adaption in the system's parameter space and if the system is an unstable equilibrium, diverge it into an

unexpected direction. In addition, over stimulation can also inhibit electrical activity, because of saturation properties in realistic neurons.

The range of stimulation in these nodes, between under and over stimulation, changes the timing of signal transmission in hybrid circuits and when those signals are the typical voltage spike of neuronal activity, a slight change in the connection (e.g. resistance) can completely stop the signal's propagation. However, through careful consideration of the hybrid circuit's design, these timing issues or parameter variation causing unwanted inhibition or excitation are relegated to the quality of the biological and electrical engineering.

The design of hybrid circuit is dependent on the type of memristive device used, where symmetry of conductance, switching speed, threshold or endurance restrict what applications are possible. For example, chemical and electrical synapses can be mimicked by tuning the symmetry of conductance: symmetric conductances can mimic electrical synapses and asymmetric conductances can mimic chemical synapses. Other tunable parameters can be exploited to create a diverse set of elements in the context of hybrid circuits, like switches, artificial synapses or spiking devices.

Overall, this work shows that hybrid systems composed of neuronal connections mediated by memristive devices have an enormous potential in neuroscience and medicine. Furthermore, it shows that the constraints imposed to effectively bridge neurons and memristive devices are amenable by today's technological standards. The time is therefore ripe to explore and construct possible applications.

Chapter 7

Future work

I have shown in a general manner that hybrid circuits can be integrated into neuronal or electrical networks, but it remains be to proved by a proof of concept. Before this proof of concept can be done, some basis work still needs to be done, mostly on the characterization of memristive devices.

Either by an extensive review of literature or by experimental work, a sufficiently diverse set of memristive devices need to be characterized when subjected to periodic (already extensively done) and non-periodic signals (less common). These tests will measure a variety of different properties, ranging from thresholds, endurance, resistance range/symmetry and change speed, because different applications call for different memristors.

When this database of properties for different devices is collected, a specific application can be studied and to simplify any future work, these applications should start by proving toy model concepts: hybrid chains, the simplest hybrid circuit, and artificial biphasic neurons.

In previous sections, I have argued that hybrid chains are one of the basic elements of hybrid circuits, and if they are proven to function in experimental conditions, it paves the way for the development of control systems between the nervous system and electronic applications. Still using the concept of hybrid chains and other more generic concepts in artificial neural networks, the proposed setup of artificial biphasic neuron presents the first non trivial application of the work developed so far, where the objective is to create a memristive array that can emit spikes.

Bibliography

[1] David Sterratt, Bruce Graham, Andrew Gillies, and David Willshaw. *Principles of Computational Modelling in Neuroscience*. Cambridge University Press (CUP), 2009.

[2] Hanne Poulsen, Himanshu Khandelia, J. Preben Morth, Maike Bublitz, Ole G. Mouritsen, Jan Egebjerg, and Poul Nissen. Neurological disease mutations compromise a c-terminal ion pathway in the na+/k+-ATPase. *Nature*, 467(7311):99–102, Aug 2010.

[3] C. A. Hubner. Ion channel diseases. *Human Molecular Genetics*, 11(20):2435–2445, Oct 2002.

[4] Trudy Pang, Ramin Atefy, and Volney Sheen. Malformations of cortical development. *The Neurologist*, 14(3):181–191, May 2008.

[5] J. Jeffrey Moffat, Minhan Ka, Eui-Man Jung, and Woo-Yang Kim. Genes and brain malformations associated with abnormal neuron positioning. *Molecular Brain*, 8(1):1–12, Nov 2015.

[6] Frederico A.C. Azevedo, Ludmila R.B. Carvalho, Lea T. Grinberg, José Marcelo Farfel, Renata E.L. Ferretti, Renata E.P. Leite, Wilson Jacob Filho, Roberto Lent, and Suzana Herculano-Houzel. Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *The Journal of Comparative Neurology*, 513(5):532–541, Apr 2009.

[7] H. Braendgaard, S. M. Evans, C. V. Howard, and H. J. G. Gundersen. The total number of neurons in the human neocortex unbiasedly estimated using optical disectors. *Journal of Microscopy*, 157(3):285–304, Mar 1990.

[8] Mark J. West and H. J. G. Gundersen. Unbiased stereological estimation of the number of neurons in the human hippocampus. *The Journal of Comparative Neurology*, 296(1):1–22, Jun 1990.

[9] Yong Tang, Jens R. Nyengaard, Didima M.G. De Groot, and Hans Jørgen G. Gundersen. Total regional and global number of synapses in the human brain neocortex. *Synapse*, 41(3):258–273, 2001.

[10] Robin IM Dunbar. Neocortex size as a constraint on group size in primates. *Journal of Human Evolution*, 22(6):469–493, Jun 1992.

[11] Mark Bear, Barry Connors, and Michael Paradiso. *Neuroscience : exploring the brain*. Wolters Kluwer, Philadelphia, 4th edition, 2016.

[12] Peter Dayan and L.F. Abbott. *Theoretical Neuroscience - Computational and Mathematical Modeling of Neural Networks*. 1st edition, Oct 2000.

[13] Simon Thorpe, Denis Fize, and Catherine Marlot. Speed of processing in the human visual system. *Nature*, 381(6582):520–522, Jun 1996.

[14] James T. Fulton. *Processes in Biological Vision*. online, 2009. Available on the Internet: http://neuronresearch.net/vision/.

[15] Greg Stuart, Nelson Spruston, Bert Sakmann, and Michael Häusser. Action potential initiation and backpropagation in neurons of the mammalian {CNS}. *Trends in Neurosciences*, 20(3):125 – 131, Mar 1997.

[16] Ichiji Tasaki. Collision of two nerve impulses in the nerve fibre. *Biochimica et Biophysica Acta*, 3:494–497, Jan 1949.

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

[18] Mitchell A. Nahmias, Bhavin J. Shastri, Alexander N. Tait, and Paul R. Prucnal. A leaky integrate-and-fire laser neuron for ultrafast cognitive computing. *IEEE J. Select. Topics Quantum Electron.*, 19(5):1–12, Sep–Oct 2013.

[19] Qintao Gan, Rui Xu, and Pinghua Yang. Exponential synchronization of stochastic fuzzy cellular neural networks with time delay in the leakage term and reaction-diffusion. *Communications in Nonlinear Science and Numerical Simulation*, 17(4):1862–1870, Apr 2012.

[20] Harvey Lodish, Arnold Berk, S Lawrence Zipursky, Paul Matsudaira, David Baltimore, and James Darnell. *Molecular Cell Biology, Section 21.4*. W. H. Freeman, 4th edition edition, 2000.

[21] Sheriar G. Hormuzdi, Mikhail A. Filippov, Georgia Mitropoulou, Hannah Monyer, and Roberto Bruzzone. Electrical synapses: a dynamic signaling system that shapes the activity of neuronal networks. *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 1662(1-2):113–137, Mar 2004.

[22] M. Mayford, S. A. Siegelbaum, and E. R. Kandel. Synapses and memory storage. *Cold Spring Harbor Perspectives in Biology*, 4(6):a005751, Jun 2012.

[23] Tomonori Takeuchi, Adrian J. Duszkiewicz, and Richard G. M. Morris. The synaptic plasticity and memory hypothesis: encoding, storage and persistence. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 369(1633), Jan 2013.

[24] Robert C. Malenka and Roger A. Nicoll. NMDA-receptor-dependent synaptic plasticity: multiple forms and mechanisms. *Trends in Neurosciences*, 16(12):521–527, Dec 1993.

[25] Robert S. Zucker and Wade G. Regehr. Short-term synaptic plasticity. *Annual Review of Physiology*, 64(1):355–405, Mar 2002.

[26] A. Citri and R.C. Malenka. Synaptic plasticity: Multiple forms, functions, and mechanisms. *Neuropsychopharmacology*, 33(1):18–41, Aug 2007.

[27] Guo-qiang Bi and Mu-ming Poo. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *The Journal of Neuroscience*, 18(24):10464–10472, Dec 1998.

[28] Carla J. Shatz. Impulse activity and the patterning of connections during cns development. *Neuron*, 5(6):745–756, Dec 1990.

[29] Narayanan Kasthuri and Jeff W. Lichtman. The role of neuronal identity in synaptic competition. *Nature*, 424(6947):426–430, jul 2003.

[30] Francisco Javier Ropero Pelaez, Mariana Antonia Aguiar-Furucho, and Diego Andina. Intrinsic plasticity for natural competition in koniocortex-like neural networks. *International Journal of Neural Systems*, 26(05):1650040, Aug 2016. PMID: 27255800.

[31] Ajaz Ahmad Bhat, Gaurang Mahajan, and Anita Mehta. Learning with a network of competing synapses. *PLoS ONE*, 6(9):e25048, Sep 2011.

[32] Kenneth D. Miller. Synaptic economics: Competition and cooperation in synaptic plasticity. *Neuron*, 17(3):371–374, Sep 1996.

[33] Gina G Turrigiano. Homeostatic plasticity in neuronal networks: the more things change, the more they stay the same. *Trends in Neurosciences*, 22(5):221–227, May 1999.

[34] Gina G. Turrigiano and Sacha B. Nelson. Homeostatic plasticity in the developing nervous system. *Nature Reviews Neuroscience*, 5(2):97–107, Feb 2004.

[35] Anthony Holtmaat and Karel Svoboda. Experience-dependent structural synaptic plasticity in the mammalian brain. *Nature Reviews Neuroscience*, 10(9):647–658, Sep 2009.

[36] Min Fu and Yi Zuo. Experience-dependent structural plasticity in the cortex. *Trends in Neurosciences*, 34(4):177–187, Apr 2011.

[37] Dmitri B. Strukov, Gregory S. Snider, Duncan R. Stewart, and R. Stanley Williams. The missing memristor found. *Nature*, 453(7191):80–83, May 2008.

[38] R. Williams. How we found the missing memristor. *IEEE Spectr.*, 45(12):28–35, Dec 2008.

[39] Yogesh N Joglekar and Stephen J Wolf. The elusive memristor: properties of basic electrical circuits. *European Journal of Physics*, 30(4):661–675, May 2009.

[40] Yenpo Ho, Garng M. Huang, and Peng Li. Nonvolatile memristor memory: Device characteristics and design implications. In *Proceedings of the 2009 International Conference on Computer-Aided Design*, ICCAD '09, pages 485–490, New York, NY, USA, 2009. ACM.

[41] Giacomo Indiveri, Bernabe Linares-Barranco, Robert Legenstein, George Deligeorgis, and Themistoklis Prodromakis. Integration of nanoscale memristor synapses in neuromorphic computing architectures. *Nanotechnology*, 24(38):384010, Sep 2013.

[42] Sung Hyun Jo, Ting Chang, Idongesit Ebong, Bhavitavya B. Bhadviya, Pinaki Mazumder, and Wei Lu. Nanoscale memristor device as synapse in neuromorphic systems. *Nano Letters*, 10(4):1297–1301, Apr 2010.

[43] H.-S. Philip Wong, Heng-Yuan Lee, Shimeng Yu, Yu-Sheng Chen, Yi Wu, Pang-Shiu Chen, Byoungil Lee, Frederick T. Chen, and Ming-Jinn Tsai. Metal–oxide rram. *Proceedings of the IEEE*, 100(6):1951–1970, Jun 2012.

[44] Leon Chua. Resistance switching memories are memristors. *Applied Physics A*, 102(4):765–783, Mar 2011.

[45] H. Y. Lee, P. S. Chen, T. Y. Wu, Y. S. Chen, C. C. Wang, P. J. Tzeng, C. H. Lin, F. Chen, C. H. Lien, and M. J. Tsai. Low power and high speed bipolar switching with a thin reactive ti buffer layer in robust hfo2 based rram. In *2008 IEEE International Electron Devices Meeting*, pages 1–4, Dec 2008.

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

[47] L. Chua. Everything you wish to know about memristors but are afraid to ask. *Radioengineering*, 24(2):319–368, Jun 2015.

[48] J. Joshua Yang, Dmitri B. Strukov, and Duncan R. Stewart. Memristive devices for computing. *Nature Nanotechnology*, 8(1):13–24, Dec 2012.

[49] Sascha Vongehr and Xiangkang Meng. The missing memristor has not been found. *Scientific Reports*, 5:11657, Jun 2015.

[50] A. Fantini, L. Goux, R. Degraeve, D. J. Wouters, N. Raghavan, G. Kar, A. Belmonte, Y. Y. Chen, B. Govoreanu, and M. Jurczak. Intrinsic switching variability in hfo2 rram. In *2013 5th IEEE International Memory Workshop*, pages 30–33, May 2013.

[51] A. Chen and M. R. Lin. Variability of resistive switching memories and its impact on crossbar array performance. In *Reliability Physics Symposium (IRPS), 2011 IEEE International*, pages MY.7.1–MY.7.4, April 2011.

[52] E. Lehtonen and M. Laiho. Stateful implication logic with memristors. In *2009 IEEE/ACM International Symposium on Nanoscale Architectures*, pages 33–36, Jul 2009.

[53] Siddharth Gaba, Patrick Sheridan, Jiantao Zhou, Shinhyun Choi, and Wei Lu. Stochastic memristive devices for computing and neuromorphic applications. *Nanoscale*, 5:5872–5878, Jul 2013.

[54] Curtis O'Kelly, Jessamyn A. Fairfield, and John J. Boland. A single nanoscale junction with programmable multilevel memory. *ACS Nano*, 8(11):11724–11729, Oct 2014.

[55] Payam Rabbani, Rasoul Dehghani, and Nima Shahpari. A multilevel memristor–cmos memory cell as a reram. *Microelectronics Journal*, 46(12, Part A):1283 – 1290, Dec 2015.

[56] Kuk-Hwan Kim, Siddharth Gaba, Dana Wheeler, Jose M. Cruz-Albrecht, Tahir Hussain, Narayan Srinivasa, and Wei Lu. A functional hybrid memristor crossbar-array/cmos system for data storage and neuromorphic applications. *Nano Letters*, 12(1):389–395, Dec 2011. PMID: 22141918.

[57] Antonio C Torrezan, John Paul Strachan, Gilberto Medeiros-Ribeiro, and R Stanley Williams. Sub-nanosecond switching of a tantalum oxide memristor. *Nanotechnology*, 22(48):485203, Nov 2011.

[58] F. Merrikh-Bayat and S. Bagheri Shouraki. Memristive neuro-fuzzy system. *IEEE Transactions on Cybernetics*, 43(1):269–285, Jan 2013.

[59] M. Klimo and O. Such. Memristors can implement fuzzy logic. *ArXiv e-prints*, October 2011.

[60] M. Chu, B. Kim, S. Park, H. Hwang, M. Jeon, B. H. Lee, and B. G. Lee. Neuromorphic hardware system for visual pattern recognition with memristor array and cmos neuron. *IEEE Transactions on Industrial Electronics*, 62(4):2410–2419, Apr 2015.

[61] Teresa Serrano-Gotarredona, Timothee Masquelier, Themistoklis Prodromakis, Giacomo Indiveri, and Bernabe Linares-Barranco. Stdp and stdp variations with memristors for spiking neuromorphic learning systems. *Frontiers in Neuroscience*, 7:2, Feb 2013.

[62] H. Kim, M. P. Sah, C. Yang, T. Roska, and L. O. Chua. Neural synaptic weighting with a pulse-based memristor circuit. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 59(1):148–158, Jan 2012.

[63] MAKOTO ITOH and LEON O. CHUA. Memristor cellular automata and memristor discrete-time cellular neural networks. *International Journal of Bifurcation and Chaos*, 19(11):3605–3656, Nov 2009.

[64] C. Dias, L. M. Guerra, J. Ventura, and P. Aguiar. Memristor-based willshaw network: Capacity and robustness to noise in the presence of defects. *Appl. Phys. Lett.*, 106(22):223505, Jun 2015.

[65] K Szot, R Dittmann, W Speier, and R Waser. Nanoscale resistive switching in srtio3 thin films. *physica status solidi (RRL)-Rapid Research Letters*, 1(2):R86–R88, Jan 2007.

[66] John Paul Strachan, Antonio C Torrezan, Gilberto Medeiros-Ribeiro, and R Stanley Williams. Measuring the switching dynamics and energy efficiency of tantalum oxide memristors. *Nanotechnology*, 22(50):505402, Nov 2011.

[67] C.A. Richter, D.R. Stewart, D.A.A. Ohlberg, and R.Stanley Williams. Electrical characterization of al/alox/molecule/ti/al devices. *Applied Physics A*, 80(6):1355–1362, Mar 2005.

[68] K. Terabe, T. Hasegawa, T. Nakayama, and M. Aono. Quantized conductance atomic switch. *Nature*, 433(7021):47–50, Jan 2005.

[69] D. R. Stewart, D. A. A. Ohlberg, P. A. Beck, Y. Chen, , R. Stanley Williams, J. O. Jeppesen, K. A. Nielsen, , and J. Fraser Stoddart. Molecule-independent electrical switching in pt/organic monolayer/ti devices. *Nano Letters*, 4(1):133–136, Dec 2003.

[70] A. Beck, J. G. Bednorz, Ch. Gerber, C. Rossel, and D. Widmer. Reproducible switching effect in thin oxide films for memory applications. *Applied Physics Letters*, 77(1):139–141, Jul 2000.

[71] Ting Chang, Sung-Hyun Jo, Kuk-Hwan Kim, Patrick Sheridan, Siddharth Gaba, and Wei Lu. Synaptic behaviors and modeling of a metal oxide memristive device. *Applied Physics A*, 102(4):857–863, feb 2011.

[72] Ahmed G. Radwan and Mohammed E. Fouda. *Memristor: Models, Types, and Applications*, page 17. Springer International Publishing, Cham, 2015.

[73] U. Russo, D. Ielmini, C. Cagli, and A. L. Lacaita. Filament conduction and reset mechanism in nio-based resistive-switching memory (rram) devices. *IEEE Transactions on Electron Devices*, 56(2):186–192, Feb 2009.

[74] G. Bersuker, D. C. Gilmer, D. Veksler, P. Kirsch, L. Vandelli, A. Padovani, L. Larcher, K. McKenna, A. Shluger, V. Iglesias, M. Porti, and M. Nafría. Metal oxide resistive memory switching mechanism based on conductive filament properties. *Journal of Applied Physics*, 110(12), Dec 2011.

[75] H. Y. Peng, G. P. Li, J. Y. Ye, Z. P. Wei, Z. Zhang, D. D. Wang, G. Z. Xing, and T. Wu. Electrode dependence of resistive switching in mn-doped zno: Filamentary versus interfacial mechanisms. *Applied Physics Letters*, 96(19), May 2010.

[76] J. D. Greenlee, W. L. Calley, M. W. Moseley, and W. A. Doolittle. Comparison of interfacial and bulk ionic motion in analog memristors. *IEEE Transactions on Electron Devices*, 60(1):427–432, Jan 2013.

[77] F. Nardi, S. Balatti, S. Larentis, D. C. Gilmer, and D. Ielmini. Complementary switching in oxide-based bipolar resistive-switching random memory. *IEEE Transactions on Electron Devices*, 60(1):70–77, Jan 2013.

[78] D. Ielmini. Modeling the universal set/reset characteristics of bipolar rram by field- and temperature-driven filament growth. *IEEE Transactions on Electron Devices*, 58(12):4309–4317, Dec 2011.

[79] S. Ambrogio, S. Balatti, D. Ielmini, and D. C. Gilmer. Analytical modelling and leakage optimization in complementary resistive switch (crs) crossbar arrays. In *2014 44th European Solid State Device Research Conference (ESSDERC)*, pages 242–245, Sep 2014.

[80] S. Ambrogio, S. Balatti, D. C. Gilmer, and D. Ielmini. Analytical modeling of oxide-based bipolar resistive memories and complementary resistive switches. *IEEE Transactions on Electron Devices*, 61(7):2378–2386, Jul 2014.

[81] Stefano Ambrogio, Simone Balatti, Seol Choi, and Daniele Ielmini. Impact of the mechanical stress on switching characteristics of electrochemical resistive memory. *Advanced Materials*, 26(23):3885–3892, Mar 2014.

[82] Bernabe Linares-Barranco, Teresa Serrano-Gotarredona, Luis Camunas-Mesa, Jose Perez-Carrasco, Carlos Zamarreno-Ramos, and Timothee Masquelier. On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex. *Frontiers in Neuroscience*, 5:26, Mar 2011.

[83] Giacomo Indiveri, Bernabe Linares-Barranco, Tara Julia Hamilton, Andre van Schaik, Ralph Etienne-Cummings, Tobi Delbruck, Shih-Chii Liu, Piotr Dudek, Philipp Häfliger, Sylvie Renaud, Johannes Schemmel, Gert Cauwenberghs, John Arthur, Kai Hynna, Fopefolu Folowosele, Sylvain Saighi, Teresa Serrano-Gotarredona, Jayawan Wijekoon, Yingxue Wang, and Kwabena Boahen. Neuromorphic silicon neuron circuits. *Front. Neurosci*, 5, 2011.

[84] Matthew D. Pickett, Gilberto Medeiros-Ribeiro, and R. Stanley Williams. A scalable neuristor built with mott memristors. *Nature Materials*, 12(2):114–117, Dec 2012.

[85] Anup Vanarse, Adam Osseiran, and Alexander Rassau. A review of current neuromorphic approaches for vision, auditory, and olfactory sensors. *Front. Neurosci.*, 10, mar 2016.

[86] H. D. Crane. The neuristor. *IEEE Transactions on Electronic Computers*, EC-9(3):370–371, Sep 1960.

[87] Tetsu Saigusa, Atsushi Tero, Toshiyuki Nakagaki, and Yoshiki Kuramoto. Amoebae anticipate periodic events. *Phys. Rev. Lett.*, 100(1), Jan 2008.

[88] Yuriy V. Pershin, Steven La Fontaine, and Massimiliano Di Ventra. Memristive model of amoeba learning. *Phys. Rev. E*, 80:021926, Aug 2009.

[89] L. O. Chua and L. Yang. Cellular neural networks: theory. *IEEE Transactions on Circuits and Systems*, 35(10):1257–1272, Oct 1988.

[90] L. O. Chua and L. Yang. Cellular neural networks: applications. *IEEE Transactions on Circuits and Systems*, 35(10):1273–1290, Oct 1988.

[91] Themistoklis Prodromakis, Boon Pin Peh, Christos Papavassiliou, and Christofer Toumazou. A versatile memristor model with nonlinear dopant kinetics. *IEEE Trans. Electron Devices*, 58(9):3099–3105, Sep 2011.

[92] Julia J. Harris, Renaud Jolivet, and David Attwell. Synaptic energy use and supply. *Neuron*, 75(5):762–777, Sep 2012.

[93] Simon B. Laughlin, Rob R. de Ruyter van Steveninck, and John C. Anderson. The metabolic cost of neural information. *Nat. Neurosci.*, 1(1):36–41, May 1998.

[94] J. Rosing and E.C. Slater. The value of g for the hydrolysis of ATP. *Biochimica et Biophysica Acta (BBA) - Bioenergetics*, 267(2):275–290, May 1972.

[95] Ioannis Vourkas and Georgios Ch. Sirakoulis. Nano-crossbar memories comprising parallel/serial complementary memristive switches. *BioNanoScience*, 4(2):166–179, Jun 2014.

[96] hines. Non-rectifying electrical synapses. https://www.neuron.yale.edu/phpBB/viewtopic.php?f=12&t=25&p=619&#p619, November 21 2005.

[97] Mary M. Heinricher. Principles of extracellular single-unit recording. In Zvi Israel and Kim J. Burchiel, editors, *Microelectrode Recording in Movement Disorder Surgery*, chapter 2, page 11. Thieme Publishing Group, 2004.

[98] C. Gold. On the origin of the extracellular action potential waveform: A modeling study. *Journal of Neurophysiology*, 95(5):3113–3128, Jan 2006.

[99] Myoung-Jae Lee, Chang Bum Lee, Dongsoo Lee, Seung Ryul Lee, Man Chang, Ji Hyun Hur, Young-Bae Kim, Chang-Jung Kim, David H. Seo, Sunae Seo, U-In Chung, In-Kyeong Yoo, and Kinam Kim. A fast, high-endurance and scalable non-volatile memory device made from asymmetric ta2o5-x/TaO2-x bilayer structures. *Nature Materials*, 10(8):625–630, Jul 2011.

[100] Yang Yin Chen, B. Govoreanu, L. Goux, R. Degraeve, A. Fantini, G. S. Kar, D. J. Wouters, G. Groeseneken, J. A. Kittl, M. Jurczak, and L. Altimime. Balancing SET/RESET pulse for > 10^{10} Endurance in HfO_2/Hf 1t1r Bipolar RRAM. *IEEE Trans. Electron Devices*, 59(12):3243–3249, dec 2012.

[101] Harold C Burger, Christian J Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2392–2399. IEEE, 2012.

[102] Andrew Maas, Quoc V Le, Tyler M O'neil, Oriol Vinyals, Patrick Nguyen, and Andrew Y Ng. Recurrent neural networks for noise reduction in robust asr. 2012.

Appendix 1 - Pseudo-code of memristor implementation in NEURON

Object initialization {
  Type of object memristive device
  Pointer pre- and post-synaptic neurons
}

Parameters {
  pre-synaptic neuron's voltage
  post-synaptic neuron's voltage
  extra arguments
}

States {
  device state
}

Initial {
  initial configuration
}

Numerical equations {
  memristor's model
}

Exit point of object {
  Current output from model equations
}

Algorithm 1: Memristive device model

Algorithm 1 is the generic template for implementing a memristive (or other electric elements) in NEURON, via NMODL syntax, with the electrodes of this module is implemented by another template in HOC.

The block of the template **Numerical equations** { ... } contains the numerical model that defines the memristive device. In this block, flux-charge and voltage-current devices are supported.

The following block of code are illustrative of the typical code structure in NEURON (in version 7.4), implementing an hybrid chain of HH neurons and flux-charge memristors defined by a numerical table of sufficiently continuous experimental data.

```
// preparatory lines
```

```
load_file("tabulating.hoc")
load_file("memo_template.hoc")

/*
SETUP:

    ( 1 )---( memr_1 ) -...- ( memr_N-1 )---( N )

1 to N are HH neurons (single compartement) and due to coding
SectionRefs need to created and used instead of the original objects

memr is an "real" memristive device
on user level, new memo(neuron A, neuron B) is the one necessary
    command needed.
Here'swhat's happening:
- placeholder neuron (named inter) is created
- inserts a density mechanism that integrates voltage difference
    between neuron A and B, corresponds a charge to flux value
- that charge is derivated in time within .mod (with reassigment of
    states)

- placeholder neuron is not necessary, but is useful for piggybacking
*/



//set of N neurons
N = 10
create soma[N]
forall { L=20 diam=20 insert hh }
//NECESSARY sectionrefs
objref s[N]
for i=0,N-1 {soma[i] s[i] = new SectionRef()}

//memristor objects
objref memr[N-1]
//memo(A,B,x,y) creates the memristor object
//x and y are scaling parameters dependent on the experimental data
for j=0,N-2 {memr[j] = new memo(s[j],s[j+1],1e10,7e2)}
```

```
//stimulus for one the neurons, square wave between 0 and 0.5 nA
objref electrode
s[0].sec electrode = new duty_cycle_electrode(0.5)
{electrode.period = 100}        //period is in milliseconds


//creation of recording objects and of file to store data
objref data
data = new File("dynamic.txt")

objref time, voltage[N]
time = new Vector()
time.record(&t)
for i=0,N-1 {
        voltage[i] = new Vector()
        voltage[i].record(&soma[i].v(0.5))
}

//total simulation time (milliseconds)
tstop = 1000
run()    //

//writing results to file
wopen("dynamic.txt")
for i=0,tstop/dt-1 {
        fprint("%f\t",time.x[i])
        for j=0,N-1 {
                fprint("%f\t",voltage[j].x[i])
        }
        fprint("\n")
}
wopen()
```

Listing 7.1: HOC level implementation of hybrid chains, with flux-charge devices

```
NEURON {
        SUFFIX intgr_flux
        POINTER v_p, v_m, scale_f, scale_q
}
```

```
PARAMETER {
        v_p     ()
        v_m     ()
        scale_f ()
        scale_q ()
}

STATE { chrg_t0 chrg_t0_dt flx t0 t0_dt current resistance div_v }

INITIAL {
        chrg_t0 = 0
        chrg_t0_dt = 0
        flx = 0
        t0 = 0
        t0_dt = 0
        current = 0
        resistance = 0
        div_v = 0
}

BREAKPOINT {
        SOLVE state METHOD cnexp
}

FUNCTION_TABLE c(x)

UNITSOFF
DERIVATIVE state {

        div_v = (v_p - v_m)/1000

        flx = (v_p - v_m)/1000

        chrg_t0_dt = chrg_t0
        chrg_t0 = c(scale_f*flx)/scale_q

        t0_dt = t0
        t0 = t
```

```
        current = ( chrg_t0 - chrg_t0_dt )/( t0 - t0_dt )

        if (current <= 1e-12) {
                resistance = resistance
        }else{
                resistance = (v_p - v_m)/(1000*current)
        }

}
UNITSON
```

Listing 7.2: Example of algorithm 1 flux-charge device

Appendix 2 - Voltage divider

```
# -*- coding: utf-8 -*-
"""
Created on Tue Mar 01 15:50:45 2016

@author: Joao_Alexandre
"""

#necessary libraries
from numpy import *
from scipy import signal
from pylab import *
from scipy.integrate import ode

#device parameters
#r_on = inverse of low resistance state
#r_off = inverse of high resistance state
#n = non-linearity of state variable
#a = constant in state variable differential equation (V^-1 s^-1)
#p = exponent in Joglekar window function
#m = non-linearity in input of state variable differential equation
r_on = array([18,18])
r_off = array([0.04,0.04])
n = 14
a = 4
p = 1
m = 1


amp = 3                          #amplitude of input signal (V)
pol = array([1,1])       #polarization of devices

#initialization of time array
t_init = 0
t_fin = 1.2
dt = 0.001
time=arange(t_init,t_fin,dt)
```

```
#input signal
period = 0.5
voltage = amp*(signal.square(2*pi*time/period,duty=0.5))

#state variable differantial equation
def kernel(t,x):
    return a*(1-(2*x-1)**(2*p))*(pol*(r_off+(r_on-r_off)*x**n)**(-1)*
       voltage[t/dt]/sum((r_off+(r_on-r_off)*x**n)**(-1)))**m

#initialization of solution arrays and iterative progression in time
   array
y_init = array([0.85,0.9])
r = ode(kernel)
r.set_integrator('dopri5')
r.set_initial_value(y_init, t_init)
y_dom = zeros((len(time),2))
y_dom[0,:] = y_init
for j in range(1,len(time)):
    y_dom[j,:] = r.integrate(r.t+dt)


#plotting of results and export to file (in directory of script)
figure(figsize=(10,5))
current = voltage/sum((r_off+(r_on-r_off)*y_dom**n)**(-1))
subplot(221)
plot(time,y_dom[:,0],'b',label='x1␣(adimensional)')
plot(time,y_dom[:,1],'r',label='x2␣(adimensional)')
grid()
legend(loc='upper␣center', bbox_to_anchor=(0.5, 1.45),ncol=1)
xlabel('time␣/␣seconds',fontsize=15)
ylim([-0.1,1.1])


subplot(222)
v1=current*(r_off[0]+(r_on[0]-r_off[0])*y_dom[:,0]**n)**(-1)
v2=current*(r_off[1]+(r_on[1]-r_off[1])*y_dom[:,1]**n)**(-1)
maxi = max(max(abs(v1)),max(abs(v2)))
plot(time[::50],current[::50]/max(abs(current)),'.',\
label='I␣/␣'+str(round(max(abs(current)),4))+'␣A',markersize=1)
```

```
plot(time,voltage/max(abs(voltage)),'--',\
label='V␣/␣'+str(round(max(abs(voltage)),5))+'␣V')
plot(time,v1/maxi,'b',label='V1␣/␣'+str(round(maxi,3))+'␣V')
plot(time,v2/maxi,'r',label='V2␣/␣'+str(round(maxi,3))+'␣V')
ylim([-1.1,1.1])
legend(loc='upper␣center', bbox_to_anchor=(0.5, 1.5),ncol=2)
xlabel('time␣/␣seconds',fontsize=15)
grid()

savefig("biphasic_art_spike.png",dpi=800,bbox_inches='tight')
```

Listing 7.3: Simulation of memristive voltage divider (with 2 linearized devices); code ran on Anacoda 1.1.0 distribution