Devices and Algorithms for Analog Deep Learning

by

O. Murat Onen

B.Sc., Middle East Technical University (2017) M.Sc., Massachusetts Institute of Technology (2019)

Submitted to the Department of Electrical Engineering and Computer Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2022

© Massachusetts Institute of Technology 2022. All rights reserved.

Certified by...... Jesús A. del Alamo

Professor of Electrical Engineering and Computer Science Thesis Supervisor

Accepted by Leslie A. Kolodziejski Professor of Electrical Engineering and Computer Science Chair, Department Committee on Graduate Students

Devices and Algorithms for Analog Deep Learning

by

O. Murat Onen

Submitted to the Department of Electrical Engineering and Computer Science on May 13, 2022, in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Abstract

Efforts to realize analog processors have skyrocketed over the last decade as having energy-efficient deep learning accelerators became imperative for the future of information processing. However, the absence of two entangled components creates an impasse before their practical implementation: devices satisfying algorithm-imposed requirements and algorithms running on nonideality-tolerant routines. This thesis demonstrates a near-ideal device technology and a superior neural network training algorithm that can ultimately propel analog computing when combined together. The CMOS-compatible nanoscale protonic devices demonstrated here show unprecedented characteristics, incorporating the benefits of nanoionics with extreme acceleration of ion transport and reactions under strong electric fields. Enabled by a material-level breakthrough of utilizing phosphosilicate glass (PSG) as a proton electrolyte, this operation regime achieves controlled shuttling and intercalation of protons in nanoseconds at room temperature in an energy-efficient manner. Then, a theoretical analysis is carried out to explain the infamous incompatibility between asymmetric device modulation and conventional neural network training algorithms. By establishing a powerful analogy with classical mechanics, a novel method, Stochastic Hamiltonian Descent, is developed to exploit device asymmetry as a useful feature. Overall, devices and algorithms developed in this thesis have immediate applications in analog deep learning, whereas the overarching methodology provides further insight for future advancements.

Thesis Supervisor: Jesús A. del Alamo Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

The study reported in this dissertation would not have been possible without the support and advice of my colleagues, family and friends. In particular, I would like to thank:

My thesis advisor, Prof. Jesús A. del Alamo for his immense insight, dedication to engineering, and fostering his students to build independent researching skills, meticulous scientific rigor, and stellar work ethic.

Prof. Bilge Yıldız and Prof. Ju Li for their invaluable guidance and shared experience in ionic transport and materials.

Nicolas Émond for his colossal contributions to optimize the protonic devices and comradery since the early days that required great persistence.

Prof. Frances M. Ross, Baoming Wang, Xiahui Yao, Wenjie Lu, Difei Zhang, and Claudia Vázquez Sanz for their valuable contributions to development and understanding of the protonic devices.

Marco Colangelo, Andrew E. Dane, Di Zhu, Kevin May, and Prof. Karl K. Berggren for their shared experience in nanofabrication and process optimization.

Tayfun Gökmen, Teodor K. Todorov, Wilfried Haensch, and Heike Riel for introducing me to the field of analog deep learning and giving me the very first opportunity to participate in top-tier research.

My colleagues at IBM Research: Malte Rasch, Vasileios Kalantzis, John Rozen, Lior Horesh and Haim Avron for their participation in numerous fruitful discussions.

Members of the XTreme Transistors Group: Alon Vardy, Taekyong Kim, Ethan S. Lee, Yanjie Shao, Xin Zhao, Aviram Massuda, and Elizabeth Kubicki for their many valuable suggestions in designing, fabricating, and testing devices.

The technical staff at NSL, EBL, MTL, MIT.nano, and MRSEC: James Daley, Mark Mondol, Donal Jamieson, Bob Bicchieri, Kurt Broderick, Dave Terry, Dennis Ward, Gary Riggott, Aubrey Penn, Whitney Hess, Charlie Settens, Jorg Scholvin, and Vicky Diadiuk for enabling cutting-edge device research at MIT.

Marco Turchetti, the best researcher I met at MIT, for his fellowship throughout

our time at graduate school and teaching me by example that serenity can be the better stance in certain predicaments.

My parents, and both grandfathers for being my source of determination in life and their unconditional support whichever path I venture down.

Sıla Deniz Çalışgan, my dearest colleague, best friend, and partner in life, for being the most compelling reason for me to remain in the present and making any adventure worthwhile as long as I can share it with her.

Contents

1	Introduction to Analog Deep Learning		11	
	1.1	Thesis Goal and Outline	15	
2	Devices for Analog Deep Learning			
	2.1	Survey of Device Technologies for Analog Computing	17	
	2.2	Scaling Attempt for Nafion-Based Protonic Programmable Resistors .	22	
	2.3	Phophosilicate Glass (PSG) as CMOS-Compatible Solid-State Pro-		
		tonic Electrolyte	24	
	2.4	Fabrication of Microscale PSG-Based Protonic Programmable Resistors	26	
	2.5	Experimental Characterization Setup	28	
	2.6	Characterization of Microscale PSG-Based Protonic Programmable Re-		
		sistors	29	
	2.7	Optimization of the WO ₃ Layer for Protonic Programmable Resistors	32	
	2.8	Fabrication of Nanoscale PSG-Based Protonic Programmable Resistors	34	
	2.9	Characterization of Nanoscale PSG-Based Protonic Programmable Re-		
		sistors	37	
	2.10	Extreme Electric Field Operation Regime for Protonic Programmable		
		Resistors	40	
	2.11	Conclusion	44	
3	Alg	orithms for Analog Deep Learning	45	
	3.1	Analog Deep Learning Based on Stochastic Gradient Descent	46	
		3.1.1 Analysis of Asymmetry Caused Effects Under SGD	48	

		3.1.2	Mathematical Modeling of Device Asymmetry Under SGD $$. $$.	50
		3.1.3	Existing Methods to Battle Asymmetry-Related Accuracy Degra-	
			dation	54
	3.2	Analog	g Deep Learning with Stochastic Hamiltonian Descent	55
		3.2.1	Mathematical Modeling of Device Asymmetry Under SHD	57
		3.2.2	Analysis of Device Asymmetry Under SHD	59
		3.2.3	Hardware Cost Reduction of SHD Implementation	63
		3.2.4	Experimental Demonstration of the SHD Algorithm	64
	3.3	Simula	ted Neural Network Training Results	68
	3.4	Conclu	usion	71
4	Con	clusior	and Future Directions	73
5	App	oendix		77
	5.1	Proces	s Engineering and Additional Device Results	77
		5.1.1	Undercut Profile of WO_3	77
		5.1.2	Optimization of Palladium Reservoir Thickness	77
		5.1.3	Fabrication Flow for Microscale Protonic Programmable Resistors	78
		5.1.4	Fabrication Flow for Nanoscale Protonic Programmable Resistors	79
		5.1.5	Alternative Layouts for Protonic Devices	80
			Alternative Active Channel Materials	81
		5.1.6		
		5.1.6 5.1.7	Field-Effect Related Volatile Conductance Changes	83
		5.1.6 5.1.7 5.1.8	Field-Effect Related Volatile Conductance Changes	83 83
		5.1.6 5.1.7 5.1.8 5.1.9	Field-Effect Related Volatile Conductance Changes Linear Regression Experiment Comparison of Retention Characteristics Under GND and FLT	83 83
		5.1.6 5.1.7 5.1.8 5.1.9	Field-Effect Related Volatile Conductance Changes Linear Regression Experiment Comparison of Retention Characteristics Under GND and FLT Gate Conditions	83 83 85

List of Figures

1-1	Schematic of a sample crossbar array	12
2-1	Review of programmable resistor technologies	18
2-2	Key operational dynamics of a 3-terminal electrochemical programmable	
	resistor	20
2-3	Nafion-based protonic programmable resistors	22
2-4	Optimization of phosphosilicate glass (PSG) electrolyte \ldots .	25
2-5	Fabrication of PSG-based microscale protonic programmable resistors	27
2-6	Experimental characterization setup for protonic programmable resistors	29
2-7	Experimental characterization of microscale protonic devices	30
2-8	Modulation behavior of V_2O_5 -PSG-based protonic devices	31
2-9	Optimization of the WO_3 layer $\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots$	33
2-10	Fabrication of PSG-based microscale protonic programmable resistors	35
2-11	Nanosecond protonic programmable resistors	36
2-12	Elemental analysis of nanoscale protonic programmable resistors	37
2-13	Ultrafast and energy-efficient modulation characteristics of nanoscale	
	protonic programmable resistors	38
2-14	Energy estimation of nanoscale protonic devices	39
2-15	Voltage dependence of conductance modulation	41
2-16	Programming voltage dependence of PSG thickness for nanoscale pro-	
	tonic programmable resistance	42
2-17	Modulation dynamics for short and long pulse durations $\ldots \ldots \ldots$	43

3-1	Implementation of the Stochastic Gradient Descent algorithm on ana-	
	log crossbar arrays	47
3-2	Linear regression under SGD algorithm using symmetric crosspoint	
	elements	48
3-3	Linear regression under SGD algorithm using asymmetric crosspoint	
	elements	49
3-4	Problem-specific residual error dependence for asymmetric devices trained	
	under SGD algorithm	50
3-5	Interaction of device asymmetry and dataset standard deviation $\ . \ .$	53
3-6	Implementation of the Stochastic Hamiltonian Descent algorithm on	
	analog crossbar arrays	56
3-7	Physical analogy between SHD algorithm and damped harmonic oscil-	
	lator	60
3-8	Comparison of single parameter regression training under SHD algo-	
	rithm for symmetric and asymmetric devices	62
3-9	Reference array initialization sensitivity of subsystems A and $C_{}$	65
3-10	Experimental setup using metal-oxide based programmable resistors $% \mathcal{L}^{(n)}$.	66
3-11	Experimental demonstration of the SHD algorithm on two-parameter	
	regression	67
3-12	Simulated training results for different resistive device technologies	70
3-13	Simulated training results for PCM-like devices	71
5-1	Undercut profile engineering for WO_3	78
5-2	Exfoliation of thick Pd layer under forming gas	78
5-3	Coplanar protonic devices with symmetric gate stack	81
5-4	Field-effect related volatile conductance change	84
5-5	Single-parameter linear regression experiment conducted on a nanoscale	
	protonic programmable resistor	85
5-6	Retention characteristics for ${\approx}100\mathrm{s}$ for floated and grounded gate con-	
	ditions	86

Chapter 1

Introduction to Analog Deep Learning

Deep learning has irreversibly changed and drastically improved how we process information. The core aspect driving this success is classifying and clustering representations of data at multiple levels of abstraction, allowing extraction of much richer information compared to raw data that classical computing paradigms have used so far [1]. However, the computational workloads to train state-of-the-art deep neural networks (DNNs) demand enormous computation time and energy costs for data centers [2]. To put into perspective, the number of operations to train a state-of-the-art DNN is already more than the number of atoms in one mole of substance, $\sim 10^{23}$ [3,4]. Since larger neural networks trained with bigger data sets generally provide better performance, this trend is expected to accelerate in the future. As a result, the necessity to provide fast and energy-efficient solutions for deep learning has invoked a massive collective research effort by industry and academia [5–7].

One way to cut the computational cost of deep learning is to use reduced-precision arithmetic for the otherwise computationally intensive matrix operations. This approach has indeed been successful for acceleration of inference tasks (i.e. classification of a new input using an already trained network). Highly optimized digital application-specific integrated circuit (ASIC) implementations using 2-bit resolution were able to provide significant benefits, without compromising classification accuracy [8]. However, the same method does not work for training applications (which are many orders of magnitude more computationally expensive than inference operations), as they were found to require at least hybrid 8-bit floating-point formats [9], which still imposes considerable energy consumption and processing time for large networks. Therefore, beyond-digital approaches that can efficiently handle training workloads are actively sought for.

The concept of analog computing has been put forward as an alternative, based on local information processing using physical device properties instead of conventional Boolean arithmetic. It is important to disambiguate here that by analog computing this thesis refers to deep neural network accelerators based on crossbar architectures and not brain-inspired/biomimetic approaches concerning spiking devices/circuits. An example crossbar array illustration is given in **Fig1-1**.



Figure 1-1: Schematic of a sample crossbar array: A 2×2 portion of the array is shown consisting of 3-terminal devices. At the end of rows and columns, peripheral circuitry is placed including: integrators, analog-digital-converters (ADCs), non-linear-function units (NLFs), digita-analog-converters (DACs), and pulse generation units. See Ref. [10] for details.

In his 1961 paper, Karl Steinbuch first described how one can use an array of resistors with a conductance matrix G, to represent an arbitrary matrix, W, for inner

product operations [11]. The idea is relatively simple, if one represents an input vector x with voltages $V_i \propto x_i$, then each element i, j will pass a current $G_{i,j}.V_i$, based on Ohm's Law. Then, at each line, these currents will be added up based on Kirchhoff's current law, yielding an output current $I_j \propto y_j$ for y = W.x. Alternatively, if the resistors in a crossbar array are nonlinear, one can encode the input information in duration of the voltage pulse (instead of the amplitude), and integrate the current at the end of lines (instead of measuring the current), such that $Q_j \propto y_j$ (**Fig.1-1**). This very basic approach is the first fully-parallel (i.e. constant-time independent of matrix size) primitive operation of analog processors, enabling the implementation of forward and backward pass cycles of backpropagation algorithm [12] for DNN training.

In addition to using physical properties instead of Boolean operations, the fact that the physical conductance matrix is both where the weight values are stored and where the multiply-accumulate (MAC) operations are executed means computations over crossbars to be considered local (sometimes referred to as in-memory computing). Therefore, one does not need to bring the weight values from memory to the arithmetic logic unit at every operation. Having said that, it would be an overstatement to claim analog processors do not suffer from the infamous von Neumann bottleneck, referring to the throughput limitation due to the data transfer rate between memory and the processor, given that one still needs to bring input matrix from the memory as well writing back the output matrix.

Overall, Steinbuch's approach only covers $\approx 2/3$ of overall training computational load, as the remaining third are outer products for updating the matrix to minimize the error function of the problem. Ideally, this means a maximum acceleration factor of $3\times$ for training, but practically, due to the many overheads spent during analog-digital conversions compensating for other nonidealities of resistors (e.g. long integration times to reduce thermal noise), there may not be a sizeable benefit for any decent-sized array. Similar arguments can be extended to inference operations, as the reduced precision alternatives mentioned above end up being more feasible for these implementations.

Following the arguments above, to benefit from using crossbar arrays, it was still

mandatory to find a way to execute the remaining third of the training operations in a fully parallel manner with analog architectures. Only after 60 years, a method that can execute rank-one outer products, relying on pulse-coincidence and thresholdbased incremental changes in device conductance was devised [13, 14]. Using this method, an entire crossbar array can be updated in parallel, without explicitly computing the outer product (i.e. the result is not returned to the user, but applied to the network) or having to read the value of any individual crosspoint element. As a result, all basic primitives for DNN training using the Stochastic Gradient Descent (SGD) algorithm can be performed in a fully-parallel fashion using analog crossbar architectures.

However, the performance benefits attained by analog computing are conditional on a set of highly strict properties: the device must be fast (~ ns), energy-efficient (< pJ), nonvolatile, reversible, and it must show symmetric conductance modulation with many (~ $10^2 - 10^3$) conductance states across a large dynamic range (> $10\times$) [10, 15]. In particular, asymmetric conductance modulation characteristics (i.e. having mismatch between positive and negative conductance adjustments) were found to deteriorate classification accuracy by causing inaccurate gradient accumulation [10, 13,15,16]. Moreover, for devices to be technologically relevant, they need to comprise CMOS-compatible materials, fabricated using a back-end-of-line (BEOL) -compatible process, have small footprint (< $0.04 \,\mu m^2$), and have a certain range of base resistance (~ $5 - 10 \,M\Omega$). These properties are equally critical as the former set as they are required for monolithic Si-integration (e.g. for memory, array periphery and digital nonlinear processing elements) of analog crossbar arrays. Unfortunately, achieving all of these properties simultaneously has so far been elusive.

In addition to widespread efforts to engineer ideal resistive elements, many customtailored algorithms have been proposed to remedy device and physical array nonidealities, in particular modulation asymmetry. The most critical issue with techniques proposed so far has been the introduction of serial operations in an "observe-andcompensate" type implementation [16–23]. Given that a key benefit of crossbar architectures is to reduce the $\mathcal{O}(N^2)$ computational complexity of a rank-1 update to $\mathcal{O}(1)$, a correction method that again requires accessing N^2 elements beats the original purpose. Moreover, these methods often do not employ implicit calculation of the outer product described above, and instead divert those operations to a nearby digital processor ($\mathcal{O}(N^2)$) [19–23]. These modifications undoubtedly resolve the device asymmetry related issues, but also forego the acceleration and energy efficiency benefits that make analog computing interesting in the first place.

1.1 Thesis Goal and Outline

The main goal of this thesis is to engineer devices and algorithms that can fulfill a decade-long-sought ideal: enabling analog computing as a next-generation computational paradigm. Unlike preceding efforts that have compartmentalized foci, the research ethos of this study will be co-optimization of device physics, materials, and algorithms, as the author believes how software should be executed and how unit elements should behave are fundamentally inseparable. It will be shown that this method opens up novel possibilities which are highly counterintuitive to conventional ways of thinking, such as using a most familiar material to obtain extreme characteristics and transforming a major predicament into a functional feature. Ultimately, the combination of the devices and algorithms investigated here should outperform the state-of-the-art and more importantly, provide a way of understanding that can guide researchers in this field for further advancements long beyond.

This thesis consists of two major chapters:

Devices for Analog Computing

This chapter will first provide a brief review of programmable resistor technologies, analyzing the advantages of ionic, in particular protonic, programmable resistors. The absence of a CMOS-compatible all-solid-state electrolyte that conducts protons but blocks electrons will be identified as the main technological bottleneck. Nanoporous phosphosilicate glass (PSG) will be proposed as an excellent candidate and will be used to first prototype microscale and then nanoscale devices with near-ideal characteristics. Channel material optimization, fabrication insights, and experimental characterization details will be provided for protonic programmable resistors. Finally, an extreme-field operation regime will be discovered, providing unprecedented characteristics for room temperature ionics. Overall, the protonic devices developed in this chapter will show better combined material, processing, and performance properties than all previous nonvolatile memory technologies.

Algorithms for Analog Computing

Despite numerous observations of device modulation asymmetry causing training accuracy degradation, reasons behind this effect have not been understood. This chapter will first explain the root cause of the issue by strategically simplifying the problem. Then, theoretical underpinnings of a novel fully-parallel training algorithm will be explained, which is compatible with asymmetric crosspoint elements. A powerful analogy with classical mechanics will be formed to demonstrate how device asymmetry can be exploited as a useful feature for analog deep learning processors. The new training algorithm described in this chapter will greatly relax the device symmetry requirement. More importantly, it will be shown in simulation that the combined performance of the protonic devices trained with this new algorithm will outperform that of nonexistent "ideal" devices trained with standard algorithms.

Chapter 2

Devices for Analog Deep Learning

Interest in engineering devices for analog deep learning applications (also referred to as programmable resistors, non-volatile memory, memristors, crosspoint elements) has skyrocketed in the last decade. [24–27]. As described in **Chap.1**, these devices need to satisfy a long set of strict properties, such that a DNN can be trained fast, energy-efficiently, and accurately (i.e. without degradation of classification performance) with analog crossbar architectures.

2.1 Survey of Device Technologies for Analog Computing

The most mature non-volatile memory technology is the phase-change memory (PCM), based on reversible phase transitions in chalcogenide glasses (**Fig.2-1**A). The active material inside these devices is highly resistive in its amorphous state, and can get conductive through crystallization due to controlled Joule heating [28]. Their maturity has enabled a few large-scale demonstrations for analog computing applications [14, 29, 30]. However, all PCM devices to date suffer from abrupt asymmetric amorphization (i.e. when one attempts to reduce device conductance, instead of a small decrement, the device conductance suddenly drops to the lowest state) [28], as well as resistance drift (i.e. device conductance steadily increase over time long after the quench is completed.) [31]. Former feature poses as a significant issue for training applications (See. **Secs.3.1.1**), whereas the latter is relevant for both training and inference implementations.



Figure 2-1: **Review of programmable resistor technologies:** Operational schematics for the modulation behavior of filamentary resistive random access memory (top left), ferroelectric tunnel junctions (bottom left), phase change memory (middle), and magnetic-tunneling devices (right). Figure reconstructed from images published in Ref. [24].

A leading competitor technology is the family of conductive filament based resistive devices (often called as resistive random access memory, ReRAM or RRAM, **Fig.2-1**B) which are based on the formation of a narrowly confined conducting path made of metal atoms or oxygen vacancies inside an insulating matrix [32–35]. The modulation dynamics of RRAMs are governed by redox reactions and ion migration, driven by electric, chemical, or thermal gradients across the device stack. Unfortunately, RRAMs also suffer from two major problems outstanding: low yield (high device-to-device variability) [36] and high stochasticity of the ion movement (modulation variability) [37, 38]. In contrast, magnetic/ferromagnetic tunneling devices (also referred to as magnetic random access memory, MRAM, **Fig.2-1**C) based on spin-transfer-torque switching [39], spin-orbital-torque switching [40], and voltage controlled magnetic anisotropy [41] have far superior reliability and reproducibility. However, achieving more than binary states per device with MRAMs has proven to be difficult, which severely limits their applicability to analog processors [42].

This very brief review undoubtedly falls short in covering all memory technologies,

as various attempts have been made based on different physical mechanisms. To give one particular example, the author of thesis previously went as far as to exploring superconducting memories to be repurposed for analog computing applications [43, 44]. However, none of such efforts have shown enough promise to become widely adopted by engineering or applied sciences communities.

One can argue that it is quite reasonable for devices that were originally designed for information storage purposes (i.e. memory) not to perform well in information processing applications. DNN training consists of many (~ $10^{20} - 10^{25}$) small incremental modifications. Therefore, devices need to be optimized for state transition (i.e. modulation/switching) properties rather than state preservation properties. Two polar opposite technologies adopt this mentality: ferroelectrics (extreme speed - strenuous control, **Fig.2-1**D) [45–47] and ionics (fine tunability - leaden motion, **Fig.2-2**) [48–58]. Ferroelectric tunnel junction devices operate by controlling the remnant polarization of the material through the application of an external electric field [59]. On the other hand, ionic devices rely on tuning the conductance of a transition metal oxide through controlling the number of ionic species inside. Both fields have great potential as well as a long-way to become the leading technology for analog computing applications. This thesis will focus onto the latter, in particular to 3-terminal electrochemical devices (**Fig.2-2**).

Electrochemical programmable resistors comprise 3 key functional layers: ion reservoir, electrolyte, and active channel. Reservoir material stores many (practically infinite) ions within, ready to be released upon the application of an electrical signal. Electrolyte layer, sandwiched between the reservoir and the active channel, conducts ions bidirectionally under electric field while insulating any electronic current. Finally, the active material in the channel layer has a variable conductivity controlled by the number of ions within. To operate the device three metal electrodes are placed: a gate (G) contact to the reservoir layer as well as source (S) and drain (D) contacts at the two ends of the active channel layer¹.

¹It should be noted here that not all electrochemical programmable resistors have 3-terminals. For example Refs. [53, 56, 57] all have 2-terminal configurations which use the same two electrodes for controlling (i.e. insertion/extraction of ions) as well as reading the state.



Figure 2-2: Key operational dynamics of a 3-terminal electrochemical programmable resistor: Increment operation refers to increasing the channel conductance, G_{DS} by a small amount. This function is obtained by inserting ions from the reservoir layer into the channel material through application of a voltage pulse to the gate (top left). Decrement operation is the opposite of increment operation, and is achieved by a gate pulse with opposite polarity (bottom left). When the gate terminal is left floating, since electrons cannot move either from outside or the inside (electrolyte is an electronic insulator), ions cannot move either. As a result, channel conductance can be read without moving ions in either direction, conserving the state.

Resultant device can then be used in three key modes (**Fig.2-2**). If a positive voltage (or current) pulse is applied to the gate terminal (while $V_D = V_S = 0$), some of the ions move from the reservoir to the channel, passing through the electrolyte². At the same time, electrons flow from the outer circuit (since electrolyte insulates electrons), in order to preserve charge neutrality. The increased number of ions within the channel material increases its conductivity (i.e. $G_{DS} \uparrow$, increment)³. Conversely, when a negative voltage (or current) pulse is applied to the gate $(V_D = V_S = 0)$, the ions that were previously inserted into the channel are extracted back. Electrons once again follow the same direction to that of protons through the outside circuitry. As a

²Ion directions are written for cationic devices, for anionic versions all should be reversed.

³This statement is true for cationic devices with n-type active channel and anionic devices with p-type active channel. For the other two combinations, increased ion concentration in the channel should decrease the channel conductivity instead and vice versa.

result, the conductivity of the channel can be reduced back (i.e. $G_{DS} \downarrow$, decrement). Importantly, when no electrical signal is applied to the gate terminal (i.e. floating gate, $I_G = 0$), since electrons cannot then move from the outside (or from the inside due to the electronically insulating electrolyte), ions cannot move within the device stack either. This allows one to read the nonvolatile conductance state (G_{DS}) of the device, by application of a small V_{DS} in a nondestructive manner⁴.

When one uses Li^+ as the working ion, then the structure shown in **Fig.2-2** becomes analogous to a solid-state battery [60]. There have been numerous attempts for lithium-based devices, utilizing the immense previous literature on lithium ion conductors developed for energy storage applications [48–51]. However, given the CMOS-incompatibility of Li-compounds, efforts were made to realize similar effects with oxygen ions, O^{2-} , instead. Importantly, these devices have bulk modulation, which makes them more controllable and reliable compared to their filamentary predecessors [52–55]. However, oxygen is a large and heavy particle to be moved around, limiting the operation speed and energy efficiency. Following this logic, researchers ultimately shifted their attention to the smallest and lightest of ions: protons (H^+) [56–58].

Unfortunately, the lack of solid-state inorganic protonic materials (electrolytes and reservoirs) critically limited the applicability and scalability of protonic devices. For example, Ref. [56] opted to use a liquid electrolyte (water infiltrated calcium aluminate with nanopores) while Ref. [57] relied on electrolysis of polymeric materials (poly(3,4-ethylenedioxythiophene) polystyrene sulfonate, PEDOT:PSS) to generate and shuttle protons. A major progress was made by Ref. [58], by utilizing Pd as a metallic proton reservoir (PdH_x), in combination with Nafion as an inorganic solid-state protonic electrolyte (**Fig.2-3**A). These devices showed promising modulation characteristics and energy efficiency unlike their electrolysis-based predecessors, thanks to the efficient proton storage and transport properties of the PdH_x layer. Moreover, given that Nafion is a material with high room temperature proton con-

 $^{^{4}}$ This property is a major design advancement over 2-terminal electrochemical devices, since the additional terminal allows decoupling of reading and programming paths, resulting with superior nonvolatile controllability

ductivity ($\approx 0.09 \,\mathrm{S \, cm^{-1}}$) [61], as well as high electronic resistivity, it also enabled devices to have good retention properties (**Fig.2-3**B). Therefore, these devices were identified as an excellent starting point, for which the scaling and Si-integration efforts are presented in **Sec.2.2**.



Figure 2-3: Nafion-based protonic programmable resistors: (A) Schematic of Nafion-based protonic programmable resistor. (B) Demonstration of channel modulation using current pulses of ± 200 nA and 5 ms. (C) Photograph of the shadowmask fabricated device on a 1×1 cm². (D) Scanning electron microscopy image of a failed nanofabrication attempt using Nafion. Due to the lack of mechanical integrity of the polymeric electrolyte, the gate structure collapsed under its own weight, squeezing the electrolyte out of the gate area. Subfigures (A) and (B) are modified from Ref. [58].

2.2 Scaling Attempt for Nafion-Based Protonic Programmable Resistors

Devices presented in Ref. [58] were fabricated using a series of shadowmasks. The resultant devices had an active area of $0.6 \times 1.2 \text{ mm}^2$, with a single device each fabricated on a $1 \times 1 \text{ cm}^2$ chip. In an attempt to scale the device dimensions down, an

electron-beam lithography based approach was developed. This process involved the following steps: (1) placement of Au contacts (evaporation + liftoff), (2) blanket deposition of the WO₃ active layer (reasons behind this material choice is explained in detail in **Sec.2.4**), (3) spin-coating of Nafion, (4) placement of a Cr/Al hard mask layer (evaporation + liftoff), (5) reactive-ion etching of Nafion and WO₃ layers using Cr/Al as a hard-mask, (6) removal of the hard mask layer with Al-liftoff, (7) placement of the Pd reservoir layer (evaporation + liftoff), and (8) placement of the Au contacts/pads (evaporation + liftoff).

Fig.2-3D shows a scanning electron microscopy (SEM) image of the failed fabrication following the Step 5 above. The key reason for the failure was found to be the lack of Nafion's sufficient mechanial integrity to be able to carry the load of the metal layer above (in the case of **Fig.2-3D**, the hardmask). Furthermore, during the process optimization, it was observed that the polymeric electrolyte material was incompatible with most solvents (e.g. isopropanol, acetone, and deionized water), as well as elevated temperatures that are commonly used in conventional lithography steps (e.g. 180 °C resist baking). Although further efforts were made which are not mentioned here (e.g. coplanar devices with a gap in between, which could be filled with the electrolyte as a final step), fabricating micro-/nanoscale devices with this electrolyte material was found to be infeasible.

Furthermore, it should be noted here that the proton conductivity of Nafion strongly relies on water absorption. Therefore, the device operation not only becomes dependent on the relative humidity of the environment, it also requires frequent hydration of the electrolyte during operation [62]. As a result, this study concluded the necessity to find a more suitable solid state protonic electrolyte for creation of more technologically-relevant protonic programmable resistors.

2.3 Phophosilicate Glass (PSG) as CMOS-Compatible Solid-State Protonic Electrolyte

In the search for an inorganic CMOS-compatible electrolyte layer, the author focused on silicate glasses (SiO₂), which are arguably the most well-known oxides in Si-technology, widely used as electron insulators [63] ⁵. Interestingly, proton (and other alkali ion) movement inside SiO₂ are conventionally perceived as a problem, as having mobile species within the gate oxide can lead to an unstable threshold voltage for MOSFETs made with such materials [64–66]. Therefore, a survey was conducted for how industry minimizes such unwanted mobility in their case, with an aim of applying the exact opposite for the purposes of this thesis.

Deposition conditions of silicate glasses can be engineered to yield a nanoporous structure with defect-OH terminated Si groups (silanol), providing a surface-site path for ion transport along the pores [67,68]. The acidic nature of silanol acts as a proton donor, which can then migrate by hopping between hydroxyl groups and structural water [69–72]. Doping silicate glasses with phosphorous sterically hinders (i.e. prevents formation of, via spatial blocking) the glass network to increase non-bridging oxygen bonds [73], replace Si-O-Si bonds with -Si-OH and -Si-O-P-OH groups, and increases both the pore volume and surface area [74]. The P-OH groups not only have higher acidity compared to silanol but are also amphoteric, meaning that they can act as both proton donors and acceptors [75]. These are all key properties that provide phosphosilicate glass (PSG, P-SiO2) high proton conductivity at room temperature $(2.54 \times 10^{-4} \,\mathrm{S \, cm^{-1}})$. This stands among the highest values when compared with several perovskites, fluorites and simple oxides proton conducting materials [76], while retaining their electron-insulating properties [77]. Furthermore, even though the proton conductivity of PSG can be further increased up to $1 \times 10^{-1} \, \mathrm{S \, cm^{-1}}$ by additional hydrothermal treatment [78] and chemisorption of water in the pore structure [74], such modifications also result in increased electronic conductivity of the

⁵The author notes that a key inspiration for this discovery was a desiccant silica-gel packet came with arts supplies. Author initially assumed the water absorption properties of SiO_2 could be a good starting point and started searching the literature.

material, which is undesirable for our design that requires non-volatile throttling of hydrogen in WO_3 .

Following the existing literature on the material, the deposition conditions of the PSG layer were then optimized to achieve: (a) low-density and nanoporous material and (b) 0.6% phosphorous dopant concentration (empirically found to be by Ref. [77]). The former increases the surface area of the material for proton conduction and can be achieved by lowering the deposition temperature, whereas the latter can be controlled via tuning the SiH₄:PH₃ ratio in the deposition system. Following this logic, the optimal PSG electrolyte was deposited using plasma-enhanced chemical vapor deposition (PECVD) processes at T = 100 °C with an RF plasma power of 60 W and a gas flow ratio of 12 sccm SiH₄ :12 sccm PH₃ diluted at 2% in H₂), yielding a deposition rate of $\approx 0.5 \text{ nm s}^{-1}$.



Figure 2-4: Optimization of phosphosilicate glass (PSG) electrolyte [79]: (A) Atomic force microscopy image of the PSG 12:12 thin film surface deposited on a Si surface. (B) X-ray photoelectron spectra of the P 2p and Si 2p peaks of the SiO2, PSG 12:12, and PSG 12:24 thin films. Inset in (B) shows zoomed spectra in the P 2p energy range [79].

An atomic force microscopy (AFM) image of a PSG film deposited under optimum conditions on a Si substrate is shown in **Fig.2-4**A. A nanogranular structure (so-called nanoglass) with a mean glassy grain diameter of $\approx 80 \text{ nm}$ and an RMS roughness \approx 1 nm is observed. This image also evidences the presence of nanopores with a high surface-to-volume ratio, an essential requirement for efficient proton transport in this material.

The stoichiometric ratio of each element in the optimized PSG film (PSG 12:12) was found to be 0.55 at.% P, 45.65 at.% Si, and 53.8 at.% O using X-ray photoelectron spectroscopy (XPS). The spectra of an undoped (SiO₂), an optimized (PSG 12:12) and a more highly-doped (PSG 12:24) film, deposited with 12:0, 12:12 and 12:24 SiH₄:PH₃ flux, respectively, were acquired (**Fig.2-4**B). While the intensity of the P 2p peak, located at a binding energy $\approx 134 \text{ eV}$ (inset), increases with PH₃ flux, that of the Si 2p peak decreases. The estimated P concentration of the PSG 12:12 film is 0.55%, a value similar to the one reported for PSG films with optimized proton conductivity ($2.54 \times 10^{-4} \text{ S cm}^{-1}$) [77]. ⁶

2.4 Fabrication of Microscale PSG-Based Protonic Programmable Resistors

Following the optimization process described in Sec.2.3, microscale 3-terminal protonic programmable resistors were made that employs a WO₃ channel, a PSG electrolyte layer, and a Pd gate reservoir. The basic operation principle of the device relies on modulating the channel conductance via the electrochemically controlled intercalation of protons into WO₃, as explained in Ref. [58]. Initially, protons are stored in the gate reservoir as Pd_x, which is achieved by the hydrogen uptake of Pd in a forming gas ambient $(3\% H_2 \text{ in } N_2)$ [80]. Depending on the polarity of a voltage pulse, a controlled number of protons are shuttled between the gate and the channel through the solid electrolyte (Fig.2-2). Protons are n-type dopants in WO₃ [81] and as they move in and out, the conductivity of the channel is incremented and decremented.

Among several oxides whose electronic conductivity can be tuned via cation intercalation (WO₃ [58], V₂O₅ [82], MoO₃ [83], Nb₂O₅ [84]), amorphous tungsten oxide (a-WO₃) was chosen here as the channel material. This selection was motivated by the well-established conductivity modulation [48, 52, 58] and electrochromism [56]

⁶The author wants to thank Nicolas Émond for his immense help with the metrology and contributions regarding PSG optimization.

dynamics with cation intercalation. a-WO₃ is a CMOS-compatible semiconductor with a bandgap of 2.8~3.2 eV whose conductivity can be precisely modulated by protonation, taking place concurrently with charge-balancing electron filling of the W 5d-orbital dominated conduction band in the dilute regime. The structure of a-WO₃ at room temperature is assumed to be similar to that of its crystalline counterpart (monoclinic based on corner-sharing WO₆ octahedra), but with disordered bond lengths and angles. The most common defect present in the WO3 lattice structure is the oxygen vacancy, which bonds to a W⁶⁺ ion, reduces the oxidation state of the neighboring W⁵⁺ ion, and increases the conductivity, in an analogous way as the electron does with proton intercalation. The extent of conductivity modulation in a-WO₃ by proton intercalation, therefore, depends on its initial defect concentration which also determines its initial conductivity [85].



Figure 2-5: Fabrication of PSG-based microscale protonic programmable resistors [79]: (A) Photolithography-based fabrication flow of the microscale devices. (B) Cross-sectional scanning electron microscope image of the PSG overhang (i.e. WO_3 undercut) region. (C) Top-view scanning electron microscope image of a finished device showing the source (S), drain (D), and gate (G) of a device with a nominal channel width (W) of 5 µm and a length (L) of 25 µm.

Microscale protonic programmable resistors were fabricated on a Si substrate covered with $10/90 \,\mathrm{nm} \,\mathrm{HfO}_2/\mathrm{Al}_2\mathrm{O}_3$ deposited by atomic layer deposition (ALD) for electrical and protonic insulation (Fig.2-5A). Channel contacts $(15/5 \,\mathrm{nm} \,\mathrm{Au/Cr})$ were first patterned using a direct-write photolithography and liftoff process. 10 nm WO₃ channel and 10 nm PSG electrolyte layers were blanket deposited using atomic layer deposition (ALD) and plasma-enhanced chemical vapor deposition (PECVD) processes, respectively. The deposition conditions for the PSG film were as discussed in Sec2.3. The PSG/WO_3 stack was subsequently patterned with a self-aligned reactive ion etching (RIE) process in CF_4 plasma, followed by TMAH-based wet etching of WO_3 to create a PSG-overhang that prevents shorting of the channel with the gate at the edges of the device. The resultant overhang can be seen in the crossectional SEM image shown in **Fig.2-5**B. Different channel dimensions (i.e. width and length) were patterned in the range between $2-100\,\mu\text{m}$. Finally, the 5 nm Pd reservoir and $150/10 \,\mathrm{nm}$ Au/Cr gate interconnect and pads were electron-beam evaporated and patterned through separate liftoff processes. Fig.2-5C shows the top-view scanning electron microscope image of a fabricated structure, whereas the inset in **Fig.2-6**A shows the photograph of the resultant chip.

2.5 Experimental Characterization Setup

Electrical characterization of the devices was conducted at room temperature in an enclosed probe station (NEXTRON MPS-PT, **Fig.2-6**A). Before starting the experiments with a given device, the reservoir layer was protonated (i.e. $Pd \rightarrow PdH_x$) using forming gas (FG, 3% H₂ in N₂). This process was conducted as follows: (1) contacting probes to the G, D, S terminals of the device under test, (2) pumping down the chamber to -70 kPa vacuum, (3) injecting the chamber with FG for $\approx 60 \text{ s}$, (4) keeping the chamber under positive pressure at $\approx 60 \text{ kPa}$ (filled with FG) for another $\approx 60 \text{ s}$, and finally pumping down the chamber to -70 kPa vacuum or forming gas during testing were found to be equivalent, whereas exposure to air resulted in no-operation as the hydrogen within the reservoir

was oxidized by the ambient O_2 .



Figure 2-6: Experimental characterization setup for protonic programmable resistors: (A) Photograph of the vacuum probe station NEXTRON MPS-PT and a device positioned in the chamber (inset). (B) Connection schemes between the device and the instruments under reading and pulsing modes.

Two of the probes (source and drain) were connected to the Source Measurement Units (SMUs) of a Keysight B1500 Semiconductor Analyzer, while the third probe (gate) was connected to the Pulse Generation Unit (PGU) of the same instrument. The experiment sequence and data acquisition were controlled via an in-house developed MATLAB suite. As explained in **Fig.2-2**, the instrument was programmed to function in two key operation modes: pulse (i.e. modulate), and read. During pulsing, the SPGU is digitally connected to the gate terminal, whereas the channel contacts are shorted to GND via their respective SMUs. Under this configuration, the pulses were then applied by the SPGU, for which the polarity, duration, amplitude, and repetition number are all defined by the aforementioned MATLAB script. On the other hand, during the readout, the gate terminal was digitally floated (i.e. $I_G = 0$), and the source current, I_S was integrated for $\approx 1 \text{ sec under } V_{DS} = 0.1 \text{ V}$ condition. The circuit schematics for these modes are shown in **Fig.2-6**B.

2.6 Characterization of Microscale PSG-Based Protonic Programmable Resistors

The sheet resistance of the ALD WO₃ was $\approx 18.2 \text{ k}\Omega/\Box$ as deposited, which then slightly decreased to $\approx 13.5 \text{ k}\Omega/\Box$ during the fabrication process. **Fig.2-7**A demonstrates clean device conductance modulation characteristics with reasonable symmetry. Moreover, these states are nonvolatile (**Fig.2-7**B), due to the electron-blocking properties of the PSG thin-film electrolyte (**Fig.2-7**C). Devices show good cycling endurance characteristics, with very little variation in conductance values even after the application of 50,000 pulses over the course of ≈ 30 h (**Fig.2-7**D). Furthermore, no noticeable open circuit voltage development across the gate stack was observed, allowing the operation under constant voltage pulses.

Note that similar devices were also fabricated using different PSG flavors using different doping concentrations (12:0-undoped, and 12:24-twice-doped, conditions described in **Sec.2.3**), deposition temperatures (200 °C and 300 °C), and different thicknesses (20 nm). As predicted by the process carried out in **Sec.2.3**, the optimized PSG layer indeed had a considerably superior performance to the other alternatives.



Figure 2-7: Experimental characterization of microscale protonic programmable resistors [79]: (A) Modulation characteristics of protonic programmable resistor (5 µm width, 50 µm length) performed by applying 50 voltage pulses of \pm 3V and 1s width in either direction. Each state is monitored for 2s by measuring the channel current while applying V_{DS}=0.1V. (B) Retention characteristics of the protonic device incrementally programmed by a voltage pulse $(V_{pulse} = -4 \text{ V}, t_{pulse} = 1 \text{ s})$ at 30 and 60 minutes. Each conductance state is read every minute by sweeping V_{SD} between $\pm 0.1 \text{ V}$ while the gate terminal is floating. (C) Gate I-V characteristics of a large device (W = 20 µm, L = 50 µm), evidencing good electronic insulation properties of the PSG electrolyte. (D) Endurance characterization throughout the application of 50,000 voltage pulses ($25 \times [1000 \uparrow 1000 \downarrow]$) of $\pm 3 \text{ V}$ and 0.1 s width over the course of $\approx 30 \text{ h}$.

Devices shown in **Fig.2-5** and characterized in **Fig.2-7** are the first prototypes of a back-end CMOS-compatible protonic programmable resistor. Conductance modulation, device scalability, and process control are ensured by the use of a thin nanoporous PSG layer, a common and CMOS-compatible material, as the proton electrolyte layer. This material choice has allowed downscaling to $\sim 5 \,\mu\text{m}$ as well as room temperature without the necessity of hydration. Therefore, it is clear that PSG can serve as a platform to explore alternative channel and hydrogen reservoir layers in protonic programmable resistors.

Indeed, an alternative channel material, V_2O_5 was tested using a PSG-based stack. Previously, an attempt of using V_2O_5 was also made with Nafion as the electrolyte, but ended up failing as the acidity of the Nafion damaged the active channel layer. Such an issue is not present with the PSG, allowing fabrication of operational devices as can be seen in **Fig.2-8**. Given the p-type nature of V_2O_5 in contrast to the n-type WO₃, the protonation of the channel material decreases the conductance (instead of increasing it). In the future, this property could potentially be utilized in complementary design elements, featuring both types of devices.



Figure 2-8: Modulation behavior of V_2O_5 -PSG-based protonic programmable resistors: Channel conductance is continuously read with $V_{DS} = 0.1 \text{ V}$ where 8 positive +1 V and 8 negative -1 V gate pulses are applied. Protonation of the channel reduces the conductance in a nonvolatile fashion, showing opposite direction to that of WO₃-based devices.

Despite the advancements described in this chapter so far, the overall device properties shown above (for both V_2O_5 and WO_3) are still highly suboptimal. Most importantly, the operation speed, 1 s), needs to be accelerated by ~8 orders of magnitude and the conductance modulation depth, $1.3 \times$, needs to be increased to > 10×. It will be shown in **Sec.2.7** that the poor performance is due to the active layer and not the PSG electrolyte. Moreover, microscale devices are still very large and need to be scaled down > 100×. This last issue is arguably the easiest to resolve (by employing a similar fabrication flow using electron-beam lithography), given the CMOS-compatible nature of the materials used.

2.7 Optimization of the WO₃ Layer for Protonic Programmable Resistors

An important observation for the devices made with amorphous ALD WO₃ is the high base (i.e. unprotonated) conductance of the channel material. It was later realized that previous literature has also reported that such substoichiometric materials showed poor electronic and optical modulation characteristics upon ion intercalation [58,85]. To further confirm the oxygen-deficient nature of the ALD-deposited WO₃, an XPS spectrogram is presented in **Fig.2-9**A (top). The W $4f_{5/2}$ and W $4f_{7/2}$ peaks deconvolution between the W6+ and W5+ oxidation states evidences a strong W⁵⁺ contribution ($\approx 1 \text{ W}^{5+}$: 4 W⁶⁺ ratio), indicative of the initial reduced (substoichiometric, WO_{3-x}) nature and in-gap states present of the film.

To mitigate this issue, the deposition method of WO₃ was optimized, with trials including various conditions of ALD, electron-beam evaporation, and sputtering. The material with superior modulation characteristics was found to result from reactive sputtering of WO₃ at room temperature, followed by a 400 °C annealing process that both oxidizes and crystallizes the material. The deposition conditions for the sputtering are as follows: W target, room temperature, 3 mTorr total pressure, 2.7/9.3 sccm O_2/Ar flux, 100 W RF power. Previous to the each deposition, target was also cleaned in pure Ar atmosphere for about 5 minutes, followed by preconditioning of the chamber for 20 minutes under the deposition conditions above. This process yielded a



Figure 2-9: **Optimization of the WO₃ layer [86]:** (A) X-ray photoelectron spectroscopy (XPS) of the ALD WO₃ (top) and reactive sputtered + annealed WO₃ (bottom). Annealing step fully oxidizes WO_{3-x} layer to fully-stoichiometric WO₃ which is has less base conductance and provides higher sensitivity for protonation. (B) X-ray diffraction (XRD) spectra of the reactive sputtered WO3 after annealing step showing characteristics of polycrystalline material in monoclinic phase. (C) Atomic force microscopy (AFM) image of the reactive sputtered WO₃ layer after the annealing step showing surface topography deviations below $\pm 1 \text{ nm}$ (D) Basic modulation behavior of a protonic device using the optimized WO₃ channel material (all other materials are identical to the ones discussed in Sec.2.4). Modulation is performed by applying 600, 1 ms, $\pm 4 \text{ V}$ voltage pulses in either direction. Each state is monitored for 5 ms, by applying $V_D S = 0.1 \text{ V}$ and measuring the channel current.

deposition rate $\sim 0.7-1 \text{ Å s}^{-1}$. Following the deposition (and patterning if necessary), the layer was annealed at 400 °C for 1 h under 8:2 N₂:O₂ atmosphere.

The XPS, XRD, and AFM results of the optimized WO₃ layer are given in **Fig.2-9**A (bottom), **Fig.2-9**B, and **Fig.2-9**C respectively. It can be seen that the resultant material is fully-stoichiometric as well as polycrystalline in monoclinic phase. To val-

idate the superior performance of the optimized WO₃ layer, a full device was made and characterized (analogous to the processes described in **Secs.2.4** and **2.6**). It is clear that the optimized material with lower base conductance has much larger dynamic range ($\approx 500 \times$) and higher modulation speed ($\sim 1 \text{ ms}$) than the earlier devices with ALD-deposited WO₃. Note that experimental results for devices made with other WO₃ flavors are not reported here as they were all inferior in performance or did not work at all, such as electron-beam evaporated, ALD-deposited with different oxygen precursors (H₂O and O₃), reactive sputtered at different O₂/Ar flux levels and RF power; all combined with different annealing conditions such as different temperatures and environments or not annealing at all. Although no definitive trend or dependency was detected, empirical results suggest that the optimized recipe above yields the highest reproducibility as well as the best performance.⁷

2.8 Fabrication of Nanoscale PSG-Based Protonic Programmable Resistors

Following the optimization of both PSG (Sec.2.3) and WO₃ (Sec.2.7) layers, a nanoscale device fabrication process based on electron beam lithography was engineered .

The nanoscale protonic programmable resistors were fabricated on a Si substrate covered with $10/90 \text{ nm HfO}_2/\text{Al}_2\text{O}_3$ deposited by atomic layer deposition (ALD) for electrical and protonic insulation (**Fig.2-10** (left)). The 10 nm WO₃ channel layer was deposited with the optimized reactive sputtering process described in **Sec.2.7**, patterned with liftoff followed by the annealing step. Channel contacts (35/5 nmAu/Cr) were then patterned using an aligned liftoff process followed by blanket deposition of 10 nm PSG electrolyte using the optimized plasma-enhanced chemical vapor deposition (PECVD) process discussed in **Sec.2.3**. The 10 nm Pd reservoir was followed by another liftoff process, and then used as a hardmask to etch PSG

⁷The author once again wants to thank Nicolas Émond for his immense help with the metrology and contributions regarding WO_3 optimization.



Figure 2-10: Fabrication of PSG-based microscale protonic programmable resistors [86]: (A) Photolithography-based fabrication flow of the nanoscale devices. (B)Transmission electron microscope image of the device crossection, focusing on the critical step of the fabrication flow (right). (C) Scanning electron microscopy (SEM) images of protonic devices with higher aspect ratios.

through reactive ion etching (RIE) using CF_4 plasma. Finally, the 150/10 nm Au/Cr gate pads were electron-beam evaporated and patterned through a photolithographybased liftoff process. Different channel dimensions (i.e. width and length) were patterned in the range between 20–1000 nm. Elemental analysis of the devices is shown in **Fig.2-12**, validating the accurate thicknesses and the placement of the respective layers.



Figure 2-11: Nanosecond protonic programmable resistors [86]: (A) 3-D illustration of the protonic programmable resistors studied in this work showing Au (yellow), WO3 (green), PSG (magenta), and Pd (grey) layers. As a result of an engineered sidewall, the Pd layer that overlaps with the channel electrodes is isolated from the remainder of the gate electrode. (B) False-colored top-view scanning electron microscope (SEM) image of a fabricated device with a 60×30 nm channel. (C) Transmission electron microscope (TEM) cross-section image of a protonic programmable resistor that had previously been extensively modulated.

The most critical feature of this process is the self-aligned gate structure employed to scale down device dimensions, avoiding mask alignment limitations. In this layout, the Pd layer is overlaid across a large region, whereas the height of the channel electrodes was calibrated such that the PSG layer can cover the sidewalls, while the Pd layer that overlaps with the channel electrodes is disconnected from the rest of the gate electrode (**Fig.2-10B**, and **Fig.2-11**C). As a result, this structure avoids field lines that do not pass through the channel material. This is intended to maximize energy efficiency. The top view SEM image and the crossectional TEM image of the finalized devices are shown in **Fig.2-11**.

Regarding the yield rate of this process, a total of 4056 devices were fabricated on 6 separate $1 \times 1 \text{ cm}^2$ chips with 4 different PSG and 2 different WO₃ thicknesses. Out of the 246 devices measured, 17 were shorted (the most aggressively scaled devices), 114 were too resistive to operate and 115 functioned successfully giving a functional
yield of 50%. Note that the yield is significantly higher, 75%, for the 55 devices with thicker WO₃ layer. Over 9 months of experimentation, only minor degradation effects were observed, where the base conductance of devices increased. This change can be attributed to self-protonation of the inactive devices due to proton leakage across PSG (as all Pd layers are protonated together in the gas environment, even for untested devices). Encapsulation of these devices should ultimately mitigate this issue (which would also require a new protonation method such as ion implantation into Pd, postencapsulation). On the other hand, devices showed no appreciable change when they were stored in N_2 box.



Figure 2-12: Elemental analysis of nanoscale protonic programmable resistors [86]: Energy Dispersive Spectroscopy (EDS) mapping in the transmission electron microscope (TEM) of the elements. The presence of Pt is due to its use as the protective layer during the focused-ion-beam milling step for the preparation of the lamella.⁹

2.9 Characterization of Nanoscale PSG-Based Protonic Programmable Resistors

Fig.2-13A shows the channel conductance modulation of a $50 \text{ nm} \times 150 \text{ nm}$ device with 10 nm thick PSG for 1000 protonating voltage pulses ($V^+ = 10 \text{ V}$) followed by



Figure 2-13: Ultrafast and energy-efficient modulation characteristics of nanoscale protonic programmable resistors [86]: (A) Modulation performance of a 50 nm×150 nm protonic device with 10 nm PSG showing fast (5 ns/pulse), nearly linear, and symmetric characteristics. (B) Retention behavior of the protonic device for ~100 s at different conductance levels over the full dynamic range. (C) Endurance characterization of the protonic device displaying non-degrading modulation over 10^5 pulses conducted over 30 h.

1000 deprotonating ones ($V^- = -8.5$ V). Between successive pulses, the channel conductance was read under $V_{DS}=0.1$ V and $I_G=0$ conditions and averaged for ~1s. The devices display nearly ideal characteristics in terms of: (1) high modulation speed, responding to 5 ns voltage pulses¹⁰, (2) fairly linear and symmetric behavior for incremental and decremental changes, (3) conductance retention characteristics over durations longer than ~ 10^{10} times the unit pulse time (**Fig.2-13**B), (4) dynamic conductance range of $20\times$, (5) optimal base resistance of 88 MΩ for readout [13], and (6) preservation of these ideal properties without any degradation over extended time and use (**Fig.2-13**C). Moreover, the devices show excellent energy efficiency under this ultrafast operation, the gate current supplied during each pulse being too small to be precisely measured for the small devices. The energy consumption during the transients is estimated to be ~2.5 fJ/pulse, which is a technology-agnostic overhead related to charging and discharging the gate capacitance. On the other hand, the energy consumed in proton transfer while the 5 ns voltage pulse is at its peak value

 $^{^{10}{\}rm Pulse}$ times are defined at the 90% of the maximum amplitude. Rise and fall times of all pulses (i.e. 0-80%) are 11.25 ns

is estimated to be $\sim 15 \text{ aJ/pulse}$ for the device shown in **Fig.2-13**. This latter value is associated with the efficient shuttling of ions within the gate stack under the high electric field. To the best of author's knowledge, no analog non-volatile memory technology has shown such ideally combined material, processing, and performance properties as the all-solid-state protonic devices produced here.



Figure 2-14: Energy estimation of nanoscale protonic devices [86]: $I_G - V_G$ curve for a large (1000 nm×750 nm) protonic device used to estimate the gate current of the smaller device from the same chip shown in **Fig.2-13**. At 10 V, average recorded gate current is 30.5 nA (marked with red lines).

The energy estimations above were calculated as following. The dynamic energy related to charging and discharging the capacitances are computed over $E = C_{gate}V_{gate}^2$, where the $Cgate = (\epsilon A_{gate})/d_{PSG}$ and $A_{gate} = 150 \times 50 \text{ nm}^2$, $d_{PSG} = 10 \text{ nm}$, and $\epsilon_{PSG} \approx 4\epsilon_0$. On the other hand, the energy related to the proton shuttling is computed as $E = t_{pulse} \times I_{avg} \times V_{pulse}$ where the t_{pulse} is the duration at which the voltage is $> 0.8 \times V_{pulse}$. To estimate the gate current and dynamic pulse energy, a larger device from the same chip (i.e. same PSG thickness) with W = 1000 nm and L = 750 nm was measured. At the same gate voltage of 10 V used in Fig.2-13A, an average gate current of 30.5 nA was recorded. Assuming area scaling, an estimate gate current of 300 pA should be passing through the device in Fig.2-13.

2.10 Extreme Electric Field Operation Regime for Protonic Programmable Resistors

In order to explain these excellent modulation characteristics, a model for device operation was developed comprising two key parts: (a) proton transport in the PSG and (b) proton-coupled electron transfer reaction rates at the PSG/electrode interfaces. Both processes have qualitatively similar formalisms and dependencies where the former is governed by hopping conduction in a disordered solid with random site energies [87–89], whereas the latter is determined by the Butler-Volmer charge transfer equation. Conventionally, the high activation energies for protons to be freed from their sites (e.g. Si-O-H or P-O-H) result in low ionic conductivity [90,91], and ultimately limit the operation speed at room temperature. For amorphous SiO₂ this activation energy was reported to be 0.38 meV which is much larger than the thermal energy in ambient conditions (25.9 meV) [92].

In the presence of a high electric field, this energy barrier to ion conduction is lowered in the field direction, yielding an enhanced proton hopping current:

$$I_G(V_{pulse}) \propto \sinh(\frac{q.a.V_{pulse}}{2kT.d_{PSG}})$$
(2.1)

where q is the electron charge, k the Boltzmann constant, T the temperature, a the apparent hopping distance, and d_{PSG} the thickness of the electrolyte [89,93]. Note that due to the high resistance of the PSG layer, all of the pulse voltage is assumed to drop across the electrolyte (i.e. $V_{PSG} \approx V_{pulse}$). Fig.2-15A shows the experimentally observed conductance change per pulse ($\Delta G_{channel}$) as a function of pulse amplitude (V_{pulse}) and pulse time (t_{pulse}) for a device with $d_{PSG} = 5 \text{ nm}$. Over a similar range of electric fields as the data in Fig. 2-13, the results of Fig.2-15 closely follow Eq.2.1 as indicated by the lines. Furthermore, experiments for devices with different d_{PSG} allow us to extract a hopping distance for the PSG of 5.6 Å (Fig.2-15B). This result is in good agreement with previously reported values for amorphous silica glasses [92, 94, 95].



Figure 2-15: Voltage dependence of conductance modulation [86]: (A) Channel conductance change per pulse ($\Delta G_{channel}$) for different pulse amplitude (V_{pulse}) and duration (t_{pulse}) pairs for a device with 5 nm PSG thickness. Points represent averaged experimental data over 1000 identical pulses for a given V_{pulse} - t_{pulse} pair whereas the meshed surface represents the fitted result. The *a* and β values used to fit the data are 5.6 Å and 1.2, respectively. (B) Estimation of the apparent hopping distance, a, from repeating the same experiment in Fig.2-13A for 25 devices selected over 4 different chips with different PSG thicknesses.

For the devices presented here, the electric field ($\approx 10 \,\mathrm{MV \, cm^{-1}}$ across the hopping distance $(\Delta U = qEa \approx 0.3 - 0.5 \,\mathrm{eV})$ is so high that it may completely remove the activation barrier within the PSG ($\approx 0.4 \,\mathrm{eV}$). This effect resolves the bottleneck of low proton conduction at room temperature, thus enabling high-speed operation. Indeed, such effect was predicted in simulations earlier [96], but has not been previously observed experimentally, as the required conditions are beyond the breakdown field or the electrochemical stability window of traditional electrolyte materials [97]. Instead, PSG allows a high critical field $(8-15 \,\mathrm{MV \, cm^{-1}})$ [98,99] in addition to a decent base proton conductivity [74, 77, 100, 101], making it an ideal electrolyte choice for this application. It should also be noted that the nonlinear $I_G - V_G$ characteristics in the gate stack are highly functional for device control. For example, for the halfselect programming scheme required for parallel updating, it is necessary that the modulation magnitude at V/2 is significantly less than that at V [13, 102], which is certainly satisfied by the exponential dependence shown in **Fig.2-15**A. Furthermore, the pulse time dependence of the protonation/deprotonation dynamics are empirically approximated as a power law ($\propto t_{pulse}^{1.2}$). The reasons behind this functional form should be investigated in a future study by decoupling the many coupled electronicionic dynamics inside the bulk materials and interfaces. Note that in Eq.2.1, V_{pulse} was used equivalent to the voltage across the PSG electrolyte (i.e. $V_{PSG} \approx V_{pulse}$). This estimation was justified by the results from devices with four different PSG thicknesses in Fig.2-16, showing overlapping modulation curves for V_{pulse}/d_{PSG} .



Figure 2-16: **Programming voltage dependence of PSG thickness for nanoscale protonic programmable resistance [86]:** Scaling of voltage linearly with PSG thickness, indicating that the PSG resistance is higher than the other resistances in the gate stack.

The ability to rapidly shuttle protons within bulk PSG at extreme speeds shifts the bottleneck to the interfaces. While the proton uptake rate of Pd is high [103] and the PdH_x/PSG interfacial reaction is likely also efficient, the same cannot be said for proton insertion into polycrystalline WO₃ [104] which shows up as PSG/WO₃ interfacial charge-transfer resistance. As protons do not have high diffusivity in WO₃ [105], it is likely that the inserted species cannot quickly vacate their sites near the interface which further reduces the insertion rate in a self-limiting fashion. These factors result in an excess of highly energetic protons at the WO₃-PSG interface and may lead to H₂ gas formation and buildup at the interface [106]. The electrical signature and morphological consequences of long-pulse-time stressing of the devices are captured in **Fig.2-17**. In **Fig.2-17**A, the same device previously studied in



Figure 2-17: Modulation dynamics for short and long pulse durations [86]: (A) Channel conductance modulation of the device presented in Fig.2-13 for increasing pulse duration under the same voltage. Between pulses, the channel conductance is read and averaged for ≈ 1 s. (B) SEM image of the device after the experiment. (C) SEM image of another device captured during early degradation. (D) Si-L_{2,3} EELS spectra of the PSG layer for tested (red, orange) and fresh (blue) devices.

Fig.2-13 is tested under the same V_{pulse} but for increasing t_{pulse} . Above 40 ns a cascading effect is apparent, where the conductivity change increases with each pulse, ultimately causing device failure for 90 ns pulses. An SEM image of the device after the experiment is shown in **Fig.2-17**B while **Fig.2-17**C shows an image of a device stopped at earlier stages (corresponding to the 40 ns pulse regime shown in **Fig.2-17**A). These and other images show damage features that are consistent with H₂ evolution, nanobubble formation and stress buildup.

Most importantly, this degradation is absent under fast-pulse operation, since the

PSG can capture free protons back once the electric field disappears. In addition to the good endurance characteristics shown in **Fig.2-13**C, as further evidence of reliable operation, electron energy loss spectroscopy (EELS) experiments were performed for the solid electrolyte in fresh and tested devices. **Fig.2-17**D shows that the Si $L_{2,3}$ energy loss spectra are similar in active and inactive regions across the PSG layer, which indicates there is no stoichiometry change [107] or the change is too subtle to be detected.

2.11 Conclusion

The two key contributions reported in this chapter are: (1) recognition of nanoporous phosphosilicate glass (PSG) as a CMOS-compatible protonic electrolyte and (2) discovery of the extreme-field operation for ultrafast modulation of nanoscale protonic devices with outstanding energy efficiency. PSG is an excellent choice for protonic programmable resistors due to its: (1) good electronic insulation, (2) high room temperature proton conductivity, (3) conduction mechanism based on P-doping (instead of structural water), and (4) ready availability in conventional Si-processing. The selection of this material has allowed using standard CMOS-fabrication techniques to scale down the device footprint. It was only later understood that PSG had another key property: high critical field. This final feature unlocked an unprecedented regime of nanosecond proton shuttling at room temperature without causing any material degradation.

Ultimately, this chapter presented the first CMOS- and BEOL-compatible nanoscale protonic programmable resistors with excellent characteristics that combine benefits of nanoionics with extreme acceleration of ion transport and reactions under strong electric fields. This operation regime enables controlled shuttling and intercalation of protons in nanoseconds at room temperature in an energy-efficient manner with nearideal modulation characteristics. As a result, the technology invented, optimized, and prototyped here has superior combined material, processing, and performance properties than the previous state-of-the-art non-volatile memories.

Chapter 3

Algorithms for Analog Deep Learning

Computational science covers a wide-range of approaches that lie in between theory and experimentation, studying and predicting behavior where the former is insufficient and the latter is impractical. These methods have been integral at frontiers of scientific discovery, including decoding the human genome, advancing drug discovery, and unraveling cosmology. Given that large-scale applications as such require analyzing of high volumes of data, processing information in an accurate, fast, and energy-efficient manner is a key requirement for the continued progress of fundamental and applied sciences.

For the last ~ 80 years, the ability to tackle ever-larger problems has been enabled by the codevelopment of more powerful computing machines and algorithms. Even though in 1960s analog processors were once considered promising, almost all computation since have been carried out with digital architectures, based on Boolean-logic implemented on CMOS-circuitry. The biggest reason behind this choice has been the erroneous nature of analog architectures, based on intrinsically noisy elements with high variability, in contrast to near-faultless operation of their digital counterparts. However, if one can design more fault-tolerant algorithms, then the analog architectures have unique properties to offer, which can provide unprecedented performance benefits.

Deep learning is by far the most popular application with considerable faulttolerance, as training operations employ a feedback loop that can correct singular mistakes post-facto (i.e. instead of building up on it). Moreover, given that the computational requirements for large-scale DNN training have been exponentially increasing with a ~ 3 month doubling time, there is a dire need to find alternative ways to support the extraordinary success of artificial intelligence. Having said that, neural network training is not the only application that can be accelerated via analog processors. This chapter will explore strengths and weaknesses of analog computing, specifically analog deep learning, in order to assess potential application fields and design specialized algorithms that operate efficiently and accurately on these processors.

3.1 Analog Deep Learning Based on Stochastic Gradient Descent

Neural networks can be construed as many layers of matrices (i.e. weights, W) performing affine transformations followed by nonlinear activation functions. Training (i.e. learning) process refers to the adjustment of W such that the network response to a given input produces the target output for a labeled dataset. The discrepancy between the network and target outputs is represented with a scalar error function, E, which the training algorithm seeks to minimize. In the case of the conventional SGD algorithm [108], values of W are incrementally modified by taking small steps (scaled by the learning rate, η) in the direction of the gradient of the error function sampled for each input. Computation of the gradients is performed by the backpropagation algorithm consisting of forward pass, backward pass, and update subroutines [12] (**Fig.3-1**A). When the discrete nature of DNN training is analyzed in the continuum limit, the time evolution of W can be written as a Langevin equation:

$$\dot{W} = -\eta \left[\frac{\partial E}{\partial W} + \epsilon(t) \right] \tag{3.1}$$

where η is the learning rate and $\epsilon(t)$ is a fluctuating term with zero-mean, accounting for the inherent stochasticity of the training procedure [109]. As a result of this training process, W converges to the vicinity of an optimum W_0 , at which $\frac{\partial E}{\partial W} = 0$ but \dot{W} is only on average 0 due to the presence of $\epsilon(t)$. For visualization, if the training dataset is a cluster of points in space, W_0 is the center of that cluster, where each individual point still exerts a force ($\epsilon(t)$) that averages out to 0 over the whole dataset.



Figure 3-1: Implementation of the Stochastic Gradient Descent algorithm on analog crossbar arrays [110]: Vectors x, y represent the input and output vectors in the forward pass whereas δ, z contain the backpropagated error information. The analog architecture schematic is only shown for a single layer, where all vectors are propagated between upper and lower network layers in general. The pseudocode only describes operations computed in the analog domain, whereas digital computations such as activation functions are not shown for simplicity.

In the case of analog crossbar-based architectures, the linear matrix operations are performed on arrays of physical devices, whereas all nonlinear computations (e.g. activation and error functions) are handled at peripheral circuitry (See **Fig.1-1**). The strictly positive nature of device conductance requires representation of each weight by means of the differential conductance of a pair of crosspoint elements (i.e. $W \propto G_{main} G_{ref}$). Consequently, vector-matrix multiplications for the forward and backward passes are computed by using both the main and the reference arrays (**Fig.3-1**A). On the other hand, the gradient accumulation and updates are only performed on the main array using bidirectional conductance changes while the values of the reference array are kept constant **Fig.3-1**. In this section, to illustrate the basic dynamics of DNN training with analog architectures, a single-parameter optimization problem is studied (linear regression) which can be considered as the simplest "neural network".

The weight updates in analog implementations are carried out through modulation of the conductance values of the crosspoint elements, which are often applied by means of pulses. These pulses cause incremental changes in device conductance $(\Delta G^{+,-})$. In an ideal device, these modulation increments are of equal magnitude in both directions and independent of the device conductance, as shown in **Fig.3-2**A. It should be noted that the series of modulations in the training process is inherently non-monotonic as different input samples in the training set create gradients with different magnitudes and signs in general. Furthermore, as stated above, even when an optimum conductance, G_0 , is reached ($W_0 \propto G_0 - G_{ref}$), continuing the training operation would continue modifying the conductance in the vicinity of G_0 , as shown in **Fig.3-2**B. Consequently, G_0 can be considered as a dynamic equilibrium point of the device conductance from the training algorithm point of view.



Figure 3-2: Linear regression under SGD algorithm using symmetric crosspoint elements [110]: (A)Sketch of conductance modulation behavior of a symmetric crosspoint device. (B) Simulated single-parameter optimization result for the symmetric device. Conductance successfully converges to the optimal value for the problem at hand, G_0 .

3.1.1 Analysis of Asymmetry Caused Effects Under SGD

As covered in **Chap.2**, infamously, most analog resistive devices do not show the ideal characteristics illustrated in **Fig.3-2**A. Instead, many technologies display asymmetric conductance modulation characteristics such that unitary (i.e. single-pulse) modulations in opposite directions do not cancel each other in general, i.e., $\Delta G^+(G) \neq$ $\Delta G^-(G)$. However, with the exception of some device technologies such as Phase Change Memory (PCM) which reset abruptly [16, 18, 23], many crosspoint elements can be modeled by a smooth, monotonic, nonlinear function that shows saturating behavior at its extrema as shown in **Fig.3-3**A [52, 58, 111]. For such devices, there exists a unique conductance point, $G_{symmetry}$, at which the magnitude of an incremental conductance change is equal to that of a decremental one. As a result, the time evolution of G during training can be rewritten as:

$$\dot{G} = -\eta \left[\frac{\partial E}{\partial G} + \epsilon(t) \right] - \eta \kappa \left| \frac{\partial E}{\partial G} + \epsilon(t) \right| f_{hardware}$$
(3.2)

where κ represents the magnitude and $f_{hardware}$ gives the functional form of the device asymmetry (See Sec.3.1.2). In this expression, the term $-\eta \left| \frac{\partial E}{\partial G} + \epsilon(t) \right|$ signifies that the direction of the change related to asymmetric behavior is solely determined by $f_{hardware}$, irrespective of the direction of the intended modulation. For the exponentially saturating device model shown in Fig.3-3A, $f_{hardware} = G - G_{symmetry}$, which indicates that each and every update event has a component that drifts the device conductance towards its symmetry point. A simple observation of this effect is when enough equal number of incremental and decremental changes are applied to these devices in a random order, the conductance value converges to the vicinity of $G_{symmetry}$ [111]. Therefore, this point can be viewed as the physical equilibrium point for the device, as it is the only conductance value that is dynamically stable.



Figure 3-3: Linear regression under SGD algorithm using asymmetric crosspoint elements [110]: (A) Sketch of conductance modulation behavior of an asymmetric crosspoint device. The point at which $\Delta G^+ = \Delta G^-$ is defined as the symmetry point of the device ($G_{symmetry}$). (B) Simulated training result for the same singleparameter optimization with the asymmetric devices. Device conductance fails to converge to G_0 , but instead settles at a level between G_0 and $G_{symmetry}$.

It is essential to realize that there is in general no relation between $G_{symmetry}$ and

 G_0 , as the former is entirely device-dependent while the latter is problem-dependent. As a result, for an asymmetric device, two equilibria of hardware and software create a competing system, such that the conductance value converges to a particular conductance somewhere between $G_{symmetry}$ and G_0 , for which the driving forces of the training algorithm and device asymmetry are balanced out. (Fig.3-3B). In the examples shown in Fig.3-2B and Fig.3-3B, G_0 of the problem is purposefully designed to be far away from $G_{symmetry}$, so as to depict a case for which the effect of asymmetry is pronounced. Indeed, it can be seen that the discrepancy between the final converged value, G_{final} , and G_0 strongly depends on the relative position of G_0 with respect to the $G_{symmetry}$ (Fig.3-4B), unlike that of ideal devices (Fig.3-1A).



Figure 3-4: Problem-specific residual error dependence for asymmetric devices trained under SGD algorithm [110]: (A) Simulated residual distance between the final converged value, G_{final} , and G_0 for training the device with symmetric characteristics shown in Fig.3-2A for datasets with different optimal values. (B) The same experiment conducted for asymmetric devices shown in Fig.3-3A.

3.1.2 Mathematical Modeling of Device Asymmetry Under SGD

Exponentially saturating conductance modulation (per pulse) of devices with the characteristics shown in **Fig.3-3**A can be described with the following equations:

$$\Delta G^+(G) = \Delta G^+(G_{symmetry}) \times (1 - \kappa (G - G_{symmetry})) \tag{3.3}$$

$$\Delta G^{-}(G) = \Delta G^{-}(G_{symmetry}) \times (1 + \kappa (G - G_{symmetry}))$$
(3.4)

By definition, $\Delta G^+(G_{symmetry}) = \Delta G^-(G_{symmetry})$ and can be considered as the unit of conductance change per single pulse (consistent with the notation for a symmetric device, which is also sometimes referred as ΔG_{min} to indicate it is ultimately the resolution of the update). As a result, two equations can be combined as:

$$\Delta G^{real}(G) = \Delta G^{intended} - \kappa |\Delta G^{intended}| \times (G - G_{symmetry})$$
(3.5)

assuming intended updates are small in magnitude (i.e. $\Delta G^{intended} \sim \pm \Delta G_{min}$).

To capture the evolution of conductance during a training operation, the discrete updates applied on the device are investigated first. Assuming the initial point (not to be confused with the optimal point) $G_0 = G_{symmetry}$, for a learning rate η :

$$G_1 = G_0 + \eta \Delta G_1 \tag{3.6}$$

$$G_2 = G_1 + \eta \Delta G_2 - \eta \kappa |\Delta G_2| (G_1 - G_{symmetry})$$
(3.7)

$$G_k = G_{k-1} + \eta \Delta G_k - \eta \kappa |\Delta G_k| (G_{k-1} - G_{symmetry})$$
(3.8)

where the intended updates ΔG_k are rank-1 portions of the update matrix to be applied on the weight matrix. Note that each ΔG is actually applied as a series of incremental updates in the units of minimum conductance change of the devices. Due to asymmetry, each successive update is affected by all preceding updates (and their order). It is important to realize that all ΔG_i are computed such that they correspond to the derivative of the error function with respect to G_{i-1} (i.e. $\sum_k \Delta G_k \propto -\eta \frac{\partial E}{\partial G}$).

Furthermore, these updates also involve a component $\epsilon(t)$, which accounts for the inherent stochasticity of this process, sampling the true gradient. Rewriting **Eq.3.8** to reflect these properties:

$$G_k = G_{k-1} + \eta \left[\frac{\partial E}{\partial G_{k-1}} + \epsilon(t) \right] - \eta \kappa \left| \frac{\partial E}{\partial G_{k-1}} + \epsilon(t) \right| (G_{k-1} - G_{symmetry})$$
(3.9)

which in return can be written in the continuum limit as:

$$\dot{G} = -\eta \left[\frac{\partial E}{\partial G} + \epsilon(t) \right] - \eta \kappa \left| \frac{\partial E}{\partial G} + \epsilon(t) \right| (G - G_{symmetry})$$
(3.10)

as previously shown in **Eq.3.2**, for $f_{hardware} = G - G_{symmetry}$.

A powerful method to prove that asymmetric devices are fundamentally incompatible to be trained under SGD operation is showing that the optimum point for an arbitrary optimization problem is not even stable in general for an asymmetric device modulated using SGD-computed updates.

For the single parameter optimization described in Sec.3.1.1 to be convergent,

$$\lim_{t \to \infty} \langle \dot{G} \rangle = 0 \tag{3.11}$$

meaning that the conductance value will settle around the vicinity of a certain level, which is labeled here as G_{final} . For the case of a single-parameter linear regression example, G_{final} represents the mean of the dataset.

It is important to realize that the updates over G are computed using the scalar error function E of the optimization problem $(E \propto (G - G_0)^2)$. Therefore, ideally, when $G \leftarrow G_{final}$:

$$\langle \dot{G} \rangle \propto \left\langle \frac{\partial E}{\partial G} \right\rangle = 0$$
 (3.12)

for a successful training operation. However, for an asymmetric device, rewriting **Eqn.3.10**, at convergence yields:

$$\left\langle \dot{G} \right\rangle = -\eta \left[\left\langle \frac{\partial E}{\partial G} \right\rangle + \left\langle \epsilon(t) \right\rangle \right] - \eta \kappa \left\langle \left| \frac{\partial E}{\partial G} + \epsilon(t) \right| \left(G - G_{symmetry} \right) \right\rangle$$
(3.13)

It is known that $\langle \epsilon(t) \rangle = 0$ by definition, and $\langle \dot{G} \rangle = 0$ from Eqn.3.12. Substituting both yields:

$$\left\langle \frac{\partial E}{\partial G} \right\rangle = -\kappa \left\langle \left| \frac{\partial E}{\partial G} + \epsilon(t) \right| (G - G_{symmetry}) \right\rangle$$
 (3.14)

Eqn.3.14 is the analytical expression, describing the competing-forces discussed in Fig.3-3 in the main text. The left-hand side of the equation describes the force exerted by the optimization process, trying to set $G_{final} \rightarrow G_0$ (optimum of the problem), whereas the right-hand side is the force that pulls $G_{final} \rightarrow G_{symmetry}$. As shown in Fig.3-3, these two forces balance out one another to ultimately satisfy Eqn.3.11, but this point bears no significance for the optimization problem at hand (i.e. it does not satisfy Eqn.3.12).



Figure 3-5: Interaction of device asymmetry and dataset standard deviation [110]: Resultant optimization error as a function of standard deviation of the input data for the single-parameter optimization example discussed in Sec.3.1.

Analytically solving **Eqn.3.14** to find G_{final} at steady state proves to be challenging, particularly due to the unknown nature of $\epsilon(t)$ for each different case. Nonetheless, the following observations can be made:

- The state G_{final} = G₀ cannot be maintained at steady state, for which ⟨G⟩ ≠ 0, unless G₀ coincides with G_{symmetry}. As distance between G₀ and G_{symmetry} increases, G_{final} stabilizes further away from G₀ (Fig.3-4B).
- 2. Increasing magnitude of stochasticity (and circuit noise elements which are not included here for simplicity), and its distribution, strays G_{final} further away from G_0 (Fig.3-5).

3. Increasing amount of asymmetry (here represented with κ) makes G_{final} stabilize further away from G_0 .

As a result, asymmetric devices cannot be used to perform training tasks with SGD, as the optimal point is not even a dynamically stable point for asymmetric devices to stay at during training.

3.1.3 Existing Methods to Battle Asymmetry-Related Accuracy Degradation

Despite numerous published simulated and experimental demonstrations, none of these studies so far provides a solution for which the analog processor still achieves its original purpose: energy-efficient acceleration of deep learning. The critical issue with the existing techniques is the requirement of serial accessing to crosspoint elements one-by-one or row-by-row [16-23]. Methods involving serial operations include reading conductance values individually, engineering update pulses to artificially force symmetric modulation, and carrying or resetting weights periodically. Furthermore, some approaches offload the gradient computation to digital processors, which not only requires consequent serial programming of the analog matrix, but also bears the cost of outer product calculation [19–23]. Updating an $N \times N$ crossbar array with these serial routines would require at least N or even N^2 operations. For practical array sizes, the update cycle would simply take too much computational time and energy. In conclusion, for implementations that compromise parallelism, whether or not the asymmetry issue is resolved becomes beside the point since computational throughput and energy efficiency benefits over conventional digital processors are lost for practical applications.

Examining the source of problem, one can realize that a method to reduce the effect of $\epsilon(t)$ would be to adopt a momentum-based method (e.g. Momentum-SGD, Adam, AdaGrad, RMSProp). Such a method would certainly ameliorate the issue, as it would strengthen the gradient, but it cannot be implemented in a parallel fashion. Momentum SGD requires previous update matrices in the computation of the next update matrix. This indicates that all update matrices are computed and stored explicitly (the outer product operation is executed in digital domain). Instead, the parallel update method computes and applies those update matrices, without returning the result of the outer product to the user. Therefore, it cannot be stored for next update matrices for applying a gradient method. Furthermore, even if one were to read the entire matrix before and after to generate the update matrix post-facto, it would still not be enough to use a momentum method with parallel operations, as there are element-wise operations required for the momentum SGD. It is therefore urgent to devise a method that deals with device asymmetry while employing only fully-parallel operations.

Recently, a novel fully-parallel training method, *Tiki-Taka*, was proposed to successfully train DNNs based on asymmetric resistive devices with asymmetric modulation characteristics [112]. This algorithm was empirically shown in simulation to deliver ideal-device-equivalent classification accuracy for a variety of network types and sizes emulated with asymmetric device models. However, the missing theoretical underpinnings of the proposed algorithmic solution as well as the cost of doubling analog hardware previously limited the method described in Ref. [112].

3.2 Analog Deep Learning with Stochastic Hamiltonian Descent

In contrast to SGD, the Stochastic Hamiltonian Descent (SHD) algorithm, illustrated in **Fig.3-6**, is a fully-parallel training algorithm that separates both the forward path and error backpropagation from the update function. For this purpose, two array pairs (instead of a single pair), namely A_{main} , A_{ref} , C_{main} , C_{ref} are utilized to represent each layer [112]. In this representation, $A = A_{main} - A_{ref}$ stands for the auxiliary array and $C = C_{main} - C_{ref}$ stands for the core array.

The new training algorithm operates as follows. At the beginning of the training process, A_{ref} and C_{ref} are initialized to $A_{main,symmetry}$ and $C_{main,symmetry}$, respectively



Figure 3-6: Implementation of the Stochastic Hamiltonian Descent algorithm on analog crossbar arrays [110]: Schematic and pseudocode of training process using the SHD algorithm. The pseudocode only describes operations computed in the analog domain, whereas digital computations such as nonlinear error functions are not shown for simplicity.

(reasons will be clarified later), following the method described in Ref. [111]. As illustrated in **Fig.3-6**, first, forward and backward pass cycles are performed on the array-pair C (Steps I and II), and corresponding updates are performed on A_{main} (scaled by the learning rate η_A) using the parallel update scheme discussed in Ref. [13] (Step III). In other words, the updates that would have been applied to C in a conventional SGD scheme are directed to A instead.

Then, every τ cycles, another forward pass is performed on A, with a vector u, which produces v = Au (Step IV). In its simplest form, u can be a vector of all "0"s but one "1", which then makes v equal to the row of A corresponding to the location of "1" in u. Finally, the vectors u and v are used to update C_{main} with the same parallel update scheme (scaled by the learning rate η_C) (Step V). These steps (IV and V shown in **Fig.3-6**A) essentially partially add the information stored in A to C_{main} .

At the end of the training procedure C alone contains the optimized network, to be later used in inference operations (hence the name core). Since A receives updates computed over $\frac{\partial E}{\partial C}$, which have zero-mean once C is optimized, its active component, A_{main} , will be driven towards $A_{main,symmetry}$. The choice to initialize the stationary reference array, A_{ref} , at $A_{main,symmetry}$ ensures that A = 0 at this point (i.e. when C is optimized), thus generating no updates to C in return.

With the choice of u vectors made above, every time steps IV and V are performed, the location of the "1" for the u vector would change in a cyclic fashion, whereas in general any set of orthogonal u vectors can be used for this purpose [112]. Note that these steps should not be confused with weight carrying [16,17], as C is updated by only a fractional amount in the direction of A as $\eta_C \ll 1$ and at no point information stored in A is externally erased (i.e. A is never reset). Instead, A and C create a coupled-dynamical-system, as the changes performed on both are determined by the values of one another.

Furthermore, it is critical to realize that the algorithm shown in **Fig.3-6** consists of only fully-parallel operations. Similar to steps I and II (forward and backward pass on C), steps IV is yet another matrix-vector multiplication that is performed by means of Ohm's and Kirchhoff's Laws. On the other hand, the update steps III and V are performed by the stochastic update scheme [13]. This update method does not explicitly compute the outer products $(x \times \delta \text{ and } u \times v)$, but instead uses a statistical method to modify all weights in parallel proportional to those outer products. As a result, no serial operations are required at any point throughout the training operation, enabling high throughput and energy efficiency benefits in deep learning computations.

3.2.1 Mathematical Modeling of Device Asymmetry Under SHD

This section will parallel the derivations carried out in Sec.3.1.2 and demonstrate the nature of the critical difference between SHD and SGD that resolves the aforementioned stability problem.

For SHD, $A = A_{main} - A_{ref}$ gets updates computed over $C = C_{main} - C_{ref}$. Writing the evolution of A in terms of the PDE given in **Eqn.3.10**:

$$\dot{A}_{main} = \eta_A \Big[\frac{\partial E}{\partial (C_{main} - C_{ref})} + \epsilon(t) \Big] - \eta_A \kappa_A \Big| \frac{\partial E}{\partial (C_{main} - C_{ref})} + \epsilon(t) \Big| (A_{main} - A_{main,symmetry})$$
(3.15)

$$\dot{A}_{ref} = 0 \tag{3.16}$$

Writing the same for C which is updated by means of partial additions of A:

$$\dot{C}_{main} = \eta_C (A_{main} - A_{ref}) - \eta_C \kappa_C | (A_{main} - A_{ref}) | (C_{main} - C_{main,symmetry}) \quad (3.17)$$

$$\dot{C}_{ref} = 0 \tag{3.18}$$

Note that in the actual discrete time evolution of these systems, the time-steps of A_{main} and C_{main} are not necessarily the same, due to the presence of τ , which is ignored here for simplicity. In steady state, both A_{main} and C_{main} will converge to the vicinity of certain values $(A_{main,final}, C_{main,final})$, for which $\langle \dot{A}_{main} \rangle = \langle \dot{C}_{main} \rangle = 0$. Therefore, taking the time averages of **Eqns.3.15** and **3.17** in steady state, and substituting $\langle \epsilon(t) \rangle = 0$ yield:

$$\left\langle \frac{\partial E}{\partial (C_{main} - C_{ref})} \right\rangle = \kappa_A \left\langle \left| \frac{\partial E}{\partial (C_{main} - C_{ref})} + \epsilon(t) \right| \left(A_{main} - A_{main,symmetry} \right) \right\rangle$$
(3.19)

$$\langle A_{main} \rangle - A_{ref} = \kappa_C \langle |(A_{main} - A_{ref})| \cdot (C_{main} - C_{main,symmetry}) \rangle$$
(3.20)

The ultimate goal of any optimization task is to achieve and maintain $G_{final} \approx G_0$ at steady state, which translates to left-hand side of **Eqn.3.19** (similar to **Eqn.3.14**) being 0. For **Eqn.3.14**, this was not possible as the right-hand side was non-zero in general for that state, making the optimum point unstable. The key difference of SHD is that since A_{main} and C_{main} are different parameters, left hand side of **Eqn.3.19** can be satisfied by $C_{main} \approx C_0$ whereas right-hand side can be 0 when $A_{main} \approx A_{main,symmetry}$ in steady state.

On the other hand, examination of **Eqn.3.20** reveals a critical requirement for SHD to work, which is setting $A_{ref} = A_{main,symmetry}$ (i.e. zero shifting, [111]). Under this condition and an appropriately small choice of η_A , $\langle A_{main} \rangle - A_{ref} \approx \langle |A_{main} - A_{ref}| \rangle \approx 0$ since A_{main} changes in the very close vicinity of $A_{main,symmetry}$ in steady state. Furthermore, $\langle A \rangle \approx 0$ also indicates that the change in C_{main} is negligible since the value of A is the driving force of C_{main} . This property allows us to treat $(C_{main} - C_{main,symmetry})$ in **Eqn.3.20** as a constant and take the term out of the averaging operator.

Overall, SHD does not suffer from SGD's fundamental incompatibility with asymmetric devices and therefore, asymmetric devices can be used in optimization tasks under SHD-based training.

3.2.2 Analysis of Device Asymmetry Under SHD

For the same linear regression problem studied in Sec.3.1.2, the discrete-time update rules given in Fig.3-6A can be rewritten as a pair of differential equations in the continuum limit that describe the time evolution of subsystems A and C as:

$$\dot{A} = -\eta_A \Big[\frac{\partial E}{\partial C} + \epsilon(t) - \eta_A \kappa_A \Big| \frac{\partial E}{\partial C} + \epsilon(t) \Big| (A_{main} - A_{main,symmetry})$$
(3.21)

$$\dot{C} = -\eta_C A + \eta_C \kappa_C |A| (C_{main} - C_{main,symmetry})$$
(3.22)

It can be noticed that this description of the coupled system has the same arrangement as the equations governing the motion of a damped harmonic oscillator (Fig.3-7). In this analogy, subsystem A corresponds to velocity, ν , while subsystem C maps to position, x, allowing the scalar error function of the optimization problem, $(C - C_0)^2$, to map onto the scalar potential energy of the physical framework, $1/2k_{spring}(x - x_0)^2$. Moreover, for implementations with asymmetric devices, an additional force term, $F_{hardware}$, needs to be included in the differential equations to reflect the hardware-induced effects on the conductance modulation. As discussed earlier, for the device model shown in **Fig.3-3**A this term is proportional to $A_{main} - A_{main,symmetry}$. Assuming $A_{ref} = A_{main,symmetry}$ (this assumption will be explained later), $F_{hardware}$ can be rewritten as a function of $A_{main} - A_{ref}$, which then resembles a drag force, F_{drag} , that is linearly proportional to velocity ($\nu \propto A = A_{main} - A_{ref}$) with a variable (but strictly nonnegative) drag coefficient k_{drag} . In general, the $F_{hardware}$ term can have various functional forms for devices with different conductance modulation characteristics but is completely absent for ideal devices. Note that, only to simplify the physical analogy, the effect of asymmetry in subsystem C is ignored here, which yields the equation shown in **Fig.3-7** (instead of **Eq.3.22**). This decision will be justified and derived in detail in **Sec.3.2.1**.

Single-Parameter Optimization with Asymmetric Device



Figure 3-7: Physical analogy between SHD algorithm and damped harmonic oscillator [110]: Differential equations describing the evolution of the parameters with the SHD training algorithm in the continuum limit, in comparison to those of motion describing the dynamics of a harmonic oscillator

Analogous to the motion of a lossless harmonic oscillator, the steady-state solution for this modified optimization problem with ideal devices (i.e. $F_{hardware} = 0$) has an oscillatory behavior (**Fig3-8A**). This result is expected, as in the absence of any dissipation mechanism, the total energy of the system cannot be minimized (it is constant) but can only be continuously transformed between its potential and kinetic components. On the other hand, for asymmetric devices, the dissipative force term $F_{hardware}$ gradually annihilates all energy in the system, allowing $A \propto \nu$ to converge to 0 ($E_{kinetic} \rightarrow 0$) while $C \propto x$ converges to $C_0 \propto x_0 (E_{potential} \rightarrow 0)$. Based on these observations, the new training algorithm is renamed as Stochastic Hamiltonian Descent (SHD) to highlight the evolution of the system parameters in the direction of reducing the system's total energy (Hamiltonian). These dynamics can be visualized by plotting the time evolution of A versus that of C, which yields a spiraling path representing decaying oscillations for the optimization process with asymmetric devices (**Fig3-8B**), in contrast to elliptical trajectories observed for ideal lossless systems (**Fig3-8A**).

Following the establishment of the necessity to have dissipative characteristics, here the conditions at which device asymmetry provides this behavior is analyzed. It is well-understood in mechanics that for a force to be considered dissipative, its product with velocity (i.e. power) should be negative (otherwise it would imply energy injection into the system). In other words, the hardware-induced force term $F_{hardware} = -\kappa_A \eta_A \left| \frac{\partial E}{\partial C} + \epsilon(t) \right| (A_{main} - A_{main,symmetry})$ and the velocity, $\nu = A_{main} - A_{ref}$, should always have opposite signs. Furthermore, from the steady-state analysis, for the system to be stationary ($\nu = 0$) at the point with minimum potential energy ($x = x_0$), there should be no net force (F = 0). Both of these arguments indicate that, for the SHD algorithm to function properly, A_{ref} must be set to $A_{main,symmetry}$. Note that as long as the crosspoint elements are realized with asymmetric devices (opposite to SGD requirement) and a symmetry point exists for each device, the shape of their modulation characteristics is not critical for successful DNN training with the SHD algorithm.

A critical aspect to note is that the SGD and the SHD algorithms are funda-



Figure 3-8: Comparison of single parameter regression training under SHD algorithm for symmetric and asymmetric devices [110]: (A)Simulated results for a single-parameter optimization task using the SHD algorithm with symmetric devices described in Fig.3-2A. B The same experiment conducted for asymmetric devices described in Fig.3-3A.

mentally disjunct methods governed by completely different dynamics. The SGD algorithm attempts to optimize the system parameters while disregarding the effect of device asymmetry and thus converges to the minimum of a wrong energy function. On the other, the system variables in an SHD-based training do not conventionally evolve in directions of the error function gradient, but instead, are tuned to minimize the total energy incorporating the hardware-induced terms. The most obvious manifestation of these properties can be observed when the training is initialized from the optimal point (i.e. the very lucky guess scenario) since any "training" algorithm should at least be able to maintain this optimal state. For the conventional SGD, when $W = W_0$, the zero-mean updates applied to the network were shown above to

drift W away from W_0 towards $W_{symmetry}$. On the other hand, for the SHD method, when A = 0 and $C = C_0$, the zero-mean updates applied on A do not have any adverse effect since A_{main} is already at $A_{main,symmetry}$ for A = 0. Consequently, no updates are applied to C either as $\dot{C} = A = 0$. Therefore, it is clear that SGD is fundamentally incompatible with asymmetric devices, even when the solution is guessed correctly from the beginning, whereas the SHD does not suffer from this problem. Note that the propositions made for SGD can be further generalized to other crossbar-compatible training methods such as equilibrium propagation [113] and deep Boltzmann machines [114], which can also be adapted to be used with asymmetric devices following the approach discussed here.

Finally, it is obvious large-scale neural networks are much more complicated systems with respect to the problem analyzed here. Similarly, different analog devices show a wide range of conductance modulation behaviors, as well as bearing other non-idealities such as analog noise, imperfect retention, and limited endurance as explained in detail in **Chap.2**. However, the theory described here finally provides an intuitive explanation for: (1) why device asymmetry is fundamentally incompatible with SGD-based training and (2) how to ensure accurate optimization while only using fully-parallel operations. The author thereby concludes that asymmetry-related issues within SGD should be analyzed in the context of competing equilibria, where the optimum for the classification problem is not even a stable solution at steady-state. In addition to this simple stability analysis, the insight to modify the optimization landscape to include nonideal hardware effects allows other fully-parallel solutions to be designed in the future using advanced concepts from optimal control theory. As a result, these parallel methods enable analog processors to provide high computational throughput and energy efficiency benefits over their conventional digital counterparts.

3.2.3 Hardware Cost Reduction of SHD Implementation

Considering a sequence of m + n incremental and n decremental changes at random order, the net modulation obtained for a symmetric device is on average m. On the other hand, for asymmetric devices the conductance value eventually converges to the symmetry point for increasing n (irrespective of m or the initial conductance). It can be seen by inspection that for increasing statistical variation present in the training data (causing more directional changes for updates), the effect of device asymmetry gets further pronounced, leading to heavier degradation of classification accuracy for networks trained with conventional SGD (See **Fig.3-5**). However, this behavior can alternatively be viewed as nonlinear filtering, where only signals with persistent sign information, m/(m+2n), are passed. Indeed, the SHD algorithm exploits this property within the auxiliary array, A, which filters the gradient information that is used to train the core array, C. As a result, C is updated with less frequency and only in directions with a high confidence level of minimizing the error function of the problem at hand. A direct implication of this statement is that the asymmetric modulation behavior of C is much less critical than that of A for successful optimization as its update signal contains less amount of statistical variation. Therefore, symmetry point information of C_{main} is not relevant either. This property is further evidenced, in Fig.3-9 when a mismatch of various degrees is intentionally introduced between the reference array and the symmetry values of its respective main array. It can be seen that while such a mismatch leads to classification performance degradation for A, such an effect is completely absent for C. Using these results and intuition, the original algorithm is modified by discarding C_{ref} and using A_{ref} (set to $A_{main,symmetry}$) as a common reference array for differential readout. This modification reduces the hardware cost of SHD implementations by 50% to significantly improve their practicality [115].

3.2.4 Experimental Demonstration of the SHD Algorithm

A small scale experimental demonstration was made with metal-oxide based electrochemical devices (also called as electrochemical random access memory, ECRAM) reported in Ref. [52] (**Fig.3-10**A). These devices are three-terminal, voltage-controlled crosspoint elements, and can be considered as an oxygen-ion based alternative to the devices presented in **Chap.2**. For the demonstration of the modified training algorithm, a 2-parameter optimization problem was chosen with a synthetic dataset $x_{1,2}$ and y of form $y = t_1x_1 + t_2x_2 + \gamma$, where $t_{1,2}$ are the unknowns searched for and γ is



Figure 3-9: Reference array initialization sensitivity of subsystems A and C [110]: (A) The learning curve for different levels (i.e. standard deviation, σ) of discrepancy introduced between A_{ref} and $A_{symmetry}$, while C_{ref} is accurately initialized to $C_{symmetry}$. (B) The same experiment conducted for C_{ref} . Training simulations were performed for the convolutional neural network on MNIST dataset, which was previously studied in Ref. [10]. The average baseline error (only the final value) for ideal devices with conventional SGD is shown with the dashed line.

the Gaussian noise.

As can be seen from the connection scheme shown in **Fig.3-10**C, A_{main} and C_{main} are represented with conductance values of physical devices, while the reference arrays containing symmetry point information are stored in digital (as they remain unchanged throughout the training). Note that allocating reference arrays in digital is not a scalable solution for large arrays and is only implemented here for simplicity. The modulation characteristics obtained for one of the devices is shown in **Fig.3-10**B, where "crossed-swords" behavior is observed with a well-defined symmetry point.

During forward and backward pass cycles, input values (from the training set) were represented with different voltage levels and output results were obtained via measuring the line currents. Note that in an actual implementation representing input values with different pulse widths rather than amplitudes might be beneficial, avoiding nonlinear conductance of the crosspoint elements for accurate vector-matrix



Figure 3-10: Experimental setup using metal-oxide based programmable resistors [110]: (A) Optical micrograph of metal-oxide based electrochemical devices, also referred to as MO-ECRAM [52]. Note that the image shows an integrated array whereas experiments were conducted with individual devices connected externally. (B) Conductance modulation characteristics obtained for one of the devices, showing "crossed-swords" behavior with a well-defined symmetry point. (C) Schematic for array configuration used in 2-parameter optimization with SHD algorithm. All steps are shown using the same notation used in Fig.3-6 except for the backward pass (Step II) which is not required for a single layer network. For training, sum of squared errors is used to calculate the scalar error and vector δ , C_{main} is updated once every 10 samples (i.e. $\tau = 10$) whereas [1,0] and [0,1] were used in Step IV (as *u* vectors). For simplicity, the reference arrays containing symmetry point information are stored in digital (as they remain unchanged throughout the training).

multiplication. Following the generation of the update vectors, x and δ , the array is programmed in parallel using stochastic updating with half-bias voltage scheme as explained in Ref. [13].



Figure 3-11: Experimental demonstration of the SHD algorithm on twoparameter regression [110]: Evolution of device conductances for the first (A_1, C_1) and the second (A_2, C_2) parameters. Plotting the values of A versus C produces the distinctive spiraling image, as expected from the theoretical analysis.

The array training results using the modified training algorithm are shown in **Fig.3-11**. It can be seen that for both parameters, the steady-state solutions for $A_{1,2}$ match the symmetry points, while those of $C_{1,2}$ successfully converge to the optimal values. Moreover, the distinctive spiraling behavior indicating the dynamics of dissipative mechanical systems was observed for both variables.

3.3 Simulated Neural Network Training Results

The description of asymmetry as the mechanism of dissipation indicates that it is a necessary and useful device property for convergence within the SHD framework. However, this argument does not imply that the convergence speed would be determined by the magnitude of device asymmetry for practical-sized applications. Unlike the single-parameter regression problem considered above, the exploration space for DNN training is immensely large, causing optimization to take place over many iterations of the dataset. In return, the level of asymmetry required to balance (i.e. damp) the system evolution is very small and can be readily achieved by any practical level of asymmetry.

To prove these assertations, simulated results are shown in **Fig.3-12** for a Long Short-Term Memory (LSTM) network, using device models with increasing levels of asymmetry, trained with both the SGD and SHD algorithms. The network was trained on Leo Tolstoy's War and Peace novel, to predict the next character for a given text string. This dataset consists of 3, 258, 246 characters, which is then split into training and test sets as 2, 933, 246 and 325, 000 characters, respectively. The network is trained to have a vocabulary of 87 distinct characters. A hidden vectors of 64-cell size was chosen, which corresponds to approximately 77,000 weights for the complete network. Full details of the network architecture can be found in Ref. [116]. For reference, training the same network with a 32-bit digital floating-point architecture yields a cross-entropy level of 1.33. This network was deliberately chosen as LSTM's are known for being particularly vulnerable to device asymmetry [117].

The simulation framework used here is the same that was used in Ref. [10, 13, 112, 117]. The simulations start with instantiating 3 devices per weight. Each device parameter (e.g. number of states, asymmetry factor, symmetry point) is generated with a given mean and standard variation, such that no two devices are the same. Moreover, these device parameters also bear cycle-to-cycle variation, defined by another parameter, to make the operation more realistic.

The asymmetry model is defined the same way shown in Eqn.3.5, yielding a behav-

ior similar to those shown in **Fig.3-3**A. The perfectly symmetric device model uses $\kappa = 0$, while other traces use values of 0.1, 0.3, and 0.5 in increasing asymmetry order. A 10% device-to-device variation was used across the crosspoint elements, which is implemented in a multiplicative fashion to the average respective κ . In other words, more symmetric device models are picked from a narrower bundle, while the highly asymmetric device models also have higher-variation. Throughout the simulation, a 30% cycle-to-cycle variation is employed to account for uncontrollable device behavior, which is commonly seen in the field. Devices belonging to A_{main} and A_{ref} are initialized at $A_{main,symmetry}$, while C_{main} was initialized with a random distribution that is determined by the layer size.

The incremental changes are set such that devices have on average 1200 programmable states within their dynamic range. Through setting the gain factors at the integrator terminals appropriately, the average full conductance range of devices are adjusted to be equivalent ± 2 arbitrary units. Consistent with this notation, the integrators are set to saturate at ± 40 arbitrary units. ADCs were defined to use 9-bit resolution whereas 7-bit resolution was selected for the DACs where the outputreferred noise level was set at 0.02 arbitrary units. This selection was made in order not to be limited by noise-related performance degradation, as studied by Ref. [10]. Noise, Bound, and Update management techniques were employed, which can be found in detail in disclosures [118] and [119] respectively.

In the update cycle, the maximum allowed number of pulses (i.e. bit length, BL) was set to be 100. However, as update management determines this number on-thego depending on certain characteristics of the update vectors and device parameters, real BL was less than 10 for the most of the training. Although it is not the version used in this thesis, IBM has recently made a similar simulation platform open-access to the public (https://github.com/ibm/aihwkit), which can be used to reproduce the same results.

The insets in **Fig.3-12**A show the average conductance modulation characteristics representative for each asymmetry level. The learning curves show the evolution of the cross-entropy error, which measures the performance of a classification model,



Figure 3-12: Simulated training results for different resistive device technologies [110]: (A) Simulated learning curves of a Long Short-Term Memory (LSTM) network trained on Leo Tolstoy's War and Peace novel, using different crosspoint device models under the SGD algorithm. Details of the network can be found in Ref. [116] (B) Simulated learning curves for the same network using the SHD algorithm.

with respect to the epochs of training. First, **Fig.3-12**A shows that even for minimally asymmetric devices (blue trace) trained with SGD, the penalty in classification performance is already severe. This result also demonstrates once more the difficulty of engineering a device that is symmetric-enough to be trained accurately with SGD. On the other hand, for SHD (**Fig.3-12**B), all depicted devices are trained successfully. Note that this statement does not apply to abrupt-switching devices, such as PCMs, as they do not bear a well-defined symmetry point. As a result, neither SGD not SHD can achieve successful training results with such devices when all operations are kept fully-parallel.

The most important implication of the data presented in **Fig.3-12**B is that when one combines the highly symmetric devices shown in this **Chap.2** (blue curve), and the SHD training algorithm shown here, the training results can be faster and more accurate, compared to those of nonexistent "ideal" devices trained with SGD. Even



Figure 3-13: Simulated training results for PCM-like devices. [110]: (A) Simulated learning curves of the LSTM network shown in Fig.3-12 using PCM device models under the SGD algorithm. Conductance modulation characteristics for a typical PCM device is given in the inset. SGD baseline corresponds to training with perfectly symmetric device. (B) Simulated learning curves of the same network using PCM device models under the SHD algorithm. SHD baseline corresponds to training with device model labeled as "low" asymmetry.

though analog computing has been long thought as a faster and more energy efficient alternative to their conventional digital counterparts, having higher accuracy or generalization performance has never been even remotely considered (due to many device and architecture imperfections). However, the work in this thesis proves that understanding the analog computing concept at multiple levels can not only resolve longstanding problems but also unlock such potential.

3.4 Conclusion

In this chapter, it was first clarified that the stochastic gradient descent algorithm is intrinsically unsuitable for analog crossbar training with non-ideal devices, due to competing dynamics imposed by the optimization problem and the physical behavior of crosspoint elements. This incompatibility explains the heavy degradation of classification accuracy for analog deep learning implementations with SGD as well as the difficulties observed so far to find a solution that does not sacrifice the parallelism of the framework. Indeed, the only fully parallel method proposed so far uses a different training algorithm altogether instead of enduring the complications of conventional SGD.

It was revealed here that this modified algorithm creates a coupled system displaying dynamics that resembles the motion of a classical system. In this method, device asymmetry plays a critical role by providing the mechanism of dissipation necessary for the convergence of the system. Therefore, as opposed to SGD, asymmetric devices with well-defined and stable symmetry points are required for the successful operation of the modified algorithm. The experimental demonstration of this method using metal-oxide based electrochemical devices validated these theoretical understandings as well as testing the algorithm in combination with real-implementation effects related to stochastic updating with half-bias programming scheme, device-todevice cross-talk, conductance retention, and modulation endurance. This algorithm is found to be suitable for many resistive technologies, as long as devices show smooth and bidirectional conductance modulation characteristics.
Chapter 4

Conclusion and Future Directions

Analog deep-learning architectures can provide orders of magnitude higher processing speed and energy-efficiency compared to traditional digital processors. This is imperative for the promise of artificial intelligence to be realized. However, the implementation of analog accelerators faces a significant barrier comprising two coupled components: 1) the absence of devices that satisfy stringent algorithm-imposed demands and 2) algorithms that can tolerate inevitable device nonidealities. This thesis showed advancement along both directions introducing a novel near-ideal device technology and a superior neural network training algorithm. The devices first realized here are CMOS-compatible nanoscale protonic programmable resistors that incorporate the benefits of nanoionics with extreme acceleration of ion transport under strong electric fields. Enabled by a material-level breakthrough of utilizing phosphosilicate glass (PSG) as a proton electrolyte, these devices achieved controlled proton intercalation in nanoseconds with high energy-efficiency. Separately, the theoretical underpinnings behind why device asymmetry is fundamentally incompatible with conventional neural network training algorithms were explained. By establishing a powerful analogy with classical mechanics, a novel method, Stochastic Hamiltonian Descent, is developed to exploit device asymmetry as a useful feature instead. In combination, the two developments presented in this thesis can be effective in ultimately realizing the potential of analog deep learning.

These advancements beg the question, what is next? Undoubtedly, nanosecond

and sub-picojoule operation is a must for any technology aiming to realize analog training accelerators. For protonic devices, this was only made possible by applying a high-enough electric field that effectively removes the energy barriers for bulk proton transport as well as charge-transfer reactions at the interface. However, for PSG, achieving this property required a high gate voltage. Future studies surely need to reduce the operation voltage under 1 V, without compromising speed or energy efficiency. Obviously, reducing the electrolyte thickness allows achieving the same field with less voltage, which was indeed utilized here. However, even an aggressive thinning down to 1 nm may not be enough due to the insertion limited dynamics, while probably losing electronic insulation capabilities in doing so. Therefore, the future work will need to optimize electrolytes with lower initial barrier heights and engineer interfaces for facile electrochemical insertion. The author suggests mechanical or chemical treatments of the electrolyte and its interfaces can achieve these goals, such as acid-bathing or He-ion irradiation.

The author strongly believes that the implications of the devices studied in **Chap.** 2 go far beyond analog deep learning, in particular to opening up new possibilities in solid-state nanoionics. Protons, among all ions, is the closest to electrons in achieving ballistic transport. Indeed in liquid water, proton motion is known to have significant quantum character, with activation-less quantum nuclear dynamics in some exchange events between water molecules [120]. Thus under a high enough electric field that tilts the energy landscape to the point at which the migration energy barrier is reduced to below a few k_BT , classical or quantum ballistic transport of protons in solids may be realized. As the proton mass is only $1800 \times$ the free-electron mass, comparable to the effective electronic band mass in some heavy-Fermion solids, it is highly intriguing to consider the potential for exotic nuclear transport under strong fields. In addition to scientific interest, these results also open up possibilities in other applications wherever fast ion motion is required, such as microbatteries, artificial photosynthesis, and light-matter interactions.

As regarding the algorithms, it must not be forgotten that the Stochastic Hamiltonian Descent algorithm only resolves the degradation issues related to asymmetric device modulation. However, there are a multitude of other analog element nonidealities such as noise, high-variability, conductance drift, limited number of states. Algorithms can be further advanced to ameliorate if not mitigate the effects of such imperfections. An interesting way forward could see implementation of SHD for digital neural network training with simulated asymmetry. The filtering dynamics described above allows SHD to guide its core component selectively in directions with high statistical persistence. Therefore, at the expense of increasing the overall memory and number of operations, SHD might outperform conventional training algorithms by providing faster convergence, better classification accuracy, and/or superior generalization performance.

The main conclusion of this thesis is analog deep learning is not simply using analog devices for deep learning algorithms. Therefore, it can't simply adopt and merge existing solutions from established fields and anything less than a complete application-specific redesign is bound to fail. For example, most mature devices are descendants of memory technologies, which never considered incremental symmetric modulation between a high number of states. Similarly, the 175 year old gradient descent algorithm obviously was not designed for anything similar to asymmetric updates. This argument goes far deeper. The highly unwanted volatility for memory applications, can be beneficial as weight decay [121], nonvolatile voltage dependence can be used to implement half-select [13], and compounded-response for successive pulses with same polarity can act as a momentum term [122]. It was demonstrated here that even the role of asymmetry can be reversed from a major technical barrier to a key feature with original, application-specific thinking. As interesting as levelspecific studies are, the author finds it implausible to realize analog deep learning with segregated efforts and urges vertical collaboration focusing on large scale applications. Even though the early attempts might likely fail, the learnings would be invaluable in setting accurate benchmarks, standardizing methods, unifying terminology, and aligning the goals of multiple disciplines. This work, can hopefully be the first of many software-hardware hybrid studies to shape the future of analog deep learning.

Chapter 5

Appendix

5.1 Process Engineering and Additional Device Results

5.1.1 Undercut Profile of WO₃

A key feature of the microscale protonic programmable resistors discussed in Sec.2.4 is the WO₃ undercut, as shown in Fig.2-5B. This step utilizes wet etching of WO₃ in CD-26 (dilute TMAH). However, as can be seen from Fig.5-1, only amorphous WO₃ gets attacked by this process, resulting with failure of polycrystalline-channel devices fail under the process described in Sec.2.4.

5.1.2 Optimization of Palladium Reservoir Thickness

In early fabrication attempts, thicker Pd reservoir layers were used in order to reduce the parasitic gate resistance. However, as can be seen in **Fig.5-2**, upon insertion into forming gas, Pd uptakes considerable amount of hydrogen and expands in volume. For layers with high volume/surface ratio, this results in a loss of adhesion and therefore is highly undesirable. Following the observation of these effects, the Pd thickness was kept under 10 nm, for which these effects have found to be completely absent.



Figure 5-1: Undercut profile engineering for WO_3 : (A) 45° top view SEM image of a microscale device with amorphous WO_3 channel. The concentric horizontal ellipses in the middle correspond to the outlines of PSG and WO_3 where the gap between the two indicates successful undercut as explained in **Fig.2-5**. (B) The same feature is absent for polycrystalline WO_3 as crystallization of the material significantly increases its resistance to TMAH-based wet etching step.



Figure 5-2: Exfoliation of thick Pd layer under forming gas: Photographs of different chips with Pd thickness >10 nm evidencing the material expands in volume when introduced into a forming gas environment and ultimately loses adhesion.

5.1.3 Fabrication Flow for Microscale Protonic Programmable Resistors

Atomic Layer Deposition (ALD) of 10/90 nm HfO₂/Al₂O₃ on 1×1 cm² SiO₂/Si pieces.

- Patterning a bilayer of polymethylglutarimide (PMGI) and Microposit S1813 using a Heidelberg maskless aligner (MLA) 150 for source/drain contact layer lift-off.
- Electron-beam evaporation of 15/5 nm of Au/Cr layer using AJA evaporation system, followed by lift-off step in NMP.
- Deposition of 10 nm amorphous WO₃ using ALD with Bis(tert-butylimino)bis(dimethylamino)tungsten (VI) (BTBMW) and O₃ precursors at 330 °C.
- Plasma-Enhanced Chemical Vapor Deposition (PECVD) of PSG layer using 1420 sccm N₂O, 12 sccm SiH₄, and 12 sccm PH₃ (2% in H₂) at 100 °C, with a RF plasma power of 60 W at 300 kHz.
- Patterning a Microposit S1813 using a Heidelberg maskless aligner (MLA) 150 as a soft-mask for the active layer.
- Patterning of both PSG and WO₃ layers using Reactive Ion Etching (RIE) with a CF_4 plasma at 100 W for 3×60 s.
- Selective wet-etching (undercut) of the WO₃ layer in MF CD-26 (diluted TMAH) at room temperature.
- Patterning a bilayer of polymethylglutarimide (PMGI) and Microposit S1813 using a Heidelberg maskless aligner (MLA) 150 for gate contact layer lift-off.
- Electron-beam evaporation of 5 nm of Pd layer using AJA evaporation system, followed by lift-off step in NMP.
- Patterning a bilayer of polymethylglutarimide (PMGI) and Microposit S1813 using a Heidelberg maskless aligner (MLA) 150 for pad lift-off.
- Electron-beam evaporation of 150/10 nm of Au/Cr layer using AJA evaporation system, followed by lift-off step in NMP.

5.1.4 Fabrication Flow for Nanoscale Protonic Programmable Resistors

• Atomic Layer Deposition (ALD) of $10/40 \text{ nm HfO}_2/\text{Al}_2\text{O}_3$ on $1 \times 1 \text{ cm}^2 \text{ SiO}_2/\text{Si}$ pieces.

- Patterning of poly(methyl methacrylate) (PMMA, e-beam resist) with Elionix FLS-125 for channel layer lift-off.
- Reactive sputtering of WO₃ layer from metallic target at room temperature in O2/Ar RF plasma using AJA sputtering system (See Sec.2.7).
- Lifting off the WO₃ layer in n-methyl pyrrolidone (NMP-Microposit 1165) followed by annealing in 8 : 2 N₂:O₂ environment at 400 °C for 1 hour.
- Patterning of poly(methyl methacrylate) (PMMA, e-beam resist) with Elionix FLS-125 for source/drain contact layer lift-off.
- Electron-beam evaporation of 35/ 5 nm of Au/Cr layer using AJA evaporation system, followed by lift-off step in NMP.
- Plasma-Enhanced Chemical Vapor Deposition (PECVD) of PSG layer using 1420 sccm N₂O, 12 sccm SiH₄, and 12 sccm PH₃ (2% in H₂) at 100 °C, with a RF plasma power of 60 W at 300 kHz.
- Patterning of poly(methyl methacrylate) (PMMA, e-beam resist) with Elionix FLS-125 for gate contact layer lift-off.
- Electron-beam evaporation of 10 nm of Pd layer using AJA evaporation system, followed by lift-off step in NMP.
- Reactive Ion Etching (RIE) of the PSG layer using Pd layer as the hard mask, under CF₄ plasma at 100 W.
- Patterning the bilayer of poly (methylglutarimide) and Microposit S1813 positive tone photoresist, using Heidelberg-MLA 150 for pad layer lift-off.
- Electron-beam evaporation of 150/15 nm of Au/Cr layer using AJA evaporation system, followed by lift-off step in NMP.

5.1.5 Alternative Layouts for Protonic Devices

In addition to the device stacks described in Sec.2.4 and 2.8, coplanar structures were also examined (Fig.5-3). These structures also employed a symmetric gate stack where instead of a Pd reservoir, a second WO₃ layer was used. In addition, the coplanar layout shown here could also enable realizing nanoscale devices with Nafion, which could be spun over the entire chip as the final step, and act as an electrolyte across the gap defined between two WO_3 regions.



Figure 5-3: Coplanar protonic devices with symmetric gate stack: (A) An SEM image of a device design that employs two WO₃ layers, one connected to the gate terminals as the reservoir, and the other connected to the source-drain terminals as the channel. Note that the long WO₃ distance between the core of the device and the metal contacts resulted in high access resistance (that cannot be modulated with protonation). (B) An SEM image with a modified design using Au contacts to reduce the aforementioned issue.

A symmetric stack as such may be beneficial by means of modulation symmetry. Protonation of the WO₃ layers were unsuccessfully attempted by heated H_2SO_4 treatment, as in the absence of Pd, proton uptake of WO₃ from forming gas is significantly lowered. Note that ignoring alignment concerns, one can also realize coplanar devices with asymmetric stack, utilizing Pd reservoir similar to the approach used in **Chap.2**.

5.1.6 Alternative Active Channel Materials

During the time of this thesis three main deposition methods were used: electronbeam evaporation, reactive sputtering, and atomic layer deposition. In addition to these three, a pulsed-laser deposition attempt was also made unsuccessfully. Following the deposition, 4 major annealing options were used: no anneal, or 1 h anneal in 8:2N₂:)₂ conditions at 300 °C, 400 °C, and 450 °C.

Electron-beam evaporated WO_3 was found to be perfectly stoichiometric (as was the target). However, during the photolithography-based fabrication, the poor step coverage of the deposition method resulted in contact issues. On the other hand, the ALD-deposited materials were found to be highly substoichiometric with low reproducibility. Although using O_3 instead of H_2O improved the oxygen content, neither deposition method was found to be suitable for this application, due to high base conductance and low modulation depth. Reactive sputtering based films have the advantage of tunability, particularly by controlling the plasma power. Higher powers reduce the time of flight of the W metal atom (from the target to the sample) and thus yield a more oxygen deficient film. However, the reproducibility rate of the material coming out of sputtering chamber was also found to be low.

In order to obtain the same result every time, the annealing step was used as an equimorphising process. Given the oxygen ambient chosen for the annealing conditions, materials were fully oxidized in the chamber, irrespective of their initial deficiencies. Interestingly, it was observed that for initially (i.e. preanneal) oxygen rich films, the crystallization occurred only above 450 °C, which is above the critical BEOL-limit. On the other hand, substoichiometric films were successfully crystallized at 400 °C, as well as being oxidized to the fullest during the annealing.

The materials that are oxygen rich and also amorphous were highly resistive and not responsive to modulation pulses. For example, when a 70 W sputtering process was used in combination with 400 °C annealing, since the initial film was oxygen-rich, it did not crystallize at this temperature, and the resultant devices were simply too resistive to measure. On the other hand, we were unable to generate an amorphous material with a right stoichiometry that is not too conductive, which might be a future direction to follow if there are additional benefits of amorphous materials compared to polycrystalline ones.

In addition to WO₃ and V_2O_5 based devices reported in **Chap.2**, trials were conducted using MoO₃, Nb₂O₅, and Ta₂O₅. These materials have been reported to have desirable properties for ion-intercalation, however, a working flavor could not be produced during the time of this thesis work, which might also be an interesting direction for future studies.

5.1.7 Field-Effect Related Volatile Conductance Changes

Careful analysis of the modulation dynamics (Fig.2-7C) unravels interesting device physics, comprising of both volatile and non-volatile changes in conductance, as programming pulses are applied. Under constant $V_{DS} = 0.1 \text{ V}$, the source current, I_S , immediately steps up/down when a gate pulse $(\pm 3 \text{ V})$ is applied and then smoothly increases/decreases for the remainder of the pulse duration (colored dots). Once the pulse disappears (i.e. floating gate), another sudden drop/rise in I_S is observed. In the absence of gate pulse, I_s remains approximately constant, at a level different from that before the application of the pulse, indicating nonvolatile programming of the channel conductance (dark dots). These sudden changes in I_S do not reflect an increase in gate current, I_{G} (<pA) but are due to a field-effect enhancement of the channel conductance (ΔG_{FE}). This behavior can be explained by a field-effect increase in the electron concentration and a resulting increase in conductance of the n-type WO_3 channel by the electrostatic field of protons driven within the electrolyte close to the WO₃ interface. Unlike the non-volatile, electrochemical intercalation-induced conductance modulation ($\Delta G_{intercalation}$), this additional channel current only flows during the application of a gate voltage pulse and so it is volatile.

5.1.8 Linear Regression Experiment

The protonic programmable resistors demonstrated in **Chap.2** were tested in a single parameter linear regression experiments, similar to those simulated in **Fig.3-1**. For this test a dummy dataset was created with a mean corresponding to $0.4\,\mu\text{S}$ with a standard deviation $0.02\,\mu\text{S}$. In this training-example, a conductance value was selected at random from this dataset, and was compared to the device conductance at that time. A single 5 ns programming pulse was then generated, with its polarity determined by the relative positions of the channel conductance and the selected datapoint. **Fig.5-5** shows that experiments starting from either side of the optimum value successfully converged to the optimum (i.e. mean of the dataset). Note that once the device conductance enters the vicinity of the optimum (green shaded area),



Figure 5-4: Field-effect related volatile conductance change [79]: (A) Detailed picture of Fig.2-7A where the channel current, I_S , is constantly recorded in the presence (colored dots), and absence (dark dots) of gate pulses.(B)Intercalation and (C) field-effect conductance modulation as a function of the pulse gate voltage for different read drain-source voltage.

since some values are above the mean and others are below, the signal becomes a mixture of increments-decrements. The device ability to keep a dynamic average value equal to the target value $(0.4\,\mu\text{S})$ indicates that the device programming was accurate.



Figure 5-5: Single-parameter linear regression experiment conducted on a nanoscale protonic programmable resistor [86]. The protonic device with W = 150 nm, L = 75 nm, and $d_{PSG} = 10 \text{ nm}$ was programmed with voltage pulses of $\pm 10 \text{ V}$ and 5 ns. The dataset was artificially generated to be around $0.4 \mu\text{S}$ with a standard deviation of $0.02 \mu\text{S}$. For both initial points above and below this optimum value, device conductance converged to the target region and remained in the vicinity of this optimum, evidencing successful optimization.

5.1.9 Comparison of Retention Characteristics Under GND and FLT Gate Conditions

In all characterizations provided in **Chap.2**, gate terminal was left floating during readout, as was explained in **Fig.2-2**. In a real-scale implementation, this can be achieved by a pass transistor can be added in series to the gate of each device, controlled by a universal "Update-Enable" signal, blocking the electron current from the channel into the reservoir. However, given that the channel resistance values are in $M\Omega$ range, this might be high enough to grant sufficient leakage protection over relevant timescales for analog deep learning accelerators. To test this hypothesis, we compared the retention behavior of the devices under grounded and floating gate configurations. **Fig.5-6** shows no noticeable difference between the two different gate biasing conditions, suggesting that a pass transistor may indeed not be necessary.



Figure 5-6: Retention characteristics for $\approx 100 \text{ s}$ for floated and grounded gate conditions [86]. (W = 200 nm, L = 100 nm, $d_{PSG} = 10 \text{ nm}$). A train of 100 10 V pulses are used to program the device to different states.No noticeable difference was observed for two different gate biasing conditions.

5.2 Analog Computing Beyond Deep Learning

As mentioned in the beginning of this chapter, deep learning is not the only application which can be implemented on analog architectures. Processes that involve high volumes of matrix-matrix multiplication or rank-1 outer product can be significantly accelerated with such processors, provided that the algorithm also doesn't require precise computation of any particular step.

(1) Power Iteration and Iterative Numerical Solvers

A most trivial use case for analog architectures is as multiply-accumulate (MAC) machines, performing rapid, fully-parallel inner products. This mode of operation can be utilized for linear algebra algorithms based on power iteration, which require computation of the inner product of a given input, x, with the matrix, A, and its transpose, A^{T} , to produce power series such as Ax, AxA^{T} , $A^{2}x$, etc. These operations also benefit from storing A^{T} for free (once the analog array is configured to represent A), thanks to the bidirectional use of the crossbar array. Using this method, Neumann series approximate inversion, quasi-Newton methods, Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm, top-k eigenvector decomposition, top-k singular value decomposition, and Krylov subspace methods (e.g. generalized minimal residual method, GMRES) can be achieved.

However, it must be noted that the matrix-matrix multiplication carried out on an analog architecture suffers from nonidealities such as noise and device nonlinearity. In order to mitigate these issues, one can first perform many fast and inaccurate iterations on an analog processor, followed by few cycles of slower but accurate digital post-processing (i.e. the analog processor acts as a preconditioner).

(2) Solving Linear Algebra Problems with Gradient Descent

Alternative to the iterative methods described above, many problems can also be described as an optimization problem, similar to DNN training. For example, let's examine inversion of the matrix A. Initially a random square matrix, B, is instantiated on an analog crossbar array that is of same size as A. Then, the rows of Aare presented to B as input (for inner product), and the resultant output $y_i = A_i B$ is compared to the respective column of the identity matrix, I_i . To exemplify, for the first row, A_1 , the inner product is compared with I_1 which is [1, 0, 0, ..., 0]. The comparison, $y_i - I_i$ is used to compute the error (i.e. between B and A^{-1}) and applied to B following a gradient descent formalism. Therefore, following sufficient steps, the initially random matrix B converges to A^{-1} . The same idea can be extended to any convex problem, where a known relation (e.g. $A \cdot A^{-1} = I$) can be used as a labeled data for optimization, such as linear system solving and eigenvector decomposition.

Since this method employs a feedback loop, it is much more robust to noise unlike the power iterations described above. However, to mitigate nonlinearity-caused degradations, digital post-processing methods such as Newton-Hotteling or digital-SGD can be performed to improve the accuracy of analog results in few cycles [123].

(3) Implementing Randomized Algorithms

Arguably the most interesting and innovative method of the trio described here is to utilize analog architectures is solving randomized algorithms [124]. These algorithms are particularly useful in big-data scenarios to obtain an approximate solution where matrix size too large to solve [125] or the complete matrix is not available at a given time (e.g. real-time sensor readout) [126]. A well-known example of such algorithms is randomized matrix sketching, that projects the original data onto a lower dimensional subspace (i.e. the sketch), to obtain an approximate solution (e.g. to find the solution of an overdetermined system of linear equations, See Ref. [127]).

Unlike the methods described above, the reason why randomized algorithms are suitable for acceleration with analog architectures is not due to MAC operations but parallel rank-1 outer products instead. For example, let's assume the case for streaming a data,= incoming in real time from 10^4 sensors over 10^6 time instances, to a reduced dimension of $10^4 \times 10$. Then, a sketch matrix, S, of size $10^4 \times 10$ is randomly instantiated on an analog crossbar array. For each input, x, of size $10^4 \times 1$, the matrix S is rank-1 updated with x and a random vector y of size 1×10 . At the end of 10^6 instance of data, the matrix S represents a reduced dimension sketch of the input system (to be used in further analysis, such as singular value decomposition, instead of doing the same on the original $10^4 \times 10^6$ one). A detailed explanation of this method, and how to mitigate asymmetry related degradations in this application can be found in Ref. [128].

Bibliography

- [1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436-444, 2015.
- [2] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for modern deep learning research. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13693–13696, 2020.
- [3] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. arXiv preprint arXiv:2101.03961, 2021.
- [4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.
- [5] Yu-Hsin Chen, Tushar Krishna, Joel S Emer, and Vivienne Sze. Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE journal of solid-state circuits*, 52(1):127–138, 2016.
- [6] Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In Proceedings of the 44th annual international symposium on computer architecture, pages 1-12, 2017.
- [7] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, pages 1–16. IEEE, 2020.
- [8] Jungwook Choi, Swagath Venkataramani, Vijayalakshmi Viji Srinivasan, Kailash Gopalakrishnan, Zhuo Wang, and Pierce Chuang. Accurate and efficient 2-bit quantized neural networks. *Proceedings of Machine Learning and* Systems, 1:348-359, 2019.
- [9] Xiao Sun, Jungwook Choi, Chia-Yu Chen, Naigang Wang, Swagath Venkataramani, Vijayalakshmi Viji Srinivasan, Xiaodong Cui, Wei Zhang, and Kailash

Gopalakrishnan. Hybrid 8-bit floating point (hfp8) training and inference for deep neural networks. Advances in neural information processing systems, 32, 2019.

- [10] Tayfun Gokmen, Murat Onen, and Wilfried Haensch. Training deep convolutional neural networks with resistive cross-point devices. *Frontiers in neuro*science, 11:538, 2017.
- [11] Karl Steinbuch. Die lernmatrix. *Kybernetik*, 1(1):36–45, 1961.
- [12] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533-536, 1986.
- [13] Tayfun Gokmen and Yurii Vlasov. Acceleration of deep neural network training with resistive cross-point devices: Design considerations. *Frontiers in neuro-science*, 10:333, 2016.
- [14] Geoffrey W Burr, Robert M Shelby, Severin Sidler, Carmelo Di Nolfo, Junwoo Jang, Irem Boybat, Rohit S Shenoy, Pritish Narayanan, Kumar Virwani, Emanuele U Giacometti, et al. Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element. *IEEE Transactions on Electron Devices*, 62(11):3498-3507, 2015.
- [15] Sapan Agarwal, Steven J Plimpton, David R Hughart, Alexander H Hsia, Isaac Richter, Jonathan A Cox, Conrad D James, and Matthew J Marinella. Resistive memory device requirements for a neural algorithm accelerator. In 2016 International Joint Conference on Neural Networks (IJCNN), pages 929–938. IEEE, 2016.
- [16] Stefano Ambrogio, Pritish Narayanan, Hsinyu Tsai, Robert M Shelby, Irem Boybat, Carmelo Di Nolfo, Severin Sidler, Massimo Giordano, Martina Bodini, Nathan CP Farinha, et al. Equivalent-accuracy accelerated neural-network training using analogue memory. *Nature*, 558(7708):60-67, 2018.
- [17] Sapan Agarwal, Robin B Jacobs Gedrim, Alexander H Hsia, David R Hughart, Elliot J Fuller, A Alec Talin, Conrad D James, Steven J Plimpton, and Matthew J Marinella. Achieving ideal accuracies in analog neuromorphic computing using periodic carry. In 2017 Symposium on VLSI Technology, pages T174-T175. IEEE, 2017.
- [18] Geoffrey W Burr, Robert M Shelby, Abu Sebastian, Sangbum Kim, Seyoung Kim, Severin Sidler, Kumar Virwani, Masatoshi Ishii, Pritish Narayanan, Alessandro Fumarola, et al. Neuromorphic computing using non-volatile memory. Advances in Physics: X, 2(1):89–124, 2017.

- [19] Fuxi Cai, Justin M Correll, Seung Hwan Lee, Yong Lim, Vishishtha Bothra, Zhengya Zhang, Michael P Flynn, and Wei D Lu. A fully integrated reprogrammable memristor-cmos system for efficient multiply-accumulate operations. *Nature Electronics*, 2(7):290–299, 2019.
- [20] Can Li, Daniel Belkin, Yunning Li, Peng Yan, Miao Hu, Ning Ge, Hao Jiang, Eric Montgomery, Peng Lin, Zhongrui Wang, et al. Efficient and self-adaptive in-situ learning in multilayer memristor neural networks. *Nature communications*, 9(1):1–8, 2018.
- [21] Can Li, Zhongrui Wang, Mingyi Rao, Daniel Belkin, Wenhao Song, Hao Jiang, Peng Yan, Yunning Li, Peng Lin, Miao Hu, et al. Long short-term memory networks in memristor crossbar arrays. *Nature Machine Intelligence*, 1(1):49– 57, 2019.
- [22] Mirko Prezioso, Farnood Merrikh-Bayat, BD Hoskins, Gina C Adam, Konstantin K Likharev, and Dmitri B Strukov. Training and operation of an integrated neuromorphic network based on metal-oxide memristors. *Nature*, 521(7550):61-64, 2015.
- [23] Abu Sebastian, Manuel Le Gallo, Riduan Khaddam-Aljameh, and Evangelos Eleftheriou. Memory devices and applications for in-memory computing. *Nature* nanotechnology, 15(7):529–544, 2020.
- [24] Zhongrui Wang, Huaqiang Wu, Geoffrey W Burr, Cheol Seong Hwang, Kang L Wang, Qiangfei Xia, and J Joshua Yang. Resistive switching materials for information processing. *Nature Reviews Materials*, 5(3):173–195, 2020.
- [25] Vinod K Sangwan and Mark C Hersam. Neuromorphic nanoelectronic materials. Nature nanotechnology, 15(7):517–528, 2020.
- [26] Qiangfei Xia and J Joshua Yang. Memristive crossbar arrays for brain-inspired computing. *Nature materials*, 18(4):309–323, 2019.
- [27] Mohammed A Zidan, John Paul Strachan, and Wei D Lu. The future of electronics based on memristive systems. *Nature electronics*, 1(1):22–29, 2018.
- [28] Geoffrey W Burr, Matthew J Brightsky, Abu Sebastian, Huai-Yu Cheng, Jau-Yi Wu, Sangbum Kim, Norma E Sosa, Nikolaos Papandreou, Hsiang-Lan Lung, Haralampos Pozidis, et al. Recent progress in phase-change memory technology. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 6(2):146-162, 2016.
- [29] Abu Sebastian, Tomas Tuma, Nikolaos Papandreou, Manuel Le Gallo, Lukas Kull, Thomas Parnell, and Evangelos Eleftheriou. Temporal correlation detection using computational phase-change memory. *Nature Communications*, 8(1):1–10, 2017.

- [30] Irem Boybat, Manuel Le Gallo, SR Nandakumar, Timoleon Moraitis, Thomas Parnell, Tomas Tuma, Bipin Rajendran, Yusuf Leblebici, Abu Sebastian, and Evangelos Eleftheriou. Neuromorphic computing with multi-memristive synapses. *Nature communications*, 9(1):1–12, 2018.
- [31] Manuel Le Gallo, Daniel Krebs, Federico Zipoli, Martin Salinga, and Abu Sebastian. Collective structural relaxation in phase-change memory devices. Advanced Electronic Materials, 4(9):1700627, 2018.
- [32] Yuchao Yang, Xiaoxian Zhang, Liang Qin, Qibin Zeng, Xiaohui Qiu, and Ru Huang. Probing nanoscale oxygen ion motion in memristive systems. *Nature* communications, 8(1):1–10, 2017.
- [33] Yuchao Yang, Peng Gao, Linze Li, Xiaoqing Pan, Stefan Tappertzhofen, Shin-Hyun Choi, Rainer Waser, Ilia Valov, and Wei D Lu. Electrochemical dynamics of nanoscale metallic inclusions in dielectrics. *Nature communications*, 5(1):1–9, 2014.
- [34] John Paul Strachan, Matthew D Pickett, J Joshua Yang, Shaul Aloni, AL David Kilcoyne, Gilberto Medeiros-Ribeiro, and R Stanley Williams. Direct identification of the conducting channels in a functioning memristive device. Advanced materials, 22(32):3573-3577, 2010.
- [35] Hao Jiang, Lili Han, Peng Lin, Zhongrui Wang, Moon Hyung Jang, Qing Wu, Mark Barnell, J Joshua Yang, Huolin L Xin, and Qiangfei Xia. Sub-10 nm ta channel responsible for superior performance of a hfo2 memristor. *Scientific reports*, 6(1):1–8, 2016.
- [36] An Chen and Ming-Ren Lin. Variability of resistive switching memories and its impact on crossbar array performance. In 2011 International Reliability Physics Symposium, pages MY-7. IEEE, 2011.
- [37] Hanwool Yeon, Peng Lin, Chanyeol Choi, Scott H Tan, Yongmo Park, Doyoon Lee, Jaeyong Lee, Feng Xu, Bin Gao, Huaqiang Wu, et al. Alloying conducting channels for reliable neuromorphic computing. *Nature Nanotechnology*, 15(7):574–579, 2020.
- [38] J Joshua Yang, Feng Miao, Matthew D Pickett, Douglas AA Ohlberg, Duncan R Stewart, Chun Ning Lau, and R Stanley Williams. The mechanism of electroforming of metal oxide memristive switches. *Nanotechnology*, 20(21):215201, 2009.
- [39] Oleg Golonzka, J-G Alzate, U Arslan, M Bohr, P Bai, J Brockman, B Buford, C Connor, N Das, B Doyle, et al. Mram as embedded non-volatile memory solution for 22ffl finfet technology. In 2018 IEEE International Electron Devices Meeting (IEDM), pages 18-1. IEEE, 2018.

- [40] Luqiao Liu, Chi-Feng Pai, Y Li, HW Tseng, DC Ralph, and RA Buhrman. Spin-torque switching with the giant spin hall effect of tantalum. *Science*, 336(6081):555-558, 2012.
- [41] Wei-Gang Wang, Mingen Li, Stephen Hageman, and CL Chien. Electric-fieldassisted switching in magnetic tunnel junctions. *Nature materials*, 11(1):64–68, 2012.
- [42] Julie Grollier, Damien Querlioz, KY Camsari, Karin Everschor-Sitte, Shunsuke Fukami, and Mark D Stiles. Neuromorphic spintronics. *Nature electronics*, 3(7):360-370, 2020.
- [43] Murat Onen, Brenden A Butters, Emily Toomey, Tayfun Gokmen, and Karl K Berggren. Design and characterization of superconducting nanowire-based processors for acceleration of deep neural network training. *Nanotechnology*, 31(2):025204, 2019.
- [44] Karl K Berggren, Murat Onen, Brenden Butters, and Emily Toomey. Superconducting nanowire-based programmable processor, December 14 2021. US Patent 11,200,947.
- [45] André Chanthbouala, Vincent Garcia, Ryan O Cherifi, Karim Bouzehouane, Stéphane Fusil, Xavier Moya, Stéphane Xavier, Hiroyuki Yamada, Cyrile Deranlot, Neil D Mathur, et al. A ferroelectric memristor. *Nature materials*, 11(10):860-864, 2012.
- [46] Ji Ma, Jing Ma, Qinghua Zhang, Renci Peng, Jing Wang, Chen Liu, Meng Wang, Ning Li, Mingfeng Chen, Xiaoxing Cheng, et al. Controllable conductive readout in self-assembled, topologically confined ferroelectric domain walls. *Nature nanotechnology*, 13(10):947–952, 2018.
- [47] X Lyu, M Si, PR Shrestha, KP Cheung, and PD Ye. First direct measurement of sub-nanosecond polarization switching in ferroelectric hafnium zirconium oxide. In 2019 IEEE International Electron Devices Meeting (IEDM), pages 15–2. IEEE, 2019.
- [48] Jianshi Tang, Douglas Bishop, Seyoung Kim, Matt Copel, Tayfun Gokmen, Teodor Todorov, SangHoon Shin, Ko-Tao Lee, Paul Solomon, Kevin Chan, et al. Ecram as scalable synaptic cell for high-speed, low-power neuromorphic computing. In 2018 IEEE International Electron Devices Meeting (IEDM), pages 13-1. IEEE, 2018.
- [49] Elliot J Fuller, Farid El Gabaly, François Léonard, Sapan Agarwal, Steven J Plimpton, Robin B Jacobs-Gedrim, Conrad D James, Matthew J Marinella, and A Alec Talin. Li-ion synaptic transistor for low power analog computing. Advanced Materials, 29(4):1604310, 2017.

- [50] Jongwon Lee, Revannath Dnyandeo Nikam, Seokjae Lim, Myonghoon Kwak, and Hyunsang Hwang. Excellent synaptic behavior of lithium-based nano-ionic transistor based on optimal wo2. 7 stoichiometry with high ion diffusivity. Nanotechnology, 31(23):235203, 2020.
- [51] Revannath Dnyandeo Nikam, Myonghoon Kwak, Jongwon Lee, Krishn Gopal Rajput, Writam Banerjee, and Hyunsang Hwang. Near ideal synaptic functionalities in li ion synaptic transistor using li3poxsex electrolyte with high ionic conductivity. *Scientific reports*, 9(1):1–11, 2019.
- [52] Seyoung Kim, Teodor Todorov, Murat Onen, Tayfun Gokmen, Douglas Bishop, Paul Solomon, Ko-Tao Lee, Matt Copel, Damon B Farmer, John A Ott, et al. Metal-oxide based, cmos-compatible ecram for deep learning accelerator. In 2019 IEEE International Electron Devices Meeting (IEDM), pages 35–7. IEEE, 2019.
- [53] Yiyang Li, Elliot J Fuller, Joshua D Sugar, Sangmin Yoo, David S Ashby, Christopher H Bennett, Robert D Horton, Michael S Bartsch, Matthew J Marinella, Wei D Lu, et al. Filament-free bulk resistive memory enables deterministic analogue switching. Advanced Materials, 32(45):2003984, 2020.
- [54] Revannath Dnyandeo Nikam, Myonghoon Kwak, and Hyunsang Hwang. Allsolid-state oxygen ion electrochemical random-access memory for neuromorphic computing. *Advanced Electronic Materials*, 7(5):2100142, 2021.
- [55] He-Yi Huang, Chen Ge, Qing-Hua Zhang, Chang-Xiang Liu, Jian-Yu Du, Jian-Kun Li, Can Wang, Lin Gu, Guo-Zhen Yang, and Kui-Juan Jin. Electrolytegated synaptic transistor with oxygen ions. Advanced Functional Materials, 29(29):1902702, 2019.
- [56] Takayoshi Katase, Takaki Onozato, Misako Hirono, Taku Mizuno, and Hiromichi Ohta. A transparent electrochromic metal-insulator switching device with three-terminal transistor geometry. *Scientific reports*, 6(1):1–9, 2016.
- [57] Yoeri Van De Burgt, Ewout Lubberman, Elliot J Fuller, Scott T Keene, Grégorio C Faria, Sapan Agarwal, Matthew J Marinella, A Alec Talin, and Alberto Salleo. A non-volatile organic electrochemical device as a low-voltage artificial synapse for neuromorphic computing. *Nature materials*, 16(4):414–418, 2017.
- [58] Xiahui Yao, Konstantin Klyukin, Wenjie Lu, Murat Onen, Seungchan Ryu, Dongha Kim, Nicolas Emond, Iradwikanari Waluyo, Adrian Hunt, Jesús A Del Alamo, et al. Protonic solid-state electrochemical synapse for physical neural networks. *Nature communications*, 11(1):1–10, 2020.
- [59] Vincent Garcia and Manuel Bibes. Ferroelectric tunnel junctions for information storage and processing. *Nature communications*, 5(1):1–12, 2014.

- [60] Yong-Gun Lee, Satoshi Fujiki, Changhoon Jung, Naoki Suzuki, Nobuyoshi Yashiro, Ryo Omoda, Dong-Su Ko, Tomoyuki Shiratsuchi, Toshinori Sugimoto, Saebom Ryu, et al. High-energy long-cycling all-solid-state lithium metal batteries enabled by silver-carbon composite anodes. *Nature Energy*, 5(4):299–308, 2020.
- [61] Yoshitsugu Sone, Per Ekdunge, and Daniel Simonsson. Proton conductivity of nafion 117 as measured by a four-electrode ac impedance method. *Journal of* the Electrochemical Society, 143(4):1254, 1996.
- [62] Lunyang Liu, Wenduo Chen, and Yunqi Li. An overview of the proton conductivity of nafion membranes through a statistical analysis. *Journal of membrane* science, 504:1–9, 2016.
- [63] A Sassella, A Borghesi, S Rojas, and L Zanotti. Optical properties of cvddeposited dielectric films for microelectronic devices. Le Journal de Physique IV, 5(C5):C5-843, 1995.
- [64] WL Warren, DM Fleetwood, JR Schwank, MR Shaneyfelt, BL Draper, PS Winokur, MG Knoll, K Vanheusden, RAB Devine, LB Archer, et al. Protonic nonvolatile field effect transistor memories in si/sio/sub 2//si structures. *IEEE Transactions on Nuclear Science*, 44(6):1789–1798, 1997.
- [65] Elena Oborina, Scott Campbell, Andrew M Hoff, Richard Gilbert, and Eric Persson. Hydrogen-related mobile charge in the phosphosilicate glass-sio 2-si structure. Journal of applied physics, 92(11):6773-6777, 2002.
- [66] K Vanheusden, WL Warren, RAB Devine, DM Fleetwood, JR Schwank, MR Shaneyfelt, PS Winokur, and ZJ Lemnios. Non-volatile memory device based on mobile protons in sio2 thin films. *Nature*, 386(6625):587–589, 1997.
- [67] Christopher W Moore, Jun Li, and Paul A Kohl. Microfabricated fuel cells with thin-film silicon dioxide proton exchange membranes. *Journal of the Electrochemical Society*, 152(8):A1606, 2005.
- [68] T Uma and M Nogami. On the development of proton conducting p2o5-zro2-sio2 glasses for fuel cell electrolytes. *Materials chemistry and physics*, 98(2-3):382– 388, 2006.
- [69] JD Chapple-Sokol, WA Pliskin, RA Conti, E Tierney, and J Batey. Energy considerations in the deposition of high-quality plasma-enhanced cvd silicon dioxide. *Journal of The Electrochemical Society*, 138(12):3723, 1991.
- [70] MT Colomer and JR Jurado. Proton conductivity in nanopore silica xerogels. Ionics, 9(3):207-213, 2003.
- [71] Tristan Pichonat and Bernard Gauthier-Manuel. Development of porous siliconbased miniature fuel cells. Journal of Micromechanics and Microengineering, 15(9):S179, 2005.

- [72] Klaus-Dieter Kreuer. Proton conductivity: materials and applications. Chemistry of materials, 8(3):610-641, 1996.
- [73] Yoshihiro Abe, Hideo Hosono, Yoshio Ohta, and LL Hench. Protonic conduction in oxide glasses: simple relations between electrical conductivity, activation energy, and the oh bonding state. *Physical Review B*, 38(14):10166, 1988.
- [74] Masayuki Nogami, Ritsuko Nagao, Cong Wong, Toshihiro Kasuga, and Tomokatsu Hayakawa. High proton conductivity in porous p2o5-sio2 glasses. *The Journal of Physical Chemistry B*, 103(44):9468–9472, 1999.
- [75] San Ping Jiang. Functionalized mesoporous structured inorganic materials as high temperature proton exchange membranes for fuel cells. *Journal of Materials Chemistry A*, 2(21):7637-7655, 2014.
- [76] Yuqing Meng, Jun Gao, Zeyu Zhao, Jake Amoroso, Jianhua Tong, and Kyle S Brinkman. Review: recent progress in low-temperature proton-conducting ceramics. *Journal of Materials Science*, 54(13):9291–9312, 2019.
- [77] Shruti Prakash, William E Mustain, SeongHo Park, and Paul A Kohl. Phosphorus-doped glass proton exchange membranes for low temperature direct methanol fuel cells. *Journal of Power Sources*, 175(1):91–97, 2008.
- [78] Haibin Li, Dongliang Jin, Xiangyang Kong, Hengyong Tu, Qingchun Yu, and Fengjing Jiang. High proton-conducting monolithic phosphosilicate glass membranes. *Microporous and mesoporous materials*, 138(1-3):63-67, 2011.
- [79] Murat Onen, Nicolas Emond, Ju Li, Bilge Yildiz, and Jesús A Del Alamo. Cmoscompatible protonic programmable resistor based on phosphosilicate glass electrolyte for analog deep learning. Nano Letters, 21(14):6111-6116, 2021.
- [80] Yang Li and Yang-Tse Cheng. Hydrogen diffusion and solubility in palladium thin films. International journal of hydrogen energy, 21(4):281-291, 1996.
- [81] Alexander W Powell, Alexandros Stavrinadis, Sotirios Christodoulou, Romain Quidant, and Gerasimos Konstantatos. On-demand activation of photochromic nanoheaters for high color purity 3d printing. *Nano Letters*, 20(5):3485–3491, 2020.
- [82] Vadym V Kulish and Sergei Manzhos. Comparison of li, na, mg and alion insertion in vanadium pentoxides and vanadium dioxides. *Rsc Advances*, 7(30):18643–18649, 2017.
- [83] Chuan Sen Yang, Da Shan Shang, Nan Liu, Gang Shi, Xi Shen, Ri Cheng Yu, Yong Qing Li, and Young Sun. A synaptic transistor based on quasi-2d molybdenum oxide. *Advanced Materials*, 29(27):1700906, 2017.

- [84] Boris Orel, Marjeta Maček, Jože Grdadolnik, and Anton Meden. In situ uv-vis and ex situ ir spectroelectrochemical investigations of amorphous and crystalline electrochromic nb2o5 films in charged/discharged states. *Journal of Solid State Electrochemistry*, 2(4):221–236, 1998.
- [85] Gunnar A Niklasson and Claes G Granqvist. Electrochromics for smart windows: thin films of tungsten oxide and nickel oxide, and devices based on these. *Journal of Materials Chemistry*, 17(2):127–156, 2007.
- [86] Murat Onen, Nicolas Emond, Baoming Wang, Difei Zhang, Frances M. Ross, Ju Li, Bilge Yildiz, and Jesus A. del Alamo. Nanosecond protonic programmable resistors under extreme electric fields. *Under Review*, 2022.
- [87] Lars Onsager. The motion of ions: principles and concepts. Science, 166(3911):1359–1364, 1969.
- [88] B Roling, LN Patro, O Burghaus, and M Gräf. Nonlinear ion transport in liquid and solid electrolytes. The European Physical Journal Special Topics, 226(14):3095-3112, 2017.
- [89] Steffen Röthel, Rudolf Friedrich, Lars Lühning, and Andreas Heuer. Theoretical description of ion conduction in disordered systems: From linear to nonlinear response. Zeitschrift für Physikalische Chemie, 224(10-12):1855–1889, 2010.
- [90] Han Gao and Keryn Lian. A comparative study of nano-sio2 and nano-tio2 fillers on proton conductivity and dielectric response of a silicotungstic acid-h3po4poly (vinyl alcohol) polymer electrolyte. ACS Applied Materials & Interfaces, 6(1):464-472, 2014.
- [91] R Oesten and RA Huggins. Proton conduction in oxides: A review. Ionics, 1(5):427-437, 1995.
- [92] RAB Devine and GV Herrera. Electric-field-induced transport of protons in amorphous sio 2. *Physical Review B*, 63(23):233406, 2001.
- [93] Andreas Heuer, Sevi Murugavel, and Bernhard Roling. Nonlinear ionic conductivity of thin solid electrolyte samples: Comparison between theory and experiment. *Physical Review B*, 72(17):174304, 2005.
- [94] Marissa ES Beatty, Eleanor I Gillette, Alexis T Haley, and Daniel V Esposito. Controlling the relative fluxes of protons and oxygen to electrocatalytic buried interfaces with tunable silicon oxide overlayers. ACS Applied Energy Materials, 3(12):12338-12350, 2020.
- [95] Julien Godet and Alfredo Pasquarello. Proton diffusion mechanism in amorphous sio 2. Physical review letters, 97(15):155901, 2006.

- [96] Magnus Kunow and Andreas Heuer. Nonlinear ionic conductivity of lithium silicate glass studied via molecular dynamics simulations. *The Journal of chemical physics*, 124(21):214703, 2006.
- [97] William D Richards, Lincoln J Miara, Yan Wang, Jae Chul Kim, and Gerbrand Ceder. Interface stability in solid-state batteries. *Chemistry of Materials*, 28(1):266-273, 2016.
- [98] Daniel Serghi and Cristian Pavelescu. Dc dielectric breakdown in phosphosilicate glass films prepared by low temperature chemical vapour deposition. *Thin* solid films, 186(1):L25–L28, 1990.
- [99] EH Snow and BE Deal. Polarization phenomena and other properties of phosphosilicate glass films on silicon. Journal of the Electrochemical Society, 113(3):263, 1966.
- [100] Dhananjay Bhusari, Jun Li, Paul Joseph Jayachandran, Christopher Moore, and Paul A Kohl. Development of p-doped sio2 as proton exchange membrane for microfuel cells. *Electrochemical and Solid-State Letters*, 8(11):A588, 2005.
- [101] Masayuki Nogami, Yoshie Tarutani, Yusuke Daiko, Seiji Izuhara, Toshiaki Nakao, and Toshihiro Kasuga. Preparation of p 2 o 5 sio2 glasses with proton conductivity of 100 ms/cm at room temperature. *Journal of the Electrochemical Society*, 151(12):A2095, 2004.
- [102] Seyoung Kim, Murat Onen, and Tayfun Gokmen. Suppressing undesired programming at half-selected devices in a crosspoint array of 3-terminal resistive memory, September 28 2021. US Patent 11,133,063.
- [103] FA Lewis. The hydrides of palladium and palladium alloys. Platinum metals review, 4(4):132–137, 1960.
- [104] PG Dickens, RMP Quilliam, and MS Whittingham. The reflectance spectra of the tungsten bronzes. *Materials Research Bulletin*, 3(12):941–949, 1968.
- [105] Simon Burkhardt, Matthias T Elm, Bernhard Lani-Wayda, and Peter J Klar. In situ monitoring of lateral hydrogen diffusion in amorphous and polycrystalline wo3 thin films. Advanced Materials Interfaces, 5(6):1701587, 2018.
- [106] S Nakabayashi, R Shinozaki, Y Senda, and HY Yoshikawa. Hydrogen nanobubble at normal hydrogen electrode. Journal of Physics: Condensed Matter, 25(18):184008, 2013.
- [107] Limor Pasternak and Yaron Paz. Low-temperature direct bonding of silicon nitride to glass. RSC advances, 8(4):2161-2172, 2018.
- [108] Augustin Cauchy et al. Méthode générale pour la résolution des systemes d'équations simultanées. Comp. Rend. Sci. Paris, 25(1847):536-538, 1847.

- [109] Yu Feng and Yuhai Tu. How neural networks find generalizable solutions: Selftuned annealing in deep learning. arXiv preprint arXiv:2001.01678, 2020.
- [110] Murat Onen, Tayfun Gokmen, Teodor Todorov, Tomasz Nowicki, Jesus A. del Alamo, John Rozen, Wilfried Haensch, and Seyoung Kim. Neural network training with asymmetric crosspoint elements. *Under Review*, 2022.
- [111] Hyungjun Kim, Malte Rasch, Tayfun Gokmen, Takashi Ando, Hiroyuki Miyazoe, Jae-Joon Kim, John Rozen, and Seyoung Kim. Zero-shifting technique for deep neural network training on resistive cross-point arrays. arXiv preprint arXiv:1907.10228, 2019.
- [112] Tayfun Gokmen and Wilfried Haensch. Algorithm for training neural networks on resistive device arrays. *Frontiers in Neuroscience*, 14:103, 2020.
- [113] Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. Frontiers in computational neuroscience, 11:24, 2017.
- [114] Ruslan Salakhutdinov and Hugo Larochelle. Efficient learning of deep boltzmann machines. In Proceedings of the thirteenth international conference on artificial intelligence and statistics, pages 693–700. JMLR Workshop and Conference Proceedings, 2010.
- [115] Tayfun Gokmen, Seyoung Kim, and Murat Onen. Area and power efficient implementations of modified backpropagation algorithm for asymmetric rpu devices, September 9 2021. US Patent App. 16/808,811.
- [116] Andrej Karpathy, Justin Johnson, and Li Fei-Fei. Visualizing and understanding recurrent networks. arXiv preprint arXiv:1506.02078, 2015.
- [117] Tayfun Gokmen, Malte J Rasch, and Wilfried Haensch. Training lstm networks with resistive cross-point devices. *Frontiers in neuroscience*, page 745, 2018.
- [118] Tayfun Gokmen and Murat Onen. Noise and bound management for rpu array, July 23 2019. US Patent 10,360,283.
- [119] Tayfun Gokmen and Murat Onen. Update management for rpu array, July 13 2021. US Patent 11,062,208.
- [120] Rodolphe Vuilleumier and Daniel Borgis. Quantum dynamics of an excess proton in water using an extended empirical valence-bond hamiltonian. The Journal of Physical Chemistry B, 102(22):4261–4264, 1998.
- [121] Anders Krogh and John Hertz. A simple weight decay can improve generalization. Advances in neural information processing systems, 4, 1991.
- [122] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International* conference on machine learning, pages 1139–1147. PMLR, 2013.

- [123] Chai Wah Wu, Murat Onen, Tayfun Gokmen, Malte Johannes Rasch, Mark S Squillante, Tomasz J Nowicki, Wilfried Haensch, Lior Horesh, and Vasileios Kalantzis. Eigenvalue decomposition with stochastic optimization, December 30 2021. US Patent App. 16/910,975.
- [124] Petros Drineas and Michael W Mahoney. Randnla: randomized numerical linear algebra. *Communications of the ACM*, 59(6):80–90, 2016.
- [125] Jiyan Yang, Xiangrui Meng, and Michael W Mahoney. Implementing randomized matrix algorithms in parallel and distributed environments. *Proceedings of* the IEEE, 104(1):58–92, 2015.
- [126] Kenneth L Clarkson and David P Woodruff. Numerical linear algebra in the streaming model. In Proceedings of the forty-first annual ACM symposium on Theory of computing, pages 205–214, 2009.
- [127] David P Woodruff. Sketching as a tool for numerical linear algebra. arXiv preprint arXiv:1411.4357, 2014.
- [128] Lior Horesh, Murat Onen, Haim Avron, Tayfun Gokmen, Vasileios Kalantzis, and Shashanka Ubaru. Matrix sketching using analog crossbar architectures, November 18 2021. US Patent App. 16/874,819.