Università degli Studi di Udine

Dipartimento Politecnico di Ingegneria e Architettura (DPIA)
PhD in Electronic Engineering

Thesis For The Degree Of Doctor Of Philosophy

Spiking Neural Networks and Ferroelectric Based Devices for Neuromorphic Computing

Advisors
Prof. Mirko Loghi
Prof. David Esseni

Candidate
Riccardo Fontanini
119926

Academic Year 2019-2022 (XXXV cycle)

Ad Aurora...

Abstract

Thanks to the rise of *Internet of Things (IoT)* and wearable electronics, smart sensors and low-power edge systems are becoming increasingly prevalent in our daily lives. In particular, the pursuit of more interactive and intelligent systems pushed research and industry toward the integration of non-conventional algorithms into electronic appliances. Consequently, *Artificial Intelligence (AI)* and deep learning, have been proposed as a solution to a multitude of algorithmic-difficult problems, such as facial and speech recognition, sentiment analysis, text synthesis, autonomous driving, etc. However, edge devices, due to their limited dimensions and serious energy constraints, present some limitations for the elaboration of deep neural networks. Hence, a popular strategy is to send the raw information acquired by the low-power devices to the cloud and wait for a processed response, instead of performing the computation locally. However, this approach increases the computational overhead of cloud servers, it leads to a quite long response latency and, moreover, it fails when the internet connection is not available. An alternative solution consists in moving computational capabilities directly to the edge through a distributed computing network, which can locally process data directly on its nodes. For edge computing, however, the energy budget required by *Deep Neural Networks (DNNs)* running on conventional elaboration systems is problematic. Therefore, because a large fraction of the energy is dissipated by moving data back and forth from the memory, significant efforts are being devoted to overcome the so-called *von-Neumann bottleneck*. In such a context, *neuromorphic computing* has been developed to improve *AI* energy requirements by exploiting biologically-inspired neural networks. This new *AI* branch exploits *Very Large Scale Integration (VLSI)* analog circuits to implement *Spiking Neural Networks (SNNs)* in hardware, thus closely mimicking the biological power reduction strategies. This thesis aims to investigate and model neuromorphic solutions for more energy-efficient *AI* applications.

In particular, we observed in deep *SNNs* that the average spike rate tends to increase with the number of layers, leading to a decreased energy efficiency during both the learning and the inference phases. In order to contrast this behavior, measures must be taken to control the spike rate without inducing a large number of silent neurons, which do not emit spikes and do not contribute to the training process. Therefore, we present a 2-phase training strategy for deep feed-forward *SNNs*: our approach modifies the loss functions and introduces two phases of training to reduce the spike rate and address the silent neuron issue. Moreover, we also examined the most important circuital implementations of *SNNs* and neuromorphic platforms to understand the challenges and the current state of the art of this topic.

Then, this thesis delves into the design techniques for ferroelectric-based memristors and their applications in neuromorphic devices, specifically focusing on *Ferroelectric Tunnel Junctions (FTJs)* as promising devices for implementing synaptic-like capabilities. In particular, we merged a model for the polarization dynamics in *Metal-Ferroelectric-Insulator-Metal (MFIM)* structures, with a novel charge-trapping model, in order to investigate the relationship between ferroelectric polarization and charge trapping in the dielectric stack. Our simulation results, calibrated against experiments, present evidence that the partial compensation of the ferroelectric polarization due to trapped charges strongly influences the operation of $Hf_{0.5}Zr_{0.5}O_2$ (HZO) based *FTJs* .

The red thread linking the activities on the training of *SNNs* to those on *FTJs* based devices is the improvement of the energy efficiency in neuromorphic systems.

Contents

List of Figures v

List of Tables vii

Acronyms viii

1 Introduction 1
- 1.1 Artificial Intelligence . . . 1
- 1.2 Energy constraints for Edge AI applications . . . 6
- 1.3 Structure and purpose of this work . . . 9

2 Spiking Neural Networks 10
- 2.1 Introduction to SNNs . . . 10
  - 2.1.1 Neural models . . . 11
  - 2.1.2 Learning rules . . . 15
- 2.2 Reducing the Spike Rate in Deep Spiking Neural Networks . . . 18
  - 2.2.1 Models . . . 20
  - 2.2.2 SNN simulator . . . 24
  - 2.2.3 Benchmarks . . . 27
  - 2.2.4 Experimental Results . . . 28
  - 2.2.5 Neuron to neuron variability on MNIST benchmark . . . 36

3 Overview of neuromorphic hardware implementations 42
- 3.1 Devices and Circuits for neuromorphic applications . . . 42
- 3.2 From synapses to memristor device concepts . . . 46
- 3.3 BEoL compatible memristors . . . 50

4 Memristors based on Ferroelectric Tunnel Junctions 52
- 4.1 A brief introduction to ferroelectric materials . . . 53
- 4.2 FTJs device concept . . . 57
- 4.3 Modelling of FTJs . . . 59
  - 4.3.1 Single-Domain MFIM structure . . . 59
  - 4.3.2 Multi-Domain MFIM structure . . . 62
  - 4.3.3 Effects of an uniform charge Q_{DF} at the FE-DE interface . . . 67
  - 4.3.4 Read operation and tunnelling current . . . 69
  - 4.3.5 Trapped charges in the MFIM structure . . . 71
  - 4.3.6 Tunneling assisted by traps at the FE-DE interface . . . 74

5 Operation and design of Ferroelectric Tunnel Junctions 82
- 5.1 Operation and design of defects free FTJs . . . 83
  - 5.1.1 Model validation and analysis of experimental data . . . 84
  - 5.1.2 Simulation based design of FTJs . . . 85

5.2 Charge trapping induced compensation of the ferroelectric polarization in quasi-static regime . . . 88
5.3 Interplay between charge trapping and polarization switching in BEOL-compatible FTJs . . . 91
5.3.1 Device fabrication and measurements . . . 91
5.3.2 Models and comparison to experiments . . . 92
5.3.3 Implications for FTJ device dynamics . . . 97
5.4 Optimal conditions for the synaptic operation of FTJs . . . 99
5.4.1 Device fabrication and experiments . . . 99
5.4.2 Modelling framework and calibration . . . 101
5.4.3 Evidence of trap contribution in FTJ operation . . . 102
5.4.4 Polarization–compensation–aware design of the FTJ . . . 105

6 Conclusions **109**

A Proof of Event-Prop backpropagation algorithm using adjoint variables in LIF Spiking Neural Networks **113**

B Extension of Event-Prop backpropagation algorithm using adjoint variables in Izhikevich-based Spiking Neural Networks **125**

C List of publications **142**
C.1 Journal papers . . . 142
C.2 Conference papers . . . 143

Bibliography **144**

List of Figures

1.1 History of AI . . . 1
1.2 Example of DNN . . . 3

2.1 Biological plausibility and computational overhead of neural models . . . 11
2.2 Example of LIF membrane potential waveform . . . 12
2.3 Example of membrane potential and membrane recovery variable of Izhikevic model 15
2.4 Sketch of network and neural state function in time . . . 19
2.5 Yin-Yang - Comparison between L_W and L . . . 23
2.6 Sketch of the implemented algorithm . . . 24
2.7 Sketch of the datasets conversion and example of neural signals. . . . 27
2.8 Exploration of the parameter α for loss L_W . . . 29
2.9 Yin-Yang - Example of switch-off . . . 29
2.10 Yin-Yang - Average number of output spikes per neuron for different α and η . . . 30
2.11 Yin-Yang - Median accuracy after 100 epochs of training. This plot shows the impact of α and η on the network's training by exploiting two different loss functions: L and L_A. . . . 31
2.12 Yin-Yang - Comparison between the total loss and the CE component of L and LA 32
2.13 Yin-Yang - Summary of the training methodologies . . . 33
2.14 Accuracy and cross-entropy for test and train sets in the MNIST case . . . 34
2.15 Behavior of the cross-entropy during the initial 150 training epochs . . . 35
2.16 Behavior of the cross-entropy in the second phase of training . . . 35
2.17 Final training results of MNIST. . . . 37
2.18 Comparison between the overall and single layer components for the average number of spikes . . . 38
2.19 MNIST - Final accuracy after the first and the second phase of training for multiple topology . . . 38
2.20 MNIST - Mean accuracy after the 2nd phase of training in presence of constants mismatch . . . 40
2.21 MNIST - 1st and 2nd phase of training accuracy results of an SNN classifier . . . 41

3.1 Kraken die . . . 43
3.2 Exponential I&F model . . . 45
3.3 Cross-bar array implementation . . . 47
3.4 Circuit of Vector-Matrix cross-bar multiplier array . . . 48
3.5 Sketch pinched I-V . . . 49
3.6 Sketch of memristive devices . . . 50

4.1 Steady states points of LGD equation . . . 53
4.2 Energy configuration of LGD polynomial . . . 55
4.3 Cross-section of an MFM based FTJ . . . 57
4.4 Single domain Ferroelectric capacitor . . . 59
4.5 Ferroelectric capacitors and related symbols . . . 62
4.6 Energy band diagram across the MFIM stack . . . 69

4.7 Sketch of charge injection fluxes . . . 74
4.8 Steady state equilibrium occupation . . . 80

5.1 Sketch of an MFIM based FTJ . . . 83
5.2 Simulated polarization versus ferroelectric voltage characteristic . . . 84
5.3 Examples of setting and reading waveforms and corresponding fractions f_{UP} . . . 85
5.4 Read current at $V_R = 2$ V . . . 86
5.5 Read current at $V_R = 2$ V versus the Al_2O_3 thickness for an HZO/Al_2O_3 . . . 86
5.6 Read current and R_I versus the SET voltage for the TiN/HZO/Al_2O_3/TiN structure 87
5.7 Simulated spontaneous polarization and interface charge . . . 89
5.8 Percentage of domains with a positive polarization versus the charge compensation produced by interface traps . . . 89
5.9 High resolution TEM cross section of the TiN/1.5 nm Al_2O_3/10 nm HZO/TiN devices 91
5.10 2θ scans taken in grazing-incidence geometry . . . 93
5.11 Polarization versus voltage characteristics measured by the PUND method . . . 95
5.12 Polarization and interface charge versus time . . . 96
5.13 Electric field across the Al_2O_3 layer during the P pulse analyzed in Fig. 5.11(a), hence for t_D=1.5 nm. . . . 96
5.14 Measured and simulated P-V curves for a PUND waveform as in for an Al_2O_3 FTJ with t_D=2.0 nm . . . 97
5.15 Polarization and interface charge during an N pulse and the following retention phase 97
5.16 FTJ current-voltage characteristics measured at different delay times after a set operation . . . 98
5.17 Sketches of the Stack Anneal–Bottom and Post Anneal–Deposition . . . 99
5.18 Examples for the band diagram across the FE–DE stack in read mode . . . 100
5.19 Simulated and experimental charge vs. electric field characteristic for a TiN/HZO/TiN device . . . 101
5.20 P–V characteristics for SAB FTJs . . . 102
5.21 I–V curves . . . 102
5.22 Comparison between simulations and experiments for the P–V curves for PAD devices 103
5.23 Simulated polarization and trapped interface charge for the SAB and PAD simulations 104
5.24 Measured current density for a SAB device at a read condition . . . 104
5.25 Minimum, $I_{R,min}$, and maximum, $I_{R,max}$, read currents . . . 105
5.26 compensated polarization vs. the trap density in read condition . . . 106
5.27 HZO conduction band minimum in read mode . . . 107
5.28 Current simulation results . . . 108

6.1 Spiking Autoencoder . . . 111

List of Tables

2.1 Model's parameters 25
2.2 Ranges of weights initialization 28
2.3 Best results in terms of accuracy for Yin-Yang 32
2.4 Best results in terms of accuracy for MNIST 36

5.1 Material parameters used in simulations 83
5.2 Simulations cross-sections and other parameters 88
5.3 Material parameters employed in simulations 94
5.4 Material parameters related to tunneling and trappin 94
5.5 Trapping cross sections used in simulations 103

Acronyms

AER Address-Event Representation

AI Artificial Intelligence

ANN Artificial Neural Network

BEOL Back end of Line

DNN Deep Neural Network

EIF Exponential Integrate-and-Fire

FE-DE Ferroelectric-Dielectric

FeFET Ferroelectric FET

FTJ Ferroelectric Tunnel Junction

HZO $Hf_{0.5}Zr_{0.5}O_2$

IoT Internet of Things

LIF Leaky Integrate and Fire

M2M Machine-to-machine

MD interface between the dielectric and the metal gate

MF interface between the ferroelectric and the metal gate

MFIM Metal-Ferroelectric-Insulator-Metal

MFM Metal-Ferroelectric-Metal

MFS Metal-Ferroelectric-Semiconductor

NC Negative Capacitance

PUND Positive-Up-Negative-Down

SNN Spiking Neural Network

SoC System-On-Chip

STDP Spike Time Dependent Plasticity

TER Tunneling Electro-Resistance

VLSI Very Large Scale Integration

Chapter 1

Introduction

1.1 Artificial Intelligence

What's the history of artificial intelligence?

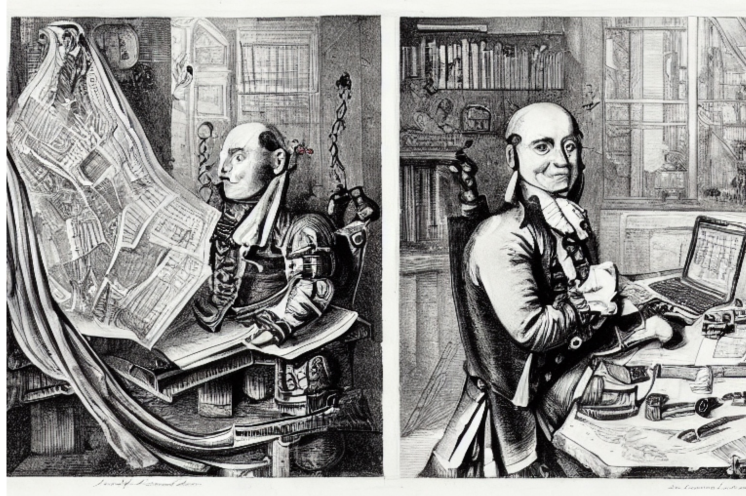

Figure 1.1: "History of AI" - Example of image automatically generated by a Text-to-Image AI.

The history of AI *can be traced back to the 1950s, when researchers first began exploring the idea of creating machines that could perform tasks that typically required human intelligence. This field of research has gone through several phases, each marked by different approaches, techniques and technologies, and here is a summary of the main stages:*

1. *Early* AI *(1940s-1970s): The early years of* AI *research focused on developing simple rule-based systems, also known as "expert systems", which could perform specific tasks such as diagnosing medical conditions or playing chess. These systems were based on the idea of encoding human expertise into a set of rules that the computer could follow [1, 2].*

2. *Symbolic* AI *(1970s-1980s): Researchers began to explore the use of symbolic representations and logic-based reasoning to create* AI *systems. This approach relied on representing knowledge in a formal symbolic language and using logical inference to deduce new knowledge [3, 4].*

3. *Connectionist* AI *(1980s-1990s): Connectionist* AI*, also known as neural networks, focused on developing systems that could learn from examples. This approach was inspired by the structure and function of the human brain and aimed to create computer systems that could "learn" from data in a way that was similar to how the brain learns [5, 6].*

4. *Machine Learning (1990s- present): Machine learning is a subset of* AI *that focuses on developing algorithms and models that can learn from data. This approach has become increasingly popular in recent years, with the advent of large datasets, powerful computing resources and the development of deep learning techniques [7–10].*

5. *Current* AI *(2010s- present): The recent years of* AI *research have seen a continued focus on machine learning, particularly deep learning, which has been used to achieve breakthroughs in a wide range of applications, including image and speech recognition, natural language processing, and decision making. Additionally, there is a growing interest in the field of neuromorphic computing, which is inspired by the structure and function of the human brain and aims to create highly efficient, low-power computing systems [11–15] .*

What's deep learning and artificial neural networks?

Deep learning is a subset of machine learning that is based on Artificial Neural Networks (ANNs)*. An* ANN *is a mathematical model that is inspired by the structure and function of the human brain and is designed to "learn" from data.*

An ANN *is composed of layers of interconnected "neurons", which are simple processing units that receive input, process it, and produce an output. These layers are interconnected, allowing information to flow through the network, and the connections between the neurons can be adjusted during training to improve the network's performance.*

Deep learning refers to the use of multiple layers in an ANN*, also known as* DNN*. Each layer in a* DNN *is designed to extract increasingly complex features from the input data, allowing the network to learn more abstract representations of the data [11, 16–20]. This allows* DNNs *to perform tasks that are difficult or impossible for traditional machine learning algorithms, such as image and speech recognition, natural language processing, and decision making [14, 19, 21].*

Deep learning models are trained using a large dataset and a process called backpropagation, where the model adjust the weights of the connections between the neurons to minimize the error between the output and the desired outcome.

Deep learning has been responsible for many recent breakthroughs in AI*, particularly in the areas of computer vision, natural language processing and speech recognition. It's widely used in many applications such as self-driving cars, image and speech recognition, and natural language processing (NLP). [22–24]*

What is the actual limit of *AI* and how it could evolve?

The limits of AI *are constantly being pushed as the field advances and new breakthroughs are made. However, there are still many challenges and limitations that need to be overcome in order for* AI *to reach its full potential.*

1. *General Intelligence: One of the main goals of* AI *is to create general intelligence, which is the ability to perform a wide range of cognitive tasks that are currently only possible for humans. However, this is a very difficult task, and current* AI *systems are still limited in their abilities and are not yet able to perform all of the tasks that humans can.*

2. *Human-like understanding: Another goal of* AI *is to create systems that can understand the world in a way that is similar to how humans do. For example, understanding natural language, recognizing objects and scenes, and making decisions based on incomplete or uncertain information. However, current* AI *systems still lack the ability to understand the world in the same way as humans.*

3. *Unstructured data: Most of the current* AI *systems are designed to work with structured data, such as images and text that have been labeled and organized in a specific way. However, most of the data in the world is unstructured and difficult to process, making it hard for* AI *to extract useful information from it.*

4. *Human-AI interaction: Another challenge is to create* AI *systems that can interact with humans in a natural and intuitive way, for example through natural language, gestures, and facial expressions.*

5. *Explainability: One of the main challenges is to make* AI *systems explainable, meaning that they can justify their decisions and predictions to humans in a way that humans can understand.*

6. *Safety and Ethics: Another challenge is to ensure that* AI *systems are safe and operate ethically. This includes preventing them from causing harm to humans, making sure they respect privacy, and ensuring that they are fair and unbiased.*

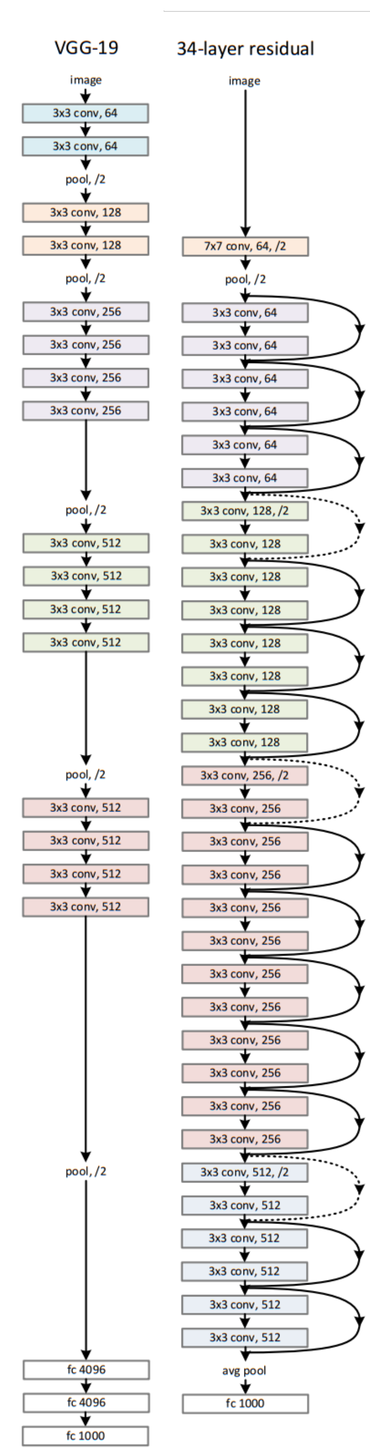

Figure 1.2: Layer depth comparison between a VGG19 DNN (on the left) and a deep residual network (on the right) in [16].

Despite these challenges, AI *is expected to continue to evolve and make progress in these areas. Advances in areas such as deep learning, natural language processing, and computer vision, as well as the development of specialized hardware such as neuromorphic chips, are expected to drive further progress in the field.*

What do you know about neuromorphic computing?

Neuromorphic computing is a form of computing that is inspired by the structure and function of the human brain. It is based on the idea of creating computer systems that can mimic the way the brain processes information, rather than relying on traditional Von Neuman architecture [25, 26].

A neuromorphic system is composed of many simple processing elements, called "neurons", that are connected to each other by "synapses". These elements work together to process information in a way that is similar to the way neurons and synapses in the brain process information.

Neuromorphic computing systems are designed to be highly energy-efficient, and to be able to perform complex computations in real-time, even with limited resources. Currently, researchers are developing neuromorphic chips, which are specialized hardware that can be used to build neuromorphic computing systems. These chips typically consist of thousands or even millions of simple processing elements, and are designed to be highly energy-efficient. They can be used to develop advanced AI *applications such as object recognition, speech recognition, and anomaly detection.*

Neuromorphic computing is still a relatively new field, and much research is still ongoing to improve the performance and capabilities of these systems. It is expected that as the technology matures, it will be used to create powerful new AI *systems that can perform tasks that are currently beyond the capabilities of traditional computers.*

The previous paragraphs were not written by the author nor by any human being, but by an *AI* instead. In particular, we exploited a natural language processing architecture called *ChatGPT*[1] to answer a few questions simply reported in the text its replies. In particular, this *AI* defines itself as:

"I am ChatGPT, a large language model developed by OpenAI. I am trained to understand and generate human-like text and I am able to assist users by providing helpful and accurate information. I can answer questions, generate text, and complete text prompts, among other things. My knowledge cutoff is 2021 and I am constantly learning and updating my knowledge base to provide better assistance. I am here to help you with any information or question you may have, feel free to ask me anything."

Who is writing found the result of this little experiment very impressive. We didn't modify

[1] ChatGPT is developed by OpenAI and it can be interrogated at https://chat.openai.com/

anything of the answers generated by the *AI*: we just copied and pasted the output of the neural model in this section. We also verified the assertions made by ChatGPT adding some references and figures to also improve the readability of this introductive part. The aim of this experiment was to introduce the baseline concepts of deep learning and also to give an idea of the level of complexity *AI* reached nowadays. Moreover, we interrogated ChatGPT with another question:

How much energy is needed to train an artificial neural network?

> *The energy consumption of training a neural network can be significant, as it requires large amounts of computational resources and energy. The exact amount of energy consumed during training depends on several factors, including:*
>
> 1. *The size of the model: Larger models require more computational resources to train, which in turn consumes more energy.*
> 2. *The complexity of the model: More complex models, such as deep neural networks, require more computational resources to train than simpler models.*
> 3. *The size of the training dataset: The larger the training dataset, the more data needs to be processed, which in turn consumes more energy.*
> 4. *The hardware used to train the model: The energy consumption of training a neural network can vary depending on the type of hardware used, such as CPU, GPU or TPU.*
> 5. *The Training algorithm: The energy consumption of training a neural network can vary depending on the algorithm used to train the model, for example, Backpropagation algorithm is used to train deep learning models and it is computationally intensive [27].*
>
> *It has been estimated that the energy consumption of training a large deep learning model can be equivalent to the energy consumption of several households for a year. However, researchers and companies are working on reducing the energy consumption of training* AI *models by developing more energy-efficient hardware and training algorithms. For example, using neuromorphic computing or Approximate computing are some of the methods that have been proposed to reduce the energy consumption of training* AI *models.*

In fact, in this Ph.D. project we aimed to pursue new strategies for power reduction in *AI* applications and, because neuromorphic computing is centered on energy saving to emulate the efficiency of biological systems, we focused our efforts on two different abstraction levels: by considering the *SNNs* concept, we first proposed a 2-phase learning procedure to reduce power consumption induced by an overexcited neural activity. Then we focused on the synaptic level by modeling and designing *Back end of Line (BEOL)* compatible *FTJs* suitable for the area and power reduction in *AI* and, in particular, neuromorphic platforms.

1.2 Energy constraints for Edge AI applications

Every day, we are surrounded by smart sensors and intelligent systems, in fact, with the rise of *IoT* and wearable electronics, our daily lives are pervaded by low-power electronic appliances. These are able to understand our habits and our needs in order to improve our lifestyles as stand-alone devices or as part of our smartphones, home automation systems, last-generation cars, healthcare appliances, video surveillance, etc. In general, the last CISCO Annual Internet Report [28] informs us that, every year, various new devices with increased capabilities and intelligence are introduced and adopted in the market. By 2023, *Machine-to-machine (M2M)* connections will be half of the total devices and connections. All these devices produce a huge amount of data that, in general, is not stored locally due to the memory constraint of *IoT* platforms, but it is collected in enormous reservoirs in the cloud. Just at that point, the data is elaborated and the result is sent back to the target device. Such an off-chip communication limits the power efficiency of the *edge devices* and, in addition, increases the computational overhead of the cloud servers. A possible solution is to move the computational capability to the edge, by creating a distributed computing network that is able to locally process data directly on each node of the network. However, the general-purpose architecture of edge computing machines does not take into consideration the nature of the processed data. This may result in inefficiency in terms of power, area, and latency [29]. In fact, standard von-Neumann architectures implement a physical separation between memory and processing unit, consequently generating a waste of power during the information exchange between these two entities. Therefore, to solve the von-Neumann bottleneck, a change in computational paradigm appears to be the best approach to deal with such a challenging landscape [25, 26]. This deviation in point of view does not consider only the computational platform, but it should take into account the whole stack of technology and, more specifically, novel algorithms, storage units and electronic implementations. Moreover, this new approach in computation is called to solve new class of problems that in many cases present fuzzy inputs and unknown results, such as habit clusterization, gesture and facial recognition, self-driving, threats estimation and prediction and many others [30–35]. Hence, taking inspiration from mammals' brain biology, data scientists are now able to replicate astonishingly well some human abilities by using conventional computational platforms and *ANNs*.

ANNs have reported great success in application domains that are difficult to be tackled with an algorithmic approach, such as text and speech recognition, classification and segmentation of images, and robotic control [11, 36–38]. The inter-neural communication in *ANNs* occurs through continuous activation values produced by non-linear but differentiable functions, and the training of the networks targets the minimization of a loss function that is also differentiable. Consequently, the back-propagation approach is a gradient-based optimization of the computational graph that is both well-established and very effective in deep learning systems. With these training methodologies, *AI* applications exploit *ANNs* and deep learning networks to generalize a specific problem and solve fuzzy tasks. However, training a *DNN* uses massive amounts of data and computing resources, which in turn requires an amount of energy that is not sustainable in terms of global electricity requirements and carbon footprint. For example, it has been estimated that training a recent state-of-the-art *AI* neural network, such as GPT-3 [36], would take more than 27 years worth of processing time on a single standard computer, and that computation would generate over 35000

kg of CO_2 emissions in total, which is more than the average carbon dioxide emission produced by an American human adult activity in two years [39]. This huge amount of energy is extremely larger compared to the one used by our brain[2]. In fact, biological neural networks, due to their high parallelism and recurrency, show an unequaled energy efficiency compared to *ANNs*. As mentioned in the previous paragraph, to reduce this gap, a branch of *AI* called *Neuromorphic Computing* tries to merge our knowledge of biological neural networks with CMOS technology in order to create a novel computational paradigm and overcome the drawbacks of von-Neumann architecture with bio-inspired silicon circuits [41]. The term *neuromorphic* was originally coined by Carver Mead, to describe *VLSI* systems containing electronic circuits that mimic neuro-biological architectures present in the nervous system [42]. Nonetheless, nowadays the original concept morphed to include different technological aspects, such as digital or full-software implementations developed to simulate complicated networks of spiking neurons, as well as memristive-based storage applications to implement energy-efficient artificial synapses.

In particular, now a neuromorphic computing solution can be thought as a combination of two different abstraction levels:

- *SNNs* is a type of *ANN* that is based on the principles of biological neural networks. *SNNs* are different from traditional *ANN* in that they model the behavior of biological neurons more closely and process data in the time domain. In *SNNs*, each neuron is modeled as a simple processing unit that generates an electrical spike (or pulse) when it receives a specific set of inputs from other neurons. These spikes are used to transmit information between neurons representing also the input and output patterns of the network.
- *Neuromorphic computing platforms* are electronic implementations of *SNNs* which are inspired by the structure and function of the human brain and are designed to process information in a way that is similar to how biological neurons process information. Neuromorphic computing platforms can be implemented using various technologies such as digital circuits, memristors, and optical devices. These technologies allow the development of low-power computing systems, that can be used for a wide range of applications such as image and speech recognition, control systems, wearable device management and many others.

These two parts of neuromorphic computing are intrinsically different from an implementation point of view. In particular, the first one is more related to informatics, data science and biology, while the second one is a mixture of electronics, physics and material science. Despite their differences, these two entities must cooperate to produce an efficient and reliable system in order to obtain smaller, more durable and energy autonomous intelligent devices. The applications related to the neuromorphic field, at least for now, are limited to small systems compared to large *ANNs* employed in the previously mentioned natural language processing or image/video classification or regression tasks. In particular, neuromorphic applications have the potential to stay relatively close to the source of data. This is paramount in those applications where the energy is supplied by a battery or a volatile source, like wearable healthcare devices implementing different kinds of physiological analysis [43], such as:

- *Electrocardiography* (ECG) measures the electrical activity generated by the cardiac tissue. Neuromorphic systems are able to analyze the heart rate and the oscillatory cardiovascular

[2]It has been observed that the maximum amount of power used by our brain is around 20 W [40]

pressure signals acquired directly on the skin, to prevent or reduce the risk of cardiovascular diseases [44, 45].

- *Electroencephalography* (EEG) is a non-invasive monitoring of the brain activity through the skull. It is useful to prevent mental disorders issues, improve driving safety and evaluate emotions [46–48];
- *Electromyography* (EMG) is a technique to acquire and elaborate the electrical activity generated by skeletal muscles. It is used to detect body motion and micro-mobility. This topic, in particular, is linked with robotics and medical prosthesis applications, but also in language recognition and detecting of neuromuscular disorders [49–52];
- *Bioimpedance spectroscopy* (BIS) is a sensing technique to analyze the spectral impedance of the skin or certain parts of animal and human tissues, like a muscle. With this technique, we can analyze mechanical and chemical properties of the tissue under test. But we can estimate also some overall health indicators like the concentration of water or fat in our bodies. It can also be helpful in the early detection of diseases, such as lymphedema, organ ischemia, and cancer [53, 54].

Additional applications for neuromorphic computing are speech recognition, vision and object recognition, sound classfication, robotics, drone tracking, navigation and obstacle avoidance and many others [55–62].

For the abovementioned reasons, in this thesis, we have dealt with energy consumption of neuromorphic systems from two different perspectives. On the one hand, we have tried to obtain the best point of work of a *SNN* in terms of the number of spikes and accuracy. Then we have focused on the reduction of the von-Neumann bottleneck by modeling and designing ferroelectric tunnel junction memristors.

1.3 Structure and purpose of this work

The main goal of this thesis is to present, investigate and model solutions for a more energy-efficient *AI* hardware. From this perspective, we focused on neuromorphic computing as a promising branch of *AI*, which gathers the knowledge of multiple disciplines (e.g. neurobiology, nanoelectronics and applied material science). In order to better describe biological neural signals compared to conventional *ANNs*, and overcome the von-Neumann computational paradigm by reaching the outstanding computational capability and energy efficiency of our human brain. However, as this topic collects different subjects, it is impossible to overview the entire discipline by looking at just one topic.

This thesis begins with chapter 2, where we introduce the *SNNs* concept as a promising biologically plausible alternative to *ANNs* . Therefore, we considered novel methodologies to improve the power efficiency of *SNNs*. In particular, we propose a 2-phase learning procedure to optimize the number of spikes emitted and, at the same time, enhance the accuracy of the network by overcoming the silent neuron issue.

After that, chapter 3 overviews the most important circuital implementations of *SNNs* and neuromorphic platforms, to expose state-of-the-art and some challenges of these physical *AI* implementations. Hence, we observe that a key concept for the progressive reduction of the von-Neumann bottleneck and, consequently, the energy waste in physical neuromorphic circuits, is to pack memristive memories in the *BEOL* by exploiting large cross-bar arrays.

Therefore, we lowered the abstraction level in chapter 4 to develop new design techniques for ferroelectric-based memristors and applications. In particular, we focused on *FTJs* as a promising device concept to implement synaptic-like capabilities in neuromorphic devices. Therefore, in Sec. 4.3 we report the in–house electrostatic model for *MFIM* structures developed in [63]. On top of that, in Sec. 4.3.6, we present a novel interfacial charge trapping model, that we exploited to understand the delicate interplay between ferroelectric polarization and charge trapping inside the structure.

After an extensive calibration again experiments, the numerical model of Sec. 4.3 is exploited for an insightful study of the design of *FTJs* as synaptic devices for neuromorphic networks in Sec. 5.1. Our analysis explains and addresses the tradeoff between the reading efficiency and the effects of the depolarization field during the retention phase.

To shed light on the role of charge trapping at the ferroelectric-dielectric interface, we report firstly a brief introduction to charge trapping in Sec. 5.2 exploiting quasi-static simulation starting from the results obtained in Sec. 5.1. We then improved our understanding on that topic in Sec. 5.3 by matching experiments with theoretical analysis of dynamic polarization switching and charge trapping in *MFIM* structures.

Last but not least, in Sec. 5.4 we present a clear evidence that the polarization compensation due to charge trapping strongly influences the ON/OFF ratio of *FTJs*. Furthermore, we identify and explain compensation conditions that enable an optimal operation for multi–level synaptic devices.

Chapter 2

Spiking Neural Networks

2.1 Introduction to SNNs

In the biological realm, the nervous tissue is one of the main components of animal bodies. It is composed of neurons, microscopic cells that are able to communicate with each other by voltage impulses. Neurons are able to propagate signals in a really efficient manner: they integrate charge in the cellular membrane and, when a threshold is reached, they emit a voltage spike followed by a reset at the initial condition. Nervous cells are connected to one another through synapses, forming the nervous tissue. A synapse is an interconnection between neurons that can modulate the amplitude of the propagated voltage spikes.

This mechanism is only partially modeled in *ANNs*, in fact, these kinds of artificial constructs are not able to integrate over time the ionic current coming from the synapses into a membrane potential and, consequently, they lose the time dependence of their activation function. Moreover, biological neural networks are intrinsically event-driven systems, while *ANNs* do not have this feature. This leads to a limitation of the energy efficiency of *ANNs* compared to their counterpart and, moreover, *ANNs* cannot take advantage of bio-inspired learning methodologies typically employed in the animal world.

Differently from *ANNs*, *SNNs* are networks of artificial spiking neurons. These artifacts are entities whose emission of spikes is ruled by dynamic equations, which model the behavior of the membrane potential of biological nervous cells. Similarly to biological neurons, artificial neurons emit spikes when the membrane potential reaches a threshold. Therefore, networks of spiking neurons, linked together through synapses, are able to mimic much better than *ANNs* the dynamics of biological nervous tissues and of our brain. Due to these features, *SNNs* are defined as the third generation of artificial neural networks [64], which behave as an event-driven computational system, where asynchronous spikes are used for communication among neurons. *SNNs* can solve the same classes of problems addressed by *ANNs*, but they can mimic much better than *ANNs* the behavior of the brain and, hopefully, emulate also the outstanding energy efficiency of the computation in biological systems. The energy efficiency of *SNNs* stems from the sparsity of events in time and from the asynchronous and local nature of the computation, which makes those networks very promising for energy-constrained, edge computing applications, such as IoT, autonomous vehicles, as well as wearable and implantable devices [43]. Another advantage of this form of information processing is the possibility of not only encoding spatial information like traditional *ANNs* do, but also adding

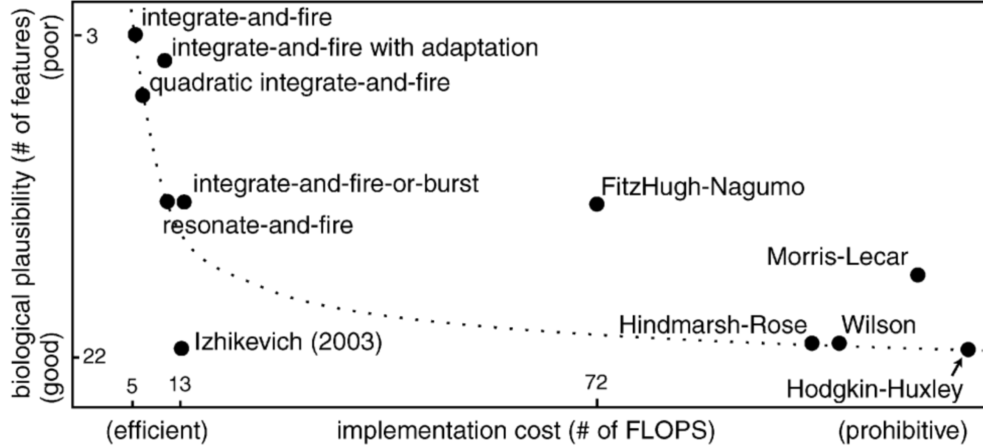

Figure 2.1: Trade-off between biological plausibility and computational overhead of multiple models presented in the literature by [65].

temporal information in the form of the precise timing of spikes. Although the aforementioned goals of *SNNs* represent a solid baseline to study and deepen this topic, developing *SNNs* is also an opportunity to understand the dynamics behind the human brain, and in particular, the still debated learning mechanisms ruling our thoughts.

In this perspective, simulations of *SNNs* represent a constitutive building block of computational neuroscience. Therefore, the whole computation and the applied learning methods should take into account the time dependence of neural state variables.

2.1.1 Neural models

Neuromorphic neural models mimic the membrane potential of biological neurons accordingly with constitutive differential equations. In literature there exist many models of spiking neurons with different degrees of biological accuracy and computational overhead. A more quantitative representation of the aforementioned trade-off is illustrated in Fig. 2.1, while, below, we introduce three relevant models in the neuroscience landscape.

The LIF model

The Leaky Integrate and Fire model (LIF) is the simplest and most widely used model in computational neuroscience. Due to its simplicity, it does not closely match the behavior of biological neurons, but it is largely used in those neuromorphic applications which focus mainly on energy consumption instead of biological plausibility. It was first proposed by Lapicque in 1907 to replicate the neuron membrane potential of biological neurons in terms of an electric circuit consisting of a resistor and capacitor in parallel, representing the leakage and the membrane capacitance, respectively [66]. In this model the membrane capacitor is charged until it reaches a certain threshold, at which time it discharges, producing an action potential (spike). After that, the membrane potential is reset.

The constitutive equation ruling the behavior of this model is:

$$\tau_m \frac{dV}{dt} = -V + RI \tag{2.1}$$

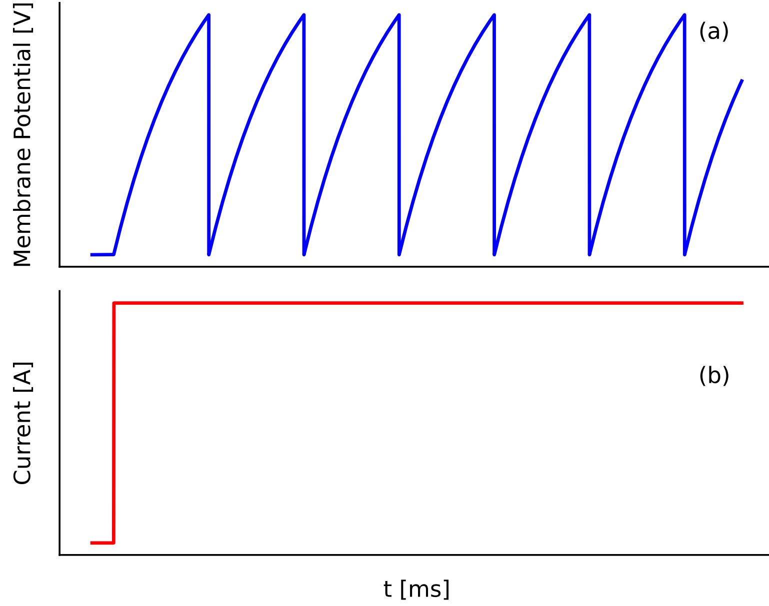

Figure 2.2: (a) Example of LIF membrane potential waveform for the step input synaptic current displayed in (b).

where V is the neural membrane potential, I the input synaptic current, τ_m is the membrane potential time constant and R is the neural input current resistivity. The *Leaky Integrate and Fire (LIF)* model is a simple but powerful model that captures the essential behavior of biological neurons. An example of *LIF* membrane potential waveform, in presence of a constant input synaptic current, is displayed in Fig. 2.2.

A variant of the *LIF* model is the *Exponential Integrate-and-Fire (EIF)* model [67]. It incorporates an exponential current in the dynamics of the membrane potential. The *EIF* model is more complex than the *LIF* model and can capture more realistic behavior of biological neurons.

In particular, the *EIF* model consists of a set of differential equations describing the time evolution of the neuron's membrane potential (V). The equation is given by:

$$\begin{cases} C\dfrac{dV}{dt} = -g_L\left(V - E_L\right) + g_L\Delta_T exp\left(\dfrac{V-\theta}{\Delta_T}\right) - g_C\left(V - E_C\right) - g_I\left(V - E_I\right) - w + I \\ \tau_w\dfrac{dw}{dt} = a\left(V - E_L\right) - w \end{cases} \tag{2.2}$$

where V is the membrane potential, E_L is the resting potential of the neuron, g_L is the leak conductance, Δ_t is the slope factor, θ is the threshold potential, and I is the input current. Moreover, E_I, E_C, g_I and g_C are the potentials and conductances that models the in-vivo excitatory and inhibitory synaptic activity. τ_w is the time constant of the adaptation current w and a represents the level of subthreshold adaptation. At each firing time, the variable w is increased by an amount b, which accounts for spike-triggered adaptation, while the membrane potential is reset to E_L as reported by:

$$\begin{cases} V \rightarrow E_L \\ w \rightarrow w + b \end{cases} \tag{2.3}$$

The membrane potential starts at the resting potential and then changes with time based on the input current and the exponential current term in the equation. The exponential current term captures the non-linear behavior of the membrane potential and allows the model to capture more realistic dynamics of the membrane potential.

The *EIF* model also includes a reset mechanism that allows the membrane potential to be reset to a certain value when it reaches the firing threshold. This reset is similar to the LIF model.

The *EIF* model is a more sophisticated model than the LIF model, it allows the modeling of neurons with different intrinsic dynamics, and it can capture more realistic behavior of biological neurons such as sub-threshold dynamics and spike frequency adaptation. The *EIF* model has been widely used in computational neuroscience and in the development of spiking neural networks and neuromorphic computing platforms.

The Hudgkin-Huxley model

The Hodgkin-Huxley model is a mathematical model of the electrical behavior of neurons that was first proposed by Alan L. Hodgkin and Andrew F. Huxley in 1952 [68]. The model is based on experimental data from the squid giant axon, a type of nerve cell that is large enough to be easily studied using electrical measurements. The model is widely considered as a foundation of neurocomputation, and it is one of the most famous and widely used models in neuroscience.

The model describes the behavior of a neuron by simulating the flow of ions across the cell membrane, which generates an electrical current, and the resulting changes in the membrane potential. The model consists of 25 parameters and 4 dynamical variables that describe the time evolution of the membrane potential, the concentrations of various ions inside and outside the cell, and the gating variables of the ion channels.

The Hodgkin-Huxley model rules the neural membrane potential by simulating the movement of ions through three types of ion channels: the sodium ($Na+$) channels, the potassium ($K+$) channels, and the leak channels. The sodium channels open when the membrane potential is depolarized, allowing $Na+$ ions to flow into the cell, which causes a further depolarization of the membrane potential. The potassium channels open when the membrane potential is hyperpolarized, allowing $K+$ ions to flow out of the cell, which causes a further hyperpolarization of the membrane potential. The leak channels are always open, allowing ions to flow in and out of the cell, which maintains the resting potential of the membrane. The model also includes a concept of "gating variables" that can change with time, which describe the opening and closing of ion channels.

The Hodgkin-Huxley model can reproduce various types of electrical activity such as action potentials, which are the electrical signals that transmit information along the nerve fibers, and subthreshold oscillations, which are the small changes in the membrane potential that occur before a spike.

However, such a model suffers from some limitations:

- it is frequently difficult to understand the nature of the neuronal dynamics observed in this model;
- it contains so many parameters that it is difficult to make a meaningful exploration;
- it cannot be calculated analytically, but only using the results of numerical simulations, which are subject to time discretization errors.

The Izhikevich model

To understand the dynamics behind our brain and the human nervous system we need to find a trade-off between computational efficiency and biological precision. The *LIF* model is the most efficient model in terms of computational overhead, but it neglects lots of neural internal dynamic behaviors, which make the *LIF* model biologically plausible only to a limited extent. In contrast, a model which mimics more closely the correct biophysics of nervous cells, like the Hudgkin-Huxley, contains many equations that make the use of such a model computationally prohibitive in large artificial networks.

In 2003 E. Izhikevich proposed a neural model that can reproduce the firing patterns of neurons recorded from the rat's motor cortex [69]. As it contains two differential equations to describe the biological membrane potential, it is suitable for large-scale simulations.

In particular, the model is described by the following system of equations:

$$\begin{cases} \dfrac{d\mathbf{V}}{dt} = 0.04\mathbf{V}^2 + 5\mathbf{V} + 140 - \mathbf{U} + \mathbf{I} \\ \dfrac{d\mathbf{U}}{dt} = a\left(b\mathbf{V} - \mathbf{U}\right) \end{cases} \tag{2.4}$$

where $\mathbf{V}$ is the membrane potential of the neuron and $\mathbf{U}$ is the membrane recovery variable that describes the activation of potassium and inactivation of sodium ionic currents: $\mathbf{U}$ acts as a negative feedback to $\mathbf{V}$, stabilizing its behavior.

When the membrane potential $\mathbf{V}$ exceeds the threshold value of 30 mV, the neuron fires and then resets at resting potential. In general, the resting potential is approximately 60-70 mV and depends only on the b parameter.

After the firing event, the state variables have a transition as ruled by:

$$\begin{cases} \mathbf{V} = c \\ \mathbf{U} = \mathbf{U} + d \end{cases} \tag{2.5}$$

By acting on the parameters a, b, c and d we can change the dynamic behavior of the model. And in particular, as reported in [69]:

- a describes the time scale of the recovery variable $\mathbf{U}$;
- b describes the sensitivity of $\mathbf{U}$ to the subthreshold fluctuations of $\mathbf{V}$;
- c describes the after-spike reset value of $\mathbf{V}$ caused by the fast high-threshold potassium conductances;
- d describes the after-spike reset of $\mathbf{U}$ caused by slow high threshold sodium and potassium conductances.

By properly choosing a, b, c and d, we achieve different behaviors when a continuous current is injected into the neuron, such as:

- Regular Spiking (RS) - It is the most common neural behavior in the cortical tissue. The neurons fire a few spikes with short interspike period and then the period increases;

- Intrinsically bursting (IB) The neuron fire a stereotypical burst of spikes followed by a repetitive single spike;
- Chattering (CH) - The neuron can fire stereotypical bursts of closely spaced spikes;
- Resonator (RZ) - The neuronal membrane potential resonates at a specific frequency defined by the parameters model before firing.

The four behaviors described above are depicted in Fig. 2.3. Many other different waveforms of the membrane potential can be achieved by adapting the aforementioned parameters.

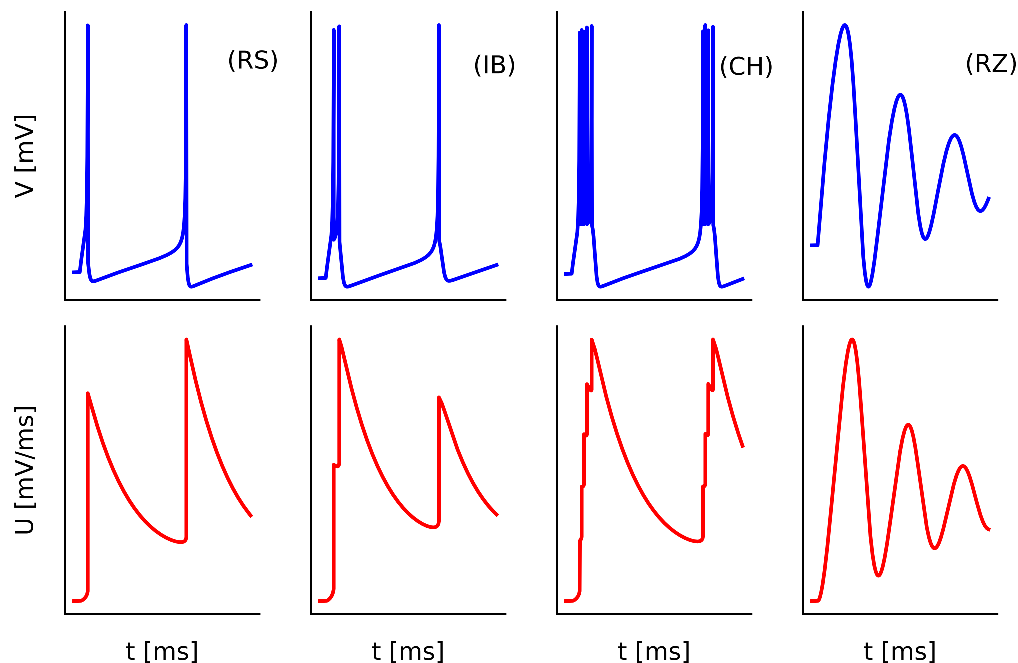

Figure 2.3: Example of membrane potential (V) and membrane recovery variable (U) of Izhikevic model for a different set of parameters. (RS) Regular spiking with a=0.02, b=0.2, c=-65 and d=8; (IB) Intrinsically bursting with a=0.02, b=0.2, c=-55 and d=4; (CH) chattering with a=0.02, b=0.2, c=-50 and d=2; (RZ) Resonator with a=0.1, b=0.26, c=-65 and 2. In these examples, the input synaptic current I is always constant.

2.1.2 Learning rules

Differently from the conventional *ANNs* approach, where the backpropagation algorithm plays a central role, in neuromorphic computing several learning methodologies have been studied in order to overcome the gap between synthetic and biologic neural networks. In fact, backpropagation and stochastic gradient descent showed astonishing results in combination with *ANNs*. However, it cannot be directly applied to *SNNs* due to the non-derivability issue of spikes, which are frequently modeled as Dirac delta distributions. Due to this, initially, SNN-based applications were performed by manually setting network weights [70, 71]. Although this approach is able to solve simple behavioral tasks, it is only feasible for lightweight networks with few connections or simple network architectures without hidden layers [72].

This lack of a standardized learning methodology pushed researchers to pursue a feasible alternative. In literature we can find offline and online learning algorithms. Offline learning methods calculate the adjustment of synaptic weights according to the entire output spike trains after the elaboration of the network, while online learning methods update the weights during the neural firing events. Offline learning algorithms are suitable for static data processing, while online learning algorithms are very convenient in real-time applications [73]. Moreover, the locality of the learning method should be considered: locality means that the learning rules are determined only by the pre-synaptic and post-synaptic activities of neurons, and the synaptic weight itself. Hence, to be defined as *local*, the quantity that modifies the weight of a synapse must be available locally [74]. Moreover, the applicability of the methods to different spiking neuron models should be taken into account. Some learning algorithms rely explicitly only on the spike times and do not refer to the particular properties of the spiking neuron models, hence they are expected to work correctly independently of the spiking neuron model used.

For unsupervised tasks, the literature mainly focuses on the exploitation of the synaptic plasticity property of biological neural networks combined with an adaptation of Hebbian learning called *Spike Time Dependent Plasticity* (*STDP*) [75]. *STDP* is one of the most used unsupervised, local, and biologically plausible, learning algorithms in neuromorphic computing and computational neuroscience [76, 77]. According to *STDP*, a synaptic weight is potentiated if a pre-synaptic spike comes before a post-synaptic spike, whereas it is conversely depressed each time the pre-synaptic neuron fires after the postsynaptic one. In order to apply the synaptic plasticity to different use cases, multiple versions of *STDP* have been proposed in literature. For example, a supervised version of *STDP* called, rewarded *STDP*, has been proposed to solve classification problems [72]. Moreover, *STDP* has been used as a learning algorithm in collaboration with Homeostatic plasticity to solve classification tasks [78]. However, biologically plausible learning rules for unsupervised learning are still to be fully explored and nowadays there is no ample consensus on a sort of standard learning rule for *SNNs*, as it is instead the case for backpropagation in *ANNs* .

For supervised learning, instead, the first proposed *SNNs* compatible learning algorithm is SpikeProp [79]. SpikeProp estimates the gradient similarly as backpropagation does for *ANNs*. However, it presents some limitations: this algorithm is limited to a network of neurons emitting just a single spike and, in addition, non-firing neurons prevent the calculation of the gradient. About this last point, Sec. 2.2.5 considers a similar issue: non-firing neurons produce a flat region of the loss which precludes the variation of the afferent parameters. In our work, this problem has been tackled by pursuing an approximated minimum of the cross-entropy loss reducing the spiking time of the output neurons with an augmented loss function. Moreover, in [80] Spikeprop has been extended to a loss dependent on an arbitrary number of output spikes, whereas [81] addressed the case of recurrent networks.

An example of interoperability between *SNNs* and *ANNs* is the *ANNs* -to-*SNNs* weights conversion: in this method, an *ANN* is trained with classical deep learning methodologies, then the weights of the network are converted thanks to a non-linear transformation [82].

Another algorithm, called *random synaptic feedback*, has been proposed in [83]. This is a backpropagation algorithm that sends the error calculated with the loss function through a feedback network with random fixed weights.

Different approaches have been developed by computing exact gradients using methods from the

optimal control theory. In particular, the sensitivity analysis has been applied to a recurrent network of *LIF* neurons that aims to exhibit a given oscillatory behavior [84]. Algorithms for recurrent SNNs have been derived by leveraging adjoint equations and accounting for the hybrid dynamic of neurons [85], as well as in [86], where the threshold for neuron firing has been replaced by a gate function that smoothens the transitions and facilitates the implementation of the adjoint methods. Another important step ahead in the biological learning understanding is E-prop [87], a biologically inspired algorithm to train recurrent spiking neural networks. It exploits eligibility traces of neural activity to adapt the weights of the network. Moreover, [88] proposes to exploit surrogate approximated functions to calculate surrogate gradients in a backpropagated *SNN* environment to overcome the non-derivability of the spiking instant.

Finally, Event-Prop is instead an adaptation of the backpropagation algorithm for *SNNs*, that calculates the precise gradient of the loss function by solving the lack of derivability of the spike events by leveraging on adjoint equations [89]. In Event-Prop, thanks to the relatively simple formulation of the *LIF* model, exact gradient computation has been derived for these kinds of neurons. On top of this, we developed in Sec. 2.2 a two-phase method to enhance accuracy of deep *SNNs* and also keep the spiking rate of the network under control. Moreover, since it has been theoretically proven that spiking networks have a Turing-equivalent computing power even for very basic mechanisms of spike generation [90], the use of *LIF* neurons should not be perceived as an actual limitation, even if more biophysically accurate neuron models have been proposed, such as the Hodgkin and Huxley or the Izhikevich models. In particular, we mathematically demonstrated, in Appendix B, that the Izhikevich model is suitable for the application with Event-Prop.

2.2 Reducing the Spike Rate in Deep Spiking Neural Networks

For a 2-layers feedforward network consisting of leaky integrate-and-fire neurons and having a loss function that depends on the times of the spikes, exact gradients have been computed in [89]. This was accomplished by using adjoint functions, which are evaluated and recorded only at the time instants corresponding to spikes. In particular, the discontinuities on the adjoint equations enforced by spikes are computed going backward in time in topological order. This approach leverages the sparseness of spikes to obtain a computation of the gradient that is effective in terms of both computation energy and memory requirements.

None of the aforementioned approaches in Sec. 2.1, however, implies any target for the possible spikes generated after the classification has been completed. Such spikes can occur when neurons are over-excited from a large number of afferents, which is statistically more probable when the network becomes deeper. While an excess of spikes is innocuous for the network operation, it is clearly detrimental for the energy consumption and, moreover, it is also biologically implausible because the refractory mechanism can effectively limit the spike rate in biological neurons. In this latter respect, we here argue that a refractory period appears indispensable in recurrent *SNNs* to prevent an unlimited spike rate [91], but this also implies an energy cost, because the refractoriness is obtained with a dissipative configuration in most circuital implementations [13, 92]. In a feed-forward network, instead, the excess of spikes can also be avoided by a judicious choice of the synaptic weights, which makes the implementation of the refractory period not strictly necessary and thus saves the corresponding energy cost.

In deep *SNNs*, however, a mere gradient descend approach has limitations even when the exact gradient can be calculated. In fact, the average spike rate tends to accelerate for an increasing number of layers, which deteriorates the energy efficiency of both the learning and the inference phase. In this respect, if measures are introduced in order to control the spike rate, one should carefully avoid to induce a large number of silent neurons, namely neurons that do not emit spikes. Such neurons, in fact, do not not practically contribute to the loss function, so that a gradient descent strategy is no longer effective to optimize their input weights. This hazard is similar to the "vanishing" gradient issue that in *ANNs* precludes the convergence toward the actual minimum of the loss function.

The targets of this section are in fact deep *LIF* feed-forward networks, and we will present a training strategy that can optimize the network performance. Moreover, we propose a modification of the loss function presented in [89], in order to reduce the spike rate and, at the same time, overcome the silent neuron issue. The last paragraph of this section raises the problem of device-to-device parameters mismatch and quantifies the drop in accuracy related to circuital non-idealities in neuromorphic implementations.

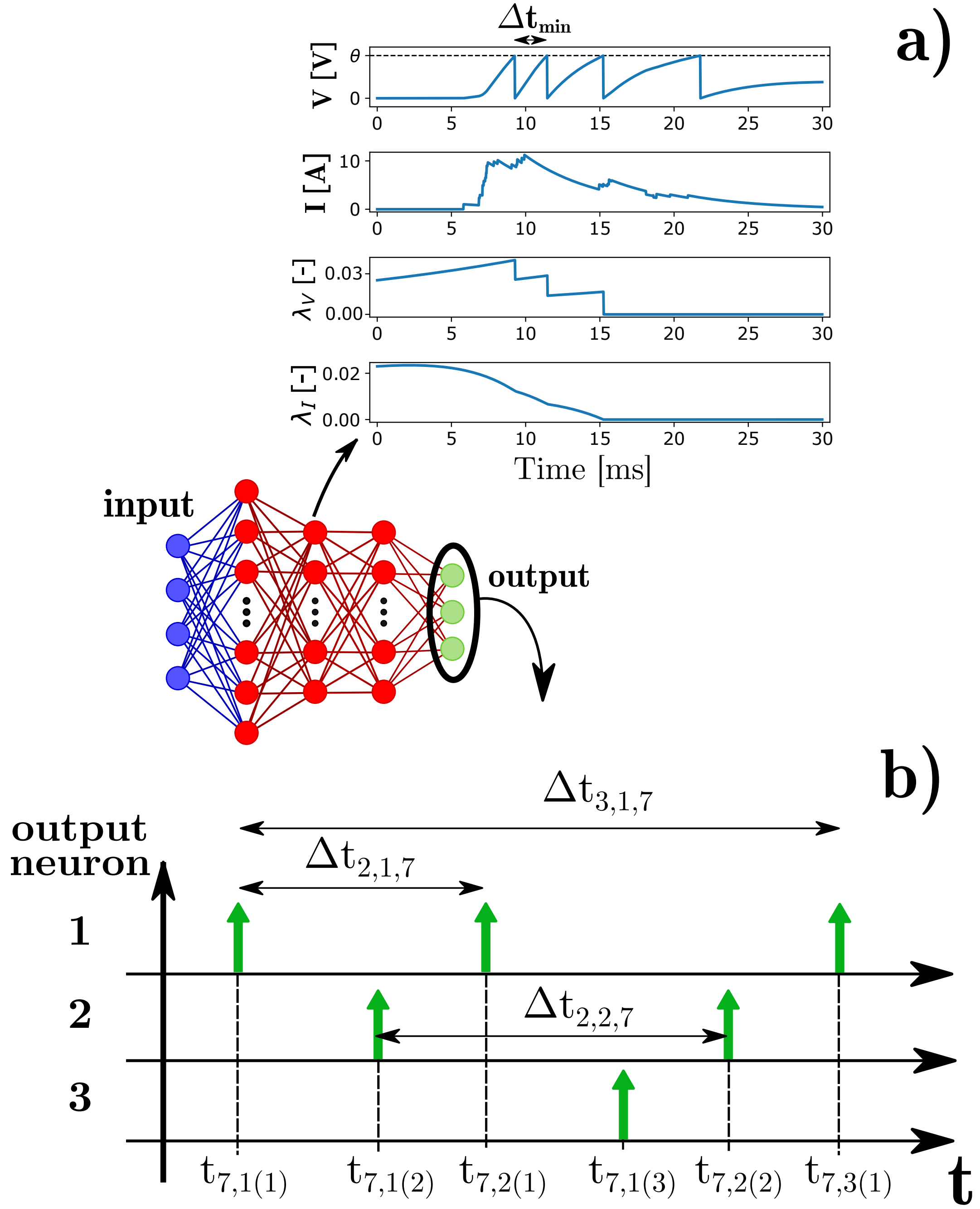

Figure 2.4: Example of general topology of the networks exploited in this work and typical neural waveforms. (a) Example of neural internal variables: V and I denote the membrane potential and the post-synaptic current, while λ_V and λ_I are the adjoint variables associated to V and I, respectively. (b) Example of output spiking activity and specific notation: every spike instant is denoted with $t_{b,p(a)}$, while the time gap between each spike and the first emitted is $\Delta t_{p,a,b}$ where b is the sample of the minibatch, a is the neural index and p is the spike index emitted by the neuron a.

2.2.1 Models

Spiking Neurons and Gradients

The constitutive equations of the *LIF* neurons are:

$$\begin{cases} \tau_m \dfrac{\partial V}{\partial t} = -V + R_m I \\ \tau_s \dfrac{\partial I}{\partial t} = -I \end{cases} \tag{2.6}$$

where V and I are the neural membrane potential and the input synaptic current respectively. τ_m and τ_s are the time constants governing the dynamics of the membrane potential and the synaptic current respectively, while R_m is the membrane resistance. The state variables V and I of each neuron are initialized to zero:

$$\begin{cases} V\left(t=0\right) = 0 \\ I\left(t=0\right) = 0 \end{cases} \tag{2.7}$$

The times of spikes are implicitly defined by:

$$\left(\mathbf{V}^{-}\right)_{n(k)} = \Theta \tag{2.8}$$

where the vector $\mathbf{V}$ and $\mathbf{I}$ gather all the membrane potentials and currents of the network, and Θ is the constant threshold potential. The notation $(\mathbf{V}^{\pm})_{n(k)}$ denotes, throughout this chapter, the n_{th} component of the membrane potential vector $\mathbf{V}$ an instant before $(-)$, or after $(+)$, the emission of the k_{th} spike. When the membrane potential of a neuron reaches the threshold, such a neuron emits a spike and resets to the rest potential ($(\mathbf{V}^{+})_{n(k)} = 0$ V), though the current does not change ($(\mathbf{I}^{-})_{n(k)} = (\mathbf{I}^{+})_{n(k)}$) since a node does not act on itself. However, the k_{th} spike produces a jump on the current of the receiving neuron given by the synaptic weight $(\mathbf{I}^{+})_m = (\mathbf{I}^{-})_{n(k)} + W_{m,n}$, where $W_{m,n}$ links the n-th spiking neuron to the m-th neuron.

To train the network we compute the exact gradient, leveraging adjoint variables and back-propagation [89], to perform a mini-batch stochastic gradient-descent. The equations of adjoint variables can be derived from Eq. 2.6 and are:

$$\begin{cases} \tau_m \dfrac{\partial \lambda_V}{\partial t} = \lambda_V \\ \tau_s \dfrac{\partial \lambda_I}{\partial t} = \lambda_I - R_m\, \lambda_V \end{cases} \tag{2.9}$$

$$\begin{cases} \left(\lambda_{\mathbf{V}}{}^{-}\right)_{n(k)} = \dfrac{1}{R_m\left(\mathbf{I}^{-}\right)_{n(k)} - \theta} \left[\left(\lambda_{\mathbf{V}}{}^{+}\right)_{n(k)} R_m \left(\mathbf{I}^{-}\right)_{n(k)} + \dfrac{\partial \mathcal{L}(t_k)}{\partial t_k} + \sum_{m \neq n}^{N_n} \left(R_m \left(\lambda_{\mathbf{V}}{}^{+}\right)_m - \left(\lambda_{\mathbf{I}}{}^{+}\right)_m \right) W_{m,n} \right] \\ \left(\lambda_{\mathbf{I}}{}^{-}\right)_{n(k)} = \left(\lambda_{\mathbf{I}}{}^{+}\right)_{n(k)} \end{cases} \tag{2.10}$$

The adjoint variables have discontinuities (ruled by Eq. 2.10) in correspondence of spikes and

are evaluated backward in time, starting from the initial condition:

$$\begin{cases} \lambda_V (t = T) = 0 \\ \lambda_I (t = T) = 0 \end{cases} \tag{2.11}$$

An example of the neural state functions and the adjoint functions is represented in Fig 2.4(a).

All the loss functions considered in this work only depend on the spike times; we do not take into account losses that depend directly on the membrane potential, since they are not consistent with an event-driven computation. Hence, the gradient of the loss function can be written as:

$$\frac{\partial \mathcal{L}}{\partial W_{i,j}} = \sum_{k=1}^{N_{spk}} \frac{d\ \mathcal{L}(t_k)}{dt_k} \frac{dt_k}{dW_{i,j}} \tag{2.12}$$

where N_{spk} is the number of spikes generated by the network.

By applying the implicit function theorem to Eq. 2.8 and exploiting the adjoint variables, the gradient of the loss becomes:

$$\frac{\partial \mathcal{L}}{\partial W_{i,j}} = -\tau_s \sum_{k \in \{spikes\, from\, j\}} (\lambda_{\mathbf{I}})_i \, (t_k) \tag{2.13}$$

Hence, to calculate the gradient of $\mathcal{L}$, we only need to sample λ_I when a spike happens. Moreover, since the state functions and their adjoints can be expressed in closed form between two discontinuities, all the evaluations can be performed with an event-driven computation.

Loss Functions

We consider deep feed-forward networks for classification tasks. The network receives input spikes, related to input data through an application-specific encoding, and propagates spikes up to the output neurons in the last layer. The output node that emits the first spike indicates the predicted class.

The first loss function taken into account is:

$$L_W = CE + \alpha CS \tag{2.14}$$

where CE (cross-entropy) and CS (classification spike) are defined as:

$$\begin{cases} CE = -\frac{1}{N_b} \sum_{b=1}^{N_b} log \left(\frac{exp\left(-t_{b,1(l)}/\tau_0\right)}{\sum_{a=1}^{N_O} exp\left(-t_{b,1(a)}/\tau_0\right)} \right) \\ \\ CS = \frac{1}{N_b} \sum_{b=1}^{N_b} \left[exp\left(\frac{-t_{b,1(l)}}{\tau_1}\right) - 1 \right] \end{cases} \tag{2.15}$$

with N_b and N_O being the number of samples of the batch and output neurons, respectively. τ_0, τ_1 are normalization constants, and α is a hyper-parameter. $t_{b,1(l)}$ is the time of the first spike emitted by the output neuron l corresponding to the correct classification, and $t_{b,1(a)}$ is, instead, the time of the first spike of the generic output neuron a.

CE is the actual target of the minimization, since it is directly related to the average classification

error because it is a measure of how much the real distribution of the dataset differs from the distribution estimated by the network [93]. Since CE depends only on the *difference among times* of the first output spikes, we can add terms to the loss function to drive the minimization towards points with convenient characteristics. Here CS aims to minimize the delay between the input and the output spikes and, as a by-product, to reduce the probability that neurons remain silent.

For shallow networks L_W provides good results [89], albeit for deep networks a good level of accuracy is obtained at the cost of a large spiking rate, as discussed in Section 2.2.1. Trying to reduce the spiking rate by decreasing the hyper-parameter α leads to a relevant reduction of the accuracy, since many neurons become silent.

Therefore, to improve the control on the spike rate, we defined a second loss function by adding a term that, for each neuron, reduces the excess of spikes without penalizing the first one:

$$L = CE + \alpha CS + \eta SP \tag{2.16}$$

SP (spike penalty) is defined as:

$$SP = \frac{1}{N_b} \sum_{b=1}^{N_b} \sum_{a=1}^{N_n} \sum_{p=2}^{P(a,b)} \frac{1}{\Delta t_{p,a,b}} \tag{2.17}$$

N_n is the number of neurons of the network, $P(a, b)$ is the total number of spikes emitted by the neuron a during the inference of sample b, and η is another hyper-parameter. $\Delta t_{p,a,b} = t_{b,p(a)} - t_{b,1(a)}$ denotes the time difference between each spike produced by the neuron a during the inference of the sample b ($t_{b,p(a)}$) and the first one ($t_{b,1(a)}$) (for the sake of clarity, Fig. 2.4(b) provides a representation of the spiking notation). If a given neuron emits only a single spike, its contribution to SP is null, thus the spike penalty tends to enlarge the delay between the first emitted spike and the following ones for each neuron of the network.

With the loss function L, the network can reach the same CE performance obtained with L_W also reducing the average spiking rate. As an example, Fig. 2.5(a) shows two different trajectories in the $\langle t_T \rangle$-$\langle t_{NT} \rangle$ plane followed by L_W and L during a training for the benchmark Yin-Yang (discussed in section 2.2.4). $\langle t_T \rangle$ is the average time of the first spike generated by the neuron corresponding to the correct class (the *target* neuron), $\langle t_{NT} \rangle$ is the average time of the first spike of the other output neurons (the *not target* nodes), and the oblique lines are the isolines of CE. This preliminary analysis shows that, using L_W, large values of α are required to reach a good accuracy (i.e., small values of CE). Such a choice, however, leads to an excess of spikes at the end of training, as shown in Fig. 2.5(b). For L, on the contrary, a large α can provide good results, in terms of CE, and keep the spike rate under control. However, the choice of such a hyper-parameter is critical: a too small α can reduce the average spike rate below 1, hence switching off some neuron and decreasing the final accuracy. Silent neurons, in fact, impact the minimization process because they do not contribute to the gradient of the loss (Eq. 2.13).

Finding suitable values for α and η is not trivial, because they strongly depend on the starting point of the minimization and on the network topology. Therefore, a strategy to reach a convenient starting point is required. For such a purpose, we introduce a third loss function, L_A:

$$L_A = CE + \alpha AS + \eta SP \tag{2.18}$$

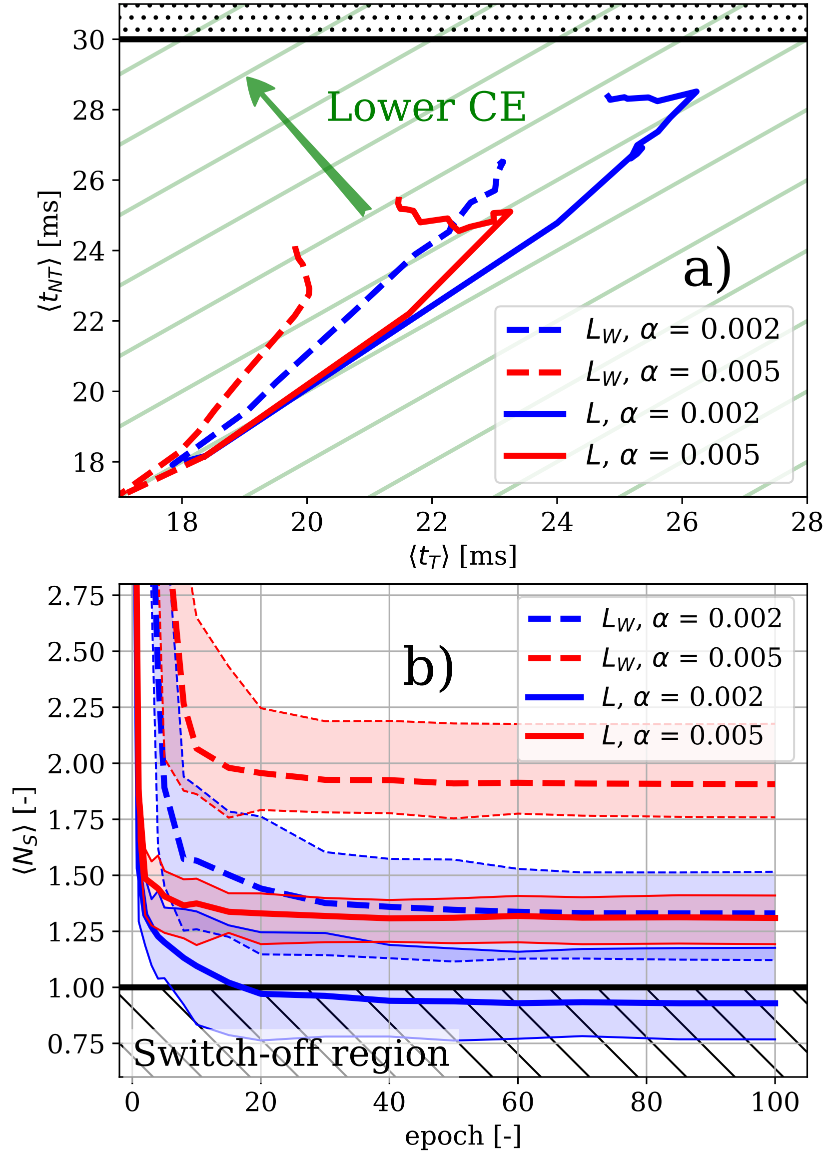

Figure 2.5: Yin-Yang - Comparison between L_W and L for two different values of α. (a) compares the average first spike arrival time for the target neuron $\langle t_T \rangle$ with the first spike time arrival of the others $\langle t_{NT} \rangle$ during training, while the solid oblique lines depict different cross-entropy levels. (b) shows the average number of spikes per neuron of the whole network, $\langle N_S \rangle$, during training. The shaded area shows the variability of $\langle N_S \rangle$ for multiple initializations. In these simulations η is set to 0.35 ms.

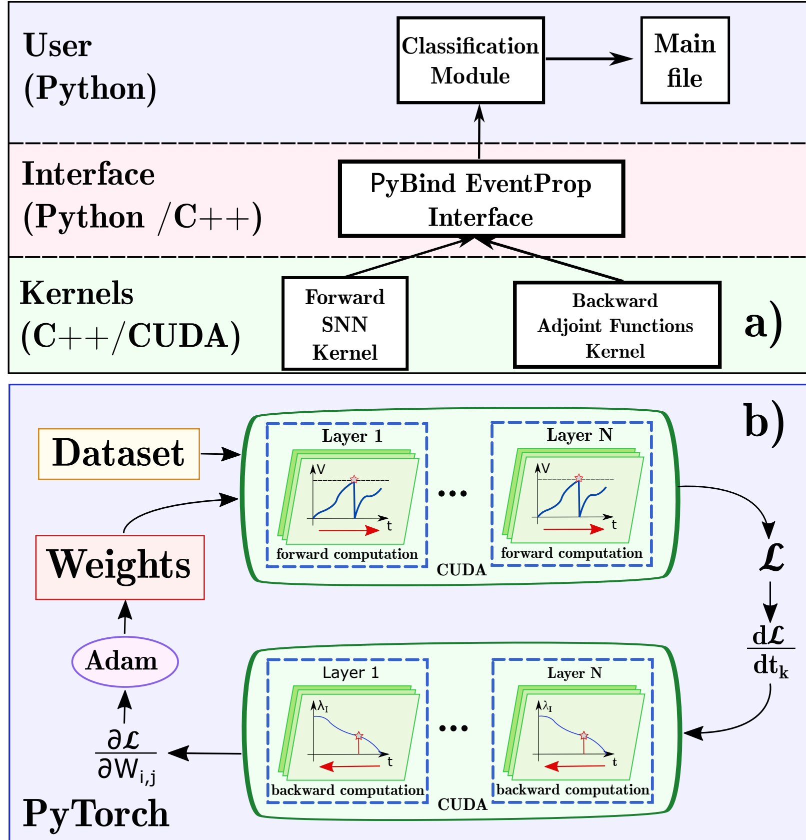

Figure 2.6: Sketch of the implemented algorithm. (a) Logical description of the simulator. (b) Behavioral representation of the algorithm. The simulator is based on Pytorch with a CUDA-implemented extension for the calculation of the neural signals and the adjoint functions.

where AS (additional spikes) is an augmentation term, defined as:

$$AS = \frac{1}{N_O N_b} \sum_{b=1}^{N_b} \sum_{a=1}^{N_O} \left[exp\left(\frac{-t_{b,1(a)}}{\tau_1} \right) - 1 \right] \tag{2.19}$$

that focuses on the first spike event of *all* output neurons. Notice that this term encloses CS (except for a constant factor), thus L_A contains the same terms as L plus the extra addends related to other spikes. An important consideration, however, is that every minimum of L is also a minimum of CE (and the same is true for L_W), but this does not hold for L_A because of the augmentation term. Nevertheless, a minimum of L_A is close enough to a minimum of CE to be considered a valid starting point for a successive minimization and, moreover, it can be reached without incurring in the switch-off issue, as empirically assessed.

Therefore, we adopt a two-phase strategy for learning: in the first phase the training aims at the minimization of L_A, in order to reach a good point to start the second phase that, minimizing L, targets a minimum of the cross-entropy.

2.2.2 SNN simulator

We simulated the networks using the PyTorch [94] framework. The simulator is implemented in Python, to interact with PyTorch, and in CUDA/C++, to accelerate the computation by exploiting parallel and fully even-driven CUDA Kernels that run on Nvidia GPU. To combine the computa-

Table 2.1: Model's parameters: the values listed in this table correspond to a purely mathematical model and they can be scaled in order to fit a physical system.

Description	Symbol	Value
Membrane time constant	τ_m	20 ms
Synaptic time constant	τ_s	5 ms
Membrane resistance	R_m	1 Ω
Threshold	Θ	1 V
Min input time spike	t_{min}	0 ms
Max input time spike	t_{max}	20 ms
Simulation time	T	30 ms
Adam parameter	β_1	0.9
Adam parameter	β_2	0.999
Adam parameter	ϵ	1×10^{-8}
Learning rate decay factor	-	0.95
Learning rate decay step	-	1 epoch
CE normalization constant	τ_0	0.5 ms
CS, AS normalization constant	τ_1	6.4 ms
Yin-Yang 1st phase Learning rate	-	5×10^{-3}
Yin-Yang 2nd phase Learning rate	-	2×10^{-4}
Yin-Yang batch size	N_b	32
MNIST Learning rate (*both phases*)	-	2×10^{-3}
MNIST batch size	N_b	20

tional power of C++/CUDA and the versatility of Python we exploited PyBind [1]. PyBind is a library that exposes C++ types in Python and vice versa, mainly to create Python bindings of existing C++ or CUDA code. Fig. 2.6(a) shows the architecture of the simulator, with the Python modules in charge of loading the dataset and orchestrating the simulation. The CUDA code computes the spike times (finding the roots of Eq. 2.8) and updates the adjoint variables for each discontinuity.

As schematically described in Fig. 2.6(a), the simulator is based on a Python front-end interface linked with the CUDA kernels. The Python code manages the interactions with the PyTorch framework that is in charge of interacting with datasets, performing data conversion and tensor manipulation. An SNN simulation consists of two phases: the forward pass and backward pass. The first one calculates the membrane potential (V) and the post-synaptic currents (I) for each neuron of the network to determine each spiking time. After that, the backward pass evaluates the adjoint variables at each spike event, calculates the loss gradient and updates the weights by exploiting Adam [95] as optimizer with a constant learning rate decay. The neural spiking activity is elaborated layer by layer, sequentially, from the first to the last layer for the forward propagation, and, in a reverse way for the backward computation. The CUDA platform exploits an Nvidia GPU environment to speed up the calculations. The simulator is based on two main kernels (function executed directly on GPU): the forward CUDA kernel is characterized by a Newton-Raphson cycle executed for each neuron of any layer of the network (see Alg. 1). The goal of this method is to calculate, with an arbitrary precision δ, the spike instants of each neuron of the network by exploiting the analytical solution of Eq. 2.6. As described by Alg. 1, by finding the roots of Eq.

[1] https://github.com/pybind/pybind11

2.8, it is possible to jump between each pre-synaptic spiking event without exploiting continuous step-driven integration of differential equations. This improves the quality of the numerical results in terms of numerical precision, memory consumption, and computing time. The backward kernel is responsible for the adjoint functions calculation: starting from the previously saved neural input synaptic current and spike time, the backward algorithm computes, backward in time, the value of the adjoint functions at each spike event.

Algorithm 1 Forward CUDA Kernel

```
I_0^- ← 0
V_0^- ← 0
for each input spike do
    t_0 ← input spiking time
    t_next ← next input spiking time
    I_0^+ ← I_0^- + βW_{m,n}                                   ▷ Current discontinuity
    t ← t_0
    Δ ← 1.0
    while True do
        while Δ ≥ δ do                                         ▷ Newton-Raphson loop
            V ← e^{-(t-t_0)/τ_m} (V_0^+ + I_0^+ τ_s/(τ_s-τ_m) (e^{(t-t_0)(τ_s-τ_m)/(τ_s τ_m)} - 1.0))
            f ← V - θ
            fp ← 1/τ_m (-V + I_0^+ exp(-(t-t_0)/τ_s))
            if fp ≤ 0 then
                break
            end if
            Δ ← -f/fp
            t ← t + Δ
        end while
        I_0^- ← I_0^+ exp(-(t-t_0)/τ_s)
        if fp > 0 && t_next > t then                           ▷ Found a spike
            I_0^+ ← I_0^-
            V_0^+ ← 0
            t_0 ← t
            Save (I_0^+, t)
        else                                                   ▷ Spike not found
            V_0^+ ← f_V (V_0^+, I_0^+, t_next)
            break
        end if
    end while
end for
```

The final computation of the gradient and the update of synaptic weights is performed by the PyTorch framework through its implementation of the Adam optimizer [95]. The whole flow of the computation is represented in Fig. 2.6(b).

The networks hyper-parameters are listed in Table 2.1, with values in the upper portion common to all the simulated networks.

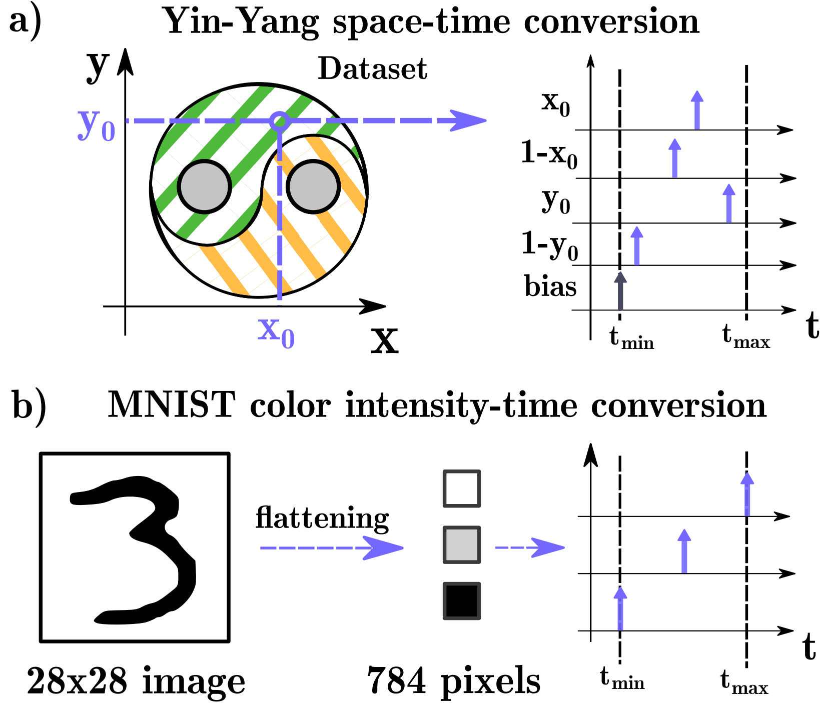

Figure 2.7: Sketch of the datasets conversion and example of neural signals. (a) Representation of the Yin-Yang Dataset and the linear transformation of spatial coordinates into spiking time events. (b) Example of conversion of a MNIST image: the pixel color intensity is translated into a spike timing with a linear transformation.

2.2.3 Benchmarks

Yin-Yang

The first benchmark taken into account is the Yin-Yang dataset [96, 97] composed of points in a 2D space. Each point lies in a region, defined by non-linear boundaries, which is the target of the classification (there are three regions). A point is translated to spikes by means of a linear space-time conversion within a fixed range $[t_{min}, t_{max}]$ of its coordinates (x, y) and their duals $(1-x, 1-y)$. The dual spikes are added to keep the average spike times of the inputs constant for the whole dataset. Moreover, an input spike at fixed time $t = t_{min}$ is added as a bias for a total of 5 inputs for the classifying *SNN* . Fig. 2.7(a) shows the structure of the dataset and an example of the space-time conversion.

This benchmark is small enough, still not trivial, to allow the fast preliminary evaluations of the learning strategies that we discussed above. To achieve fair and meaningful comparisons, we use a network with almost the same number of synapses as the network used in [89], but deeper and, thus, with fewer neurons: for this benchmark we employ a 5-40-25-13-3 feed-forward network with synaptic weights initialized with a uniform distribution in the range reported in Table 2.2.

MNIST

The second benchmark considered is the MNIST dataset [98]. MNIST is composed of images (28x28 pixel large, gray-scale) that depict the hand-written digits to be recognized by the network. The training set contains 60000 images, while the images in the test set are 10000. Each pixel of the input image is translated to an input spike: the spiking time is a linear mapping of the pixel intensity

Table 2.2: Ranges of weights initialization

Layer	# Neurons	Minimum	Maximum
	Yin-Yang		
1	40	1.0	3.0
2	25	0.2	1.0
3	13	0.0	1.0
4	3	0.0	1.0
	MNIST		
1	280	-0.1	0.3
2	160	-0.5	0.15
3	70	-0.2	0.6
4	10	-0.4	0.8

to the range $[t_{min}, t_{max}]$ as show in Fig. 2.7(b). Since the majority of pixel has the background color and the input data are large enough (784 pixels), there is no need to add dual inputs or a bias.

For this benchmark we use a 784-280-160-70-10 network, that requires more than $2 \cdot 10^5$ synapses. The number of synapses is large enough, compared to the number of images in the training set, to avoid overfitting. Therefore, we augmented the training set by introducing random shifts (horizontal and vertical, up to 10%) and random rotations (up to 20°) of the images. Even for this benchmark, the synaptic weights are initialized with a uniform distribution: the ranges are reported in Table 2.2.

The starting weight ranges of both benchmarks are tuned to initially ignite all neurons of the network and produce a sparse enough spiking activity to prevent an over-excitation of the output layer's neurons. The learning rate for the Yin-Yang analysis has been changed between the first and the second phase of training to better follow the distribution of the average weights of the network. Conversely, the MNIST preliminary exploration reported in the next paragraph reveals a good behavior of the training process even by keeping the same learning rate for both phases.

2.2.4 Experimental Results

Yin-Yang Simulation Results

The first evaluation is the behavior of the 4-layer network, applied to the Yin-Yang benchmark, with the reference loss L_W and by varying the hyper-parameter α. Fig. 2.8 reports (a) the accuracy and the mean arrival time of the first classification spike, and (b) the average number of spikes per neuron ($\langle N_S \rangle$), reported separately for each layer. Data are related to one inference performed by a trained network. For low values of α (the dashed area) the accuracy is quite poor, since at least one layer (in particular the output layer 4) has $\langle N_S \rangle < 1$. This implies that some neurons are silent hence, at some point, the gradient descent lost the control to change the weights of synapses affering to such neurons. Larger values of α provide better accuracy and, as a by-product, a faster response, but with the penalty of an excess of spikes. As stated above, too many spikes are harmful in terms of energy efficiency and should be thus avoided. The ideal condition is no more than a single spike per neuron per inference.

As an example of the behavior of the network *during* the training, we report two cases in Fig. 2.9 for 100 epochs. The first case (dashed lines: $\alpha = 5 \cdot 10^{-3}$, large enough to provide a good accuracy)

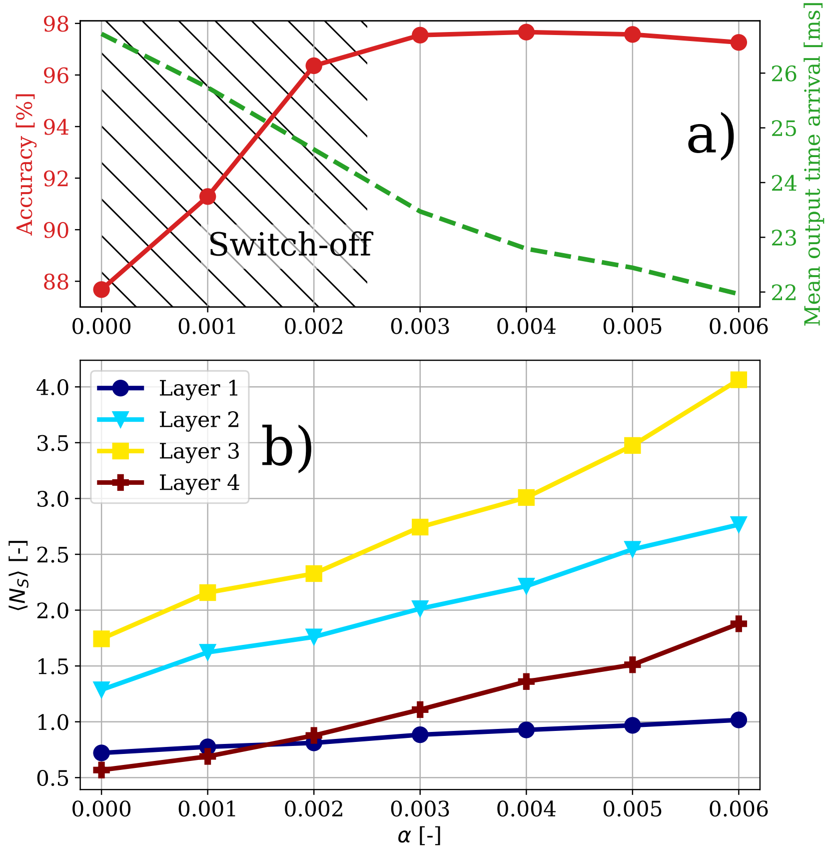

Figure 2.8: Yin-Yang - Exploration of the parameter α for loss L_W. (a) Comparison between the accuracy of the network (solid line) and the mean first spike arrival at the output layer (dashed line). (b) Dependency of the mean number of spikes emitted by each layer of the network.

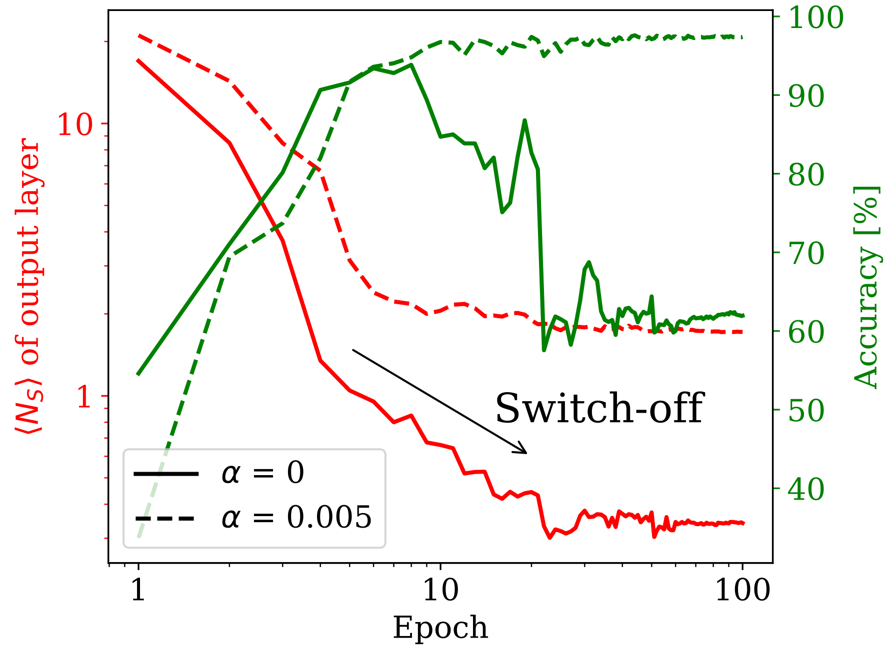

Figure 2.9: Yin-Yang - Example of switch-off. Comparison between the average number of spikes per neuron $\langle N_S \rangle$ of the output layer and the accuracy of the network during two 100-epoch training, starting from the same realization of weights and exploiting the L_W loss function.

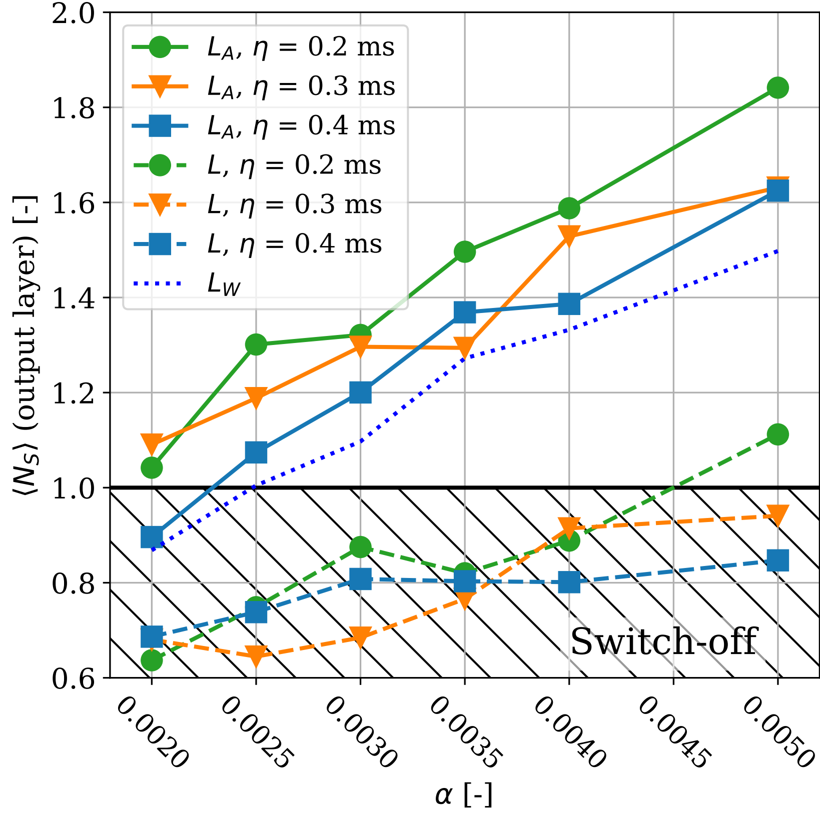

Figure 2.10: Yin-Yang - Average number of output spikes per neuron for different α, η in the L and L_A cases. As a comparison, it is also shown the output spiking rate of Fig. 2.8(b) (dotted line).

shows how the spiking rate, albeit reduced while the training is in progress, saturates at a value that is near to 2. The second case (solid lines: $\alpha = 0$) shows that the accuracy improves only for the first few epochs; when the spike rate becomes too small, too many nodes become quiescent, and the network accuracy drops to uselessness.

This first exploration shows that, even for a simple benchmark, keeping the spike rate under control while training a deep network is challenging, if not impossible, and it is the driving motivation for the changes in the loss function proposed in this chapter.

As mentioned above, the loss functions L and L_A introduce a new hyper-parameter (η). Fig. 2.10 shows the impact of α on the average number of spikes of the output layer, for some values of η and at the end of a 100 epochs training, while Fig. 2.11 shows the accuracy for the same configurations. In both plots, dashed lines are for L and continuous lines are for L_A. Fig. 2.10 points out how the excess of spikes can be totally eliminated by using L as loss function, hence remarking the effectiveness of the added penalty term. The accuracy, however, is negatively impacted (Fig. 2.11) and, to reach acceptable values, large values of the hyper-parameter α are required, with the resulting increase of the number of spikes. Furthermore, the minimization of L is very dependent on the starting point and in general it is not very reliable because, in many cases, some neurons become quiescent, hence adversely affecting the optimization. This is the reason for the non-monotonic behavior of the accuracy for L shown in Fig. 2.11.

Conversely, minimization with L_A, keeps all neurons active, because of the AS term, and provides an appreciable accuracy (but for the smallest values of α). This indicates that the minimization

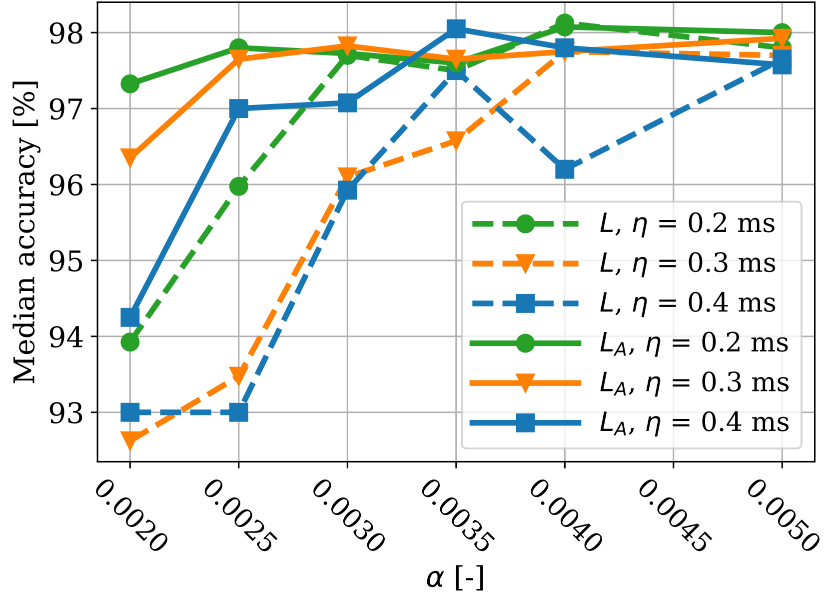

Figure 2.11: Yin-Yang - Median accuracy after 100 epochs of training. This plot shows the impact of α and η on the network's training by exploiting two different loss functions: L and L_A.

does not suffer from the switch-off condition, as confirmed by the number of spikes of the output layer, hence the minimum of L_A is approached without halting in some plateaux. However, as discussed in Section 2.2.1, L_A cannot provide the actual minimum of the cross-entropy for the very nature of AS, hence a better accuracy may be achievable. The purpose of L_A, in fact, is allowing a *surrogate* minimization of L while forcing the nodes to stay active. Then, using the result of the minimization through L_A as the starting point of a second minimization, that uses L, we can avoid the aforementioned instability (because the initial point is quite close to the final minimum) and reach the real minimum of CE.

Fig. 2.12 shows an example of 2-phase training, where the first phase (100 epochs, plot on the left) is performed minimizing L_A, while in the second phase (60 epochs, plot on the right) the target of the minimization is L. In the left plot, we also show the behavior of a training that only uses L and that successfully achieves a valid minimum; here it should be noticed that this is a selected lucky case, chosen among many unsuccessful tests, while all the training minimizations performed with L_A converge to a minimum. In the right plot, we can observe the improvement of the second phase, that brings the actual minimum of CE (dotted line), when L is used while L_A cannot provide any further improvement (continuous line).

The overall results of the 2-phase training, in terms of accuracy and average number of spikes (for the whole network and for the output layer only), are depicted in Fig. 2.13 and are compared with the same results for trainings that use only L or L_A. We can notice that the resulting accuracy of the 2-phase training is consistently the best (except for the lowest values of α), while the spike rate is kept under control. A lower spike rate is only found for the training that uses only L, but this is due to the switching off of some neurons that in turn precludes a good accuracy.

Table 2.3 reports the accuracy, the cross-entropy, and the average number of spikes per neuron and per inference that are obtained in the best case for the three loss functions adopted and for the 2-phase strategy. We also reported the values of the hyper-parameters that lead to such best cases.

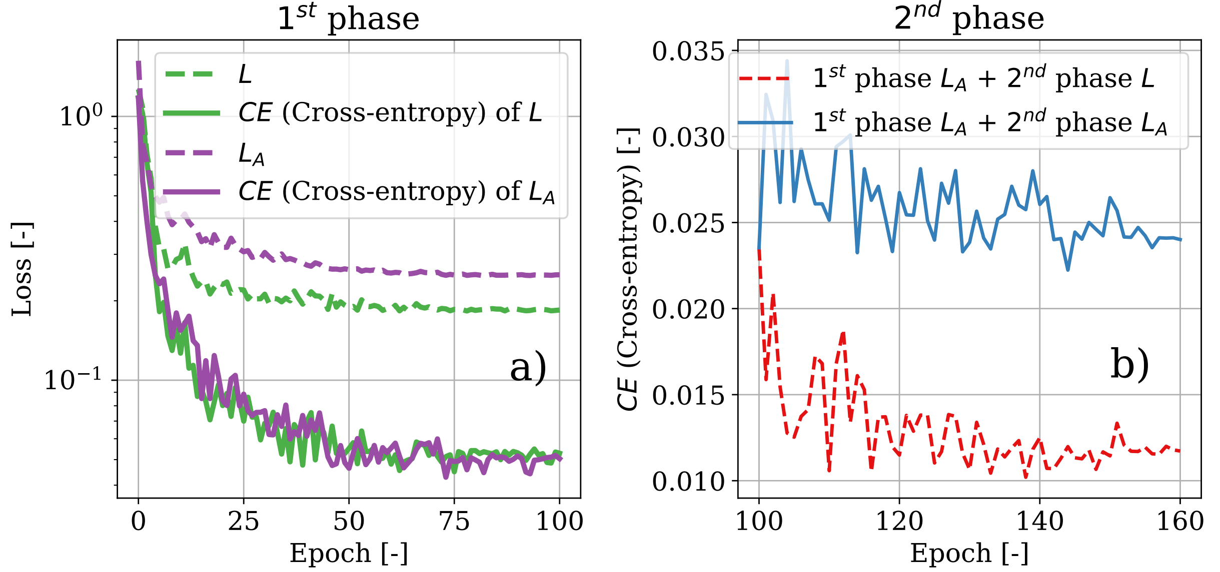

Figure 2.12: Yin-Yang - (a) Comparison between the total loss and the CE component of L and L_A during training and for $\alpha = 0.004$ and $\eta = 0.4$ ms. (b) CE component trends during the 2nd phase of training for L and L_A, started from the final result achieved for L_A in (a). The 2nd phase was performed with the same losses and parameters except for the learning rate, set to 2×10^{-4}. The final accuracy in the 2nd training phase improves, for the L case, from $\sim 98.4\%$ to $\sim 98.7\%$.

Table 2.3: Best results in terms of accuracy for Yin-Yang

Training	*Accuracy*	*Cross-Entropy*	$\langle N_S \rangle$	$\alpha \times 10^3$	η *[ms]*
L_W	98.30%	0.078	1.68	-	-
L	99.05%	0.074	1.52	4.0	0.2
L_A	98.90%	0.087	1.49	4.0	0.2
2-phase	**99.00%**	**0.053**	**1.44**	**4.0**	**0.3**

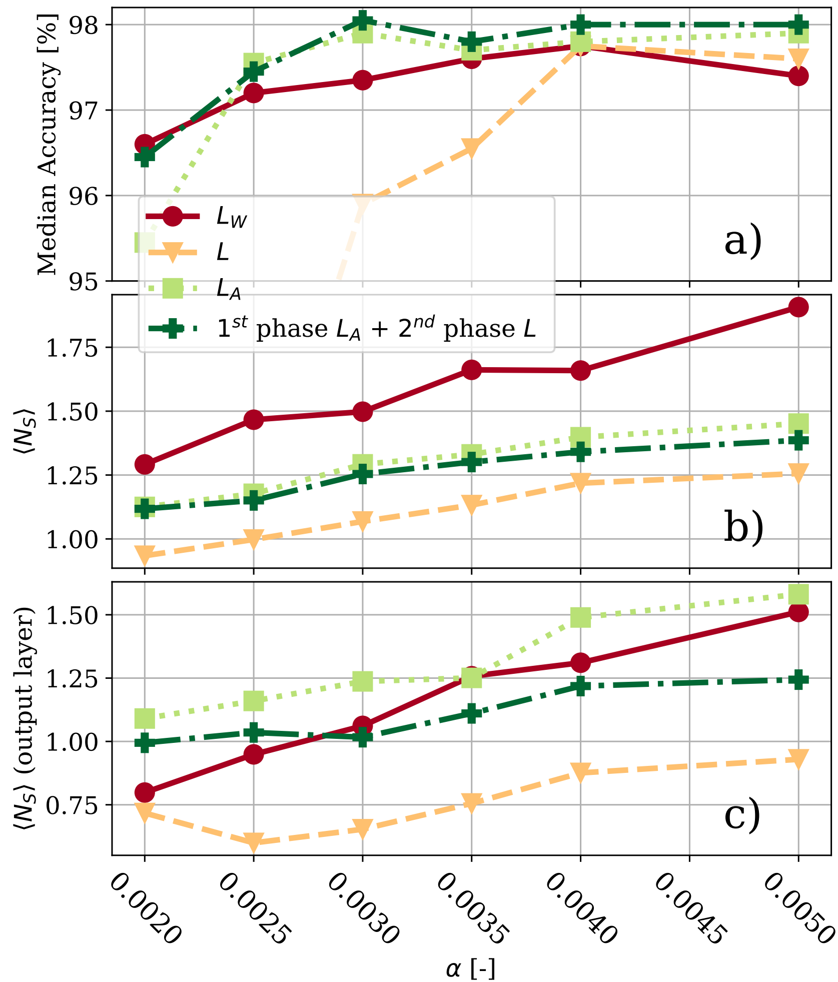

Figure 2.13: Yin-Yang - Summary of the training methodologies. (a) shows the median accuracy of the network, (b) shows the average number of spikes per neuron of the whole network, while (c) shows the average number of spikes emitted by the output layer. These results were obtained with $\eta = 0.3$ ms and averaged over 10 seeds per point.

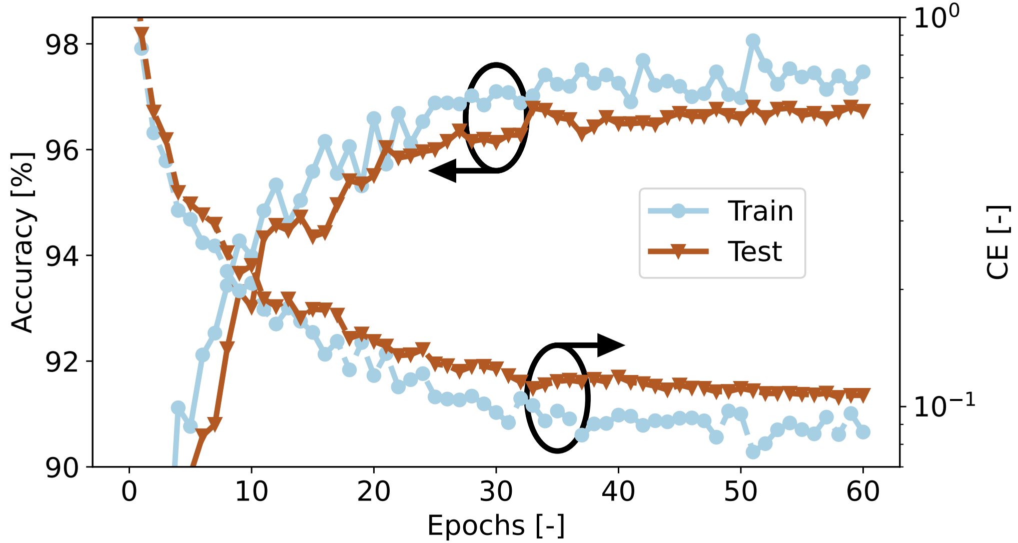

Figure 2.14: MNIST - General behavior of accuracy and cross-entropy during training. In particular, it can be observed the saturation of the two curves after 60 training epochs, without any anomaly on the test set, symptom of a parameter over-fitting.

The best results are obtained for L and for the 2-phase strategy, as expected, with a slightly better performance in terms of spike rate for the 2-phase learning. However, it must be considered that for the 2-phase learning the results in the table are easily obtained (all the trainings have converged to a good minimum of the cross-entropy), while for L only a small fraction of the tests has not incurred in the switch-off issue.

MNIST Simulation Results

For the MNIST benchmark a larger network, compared to the Yin-Yang benchmark, is needed to obtain a sufficient accuracy. In this respect, while the scope of this work is not a record breaking performance on the MNIST dataset, we still believe that any accuracy below 97% is not meaningful enough to be considered. Therefore, we adopted a 4-layer network with significant larger hidden neurons, as already mentioned in Section 2.2.1. Training such a network with the L_W or the L loss functions exacerbates the issues already discussed for the Yin-Yang benchmark, up to the point that the training becomes impossible. Since from the first epochs of learning, when a small value of α is used, the switch off involves so many neurons that the whole last layer becomes inactive. For larger values of α the behavior changes abruptly, with a sharp increment of the number of spikes emitted by the network resulting in a neural over-excitation. In both cases, the training cannot be completed with success: finding a value of αs that lies exactly between those two region of behavior is not feasible, even because the resulting behavior is strongly impacted by the initial condition too.

Using L_A as loss function, however, allows a smooth training for a large range of α values. We verified that with several simulations whose behavior is exemplified in Fig. 2.14 where data about a training are shown over time. Such a plot reports the accuracy and the CE computed on the training set and on the test set for the first 60 epochs ($\alpha = 0.003$ and $\eta = 3$ms are used). Both accuracy and CE behave as expected, with a smooth saturation towards the minimum of the loss function.

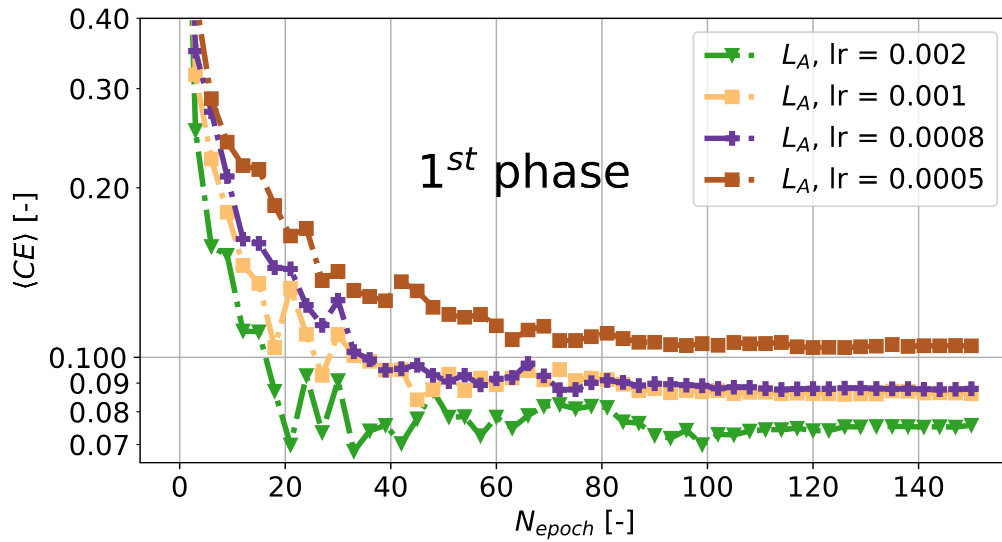

Figure 2.15: Preliminary analyses on MNIST: behavior of the cross-entropy during the initial 150 training epochs for multiple learning rates.

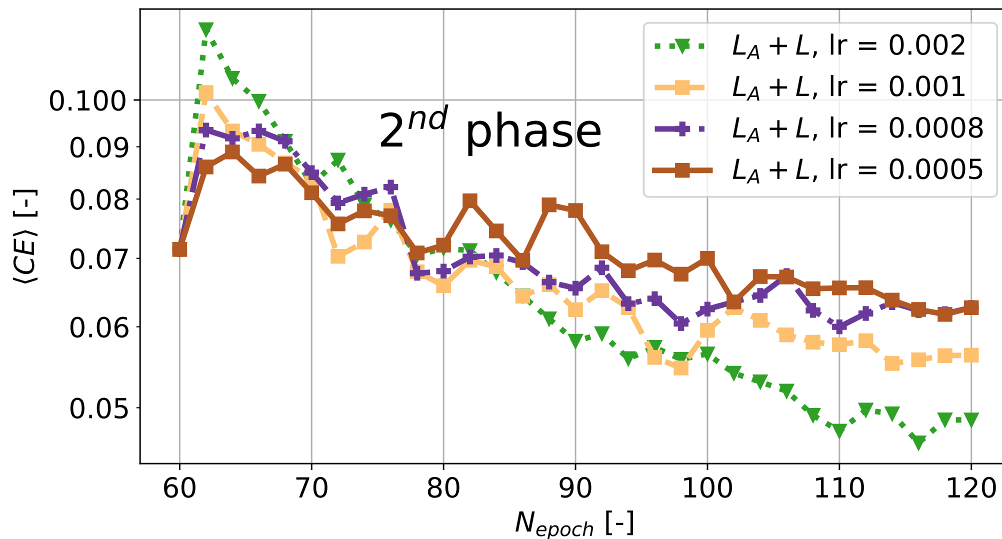

Figure 2.16: Preliminary analyses on MNIST: behavior of the cross-entropy in the second phase of training starting from the same initial condition (1st phase with L_A and lr $= 0.001$). The initial increment of the CE is due to the CE minimum discontinuity from L_A to L.

Table 2.4: Best results in terms of accuracy for MNIST

Training	*Accuracy*	*Cross-Entropy*	$\langle N_S \rangle$	$\alpha \times 10^3$	η *[ms]*
L_A	97.63%	0.081	0.82	3.0	3.0
2-phase	**98.42%**	**0.059**	**0.45**	**3.0**	**3.0**

Since L_A cannot provide the true minimum of the cross-entropy, the 2-phase learning has been used to perform the actual training of the network. In Fig. 2.15, the CE over time is shown for the first phase and for some learning rate up to epoch 150. Furthermore, Fig. 2.16 illustrates the CE behavior during the second phase, for the same values of the learning rate. We can notice how the CE, that has saturated in the first phase, can be further reduced when L is adopted as the loss function. Moreover, the nice decrement of CE in that second phase is also an indication that the neuronal switch-off has been avoided and that the training can improve during the second phase. This behavior is consistent for all the values of the learning rate we adopted. Therefore, to speed up the training in our further exploration, we choose to keep the same value ($lr = 2 \cdot 10^{-3}$) for both phases.

Fig. 2.17 shows the accuracy and the cross-entropy reached by the network at the end of the training for some values of the hyper-parameters α and η. Such a plot also shows the results for a network trained using L_A only. Plots show how the 2-phase learning can always converge to good results in terms of accuracy and cross-entropy, consistently better than the results obtained by L_A alone. We also recall that neither the L nor the L_W alone can be used to train this network because they do not reach proper convergence during the training.

Fig. 2.18 shows the average number of spikes per neuron that are required to perform a single inference. In the plot on the right, the average is computed over all the nodes, while in the plot on the left it is reported separately for each layer (for a single value of α). In both cases, the 2-phase learning keeps the spike rate well below the rate that is reached using L_A only, showing its efficacy on both the metrics, namely accuracy and spike rate.

Even for MNIST, we report, in Table 2.4, the accuracy, the cross-entropy, and the average number of spikes per neuron and per inference that are obtained in the best case for the usable learning strategies. Again, the 2-phase strategy reaches superior results, in terms of accuracy and CE, while keeping the spike rate under control.

Finally, we also considered different topologies of the network, to ensure that the effectiveness of the 2-phase learning is not restricted to a single case. We trained three other networks, after choosing the α and η that provided the best results, and compared the accuracy of the 2-phase learning against the one obtained using L_A only. It turned out that the proposed strategy is consistently effective in reaching good a accuracy for all the cases that we have explored.

2.2.5 Neuron to neuron variability on MNIST benchmark

The statistical distribution of the parameters in real neuromorphic devices due to the device-to-device variation should be considered before the design of neuromorphic chips to quantify the robustness of these networks in terms of accuracy and CE. However, nowadays it does not exist a standardization of neuromorphic circuits and relative implementations: this lack of standards

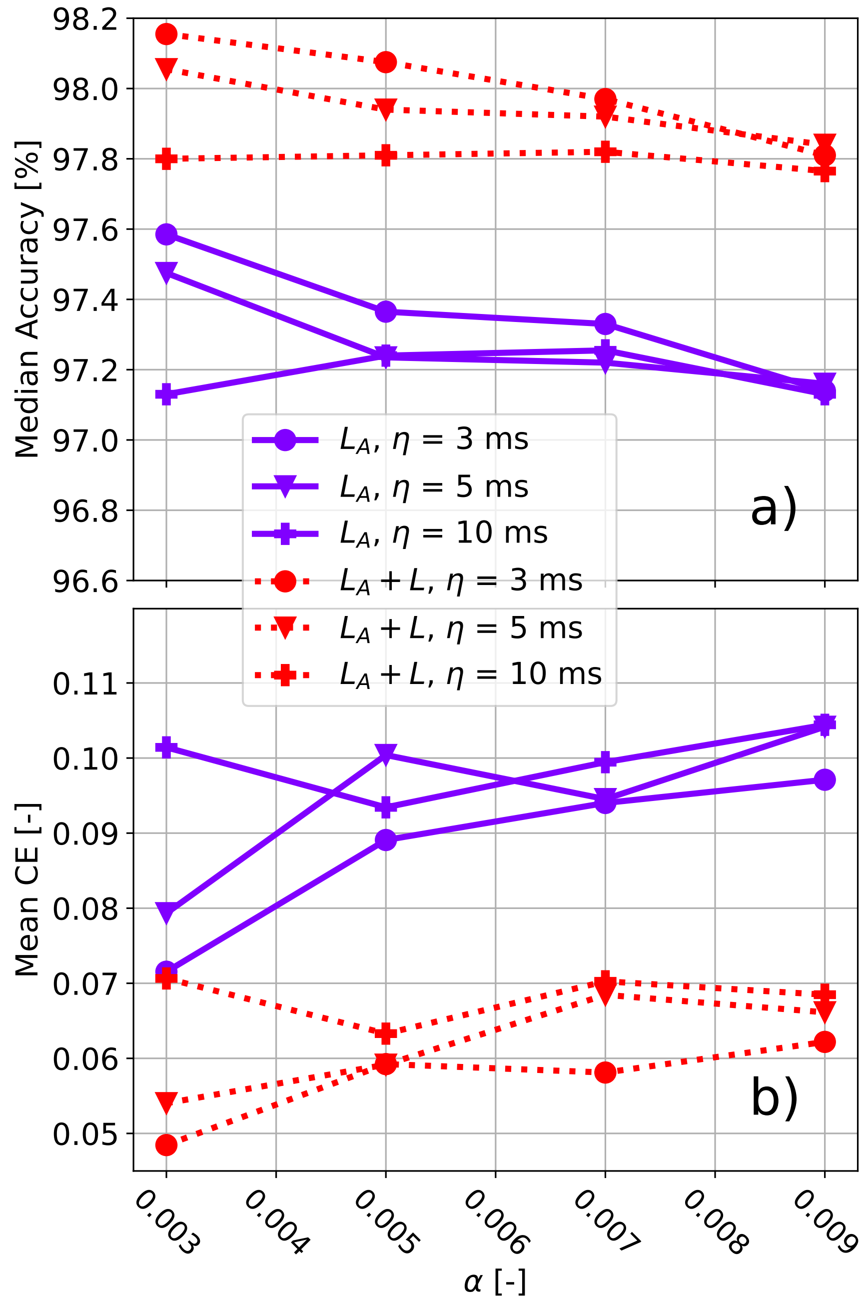

Figure 2.17: MNIST - Final training results of MNIST. (a) Median accuracy of the network and (b) mean cross-entropy for both training phase and different α values.

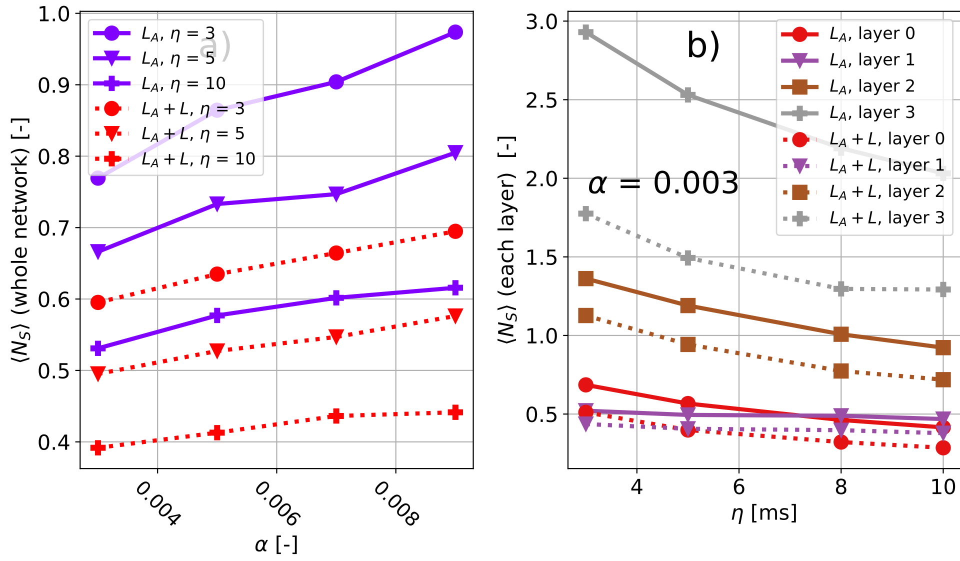

Figure 2.18: MNIST - Comparison between (a) the overall average number of spikes produced by the network and (b) the single layer components for the $\alpha = 0.003$ case. (a) shows an average number of spikes produced by the whole network much below the threshold value of 1 spike per neuron due to the strong dependence on the average behavior of the first (280 neurons) and the second layers (160 neurons). The network switch-off is prevented by the excitation of the last layer (layer 3, plot (b)) which doesn't affect too much the results of (a).

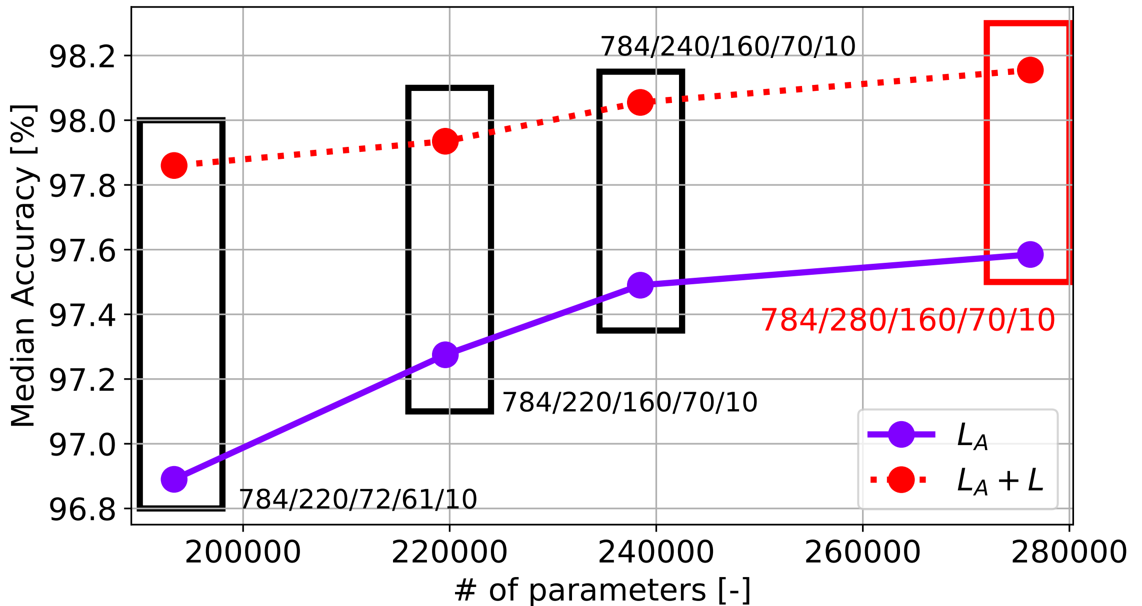

Figure 2.19: MNIST - Final accuracy after the first and the second phase of training for multiple topologies of the network and, in particular, for a different number of networks' parameters. The average accuracy results obtained for the reference network are highlighted in red. In this figure, the hyper-parameters are fixed for all networks, and in particular $\alpha = 0.003$ and $\eta = 3$ ms.

prevents a direct comparison of neuromorphic models exploiting, for example, a circuit-level Monte Carlo analysis. On the other hand, an approximated estimation of the behavior of the circuit can be made by increasing the considered abstraction layer from the circuit level to the behavioral one. Therefore, this paragraph introduces the impact of noise on neuromorphic implementations as a study of a deviation of the neural parameters from their nominal values and it tries to quantify the behavior of the MNIST classifier network with a different degree of uncertainty. This, as a general analysis, is not strictly linked to a circuit implementation, but it gathers all the uncertainties of a circuit on deviations of the membrane potential constant (τ_m), and the synaptic device constant (τ_s). Until now, all the reported results neglect the presence of device variability, however, the following analysis shows the dependence of some results reported in Fig. 2.13 when noise is injected during the inference or the training process, as an abstraction of electronic circuit uncertainties.

The analysis presented in this section considers two different methodologies: the first one is meant to mimic real offline training followed by an online inference, while the second study is made by injecting noise before the training process and then testing the network with the same observation of neural parameters.

Therefore, in the first case, by exploiting the trained network for the MNIST dataset whose results are reported in Tab. 2.4, a Gaussian noise is injected just before the inference of the MNIST test set. The inference results are gathered in Fig. 2.20 and, in particular, those results are calculated by selecting the most accurate set of weights among 10 seeds of fully trained *SNN* without any parameters' variability; then the resulting network is tested on the full test set by injecting a null-mean Gaussian distributed noise with a relative standard deviation σ_{τ_m} and σ_{τ_s} on τ_m and τ_s respectively. This, is for multiple values of α and η. As it can be observed in Fig. 2.20(a) and (b), the accuracy decreases as much noise on the model's parameters is injected, while, Figs. 2.20(c) and (d) also show a more prominent impact of the noise on τ_m compared to the one injected on τ_s. In fact, a substantial deviation of the membrane potential time constant from the nominal τ_m value impacts not only the mean final accuracy of the system but also increases its standard deviation producing an unpredictable network. In addition, Fig. 2.20 also shows a slight dependence of the η hyperparameter on the accuracy, while α seems not to produce a noteworthy trend for the tested values. In summary, the results reported in Fig. 2.20 show a strong degradation of the system accuracy in presence of noise, especially on the neural parameter τ_m. This is due to a strong decoupling of the network's hyperparameters before and after the noise injection: in the adopted scheme, it does not exist any network's parameters adaptation methodology to overcome the hyperparameter mismatch. In this perspective, on the other hand, the second study aims to quantify the network's accuracy loss when the training process takes into account the specific statistical realization between training and inference.

To simplify the whole picture, this second analysis spans over multiple values of σ_{τ_m} and σ_{τ_s}, but takes into account just one set of hyperparameters ($\alpha = 0.003$ and $\eta = 3$ ms). Interestingly, by injecting noise into the training process and then testing the obtained network with the same observation for the model parameters τ_m and τ_s, the accuracy results of Fig. 2.21 tend to match the one reported in Fig. 2.17 with just an average drop of the accuracy of 0.3% in the $\sigma_{\tau_m} = 0.2$ case. Moreover, the deviation on the synaptic time constant does not produce any relevant difference during training. These results prove the great adaptability of the training process in case of device-to-device constant mismatches, however, training the network with a single observation does not

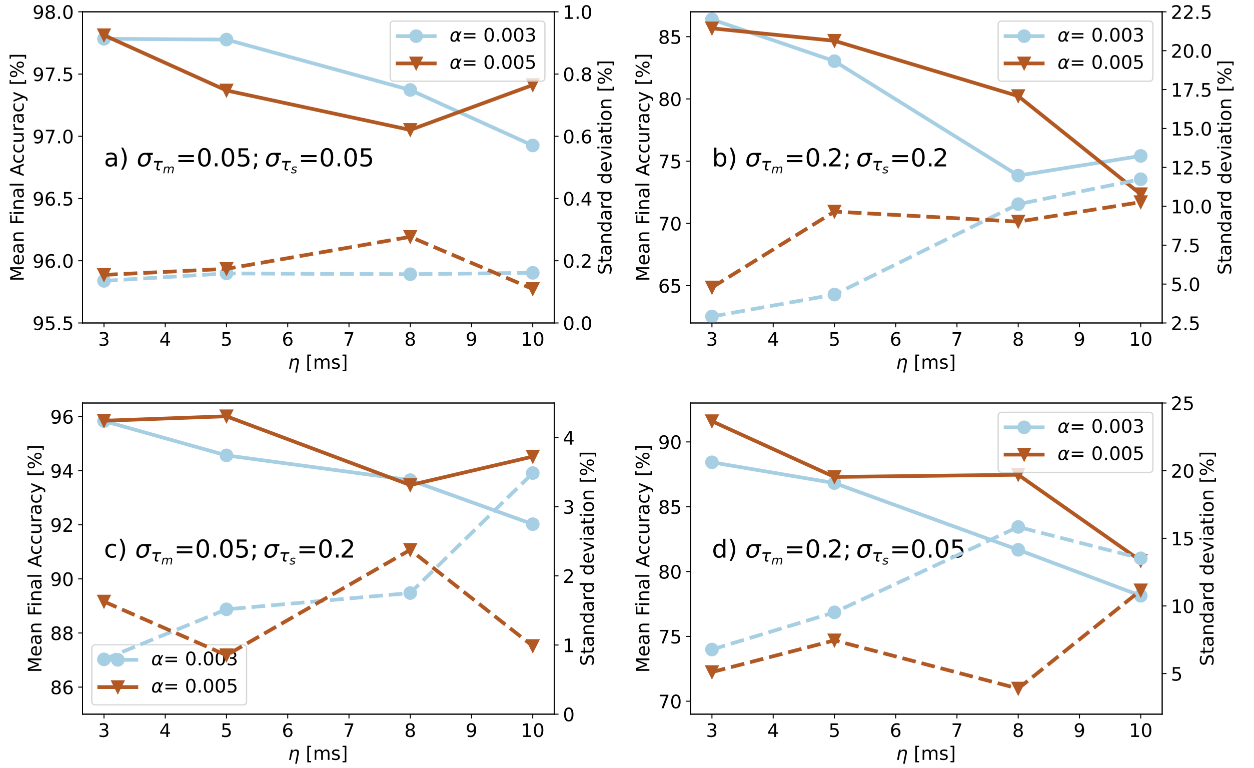

Figure 2.20: MNIST - Mean accuracy after the 2nd phase of training (solid lines), and relative standard deviation (dashed lines), in presence of constants mismatch between each neuron of the network for multiple values of η and α. σ_{τ_m} and σ_{τ_s} represent the standard deviation of the Gaussian noise applied on the neural constants τ_m and τ_s respectively. These results are obtained by averaging 10 realizations of noisy parameters on the nominal networks. The results reported in this figure reduce to the one shown in Fig. 2.17 when σ_{τ_m} and σ_{τ_s} are both equal to 0. This plot has been obtained with $\alpha = 0.003$ and $\eta = 3$ ms.

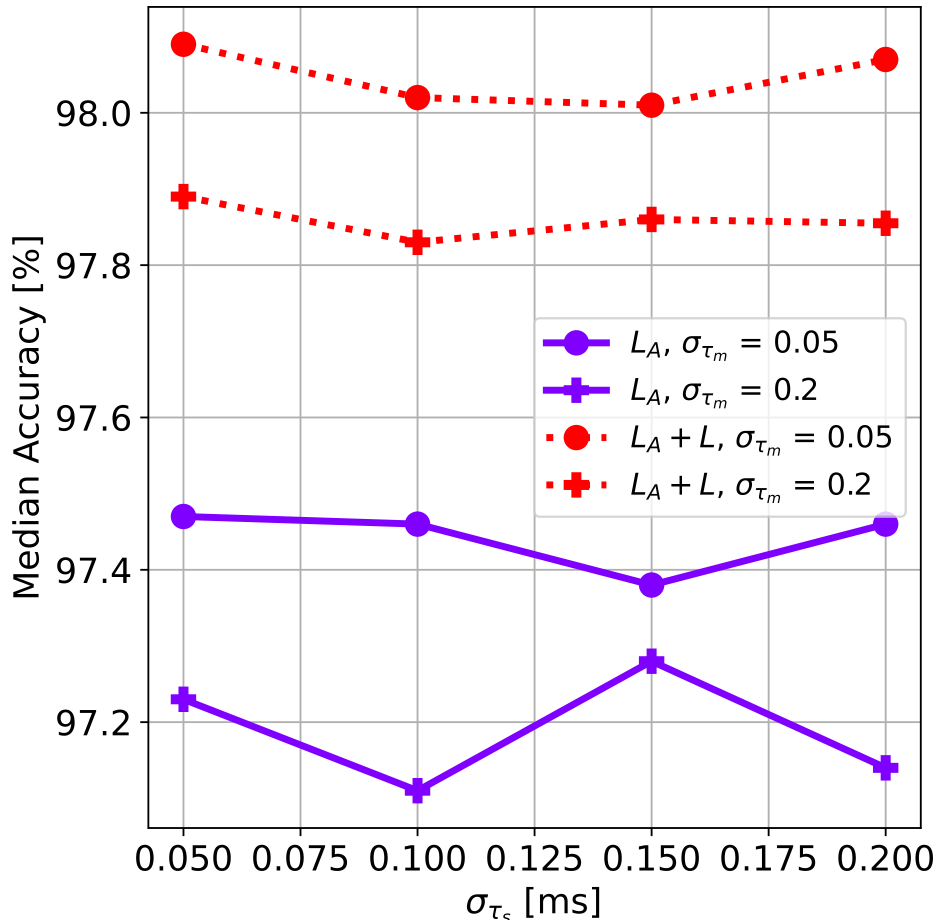

Figure 2.21: MNIST - 1st and 2nd phase of training accuracy results of an SNN classifier with statistically distributed parameters. It can be observed that the system, even in presence of not nominal neurons and synapses time constants, converges towards the accuracy results presented in Fig. 2.17 if tested with the same realization of the training. In particular, this plot has been obtained with $\alpha = 0.003$ and $\eta = 3$ ms.

solve the problem raised when the weights are applied to a system with a different statistical observation of the time constants. In fact, by testing the networks of Fig. 2.21 with a different parameters realization, the accuracy of the system converges towards the results showed in Fig. 2.20. Even though this critical scenario represents a hindrance to the offline-training-online-inference mechanism, it could be mitigated by exploiting a localized learning methodology, like STDP [76], directly implemented on the neuromorphic chip. Online learning methodologies, governed by local learning mechanisms, may sew up the tear between the offline-trained network and the localized one, by adjusting the weights or the local hyperparameters of the network in order to obtain a reliable and accurate system.

Chapter 3

Overview of neuromorphic hardware implementations

In the previous section, we introduced *SNNs* and novel learning methodologies to improve the accuracy and energy efficiency of these networks in a classification task. In this chapter, we will briefly review the state of the art in neuromorphic hardware implementations to understand and evaluate ways to further reduce energy consumption. The field of the hardware implementations of neuromorphic systems has widely enlarged in recent years, consequently, our review will be limited to the aspects most tightly linked to the scope of this thesis.

3.1 Devices and Circuits for neuromorphic applications

Neural models and learning methodologies are the fundamental building blocks of *SNNs* and neuromorphic engineering: by combining different neural models with bio-inspired learning algorithms, researchers and engineers all over the world have designed and developed many neuromorphic networks and platforms [99–101]. Nonetheless, by looking at the state of the art of modern neuroscience and chip design, we see that nowadays we can build CMOS circuits that replicate the operation of biological neural networks [102, 103]. However, the manufacturing technology is still lagging behind biology in terms of transistor area and wiring resources. In particular, we are still far from the density of three-dimensional biological systems and their self-assembling capabilities. In general terms, the features of artificial neural systems are quite different from their biological counterparts. For example, we can design very fast circuits compared to biological timescales. While biology operates at kilohertz scales, our CMOS circuits operate at gigahertz. We can design reliable circuits that operate deterministically with high precision, while in the brain the individual synapses and neurons are neither very precise nor very reliable, and the successful operation of the system largely benefits from a massive redundancy. Moreover, the highly parallel and event-based computation of *SNNs* does not fit too well conventional computers due to the aforementioned limitations of the general-purpose von-Neumann architectures [26, 43]. Hence, the computational platforms must be re-thought in order to improve energy efficiency and maximize parallelization [104].

In this perspective, two different schools of thought have been raised over the years: the first one focuses on *simulating* neural networks in a full-digital ad-hoc platform, while the second one aims to integrate the whole neuromorphic cores, including analogical neural dynamics, into *VLSI* mixed-

signal chips [105]. The full-digital simulating platforms have a high potential for solving machine-learning tasks, but they do not emulate directly the dynamics of real neural systems. Conversely, the mixed-signal platforms embed an analog core that can reproduce quite closely the biological signals with very promising energy efficiency.

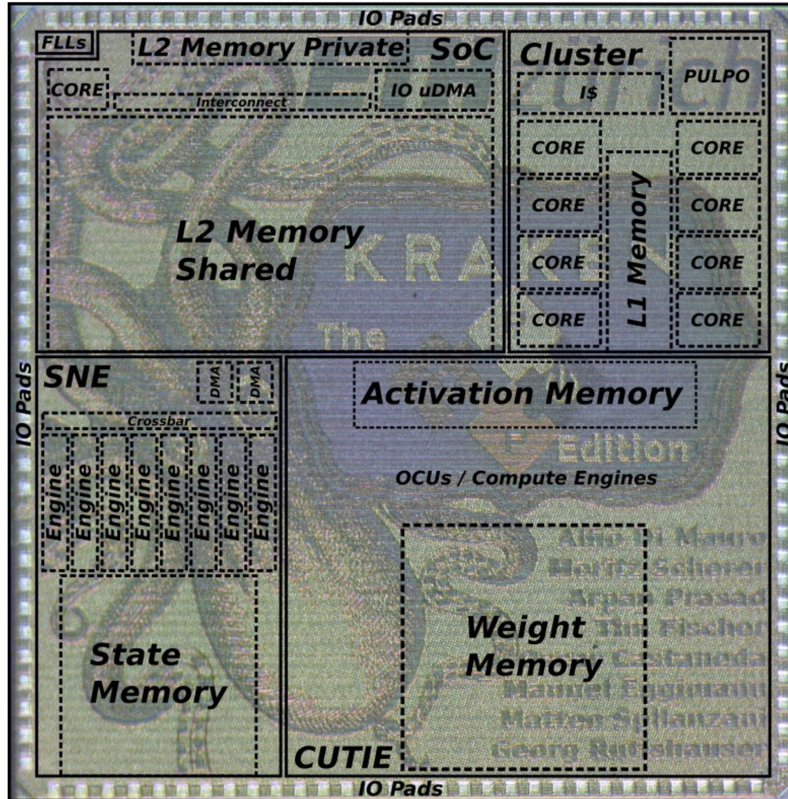

Figure 3.1: Die and logic subdivision of Kraken neuromorphic SoC [62]

Digital neuromorphic processors show a more mature development: the *SpiNNaker* system, for example, is a multi-core computer designed with the goal of simulating very large numbers of neurons in real-time [106]. The SpiNNaker machine is a stack of 600 printed circuit boards each carrying 48 SpiNNaker processors, which supports the simulation of hundreds of millions of neurons. The key innovation in the SpiNNaker architecture is the communications infrastructure, which is optimized to carry very large numbers of very small packets. Differently from the SpiNNaker project, the TrueNorth architecture proposed by IBM in 2014 is a fully-digital neuromorphic system that integrates on the same chip 4096 cores, each comprising pure digital asynchronous circuits able to simulate 256 neurons with 256 $\times$ 256 synaptic connections [107]. Moreover, the last-born is *Intel Loihi* [108]. The Loihi platform is a 60-mm^2 chip fabricated in Intel's 14-nm process. This platform integrates 128 neural cores, three embedded x86 processor cores, and off-chip communication interfaces that hierarchically extend the mesh in four planar directions to other chips. A synchronous network-on-chip transports all communication between cores in the form of packetized messages. Loihi, a fully digital architecture, approximates the continuous-time dynamics using a fixed-size discrete time-step model. In this model, all neurons need to maintain a consistent understanding of time so that their distributed dynamics can evolve in a well-defined, synchronized manner. A different approach was followed to design the *System-On-Chip (SoC)* called Kraken [62]. The Kraken *SoC* (see Fig. 3.1) is built around a 32bit fabric controller RISC-V core, it hosts 1 MB of SRAM

memory, many standard peripherals to interact with the external world and it enables highly sparse event-driven sub-µJ/inf *SNNs* inference on a dedicated neuromorphic energy-proportional accelerator. In particular, it elaborates spiking convolutional neural networks with 4bit 3x3 kernels and 8bit *LIF* neuron states. It also implements a "Completely Unrolled Ternary Inference Engine" accelerator designed to maximize energy efficiency by minimizing data movement during inference. It is able to run complex visual tasks like object detection, tracking, drone navigation and obstacle avoidance, at high speed and robustness, under tight payload and power constraints.

On the other hand, an intriguing example of *VLSI* integration of analog-digital interoperability is included in Dynap-SEL [103], where analog neuromorphic cores are responsible for the integration and generation of spikes, while the digital part distributes the events and links different neural layers throughout the circuit in a scalable and efficient manner. Dynap-SEL was fabricated with a 28 nm Fully Depleted Silicon on Insulator (FDSOI) process. The core in the Dynap-SEL device comprises 64 analog neurons. Each neuron has 128 mixed-signal plastic synapses, 64 mixed-signal non-plastic synapses and 4 linear synapse circuits. Up to 16 different chips can be easily combined and merged together to form a 4×4 chip core. Moreover, its neural circuits and the spiking distribution protocol were introduced in [109]. In particular, the MOSFET transistors of these low-power circuits are operated in the subthreshold regime, so that the main mechanism of carrier transport is diffusion, as it is for ions flowing through channels across neuron membranes. Thanks to the subthreshold operation, transistor currents can be lowered down to a range between femto to nano Amperes. Consequently, circuits have biologically realistic time constants, ranging from tens to hundreds of milliseconds, with reasonable capacitance values from a silicon integration standpoint. Moreover, a digital event-based distribution representation and signal communication protocol, called *Address-Event Representation (AER)*, has been included to improve flexibility and energy consumption[103].

In this representation, a digital word address is assigned to neurons and transmitted as soon they produce an event using asynchronous digital circuits. Information is therefore encoded in both spiking time and neural address. Unlike classical digital logic circuits, *VLSI* implementations of *SNNs* are typically characterized by very large fan-in and fan-out numbers, due to the huge number of neurons composing the network (e.g., we have observed in chapter 2, *SNNs* composed by layers of hundreds of neurons and connected by thousands of synapses). It is consequently important to design neuromorphic computing platforms that can be configurable to support different network topologies, neurons and synapses. In this respect, *AER* routing plays a crucial role in internal communication schemes improving the scalability of neuromorphic systems. Indeed, some of the main bottlenecks in the construction of large-scale re-configurable neuromorphic computing platforms are the bandwidth, latency, and memory requirements for routing address-events among neurons [103].

Another recent implementation of a hybrid spiking generation and communication scheme, is BrainScaleS-2 [110]. The BrainScaleS-2 core consists, in fact, of a full-custom analog core combining a synaptic crossbar, neuron circuits, analog parameter storage, two digital plasticity processors, and the event routing network responsible for spike communication. Authors, indeed, refer to the training-tuning-inference process of the BrainScaleS-2 neuromorphic circuit as an "in-the-loop" operation. This operation principle uses the “hybrid plasticity” scheme proposed in [111], combining analog measurements with digital calculations. This increases the flexibility, while keeping the

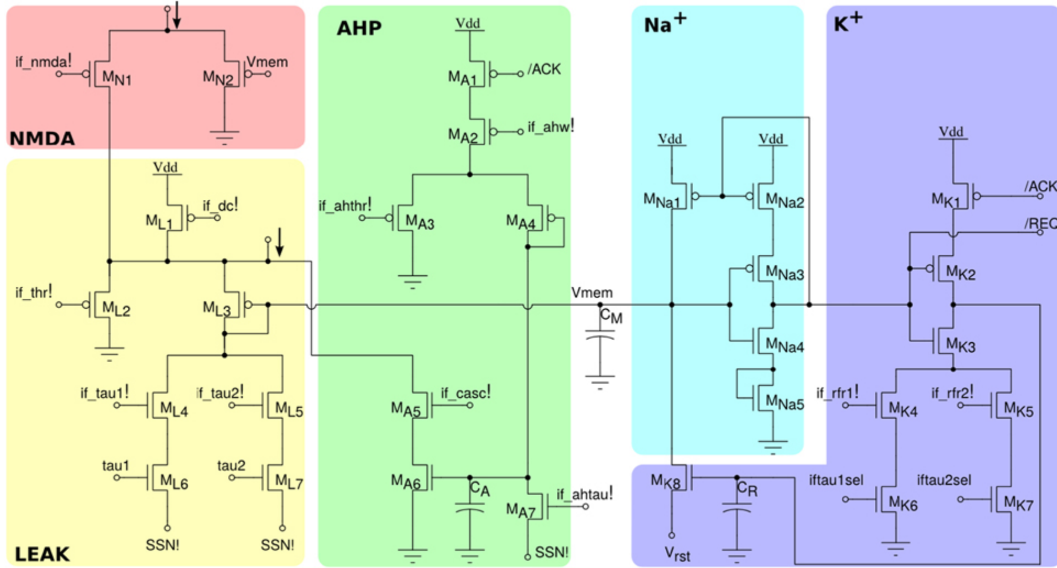

Figure 3.2: Schematic that implements the adaptive exponential Integrate-and-Fire (I&F) neural model in [104].

advantages of an accelerated physical model, like simultaneously observing all correlations between pre-and postsynaptic signals. In fact, as proposed by Friedmann et al. in [111], it is possible to improve the flexibility of neuromorphic platforms while keeping the high energy efficiency associated with neuromorphic implementations, by combining a general-purpose processor with full-custom analog elements. In this way, backpropagated learning methodologies, such as the one exploited in chapter 2 of the present manuscript, can be actively used to train physical neuromorphic circuits. In particular, as stated in [89], the precise gradient of the loss function in deep *SNNs* can be obtained by simply sampling the value of the neural input current at each spiking event. The flexibility of this methodology makes it possible a batch mode operation. In this mode, experiment instances are queued and sequentially executed on the system, without any data dependency among the different instances.

Another important step ahead into integration of neuromorphic methodologies into *VLSI* circuits was proposed in [104], where an adaptive exponential I&F neuron was successfully integrated into a neuromorphic chip called ROLLS, developed in standard 180 nm CMOS. The neural model is an adaptation of the *LIF* that can exhibit a wide range of neural behaviors, such as spike-frequency adaptation properties, refractory period mechanism and adjustable spiking threshold mechanism [112]. The functionalities implemented in the neural circuit schematic shown in Fig. 3.2 include a neural leak conductance, a backtracked spike-frequency adapter working together with a positive feedback block in order to model the biological neural Sodium activation and inactivation channels during the spike production, and finally a block which models the effect of the Potassium conductance, namely resets the neuron and implements a refractory period mechanism.

Moreover, some of the aforementioned learning algorithms have been integrated together with electronic neurons in physical mixed-signal *SNNs* . For example, a spiking neuron combined with synapses having *STDP* capabilities has been recently validated experimentally [113]. In that work, Polidori and co-workers show an implementation of a fully analog *LIF* spiking neuron compatible with a standard CMOS process combined with an online learning method with long-term memory elements. The synaptic weights are stored as a charge in the floating gate of a standard MOS

transistor, where the charge is modified using only the Fowler-Nordheim tunneling mechanism.

These mixed-signal analog-digital implementations of neuromorphic circuits generally exploit low-power transistors to reduce energy consumption by emulating biological time scales [114, 115]. Moreover, the energy–per–spike is a useful figure of merit to evaluate these *VLSI* implementations and it highlights the high efficiency of these circuital implementations: in fact, many works report an energy consumption close to 10 fJ per spike [113, 114, 116, 117], that comes close to the estimated energy consumption of biological neurons [116].

Adapting the temporal properties of such circuits to the temporal dynamics of the signals being processed can save the area needed to store data and the state of the processing elements, because circuits operate in real-time directly on the signals being acquired by a sensor [43]. Moreover, by combining the adaptive analog signal processing strategies of these neuromorphic circuits with digital event-based asynchronous communication schemes, it is possible to build large-scale multi-core neuromorphic processors that combine the best of both worlds (analog and digital) for low-power signal processing, computation, and communication [114]. Although these analog implementations are able to mimic quite well the features of biological neurons in terms of functionalities and energy efficiency, they are still a subject of research.

Both these two *SNNs* computational paradigms, mixed-analog-digital or fully digital, are extremely parallel and limit the information exchange outside the elaboration chips to save energy. These two concepts are fundamental in neuromorphic computing and many *VLSI* implementations exploit cross-bar arrays to shrink the memory-related area and boost their energy-efficient characteristics. As discussed in more detail in the next section, cross-bar arrays are memory architectures that consist of crossed metal lines joint together by memory elements. These large memory arrays link the neural circuits inside the chip, therefore, the information encoded into spikes travels from one neuron to another without leaving the chip in which they have been generated. The information is consequently confined in a single neuromorphic platform largely reducing the aforementioned von-Neumann bottleneck.

3.2 From synapses to memristor device concepts

In spite of different designs and technologies adopted, both the approaches presented in the previous section (fully digital or mixed signal implementations) shares two fundamental aspects of neuromorphic computing to reduce energy consumption: the asynchronous spike sparsity and the locality. The first principle aims to improve energy efficiency by leveraging two different assets: sparsity in time and space. As previously stated in Sec. 2.2, the spike sparsity in time is directly linked to the neural emission frequency. In particular, in order to minimize the energy consumption of a neuromorphic device. On the other hand, the distributed elaboration of a Neuromorphic circuit is strictly linked to the locality principle prescribing that, differently from the Von-Neuman machine, the information should be elaborated as close as possible to the emission point. Therefore, the information enclosed in a single spike or a set of spikes should not be moved back and forth from the main memory of the system in order to be processed. In fact, in order to overcome the memory bottleneck of conventional computing machines [25], the Neuromorphic concept aims to move information inside only the chip itself. This can be done by exploiting cross-bar arrays (see

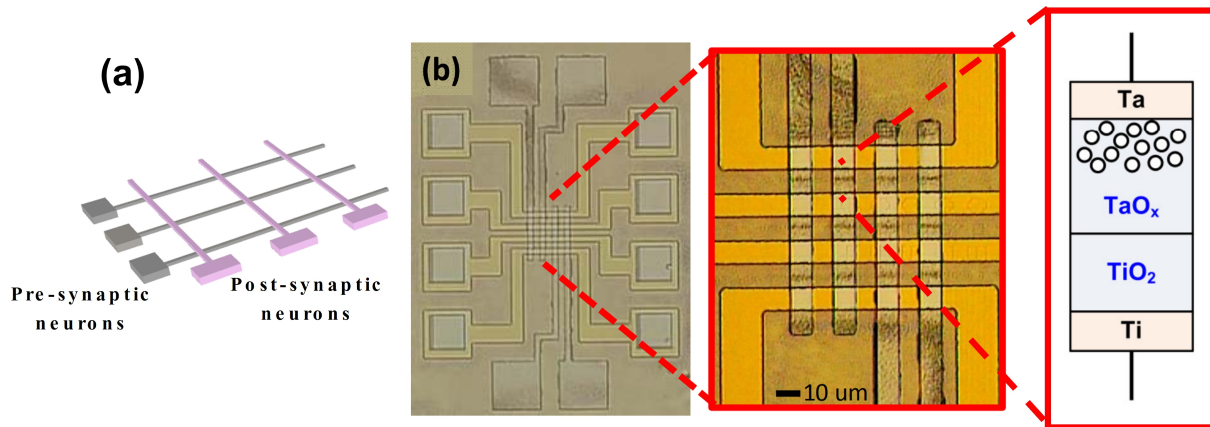

Figure 3.3: (a) Schematic illustration and (b) top-view optical microscope image of the 3D double-layer RRAM array reported in [120].

Fig. 3.3(a)), which link every pre-synaptic neuron to each post-synaptic neuron of a fully connected neural network [118, 119]. These cross-bar arrays are basically implemented with a bunch of cross interconnections where two lines are linked together with a memory element called 'synapse' (from the biological nomenclature). Each synapse represents a weight of the neural network. Regardless of the technology adopted into the neural design (fully analog or mixed analog-digital), a cross-bar connects two or multi neurons by keeping the information inside the same chip.

Generally, the rows of crossbar arrays are driven by spikes and integration takes place at each column [120]. In such implementations, e.g. the one illustrated in Fig. 3.3, the synaptic weight function is adjusted by using the number, the amplitude or the duration of potentiating and depressing pulses. By doing so, biologically plausible mechanisms, such as *STDP*, can be directly applied to potentiate or depress the synaptic weight thus enabling online training mechanisms. This is also beneficial in reducing read power and mitigating sneak path problems [74].

On the other hand, the circuital implementation of cross-bar arrays proposed in [74] differs from the previous one in that it employs the cross-interconnection as a vector-matrix multiplier. In particular, this alternative circuit modulates the neural pre-synaptic spiking trace with the memristor weight matrix. Using this strategy, a single trace supports both inference and learning, improving noise immunity and also device-to-device variation. Figure 3.4 depicts the details of the learning circuits in a crossbar-like architecture. This circuit is compatible with the use of the aforementioned *AER* as the communication scheme for mixed-signal neuromorphic circuits [103, 121].

In particular, in this circuit, a Differential Pair Integrator generates a continuous current starting from the neural spiking activity. Then this current is translated into a voltage thanks current-voltage converter block. After that, a buffer stage decouples the input stage from the synaptic sub-circuit which sees a differential voltage between the input and the pinned voltage of the output stage. Consequently, the read current observed by the output stage is proportional to the sum of resistive values of the memristors in that specific column.

In both implementations of the previously exposed cross-bar arrays, an important role is played by the *memristor*. The concept of *memristor* (contraction of *memory resistor*) was firstly introduced in [122] and then extended in [123] in response to the axiomatic need for completeness of relationship

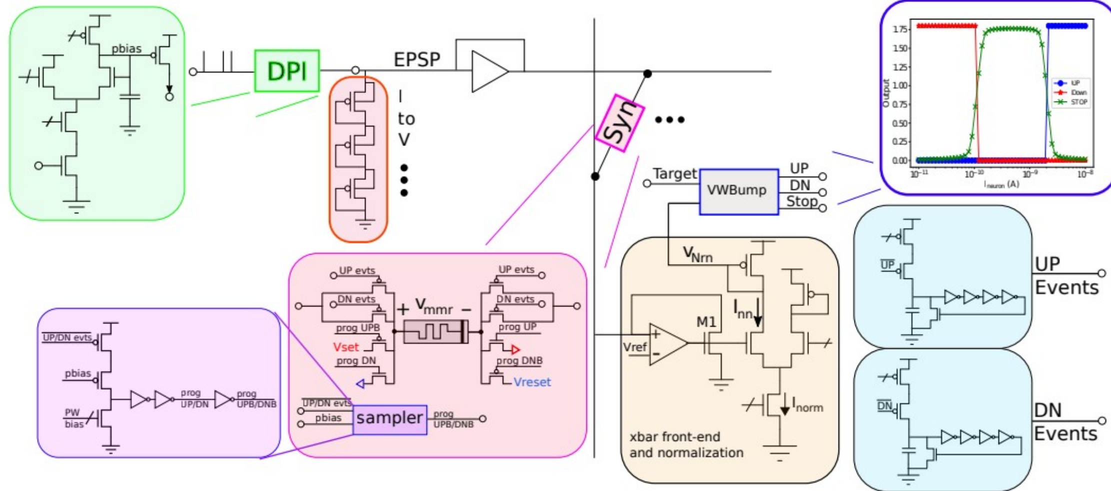

Figure 3.4: Circuit of Vector-Matrix cross-bar multiplier array reported in [74].

between the four fundamentals circuit variables (current i, voltage v, charge q and flux-linkage ϕ). In particular, two of them are covered by the definition of charge ($q = \int idt$) and flux-linkage ($\phi = \int vdt$), while other three relations are given by the definition of resistance ($v = Ri$), capacitance ($dq = Cdv$) and inductance ($d\phi = Ldi$). Therefore, L. Chua in [122] postulated the existence of the 'memristor' as the fourth missing two-terminal circuit element characterized by a ϕ-q relationship. In particular, a charge-controlled memristor shows a proportionality between voltage and current ($v = M(q)i$) defined by:

$$M(q) \triangleq \frac{d\phi(q)}{dq}.$$

For the sake of clarity, the above-mentioned definition can be seen as the relationship of a common resistor whose resistance $R = f(w, q, t)$ presents memory effects and changes its value against an internal state variable w, which can be described as a first-order dynamic equation [123]:

$$\frac{dw}{dt} = f(w, q, t)$$

Among all proprieties listed in the work of L. Chua, the pinched hysteretic $v - i$ curve depicts the general behavior of these devices: fig. 3.5 shows the numerical results of a simulated memristor attached to a periodic voltage input. As it can be observed, the pinched curve implies a null current through the device at an applied voltage of 0V. Moreover, by increasing the input signal frequency, the hysteretic curve tends to shrink, reaching the linear characteristic of a simple resistor at the limit of infinite frequency. This general behavior has been observed in several structures and different physical effects can be involved in this peculiar resistance modulation [124–127].

Even if the ability to perform computing at the site where data is stored – also called in-memory computing – was first proposed in 1960 by [128] in the digital domain, nowadays memristive devices, by exploiting different physical phenomena (e.g. spin, phase transition, ferroelectricity, or ionic transport), have been proposed and developed as key concepts to reach unprecedented targets for energy efficiency in edge computing as synaptic devices and non-volatile memories.
Moreover, a modern neural network presents hundreds of thousands of weights and, consequently,

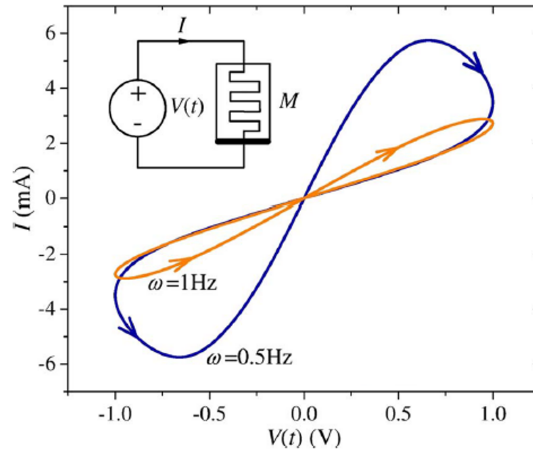

Figure 3.5: Simulated pinched I-V curve of a memristor reported in [123]. By increasing the frequency of the voltage generator, the memristor shows a more linear relationship, similar to a simple resistor.

the energy associated with reading and programming these devices should be very small, as their area occupation. To fulfill these requirements, synaptic devices should be organized into large arrays possibly exploiting the *BEOL* integration of the chip, thus saving silicon area. Thanks to the *BEOL* integration, the intermediate computing unit elements (representing the weights in a neural network) can be stored locally as the conductance of a single, non-volatile memristor. In order to be attractive as synaptic elements of a neuromorphic system, a memristive technology should offer:

- Multiple resistance states, which are the basic quantized states of the memristor;
- Limited device-to-device and cycle-to-cycle variations. Even if biological neural networks show finite stochastic variability in the synaptic behavior, in electronic implementations large variability in the memristor characteristic may produce instabilities in the neuromorphic circuit and, consequently, a degradation of the performance;
- Linearity in the I-V relationship. This is necessary to make direct use of Ohm's law to compute, whereby electric pulses with different amplitudes can be conveniently used as a multiplication factor. This point could be mitigated by interrogating the conductance states of synapses with pulses of a fixed amplitude but different widths (or equivalently with different numbers of pulses), and measuring the output charge as the multiplication result;
- Linearity and symmetry in weight updating. This means that the synaptic weight can be increased or reduced in an approximately linear proportion to the number of input pulses;
- BEOL compatibility, due to the aforementioned great advantages offered by a *BEOL* implementation.

All these points should be taken into account during the design of a memristor-based neuromorphic circuit. However, a proper memristor design can not be carried out at device level only, it should be put in a broader context accounting also for higher abstraction layers. In this perspective, we leveraged the work in previous sections in order to put the operation and design of *FTJs* memristors in a better and more pragmatic perspective.

3.3 BEoL compatible memristors

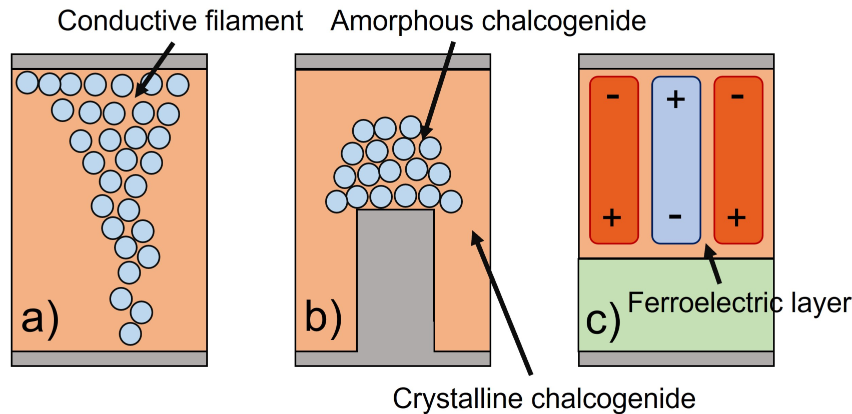

Figure 3.6: Sketch of memristive devices for neuromorphic computing. a) Filamentary RRAM device; b) Phase change memory device; c) FTJ device.

Memristors are two-terminal devices able to reversibly and gradually change their conductance. Material scientists, physicists and electronic engineers investigated many working principles and device concepts for memristors, and we here report a few examples::

- **Filamentary Resistive Switching Devices** - are a type of resistive random access memory (ReRAM) that utilizes a filamentary mechanism for resistive switching. They are characterized by their ability to switch between high and low resistance states through the formation and rupture of a conducting filament within the device. The filament is typically composed of metal ions or oxygen vacancies, and is formed through the application of an electrical current. The resistance of the filament can be modulated by adjusting the current, allowing the device to be used for data storage. In particular, the conductive filament can be anionic (the filament is composed of oxygen vacancies deriving from a reduction reaction) or cationic (the filament is composed of metal ions that migrate from one of the electrodes into the oxide layer) [129, 130].

- **Phase Change Memories** - are based on chalcogenide alloys that can change their state between a crystalline and an amorphous state showing low and high resistivity, respectively. The phase transition is thermally induced. To achieve an amorphous state, the alloy has to be heated above its melting point and then rapidly cooled down. Crystallization is instead achieved by heating the material above its crystallization temperature for a long enough time, so as to allow the atoms in the chalcogenide to rearrange in an ordered crystalline lattice. Phase change memories offer high endurance, fast write speeds, and low power consumption compared to other non-volatile memory technologies, such as flash memory. They are also scalable and can be integrated with other devices in a 3D structure [131]. However, PCMs also have some limitations such as low endurance and poor performance at high temperatures. Moreover, depending on the chalcogenide material, melting temperatures up to 600 °C have to be reached. As a consequence, relatively high currents have to be used to reach the melting temperature (in the order of hundreds of µA) [132, 133].

- **Ferroelectric devices** - are based on the ability of certain materials, known as ferroelectrics, to switch between different polarization states, when an electric field is applied. This polar-

ization switching can change the band bending and thus the resistance of the device, which thus becomes capable of storing data. Ferroelectric memristors have the potential to offer high density, low power and fast data storage and computing capabilities. Three examples of ferroelectric materials that have been used to create ferroelectric memristors include zirconate titanate (PZT), barium strontium titanate (BST) and hafnium oxide (HfO_2). Some examples of ferroelectric memristor devices are *Ferroelectric FET (FeFETs)* or *FTJs* [134, 135].

The above-mentioned memristor technologies are more suitable for neuromorphic computing compared to, for example, fully-CMOS static RAMs. In fact, the integration of memristive devices in the *BEOL* is considered a key enabler for neuromorphic computing applications, because it could significantly increase the effective memory density of a chip, without introducing major changes in the silicon substrate.

In this perspective, the discovery of ferroelectricity in the Hafnium oxides (HfO_2) [136] paved the way to a set of *BEOL* compatible, ferroelectric devices including Ferroelectric Tunnel Junctions and Ferroelectric FETs. The focus of the following chapter, in particular, is on *FTJs* as promising candidates for artificial synapses [137, 138]. In particular, chapter 4 introduces the baseline notions to model the ferroelectric polarization and the stacks consisting of a ferroelectric and a dielectric material. Then, we focus on the interplay between charge trapping and ferroelectric polarization in ferroelectric-dielectric stacks [139]. Finally, we conclude our analysis with some design guidelines to increase *Tunneling Electro-Resistance (TER)* [140] and maximum reading current in *FTJs* intended as artificial synapses.

Chapter 4

Memristors based on Ferroelectric Tunnel Junctions

In the previous chapter, we reviewed the literature on hardware implementations of *SNN* . With this understanding, we have chosen to focus on the study of *FTJs* as a potential solution for reducing energy consumption in neuromorphic hardware in this chapter, as they can be implemented as *BEOL* memristors in cross-bar applications. In particular, this chapter covers the role of ferroelectric materials in synaptic device concepts, from a physical perspective to the design of *FTJ* devices. In particular, firstly, Sec. 3.2 introduces the general memristor concept as a synaptic device. In fact, these tunable resistors, and in particular the ferroelectric-based ones, are considered valid candidates for in-memory computing applications. Therefore, Sec. 4.1 briefly explains the ferroelectric phenomena in order to introduce the *FTJs* working principle in Sec. 4.2. Then Sec. 4.3 reports a comprehensive modeling of a *MFIM* stack as baseline physical system for *FTJ* devices.

After the introductive part, in Sec. 5.1, after a model calibration between simulations and the experimental results reported in [141], we proposed a new design perspective to enhance *FTJ* devices. After that, in Sec. 5.2, we considered different conditions of compensation of the interfacial ferroelectric polarization in a quasi-static regime. Moreover, in order to reconcile *qualitatively* discrepancies between the spontaneous polarization versus applied voltage curves that emerged in the previous paragraph, we propose, in Sec. 5.3, a novel design technique that takes into account charge traps induced compensation of ferroelectric polarization at the interface between the ferroelectric and the dielectrc material. Last but not least, in Sec. 5.4, we conclude our analysis on *FTJs* by identifying the compensation conditions that enable an optimal operation of *FTJs*.

4.1 A brief introduction to ferroelectric materials

Ferroelectricity is observed in some crystalline or poly-crystalline materials (called ferroelectric materials), which show a non-linear response in polarization when a varying electric field is applied to the sample. In particular, ferroelectricity was firstly observed in 1921 by analyzing a "Rochelle salt" (i.e. potassium sodium tartrate tetrahydrate) sample. The name ferroelectricity was chosen after the observation of the typical hysterical voltage-polarization characteristics, very similar to the behavior of ferromagnetics in a variable magnetic field.

A ferroelectric material shows, in general, two stable states of the electric dipole moment, referred to as *remnant polarization (P_R)*, when zero voltage is applied externally. Moreover, it is possible to switch between them by sweeping the external field. In fact, near the *coercive field (E_C)*, the polarization switches between the stable polarization states, see Fig. 4.1. Ferroelectricity depends strongly also on temperature: at high temperatures, ferroelectricity first reduces and then it completely vanishes when the so-called Curie temperature is reached and surpassed, leading to the paraelectric phase of the material [142].

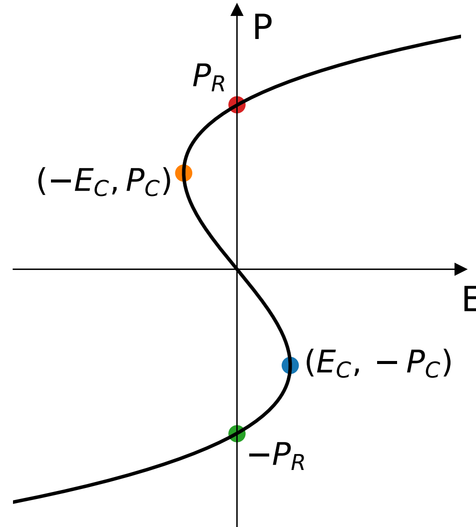

Figure 4.1: Typical "S" shaped curve corresponding to steady states points of Eq. 4.2 calculated thanks to Eq. 4.4 where P_R is the remnant polarization, while E_C and P_C detect the coercive point.

In a sample of ferroelectric material, it is possible to identify microscopic regions, called domains, where the remnant polarization is constant. Moreover, these regions are separated by an intermediate region called *domain-wall*: the characteristics of each domain tend to influence neighboring domains through a direct domain wall coupling term and an indirect coupling due e to electrostatic interactions [63, 142].

Nowadays large collections of ferroelectric materials have been discovered and employed in many devices and applications such as perovskites (e.g. $BaTiO_3$ or $PbZr_xTi_{1-x}O_3$). However, until a few years ago ferroelectricity didn't become very popular in modern silicon technologies due to the limited CMOS processing compatibility of perovskites [143]. Nonetheless, the recent discovery of ferroelectricity in HfO_2 boosted the interest in this device concept due to the successful integration with CMOS technology [136]. The formation of a non-centrosymmetric, orthorhombic phase is responsible for the ferroelectric behavior of this material. Although the orthorhombic phase of

HfO_2 can be obtained by introducing various dopants into the crystal lattice, such as Si, Zr, or Al. The doping through Zr is the most promising dopant for integration into the *BEOL* [144]. Hence, the most promising Hafnia and Zirconium alloy is the $Hf_{0.5}Zr_{0.5}O_2$ compound, often denoted as *HZO*, which presents good characteristics in terms of polarizability and resilience [143].

The Landau–Ginzburg–Devonshire theory is employed in this thesis to model and predict the characteristics of phase transitions and polarization switching properties of ferroelectrics. In particular, to describe the behavior of ferroelectrics a dynamic theory is needed to describe the switching mechanism of the polarization. Landau and Khalatnikov have shown in [145] that in a homogeneous ferroelectric the dynamics of the spontaneous polarization, P, can be described by combining the Lagrange and the Gibb's free-energy equation, as:

$$\rho \frac{dP}{dt} = -\frac{\partial U}{\partial P} \tag{4.1}$$

where U is the system's Gibbs energy and ρ is a resistivity governing the switching kinetics of the domains. In general, the bulk ferroelectric state is determined by temperature, polarization, electric field, stress, and strain, even if, in order to simplify the overall picture, we neglected the role of stress and strain in our simulated devices. Moreover, the resistivity ρ is strictly linked with the switching dynamics of the domains. In this context, the Merz law has been proposed as an empirical relationship between the dynamics of the domains and the electric field of the ferroelectric material [146, 147]. However, for the sake of simplicity, we kept constant the value of ρ through this thesis. This decision has been taken in order to simplify the overall analysis. Moreover, despite this approximation, by fitting the value of ρ with a direct comparison between our simulations and experimental data, a good average dynamic behavior of the numerical data can be observed [148]. In particular, the Landau-Khalatnikov equation describes the free energy of the system as a thermodynamic function defining the polarization state [145, 149]:

$$G(P, E, T) = G0 + \frac{1}{2}\alpha(T)P^2 + \frac{1}{4}\beta P^4 + \frac{1}{6}\gamma P^6 - EP, \tag{4.2}$$

where the expansion is usually truncated at sixth order. In Eq. 4.2 E is the electric field in the ferroelectric, and α, β, and γ known as anisotropy constants. Moreover, the temperature dependence of α is usually expressed as

$$\alpha(T) = \alpha_0 (T - T_c), \tag{4.3}$$

where, for a ferroelectric material, α is negative for a temperature below the Curie temperature Tc, above which a paraelectric behavior is observed. The β constant can be either positive or negative instead. This parameter classifies the ferroelectric material in two different orders. The first order ($\beta < 0$) corresponds to two local minimums of the Gibbs' free energy at non-zero polarization higher than the paraelectric global minimum at P=0. By increasing the temperature, these two local minimums tend to rise in energy following mainly the relationship of α with the temperature reported above. This behavior progressively decreases the energy well between the local minimums and the absolute minimum in P=0 until reaching the temperature at which the system is energetically stable at the paraelectric state (P=0). At that particular temperature, a steep drop of spontaneous polarization can be observed. On the other hand, a second-order ferroelectric ($\beta > 0$) does not present any local minimum in the free-energy function larger than the paraelectric state (P=0).

This means that, by increasing the temperature, just only a smooth transition from the ferroelectric phase (two well-distinguished energy minimums below the energy of the paraelectric state P=0) to the paraelectric phase (single absolute minimum at P=0) is possible at the Curie temperature [150]. In this manuscript, we did not perform a temperature phase transition analysis of our samples, therefore we exploited the three anisotropy constants only as fitting parameters. Moreover, we highlight that, in principle, P is a three-component vector, so the free energy should be written using the vector components. Here, for simplicity, we consider only a scalar value of P, which, for thin ferroelectrics, can be simply taken as the polarization component along the thickness direction.

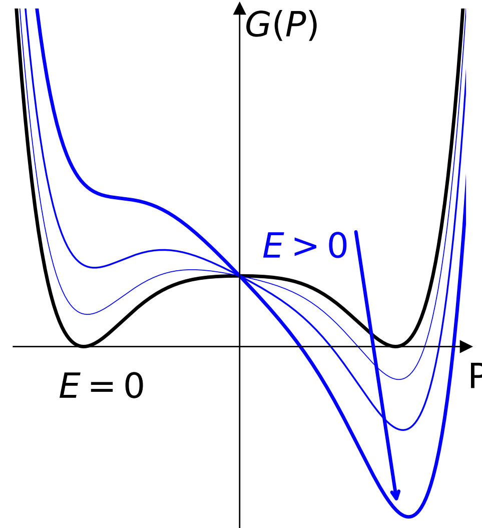

Figure 4.2: Examples of energy configurations of Eq. 4.2 for different values of the electric field

We can observe a two-well configuration where the polarization could stabilize by plotting the free energy configuration of Eq. 4.2. Moreover, as it can be observed in Fig. 4.2, the electric field E can make one minimum progressively more energetically favorable compared to the second one, thus finally forcing the switching between the two stable states.

The steady-state points corresponding to the trajectories described by Eq. 4.2 can be found by minimizing the total energy in Eq. 4.2, which leads to:

$$\frac{\partial G(P,E,T)}{\partial P} = 0 \rightarrow E = \alpha(T)P + \beta P^3 + \gamma P^5 \tag{4.4}$$

By plotting Eq. 4.4 it can be observed the typical "S-shaped" Polarization-Electric Field curve displaying all stable points of the polarization, see Fig.4.1. It is important to notice that this static curve shows a central negative-slope branch intersecting the origin of the plot, which corresponds to a negative differential capacitance $C_{FE} = \partial P/\partial E < 0$. This *Negative Capacitance (NC)* branch is not directly observed in experiments, in fact, the quasi-static experimental characteristic has the well-known hysteretic behavior [141] because the *NC* region is intrinsically unstable. Therefore, as thoroughly mentioned in [63], *NC* can be stabilized with the use of a proper value capacitance in series. In other words, by the use of the negative-positive capacitance series, we can obtain a voltage gain between the capacitances themselves, hence a sort of step-up voltage transformer that relies only on capacitances and can be, in principle, embodied in the gate-stack of a transistor. This voltage gain is the key concept of the so-called NC-FETs, proposed for the first time by

Salahuddin and Datta in [151]. They, in fact, proposed to replace the conventional gate insulator with a ferroelectric insulator with the aim of obtaining a value of *subthreshold-swing* lower than 60 mV/decade, and, consequently, reduce further the leakage current in modern semiconductor technologies. The *NC* operation of ferroelectric capacitors is an interesting and partly elusive topic from a material science perspective [152–155], and it is a promising option for the design of nanoscale transistors [151, 156, 157]. In this latter respect, several studies have experimentally reported *NC* operation in MOS transistors and discussed the potential benefits up to industrial level CMOS circuits [158, 159], and other contributions have addressed different design aspects related to NC transistors [160–169]. Even if these kinds of devices are still a matter of research, we will not cover this topic in this thesis, but we explored the ferroelectric stability concept in [148] to better understand the role of domain coupling and interactions in ferroelectric dynamics. The results obtained in [148] help to understand the depolarization instability principle of ferroelectric domains reported in this manuscript. In fact, with the phenomenological discussion of domain alignment and *NC* we were able to estimate not only the domain wall coupling constant, but also provide a quantitative description of the dynamics ruling ferroelectric devices. This is a fundamental aspect to properly describe the synaptic behavior of the *FTJ* devices described in the following chapters of the manuscript.

4.2 FTJs device concept

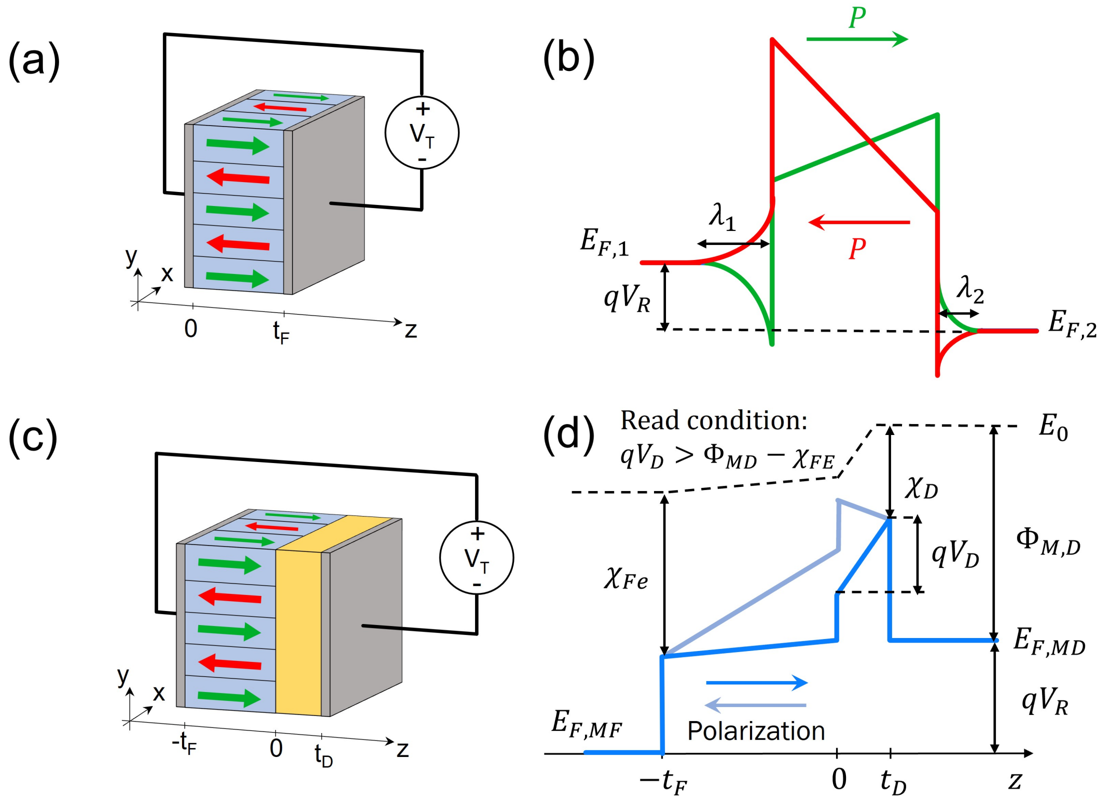

Figure 4.3: (a) Cross-section of an MFM based FTJ. (b) Band diagram for an MFM based FTJ, where λ_1 and λ_2 (with $\lambda_1 > \lambda_2$) are the screening lengths in the metal electrodes. (c) Same as in (b) but for the MFIM structure. (d) Band diagram across a MFIM based FTJ during reading $V_T = V_R$. qV_D should be larger than the ferroelectric tunnelling barrier $[\Phi_{M,D} - \chi_F]$, so that the ferroelectric conduction band profile can drop below $E_{f,MD}$. E_0 and $\Phi_{M,D}$ are respectively the vacuum level and the work function of the MD electrodes. χ_F, χ_D are the electron affinity of the ferroelectric and dielectric, $E_{f,MD}$, $E_{f,MF}$ are the Fermi levels of the MD and MF electrode.

The Ferroelectric Tunnel Junction device consists of a ferroelectric layer sandwiched between two metals (see Fig. 4.3(a)). In general, the finite screening length of the electrodes results in a band bending dependent on the polarization of the ferroelectric itself. In fact, the positive or negative charge changes the effective tunnel barrier height and consequently affects the tunneling resistance. In general, different electrodes should be used to ensure a different effective tunneling barrier is obtained for the two opposite polarizations, see Fig. 4.3(b). However, the main issue with the *Metal-Ferroelectric-Metal (MFM)* stack device is the need for a very thin ferroelectric layer to ensure large enough tunneling current densities.

In this respect, in order to overcome the challenge of ultra–thin ferroelectric layers (1–3 nm) needed in *MFM* stacks, the *FTJs* design has been steered towards two different device structures: *Metal-Ferroelectric-Semiconductor (MFS)* and *MFIM* stacks [29, 138, 170, 171]. *MFS* replace one metal electrode with a semiconducting contact in order to produce a controlled band-bending and ensure a non-symmetric tunneling barrier for different ferroelectric polarity orientations. Recently, good results have been obtained in this direction, exploiting Germanium as semiconducting material. In fact, very thin *FTJs* were reported in [171], showing excellent polarization characteristics and low coercive voltages.

On the other hand, *MFIM* devices (see Fig. 4.3(c)) integrate a dielectric interlayer between the ferroelectric material and one of the metal contacts. These devices are particularly suited for the integration in the *BEOL* of CMOS circuits [170, 172]. Moreover, as it can be inferred from Fig. 4.3(d), in the *MFIM* case, the tunneling current should be mainly limited by the dielectric for a positive ferroelectric polarization (P). On the other hand, when the polarity flips, the internal electric field generated by the uncompensated charge pushes upward the conduction band minimum at the *Ferroelectric-Dielectric (FE-DE)* interface between the two oxides. Hence, the current in the low-conductive state is limited by the thicker potential energy barrier. Hence, in MFIM structures, relatively thick ferroelectric films can be used and still achieve not too small current densities and good values for the *TER* ratio[1] [29]. *FTJ* is a promising candidate as synaptic device with high energy efficiency. A four-level operation has been already experimentally reported in a *MFIM* architecture (Fig. 4.3(a)) [141]. However, the design of a *MFIM FTJ* has a delicate trade–off between the read operation and the retention condition. Indeed, during reading in the low resistance state, the voltage drop V_D across the dielectric layer should be large enough to induce a tunneling limited by the thin oxide, namely $qV_D>[\Phi_{MD}-\chi_F]$. This requires a small dielectric capacitance $C_D=\varepsilon_D/t_D$. The retention condition requires, instead, a large C_D/C_F ratio (with $C_F=\varepsilon_F/t_F$ being the capacitance due to background polarization of the ferroelectric), so as to minimize the depolarization field E_{DEP} and prevent the backswitching of the ferroelectric layer [138]. Furthermore, the dielectric thickness t_D and its electron affinity χ_D have a large impact on the read tunnelling current.

Due to this complex trade–off and the many material and device options, there is an urgent need for a simulation driven optimization of the *FTJs*. However, the modeling of *FTJs* is challenging, as it entails the ferroelectric dynamics for a three dimensional (3D) electrostatics and the tunneling through the dielectric stack. Furthermore, since the non-idealities of the different materials in the stack and/or at the interfaces can play an important role in the operation and in the performance of the *FTJs* [173–176], modelling strategies to include these effects are needed.

[1] The so-called *TER* is measured as the ratio between the conductance of the device in high-conductive state and low-conductive state, respectively

4.3 Modelling of FTJs

In this section, we report a comprehensive modeling of *FTJs* metal-ferroelectric-dielectric-metal (MFIM) structures, used in the results shown in the following chapters of this manuscript. Firstly, for the sake of clarity, we expose the electrodynamic modeling developed in [63], which is the starting point of the analysis in Sec. 5.1. In particular, Sec. 4.3.1 introduces the modeling of the aforementioned structure in the single domain case. However, the mono-domain model could not explain all the features of a complex multi-crystalline and multi-domain model shown by a ferroelectric device (e.g. Hafnia-based ferroelectric tunnel junction multi-level conductances analyzed in [177]). Therefore, Sec. 4.3.2 expands the discussion to multi-domain structures deriving the constitutive equation ruling the dynamics of the ferroelectric polarization in such devices. In particular, in order to simplify the abstraction and the modeling of these kinds of devices, we will consider systems of domains electrostatically coupled to each other but with the same dimension in all spatial directions. Moreover, for the sake of simplicity, within each domain, an anisotropic ferroelectric material condition has been considered. Therefore, the polarization within each domain of the structure is homogeneous. As we will see in the next chapter, these assumptions simplify the whole picture without weakening the link between simulations and experiments. Moreover, we report the expressions describing the uniform fixed charge effects at the *FE-DE* interface. On top of that, we design our charge trapping model to rule the interactions between polarization and trapped charge at the interface between the ferroelectric and dielectric material. Therefore, in Sec. 4.3.5, we link the previous electrodynamic analysis with the presence of charge trapped at the *FE-DE* interface, while in Sec. 4.3.6, we present a novel model that rules the charging/discharging of trapping states at *FE-DE* interface considering electrons elastic tunneling fluxes. At the end of this chapter, we provide expressions to calculate the charge and currents induced in the metals of the capacitors (with or without charge trapping) and, consequently, a comprehensive set of equations to compare simulations with experiments.

4.3.1 Single-Domain MFIM structure

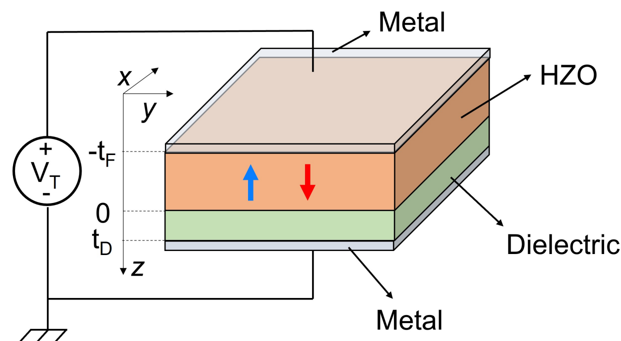

Figure 4.4: Ferroelectric capacitors and related symbols. a) Sketch of a MFIM capacitor, where t_F and t_D are the ferroelectric and dielectric thicknesses. A positive ferroelectric polarization points towards the dielectric. V_T is the external bias, while $V_D(\bar{r})$ and $V_F(\bar{r})$ are the voltage drop respectively across the dielectric and ferroelectric.

Our starting point is the mono-domain of the Landau, Ginzburg, Devonshire (LGD) model for an *MFIM* capacitor presented in [63, 148, 178, 179]. The equation, describing the energy density of

a Ferroelectric material is:

$$U_F = \alpha P^2 + \beta P^4 + \gamma P^6 + k(\nabla P)^2 + \frac{\varepsilon_0 \varepsilon_B}{2} E_F^2 \qquad [J/m^3] \tag{4.5}$$

where α, β and γ are the ferroelectric anisotropy constants, ε_0 and ε_B are the vacuum permittivity and the ferroelectric relative permittivity, k is the domain wall term for the multi-domain structure, P is the spontaneous polarization while E_F is the electric field in the ferroelectric material. The boundary conditions between the ferroelectric and the dielectric material are expressed as:

$$\begin{aligned} \varepsilon_0 \varepsilon_r E_D &= P + \varepsilon_0 \varepsilon_B E_F \\ V_T &= t_F E_F + t_D E_D \end{aligned} \tag{4.6}$$

where t_D and t_F are the dielectric and the ferroelectric thickness, respectively, while V_T is the applied voltage to the MFIM structure. From Eqs. 4.6, we can extract the dependence of the electric fields of the system from P and V_T as:

$$\begin{aligned} E_F &= \frac{1}{t_F C_0} \left(C_D V_T - P \right) \\ E_D &= \frac{1}{t_D C_0} \left(C_F V_T + P \right) \end{aligned} \tag{4.7}$$

In Eqs. 4.7, C_D e C_F are the capacitance per unit area of the dielectric and the paraelectric component of the ferroelectric ($C_D = \varepsilon_0 \varepsilon_r / t_D$, $C_F = \varepsilon_0 \varepsilon_B / t_F$) while $C_0 = C_D + C_F$. Therefore, we can describe the electrostatic term of the free energy remembering that $V_D = E_D t_D$ is the *FE-DE* interface potential (at z=0 in Fig. 4.4).

Moreover, the electrostatic energy lis expressed as:

$$\begin{aligned} U_E &= \frac{1}{2} P V_D + \frac{\varepsilon_0 \varepsilon_F}{2} E_F V_T = \\ &= \frac{1}{2} P t_D \frac{1}{t_D C_0} \left(P + C_F V_T \right) + \frac{V_T C_F}{2 C_0} \left(-P + C_D V_T \right) = \\ &= \frac{P^2}{2 C_0} + \frac{V_T^2 C_s}{2} \qquad [J/m^2] \end{aligned} \tag{4.8}$$

where $C_s = (C_F C_D)/(C_F + C_D)$. The energy term related to an external battery can be expressed as:

$$U_B = -V_T \, P_T = -V_T \left(P + \varepsilon_0 \varepsilon_F E_F \right) = -P V_T \frac{C_D}{C_0} - V_T^2 C_s \qquad [J/m^2] \tag{4.9}$$

where the total polarization is:

$$P_T = P + \varepsilon_0 \varepsilon_F E_F \qquad [C/m^2] \tag{4.10}$$

Therefore, the free energy of the system will be written as the sum between the battery, the electrostatic and the LKE term:

$$\begin{aligned} U_T &= U_E + U_B + U_{LKE} = \\ &= \frac{P^2}{2 C_0} + \frac{V_T^2 C_s}{2} - \left(P V_T \frac{C_D}{C_0} + V_T^2 C_s \right) + t_F \left(\alpha P^2 + \beta P^4 + \gamma P^6 \right) \end{aligned} \tag{4.11}$$

By recalling Eq. 4.1 and differentiating U_T against P, we obtain the equation governing the polarization dynamics:

$$\begin{aligned} t_F\,\rho\frac{dP}{dt} &= -\frac{\partial U_T}{\partial P} = -\left[t_F\left(2\alpha P + 4\beta P^3 + 6\gamma P^5\right) + \frac{P}{C_0} - \frac{C_D}{C_0}V_T\right] \\ &= -\left(2\alpha t_F + \frac{1}{C_0}\right)P - 4t_F\beta P^3 - 6t_F\gamma P^5 + \frac{C_D}{C_0}V_T\ \ [V]. \end{aligned} \tag{4.12}$$

Eq. 4.12 is valid only for a single domain system, namely for a homogeneous ferroelectric material.

4.3.2 Multi-Domain MFIM structure

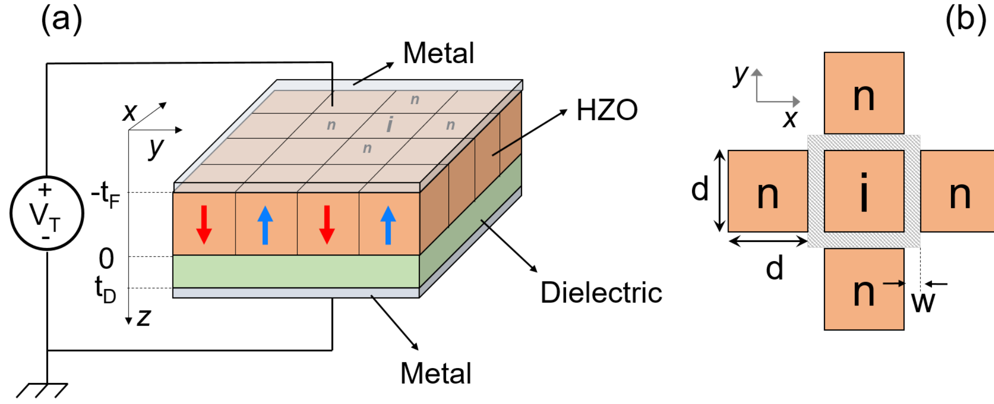

Figure 4.5: Ferroelectric capacitors and related symbols. a) Sketch of a MFIM capacitor, where t_F and t_D are the ferroelectric and dielectric thicknesses. A positive ferroelectric polarization points towards the dielectric (red arrow). (b) Zoom on the ferroelectric domain wall regions, where d is the side of the square domain and w is the width of the domain-wall region, which are used to calculate the domain wall energy [63]. V_T is the external bias, while $V_D(\bar{r})$ and $V_F(\bar{r})$ are the voltage drop respectively across the dielectric and ferroelectric, depending on the position $\bar{r}$=(x, y) in the (x,y) plane.

In a multi-domain framework, we describe the electrostatic energy of the system ($\mathcal{U}_E$) by considering the contribution of all domains as

$$\mathcal{U}_E = \sum_{i=1}^{n_D} \left[\frac{1}{2} \int_{D_i} P_i V_D(\bar{r}) d\bar{r} \right] + \frac{V_T}{2} \int_A \varepsilon_0 \varepsilon_F E_{F,z}(\bar{r}, -t_F) d\bar{r} \;\; [J] \tag{4.13}$$

where $V_D(\bar{r})$ is the FE-DE interface potential at $z = 0$. Here $E_{F,z}(\bar{r}, -t_F)$ is the z-component of the electric field at the interface between the ferroelectric and the metal gate (MF) at $z = -t_F$, while n_D is the total number of domains and $A = n_D A_D$ is the device area. In order to simplify our approach, it has been assumed that the area of any domain in the structure is squared and equal to $A_D = d^2$, where d is the side dimension of any domain in the structure.

In the multi-domain framework, the energy term related to the external battery is described as

$$\mathcal{U}_B = -V_T \left[\sum_{i=1}^{n_D} [P_i A_D] + \int_A \varepsilon_0 \varepsilon_F E_{Fz}(\bar{r}) d\bar{r} \right] \;\; [J]. \tag{4.14}$$

and the total polarization of the ferroelectric is:

$$\begin{aligned} P_T &= \frac{1}{A} \left[\sum_{i=1}^{n_D} [P_i A_D] + \int_A \varepsilon_0 \varepsilon_F E_{Fz}(\bar{r}) d\bar{r} \right] = \\ &= \frac{1}{n_D} \sum_{i=1}^{n_D} P_i + \frac{1}{A} \int_A \varepsilon_0 \varepsilon_F E_{Fz}(\bar{r}) d\bar{r} \quad [C/m^2] \end{aligned} \tag{4.15}$$

Adding together these two terms (Eqs. 4.13 and 4.14) we obtain the total electrostatic energy

of the structure:

$$\mathcal{U}_{ET} = \mathcal{U}_E + \mathcal{U}_B = \sum_{i=1}^{n_D} \left[\frac{P_i}{2} \int_{D_i} V_D(\bar{r}) d\bar{r} - A_D V_T P_i \right] - \frac{V_T}{2} \int_A \varepsilon_0 \varepsilon_F E_{Fz}(\bar{r}) d\bar{r} \ \ [J]. \tag{4.16}$$

In general, V_D and E_{Fz} can be calculated thanks to the superimposition of the effects of P_i in each domain and of V_T. The final dynamic equations ruling the system will be obtained differentiating by each domain polarization P_i. By exploiting the superimposition of effects, we can also express the z-component of the electric field:

$$E_{F,z}(\bar{r}, z) = \sum_{h=1}^{n_D} P_h \bar{G}_{F,h}(\bar{r}, z) + \frac{C_D V_T}{t_F C_0} \hat{i}_z \tag{4.17}$$

it is valid for each $(\bar{r}, z)$ with $-t_F \leq z < 0$. In particular, we would like to calculate the $E_{Fe,z}$ at the interface between the oxide and TOP electrode (i.e. a $z = -t_F$), therefore, we write:

$$E_{Tz} = E_F(\bar{r}, -t_F) = \sum_{h=1}^{n_D} P_h G_{Tz,h}(\bar{r}) + \frac{C_D}{t_F C_0} V_T \quad . \tag{4.18}$$

where $G_{Tz,h}$ are the Green's functions of P_h in combination with E_{Tz} expressed in $[m/F]$. Moreover, we define the electrostatic potential at the *FE-DE* interface ($z = 0$), as:

$$V_D(\bar{r}) = \sum_{h=1}^{n_D} P_h G_{D,h}(\bar{r}) + \frac{C_F}{C_0} V_T \ \ . \tag{4.19}$$

Recalling now Eq. 4.16, we can split the two terms as $\mathcal{U}_{ET,i}^{(1)}$ and $\mathcal{U}_{ET}^{(2)}$:

$$\begin{aligned} \mathcal{U}_{ET,i}^{(1)} &= \frac{P_i}{2} \int_{D_i} \left[\sum_h P_h G_{D,h}(\bar{r}) + \frac{C_F}{C_0} V_T \right] d\bar{r} - d^2 V_T P_i \\ &= \frac{P_i}{2} \sum_h P_h \int_{D_i} G_{D,h}(\bar{r}) d\bar{r} + d^2 \frac{P_i C_F}{2C_0} V_T - d^2 V_T P_i \quad [J]. \end{aligned} \tag{4.20}$$

Normalizing against the area A, we obtain:

$$U_{ET,i}^{(1)} = \frac{P_i}{2} \sum_h \frac{P_h}{C_{i,h}} + P_i V_T \left(\frac{C_F}{2C_0} - 1 \right) = \underbrace{\frac{P_i}{2} \sum_h \frac{P_h}{C_{i,h}}}_{U_{dep,i}} - \frac{V_T}{2} P_i \left(1 + \frac{C_D}{C_0} \right) \qquad [J/m^2] \tag{4.21}$$

where the capacitance $C_{i,h}$ is defined as:

$$\frac{1}{C_{i,h}} = \frac{1}{d^2} \int_{D_i} G_{D,h}(\bar{r}) d\bar{r} \quad [m^2/F]. \tag{4.22}$$

The second term $\mathcal{U}_{ET}^{(2)}$ can be rewritten as:

$$\begin{aligned}\mathcal{U}_{ET}^{(2)} &= -\frac{V_T}{2}\int_A \varepsilon_0\varepsilon_F \left[\sum_h P_h G_{Tz,h}(\bar{r}) + \frac{C_D}{t_F C_0} V_T\right] d\bar{r} \\ &= -\frac{d^2 V_T}{2}\sum_h P_h B_h - \frac{AC_s}{2} V_T^2 \quad [J]\end{aligned} \tag{4.23}$$

where B_h is dimensionless and it is defined as:

$$B_h = \frac{\varepsilon_0\varepsilon_F}{d^2}\int_A G_{Tz,h}(\bar{r}) d\bar{r} \tag{4.24}$$

Normalizing against the domain's area we obtain:

$$U_{ET}^{(2)} = -\frac{V_T}{2}\sum_h P_h B_h - n_D \frac{C_s V_T^2}{2} \quad [J/m^2] \tag{4.25}$$

Consequently, we can rewrite Eq.4.15, as:

$$P_T = \frac{1}{n_D}\sum_{i=1}^{n_D} P_i + \frac{1}{n_D}\sum_{i=1}^{n_D} P_h\, B_h + C_S\, V_T \tag{4.26}$$

Important notes:

1. Because $G_{Tz,h}$ and $G_{D,h}$ are proportional to the domain's area d^2, Eqs.4.22, 4.24, where $C_{i,h}$ and B_h (normalized against the domain's area as well) ensure the model tends to the continuum case where $d \to 0$;

2. Because the $C_{i,h}$ definition, by summing up all capacitance components against every h for a single i, we obtain the potential on the i-domain produced by an uniform distribution of charges, corresponding to the case of Eq.4.7. Due to this, in this case, we can introduce the following *sum rule*:
$$\sum_{h=1}^{n_D} \frac{1}{C_{i,h}} = \frac{1}{C_0} \tag{4.27}$$
The *sum rule* in Eq.4.27 is i-domain indipendent, consequently we can also write:
$$\sum_{i,h=1}^{n_D} \frac{1}{C_{i,h}} = \frac{n_D}{C_0}\ . \tag{4.28}$$

3. Because the B_h definition, by summing-up on avery h we are integrating the ferroelectric electric field at the top-interface on a uniform distribution of charges, this is related to the uni-dimensional electrostatics exposed in the single domain section (Eq.4.7). Due to this we can define a second *sum rule* for the b_h components:
$$\sum_{h=1}^{n_D} b_h = n_D\left(-\frac{C_F}{C_0}\right) \tag{4.29}$$
Moreover, the numerical elaboration of the b_h components by exploiting null electric fields

boundary conditions on the x and y direction, shows that:

$$B_h \simeq -C_F/C_0 \tag{4.30}$$

Therefore, we can write U_{ET} as:

$$\begin{aligned} U_{ET} &= \sum_{j=1}^{n_D} \left[\frac{P_j}{2} \left(\sum_h \frac{P_h}{C_{j,h}} + \frac{C_F}{C_0} V_T \right) - V_T P_j \right] - \frac{V_T}{2} \sum_{j=1}^{n_D} P_j B_j - \frac{n_D C_s V_T^2}{2} \\ &= \frac{1}{2} \sum_{j,h=1}^{n_D} \frac{P_j P_h}{C_{j,h}} - \frac{V_T}{2} \sum_j^{n_D} P_j \left(B_j + 1 + C_D/C_0 \right) - \frac{n_D C_s V_T^2}{2} \end{aligned} \tag{4.31}$$

By differentiate against P_i, we obtain:

$$\frac{\partial U_{ET}}{\partial P_i} = \frac{1}{2} \sum_{j=1}^{n_D} \left[\frac{1}{C_{i,j}} + \frac{1}{C_{j,i}} \right] P_j - \frac{V_T}{2} \left[B_i + 1 + \frac{C_D}{C_0} \right]. \tag{4.32}$$

where $\sum_{j=1}^{n_D} \left[1/C_{i,j} + 1/C_{j,i} \right] P_j$ incorporates the depolarization energy term.

Moreover, by considering Eq.4.30 we have:

$$\left[B_i + 1 + \frac{C_D}{C_0} \right] \simeq \frac{2\, C_D}{C_0} \tag{4.33}$$

Considering now the domain-walls energy term $U_{W,i}$ we have:

$$u_{W,i} = \sum_j k \left(\frac{P_i - P_j}{w} \right)^2 \quad [J/m^3] \tag{4.34}$$

where w is the domain-walls thickness. Now, by integrating the Eq. 4.34 on the $w/2$ side shell surrounding the domain trough all the ferroelectric thickness (t_F), and by also dividing by the considered area $(d+w)^2$, we have:

$$\begin{aligned} U_{W,i} &= \frac{d\, w\, t_F}{2(d+w)^2} \sum_j k \left(\frac{P_i - P_j}{w} \right)^2 \\ &= \frac{t_F}{2} \frac{d}{(d+w)^2} \sum_j \frac{k}{w} \left(P_i - P_j \right)^2 \quad [J/m^2] \end{aligned} \tag{4.35}$$

While the U_{LKE} term is :

$$U_{LKE,i} = \left(\alpha_i P_i^2 + \beta_i P_i^4 + \gamma_i P_i^6 \right) t_F \tag{4.36}$$

We gather all energy components into the total energy

$$U_{TOT} = U_{LKE} + U_W + U_{ET} \tag{4.37}$$

to be differentiated against each domain polarization P_i:

$$\begin{aligned} t_F\rho_i\frac{dP_i}{dt} =& -\left(2\alpha_i P_i + 4\beta_i P_i^3 + 6\gamma_i P_i^5\right) t_F + \\ & -\left[\frac{t_F d}{(d+w)^2}\sum_{j-neighbor}\frac{k}{w}\left(P_i - P_j\right)\right] + \\ & -\left[\frac{1}{2}\sum_{j=1}^{n_D}\left[\frac{1}{C_{i,j}} + \frac{1}{C_{j,i}}\right]P_j - \frac{V_T}{2}\left(B_i + 1 + \frac{C_D}{C_0}\right)\right] \quad [V]. \end{aligned} \tag{4.38}$$

where α_i, β_i and γ_i are the domain-dependent ferroelectric anisotropy constants. When $n_D = 1$ domain-wall term becomes null, therefore Eq.4.38 equals the single domain constitutive Eq. 4.8.

When $t_D \to 0$, $1/C_0$, $1/C_{j,h}$ and B_i become negligible, while $[C_D/C_0] \to 1$, therefore Eq.4.38 tends to the MFM case.

Equation 4.38 provides at each time t and bias $V_T(t)$ all the domain polarizations $P_i(t)$, so that the dielectric, $V_{D,i}$, and ferroelectric, $V_{F,i}$, voltage drops are given by [63]:

$$\begin{aligned} V_{D,i} =& \frac{1}{d^2}\int_{D_i} V_D(\bar{r})d\bar{r} = \sum_{j=1}^{n_D}\frac{1}{C_{i,j}}P_j + \frac{C_F}{C_0}V_T, \\ V_{F,i} =& V_T - V_{D,i} = -\sum_{j=1}^{n_D}\frac{1}{C_{i,j}}P_j + \frac{C_D}{C_0}V_T \end{aligned} \tag{4.39}$$

As it can be seen, the resistivity ρ sets a time scale $t_\rho = \rho/(2|\langle\alpha\rangle|)$ of the ferroelectric dynamics, where $\langle\alpha\rangle$ is the average α across the domains. In this respect, a slow $V_T(t)$ bias compared to t_ρ results in a quasi-static behavior.

4.3.3 Effects of an uniform charge Q_{DF} at the FE-DE interface

Due to the system linearity, the superimposition principle can be used to account also for possible fixed charges, Q_{DF}, at the *FE-DE* interface. In the presence of fixed charges, the overall charge per unit area in each domain can be expressed as

$$\widetilde{P}_i = P_i + Q_{DF} \tag{4.40}$$

An additional charge at the interface modifes Eqs. 4.18 and 4.19 as

$$E_{Tz} = E_F(\bar{r}, -t_F) = \sum_{h=1}^{n_D} \widetilde{P_h}\, G_{Tz,h}(\bar{r}) + \frac{C_D}{t_F C_0} V_T \tag{4.41}$$

$$V_D(\bar{r}) = \sum_{h=1}^{n_D} \widetilde{P_h} G_{D,h}(\bar{r}) + \frac{C_F}{C_0} V_T \quad . \tag{4.42}$$

and the total polarization of the ferroelectric material becomes:

$$P_T = \frac{1}{n_D} \sum_{i=1}^{n_D} P_i + \frac{1}{n_D} \sum_{i=1}^{n_D} \widetilde{P_h}\, B_h + C_S\, V_T \tag{4.43}$$

In the battery energy term, we should consider not only the spontaneous polarization P_i but also the energy contribution of the electric field due to fixed charges.

The battery energy term is consequently defined as:

$$\mathcal{U}_B = -V_T \left[\sum_{i=1}^{n_D} [\, P_i\, A_D] + \int_A \varepsilon_0 \varepsilon_F E_{Fz}(\bar{r}) d\bar{r} \right] \; [J]. \tag{4.44}$$

therefore, normalizing by A_D=d^2 for each domain we have:

$$U_B = -V_T \left[\sum_{i=1}^{n_D} P_i + \sum_{i=1}^{n_D} \widetilde{P_h}\, B_h + n_D\, C_S\, V_T \right] \; [J/m^2]. \tag{4.45}$$

All electrostatic energy terms are practically equal to the expressions presented in the previous section (by substituting P_i with $\widetilde{P}_i$), except for the first term between square brackets of Eq. 4.45, where the equation includes just the spontaneous polarization P_i. Therefore Eq. 4.31 becomes:

$$\begin{aligned} U_{ET} &= \sum_{j=1}^{n_D} \left[\frac{\widetilde{P_j}}{2} \left(\sum_h \frac{\widetilde{P_h}}{C_{j,h}} + \frac{C_F}{C_0} V_T \right) - V_T P_j \right] - \frac{V_T}{2} \sum_{j=1}^{n_D} \widetilde{P_j} B_j - \frac{n_D C_s V_T^2}{2} \\ &= \sum_{j,h} \frac{\widetilde{P_j}\widetilde{P_h}}{2C_{j,h}} + \frac{C_F}{2C_0} V_T \sum_j \widetilde{P_j} - V_T \sum_j P_j - \frac{V_T}{2} \sum_j \widetilde{P_j} B_j - \frac{n_D C_s V_T^2}{2} \end{aligned} \tag{4.46}$$

We can differentiate the electrostatic total energy P_i:

$$\begin{aligned}\frac{\partial U_{ET}}{\partial P_i} =& \frac{1}{2}\sum_j P_j\left[\frac{1}{C_{i,j}}+\frac{1}{C_{j,i}}\right]-V_T-\frac{V_T}{2}B_i+\\ &+\frac{C_F}{2C_0}V_T+\frac{Q_{DF}}{2}\sum_j\left[\frac{1}{C_{i,j}}+\frac{1}{C_{j,i}}\right]\\ =&\frac{1}{2}\sum_j P_j\left[\frac{1}{C_{i,j}}+\frac{1}{C_{j,i}}\right]-\frac{V_T}{2}\left[B_i+1+\frac{C_D}{C_0}\right]+\frac{Q_{DF}}{2}\sum_j\left[\frac{1}{C_{i,j}}+\frac{1}{C_{j,i}}\right]\\ =&\frac{1}{2}\sum_j P_j\left[\frac{1}{C_{i,j}}+\frac{1}{C_{j,i}}\right]-\frac{V_T}{2}\left[B_i+1+\frac{C_D}{C_0}\right]+\frac{Q_{DF}}{C_0}\end{aligned} \tag{4.47}$$

and obtain the constitutive equation (replacing Eq. 4.38):

$$\begin{aligned}t_F\rho_i\frac{dP_i}{dt} =& -\left(2\alpha_i P_i+4\beta_i P_i^3+6\gamma_i P_i^5\right)t_F+\\ &-\left[\frac{t_F d}{(d+w)^2}\sum_{j-vicini}\frac{k}{w}\left(P_i-P_j\right)\right]+\\ &-\left[\frac{1}{2}\sum_j P_j\left[\frac{1}{C_{i,j}}+\frac{1}{C_{j,i}}\right]-\frac{V_T}{2}\left[B_i+1+\frac{C_D}{C_0}\right]+\frac{Q_{DF}}{C_0}\right]\quad[V].\end{aligned} \tag{4.48}$$

4.3.4 Read operation and tunnelling current

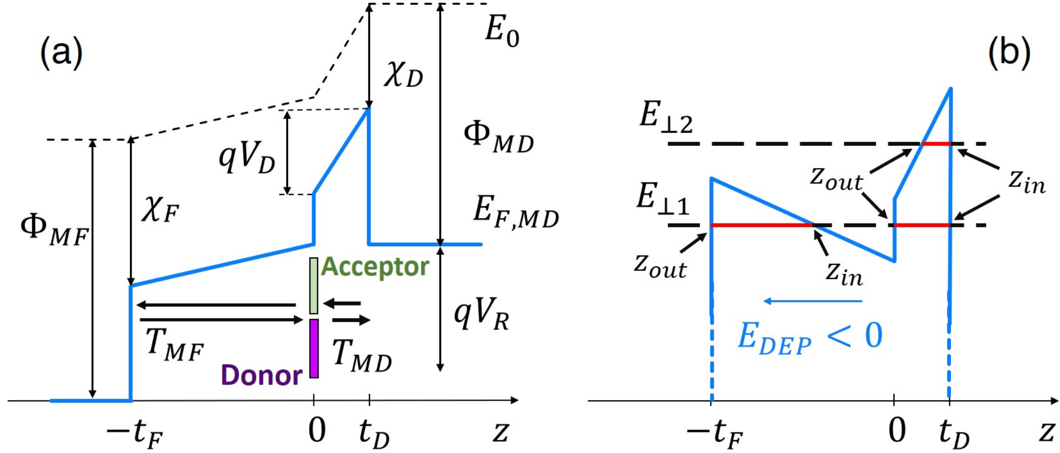

Figure 4.6: Energy band diagram across the MFIM stack. E_0, Φ_{MF} and Φ_{MD} are respectively the vacuum level, and the work function of the MF and MD electrodes. χ_F, χ_D are the electron affinity of the ferroelectric and dielectric, $E_{f,MD}$, $E_{f,MF}$ are the Fermi levels in MD and MF. (a) Read condition with a voltage V_T=V_R: qV_D should be larger than the ferroelectric tunnelling barrier $[\Phi_{MD}-\chi_F]$, so that the ferroelectric conduction band profile can drop below $E_{f,MD}$; (b) Retention condition for $V_T = 0$ V: the depolarization field $E_{DEP} \approx P_r \left[\varepsilon_F \left(C_D/C_F + 1\right)\right]^{-1}$ should be minimized to prevent backswitching (P_r is the remnant polarization).

We assume in the model that the current in the read operation, I_R, is dominated by the tunnelling through the dielectric stack, which is estimated as the sum of the $\mathrm{I}_{R,i}$ in each domain. The $\mathrm{I}_{R,i}$ is in turn expressed by using a Landauer model as [180]:

$$\mathrm{I}_{R,i} = \frac{q}{\pi\hbar} \int_{-\infty}^{\infty} \sum_{k_x,k_y} T_i(E_\perp) \left[f_{0,MD}(E_\perp + \varepsilon(\mathbf{k})) - f_{0,MF}(E_\perp + \varepsilon(\mathbf{k}))\right] dE_\perp \tag{4.49}$$

while the transverse energy is

$$\varepsilon(\mathbf{k}) = \frac{\hbar^2}{2m_\parallel} \frac{1}{{k_x}^2 + {k_y}^2}. \tag{4.50}$$

The Fermi functions in Eq. 4.49 are defined as

$$f_{0,MD(F)}(E) = \frac{1}{1 + \exp\left(E - E_{f,MD(F)}\right)} \tag{4.51}$$

with $E_{f,MD}$, $E_{f,MF}$ being the Fermi levels of the electrodes. Equation 4.49 assumes an effective mass approximation and an energy separability E=$E_\perp$+$\varepsilon(\mathbf{k})$, with the transverse energy $\varepsilon(\mathbf{k})$ being conserved in the tunnelling process [181]. For a tunnelling transmission, $T_i(E_\perp)$, independent of (k_x, k_y), the sum over (k_x, k_y) can be evaluated analytically for each electrode M = MD or MF as [182]

$$\begin{aligned} \sum_{k_x k_y} f_{0,M}(E_\perp + \varepsilon(\mathbf{k})) &= \frac{A}{(2\pi)^2} \int_{\mathbf{k}} f_{0,M}(E_\perp + \varepsilon(\mathbf{k})) d\mathbf{k} \\ &= \frac{A K_B T m_\parallel}{2\pi\hbar^2} \ln\left[1 + \exp\left(\frac{E_{f,M} - E_\perp}{K_B T}\right)\right] \end{aligned} \tag{4.52}$$

Hence the expression for $I_{R,i}$ simplifies as

$$I_{R,i} = \frac{AK_BTm_{\|}q}{2\pi^2\hbar^3}\int_{-\infty}^{+\infty} T_i(E_\perp)\left[ln\left[1+\exp\left(\frac{E_{f,MD}-E_\perp}{K_BT}\right)\right]-ln\left[1+\exp\left(\frac{E_{f,MF}-E_\perp}{K_BT}\right)\right]\right]dE_\perp \tag{4.53}$$

For a read voltage V_R applied to the MF electrode, we have $E_{f,MF}$=$E_{f,MD}$−qV_R. Finally, $T_i(E_\perp)$ is calculated accordingly with a WKB approximation and an effective oxide mass m_{ox}. Here we notice that the $m_{\|}$ in Eq. 4.53 corresponds to an effective mass for the density of states of the metal electrodes. In the lack of a better determination of $m_{\|}$, we used the popular assumption $m_{\|} \approx m_0$ [183].

For the purpose of WKB calculations the conduction band profile $E_{CD,i}(z)$, $E_{CF,i}(z)$ in the oxide and ferroelectric is assumed to be linear and set by the $V_{D,i}$ and $V_{F,i}$ given by Eqs.4.39; this simplifies the determination of the extrema z_{in}, z_{out} of the tunnelling paths illustrated in Fig. 4.6(b). Depending on the specific polarization condition, two tunnelling paths may be involved in the WKB calculation at a given $E_\perp$ (see Fig. 4.6(b)), in which case $T_i(E_\perp)$ is obtained as the product of the two tunnelling transmission probabilities. This approach neglects the influence on $T_i(E_\perp)$ of interference effects, which is a reasonable approximation also in virtue of the empirical calibration of some modelling parameters discussed below.

4.3.5 Trapped charges in the MFIM structure

It has been recently pointed out that in ferroelectric–dielectric systems the charge injection through the thin dielectric and the charge trapping can play an important role [173–176]. Hereafter we will assume that tunnelling is the dominant conduction mechanism across the Al_2O_3 layer, even if additional mechanisms assisted by defects are also possible in thin oxides [184]. The following paragraphs provide simplified expressions for the charge induced at the metal contact and relative currents in *MFIM* structure with or without charge trapping at the *FE-DE* interface. The following expressions will be integrated in a more precise manner in Sec. 4.3.6.

MFIM without any charge trapped at the FE-DE interface

The charge per unit area on the metal contacts (in the MFIM structure without charge trapping) should be calculated with the following expressions:

$$Q^{M,F} = \frac{1}{A}\left[d^2 \sum_{h=1}^{n_D} P_h\left(B_h + 1\right) + V_T C_S \cdot n_D d^2\right] \tag{4.54}$$

$$Q^{M,D} = \frac{1}{A}\left[d^2 \sum_{h=1}^{n_D} P_h D_h - V_T C_S \cdot n_D d^2\right] \tag{4.55}$$

where the whole device area is denoted $A = n_D d^2$, while B_h and D_h provide the ferroelectric and dielectric metal induced charge from a *FE-DE* interfacial charge.

It should be noted that in $Q^{M,F}$ the polarization P_h is multiplied by (B_h+1) to take into account both polarization at the interface and the opposite polarization close to the *MF* contact $(-P_h)$. As already mentioned B_h and D_h can be expressed analytically by using the following equation:

$$B_h \simeq -C_F/C_0 \qquad D_h \simeq -C_D/C_0 \qquad B_h + D_h = -1 \tag{4.56}$$

Consequently, by introducing the average polarization $P_{AV}=(\sum_{i=1}^{n_D} P_i)/n_D$, we can re-write Eqs. 4.54 and 4.55 as

$$Q^{M,F} = \left(1 - \frac{C_F}{C_0}\right) P_{AV} + V_T C_S = \frac{C_D}{C_0} P_{AV} + V_T C_S \tag{4.57}$$

$$Q^{M,D} = \frac{-C_D}{C_0} P_{AV} - V_T C_S. \tag{4.58}$$

therefore, we have that $Q^{M,F}(t) + Q^{M,D}(t) = 0$.

The corresponding displacement currents are:

$$\begin{aligned} i^{M,F}(t) &= \frac{\partial Q^{M,F}(t)}{\partial t} = C_S \frac{\partial V_T}{\partial t} + \frac{C_D}{C_0}\frac{\partial P_{AV}}{\partial t} \\ i^{M,D}(t) &= \frac{\partial Q^{M,D}(t)}{\partial t} = -C_S \frac{\partial V_T}{\partial t} - \frac{C_D}{C_0}\frac{\partial P_{AV}}{\partial t} = -i^{M,F}(t) \end{aligned} \tag{4.59}$$

MFIM with trapped charge at the FE-DE interface

The charge per unit area on the metal contacts should be calculated with the following expressions:

$$Q^{M,F} = \frac{1}{A}\left[d^2 \sum_{h=1}^{n_D} B_h\, Q_{int,h} + d^2 \sum_{h=1}^{n_D} P_h(B_h+1) + V_T C_S \cdot n_D d^2\right] \tag{4.60}$$

$$Q^{M,D} = \frac{1}{A}\left[d^2 \sum_{h=1}^{n_D} (P_h + Q_{int}) D_h - V_T C_S \cdot n_D d^2\right] \tag{4.61}$$

where $Q_{int,h}$ is the interface charge, and it could be expressed as $Q_{int,h}$=$(Q_{IF} + Q_{T,h})$, where Q_{IF} and $Q_{T,h}$ are fixed charge and the charge trapped in the hth domain. Therefore, by exploiting Eq. 4.56 we have:

$$Q^{M,F} = -\frac{C_F}{C_0} Q_{int,AV} + \frac{C_D}{C_0} P_{AV} + V_T C_S \tag{4.62}$$

$$Q^{M,D} = \frac{-C_D}{C_0} Q_{int,AV} - \frac{C_D}{C_0} P_{AV} - V_T C_S. \tag{4.63}$$

Recalling now the total charge in the ferroelectric:

$$Q = P_{AV} + \varepsilon_0 \varepsilon_F E_{fe,AV} \tag{4.64}$$

we can observe that, by substituting $E_{fe,AV}$ with Eq. 4.7 and by denoting with $Q_{int,AV}$ the total trapped charge at the *FE-DE* interface, we obtain an updated expression of $Q^{M,F}$:

$$\begin{aligned} Q_{AV} = P_{AV} + \frac{\varepsilon_0 \varepsilon_F}{t_F} \frac{1}{C_0} (C_D V_T - P_{AV} - Q_{int,AV}) = \\ -\frac{C_F}{C_0} Q_{int,AV} + \frac{C_D}{C_0} P_{AV} + C_S V_T = Q^{M,F} \end{aligned} \tag{4.65}$$

Now, the current at the metal contact is composed by two different components: the first is the displacement current due to the time derivative of the charges $Q^{M,F}(t)$ and $Q^{M,D}(t)$, while the second component is a current flow via tunneling between the metal electrodes and the traps at the *FE-DE* interface.

In general, f_D denotes the fraction of the charge exchanged with the interface between the dielectric and the metal gate (MD) and the *FE-DE* interface. Consequently, the ratio of the charge exchange between the *MF* and the *FE-DE* interface is $(1 - f_D)$. Thereby, we can describe the currents as:

$$\begin{aligned} i^{M,F}(t) &= \frac{\partial Q^{M,F}(t)}{\partial t} + (1 - f_D)\, \frac{\partial Q_{int,AV}(t)}{\partial t} \\ i^{M,D}(t) &= \frac{\partial Q^{M,D}(t)}{\partial t} + f_D\, \frac{\partial Q_{int,AV}(t)}{\partial t} \end{aligned} \tag{4.66}$$

which can be rewritten as:

$$\begin{aligned} i^{M,F}(t) &= -\frac{C_F}{C_0} \frac{\partial Q_{int,AV}}{\partial t} + C_S \frac{\partial V_T}{\partial t} + \frac{C_D}{C_0} \frac{\partial P}{\partial t} + (1 - f_D)\, \frac{\partial Q_{int,AV}(t)}{\partial t} \\ i^{M,D}(t) &= -\frac{C_D}{C_0} \frac{\partial Q_{int,AV}}{\partial t} - C_S \frac{\partial V_T}{\partial t} - \frac{C_D}{C_0} \frac{\partial P}{\partial t} + f_D\, \frac{\partial Q_{int,AV}(t)}{\partial t} \end{aligned} \tag{4.67}$$

Equation 4.67 ensures $i^{M,F}(t) + i^{M,D}(t) = 0$. If the dielectric is much thinner than the ferroelectric, it may be a reasonable approximation to consider that the tunneling current is almost completely exchanged with the *MD* electrode, namely, we can use the approximation $f_D \approx 1$

4.3.6 Tunneling assisted by traps at the FE-DE interface

In this section, we introduce a novel model for charge injection and trapping in *MFIM* structures. We describe the trap density and trapped charge in terms of areal densities at the *FE-DE* interface. However, it is acknowledged that these figures should be considered as equivalent areal densities possibly including also a charge trapping in the ferroelectric and dielectric films.

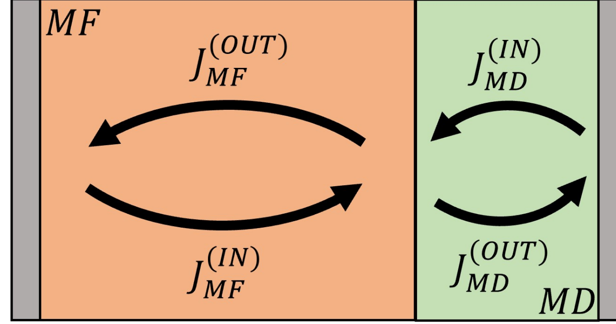

Figure 4.7: Sketch of charge injection fluxes through the MFIM structure

A first-order dynamic equation for a trap level at the *FE-DE* interface having an energy E_T and an occupation f_T can be written as

$$q\frac{\partial f_T(E_T)}{\partial t} = I_{MD}^{(IN)}(E_T) + I_{MF}^{(IN)}(E_T) - I_{MD}^{(OUT)}(E_T) - I_{MF}^{(OUT)}(E_T) \tag{4.68}$$

where f_T is the traps occupation function. Moreover, the current at the *MD* electrode injected into the trap at E_T can be written as:

$$I_{MD}^{(IN)}(E_T) = \sigma_T(1 - f_T) \int_{E_T-\frac{\sigma_E}{2}}^{E_T+\frac{\sigma_E}{2}} J_{MD}\,(E)dE \tag{4.69}$$

which can be approximated as:

$$I_{MD}^{(IN)} = \sigma_T\sigma_E J_{MD}(1 - f_T) \tag{4.70}$$

where:

- σ_T $[m^2]$ is the trap cross-section in which the traps collect the electron flux;
- σ_E $[J]$ o $[eV]$ is the energy cross-section;
- J_{MD} $\left[\frac{A}{m^2\,J}\right]$ is the impinging current density (per unit energy and unit area) at the *FE-DE* interface.

Then, the trap's occupation must be in equilibrium with the Fermi level of the metal in a steady-state condition ($\partial f_T/\partial t \rightarrow 0$). This is done by applying the detailed balance principle to the current fluxes of Eq. 4.68 for each metal. In this condition, $I_{MD}^{(IN)} = I_{MD}^{(OUT)}$ and consequently we can define the output current towards the *MD* contact as

$$I_{MD}^{(OUT)} = \sigma_T\sigma_E J_{MD}(E_T)\,[1 - f_{0,MD}(E_T)] \tag{4.71}$$

In the same manner, we define the input current at the FE contact as:

$$I_{MF}^{(IN)}(E_T) = \sigma_T \sigma_E J_{MF}(E_T)(1 - f_T) \tag{4.72}$$

And consistently with the *MD* conctact, we define the output current of *MF* as

$$I_{MF}^{(OUT)}(E_T) = \sigma_T \sigma_E J_{MF}(E_T)\left[1 - f_{0,MF}(E_T)\right] \tag{4.73}$$

To calculate $J_{MD}(E_T)$ we exploted the Landauer model:

$$I_{MD}(E_T) \approx \frac{q}{\pi\hbar} \sum_{\mathbf{k}} T_{\mathbf{k}}^{(MD)}(E_\perp) f_{0,MD}(E_T) \qquad \left[\frac{A}{J}\right] \tag{4.74}$$

where $\mathbf{k} = (k_x, k_y)$ with z the elastic tunnelling direction. Moreover, the transmission coefficient $T_{\mathbf{k}}^{(MD)}(E_\perp) \approx T_{MD}(E_\perp)$ depends on $E_\perp = E_T - \varepsilon_{\mathbf{k}}$ with $\varepsilon_{\mathbf{k}} = \frac{\hbar^2 k^2}{2m_\parallel}$ thanks the effective mass approximation [183]. Therefore, we can write:

$$I_{MD}(E_T) = \frac{q}{\pi\hbar^2} f_{0,MD}(E_T) \sum_{\mathbf{k}} T_{MD}(E_T - \varepsilon_{\mathbf{k}}) \tag{4.75}$$

the sum over $\mathbf{k}$ can be converted into an integral proportional to the area:

$$\begin{aligned} \sum_{\mathbf{k}} T(E_T - \varepsilon_{\mathbf{k}}) &= \frac{A}{(2\pi)^2} \int_0^{2\pi} d\theta \int_0^{+\infty} T_{MD}(E_T - \varepsilon_{\mathbf{k}}) k dk \\ &= \frac{A m_\parallel}{2\pi\hbar^2} \int_0^{+\infty} T_{MD}(E_T - \varepsilon) d\varepsilon \end{aligned} \tag{4.76}$$

We can write the final density of current from the *MD* contact as:

$$J_{MD}(E_T) = \frac{I_{MD}}{A} = f_{0,MD}(E_T) \frac{q m_\parallel}{2\pi^2\hbar^3} \int_0^{+\infty} T_{MD}(E_T - \varepsilon) d\varepsilon \tag{4.77}$$

Thanks to the WKB approximation, the transmission coefficient is expressed as an exponential function which has the imaginary exponent proportional to the wave vector k_T evaluated through the tunneling direction:

$$T_{MD}(z, E) \propto exp\left(i k_T(z, E)\right) = exp\left(i \frac{\sqrt{2 m_{eff}^D (E - E_c(z))}}{\hbar}\right) \tag{4.78}$$

where m_{eff}^D is the effective mass of the dielectric, $E_c(z)$ is the conduction band minimum at point z. In our case $E_c(z) > E$, therefore the transmission coefficient is real. It can be demonstrated that T_{MD} can be expressed in the form:

$$T_{MD}(E_\perp) = \exp\left\{-\frac{4}{3\hbar q} \frac{\sqrt{2 m_{ox}}}{E_{ox}} \left[(\beta + q E_{ox} t_{ox})^{\frac{3}{2}} - \beta^{\frac{3}{2}}\right]\right\} \tag{4.79}$$

with

$$\beta(E_{\perp}) = \Phi_m - \chi_{ox} - qV_{bias} + qE_F t_F - (E_{\perp}) \tag{4.80}$$

For the *MF* contact we have the same behavior:

$$J_{MF}(E_T) = f_{0,MF}(E_T)\frac{qm_{\parallel}}{2\pi^2\hbar^3}\int_0^{+\infty} T_{MF}(E_T-\varepsilon)d\varepsilon \tag{4.81}$$

and the transmission coefficient is:

$$T_{MF}(z,E) \propto exp\left(ik_T(z,E)\right) = exp\left(i\frac{\sqrt{2m_{eff}^F(E-E_c(z))}}{\hbar}\right) \tag{4.82}$$

where m_{eff}^F is the effective mass of the ferroelectric.
Where T_{MF} can be expressed as:

$$T_{MF}(E_{\perp}) = \exp\left\{-\frac{4}{3\hbar q}\frac{\sqrt{2m_F}}{E_F}\left[(\alpha+qE_F t_F)^{\frac{3}{2}} - \alpha^{\frac{3}{2}}\right]\right\} \tag{4.83}$$

with

$$\alpha(E_{\perp}) = \Phi_m - \chi_F - qV_{bias} - E_{\perp} \tag{4.84}$$

In particular, J_{MD} and J_{MF} depend on T_{MD} and T_{MF} with the following relation:

- $J_{MD} \propto f_{0,MD}(E_T)\int_0^{+\infty} T_{MD}(E_T-\varepsilon)d\varepsilon$
- $J_{MF} \propto f_{0,MF}(E_T)\int_0^{+\infty} T_{MF}(E_T-\varepsilon)d\varepsilon$

When E_T goes below the Fermi-level of the *MD* contact $E_{f,MD}$, the electron emission becomes negligible compared to the capture rate, therefore $f_T \to 1$ in a steady-state condition. Conversely, when E_T exceeds $E_{f,MD}$ the capture rate becomes negligible and $f_T \to 0$.
The trap-assisted tunneling currents can be calculated as:

$$I_{MF}^{(TAT)} = \sum_{\text{traps}}\left(I_{MF}^{(IN)} - I_{MF}^{(OUT)}\right) \tag{4.85}$$

$$I_{MD}^{(TAT)} = \sum_{\text{traps}}\left(I_{MD}^{(IN)} - I_{MD}^{(OUT)}\right) \tag{4.86}$$

The charge in the traps should be calculated with:

$$Q_T(E_T) = \underbrace{f_T N_T^{acc}}_{=n_{tr}}\underbrace{(-q)}_{\text{acc. traps}} + \underbrace{(1-f_T)N_T^{don}}_{=p_{tr}}\underbrace{(+q)}_{\text{donor traps}} \tag{4.87}$$

where N_T^{acc} and N_T^{don} $\left[\frac{1}{cm^2}\right]$ are the density per unit area of the acceptors and donor traps, respectively. The total trapped charge at the *FE-DE* interface can be calculated also with the formula:

$$Q_T(E_T) = \underbrace{f_T N_{acc}\Delta E}_{=n_{tr}}\underbrace{(-q)}_{\text{acc. traps}} + \underbrace{(1-f_T)N_{don}\Delta E}_{=p_{tr}}\underbrace{(+q)}_{\text{donor traps}} \tag{4.88}$$

where N_{acc} and N_{don} $\left[\frac{1}{Jcm^2}\right]$ are the density per unit area and unit energy for acceptors and donors

traps, respectively. Comparing Eq. 4.87 with Eq. 4.88 we can set:

$$\begin{cases} N_{acc}\Delta E = N_T^{acc} \\ N_{don}\Delta E = N_T^{don} \end{cases} \tag{4.89}$$

where ΔE is the finite energy discretization step to evaluate numerically the energy integral of Eqs. 4.77 and 4.81. Moreover, the average charge trapped in the acceptor (Q_{acc}) is:

$$Q_{acc} = \frac{(-q)}{n_D} \sum_{n_D, E_T} N_{acc} f_T(E_T) \Delta E = \frac{(-q)}{n_D} \sum_{n_D, E_T} N_T^{acc} f_T(E_T) \tag{4.90}$$

while a donor trap charge can be expressed as

$$\begin{aligned} Q_{don} &= (+q) \sum_{E_T} N_{don}(1 - f_T(E_T)) \Delta E = \\ &= (+q) \sum_{E_T} N_T^{don}(1 - f_T(E_T)) \end{aligned} \tag{4.91}$$

Now, in order to gather all the results obtained until now and link them with the *detailed balance principle*, we rewrite the dynamic traps equation (Eq. 4.104) as a function of capture and emission rates of the carriers at the MD and MF contact, respectively:

$$\frac{\partial f_T}{\partial t} = c_{n,MD} \left(1 - f_T\right) - e_{n,MD} f_T + c_{n,MF} \left(1 - f_T\right) - e_{n,MF} f_T \tag{4.92}$$

where $c_{n,M}$ and $e_{n,M}$ (M = MD or MF) denote capture and emission rate of each trap. When f_T is equal to the Fermi function at the electrode *MD* or MF, a detailed balance must be satisfied between the capture and emission rate towards the electrodes. Consequently, we can write the detailed balanced principle in steady-state ($\partial f_T / \partial t = 0$) at the *MD* contact:

$$c_{n,MD} \left(1 - f_T\right) - e_{n,MD} f_T = 0 \qquad f_T = f_{0,MD} \tag{4.93}$$

$$e_{n,MD} = c_{n,MD} \frac{1 - f_{0,MD}}{f_{0,MD}} = c_{n,MD} \text{exp} \left(\frac{E_T - E_{f,MD}}{K_B T} \right) \tag{4.94}$$

and at MF contact:

$$c_{n,MF} \left(1 - f_T\right) - e_{n,MF} f_T = 0 \qquad f_T = f_{0,MF} \tag{4.95}$$

$$e_{n,MF} = c_{n,MF} \frac{1 - f_{0,MF}}{f_{0,MF}} = c_{n,MF} \text{exp} \left(\frac{E_T - E_{f,MF}}{K_B T} \right) \tag{4.96}$$

Now, considering the definitions of the carrier's currents in Eqs. 4.70, 4.71, 4.72 and 4.73 together with Eqs. 4.77 and 4.81, we can link the capture and emission rates in terms of the tunnelling

transmission coefficient between the FE-DE interface and each terminal *MD* or *MF* according to:

$$c_{n,MD0} = e_{n,MD0} = \sigma_T \sigma_E \frac{m_\parallel}{2\pi^2\hbar^3} \int_0^{+\infty} T_{MD}(E_T - \varepsilon) d\varepsilon \tag{4.97}$$

Then, we can rewrite the emission and capture rates for the *MD* contact as:

$$c_{n,MD} = c_{n,MD0}\, f_{0,MD}(E_T) \tag{4.98}$$

$$e_{n,MD} = c_{n,MD}\,(1 - f_{0,MD}(E_T)) = c_{n0}\, \frac{\exp((E_T - E_{f,MD})/KT)}{1 + \exp((E_T - E_{f,MD})/KT)} \tag{4.99}$$

At the *MF* electrode, we can similarly :

$$c_{n,MF0} = e_{n,MF0} = \sigma_T \sigma_E \frac{m_\parallel}{2\pi^2\hbar^3} \int_0^{+\infty} T_{MF}(E_T - \varepsilon) d\varepsilon \tag{4.100}$$

and re-write the emission and capture rates for the *MF* contact as:

$$c_{n,MF} = c_{n,MF0}\, f_{0,MF}(E_T) \tag{4.101}$$

$$e_{n,MF} = c_{n,MF0}\,(1 - f_{0,MF}(E_T)) = c_{n0}\, \frac{\exp((E_T - E_{f,MF})/KT)}{1 + \exp((E_T - E_{f,MF})/KT)} \tag{4.102}$$

With Eqs. 4.98, 4.99, 4.101 and 4.102 we can further simplify the r.h.s. of Eq. 4.68, in fact, at both electrodes we have:

$$\begin{aligned} c_n\,(1 - f_T) - e_n\, f_T &= \\ &= e_{n0}\, f_{0,M}(E_T)\,(1 - f_T) - e_{n0}\, f_T\,(1 - f_{0,M}(E_T)) = \\ &= e_{n0}\,(f_{0,M}(E_T) - f_T) \end{aligned} \tag{4.103}$$

and consequently Eq. 4.104 can be re-written as:

$$\frac{\partial f_T}{\partial t} = e_{n,MD0}\,[f_{0,MD}(E_T) - f_T] + e_{n,MF0}\,[f_{0,MF}(E_T) - f_T] \tag{4.104}$$

with $e_{n,MD0}$ and $e_{n,MF0}$ are given by Eqs. 4.97 and 4.100. Equations 4.104 refers to a single domain in the *FTJ* structure, but it is understood that it was solved in each domain and self-consistently with the LGD equations for the ferroelectric dynamics.

Moreover, in steady-state ($\partial f_T/\partial t \to 0$), the traps occupation can be calculated taking into

account Eqs. 4.98, 4.99, 4.101, 4.102 and 4.104 as:

$$\begin{aligned} f_T &= \frac{c_{n,MD} + c_{n,MF}}{c_{n,MD} + c_{n,MF} + e_{n,MD} + e_{n,MF}} \\ &= \frac{c_{n,MD} + c_{n,MF}}{c_{n,MD}\left[1 + \exp\left(\frac{E_T - E_{f,MD}}{K_B T}\right)\right] + c_{n,MF}\left[1 + \exp\left(\frac{E_T - E_{f,MF}}{K_B T}\right)\right]} \\ &= \frac{e_{n,MD0}\, f_{0,MD}(E_T) + e_{n,MF0}\, f_{0,MF}(E_T)}{e_{n,MD0} + e_{n,MF0}} \end{aligned} \tag{4.105}$$

Equation 4.105 shows that at the equilibrium, namely in steady state for $V_T = 0$ V and thus for $E_{f,MD} = E_{f,MF}$, the f_T of acceptor type traps is duly set by the equilibrium Fermi function. When a non zero V_T is applied to the structure, instead, the steady state f_T approaches the equilibrium condition with one of the two electrodes only if the capture rate from that one electrode is much larger than the capture rate from the second electrode.

In particular, if a trap is located inside the Fermi window ($E_{f,MF} < E_T < E_{f,MD}$), we can approximate the Fermi function at the *MD* as $f_{0,MD}(E_T) \approx 1$, while at *MF* with $f_{0,MF}(E_T) \approx \exp(-(E_T - E_{f,MF})/KT)$. Thereby, Eq. 4.105 becomes:

$$f_T \simeq \frac{e_{n,MD0} + e_{n,MF0} \exp(-(E_T - E_{f,MF})/KT)}{e_{n,MD0} + e_{n,MF0}} \tag{4.106}$$

Eq. 4.106 describes the dependence of the traps occupation f_T as a function of the trap energy ET, and of $e_{n,MD0}$ and $e_{n,MF0}$. As it is well known, the trap-assisted tunnelling is mainly dependent on the traps located inside at the Fermi window.

Figure 4.8 shows an example of the steady state occupation of acceptor type traps with an energy located 0.4 eV below the HZO conduction band edge and after a SET pulse (see the V_T waveform in the inset). In the example at study, the capture rate c_{MD0} is much larger than c_{MF0}, because of the very different thickness of the dielectric and ferroelectric layers, so that the steady state occupation of the trap is essentially in equilibrium with the Fermi level $E_{f,MD}$ of the *MD* electrode.

Now, in order to calculate the total current flowing through the metal contacts (generalized by Eq. 4.67), we need to link the time derivative of the acceptor and donor traps with the tunneling fluxes reported below. Therefore, for the acceptor traps we have:

$$\begin{aligned} \frac{\partial Q_{acc}}{\partial t} = \frac{1}{n_D} \sum_{n_D, E_T} \Big[& J_{MD}^{(IN)}(E_T) + J_{MD}^{(OUT)}(E_T) + \\ & J_{MF}^{(IN)}(E_T) + J_{MF}^{(OUT)}(E_T) \Big]_{acc} \Delta E \end{aligned} \tag{4.107}$$

while for the donor:

$$\begin{aligned} \frac{\partial Q_{don}}{\partial t} = \frac{1}{n_D} \sum_{n_D, E_T} \Big[& J_{MD}^{(IN)}(E_T) + J_{MD}^{(OUT)}(E_T) + \\ & J_{MF}^{(IN)}(E_T) + J_{MF}^{(OUT)}(E_T) \Big]_{don} \Delta E \end{aligned} \tag{4.108}$$

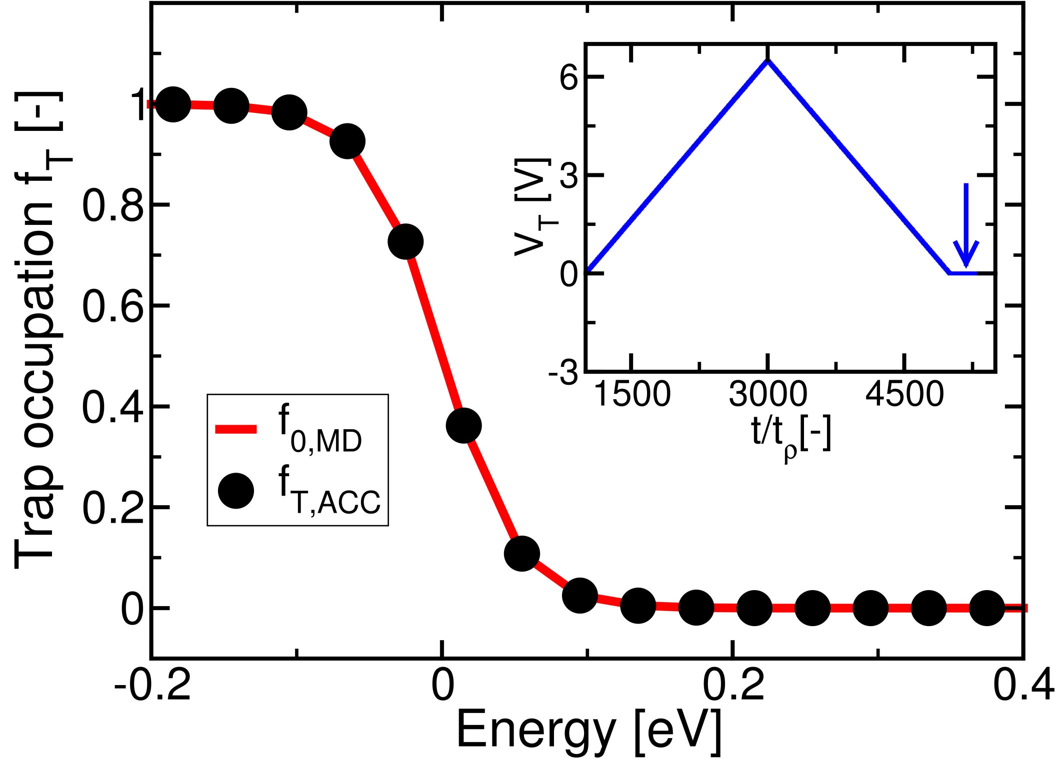

Figure 4.8: Steady state equilibrium occupation (i.e. at $V_T = 0$ V) for acceptor–type traps after a SET pulse. The trap energy on the x–axis is referred to the Fermi level $E_{f,MD}$ of the MD electrode. The red line is a plot of the Fermi occupation function $f_{0,MD(F)}(E)$ of the MD electrode. The inset shows the V_T waveform used for the simulations and the vertical arrow indicates the time at which the trap occupation f_T has been plotted.

Now we can rewrite Eq. 4.67 adding the charge trapping terms:

$$
\begin{aligned}
i_{MF}(t) = & -\frac{C_F}{C_0}\frac{\partial Q_{acc,AV}}{\partial t} + C_S\frac{\partial V_T}{\partial t} + \frac{C_D}{C_0}\frac{\partial P}{\partial t} \\
& + \frac{1}{n_D}\sum_{n_D,E_T}\Big[J^{(IN)}_{MF,acc}(E_T) + J^{(OUT)}_{MF,acc}(E_T) \\
& J^{(IN)}_{MF,don}(E_T) + J^{(OUT)}_{MF,don}(E_T)\Big]\Delta E - i^{(dir)}_{tunn}(t) = \\
& = -\frac{C_F}{C_0}\frac{\partial Q_{acc,AV}}{\partial t} + C_S\frac{\partial V_T}{\partial t} + \frac{C_D}{C_0}\frac{\partial P}{\partial t} + \\
& \left[I^{in}_{MF} - I^{out}_{MF}\right] - i^{(dir)}_{tunn}(t)
\end{aligned}
\tag{4.109}
$$

$$
\begin{aligned}
i_{MD}(t) &= -\frac{C_D}{C_0}\frac{\partial Q_{acc,AV}}{\partial t} - C_S\frac{\partial V_T}{\partial t} - \frac{C_D}{C_0}\frac{\partial P}{\partial t} \\
&+ \frac{1}{n_D}\sum_{n_D,E_T}\Big[J^{(IN)}_{MD,acc}(E_T) + J^{(OUT)}_{MD,acc}(E_T) \\
&J^{(IN)}_{MD,don}(E_T) + J^{(OUT)}_{MD,don}(E_T)\Big]\Delta E + i^{(dir)}_{tunn}(t) \\
&= -\frac{C_D}{C_0}\frac{\partial Q_{acc,AV}}{\partial t} - C_S\frac{\partial V_T}{\partial t} - \frac{C_D}{C_0}\frac{\partial P}{\partial t} + \\
&\big[I^{in}_{MD} - I^{out}_{MD}\big] + i^{(dir)}_{tunn}(t)
\end{aligned}
\tag{4.110}
$$

which, by recalling Eq.4.107, ensures $i^{M,F}(t)+i^{M,D}(t)=0$. In general terms, the conduction current could be non zero in state condition.

Chapter 5

Operation and design of Ferroelectric Tunnel Junctions

In this chapter, we initially calibrate our model with some experimental results reported in the literature, providing a set of parameters to reproduce experiments for different stacks of oxides. In particular, Sec. 5.1, extract from the *MFM* structure a full set of anisotropy constants for the ferroelectric material. By exploiting those values we are able to predict the read tunneling current of the devices reported in [141] and we also provide different design options to enhance *FTJs* . Hereafter we take into account the presence of charge trapped at the *FE-DE* interface as discussed in [176]. Consequently, we reported a quasi-static analysis in Sec. 5.2 that shows the impact of the charge-trapped compensation of the ferroelectric polarization in *FTJs* . The results of Sec. 5.2 have been confirmed and extended in the following discussion of Sec. 5.3, where a joint effort based on experiments and comprehensive numerical modelling investigates the role of polarization switching, charge trapping and depolarization effects in *MFIM* structures. Last but not least, Sec. 5.4 quantitatively investigates the trade–offs implied in the operation of *FTJs* as synaptic devices. Moreover, we report optimal compensation conditions that can be exploited to optimize *FTJs* as synaptic devices with multiple conductance levels.

5.1 Operation and design of defects free FTJs

In this section we calibrate and validate our model with measurements [141, 177]: firstly, we compare some measurements of a *MFM* stack with our simulation results to extract the anisotropy constants of the ferroelectric material. Then we used those values to simulate the *MFIM* stack and compare read currents in the quasi-static regime with the experimental values reported in [141]. After that, we use our calibrated simulator to investigate some design trade–off, exploring different oxide materials and metal contacts to improve the maximum read current and the *TER* ratio of these devices.

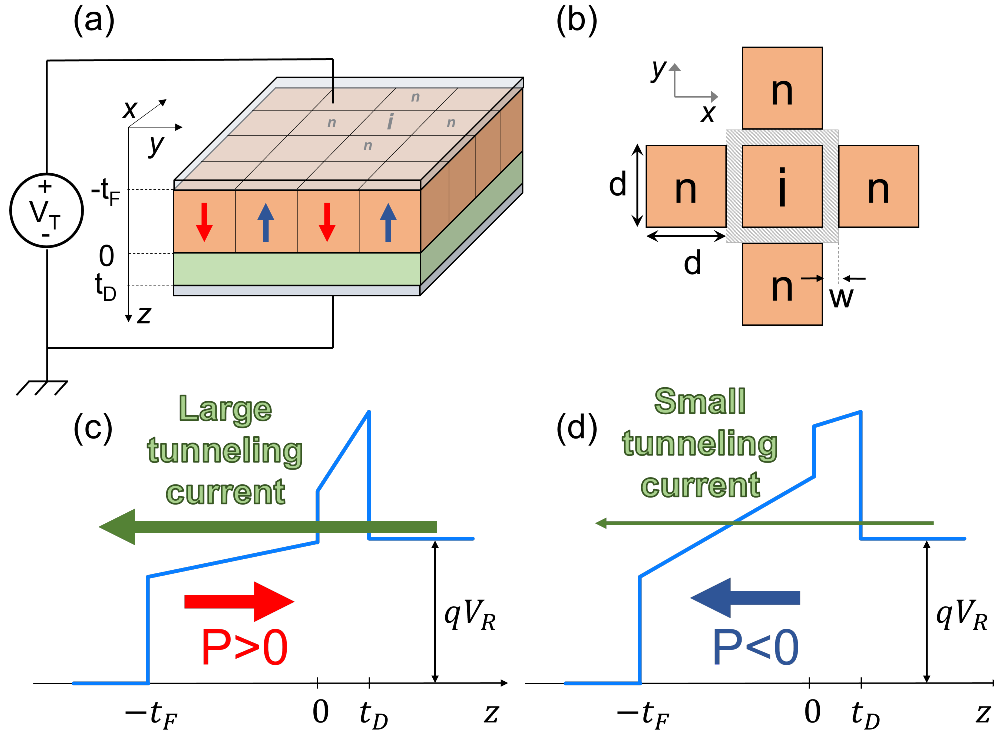

Figure 5.1: (a) Sketch of an MFIM based FTJ, where MF, MD are the electrodes contacting respectively the ferroelectric and the dielectric, while t_F, t_D, ε_F and ε_D are the ferroelectric and dielectric thickness and permittivity, respectively. A positive ferroelectric polarization points towards the dielectric (red arrow). (b) Zoom on ferroelectric domains where d is the side of the square domain and w is the width of the domain-wall region used for the domain wall energy in LGD (Eq. 4.38) [63]. Throughout the work we used $d = 5$ nm, $w/d = 0.1$ and the domain wall coupling factor in Eq. 4.38 was set to $k/w = 2 \times 10^{-3}$ [m^2/F]. V_T is the external bias. (c),(d) Qualitative band diagram across the dielectric stack during the read operation, where V_R denotes the read voltage. (c) Positive polarization and low resistance state; (d) Negative polarization and high resistance state.

	HZO	Al_2O_3	SiO_2
χ_D, χ_F [eV]	2.1	1.4	0.95
$m_{ox(F,D)}[m_0]$	0.4	0.3	0.5
ϵ_D, $\epsilon_F[\epsilon_0]$	30	10	3.9

Table 5.1: Material parameters used in simulations. The work function Φ_M of Al and TiN were taken as $4.08eV$ and $4.55eV$, respectively.

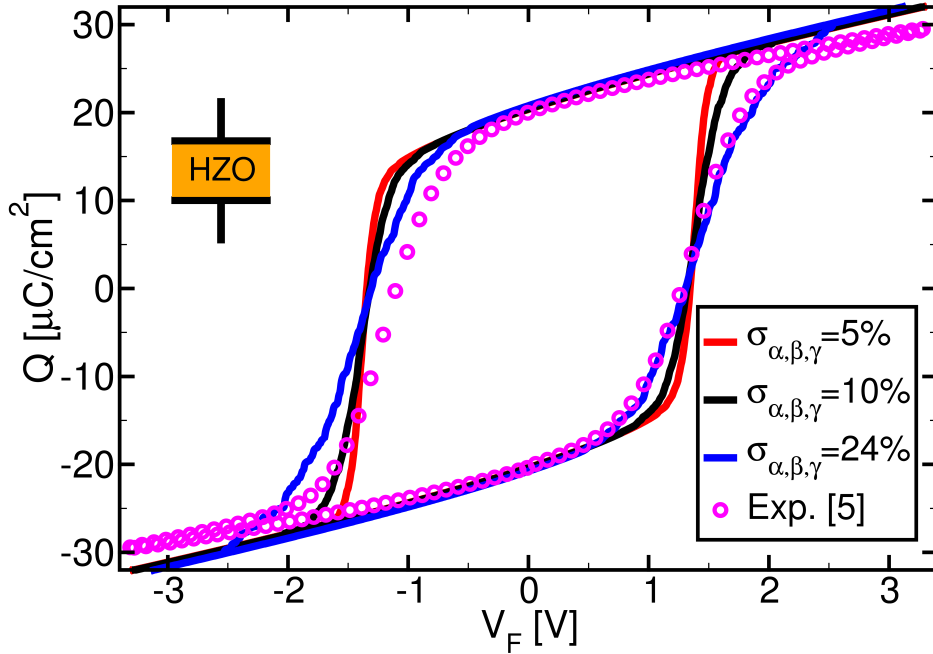

Figure 5.2: Experiments from [177] (symbols) and simulated polarization (lines) versus ferroelectric voltage characteristic of a $Hf_{0.5}Zr_{0.5}O_2$ layer with $t_F = 12$ nm. The nominal values of the anisotropy constants in the LGD equations are $\alpha = -5.8 \cdot 10^8$ m/F, $\beta = 2.9 \cdot 10^9$ m^5/F/C^2, $\gamma = 6.5 \cdot 10^{10}$ m^9/F/C^4, and domain to domain variations are introduced according to a normal distribution of α_i, β_i, γ_i, where $\sigma_{\alpha,\beta,\gamma}$ denote the standard deviations normalized to the mean values. Both experiments and simulations correspond to a quasi-static condition.

5.1.1 Model validation and analysis of experimental data

The models have been validated by comparison with recent experimental data for both the ferroelectric Q–V_F curve (with $Q = P + \varepsilon_0\varepsilon_F E_F$) [177], and the read current in corresponding *FTJs* [141]. Figure 5.2 shows the Q-V_F curve measured in the *MFM* system of [177], featuring a 12nm thick $Hf_{0.5}Zr_{0.5}O_2$ (HZO); the simulations with the LGD multi–domain model described in Sec. 4.3 are also reported. The agreement between simulations and experiments is good and the matching in the switching region improves by accounting for the domain–to–domain variations of α_i, β_i and γ_i. Throughout this section we will use the nominal values for α, β and γ reported in the caption of Fig. 5.2.

We simulated a *MFIM* structure featuring a 12 nm *HZO* ferroelectric, a 2 nm Al_2O_3 dielectric and TiN metal electrodes [141] (see Tab. 5.1 for material parameters). Figure 5.3 (top) shows examples of the setting and reading waveforms applied during the simulations of the *FTJs*, that were shaped to emulate the triangular waveforms used in the experiments of [141]. As it can be seen from the x-axis, the V_T waveforms are very slow compared to t_ρ, hence the simulations correspond to a quasi static operation.

In Fig. 5.3 (bottom), different maximum SET voltage values (V_{SET}) have been used, clearly resulting in different fractions, f_{UP}, of domains with a positive polarization, stemming from the minor loops in the Q versus V_T curve shown in the inset. By inspecting the f_{UP} in the set and read operation we also see that the *MFIM* device suffers from a quite strong depolarization effect. In fact, for a given V_{SET}, the f_{UP} during retention (i.e. for $V_T = 0$ V) and read (i.e. for $V_T = 2$ V) is significantly smaller than the corresponding f_{UP} reached in the SET operation. This occurs because

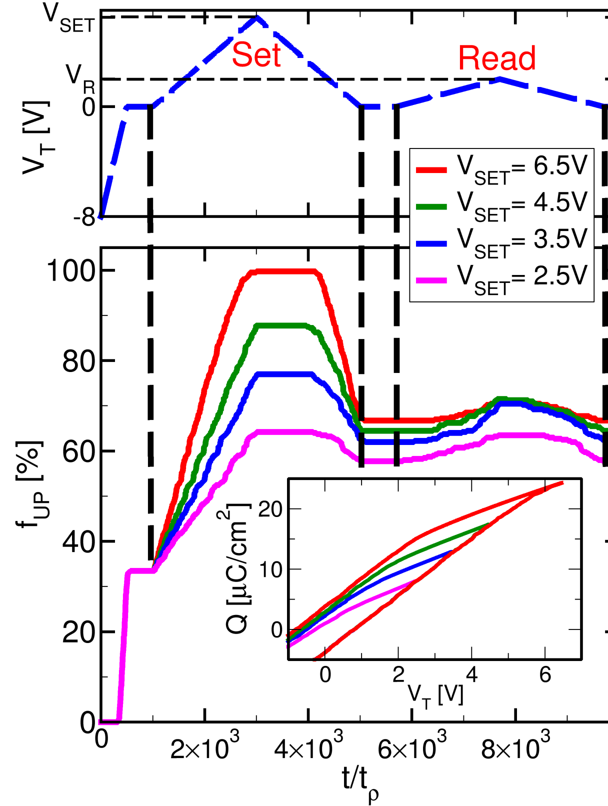

Figure 5.3: (Top) Examples of setting and reading waveforms used in the simulations of the *MFIM* based *FTJs* of [141] (see parameters in Tab. 5.1); read voltage is $V_R = 2$ V. (Bottom) Corresponding fractions f_{UP}, defined as the percentage of the domains having a positive polarization during the set and read operation and for different V_{SET}. The inset shows minor loops in the Q versus V_T plots corresponding to different V_{SET} values.

the fairly large dielectric thickness t_D=2 nm results in a strong depolarization field (see the sketch in Fig. 4.6(b)), producing a backswitching to $P_i < 0$ of some of the domains that had switched to $P_i > 0$ during the setting phase.

Figure 5.4 finally compares simulations and experiments for the I_R of *FTJs* at a read voltage $V_R = 2$ V. Our simulations in Fig. 5.4 can track the experiments quite well with the reasonable values of the oxide mass m_{ox} reported in Tab. 5.1 [185].

5.1.2 Simulation based design of FTJs

The minimum I_R value required by applications is set by the transistors leakage current and by the noise of the sense amplifier and, for recent designs of neuromorphic processors, a reasonable target is approximately 100 pA [102, 186, 187]. The results in Fig. 5.4 show that, for $V_{SET} = 2.5$ V, a device area larger than 10^4 μm^2 is needed to reach the $I_R = 100$ pA limit. Moreover the ON/OFF current ratio $R_I = [I_{R,max}/I_{R,min}]$ is only about ten, that seems too small for the desired 4–bit resolution of the synaptic weights [188]. Hence the primary goals of our *FTJ* design exploration are the increase of I_R and R_I.

The most obvious route to increase I_R is the scaling of the dielectric thickness t_D, whose effects are illustrated in Fig. 5.5 for the HZO/Al_2O_3 stack. As it can be seen, by thinning the Al_2O_3 layer the I_R increases, but R_I degrades (see inset). A marked R_I reduction with decreasing t_D has been

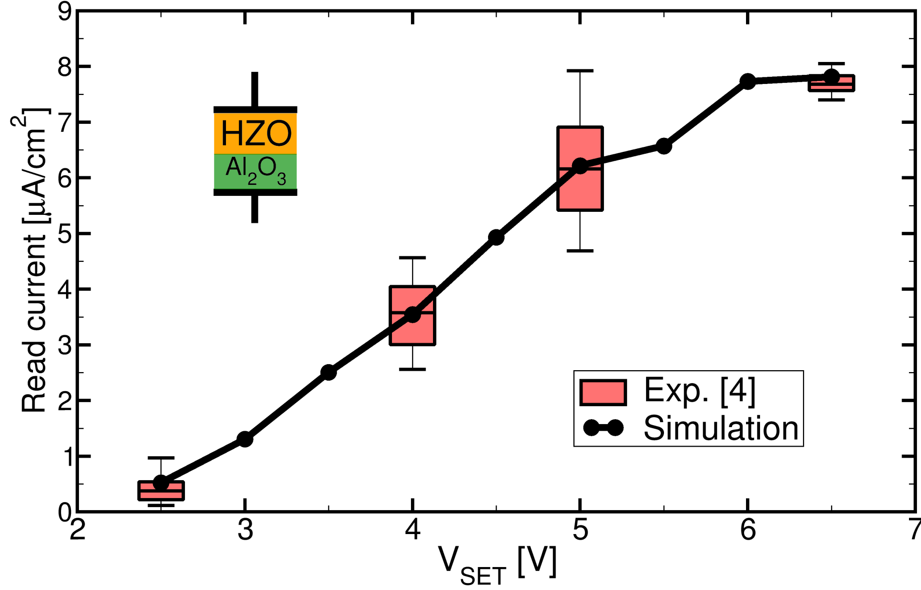

Figure 5.4: Experiments from [141] (boxes, device area $\approx 3.14 \cdot 10^{-4}$ cm^2), and simulations (black solid line) for the read current at $V_R = 2$ V of an HZO/Al_2O_3 *FTJ* (12 nm / 2 nm) and versus the set voltage V_{SET}. The error bars for experiments were inferred from the cycle to cycle variations reported in Fig. 3(d) of [141].

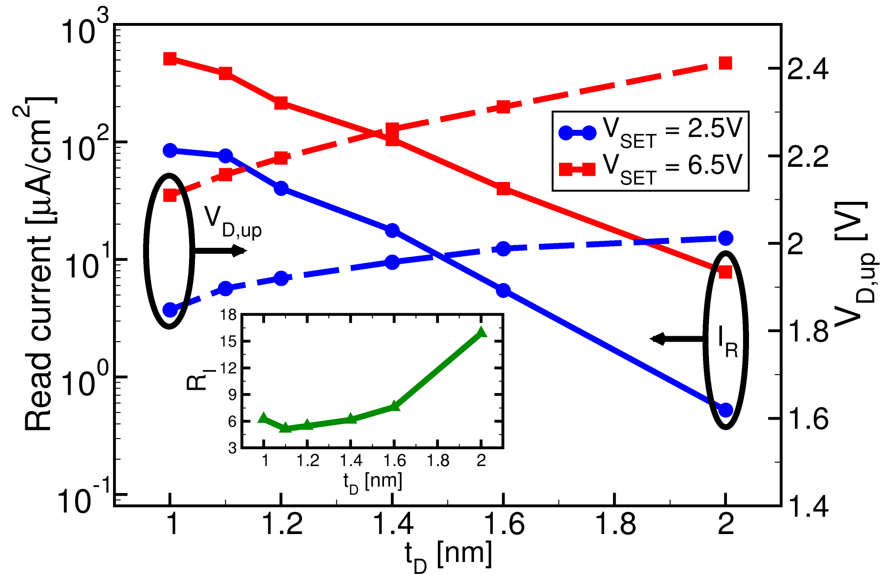

Figure 5.5: Read current at $V_R = 2$ V (left y axis) versus the Al_2O_3 thickness for an HZO/Al_2O_3 FTJ ($t_F = 12$ nm) and for $V_{SET} = 2.5$ V or 6.5 V. The average voltage drop, $V_{D,up}$, for the positive polarization domains is also reported (right y axis) in read condition, and the current ratio R_I=[$I_{R,max}/I_{R,min}$] is shown in the inset.

observed also in experiments [177]. Fig. 5.5 also shows that, by thinning Al_2O_3 and thus increasing C_D, the average $V_{D,up}$ for the positive polarization domains decreases (right y axis), and eventually $qV_{D,up}$ cannot overcome $[\Phi_{MD} - \chi_F] \simeq 2.45$ eV (see Fig. 4.6(a)).

In order to reduce C_D for a given t_D, we replaced the tunneling oxide with SiO_2, having a dielectric constant about 2.5 times smaller than Al_2O_3. Figure 5.6(a) reports the read current I_R and R_I versus V_{SET} for two variants of a HZO/SiO_2 based FTJ, and compared to the HZO/Al_2O_3 baseline case of Fig. 5.4 (black line). By using a 1 nm SiO_2 layer and maintaining TiN electrodes the I_R at V_R=2.0 V largely increases (red line), but R_I does not improve. In the second option of

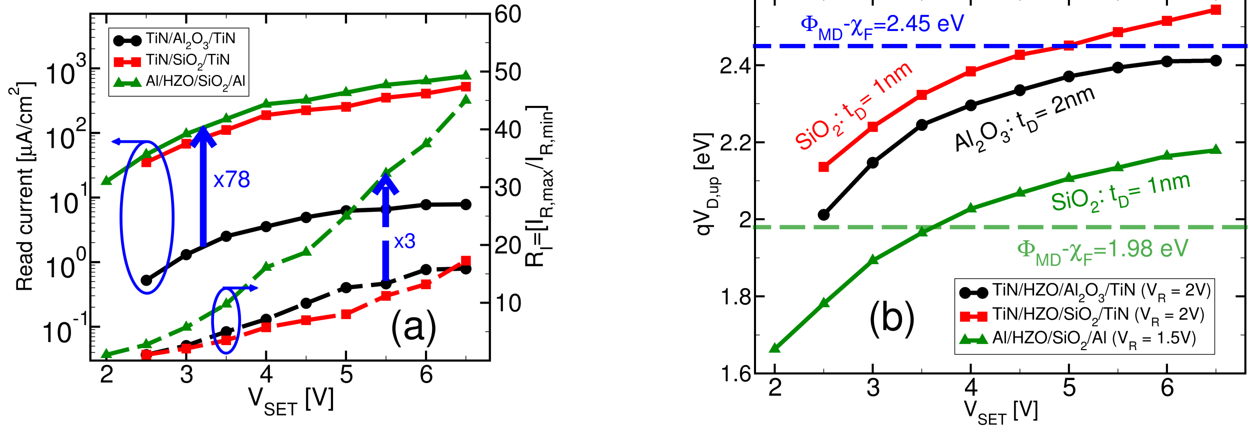

Figure 5.6: (a) Read current (left y axis) and $R_I = [I_{R,max}/I_{R,min}]$ (right y axis) versus the SET voltage for the TiN/HZO/Al_2O_3/TiN structure ($t_D = 2$ nm, $V_R = 2$ V) in Fig. 5.4, and for two variants of an HZO/SiO_2 based FTJ, namely for TiN/HZO/SiO_2/TiN ($t_D = 1$ nm, $V_R = 2$ V), and for Al/HZO/SiO_2/Al ($t_D = 1$ nm, $V_R = 1.5$ V). The HZO thickness is $t_F = 12$ nm in all cases. (b) Average voltage drop, $V_{D,up}$, for the positive polarization domains (as in Fig. 5.5), for the different design options. The ferroelectric tunnelling barrier $[\Phi_{MD} - \chi_F]$ is substantially reduced for the Al electrode.

the HZO/SiO_2 based *FTJ* we considered a low workfunction Al electrode ($\Phi_M \simeq 4.08$ eV), so as to reduce the SiO_2 tunnelling barrier $[\Phi_{MD} - \chi_D]$. This leads to a large I_R increase at fixed V_R, that we exploited to decrease both V_R to 1.5 V and the minimum V_{SET} to 2 V. The corresponding results in Fig. 5.6(a) (green line) show a large improvement for both I_R and R_I compared to the HZO/Al_2O_3 case. Indeed, Fig. 5.6(b) reveals that the 1 nm SiO_2 design leads to $V_{D,up}$ values comparable to the 2 nm Al_2O_3 for $V_R = 2$ V; moreover for Al electrodes the $qV_{D,up}$ can overcome $[\Phi_{MD} - \chi_F]$ even for $V_R = 1.5$ V.

The engineering of the metal work-function for the capping electrodes is still quite actively investigated for the design of *FTJs* [189, 190]. In this respect, while the crystallization annealing with a TiN electrode is the most popular option to induce ferroelectricity in the HZO, a robust *HZO* ferroelectricity has been experimentally reported also for different metal electrodes [189, 191].

In this section, we compared experiments reported in [177] with our LGD in-house developed simulator. We began our analyses by fitting charge-versus-voltage measurements of *MFM* capacitors with our model, then we analyzed in a quasi-static regime the behavior of *MFIM* structure, providing also a comparison between simulations and experiments of the read tunneling current. After that, we engineered different *FTJs* structures exploiting also different work-functions. Our simulation results show an enhanced read current with a decent *TER* ratio that can be useful to improve the number of discrete levels and scalability of *FTJ* devices. However, as recently stated in [176], charge trapped in the oxide stack could affect the general behavior of the device. Therefore, in the next session, we discuss the presence of charge trapping at the *FE-DE* interface and how it affects the quasi-static behavior of *FTJs* .

5.2 Charge trapping induced compensation of the ferroelectric polarization in quasi-static regime

In the previous section, we investigated the quasi-static behavior of *FTJ* devices under the simplistic assumption of a clean interface between the ferroelectric and the dielectric layer. We also proposed a novel design perspective to enhance the window between the minimum and maximum currents during the read operation by combining the perks of exploiting a SiO_2 interlayer instead of the alumina and a non-negligible work-function engineering.

However, because it has been argued that in ferroelectric–dielectric stacks the charge injection and trapping can compensate to a large extent the ferroelectric polarization [173], we have here analyzed the influence of such a charge trapping on the operation of *FTJs*. In order to be as much consistent as possible with the previous analysis, we initially investigated the behavior of such devices considering the same structures and an equal set of parameters as the one reported in Sec. 5.1. In this perspective, with this first quasi-static and FE/DE interface charge trapping enabled exploration, our aim is to provide a qualitative direct comparison between the results of Sec. 5.1 and the following analysis. Therefore, we here first introduce the quasi-static results of the trapping model described in Sec. 4.3.6. In particular, this model allows us to have the trap occupation f_T for any waveform of the external bias V_T, which can be used to calculate the charges Q_{acc}, Q_{don} trapped in respective acceptor and donor traps as expressed in Eqs. 4.90 and 4.91, thus reported below to for the sake of clarity:

$$Q_{acc} = \frac{(-q)}{n_D} \sum_{E_T} N_{acc}\, f_T(E_T)\, \Delta E,$$

$$Q_{don} = \frac{q}{n_D} \sum_{E_T} N_{don}\, \left(1 - f_T(E_T)\right)\, \Delta E,$$

where N_{acc}, N_{don} denote the trap densities and ΔE is the energy step between the discrete trap levels. The overall interface charge Q_{int} is simply the sum of Q_{acc} and Q_{don}. Moreover, in all simulations including traps, in each domain the charge Q_{int} was duly added to the polarization P in the calculation of all relevant quantities, such as $V_{D,i}$, $V_{F,i}$ in Eqs. 4.39. As discussed above, the tunneling transmission $T_{MD}(E_T)$ and $T_{MF}(E_T)$ of Eqs. 4.79 and 4.83 were calculated through the WKB approximation, by using the effective tunnelling masses m_D, m_F, and the energy barriers $\Phi_D = (\Phi_{TiN} - \chi_D)$, $\Phi_F = (\Phi_{TiN} - \chi_F)$ in Tabs. 5.1 and 5.2.

σ_E [meV]	σ_T [cm^2]	m$_D$ [m$_0$]	m$_F$
7	$5 \cdot 10^{-14}$ - $1 \cdot 10^{-15}$	$0.23 - 0.3$	0.4

Table 5.2: The acceptor and donor traps are located 0.4 eV and 2.4 eV below the conduction band minimum at the ferroelectric–dielectric interface, respectively, and they both extend for 2 eV wide range in the energy gap.

Figure 5.7 shows an example of the waveforms for the average spontaneous polarization, P, and interface charge, Q_{int}, during a SET and a RESET pulses followed by a read pulse. As it can be seen, the acceptor and donor trap densities are such that approximately 50% of the ferroelectric polarization is compensated by Q_{int} during the SET/RESET phase, as well as during the read operation. Furthermore, the average P and Q_{int} tend to follow specular trajectories in Fig. 5.7. If

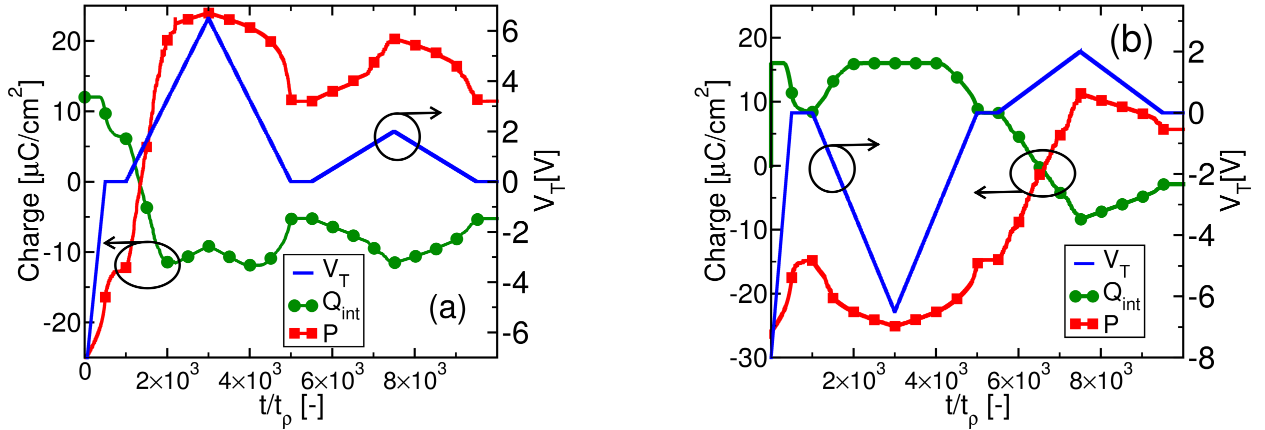

Figure 5.7: Simulated spontaneous polarization (P, red line) and interface charge (Q_{int}, green line) averaged across the device area. (a) Results for a SET pulse followed by a 2 V read waveform; (b) results for a RESET pulse followed by the same read waveform. The interfacial trap densities used in these simulations are $N_{acc} = N_{don} = 5 \times 10^{13}$ [cm^{-2}eV^{-1}].

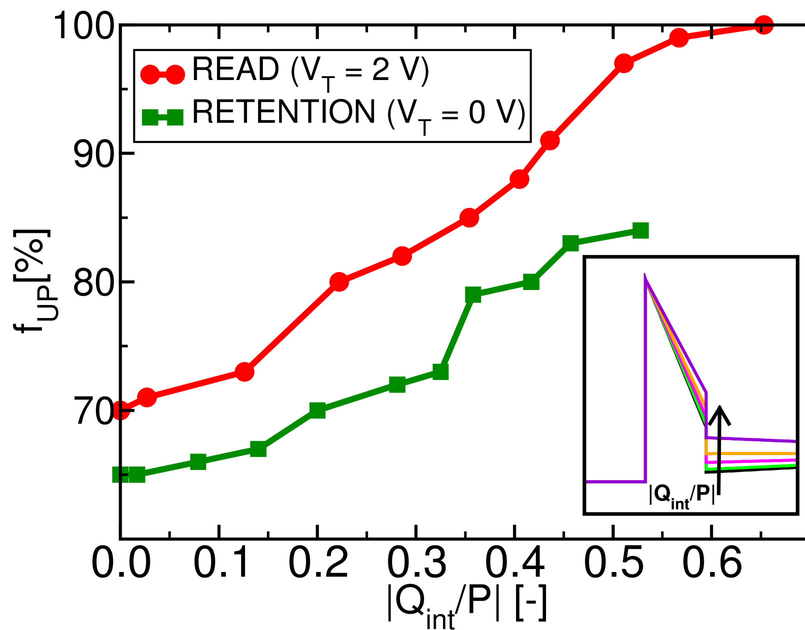

Figure 5.8: Percentage of domains with a positive polarization, f_{UP}, versus the charge compensation produced by interface traps and here quantified as $|Q_{int}/P|$). The f_{UP} is evaluated after a SET pulse during either retention or read phases. The inset illustrates the conduction band profile (averaged across the device area) close to the ferroelectric–dielectric interface and evaluated during the read pulse. The interfacial trap densities (with $N_{acc} = N_{don}$) varies between 0 and 5×10^{13} [cm^{-2}eV^{-1}]

we focus on the rising V_T ramp in Fig. 5.7(a), for example, this can be understood because, when Q_{int} produces a significant compensation of P, then the Q_{int} detrapping and the rising V_T ramp have an additive effect on the increase of the ferroelectric field that produces the P switching. This creates the link between Q_{int} and P that results in roughly specular variations.

We now notice that the trapping induced compensation of the ferroelctric polarization observed in Fig. 5.7 is instrumental in order to stabilize the polarization by reducing the otherwise very large depolarization field in the ferroelectric. This latter point is best illustrated by Fig. 5.8, reporting the percentage f_{UP} of domains with a positive polarization after a SET pulse versus the charge compensation quantified as $|Q_{int}/P|$. The results are shown for either the retention (i.e. $V_T = 0$ V)

or the read phase (i.e. $V_T = 2$ V). As expected, a larger polarization can be stabilized by increasing Q_{int}, however this does not result in an improved band bending in read mode. In fact, the inset shows that, by increasing $|Q_{int}/P|$, the average voltage drop, V_D, across the dielectric degrades and the *HZO* conduction band minimum at the ferroelectric–dielectric interface is pushed up. This is a serious drawback of the charge compensation via Q_{int}, which hinders the attainment of a band diagram favorable for the tunneling injection through the thin dielectric layer.

It is presently difficult to be more quantitative about the trade–off between the favorable and the detrimental effects of the trapping-induced compensation of the ferroelectric polarization, particularly because it is difficult to estimate the trapped charge in actual devices. In this latter respect, the values of trap densities considered in Figs. 5.7, 5.8 are admittedly quite large, however, they are not at all inconsistent with recent experimental papers that have reported estimates for the areal density of trapped charge in *HZO* based *FTJs* or FeFETs in the range of 10^{14} cm^{-2} [174], [192].

In summary, by comparing the charge trapping free *FTJ* results reported in Fig. 5.3 with the one reported in Figs. 5.7 and 5.8, we observe a strong dependence of the stable average polarization with the amount of compensation provided by the trapped charge at the interface between dielectric and ferroelectric material. This discrepancy is more visible by comparing the ratio of domains with positive polarization. In fact, in the case without any charge trapping, f_{UP} tends approximately 70% during the read pulse when the maximum set voltage is applied. On the other hand, when strong compensation is taken into account, that value reaches the maximum, synth of a homogeneous polarization condition (see the red curve in Fig. 5.8). This means that traps could play a crucial role in the switching dynamics of ferroelectric materials. To shed some light on this, in the next section, we will discuss more thoroughly this interplay between charge trapping and ferroelectric polarization by comparing and matching dynamic measurements with a complete set of dynamic (non quasi-static) simulations.

5.3 Interplay between charge trapping and polarization switching in BEOL-compatible FTJs

Energy efficiency is a main target for neuromorphic computing and it is a major concern for all computational technologies [193, 194]. As shown in Sec. 5.1, thanks to the field-driven polarization switching, *FTJs* can provide high impedance and low energy synaptic devices. A four-level operation has been demonstrated in a *MFIM* structure [141]. Moreover, as anticipated in Sec. 5.1, charge injection through the thin dielectric and charge trapping are expected to play a prominent role in the switching and stabilization of the polarization [173, 195] in ferroelectric-dielectric (FE-DE) systems. In fact, large interface charge densities have been reported both for *FeFETs* [174, 196] and *MFIM* capacitors [175, 192], albeit some quantitative aspects are still actively debated [176].

In this section we present a joint effort based on experiments and comprehensive numerical modelling to investigate the role of polarization switching, charge trapping and depolarization effects in TiN/$HfZrO_4$/Al_2O_3/TiN CMOS compatible FTJs. Our results help in clarifying the physical operation of the devices and provide a sound basis for the device design.

5.3.1 Device fabrication and measurements

Large area (7850 μm^2) FTJ structures were fabricated, featuring TiN top- and bottom-electrodes, an aluminum oxide (Al_2O_3) tunneling layer at the top electrode interface and a 10 nm thick hafnium zirconium oxide ($Hf_{0.5}Zr_{0.5}O_2$ or, equivalently, $HfZrO_4$) ferroelectric between the bottom electrode and the tunneling layer. The Al_2O_3 layer thickness was varied between 1.5 nm and 2.0 nm.

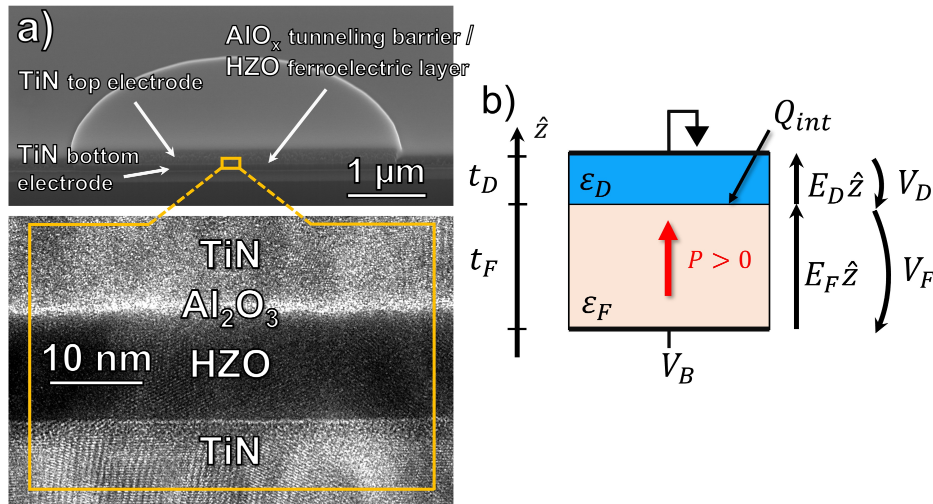

Figure 5.9: a) High resolution TEM cross section of the TiN/1.5 nm Al_2O_3/10 nm HZO/TiN devices used in experiments; b) Sketch of the device template for numerical simulations.

The TiN electrodes were deposited by physical vapor deposition at 350°C. Figure 5.9(a) reports a high-resolution TEM cross-section of the device. Prior to the *HZO* deposition, the TiN bottom electrode underwent chemical mechanical polishing (CMP) to improve roughness (rms= 0.1 nm). Both oxide layers were grown by atomic layer deposition at 280°C. $HfZrO_4$ films were deposited with HyALD and ZyALD precursors and O_3 as oxidant, at a rate of 0.8 /cycle using 1:1 supercycles of HfO_2 and ZrO_2 to ensure uniform stoichiometry. Al_2O_3 was deposited at a rate of 1 /cycle using TMA as a precursor and O_3 as oxidant. The maximum thermal budget for all structures was 450°C

in order to be compatible with *BEOL* integration. Reactive Ion etching (RIE) was used to etch the TiN/Al_2O_3/$HfZrO_4$ stack with an etch stop in the TiN bottom electrode.

Grazing-incidence X-ray diffraction measurements were undertaken on nominally identical film stacks without CMP bottom electrodes, annealed at 500°C, on a Bruker D8 Discover X-ray analyzer. 2θ scans were measured in the grazing incidence geometry, at an incidence angle of 0.45°, to minimize the signal from the underlying substrate and allow the identification of crystal phases of the $HfZrO_4$ film. These data are plotted in Fig. 5.10(a) for stacks with 1.5 and 2 nm Al_2O_3, and the dashed lines indicate the expected peak positions of the strongest crystal reflections in these films [197]. The spectra were fitted with Gaussian peaks centered around these positions and the areas of the fitted peaks were used to calculate the phase fractions of the m-, o- and t-phases in our films.

Ferroelectricity in $HfZrO_4$ is attributed to the polar orthorhombic (o-) phase, while phase transformations can also occur during electric field cycling, contributing to the wake-up effect [170, 198]. Thus the high o-phase fraction of these films in the pristine state (57% and 54% for 1.5 and 2 nm, respectively) suggests their good ferroelectric properties, despite the relatively thick dielectric interlayers. At the annealing temperatures used for both *BEOL* and reference samples, the Al_2O_3 layer remains amorphous.

This section compares simulations with measurements obtained with the *Positive-Up-Negative-Down (PUND)* technique, which was originally conceived for *MFM* systems [142, 199], and is nowadays also routinely employed in *MFIM* device structures [200–204]. The *PUND* method consists of a series of triangular pulses (see also Fig. 5.10(c)) and it is widely used in the characterization of ferroelectric devices [139, 142, 170]. Hereafter, we used a 10kHz *PUND* with a delay time of 5ns between each pulse. Wake-up of the experimental results was achieved by bipolar triangular cycling (1000 cycles - 100kHz frequency) with a maximal voltage the same as the PUND.

Figure 5.10(b) represents the switched charge measured for 10 kHz triangular pulses of 6 V, for samples with (blue solid lines) and without (red dashed lines) CMP bottom electrodes. The bias was applied on the bottom electrode. The plotted curves are obtained by integrating the difference in P-U and N-D currents, which should provide an estimate of the switched polarization without the influence of leakage and displacement currents. It should be pointed out that this does not necessarily correspond entirely to the polarization switching current, as highlighted recently [204]. From Fig. 5.10(b) it is clear, nonetheless, that the switched charge is larger for CMP devices. As the film stacks are nominally identical, the increase in switched charge may be related to an increased Pr due to different texturing of the TiN in the BEOL samples [205]. In addition, the reduced negative coercive voltage for the CMP process indicates a reduction in non-switching dead layers, and thus attests to the improvement of the bottom interface.

5.3.2 Models and comparison to experiments

The ferroelectric dynamics and electrostatics of an FTJ with the structure illustrated in Fig. 5.9(b) were self-consistently solved by using the multi-domain Landau, Ginzburg, Devonshire (LGD) model thoroughly described in Sec.4.3. The nominal values for the anisotropic constants used in simulations are α=$-1.1 \cdot 10^8$ m/F, β=$-1.5 \cdot 10^{10}$ m^5/F/C^2, γ=$1.85 \cdot 10^{11}$ m^9/F/C^4, which result in a remnant polarization $P_r \approx 24$ μC/cm^2 and coercive field $E_c \approx 1.8$ MV/cm, that are in fairly good agreement with polarization-voltage response of *MFM* samples deposited under similar conditions [206]. Our simulations typically include n_D=100 domains, and we account for domain-to-domain

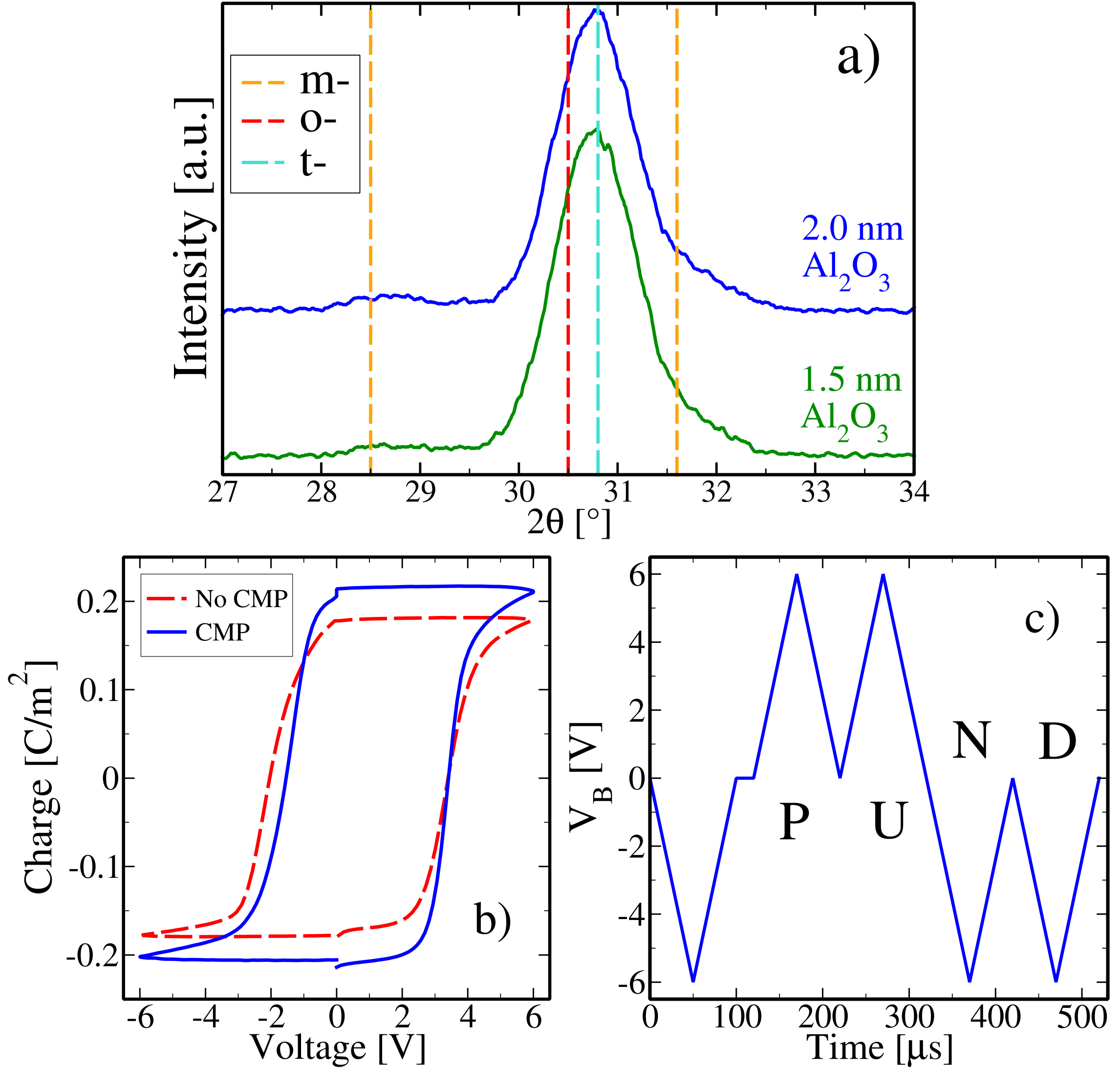

Figure 5.10: a) 2θ scans taken in grazing-incidence geometry for films stacks with nominally 10 nm HSO and 1.5 nm Al_2O_3 (green) and 2 nm Al_2O_3 (blue). Dashed lines indicate the expected peak positions for the monoclinic (m-), polar orthorhombic (o-) and tetragonal (t-) phases; b) comparison of the switched charge measured on a 6 V - 10 kHz PUND waveform (represented in c), for samples with/without CMP bottom electrodes

random variations of α, β, γ (see Tab.5.3). We have also verified that simulation results are practically insensitive to a further increase in the number of domains.

At each time t and external bias $V_B(t)$, the LGD equations provide the domain polarization $P_i(t)$, the electric fields in the ferroelectric $E_{F,i}$ and in the dielectric $E_{D,i}$, and thus the band diagram in each domain (with $i{=}1, 2 \cdots n_D$).

As shown in Fig. 5.3 and also in Fig. 5.11, simulations neglecting any trapped charge at the FE-DE interface (or inside the dielectrics) result in much more stretched P-V curves compared to experiments. These simulations are instead consistent experimental P-V curves reported for an *HZO* capacitor serially connected to a discrete ceramic capacitor, which ensures a negligible charge injection through the dielectric [173], or to measurements in *MFIM* structures with thicker Al_2O_3 layers similarly suppressing charge injection [155]. These results are also qualitatively consistent with the analysis previously reported in [214].

Material	Thick.[nm]	χ[eV]	ε_r [$ad.$]	Φ_M [eV]
$HfZrO_4$	10	2.4	34	-
Al_2O_3	$1.5, 2$	1.4	10	-
TiN	-	-	-	4.55

Table 5.3: Material parameters employed in simulations: ε_r is the relative permittivity, and χ, Φ_M are respectively the electron affinities of the dielectrics and the TiN workfunction. Calculations include domain to domain variations of α_i, β_i, γ_i parameters (with i=$1, 2 \cdots n_D$), corresponding to a ratio σ_{E_C} = 10% between the standard deviation and the mean value of the coercive field E_C. The resistivity for the ferroelectric is ρ=112 Ωm which is consistent with recently reported values for HZO based capacitors [207].

t_D	$\sigma_{T,acc}$ [cm^2]	$\sigma_{T,don}$	m$_D$ [m$_0$]	m$_F$
1.5 nm	$2.5 \cdot 10^{-16}$	10^{-16}	0.2	0.4
2.0 nm	$5 \cdot 10^{-15}$	10^{-15}	0.15	0.4

Table 5.4: Material parameters related to tunneling and trapping in $TiN/HfZrO_4/Al_2O_3/TiN$ FTJs. The value of m_D for Al_2O_3 and m_F for HZO are within the range reported in [208–210]; likewise, the values for $\sigma_{T,acc}$ and $\sigma_{T,don}$ are within the admittedly quite wide range of values reported in the literature [211–213]. The energy cross-section σ_E was set to 7 meV in all simulations. Acceptor and donor type traps are uniformly distributed in energy respectively from 0.6 to 2.6 eV and from 1.8 to 3.8 eV below the HZO conduction band, respectively (see also Fig. 4.6).

Figure 5.11(c) illustrates the I-V curves during the P and N pulses. In simulations with no traps (black solid line) the displacement current, $C_S(\partial V_B/\partial t)$, due to the linear polarization response is reproduced well (with C_S=$[1/C_D + 1/C_F]^{-1}$ and C_D=$\varepsilon_0\varepsilon_D/t_D$, C_F=$\varepsilon_0\varepsilon_F/t_F$). However, the simulated switching current ($\partial P/\partial t$) is spread over a large voltage range, which results in a relatively wide current plateau rather than a fairly narrow current peak exceeding the $C_S(\partial V_B/\partial t)$ contribution. This feature is again in stark disagreement with experiments.

The discrepancies between simulations and experiments in Fig. 5.11 suggest that charge injection and trapping in the dielectric stack plays an important role in the polarization switching for the *FTJs*, when considering the small dielectric thickness [173, 215]. Hereafter we will assume that conduction in the Al_2O_3 layer is limited by tunnelling, even if transport mechanisms assisted by defects through Poole-Frenkel and hopping mechanisms are also possible in thin oxides [184]. We assume that the most important trapping effects for the polarization switching occur close to the FE-DE interface, and thus we describe the trap density and trapped charge in terms of areal densities. However, it is understood that these figures should be regarded as equivalent areal densities possibly summarizing also a charge trapping in the DE and FE films.

The trapping model exploited in this section is thoroughly presented in Sec. 4.3.6. Acceptor and donor type traps at the FE-DE interface were described according to the first order dynamic equation (see Eq. 4.104). Moreover, all simulation parameters regarding the dynamic trapping model are reported in Tabs. 5.3 and 5.4.

Trapped acceptors and donor charges are defined in Eqs. 4.90 and 4.91 respectively. We here reported the expressions of Q_{acc} and Q_{don} for the sake of clarity:

$$Q_{acc} = \frac{-q}{n_D} \sum_{E_T} N_{acc}\, f_T(E_T)\, \Delta E,$$

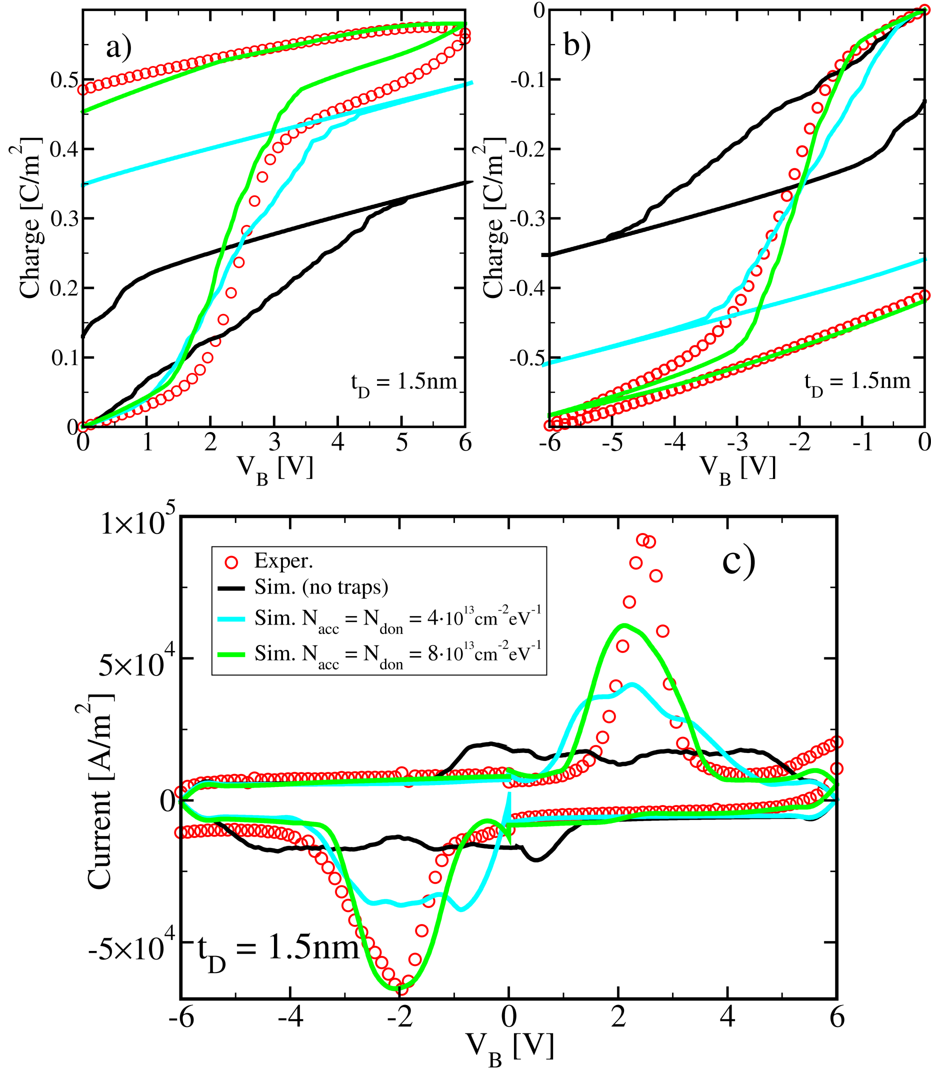

Figure 5.11: Polarization versus voltage characteristics measured by the PUND method for a 50 μs width of the triangular pulse and for an Al_2O_3 FTJ with t_D=1.5 nm. Results are reported for the P pulse (a), and for the N pulse (b). (c) Current versus voltage characteristics corresponding to the PUND measurements. Corresponding simulations are shown for no trapped charge (black solid line), and for different equivalent areal density of acceptor and donor type traps.

$$Q_{don} = \frac{q}{n_D} \sum_{E_T} N_{don} \left(1 - f_T(E_T)\right) \Delta E,$$

where N_{acc}, N_{don} denote the trap densities, ΔE is the energy step between the discrete trap levels and f_T is the traps occupation. The total trapped charge at the FE-DE interface can be consequently calculated as $Q_{int} = Q_{acc} + Q_{don}$.

Figure 5.11 suggests that simulations can be reconciled with experiments only by assuming a large equivalent trap density, and Fig. 5.11(c) shows that the corresponding simulated I-V plot can also track quite well the coercive voltage and the shape of the measured current. Figure 5.12 offers a simulation-based insight about the behavior of the average polarization and interface charge (P and Q_{int}, both averaged over the device area) along a PUND waveform. In this example both positive and negative P are compensated by Q_{int} to a large extent. This is the basic mechanism by which the depolarization field can be reduced compared to the case with $Q_{int} \approx 0$ C/m^2, and the simulated P-V curves reconciled with experiments. Figure 5.13 also reports the simulated field, E_D,

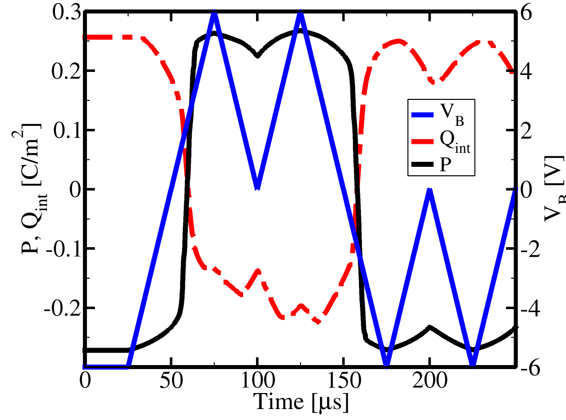

Figure 5.12: Polarization, P, and interface charge, Q_{int}=Q_{acc}+Q_{don}, versus time during for the same PUND simulations as in Figs. 5.11, 5.11. The Al_2O_3 thickness is t_D=1.5 nm and the trap densities are $N_{acc} = N_{don} = 8 \cdot 10^{13}$ 1/(cm^{2}eV). The V_B waveform is also shown (blue line).

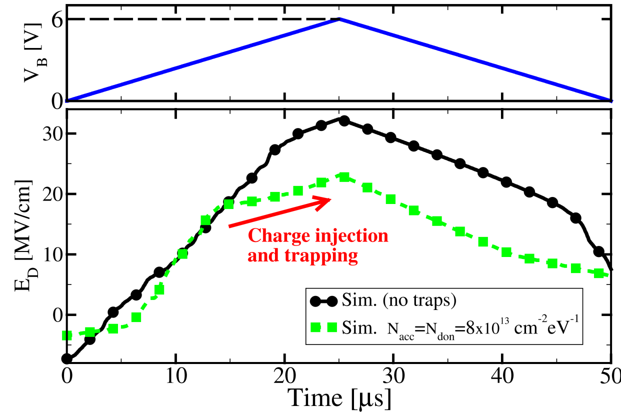

Figure 5.13: Electric field across the Al_2O_3 layer during the P pulse analyzed in Fig. 5.11(a), hence for t_D=1.5 nm.

across the Al_2O_3 layer during the P pulse analyzed in Fig. 5.11(a). In simulations with no traps, E_D exceeds 30 MV/cm at the V_B peak, which is an unrealistically large value and would also lead to a huge tunnelling current. In the presence of traps, instead, when E_D exceeds approximately 15 MV/cm electrons are injected through the dielectric and trapped at the FE-DE interface. The resulting building up of negative charge quenches the increase of E_D, which is a physical picture consistent with the one proposed in [173]. The reduction of E_D induced by the trapped charge is accompanied by an increase of E_F, which is however much smaller in magnitude than E_D.

Figure 5.14 shows that the results are qualitatively similar for the FTJs having a slightly thicker Al_2O_3 layer. Even for t_D = 2.0 nm the simulations with no traps are in sharp disagreement with experiments. Moreover, the same trap densities result in a fairly good agreement with P-V and I-V curves for both t_D = 1.5 nm and 2.0 nm.

The experiments in Figs. 5.11 and 5.14 suggest that, if it is the charge injection through Al_2O_3 that feeds the charge trapping at the FE-DE interface, then such an injection must be similarly effective for t_D = 1.5 nm and 2.0 nm. To obtain this behaviour in our tunnelling based model, for the t_D = 2.0 nm case it was necessary to increase the trap cross section σ_T and slightly decrease the Al_2O_3 tunnelling mass (see Tab.5.4). The need for an empirical adjustment of these parameters may suggest that additional transport mechanisms are involved in Al_2O_3 [184].

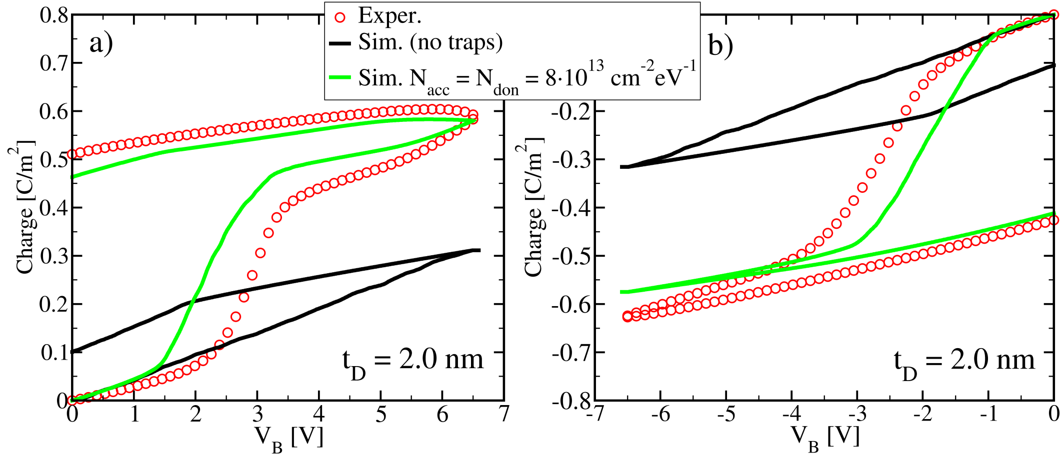

Figure 5.14: Measured and simulated P-V curves for a PUND waveform as in Fig. 5.11, but for an Al_2O_3 FTJ with t_D=2.0 nm.

5.3.3 Implications for FTJ device dynamics

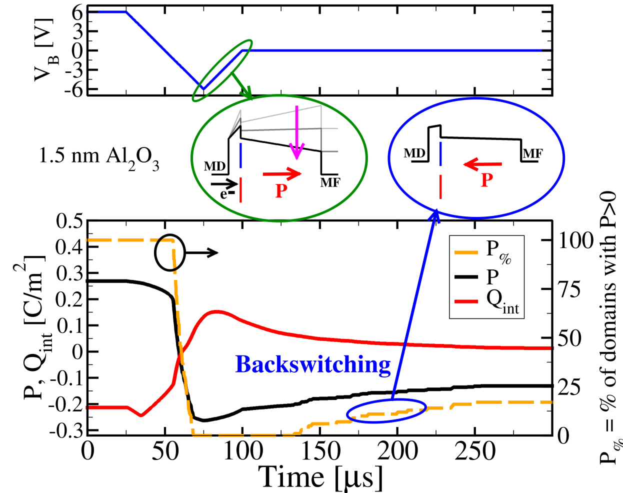

Figure 5.15: Polarization (black line) and interface charge (red line) during an N pulse and the following retention phase at $V_B = 0$ V. The percentage of positive polarization domains (yellow dashed line, right y axis) increases during the retention phase. Insets show the band diagram for negative (left) and a backswitched (right) polarization configurations.

The results of the previous section suggest that charge injection and trapping play an important role in the polarization switching and stabilization in the MFIM-based FTJs at study. As a corollary, we found that the interface charge has implications for different aspects of the device design. Figure 5.15, for example, examines the polarization loss at $V_B = 0$ V after the N pulse of a PUND sequence, which is linked to detrapping. In the simulations of Fig. 5.15, in fact, a fraction of the donor traps that have positively charged during the N pulse lie below the Fermi level when V_B goes back to zero. During the retention at V_B=0 V such donor traps capture electrons and thus become neutral (see Eq. 4.104). The resulting reduction of the positive interface charge Q_{int}≈Q_{don} enhances the depolarization field, which eventually leads to the back-switching of a fraction of domains.

Of course the behaviour in Fig. 5.15 is critically influenced by the position of the trap energy levels. However, a similar interplay between detrapping and back-switching is plausible in actual devices. As an example, the polarization loss along time was experimentally investigated in Fig. 5.16,

where the delay between two consecutive, negative identical pulses was varied within nine orders of magnitude. The polarization loss was here defined as the ratio between the polarization switched during the S2 pulse divided by the counterpart during the S1 pulse. Figure 5.16 shows that the lost polarization increases with time from 100 ns to 10 ms and then saturates for longer times, with a plateau at typically 30% of the total polarization for an alumina thickness of 1.5 nm. Increasing the alumina thickness from 1.5 nm to 2.0 nm is expected to increase the depolarization field and therefore the polarization loss at a fixed trapped charge areal density, which is in fact observed in Fig. 5.16.

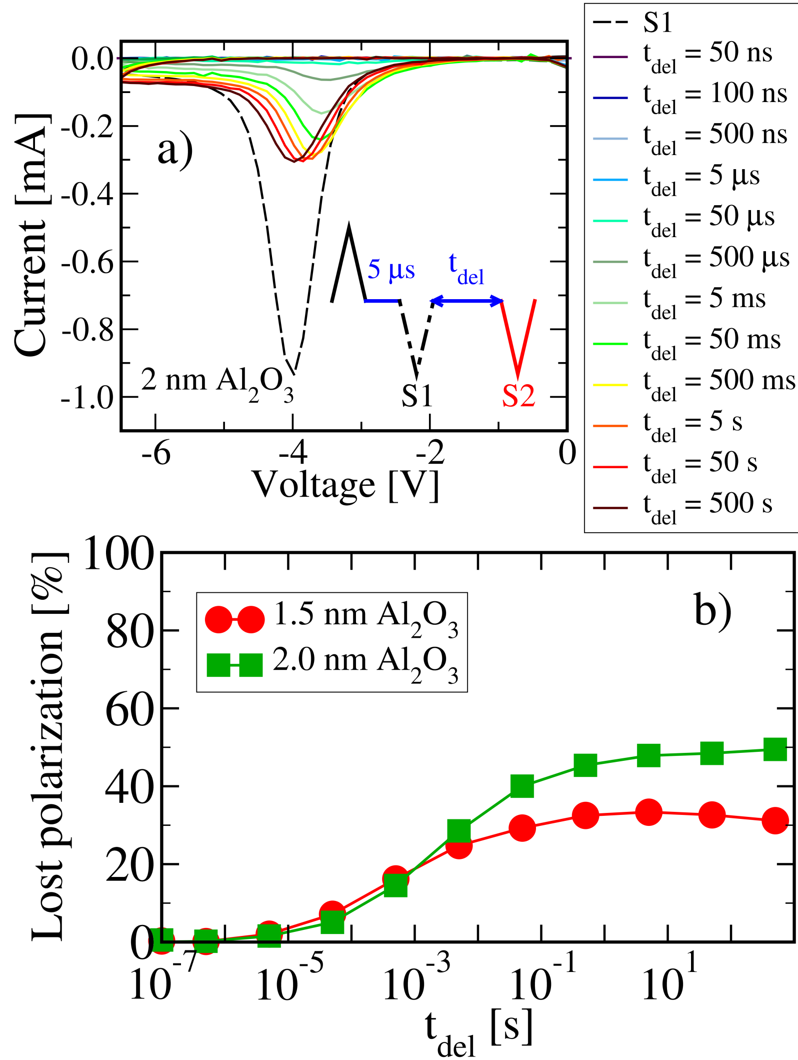

Figure 5.16: FTJ current-voltage characteristics measured at different delay times after a set operation S1 at T = 300 K (schematic of the measurement waveform in inset), showing P_r loss due to depolarization field and charge detrapping. b) Extracted percentage of remnant polarization lost during t_{delay} observed on a).

5.4 Optimal conditions for the synaptic operation of FTJs

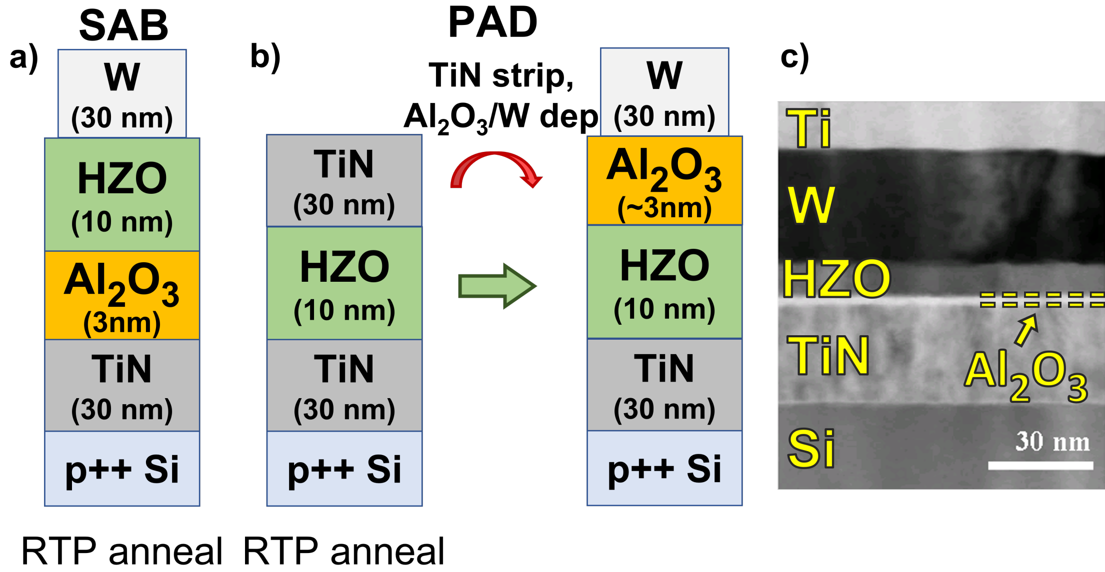

Figure 5.17: a), b) Sketches of the Stack Anneal–Bottom (SAB) FTJ and of the Post Anneal–Deposition (PAD) device, respectively. The main difference between SAB and PAD is the annealing process undergone by the Al_2O_3. Devices have a top electrode area of $95\times95\ \mu m^2$. c) Cross section TEM image of the SAB FTJ stack in this work.

As shown in the previous section, the operation of *MFIM FTJs* inherently relies on delicate trade–offs. In fact, the depolarization field enables a polarization-dependent tunneling read current, but it also tends to destabilize the polarization in retention and read mode. Moreover, in ferroelectric–dielectric stacks the charge injection and trapping substantially influence the stability of polarization and its switching dynamics [173], as it has also been recognized and debated for ferroelectric–based FETs [174–176].

In this last chapter, we exploit the information legacy of previous sections to understand and quantitatively investigate the trade–offs implied in the operation of FTJs as synaptic devices. The calibration of the simulator against experiments clarifies some crucial aspects of the device operation related to the charge trapping inside the *MFIM* stack. In particular, our results show that either a small or a very large trapping induce, respectively, negligible or complete compensation of the ferroelectric polarization, that hinder the FTJ operation in both cases. In this respect, we report optimal compensation conditions that can be exploited to optimize *FTJs* as synaptic devices with multiple conductance levels.

5.4.1 Device fabrication and experiments

Two types of *MFIM FTJs* stacks were fabricated with the same nominal thickness for FE and DE films, but through different process sequences. The sketch of the devices is reported in Fig. 5.17. The first stack named 'Stack Anneal–Bottom' [SAB, Fig. 5.17a)] was fabricated as follows: a 30 nm TiN layer was sputtered on *p++* Si substrate as the bottom electrode, then a 3 nm Al_2O_3 film followed by a 10 nm $Hf_{0.5}Zr_{0.5}O_2$ layer were deposited by atomic layer deposition at 250°C. Finally, a 30 nm W layer was deposited as top electrode by sputtering at room temperature. The

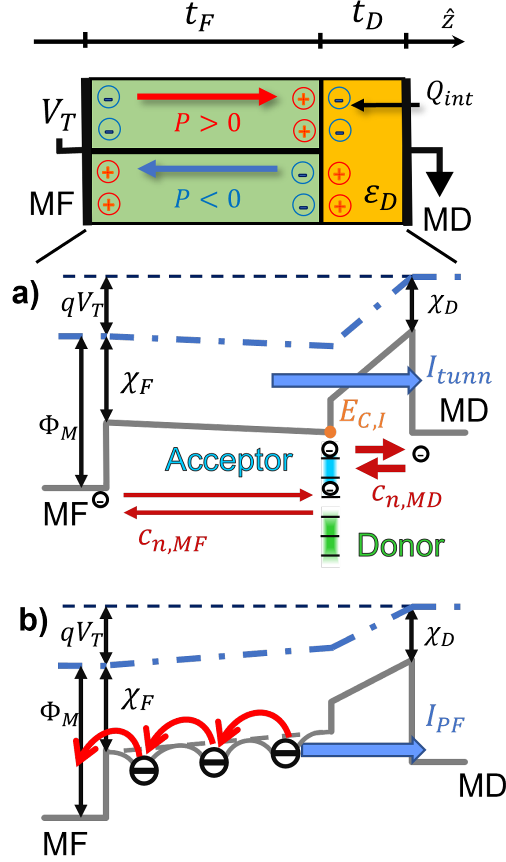

Figure 5.18: Examples for the band diagram across the FE–DE stack in read mode. Polarization is taken as positive when it points toward the DE–FE interface. V_T is the external applied voltage at the metal contacting the ferroelectric. Φ_M is the electrodes work function. $\chi_F = 2.4$ eV, $\chi_D = 1.6$ eV are the electron affinity of the HZO and Al_2O_3 layer, $E_{f,MD}$, $E_{f,MF}$ are the Fermi levels of the MD and MF electrodes. (a) The voltage drop V_D in the DE is such that the HZO conduction band minimum, $E_{C,I}$, at the FE–DE interface is smaller than the Fermi level at the MD electrodeI, thus enabling a tunnelling injection, I_{tunn}, limited by the dielectric; (b) A smaller V_D compared to results in (a) is expected in case of a Poole–Frenkel current, I_{PF}, in shallow HZO traps [184].

entire stack underwent a crystallization anneal by RTP at 400°C for 120 s in N_2 ambient. The top electrodes were patterned by Ti/Pt lift–off followed by W etch in H_2O_2. A TEM image of a SAB FTJ is shown in Fig. 5.17c).

The second FTJ named 'Post–Anneal Deposition' [PAD, Fig. 5.17b)] was fabricated as follows. After the deposition of the 30 nm bottom TiN electrode onto the $p{+}{+}$ Si substrate, a 10 nm HZO layer followed by a 30 nm TiN film were first deposited. The obtained TiN/HZO/TiN stack was RTP annealed at 400°C for 120 s in N_2 ambient. After the anneal, the top TiN layer was completely etched away and the 3 nm Al_2O_3 film was deposited at 250°C. The deposition and patterning of top W pads was then performed with the process described earlier for the SAB device. Note that, in the PAD FTJ case, the HZO was not in contact with the Al_2O_3 layer during annealing, while in the SAB stack the HZO and Al_2O_3 layers were annealed together. This is expected to lead to different HZO/Al_2O_3 interfaces in the two devices and, thus, to different chemical and electrical boundary conditions for the switching and stabilization of the HZO polarization. In particular, one may expect a significantly larger density of traps at the annealed HZO/Al_2O_3 interface of the SAB

stack compared to the PAD case. TiN/HZO/TiN capacitors with a 10 nm HZO layer were also fabricated as a reference.

After fabrication, SAB and PAD *FTJs* are woken–up through 2000 cycles of a triangular waveform at 1 kHz with 4.5 V amplitude. The number of wake–up cycles corresponds to what is needed to wake–up the reference TiN/HZO/TiN capacitors. Then, polarization–voltage (P–V) and current–volage (I–V) loops are measured by applying 1 ms long triangular pulses to the top electrode, with the substrate at ground potential. To avoid the breakdown of devices, the maximum voltage amplitudes used for SAB and PAD FTJs are 5 and 4.5 V, respectively.

5.4.2 Modelling framework and calibration

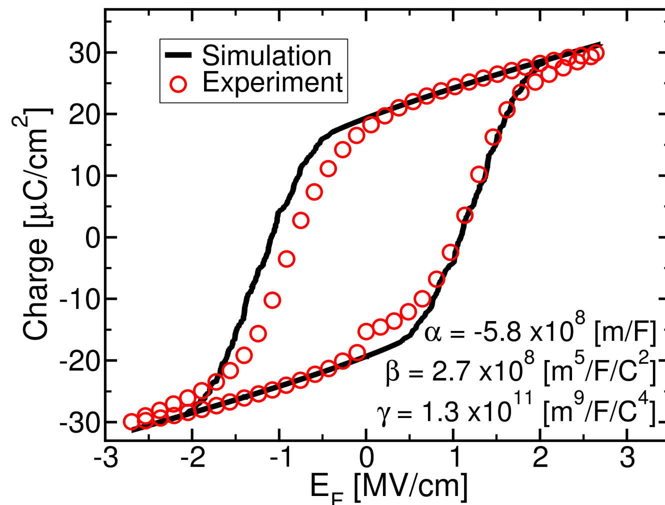

Figure 5.19: Simulated and experimental charge vs. electric field characteristic for a TiN/HZO/TiN device. Mean values of the gaussian distributed LGD constants are reported. The model reproduces well the experiment, validating also the HZO permittivity used in our calculations.

For the *FTJs* simulations, we used the in–house numerical model reported in Sec. 4.3, the trapping model of Sec.4.3.6, while the read current framework was developed in Sec. 5.1. In all simulations, we used $\rho = 100\ \Omega{\cdot}\mathrm{m}$ and $k \simeq 0$, consistently with the analysis in [148].

Concerning the read current, I_R, it is assumed to be dominated by the tunnelling across the Al_2O_3 layer, even if additional mechanisms assisted by defects are also possible [184]. In this respect, Fig. 5.18 qualitatively shows that a similar band bending across the Al_2O_3 layer is necessary for either a current dominated by tunneling (I_{tunn}, a) or by a Poole–Frenkel mechanism in shallow HZO traps (I_{PF}, b).

We calibrated the model against experiments. In Fig. 5.19 the simulations based on the LGD equations agree fairly well with the experimental charge versus electric field, E_F, curve for a TiN/HZO/TiN structure, except for the slight asymmetry in the measured P–V. This could be due to differences in top and bottom TiN–HZO interfaces, that are not considered in the simulations. Such a comparison allowed us to extract the LGD constants. In particular, in order to reproduce the experiments well, we considered domain–to–domain variations of the LGD parameters by assuming a normal distribution of α, β and γ. The mean values of such distributions are reported in Fig. 5.19, while the standard deviation of each constant are $\sigma_\alpha = 25\%$, $\sigma_\beta = 5\%$ and

$\sigma_\gamma = 8\%$. We will use these nominal values throughout this section, if not otherwise stated.

In all simulations, the work functions of W and TiN electrodes were taken as 4.5 eV, while for the tunneling mass and the permittivity we assumed $m_{FE} = 0.38 \cdot m_0$ and $\varepsilon_{FE} = 34 \cdot \varepsilon_0$ for the HZO, $m_{DE} = 0.15 \cdot m_0$ and $\varepsilon_{DE} = 10 \cdot \varepsilon_0$ for the Al_2O_3, with m_0 being the free electron mass and ε_0 the permittivity of vacuum. The values for these material parameters are empirically validated by the fairly good agreement between simulations and experiments.

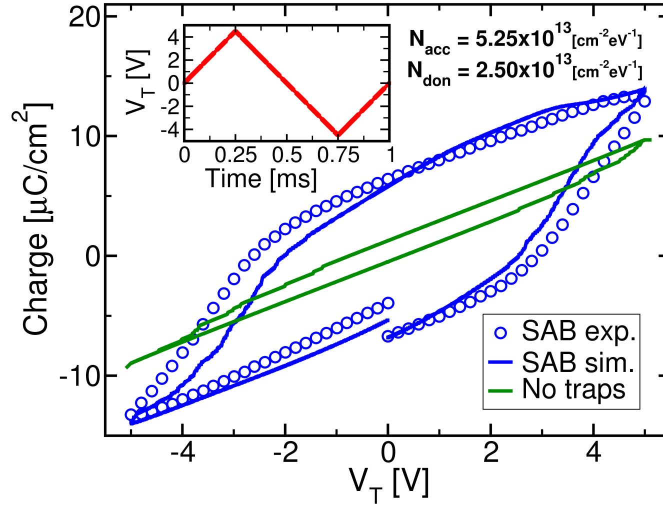

Figure 5.20: P–V characteristics for SAB FTJs (blue circles) measured by using the triangular waveform shown in the inset; Al_2O_3 thickness is $t_D = 3$ nm. Corresponding simulations for no trapped charge (green line) or by accounting for acceptor and donor traps (blue line) are shown.

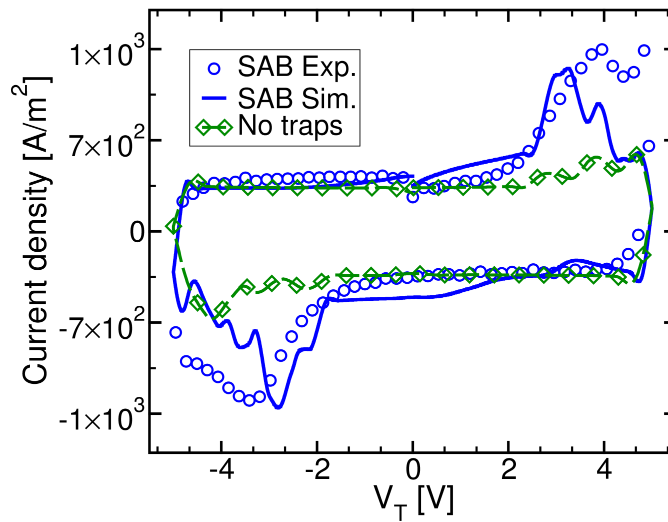

Figure 5.21: I–V curves corresponding to the P–V plots in Fig. 5.20. Trap densities are the same as in Fig. 5.20 and are summarized in Tab. 5.5.

5.4.3 Evidence of trap contribution in FTJ operation

Figure 5.20 compares simulations and experiments concerning the polarization–voltage (P–V) curves of the SAB devices. As it can be seen, consistently with Fig. 5.3 of Sec. 5.1 and Fig. 5.11 of Sec. 5.3, simulations neglecting charge injection and trapping (green line) exhibit a much more stretched P–

V curves compared to experiments. These simulation results are actually qualitatively consistent with experiments reported for an HZO capacitor serially connected to a discrete ceramic capacitor ensuring zero charge injection [155, 173]. Similarly, in Fig. 5.21, also current–voltage (I–V) simulations hardly show any switching current when trapping is neglected.

A fairly good agreement with P–V and I–V experiments can instead be achieved by including, in the simulations, an adequate charge density at the FE–DE interface (see blue lines in Figs. 5.20 and 5.21). This is a clear evidence of the contribution of charge trapping in the operation of *FTJs*, that is in line with recent results reported for FeFET devices [174–176]. It is worth mentioning that in Figs. 5.20, 5.21 and 5.22 the LGD constants had to be changed compared to Fig. 5.19, in order to increase the coercive field E_c, thus improving the agreement with experiments. This increase of the apparent E_c in *MFIM* compared to *MFM* structures has been previously reported and it has been ascribed to the division of external voltage between DE and FE layer. The capacitance of the ferroelectric increases as the field approaches E_c, causing an increased drop across the DE layer and thus higher apparent E_c than in *MFM* [216].

Table 5.5: Trapping cross sections $\sigma_{T,acc}$, $\sigma_{T,don}$, σ_E used in simulations throughout this section and trap densities used in the simulations of Fig. 5.20 (SAB) and Fig. 5.22 (PAD).

	N_{acc} $[cm^{-2}eV^{-1}]$	N_{don} $[cm^{-2}eV^{-1}]$	$\sigma_{T,acc}$ $[cm^2]$	$\sigma_{T,don}$ $[cm^2]$	σ_E $[eV]$
SAB	$5.25 \cdot 10^{13}$	$2.5 \cdot 10^{13}$	$3.5 \cdot 10^{-14}$	$8 \cdot 10^{-16}$	$7 \cdot 10^{-3}$
PAD	$1.12 \cdot 10^{13}$	$1 \cdot 10^{13}$	$7 \cdot 10^{-14}$	$7 \cdot 10^{-14}$	$7 \cdot 10^{-3}$

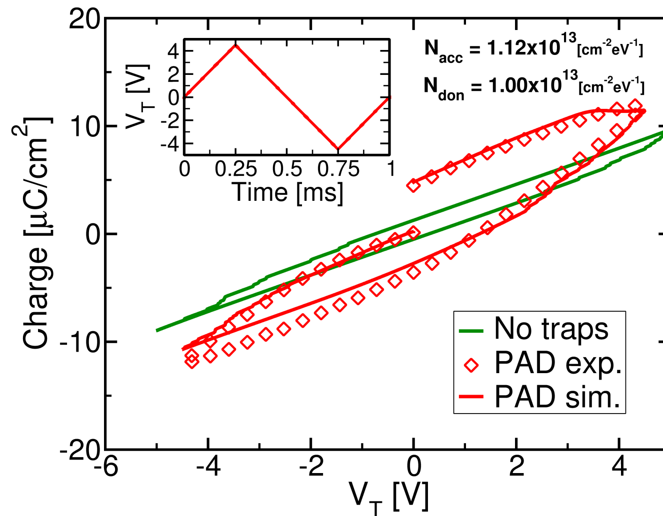

Figure 5.22: Comparison similar that in Fig. 5.20 between simulations and experiments for the P–V curves for PAD devices. Corresponding trap densities are reported in Tab. 5.5. Green line shows the no traps case, while red line is the simulation with traps included.

Figure 5.22 reports an analysis similar to that in Fig. 5.20 for the PAD devices. Even for the PAD *FTJs*, the simulations neglecting trapping (green line) exhibit large discrepancies with experiments (symbols). However, the trap density matching simulations with experiments is smaller than the concentration used for the SAB devices (see Tab. 5.5). It should be mentioned that not only the processing conditions but also the FTJ wake–up may influence the density of the electrically active defects. However, since both *FTJs* underwent the same wake–up sequence, we expect that this

difference is due to the different annealing conditions for the two devices. The discrepancy in the simulated interfacial charge in SAB and PAD samples is best illustrated by Fig. 5.23, reporting the average polarization and trapped charge $Q_{int} = (Q_{acc} + Q_{don})$ along a triangular V_T waveform. Figure 5.23 shows that the polarization is partly compensated by Q_{int} in both devices, but the effect is more prominent in SAB *FTJs*. Again, this is ascribed to the different annealing sequence between the two devices and Fig. 5.23 provides an insight about the role of the trap density at the HZO/Al_2O_3 interface in the FTJ operation.

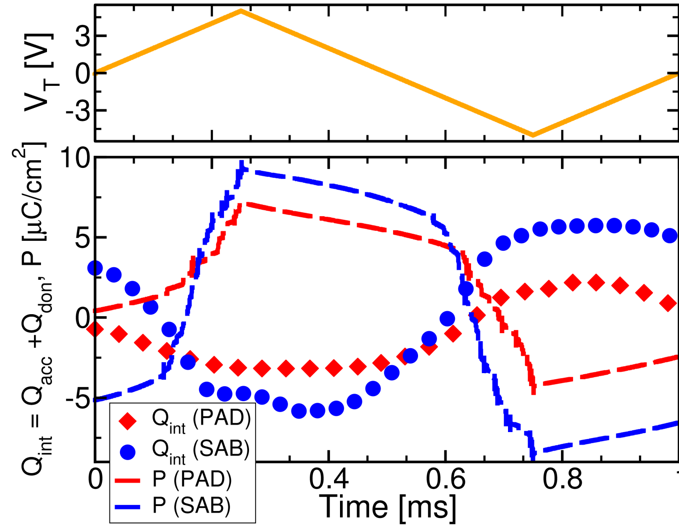

Figure 5.23: Simulated polarization (P, dashed lines) and trapped interface charge (Q_{int}, symbols) averaged across the device area for the SAB and PAD simulations in Figs. 5.20 and 5.22; the V_T waveform is shown as a yellow solid line. The used interfacial trap densities are reported in Tab. 5.5.

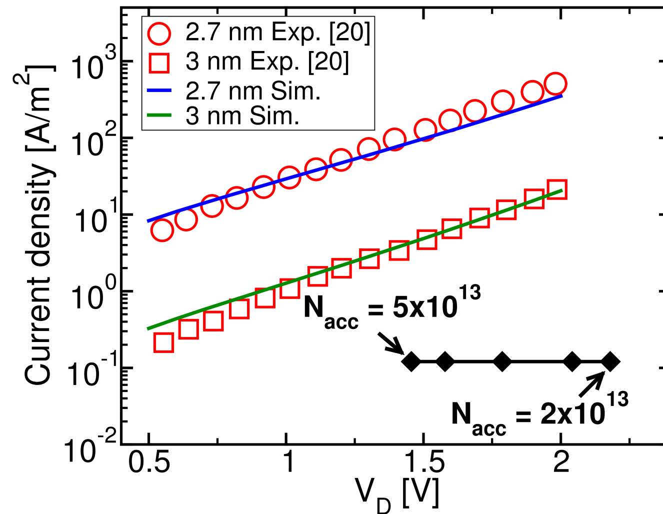

Figure 5.24: Measured current density for a SAB device at a read voltage $V_R = 2$ V (black diamonds) versus the corresponding voltage drop, V_D, across the 3 nm Al_2O_3 estimated by using simulations with different density, N_{acc} [$cm^{-2}eV^{-1}$], of acceptor traps. In these simulations, the density of donors does not affect the final result. Measured tunnelling current in thin SiO_2 layers (red symbols) and corresponding simulations with the tunnelling model of this work are reported as a reference.

In read mode, the compensation of the positive polarization is expected to reduce the band bending and thus the read current I_R (see Fig. 5.18). In this regard, Fig. 5.24 shows the measured

ON current density of about 0.12 A/m^2 (black diamonds) for SAB FTJs, together with the tunnelling current density in metal–SiO_2–metal (MIM) systems [217]; incidentally, our tunnelling model is in good agreement with experiments in the MIM systems. For the FTJ device, the voltage drop, V_D, across the Al_2O_3 layer is estimated by simulations and for different N_{acc}. Figure 5.24 shows that the estimated V_D for $N_{acc} \simeq 5 \cdot 10^{13}$ cm^{-2}eV^{-1} (see SAB parameters in Tab. 5.5) is clearly smaller than the value $V_D = (\Phi_M - \chi_F)/q = 2.1$ V required to have a read current I_R limited by the I_{tunn} through the Al_2O_3 film alone [see Fig. 5.18a)]. Consistently with this picture, the measured I_R is much smaller than the I_{tunn} in MIM systems, and it may be limited by Poole–Frenkel conduction [see I_{PF} in Fig. 5.18b)].

5.4.4 Polarization–compensation–aware design of the FTJ

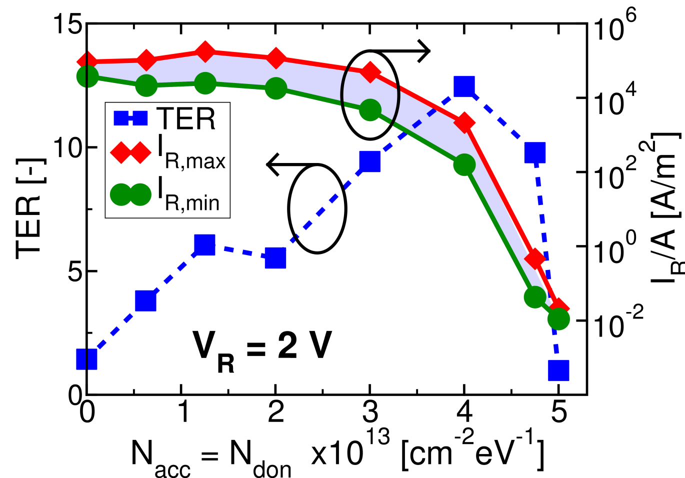

Figure 5.25: Minimum, $I_{R,min}$, and maximum, $I_{R,max}$, simulated read current (right y–axis) versus the trap density at read voltage $V_R = 2$ V. $I_{R,min}$ and $I_{R,max}$ correspond to a SET pulse voltage of $V_{SET,min} = 2.5$ V and $V_{SET,max} = 4.5$ V, respectively. The corresponding tunnel electroresistance, TER $= (I_{R,max} - I_{R,min})/I_{R,min}$ is reported (left y–axis).

In recent designs of neuromorphic processors, a reasonable target for the minimum read current is set to about 100 pA [102, 186], which requires a read current density of about $I_R/A \simeq 100$ A/m^2 for an FTJ area of $A \approx 1$ μm^2. Hereafter, we use our calibrated simulations to examine the optimal design of FTJs in terms of I_R and tunneling electroresistance TER $= (I_{R,max} - I_{R,min})/I_{R,min}$. For the simulations in this paragraph, the LGD constants are those in Fig. 5.19, calibrated on the *MFM* device, in order to precisely account for the characteristics of the integrated HZO layer.

Figure 5.25 reports the simulated minimum and maximum read current densities and TER for a FTJ having an Al_2O_3 thickness scaled down to $t_D = 2.5$ nm and for different trap densities $N_{acc} = N_{don}$[1]. Despite the uncertainties that admittedly affect our calculations of the read current and have been also discussed in Fig. 5.24, the I_R and the TER values calculated at large trap densities are in–line with recent literature on SoA FTJ devices, showing read currents in the 0.05–0.6 A/m^2 range and TER values between 5 and 12.5 [141, 202, 218].

Quite interestingly, Figure 5.25 shows that the TER exhibits an optimal value for N_{acc} around

[1]This simplifying assumption of an equal density of acceptor and donor traps has been introduced in order to limit the number of free parameters in the analysis carried out in this section. We expect that the main outcome of this discussion does not change if N_{acc} and N_{don} are allowed to be different.

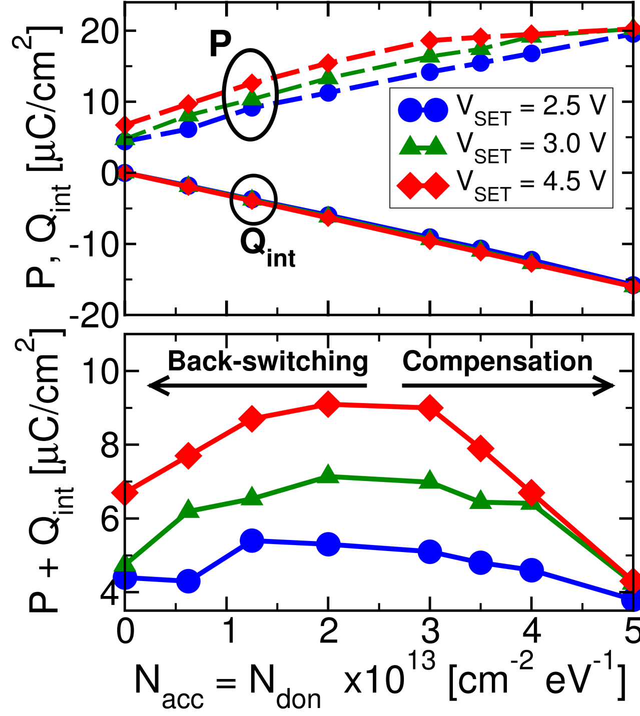

Figure 5.26: Average polarization, P, interface charge, Q_{int}, and net or compensated polarization ($P+Q_{int}$) vs. the trap density in read condition ($V_R = 2$ V) and for different SET voltages. At low N_{acc}, ($P + Q_{int}$) degrades due to depolarization field and back–switching, while at large N_{acc}, ($P + Q_{int}$) decreases due to a large P compensation.

$4 \cdot 10^{13}$ cm^{-2}eV^{-1}, which stems from the behavior of the net or compensated polarization ($P+Q_{int}$) illustrated in Fig. 5.26. In fact, at low N_{acc} values, the compensation of the positive P is too weak, so that the depolarization field E_F increases and destabilizes the polarization. In this respect, we calculated the difference between the number of domains with positive spontaneous polarization P at the maximum value of the SET voltage and at zero external bias (i.e. during retention) and then we divided this value by the overall number of domains. This is indeed the fraction of the domains set to positive P during the SET pulse that back–switch to negative P during retention. We defined such value as back—switching in Fig. 5.27b), which shows that at low N_{acc} a significant back–switching of the ferroelectric domains occurs. The large back—switching at low N_{acc} deteriorates the TER in Fig. 5.25.

At large N_{acc}, instead, there is an excessive compensation that reduces ($P + Q_{int}$). This implies an increase of the HZO conduction band minimum, $E_{C,I}$, at the FE–DE interface during the read mode, which is reported in Fig. 5.27a). Such an $E_{C,I}$ raise leads to the large I_R drop at high N_{acc} shown in Fig. 5.25. Moreover, a large compensation tends to merge the states obtained by different V_{SET} values, which seriously hinders a multi–level read current and thus the operation as a synaptic device.

The results of Fig. 5.25 and the inspection of the internal quantities performed in Figs. 5.26 and 5.27, clearly show that an optimal compensation condition exists for the operation of *FTJs*. In this respect, as discussed above, both negligible and excessive charge–trapping–induced compensation of the ferroelectric polarization are detrimental for the operation of the *FTJs* under study. Indeed, there exists a quite delicate balance between the suppression of back–switching and *TER* values

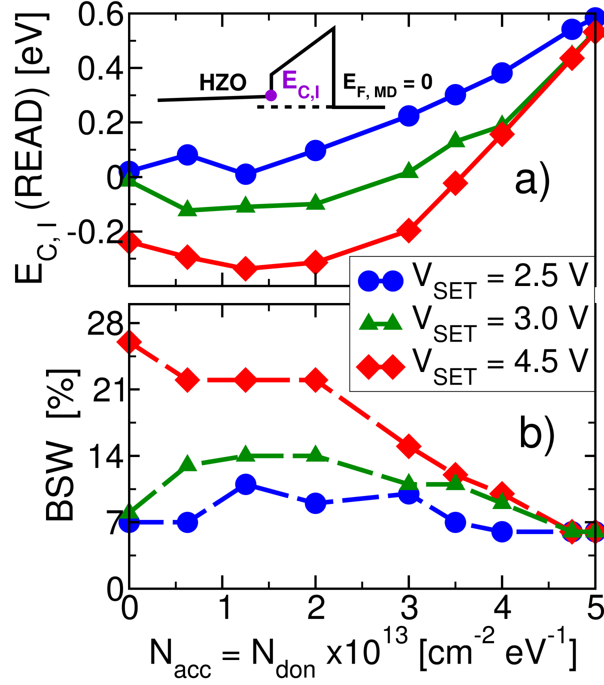

Figure 5.27: a) HZO conduction band minimum, $E_{C,I}$, in read mode versus the trap density, where the Fermi level at the MD electrode is $E_{F,MD} = 0$ eV (see inset). b) Ferroelectric domain back–switching, BSW, defined as the percentage of domains having a negative P in retention after being set to a positive P during the SET pulse.

enabling a multi–level operation in read mode. In Fig. 5.28, for a given FTJ structure, we leverage the optimal design condition in Fig. 5.25 and show that eight current levels can be placed in the memory window of the FTJ for the optimal compensation condition corresponding to $N_{acc} = 4 \cdot 10^{13}$ cm^{-2}eV^{-1}, thus enabling a 3–bit synaptic weight resolution. It is worth mentioning that such an optimal condition corresponds to a V_D across the Al_2O_3 layer that is very close to or larger than the value $(\Phi_M - \chi_F)/q = 2.1$ V required to have tunnelling through the thin dielectric film alone [see Figs. 5.18a) and 5.27]. A comparison of the calculated I_R with experiments suggests that it is probably difficult in actual *FTJs* to reach this favorable condition where I_R is only limited by the tunnelling through the Al_2O_3 layer, as also discussed in Fig. 5.24.

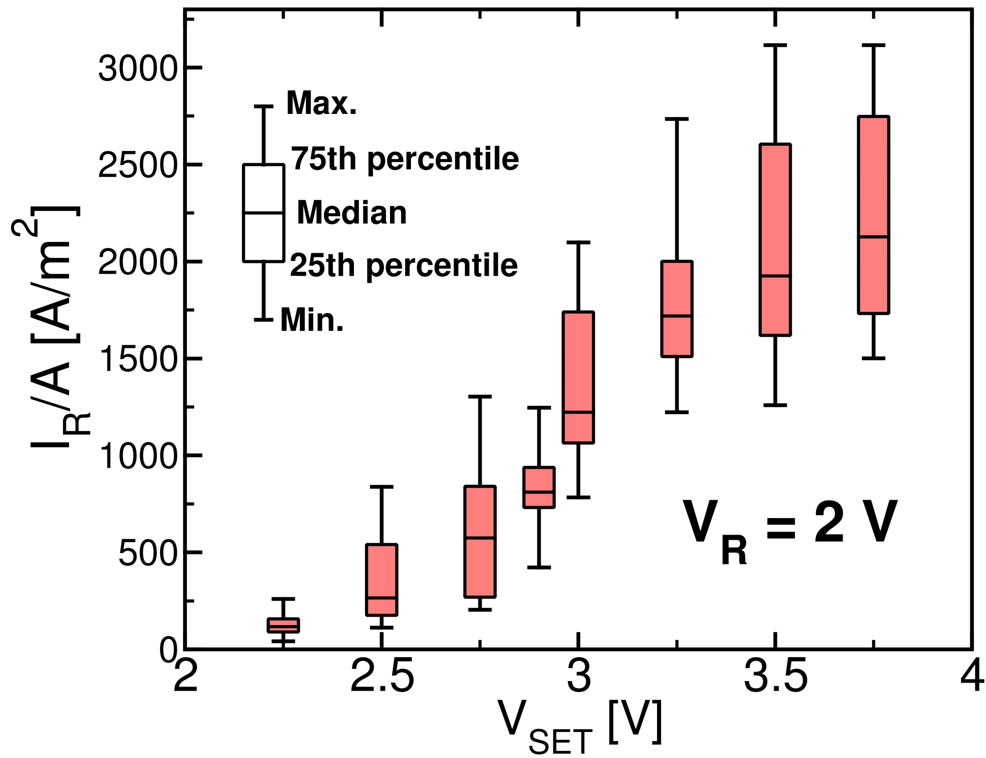

Figure 5.28: Simulation results for the FTJ of Fig. 5.25 under optimal compensation conditions corresponding to $N_{acc} = 4 \cdot 10^{13}$ cm^{-2}eV^{-1}. The box plots reveal overlapping distributions for the current levels for small area FTJs with $n_D = 100$ ferroelectric domains. Simulations have been performed for a small device area of 2500 nm^2 (corresponding to a domain size $d = 5$ nm).

Chapter 6

Conclusions

The project of this Ph.D. has focused on the development of energy reduction methodologies for information technology and, in particular, for *AI* applications. In this thesis, we report a top-down overview of neuromorphic computing solutions, suitable to create novel and more efficient computational paradigms compared to von-Neumann-based *AI* computational platforms. In this perspective, we first exploited a learning strategy that leverages the exact computation of the gradient in *SNNs*. We observed that a gradient descent strategy alone is not sufficient to obtain a small spike rate and good accuracy in deep *SNNs*, because a challenging trade-off between the frequency of spikes and the minimization of the loss function emerges during the training. Acting on the spike rate only, in fact, can cause many neurons to be silent, hence resulting in a gradient vanishing problem. On the other hand, focusing only on the network accuracy almost invariably results in a large spike rate. The neural over-excitation issue gains more relevance as the network becomes deeper or the number of neurons becomes larger, hence we proposed a learning strategy composed of two phases. The first phase uses an "augmented" loss function (L_A) that is built to avoid silent neurons. While the minimization of L_A does not converge to the minimum of the real metric of interest (the cross-entropy), it tends to converge to a point that is close to such a minimum and it is thus an effective starting point for the second training phase. Here, a "correct" loss (L) is used that still pursues also a minimization of the spike rate, thus leading to the convergence of the network towards the minimum of the cross-entropy, while maintaining at the same time a small average spike rate. Even though we applied this methodology to a relatively simple network that takes advantage of the computational efficiency of Leaky Integrate and Fire neurons, the mathematical framework can be exploited for different tasks, network topologies, or neural implementations. It would be interesting, in fact, to apply the combination of the two-phase learning and the Event-Prop back-propagation algorithm to different applications, such as Autoencoders. These networks are designed to shrink information from a large set of inputs/signals to a more compact and dense representation. These networks are used in many fields as data aggregators, transformers, or anomaly detectors [219]. Moreover, spikes-based Autoencoders can take advantage of the intrinsic duality of space-time representation of the data as spikes. Therefore, differently from normal Autoencoders based on classical neural networks, spike-based Autoencoders can exploit time to represent information and reduce the overhead of the area needed to acquire and process data. In this conversion process, however, the minimum average spike rate needed to represent information in the internal layer limits the performance of the network in terms of energy

consumption. Consequently, the two-phase learning introduced in this thesis can be applied not only to improve the efficacy of the learning process, but also to find out an optimum in terms of reconstruction capability and power efficiency. As a proof of concept, we reported in Fig. 6.1 a sketch of a transformer Autoencoder made by a 5 spiking layers network. In this simple example, the network takes as input an MNIST image, shrinks the input data with an encoder network to a hidden internal state of 60 neurons, and then expands the internal representation with a decoder network to recreate the input image. The input/output encoding and learning methodologies are compatible with the one reported in chapter 2. As can be observed, after training the network, the internal representation of the input data shows a prominent neural activity that must be taken under control in order to limit the power consumption of the system (each point in the graph depicts a spike emitted by a neuron of the middle layer). The one reported in Fig. 6.1 is just an example of a possible reconstruction application, and the details of this implementation are not reported to keep the focus on the working principle and approach followed in this thesis. Moreover, among several open questions on the efficacy of the Event-Prop algorithm, the behavioral relationship between networks implemented with different neural models and trained with the Event-Prop algorithm remains to be explored. Nonetheless, a step ahead in this direction has been reported in this thesis in Appendix B, where a full set of equations extends the training algorithm Event-Prop to the Izhikevich neural model. Therefore, a direct comparison between *SNNs* based on the Izhikevic or the LIF models trained with Event-Prop can be performed. Moreover, the two–phase learning can be considered during the training of those different implementations of neural networks.

After a deep analysis of artificial neural behavior, we then focused on another fundamental building block of neuromorphic computing: artificial synapses. In fact, as already mentioned, neuromorphic computing aims to change the computational paradigm not just by the use of a more biologically plausible neural activation, but also with the implementation of *SNNs* in *VLSI* circuits. In this respect, memristive memories are a promising option to implement artificial synapses in the *BEOL* stacked in cross-bar arrays. Among different memristive technologies reported in the literature, we focused on *FTJs*. Hence, we first introduced the working principle of *FTJs* and the physical modeling of the *MFIM* structure. After that, we calibrated our model against experiments. Thanks to simulations, we observed delicate tradeoffs between the reading current modulation and the depolarization field that hampers the retention of *FTJs* based synaptic devices. Then, we investigated the behavior of *FTJs* in the presence of charge trapping. As for the detrimental effects of the depolarization field, we demonstrated that the contribution of charge trapping can be beneficial for the stabilization of the polarization during retention. However, the trapped charge may reduce the read current dynamics, thus limiting the benefits of an optimized design for ferroelectric tunneling junctions. From this perspective, we thoroughly studied the switching dynamics of the *MFIM* ferroelectric capacitors and we observed a strong dependence on the injection and trapping of charge into the dielectric stack, which compensates the ferroelectric polarization to a large extent. We argue that the understanding and control of such a charge compensation is crucial for the design of *FTJs* based on an *MFIMs* structure. Therefore to better understand the interplay between ferroelectric polarization and charge trapping, in Sec. 5.4, we presented a joint effort between numerical modeling and experiments. We identified and physically explained the optimal compensation condition for a robust operation of *FTJs*. Even if tailoring trap densities may be challenging from a technological standpoint, we experimentally demonstrated that appropriate processing steps and annealing

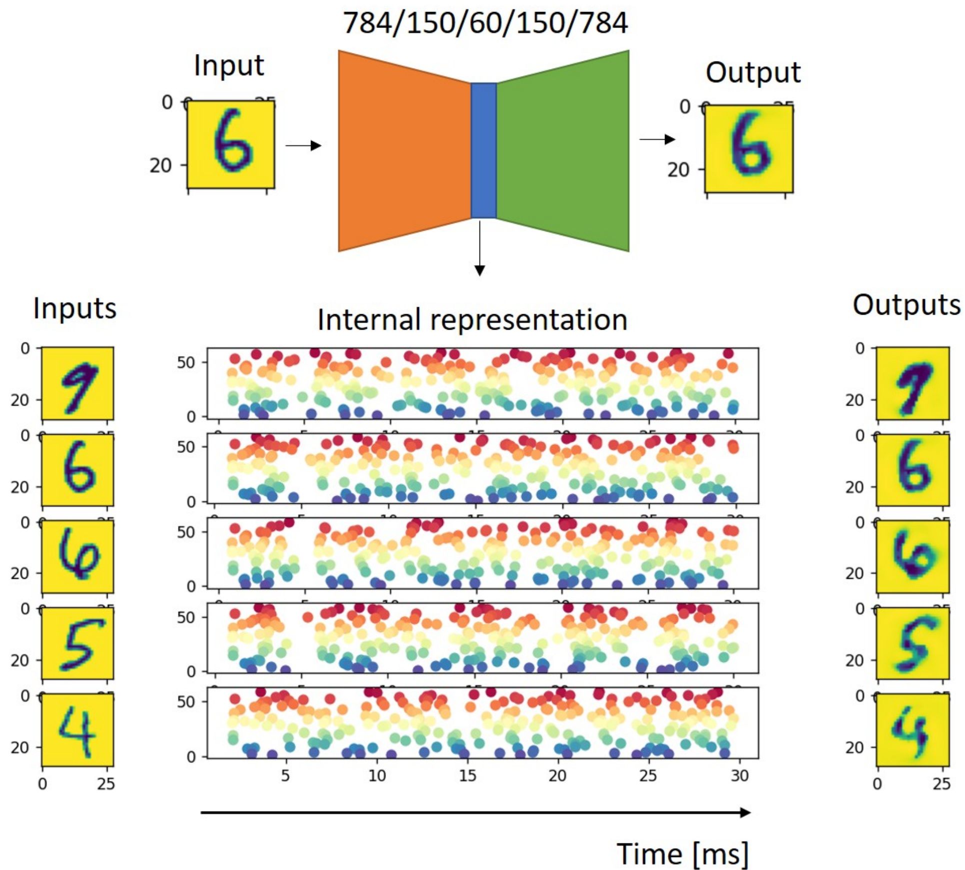

Figure 6.1: Example of Spike-based Autoencoder which generates a lossy version of the MNIST image in input. In particular, the Autoencoder is formed by 5 layers (784/150/60/150/784) of fully connected LIF spiking neurons. In the middle of the picture, 5 examples of image reconstructions with the corresponding central layer activity are reported. In particular, the five graphs depict, in time, the neural spike activity of the internal neurons with a dot. Neurons are shown in different colors.

conditions can lead to different trap densities at the *FE-DE* interface of actual *FTJs*. Moreover, the trap density can be partly controlled in ALD deposited Al_2O_3 films by tuning the deposition parameters, such as the precursor pulse duration, the oxidant precursor type (water, O_2 plasma, ozone), the deposition temperature, and the dosing time. These processing parameters have been shown to change the Al_2O_3 defect density at least by an order of magnitude [220–222]. Another pathway is to include an ultrathin charge trapping layer between the tunneling dielectric and HZO [223]. As a final remark, the insertion of an ultra–thin metal layer at the *FE-DE* interface may also be an interesting design option to control the compensation of the ferroelectric polarization in *FTJs*, as it has been suggested in [160].

In the pursuit of a good matching between simulations and experiments, a simplified analysis helped to understand the main building blocks of the theory behind ferroelectricity. Therefore, some approximations have been taken into account to reduce the number of fitting parameters and degrees of uncertainty. This means future developments of this work should discuss and verify the validity of our analysis, taking also into account an improved model validated on more experimental conditions compared to the one reported in this manuscript. First of all, we neglected the dependency between the electric field and the intrinsic resistivity of the ferroelectric material, which regulates the switching dynamics of the ferroelectric domains (ρ parameter). However, a deeper understanding of this parameter can improve simulation results with different external slew–rate conditions. In fact, the interplay between domain switching and charge traps is still not fully understood, and it requires further analysis with different experimental conditions to derive a set of equations that satisfies the whole landscape of possibilities: the presence of charges trapped at the interface between *FE-DE* can modify the internal field of the structure and, consequently, the switching speed of the ferroelectric domains. On the other hand, the dynamics of the domains may influence the band diagram of the structure, thus changing the transmission coefficients linked with tunneling fluxes of carriers between metals and the *FE-DE* interface.

Moreover, through this manuscript we assumed direct tunneling as the principal transport mechanism. However, as discussed in chapter 5.4, different transport mechanisms can play a prominent role in the charge trapping and the tunneling current of FTJs. Therefore, a direct comparison between transport mechanisms (such as direct tunneling, indirect tunneling, Pool-Frenkel etc.) supported by experiments can shed light on this topic and help the research community to improve further the *FTJs* usability and reliability, allowing the creation of even more energy efficient and intelligent technological devices.

Appendix A

Proof of Event-Prop backpropagation algorithm using adjoint variables in LIF Spiking Neural Networks

The equations reported below describe the free dynamic of a simple model for *LIF* spiking neurons:

$$\begin{cases} \tau_m \frac{d\mathbf{V}}{dt} = -\mathbf{V} + \Phi(\mathbf{D}) R\mathbf{I} \\ \tau_s \frac{d\mathbf{I}}{dt} = -\mathbf{I} \\ \tau_d \frac{d\mathbf{D}}{dt} = -\mathbf{D} \end{cases} \tag{A.1}$$

then we can define implicit version of A.1:

$$\mathbf{f_V} := \tau_m \dot{\mathbf{V}} + \mathbf{V} - R\mathbf{I} \tag{A.2}$$

$$\mathbf{f_I} := \tau_s \dot{\mathbf{I}} + \mathbf{I} \tag{A.3}$$

$$\mathbf{f_D} := \tau_d \dot{\mathbf{D}} + \mathbf{D} \tag{A.4}$$

where $\mathbf{V}, \mathbf{I}, , \mathbf{D}, \mathbf{f_V}, \mathbf{f_I}, \mathbf{f_D} \in \Re^{N_n}$ and N_n is the number of neurons in the network. Moreover the transition condition of the the state variables reported below for a spiking neuron (denoted with $n(k)$) and a non spiking neuron (denoted with m) are:

$$\left(\mathbf{V}^+\right)_{n(k)} = \gamma \tag{A.5}$$

$$\left(\mathbf{V}^+\right)_m = \left(\mathbf{V}^-\right)_m \tag{A.6}$$

$$\left(\mathbf{I}^+\right)_{n(k)} = G\left(\mathbf{I}^-\right)_{n(k)} \tag{A.7}$$

$$\left(\mathbf{I}^+\right)_m = \left(\mathbf{I}^-\right)_m + \beta \mathbf{W_{m,n}} \tag{A.8}$$

$$\left(\mathbf{D}^+\right)_{n(k)} = \epsilon \tag{A.9}$$

$$\left(\mathbf{D}^{+}\right)_{m} = \left(\mathbf{D}^{-}\right)_{m} \tag{A.10}$$

While the threshold condition for spiking is defined as:

$$\left(\mathbf{V}^{-}\right)_{n(k)} = \theta \tag{A.11}$$

The cost function has been defined as follows:

$$\mathscr{L} = l_p(t^{post}) + \int_T l_V\left(t^{post}, \mathbf{V}\right) dt^{post} \tag{A.12}$$

and, in general, the integral can be split between spikes events because $l_V\left(t^{post}, \mathbf{V}\right)$ is always defined and limited in time:

$$\mathscr{L} = l_p(t^{post}) + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} l_V\left(t^{post}, \mathbf{V}\right)\ dt^{post} \tag{A.13}$$

where $l_V\left(t^{post}, \mathbf{V}\right) : \Re^{N_n} \to \Re$. Moreover, the sum over k starts from 0, consequently, t_0 is the beginning of the simulation and, in general, is equal to 0 s. The time gap $t_0 - t_1$ is the inactivity period of the network where no spike has been emitted by the network.

In order to calculate the gradient of the loss function, we need to add the adjoint variables $\lambda_{\mathbf{V}}$, $\lambda_{\mathbf{I}}$ and $\lambda_{\mathbf{D}}$ to A.13 :

$$\mathscr{L} = l_p(t^{post}) + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} l_V\left(t^{post}, \mathbf{V}\right) + \lambda_{\mathbf{V}} \cdot \mathbf{f_V} + \lambda_{\mathbf{I}} \cdot \mathbf{f_I} + \lambda_{\mathbf{D}} \cdot \mathbf{f_D}\ dt^{post} \tag{A.14}$$

Therefore, by differentiating the payoff function we obtain:

$$\begin{aligned}\frac{d\mathscr{L}}{dW_{i,j}} =& \frac{dl_p(t^{post})}{dW_{i,j}} + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} dt^{post} \frac{dl_V\left(t_k^{post}, \mathbf{V}\right)}{dW_{i,j}} + \\ &+ \frac{d\lambda_{\mathbf{V}} \cdot \mathbf{f_V}}{dW_{i,j}} + \frac{d\lambda_{\mathbf{I}} \cdot \mathbf{f_I}}{dW_{i,j}} + \frac{d\lambda_{\mathbf{D}} \cdot \mathbf{f_D}}{dW_{i,j}}\end{aligned} \tag{A.15}$$

and because the adjoint functions don't depend on the weights:

$$\begin{aligned}\frac{d\mathscr{L}}{dW_{i,j}} =& \frac{dl_p(t^{post})}{dW_{i,j}} + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} dt^{post} \frac{dl_V\left(t^{post}, \mathbf{V}\right)}{dW_{i,j}} + \\ &+ \lambda_{\mathbf{V}} \cdot \frac{d\mathbf{f_V}}{dW_{i,j}} + \lambda_{\mathbf{I}} \cdot \frac{d\mathbf{f_I}}{dW_{i,j}} + \lambda_{\mathbf{D}} \cdot \frac{d\mathbf{f_D}}{dW_{i,j}}\end{aligned} \tag{A.16}$$

Now each term should be expanded. Let's begin with the loss functions terms $l_V\left(t^{post}, \mathbf{V}\right)$ and $l_p(t^{post})$. These loss functions depend indirectly from the weights of networks, therefore, we can compute the total derivative with the chain rule:

$$\frac{dl_V\left(t_k^{post}, \mathbf{V}\right)}{dW_{i,j}} = \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} + \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \frac{\partial \mathbf{V}}{\partial W_{i,j}} \tag{A.17}$$

$$\frac{dl_p(t^{post})}{dW_{i,j}} = \sum_{k=0}^{N_{post}} \frac{\partial l_p(t_k^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \tag{A.18}$$

Then, taking into account A.2 and A.3:

$$\lambda_{\mathbf{V}} \cdot \frac{d\mathbf{f_V}}{dW_{i,j}} = \lambda_{\mathbf{V}} \cdot \left(\tau_m \frac{d\dot{\mathbf{V}}}{dW_{i,j}} + \frac{d\mathbf{V}}{dW_{i,j}} - \Phi(\mathbf{D}) R \frac{d\mathbf{I}}{dW_{i,j}} \right) \tag{A.19}$$

$$\lambda_{\mathbf{I}} \cdot \frac{d\mathbf{f_I}}{dW_{i,j}} = \lambda_{\mathbf{I}} \cdot \left(\tau_s \frac{d\dot{\mathbf{I}}}{dW_{i,j}} + \frac{d\mathbf{I}}{dW_{i,j}} \right) \tag{A.20}$$

$$\lambda_{\mathbf{D}} \cdot \frac{d\mathbf{f_D}}{dW_{i,j}} = \lambda_{\mathbf{D}} \cdot \left(\tau_D \frac{d\dot{\mathbf{D}}}{dW_{i,j}} + \frac{d\mathbf{D}}{dW_{i,j}} \right) \tag{A.21}$$

Because the weights of the network are fixed in time, and in particular between two spikes events (also due to Gronwall's theorem) we can exchange the position of the differentials operators:

$$\frac{d}{dW_{i,j}} \left[\frac{d}{dt} \right] = \frac{d}{dt} \left[\frac{d}{dW_{i,j}} \right]$$

Substituting then Eqs. A.17, A.18, A.19, A.20 and A.21 in eq. A.16, we obtain:

$$\begin{aligned} \frac{d\mathscr{L}}{dW_{i,j}} = & \sum_{k=0}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \\ & + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} dt^{post} \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} + \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \frac{\partial \mathbf{V}}{\partial W_{i,j}} \\ & \qquad + \lambda_{\mathbf{V}} \cdot \left(\tau_m \frac{d\dot{\mathbf{V}}}{dW_{i,j}} + \frac{d\mathbf{V}}{dW_{i,j}} - \Phi(\mathbf{D}) R \frac{d\mathbf{I}}{dW_{i,j}} \right) \\ & \qquad + \lambda_{\mathbf{I}} \cdot \left(\tau_s \frac{d\dot{\mathbf{I}}}{dW_{i,j}} + \frac{d\mathbf{I}}{dW_{i,j}} \right) + \lambda_{\mathbf{D}} \cdot \left(\tau_D \frac{d\dot{\mathbf{D}}}{dW_{i,j}} + \frac{d\mathbf{D}}{dW_{i,j}} \right) \end{aligned} \tag{A.22}$$

Then, we need to integrate by parts:

$$\begin{aligned} & \int_{t_k^{post}}^{t_{k+1}^{post}} \frac{\partial l_V\left(t^{post}, \mathbf{V}\right)}{\partial t^{post}} \frac{\partial t}{\partial W_{i,j}} \, dt^{post} = \\ & \qquad = \left[l_V\left(t^{post}, \mathbf{V}\right) \frac{\partial t}{\partial W_{i,j}} \right]_{t_k}^{t_{k+1}} - \int_{t_k^{post}}^{t_{k+1}^{post}} l_V\left(t^{post}, \mathbf{V}\right) \underbrace{\frac{\partial}{\partial W_{i,j}} \left[\underbrace{\frac{\partial t_k}{\partial t}}_{=1} \right]}_{=0} dt^{post} \\ & \qquad = l_V\left(t_{k+1}^{post}, \mathbf{V}\right) \frac{\partial t_{k+1}}{\partial W_{i,j}} - l_V\left(t_k^{post}, \mathbf{V}\right) \frac{\partial t_k}{\partial W_{i,j}} \end{aligned} \tag{A.23}$$

$$\int_{t_k^{post}}^{t_{k+1}^{post}} \tau_m \lambda_{\mathbf{V}} \cdot \underbrace{\frac{\partial \dot{\mathbf{V}}}{\partial W_{i,j}}}_{=\frac{d}{dt}\left[\frac{\partial \mathbf{V}}{\partial W_{i,j}}\right]} dt^{post} = \left[\tau_m \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\right]_{t_k}^{t_{k+1}} - \tau_m \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda_{\mathbf{V}}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\, dt^{post} =$$

$$= \tau_m \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_m \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\bigg|_{t_k} - \tau_m \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda_{\mathbf{V}}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\, dt^{post} \tag{A.24}$$

The same happens for I and D:

$$\int_{t_k^{post}}^{t_{k+1}^{post}} \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \dot{\mathbf{I}}}{\partial W_{i,j}}\, dt^{post} = \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_k} - \tau_s \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda_{\mathbf{I}}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\, dt^{post} \tag{A.25}$$

$$\int_{t_k^{post}}^{t_{k+1}^{post}} \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \dot{\mathbf{D}}}{\partial W_{i,j}}\, dt^{post} = \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_k} - \tau_d \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda_{\mathbf{D}}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\, dt^{post} \tag{A.26}$$

Combining everything in A.22, we obtain:

$$\frac{d\mathscr{L}}{dW_{i,j}} = A + B + C \tag{A.27}$$

where A,B and C are respectively:

$$A = \sum_{k=0}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \tag{A.28}$$

$$\begin{aligned} B = \sum_{k=0}^{N_{post}} l_V\left(t_{k+1}^{post}, \mathbf{V}\right) \frac{\partial t_{k+1}^{post}}{\partial W_{i,j}} - l_V\left(t_k^{post}, \mathbf{V}\right) \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\ + \tau_m \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_m \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\bigg|_{t_k} + \\ + \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_k} + \\ + \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_k} \end{aligned} \tag{A.29}$$

$$C = \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} dt^{post} \left[-\tau_m \dot{\lambda_{\mathbf{V}}} + \lambda_{\mathbf{V}} + \frac{\partial l_V\left(t^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \right] \frac{\partial \mathbf{V}}{\partial W_{i,j}} + \left[-\tau_s \dot{\lambda_{\mathbf{I}}} + \lambda_{\mathbf{I}} - \Phi(\mathbf{D}) R \lambda_{\mathbf{I}} \right] \frac{\partial \mathbf{I}}{\partial W_{i,j}} + \left[-\tau_d \dot{\lambda_{\mathbf{D}}} + \lambda_{\mathbf{D}} \right] \frac{\partial \mathbf{D}}{\partial W_{i,j}} \tag{A.30}$$

In particular, C becomes = 0 when the terms between square brakets are imposed = 0, defining in this way the dynamic response for the adjoint functions:

$$\tau_m \frac{d\lambda_{\mathbf{V}}}{dt} = \lambda_{\mathbf{V}} + \frac{\partial l_V\left(t^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \tag{A.31}$$

$$\tau_s \frac{d\lambda_{\mathbf{I}}}{dt} = \lambda_{\mathbf{I}} - \Phi(\mathbf{D}) R \lambda_{\mathbf{V}} \tag{A.32}$$

$$\tau_d \frac{d\lambda_{\mathbf{D}}}{dt} = \lambda_{\mathbf{D}} \tag{A.33}$$

The loss function $l_p(t^{post})$ is null in the time interval $[t_0; t_1]$ (the inactivity period between the beginning of the simulation and the first spike time emitted by the network). Consequently, all its derivative in that time gap are null. Therefore the A term could be reduced as:

$$A = \sum_{k=1}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \tag{A.34}$$

Now, B presents a numerical series that could be group as:

$$\sum_{k=0}^{N_{post}} a_{k+1}^- - a_k^+ = a^-\big|_1 - a^+\big|_0 + a^-\big|_2 - a^+\big|_1 + ... + \underbrace{a^-\big|_{N_{post}+1}}_{=0} - a^+\big|_{N_{post}} = = - a^+\big|_0 + \sum_{k=1}^{N_{post}} \left[a^- - a^+\right]\big|_{t_k} \tag{A.35}$$

Therefore, applying this formalism we obtain:

$$\begin{aligned} B = &- l_V^+\left(t_0^{post}, \mathbf{V}\right) \frac{\partial t_0^{post}}{\partial W_{i,j}} + \sum_{k=1}^{N_{post}} \left[l_V^-\left(t_k^{post}, \mathbf{V}\right) - l_V^+\left(t_k^{post}, \mathbf{V}\right) \right] \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\ &- \tau_m \left[\lambda_{\mathbf{V}}^+ \cdot \frac{\partial \mathbf{V}^+}{\partial W_{i,j}} \right]_{k=0} + \tau_m \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{V}}^- \cdot \frac{\partial \mathbf{V}^-}{\partial W_{i,j}} - \lambda_{\mathbf{V}}^+ \cdot \frac{\partial \mathbf{V}^+}{\partial W_{i,j}} \right]_{t_k} \\ &- \tau_s \left[\lambda_{\mathbf{I}}^+ \cdot \frac{\partial \mathbf{I}^+}{\partial W_{i,j}} \right]_{k=0} + \tau_s \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{I}}^- \cdot \frac{\partial \mathbf{I}^-}{\partial W_{i,j}} - \lambda_{\mathbf{I}}^+ \cdot \frac{\partial \mathbf{I}^+}{\partial W_{i,j}} \right]_{t_k} \\ &- \tau_d \left[\lambda_{\mathbf{D}}^+ \cdot \frac{\partial \mathbf{D}^+}{\partial W_{i,j}} \right]_{k=0} + \tau_d \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{D}}^- \cdot \frac{\partial \mathbf{D}^-}{\partial W_{i,j}} - \lambda_{\mathbf{D}}^+ \cdot \frac{\partial \mathbf{D}^+}{\partial W_{i,j}} \right]_{t_k} \end{aligned} \tag{A.36}$$

Now, the starting condition of the internal variable $\mathbf{V}$, $\mathbf{I}$ and $\mathbf{D}$ are parameter independent ($\mathbf{V}|_{k=0} = 0,\ \mathbf{I}|_{k=0} = 0,\ \mathbf{D}|_{k=0} = 0$), therefore:

$$\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_n\bigg|_{k=0} = \left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_n\bigg|_{k=0} = \left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_n\bigg|_{k=0} = 0$$

Moreover, the loss function $l_V^+\left(t_0^{post}, \mathbf{V}\right)$ is also null at the beginning of the simulation. Therefore we can re-write the B term as:

$$B = \sum_{k=1}^{N_{post}} \left[l_V^-\left(t_k^{post}, \mathbf{V}\right) - l_V^+\left(t_k^{post}, \mathbf{V}\right)\right] \frac{\partial t_k^{post}}{\partial W_{i,j}} + \tau_m \left[\lambda_{\mathbf{V}}^- \cdot \frac{\partial \mathbf{V}^-}{\partial W_{i,j}} - \lambda_{\mathbf{V}}^+ \cdot \frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right]_{t_k} + \tau_s \left[\lambda_{\mathbf{I}}^- \cdot \frac{\partial \mathbf{I}^-}{\partial W_{i,j}} - \lambda_{\mathbf{I}}^+ \cdot \frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right]_{t_k} + \tau_d \left[\lambda_{\mathbf{D}}^- \cdot \frac{\partial \mathbf{D}^-}{\partial W_{i,j}} - \lambda_{\mathbf{D}}^+ \cdot \frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right]_{t_k} \tag{A.37}$$

Now, we can apply the implicit function theorem to the discontinuity equations of all neural state variable (Eqs. A.5, A.6, A.7, A.8, A.7, A.8 and A.11):

$$\frac{dt_k}{dW_{i,j}} = -\frac{\left(\frac{\partial(\mathbf{V}^- - \theta)}{\partial W_{i,j}}\right)_{n(k)}}{\left(\frac{\partial(\mathbf{V}^- - \theta)}{\partial t_k}\right)_{n(k)}} = -\frac{\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}}{\left(\dot{\mathbf{V}^-}\right)_{n(k)}} \tag{A.38}$$

$$\frac{dt_k}{dW_{i,j}} = -\frac{\left(\frac{\partial(\mathbf{V}^+ - \gamma)}{\partial W_{i,j}}\right)_{n(k)}}{\left(\frac{\partial(\mathbf{V}^+ - \gamma)}{\partial t_k}\right)_{n(k)}} = -\frac{\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)}}{\left(\dot{\mathbf{V}^+}\right)_{n(k)}} \tag{A.39}$$

$$\frac{dt_k}{dW_{i,j}} = -\frac{\left(\frac{\partial(\mathbf{V}^- - \mathbf{V}^+)}{\partial W_{i,j}}\right)_m}{\left(\frac{\partial(\mathbf{V}^- - \mathbf{V}^+)}{\partial t_k}\right)_m} = -\frac{\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_m}{\left(\dot{\mathbf{V}^-}\right)_m - \left(\dot{\mathbf{V}^+}\right)_m} \tag{A.40}$$

Then, taking into account that eq. A.1 is always verified for a non-spiking neuron we can write: Consequently, considering eq. A.10:

$$\left(\dot{\mathbf{V}^-}\right)_m - \left(\dot{\mathbf{V}^+}\right)_m = -\frac{R\phi(\mathbf{D}_{\mathbf{m}}^-)}{\tau_m}\left(\left(\mathbf{I}^-\right)_m - \left(\mathbf{I}^+\right)_m\right) + \frac{1}{\tau_m}\underbrace{\left(\left(\mathbf{V}^-\right)_m - \left(\mathbf{V}^+\right)_m\right)}_{=0} \tag{A.41}$$

Therefore, substituting eqs. A.6 and A.8:

$$\begin{aligned}\left(\dot{\mathbf{V}^-}\right)_m - \left(\dot{\mathbf{V}^+}\right)_m &= -\frac{R\phi\left(\left(\mathbf{D}^-\right)_m\right)}{\tau_m}\left(\left(\mathbf{I}^-\right)_m + \beta\mathbf{W}_{\mathbf{m.n}} - \left(\mathbf{I}^-\right)_m\right) \\ &= -\frac{R\phi\left(\left(\mathbf{D}^-\right)_m\right)}{\tau_m}\beta\mathbf{W}_{\mathbf{m.n}}\end{aligned} \tag{A.42}$$

Now, we can substitute the result just obtained in eq. A.42 in eq. A.40:

$$\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_m = \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m - \frac{R\phi\left(\left(\mathbf{D}^-\right)_m\right)}{\tau_m}\beta\mathbf{W}_{\mathbf{m.n}}\frac{dt_k}{dW_{i,j}} \tag{A.43}$$

Then, the dt_k/dW can be expressed by considering eq. A.38:

$$\begin{aligned}
\frac{dt_k}{dW_{i,j}} &= -\frac{1}{\left(\dot{\mathbf{V}}^-\right)_{n(k)}}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \\
&= -\frac{\tau_m}{\left(-\underbrace{\left(\mathbf{V}^-\right)_{n(k)}}_{=\theta} + R\left(\mathbf{I}^-\right)_{n(k)}\underbrace{\phi\left(\left(\mathbf{D}^-\right)_{n(k)}\right)}_{=1}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \\
&= -\frac{\tau_m}{\left(-\theta + R\left(\mathbf{I}^-\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}
\end{aligned} \tag{A.44}$$

Natural existence condition is $\left(\dot{\mathbf{V}}\right)_{n(k)} \neq 0$ that should be always verified. Considering eq. A.39 we have:

$$\begin{aligned}
\frac{dt_k}{dW_{i,j}} &= -\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)}\frac{1}{\left(\dot{\mathbf{V}}^+\right)_{n(k)}} \\
&= -\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)}\frac{\tau_m}{\left(-\underbrace{\left(\mathbf{V}^+\right)_{n(k)}}_{=\gamma} + R\underbrace{\left(\mathbf{I}^+\right)_{n(k)}}_{G\left(\mathbf{I}^-\right)_{n(k)}}\phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\right)} \\
&= -\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)}\frac{\tau_m}{\left(-\gamma + RG\left(\mathbf{I}^-\right)_{n(k)}\phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\right)}
\end{aligned} \tag{A.45}$$

Substituting A.39 in A.45 we obtain:

$$\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)} = \frac{\gamma - RG\left(\mathbf{I}^-\right)_{n(k)}\phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)}{\left(\theta - R\left(\mathbf{I}^-\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \tag{A.46}$$

By applying the definition of partial derivative to eq. A.7 we obtain:

$$\begin{aligned}
\frac{d\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{dW_{i,j}} &= \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial W_{i,j}}\underbrace{\frac{dW_{i,j}}{dW_{i,j}}}_{=1} + \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial t_k}\frac{dt_k}{dW_{i,j}} \\
&= \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial W_{i,j}} + \left(\left(\dot{\mathbf{I}}^+\right)_{n(k)} - G\left(\dot{\mathbf{I}}^-\right)_{n(k)}\right)\frac{dt_k}{dW_{i,j}} \\
&= \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial W_{i,j}} + \frac{1}{\tau_s}\underbrace{\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}_{=0}\frac{dt_k}{dW_{i,j}}
\end{aligned} \tag{A.47}$$

This means that the total derivative is equal to the partial derivative, consequently we have:

$$\left(\frac{\partial \mathbf{I}^{+}}{\partial W_{i,j}}\right)_{n(k)} = G\left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_{n(k)} \tag{A.48}$$

By applying the implicit function theorem to eq. A.8 we obtain:

$$\begin{aligned}\frac{dt_k}{dW_{i,j}} &= -\frac{\left(\frac{\partial \mathbf{I}^{+}}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_m - \beta\left(\frac{\partial W_{m,n}}{\partial W_{i,j}}\right)_m}{\underbrace{\left(\frac{\partial \mathbf{I}^{+}}{\partial t_k}\right)_m}_{(\dot{\mathbf{I}^{+}})_m} - \underbrace{\left(\frac{\partial \mathbf{I}^{-}}{\partial t_k}\right)_m}_{(\dot{\mathbf{I}^{-}})_m} - \beta\underbrace{\left(\frac{\partial W_{m,n}}{\partial t_k}\right)_m}_{=0}} \\ &= -\tau_s\frac{\left(\frac{\partial \mathbf{I}^{+}}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_m - \beta\delta_{i,m}\delta_{j,n}}{-\left(\mathbf{I}^{+}\right)_m + \left(\mathbf{I}^{-}\right)_m} \\ &= \tau_s\frac{\left(\frac{\partial \mathbf{I}^{+}}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_m - \beta\delta_{i,m}\delta_{j,n}}{\beta W_{m,n}}\end{aligned} \tag{A.49}$$

Therefore, we can rewrite the equation below as:

$$\left(\frac{\partial \mathbf{I}^{+}}{\partial W_{i,j}}\right)_m = \frac{\beta W_{m,n}}{\tau_s}\frac{dt_k}{dW_{i,j}} + \beta\delta_{i,m}\delta_{j,n} + \left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_m \tag{A.50}$$

We applied the implicit function theorem on eq. A.9:

$$\frac{dt_k}{dW_{i,j}} = -\frac{\left(\frac{\partial \mathbf{D}^{+}}{\partial W_{i,j}}\right)_{n(k)}}{\underbrace{\left(\frac{\partial \mathbf{D}^{+}}{\partial t_k}\right)_{n(k)}}_{(\dot{\mathbf{D}^{+}})_m}} = \tau_D\frac{\left(\frac{\partial \mathbf{D}^{+}}{\partial W_{i,j}}\right)_{n(k)}}{\left(\mathbf{D}^{+}\right)_{n(k)}} \tag{A.51}$$

And consequently we have:

$$\begin{aligned}\left(\frac{\partial \mathbf{D}^{+}}{\partial W_{i,j}}\right)_{n(k)} &= \frac{1}{\tau_D}\frac{dt_k}{dW_{i,j}}\underbrace{\left(\mathbf{D}^{+}\right)_{n(k)}}_{=\epsilon} \\ &= -\frac{\tau_m\epsilon}{\tau_D\left(-\theta + R\left(\mathbf{I}^{-}\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^{-}}{\partial W_{i,j}}\right)_{n(k)}\end{aligned} \tag{A.52}$$

While the m-component is similar to eq. A.48 and corresponds to:

$$\left(\frac{\partial \mathbf{D}^{+}}{\partial W_{i,j}}\right)_m = \left(\frac{\partial \mathbf{D}^{-}}{\partial W_{i,j}}\right)_m \tag{A.53}$$

Then, combining eqs. A.54, A.44,A.52 and A.50, we obtain:

$$\begin{aligned}\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_m &= \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m - \frac{R\phi\left((\mathbf{D}^-)_m\right)}{\tau_m}\beta W_{m,n}\frac{dt_k}{dW_{i,j}} \\ &= \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m + \frac{R\beta W_{m,n}\phi\left((\mathbf{D}^-)_m\right)}{-\theta + R\left(\mathbf{I}^-\right)_{n(k)}}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}\end{aligned} \tag{A.54}$$

$$\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_m = -\frac{\beta W_{m,n}\tau_m}{\tau_s\left(-\theta + R\left(\mathbf{I}^-\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \beta\delta_{i,m}\delta_{j,n} + \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m \tag{A.55}$$

Now, considering the inner products of eq. A.37 we can expand each term as represented below:

$$\lambda\cdot\mathbf{d} = (\lambda)_{n(k)}(\mathbf{d})_{n(k)} + \sum_{m\neq n(k)}^{N_n-1}(\lambda)_m(\mathbf{d})_m$$

Consequently, eq. A.36 expands in:

$$\begin{aligned}B = &\sum_{k=1}^{N_{post}}\left[l_V^-\left(t_k^{post},\mathbf{V}\right) - l_V^+\left(t_k^{post},\mathbf{V}\right)\right]\frac{\partial t_k^{post}}{\partial W_{i,j}} + \\ &+ \tau_m\left(\lambda_{\mathbf{V}}^-\right)_{n(k)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \tau_m\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{V}}^-\right)_m\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m + \\ &- \tau_m\left(\lambda_{\mathbf{V}}^+\right)_{n(k)}\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)} - \tau_m\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{V}}^+\right)_m\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_m + \\ &+ \tau_s\left(\lambda_{\mathbf{I}}^-\right)_{n(k)}\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_{n(k)} + \tau_s\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{I}}^-\right)_m\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m + \\ &- \tau_s\left(\lambda_{\mathbf{I}}^+\right)_{n(k)}\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_{n(k)} - \tau_s\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{I}}^+\right)_m\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_m + \\ &\tau_D\left(\lambda_{\mathbf{D}}^-\right)_{n(k)}\left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_{n(k)} + \tau_D\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{D}}^-\right)_m\left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_m + \\ &- \tau_D\left(\lambda_{\mathbf{D}}^+\right)_{n(k)}\left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_{n(k)} - \tau_D\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{D}}^+\right)_m\left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_m +\end{aligned} \tag{A.56}$$

$$
\begin{aligned}
B = & \sum_{k=1}^{N_{post}} \left[l_V^- \left(t_k^{post}, \mathbf{V} \right) - l_V^+ \left(t_k^{post}, \mathbf{V} \right) \right] \frac{\tau_m}{\left(\theta - R \left(\mathbf{I}^- \right)_{n(k)} \right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} + \\
& + \tau_m \left(\lambda_{\mathbf{V}}^- \right)_{n(k)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} + \tau_m \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{V}}^- \right)_m \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_m + \\
& - \tau_m \left(\lambda_{\mathbf{V}}^+ \right)_{n(k)} \frac{\gamma - RG \left(\mathbf{I}^- \right)_{n(k)} \phi \left(\left(\mathbf{D}^+ \right)_{n(k)} \right)}{\left(\theta - R \left(\mathbf{I}^- \right)_{n(k)} \right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} + \\
& - \tau_m \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{V}}^+ \right)_m \left[\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_m - \frac{R \beta W_{m,n} \phi \left(\left(\mathbf{D}^- \right)_m \right)}{\theta - R \left(\mathbf{I}^- \right)_{n(k)}} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} \right] + \\
& + \tau_s \left(\lambda_{\mathbf{I}}^- \right)_{n(k)} \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}} \right)_{n(k)} + \tau_s \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{I}}^- \right)_m \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}} \right)_m + \\
& - \tau_s G \left(\lambda_{\mathbf{I}}^+ \right)_{n(k)} \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}} \right)_{n(k)} + \\
& - \tau_s \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{I}}^+ \right)_m \left[\frac{\beta W_{m,n} \tau_m}{\tau_s \left(\theta - R \left(\mathbf{I}^- \right)_{n(k)} \right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} + \beta \delta_{i,m} \delta_{j,n} + \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}} \right)_m \right] + \\
& + \tau_D \left(\lambda_{\mathbf{D}}^- \right)_{n(k)} \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}} \right)_{n(k)} + \tau_D \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{D}}^- \right)_m \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}} \right)_m + \\
& - \left(\lambda_{\mathbf{D}}^+ \right)_{n(k)} \frac{\tau_m \epsilon}{\left(\theta - R \left(\mathbf{I}^- \right)_{n(k)} \right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} + \\
& - \tau_D \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{D}}^+ \right)_m \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}} \right)_m
\end{aligned}
\tag{A.57}
$$

$$
A = \sum_{k=1}^{N_{post}} \frac{\partial l_p (t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} = - \sum_{k=1}^{N_{post}} \frac{\tau_m}{\left(-\theta + R \left(\mathbf{I}^- \right)_{n(k)} \right)} \frac{\partial l_p (t^{post})}{\partial t_k^{post}} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} \tag{A.58}
$$

Let's now group different terms:

$$\begin{aligned}\frac{d\mathscr{L}}{dW_{i,j}} = \sum_{k=1}^{N_{post}} \frac{\tau_m}{\left(\theta - R\left(\mathbf{I}^-\right)_{n(k)}\right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \Bigg[& -\frac{\partial l_p(t^{post})}{\partial t_k^{post}} + \left(l_V^-\left(t_k^{post}, \mathbf{V}\right) - l_V^+\left(t_k^{post}, \mathbf{V}\right)\right) + \\ & \left(\lambda_{\mathbf{V}}^-\right)_{n(k)} \left(\theta - R\left(\mathbf{I}^-\right)_{n(k)}\right) - \left(\lambda_{\mathbf{V}}^+\right)_{n(k)} \left(\gamma - RG\phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\left(\mathbf{I}^-\right)_{n(k)}\right) + \\ & \sum_{m \neq n(k)}^{N_n - 1} \beta W_{m,n} \left(\left(\lambda_{\mathbf{V}}^+\right)_m R\phi\left(\left(\mathbf{D}^+\right)_m\right) - \left(\lambda_{\mathbf{I}}^+\right)_m\right) - \left(\lambda_D^+\right)_{n(k)} \epsilon \Bigg] + \\ & \sum_{m \neq n(k)}^{N_n - 1} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m \tau_m \left[\left(\lambda_{\mathbf{V}}^-\right)_m - \left(\lambda_{\mathbf{V}}^+\right)_m\right] + \\ & \sum_{m \neq n(k)}^{N_n - 1} \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m \tau_s \left[\left(\lambda_{\mathbf{I}}^-\right)_m - \left(\lambda_{\mathbf{I}}^+\right)_m\right] + \\ & \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_{n(k)} \tau_s \left[\left(\lambda_{\mathbf{I}}^-\right)_{n(k)} - G\left(\lambda_{\mathbf{I}}^+\right)_{n(k)}\right] + \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_{n(k)} \tau_D \left(\lambda_{\mathbf{D}}^-\right)_{n(k)} + \\ & \sum_{m \neq n(k)}^{N_n - 1} \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_m \tau_D \left[\left(\lambda_{\mathbf{D}}^-\right)_m - \left(\lambda_{\mathbf{D}}^+\right)_m\right] + \\ & - \sum_{m \neq n(k)}^{N_n - 1} \tau_s \beta \delta_{i,m} \delta_{j,n} \left(\lambda_{\mathbf{I}}^-\right)_{n(k)}\end{aligned} \tag{A.59}$$

In order to delete as much term possible we can define the following continuity conditions:

$$\left(\lambda_{\mathbf{V}}^-\right)_m = \left(\lambda_{\mathbf{V}}^+\right)_m \tag{A.60}$$

$$\left(\lambda_{\mathbf{I}}^-\right)_m = \left(\lambda_{\mathbf{I}}^+\right)_m \tag{A.61}$$

$$\left(\lambda_{\mathbf{I}}^-\right)_{n(k)} = G\left(\lambda_{\mathbf{I}}^+\right)_m \tag{A.62}$$

$$\left(\lambda_{\mathbf{D}}^-\right)_m = \left(\lambda_{\mathbf{D}}^+\right)_m \tag{A.63}$$

$$\left(\lambda_{\mathbf{D}}^-\right)_{n(k)} = 0 \tag{A.64}$$

$$
\begin{aligned}
\left(\lambda_{\mathbf{V}}^{-}\right)_{n(k)} =& \frac{1}{\left(\theta - R\left(\mathbf{I}^{-}\right)_{n(k)}\right)} \Bigg[\frac{\partial l_p(t^{post})}{\partial t_k^{post}} + \left(l_V^{+}\left(t_k^{post}, \mathbf{V}\right) - l_V^{-}\left(t_k^{post}, \mathbf{V}\right)\right) + \\
& \left(\lambda_{\mathbf{V}}^{+}\right)_{n(k)} \left(\gamma - RG\phi\left(\left(\mathbf{D}^{+}\right)_{n(k)}\right)\left(\mathbf{I}^{-}\right)_{n(k)}\right) + \\
& \sum_{m \neq n(k)}^{N_n - 1} \beta W_{m,n} \left(\left(\lambda_{\mathbf{I}}^{+}\right)_m - \left(\lambda_{\mathbf{V}}^{+}\right)_m R\phi\left(\left(\mathbf{D}^{+}\right)_m\right)\right) + \left(\lambda_D^{+}\right)_{n(k)} \epsilon \Bigg]
\end{aligned}
\tag{A.65}
$$

And consequently eq. A.59 reduces to:

$$
\begin{aligned}
\frac{d\mathscr{L}}{dW_{i,j}} &= -\tau_s \sum_{k=1}^{N_{post}} \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{I}}^{+}\right)_m \beta \delta_{i,m} \delta_{j,n} = \\
&= -\tau_s \sum_{spikes from j} \left(\lambda_{\mathbf{I}_i}^{+}\right)_m \beta
\end{aligned}
\tag{A.66}
$$

Appendix B

Extension of Event-Prop backpropagation algorithm using adjoint variables in Izhikevich-based Spiking Neural Networks

In this section, we extend the backpropagation learning algorithm Event-Prop [89] with the biologically plausible Izhikevich neural model [69] to provide a reliable methodology to train biological compatible artificial spiking neural networks.

As mentioned in [89], the Event-prop algorithm is developed for *LIF* spiking neural networks. This makes unsuitable the backpropagation-based algorithm to train a network of Izhikevich neurons as it is. Nevertheless, we report in this paragraph an adaptation of Even-prop that substitutes the *LIF* with the Izhikevich model.

In order to simplify and generalize the treatment, we used the same nomenclature adopted in Sec. 2.2 in this paragraph. Moreover, for the sake of simplicity, we consider a parametric version of the membrane potential equation expressed in Eq. 2.4:

$$\frac{d\mathbf{V}}{dt} = M_0\mathbf{V}^2 + M_1\mathbf{V} + M_2 + \mathbf{I} - M_3\mathbf{U}$$

After the spiking event, the membrane potential resets at $(\mathbf{V}^+)_{n(k)} = \gamma = c$, while during other neuron spikes the membrane potential is continuous $(\mathbf{V}^+)_m = (\mathbf{V}^-)_m$. Consistently, also the membrane recovery potential $\mathbf{U}$ presents a discontinuous behavior during spikes $(\mathbf{U}^+)_{n(k)} = (\mathbf{U}^-)_{n(k)} + d$ and otherwise a continuous one $(\mathbf{U}^+)_m = (\mathbf{U}^-)_m$. Moreover, the spiking threshold condition is expressed by:

$$\left(\mathbf{V}^-\right)_{n(k)} = \theta \tag{B.1}$$

Then, we impose that post-synaptic current follows an exponential behavior as it charges a neural membrane which can be approximated with a linear capacitor, as described in literature [89]. Therefore, $\mathbf{I}$ follows:

$$\tau_s\frac{d\mathbf{I}}{dt} = -\mathbf{I} \tag{B.2}$$

where τ_s is the post-synaptic time constant.

The strong non-linearity of the Izhikevich model, combined with the presence of the membrane recovery variable which doesn't reset at each spiking event, generates a non-linear spiking time distribution. Due to this, the application of the 2-phase method proposed in Sec. 2.2 is not as straightforward as in the *LIF* case. Moreover, the Izhikevich model doesn't include any minimum refractory period to easily limit the spiking frequency, due to this, the application of the 2-phase method proposed in Sec. 2.2 can help in this sense, as well as include a dumping function that locally decouples the membrane potential and the input synaptic current for a limited period of time to prevent uncontrolled neural overexcitation during training. However, this last method leads to dissipative energy wastes in a neuromorphic application. To model an externally imposed refractory period T_r, we include a fictitious state variable $\mathbf{D}$ that resets to a fixed value ϵ at each spike emission and decades exponentially in time. As it will later be shown, this additive state variable should not be calculated during the normal operation of the algorithm, but it is useful just to mathematically demonstrate that the adjoint function related to the refractory period is always null and, consequently, the calculation of $\mathbf{D}$ itself is not needed during the training and the inference process. The dynamic equation that describes the behavior of $\mathbf{D}$ is:

$$\tau_d \frac{d\mathbf{D}}{dt} = -\mathbf{D} \tag{B.3}$$

where τ_D is the time constant associated with the state function $\mathbf{D}$. The discontinuity condition for this state function is, as mentioned before, $(\mathbf{D}^+)_{n(k)} = \epsilon$, while it is continuous during other neural spiking events $(\mathbf{D}^+)_m = (\mathbf{D}^-)_m$. Then the dumping function $\Phi(\mathbf{D})$ is included in membrane potential dynamic equation as:

$$\frac{d\mathbf{V}}{dt} = M_0\mathbf{V}^2 + M_1\mathbf{V} + M_2 + \Phi(\mathbf{D})\left(R\mathbf{I} - M_3\mathbf{U}\right)$$

where

$$\Phi(D) = \begin{cases} 0 & \text{when } D > \epsilon \exp\left(-T_r/\tau_r\right) \\ 1 & \text{otherwise} \end{cases} \tag{B.4}$$

The equations reported below describe the free dynamic of the complete model:

$$\begin{cases} \dfrac{d\mathbf{V}}{dt} = M_0\mathbf{V}^2 + M_1\mathbf{V} + M_2 + \Phi(\mathbf{D})\left(R\mathbf{I} - M_3\mathbf{U}\right) \\ \dfrac{d\mathbf{U}}{dt} = a\left(b\mathbf{V} - \mathbf{U}\right) \\ \tau_s \dfrac{d\mathbf{I}}{dt} = -\mathbf{I} \\ \tau_d \dfrac{d\mathbf{D}}{dt} = -\mathbf{D} \end{cases} \tag{B.5}$$

Then we can define implicit version of B.5:

$$\mathbf{f_V} := \tau_m\dot{\mathbf{V}} - M_0\mathbf{V}^2 - M_1\mathbf{V} - M_2 - \Phi(\mathbf{D})\left(R\mathbf{I} - M_3\mathbf{U}\right) \tag{B.6}$$

$$\mathbf{f_I} := \tau_s\dot{\mathbf{I}} + \mathbf{I} \tag{B.7}$$

$$\mathbf{f_D} := \tau_d\dot{\mathbf{D}} + \mathbf{D} \tag{B.8}$$

$$\mathbf{f_U} := \tau_d \dot{\mathbf{U}} - a\left(b\mathbf{V} - \mathbf{U}\right) \tag{B.9}$$

where $\mathbf{V}, \mathbf{I}, \mathbf{D}, \mathbf{U}, \mathbf{f_V}, \mathbf{f_I}, \mathbf{f_D}, \mathbf{f_U} \in \Re^{N_n}$ and N_n is the number of neurons in the network. Moreover the transition condition of the the state variables reported below for a spiking neuron (denoted with $n(k)$) and a non spiking neuron (denoted with m) are:

$$\left(\mathbf{V}^{+}\right)_{n(k)} = \gamma = c \tag{B.10}$$

c is the parameter defined by the Izhikevich model.

$$\left(\mathbf{V}^{+}\right)_{m} = \left(\mathbf{V}^{-}\right)_{m} \tag{B.11}$$

$$\left(\mathbf{I}^{+}\right)_{n(k)} = G\left(\mathbf{I}^{-}\right)_{n(k)} \tag{B.12}$$

G is a scale factor applied to input synaptic current during the spiking event. Usually, G is imposed at 1.0

$$\left(\mathbf{I}^{+}\right)_{m} = \left(\mathbf{I}^{-}\right)_{m} + \beta \mathbf{W_{m,n}} \tag{B.13}$$

where $\mathbf{W_{m,n}}$ is the synaptic conductance that links the neuron n with the post-synaptic neuron m. β is the post-neural potential.

$$\left(\mathbf{D}^{+}\right)_{n(k)} = \epsilon \tag{B.14}$$

$$\left(\mathbf{D}^{+}\right)_{m} = \left(\mathbf{D}^{-}\right)_{m} \tag{B.15}$$

$$\left(\mathbf{U}^{+}\right)_{m} = \left(\mathbf{U}^{-}\right)_{m} \tag{B.16}$$

$$\left(\mathbf{U}^{+}\right)_{n(k)} = \left(\mathbf{U}^{-}\right)_{n(k)} + d \tag{B.17}$$

Moreover, as described by [69], the fictitious state variable $\mathbf{U}$ is augmented by a d factor during the fire event. The threshold condition is defined as:

$$\left(\mathbf{V}^{-}\right)_{n(k)} = \theta \tag{B.18}$$

While the initial conditions of the state functions are defined as the static solution of the state variable differential equation:

$$\begin{cases} \dfrac{d\mathbf{V}}{dt} = M_0\mathbf{V}^2 + M_1\mathbf{V} + M_2 + \Phi(\mathbf{D})\left(R\mathbf{I} - M_3\mathbf{U}\right) = 0 \\ \dfrac{d\mathbf{U}}{dt} = a\left(b\mathbf{V} - \mathbf{U}\right) = 0 \end{cases} \tag{B.19}$$

In this case $\Phi(\mathbf{D}) = 1$, $\mathbf{I} \to 0$ and

$$\mathbf{U} = b\mathbf{V} \tag{B.20}$$

By substituting Eq. B.20 in Eq. B.19, we obtain:

$$M_0\mathbf{V}^2 + (M_1 - bM_3)\,\mathbf{V} + M_2 = 0 \tag{B.21}$$

Which solutions are:

$$V_{1,2} = \frac{-M_1 + bM_3 \pm \sqrt{(M_1 - bM_3)^2 - 4M_0M_2}}{2M_0} \tag{B.22}$$

That is valid, just when:

$$|M_1 - M_3b| > \sqrt{4M_0M_2} \tag{B.23}$$

The vectors $\mathbf{V}$ and $\mathbf{U}$ should be initialized as Eqs. B.22 and B.20. When the condition expressed by Eq. B.23 is not verified, the starting value for the membrane potential is $\mathbf{V} = c$. The initial value of the state functions doesn't affect the adjoint function method to calculate the weight's gradient.

The cost function to minimize has been generalized as follows:

$$\mathscr{L} = l_p(t^{post}) + \int_T l_V\left(t^{post}, \mathbf{V}\right) dt^{post} \tag{B.24}$$

and, in general, the integral can be split between spikes events because $l_V\left(t^{post}, \mathbf{V}\right)$ is always defined and limited in time:

$$\mathscr{L} = l_p(t^{post}) + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} l_V\left(t^{post}, \mathbf{V}\right)\; dt^{post} \tag{B.25}$$

where $l_V\left(t^{post}, \mathbf{V}\right) : \Re^{N_n} \rightarrow \Re$. Moreover, the sum over k starts from 0, consequently, t_0 denotes the beginning of the simulation. The time gap $t_0 - t_1$ is the inactivity period of the network where no spike has been emitted by the network.

In order to calculate the gradient of the loss function, we need to add the adjoint variables $\lambda_{\mathbf{V}}$, $\lambda_{\mathbf{I}}$ and $\lambda_{\mathbf{D}}$ to B.25 :

$$\mathscr{L} = l_p(t^{post}) + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} l_V\left(t^{post}, \mathbf{V}\right) + \lambda_{\mathbf{V}}\cdot\mathbf{f_V} + \lambda_{\mathbf{I}}\cdot\mathbf{f_I} + \lambda_{\mathbf{D}}\cdot\mathbf{f_D} + \lambda_{\mathbf{U}}\cdot\mathbf{f_U}\; dt^{post} \tag{B.26}$$

Therefore, by differentiating the payoff function we obtain:

$$\begin{aligned}\frac{d\mathscr{L}}{dW_{i,j}} =& \frac{dl_p(t^{post})}{dW_{i,j}} + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} dt^{post} \frac{dl_V\left(t_k^{post}, \mathbf{V}\right)}{dW_{i,j}} + \\ &+ \frac{d\lambda_{\mathbf{V}}\cdot\mathbf{f_V}}{dW_{i,j}} + \frac{d\lambda_{\mathbf{I}}\cdot\mathbf{f_I}}{dW_{i,j}} + \frac{d\lambda_{\mathbf{D}}\cdot\mathbf{f_U}}{dW_{i,j}} + \frac{d\lambda_{\mathbf{U}}\cdot\mathbf{f_U}}{dW_{i,j}}\end{aligned} \tag{B.27}$$

And because the adjoint functions don't depend on the weights:

$$\begin{aligned}\frac{d\mathscr{L}}{dW_{i,j}} = \frac{dl_p(t^{post})}{dW_{i,j}} + \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} dt^{post} \frac{dl_V\left(t^{post}, \mathbf{V}\right)}{dW_{i,j}} + \\ + \lambda_{\mathbf{V}}\cdot\frac{d\mathbf{f_V}}{dW_{i,j}} + \lambda_{\mathbf{I}}\cdot\frac{d\mathbf{f_I}}{dW_{i,j}} + \lambda_{\mathbf{D}}\cdot\frac{d\mathbf{f_D}}{dW_{i,j}} + \lambda_{\mathbf{U}}\cdot\frac{d\mathbf{f_U}}{dW_{i,j}}\end{aligned} \tag{B.28}$$

Now each term should be expanded. Let's begin with the loss functions terms $l_V\left(t^{post}, \mathbf{V}\right)$ and $l_p(t^{post})$. These loss functions depend indirectly from the weights of networks, therefore, we can compute the total derivative with the chain rule:

$$\frac{dl_V\left(t_k^{post}, \mathbf{V}\right)}{dW_{i,j}} = \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} + \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \frac{\partial \mathbf{V}}{\partial W_{i,j}} \tag{B.29}$$

$$\frac{dl_p(t^{post})}{dW_{i,j}} = \sum_{k=0}^{N_{post}} \frac{\partial l_p(t_k^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \tag{B.30}$$

Then, by taking into account B.6 and B.7:

$$\begin{aligned}
\lambda_{\mathbf{V}} \cdot \frac{d\mathbf{f_V}}{dW_{i,j}} =& \lambda_{\mathbf{V}} \cdot \left[\frac{d\dot{\mathbf{V}}}{dW_{i,j}} - M_0 \frac{d\mathbf{V}^2}{dW_{i,j}} - M_1 \frac{d\mathbf{V}}{dW_{i,j}} + \right. \\
& \left. +\Phi(\mathbf{D}) \left(M_3 \frac{d\mathbf{U}}{dW_{i,j}} - R \frac{d\mathbf{I}}{dW_{i,j}}\right)\right] \\
=& \lambda_{\mathbf{V}} \cdot \left[\frac{d\dot{\mathbf{V}}}{dW_{i,j}} - 2M_0 \mathbf{V} \frac{d\mathbf{V}}{dW_{i,j}} - M_1 \frac{d\mathbf{V}}{dW_{i,j}} + \right. \\
& \left. +\Phi(\mathbf{D}) \left(M_3 \frac{d\mathbf{U}}{dW_{i,j}} - R \frac{d\mathbf{I}}{dW_{i,j}}\right)\right]
\end{aligned} \tag{B.31}$$

$$\lambda_{\mathbf{I}} \cdot \frac{d\mathbf{f_I}}{dW_{i,j}} = \lambda_{\mathbf{I}} \cdot \left(\tau_s \frac{d\dot{\mathbf{I}}}{dW_{i,j}} + \frac{d\mathbf{I}}{dW_{i,j}}\right) \tag{B.32}$$

$$\lambda_{\mathbf{D}} \cdot \frac{d\mathbf{f_D}}{dW_{i,j}} = \lambda_{\mathbf{D}} \cdot \left(\tau_D \frac{d\dot{\mathbf{D}}}{dW_{i,j}} + \frac{d\mathbf{D}}{dW_{i,j}}\right) \tag{B.33}$$

$$\lambda_{\mathbf{U}} \cdot \frac{d\mathbf{f_U}}{dW_{i,j}} = \lambda_{\mathbf{U}} \cdot \left(-ab \frac{d\mathbf{V}}{dW_{i,j}} + a \frac{d\mathbf{U}}{dW_{i,j}} + \frac{d\dot{\mathbf{U}}}{dW_{i,j}}\right) \tag{B.34}$$

Because the weights of the network are fixed in time, and in particular between two spikes events (also due to the Gronwall's theorem) we can exchange the position of the differential operators:

$$\frac{d}{dW_{i,j}}\left[\frac{d}{dt}\right] = \frac{d}{dt}\left[\frac{d}{dW_{i,j}}\right]$$

Substituting then eqs. B.29, B.30, B.31, B.32 and B.34 in eq. B.28, we obtain:

$$\begin{aligned}
\frac{d\mathscr{L}}{dW_{i,j}} &= \sum_{k=0}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\
&+ \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} + \frac{\partial l_V\left(t_k^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \frac{\partial \mathbf{V}}{\partial W_{i,j}} + \\
&+ \lambda_{\mathbf{V}} \cdot \left[\frac{d\dot{\mathbf{V}}}{dW_{i,j}} - 2M_0 \mathbf{V} \frac{d\mathbf{V}}{dW_{i,j}} - M_1 \frac{d\mathbf{V}}{dW_{i,j}} + \right. \\
&\left. + \Phi(\mathbf{D}) \left(M_3 \frac{d\mathbf{U}}{dW_{i,j}} - R \frac{d\mathbf{I}}{dW_{i,j}} \right) \right] + \\
&+ \lambda_{\mathbf{I}} \cdot \left(\tau_s \frac{d\dot{\mathbf{I}}}{dW_{i,j}} + \frac{d\mathbf{I}}{dW_{i,j}} \right) \\
&+ \lambda_{\mathbf{D}} \cdot \left(\tau_D \frac{d\dot{\mathbf{D}}}{dW_{i,j}} + \frac{d\mathbf{D}}{dW_{i,j}} \right) + \\
&+ \lambda_{\mathbf{U}} \cdot \left(\frac{d\dot{\mathbf{U}}}{dW_{i,j}} - ab \frac{d\mathbf{V}}{dW_{i,j}} + a \frac{d\mathbf{U}}{dW_{i,j}} \right) dt^{post}
\end{aligned} \tag{B.35}$$

Then, we need to integrate by parts:

$$\begin{aligned}
\int_{t_k^{post}}^{t_{k+1}^{post}} \frac{\partial l_V\left(t^{post}, \mathbf{V}\right)}{\partial t^{post}} \frac{\partial t^{post}}{\partial W_{i,j}} \, dt^{post} &= \left[l_V\left(t^{post}, \mathbf{V}\right) \frac{\partial t^{post}}{\partial W_{i,j}} \right]_{t_k}^{t_{k+1}} \\
&- \int_{t_k^{post}}^{t_{k+1}^{post}} l_V\left(t^{post}, \mathbf{V}\right) \underbrace{\frac{\partial}{\partial W_{i,j}} \left[\underbrace{\frac{\partial t^{post}}{\partial t^{post}}}_{=1} \right]}_{=0} dt^{post} = \\
&= l_V\left(t_{k+1}^{post}, \mathbf{V}\right) \frac{\partial t_{k+1}^{post}}{\partial W_{i,j}} - l_V\left(t_k^{post}, \mathbf{V}\right) \frac{\partial t_k^{post}}{\partial W_{i,j}}
\end{aligned} \tag{B.36}$$

$$\begin{aligned}
\int_{t_k^{post}}^{t_{k+1}^{post}} \lambda_{\mathbf{V}} \cdot \underbrace{\frac{\partial \dot{\mathbf{V}}}{\partial W_{i,j}}}_{=\frac{d}{dt}\left[\frac{\partial \mathbf{V}}{\partial W_{i,j}}\right]} dt^{post} &= \left[\lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}} \right]_{t_k}^{t_{k+1}} - \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda_{\mathbf{V}}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}} \, dt^{post} = \\
&= \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}} \bigg|_{t_{k+1}} - \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}} \bigg|_{t_k} - \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda_{\mathbf{V}}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}} \, dt^{post}
\end{aligned} \tag{B.37}$$

The same happens for I and D:

$$\int_{t_k^{post}}^{t_{k+1}^{post}} \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \dot{\mathbf{I}}}{\partial W_{i,j}} \, dt^{post} = \\ = \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_k} - \tau_s \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda}_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}} \, dt^{post} \tag{B.38}$$

$$\int_{t_k^{post}}^{t_{k+1}^{post}} \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \dot{\mathbf{D}}}{\partial W_{i,j}} \, dt^{post} = \\ = \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_k} - \tau_d \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda}_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}} \, dt^{post} \tag{B.39}$$

$$\int_{t_k^{post}}^{t_{k+1}^{post}} \lambda_{\mathbf{U}} \cdot \frac{\partial \dot{\mathbf{U}}}{\partial W_{i,j}} \, dt^{post} = \\ = \lambda_{\mathbf{U}} \cdot \frac{\partial \mathbf{U}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \lambda_{\mathbf{U}} \cdot \frac{\partial \mathbf{U}}{\partial W_{i,j}}\bigg|_{t_k} - \int_{t_k^{post}}^{t_{k+1}^{post}} \dot{\lambda}_{\mathbf{U}} \cdot \frac{\partial \mathbf{U}}{\partial W_{i,j}} \, dt^{post} \tag{B.40}$$

Combining Eqs. B.36, B.37, B.38, B.39 and B.40 in B.35, we obtain:

$$\frac{d\mathscr{L}}{dW_{i,j}} = A + B + C \tag{B.41}$$

where A,B and C are respectively:

$$A = \sum_{k=0}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \tag{B.42}$$

$$\begin{aligned} B = \sum_{k=0}^{N_{post}} & l_V\left(t_{k+1}^{post}, \mathbf{V}\right) \frac{\partial t_{k+1}^{post}}{\partial W_{i,j}} - l_V\left(t_k^{post}, \mathbf{V}\right) \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\ & + \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \lambda_{\mathbf{V}} \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}}\bigg|_{t_k} + \\ & + \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_s \lambda_{\mathbf{I}} \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}}\bigg|_{t_k} + \\ & + \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \tau_d \lambda_{\mathbf{D}} \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}}\bigg|_{t_k} \\ & + \lambda_{\mathbf{U}} \cdot \frac{\partial \mathbf{U}}{\partial W_{i,j}}\bigg|_{t_{k+1}} - \lambda_{\mathbf{U}} \cdot \frac{\partial \mathbf{U}}{\partial W_{i,j}}\bigg|_{t_k} + \end{aligned} \tag{B.43}$$

$$
\begin{aligned}
C = \sum_{k=0}^{N_{post}} \int_{t_k^{post}}^{t_{k+1}^{post}} & \left[-\dot{\lambda_{\mathbf{V}}} - 2M_0\lambda_{\mathbf{V}}\mathbf{V} - M_1\mathbf{V} - ab\lambda_{\mathbf{U}} + \frac{\partial l_V\left(t^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \right] \cdot \frac{\partial \mathbf{V}}{\partial W_{i,j}} + \\
& + \left[-\tau_s \dot{\lambda_{\mathbf{I}}} + \lambda_{\mathbf{I}} - \Phi(\mathbf{D}) R \lambda_{\mathbf{I}} \right] \cdot \frac{\partial \mathbf{I}}{\partial W_{i,j}} + \\
& + \left[-\tau_d \dot{\lambda_{\mathbf{D}}} + \lambda_{\mathbf{D}} \right] \cdot \frac{\partial \mathbf{D}}{\partial W_{i,j}} \\
& + \left[-\dot{\lambda_{\mathbf{U}}} + a\lambda_{\mathbf{U}} + \Phi(\mathbf{D}) M_3 \lambda_{\mathbf{V}} \right] \cdot \frac{\partial \mathbf{U}}{\partial W_{i,j}} dt^{post}
\end{aligned}
\tag{B.44}
$$

In particular, C becomes $= 0$ when the terms between square brackets are imposed $= 0$, defining in this way the dynamic response for the adjoint functions:

$$
\frac{d\lambda_{\mathbf{V}}}{dt} = -2M_0\lambda_{\mathbf{V}}\mathbf{V} - M_1\lambda_{\mathbf{V}} - ab\lambda_{\mathbf{U}} + \frac{\partial l_V\left(t^{post}, \mathbf{V}\right)}{\partial \mathbf{V}} \tag{B.45}
$$

$$
\tau_s \frac{d\lambda_{\mathbf{I}}}{dt} = \lambda_{\mathbf{I}} - \Phi(\mathbf{D}) R \lambda_{\mathbf{V}} \tag{B.46}
$$

$$
\tau_d \frac{d\lambda_{\mathbf{D}}}{dt} = \lambda_{\mathbf{D}} \tag{B.47}
$$

$$
\frac{d\lambda_{\mathbf{U}}}{dt} = a\lambda_{\mathbf{U}} + M_3 \Phi(\mathbf{D}) \lambda_{\mathbf{V}} \tag{B.48}
$$

The loss function $l_p(t^{post})$ is null in the time interval $[t_0; t_1]$ (the inactivity period between the beginning of the simulation and the first spike time emitted by the network). Consequently, all its derivatives in that time gap are null. Therefore the A term could be reduced as:

$$
A = \sum_{k=1}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} \tag{B.49}
$$

Now, B presents a numerical series that could be group as:

$$
\begin{aligned}
& \sum_{k=0}^{N_{post}} a_{k+1}^{-} - a_k^{+} = \\
& = a^{-}\big|_1 - a^{+}\big|_0 + a^{-}\big|_2 - a^{+}\big|_1 + \ldots + \underbrace{a^{-}\big|_{N_{post}+1}}_{=0} - a^{+}\big|_{N_{post}} = \\
& = -\, a^{+}\big|_0 + \sum_{k=1}^{N_{post}} \left[a^{-} - a^{+}\right]\big|_{t_k}
\end{aligned}
\tag{B.50}
$$

Therefore, applying this formalism we obtain:

$$
\begin{aligned}
B = & - l_V^+ \left(t_0^{post}, \mathbf{V}\right) \frac{\partial t_0^{post}}{\partial W_{i,j}} + \sum_{k=1}^{N_{post}} \left[l_V^- \left(t_k^{post}, \mathbf{V}\right) - l_V^+ \left(t_k^{post}, \mathbf{V}\right)\right] \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\
& - \left[\lambda_{\mathbf{V}}^+ \cdot \frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right]_{k=0} + \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{V}}^- \cdot \frac{\partial \mathbf{V}^-}{\partial W_{i,j}} - \lambda_{\mathbf{V}}^+ \cdot \frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right]_{t_k} \\
& - \tau_s \left[\lambda_{\mathbf{I}}^+ \cdot \frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right]_{k=0} + \tau_s \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{I}}^- \cdot \frac{\partial \mathbf{I}^-}{\partial W_{i,j}} - \lambda_{\mathbf{I}}^+ \cdot \frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right]_{t_k} \\
& - \tau_d \left[\lambda_{\mathbf{D}}^+ \cdot \frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right]_{k=0} + \tau_d \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{D}}^- \cdot \frac{\partial \mathbf{D}^-}{\partial W_{i,j}} - \lambda_{\mathbf{D}}^+ \cdot \frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right]_{t_k} \\
& - \left[\lambda_{\mathbf{U}}^+ \cdot \frac{\partial \mathbf{U}^+}{\partial W_{i,j}}\right]_{k=0} + \sum_{k=1}^{N_{post}} \left[\lambda_{\mathbf{U}}^- \cdot \frac{\partial \mathbf{U}^-}{\partial W_{i,j}} - \lambda_{\mathbf{U}}^+ \cdot \frac{\partial \mathbf{U}^+}{\partial W_{i,j}}\right]_{t_k}
\end{aligned}
\tag{B.51}
$$

Now, the starting condition of the internal variable $\mathbf{V}$, $\mathbf{I}$ and $\mathbf{D}$ are parameter independent ($\mathbf{V}|_{k=0} = 0,\ \mathbf{I}|_{k=0} = 0,\ \mathbf{D}|_{k=0} = 0$), therefore:

$$
\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_n \Bigg|_{k=0} = \left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_n \Bigg|_{k=0} = \left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_n \Bigg|_{k=0} = \left(\frac{\partial \mathbf{U}^+}{\partial W_{i,j}}\right)_n \Bigg|_{k=0} = 0
$$

Moreover, the loss function $l_V^+ \left(t_0^{post}, \mathbf{V}\right)$ is null at the beginning of the simulation by definition. Therefore we can re-write the B term as:

$$
\begin{aligned}
B = & \sum_{k=1}^{N_{post}} \left[l_V^- \left(t_k^{post}, \mathbf{V}\right) - l_V^+ \left(t_k^{post}, \mathbf{V}\right)\right] \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\
& \left[\lambda_{\mathbf{V}}^- \cdot \frac{\partial \mathbf{V}^-}{\partial W_{i,j}} - \lambda_{\mathbf{V}}^+ \cdot \frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right]_{t_k} + \\
& \tau_s \left[\lambda_{\mathbf{I}}^- \cdot \frac{\partial \mathbf{I}^-}{\partial W_{i,j}} - \lambda_{\mathbf{I}}^+ \cdot \frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right]_{t_k} + \\
& \tau_d \left[\lambda_{\mathbf{D}}^- \cdot \frac{\partial \mathbf{D}^-}{\partial W_{i,j}} - \lambda_{\mathbf{D}}^+ \cdot \frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right]_{t_k} \\
& \left[\lambda_{\mathbf{U}}^- \cdot \frac{\partial \mathbf{U}^-}{\partial W_{i,j}} - \lambda_{\mathbf{U}}^+ \cdot \frac{\partial \mathbf{U}^+}{\partial W_{i,j}}\right]_{t_k}
\end{aligned}
\tag{B.52}
$$

Now, we can apply the implicit function theorem to the discontinuity equations of all neural state variable (eqs. B.10, B.11, B.12, B.13, B.14, B.15, B.17, B.16 and B.18):

$$
\frac{dt_k}{dW_{i,j}} = - \frac{\left(\frac{\partial(\mathbf{V}^- - \theta)}{\partial W_{i,j}}\right)_{n(k)}}{\left(\frac{\partial(\mathbf{V}^- - \theta)}{\partial t_k}\right)_{n(k)}} = - \frac{\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}}{\left(\dot{\mathbf{V}}^-\right)_{n(k)}}
\tag{B.53}
$$

$$
\frac{dt_k}{dW_{i,j}} = - \frac{\left(\frac{\partial(\mathbf{V}^+ - \gamma)}{\partial W_{i,j}}\right)_{n(k)}}{\left(\frac{\partial(\mathbf{V}^+ - \gamma)}{\partial t_k}\right)_{n(k)}} = - \frac{\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)}}{\left(\dot{\mathbf{V}}^+\right)_{n(k)}}
\tag{B.54}
$$

$$\frac{dt_k}{dW_{i,j}} = -\frac{\left(\frac{\partial(\mathbf{V}^- - \mathbf{V}^+)}{\partial W_{i,j}}\right)_m}{\left(\frac{\partial(\mathbf{V}^- - \mathbf{V}^+)}{\partial t_k}\right)_m} = -\frac{\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_m}{\left(\dot{\mathbf{V}}^-\right)_m - \left(\dot{\mathbf{V}}^+\right)_m} \tag{B.55}$$

Then, taking into account that eq. B.5 is always verified for a non-spiking neuron we can write:

$$\begin{aligned} &\left(\dot{\mathbf{V}}^- - M_0\left(\mathbf{V}^-\right)^2 - M_1\mathbf{V}^- - M_2 + \phi(\mathbf{D}^-)M_3\mathbf{U}^- - \phi(\mathbf{D}^-)R\mathbf{I}^-\right)_m = \\ &\left(\dot{\mathbf{V}}^+ - M_0\left(\mathbf{V}^+\right)^2 - M_1\mathbf{V}^+ - M_2 + \phi(\mathbf{D}^+)M_3\mathbf{U}^+ - \phi(\mathbf{D}^+)R\mathbf{I}^+\right)_m \end{aligned} \tag{B.56}$$

Consequently, considering eq. B.15:

$$\left(\dot{\mathbf{V}}^-\right)_m - \left(\dot{\mathbf{V}}^+\right)_m = -R\phi(\mathbf{D}^-_\mathbf{m})\left(\left(\mathbf{I}^-\right)_m - \left(\mathbf{I}^+\right)_m\right) \tag{B.57}$$

Therefore, substituting eqs. B.11 and B.13:

$$\begin{aligned} \left(\dot{\mathbf{V}}^-\right)_m - \left(\dot{\mathbf{V}}^+\right)_m &= -R\phi\left(\left(\mathbf{D}^-\right)_m\right)\left(\left(\mathbf{I}^-\right)_m + \beta\mathbf{W}_{\mathbf{m.n}} - \left(\mathbf{I}^-\right)_m\right) \\ &= -R\phi\left(\left(\mathbf{D}^-\right)_m\right)\beta\mathbf{W}_{\mathbf{m.n}} \end{aligned} \tag{B.58}$$

Now, we can substitute the result just obtained in eq. B.58 in eq. B.55:

$$\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_m = \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m - R\phi\left(\left(\mathbf{D}^-\right)_m\right)\beta\mathbf{W}_{\mathbf{m.n}}\frac{dt_k}{dW_{i,j}} \tag{B.59}$$

Then, the dt_k/dW can be expressed by considering eq. B.53:

$$\begin{aligned} \frac{dt_k}{dW_{i,j}} &= -\frac{1}{\left(\dot{\mathbf{V}}^-\right)_{n(k)}}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \\ &= -\frac{\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}}{\left(M_0\underbrace{\left(\mathbf{V}^{-2}\right)_{n(k)}}_{=\theta^2} + M_1\underbrace{\left(\mathbf{V}^-\right)_{n(k)}}_{=\theta} + M_2 + \underbrace{\phi\left(\left(\mathbf{D}^-\right)_{n(k)}\right)}_{=1}\left(-M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)\right)} \\ &= -\frac{1}{\left(M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \end{aligned} \tag{B.60}$$

A natural existence condition is $\left(\dot{\mathbf{V}}\right)_{n(k)} \neq 0$ which should be always verified. Considering eq. B.54 we have:

$$
\begin{aligned}
\frac{dt_k}{dW_{i,j}} &= -\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)} \frac{1}{\left(\dot{\mathbf{V}^+}\right)_{n(k)}} \\
&= -\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)} \frac{1}{\left(M_0 \underbrace{\left(\mathbf{V}^+\right)^2_{n(k)}}_{=\gamma^2} + M_1 \underbrace{\left(\mathbf{V}^+\right)_{n(k)}}_{=\gamma} + M_2 + \phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\left(R \underbrace{\left(\mathbf{I}^+\right)_{n(k)}}_{G\left(\mathbf{I}^-\right)_{n(k)}} - M_3 \underbrace{\left(\mathbf{U}^+\right)_{n(k)}}_{\left(\mathbf{U}^-\right)_{n(k)}+d}\right)\right)} \\
&= -\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)} \frac{1}{\left(M_0\gamma^2 + M_1\gamma + M_2 + \phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\left(RG\left(\mathbf{I}^-\right)_{n(k)} - M_3\left(\left(\mathbf{U}^-\right)_{n(k)} + d\right)\right)\right)}
\end{aligned} \tag{B.61}
$$

Substituting B.54 in B.61 we obtain:

$$
\left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}}\right)_{n(k)} = \frac{\left(M_0\gamma^2 + M_1\gamma + M_2 + \phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\left(RG\left(\mathbf{I}^-\right)_{n(k)} - M_3\left(\left(\mathbf{U}^-\right)_{n(k)} + d\right)\right)\right)}{\left(M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \tag{B.62}
$$

By applying the definition of partial derivative to eq. B.12 we obtain:

$$
\begin{aligned}
\frac{d\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{dW_{i,j}} &= \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial W_{i,j}} \underbrace{\frac{dW_{i,j}}{dW_{i,j}}}_{=1} + \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial t_k} \frac{dt_k}{dW_{i,j}} \\
&= \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial W_{i,j}} + \left(\left(\dot{\mathbf{I}^+}\right)_{n(k)} - G\left(\dot{\mathbf{I}^-}\right)_{n(k)}\right) \frac{dt_k}{dW_{i,j}} \\
&= \frac{\partial\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}{\partial W_{i,j}} + \frac{1}{\tau_s} \underbrace{\left(\left(\mathbf{I}^+\right)_{n(k)} - G\left(\mathbf{I}^-\right)_{n(k)}\right)}_{=0} \frac{dt_k}{dW_{i,j}}
\end{aligned} \tag{B.63}
$$

This means that the total derivative is equal to the partial derivative, consequently we have:

$$
\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_{n(k)} = G\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_{n(k)} \tag{B.64}
$$

By applying the implicit function theorem to eq. B.13 we obtain:

$$\begin{aligned}
\frac{dt_k}{dW_{i,j}} &= -\frac{\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m - \beta\left(\frac{\partial W_{m,n}}{\partial W_{i,j}}\right)_m}{\underbrace{\left(\frac{\partial \mathbf{I}^+}{\partial t_k}\right)_m}_{(\dot{\mathbf{I}^+})_m} - \underbrace{\left(\frac{\partial \mathbf{I}^-}{\partial t_k}\right)_m}_{(\dot{\mathbf{I}^-})_m} - \beta \underbrace{\left(\frac{\partial W_{m,n}}{\partial t_k}\right)_m}_{=0}} \\
&= -\tau_s \frac{\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m - \beta\delta_{i,m}\delta_{j,n}}{-\left(\mathbf{I}^+\right)_m + \left(\mathbf{I}^-\right)_m} \\
&= \tau_s \frac{\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_m - \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m - \beta\delta_{i,m}\delta_{j,n}}{\beta W_{m,n}}
\end{aligned} \tag{B.65}$$

Therefore, we can rewrite the equation below as:

$$\left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}}\right)_m = \frac{\beta W_{m,n}}{\tau_s}\frac{dt_k}{dW_{i,j}} + \beta\delta_{i,m}\delta_{j,n} + \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m \tag{B.66}$$

We applied the implicit function theorem on eq. B.14:

$$\frac{dt_k}{dW_{i,j}} = -\frac{\left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_{n(k)}}{\underbrace{\left(\frac{\partial \mathbf{D}^+}{\partial t_k}\right)_{n(k)}}_{(\dot{\mathbf{D}^+})_m}} = \tau_D \frac{\left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_{n(k)}}{\left(\mathbf{D}^+\right)_{n(k)}} \tag{B.67}$$

And consequently we have:

$$\begin{aligned}
\left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_{n(k)} &= \frac{1}{\tau_D}\frac{dt_k}{dW_{i,j}} \underbrace{\left(\mathbf{D}^+\right)_{n(k)}}_{=\epsilon} \\
&= -\frac{\epsilon}{\tau_D\left(M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}
\end{aligned} \tag{B.68}$$

While the m-component is similar to eq. B.64 and corresponds to:

$$\left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}}\right)_m = \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_m \tag{B.69}$$

By applying the chain rule to eq. B.16 we obtain:

$$\begin{aligned}
0 = \frac{d\left[(\mathbf{U}^+)_m - (\mathbf{U}^-)_m\right]}{dW_{i,j}} &= \frac{\partial\left[(\mathbf{U}^+)_m - (\mathbf{U}^-)_m\right]}{\partial W_{i,j}}\frac{\partial W_{i,j}}{\partial W_{i,j}} + \frac{\partial}{\partial t_k}\left[\left(\mathbf{U}^+\right)_m - \left(\mathbf{U}^-\right)_m\right]\frac{\partial t_k}{\partial W_{i,j}} \\
&= \frac{\partial\left(\mathbf{U}^+\right)_m}{\partial W_{i,j}} - \frac{\partial\left(\mathbf{U}^-\right)_m}{\partial W_{i,j}} + \left[\left(\dot{\mathbf{U}}^+\right)_m - \left(\dot{\mathbf{U}}^-\right)_m\right]\frac{\partial t_k}{\partial W_{i,j}} \\
&= \frac{\partial\left(\mathbf{U}^+\right)_m}{\partial W_{i,j}} - \frac{\partial\left(\mathbf{U}^-\right)_m}{\partial W_{i,j}} + \underbrace{\left[a\left(b\left(\mathbf{V}^+\right)_m - \left(\mathbf{U}^+\right)_m\right) - a\left(b\left(\mathbf{V}^-\right)_m - \left(\mathbf{U}^-\right)_m\right)\right]}_{=0}\frac{\partial t_k}{\partial W_{i,j}}
\end{aligned} \tag{B.70}$$

Therefore, we obtain:

$$\frac{\partial\left(\mathbf{U}^{+}\right)_{m}}{\partial W_{i,j}}=\frac{\partial\left(\mathbf{U}^{+}\right)_{m}}{\partial W_{i,j}} \tag{B.71}$$

By applying the implicit function theorem to B.17 to the $(I)_{n(k)}$ component:

$$\begin{aligned}
\frac{\partial t_k}{\partial W_{i,j}} &= -\frac{\frac{\partial}{\partial W_{i,j}}\left[(\mathbf{U}^{+})_{n(k)}-(\mathbf{U}^{-})_{n(k)}-d\right]}{\frac{\partial}{\partial t_k}\left[(\mathbf{U}^{+})_{n(k)}-(\mathbf{U}^{-})_{n(k)}-d\right]} \\
&= -\frac{\frac{\partial\left(\mathbf{U}^{+}\right)_{n(k)}}{\partial W_{i,j}}-\frac{\partial\left(\mathbf{U}^{-}\right)_{n(k)}}{\partial W_{i,j}}}{\left(\dot{\mathbf{U}}^{+}\right)_{n(k)}-\left(\dot{\mathbf{U}}^{-}\right)_{n(k)}} \\
&= -\frac{\frac{\partial\left(\mathbf{U}^{+}\right)_{n(k)}}{\partial W_{i,j}}-\frac{\partial\left(\mathbf{U}^{-}\right)_{n(k)}}{\partial W_{i,j}}}{a\left(b\underbrace{\left(\mathbf{V}^{+}\right)_{n(k)}}_{=\gamma}-\underbrace{\left(\mathbf{U}^{+}\right)_{n(k)}}_{(\mathbf{U}^{-})_{n(k)}+d}\right)-a\left(b\underbrace{\left(\mathbf{V}^{-}\right)_{n(k)}}_{\theta}-(\mathbf{U}^{-})_{n(k)}\right)} \\
&= \frac{\frac{\partial\left(\mathbf{U}^{+}\right)_{n(k)}}{\partial W_{i,j}}-\frac{\partial\left(\mathbf{U}^{-}\right)_{n(k)}}{\partial W_{i,j}}}{-ab\left(\gamma-\theta\right)+ad}
\end{aligned} \tag{B.72}$$

Then, combining eqs. B.73, B.60,B.68 and B.66, we obtain:

$$\begin{aligned}
\left(\frac{\partial \mathbf{V}^{+}}{\partial W_{i,j}}\right)_m &= \left(\frac{\partial \mathbf{V}^{-}}{\partial W_{i,j}}\right)_m - R\phi\left(\left(\mathbf{D}^{-}\right)_m\right)\beta W_{m,n}\frac{dt_k}{dW_{i,j}} \\
&= \left(\frac{\partial \mathbf{V}^{-}}{\partial W_{i,j}}\right)_m + \frac{R\phi\left(\left(\mathbf{D}^{-}\right)_m\right)\beta W_{m,n}}{\left(M_0\theta^2+M_1\theta+M_2+-M_3\left(\mathbf{U}^{-}\right)_{n(k)}+R\left(\mathbf{I}^{-}\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^{-}}{\partial W_{i,j}}\right)_{n(k)}
\end{aligned} \tag{B.73}$$

$$\begin{aligned}
\left(\frac{\partial \mathbf{I}^{+}}{\partial W_{i,j}}\right)_m = &-\frac{\beta W_{m,n}}{\tau_s\left(M_0\theta^2+M_1\theta+M_2+-M_3\left(\mathbf{U}^{-}\right)_{n(k)}+R\left(\mathbf{I}^{-}\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^{-}}{\partial W_{i,j}}\right)_{n(k)}+ \\
&\beta\delta_{i,m}\delta_{j,n}+\left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_m
\end{aligned} \tag{B.74}$$

$$\left(\frac{\partial \mathbf{U}^{+}}{\partial W_{i,j}}\right)_{n(k)} = -\frac{ad-ab\left(\gamma-\theta\right)}{\left(M_0\theta^2+M_1\theta+M_2+-M_3\left(\mathbf{U}^{-}\right)_{n(k)}+R\left(\mathbf{I}^{-}\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^{-}}{\partial W_{i,j}}\right)_{n(k)}+\left(\frac{\partial \mathbf{I}^{-}}{\partial W_{i,j}}\right)_{n(k)} \tag{B.75}$$

Now, considering the inner products of eq. B.52 we can expand each term as represented below:

$$\lambda \cdot \mathbf{d} = (\lambda)_{n(k)} (\mathbf{d})_{n(k)} + \sum_{m \neq n(k)}^{N_n - 1} (\lambda)_m (\mathbf{d})_m$$

Consequently, eq. B.51 expands in:

$$\begin{aligned} B = & \sum_{k=1}^{N_{post}} \left[l_V^- \left(t_k^{post}, \mathbf{V} \right) - l_V^+ \left(t_k^{post}, \mathbf{V} \right) \right] \frac{\partial t_k^{post}}{\partial W_{i,j}} + \\ & + \left(\lambda_{\mathbf{V}}^-\right)_{n(k)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_{n(k)} + \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{V}}^-\right)_m \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}} \right)_m + \\ & - \left(\lambda_{\mathbf{V}}^+\right)_{n(k)} \left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}} \right)_{n(k)} - \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{V}}^+\right)_m \left(\frac{\partial \mathbf{V}^+}{\partial W_{i,j}} \right)_m + \\ & + \tau_s \left(\lambda_{\mathbf{I}}^-\right)_{n(k)} \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}} \right)_{n(k)} + \tau_s \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{I}}^-\right)_m \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}} \right)_m + \\ & - \tau_s \left(\lambda_{\mathbf{I}}^+\right)_{n(k)} \left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}} \right)_{n(k)} - \tau_s \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{I}}^+\right)_m \left(\frac{\partial \mathbf{I}^+}{\partial W_{i,j}} \right)_m + \\ & \tau_D \left(\lambda_{\mathbf{D}}^-\right)_{n(k)} \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}} \right)_{n(k)} + \tau_D \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{D}}^-\right)_m \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}} \right)_m + \\ & - \tau_D \left(\lambda_{\mathbf{D}}^+\right)_{n(k)} \left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}} \right)_{n(k)} - \tau_D \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{D}}^+\right)_m \left(\frac{\partial \mathbf{D}^+}{\partial W_{i,j}} \right)_m + \\ & + \left(\lambda_{\mathbf{U}}^-\right)_{n(k)} \left(\frac{\partial \mathbf{U}^-}{\partial W_{i,j}} \right)_{n(k)} + \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{U}}^-\right)_m \left(\frac{\partial \mathbf{U}^-}{\partial W_{i,j}} \right)_m + \\ & - \left(\lambda_{\mathbf{U}}^+\right)_{n(k)} \left(\frac{\partial \mathbf{U}^+}{\partial W_{i,j}} \right)_{n(k)} - \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{U}}^+\right)_m \left(\frac{\partial \mathbf{U}^+}{\partial W_{i,j}} \right)_m + \end{aligned} \tag{B.76}$$

$$
\begin{aligned}
B = &\sum_{k=1}^{N_{post}} \left[l_V^- \left(t_k^{post}, \mathbf{V}\right) - l_V^+ \left(t_k^{post}, \mathbf{V}\right)\right] \frac{1}{\left(M_0\theta^2 + M_1\theta + M_2 + -M_3 \left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \\
&+ \left(\lambda_{\mathbf{V}}^-\right)_{n(k)} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \sum_{m\neq n(k)}^{N_n-1} \left(\lambda_{\mathbf{V}}^-\right)_m \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m + \\
&- \left(\lambda_{\mathbf{V}}^+\right)_{n(k)} \frac{M_0\gamma^2 + M_1\gamma + M_2 + \Phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\left[-M_3\left(\left(\mathbf{U}^-\right)_{n(k)} + d\right) + RG\left(\mathbf{I}^-\right)_{n(k)}\right]}{M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \\
&- \sum_{m\neq n(k)}^{N_n-1} \left(\lambda_{\mathbf{V}}^+\right)_m \left[\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m + \frac{R\beta W_{m,n}\Phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)}{M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}\right] + \\
&+ \tau_s\left(\lambda_{\mathbf{I}}^-\right)_{n(k)}\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_{n(k)} + \tau_s\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{I}}^-\right)_m\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m - \tau_s G\left(\lambda_{\mathbf{I}}^+\right)_{n(k)}\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_{n(k)} + \\
&- \tau_s\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{I}}^+\right)_m\left[-\frac{\beta W_{m,n}}{\tau_s\left(M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \right. \\
&\left. + \beta\delta_{i,m}\delta_{j,n} + \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m\right] + \tau_D\left(\lambda_{\mathbf{D}}^-\right)_{n(k)}\left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_{n(k)} + \tau_D\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{D}}^-\right)_m\left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_m + \\
&+ \left(\lambda_{\mathbf{D}}^+\right)_{n(k)}\frac{\epsilon}{M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} + \\
&- \tau_D\sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{D}}^+\right)_m\left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_m + \left(\lambda_{\mathbf{U}}^-\right)_{n(k)}\left(\frac{\partial \mathbf{U}^-}{\partial W_{i,j}}\right)_{n(k)} + \sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{U}}^-\right)_m\left(\frac{\partial \mathbf{U}^-}{\partial W_{i,j}}\right)_m + \\
&- \left(\lambda_U^+\right)_{n(k)}\left[\left(\frac{\partial \mathbf{U}^-}{\partial W_{i,j}}\right)_{n(k)} - \frac{ad - ab\left(\gamma - \theta\right)}{M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}\right] + \\
&- \sum_{m\neq n(k)}^{N_n-1}\left(\lambda_{\mathbf{U}}^-\right)_m\left(\frac{\partial \mathbf{U}^-}{\partial W_{i,j}}\right)_m
\end{aligned}
\tag{B.77}
$$

$$
\begin{aligned}
A = &\sum_{k=1}^{N_{post}} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \frac{\partial t_k^{post}}{\partial W_{i,j}} = \\
= &- \sum_{k=1}^{N_{post}} \frac{1}{\left(M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right)} \frac{\partial l_p(t^{post})}{\partial t_k^{post}} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)}
\end{aligned}
\tag{B.78}
$$

In the following equation gathered different terms:

$$\frac{d\mathscr{L}}{dW_{i,j}} = \sum_{k=1}^{N_{post}} \frac{1}{M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}} \left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_{n(k)} \Bigg[- \frac{\partial l_p(t^{post})}{\partial t_k^{post}} + \left(l_V^-\left(t_k^{post}, \mathbf{V}\right) - l_V^+\left(t_k^{post}, \mathbf{V}\right)\right) + \left(\lambda_{\mathbf{V}}^-\right)_{n(k)}\left(M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^-\right)_{n(k)} + R\left(\mathbf{I}^-\right)_{n(k)}\right) + - \left(\lambda_{\mathbf{V}}^+\right)_{n(k)}\left(M_0\gamma^2 + M_1\gamma + M_2 + \Phi\left(\left(\mathbf{D}^+\right)_{n(k)}\right)\left[-M_3\left(\left(\mathbf{U}^-\right)_{n(k)} + d\right) + RG\left(\mathbf{I}^-\right)_{n(k)}\right]\right)\left(\mathbf{I}^-\right)_{n(k)} + \sum_{m \neq n(k)}^{N_n - 1} \beta W_{m,n}\left(\left(\lambda_{\mathbf{I}}^+\right)_m - \left(\lambda_{\mathbf{V}}^+\right)_m R\phi\left(\left(\mathbf{D}^+\right)_m\right)\right) + \left(\lambda_D^+\right)_{n(k)}\epsilon + \left(\lambda_{\mathbf{U}}^+\right)_{n(k)}\left(ad - ab\left(\gamma - \theta\right)\right)\Bigg] + \sum_{m \neq n(k)}^{N_n - 1}\left(\frac{\partial \mathbf{V}^-}{\partial W_{i,j}}\right)_m \tau_m\left[\left(\lambda_{\mathbf{V}}^-\right)_m - \left(\lambda_{\mathbf{V}}^+\right)_m\right] + \sum_{m \neq n(k)}^{N_n - 1}\left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_m \tau_s\left[\left(\lambda_{\mathbf{I}}^-\right)_m - \left(\lambda_{\mathbf{I}}^+\right)_m\right] + \left(\frac{\partial \mathbf{I}^-}{\partial W_{i,j}}\right)_{n(k)} \tau_s\left[\left(\lambda_{\mathbf{I}}^-\right)_{n(k)} - G\left(\lambda_{\mathbf{I}}^+\right)_{n(k)}\right] + \left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_{n(k)} \tau_D\left(\lambda_{\mathbf{D}}^-\right)_{n(k)} + \sum_{m \neq n(k)}^{N_n - 1}\left(\frac{\partial \mathbf{D}^-}{\partial W_{i,j}}\right)_m \tau_D\left[\left(\lambda_{\mathbf{D}}^-\right)_m - \left(\lambda_{\mathbf{D}}^+\right)_m\right] + - \sum_{m \neq n(k)}^{N_n - 1} \tau_s \beta \delta_{i,m}\delta_{j,n}\left(\lambda_{\mathbf{I}}^-\right)_{n(k)} \tag{B.79}$$

In order to delete as much term possible we can define the following continuity conditions:

$$\left(\lambda_{\mathbf{V}}^-\right)_m = \left(\lambda_{\mathbf{V}}^+\right)_m \tag{B.80}$$

$$\left(\lambda_{\mathbf{I}}^-\right)_m = \left(\lambda_{\mathbf{I}}^+\right)_m \tag{B.81}$$

$$\left(\lambda_{\mathbf{I}}^-\right)_{n(k)} = G\left(\lambda_{\mathbf{I}}^+\right)_m \tag{B.82}$$

$$\left(\lambda_{\mathbf{D}}^-\right)_m = \left(\lambda_{\mathbf{D}}^+\right)_m \tag{B.83}$$

$$\left(\lambda_{\mathbf{D}}^-\right)_{n(k)} = 0 \tag{B.84}$$

$$\left(\lambda_{\mathbf{U}}^-\right)_m = \left(\lambda_{\mathbf{U}}^+\right)_m \tag{B.85}$$

$$\left(\lambda_{\mathbf{U}}^-\right)_{n(k)} = \left(\lambda_{\mathbf{U}}^+\right)_m \tag{B.86}$$

$$
\begin{aligned}
\left(\lambda_{\mathbf{V}}^{-}\right)_{n(k)} =& \frac{1}{M_0\theta^2 + M_1\theta + M_2 + -M_3\left(\mathbf{U}^{-}\right)_{n(k)} + R\left(\mathbf{I}^{-}\right)_{n(k)}} \left[\frac{\partial l_p(t^{post})}{\partial t_k^{post}} + \left(l_V^{+}\left(t_k^{post}, \mathbf{V}\right) - l_V^{-}\left(t_k^{post}, \mathbf{V}\right)\right) + \right. \\
& \left(\lambda_{\mathbf{V}}^{+}\right)_{n(k)} \left(M_0\gamma^2 + M_1\gamma + M_2 + \Phi\left(\left(\mathbf{D}^{+}\right)_{n(k)}\right)\left[-M_3\left(\left(\mathbf{U}^{-}\right)_{n(k)} + d\right) + RG\left(\mathbf{I}^{-}\right)_{n(k)}\right]\right) + \\
& \left. \sum_{m \neq n(k)}^{N_n - 1} \beta W_{m,n}\left(\left(\lambda_{\mathbf{V}}^{+}\right)_m R\phi\left(\left(\mathbf{D}^{+}\right)_m\right) - \left(\lambda_{\mathbf{I}}^{+}\right)_m\right) + \left(\lambda_D^{+}\right)_{n(k)}\epsilon - \left(\lambda_{\mathbf{U}}^{+}\right)_{n(k)}\left(ad - ab\left(\gamma - \theta\right)\right)\right]
\end{aligned}
\tag{B.87}
$$

And consequently eq. B.79 reduces to:

$$
\begin{aligned}
\frac{d\mathscr{L}}{dW_{i,j}} &= -\tau_s \sum_{k=1}^{N_{post}} \sum_{m \neq n(k)}^{N_n - 1} \left(\lambda_{\mathbf{I}}^{+}\right)_m \beta\delta_{i,m}\delta_{j,n} = \\
&= -\tau_s \sum_{spikes from j} \left(\lambda_{\mathbf{I}_i}^{+}\right)_m \beta
\end{aligned}
\tag{B.88}
$$

Appendix C

List of publications

C.1 Journal papers

Fontanini, R., Segatto, M., Nair, K.S., Holzer, M., Driussi, F., Hausler, I., Koch, C.T., Dubourdieu, C., Deshpande, V., Esseni, D.
Charge-Trapping-Induced Compensation of the Ferroelectric Polarization in FTJs: Optimal Conditions for a Synaptic Device Operation
(2022) IEEE Transactions on Electron Devices, 69 (7), pp. 3694-3699. DOI: 10.1109/TED.2022.3175684

Fontanini, R., Barbot, J., Segatto, M., Lancaster, S., Duong, Q., Driussi, F., Grenouillet, L., Triozon, L., Coignus, J., Mikolajick, T., Slesazeck, S., Esseni, D.
Interplay between Charge Trapping and Polarization Switching in BEOL-Compatible Bilayer Ferroelectric Tunnel Junctions
(2022) IEEE Journal of the Electron Devices Society, 10, pp. 593-599. DOI: 10.1109/JEDS.2022.3171217

Segatto, M., **Fontanini, R.**, Driussi, F., Lizzit, D., Esseni, D.
Limitations to Electrical Probing of Spontaneous Polarization in Ferroelectric-Dielectric Heterostructures
(2022) IEEE Journal of the Electron Devices Society, 10, pp. 324-333. DOI: 10.1109/JEDS.2022.3164652

Hoffmann, M., Gui, M., Slesazeck, S., **Fontanini, R.**, Segatto, M., Esseni, D., Mikolajick, T.
Intrinsic Nature of Negative Capacitance in Multidomain Hf0.5Zr0.5O2-Based Ferroelectric/Dielectric Heterostructures
(2022) Advanced Functional Materials, 32 (2), art. no. 2108494,. DOI: 10.1002/adfm.202108494

Esseni, D., **Fontanini, R.**
Macroscopic and microscopic picture of negative capacitance operation in ferroelectric capacitors
(2021) Nanoscale, 13 (21), pp. 9641-9650. DOI: 10.1039/d0nr06886a

Fontanini, R., Segatto, M., Massarotto, M., Specogna, R., Driussi, F., Loghi, M., Esseni, D.E.
Modeling and Design of FTJs as Multi-Level Low Energy Memristors for Neuromorphic Computing

(2021) IEEE Journal of the Electron Devices Society, 9, pp. 1202-1209. DOI: 10.1109/JEDS.2021.3120200

C.2 Conference papers

Fontanini, R., Esseni, D., Loghi, M.
Reducing the Spike Rate in Deep Spiking Neural Networks
(2022) ACM International Conference Proceeding Series, art. no. 8, . DOI: 10.1145/3546790.3546798

Segatto, M., Massarotto, M., Lancaster, S., Duong, Q.T., Affanni, A., **Fontanini, R.**, Driussi, F., Lizzit, D., Mikolajick, T., Slesazeck, S., Esseni, D.
Polarization switching and AC small-signal capacitance in Ferroelectric Tunnel Junctions
(2022) European Solid-State Device Research Conference, 2022-September, pp. 340-343. DOI: 10.1109/ESSDERC55479.2022.9947185

Esseni, D., **Fontanini, R.**, Lizzit, D., Massarotto, M., Driussi, F., Loghi, M.
Ferroelectric based FETs and synaptic devices for highly energy efficient computational technologies
(2021) 2021 5th IEEE Electron Devices Technology and Manufacturing Conference, EDTM 2021, art. no. 9420848, . DOI: 10.1109/EDTM50988.2021.9420848

Fontanini, R., Massarotto, M., Specogna, R., Driussi, F., Loghi, M., Esseni, D.
Modelling and design of FTJs as high reading-impedance synaptic devices
(2021) 2021 5th IEEE Electron Devices Technology and Manufacturing Conference, EDTM 2021, art. no. 9420863, . DOI: 10.1109/EDTM50988.2021.9420863

Fontanini, R., Barbot, J., Segatto, M., Lancaster, S., Duong, Q., Driussi, F., Grenouillet, L., Triozon, F., Coignus, J., Mikolajick, T., Slesazeck, S., Esseni, D.
Polarization switching and interface charges in BEOL compatible Ferroelectric Tunnel Junctions
(2021) European Solid-State Device Research Conference, 2021-September, pp. 255-258. DOI: 10.1109/ESSDERC53440.2021.9631812

Bibliography

[1] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, Dec 1943. doi: 10.1007/BF02478259. [Online]. Available: https://doi.org/10.1007/BF02478259

[2] J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon, "A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955," *AI Magazine*, vol. 27, no. 4, p. 12, Dec. 2006. doi: 10.1609/aimag.v27i4.1904. [Online]. Available: https://ojs.aaai.org/index.php/aimagazine/article/view/1904

[3] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968. doi: 10.1109/TSSC.1968.300136

[4] J. McCarthy and P. J. Hayes, "Some philosophical problems from the standpoint of artificial intelligence," pp. 463–502, 1969, reprinted in McC90.

[5] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 12 1989. doi: 10.1162/neco.1989.1.4.541. [Online]. Available: https://doi.org/10.1162/neco.1989.1.4.541

[6] J. J. Hopfield and D. W. Tank, ""neural" computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, no. 3, pp. 141–152, Jul 1985. doi: 10.1007/BF00339943. [Online]. Available: https://doi.org/10.1007/BF00339943

[7] V. Vapnik, *The Nature of Statistical Learning Theory*, ser. Information Science and Statistics. Springer New York, 2013. ISBN 9781475732641. [Online]. Available: https://books.google.it/books?id=EqgACAAAQBAJ

[8] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] P. Domingos, "A few useful things to know about machine learning," *Commun. ACM*, vol. 55, no. 10, p. 78–87, oct 2012. doi: 10.1145/2347736.2347755. [Online]. Available: https://doi.org/10.1145/2347736.2347755

[10] A. Ng and M. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Advances in Neural Information Processing Systems*, T. Dietterich, S. Becker, and Z. Ghahramani, Eds., vol. 14.

MIT Press, 2001. [Online]. Available: https://proceedings.neurips.cc/paper/2001/file/7b7a53e239400a13bd6be6c91c4f6c4e-Paper.pdf

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[13] G. Indiveri, B. Linares-Barranco, T. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. SAÏGHI, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen, "Neuromorphic silicon neuron circuits," *Frontiers in Neuroscience*, vol. 5, 2011. doi: 10.3389/fnins.2011.00073. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2011.00073

[14] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423 pp. 4171–4186. [Online]. Available: https://doi.org/10.18653/v1/n19-1423

[15] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, no. 7873, pp. 583–589, Aug 2021. doi: 10.1038/s41586-021-03819-2. [Online]. Available: https://doi.org/10.1038/s41586-021-03819-2

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[17] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. doi: 10.1109/5.726791

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: https://arxiv.org/abs/1409.1556

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015. doi: 10.1007/s11263-015-0816-y

[21] J. Z. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel, “Multi-agent reinforcement learning in sequential social dilemmas,” in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS ’17. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2017, p. 464–473.

[22] P. J. Navarro, L. Miller, F. Rosique, C. Fernández-Isla, and A. Gila-Navarro, “End-to-end deep neural network architectures for speed and steering wheel angle prediction in autonomous driving,” *Electronics*, vol. 10, no. 11, 2021. doi: 10.3390/electronics10111266. [Online]. Available: https://www.mdpi.com/2079-9292/10/11/1266

[23] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, “Language models are unsupervised multitask learners,” *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[24] G. Hinton, L. Deng, D. Yu, G. Dahl, A. rahman Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, “Deep neural networks for acoustic modeling in speech recognition,” *Signal Processing Magazine*, 2012.

[25] D. Ielmini and H.-S. P. Wong, “In-memory computing with resistive switching devices,” *Nature Electronics*, vol. 1, no. 6, pp. 333–343, Jun 2018. doi: 10.1038/s41928-018-0092-2. [Online]. Available: https://doi.org/10.1038/s41928-018-0092-2

[26] W. A. Wulf and S. A. McKee, “Hitting the memory wall: Implications of the obvious,” *SIGARCH Comput. Archit. News*, vol. 23, no. 1, p. 20–24, mar 1995. doi: 10.1145/216585.216588. [Online]. Available: https://doi.org/10.1145/216585.216588

[27] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, Oct 1986. doi: 10.1038/323533a0. [Online]. Available: https://doi.org/10.1038/323533a0

[28] CISCO, “Cisco annual internet report,” 2022. [Online]. Available: https://cisco.com

[29] E. Covi, Q. T. Duong, S. Lancaster, V. Havel, J. Coignus, J. Barbot, O. Richter, P. Klein, E. Chicca, L. Grenouillet, A. Dimoulas, T. Mikolajick, and S. Slesazeck, “Ferroelectric tunneling junctions for edge computing,” in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021. doi: 10.1109/ISCAS51556.2021.9401800 pp. 1–5.

[30] Z. Wang, M. Jiang, Y. Hu, and H. Li, “An incremental learning method based on probabilistic neural networks and adjustable fuzzy clustering for human activity recognition by using wearable sensors,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 4, pp. 691–699, 2012. doi: 10.1109/TITB.2012.2196440

[31] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016. doi: 10.1109/TIV.2016.2578706

[32] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, "A survey of deep learning methods for cyber security," *Information*, vol. 10, no. 4, 2019. doi: 10.3390/info10040122. [Online]. Available: https://www.mdpi.com/2078-2489/10/4/122

[33] T. Starner, J. Weaver, and A. Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1371–1375, 1998. doi: 10.1109/34.735811

[34] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014. doi: 10.1109/CVPR.2014.220 pp. 1701–1708.

[35] N. Scheiner, N. Appenrodt, J. Dickmann, and B. Sick, "Radar-based road user classification and novelty detection with recurrent neural network ensembles," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019. doi: 10.1109/IVS.2019.8813773 pp. 722–729.

[36] L. Floridi and M. Chiriatti, "GPT-3: Its Nature, Scope, Limits, and Consequences," *Minds and Machines*, vol. 30, no. 4, pp. 681–694, 2020. doi: 10.1007/s11023-020-09548-1

[37] F. Tavella, A. Galata, and A. Cangelosi, "Phonology recognition in american sign language," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022. doi: 10.1109/ICASSP43922.2022.9747212 pp. 8452–8456.

[38] F. Tavella, A. Giaretta, M. Conti, and S. Balasubramaniam, "A machine learning-based approach to detect threats in bio-cyber dna storage systems," *Computer Communications*, vol. 187, pp. 59–70, 2022. doi: https://doi.org/10.1016/j.comcom.2022.01.023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366422000378

[39] G. Indiveri, "Introducing 'neuromorphic computing and engineering'," *Neuromorphic Computing and Engineering*, vol. 1, no. 1, p. 010401, jul 2021. doi: 10.1088/2634-4386/ac0a5b. [Online]. Available: https://dx.doi.org/10.1088/2634-4386/ac0a5b

[40] L. Kováč, "The 20 W sleep-walkers," *EMBO reports*, vol. 11, no. 1, pp. 2–2, 2010. doi: https://doi.org/10.1038/embor.2009.266. [Online]. Available: https://www.embopress.org/doi/abs/10.1038/embor.2009.266

[41] E. Chicca and G. Indiveri, "A recipe for creating ideal hybrid memristive-cmos neuromorphic processing systems," *Applied Physics Letters*, vol. 116, no. 12, p. 120501, 2020. doi: 10.1063/1.5142089. [Online]. Available: https://doi.org/10.1063/1.5142089

[42] C. Mead, "Analog vlsi and neural systems," 1989.

[43] E. Covi, E. Donati, X. Liang, D. Kappel, H. Heidari, M. Payvand, and W. Wang, "Adaptive extreme edge computing for wearable devices," *Frontiers in Neuroscience*, vol. 15, 2021. doi: 10.3389/fnins.2021.611300. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2021.611300

[44] Y. Xing, L. Zhang, Z. Hou, X. Li, Y. Shi, Y. Yuan, F. Zhang, S. Liang, Z. Li, and L. Yan, "Accurate ECG Classification Based on Spiking Neural Network and Attentional Mechanism for Real-Time Implementation on Personal Portable Devices," *Electronics*, vol. 11, no. 12, 2022. doi: 10.3390/electronics11121889. [Online]. Available: https://www.mdpi.com/2079-9292/11/12/1889

[45] E. J. da S. Luz, W. R. Schwartz, G. Cámara-Chávez, and D. Menotti, "ECG-based heartbeat classification for arrhythmia detection: A survey," *Computer Methods and Programs in Biomedicine*, vol. 127, pp. 144–164, 2016. doi: https://doi.org/10.1016/j.cmpb.2015.12.008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260715003314

[46] K.-Q. Shen, X.-P. Li, C.-J. Ong, S.-Y. Shao, and E. P. Wilder-Smith, "EEG-based mental fatigue measurement using multi-class support vector machines with confidence estimate," *Clinical Neurophysiology*, vol. 119, no. 7, pp. 1524–1533, 2008. doi: https://doi.org/10.1016/j.clinph.2008.03.012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1388245708002034

[47] C.-T. Lin, C.-H. Chuang, C.-S. Huang, S.-F. Tsai, S.-W. Lu, Y.-H. Chen, and L.-W. Ko, "Wireless and Wearable EEG System for Evaluating Driver Vigilance," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 2, pp. 165–176, 2014. doi: 10.1109/TBCAS.2014.2316224

[48] X.-W. Wang, D. Nie, and B.-L. Lu, "Emotional state classification from EEG data using machine learning approach," *Neurocomputing*, vol. 129, pp. 94–106, 2014. doi: https://doi.org/10.1016/j.neucom.2013.06.046. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231213009867

[49] C. Cipriani, F. Zaccone, S. Micera, and M. C. Carrozza, "On the shared control of an EMG-controlled prosthetic hand: analysis of user–prosthesis interaction," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 170–184, 2008.

[50] I. Moon, M. Lee, J. Chu, and M. Mun, "Wearable EMG-based HCI for electric-powered wheelchair users with motor disabilities," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation.* IEEE, 2005, pp. 2649–2654.

[51] I. Mendez, B. W. Hansen, C. M. Grabow, E. J. L. Smedegaard, N. B. Skogberg, X. J. Uth, A. Bruhn, B. Geng, and E. N. Kamavuako, "Evaluation of the Myo armband for the classification of hand motions," in *2017 International Conference on Rehabilitation Robotics (ICORR).* IEEE, 2017, pp. 1211–1214.

[52] A. L. Rincon, H. Yamasaki, and S. Shimoda, "Design of a video game for rehabilitation using motion capture, EMG analysis and virtual reality," in *2016 International Conference on Electronics, Communications and Computers (CONIELECOMP).* IEEE, 2016, pp. 198–204.

[53] J. R. Matthie, "Bioimpedance measurements of human body composition: critical analysis and outlook," *Expert review of medical devices*, vol. 5, no. 2, pp. 239–261, 2008.

[54] T.-P. Sun, C. T.-S. Ching, C.-S. Cheng, S.-H. Huang, Y.-J. Chen, C.-S. Hsiao, C.-H. Chang, S.-Y. Huang, H.-L. Shieh, W.-H. Liu *et al.*, "The use of bioimpedance in the detection/screening of tongue cancer," *Cancer Epidemiology*, vol. 34, no. 2, pp. 207–211, 2010.

[55] J. Wu, E. Yılmaz, M. Zhang, H. Li, and K. C. Tan, "Deep spiking neural networks for large vocabulary automatic speech recognition," *Frontiers in Neuroscience*, vol. 14, 2020. doi: 10.3389/fnins.2020.00199. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2020.00199

[56] G. Datta, S. Kundu, A. R. Jaiswal, and P. A. Beerel, "Ace-snn: Algorithm-hardware co-design of energy-efficient & low-latency deep spiking neural networks for 3d image recognition," *Frontiers in Neuroscience*, vol. 16, 2022. doi: 10.3389/fnins.2022.815258. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2022.815258

[57] R. Kabilan and N. Muthukumaran, "A Neuromorphic Model for Image Recognition using SNN," in *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, 2021. doi: 10.1109/ICICT50816.2021.9358663 pp. 720–725.

[58] E. Sadovsky, R. Jarina, and R. Orjesek, "Image recognition using spiking neural networks," in *2021 31st International Conference Radioelektronika (RADIOELEKTRONIKA)*, 2021. doi: 10.1109/RADIOELEKTRONIKA52220.2021.9420192 pp. 1–5.

[59] S. Xiang, S. Jiang, X. Liu, T. Zhang, and L. Yu, "Spiking vgg7: Deep convolutional spiking neural network with direct training for object recognition," *Electronics*, vol. 11, no. 13, 2022. doi: 10.3390/electronics11132097. [Online]. Available: https://www.mdpi.com/2079-9292/11/13/2097

[60] J. Wu, Y. Chua, M. Zhang, H. Li, and K. C. Tan, "A spiking neural network framework for robust sound classification," *Frontiers in Neuroscience*, vol. 12, 2018. doi: 10.3389/fnins.2018.00836. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2018.00836

[61] Z. Bing, C. Meschede, F. Röhrbein, K. Huang, and A. C. Knoll, "A survey of robotics control based on learning-inspired spiking neural networks," *Frontiers in Neurorobotics*, vol. 12, 2018. doi: 10.3389/fnbot.2018.00035. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnbot.2018.00035

[62] A. Di Mauro, M. Scherer, D. Rossi, and L. Benini, "Kraken: A Direct Event/Frame-Based Multi-sensor Fusion SoC for Ultra-Efficient Visual Processing in Nano-UAVs," in *2022 IEEE Hot Chips 34 Symposium (HCS)*, 2022. doi: 10.1109/HCS55958.2022.9895621 pp. 1–19.

[63] T. Rollo, F. Blanchini, G. Giordano, R. Specogna, and D. Esseni, "Stabilization of negative capacitance in ferroelectric capacitors with and without a metal interlayer," *Nanoscale*, vol. 12,, pp. 6121–6129, 2020. doi: 10.1039/c9nr09470a

[64] W. Maass, "Networks of spiking neurons: The third generation of neural network models," *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, 1997. doi: https://doi.org/10.1016/S0893-6080(97)00011-7. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608097000117

[65] E. Izhikevich, "Which model to use for cortical spiking neurons?" *IEEE Transactions on Neural Networks*, vol. 15, no. 5, pp. 1063–1070, 2004. doi: 10.1109/TNN.2004.832719

[66] A. N. Burkitt, "A review of the integrate-and-fire neuron model: I. homogeneous synaptic input," *Biological Cybernetics*, vol. 95, no. 1, pp. 1–19, Jul 2006. doi: 10.1007/s00422-006-0068-6. [Online]. Available: https://doi.org/10.1007/s00422-006-0068-6

[67] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *Journal of Neurophysiology*, vol. 94, no. 5, pp. 3637–3642, 2005. doi: 10.1152/jn.00686.2005 PMID: 16014787. [Online]. Available: https://doi.org/10.1152/jn.00686.2005

[68] 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*, vol. 117, no. 4, pp. 500–544, 1952. doi: https://doi.org/10.1113/jphysiol.1952.sp004764. [Online]. Available: https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1952.sp004764

[69] E. Izhikevich, "Simple model of spiking neurons," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1569–1572, Nov 2003. doi: 10.1109/TNN.2003.820440

[70] G. Indiveri, "Neuromorphic analog VLSI sensor for visual tracking: Circuits and application examples," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 11, pp. 1337–1347, 1999.

[71] M. A. Lewis, R. Etienne-Cummings, A. H. Cohen, and M. Hartmann, "Toward biomorphic control using custom aVLSI CPG chips," in *Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. symposia proceedings (Cat. No. 00CH37065)*, vol. 1. IEEE, 2000, pp. 494–500.

[72] Z. Bing, I. Baumann, Z. Jiang, K. Huang, C. Cai, and A. Knoll, "Supervised Learning in SNN via Reward-Modulated Spike-Timing-Dependent Plasticity for a Target Reaching Vehicle," *Frontiers in Neurorobotics*, vol. 13, 2019. doi: 10.3389/fnbot.2019.00018. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnbot.2019.00018

[73] X. Wang, X. Lin, and X. Dang, "Supervised learning in spiking neural networks: A review of algorithms and evaluations," *Neural Networks*, vol. 125, pp. 258–280, 2020. doi: https://doi.org/10.1016/j.neunet.2020.02.011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608020300563

[74] M. Payvand, M. E. Fouda, F. Kurdahi, A. Eltawil, and E. O. Neftci, "Error-triggered three-factor learning dynamics for crossbar arrays," in *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, 2020. doi: 10.1109/AICAS48895.2020.9073998 pp. 218–222.

[75] E. M. Izhikevich, "Solving the Distal Reward Problem through Linkage of STDP and Dopamine Signaling," *Cerebral Cortex*, vol. 17, no. 10, pp. 2443–2452, 01 2007. doi: 10.1093/cercor/bhl152. [Online]. Available: https://doi.org/10.1093/cercor/bhl152

[76] T. Masquelier, R. Guyonneau, and S. J. Thorpe, "Spike timing dependent plasticity finds the start of repeating patterns in continuous spike trains," *PLOS ONE*, vol. 3, no. 1, pp. 1–9, 01 2008. doi: 10.1371/journal.pone.0001377. [Online]. Available: https://doi.org/10.1371/journal.pone.0001377

[77] A. Morrison, M. Diesmann, and W. Gerstner, "Phenomenological models of synaptic plasticity based on spike timing," *Biological Cybernetics*, vol. 98, no. 6, pp. 459–478, Jun 2008. doi: 10.1007/s00422-008-0233-1. [Online]. Available: https://doi.org/10.1007/s00422-008-0233-1

[78] J. Wu, M. Zhang, H. Li, and Y. Chua, "Competitive stdp-based feature representation learning for sound event classification," in *2019 International Joint Conference on Neural Networks (IJCNN)*, 2019. doi: 10.1109/IJCNN.2019.8851688 pp. 1–8.

[79] S. M. Bohté, J. N. Kok, and H. L. Poutré, "Spikeprop: backpropagation for networks of spiking neurons," in *ESANN*, 2000. ISBN 2-930307-00-5 pp. 419–424.

[80] O. Booij and H. tat Nguyen, "A gradient descent rule for spiking neurons emitting multiple spikes," *Information Processing Letters*, vol. 95, no. 6, pp. 552–558, 2005. doi: https://doi.org/10.1016/j.ipl.2005.05.023 Applications of Spiking Neural Networks. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020019005001560

[81] Y. Xu, X. Zeng, L. Han, and J. Yang, "A supervised multi-spike learning algorithm based on gradient descent for spiking neural networks," *Neural Networks*, vol. 43, pp. 99–113, 2013. doi: https://doi.org/10.1016/j.neunet.2013.02.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608013000440

[82] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification," *Frontiers in Neuroscience*, vol. 11, 2017. doi: 10.3389/fnins.2017.00682. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2017.00682

[83] T. P. Lillicrap, D. Cownden, D. B. Tweed, and C. J. Akerman, "Random synaptic feedback weights support error backpropagation for deep learning," *Nature Communications*, vol. 7, no. 1, p. 13276, Nov 2016. doi: 10.1038/ncomms13276. [Online]. Available: https://doi.org/10.1038/ncomms13276

[84] K. Selvaratnam, Y. Kuroe, and T. Mori, "Learning methods of recurrent spiking neural networks," *Trans. Inst. Syst. Control Inf. Eng.*, vol. 13, no. 3, pp. 95–104, 2000. doi: 10.5687/iscie.13.3_95

[85] Y. Kuroe and T. Ueyama, "Learning methods of recurrent spiking neural networks based on adjoint equations approach," in *The 2010 International Joint Conference on Neural Networks (IJCNN)*, 2010. doi: 10.1109/IJCNN.2010.5596914 pp. 1–8.

[86] D. Huh and T. J. Sejnowski, "Gradient descent for spiking neural networks," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018.

[87] G. Bellec, F. Scherr, A. Subramoney, E. Hajek, D. Salaj, R. Legenstein, and W. Maass, "A solution to the learning dilemma for recurrent networks of spiking neurons," *Nature Communications*, vol. 11, no. 1, p. 3625, Jul 2020. doi: 10.1038/s41467-020-17236-y. [Online]. Available: https://doi.org/10.1038/s41467-020-17236-y

[88] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 51–63, 2019. doi: 10.1109/MSP.2019.2931595

[89] T. C. Wunderlich and C. Pehle, "Event-based backpropagation can compute exact gradients for spiking neural networks," *Scientific Reports*, vol. 11, no. 1, p. 12829, Jun 2021. doi: 10.1038/s41598-021-91786-z. [Online]. Available: https://doi.org/10.1038/s41598-021-91786-z

[90] W. Maass, "Lower bounds for the computational power of networks of spiking neurons," *Neural Computation*, vol. 8, no. 1, pp. 1–40, Jan 1996. doi: 10.1162/neco.1996.8.1.1

[91] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.

[92] R. Fontanini, D. Esseni, and M. Loghi, "Reducing the spike rate in deep spiking neural networks," in *Proceedings of the International Conference on Neuromorphic Systems 2022*, ser. ICONS '22. New York, NY, USA: Association for Computing Machinery, 2022. doi: 10.1145/3546790.3546798. ISBN 9781450397896. [Online]. Available: https://doi.org/10.1145/3546790.3546798

[93] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, Mar. 1951. doi: 10.1214/aoms/1177729694. [Online]. Available: https://doi.org/10.1214/aoms/1177729694

[94] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[95] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[96] L. Li, A. Pratap, H.-T. Lin, and Y. S. Abu-Mostafa, "Improving generalization by data categorization," in *Knowledge Discovery in Databases: PKDD 2005*, A. M. Jorge, L. Torgo, P. Brazdil, R. Camacho, and J. Gama, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. ISBN 978-3-540-31665-7 pp. 157–168.

[97] L. Kriener, J. Göltz, and M. A. Petrovici, "The yin-yang dataset," *CoRR*, vol. abs/2102.08211, 2021. [Online]. Available: https://arxiv.org/abs/2102.08211

[98] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. doi: 10.1109/5.726791

[99] M. Stimberg, R. Brette, and D. F. Goodman, "Brian 2, an intuitive and efficient neural simulator," *eLife*, vol. 8, p. e47314, Aug. 2019. doi: 10.7554/eLife.47314

[100] M.-O. Gewaltig and M. Diesmann, "NEST (NEural Simulation Tool)," *Scholarpedia*, vol. 2, no. 4, p. 1430, 2007.

[101] M. Mozafari, M. Ganjtabesh, A. Nowzari-Dalini, and T. Masquelier, "SpykeTorch: Efficient Simulation of Convolutional Spiking Neural Networks With at Most One Spike per Neuron," *Frontiers in Neuroscience*, vol. 13, 2019. doi: 10.3389/fnins.2019.00625. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2019.00625

[102] N. Qiao and G. Indiveri, "Scaling mixed-signal neuromorphic processors to 28 nm FD-SOI technologies," in *2016 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2016. doi: 10.1109/BioCAS.2016.7833854 pp. 552–555.

[103] S. Moradi, N. Qiao, F. Stefanini, and G. Indiveri, "A Scalable Multicore Architecture With Heterogeneous Memory Structures for Dynamic Neuromorphic Asynchronous Processors (DYNAPs)," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 1, pp. 106–122, 2018. doi: 10.1109/TBCAS.2017.2759700

[104] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, F. Stefanini, D. Sumislawska, and G. Indiveri, "A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses," *Frontiers in Neuroscience*, vol. 9, 2015. doi: 10.3389/fnins.2015.00141. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2015.00141

[105] C. S. Thakur, J. L. Molin, G. Cauwenberghs, G. Indiveri, K. Kumar, N. Qiao, J. Schemmel, R. Wang, E. Chicca, J. Olson Hasler, J.-s. Seo, S. Yu, Y. Cao, A. van Schaik, and R. Etienne-Cummings, "Large-scale neuromorphic spiking array processors: A quest to mimic the brain," *Frontiers in Neuroscience*, vol. 12, 2018. doi: 10.3389/fnins.2018.00891. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2018.00891

[106] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The spinnaker project," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 652–665, 2014. doi: 10.1109/JPROC.2014.2304638

[107] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014. doi: 10.1126/science.1254642. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1254642

[108] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, Y. Liao, C.-K. Lin, A. Lines, R. Liu, D. Mathaikutty, S. McCoy, A. Paul, J. Tse, G. Venkataramanan, Y.-H. Weng, A. Wild, Y. Yang, and H. Wang, "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018. doi: 10.1109/MM.2018.112130359

[109] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri, "Neuromorphic electronic circuits for building autonomous cognitive systems," *Proceedings of the IEEE*, vol. 102, pp. 1367–1388, 2014.

[110] C. Pehle, S. Billaudelle, B. Cramer, J. Kaiser, K. Schreiber, Y. Stradmann, J. Weis, A. Leibfried, E. Müller, and J. Schemmel, "The brainscales-2 accelerated neuromorphic system with hybrid plasticity," *Frontiers in Neuroscience*, vol. 16, 2022. doi: 10.3389/fnins.2022.795876. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2022.795876

[111] S. Friedmann, J. Schemmel, A. Grübl, A. Hartel, M. Hock, and K. Meier, "Demonstrating hybrid learning in a flexible neuromorphic hardware system," *IEEE transactions on biomedical circuits and systems*, vol. 11, no. 1, pp. 128–142, 2016.

[112] G. Indiveri, E. Chicca, and R. Douglas, "A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity," *IEEE transactions on neural networks*, vol. 17, no. 1, pp. 211–221, 2006.

[113] E. Polidori, G. Camisa, A. Mesri, G. Ferrari, C. Polidori, M. Mastella, and E. Prati, "Experimental validation of an analog spiking neural network with STDP learning rule in CMOS technology," in *2022 IEEE International Conference on Metrology for Extended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*, 2022. doi: 10.1109/MetroXRAINE54828.2022.9967583 pp. 187–192.

[114] A. Rubino, C. Livanelioglu, N. Qiao, M. Payvand, and G. Indiveri, "Ultra-Low-Power FDSOI Neural Circuits for Extreme-Edge Neuromorphic Intelligence," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 1, pp. 45–56, 2021. doi: 10.1109/TCSI.2020.3035575

[115] M. Mahowald and R. Douglas, "A silicon neuron," *Nature*, vol. 354, no. 6354, pp. 515–518, Dec 1991. doi: 10.1038/354515a0. [Online]. Available: https://doi.org/10.1038/354515a0

[116] I. Sourikopoulos, S. Hedayat, C. Loyez, F. Danneville, V. Hoel, E. Mercier, and A. Cappy, "A 4-fJ/Spike Artificial Neuron in 65 nm CMOS Technology," *Frontiers in Neuroscience*, vol. 11, 2017. doi: 10.3389/fnins.2017.00123. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2017.00123

[117] S. Brink, S. Nease, P. Hasler, S. Ramakrishnan, R. Wunderlich, A. Basu, and B. Degnan, "A Learning-Enabled Neuron Array IC Based Upon Transistor Channel Models of Biological Phenomena," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 7, no. 1, pp. 71–81, 2013. doi: 10.1109/TBCAS.2012.2197858

[118] Q. Xia and J. J. Yang, “Memristive crossbar arrays for brain-inspired computing,” *Nature Materials*, vol. 18, no. 4, pp. 309–323, Apr 2019. doi: 10.1038/s41563-019-0291-x. [Online]. Available: https://doi.org/10.1038/s41563-019-0291-x

[119] M. Payvand, F. Moro, K. Nomura, T. Dalgaty, E. Vianello, Y. Nishi, and G. Indiveri, “Self-organization of an inhomogeneous memristive hardware for sequence learning,” *Nature Communications*, vol. 13, no. 1, p. 5793, Oct 2022. doi: 10.1038/s41467-022-33476-6. [Online]. Available: https://doi.org/10.1038/s41467-022-33476-6

[120] I.-T. Wang, Y.-C. Lin, Y.-F. Wang, C.-W. Hsu, and T.-H. Hou, “3D synaptic architecture with ultralow sub-10 fJ energy per spike for neuromorphic computation,” in *2014 IEEE International Electron Devices Meeting*, 2014. doi: 10.1109/IEDM.2014.7047127 pp. 28.5.1–28.5.4.

[121] W. Maass and C. M. Bishop, “A Pulse-Coded Communications Infrastructure for Neuromorphic Systems,” in *Pulsed Neural Networks.* The MIT Press, 11 1998. ISBN 9780262278768. [Online]. Available: https://doi.org/10.7551/mitpress/5704.003.0011

[122] L. Chua, “Memristor-the missing circuit element,” *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, pp. 507–519, 1971. doi: 10.1109/TCT.1971.1083337

[123] M. Di Ventra, Y. Pershin, and L. Chua, “Circuit elements with memory: Memristors, memcapacitors, and meminductors,” *Proceedings of the IEEE*, vol. 97, pp. 1717 – 1724, 11 2009. doi: 10.1109/JPROC.2009.2021077

[124] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, “The missing memristor found,” *Nature*, vol. 453, no. 7191, pp. 80–83, May 2008. doi: 10.1038/nature06932. [Online]. Available: https://doi.org/10.1038/nature06932

[125] M. Di Ventra, F. L. Traversa, F. Bonani, and Y. V. Pershin, “Dynamic computing random access memory: A brain-inspired computing paradigm with memelements,” in *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2014. doi: 10.1109/ISCAS.2014.6865324 pp. 1070–1073.

[126] A. Beck, J. G. Bednorz, C. Gerber, C. Rossel, and D. Widmer, “Reproducible switching effect in thin oxide films for memory applications,” *Applied Physics Letters*, vol. 77, no. 1, pp. 139–141, 07 2000. doi: 10.1063/1.126902. [Online]. Available: https://doi.org/10.1063/1.126902

[127] A. Chanthbouala, V. Garcia, R. O. Cherifi, K. Bouzehouane, S. Fusil, X. Moya, S. Xavier, H. Yamada, C. Deranlot, N. D. Mathur, M. Bibes, A. Barthélémy, and J. Grollier, “A ferroelectric memristor,” *Nature Materials*, vol. 11, no. 10, pp. 860–864, Oct 2012. doi: 10.1038/nmat3415. [Online]. Available: https://doi.org/10.1038/nmat3415

[128] W. Kautz, “Cellular Logic-in-Memory Arrays,” *IEEE Transactions on Computers*, vol. C-18, no. 8, pp. 719–727, 1969. doi: 10.1109/T-C.1969.222754

[129] K.-J. Heo, H.-S. Kim, J.-Y. Lee, and S.-J. Kim, “Filamentary Resistive Switching and Capacitance-Voltage Characteristics of the a-IGZO/TiO2 Memory,” *Scientific Reports*,

vol. 10, no. 1, p. 9276, Jun 2020. doi: 10.1038/s41598-020-66339-5. [Online]. Available: https://doi.org/10.1038/s41598-020-66339-5

[130] U. Das, D. Das, B. Paul, T. Rabha, S. Pattanayak, A. Kanjilal, S. Bhattacharjee, P. Sarkar, and A. Roy, "Induced Vacancy-Assisted Filamentary Resistive Switching Device Based on RbPbI3–xClx Perovskite for RRAM Application," *ACS Applied Materials & Interfaces*, vol. 12, no. 37, pp. 41 718–41 727, Sep 2020. doi: 10.1021/acsami.0c10123. [Online]. Available: https://doi.org/10.1021/acsami.0c10123

[131] Y. Demirağ, F. Moro, T. Dalgaty, G. Navarro, C. Frenkel, G. Indiveri, E. Vianello, and M. Payvand, "Pcm-trace: Scalable synaptic eligibility traces with resistivity drift of phase-change materials," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021. doi: 10.1109/ISCAS51556.2021.9401446 pp. 1–5.

[132] E. Covi, S. Lancaster, S. Slesazeck, V. Deshpande, T. Mikolajick, and C. Dubourdieu, "Challenges and perspectives for energy-efficient brain-inspired edge computing applications (invited paper)," in *2022 IEEE International Conference on Flexible and Printable Sensors and Systems (FLEPS)*, 2022. doi: 10.1109/FLEPS53764.2022.9781597 pp. 1–4.

[133] A. Lacaita, "Phase change memories: State-of-the-art, challenges and perspectives," *Solid-State Electronics*, vol. 50, no. 1, pp. 24–31, 2006. doi: https://doi.org/10.1016/j.sse.2005.10.046 Special Issue: Papers selected from the 2005 ULIS Conference. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0038110105003230

[134] D. Esseni, R. Fontanini, D. Lizzit, M. Massarotto, F. Driussi, and M. Loghi, "Ferroelectric based fets and synaptic devices for highly energy efficient computational technologies," in *2021 5th IEEE Electron Devices Technology & Manufacturing Conference (EDTM)*, 2021. doi: 10.1109/EDTM50988.2021.9420848 pp. 1–3.

[135] C. Zacharaki, S. Chaitoglou, N. Siannas, P. Tsipas, and A. Dimoulas, "Hf0.5Zr0.5O2-Based Germanium Ferroelectric p-FETs for Nonvolatile Memory Applications," *ACS Applied Electronic Materials*, vol. 4, no. 6, pp. 2815–2821, Jun 2022. doi: 10.1021/acsaelm.2c00324. [Online]. Available: https://doi.org/10.1021/acsaelm.2c00324

[136] T. S. Böscke, S. Teichert, D. Bräuhaus, J. Müller, U. Schröder, U. Böttger, and T. Mikolajick, "Phase transitions in ferroelectric silicon doped hafnium oxide," *Applied Physics Letters*, vol. 99, no. 11, p. 112904, 2011. doi: 10.1063/1.3636434. [Online]. Available: https://doi.org/10.1063/1.3636434

[137] T. Mikolajick, U. Schroeder, P. D. Lomenzo, E. T. Breyer, H. Mulaosmanovic, M. Hoffmann, T. Mittmann, F. Mehmood, B. Max, and S. Slesazeck, "Next generation ferroelectric memories enabled by hafnium oxide," in *2019 IEEE International Electron Devices Meeting (IEDM)*, 2019. doi: 10.1109/IEDM19573.2019.8993447 pp. 15.5.1–15.5.4.

[138] S. Slesazeck and T. Mikolajick, "Nanoscale resistive switching memory devices: a review," *Nanotechnology*, vol. 30, no. 35, p. 352003, jun 2019. doi: 10.1088/1361-6528/ab2084. [Online]. Available: https://doi.org/10.1088/1361-6528/ab2084

[139] R. Fontanini, J. Barbot, M. Segatto, S. Lancaster, Q. Duong, F. Driussi, L. Grenouillet, L. Triozon, J. Coignus, T. Mikolajick, S. Slesazeck, and D. Esseni, "Interplay Between Charge Trapping and Polarization Switching in BEOL-Compatible Bilayer Ferroelectric Tunnel Junctions," *IEEE Journal of the Electron Devices Society*, vol. 10, pp. 593–599, 2022. doi: 10.1109/JEDS.2022.3171217

[140] R. Fontanini, M. Segatto, K. S. Nair, M. Holzer, F. Driussi, I. Häusler, C. T. Koch, C. Dubourdieu, V. Deshpande, and D. Esseni, "Charge-trapping-induced compensation of the ferroelectric polarization in ftjs: Optimal conditions for a synaptic device operation," *IEEE Transactions on Electron Devices*, vol. 69, no. 7, pp. 3694–3699, 2022. doi: 10.1109/TED.2022.3175684

[141] B. Max, M. Hoffmann, S. Slesazeck, and T. Mikolajick, "Direct correlation of ferroelectric properties and memory characteristics in ferroelectric tunnel junctions," *IEEE Journal of the Electron Devices Society*, vol. 7, pp. 1175–1181, 2019. doi: 10.1109/JEDS.2019.2932138

[142] K. M. Rabe, M. Dawber, C. Lichtensteiger, C. Ahn, and J. Triscone, *Physiscs of Ferroelectrics, a Modern Perspective.* Springer, 2007.

[143] T. Mikolajick, S. Slesazeck, M. H. Park, and U. Schroeder, "Ferroelectric hafnium oxide for ferroelectric random-access memories and ferroelectric field-effect transistors," *MRS Bulletin*, vol. 43, no. 5, pp. 340–346, May 2018. doi: 10.1557/mrs.2018.92. [Online]. Available: https://doi.org/10.1557/mrs.2018.92

[144] D. Lehninger, M. Lederer, T. Ali, T. Kämpfe, K. Mertens, and K. Seidel, "Enabling Ferroelectric Memories in BEoL - towards advanced neuromorphic computing architectures," in *2021 IEEE International Interconnect Technology Conference (IITC)*, 2021. doi: 10.1109/IITC51362.2021.9537346 pp. 1–4.

[145] L. D. Landau and I. Khalatnikov, "On the anomalous absorption of sound near a second order phase transition point," *Dokl. Akad. Nauk*, vol. 96, pp. 469–472, 1954.

[146] W. J. Merz, "Domain formation and domain wall motions in ferroelectric batio$_3$ single crystals," *Phys. Rev.*, vol. 95, pp. 690–698, Aug 1954. doi: 10.1103/PhysRev.95.690. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.95.690

[147] D. Zhao, T. Lenz, G. H. Gelinck, P. Groen, D. Damjanovic, D. M. de Leeuw, and I. Katsouras, "Depolarization of multidomain ferroelectric materials," *Nature Communications*, vol. 10, no. 1, p. 2547, Jun 2019. doi: 10.1038/s41467-019-10530-4. [Online]. Available: https://doi.org/10.1038/s41467-019-10530-4

[148] D. Esseni and R. Fontanini, "Macroscopic and microscopic picture of negative capacitance operation in ferroelectric capacitors," *Nanoscale*, vol. 13, pp. 9641–9650, 2021. doi: 10.1039/D0NR06886A

[149] N. Siannas, C. Zacharaki, P. Tsipas, S. Chaitoglou, L. Bégon-Lours, C. Istrate, L. Pintilie, and A. Dimoulas, "Metastable ferroelectricity driven by depolarization fields in ultrathin Hf0.5Zr0.5O2," *Communications Physics*, vol. 5, no. 1, p. 178, Jul 2022. doi: 10.1038/s42005-022-00951-x. [Online]. Available: https://doi.org/10.1038/s42005-022-00951-x

[150] P. Chandra and P. B. Littlewood, "A landau primer for ferroelectrics," 2006.

[151] S. Salahuddin and S. Datta, "Use of Negative Capacitance to Provide Voltage Amplification for Low Power Nanoscale Devices," *Nano Letters*, vol. 8, no. 2, 2008. doi: 10.1021/nl071804g

[152] M. A. Alam, M. Si, and P. D. Ye, "A critical review of recent progress on negative capacitance field-effect transistors," *Applied Physics Letters*, vol. 114, no. 9, p. 090401, 2019. doi: 10.1063/1.5092684. [Online]. Available: https://doi.org/10.1063/1.5092684

[153] J. Íñiguez, P. Zubko, I. Luk'yanchuk, and A. Cano, "Ferroelectric negative capacitance," *Nature Materials*, vol. 4, pp. 243–256, 2019.

[154] H. W. Park, J. Roh, Y. B. Lee, and C. S. Hwang, "Modeling of negative capacitance in ferroelectric thin films," *Advanced Materials*, vol. 31, no. 32, p. 1805266, 2019. doi: https://doi.org/10.1002/adma.201805266

[155] M. Hoffmann, M. Gui, S. Slesazeck, R. Fontanini, M. Segatto, D. Esseni, and T. Mikolajick, "Intrinsic Nature of Negative Capacitance in Multidomain $Hf_{0.5}Zr_{0.5}O_2$-Based Ferroelectric/Dielectric Heterostructures," *Advanced Functional Materials*, vol. 32, no. 2, p. 2108494, 2022. doi: https://doi.org/10.1002/adfm.202108494. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adfm.202108494

[156] A. C. Seabaugh and Q. Zhang, "Low-Voltage Tunnel Transistors for Beyond CMOS Logic," *Proceedings of the IEEE*, vol. 98, no. 12, pp. 2095–2110, Dec 2010. doi: 10.1109/JPROC.2010.2070470

[157] J. C. Wong and S. Salahuddin, "Negative Capacitance Transistors," *Proceedings of the IEEE*, vol. 107, no. 1, pp. 49–62, Jan. 2019.

[158] Z. Krivokapic, U. Rana, R. Galatage, A. Razavieh, A. Aziz, J.Liu, J.Shi, H. Kim, R. Sporer, C. Serrao, A.Busquet, P. Polakowski, J. Mülle, W. Kleemeier, A. Jacob, D. Brown, A. Knorr, R. Carter, and S. Banna, GLOBALFOUNDRIES, "14nm Ferroelectric FinFET Technology with Steep Subthreshold Slope for Ultra Low Power Applications," in *2017 IEEE International Electron Devices Meeting (IEDM)*, December 2017. doi: 10.1109/IEDM.2017.8268393. ISSN 2156-017X pp. 357–360.

[159] D. Kwon, Y. Liao, Y. Lin, J. P. Duarte, K. Chatterjee, A. J. Tan, A. K. Yadav, C. Hu, Z. Krivokapic, and S. Salahuddin, "Response Speed of Negative Capacitance FinFETs," in *2018 IEEE Symposium on VLSI Technology*, June 2018. doi: 10.1109/VLSIT.2018.8510626. ISSN 2158-9682 pp. 49–50.

[160] D. J. Frank, P. M. Solomon, C. Dubourdieu, M. M. Frank, V. Narayanan, and T. N. Theis, "The Quantum Metal Ferroelectric Field-Effect Transistor," *IEEE Transactions on Electron Devices*, vol. 61, no. 6, pp. 2145–2153, 2014. doi: 10.1109/TED.2016.2614432

[161] A. Jain and M. A. Alam, "Stability constraints define the minimum subthreshold swing of a negative capacitance field-effect transistor," *IEEE Transactions on Electron Devices*, vol. 61, no. 7, pp. 2235–2242, 2014. doi: 10.1109/TED.2013.2286997

[162] K. Majumdar, S. Datta, and S. P. Rao, "Revisiting the theory of ferroelectric negative capacitance," *IEEE Transactions on Electron Devices*, vol. 63, no. 5, pp. 2043–2049, 2016. doi: 10.1109/TED.2013.2286997

[163] A. I. Khan, U. Radhakrishna, K. Chatterjee, S. Salahuddin, and D. A. Antoniadis, "Negative capacitance behavior in a leaky ferroelectric," *IEEE Transactions on Electron Devices*, vol. 63, no. 11, pp. 4416–4422, Nov 2016.

[164] A. I. Khan, U. Radhakrishna, S. Salahuddin, and D. Antoniadis, "Work Function Engineering for Performance Improvement in Leaky Negative Capacitance FETs," *IEEE Electron Device Letters*, vol. 38, no. 9, pp. 1335–1338, Sep. 2017. doi: 10.1109/LED.2017.2733382

[165] T. Rollo and D. Esseni, "Energy Minimization and Kirchhoff's Laws in Negative Capacitance Ferroelectric Capacitors and MOSFETs," *IEEE Electron Device Letters*, vol. 38, no. 6, pp. 814–817, June 2017. doi: 10.1109/LED.2017.2691002

[166] T. Rollo and D. Esseni, "New Design Perspective for Ferroelectric NC-FETs," *IEEE Electron Device Letters*, vol. 39, no. 4, pp. 603–606, April 2018. doi: 10.1109/LED.2018.2795026

[167] T. Rollo and D. Esseni, "Influence of Interface Traps on Ferroelectric NC-FETs," *IEEE Electron Device Letters*, vol. 39, no. 7, pp. 1100–1103, July 2018. doi: 10.1109/LED.2018.2842087

[168] S. Pentapati, R. Perumal, S. Khandelwal, M. Hoffmann, S. K. Lim, and A. I. Khan, "Cross-Domain Optimization of Ferroelectric Parameters for Negative Capacitance Transistors—Part I: Constant Supply Voltage," *IEEE Transactions on Electron Devices*, vol. 67, no. 1, pp. 365–370, 2020. doi: 10.1109/TED.2013.2286997

[169] S. Pentapati, R. Perumal, A. I. Khan, and S. K. Lim, "Optimal Ferroelectric Parameters for Negative Capacitance Field-Effect Transistors Based on Full-Chip Implementations—Part II: Scaling of the Supply Voltage," *IEEE Transactions on Electron Devices*, vol. 67, no. 1, pp. 371–376, 2020. doi: 10.1109/TED.2013.2286997

[170] R. Fontanini, J. Barbot, M. Segatto, S. Lancaster, Q. Duong, F. Driussi, L. Grenouillet, F. Triozon, J. Coignus, T. Mikolajick, S. Slesazeck, and D. Esseni, "Polarization switching and interface charges in BEOL compatible Ferroelectric Tunnel Junctions," in *ESSDERC 2021 - IEEE 51st European Solid-State Device Research Conference (ESSDERC)*, 2021. doi: 10.1109/ESSDERC53440.2021.9631812 pp. 255–258.

[171] N. Siannas, C. Zacharaki, P. Tsipas, S. Chaitoglou, L. Begon-Lours, and A. Dimoulas, "Scaling ferroelectric HZO thickness for low power Ge MFS-FTJ memories," in *ESSDERC 2021 - IEEE 51st European Solid-State Device Research Conference (ESSDERC)*, 2021. doi: 10.1109/ESSDERC53440.2021.9631382 pp. 287–290.

[172] M. Y. Zhuravlev, Y. Wang, S. Maekawa, and E. Y. Tsymbal, "Tunneling electroresistance in ferroelectric tunnel junctions with a composite barrier," *Applied Physics Letters*, vol. 95, no. 5, p. 052902, 2009. doi: 10.1063/1.3195075. [Online]. Available: https://doi.org/10.1063/1.3195075

[173] H. W. Park, S. D. Hyun, I. S. Lee, S. H. Lee, Y. B. Lee, M. Oh, B. Y. Kim, S. G. Ryoo, and C. S. Hwang, "Polarizing and depolarizing charge injection through a thin dielectric layer in a ferroelectric–dielectric bilayer," *Nanoscale*, vol. 13, pp. 2556–2572, 2021. doi: 10.1039/D0NR07597C

[174] K. Toprasertpong, M. Takenaka, and S. Takagi, "Direct observation of interface charge behaviors in FeFET by quasi-static split C-V and hall techniques: Revealing FeFET operation," in *2019 IEEE International Electron Devices Meeting (IEDM)*, 2019. doi: 10.1109/IEDM19573.2019.8993664 pp. 23.7.1–23.7.4.

[175] J. Li, Y. Qu, M. Si, X. Lyu, and P. D. Ye, "Multi-Probe characterization of ferroelectric/dielectric interface by C-V, P-V and conductance methods," in *2020 IEEE Symposium on VLSI Technology*, 2020. doi: 10.1109/VLSITechnology18217.2020.9265069 pp. 1–2.

[176] S. Deng, Z. Jiang, S. Dutta, H. Ye, W. Chakraborty, S. Kurinec, S. Datta, and K. Ni, "Examination of the interplay between polarization switching and charge trapping in Ferroelectric FET," in *2020 IEEE International Electron Devices Meeting (IEDM)*, 2020. doi: 10.1109/IEDM13553.2020.9371999 pp. 4.4.1–4.4.4.

[177] B. Max, M. Hoffmann, S. Slesazeck, and T. Mikolajick, "Ferroelectric tunnel junctions based on ferroelectric-dielectric $Hf_{0.5}Zr_{0.5}O_2/Al_2O_3$ capacitor stacks," in *2018 48th European Solid-State Device Research Conference (ESSDERC)*, 2018. doi: 10.1109/ESSDERC.2018.8486882 pp. 142–145.

[178] T. Rollo, F. Blanchini, G. Giordano, R. Specogna, and D. Esseni, "Revised analysis of negative capacitance in ferrolectric-insulator capacitors: analytical and numerical results, physical insight, comparison to experiments. ," in *2019 IEEE International Electron Devices Meeting (IEDM)*, Dec 2019, pp. 7.2.1–7.2.4.

[179] M. Hoffmann, M. Pešić, S. Slesazeck, U. Schroeder, and T. Mikolajick, "On the stabilization of ferroelectric negative capacitance in nanoscale devices," *Nanoscale*, vol. 10, pp. 10 891–10 899, 2018. doi: 10.1039/C8NR02752H. [Online]. Available: http://dx.doi.org/10.1039/C8NR02752H

[180] M. Lundstrom and C. Jeong, *Near-Equilibrium Transport*. WORLD SCIENTIFIC, 2013. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/7975

[181] M. V. Fischetti, S. E. Laux, and E. Crabbé, "Understanding hot-electron transport in silicon devices: Is there a shortcut?" *Journal of Applied Physics*, vol. 78, no. 2, pp. 1058–1087, 1995. doi: 10.1063/1.360340. [Online]. Available: https://doi.org/10.1063/1.360340

[182] D. Esseni, P. Palestri, and L. Selmi, *Nanoscale MOS Transistors: Semi-Classical Transport and Applications*. Cambridge University Press, 2011.

[183] F. Driussi, P. Palestri, and L. Selmi, "Modeling, simulation and design of the vertical graphene base transistor," *Microelectronic Engineering*, vol. 109, pp. 338–341, 2013. doi: https://doi.org/10.1016/j.mee.2013.03.134 Insulating Films on Semiconductors 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167931713003638

[184] H. Schroeder, "Poole-Frenkel-effect as dominating current mechanism in thin oxide films—An illusion?!" *Journal of Applied Physics*, vol. 117, no. 21, p. 215103, 2015. doi: 10.1063/1.4921949. [Online]. Available: https://doi.org/10.1063/1.4921949

[185] F. Driussi, S. Spiga, A. Lamperti, G. Congedo, and A. Gambi, "Simulation Study of the Trapping Properties of HfO_2 -Based Charge-Trap Memory Cells," *IEEE Transactions on Electron Devices*, vol. 61, no. 6, pp. 2056–2063, 2014. doi: 10.1109/TED.2014.2316374

[186] M. Sharifshazileh, K. Burelo, T. Fedele, J. Sarnthein, and G. Indiveri, "A neuromorphic device for detecting high-frequency oscillations in human iEEG," in *2019 26th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, 2019. doi: 10.1109/ICECS46596.2019.8965192 pp. 69–72.

[187] R. Fontanini, M. Massarotto, R. Specogna, F. Driussi, M. Loghi, and D. Esseni, "Modelling and design of FTJs as high reading-impedance synaptic devices," in *2021 5th IEEE Electron Devices Technology Manufacturing Conference (EDTM)*, 2021. doi: 10.1109/EDTM50988.2021.9420863 pp. 1–3.

[188] T. Pfeil, T. Potjans, S. Schrader, W. Potjans, J. Schemmel, M. Diesmann, and K. Meier, "Is a 4-bit synaptic weight resolution enough? – constraints on enabling spike-timing dependent plasticity in neuromorphic hardware," *Frontiers in Neuroscience*, vol. 6, p. 90, 2012. doi: 10.3389/fnins.2012.00090. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2012.00090

[189] R. Cao, Y. Wang, S. Zhao, Y. Yang, X. Zhao, W. Wang, X. Zhang, H. Lv, Q. Liu, and M. Liu, "Effects of capping electrode on ferroelectric properties of $Hf_{0.5}Zr_{0.5}O_2$ thin films," *IEEE Electron Device Letters*, vol. 39, no. 8, pp. 1207–1210, 2018. doi: 10.1109/LED.2018.2846570

[190] B. Max, T. Mikolajick, M. Hoffmann, S. Slesazeck, and T. Mikolajick, "Retention characteristics of $Hf_{0.5}Zr_{0.5}O_2$-based ferroelectric tunnel junctions," in *2019 IEEE 11th International Memory Workshop (IMW)*, 2019. doi: 10.1109/IMW.2019.8739765 pp. 1–4.

[191] Y.-C. Lin, F. McGuire, and A. D. Franklin, "Realizing ferroelectric $Hf_{0.5}Zr_{0.5}O_2$ with elemental capping layers," *Journal of Vacuum Science & Technology B*, vol. 36, no. 1, p. 011204, 2018. doi: 10.1116/1.5002558. [Online]. Available: https://doi.org/10.1116/1.5002558

[192] J. Li, M. Si, Y. Qu, X. Lyu, and P. D. Ye, "Quantitative characterization of ferroelectric/dielectric interface traps by pulse measurements," *IEEE Transactions on Electron Devices*, vol. 68, no. 3, pp. 1214–1220, 2021. doi: 10.1109/TED.2021.3053497

[193] A. M. Ionescu and H. Riel, "Tunnel field-effect transistors as energy-efficient electronic switches," *Nature*, vol. 479, p. 329–337, 2011. doi: 10.1038/nature10679

[194] D. Esseni, M. Pala, P. Palestri, C. Alper, and T. Rollo, "A review of selected topics in physics based modeling for tunnel field-effect transistors," *Semiconductor Science and Technology*, vol. 32, no. 8, p. 083005, 2017. [Online]. Available: http://stacks.iop.org/0268-1242/32/i=8/a=083005

[195] R. Fontanini, M. Segatto, M. Massarotto, R. Specogna, F. Driussi, M. Loghi, and D. Esseni, "Modeling and Design of FTJs as Multi-Level Low Energy Memristors for Neuromorphic Computing," *IEEE Journal of the Electron Devices Society*, vol. 9, pp. 1202–1209, 2021. doi: 10.1109/JEDS.2021.3120200

[196] K. Toprasertpong, Z. Y. Lin, T. E. Lee, M. Takenaka, and S. Takagi, "Asymmetric Polarization Response of Electrons and Holes in Si FeFETs: Demonstration of Absolute Polarization Hysteresis Loop and Inversion Hole Density over $2{\times}10^{13}$ cm^{-2}," in *2020 IEEE Symposium on VLSI Technology*, 2020. doi: 10.1109/VLSITechnology18217.2020.9265015 pp. 1–2.

[197] M. Hyuk Park, H. Joon Kim, Y. Jin Kim, W. Lee, T. Moon, and C. Seong Hwang, "Evolution of phases and ferroelectric properties of thin $Hf_{0.5}Zr_{0.5}O_2$ films according to the thickness and annealing temperature," *Applied Physics Letters*, vol. 102, no. 24, p. 242905, 2013.

[198] E. D. Grimley, T. Schenk, X. Sang, M. Pešić, U. Schroeder, T. Mikolajick, and J. M. LeBeau, "Structural changes underlying field-cycling phenomena in ferroelectric HfO_2 thin films," *Advanced Electronic Materials*, vol. 2, no. 9, p. 1600173, 2016.

[199] M. Fukunaga and Y. Noda, "New technique for measuring ferroelectric and antiferroelectric hysteresis loops," *Journal of the Physical Society of Japan*, vol. 77, no. 6, p. 064706, 2008. doi: 10.1143/JPSJ.77.064706. [Online]. Available: https://doi.org/10.1143/JPSJ.77.064706

[200] P. D. Lomenzo, P. Zhao, Q. Takmeel, S. Moghaddam, T. Nishida, M. Nelson, C. M. Fancher, E. D. Grimley, X. Sang, J. M. LeBeau, and J. L. Jones, "Ferroelectric phenomena in Si-doped HfO_2 thin films with TiN and Ir electrodes," *Journal of Vacuum Science & Technology B*, vol. 32, no. 3, p. 03D123, 2014. doi: 10.1116/1.4873323. [Online]. Available: https://doi.org/10.1116/1.4873323

[201] V. Mikheev, A. Chouprik, Y. Lebedinskii, S. Zarubin, Y. Matveyev, E. Kondratyuk, M. G. Kozodaev, A. M. Markeev, A. Zenkevich, and D. Negrov, "Ferroelectric second-order memristor," *ACS Applied Materials & Interfaces*, vol. 11, no. 35, pp. 32 108–32 114, 2019. doi: 10.1021/acsami.9b08189 PMID: 31402643. [Online]. Available: https://doi.org/10.1021/acsami.9b08189

[202] H. Ryu, H. Wu, F. Rao, and W. Zhu, "Ferroelectric tunneling junctions based on aluminum oxide/ zirconium-doped hafnium oxide for neuromorphic computing," *Scientific Reports*, vol. 9, no. 1, p. 20383, Dec 2019. doi: 10.1038/s41598-019-56816-x. [Online]. Available: https://doi.org/10.1038/s41598-019-56816-x

[203] V. Deshpande, K. S. Nair, M. Holzer, S. Banerjee, and C. Dubourdieu, "CMOS back-end-of-line compatible ferroelectric tunnel junction devices," *Solid-State Electronics*, vol. 186, p. 108054, 2021. doi: https://doi.org/10.1016/j.sse.2021.108054. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S003811012100099X

[204] M. Segatto, R. Fontanini, F. Driussi, D. Lizzit, and D. Esseni, "Limitations to electrical probing of spontaneous polarization in ferroelectric-dielectric heterostructures," *IEEE Journal of the Electron Devices Society*, pp. 1–1, 2022. doi: 10.1109/JEDS.2022.3164652

[205] R. Athle, A. E. Persson, A. Irish, H. Menon, R. Timm, and M. Borg, "Effects of TiN top electrode texturing on ferroelectricity in $Hf_{1-x}Zr_xO_2$," *ACS applied materials & interfaces*, vol. 13, no. 9, pp. 11 089–11 095, 2021.

[206] R. Alcala, C. Richter, M. Materano, P. D. Lomenzo, C. Zhou, J. L. Jones, T. Mikolajick, and U. Schroeder, "Influence of oxygen source on the ferroelectric properties of ALD grown $Hf_{1-x}Zr_xO_2$ films," *Journal of Physics D: Applied Physics*, vol. 54, no. 3, p. 035102, oct 2020. doi: 10.1088/1361-6463/abbc98. [Online]. Available: https://doi.org/10.1088/1361-6463/abbc98

[207] J. A. T. Kim and D. A. Antoniadis, "Dynamics of $HfZrO_2$ Ferroelectric Structures: Experiments and Models," in *2020 IEEE International Electron Devices Meeting (IEDM)*, Dec 2020, pp. 441–444.

[208] A. Kerber, E. Cartier, R. Degraeve, P. Roussel, L. Pantisano, T. Kauerauf, G. Groeseneken, H. Maes, and U. Schwalke, "Charge trapping and dielectric reliability of SiO_2-Al_2O_3 gate stacks with TiN electrodes," *IEEE Transactions on Electron Devices*, vol. 50, no. 5, pp. 1261–1269, 2003. doi: 10.1109/TED.2003.813486

[209] Q. Q. Shu and W. G. Ma, "Barrier parameter variation in Al-Al_2O_3-metal tunnel junctions," *Applied Physics Letters*, vol. 61, no. 21, pp. 2542–2544, 1992. doi: 10.1063/1.108145. [Online]. Available: https://doi.org/10.1063/1.108145

[210] Z. Dong, X. Cao, T. Wu, and J. Guo, "Tunneling current in HfO_2 and $Hf_{0.5}Zr_{0.5}O_2$-based ferroelectric tunnel junction," *Journal of Applied Physics*, vol. 123, no. 9, p. 094501, 2018. doi: 10.1063/1.5016823. [Online]. Available: https://doi.org/10.1063/1.5016823

[211] N. Saks and M. Ancona, "Determination of interface trap capture cross sections using three-level charge pumping," *IEEE Electron Device Letters*, vol. 11, no. 8, pp. 339–341, 1990. doi: 10.1109/55.57927

[212] S. Gupta, E. Simoen, R. Loo, O. Madia, D. Lin, C. Merckling, Y. Shimura, T. Conard, J. Lauwaert, H. Vrielinck, and M. Heyns, "Density and Capture Cross-Section of Interface Traps in $GeSnO_2$ and GeO_2 Grown on Heteroepitaxial GeSn," *ACS Applied Materials & Interfaces*, vol. 8, no. 21, pp. 13 181–13 186, Jun 2016. doi: 10.1021/acsami.6b01582. [Online]. Available: https://doi.org/10.1021/acsami.6b01582

[213] J. T. Ryan, A. Matsuda, J. P. Campbell, and K. P. Cheung, "Interface-state capture cross section—why does it vary so much?" *Applied Physics Letters*, vol. 106, no. 16, p. 163503, 2015. doi: 10.1063/1.4919100. [Online]. Available: https://doi.org/10.1063/1.4919100

[214] A. K. Tagantsev and G. Gerra, "Interface-induced phenomena in polarization response of ferroelectric thin films," *Journal of Applied Physics*, vol. 100, no. 5, p. 051607, 2006. doi: 10.1063/1.2337009. [Online]. Available: https://doi.org/10.1063/1.2337009

[215] M. Segatto, M. Massarotto, S. Lancaster, Q. T. Duong, A. Affanni, R. Fontanini, F. Driussi, D. Lizzit, T. Mikolajick, S. Slesazeck, and D. Esseni, "Polarization switching and ac small-signal capacitance in ferroelectric tunnel junctions," in *ESSDERC 2022 - IEEE 52nd*

European Solid-State Device Research Conference (ESSDERC), 2022. doi: 10.1109/ESSDERC55479.2022.9947185 pp. 340–343.

[216] L. Jiang, W. Hu, Z. Wei, W. Xu, and H. Meng, "High-performance organic single-crystal transistors and digital inverters of an anthracene derivative," *Advanced Materials*, vol. 21, no. 36, pp. 3649–3653, 2009. doi: https://doi.org/10.1002/adma.200900503. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.200900503

[217] N. Matsuo, T. Miura, A. Urakami, and T. Miyoshi, "Analysis of direct tunneling for thin SiO_2 film," *Japanese Journal of Applied Physics*, vol. 38, no. Part 1, No. 7A, pp. 3967–3971, jul 1999. doi: 10.1143/jjap.38.3967. [Online]. Available: https://doi.org/10.1143/jjap.38.3967

[218] J. Hur, Y.-C. Luo, P. Wang, N. Tasneem, A. I. Khan, and S. Yu, "Ferroelectric tunnel junction optimization by plasma-enhanced atomic layer deposition," in *2020 IEEE Silicon Nanoelectronics Workshop (SNW)*, 2020. doi: 10.1109/SNW50361.2020.9131649 pp. 11–12.

[219] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special lecture on IE*, vol. 2, no. 1, pp. 1–18, 2015.

[220] M. M. Rahman, K.-Y. Shin, and T.-W. Kim, "Characterization of electrical traps formed in Al_2O_3 under various ALD conditions," *Materials*, vol. 13, no. 24, 2020. doi: 10.3390/ma13245809. [Online]. Available: https://www.mdpi.com/1996-1944/13/24/5809

[221] J. B. Kim, D. R. Kwon, K. Chakrabarti, C. Lee, K. Y. Oh, and J. H. Lee, "Improvement in Al_2O_3 dielectric behavior by using ozone as an oxidant for the atomic layer deposition technique," *Journal of Applied Physics*, vol. 92, no. 11, pp. 6739–6742, 2002. doi: 10.1063/1.1515951. [Online]. Available: https://doi.org/10.1063/1.1515951

[222] K. Henkel, M. Kot, and D. Schmeißer, "Localized defect states and charge trapping in atomic layer deposited-Al_2O_3 films," *Journal of Vacuum Science & Technology A*, vol. 35, no. 1, p. 01B125, 2017. doi: 10.1116/1.4971991. [Online]. Available: https://doi.org/10.1116/1.4971991

[223] J.-Y. Wu, Y.-T. Chen, M.-H. Lin, and T.-B. Wu, "Ultrathin HfON trapping layer for charge-trap memory made by Atomic Layer Deposition," *IEEE Electron Device Letters*, vol. 31, no. 9, pp. 993–995, 2010. doi: 10.1109/LED.2010.2052090

Acknowledgements

First and foremost, I would like to thank my advisors, Prof. Mirko Loghi and Prof. David Esseni for their excellent supervision and continuous guidance. Moreover, a special thanks to Mattia Segatto, not only a colleague but especially a loyal friend over all these years. Now, you are not a padawan anymore, but a master instead! Moreover, I would like to thank all BeFerroSynaptic people, and, in particular, Erika, Suzanne and Justine! We spent a good time together and I hope this friendship will last in the coming years! Then, I would like to express my gratitude to my friends and my family for their endless and invaluable support. Last but not least, I am deeply grateful to my wife Giulia for her endless patience and loving care throughout this journey.

Riccardo Fontanini, Udine, January 2023