

Exploiting Learning Capabilities of Neuromorphic Nanowire Networks

Ruomin Zhu

朱若珉

School of Physics,
Faculty of Science,
The University of Sydney

A thesis submitted in fulfillment of
the requirements for the degree of
Doctor of Philosophy

Sydney, Australia

November 2023

Declaration

Declaration of original contribution

of the thesis submitted by

Ruomin Zhu

This is to certify that to the best of my knowledge, the content of this thesis is my own work. Outside the listed publications, this thesis or any of its contents have not been submitted for any degree or other purposes.

I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

Ruomin Zhu, Author

Date

Acknowledgements

Time really does fly! It feels like just yesterday when I embarked on this research journey, and now, here we are at its conclusion. What an amazing journey this has been! A massive thank you to everyone who's been with me on this journey through my research. Your support, in all its forms, has been my rock, and honestly, this thesis would not have seen the light of day without you all. From late-night coding sessions to those much-needed coffee runs, every little and big thing you have done has made a world of difference to me. So, this one's for all of you - my guides, friends, and family - who have been my cheerleaders from day one.

A huge thanks to Professor Zdenka Kuncic, who's been an outstanding supervisor throughout this journey. You have been super patient, always cheering me on, and never hesitated about giving me feedback when I needed it. You have generously given so much of her time, and honestly, every time I hit a wall with my research, you had the knack for giving just the right advice to get me moving again.

A massive thank you to my co-supervisor, Professor Joseph Lizier! Joe, your knowledge in information theory and complex systems has been nothing short of a lifeline for this project. I can't thank you enough for all those insightful chats that cleared up so many things for me and always seemed to point me in the right direction.

I would like to extend my thanks to the staff of school of physics and the university of Sydney for their support and assistance throughout the period of my studies.

My sincere thanks go out to the team at UCLA-CNSI, especially Sam Lilak, Adam Stieg and James Gimzewski, who helped me with physical devices and hardware experiments. Your insights and expertise have been invaluable to my work. These gratitude also go to Adrian Diaz-Alvarez and Tomonobu Nakayama from NIMS-MANA for your precious conversations and collaborations.

Big thanks are also in order for Jason Eshraghian. From those drunk nights in Telluride (even though you don't drink) to your expensive mattress in Santa Cruz, and from the dark memes on Instagram to the insightful voice messages on WhatsApp, you've been a constant. You are my role model as a young researcher. Thanks for being there every step of the way.

Valentina Baccetti and Francesco Caravelli, thank you for the invites to visit and give talks! I had not realised how awesome that part of academia was until then! Our conversations in Melbourne and Los Alamos were top-notch. You both played a big part in my decision to stick with academia.

Alon and Joel, thanks heaps for all the good times and the productivity boost when we teamed up. Won't forget our munch and sip sessions in Newtown and the Rose Hotel. And yep, beef curry with beef is gonna be my go-to in every Thai restaurant from now on!

Mac, Omid, Mike, Kevin, Yinhao, Steven, David, Aaron, Natesh and all other researchers I have worked with - even though we might not have spent a ton of time together, I still hold our work and collaborations in high regard. Every moment and every project has been valuable in this journey.

Simon, Josh, and Gianluca, we work on very similar topics. That friendly rivalry has been a subtle but motivating force throughout this journey. A big thank you for the exchange of thoughts and ideas. Gianluca, while we haven't met in person, I have great admiration for your work and am looking forward to the day we can have an in-depth discussion.

Pulin, Chia-Ling, Priyanka, Michael, Fiona, Alex, Tristram, Helen and everyone else I've had the privilege of teaching alongside, thank you for teaching me how to teach! I've never been so prepared to pass on what I know to the next generation of physics students.

Mikhail, Geraint, Daniel, Kevin, Jaime, Ivan, Firouzah, Michael, Michelle, Sheryl, Christina, Aria, Brendan, Annie, Natasha, and everyone else from Heron Island - massive thanks to all of you! Those epic nights at the bar, unforgettable snorkeling sessions, stargazing, and all the deep beachside chats were just incredible. Cheers for all the wonderful memories!

Peng, Hugo, Simon, Hanle, Steve, Cindy, Felix, Sadique, Gregor, Rej, Zhenming, Tobi, Jon, Terry, Guido, Tim, Shashank, Adil, Marco, Luna, Andrew, Doo Seok, Chang, Qinyu, Giorgia, Maxime, Melika, Sofie, - a heartfelt thank you to each one of you! The fantastic times, stellar company, and unconditional help in Telluride, Santa Cruz, and Santa Fe have been unforgettable. Truly a splendid experience on my journey.

Leslie, Yangyang, Bruce, Dage, Yinghao, Shou, Zihao, Jacob, Azar, Eddie, Hugh, MJC, Wuyang, Qingfeng, Cara, GFB, William, Jiahao, Haoze, Yi, Jason, Caroline, Leo, Nannan, Renee, Ashley, my basketball mates and all my dearest friends, thank you for your company and all the times we've shared. I couldn't have gotten through this without each and every one of you.

Dante, your support throughout this journey has been unwavering. Thank you for being my food adventure buddy, lending an ear to all my complaints, assisting me through house

moves, and standing by me during tough times. Your friendship is a priceless treasure in my life.

I would like to dedicate this last bit to my family and loved ones. You have been the silent heroes behind every page of this thesis. Through every late night, every meltdown, and every little victory, your love and encouragement have been my steady constant. This isn't just my win, it's ours. We made it happen!

Mom and Dad, your sacrifices and all those unspoken gestures haven't slipped by me. I might not voice it often, but I see it, I feel it, and I'm more grateful than I can put into words. Your belief in me, even amidst my own doubts, has been a silent, steadfast force propelling me forward. Your love and support have been my sanctuary, a place where I found the strength to keep going, even when the journey got tough. The values you instilled in me, the ethics and resilience, have been my guiding light, shaping my path and decisions throughout this journey. In every challenge faced and every hurdle overcome, I've seen reflections of your strength, your perseverance, and your grace. Thank you for being my endless source of love and support.

My dearest aunt, you've been my hero from the moment I entered this world. Whenever times get tough, you're the first person who comes to mind. Your unconditional love means the world to me, and deep down, I know I'm like a son to you too! Your impact on my life is immeasurable, and my gratitude for all you've done knows no bounds.

Big thanks to my dear uncle - always there for me, even with your crazy busy schedule, always picking me up from the airport. You have shown me what it means to be hardworking and responsible. Your mindset has been a crucial guide through my PhD journey.

To my girlfriend, Tiffany, a world of thanks for introducing me to this incredible continent; it's a decision I've never regretted. The dishes you've cooked, the surprising gifts you brought, the places we went together, and your unwavering support and uplifting words have been the pillars of my life. During all the tough times, your soothing presence and uplifting words have been my safe haven, offering peace and fortitude when I needed it the most. We've explored so many spots across various countries and continents together. I'm stoked to continue this journey and see the rest of the world with you. Also, a massive shoutout to your family! Thanks for all the amazing food and drinks in Sydney. You guys know how to make someone feel welcome and well-fed!

Jiahui, Ding Ding, Yifan, Yibo, Zhuocheng, Lunan, Yiqin, Yishu, thanks for all the fun distractions, the laughs, and those much-needed chill breaks. And thanks for dragging me to all those cool spots when my PhD budget was looking kinda sad. Your magic trick of turning

my stress into smiles and worries into chill vibes? That's been my secret weapon through all the madness. You rock!

Wrapping this up, a huge thanks to everyone who's been on this wild ride with me. Your vibes, support, and all the little things in between have shaped this thesis and me in more ways than I can count. This one's as much yours as it is mine. Cheers to all the late nights, early mornings, and every crazy moment in between.

Abstract

As the Moore's Law era approaches its end and the constraints of conventional computing paradigms become starkly evident, the pursuit of alternative, efficient computational models has emerged as a critical necessity. The human brain, with its remarkable capability and energy efficiency, stands as a compelling model from which we can draw inspiration to engineer intelligence into machines. This thesis explores the potential of Neuromorphic Nanowire Networks (NWNs), aiming to exploit their processing power and learning capabilities similar to biological neural systems. With their brain-like structures and electrical junctions resembling synapses, NWNs stand as a promising physical substrate for creating hardware systems that capture the brain's adaptability and efficiency.

This thesis investigates both simulated and physical NWNs, with a concentrated focus on understanding and leveraging their learning potential. Through in-depth investigations into the information dynamics of NWNs during learning tasks, and exploration of their learning capability under a reservoir computing framework, this research unveils new insights into unleashing the potentials of NWNs. By employing NWNs under a reservoir computing framework, it was found that their optimal learning performance correlates with specific information-theoretic measures. Additionally, drawing inspiration from the machine learning community, transfer learning, convolutional kernels, meta learning, and recursive least square were implemented to exploit the inherent learning and memory capacities of NWNs.

The findings presented in this thesis underscore the significant potential of NWNs in neuromorphic computing, paving the way for future research and applications in crafting energy-efficient, learning-capable computational systems.

Table of contents

List of Publications	xv
List of figures	xxiii
List of tables	xxvii
Nomenclature	xxix
1 Introduction	1
2 Literature Review	5
2.1 Chasing Intelligence in Machines	5
2.2 Neuromorphic Intelligence	7
2.2.1 Spike-based Systems	9
2.2.2 Memristors	10
2.2.3 Memristor-based Systems	12
2.3 Self-assembled Neuromorphic Systems	13
2.3.1 Carbon Nanotubes	14
2.3.2 Nanoparticles	15
2.3.3 Nanowire networks	15
2.4 Reservoir Computing	17
2.4.1 Reservoir computing algorithms	17
2.4.2 Non-algorithmic reservoirs	19
2.4.3 Performance Optimisation	22
2.5 Information Dynamics	24
2.5.1 Information-theoretic measures	24
2.5.2 Applications	26
2.6 Motivation and Rationale for this thesis	26

References	28
3 Information Dynamics in Neuromorphic Nanowire Networks	47
3.1 Introduction	49
3.2 Results	51
3.2.1 Junction centrality and information flow	51
3.2.2 Network dynamics of information transfer and storage	54
3.2.3 Memory capacity and learning	58
3.3 Discussion	61
3.4 Methods	64
3.4.1 Electrical network model	64
3.4.2 Junction Model	65
3.4.3 Centrality	66
3.4.4 Modularity	66
3.4.5 Information Dynamics	67
3.4.6 Memory and learning tasks	68
References	70
4 Harnessing adaptive dynamics in NWNs for transfer learning	81
4.1 Introduction	83
4.2 Methods	84
4.2.1 Modelling network connectivity and memristive junctions	84
4.2.2 Mackey–Glass time series prediction	85
4.2.3 Transfer learning	86
4.3 Results	87
4.3.1 Adaptive dynamics	87
4.3.2 Mackey–Glass time–series prediction	88
4.3.3 Transfer learning	89
4.4 Conclusions	91
References	92
5 MNIST classification using Neuromorphic Nanowire Networks	97
5.1 Introduction	99
5.2 Methods	99
5.2.1 Kernel scanning	99
5.2.2 Information Dynamics	101

5.3	Results	102
5.3.1	MNIST classification	102
5.3.2	Information dynamics	103
5.4	Conclusions	105
	References	106
6	Memristive Reservoirs Learn to Learn	109
6.1	Introduction	111
6.2	Research Methods	112
6.2.1	The L2L Framework	112
6.2.2	Learning Volterra dynamics	114
6.2.3	Learning membrane dynamics of SNNs	115
6.3	Results	117
6.3.1	L2L Volterra dynamics	117
6.3.2	L2L SNN dynamics	119
6.4	Discussion	121
6.5	Conclusion	122
	References	124
7	Online Dynamical Learning and Sequence Memory in NWN	129
7.1	Introduction	131
7.2	Results	132
7.2.1	Dynamical feature generation	133
7.2.2	Online learning	134
7.2.3	Mutual information	136
7.2.4	Sequence memory task	138
7.3	Discussion	143
7.4	Methods	146
7.4.1	Experimental setup	146
7.4.2	Online learning	147
7.4.3	Mutual information	148
7.4.4	Sequence memory task	149
	References	151
8	Concluding Remarks	161
8.1	Summary of Results	162

8.2	Future Work and Outlook	163
8.2.1	Equilibrium Propagation for NWNs	163
8.2.2	Interfacing NWNs with other spike-based systems	163
8.2.3	Connect multiple NWNs as different brain regions	164
8.3	Conclusion and Personal Remarks	164
	References	165
Appendix A Supplementary materials for original research papers		167
A.1	Supplementary information for Chapter 3	167
A.2	Supplementary information for Chapter 7	171
	References	185

List of Publications

Initials

RZ - Ruomin Zhu

ZK - Zdenka Kuncic

JL - Joseph Lizier

AL - Alon Loeffler

JH - Joel Hochstetter

JE - Jason Eshraghian

AD - Adrian Diaz-Alvarez

TN - Tomonobu Nakayama

SL - Sam Lilak

AS - Adam Stieg

JG - James Gimzewski

Publications

Chapter 3 of this thesis is published as:

Zhu, R., Hochstetter, J., Loeffler, A., Diaz-Alvarez, A., Nakayama, T., Lizier, J.T., Kuncic, Z., 2021. Information dynamics in neuromorphic nanowire networks. Scientific Reports 11, 13047. <https://doi.org/10.1038/s41598-021-92170-7>

R.Z, J.L. and Z.K. conceived and designed the study. R.Z., Z.K., J.L., J.H. and A.L. developed the model. A.D. and T.N. provided experimental validation of the model. R.Z. performed the simulations and analysed the results. Z.K. and J.L. supervised the project. R.Z. wrote the manuscript, with consultation from all authors.

Chapter 4 of this thesis is published as:

Zhu, R., Hochstetter, J., Loeffler, A., Diaz-Alvarez, A., Stieg, A., Gimzewski, J., Nakayama, T., Kuncic, Z., 2020. Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning, in: 2020 International Conference on Rebooting Computing (ICRC). pp. 102–106. <https://doi.org/10.1109/ICRC2020.2020.00007>

R.Z, and Z.K. conceived and designed the study. R.Z., Z.K., J.H. and A.L. developed the model. A.D., T.N., A.S. and J.G. provided experimental validation of the model. R.Z. performed the simulations and analysed the results. Z.K. supervised the project. R.Z. wrote the manuscript, with consultation from all authors.

Chapter 5 of this thesis is published as:

Zhu, R., Loeffler, A., Hochstetter, J., Diaz-Alvarez, A., Nakayama, T., Stieg, A., Gimzewski, J., Lizier, J., Kuncic, Z., 2021. MNIST classification using Neuromorphic Nanowire Networks, in: International Conference on Neuromorphic Systems 2021 (ICONS 2021). Association for Computing Machinery, New York, NY, USA, pp. 1–4. <https://doi.org/10.1145/3477145.3477162>

R.Z, A.S., J.L. and Z.K. conceived and designed the study. R.Z., Z.K., J.H. and A.L. developed the model. A.D., T.N., and J.G. provided experimental validation of the model.

R.Z. performed the simulations and analysed the results. Z.K. supervised the project. R.Z. wrote the manuscript, with consultation from all authors.

Chapter 6 of this thesis is published as:

Zhu, R., Eshraghian, J., Kuncic, Z., 2023. Memristive Reservoirs Learn to Learn, in: Proceedings of the 2023 International Conference on Neuromorphic Systems (ICONS 2023), ACM, Santa Fe NM USA, pp. 1–7. <https://doi.org/10.1145/3589737.3605989>

R.Z, J.E. and Z.K. conceived and designed the study. R.Z. performed the simulations and analysed the results. Z.K. supervised the project. R.Z. wrote the manuscript, with consultation from all authors.

Chapter 7 has been accepted for publications as:

Zhu, R., Lilak, S., Loeffler, A., Lizier, J., Stieg, A., Gimzewski, J., Kuncic, Z., 2023. Online dynamical learning and sequence memory in neuromorphic nanowire networks, Nature Communications 14, 6697. <https://doi.org/10.1038/s41467-023-42470-5>

R.Z. and Z.K. conceived and designed the study. S.L., A.Z.S. and J.G. fabricated the device. S.L. performed the experiments, with guidance from R.Z. and Z.K.. R.Z., A.L., J.L. and Z.K. analysed the data. R.Z. wrote the manuscript, with consultation from the other authors. Z.K. supervised the project.

I also contributed to a number of publications listed below that are not discussed in detail in this thesis, other than in the literature review:

Journal Articles

Loeffler, A., Diaz-Alvarez, A., **Zhu, R.**, Ganesh, N., Shine, J.M., Nakayama, T., Kuncic, Z., 2023. Neuromorphic learning, working memory, and metaplasticity in nanowire networks. Science Advances. <https://doi.org/10.1126/sciadv.adg3289>

Dunham, C.S., Lilak, S., Hochstetter, J., Loeffler, A., **Zhu, R.**, Chase, C., Stieg, A.Z., Kuncic, Z., Gimzewski, J.K., 2021. Nanoscale neuromorphic networks and criticality: a

perspective. *Journal of Physics: Complexity* 2, 042001. <https://doi.org/10.1088/2632-072X/ac3ad3>

Loeffler, A., **Zhu, R.**, Hochstetter, J., Diaz-Alvarez, A., Nakayama, T., Shine, J.M., Kuncic, Z., 2021. Modularity and multitasking in neuro-memristive reservoir networks. *Neuromorphic Computing and Engineering* 1, 014003. <https://doi.org/10.1088/2634-4386/ac156f>

Hochstetter, J., **Zhu, R.**, Loeffler, A., Diaz-Alvarez, A., Nakayama, T., Kuncic, Z., 2021. Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nature Communications* 12, 4008. <https://doi.org/10.1038/s41467-021-24260-z>

Li, Q., Diaz-Alvarez, A., Iguchi, R., Hochstetter, J., Loeffler, A., **Zhu, R.**, Shingaya, Y., Kuncic, Z., Uchida, K., Nakayama, T., 2020. Dynamic Electrical Pathway Tuning in Neuromorphic Nanowire Networks. *Advanced Functional Materials* 30, 2003679. <https://doi.org/10.1002/adfm.202003679>

Loeffler, A., **Zhu, R.**, Hochstetter, J., Li, M., Fu, K., Diaz-Alvarez, A., Nakayama, T., Shine, J.M., Kuncic, Z., 2020. Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* 14, 184. <https://doi.org/10.3389/fnins.2020.00184>

Manuscripts Submitted

Baccetti, V., **Zhu, R.**, Kuncic, Z., Caravelli, F., 2023. Ergodicity, lack thereof, and the performance of reservoir computing with memristive networks, *Nano Express*, under review, on arXiv. <http://arxiv.org/abs/2310.09530>

Conference Proceedings

Fu, K., **Zhu, R.**, Loeffler, A., Hochstetter, J., Diaz-Alvarez, A., Stieg, A., Gimzewski, J., Nakayama, T., Kuncic, Z., 2020. Reservoir Computing with Neuromemristive Nanowire Networks, in: 2020 International Joint Conference on Neural Networks (IJCNN). Presented at the 2020 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. <https://doi.org/10.1109/IJCNN48605.2020.9207727>

Kuncic, Z., Kavehei, O., **Zhu, R.**, Loeffler, A., Fu, K., Hochstetter, J., Li, M., Shine, J.M., Diaz-Alvarez, A., Stieg, A., Gimzewski, J., Nakayama, T., 2020. Neuromorphic Information Processing with Nanowire Networks, in: 2020 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 1–5.

<https://doi.org/10.1109/ISCAS45731.2020.9181034>

Talks and Presentations:

During my PhD, I also attended a number of conferences and presented talks and posters:

Memristive Reservoirs Learn to Learn, International Conference on Neuromorphic Systems (ICONS), **Contributed Talk**, Santa Fe, NM, Aug 2023

Meta-learning and Online-learning with Neuromorphic Nanowire Networks, Los Alamos National Lab Internal Seminar, **Invited Talk**, Los Alamos, NM, July 2023

Reservoir Computing with Neuromorphic Nanowire Networks, Neuromorphic Materials, Devices, Circuits and Systems, **Contributed Talk**, Online, Jan 2023

Priming Nanowire Networks For Neuromorphic Computing, Complexity, Criticality and Computation (C^3) Symposium, **Contributed Talk**, Heron Island, QLD, Jan 2023

Neuromorphic Computing with Nanowire Networks, RMIT STEM college Seminar Melbourne, **Invited Talk**, VIC, Nov 2022

Dynamic Self-supervised Learning and Information Recall in Neuromorphic Nanowire Networks, Emerging Aspirations in Complex Systems Workshop, **Contributed Talk**, Sydney, NSW, Sept 2022

MNIST classification using Neuromorphic Nanowire Networks, International Conference on Neuromorphic Systems (ICONS), **Contributed Talk**, Online, July 2021

Harnessing Adaptive Dynamics in Neuro-Memristive Nanowire Networks for Transfer Learning, International Conference on Rebooting Computing, **Contributed Talk**, Online, Dec 2020

Information Dynamics in Neuromorphic Nanowire Networks, Brain Criticality, **Virtual Poster**, Online, Oct, 2020

Information Dynamics in Neuromorphic Nanowire Networks, International Conference on Complex Systems, **Contributed Talk**, Online, July 2020

Publication Authorship Attribution

In addition to the statements above, in cases where I am not the corresponding author of a published item, permission to include the published material has been granted by the corresponding author.

Ruomin Zhu, Student

Date

As the supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.

Zdenka Kuncic, Supervisor

Date

List of figures

2.1	Timeline of Artificial Intelligence	6
2.2	The rapid increase of computational demands	7
2.3	The landscape of Neuromorphic systems	8
2.4	Single neuron models	10
2.5	Schematic illustration of single memristor dynamics	11
2.6	Different resistive switching mechanisms	12
2.7	Memristor crossbar array	13
2.8	Network of CNTs complexed with POM	14
2.9	Percolating network of nanoparticles	15
2.10	Nanowire networks	16
2.11	Schematic illustration of reservoir computing	18
2.12	Comparison of algorithmic RC and physical RC	20
2.13	NWN reservoir used for spoken digit classification	21
2.14	RC performance optimisation examples	22
2.15	Schematic illustration of information theory metrics	24
3.1	Simulation of neuromorphic NWN and voltage/conductance distribution	52
3.2	Junction TE as a function of centrality at different stages of activations	54
3.3	Nanowire network dynamics time series	55
3.4	Network information flow dynamics	57
3.5	Information processing tasks and corresponding information dynamics measures for networks pre-initialised with different initial states	59
3.6	Memory capacity (MC) for networks with varying information processing resources	60
3.7	A Simulated NWN and its corresponding graphical representation	64
4.1	Graph representations of 300-node networks	85

4.2	Indiviudal junction conductances and network snapshots	87
4.3	Mackey-Glass time series forecast	89
4.4	Average MG forecasting accuracy as a function of delay parameter τ	89
4.5	MG prediction accuracy with different τ_1 signals used to prime the network	90
4.6	Heatmap for MG prediction accuracy with different τ_1 and τ_2 signals	90
5.1	Schematic illustration of the kernel scanning method	100
5.2	Schematic illustration of feature extraction	101
5.3	MNIST classification result	103
5.4	Information dynamics of learning the digits “0” and “9”	104
6.1	Schematic diagram of the L2L framework.	113
6.2	Schematic diagram for the learning tasks	116
6.3	L2L Volterra dynamics	118
6.4	Reservoir activity with different pre-initialization pulse widths	119
6.5	Reservoir node voltage histograms for different pre-initialization pulse widths T_0	120
6.6	L2L SNN membrane dynamics	121
7.1	Schematic illustration of nanowire network device setup for MNIST digit classification demonstrating online dynamical learning.	133
7.2	Input–output mapping of MNIST digits in a nanowire network device	134
7.3	Online learning of MNIST digits	136
7.4	Learning rate and mutual information	137
7.5	Spatial distribution of mutual information (MI) for readout channels	138
7.6	Schematic outline of the sequence memory task	139
7.7	NWN sequence memory	141
7.8	Digit reconstruction in sequence memory task using the online-learning algorithm	142
7.9	Images of NWN device	146
S1	Junction active information storage (AIS) as a function of edge betweenness centrality c_{EB} calculated for 50 simulations	167
S2	Deviation from linearity of NWNs	168
S3	Intra-module weight ratio Γ and modularity Q as a function of time	169
S4	Flexibility of NWNs during learning tasks	170
S5	SEM images of the nanowire network without electrodes	171

S6	Simulated nanowire network	172
S7	Different device electrode setups	173
S8	Readout voltages for digits ‘0’ - ‘4’ from all 16 channels	174
S9	Readout voltages for digits ‘5’ - ‘9’ from all 16 channels	175
S10	$I - V$ curves under AC sweeps	176
S11	MNIST online classification accuracy as a function of number of channels using different electrodes for (input,drain) and different voltage amplitude and bias	177
S12	Misclassified digits in the online MNIST classification task	178
S13	Input voltage and network conductance for different MNIST digit inputs . .	179
S14	SSIM as a function of number of training samples used to train the online reconstructor	180
S15	Maximum SSIM averaged across all digits as a function of memory window length	181
S16	Reconstruction of digit 5 in the sequence memory task	182

List of tables

5.1	MNIST pixel's Mutual Information (nats) with Class	104
6.1	NRMSE for Task 1 with and without meta-learning	117
6.2	NRMSE for Task 2 with and without meta-learning	120
7.1	MNIST digit classification using an NWN device with one and five readout channels for online and batch (offline) linear classifiers.	135
A.1	MNIST classification accuracies without active error correction in the online algorithm	183

Nomenclature

Acronyms / Abbreviations

AFM Atomic Force Microscopy

AI Artificial Intelligence

AIS Active Information Storage

ANN Artificial Neural Network

ASIC Application-specific Integrated Circuits

BPTT Backward Propagation Through Time

CNN Convolutional Neural Network

CNT Carbon Nanotube

ECM Electrochemical Metallisation

ES Evolutionary Strategies

ESN Echo State Network

HRS High Resistance State

L2L Learn to Learn

LDA Linear Discriminant Analysis

LRS Low Resistance State

LSM Liquid State Machine

LTP Long-term Potentiation

MC	Memory Capacity
MEA	Multi-electrode Array
MG	Mackey Glass
MI	Mutual Information
MIM	Metal-insulator-metal
MLP	Multi-layer Perceptron
NLT	Non-linear Transformation
NRMSE	Normalised Root Mean Squared Error
NWN	Nanowire Networks
PNN	Percolating networks of nanoparticles
POM	Polyoxometalate
RAM	Random Access Memory
RBN	Random Boolean Network
RC	Reservoir Computing
ReRAM	Resistive Random Access Memory
RLS	Recursive Least Square
RNN	Recurrent Neural Network
SA	Simulated Annealing
SEM	Scanning Electron Microscopy
SNN	Spiking Neural Network
SSIM	Structural Similarity Index Measure
STDP	Spike-timing Dependent Plasticity
STP	Short-term Plasticity

SVM Support Vector Machine

TE Transfer Entropy

VLSI Very Large Scale Integration

WTA Winner-takes-all

Chapter 1

Introduction

*"The true delight is in the finding out
rather than in the knowing."*

Isaac Asimov

The human brain, a complex orchestration of roughly 86 billion neurons, stands as one of nature's most astounding creations. Its information processing capabilities surpass those of the most advanced digital supercomputers, allowing us to process vast amounts of information in mere fractions of a second and with minimal energy requirements. From the intricate art of recognising faces to the abstract understanding of emotions and concepts, the brain's power is unparalleled. Its adaptability and learning capacity, demonstrated through processes like synaptic plasticity, enable us to continually evolve, adapt, and innovate. This cognitive flexibility and efficiency have driven humans to the forefront of evolutionary success, prompting scientists and researchers to seek inspiration from the brain's architecture in the hope of replicating its abilities artificially.

Over many decades, a myriad of approaches have been proposed to replicate the intelligence of the human brain, ranging from sophisticated neural network software architectures to the more philosophical concept of the "brain in a vat" (a hypothetical scenario in which a brain is sustained in a vat and connected to external stimuli) challenging our perceptions of reality and consciousness. While efforts have been made to emulate the brain's intricate network of neurons and synapses purely in software, hardware or wetware remains an aspirational goal, significant strides have been made in neuromorphic engineering. This field attempts to design systems, not necessarily to mimic the brain in exact detail, but to capture its efficiency, adaptability, and learning mechanisms. Through these efforts, the dream of harnessing the brain's intelligence in artificial systems inches closer to reality.

Nanowire Networks (NWNs), the subject of this thesis, stand out as a particularly promising avenue in this quest. Their inherent complex brain-like topology mimics the intricate connectomes found within biological neural systems. This structural complexity is further enriched by synapse-like cross-point electrical junctions, which enables dynamic information processing akin to natural neural networks. Beyond mere structural similarities, NWNs offer the potential to emulate the adaptability and learning capabilities of the brain. Throughout this thesis, we delve into both simulated results and experimental measurements of neuromorphic NWNs, with a keen focus on understanding their learning potential. The overarching ambition of my research is to construct a system that can learn not just with the prowess of the human brain, but also with its efficiency. This thesis documents my preliminary venture into this domain, and the findings presented here are encouraging indicators of the path ahead.

Chapter 2 presents a comprehensive review of related studies in the literature, providing a foundational understanding of the field. It not only contextualises our research but also highlights the established knowledge and methodologies that our study seeks to expand upon.

To fully understand the learning capabilities of NWNs, it is essential to first unravel their information processing mechanisms. Chapter 3 delves into the information dynamics of NWNs during learning tasks. We primarily focus on how nanowires communicate with each other and how memristive junctions store and process information. Additionally, our study measures the network's information dynamics at different activation levels.

Chapters 4 and 5 further investigate the learning capability of NWNs under a reservoir computing framework. In Chapter 4, we investigate the transfer of knowledge between learning tasks, with the objective of leveraging past knowledge for subsequent learning tasks. Chapter 5 employs a kernel to feed information to the network convolutionally, enhancing the network's ability to learn spatial information.

A primary challenge in reservoir computing with NWNs is determining the ideal initial conditions and operational regime for specific tasks. Addressing this challenge often requires a substantial amount of trial and error. Chapter 6 introduces a structured method to tackle this issue, identifying the best initial conditions for a family of related tasks through meta-learning.

In Chapter 7, we implement a recursive least square algorithm on the readouts from a physical NWN device, enabling online MNIST handwritten digit classification. The inherent memory of the system is further utilised to learn the patterns in sequences formed from the MNIST digits. This represents the first instance of the MNIST classification benchmark task being performed with a physical NWN device.

Chapter 8 summarises the key findings of this thesis and offers insights into potential avenues for future research.

The findings of this thesis pave the way for novel applications and deeper insights within the realm of neuromorphic computing and engineering. Through rigorous exploration and innovative methodologies, we have demonstrated the vast potential of NWNs in various learning tasks. As the world edges closer to realising efficient brain-inspired computational systems, the research documented here offers a beacon, illuminating the path forward and underscoring the profound possibilities that lie ahead in this exciting domain.

Chapter 2

Literature Review

“吾尝终日而思矣，不如须臾之所学也。”

“I have pondered all day, but it's not as good as what I learn in a moment.”

劝学 An Exhortation to Learning

- 荀子 Xun Zi

2.1 Chasing Intelligence in Machines

In recent years, Artificial Intelligence (AI) has taken center stage in technological innovation. Driven by advances in machine learning, data availability, and computing resources, AI's impact is being felt across a broad range of areas. For instance, AlphaGo astonished the world in 2015-16 by defeating world champions in the complex game of Go [1] while in the last 12 months, chatGPT has elevated natural language processing to a whole new level [2].

The idea of realising intelligence in machines was first explored by Alan Turing in his work titled "Intelligent Machinery" [3] in 1948. Subsequent efforts shifted towards looking at the brain's neural network for inspiration and the phrase *artificial intelligence* was then formally established at the Dartmouth Workshop later in the 1950s [4]. As shown in Fig. 2.1, the first significant model of an artificial neuron came in the form of the Perceptron, developed by Frank Rosenblatt in 1957 [5]. The Perceptron model was a single-layer neural network that could perform simple classification tasks, and it drew interest in the then new idea that machines could be trained to learn from data. Subsequent studies produced the multi-layer perceptron (MLP) model to study more complex and nonlinear relationships in

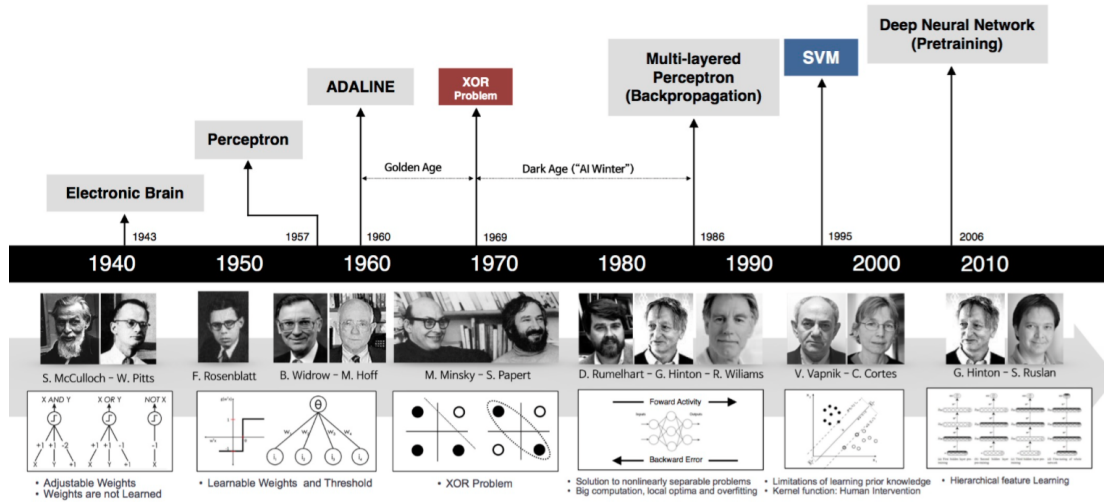


Fig. 2.1 Key milestones in the development of artificial intelligence and neural networks. Adapted from [beamlab](#).

data [6, 7]. Nevertheless, skepticism over the capabilities of these early models increased as they did not meet the inflated expectations for solving “real-world” problems (like the XOR problem), resulting in *the first AI winter* in the 1970s [8–10]. The development of the well-known backward propagation algorithm in the 1980s offered a way to train MLPs efficiently and help revive the field [11]. Yet, despite these advances, the field faced *the second AI winter* in the late 1980s and early 1990s, due to overpromised capabilities and insufficient computing and data resources [7, 9]. Supervised and unsupervised machine learning algorithms like Support Vector Machines (SVMs) [12] and Random Forests [13] proved effective across a range of tasks and applications and brought attention back to the field. The AI landscape began to change dramatically in the 2000s, notably with the advent of GPUs for highly paralleled generendeavour computation, which provided exponential increases in processing power and enabled the training of much larger and more complex models based on Artificial Neural Networks (ANNs), leading to the era of deep learning using multi-layer feed-forward ANNs (effectively the same as the MLPs introduced years earlier) [14–16]. Combined with the massive amount of data made available by the rise of the internet, and the associated centralised data centers, these developments have catalysed the current surge in AI.

As the AI landscape continues to evolve, it simultaneously encounters new frontiers that challenge conventional computing methods, notably through an unsustainable surge in power requirements and inherent limitations in existing hardware architectures. In response, the field has started to draw inspiration from biological neural systems, leading to the rise of

neuromorphic intelligence, which aims to mimic the efficiency and adaptability of human brains.

2.2 Neuromorphic Intelligence

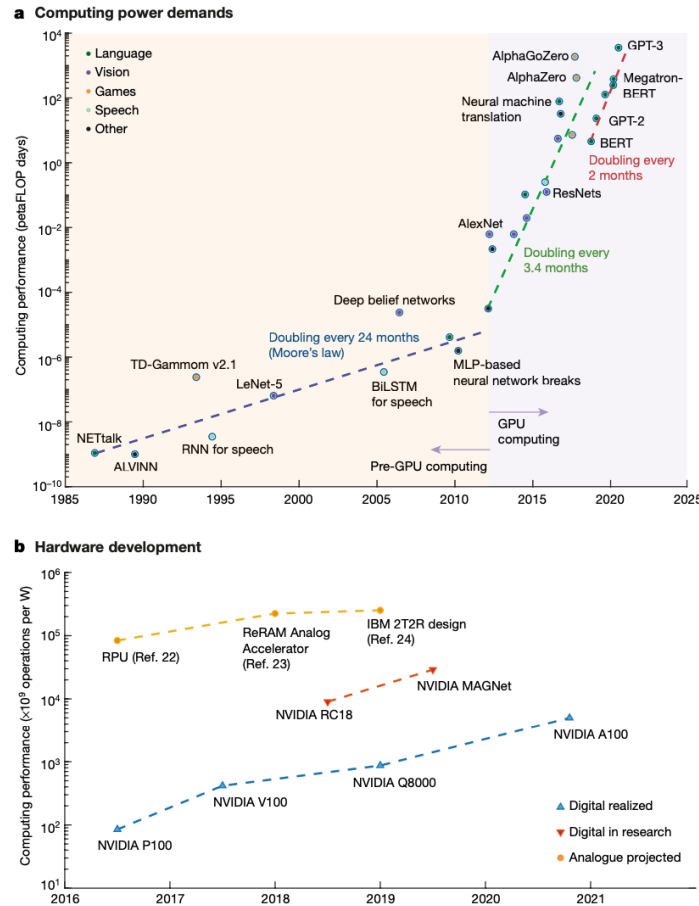


Fig. 2.2 The increase in computational demands of AI has outpaced the development of hardware. (a) Computing power of major AI models over the past four decades presented in petaFLOP days. Recent advances in large language models double in demand every 2 months. (b) Development of digital and analogue compute hardware since 2016. Adapted from Mehonic and Kenyon [17].

ANNs used in AI are increasingly powerful, but they are also accompanied with significant hidden costs. Fig. 2.2 (a) demonstrates the exponential increase of computing power demands to train AI models. Moreover, recent breakthroughs in language models have resulted in a doubling demand for computing power every 2 months, which has significantly outpaced the hardware trend predicted by Moore's law, as shown in Fig. 2.2 (b) [17]. With

the end of the Moore's law era, it appears this challenge cannot be addressed by conventional computing paradigms.

In contrast to ANNs, the human brain, comprised of around 10^{11} neurons and 10^{15} synapses [18], operates with exceptional efficiency, with a power consumption of approximately 20 W. In comparison, a digital simulation of an ANN with similar size consumes 7.9 MW [19]. Much of the disparity in power consumption between the human brain and digital ANN computations can be attributed to the inherent inefficiencies of the von Neumann architecture, in which memory and processing units are located separately, thus using significant energy to bus data back and forth. Brain-inspired, or *neuromorphic*, approaches to computing look to the brain for efficiency gains. Immediate energy efficiency gains can be realised by simply integrating processing and memory, but the human brain is also an exemplar of “intelligence”, and thus offers the potential for enhancing learning efficiency and efficacy.

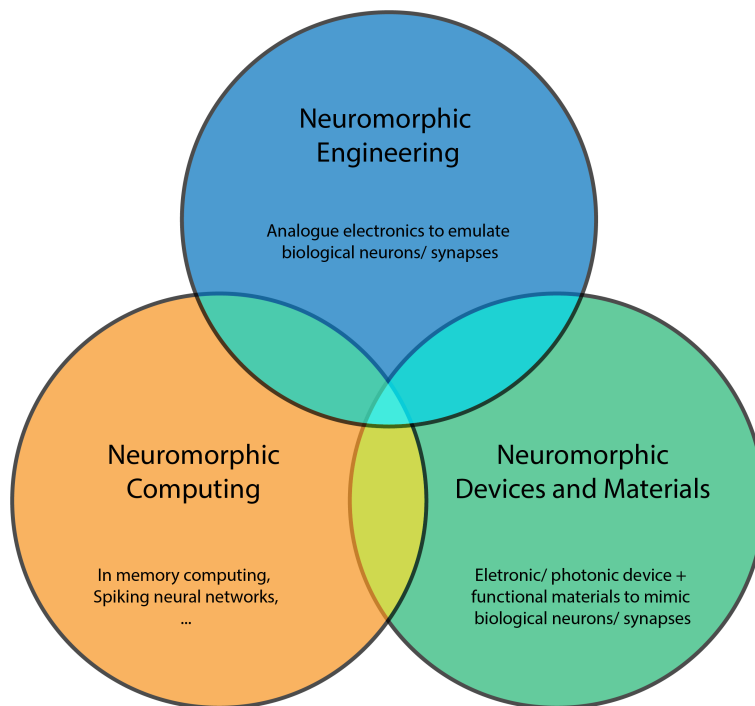


Fig. 2.3 The landscape of neuromorphic or brain-inspired systems, based on the definition by Mehonic and Kenyon [17].

The concept of neuromorphic engineering was first proposed by Carver Mead in the late 1980s and aims to engineer a new type of electronic hardware which mimicks the architecture and efficiency of biological brains for computational purposes [20–22]. After several decades

of development, the field of neuromorphics has matured to a point where it can be broadly categorised into at least three key areas (cf. Fig. 2.3) [17, 23]:

1. “Neuromorphic engineering”, which focuses on emulating the functions of biological neurons and synapses using analogue electronics [24];
2. “Neuromorphic computing”, which seeks to conceive new data processing techniques like co-locating memory and computation [25], integrating spiking mechanisms from biological neurons into artificial neural networks [26, 27]; and
3. “Neuromorphic devices and materials”, which employs novel electronic and photonic devices leveraging functional materials with innate physical properties that mimic biological neurons and/or synapses [28–30].

These areas are not mutually exclusive or exhaustive, and some overlap with broader research efforts in unconventional computing paradigms, for example, computing with physical systems [31]. This thesis focusses on neuromorphic devices and materials and highlights important intersections with the other two areas.

2.2.1 Spike-based Systems

Spiking neural networks (SNNs) represent a unique class of ANNs that incorporates spiking neurons as their fundamental computational units [32, 33]. As demonstrated in Fig. 2.4, in contrast to traditional ANNs, which use continuous activation functions and operate with floating-point states, SNNs use a discrete, spike-based information processing scheme inspired by action potentials observed in biological neural systems [34]. This spike event-driven architecture captures the temporal relationships embedded in data and offers a feasible way to bridge the energy efficiency gap between neural networks and biological brains, making it a promising candidate for next generation neuromorphic machine intelligence [32].

Spike-based neuromorphic chips are designed to exploit the functionality and efficiency of SNNs [23, 35–39]. These specialised chips integrate compute and memory in a non-von Neumann fashion and employ circuits that emulate spiking neurons to achieve low-latency and low-power consumption. Notable examples include IBM’s TrueNorth [40], Intel’s Loihi [41] and several other systems developed under the Human Brain Project such as SpiNNaker [42], BrainScaleS [43] and DYNAPs [44].

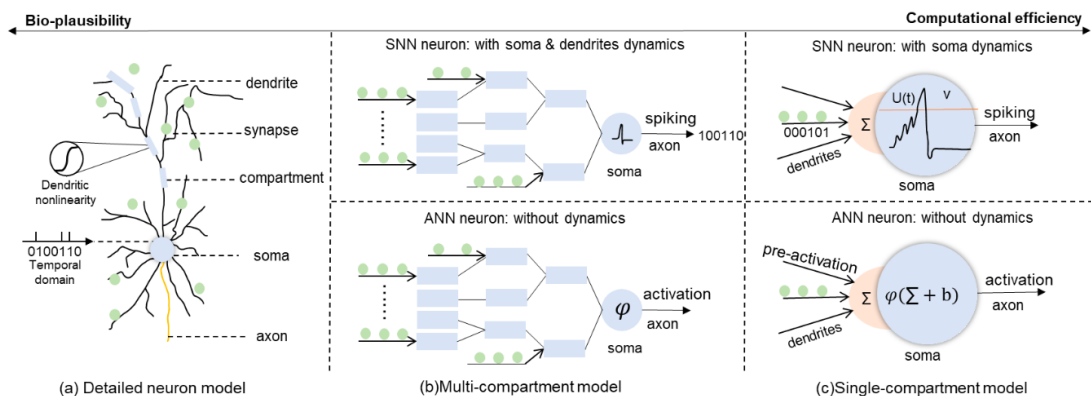


Fig. 2.4 Single neuron models. From left to right, the models are computationally more efficient but biologically less plausible. (a) Fine-structured neuron model with thousands of compartments on the dendrites. (b) Neuron model with dendrites and soma. (c) Point neuron model with soma dynamics only (perceptron model). The top panels in (b) and (c) show spiking neuron models, in which temporal dynamics are taken into consideration and the neurons communicate via spikes. The models shown in the bottom panels are static artificial neurons. Adapted from Li et al. [39].

Nevertheless, training SNNs presents unique challenges compared to traditional ANNs. Traditional training algorithms like backpropagation are often unsuitable for SNNs due to the temporal dynamics of SNNs as well as the discontinuous and non-differentiable nature of spike events. Neftci et al. employed surrogate functions to approximate the derivative of spike events, thus enabled the training of SNNs via backpropagation through time (BPTT) [45]. However, unrolling the computing graph in BPTT across numerous time steps escalates the computational complexity quadratically [26]. While SNNs hold promise for energy-efficient neuromorphic hardware and emulating more biologically plausible computation, realising their full potential demands novel techniques and methodologies to overcome these training challenges [46].

2.2.2 Memristors

First predicted by Leon Chua in 1971 as the missing fourth fundamental circuit element [47], the concept of memristors (short for “memory resistors”) remained theoretical until a practical device was finally created by Strukov et al. at the HP Labs in 2008 [48]. Memristors are two-terminal (three-terminal realisations can be found in [49]) devices unique for their variable resistance, which is modulated by the time integral of applied electrical stimuli, thereby enabling both memory and nonlinearity [50]. While still under development, memristors

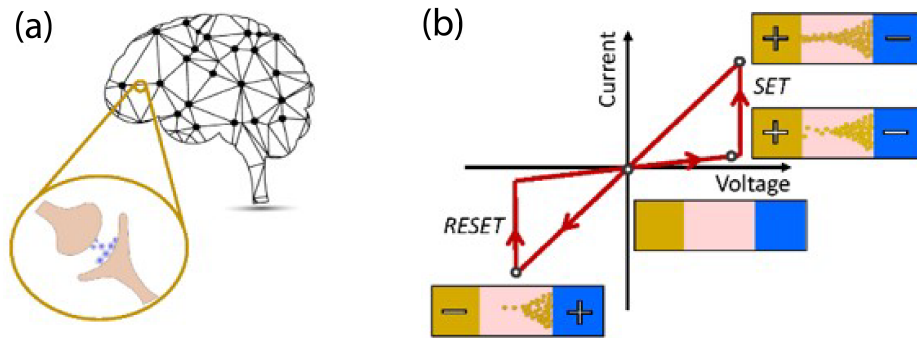


Fig. 2.5 Schematic illustration of single memristor dynamics. (a) In biological brains, synapses play an important role in storing and transmitting information. (b) The operation cycle of a filamentary memristor, showing the characteristic hysteresis curve in $I - V$ phase space, caused by the evolution of a conductive nano-filament. Adapted from [Empa](#).

are rapidly emerging as strong candidates for next-generation non-CMOS and post-von Neumann computational systems [30, 50–54].

The phenomenon of resistance switching in memristors can be characterised by two primary states: the High Resistance State (HRS) and the Low Resistance State (LRS). In common metal-insulator-metal (MIM) memristors, the transition between these states is controlled by a thin layer that is electrically insulating, but ionically conducting. As Fig. 2.5 (b) indicates, when an external current or voltage is applied to a MIM memristor, metallic ions migrate within the thin insulating film and redox reactions neutralise the ions to form metallic nano-scale filament in a process referred to as electrochemical metallisation (ECM) [54]. Once established, thermodynamic fluctuations can rupture conductive nano-filaments, driving transitions between HRS and LRS. A similar process occurs in metal-oxide memristors due to migration of oxygen vacancies [52]. These nanoscale processes play a pivotal role in determining the efficiency and behaviour of memristive systems. The shift from HRS to the LRS is often referred to as the SET process while the process from the LRS back to the HRS is known as the RESET process [50, 52]. Fig. 2.6 demonstrates the two types of switching behaviour, namely unipolar and bipolar. In unipolar switching, the SET and RESET processes occur at different voltage levels but in the same polarity. In contrast, bipolar switching requires opposite polarities for the SET and RESET processes [55].

Memristive devices can be further categorised based on their memory characteristics into volatile and non-volatile types. The conductance of volatile memristors decays when external stimulation is removed, which allows them to simulate specific synaptic and neural dynamics. This attribute has been leveraged in various applications, ranging from spiking neural networks to combinatorial optimisations. Conversely, non-volatile devices retain their

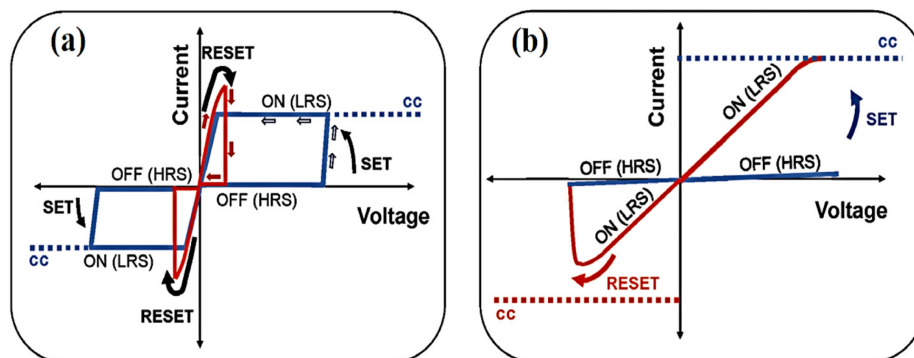


Fig. 2.6 Different resistive switching mechanisms. (a) Unipolar resistive switching, where voltages of the same polarity switch the memristor ON or OFF. (b) Bipolar resistive switching, where voltages of opposite polarities switch the memristor ON or OFF. Adapted from Mohammad et al. [56].

resistance state even when power is disconnected, which makes them ideal for long-term data storage and essential for performing matrix multiplication in memristor arrays [54]. This thesis focuses on volatile, bipolar MIM memristors that form between intersecting silver nanowires coated in an insulating nano-layer, so resistive switching is due to ECM.

2.2.3 Memristor-based Systems

Memristors have opened up new possibilities across a wide range of neuromorphic computational applications thanks to their distinctive characteristics like nonlinearity and memory. Notably, as shown in Fig. 2.7, the crossbar architecture is often employed in memristor applications as it closely mimics the structure of feedforward neural networks. In a crossbar array, memristors are placed at the junctions between horizontal and vertical conductive bars, effectively serving as synaptic weights in neural networks [57, 58]. When voltage is applied to the horizontal bars, rapid matrix multiplication operations occur, and the currents measured at the vertical bars represent the output, in accordance with Ohm's law [23, 28, 36–39, 53, 54, 59].

With their unique capability to co-locate memory and computing, memristors emerge as natural candidates for in-memory computing [25]. For instance, the HRS and LRS of memristors can effectively serve as digital “0”s and “1”s respectively, enabling them to perform logic operations directly through voltage inputs [62, 63]. Haj-Ali et al. have implemented this idea to realise image processing in-memory [64]. Moreover, Papandreou et al. showed that certain memristive devices can be programmed to specific conductance levels, enhancing the capabilities of in-memory computing systems [65]. More examples can be found in the review by Ielmini et al. [25].

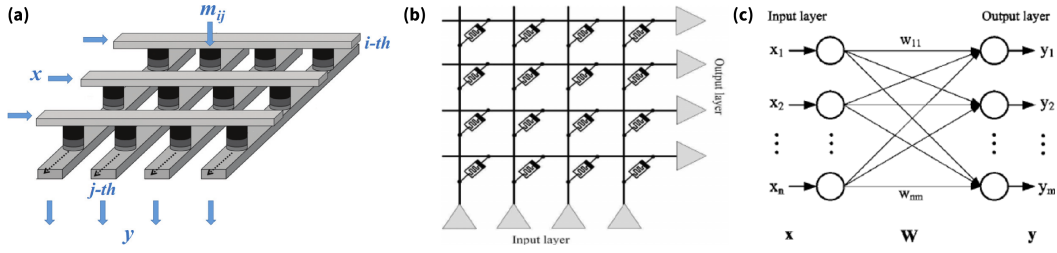


Fig. 2.7 Memristor crossbar array. (a) Schematic illustration of a memristor crossbar array. The top and bottom metal bars are placed perpendicularly, with the cross points in between filled with memristive materials. Voltages are applied to the horizontal bars and the current of the vertical bars are read out. Adapted from Huang et al. [60]. (b) Memristor crossbar circuit. (c) Graph representation of the cross bar architecture. (b) and (c) are adapted from Merrikh-Bayat et al. [61].

Additionally, their capability for rapid matrix multiplication makes memristors well-suited as accelerators in deep learning applications [66–68]. When integrated with SNNs, memristors not only align well with the encoding mechanisms of SNNs but also offer architectural advantages that collectively enhance computational efficiency [69–71]. Furthermore, memristors are also finding applications in reservoir computing, discussed in more detail below in Section 2.4. Various applications have been demonstrated using reservoir computing with memristors, including time-series prediction [72], image classification [72, 73], and spatio-temporal signal processing [74, 75].

This exploration into memristors’ potential in neuromorphic applications paves the way for an exciting development: the use of self-assembled nanomaterials with inherent memristive properties. These novel materials promise to revolutionise neuromorphic computing by offering enhanced scalability, fabrication efficiency, and more intricate network structures, addressing some of the constraints encountered with traditional memristor-based architectures.

2.3 Self-assembled Neuromorphic Systems

As discussed in the previous section, memristors represent a class of nanomaterials with intrinsic neuromorphic properties, namely memristive switching, due to nanoscale electrochemical processes. Whereas memristor devices are typically fabricated using conventional bulk materials with top-down methods, another class of neuromorphic nanomaterials is one in which numerous memristive elements can be synthesised using bottom-up techniques. This section focuses on novel, low-dimensional memristive nanomaterial systems synthesised

via self-assembly. Such systems not only exhibit the neuromorphic features of traditional memristors, but also provide additional advantages, such as improved scalability, easier fabrication, and denser interconnects owing to the formation of a disordered network structure [28–30, 76].

2.3.1 Carbon Nanotubes

Carbon nanotubes (CNTs) are unique one-dimensional materials with exceptional electrical, thermal, and mechanical properties [77–80]. A carbon nanotube is a cylinder of rolled-up, one-atom-thick carbon atoms, exhibiting metallic or semiconducting properties. Synapses constructed from CNT networks exhibit memristive dynamics and are able to mimic the behaviour of biological synapses, thereby allowing the creation of synaptic circuits for use in neuromorphic computing systems [81, 82].

Zhao et al. assembled optically gated CNT synapses into a crossbar array, which allows very efficient parallel learning of multiple functions within large arrays and enables high reliability, high density as well as fast learning speed [83]. Random networks of CNTs exhibit electrical nonlinearity and random fluctuations that mimic both the complex connectivity and neural dynamics in biological brains [80, 84–86]. As shown in Fig. 2.8, Tanaka et al. complexed a network of CNTs with polyoxometalate (POM) which could work as a good candidate for neuronal membrane device. The CNTs in this system work as a synapse connecting the POM cells, which generate spikes and noise spontaneously. As a result, the network exhibits rich spiking dynamics and learning capabilities, making it well-suited for reservoir computing applications [85].

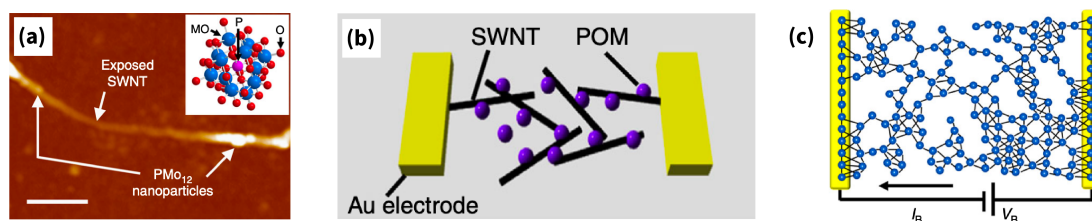


Fig. 2.8 Network of CNTs complexed with POM. (a) AFM image of the single wall carbon nanotube (SWNT)/ POM complex. Scale bar: 200 nm. (Inset: Molecular structure of PMo_{12}). (b) Schematic illustration of the network and device setup. (c) Cellular automata model on a 2D grid consisting of POM cells (blue spheres) connected with SWNT wires (black). Adapted from Tanaka et al. [85].

2.3.2 Nanoparticles

Metal nanoparticles and associated thin films are also drawing increasing attention as a promising substrate for neuromorphic computing. Vahl et al. used gas phase synthesis to deposit alloy nanoparticles like AgAu and AgPt between nanoscopic SiO₂ thin films [87]. Diffusive memristive switching was observed when probing individual or small clusters of these nanoparticles arranged in a SiO₂/NP/SiO₂ or SiO_xNy/NP/SiO_xNy configuration using conductive atomic force microscopy (c-AFM) [88]. This phenomenon is associated with the local formation and dissolution of Ag filaments [89]. Similar resistive switching behaviour and complex spiking dynamics were also observed in cluster assembled Au films, which made them suitable for neuromorphic computing [90–92].

Thin films of percolating nanoparticles exhibit resistive switching behaviour and quantised conductance at room temperature, facilitated by atomic-scale filaments forming in tunnel gaps between the particles [93, 94]. Beyond simple switching, these nanoparticle can self-assemble into complex networks (percolating nanoparticle networks, PNNs, cf. Fig. 2.9) that inherently exhibit neuromorphic behaviours, including stochastic conductance switching [95] and the ability to convert from short-term memory to long-term potentiation [94]. These networks operate at a percolating phase transition, where they demonstrate brain-like spatiotemporal correlations and criticality, giving rise to phenomena like avalanches of switching events, which is commonly observed in biological brains [96–98].

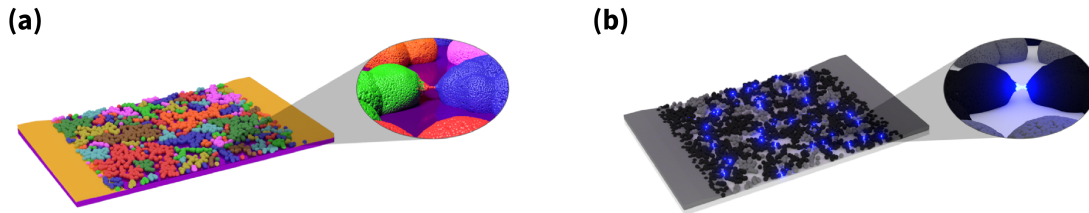


Fig. 2.9 Schematic illustration of percolating networks of nanoparticles. (a) Different colours represent groups of particles that are in contact with each other. The zoom-in window shows the growth of an atomic conductive nano-filament. (b) Conducting pathways (black) formed as a result of filament growth. The zoom-in window shows the atomic conductive nano-filament as well. Adapted from Mallinson et al. [96].

2.3.3 Nanowire networks

Nanowire networks (NWNs, cf. Fig. 7.9) represent a unique class of memristive systems with which to realise neuromorphic intelligence [99, 100]. The first studies to investigate NWNs for brain-inspired computing and learning tasks can be identified with the development

of “atomic switch networks” by Stieg et al. at UCLA [101, 102]. These self-assembled networks of Ag_2S nanowires were inspired by earlier work on Ag_2S atomic switch devices that exhibit memristive switching due to ECM-induced Ag filament evolution [103, 104]. Hasegawa et al. showed that single atomic switches can achieve learning via their short-term and long-term memory [105] and a related study later showed that these single devices demonstrate synapse-like short-term plasticity and long-term potentiation [106]. These studies explored the learning capacity of individual memristive junctions, which laid the foundation for unravelling the learning potential of NWNs.

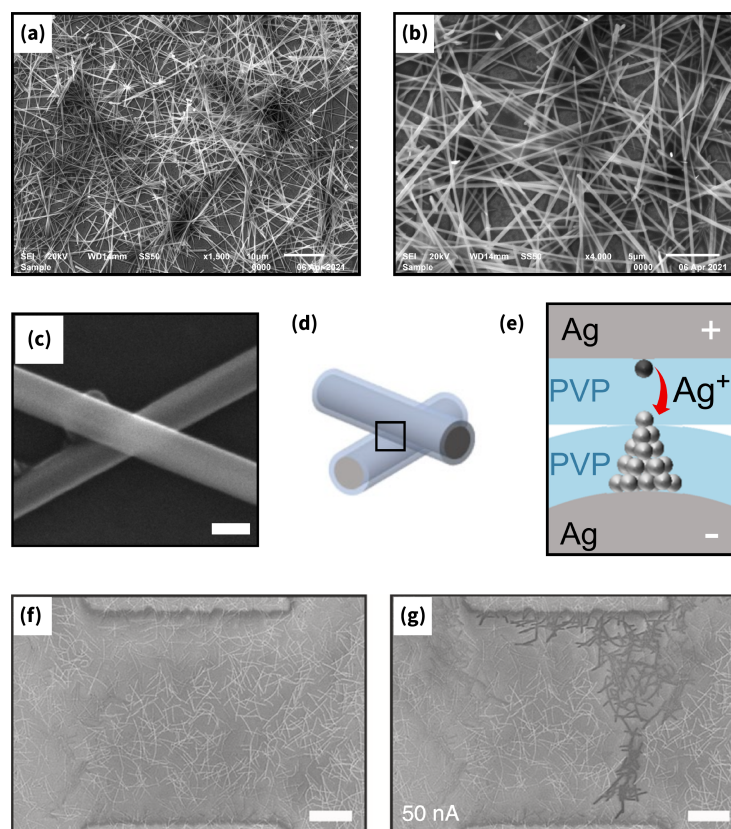


Fig. 2.10 Nanowire networks. (a), (b) SEM images of a AgSe nanowire network at different scales. Scale bars: (a) $10\mu\text{ m}$, (b) $5\mu\text{ m}$. Credit: Sam Lilak, UCLA. (c) SEM image of a cross-point junction between two Ag-PVP nanowires. Scale bar: 100 nm . Adapted from Milano et al. [107]. (d), (e) Schematic illustration of Ag-PVP cross-point junctions. The switching dynamics is modulated by the formation/ rupture of the metallic Ag filament. Adapted from Milano et al. [99]. (f), (g) SEM images of the emergence of a “winner-takes-all” conducting pathway between source and drain electrodes in a Ag-PVP NWN. Scale bar: $2\mu\text{ m}$. Adapted from Manning et al. [108].

Subsequent studies demonstrated that the self-assembled structure of NWNs exhibit small-world and brain-like connectivities [109–111]. When a DC voltage bias is applied across these networks via electrodes, the conductance on individual junctions grows unevenly [112] and a “winner-takes-all” current pathway can be observed [108], resulting in complex responses in device-level dynamics [113], and enabling programability [114]. Studies also indicated that NWNs exhibit emergent dynamics [101, 115] and avalanche criticality [116, 117] during dynamical phase transitions which are thought to be crucial for information processing in the brain [118]. With these complex dynamics and adaptability, NWNs stand out among self-assembled systems for their demonstrated superior performance as physical reservoirs, with their diverse dynamics enabling more efficient and brain-like computations. Further attributes of neuromorphic nanowire networks are detailed in the a recent review paper by Kuncic and Nakayama [100].

2.4 Reservoir Computing

As discussed above, self-assembled nanomaterial systems exhibit complex connectivity and nonlinear dynamics mimicking biological brains. These features make them well-suited for reservoir computing, which projects input data into a higher-dimensional space, enabling training to be reduced to a relatively simple linear readout layer.

2.4.1 Reservoir computing algorithms

Reservoir computing (RC) is a machine learning paradigm that utilises a special type of recurrent neural network (RNN) called a reservoir to perform computations [119–121]. It was introduced independently by Maass et al. [122] (as liquid state machine, LSM) and Jaeger [123] (as echo state network, ESN) in the early 2000s. As shown in 2.11, a typical reservoir computing architecture consists of three main layers: the input layer, the reservoir layer, and the output layer. In this configuration, the input weights $\mathbf{W}_{\text{in}}$ and internal weights $\mathbf{W}_{\text{res}}$ of the reservoir are pre-generated randomly and fixed and only the readout weights $\mathbf{W}_{\text{out}}$ are trained using linear algorithms. Moreover, the reservoir integrates a pre-configured RNN with a random and sparse connectivity, where the behaviour of each neuron is described by a non-linear activation function [122, 123]. The system was designed to address the computational burden as well as the vanishing/exploding gradient problem of traditional RNN training methods by limiting the training to the output layer [121, 124, 125].

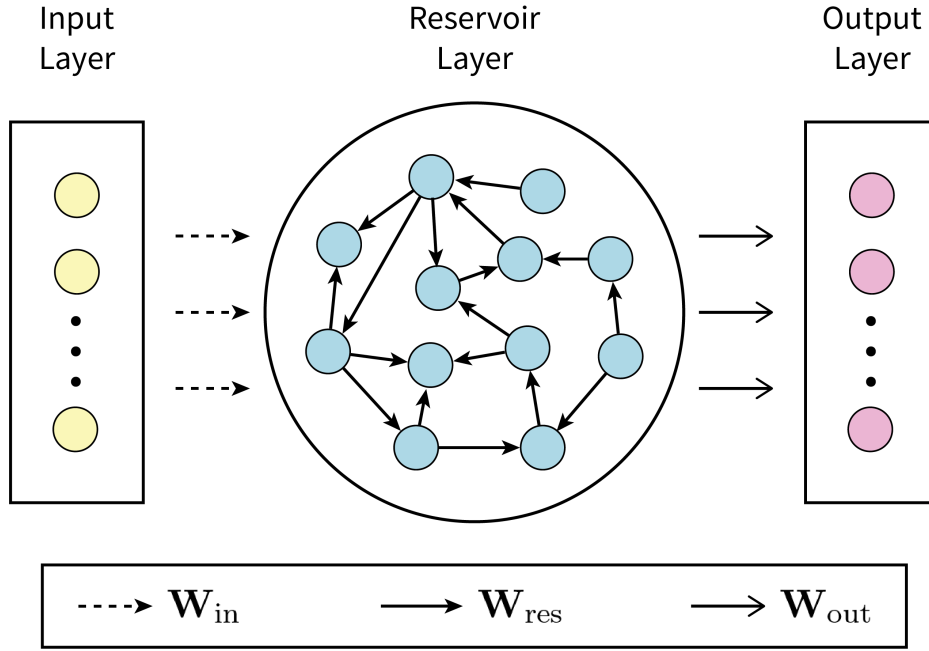


Fig. 2.11 Schematic illustration of reservoir computing, consisting of three layers: input, reservoir and output. While $\mathbf{W}_{in}$ and $\mathbf{W}_{res}$ are pre-generated, $\mathbf{W}_{out}$ is trained in a supervised manner.

While ANNs are generally optimised for static data, RC is designed to handle dynamic spatio-temporal inputs [125]. Algorithmically, the dynamics of an RC system can be described as:

$$\mathbf{x}(t+1) = f(\mathbf{W}_{in}\mathbf{u}(t+1) + \mathbf{W}_{res}\mathbf{x}(t)), \quad (2.1)$$

where $\mathbf{x}(t)$ represents the states of the neurons inside the reservoir at time t , and $\mathbf{u}(t+1)$ is the input at the next time step, and f is a nonlinear activation function (commonly tanh). The readout layer of the reservoir performs a linear combination of the output neuron states $\mathbf{x}_{out}(t)$:

$$\hat{\mathbf{y}}(t) = \mathbf{W}_{out}\mathbf{x}_{out}(t), \quad (2.2)$$

The implementation of RC with temporal inputs involves three main steps: initialisation, training, and evaluation. During the initialisation phase, the input and internal weights of the reservoir, as well as the network's connectivity, are randomly generated. A fraction of the training signal is fed into the reservoir, allowing the internal phase space to adapt to the input dynamics and wash out the influence from random initial conditions [125]. In the training

phase, the readout weights are trained using a learning algorithm such as linear regression or ridge regression to minimise the loss for all training samples in a supervised manner:

$$\mathcal{L} = \|\mathbf{y} - \mathbf{W}_{\text{out}}\mathbf{x}_{\text{out}}\|_2, \quad (2.3)$$

where $\mathbf{y}$ is the target signal and $\|\cdot\|_2$ represents the L_2 norm. In the evaluation phase, the reservoir states corresponding to new input data are linearly combined using the trained readout layer as described by Eq. 2.2, and the reservoir's performance is thereby estimated [119, 124, 125].

As a result of the random connectivities and the nonlinear dynamics of individual neurons, the reservoir demonstrates complex and rich temporal behaviors in its collective dynamics, enabling it to project the input data into a high-dimensional feature space [122–124, 126]. In the case of ESNs, Hermans and Schrauwen showed that in the limit of infinitely large networks, the reservoir operates in a manner effectively equivalent to a recursive kernel such as that used in recursive SVMs [127]. Jaeger et al. showed that reservoirs also exhibit short-term memory due to the recurrent connectivity, allowing them to retain and process information from past inputs [124, 128, 129]. As new inputs are fed into the reservoir, its internal dynamics evolve, capturing and encoding temporal patterns and the reservoir's state at any given time contains a collective memory of the past inputs it has encountered [122, 123]. This memory enables reservoirs to exhibit temporal dependencies and effectively handle tasks that require the analysis and prediction of sequential data [128, 130, 131]. The nonlinearity and memory within reservoirs forms the foundation for their ability to learn and generalise from temporal information, making them powerful tools in various machine learning applications [125, 132–135].

2.4.2 Non-algorithmic reservoirs

Algorithmic reservoirs have shown promising results in various applications such as time series analysis and prediction [136–138], audio processing and speech recognition [139], control tasks [140, 141], and pattern recognition [142]. As shown in Fig. 2.12, the idea of employing hardware for RC arose naturally and has attracted significant interest, along with various feasible applications [143]. In RC, the primary role of the reservoir is to nonlinearly map the input into a high-dimensional space while retaining memory [121] and physical systems offer numerous advantages such as faster, more energy-efficient computation, in-material data processing, optimised resource usage and easy integration into different applications [125, 143]. Additionally, physical reservoirs are constrained by

conservation laws (e.g. Kirchoff's circuit laws), thus circumventing the main drawback of algorithmic RC, hyperparameter optimisation, and their activation functions, rather than being represented as abstract mathematical functions, are governed by the physics of their individual elements (e.g. memristors). As a result, the quest for suitable physical RC hardware has concentrated on two areas: dynamical systems capable of generating nonlinear responses to the input and neural network hardware that exhibits RNN structure [143].

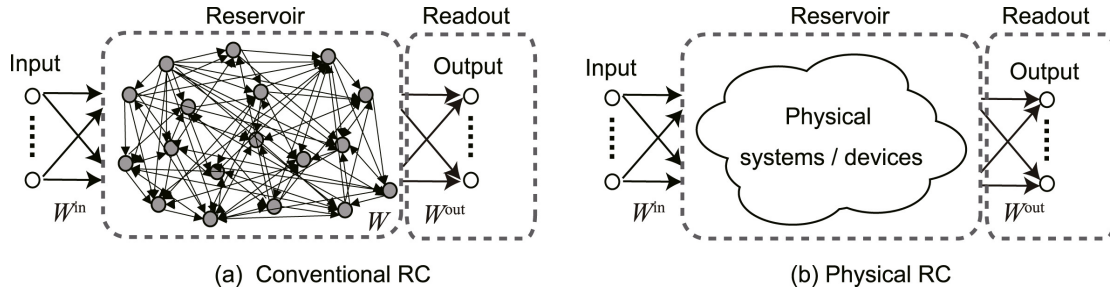


Fig. 2.12 Comparison of algorithmic RC and physical RC. (a) Conventional algorithmic RC using recurrent neural networks as reservoirs. (b) Physical RC using physical systems/devices as reservoirs. Adapted from [143].

Neuromorphic Reservoirs

Neuromorphic circuit play an important role for physical RC. Very large scale integration (VLSI) are capable of realizing brain or bio-inspired circuits and application-specific integrated circuits (ASICs) have been designed to realise RC [144, 145]. The inherent long-term memory of memristors in particular makes them well-suited for RC and integrating them into a network enables computing for complex tasks. Du et al. and Moon et al. implemented dynamical memristor arrays for RC to perform tasks like digit classification and time-series predictions [72, 146]. Spintronic systems have also been used for RC as they exhibit non-volatile memory (see definition in Section 2.2.2) and rich dynamics. Torrejon et al. utilised spintronics under the RC framework and successfully demonstrated spoken digit recognition [147]. More details can be found in the recent review paper by Tanaka et al [143].

In the context of neuromorphic nanomaterial systems, NWNs show interesting neuromorphic properties which enable them to process information in a manner suitable for RC. Related studies have shown that they are able to achieve current pathway tuning [148], associative learning [149], as well as cognitive functions [150]. Nevertheless, given their nonlinearity, high dimensionality and fading memory, NWNs' most promising application for neuromorphic computing is to use them as reservoirs in RC [100, 151]. Sillin et al. [152] and Demis et al. [102] implemented multi-electrode arrays (MEA) on self-assembled

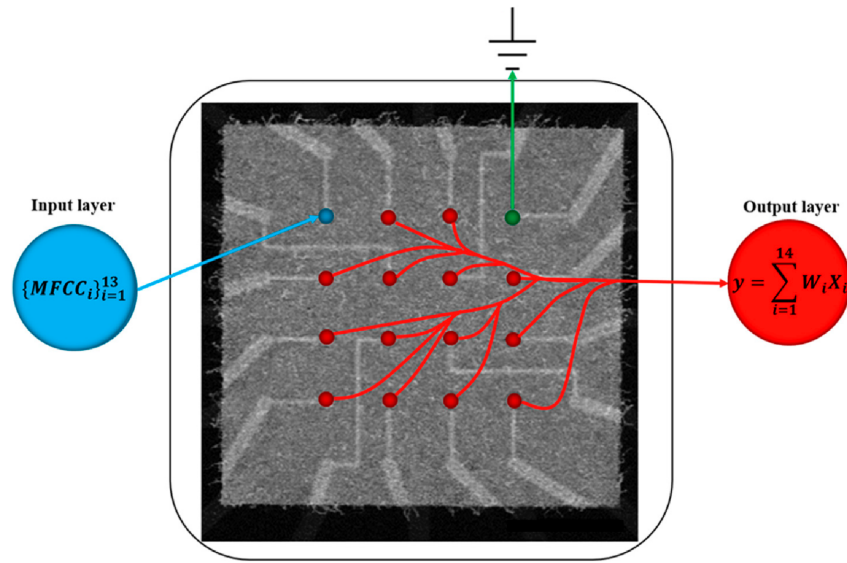


Fig. 2.13 AgI NWN multi-electrode device employed as a reservoir for spoken digit classification. Adapted from Lilak et al. [153].

Ag₂S NWNs to send in and read out information, enabling reservoir computing using these networks. Their results showed that NWNS are able to generate higher-order harmonics and they demonstrated a waveform regression task using both a numerical model and a physical network. As demonstrated in Fig. 2.13, using the same MEA device with AgI NWNs, Lilak et al. were able to deliver auditory signals into the network and successfully performed the spoken digit classification task [153]. Milano et al. performed the MNIST handwritten digit classification task using a NWN simulation to generate conductance readouts which were then used in a RRAM array to train the output weights [154]. In addition to these tasks, Fu et al. utilised simulated NWNs for Mackey-Glass chaotic time series prediction [151].

Other nanomaterial networks discussed in Section 2.3 have also been employed as reservoirs in RC. For instance, Akai-Kasaya et al. used a random network of single-walled carbon nanotubes complexed with polyoxometalate as a reservoir to perform tasks like waveform reconstruction, time-series prediction and memory capacity [155]. Banerjee et al. demonstrated an object classification task with robotic tactile sensory inputs using a single-walled carbon nanotube–porphyrin polyoxometalate (Por-POM) random network [86]. In the regime of nanoparticles and thin films, Mirigliano et al. built a binary classifier using nanostructured Ag films [156] and Mallinson et al. studied the effects of topology and heterogeneity on the performance of reservoirs made of percolating networks of nanoparticles [157].

2.4.3 Performance Optimisation

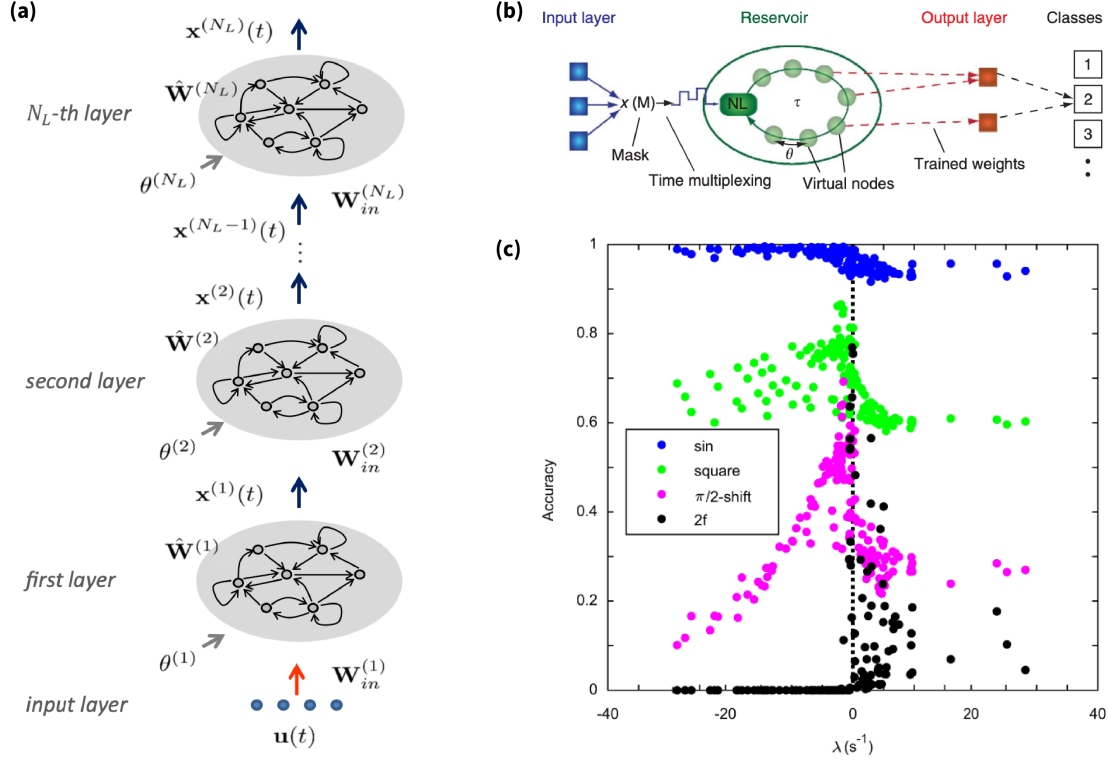


Fig. 2.14 RC performance optimisation examples. (a) RC with a deep architecture. Adapted from Gallicchio et al. [158]. (b) Time multiplexing of a single node reservoir. Adapted from Appeltant et al. [159]. (c) Task performance of a simulated NWN as a reservoir tuned to states around the edge of chaos. Adapted from Hochstetter et al. [116].

As described earlier in Section 2.4, reservoirs are particularly well-suited for a wide range of computational tasks. However, exploiting their full potential requires a dedicated effort to fine-tune the system and improve the performance [121, 125]. Various studies have explored the possibilities for enhancing the RC framework, including for memristive reservoirs [80]. Here we mainly focus on the model and configuration of the system, the training process and the internal dynamics (c.f. Fig. 2.14) for both algorithmic and physical RC.

Configuration The most intuitive approach is to utilise various connectivities for the reservoir [121, 125]. In particular, small-world [160], scale-free [161], and biological neural connectivities have captured the most attention, and various endeavours have been undertaken in this regard [121, 125, 162, 163]. Meanwhile, modifying the neuron model or dynamic model is another common approach. For instance, in the original LSM work, Maass et al. introduced a continuous time model and incorporated spiking neurons, making the

system more realistic biologically [122]. Including an leaky component in Eq. 2.1 is another common method and it enables the system to operate at different time scales [164]. Adding a feedback term in Eq. 2.1 is potentially able to shift the internal dynamics of the reservoir to enhance the linear combination for readout, but it also brings stability issues and increases the computation complexity [165]. Combining multiple reservoirs could enrich the dynamic states and thus improve task performance. Gallicchio et al. demonstrated that putting reservoirs into a deep structure amplifies the diversity of nonlinear dynamics and increases the short-term memory capacity [158, 166]. Other studies suggest that using reservoirs in parallel can be advantageous for particular tasks [167, 168].

Training For both algorithmic and physical systems, the reservoir’s responses are gathered as readouts and linear training algorithms are employed to train the readout weights [125]. In the majority of cases, least square regression and ridge regression are used to minimise the MSE loss between the training result and the target as shown in Eq. 2.3 [124, 125]. In particular, Tapson and Van Schaik suggested that a recursive least square (RLS) technique could be used to train high-dimensional systems in an online manner with fast convergence and robust results [169, 170]. Another common approach is to use time-multiplexing on readout neurons to create virtual nodes, so that extensive temporal information can be captured [159]. This approach is particularly effective in the case of insufficient readout neurons in hardware reservoirs (e.g. memristor cross-bar arrays). Furthermore, overfitting is a frequently encountered problem when training a reservoir, and cross validation is a useful strategy for determining suitable ridge parameters to mitigate the issue [125].

Dynamics As described in Section 2.4, a trade-off between a reservoir’s memory capacity and nonlinearity exists in reservoir systems and a *goldilocks zone* in between could be ideal for computing [133, 135]. These properties of reservoirs are primarily governed by their internal dynamics [125]. Various studies have referred to this regime as a phase transition (or “edge of chaos”) and have consistently reported that the best performances of reservoirs can be observed here for both algorithmic reservoirs [171–174] or hardware systems [96, 116, 117]. Multiple approaches are employed to optimise the reservoirs and find the best configuration that exploit the computing resources of the device. For example, a common step in the optimisation process involves normalizing the input and target signals to a range that is suitable for the reservoir to operate [125]. While most optimisation techniques tend to use a trial-and-error approach, it has been demonstrated that, even if reservoirs are not fully differentiable, the meta-learning (or learning-to-learn) framework can help identify

the optimal hyperparameters which allow the network to operate within an optimal regime [175–177].

2.5 Information Dynamics

Information theory was first introduced by Shannon in 1948 to provide fundamental principles and tools for understanding how information can be efficiently encoded, transmitted, and decoded [178, 179]. It has been widely applied to analyse complex self-organised systems [180], in which the emergent global behavior of the system arises non-trivially from the local interactions among individual units [181, 182]. The study of information dynamics is conducted in the spatial-temporal domain by investigating the transfer of information between individual units, its storage mechanisms within these units, as well as how information from different sources combine in these systems [183–185]. This thesis focusses on several prevalent measures such as mutual information [179], transfer entropy [186] and active information storage [184] to analyse the local computation of neuromorphic NWNs. The following section provides a brief introduction to these measures. A more detailed exposition may be found in the JIDT paper by Lizier et al. [187].

2.5.1 Information-theoretic measures

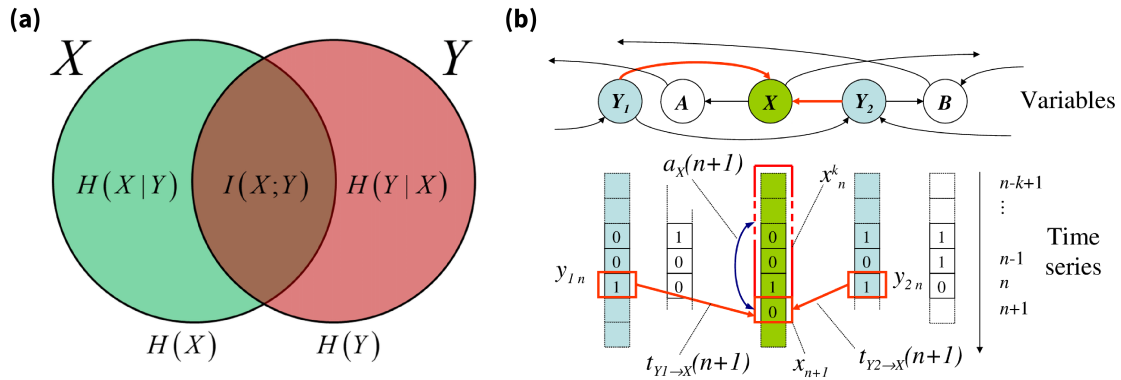


Fig. 2.15 Schematic illustration of information theory metrics. (a) Mutual information between two variables. Adapted from Uda et al. [188]. (b) Transfer entropy and active information storage. Adapted from Lizier et al. [187].

Mutual Information Mutual information (MI) quantifies the amount of information that one variable X conveys about another variable Y [179, 187]. As shown in Fig. 2.15 (a),

it measures the relationship between variables and provides insights into their statistical association. Mathematically it is defined as:

$$I(X;Y) = H(X) + H(Y) - H(X,Y), \quad (2.4)$$

$$= \sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)}, \quad (2.5)$$

where $H(\cdot)$ is the Shannon entropy and $p(\cdot)$ represents probability. It is also noteworthy that MI is symmetric and non-negative, for which $I(X;Y) = 0$ if and only if X and Y are statistically independent. It provides a way to understand the degree to which knowledge of one variable reduces uncertainty about the other.

Transfer Entropy Transfer entropy (TE) measures the amount of directed (asymmetric) transfer of information from one source time-series process $\mathbf{Y}$ to another target process $\mathbf{X}$ [186, 189]. As illustrated in Fig. 2.15 (b), it is defined as the conditional mutual information between these processes:

$$\begin{aligned} T_{Y \rightarrow X}(k,l) &= I(\mathbf{Y}_n^{(l)}; X_{n+1} | \mathbf{X}_n^{(k)}), \\ &= \left\langle \log_2 \frac{p(x_{n+1} | \mathbf{x}_n^{(k)}, \mathbf{y}_n^{(l)})}{p(x_{n+1} | \mathbf{x}_n^{(k)})} \right\rangle, \end{aligned} \quad (2.6)$$

in which n is the time index for the variables, $\mathbf{Y}_n^{(l)}$ is the variable representing the past l values of the source up to and including time n , X_{n+1} the next value of the target at time $n+1$, and $\mathbf{X}_n^{(k)}$ the past k values of the target. The lower case identifiers $(x_{n+1}, \mathbf{x}_n^{(k)}, \mathbf{y}_n^{(l)})$ are specific realisations of the variables. TE calculates the amount of information that the past of one process provides about the future of another, beyond what can be explained by their shared history.

Active Information Storage Active information storage (AIS) measures the amount of information in the past of a time-series process X that is used to predict its own future [184, 187]. Fig. 2.15 indicates that AIS of X can be defined as the mutual information

between its own past states $\mathbf{X}_n^{(k)}$ and the future state X_{n+1} :

$$A_X(k) = I(\mathbf{X}_n^{(k)}; X_{n+1}), \quad (2.7)$$

$$= \left\langle \log_2 \frac{p(\mathbf{x}_n^{(k)}, x_{n+1})}{p(\mathbf{x}_n^{(k)})p(x_{n+1})} \right\rangle. \quad (2.8)$$

It quantifies the ability of a system to encode and maintain information in a way that is relevant to the future of its activity.

2.5.2 Applications

Information theory offers powerful tools for analyzing systems in various disciplines such as canonical complex systems, computational neuroscience, cognitive science and artificial neural networks [178, 179, 187, 190]. This type of analysis has proven instrumental in investigating the intrinsic information processing, especially during the phase transition regime, of numerous canonical complex systems including the Ising model [191, 192], echo state networks (reservoirs) [173], spiking neural networks [193, 194], random Boolean networks [195, 196] and neural mass models [197]. Furthermore, in computational neuroscience, the application of information dynamics has enhanced the understanding of neural coding [198, 199], connectivity [200–203] and information flow within as well as between different brain regions [190, 204, 205]. Last but not least, information theory provides a powerful framework for delving into the intricacies of both cognitive systems [206, 207] and artificial neural networks [208, 209], offering insights into their fundamental mechanisms. Further applications can be explored in the broader academic literature [189, 210].

2.6 Motivation and Rationale for this thesis

The research presented in this thesis is motivated by the convergent limitations faced by conventional CMOS hardware and AI software: the current trend is unsustainable in terms of both energy efficiency and scalability. While much effort has been directed toward quantum computing, this approach will also be considerably resource-intensive and so does not address these critical challenges. A neuromorphic approach offers arguably the most feasible pathway to realise efficient, sustainable computing and machine intelligence. The rationale behind focussing this research on memristive nanowire networks is that these systems present several intrinsically neuromorphic properties that remained unexplored until relatively recently. This

includes their neural network-like circuit topology and rich spatiotemporal dynamics, which lend themselves directly to a physical reservoir computing approach, as explored here.

Previous studies on NWNs mainly focused on the neuromorphic properties of NWNs, including their brain-like connectivity and dynamics, as well as cognitive abilities. Additionally, there have been some preliminary explorations into RC applications for simple tasks like wave form regression, offering initial insights on their potential applications. However, there remains significant scope to delve deeper into these applications. This thesis aims to bridge this knowledge gap by presenting a thorough study of the learning capabilities of neuromorphic nanowire networks within a reservoir computing framework, pushing forward our understanding of their dynamical properties and potential for practical, real-world applications. In doing so, the ultimate goal is to advance the concept of “neuromorphic machine intelligence”, underpinned by the physics of nanomaterials and dynamical systems.

References

- [1] Silver, D. *et al.* Mastering the game of Go with deep neural networks and tree search. *Nature* **529** (7587), 484–489 (2016). doi:[10.1038/nature16961](https://doi.org/10.1038/nature16961) .
- [2] Introducing ChatGPT. <https://openai.com/blog/chatgpt>.
- [3] Turing, A. M. Mechanical intelligence. (*No Title*) .
- [4] McCarthy, J., Minsky, M. L., Rochester, N. & Shannon, C. E. A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine* **27** (4), 12–12 (2006) .
- [5] Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review* **65** (6), 386 (1958) .
- [6] Ivakhnenko, A. G. Polynomial theory of complex systems. *IEEE transactions on Systems, Man, and Cybernetics* (4), 364–378 (1971) .
- [7] Schmidhuber, J. Annotated history of modern ai and deep learning. *arXiv preprint arXiv:2212.11279* (2022) .
- [8] Lighthill, S. J. Artificial Intelligence: A General Survey (The Lighthill Report). <https://aitopics.org/doc/classics:D8235CF9> (1972).
- [9] Crevier, D. *AI: The Tumultuous History of the Search for Artificial Intelligence* (1993).
- [10] Minsky, M. & Papert, S. A. *Perceptrons: An Introduction to Computational Geometry* (The MIT Press, 2017).
- [11] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *nature* **323** (6088), 533–536 (1986) .
- [12] Cortes, C. & Vapnik, V. Support-vector networks. *Machine learning* **20**, 273–297 (1995) .
- [13] Breiman, L. Random forests. *Machine learning* **45**, 5–32 (2001) .
- [14] Oh, K.-S. & Jung, K. Gpu implementation of neural networks. *Pattern Recognition* **37** (6), 1311–1314 (2004) .

- [15] Bengio, Y. *et al.* Learning deep architectures for ai. *Foundations and trends® in Machine Learning* **2** (1), 1–127 (2009) .
- [16] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521** (7553), 436–444 (2015). doi:[10.1038/nature14539](https://doi.org/10.1038/nature14539) .
- [17] Mehonic, A. & Kenyon, A. J. Brain-inspired computing needs a master plan. *Nature* **604** (7905), 255–260 (2022). doi:[10.1038/s41586-021-04362-w](https://doi.org/10.1038/s41586-021-04362-w) .
- [18] Herculano-Houzel, S. The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in human neuroscience* **31** (2009) .
- [19] Wong, T. M. *et al.* 10¹⁴. Report No. RJ10502 (ALM1211-004) (2012). (IBM) .
- [20] Mead, C. & Ismail, M. *Analog VLSI implementation of neural systems* Vol. 80 (Springer Science & Business Media, 1989).
- [21] Mead, C. Neuromorphic electronic systems. *Proceedings of the IEEE* **78** (10), 1629–1636 (1990). doi:[10.1109/5.58356](https://doi.org/10.1109/5.58356) .
- [22] Mead, C. How we created neuromorphic engineering. *Nature Electronics* **3** (7), 434–435 (2020). doi:[10.1038/s41928-020-0448-2](https://doi.org/10.1038/s41928-020-0448-2) .
- [23] Christensen, D. V. *et al.* 2022 roadmap on neuromorphic computing and engineering. *Neuromorphic Computing and Engineering* **2** (2), 022501 (2022). doi:[10.1088/2634-4386/ac4a83](https://doi.org/10.1088/2634-4386/ac4a83) .
- [24] Indiveri, G. *et al.* Neuromorphic Silicon Neuron Circuits. *Frontiers in Neuroscience* **5** (2011). doi:[10.3389/fnins.2011.00073](https://doi.org/10.3389/fnins.2011.00073) .
- [25] Ielmini, D. & Wong, H.-S. P. In-memory computing with resistive switching devices. *Nature Electronics* **1** (6), 333–343 (2018). doi:[10.1038/s41928-018-0092-2](https://doi.org/10.1038/s41928-018-0092-2) .
- [26] Eshraghian, J. K. *et al.* Training spiking neural networks using lessons from deep learning. *Proceedings of the IEEE* (2023). doi:[10.1109/JPROC.2023.3308088](https://doi.org/10.1109/JPROC.2023.3308088) .
- [27] Schmidgall, S. *et al.* Brain-inspired learning in artificial neural networks: A review (2023). [2305.11252](https://doi.org/10.2305.11252).
- [28] Zhu, J., Zhang, T., Yang, Y. & Huang, R. A comprehensive review on emerging artificial neuromorphic devices. *Applied Physics Reviews* **7** (1), 011312 (2020). doi:[10.1063/1.5118217](https://doi.org/10.1063/1.5118217) .

- [29] Sangwan, V. K. & Hersam, M. C. Neuromorphic nanoelectronic materials. *Nature Nanotechnology* **15** (7), 517–528 (2020). doi:[10.1038/s41565-020-0647-z](https://doi.org/10.1038/s41565-020-0647-z) .
- [30] Marković, D., Mizrahi, A., Querlioz, D. & Grollier, J. Physics for neuromorphic computing. *Nature Reviews Physics* (2020). doi:[10.1038/s42254-020-0208-2](https://doi.org/10.1038/s42254-020-0208-2) .
- [31] Stern, M. & Murugan, A. Learning without neurons in physical systems. *Annual Review of Condensed Matter Physics* **14** (1), 417–441 (2023). doi:[10.1146/annurev-conmatphys-040821-113439](https://doi.org/10.1146/annurev-conmatphys-040821-113439) .
- [32] Maass, W. Networks of spiking neurons: The third generation of neural network models. *Neural Networks* **10** (9), 1659–1671 (1997). doi:[10.1016/S0893-6080\(97\)00011-7](https://doi.org/10.1016/S0893-6080(97)00011-7) .
- [33] Roy, K., Jaiswal, A. & Panda, P. Towards spike-based machine intelligence with neuromorphic computing. *Nature* **575** (7784), 607–617 (2019). doi:[10.1038/s41586-019-1677-2](https://doi.org/10.1038/s41586-019-1677-2) .
- [34] Maass, W. & Markram, H. On the computational power of circuits of spiking neurons. *Journal of Computer and System Sciences* **69** (4), 593–616 (2004). doi:[10.1016/j.jcss.2004.04.001](https://doi.org/10.1016/j.jcss.2004.04.001) .
- [35] Merolla, P. A. *et al.* A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345** (6197), 668–673 (2014). doi:[10.1126/science.1254642](https://doi.org/10.1126/science.1254642) .
- [36] Schuman, C. D. *et al.* A Survey of Neuromorphic Computing and Neural Networks in Hardware (2017). [1705.06963](https://arxiv.org/abs/1705.06963).
- [37] Bouvier, M. *et al.* Spiking Neural Networks Hardware Implementations and Challenges: A Survey. *ACM Journal on Emerging Technologies in Computing Systems* **15** (2), 22:1–22:35 (2019). doi:[10.1145/3304103](https://doi.org/10.1145/3304103) .
- [38] Zhang, W. *et al.* Neuro-inspired computing chips. *Nature electronics* **3** (7), 371–382 (2020) .
- [39] Li, G. *et al.* Brain inspired computing: A systematic survey and future trends (2023) .
- [40] Akopyan, F. *et al.* Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip. *IEEE transactions on computer-aided design of integrated circuits and systems* **34** (10), 1537–1557 (2015) .

- [41] Davies, M. *et al.* Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro* **38** (1), 82–99 (2018) .
- [42] Furber, S. B., Galluppi, F., Temple, S. & Plana, L. A. The spinnaker project. *Proceedings of the IEEE* **102** (5), 652–665 (2014) .
- [43] Schmitt, S. *et al.* Neuromorphic hardware in the loop: Training a deep spiking network on the brainscales wafer-scale system, 2227–2234 (IEEE, 2017).
- [44] Moradi, S., Qiao, N., Stefanini, F. & Indiveri, G. A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (dynaps). *IEEE transactions on biomedical circuits and systems* **12** (1), 106–122 (2017) .
- [45] Neftci, E. O., Mostafa, H. & Zenke, F. Surrogate Gradient Learning in Spiking Neural Networks (2019). [1901.09948](https://arxiv.org/abs/1901.09948).
- [46] Eshraghian, J. K., Wang, X. & Lu, W. D. Memristor-Based Binarized Spiking Neural Networks: Challenges and applications. *IEEE Nanotechnology Magazine* **16** (2), 14–23 (2022). doi:[10.1109/MNANO.2022.3141443](https://doi.org/10.1109/MNANO.2022.3141443) .
- [47] Chua, L. & Kang, S. M. Memristive devices and systems. *Proceedings of the IEEE* **64** (2), 209–223 (1976). doi:[10.1109/PROC.1976.10092](https://doi.org/10.1109/PROC.1976.10092) .
- [48] Strukov, D. B., Snider, G. S., Stewart, D. R. & Williams, R. S. The missing memristor found. *Nature* **453** (7191), 80–83 (2008). doi:[10.1038/nature06932](https://doi.org/10.1038/nature06932) .
- [49] Widrow, B. *Adaptive" Adaline" Neuron Using Chemical" Memistors."*. (1960).
- [50] Caravelli, F. & Carbajal, J. P. Memristors for the Curious Outsiders (2018). [1812.03389](https://arxiv.org/abs/1812.03389).
- [51] Zidan, M. A., Strachan, J. P. & Lu, W. D. The future of electronics based on memristive systems. *Nature Electronics* **1** (1), 22–29 (2018). doi:[10.1038/s41928-017-0006-8](https://doi.org/10.1038/s41928-017-0006-8) .
- [52] Mehonic, A. *et al.* Memristors—From In-Memory Computing, Deep Learning Acceleration, and Spiking Neural Networks to the Future of Neuromorphic and Bio-Inspired Computing. *Advanced Intelligent Systems* **2** (11), 2000085 (2020). doi:[10.1002/aisy.202000085](https://doi.org/10.1002/aisy.202000085) .

- [53] Krestinskaya, O., James, A. P. & Chua, L. O. Neuro-memristive Circuits for Edge Computing: A review. *IEEE Transactions on Neural Networks and Learning Systems* **31** (1), 4–23 (2020). doi:[10.1109/TNNLS.2019.2899262](https://doi.org/10.1109/TNNLS.2019.2899262), [arxiv:1807.00962](https://arxiv.org/abs/1807.00962) [cs].
- [54] Zhou, G. *et al.* Volatile and Nonvolatile Memristive Devices for Neuromorphic Computing. *Advanced Electronic Materials* **8** (7), 2101127 (2022). doi:[10.1002/aelm.202101127](https://doi.org/10.1002/aelm.202101127).
- [55] Waser, R., Dittmann, R., Staikov, G. & Szot, K. Redox-based resistive switching memories—nanoionic mechanisms, prospects, and challenges. *Advanced materials* **21** (25-26), 2632–2663 (2009).
- [56] Mohammad, B. *et al.* State-of-the-art of metal-oxide memristor devices. *Nanotechnology Reviews* **5** (2016). doi:[10.1515/ntrev-2015-0029](https://doi.org/10.1515/ntrev-2015-0029).
- [57] Jo, S. H. *et al.* Nanoscale Memristor Device as Synapse in Neuromorphic Systems. *Nano Letters* **10** (4), 1297–1301 (2010). doi:[10.1021/nl904092h](https://doi.org/10.1021/nl904092h).
- [58] Indiveri, G., Linares-Barranco, B., Legenstein, R., Deligeorgis, G. & Prodromakis, T. Integration of nanoscale memristor synapses in neuromorphic computing architectures. *Nanotechnology* **24** (38), 384010 (2013). doi:[10.1088/0957-4484/24/38/384010](https://doi.org/10.1088/0957-4484/24/38/384010).
- [59] Xia, Q. & Yang, J. J. Memristive crossbar arrays for brain-inspired computing. *Nature Materials* **18** (4), 309–323 (2019). doi:[10.1038/s41563-019-0291-x](https://doi.org/10.1038/s41563-019-0291-x).
- [60] Huang, A., Zhang, X., Li, R. & Chi, Y. Memristor neural network design. *Memristor and Memristive Neural Networks* 1–35 (2018).
- [61] Merrih-Bayat, F. & Shouraki, S. B. Memristive neuro-fuzzy system. *IEEE transactions on cybernetics* **43** (1), 269–285 (2012).
- [62] Borghetti, J. *et al.* ‘memristive’ switches enable ‘stateful’ logic operations via material implication. *Nature* **464** (7290), 873–876 (2010).
- [63] Vourkas, I. & Sirakoulis, G. C. Emerging memristor-based logic circuit design approaches: A review. *IEEE circuits and systems magazine* **16** (3), 15–30 (2016).
- [64] Haj-Ali, A., Ben-Hur, R., Wald, N., Ronen, R. & Kvatinsky, S. Imaging: In-memory algorithms for image processing. *IEEE Transactions on Circuits and Systems I: Regular Papers* **65** (12), 4258–4271 (2018).

- [65] Papandreou, N. *et al.* *Programming algorithms for multilevel phase-change memory*, 329–332 (IEEE, 2011).
- [66] Burr, G. W. *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) .
- [67] Joshi, V. *et al.* Accurate deep neural network inference using computational phase-change memory. *Nature communications* **11** (1), 2473 (2020) .
- [68] Yao, P. *et al.* Fully hardware-implemented memristor convolutional neural network. *Nature* **577** (7792), 641–646 (2020) .
- [69] Boybat, I. *et al.* Neuromorphic computing with multi-memristive synapses. *Nature communications* **9** (1), 2514 (2018) .
- [70] Serb, A. *et al.* Unsupervised learning in probabilistic neural networks with multi-state metal-oxide memristive synapses. *Nature communications* **7** (1), 12611 (2016) .
- [71] Nandakumar, S. *et al.* Experimental demonstration of supervised learning in spiking neural networks with phase-change memory synapses. *Scientific reports* **10** (1), 8080 (2020) .
- [72] Du, C. *et al.* Reservoir computing using dynamic memristors for temporal information processing. *Nature Communications* **8** (1), 2204 (2017). doi:[10.1038/s41467-017-02337-y](https://doi.org/10.1038/s41467-017-02337-y) .
- [73] Midya, R. *et al.* Reservoir Computing Using Diffusive Memristors. *Advanced Intelligent Systems* **1** (7), 1900084 (2019). doi:[10.1002/aisy.201900084](https://doi.org/10.1002/aisy.201900084) .
- [74] Zhong, Y. *et al.* Dynamic memristor-based reservoir computing for high-efficiency temporal signal processing. *Nature Communications* **12** (1), 408 (2021). doi:[10.1038/s41467-020-20692-1](https://doi.org/10.1038/s41467-020-20692-1) .
- [75] Zhong, Y. *et al.* A memristor-based analogue reservoir computing system for real-time and power-efficient signal processing. *Nature Electronics* **5** (10), 672–681 (2022). doi:[10.1038/s41928-022-00838-3](https://doi.org/10.1038/s41928-022-00838-3) .
- [76] Kuncic, Z., Nakayama, T. & Gimzewski, J. Editorial: Focus on disordered, self-assembled neuromorphic systems. *Neuromorphic Computing and Engineering* **2** (4), 040201 (2022). doi:[10.1088/2634-4386/ac91a0](https://doi.org/10.1088/2634-4386/ac91a0) .

- [77] Collins, P. G., Zettl, A., Bando, H., Thess, A. & Smalley, R. Nanotube nanodevice. *Science* **278** (5335), 100–102 (1997) .
- [78] Tans, S. J., Verschueren, A. R. & Dekker, C. Room-temperature transistor based on a single carbon nanotube. *Nature* **393** (6680), 49–52 (1998) .
- [79] Baughman, R. H., Zakhidov, A. A. & De Heer, W. A. Carbon nanotubes—the route toward applications. *science* **297** (5582), 787–792 (2002) .
- [80] Tanaka, H. *et al.* In-materio computing in random networks of carbon nanotubes complexed with chemically dynamic molecules: A review. *Neuromorphic Computing and Engineering* **2** (2), 022002 (2022). doi:[10.1088/2634-4386/ac676a](https://doi.org/10.1088/2634-4386/ac676a) .
- [81] Kim, K., Chen, C.-L., Truong, Q., Shen, A. M. & Chen, Y. A carbon nanotube synapse with dynamic logic and learning. *Advanced Materials* **25** (12), 1693–1698 (2013) .
- [82] Shen, A. M. *et al.* Analog neuromorphic module based on carbon nanotube synapses. *ACS nano* **7** (7), 6117–6122 (2013) .
- [83] Zhao, W. *et al.* Nanotube devices based crossbar architecture: toward neuromorphic computing. *Nanotechnology* **21** (17), 175202 (2010) .
- [84] Snow, E., Novak, J., Campbell, P. & Park, D. Random networks of carbon nanotubes as an electronic material. *Applied Physics Letters* **82** (13), 2145–2147 (2003) .
- [85] Tanaka, H. *et al.* A molecular neuromorphic network device consisting of single-walled carbon nanotubes complexed with polyoxometalate. *Nature Communications* **9** (1), 2693 (2018). doi:[10.1038/s41467-018-04886-2](https://doi.org/10.1038/s41467-018-04886-2) .
- [86] Banerjee, D. *et al.* Emergence of in-materio intelligence from an incidental structure of a single-walled carbon nanotube–porphyrin polyoxometalate random network. *Advanced Intelligent Systems* **4** (4), 2100145 (2022) .
- [87] Vahl, A. *et al.* Single target sputter deposition of alloy nanoparticles with adjustable composition via a gas aggregation cluster source. *Nanotechnology* **28** (17), 175703 (2017) .
- [88] Vahl, A., Carstens, N., Strunskus, T., Faupel, F. & Hassanien, A. Diffusive memristive switching on the nanoscale, from individual nanoparticles towards scalable nanocomposite devices. *Scientific reports* **9** (1), 17367 (2019) .

- [89] Carstens, N., Strunskus, T., Faupel, F., Hassanien, A. & Vahl, A. Neuronal-like irregular spiking dynamics in highly volatile memristive intermediate-scale agpt-nanoparticle assemblies. *Particle & Particle Systems Characterization* **40** (3), 2200131 (2023) .
- [90] Mirigliano, M. *et al.* Non-ohmic behavior and resistive switching of au cluster-assembled films beyond the percolation threshold. *Nanoscale Advances* **1** (8), 3119–3130 (2019) .
- [91] Mirigliano, M. *et al.* Anomalous electrical conduction and negative temperature coefficient of resistance in nanostructured gold resistive switching films. *Scientific Reports* **10** (1), 19613 (2020) .
- [92] Mirigliano, M. *et al.* Complex electrical spiking activity in resistive switching nanostructured au two-terminal devices. *Nanotechnology* **31** (23), 234001 (2020) .
- [93] Sattar, A., Fostner, S. & Brown, S. A. Quantized conductance and switching in percolating nanoparticle films. *Physical review letters* **111** (13), 136808 (2013) .
- [94] Fostner, S. & Brown, S. A. Neuromorphic behavior in percolating nanoparticle films. *Phys. Rev. E* **92**, 052134 (2015). doi:[10.1103/PhysRevE.92.052134](https://doi.org/10.1103/PhysRevE.92.052134) .
- [95] Bose, S. K., Mallinson, J. B., Gazoni, R. M. & Brown, S. A. Stable self-assembled atomic-switch networks for neuromorphic applications. *IEEE Transactions on Electron Devices* **64** (12), 5194–5201 (2017). doi:[10.1109/TED.2017.2766063](https://doi.org/10.1109/TED.2017.2766063) .
- [96] Mallinson, J. B. *et al.* Avalanches and criticality in self-organized nanoscale networks. *Science Advances* **5** (11), eaaw8438 (2019). doi:[10.1126/sciadv.aaw8438](https://doi.org/10.1126/sciadv.aaw8438) .
- [97] Shirai, S. *et al.* Long-range temporal correlations in scale-free neuromorphic networks. *Network Neuroscience* **4** (2), 432–447 (2020) .
- [98] Pike, M. D. *et al.* Atomic scale dynamics drive brain-like avalanches in percolating nanostructured networks. *Nano letters* **20** (5), 3935–3942 (2020) .
- [99] Milano, G., Porro, S., Valov, I. & Ricciardi, C. Recent Developments and Perspectives for Memristive Devices Based on Metal Oxide Nanowires. *Advanced Electronic Materials* **5** (9), 1800909 (2019). doi:[10.1002/aelm.201800909](https://doi.org/10.1002/aelm.201800909) .

- [100] Kuncic, Z. & Nakayama, T. Neuromorphic nanowire networks: Principles, progress and future prospects for neuro-inspired information processing. *Advances in Physics: X* **6** (1), 1894234 (2021). doi:[10.1080/23746149.2021.1894234](https://doi.org/10.1080/23746149.2021.1894234) .
- [101] Stieg, A. Z. *et al.* Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials* **24** (2), 286–293 (2012). doi:[10.1002/adma.201103053](https://doi.org/10.1002/adma.201103053) .
- [102] Demis, E. C. *et al.* Atomic switch networks—nanoarchitectonic design of a complex system for natural computing. *Nanotechnology* **26** (20), 204003 (2015). doi:[10.1088/0957-4484/26/20/204003](https://doi.org/10.1088/0957-4484/26/20/204003) .
- [103] Terabe, K., Nakayama, T., Hasegawa, T. & Aono, M. Formation and disappearance of a nanoscale silver cluster realized by solid electrochemical reaction. *Journal of Applied Physics* **91** (12), 10110–10114 (2002). doi:[10.1063/1.1481775](https://doi.org/10.1063/1.1481775) .
- [104] Terabe, K., Hasegawa, T., Nakayama, T. & Aono, M. Quantized conductance atomic switch. *Nature* **433** (7021), 47–50 (2005). doi:[10.1038/nature03190](https://doi.org/10.1038/nature03190) .
- [105] Hasegawa, T. *et al.* Learning Abilities Achieved by a Single Solid-State Atomic Switch. *Advanced Materials* **22** (16), 1831–1834 (2010). doi:[10.1002/adma.200903680](https://doi.org/10.1002/adma.200903680) .
- [106] Ohno, T. *et al.* Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials* **10** (8), 591–595 (2011). doi:[10.1038/nmat3054](https://doi.org/10.1038/nmat3054) .
- [107] Milano, G., Cultrera, A., Boarino, L., Callegaro, L. & Ricciardi, C. Tomography of memory engrams in self-organizing nanowire connectomes. *Nature Communications* **14** (1), 5723 (2023). doi:[10.1038/s41467-023-40939-x](https://doi.org/10.1038/s41467-023-40939-x) .
- [108] Manning, H. G. *et al.* Emergence of winner-takes-all connectivity paths in random nanowire networks. *Nature Communications* **9** (1), 3219 (2018). doi:[10.1038/s41467-018-05517-6](https://doi.org/10.1038/s41467-018-05517-6) .
- [109] Pantone, R. D., Kendall, J. D. & Nino, J. C. Memristive nanowires exhibit small-world connectivity. *Neural Networks* **106**, 144–151 (2018). doi:[10.1016/j.neunet.2018.07.002](https://doi.org/10.1016/j.neunet.2018.07.002) .
- [110] Loeffler, A. *et al.* Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* **14**, 184 (2020). doi:[10.3389/fnins.2020.00184](https://doi.org/10.3389/fnins.2020.00184) .

- [111] Loeffler, A. *et al.* Modularity and multitasking in neuro-memristive reservoir networks. *Neuromorphic Computing and Engineering* **1** (1), 014003 (2021). doi:[10.1088/2634-4386/ac156f](https://doi.org/10.1088/2634-4386/ac156f) .
- [112] Bellew, A. T., Manning, H. G., Gomes da Rocha, C., Ferreira, M. S. & Boland, J. J. Resistance of Single Ag Nanowire Junctions and Their Role in the Conductivity of Nanowire Networks. *ACS Nano* **9** (11), 11422–11429 (2015). doi:[10.1021/acsnano.5b05469](https://doi.org/10.1021/acsnano.5b05469) .
- [113] O’Callaghan, C. *et al.* Collective capacitive and memristive responses in random nanowire networks: Emergence of critical connectivity pathways. *Journal of Applied Physics* **124** (15), 152118 (2018). doi:[10.1063/1.5037817](https://doi.org/10.1063/1.5037817) .
- [114] Bellew, A. T., Bell, A. P., McCarthy, E. K., Fairfield, J. A. & Boland, J. J. Programmability of nanowire networks. *Nanoscale* **6** (16), 9632–9639 (2014). doi:[10.1039/C4NR02338B](https://doi.org/10.1039/C4NR02338B) .
- [115] Diaz-Alvarez, A. *et al.* Emergent dynamics of neuromorphic nanowire networks. *Scientific Reports* **9** (1), 14920 (2019). doi:[10.1038/s41598-019-51330-6](https://doi.org/10.1038/s41598-019-51330-6) .
- [116] Hochstetter, J. *et al.* Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nature Communications* **12** (1), 4008 (2021). doi:[10.1038/s41467-021-24260-z](https://doi.org/10.1038/s41467-021-24260-z) .
- [117] Dunham, C. S. *et al.* Nanoscale neuromorphic networks and criticality: a perspective. *Journal of Physics: Complexity* **2** (4), 042001 (2021). doi:[10.1088/2632-072X/ac3ad3](https://doi.org/10.1088/2632-072X/ac3ad3) .
- [118] Shew, W. L. & Plenz, D. The functional benefits of criticality in the cortex. *The neuroscientist* **19** (1), 88–100 (2013) .
- [119] Schrauwen, B. An overview of reservoir computing: Theory, applications and implementations (2007) .
- [120] Verstraeten, D., Schrauwen, B., D’Haene, M. & Stroobandt, D. An experimental unification of reservoir computing methods. *Neural Networks* **20** (3), 391–403 (2007). doi:[10.1016/j.neunet.2007.04.003](https://doi.org/10.1016/j.neunet.2007.04.003) .
- [121] Lukoševičius, M. & Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review* **3** (3), 127–149 (2009). doi:[10.1016/j.cosrev.2009.03.005](https://doi.org/10.1016/j.cosrev.2009.03.005) .

- [122] Maass, W., Natschläger, T. & Markram, H. Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Computation* **14** (11), 2531–2560 (2002). doi:[10.1162/089976602760407955](https://doi.org/10.1162/089976602760407955) .
- [123] Jaeger, H. The “echo state” approach to analysing and training recurrent neural networks – with an Erratum note 47 (2001) .
- [124] Jaeger, H. A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach (2002) .
- [125] Cucchi, M., Abreu, S., Ciccone, G., Brunner, D. & Kleemann, H. Hands-on reservoir computing: A tutorial for practical implementation. *Neuromorphic Computing and Engineering* **2** (3), 032002 (2022). doi:[10.1088/2634-4386/ac7db7](https://doi.org/10.1088/2634-4386/ac7db7) .
- [126] Legenstein, R. & Maass, W. What makes a dynamical system computationally powerful? 31 (2004) .
- [127] Hermans, M. & Schrauwen, B. Recurrent kernel machines: Computing with infinite echo state networks. *Neural Computation* **24** (1), 104–133 (2012) .
- [128] Jaeger, H. Short Term Memory in Echo State Networks 60 .
- [129] Jaeger, H. Long Short-Term Memory in Echo State Networks: Details of a Simulation Study 29 (2012) .
- [130] Takens, F. in *Detecting strange attractors in turbulence* (eds Rand, D. & Young, L.-S.) *Dynamical Systems and Turbulence, Warwick 1980*, Vol. 898 366–381 (Springer Berlin Heidelberg, Berlin, Heidelberg, 1981).
- [131] Farkaš, I., Bosák, R. & Gergel', P. Computational analysis of memory capacity in echo state networks. *Neural Networks* **83**, 109–120 (2016). doi:[10.1016/j.neunet.2016.07.012](https://doi.org/10.1016/j.neunet.2016.07.012) .
- [132] Verstraeten, D., Dambre, J., Dutoit, X. & Schrauwen, B. *Memory versus non-linearity in reservoirs*, 1–8 (2010).
- [133] Dambre, J., Verstraeten, D., Schrauwen, B. & Massar, S. Information Processing Capacity of Dynamical Systems. *Scientific Reports* **2** (1), 514 (2012). doi:[10.1038/srep00514](https://doi.org/10.1038/srep00514) .

- [134] Butcher, J., Verstraeten, D., Schrauwen, B., Day, C. & Haycock, P. Reservoir computing and extreme learning machines for non-linear time-series data analysis. *Neural Networks* **38**, 76–89 (2013). doi:[10.1016/j.neunet.2012.11.011](https://doi.org/10.1016/j.neunet.2012.11.011) .
- [135] Inubushi, M. & Yoshimura, K. Reservoir Computing Beyond Memory-Nonlinearity Trade-off. *Scientific Reports* **7** (1), 10199 (2017). doi:[10.1038/s41598-017-10257-6](https://doi.org/10.1038/s41598-017-10257-6) .
- [136] Wyffels, F. & Schrauwen, B. A comparative study of reservoir computing strategies for monthly time series prediction. *Neurocomputing* **73** (10-12), 1958–1964 (2010) .
- [137] Shahi, S., Fenton, F. H. & Cherry, E. M. Prediction of chaotic time series using recurrent neural networks and reservoir computing techniques: A comparative study. *Machine learning with applications* **8**, 100300 (2022) .
- [138] Thorne, B., Jüngling, T., Small, M., Corrêa, D. & Zaitouny, A. Reservoir time series analysis: Using the response of complex dynamical systems as a universal indicator of change. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **32** (3), 033109 (2022). doi:[10.1063/5.0082122](https://doi.org/10.1063/5.0082122) .
- [139] Skowronski, M. D. & Harris, J. G. Automatic speech recognition using a predictive echo state network classifier. *Neural networks* **20** (3), 414–423 (2007) .
- [140] Xu, D., Lan, J. & Principe, J. C. *Direct adaptive control: an echo state network and genetic algorithm approach*, Vol. 3, 1483–1486 (IEEE, 2005).
- [141] Burgsteiner, H. *Training networks of biological realistic spiking neurons for real-time robot control*, 129–136 (2005).
- [142] Fernando, C. & Sojakka, S. Banzhaf, W., Ziegler, J., Christaller, T., Dittrich, P. & Kim, J. T. (eds) *Pattern Recognition in a Bucket*. (eds Banzhaf, W., Ziegler, J., Christaller, T., Dittrich, P. & Kim, J. T.) *Advances in Artificial Life*, Lecture Notes in Computer Science, 588–597 (Springer, Berlin, Heidelberg, 2003).
- [143] Tanaka, G. *et al.* Recent advances in physical reservoir computing: A review. *Neural Networks* **115**, 100–123 (2019). doi:[10.1016/j.neunet.2019.03.005](https://doi.org/10.1016/j.neunet.2019.03.005) .
- [144] Schürmann, F., Meier, K. & Schemmel, J. *Edge of Chaos Computation in Mixed-Mode VLSI - A Hard Liquid*, Vol. 17 (MIT Press, 2004).
- [145] Petre, P. & Cruz-Albrecht, J. *Neuromorphic mixed-signal circuitry for Asynchronous Pulse Processing*, 1–4 (2016).

- [146] Moon, J. *et al.* Temporal data classification and forecasting using a memristor-based reservoir computing system. *Nature Electronics* **2** (10), 480–487 (2019). doi:[10.1038/s41928-019-0313-3](https://doi.org/10.1038/s41928-019-0313-3) .
- [147] Torrejon, J. *et al.* Neuromorphic computing with nanoscale spintronic oscillators. *Nature* **547** (7664), 428–431 (2017). doi:[10.1038/nature23011](https://doi.org/10.1038/nature23011) .
- [148] Li, Q. *et al.* Dynamic Electrical Pathway Tuning in Neuromorphic Nanowire Networks. *Advanced Functional Materials* **30** (43), 2003679 (2020). doi:[10.1002/adfm.202003679](https://doi.org/10.1002/adfm.202003679) .
- [149] Diaz-Alvarez, A., Higuchi, R., Li, Q., Shingaya, Y. & Nakayama, T. Associative routing through neuromorphic nanowire networks. *AIP Advances* **10** (2), 025134 (2020). doi:[10.1063/1.5140579](https://doi.org/10.1063/1.5140579) .
- [150] Loeffler, A. *et al.* Neuromorphic learning, working memory, and metaplasticity in nanowire networks. *SCIENCE ADVANCES* (2023). doi:[10.1126/sciadv.adg3289](https://doi.org/10.1126/sciadv.adg3289) .
- [151] Fu, K. *et al.* *Reservoir Computing with Neuromemristive Nanowire Networks*, 1–8 (2020).
- [152] Sillin, H. O. *et al.* A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology* **24** (38), 384004 (2013). doi:[10.1088/0957-4484/24/38/384004](https://doi.org/10.1088/0957-4484/24/38/384004) .
- [153] Lilak, S. *et al.* Spoken Digit Classification by In-Materio Reservoir Computing With Neuromorphic Atomic Switch Networks. *Frontiers in Nanotechnology* **3** (2021). doi:[10.3389/fnano.2021.675792](https://doi.org/10.3389/fnano.2021.675792) .
- [154] Milano, G. *et al.* In materia reservoir computing with a fully memristive architecture based on self-organizing nanowire networks. *Nature Materials* (2021). doi:[10.1038/s41563-021-01099-9](https://doi.org/10.1038/s41563-021-01099-9) .
- [155] Akai-Kasaya, M. *et al.* Performance of reservoir computing in a random network of single-walled carbon nanotubes complexed with polyoxometalate. *Neuromorphic Computing and Engineering* **2** (1), 014003 (2022) .
- [156] Mirigliano, M. *et al.* A binary classifier based on a reconfigurable dense network of metallic nanojunctions. *Neuromorphic Computing and Engineering* **1** (2), 024007 (2021) .

- [157] Mallinson, J. *et al.* Reservoir computing using networks of memristors: effects of topology and heterogeneity. *Nanoscale* **15** (22), 9663–9674 (2023) .
- [158] Gallicchio, C. & Micheli, A. Echo State Property of Deep Reservoir Computing Networks. *Cognitive Computation* **9** (3), 337–350 (2017). doi:[10.1007/s12559-017-9461-9](https://doi.org/10.1007/s12559-017-9461-9) .
- [159] Appeltant, L. *et al.* Information processing using a single dynamical node as complex system. *Nature Communications* **2** (1), 468 (2011). doi:[10.1038/ncomms1476](https://doi.org/10.1038/ncomms1476) .
- [160] Watts, D. J. & Strogatz, S. H. Collective dynamics of ‘small-world’ networks. *Nature* **393** (6684), 440–442 (1998). doi:[10.1038/30918](https://doi.org/10.1038/30918) .
- [161] Barabasi, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286** (5439), 509–512 (1999). doi:[10.1126/science.286.5439.509](https://doi.org/10.1126/science.286.5439.509), [arxiv:cond-mat/9910332](https://arxiv.org/abs/cond-mat/9910332) .
- [162] Suárez, L. E., Richards, B. A., Lajoie, G. & Misić, B. Learning function from structure in neuromorphic networks. *Nature Machine Intelligence* **3** (9), 771–786 (2021). doi:[10.1038/s42256-021-00376-1](https://doi.org/10.1038/s42256-021-00376-1) .
- [163] Suárez, L. E. *et al.* Conn2res: A toolbox for connectome-based reservoir computing . doi:[10.1101/2023.05.31.543092](https://doi.org/10.1101/2023.05.31.543092) .
- [164] Jaeger, H., Lukoševičius, M., Popovici, D. & Siewert, U. Optimization and applications of echo state networks with leaky- integrator neurons. *Neural Networks* **20** (3), 335–352 (2007). doi:[10.1016/j.neunet.2007.04.016](https://doi.org/10.1016/j.neunet.2007.04.016) .
- [165] Lukoševičius, M. Echo State Networks with Trained Feedbacks .
- [166] Gallicchio, C., Micheli, A. & Pedrelli, L. Deep reservoir computing: A critical experimental analysis. *Neurocomputing* **268**, 87–99 (2017). doi:[10.1016/j.neucom.2016.12.089](https://doi.org/10.1016/j.neucom.2016.12.089) .
- [167] Alomar, M. L. *et al.* Efficient parallel implementation of reservoir computing systems. *Neural Computing and Applications* **32** (7), 2299–2313 (2020). doi:[10.1007/s00521-018-3912-4](https://doi.org/10.1007/s00521-018-3912-4) .
- [168] Wang, M. & Fu, Z. A new method of nonlinear causality detection: Reservoir computing Granger causality. *Chaos, Solitons & Fractals* **154**, 111675 (2022). doi:[10.1016/j.chaos.2021.111675](https://doi.org/10.1016/j.chaos.2021.111675) .

- [169] Tapson, J. & van Schaik, A. Learning the pseudoinverse solution to network weights. *Neural Networks* **45**, 94–100 (2013). doi:[10.1016/j.neunet.2013.02.008](https://doi.org/10.1016/j.neunet.2013.02.008) .
- [170] van Schaik, A. & Tapson, J. Online and adaptive pseudoinverse solutions for ELM weights. *Neurocomputing* **149**, 233–238 (2015). doi:[10.1016/j.neucom.2014.01.071](https://doi.org/10.1016/j.neucom.2014.01.071) .
- [171] Bertschinger, N. & Natschläger, T. Real-Time Computation at the Edge of Chaos in Recurrent Neural Networks. *Neural Computation* **16** (7), 1413–1436 (2004). doi:[10.1162/089976604323057443](https://doi.org/10.1162/089976604323057443) .
- [172] Legenstein, R. & Maass, W. Edge of chaos and prediction of computational performance for neural circuit models. *Neural Networks* **20** (3), 323–334 (2007). doi:[10.1016/j.neunet.2007.04.017](https://doi.org/10.1016/j.neunet.2007.04.017) .
- [173] Boedecker, J., Obst, O., Lizier, J. T., Mayer, N. M. & Asada, M. Information processing in echo state networks at the edge of chaos. *Theory in Biosciences* **131** (3), 205–213 (2012). doi:[10.1007/s12064-011-0146-8](https://doi.org/10.1007/s12064-011-0146-8) .
- [174] Matsuki, T. & Shibata, K. Adaptive balancing of exploration and exploitation around the edge of chaos in internal-chaos-based learning. *Neural Networks* **132**, 19–29 (2020). doi:[10.1016/j.neunet.2020.08.002](https://doi.org/10.1016/j.neunet.2020.08.002) .
- [175] Bellec, G., Salaj, D., Subramoney, A., Legenstein, R. & Maass, W. Long short-term memory and learning-to-learn in networks of spiking neurons (2018). [1803.09574](https://arxiv.org/abs/1803.09574).
- [176] Bohnstingl, T., Scherr, F., Pehle, C., Meier, K. & Maass, W. Neuromorphic Hardware Learns to Learn. *Frontiers in Neuroscience* **13** (2019) .
- [177] Subramoney, A., Scherr, F. & Maass, W. in *Reservoirs learn to learn* 59–76 (2021). [1909.07486](https://arxiv.org/abs/1909.07486).
- [178] Cover, T. M. & Thomas, J. A. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)* (Wiley-Interscience, USA, 2006).
- [179] MacKay, D. J. *Information Theory, Inference and Learning Algorithms* (Cambridge university press, 2003).
- [180] Prokopenko, M., Boschetti, F. & Ryan, A. J. An information-theoretic primer on complexity, self-organization, and emergence. *Complexity* **15** (1), 11–28 (2009) .

- [181] Bar-Yam, Y. General Features of Complex Systems. *KNOWLEDGE MANAGEMENT* (2002) .
- [182] Mitchell, M. *Complexity: A Guided Tour* (Oxford university press, 2009).
- [183] Lizier, J. T. *The Local Information Dynamics of Distributed Computation in Complex Systems* (Springer Science & Business Media, 2012).
- [184] Lizier, J. T., Prokopenko, M. & Zomaya, A. Y. Local measures of information storage in complex distributed computation. *Information Sciences* **208**, 39–54 (2012). doi:[10.1016/j.ins.2012.04.016](https://doi.org/10.1016/j.ins.2012.04.016) .
- [185] Lizier, J. T. in *Measuring the dynamics of information processing on a local scale in time and space* 161–193 (Springer, 2014).
- [186] Schreiber, T. Measuring Information Transfer. *Physical Review Letters* **85** (2), 461–464 (2000). doi:[10.1103/PhysRevLett.85.461](https://doi.org/10.1103/PhysRevLett.85.461) .
- [187] Lizier, J. T. JIDT: An Information-Theoretic Toolkit for Studying the Dynamics of Complex Systems. *Frontiers in Robotics and AI* **1** (2014). doi:[10.3389/frobt.2014.00011](https://doi.org/10.3389/frobt.2014.00011) .
- [188] Uda, S. Application of information theory in systems biology. *Biophysical reviews* **12** (2), 377–384 (2020) .
- [189] Bossomaier, T., Barnett, L., Harré, M. & Lizier, J. T. *An Introduction to Transfer Entropy: Information Flow in Complex Systems* (Springer International Publishing, 2016).
- [190] Wibral, M., Vicente, R. & Lizier, J. T. *Directed information measures in neuroscience* Vol. 724 (Springer, 2014).
- [191] Barnett, L., Lizier, J. T., Harré, M., Seth, A. K. & Bossomaier, T. Information Flow in a Kinetic Ising Model Peaks in the Disordered Phase. *Physical Review Letters* **111** (17), 177203 (2013). doi:[10.1103/PhysRevLett.111.177203](https://doi.org/10.1103/PhysRevLett.111.177203) .
- [192] Marinazzo, D., Angelini, L., Pellicoro, M. & Stramaglia, S. Synergy as a warning sign of transitions: The case of the two-dimensional ising model. *Physical Review E* **99** (4), 040101 (2019) .

- [193] Mediano, P. A. & Shanahan, M. Balanced information storage and transfer in modular spiking neural networks. *arXiv preprint arXiv:1708.04392* (2017) .
- [194] Cramer, B. *et al.* Control of criticality and computation in spiking neuromorphic networks with plasticity. *Nature Communications* **11** (1), 2853 (2020). doi:[10.1038/s41467-020-16548-3](https://doi.org/10.1038/s41467-020-16548-3) .
- [195] Lizier, J., Prokopenko, M. & Zomaya, A. The Information Dynamics of Phase Transitions in Random Boolean Networks. *Proc. Eleventh International Conference on the Simulation and Synthesis of Living Systems (ALife XI)* (2008) .
- [196] Lizier, J. T., Pritam, S. & Prokopenko, M. Information Dynamics in Small-World Boolean Networks. *Artificial Life* **17** (4), 293–314 (2011). doi:[10.1162/artl_a_00040](https://doi.org/10.1162/artl_a_00040) .
- [197] Li, M. *et al.* Transitions in information processing dynamics at the whole-brain network level are driven by alterations in neural gain. *PLoS computational biology* **15** (10), e1006957 (2019) .
- [198] Borst, A. & Theunissen, F. E. Information theory and neural coding. *Nature neuroscience* **2** (11), 947–957 (1999) .
- [199] Quiñero, R. & Panzeri, S. Extracting information from neuronal populations: information theory and decoding approaches. *Nature Reviews Neuroscience* **10** (3), 173–185 (2009) .
- [200] Honey, C. J., Kötter, R., Breakspear, M. & Sporns, O. Network structure of cerebral cortex shapes functional connectivity on multiple time scales. *Proceedings of the National Academy of Sciences* **104** (24), 10240–10245 (2007) .
- [201] Vicente, R., Wibral, M., Lindner, M. & Pipa, G. Transfer entropy—a model-free measure of effective connectivity for the neurosciences. *Journal of computational neuroscience* **30** (1), 45–67 (2011) .
- [202] Lizier, J. T., Heinzle, J., Horstmann, A., Haynes, J.-D. & Prokopenko, M. Multivariate information-theoretic measures reveal directed information structure and task relevant changes in fmri connectivity. *Journal of computational neuroscience* **30**, 85–107 (2011) .
- [203] Novelli, L. & Lizier, J. T. Inferring network properties from time series using transfer entropy and mutual information: Validation of multivariate versus bivariate approaches. *Network Neuroscience* **5** (2), 373–404 (2021). doi:[10.1162/netn_a_00178](https://doi.org/10.1162/netn_a_00178) .

-
- [204] Lungarella, M. & Sporns, O. Mapping information flow in sensorimotor networks. *PLoS computational biology* **2** (10), e144 (2006) .
- [205] Rabinovich, M. I., Afraimovich, V. S., Bick, C. & Varona, P. Information flow dynamics in the brain. *Physics of life reviews* **9** (1), 51–73 (2012) .
- [206] Bialek, W., Nemenman, I. & Tishby, N. Predictability, complexity, and learning. *Neural computation* **13** (11), 2409–2463 (2001) .
- [207] Tishby, N. & Polani, D. in *Information theory of decisions and actions* 601–636 (Springer, 2010).
- [208] Shwartz-Ziv, R. & Tishby, N. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810* (2017) .
- [209] Saxe, A. M. *et al.* On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment* **2019** (12), 124020 (2019) .
- [210] Timme, N. M. & Lapish, C. A tutorial for information theory in neuroscience. *eneuro* **5** (3) (2018) .

Chapter 3

Information Dynamics in Neuromorphic Nanowire Networks

*“Information is a source of learning.
But unless it is organized, processed,
and available to the right people in a
format for decision making, it is a
burden, not a benefit.”*

William Pollard

This chapter is published as:

Zhu, R., Hochstetter, J., Loeffler, A., Diaz-Alvarez, A., Nakayama, T., Lizier, J.T., Kuncic, Z., 2021. Information dynamics in neuromorphic nanowire networks. Scientific Reports 11, 13047. <https://doi.org/10.1038/s41598-021-92170-7>

Abstract

Neuromorphic systems comprised of self-assembled nanowires exhibit a range of neural-like dynamics arising from the interplay of their synapse-like electrical junctions and their complex network topology. Additionally, various information processing tasks have been demonstrated with neuromorphic nanowire networks. Here, we investigate the dynamics of how these unique systems process information through information-theoretic metrics. In particular, Transfer Entropy (TE) and Active Information Storage (AIS) are employed to investigate dynamical information flow and short-term memory in nanowire networks. In addition to finding that the topologically central parts of networks contribute the most to the information flow, our results also reveal TE and AIS are maximised when the networks transitions from a quiescent to an active state. The performance of neuromorphic networks in memory and learning tasks is demonstrated to be dependent on their internal dynamical states as well as topological structure. Optimal performance is found when these networks are pre-initialised to the transition state where TE and AIS are maximal. Furthermore, an optimal range of information processing resources (i.e. connectivity density) is identified for performance. Overall, our results demonstrate information dynamics is a valuable tool to study and benchmark neuromorphic systems.

3.1 Introduction

The brain is recognised as a powerful and efficient information processing system [1, 2]. Its network structure has inspired a plethora of artificial neural network algorithms now widely used in machine learning [3, 4]. Notwithstanding the success of these neuro-inspired learning approaches, the physical nature of real neurons and synapses has also inspired hardware-based approaches to replicating the unique information processing capabilities of biological nervous systems [5]. A key milestone in this endeavour was the dramatic reduction in power consumption achieved by integrating memory and processing in so-called neuromorphic chips [6]. This has been exploited to improve the efficiency of neural network training, a computationally intensive task. Other neuromorphic computing approaches have successfully demonstrated neuron spike-based learning implemented in CMOS hardware [7–9] and synapse-based learning achieved using novel post-CMOS device components with resistive memory (memristive) switching properties [10–15].

Neuromorphic information processing capabilities have also been demonstrated in low-dimensional nanomaterials constructed by bottom-up methods, e.g. quantum dots [16], carbon nanotubes [17], nanoparticles [18] - see Sangwan and Hersham (2020) [19] for a comprehensive review. These represent a unique class of neuromorphic systems, as bio-inspired self-assembly can produce highly disordered structures with emergent collective computational abilities [20, 21]. This draws strong similarities to complex systems approaches used to study the brain's network dynamics [22]. The present study focuses on neuromorphic Nanowire Networks (NWNs), comprised of metallic nanowires that self-assemble to form a complex network topology, with memristive cross-point junctions [23]. Previous studies have shown that NWNs exhibit complex, neuromorphic structural properties such as small-worldness, modularity and recurrent feedback loops [24]. In response to electrical stimulation, NWNs exhibit emergent collective neural-like dynamics [25–31], including a “winner-takes-all” (WTA) electrical transport path [18, 27–30, 32, 33]. This feature emerges above a threshold voltage when connected edge junctions collectively switch to a low resistance state, thereby activating the network into a highly conducting state. Theoretical studies suggest that a WTA gate module in recurrent neural networks may be optimal for computation [34]. Furthermore, activation of disordered memristive networks may be concomitant with a phase transition [35], which is often associated with optimal information processing capabilities at the edge-of-chaos [36, 37]. Simulation studies [38–41] and experimental measurements [25, 26] using a multi-electrode array device have demonstrated neuro-inspired learning using a reservoir computing approach. In that approach, only the (memoryless) readout layer

is trained, while the network dynamically self-adjusts its synapse-like memristive junctions in response to changing input signals.

This study aims to gain deeper insights into the neuromorphic information processing capabilities of NWNs by quantifying the dynamics of information transfer and storage. The information dynamics measures of transfer entropy (TE) and active information storage (AIS) have been used to quantify the intrinsic nonlinear dynamical properties of complex systems, especially systems performing distributed computing [42–48]. Whilst mutual information has previously been applied to neural network algorithms [49, 50], this metric does not capture the dynamical properties of directed information flow and storage. In contrast, TE quantifies how much information the past activity of a source component provides about the state update of a target component (beyond that provided by the target’s own past) and captures directed information flow between the components [42, 48]. AIS quantifies how much information the past activity of a component provides about its next value, capturing the memory that is active in the computation to update that component’s state [45]. TE is particularly useful in measuring how information carried by signals is propagated across a network, and can help identify the most important components involved in the dynamics of a network [44, 51], and performance of the network can thus be optimised accordingly [52].

Crucially, TE and AIS have been used to study the dynamics of intrinsic information processing in many other systems undergoing order-chaos phase transitions through critical-like points, including canonical complex systems such as Random Boolean Networks [53, 54] and the Ising model [55–57], echo state networks [58], spiking neural networks [59], in vitro neural networks [60], and neural mass models [61]. Typically, TE and AIS indicate maximisation of information storage and transfer in the intrinsic information processing at or near the edge-of-chaos, and also often point to a balance between these operations in this regime. Such a maximisation of information processing capabilities in intrinsic dynamics provides an explanation for optimal computational capabilities found at the edge-of-chaos in some systems such as recurrent neural networks [36, 58] (and conjectured in others). In a related study to ours, Cramer et al. [62] measured TE and AIS in a different neuromorphic system, finding that information transfer and storage in the intrinsic dynamics of the system increased as the system approached criticality from an asynchronous/bursting state. Whether and how those findings extend to neuromorphic NWNs – where nodes undergo inactive-active transitions not only individually but strongly in tandem with their ongoing time-series activity rather than in response to some external tuning – and how this relates to implementations of reservoir computing therein, is an open question.

This work is organised as follows. First, we present the spatio-temporal correlation between properties of the NWN components and their information dynamics. Then we show the time-series evolution of relevant network properties, including average TE and AIS. We then present results showing how a network’s performance in learning and memory can be interpreted in terms of the information dynamics measures. We also present results showing the influence of network connectivity density on information processing performance.

3.2 Results

Here, we use simulations based on a physically-motivated model [30, 39, 40, 63] to identify the junctions responsible for signal transmission and investigate the corresponding dynamical network characteristics in terms of the information theoretic measures TE and AIS. The model captures the structural features of nanowire network self assembly. Nanowire-nanowire intersections are modelled as memristive electrical junctions, based on an experimental system [30]. An externally applied voltage is redistributed across the networked junctions (in accordance with Kirchoff’s circuit laws), causing some to switch from a low-conductance state to a high-conductance state (see Methods for details). Time-series analysis and benchmark learning and memory tasks are used to demonstrate how TE and AIS can provide new insight into the dynamical information processing capacity of NWNs.

3.2.1 Junction centrality and information flow

Junction state and network current flow. In network analysis, centrality is a key metric that can help identify the relative importance of nodes in relation to the underlying network structure and connectivity [64, 65]. Despite the existence of numerous classes of centrality algorithms, betweenness centrality can be considered as one of the most well-known measures as it quantifies how often a node (or edge) performs as a “bridge” on the shortest paths between arbitrary nodes in the network [66, 67]. This study employs a variation of betweenness centrality based on current-flow [68], relevant for nodes and edges within electrical systems (see Methods for details). At a specific time t during network activation, current-flow edge betweenness centrality c_{EB} is calculated using Eq. 3.7.

Fig. 3.1(a) shows a snapshot of the network graph at $t = 1.3$ s. The orange nodes indicate the first current transport path, formed earlier at $t = 1.218$ s and coinciding with network activation. This WTA path is equivalent here to the shortest path length between the source and drain nodes. The locations of high-conductance junctions beside the WTA pathway

demonstrate the importance of centrality – junctions at more central locations typically switch on earlier and thus regulate the subsequent dynamics of information spread by current. This suggests that the interplay between network connectivity and junction dynamics results in adaptive information flow.

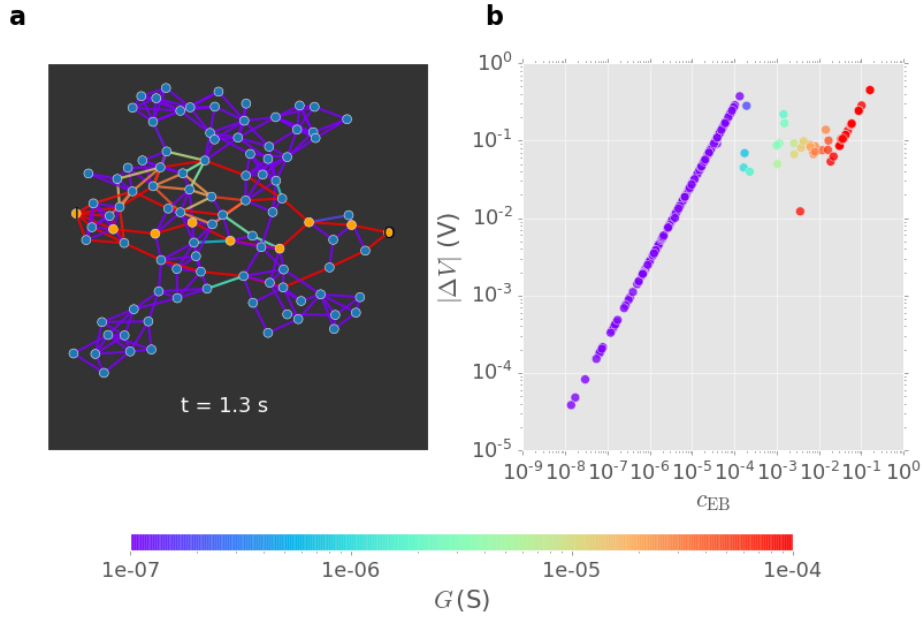


Fig. 3.1 Simulation of a neuromorphic NWN with 100-nanowire nodes and 261-junction edges. (a) Snapshot of network graph at $t = 1.3$ s, showing the first current path (between orange nodes, established earlier at $t = 1.218$ s) and edge (junction) conductance (colourbar). (b) Magnitude of voltage difference $|\Delta V|$ and conductance G across network junctions as a function of their current edge betweenness centrality c_{EB} at $t = 1.3$ s.

Fig. 3.1(b) plots the magnitude of voltage difference $|\Delta V|$ across all network junctions as a function of c_{EB} at $t = 1.3$ s, with colour indicating each junction's conductance state. Three regimes can be clearly identified: (i) an extensive linear regime spanning approximately 4 orders of magnitude in both $|\Delta V|$ and c_{EB} characterised by low conductance ($\lesssim 10^{-6}$ S) states; (ii) a shorter transition regime with intermediate conductance values ($10^{-4} - 10^{-5}$ S), corresponding to junctions in the tunneling transport regime; and (iii) a short linear regime characterised by high conductance ($\gtrsim 10^{-5}$ S) states, corresponding to junctions that have switched on and carry Ohmic current. Junctions in the high conductance regime therefore form the WTA path, pushing the collective conductance of the network to a higher level. Furthermore, junctions in the transition regime will form subsequent current paths. Within the low conductance regime, junctions with higher c_{EB} enter the tunneling regime earlier and thus switch on earlier. For instance, at $t = 0$ s, all junctions reside in the low conductance

regime. The centrality c_{EB} at $t = 0$ s reflects the order of junctions to be turned on through the activation process. These results show that junctions in the highest conductance state are generally also those with the highest centrality.

Transfer entropy – information flow. The adaptive dynamics of NWNs makes them ideally suited to temporal information processing tasks [27, 30]. In general, the performance of a given network structure depends on the specific task [38–41, 69, 70]. However, information theory provides an effective approach to assessing and characterising intrinsic network performance in a task-independent manner [71, 72]. Information theoretic measures enable the quantification of the information transfer between nodes and associated information storage [43–45, 47, 73]. Transfer entropy (TE), in particular, is a key information theoretic measure of predictive information provided by a source node to a destination node [43, 44]. Estimation of TE enables identification of information propagation and thereby active areas of a given network structure. In this study, TE is based on edge connectivity – for each edge (junction) in the network, TE is estimated in both directions and the sum measures net activity on the edge. For the following results, 50 network simulations were performed with different source-drain locations (keeping the graphical distance between these nodes constant).

Fig. 3.2 shows junction TE as a function of c_{EB} at $t = 0.5$ s, $t = t^* = 1.22$ s, and $t = 2.5$ s, where t^* represents the network activation time. Each TE– c_{EB} data point on the scatter plot represents one junction in 50 network simulations. Since TE estimation at a specific time point includes bias due to the estimator (see Methods), a window of 100 time steps (0.1 s) is employed to mitigate uncertainty. Fig. 3.2(a) shows TE vs c_{EB} at $t = 0.5$ s. At this time, networks are in a pre-activation stage. Most junctions remain in a low conductance state. TE for most junctions is negligible. Only junctions with the highest c_{EB} values exhibit small TE fluctuations.

Fig. 3.2(b) shows TE vs c_{EB} at $t = t^*$. At this time, the most central junctions ($c_{EB} \gtrsim 10^{-2}$) switch on (red), while less central junctions (blue) exhibit higher TE and thus stronger information dynamics. A notable number of junctions are in the transition regime, between off and on ($10^{-4} \lesssim c_{EB} \lesssim 10^{-2}$). The moving window average suggests that junctions entering this regime ($c_{EB} \sim 10^{-4}$) have the highest TE values and are thus poised to transmit substantial information.

Fig. 3.2(c) shows TE vs c_{EB} at $t = 2.5$ s, at which point networks have activated. Here, fewer junctions are in the transition regime compared to $t = t^*$, as most are already switched on. Again, more central junctions exhibit stronger information dynamics, and junctions

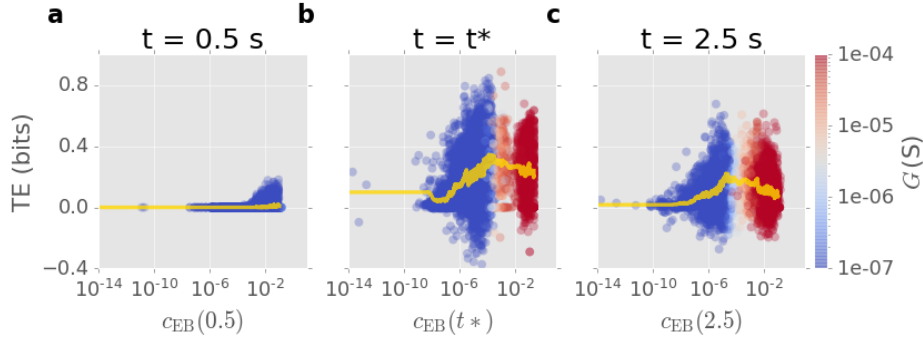


Fig. 3.2 Junction TE as a function of edge betweenness centrality c_{EB} calculated for 50 simulations of a 100-node network with varying source-drain locations (fixed distance) at three different time points: (a) $t = 0.5$ s; (b) $t = t^*$, where $t^* = 1.22$ s is network activation time; and (c) $t = 2.5$ s. Each data point represents one junction at the specified time. The colourbar represents the corresponding junction conductance. The yellow curve is a moving window average (size 0.1 s) of the scatter points calculated by sorting the scatter points based on their c_{EB} values and averaging c_{EB} , TE values within each window.

entering the transition regime have the highest TE. However, TE at this network state is generally lower compared to at $t = t^*$. Information dynamics thus tends to be weaker after activation, as fewer junctions are in the process of switching.

These results demonstrate that the structure and connectivity of the network plays an important role in the information dynamics of neuromorphic NWNs. Junctions at more central positions tend to exhibit stronger TE. Similar results can be observed for AIS as well (see supplementary Fig. 1). Collectively, information transfer and storage achieves a higher level when the network begins to activate. Locally, at the junction level, network activation coincides with junctions switching to a high conductance state.

3.2.2 Network dynamics of information transfer and storage

Network time series Fig. 3.3 shows the time-series of key network properties in response to a Mackey-Glass input signal (with $\tau = 50$): (a) conductance G ; (b) its derivative $\delta G / \delta t$; (c) TE, calculated for both Gaussian and Kraskov estimators, and averaged over all junctions; (d) AIS (Kraskov); and (e) conductance-weighted modularity Q , which is a metric that captures and characterises the community structure of networks (see Methods) [74–76]. The time series is split into three regimes: pre-activation, activation and post-activation, shaded red, green and beige, respectively.

In the pre-activation regime, all junctions are off, with very low G . Junction conductance states start to evolve based on their voltage. G increases initially in junctions with higher

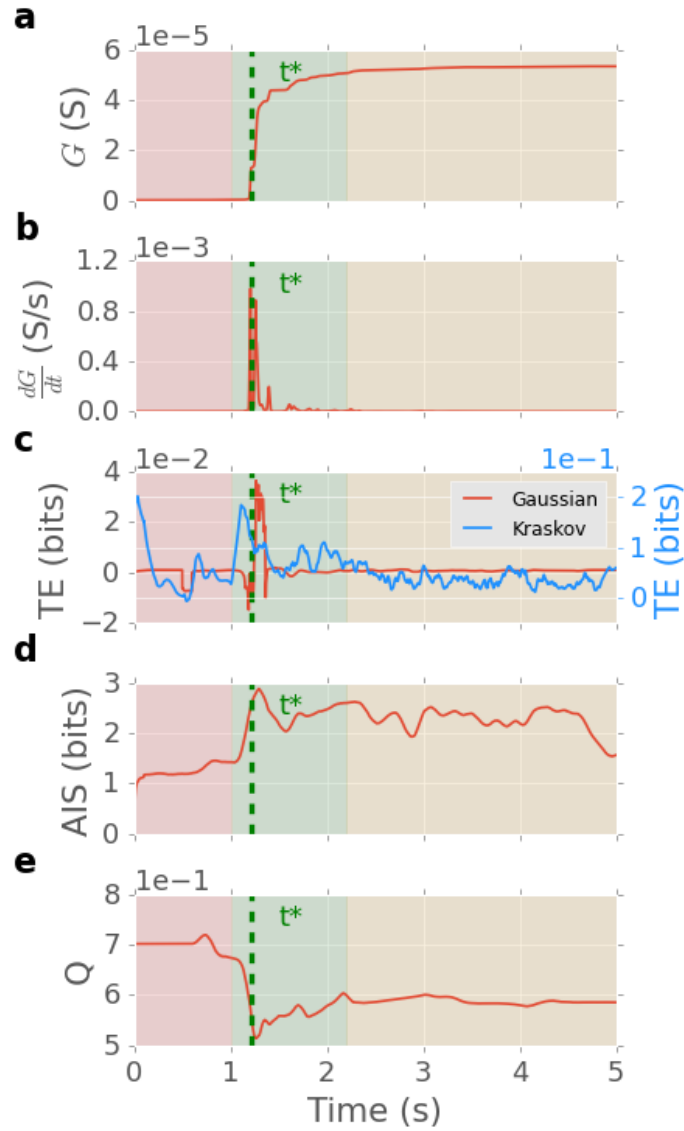


Fig. 3.3 Network time series. (a) Network conductance G . (b) Time derivative of G . (c) Transfer entropy TE, calculated with the Gaussian and Kraskov estimators and averaged over all junctions. (d) Active information storage AIS. (e) Network modularity Q . Red, green and beige shaded regions indicate pre-activation, activation and post-activation periods, respectively, for a Mackey-Glass input signal. The dashed vertical green line indicates the activation time, $t^* = 1.22$ s, coinciding with formation of the first current path.

voltages. TE increases when the signal is first applied to the network, but in general remains relatively low in this regime. AIS remains relatively low in this regime since there is little memory of past states. Here only AIS calculated based on the Kraskov estimator is presented, since little linear information storage was captured here by the Gaussian estimator. Q remains constant since all junctions remain in a low- G state. The activation regime is characterised by a steep increase in G , sharp spikes in $\delta G/\delta t$, a peak in TE and AIS, and a drop in Q . During activation, several neighbouring central junctions switch on and G jumps, resulting in time derivative spikes. The first spike in $\delta G/\delta t$ coincides with the formation of the first WTA current path at $t = t^* = 1.22$ s (green dashed line). This also coincides with a sharp dip in Q , which indicates that the network suddenly becomes highly integrated. TE also exhibits dramatic changes near t^* , particularly for the Gaussian estimator. The Kraskov-estimated TE shows the largest increase just before t^* , which may suggest potential predictive capacity. Indeed, the TE results from this non-linear estimator were an order of magnitude larger than those of the Gaussian estimator near the activation time, suggesting strong non-linear interactions in that regime. In the post-activation regime, many junctions are in high-conductance states and consequently, G is in a high stable state. TE returns to a low, approximately constant level, indicating little information flow. In contrast, the AIS increases to a higher level at activation and remains there post-activation; in conjunction with the lower information flow this indicates more rich, self-predictable dynamics of the nodes in comparison to pre-activation. Q remains stable in this regime and at a lower level than during pre-activation, indicating the network has evolved to a more integrated state post-activation.

Snapshots of the network at different stages of activation are shown in Fig. 6.4 (a). Here, nodes and edges are coloured by their respective TE (Kraskov) averaged via a moving time window (0.1 s). Several time points are chosen from the three regimes referred to in Fig. 3.3. At $t = 0.5$ s, the whole network is in the pre-activation regime, during which information dynamics are barely discernable. At $t = 1.1$ s, when the network is approaching activation, the majority of nodes/edges are involved in the information dynamics. The most active nodes and edges are observable around the center of the network (i.e. high c_{EB}), between source and drain. At $t = t^* = 1.22$ s, the WTA path emerges from the area that was previously most active (at $t = 1.1$ s), while areas adjacent to the WTA path also exhibit strong dynamics. These areas form additional current pathways at later times, as evident at $t = 2.0$ s. At this point, the network's activation winds down and its dynamics start to fade. At $t = 2.5$ s, the network is in the post-activation regime, where – despite the conductance being maximal – the information dynamics decrease to a lower (but still observable) level. These network snapshots also highlight the predictive capacity of TE activity. Network nodes/edges with

the strongest TE dynamics (e.g. at $t = 1.1$ s) are subsequently involved in the first current path formation and ongoing network activity.

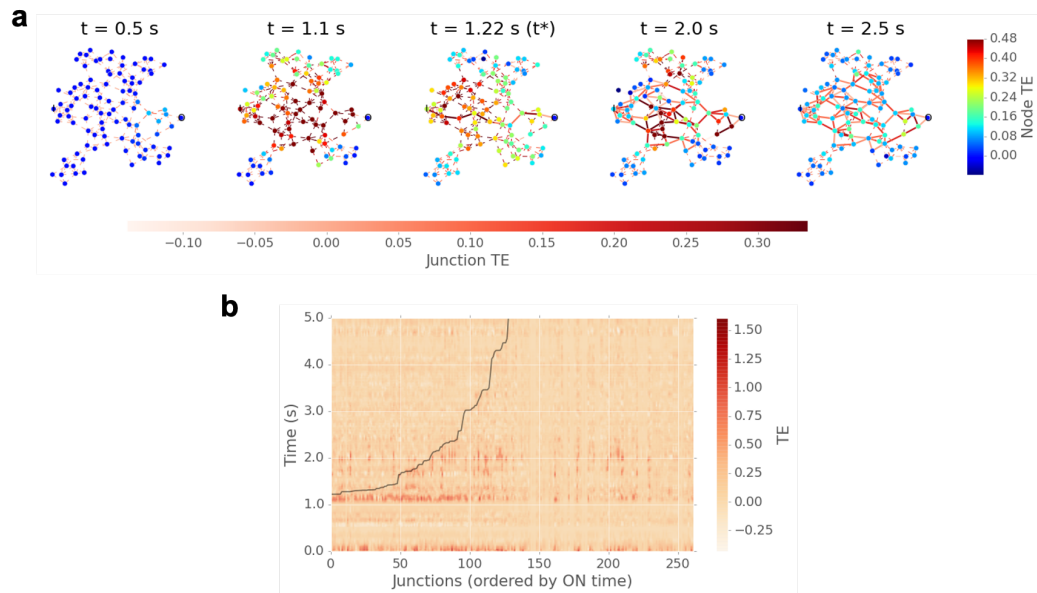


Fig. 3.4 Network information flow dynamics. Top panel – Snapshots of the network taken at time points before activation ($t = 0.5$ s), at the onset of activation ($t = 1.1$ s), when the first current path forms ($t = 1.22$ s), and after network activation ($t = 2.0, 2.5$ s). Nodes are coloured with time-averaged TE flow (inTE + outTE) within the last 0.2 second window using the Kraskov estimator. Junctions are coloured by the sum of corresponding TE in both directions. Bottom panel – heatmap of junction TE as a function of time, with junction number ordered by switch on time.

Fig. 6.4 (b) shows a heat map of each junction's TE during activation. Here the x -axis represents junction indices ordered by the time they switch on (i.e. lower means earlier). The black curve denotes their corresponding switch on time. Hot zones (dark red) in the heat map demonstrate strong transfer events occur just before junctions switch on, particularly for those that switch on earlier and are involved in network activation. Based on the results from Fig. 3.2 (b), the more central a junction is, the earlier it switches on and the stronger information dynamics it exhibits. Weaker TE events that are still evident at later times correspond to junctions whose conductance continues to evolve, but does not reach a sufficiently high state for the junction to switch on. Some TE events evident on the right half of the panel are not coupled with junction switching behaviours. These dynamics are induced by the collective conductance change of the network due to the WTA path formation.

3.2.3 Memory capacity and learning

The previous results shown in Fig. 6.4 reveal how information is dynamically transmitted through a neuromorphic NWN. Here, we investigate information storage and transfer dynamics involved in two different reservoir computing benchmark tasks: memory capacity (MC) [77] and non-linear transformation (NLT) [38].

Network pre-initialisation As shown in Fig. 3.3 and Fig. 6.4, the continuously evolving internal states of a NWN give rise to network dynamics with distinctive features around activation. Thus, we pre-initialised networks with different initial states (i.e. pre-activation, activation and post-activation states) to investigate the influence of initial network state on information storage and transfer, as measured by AIS and TE, respectively. Networks were pre-initialised using the same Mackey-Glass signal as for Fig. 3.3. Network states, defined as the instantaneous junction conductance state distribution, were recorded every 50 time steps (0.05 s) and used as the initial states for the MC and NLT tasks. The time at which the network state was sampled is defined as its pre-initialisation time. A pre-initialisation time of zero corresponds to all junctions set to the same low-conductance initial state, as used in the results presented thus far.

Fig. 3.5 (a) shows MC as a function of network pre-initialisation time. The corresponding AIS, averaged over its time-series, is shown in Fig. 3.5 (b). Both MC and AIS peak when the network is pre-initialised to the state at its activation time (green dashed line). Similarly, Fig. 3.5 (c) shows NLT performance is optimal for pre-initialisation states approaching and at network activation. The corresponding TE, averaged over its time-series, as shown in Fig. 3.5 (d), also peaks when the network is pre-initialised to states approaching activation. These results reveal that network states at or approaching activation are optimal for these information processing tasks.

Information storage and transfer in the networks exhibits similar optimal behaviour when the initial state approaches the critical-like network activation point. However, each task benefits from different network dynamical features. The MC task exploits the short-term memory of memristive elements in the network, as well as its recurrent structure. Information storage captures self-predictability at each node, incorporating strength of recurrence on nodes (i.e. relevant past information), and thus reflects the memory capacity of the network. The NLT learning task, on the other hand, involves the transfer and modification of a continuous signal and thus, the node-to-node predictive information contributing to the computation is best captured by TE. Furthermore, the energy-based metric “deviation from linearity” [78] (see supplementary Fig. 2), which provides a measure of a network’s capability

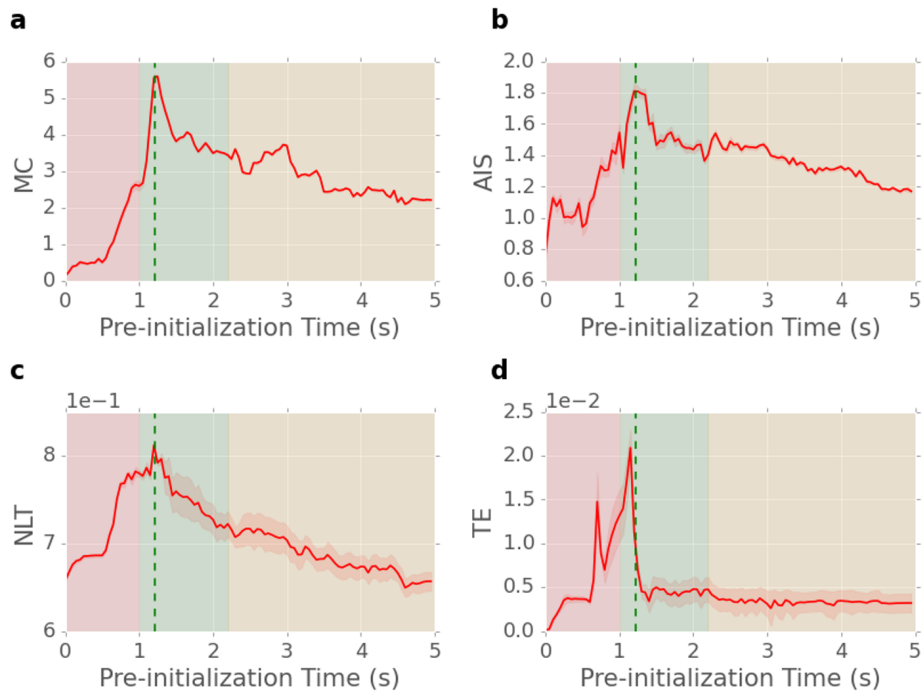


Fig. 3.5 Information processing tasks and corresponding information dynamics measures for networks pre-initialised with different initial states (shaded regions represent standard error): (a) memory capacity (MC) task performance; (b) average active information storage (AIS) for MC; (c) non-linear transformation (NLT) accuracy; (d) average transfer entropy (TE) of NLT. Plots in (a) & (b) are averaged over multiple iterations of MC due to the inherent randomness of the task. Standard error is too small to be noticed. Plots in (c) & (d) are averaged over different amplitudes of the NLT task ranging between 0.6 – 1V.

of nonlinearly transforming the input into different frequency components, is maximised when the network's initial state approaches activation, which is optimal for both tasks.

Information processing capacity As shown in the previous section, pre-initialisation enables optimal memory and learning performance. Here we investigate this further by considering networks of varying information processing capacity, as represented by average degree $\langle k \rangle$ (i.e. average number of connections per node). We generated 16 100-node NWNs with varying number of junctions (ranging from 261 to 4684), to determine whether increasing connectivity and therefore, information processing resources, increases network performance. We used the same MC task as in fig. 3.5 to assess relative performance for different $\langle k \rangle$.

Fig. 3.6(a) shows MC with respect to pre-initialisation time for networks with relatively low $\langle k \rangle$ (5–15). Similar to the results in fig. 3.5, MC is optimised when the networks are

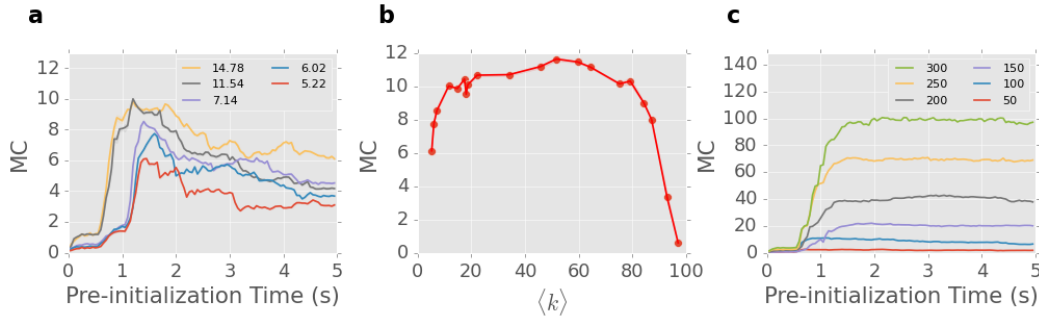


Fig. 3.6 Memory capacity (MC) for networks with varying information processing resources. (a) Average MC as a function of pre-initialisation time for 100-node networks with sparse connectivity ($5.22 \leq \langle k \rangle \leq 14.78$). (b) Maximum MC achieved for 100-node networks as a function of $\langle k \rangle$. (c) Average MC vs. pre-initialisation time for networks with different number of nodes (50 – 300) and similar $\langle k \rangle$.

pre-initialised to their respective critical-like activation state. Additionally, networks with higher $\langle k \rangle$ tend to show better performance in the MC task when in this state.

Fig. 3.6(b) shows the best MC performance of each of the 16 networks (pre-initialised to its optimal state) as a function of $\langle k \rangle$. Networks with $\langle k \rangle < 20$, show increasing performance as density increases, due to more possible current pathways being formed during network activation. For networks with $20 < \langle k \rangle < 70$, performance plateaus, while for $\langle k \rangle > 70$, performance decreases significantly. These results indicate that there exist an optimal connectivity density for NWNs to perform information processing tasks. A small drop in MC can be observed for a specific network realisation with $\langle k \rangle \sim 20$, suggesting that other topological properties could affect the network's memory as well. The optimal computational density regime coincides with the maximum deviation from linearity (see supplementary Fig. 2) – networks with better performance exhibit stronger deviation from linearity.

Fig. 3.6(c) plots MC for networks of different sizes (number of nanowire nodes) but similar $\langle k \rangle (\approx 20)$, as a function of pre-initialisation time. A trend similar to Fig. 3.6(a) is evident – the best performance can be achieved with appropriate pre-initialisation. Furthermore, the plot demonstrates that larger networks tend to exhibit better performance as they are able to incorporate more information processing resources.

These results demonstrate that the optimisation of NWN memory and learning performance around critical-like points holds for different network realisations. For networks with the same number of nanowire nodes, there exists an optimal regime of $\langle k \rangle$ for the NWNs to exhibit the best performance. On the other hand, larger networks (more nanowires) generally makes the performance even better.

3.3 Discussion

This study is the first to investigate the spatio-temporal dynamics of information transfer and storage in neuromorphic nanowire networks. We found that topological network dynamics, as measured by edge betweenness centrality c_{EB} , influences information transfer, with more central junctions exhibiting stronger TE. We also found TE is maximised when individual junctions approach their respective transient dynamics regime (corresponding to a tunneling electron transport regime). These results corroborate those of Joyce et al.’s study, which showed that high centrality components in brain networks tend to be more influential in network dynamics [79]. However, the centrality metric used in that study was based on static structural properties and the effect on information dynamics was not investigated. TE in contrast has been widely used in neuroscience to study the dynamic activity in a network in terms of measuring information flow between nodes. This has included, for example, inferring functional/effective connectivity from analysing EEG/MEG data [44] and spiking cortical networks [51], as well as finding that TE is correlated to connectivity strength [80].

Our time-series analysis (fig. 3.3) revealed that during network activation, TE and AIS are maximised when the first (WTA) current path is about to form. Together with our previous results showing that neuromorphic nanowire networks exhibit ordered and chaotic dynamical regimes, as well as a regime near activation resembling a “critical-like” state at the edge-of-chaos under self-regulated dynamics [41], these results corroborate several other analyses of information dynamics in systems undergoing order-chaos phase transitions through critical-like points such as random Boolean networks [53, 54], the Ising model [55–57], echo state networks [58] and other systems [59–61]. Some of these studies suggested maximisation of both information storage and transfer at the edge of chaos (e.g. in echo state networks [58]), whilst others found AIS peaks under ordered dynamics and TE peaks under chaotic dynamics with a balance between these operations at the edge of chaos (e.g. in random Boolean networks [53, 54]). Our results align with the former group, with both TE and AIS peaking at the critical-like activation point. These differences may be attributed to the different types of networks and activation methods used in the different studies. Nevertheless, in both groups the edge of chaos represents a regime where both operations of information storage and transfer are strong, and the similarity of information dynamics behaviour is intriguing, especially considering the suggestion that information processing may be optimised around the edge-of-chaos [34]. Additionally in our study, a local measure of TE on junctions exhibited a pre-emptive signature with respect to their

individual switching behaviour, echoing early information transfer observed prior to the critical point in the Ising model [55–57] and in Kuramoto oscillator synchronisation [81].

Neuroscience studies on network modularity typically investigate the relationship between a static modular structure (unweighted) and functionality [82, 83]. Bassett et al., however, found that brain networks exhibit dynamic reconfiguration during learning and the modular community structure of the network changes with time [84]. In our study, we calculate modularity based on the evolving junction conductance. The corresponding time-series, fig. 3.3 (e), indicates that modularity first increases when conductance increases across several junctions. The network then becomes most integrated (modularity minimised) at the critical-like point. This can be attributed to inter-module connections formed by stronger junctions and some nodes involved in the WTA path divested to form fewer larger modules (see supplementary Fig. 3). In a related study, Husken et al. investigated the evolution of modularity in feed-forward neural networks in a task-dependent manner [85]. They found modularity increases as the learning process continues before dropping to a steady-state. A similar peak in modularity evolution with respect to training was also shown by Shine et al. in deep neural networks performing the MNIST digit recognition task [86]. They found that learning accuracy saturates at maximum modularity. Our results likely reflect the small-world and recurrent structure of NWNs, which are fundamentally different from feed-forward artificial neural networks.

Our study also revealed that learning performance (NLT) and memory capacity (MC) of NWNs can be optimised by pre-initialising the network to its critical-like activation state, where AIS and TE are both optimised (fig. 3.5). A similar approach was also employed by Klos et al. to perform dynamical learning of different tasks. They found that appropriate pre-initialisation enables the network to better adapt to a target [87]. Our results are also consistent with those of Boedecker et al.’s investigation of optimal network performance around the edge-of-chaos in RBNs [58] and Bertschinger et al.’s results using recurrent neural networks [36]. Furthermore, Cramer et al.’s study on criticality in spiking neural networks implemented on a memristor hardware system showed that tasks of different computational complexity benefit by varying amounts from the network’s dynamical regimes [62]. In our study, the MC task, which measures short-term memory, correlates with AIS dynamics, while the NLT task correlates with TE dynamics. Another metric called ‘deviation from linearity’ has been proposed to measure non-linearity within reservoir systems [78]. Butcher et al. found that larger deviation from linearity correlates with better performance for a reservoir with random static projections (R²SP) [88]. Similarly, we found that the best memory and learning performance of pre-initialised NWNs correlates with the maximum deviation

from linearity based on the network's intrinsic dynamics. We also found that the learning performance of pre-initialised NWNs correlates with "flexibility" (see supplementary Fig. 4), measured by the frequency of changes in network modular structure, as proposed by Bassett et al. [84].

The performance of conventional silicon based chips is mainly determined by the number of transistors, which has doubled every two years in accordance with Moore's law. In the context of post-Moore's law information processing with neuromorphic reservoir networks, studies have shown that the performance can be strongly affected by the network size, structure and connectivity [89]. Our results (Fig. 3.6) show that, unlike traditional CMOS chips, the performance of NWNs does not increase monotonically with respect to its connectivity, i.e. computational density. Instead, there exists an optimal range of $\langle k \rangle$. In addition, larger networks tend to exhibit significantly better performance when pre-initialised. These results agree with Legenstein et al.'s results on the computational performance of neural circuit models, which suggest that an optimal $\langle k \rangle$ may be beneficial for computing, especially at the edge-of-chaos [37, 90]. Our finding of an optimal range of $\langle k \rangle$ for information processing also corroborates similar findings by Snyder et al. in their studies on RBNs and random automata networks [91, 92]. Furthermore, we found that networks within the optimal range of $\langle k \rangle$ exhibit larger deviations from linearity (see supplementary Fig. 2).

All the simulations presented here, including network pre-initialisation, were performed using one input node. However, more complex tasks would generally require input signals to be delivered to multiple input nodes [40, 93], which in general would change the pre-initialised network states. In a real experimental hardware device, it would be difficult to control pre-initialisation to an optimal state. However, other studies suggest that some pre-initialisation, even if not to optimal states, could still be beneficial to learning tasks, enabling strategies such as transfer learning [41].

In conclusion, we investigated the spatio-temporal dynamics of information transfer and storage in neuromorphic nanowire networks subject to a complex time-varying electrical signal. Our results demonstrate that information dynamics metrics provide a valuable tool for analysing the information processing capability of neuromorphic systems. We found information dynamics to be maximised around the critical-like activation point in nanowire networks, which agrees with previous studies on other similar computing systems based on recurrent neuromorphic networks. Furthermore, based on the information dynamics, we found that pre-initialisation enables these networks to achieve optimal performance in memory and learning tasks. We also found that there exists an optimal range of network density for

computation. These results warrant further studies of information dynamics involved in more computationally complex tasks that are non-static, such as action recognition.

3.4 Methods

3.4.1 Electrical network model

Fig. 3.7 (a) shows a visualisation of a simulated nanowire network containing 100 nanowires and 261 junctions. Self-assembly is modelled by distributing nanowires on a $3 \times 3 \mu\text{m}^2$ 2D plane, with their centers uniformly sampled from $[0, 3]$ and orientation uniformly sampled from $[0, \pi]$. The lengths of the nanowires are sampled from a gamma distribution (mean = 100 nm, standard deviation 10 nm), based on experimental measurements [30]. Fig. 3.7 (b) shows the graphical representation of the NWN in fig. 3.7 (a). Nanowires are represented as nodes, while junctions are represented as edges between corresponding nodes.

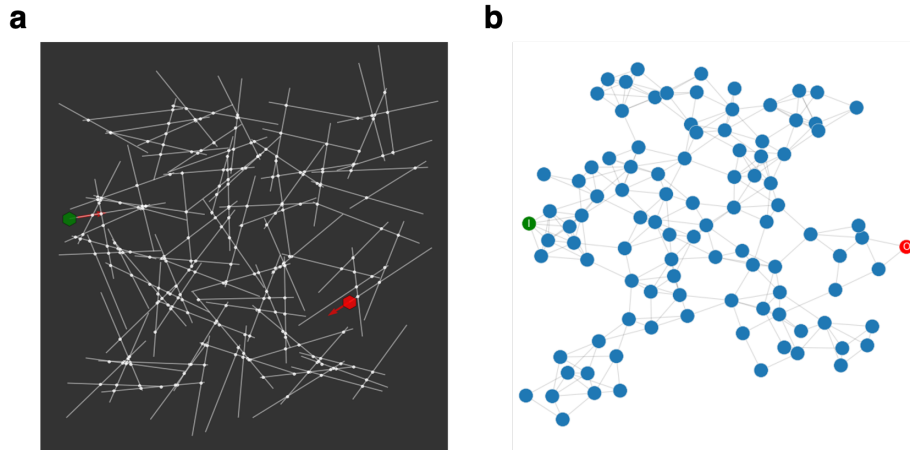


Fig. 3.7 (a) Simulated NWN with 100 nanowires and 261 junctions. (b) Graphical representation of the NWN in (a). Green denotes input nanowire contact electrode-node and red represents drain node.

Unless otherwise specified, all simulation results are obtained using a NWN with 100 nanowires and 261 junctions. All variables, except the adjacency matrix A , are time-dependent. Two nodes on the left and right end of the network (as shown in fig. 3.7) are used as source and drain electrodes. A Mackey-Glass time-series signal with delay parameter $\tau = 50$ is used as the input voltage signal, delivered to the source electrode. Kirchhoff's voltage and current conservation laws are solved at each time step using the nodal analysis approach applied to the graphical representation of the network [94]. The junction

conductance distribution evolves as voltage bias is continuously applied to the network. The Laplacian matrix of the junction conductance is used to describe the network's state. Hence, at each time step, the junction voltage distribution V across the network is obtained by solving [95]

$$\mathcal{L}^\dagger V = I \quad , \quad (3.1)$$

where I is current and $\mathcal{L}^\dagger$ is the expanded graph Laplacian of the network, expressed in block matrix representation as

$$\mathcal{L}^\dagger = \left[\begin{array}{c|c} \mathcal{L} & C \\ \hline C & 0 \end{array} \right] \quad , \quad (3.2)$$

where $\mathcal{L}$ is the graph Laplacian and where the elements of C are either 1 if the nanowire node is connected to an external electrode or 0 otherwise. The Laplacian is

$$\mathcal{L} = D - W \quad , \quad (3.3)$$

where W is the weighted adjacency matrix of the network, with the weights representing junction conductance distribution:

$$W_{ij} = A_{ij}G(i, j) \quad , \quad (3.4)$$

where $G(i, j)$ is conductance on the edge connecting nodes i and j and D is the weighted degree matrix generated from W :

$$D = \mathbf{diag}(d_i) \quad , \quad d_i = \sum_{k=1}^N W_{ik} \quad . \quad (3.5)$$

3.4.2 Junction Model

Nanowire-nanowire cross-points are modelled as ideal, electrically insulating, ionically conducting junctions, with voltage-threshold memristive switching[30, 38, 39, 96], modulated by electron tunnelling transport [97]. The model does not take into account noise fluctuations, which can contribute additional nonlinearities that can generate even richer dynamics [97]. Junction conductance, $G = G(\lambda)$, depends on a state variable $\lambda(t)$ that parametrises the conducting filament responsible for memristive switching. All junctions are initially in a high resistance “off” state. For each junction in the network, resistance switches to “on” state when $\lambda \geq \lambda_{\text{crit}}$, where λ_{crit} is a set threshold. The ratio of these resistance states is $R_{\text{off}}/R_{\text{on}} = 10^3$,

with $R_{\text{on}} = G_0^{-1}$, and $G_0 = (13 \text{ k}\Omega)^{-1}$ is the conductance quanta. The evolution of $\lambda(t)$ is described by a polarity-dependent voltage-threshold model [39, 63, 96, 97]:

$$\frac{d\lambda}{dt} = \begin{cases} (|V(t)| - V_{\text{set}})\text{sign}[V(t)], & |V(t)| > V_{\text{set}} \\ 0, & V_{\text{reset}} < |V(t)| < V_{\text{set}} \\ b(|V(t)| - V_{\text{reset}})\text{sign}[\lambda(t)], & |V(t)| < V_{\text{reset}} \end{cases} \quad (3.6)$$

where V_{set} is the on-threshold and V_{reset} is the off-threshold, and b is a positive constant defining the relative rates of decay of the filament. The following default parameter values are used for all the simulation results presented in this study: $V_{\text{set}} = 10^{-2}\text{V}$, $V_{\text{reset}} = 10^{-3}\text{V}$ and $b = 10$. Experimental validation of this model is presented elsewhere [30, 63, 97].

3.4.3 Centrality

Centrality identifies the most important components in a network, with various definitions used in the literature [74]. In this study, a variation of betweenness centrality is used, based on current flow in circuits proposed by Brandes and Fleischer [68]. The current betweenness centrality of an edge in the network is defined as

$$c_{EB}(e) = \frac{\sum_{i \neq j \in N} \tau_{ij}(e)}{(N-1)(N-2)}, \quad (3.7)$$

where $\tau_{ij}(e)$ is the current flow through edge e between nodes i and j and N is the number of nodes in the network.

3.4.4 Modularity

Modularity is a metric that characterises the community structure of networks [74–76]. In this study, communities are determined by the Louvain method [98]. The resolution parameter of the detection method is chosen according to a mutual information based method proposed by Ronhovde et al [99]. To capture the influence of network dynamics, we define a modularity weighted by junction conductance:

$$Q^w = \frac{1}{g} \sum_{i,j} \left[G(i,j) - \frac{d_i d_j}{g} \right] \delta_{m_i m_j} \quad , \quad g = \sum_{i,j \in N} G(i,j) \quad (3.8)$$

where m_i is the community that node i belongs to, and δ is the Kronecker delta function.

3.4.5 Information Dynamics

Transfer entropy measures how much information the past of time-series process $\mathbf{Y}$ contributes to predicting the next value of time-series process $\mathbf{X}$, in the context of the past of $\mathbf{X}$. Specifically, transfer entropy is conditional mutual information between these variables, written as [42, 48]:

$$\begin{aligned} T_{Y \rightarrow X}(k, l) &= I(\mathbf{Y}_n^{(l)}; X_{n+1} | \mathbf{X}_n^{(k)}), \\ &= \left\langle \log \frac{p(x_{n+1} | \mathbf{x}_n^{(k)}, \mathbf{y}_n^{(l)})}{p(x_{n+1} | \mathbf{x}_n^{(k)})} \right\rangle, \end{aligned} \quad (3.9)$$

where n is a time index for the variables, $\mathbf{Y}_n^{(l)}$ is the variable representing the past l values of the source up to and including time n , X_{n+1} the next value of the target at time $n + 1$, and $\mathbf{X}_n^{(k)}$ the past k values of the target. The lower case identifiers $(x_{n+1}, \mathbf{x}_n^{(k)}, \mathbf{y}_n^{(l)})$, indicate a specific realisation of the variables.

Active information storage A_X measures how much information the past of process $\mathbf{X}$ contributes to predicting the next value of that process [45, 47]. This is a mutual information between $\mathbf{X}_n^{(k)}$ and X_{n+1} :

$$A_X(k) = I(\mathbf{X}_n^{(k)}; X_{n+1}). \quad (3.10)$$

In the context of this study, information transfer can be calculated between any two nodes, as the voltage distribution across all the network nodes, and hence junction conductance, changes dynamically in response to the input signal. Transfer entropy (TE) from node i to node j is calculated based on their respective voltage time-series $\mathbf{V}_i$ and $\mathbf{V}_j$:

$$\text{TE}_{i,j} = T_{\mathbf{V}_i \rightarrow \mathbf{V}_j}. \quad (3.11)$$

TE activity on the edge connecting nodes i and j is represented heuristically as a sum of the TEs on both directions of the edge:

$$\text{TE}_e = \text{TE}_{i,j} + \text{TE}_{j,i} \quad (3.12)$$

In NWNs, memory is incorporated by the conductance evolution of memristive junctions, whose instantaneous state depends on past history of states. Therefore, active information storage (AIS) is calculated based on the conductance time-series $\mathbf{G}$ on junctions, which can

be written as:

$$\text{AIS}_e = A_G. \quad (3.13)$$

TE and AIS are calculated using the JIDT open-source software [46]. Two estimators are employed to capture different aspects of information dynamics. The linear-Gaussian estimator models the variables as having Gaussian distributions with linear interactions; it is fast, and very efficient when the model assumption is correct, but only captures the linear component of an interaction. In contrast, the Kraskov or KSG estimator [100–102] uses a nearest-neighbour technique to estimate the relationships in a model-free manner; this can detect arbitrary non-linear relationships, though typically requires more data to do so.

3.4.6 Memory and learning tasks

To investigate the information processing performance of NWNs, two reservoir computing benchmark tasks were implemented on the network: memory capacity and non-linear transformation. For both tasks, the input voltage signal was delivered to one source node and voltage was read out from multiple nodes, replicating experimental implementation [38].

Pre-initialisation

Networks are pre-initialised to different initial states to investigate the effect on task performance. The different initial states are defined as the unique junction conductance distributions captured every 0.05 s, in response to the input Mackey-Glass time-series signal. Then using each stored state as a unique initial state, memory capacity and non-linear transformation tasks are performed and corresponding AIS and TE calculated. Junction conductance levels are reset to zero homogeneously across the network prior to pre-initialisation. However, in physical NWNs it may take on the order of 24 hours for junction conductance levels to reset [30].

Memory Capacity

Memory capacity (MC) is a measure of a reservoir’s ability to recall information from its fading memory property [77]. A time series input voltage signal is generated from a uniform random distribution in the interval $[-2, 2]$. A linear combination of the network’s state (node voltage) at t is applied to reconstruct the previous input at $t - k$. Using $k = 1 \dots k_{\max}$ to include

different delayed intervals, MC is defined as

$$\text{MC} = \sum_{k=1}^{k_{\max}} \text{MC}_k, \text{ with } \text{MC}_k = \frac{\text{cov}^2(\mathbf{u}_{t-k}, \mathbf{V}_t)}{\sigma^2(\mathbf{u}_{t-k})\sigma^2(\mathbf{V}_t)}. \quad (3.14)$$

where MC_k represents the network's memory of the input from k steps before, $k_{\max} = N$, in which N is the number of nodes in the network, $\mathbf{u}$ is the input time-series signal and $\mathbf{V}$ is the voltage time-series readout from the network.

Non-linear transformation

Non-linear waveform transformation tests a reservoir's learning ability [38]. An input sinusoidal wave is nonlinearly transformed by the network into either a square or sawtooth waveform. Similar to the MC test, linear combinations of the network's states are used to fit the target signal:

$$\mathbf{Y} = \omega \cdot \mathbf{V}, \quad (3.15)$$

where ω is the fitted weight trained by linear regression and $\mathbf{V}$ is the voltage readout from the network. The accuracy is calculated as $1 - \text{RNMSE}$, where

$$\text{RNMSE} = \sqrt{\frac{\sum (\mathbf{Y} - \mathbf{T})^2}{\sum \mathbf{T}^2}}. \quad (3.16)$$

is the root-normalised mean square error, and where $\mathbf{Y}$ is the trained result while $\mathbf{T}$ represents the target time-series signal.

References

- [1] Roth, G. & Dicke, U. Evolution of the brain and intelligence. *Trends in Cognitive Sciences* **9** (5), 250–257 (2005). doi:[10.1016/j.tics.2005.03.005](https://doi.org/10.1016/j.tics.2005.03.005) .
- [2] Bassett, D. S. & Gazzaniga, M. S. Understanding complexity in the human brain. *Trends in Cognitive Sciences* **15** (5), 200–209 (2011). doi:[10.1016/j.tics.2011.03.006](https://doi.org/10.1016/j.tics.2011.03.006) .
- [3] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521** (7553), 436–444 (2015). doi:[10.1038/nature14539](https://doi.org/10.1038/nature14539) .
- [4] Arel, I., Rose, D. C. & Karnowski, T. P. Deep Machine Learning - A New Frontier in Artificial Intelligence Research [Research Frontier]. *IEEE Computational Intelligence Magazine* **5** (4), 13–18 (2010). doi:[10.1109/MCI.2010.938364](https://doi.org/10.1109/MCI.2010.938364) .
- [5] Tang, J. *et al.* Bridging Biological and Artificial Neural Networks with Emerging Neuromorphic Devices: Fundamentals, Progress, and Challenges. *Advanced Materials* **31** (49), 1902761 (2019). doi:[10.1002/adma.201902761](https://doi.org/10.1002/adma.201902761) .
- [6] Zhang, W. *et al.* Neuro-inspired computing chips. *Nature Electronics* **3** (7), 371–382 (2020). doi:[10.1038/s41928-020-0435-7](https://doi.org/10.1038/s41928-020-0435-7) .
- [7] Indiveri, G. *et al.* Neuromorphic Silicon Neuron Circuits. *Frontiers in Neuroscience* **5** (2011). doi:[10.3389/fnins.2011.00073](https://doi.org/10.3389/fnins.2011.00073) .
- [8] Pfeil, T. *et al.* Six Networks on a Universal Neuromorphic Computing Substrate. *Frontiers in Neuroscience* **7** (2013). doi:[10.3389/fnins.2013.00011](https://doi.org/10.3389/fnins.2013.00011) .
- [9] Merolla, P. A. *et al.* A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345** (6197), 668–673 (2014). doi:[10.1126/science.1254642](https://doi.org/10.1126/science.1254642) .
- [10] Jo, S. H. *et al.* Nanoscale Memristor Device as Synapse in Neuromorphic Systems. *Nano Letters* **10** (4), 1297–1301 (2010). doi:[10.1021/nl904092h](https://doi.org/10.1021/nl904092h) .
- [11] Ohno, T. *et al.* Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials* **10** (8), 591–595 (2011). doi:[10.1038/nmat3054](https://doi.org/10.1038/nmat3054) .
- [12] Yang, J. J., Strukov, D. B. & Stewart, D. R. Memristive devices for computing. *Nature Nanotechnology* **8** (1), 13–24 (2013). doi:[10.1038/nnano.2012.240](https://doi.org/10.1038/nnano.2012.240) .

- [13] La Barbera, S., Vuillaume, D. & Alibart, F. Filamentary Switching: Synaptic Plasticity through Device Volatility. *ACS Nano* **9** (1), 941–949 (2015). doi:[10.1021/nn506735m](https://doi.org/10.1021/nn506735m) .
- [14] Ielmini, D. & Wong, H.-S. P. In-memory computing with resistive switching devices. *Nature Electronics* **1** (6), 333–343 (2018). doi:[10.1038/s41928-018-0092-2](https://doi.org/10.1038/s41928-018-0092-2) .
- [15] Wang, Z. *et al.* Resistive switching materials for information processing. *Nature Reviews Materials* **5** (3), 173–195 (2020). doi:[10.1038/s41578-019-0159-3](https://doi.org/10.1038/s41578-019-0159-3) .
- [16] Roychowdhury, V. P., Janes, D. B., Bandyopadhyay, S. & Xiaodong Wang. Collective computational activity in self-assembled arrays of quantum dots: A novel neuromorphic architecture for nanoelectronics. *IEEE Transactions on Electron Devices* **43** (10), 1688–1699 (1996). doi:[10.1109/16.536815](https://doi.org/10.1109/16.536815) .
- [17] Tanaka, H. *et al.* A molecular neuromorphic network device consisting of single-walled carbon nanotubes complexed with polyoxometalate. *Nature Communications* **9** (1), 2693 (2018). doi:[10.1038/s41467-018-04886-2](https://doi.org/10.1038/s41467-018-04886-2) .
- [18] Mallinson, J. B. *et al.* Avalanches and criticality in self-organized nanoscale networks. *Science Advances* **5** (11), eaaw8438 (2019). doi:[10.1126/sciadv.aaw8438](https://doi.org/10.1126/sciadv.aaw8438) .
- [19] Sangwan, V. K. & Hersam, M. C. Neuromorphic nanoelectronic materials. *Nature Nanotechnology* **15** (7), 517–528 (2020). doi:[10.1038/s41565-020-0647-z](https://doi.org/10.1038/s41565-020-0647-z) .
- [20] Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences* **79** (8), 2554–2558 (1982). doi:[10.1073/pnas.79.8.2554](https://doi.org/10.1073/pnas.79.8.2554) .
- [21] Forrest, S. EMERGENT COMPUTATION: SELF-ORGANIZING, COLLECTIVE, AND COOPERATIVE PHENOMENA IN NATURAL AND ARTIFICIAL COMPUTING NETWORKS. *Emergent computation* 11 (1990) .
- [22] Lynn, C. W. & Bassett, D. S. The physics of brain network structure, function and control. *Nature Reviews Physics* **1** (5), 318–332 (2019). doi:[10.1038/s42254-019-0040-8](https://doi.org/10.1038/s42254-019-0040-8) .
- [23] Kuncic, Z. & Nakayama, T. Neuromorphic nanowire networks: Principles, progress and future prospects for neuro-inspired information processing. *Advances in Physics X* **6**, 1894234 (2021) .

- [24] Loeffler, A. *et al.* Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* **14**, 184 (2020). doi:[10.3389/fnins.2020.00184](https://doi.org/10.3389/fnins.2020.00184) .
- [25] Avizienis, A. V. *et al.* Neuromorphic Atomic Switch Networks. *PLoS ONE* **7** (8), e42772 (2012). doi:[10.1371/journal.pone.0042772](https://doi.org/10.1371/journal.pone.0042772) .
- [26] Stieg, A. Z. *et al.* Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials* **24** (2), 286–293 (2012). doi:[10.1002/adma.201103053](https://doi.org/10.1002/adma.201103053) .
- [27] Bellew, A. T., Bell, A. P., McCarthy, E. K., Fairfield, J. A. & Boland, J. J. Programmability of nanowire networks. *Nanoscale* **6** (16), 9632–9639 (2014). doi:[10.1039/C4NR02338B](https://doi.org/10.1039/C4NR02338B) .
- [28] Manning, H. G. *et al.* Emergence of winner-takes-all connectivity paths in random nanowire networks. *Nature Communications* **9** (1), 3219 (2018). doi:[10.1038/s41467-018-05517-6](https://doi.org/10.1038/s41467-018-05517-6) .
- [29] O’Callaghan, C. *et al.* Collective capacitive and memristive responses in random nanowire networks: Emergence of critical connectivity pathways. *Journal of Applied Physics* **124** (15), 152118 (2018). doi:[10.1063/1.5037817](https://doi.org/10.1063/1.5037817) .
- [30] Diaz-Alvarez, A. *et al.* Emergent dynamics of neuromorphic nanowire networks. *Scientific Reports* **9** (1), 14920 (2019). doi:[10.1038/s41598-019-51330-6](https://doi.org/10.1038/s41598-019-51330-6) .
- [31] Milano, G. *et al.* Brain-Inspired Structural Plasticity through Reweighting and Rewiring in Multi-Terminal Self-Organizing Memristive Nanowire Networks. *Advanced Intelligent Systems* **2** (8), 2000096 (2020). doi:[10.1002/aisy.202000096](https://doi.org/10.1002/aisy.202000096) .
- [32] Li, Q. *et al.* Dynamic Electrical Pathway Tuning in Neuromorphic Nanowire Networks. *Advanced Functional Materials* **30** (43), 2003679 (2020). doi:[10.1002/adfm.202003679](https://doi.org/10.1002/adfm.202003679) .
- [33] Diaz-Alvarez, A., Higuchi, R., Li, Q., Shingaya, Y. & Nakayama, T. Associative routing through neuromorphic nanowire networks. *AIP Advances* **10** (2), 025134 (2020). doi:[10.1063/1.5140579](https://doi.org/10.1063/1.5140579) .
- [34] Maass, W. On the Computational Power of Winner-Take-All. *Neural Computation* **12** (11), 2519–2535 (2000). doi:[10.1162/089976600300014827](https://doi.org/10.1162/089976600300014827) .

- [35] Sheldon, F. C. & Di Ventra, M. Conducting-insulating transition in adiabatic memristive networks. *Physical Review E* **95** (1), 012305 (2017). doi:[10.1103/PhysRevE.95.012305](https://doi.org/10.1103/PhysRevE.95.012305) .
- [36] Bertschinger, N. & Natschläger, T. Real-Time Computation at the Edge of Chaos in Recurrent Neural Networks. *Neural Computation* **16** (7), 1413–1436 (2004). doi:[10.1162/089976604323057443](https://doi.org/10.1162/089976604323057443) .
- [37] Legenstein, R. & Maass, W. Edge of chaos and prediction of computational performance for neural circuit models. *Neural Networks* **20** (3), 323–334 (2007). doi:[10.1016/j.neunet.2007.04.017](https://doi.org/10.1016/j.neunet.2007.04.017) .
- [38] Sillin, H. O. *et al.* A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology* **24** (38), 384004 (2013). doi:[10.1088/0957-4484/24/38/384004](https://doi.org/10.1088/0957-4484/24/38/384004) .
- [39] Kuncic, Z. *et al.* *Neuromorphic Information Processing with Nanowire Networks*, 1–5 (2020).
- [40] Fu, K. *et al.* *Reservoir Computing with Neuromemristive Nanowire Networks*, 1–8 (2020).
- [41] Zhu, R. *et al.* *Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning*, 102–106 (2020).
- [42] Schreiber, T. Measuring Information Transfer. *PHYSICAL REVIEW LETTERS* **85** (2), 4 (2000) .
- [43] Lizier, J. T., Prokopenko, M. & Zomaya, A. Y. Local information transfer as a spatiotemporal filter for complex systems. *Physical Review E* **77** (2), 026110 (2008). doi:[10.1103/PhysRevE.77.026110](https://doi.org/10.1103/PhysRevE.77.026110) .
- [44] Vicente, R., Wibral, M., Lindner, M. & Pipa, G. Transfer entropy—a model-free measure of effective connectivity for the neurosciences. *Journal of Computational Neuroscience* **30** (1), 45–67 (2011). doi:[10.1007/s10827-010-0262-3](https://doi.org/10.1007/s10827-010-0262-3) .
- [45] Lizier, J. T., Prokopenko, M. & Zomaya, A. Y. Local measures of information storage in complex distributed computation. *Information Sciences* **208**, 39–54 (2012). doi:[10.1016/j.ins.2012.04.016](https://doi.org/10.1016/j.ins.2012.04.016) .

- [46] Lizier, J. T. JIDT: An Information-Theoretic Toolkit for Studying the Dynamics of Complex Systems. *Frontiers in Robotics and AI* **1** (2014). doi:[10.3389/frobt.2014.00011](https://doi.org/10.3389/frobt.2014.00011) .
- [47] Wibral, M., Lizier, J. T., Vögler, S., Priesemann, V. & Galuske, R. Local active information storage as a tool to understand distributed neural information processing. *Frontiers in Neuroinformatics* **8** (2014). doi:[10.3389/fninf.2014.00001](https://doi.org/10.3389/fninf.2014.00001) .
- [48] Bossomaier, T., Barnett, L., Harré, M. & Lizier, J. T. *An Introduction to Transfer Entropy: Information Flow in Complex Systems* (Springer, Cham, Switzerland, 2016).
- [49] Hjelm, R. D. *et al.* Learning deep representations by mutual information estimation and maximization. *arXiv:1808.06670 [cs, stat]* (2019). [arXiv:1808.06670](https://arxiv.org/abs/1808.06670) [cs, stat].
- [50] Löwe, S., O'Connor, P. & Veeling, B. Putting An End to End-to-End: Gradient-Isolated Learning of Representations. *Advances in Neural Information Processing Systems* **32**, 3039–3051 (2019) .
- [51] Ito, S. *et al.* Extending Transfer Entropy Improves Identification of Effective Connectivity in a Spiking Cortical Network Model. *PLoS ONE* **6** (11), e27431 (2011). doi:[10.1371/journal.pone.0027431](https://doi.org/10.1371/journal.pone.0027431) .
- [52] Obst, O., Boedecker, J. & Asada, M. in *Improving Recurrent Neural Network Performance Using Transfer Entropy* (eds Wong, K. W., Mendis, B. S. U. & Bouzerdoun, A.) *Neural Information Processing. Models and Applications*, Vol. 6444 193–200 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2010).
- [53] Lizier, J., Prokopenko, M. & Zomaya, A. The Information Dynamics of Phase Transitions in Random Boolean Networks. *Proc. Eleventh International Conference on the Simulation and Synthesis of Living Systems (ALife XI)* (2008) .
- [54] Lizier, J. T., Pritam, S. & Prokopenko, M. Information dynamics in small-world Boolean networks. *Artificial Life* **17** (4), 293–314 (2011). doi:[10.1162/artl_a_00040](https://doi.org/10.1162/artl_a_00040) .
- [55] Barnett, L., Lizier, J. T., Harré, M., Seth, A. K. & Bossomaier, T. Information Flow in a Kinetic Ising Model Peaks in the Disordered Phase. *Physical Review Letters* **111** (17), 177203 (2013). doi:[10.1103/physrevlett.111.177203](https://doi.org/10.1103/physrevlett.111.177203) .
- [56] Sootla, S., Theis, D. & Vicente, R. Analyzing Information Distribution in Complex Systems. *Entropy* **19** (12), 636 (2017). doi:[10.3390/e19120636](https://doi.org/10.3390/e19120636) .

- [57] Marinazzo, D., Angelini, L., Pellicoro, M. & Stramaglia, S. Synergy as a warning sign of transitions: The case of the two-dimensional Ising model. *Physical Review E* **99** (4), 040101 (2019). doi:[10.1103/PhysRevE.99.040101](https://doi.org/10.1103/PhysRevE.99.040101) .
- [58] Boedecker, J., Obst, O., Lizier, J. T., Mayer, N. M. & Asada, M. Information processing in echo state networks at the edge of chaos. *Theory in Biosciences* **131** (3), 205–213 (2012). doi:[10.1007/s12064-011-0146-8](https://doi.org/10.1007/s12064-011-0146-8) .
- [59] Mediano, P. A. & Shanahan, M. Balanced information storage and transfer in modular spiking neural networks. *arXiv preprint arXiv:1708.04392* (2017) .
- [60] Priesemann, V. *et al.* Self-organization of information processing in developing neuronal networks. *BMC Neuroscience* **16 Suppl 1**, P221 (2015). doi:[10.1186/1471-2202-16-s1-p221](https://doi.org/10.1186/1471-2202-16-s1-p221) .
- [61] Li, M. *et al.* Transitions in information processing dynamics at the whole-brain network level are driven by alterations in neural gain. *PLOS Computational Biology* **15** (10), e1006957 (2019). doi:[10.1371/journal.pcbi.1006957](https://doi.org/10.1371/journal.pcbi.1006957) .
- [62] Cramer, B. *et al.* Control of criticality and computation in spiking neuromorphic networks with plasticity. *Nature Communications* **11** (1), 2853 (2020). doi:[10.1038/s41467-020-16548-3](https://doi.org/10.1038/s41467-020-16548-3) .
- [63] Kuncic, Z. *et al.* Emergent brain-like complexity from nanowire atomic switch networks: Towards neuromorphic synthetic intelligence, 1–3 (2018).
- [64] Bonacich, P. Power and Centrality: A Family of Measures. *American Journal of Sociology* **92** (5), 1170–1182 (1987) .
- [65] Newman, M. *Networks* Vol. 1 (Oxford University Press, 2018).
- [66] Freeman, L. C. A Set of Measures of Centrality Based on Betweenness. *Sociometry* **40** (1), 35 (1977). doi:[10.2307/3033543](https://doi.org/10.2307/3033543) .
- [67] Barrat, A., Barthélemy, M., Pastor-Satorras, R. & Vespignani, A. The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences* **101** (11), 3747–3752 (2004). doi:[10.1073/pnas.0400087101](https://doi.org/10.1073/pnas.0400087101) .
- [68] Brandes, U. & Fleischer, D. in *Centrality Measures Based on Current Flow* (eds Hutchison, D. *et al.*) *STACS 2005*, Vol. 3404 533–544 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2005).

- [69] Lukoševičius, M. & Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review* **3** (3), 127–149 (2009). doi:[10.1016/j.cosrev.2009.03.005](https://doi.org/10.1016/j.cosrev.2009.03.005) .
- [70] Moon, J. *et al.* Temporal data classification and forecasting using a memristor-based reservoir computing system. *Nature Electronics* **2** (10), 480–487 (2019). doi:[10.1038/s41928-019-0313-3](https://doi.org/10.1038/s41928-019-0313-3) .
- [71] Shannon, C. E. A Mathematical Theory of Communication. *Bell System Technical Journal* **27** (3), 379–423 (1948). doi:[10.1002/j.1538-7305.1948.tb01338.x](https://doi.org/10.1002/j.1538-7305.1948.tb01338.x) .
- [72] El Gamal, A. A. & Kim, Y.-H. *Network Information Theory* (Cambridge University Press, Cambridge ; New York, 2011).
- [73] Lizier, J. T., Prokopenko, M. & Zomaya, A. Y. in *A Framework for the Local Information Dynamics of Distributed Computation in Complex Systems* (ed. Prokopenko, M.) *Guided Self-Organization: Inception*, Vol. 9 of *Emergence, Complexity and Computation* 115–158 (Springer, Berlin/Heidelberg, 2014).
- [74] Newman, M. E. J. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America* **103** (23), 8577–8582 (2006). doi:[10.1073/pnas.0601602103](https://doi.org/10.1073/pnas.0601602103) .
- [75] Sosa, M. E., Eppinger, S. D. & Rowles, C. M. A Network Approach to Define Modularity of Components in Complex Products. *Journal of Mechanical Design* **129** (11), 1118–1129 (2007). doi:[10.1115/1.2771182](https://doi.org/10.1115/1.2771182) .
- [76] Rubinov, M. & Sporns, O. Complex network measures of brain connectivity: Uses and interpretations. *NeuroImage* **52** (3), 1059–1069 (2010). doi:[10.1016/j.neuroimage.2009.10.003](https://doi.org/10.1016/j.neuroimage.2009.10.003) .
- [77] Jaeger, H. *Short Term Memory in Echo State Networks* GMD Report (GMD Forschungszentrum Informationstechnik, Sankt Augustin, 2001).
- [78] Verstraeten, D., Dambre, J., Dutoit, X. & Schrauwen, B. *Memory versus non-linearity in reservoirs*, 1–8 (2010).
- [79] Joyce, K. E., Laurienti, P. J., Burdette, J. H. & Hayasaka, S. A New Measure of Centrality for Brain Networks. *PLoS ONE* **5** (8), 13 (2010) .

- [80] Ursino, M., Ricci, G. & Magosso, E. Transfer Entropy as a Measure of Brain Connectivity: A Critical Analysis With the Help of Neural Mass Models. *Frontiers in Computational Neuroscience* **14**, 45 (2020). doi:[10.3389/fncom.2020.00045](https://doi.org/10.3389/fncom.2020.00045) .
- [81] Ceguerra, R. V., Lizier, J. T. & Zomaya, A. Y. *Information storage and transfer in the synchronization process in locally-connected networks*, 54–61 (IEEE, Paris, 2011).
- [82] Meunier, D., Lambiotte, R., Fornito, A., Ersche, K. & Bullmore, E. T. Hierarchical modularity in human brain functional networks. *Frontiers in Neuroinformatics* **3** (2009). doi:[10.3389/neuro.11.037.2009](https://doi.org/10.3389/neuro.11.037.2009) .
- [83] Ferrarini, L. *et al.* Hierarchical functional modularity in the resting-state human brain. *Human Brain Mapping* **30** (7), 2220–2231 (2009). doi:[10.1002/hbm.20663](https://doi.org/10.1002/hbm.20663) .
- [84] Bassett, D. S. *et al.* Dynamic reconfiguration of human brain networks during learning. *Proceedings of the National Academy of Sciences* **108** (18), 7641–7646 (2011). doi:[10.1073/pnas.1018985108](https://doi.org/10.1073/pnas.1018985108) .
- [85] Hüsken, M., Igel, C. & Toussaint, M. Task-dependent evolution of modularity in neural networks. *Connection Science* **14** (3), 219–229 (2002). doi:[10.1080/09540090208559328](https://doi.org/10.1080/09540090208559328) .
- [86] Shine, J. M., Li, M., Koyejo, O., Fulcher, B. & Lizier, J. T. Topological Augmentation of Latent Information Streams in Feed-Forward Neural Networks. Preprint, Neuroscience (2020).
- [87] Klos, C., Kalle Kossio, Y. F., Goedeke, S., Gilra, A. & Memmesheimer, R.-M. Dynamical Learning of Dynamics. *Physical Review Letters* **125** (8), 088103 (2020). doi:[10.1103/PhysRevLett.125.088103](https://doi.org/10.1103/PhysRevLett.125.088103) .
- [88] Butcher, J., Verstraeten, D., Schrauwen, B., Day, C. & Haycock, P. Reservoir computing and extreme learning machines for non-linear time-series data analysis. *Neural Networks* **38**, 76–89 (2013). doi:[10.1016/j.neunet.2012.11.011](https://doi.org/10.1016/j.neunet.2012.11.011) .
- [89] Büsing, L., Schrauwen, B. & Legenstein, R. Connectivity, Dynamics, and Memory in Reservoir Computing with Binary and Analog Neurons. *Neural Computation* **22** (5), 1272–1311 (2010). doi:[10.1162/neco.2009.01-09-947](https://doi.org/10.1162/neco.2009.01-09-947) .
- [90] Legenstein, R. & Maass, W. *What makes a dynamical system computationally powerful?*, 127–154 (MIT Press, 2007), 1 edn.

- [91] Snyder, D., Goudarzi, A. & Teuscher, C. *Finding Optimal Random Boolean Networks for Reservoir Computing*, 259–266 (MIT Press, 2012).
- [92] Snyder, D., Goudarzi, A. & Teuscher, C. Computational capabilities of random automata networks for reservoir computing. *Physical Review E* **87** (4), 042808 (2013). doi:[10.1103/PhysRevE.87.042808](https://doi.org/10.1103/PhysRevE.87.042808) .
- [93] Zhu, R. *et al.* MNIST classification using Neuromorphic Nanowire Networks. *Under Review* 4 (2021) .
- [94] Chung-Wen Ho, Ruehli, A. & Brennan, P. The modified nodal approach to network analysis. *IEEE Transactions on Circuits and Systems* **22** (6), 504–509 (1975). doi:[10.1109/TCS.1975.1084079](https://doi.org/10.1109/TCS.1975.1084079) .
- [95] Dorfler, F., Simpson-Porco, J. W. & Bullo, F. Electrical Networks and Algebraic Graph Theory: Models, Properties, and Applications. *Proceedings of the IEEE* **106** (5), 977–1005 (2018). doi:[10.1109/JPROC.2018.2821924](https://doi.org/10.1109/JPROC.2018.2821924) .
- [96] Pershin, Y. V., Slipko, V. A. & Di Ventra, M. Complex dynamics and scale invariance of one-dimensional memristive networks. *Phys. Rev. E* **87**, 022116 (2013). doi:[10.1103/PhysRevE.87.022116](https://doi.org/10.1103/PhysRevE.87.022116) .
- [97] Hochstetter, J. *et al.* Avalanches and edge of chaos learning in neuromorphic nanowire networks. *Nature Communications (under review)* (2021) .
- [98] Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008** (10), P10008 (2008). doi:[10.1088/1742-5468/2008/10/P10008](https://doi.org/10.1088/1742-5468/2008/10/P10008), [arXiv:0803.0476](https://arxiv.org/abs/0803.0476) .
- [99] Ronhovde, P. & Nussinov, Z. Multiresolution community detection for megascale networks by information-based replica correlations. *Physical Review E* **80** (1), 016109 (2009). doi:[10.1103/PhysRevE.80.016109](https://doi.org/10.1103/PhysRevE.80.016109) .
- [100] Kraskov, A., Stögbauer, H. & Grassberger, P. Estimating mutual information. *Physical Review E* **69** (6), 066138 (2004). doi:[10.1103/PhysRevE.69.066138](https://doi.org/10.1103/PhysRevE.69.066138) .
- [101] Frenzel, S. & Pompe, B. Partial Mutual Information for Coupling Analysis of Multivariate Time Series. *Physical Review Letters* **99** (20), 204101 (2007). doi:[10.1103/PhysRevLett.99.204101](https://doi.org/10.1103/PhysRevLett.99.204101) .

-
- [102] Gómez-Herrero, G. *et al.* Assessing Coupling Dynamics from an Ensemble of Time Series. *Entropy* **17** (4), 1958–1970 (2015). doi:[10.3390/e17041958](https://doi.org/10.3390/e17041958) .

Chapter 4

Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning

“Experience is the teacher of all things.”

Julius Caesar

This chapter is published as:

Zhu, R., Hochstetter, J., Loeffler, A., Diaz-Alvarez, A., Stieg, A., Gimzewski, J., Nakayama, T., Kuncic, Z., 2020. Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning, in: 2020 International Conference on Rebooting Computing (ICRC). pp. 102–106. <https://doi.org/10.1109/ICRC2020.2020.00007>

Abstract

Nanowire networks (NWNs) represent a unique hardware platform for neuromorphic information processing. In addition to exhibiting synapse-like resistive switching memory at their cross-point junctions, their self-assembly confers a neural network-like topology to their electrical circuitry, something that is impossible to achieve through conventional top-down fabrication approaches. In addition to their low power requirements, cost effectiveness and efficient interconnects, neuromorphic NWNs are also fault-tolerant and self-healing. These highly attractive properties can be largely attributed to their complex network connectivity, which enables a rich repertoire of adaptive nonlinear dynamics, including edge-of-chaos criticality. Here, we show how the adaptive dynamics intrinsic to neuromorphic NWNs can be harnessed to achieve transfer learning. We demonstrate this through simulations of a reservoir computing implementation in which NWNs perform the well-known benchmarking task of Mackey-Glass (MG) signal forecasting. First we show how NWNs can predict MG signals with arbitrary degrees of unpredictability (i.e. chaos). We then show that NWNs pre-exposed to a MG signal perform better in forecasting than NWNs without prior experience of an MG signal. This type of transfer learning is enabled by the network's collective memory of previous states. Overall, their adaptive signal processing capabilities make neuromorphic NWNs promising candidates for emerging real-time applications in IoT devices in particular, at the far edge.

4.1 Introduction

The field of neuromorphic engineering is widely recognised as the realisation of Carver Mead's original vision for a new type of electronic hardware engineered to mimic information processing in biological nervous systems [1, 2]. Today, the lowest common denominator of virtually all neuromorphic hardware systems is the co-location of memory and processing units (i.e. non-von Neumann architecture). This minimal neuromorphic feature alone has dramatically improved power efficiency in training various artificial neural network (ANN) models [3].

A higher level neuromorphic attribute is the ability to learn and while ANN models demonstrate learning in software, learning in hardware is desirable for next-generation stand-alone cognitive devices, especially at the IoT edge [4]. In hardware, spike-based learning has been successfully implemented in conventional silicon CMOS technology (e.g. [5–8]). Beyond silicon, nanoelectronic materials with intrinsic neuromorphic properties, including memory and the ability to emulate synaptic connections [3, 9], have attracted enormous attention for on-chip learning [10]. In particular, resistive switching memory (memristive) devices [11–14] are leading candidates for efficient neuromorphic computing architectures, with demonstrated neuromorphic learning functionalities such as short-term and long-term potentiation (STP/LTP) and spike-timing dependent plasticity (STDP) [15–18].

At the device level, neuromorphic functionalities can be broadly attributed to modification of electronic transport mechanisms by nanoscale geometric confinement, usually across a metal-insulator-metal (MIM) junction [19]. Importantly, synapse-like memristive switching is observed not just in memristors fabricated from conventional bulk materials (e.g. metal oxides), but also in neuromorphic systems self-assembled from nanomaterials using bottom-up techniques [10]. Here, we focus on self-assembled metallic nanowires because not only do they form memristive switching MIM junctions, but they also form a complex neural-like network topology, with all-in-one connectivity properties such as small-worldness, modularity and recurrent feedback loops [20–27]. The unique neuromorphic topology of self-assembled nanowire networks (NWNs) is responsible for collective functionalities emerging from the interplay between network connectivity and synaptic nonlinear dynamics [25, 28–32].

Learning in NWN hardware does not require implementation of an ANN model, as has been demonstrated with associative memory tasks [33, 34] and with temporal information processing tasks using a reservoir computing approach, where the network self-regulates in response to continuous-time input signals and only the readout is trained [29, 35, 36]. Varying spatio-temporal input signals (i.e. delivered via different contact electrodes and

with time-varying amplitudes) results in the formation of new electrical pathways, analogous to synaptogenetic learning [37, 38]. Here, we show that NWNs with prior experience of a complex, nonlinear time-series signal can perform better in forecasting the signal than a NWN without prior exposure, thus demonstrating capacity for transfer learning, an important attribute for general intelligence (see [39] for a recent comprehensive review).

4.2 Methods

4.2.1 Modelling network connectivity and memristive junctions

We performed simulations using a physically motivated model based on polymer-coated Ag nanowires that self-assemble into a complex network [25, 31]. Self-assembly was modelled by distributing individual nanowires on a 2D plane, with uniformly random positions and orientations, and with lengths uniformly sampled from a gamma distribution (mean $100\ \mu\text{m}$, stdev $10\ \mu\text{m}$). The variance in nanowire length is based on experimental observations [20, 21, 25, 33] and increases the probability of forming cross-point junctions between overlapping nanowires. This mimics biological neural networks, in which individual neurons can each make several thousand synaptic connections to neighbouring neurons. In our model of self-assembled nanowire networks (NWNs), a range of nanowire connectivities are possible for a fixed number of nanowires. Importantly, the resulting network structure is more complex than a purely random topology or fully connected network (Fig. 4.1), with sparseness and recurrence characteristics that are responsible for efficient signal transduction and emergent cognitive function in biological neural networks [27, 40, 41]. It is also noteworthy that the complex network topology of self-assembled networks differs from the bipartite structure used in ANN models.

Nanowire-nanowire cross-points were modelled as voltage-controlled memristive junctions described by a state-dependent Ohm's law, $I = G(\lambda)V$, where the conductance $G(\lambda)$ is a function of the state variable $\lambda(t)$ that depends on the past history of voltage input. Physically, $\lambda(t)$ parameterises the evolution of a conductive filament that forms across the MIM junction above a threshold bias. For polymer-coated Ag nanowires, the polymer is electrically insulating, but ionically conducting, so Ag^+ cations can migrate across the biased junction [42]. The conducting atomic filament that forms in this way switches the junction from a high-resistance “off” state, to a low-resistance “on” state, when $\lambda \geq \lambda_{\text{crit}}$, where λ_{crit} is a threshold. As the polymer thickness ($\approx 2 - 3\ \text{nm}$) is comparable to the Fermi length of Ag ($\approx 0.5\ \text{nm}$), resistive switching is modelled as a change in the junction conductance

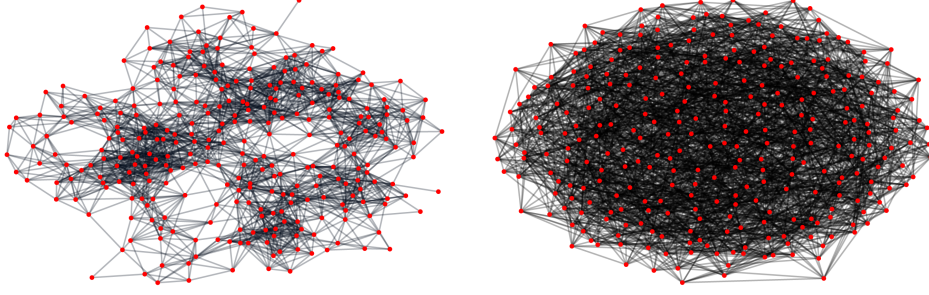


Fig. 4.1 Graph representations of 300-node networks: left – self-assembled nanowire network (2434 edge junctions, average degree 16, small-world propensity 0.67); right – random network (2400 junctions, average degree 16, small-world propensity 0.29). Nodes in red, edges in black.

state $G(\lambda)$ by an amount equal to the conductance quanta $G_0 = (13\text{ k}\Omega)^{-1}$, consistent with measurements of individual nanowire junctions [43]. The corresponding resistance states are $R_{\text{on}} = G_0^{-1}$ and $R_{\text{off}} = \zeta R_{\text{on}}$, with $\zeta = 10^4$ used in the simulation results presented here. Additionally, the Simmons formula is used to model the low voltage tunneling regime in $G(\lambda)$ when the conductive filament is close to the opposite nanowire [44]. Network conductance is calculated using a modified nodal analysis [45] to solve Kirchoff's circuit law equations at each time point.

4.2.2 Mackey–Glass time series prediction

Reservoir computing was implemented on a network with $N = 100$ nanowire nodes and 577 memristive junctions. The Mackey–Glass (MG) signal was delivered to one source node as input voltage bias relative to a drain node. MG signals with varying time delays $\tau \geq 17$ were predicted, with $\tau = 17$ corresponding to the onset of chaotic dynamics.

The network state as a function of time t is represented by instantaneous voltage on all the N nodes. The MG signal at a future time $u_{t+\delta t}$ was predicted using a subset of $n = 10$ node states weighted by a vector w :

$$u_{t+\delta t} = w \cdot \eta_t, \quad (4.1)$$

where η_t is an 11-element vector that includes a 1 V linear shift element and where w was determined by least squares regression using all past states of the n nodes and the input (teacher) signal in the time interval $t \in [0, T]$, i.e.

$$[u_{\delta t}, u_{\delta t+1}, \dots, u_T] = w \cdot [\eta_0, \eta_1, \dots, \eta_{T-\delta t}]. \quad (4.2)$$

A history length of $T > \tau$ was used to train the n output weights and the prediction step was set to $\delta t = \tau$. Accuracy of the prediction task was calculated as

$$\text{Accuracy} = 1 - \text{RNMSE} \quad (4.3)$$

where RNMSE is the root-normalised mean square error. Statistical uncertainties were determined by randomly selecting the $n = 10$ readout nodes for 100 simulations and averaging the accuracy.

4.2.3 Transfer learning

In conventional reservoir computing, the initial state of the network is homogeneous and for the MG prediction task described above, we set $\eta_0 = 0$. We modified the task by first exposing the network to a source MG signal with delay parameter τ_1 before training and predicting a second target MG signal with delay parameter τ_2 . Delivering the source MG (τ_1) signal for 1.5 s effectively primed the network to an initial state $\eta'_0 \neq 0$ that has memory of previous states associated with the source MG signal. This is analogous to transfer learning methods applied to ANN models, where synaptic weights are trained on a source domain and the knowledge gained is transferred to a different, but related, target domain [39]. Our case is somewhat different as the network dynamically self-adjusts its own synaptic junction states during the priming period (since in reservoir computing, only the output weights are trained, not the network weights). We compared the accuracy in predicting the MG target signal to that obtained for a network without prior exposure to the source MG signal during a pre-training period.

4.3 Results

4.3.1 Adaptive dynamics

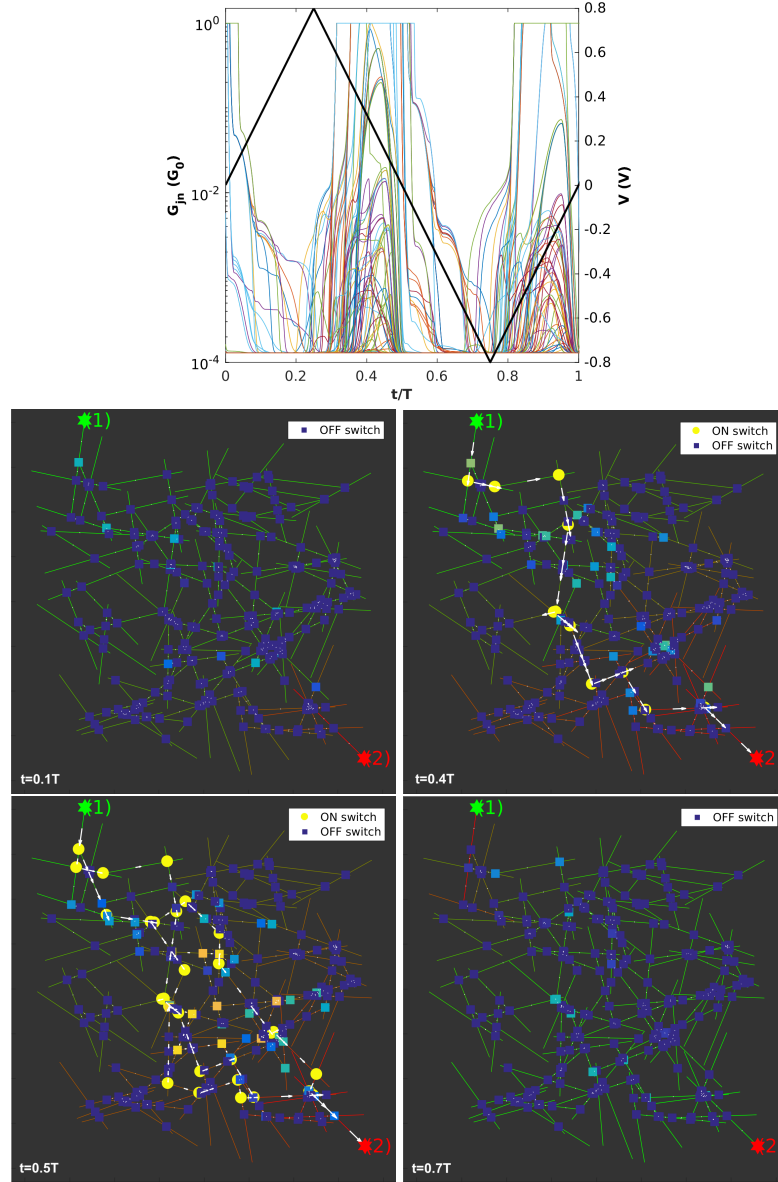


Fig. 4.2 Top panel – Individual memristive junction conductances G_{jn} (in units of conductance quanta G_0) as a function of time t (in units of total simulation time T) for a triangular voltage signal (black) input to a 261-junction NWN. Bottom panel – snapshots of the network at four sequential time points ($t/T = 0.1, 0.4, 0.5, 0.7$) with colourbar indicating G_{jn} . Dark blue junctions denote memristive switches in the off state.

Figure 4.2 shows the NWN response to a triangular input signal. In the top panel, each coloured curve represents the evolution in time of conductance G_{jn} across an individual memristive junction. The network connectivity determines the spatial distribution of voltage at each moment in time. This connectivity influences the voltage-controlled memristive dynamics of each junction, resulting in collective switching as G_{jn} continuously adapts. The bottom panel shows this self-regulation of the synaptic junctions in snapshot visualisations of the network at successive time points during evolution. Brightly coloured circles evident in the frames at $t = 0.4T$ and $t = 0.5T$ represent memristive switches in their on state, with current paths indicated (white). The intrinsic adaptive dynamics of NWNs can in principle be harnessed for information processing. For the parameters used in Fig. 4.2 ($f = 0.75$ Hz, $A = 0.8$ V), the network exhibits “edge-of-chaos” dynamics (e.g. $I - V$ trajectories begin to diverge), which may be optimal for information processing [46].

4.3.2 Mackey–Glass time-series prediction

Fig. 4.3 plots the time series for training and predicting a MG signal with $\tau = 20$. Network output weights w are trained using the first 2400 time steps (i.e. $T = 2.4$ s) after which the signal is predicted 20 steps ahead using eq. (4.1) (i.e. with $\delta t = 0.02$ s in this case). The target signal is overplotted for comparison. The resulting prediction accuracy is 75%. Fig. 4.4 plots the prediction accuracy as a function of $\tau \geq 17$. Accuracy decreases with τ because errors amplify exponentially as the MG signal becomes more chaotic.

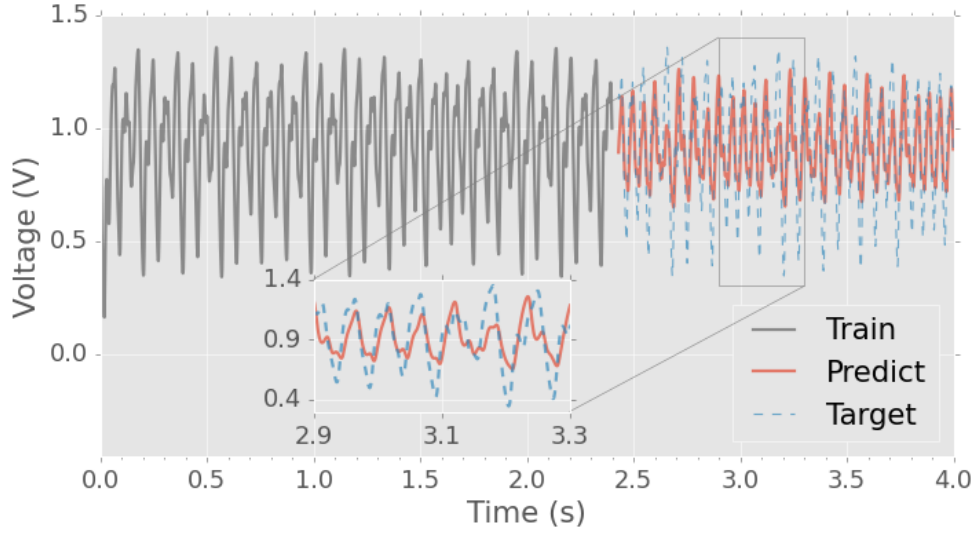


Fig. 4.3 Time series of MG source signal during training (gray), followed by target (dashed blue) and predicted (red) signals for $\tau = 20$. The inset shows a zoom-in of part of the prediction period.

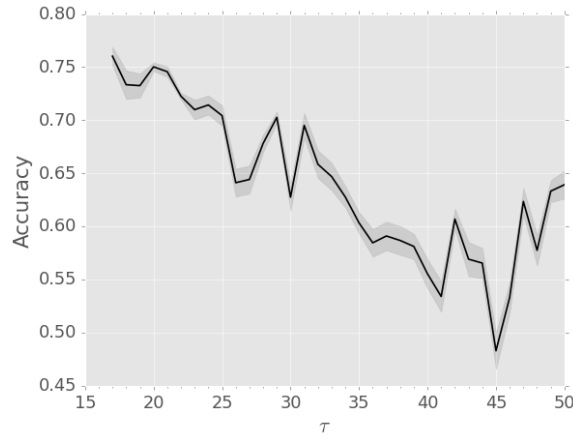


Fig. 4.4 Average MG forecasting accuracy as a function of delay parameter τ . Shading indicates standard error.

4.3.3 Transfer learning

Figure 4.5 plots MG prediction accuracy when the network is primed by a MG signal prior to training. Accuracy is plotted for three different pre-training MG signals ($\tau_1 = 20, 40, 150$) as a function of τ_2 used for MG signal training. For comparison, prediction accuracy without the pre-training (cf. Fig. 4.4) is also overplotted. Accuracy improves when the network is first primed with a MG signal. This demonstrates the principle of transfer learning, where

knowledge is extracted from a source domain and then leveraged for learning in a related target domain.

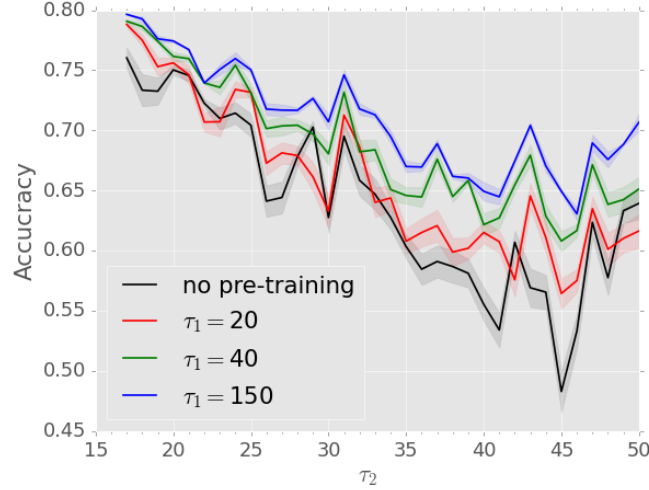


Fig. 4.5 Average MG signal prediction accuracy as a function of signal delay parameter τ_2 of the predicted signal and different τ_1 signals used to prime the network before training. Average accuracy without pre-training is shown for comparison (cf. Fig. 4.4). Shading indicates standard error.

Learning performance is expected to improve especially when there is insufficient information in the target domain compared to the source domain. In this example, prediction accuracy improves more when the network is primed by a source MG signal that is more chaotic (i.e. more degrees of freedom) than the target MG signal (i.e. $\tau_1 > \tau_2$). This is shown by the blue curve (for $\tau_1 = 150$) in Fig. 4.5 and by the accuracy difference heatmap in Fig. 4.6.

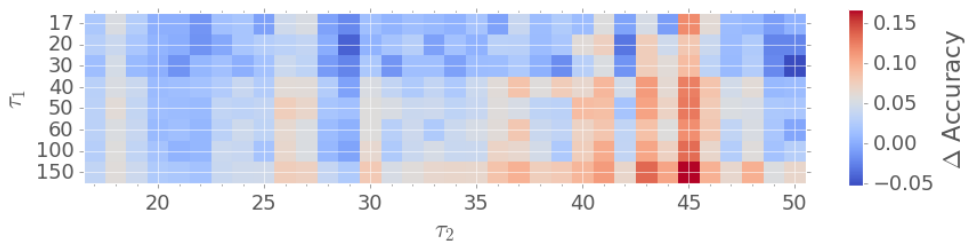


Fig. 4.6 Heatmap showing change in average accuracy in predicting a MG signal with τ_2 when the network is primed using a MG signal with τ_1 relative to prediction without priming.

Importantly, the target MG signal is predicted without relying on any teacher signal for recall. This suggests learning is achieved by harnessing the network's collective memory of past dynamical states. Priming the network before training improves learning by

strengthening the memristive connections in an adaptive way, enabling longer-term memory consolidation.

Prediction accuracy also depends on the instantaneous network state selected for priming. Regardless of the value of τ_2 of the MG signal being predicted, we find accuracy is optimised for a small range of primed network states. This optimal range of states occurs around network activation, coinciding with the formation of a winner-takes-all (WTA) current path (cf. Fig. 4.2, bottom panel). Such WTA gate modules in network circuits are purported to have universal computational power for both digital and analog information processing [47].

4.4 Conclusions

We have demonstrated that the complex interplay between the neural network-like circuitry of nanowire networks and their memristive junctions results in adaptive dynamics, where the network self-regulates to find the optimal signal transduction routes. We showed how the adaptive dynamics can be harnessed for signal processing using a reservoir computing implementation. Prediction of the highly nonlinear Mackey–Glass signal was demonstrated well into the strongly chaotic regime. This has not previously been demonstrated with other memristive reservoir computing approaches. Moreover, we found performance accuracy of this task is improved by transfer learning, where the network is primed by a Mackey–Glass signal before training. Our results show that transfer learning improves performance the most when pre-training with a source signal that is more complex than the target signal to be predicted.

References

- [1] Mead, C. Neuromorphic electronic systems. *Proceedings of the IEEE* **78** (10), 1629–1636 (1990). doi:[10.1109/5.58356](https://doi.org/10.1109/5.58356) .
- [2] Mead, C. How we created neuromorphic engineering. *Nature Electronics* **3** (7), 434–435 (2020). doi:[10.1038/s41928-020-0448-2](https://doi.org/10.1038/s41928-020-0448-2) .
- [3] Zhang, W. *et al.* Neuro-inspired computing chips. *Nature Electronics* **3** (7), 371–382 (2020). doi:[10.1038/s41928-020-0435-7](https://doi.org/10.1038/s41928-020-0435-7) .
- [4] Krestinskaya, O., James, A. P. & Chua, L. O. Neuromemristive circuits for edge computing: A review. *IEEE transactions on neural networks and learning systems* **31** (1), 4–23 (2019) .
- [5] Indiveri, G. *et al.* Neuromorphic Silicon Neuron Circuits. *Frontiers in Neuroscience* **5** (2011). doi:[10.3389/fnins.2011.00073](https://doi.org/10.3389/fnins.2011.00073) .
- [6] Merolla, P. A. *et al.* A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345** (6197), 668–673 (2014). doi:[10.1126/science.1254642](https://doi.org/10.1126/science.1254642) .
- [7] Pfeil, T. *et al.* Six Networks on a Universal Neuromorphic Computing Substrate. *Frontiers in Neuroscience* **7** (2013). doi:[10.3389/fnins.2013.00011](https://doi.org/10.3389/fnins.2013.00011) .
- [8] Wunderlich, T. *et al.* Demonstrating Advantages of Neuromorphic Computation: A Pilot Study. *Frontiers in Neuroscience* **13**, 260 (2019). doi:[10.3389/fnins.2019.00260](https://doi.org/10.3389/fnins.2019.00260) .
- [9] Burr, G. W. *et al.* Neuromorphic computing using non-volatile memory. *Advances in Physics: X* **2** (1), 89–124 (2017). doi:[10.1080/23746149.2016.1259585](https://doi.org/10.1080/23746149.2016.1259585) .
- [10] Sangwan, V. K. & Hersam, M. C. Neuromorphic nanoelectronic materials. *Nature Nanotechnology* **15** (7), 517–528 (2020). doi:[10.1038/s41565-020-0647-z](https://doi.org/10.1038/s41565-020-0647-z) .
- [11] Waser, R. & Aono, M. Nanoionics-based resistive switching memories. *Nature Materials* **6** (11), 833–840 (2007). doi:[10.1038/nmat2023](https://doi.org/10.1038/nmat2023) .
- [12] Zidan, M. A., Strachan, J. P. & Lu, W. D. The future of electronics based on memristive systems. *Nature Electronics* **1** (1), 22–29 (2018). doi:[10.1038/s41928-017-0006-8](https://doi.org/10.1038/s41928-017-0006-8) .

- [13] Ielmini, D. & Wong, H.-S. P. In-memory computing with resistive switching devices. *Nature Electronics* **1** (6), 333–343 (2018). doi:[10.1038/s41928-018-0092-2](https://doi.org/10.1038/s41928-018-0092-2) .
- [14] Wang, Z. *et al.* Resistive switching materials for information processing. *Nature Reviews Materials* **5** (3), 173–195 (2020). doi:[10.1038/s41578-019-0159-3](https://doi.org/10.1038/s41578-019-0159-3) .
- [15] Ohno, T. *et al.* Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials* **10** (8), 591–595 (2011). doi:[10.1038/nmat3054](https://doi.org/10.1038/nmat3054) .
- [16] Serrano-Gotarredona, T., Masquelier, T., Prodromakis, T., Indiveri, G. & Linares-Barranco, B. STDP and STDP variations with memristors for spiking neuromorphic learning systems. *Frontiers in Neuroscience* **7** (2013). doi:[10.3389/fnins.2013.00002](https://doi.org/10.3389/fnins.2013.00002) .
- [17] Serb, A. *et al.* Unsupervised learning in probabilistic neural networks with multi-state metal-oxide memristive synapses. *Nature Communications* **7** (1), 12611 (2016). doi:[10.1038/ncomms12611](https://doi.org/10.1038/ncomms12611) .
- [18] Mehonic, A. *et al.* Memristors—From In-Memory Computing, Deep Learning Acceleration, and Spiking Neural Networks to the Future of Neuromorphic and Bio-Inspired Computing. *Advanced Intelligent Systems* **2** (11), 2000085 (2020). doi:[10.1002/aisy.202000085](https://doi.org/10.1002/aisy.202000085) .
- [19] Pershin, Y. V. & Ventra, M. D. Memory effects in complex materials and nanoscale systems. *Advances in Physics* **60** (2), 145–227 (2011). doi:[10.1080/00018732.2010.544961](https://doi.org/10.1080/00018732.2010.544961) .
- [20] Nirmalraj, P. N. *et al.* Manipulating Connectivity and Electrical Conductivity in Metallic Nanowire Networks. *Nano Letters* **12** (11), 5966–5971 (2012). doi:[10.1021/nl303416h](https://doi.org/10.1021/nl303416h) .
- [21] Bellew, A. T., Bell, A. P., McCarthy, E. K., Fairfield, J. A. & Boland, J. J. Programmability of nanowire networks. *Nanoscale* **6** (16), 9632–9639 (2014). doi:[10.1039/C4NR02338B](https://doi.org/10.1039/C4NR02338B) .
- [22] Avizienis, A. V. *et al.* Neuromorphic Atomic Switch Networks. *PLoS ONE* **7** (8), e42772 (2012). doi:[10.1371/journal.pone.0042772](https://doi.org/10.1371/journal.pone.0042772) .
- [23] Demis, E. C. *et al.* Atomic switch networks—nanoarchitectonic design of a complex system for natural computing. *Nanotechnology* **26** (20), 204003 (2015). doi:[10.1088/0957-4484/26/20/204003](https://doi.org/10.1088/0957-4484/26/20/204003) .

- [24] Milano, G., Porro, S., Valov, I. & Ricciardi, C. Recent Developments and Perspectives for Memristive Devices Based on Metal Oxide Nanowires. *Advanced Electronic Materials* **5** (9), 1800909 (2019). doi:[10.1002/aelm.201800909](https://doi.org/10.1002/aelm.201800909) .
- [25] Diaz-Alvarez, A. *et al.* Emergent dynamics of neuromorphic nanowire networks. *Scientific Reports* **9** (1), 14920 (2019). doi:[10.1038/s41598-019-51330-6](https://doi.org/10.1038/s41598-019-51330-6) .
- [26] Pantone, R. D., Kendall, J. D. & Nino, J. C. Memristive nanowires exhibit small-world connectivity. *Neural Networks* **106**, 144–151 (2018). doi:[10.1016/j.neunet.2018.07.002](https://doi.org/10.1016/j.neunet.2018.07.002) .
- [27] Loeffler, A. *et al.* Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* **14**, 184 (2020). doi:[10.3389/fnins.2020.00184](https://doi.org/10.3389/fnins.2020.00184) .
- [28] Stieg, A. Z. *et al.* Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials* **24** (2), 286–293 (2012). doi:[10.1002/adma.201103053](https://doi.org/10.1002/adma.201103053) .
- [29] Sillin, H. O. *et al.* A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology* **24** (38), 384004 (2013). doi:[10.1088/0957-4484/24/38/384004](https://doi.org/10.1088/0957-4484/24/38/384004) .
- [30] Manning, H. G. *et al.* Emergence of winner-takes-all connectivity paths in random nanowire networks. *Nature Communications* **9** (1), 3219 (2018). doi:[10.1038/s41467-018-05517-6](https://doi.org/10.1038/s41467-018-05517-6) .
- [31] Kuncic, Z. *et al.* Emergent brain-like complexity from nanowire atomic switch networks: Towards neuromorphic synthetic intelligence, 1–3 (2018).
- [32] Milano, G. *et al.* Brain-Inspired Structural Plasticity through Reweighting and Rewiring in Multi-Terminal Self-Organizing Memristive Nanowire Networks. *Advanced Intelligent Systems* **2** (8), 2000096 (2020). doi:[10.1002/aisy.202000096](https://doi.org/10.1002/aisy.202000096) .
- [33] Diaz-Alvarez, A., Higuchi, R., Li, Q., Shingaya, Y. & Nakayama, T. Associative routing through neuromorphic nanowire networks. *AIP Advances* **10** (2), 025134 (2020). doi:[10.1063/1.5140579](https://doi.org/10.1063/1.5140579) .
- [34] Li, Q. *et al.* Dynamic Electrical Pathway Tuning in Neuromorphic Nanowire Networks. *Advanced Functional Materials* **30** (43), 2003679 (2020). doi:[10.1002/adfm.202003679](https://doi.org/10.1002/adfm.202003679) .

- [35] Fu, K. *et al.* *Reservoir Computing with Neuromemristive Nanowire Networks*, 1–8 (2020).
- [36] Kuncic, Z. *et al.* *Neuromorphic Information Processing with Nanowire Networks*, 1–5 (2020).
- [37] Zito, K. & Svoboda, K. Activity-Dependent Synaptogenesis in the Adult Mammalian Cortex. *Neuron* **35** (6), 1015–1017 (2002). doi:[10.1016/S0896-6273\(02\)00903-0](https://doi.org/10.1016/S0896-6273(02)00903-0) .
- [38] Cui, Y., Ahmad, S. & Hawkins, J. Continuous Online Sequence Learning with an Unsupervised Neural Network Model. *Neural Computation* **28** (11), 2474–2504 (2016). doi:[10.1162/NECO_a_00893](https://doi.org/10.1162/NECO_a_00893) .
- [39] Zhuang, F. *et al.* A Comprehensive Survey on Transfer Learning (2020). [arXiv:1911.02685](https://arxiv.org/abs/1911.02685).
- [40] Bullmore, E. & Sporns, O. Complex brain networks: Graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience* **10** (3), 186–198 (2009). doi:[10.1038/nrn2575](https://doi.org/10.1038/nrn2575) .
- [41] Lynn, C. W. & Bassett, D. S. The physics of brain network structure, function and control. *Nature Reviews Physics* **1** (5), 318–332 (2019). doi:[10.1038/s42254-019-0040-8](https://doi.org/10.1038/s42254-019-0040-8) .
- [42] Zhu, J., Zhang, T., Yang, Y. & Huang, R. A comprehensive review on emerging artificial neuromorphic devices. *Applied Physics Reviews* **7** (1), 011312 (2020). doi:[10.1063/1.5118217](https://doi.org/10.1063/1.5118217) .
- [43] Terabe, K., Hasegawa, T., Nakayama, T. & Aono, M. Quantized conductance atomic switch. *Nature* **433** (7021), 47–50 (2005). doi:[10.1038/nature03190](https://doi.org/10.1038/nature03190) .
- [44] Simmons, J. G. Generalized Formula for the Electric Tunnel Effect between Similar Electrodes Separated by a Thin Insulating Film .
- [45] Chung-Wen Ho, Ruehli, A. & Brennan, P. The modified nodal approach to network analysis. *IEEE Transactions on Circuits and Systems* **22** (6), 504–509 (1975). doi:[10.1109/TCS.1975.1084079](https://doi.org/10.1109/TCS.1975.1084079) .
- [46] Bertschinger, N. & Natschläger, T. Real-Time Computation at the Edge of Chaos in Recurrent Neural Networks. *Neural Computation* **16** (7), 1413–1436 (2004). doi:[10.1162/089976604323057443](https://doi.org/10.1162/089976604323057443) .

- [47] Maass, W. On the Computational Power of Winner-Take-All. *Neural Computation* **12** (11), 2519–2535 (2000). doi:[10.1162/089976600300014827](https://doi.org/10.1162/089976600300014827) .

Chapter 5

MNIST classification using Neuromorphic Nanowire Networks

“博学之，审问之，慎思之，明辨之，笃行之。”

“Learn extensively, inquire earnestly, think profoundly, discriminate clearly and practice sincerely.”

中庸 Doctrine of the Mean

- 子思 Zi Si

This chapter is published as:

Zhu, R., Loeffler, A., Hochstetter, J., Diaz-Alvarez, A., Nakayama, T., Stieg, A., Gimzewski, J., Lizier, J., Kuncic, Z., 2021. MNIST classification using Neuromorphic Nanowire Networks, in: International Conference on Neuromorphic Systems 2021 (ICONS 2021). Association for Computing Machinery, New York, NY, USA, pp. 1–4.

<https://doi.org/10.1145/3477145.3477162>

Abstract

Neuromorphic Nanowire Networks (NWNs) are a novel class of information processing hardware devices, combining the advantage of memristive cross-point junctions and neural-like complex network topology. In addition to their low operating power, NWNs are also easy to fabricate with bottom-up self-assembly. Here, we implement the MNIST handwritten digit classification task on simulated NWNs to demonstrate their ability to perform complex learning tasks. Using a CNN-inspired kernel method and a reservoir computing framework, our simulation results attain an accuracy of nearly 98%. Moreover, this is achieved using only a fraction of the total MNIST training data and without requiring hardware accelerators. We also investigate the information theoretic metrics of mutual information (MI), transfer entropy (TE) and active information storage (AIS) to analyse the MNIST learning dynamics of NWNs. We find that MI with respect to classes is maximised after network feature extraction and that TE is largest when MNIST digit boundaries are processed, while AIS is strongest when areas with lower pixel values are presented. Overall, these results suggest the information processing capabilities of neuromorphic NWNs make them promising candidates for complex learning applications.

5.1 Introduction

This study concerns a specific class of memristive neuromorphic systems constructed using bottom-up self-assembly: nanowire networks (NWNs) [1]. Neuromorphic NWNs are disordered systems comprised of passivated metallic nanowires that self-assemble to form a complex network topology with memristive cross-point junctions [2, 3]. Previous studies have demonstrated that NWNs exhibit complex, neural network-like structural properties including small-worldness, modularity and recurrent feedback loops [4]. In response to external stimuli, NWNs exhibit emergent collective dynamics [2, 3, 5–9], which have been shown to be essential for information processing in these neuromorphic devices [3, 7, 10, 11].

Simulation studies of NWNs have demonstrated learning ability under a reservoir computing framework, in which only the readout layer is trained and the network self-adjusts its synapse-like memristive junctions. Moreover, the computing resources within NWNs can be utilised most efficiently when the input is time-dependent [2, 5, 12–14]. The MNIST handwritten digit classification task has become the “hello world” in the field of machine learning [15]. Here, we convert the original MNIST dataset to temporal streams via a shifting kernel and test the performance of NWNs on this task as a benchmark. Furthermore, we use information-theoretic measures to investigate the dynamics within NWNs while learning MNIST digits, to gain deeper insight into the information processing capability of these neuromorphic systems.

5.2 Methods

We use a physically motivated model (detailed elsewhere [12, 16, 17]) to simulate the MNIST classification task implemented on NWNs under a reservoir computing framework. The MNIST dataset is pre-processed and delivered to NWNs temporally to utilise the signal processing capacity of the recurrent memristive network [14]. The training set used in this study is comprised of a randomly selected subset of the original MNIST training dataset. The entire MNIST testing dataset is employed and shuffled to test the classification accuracy. A kernel method inspired by convolutional neural networks [15, 18] is applied to convert the original MNIST image pixel values to temporal input signals delivered to the network.

5.2.1 Kernel scanning

MNIST images are pre-processed with a shifting kernel of size $\kappa \times \kappa$ and duration $\Delta t = 1$ s to extract features. The traverse of the kernel starts from the top left corner of the image,

moving horizontally with a stride of 1 pixel at each Δt until its right boundary reaches the last column of the image. The kernel then moves back to the left most column and shifts one row down from the original position. Fig. 5.1 illustrates this method for a kernel with $\kappa = 5$. Each MNIST image (comprised of 28×28 pixels) is scanned by the kernel $(29 - \kappa)^2$ times in total, and the features re-arranged into a 2D shape as shown in fig. 5.1 (c). At each Δt , the kernel pixel values (fig. 5.1 (b)) are converted into normalised voltages and delivered to dedicated input electrodes in the network (fig. 5.1 (d)).

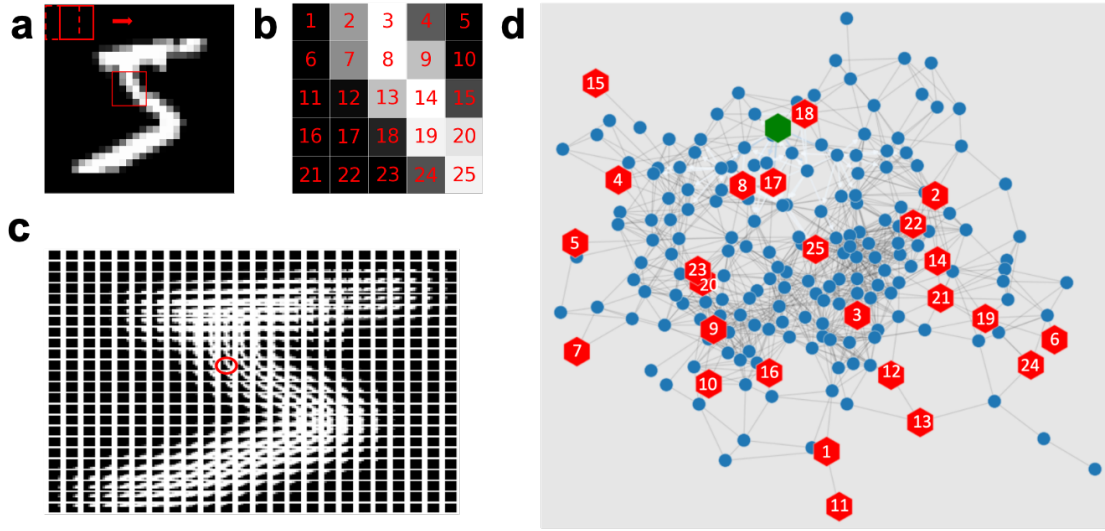


Fig. 5.1 Schematic illustration of the kernel scanning method. Each MNIST image (a) is scanned by a kernel (size 5×5 shown here) (b) that shifts by 1 pixel, thus creating a 2D feature representation (c). Kernel pixel intensities are converted to voltage signals which are (d) input to the network temporally via dedicated electrodes (25 in this case) over a duration of 1 s per kernel pixel.

The role of the network is to project features into a higher dimensional space such that they become linearly separable. Currents on the κ^2 input electrodes are read simultaneously as the input voltages are delivered. Through the learning period of each MNIST image, every electrode has $(29 - \kappa)^2$ readouts, with Δt duration of each readout. The final readout is recorded for each scan of period Δt for each electrode, and re-arranged into κ^2 2D matrices of shape $I_{(29-\kappa) \times (29-\kappa)}$, as shown in fig. 5.2 (a). The most salient features are then extracted by taking the maximum of I in each row and column, as shown in fig. 5.2 (b):

$$F_i = \max(I, \text{axis} = i), \quad i = 1, 2 \quad (5.1)$$

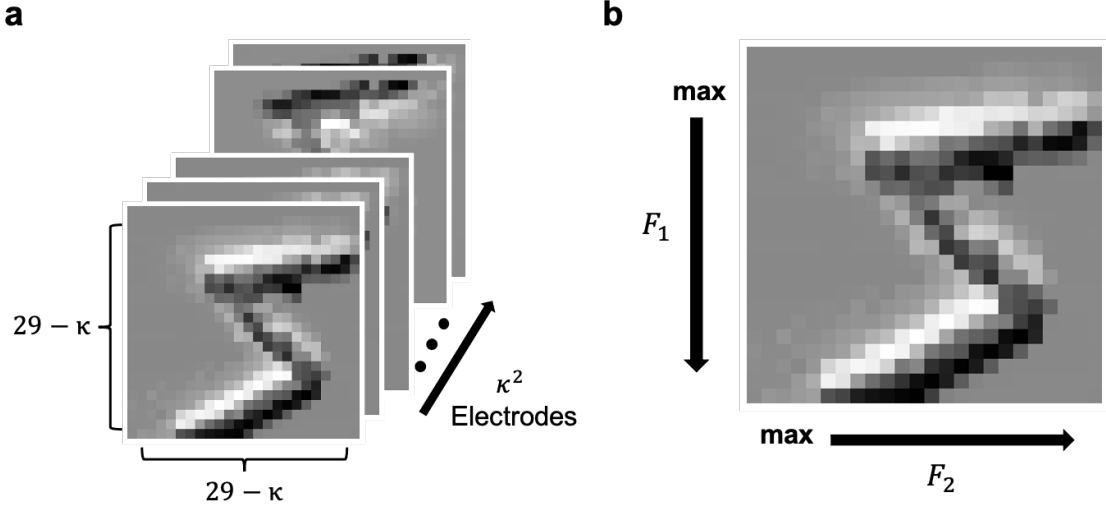


Fig. 5.2 Schematic illustration of feature extraction. (a) During the learning period of each MNIST image, current on κ^2 input electrodes is recorded and re-arranged into 2D matrices $I_{(29-\kappa) \times (29-\kappa)}$. (b) For each electrode, features are sampled as max values along each column (F_1) and each row (F_2).

where F_1 and F_2 are both of length $29 - \kappa$ for each electrode. As a result, $2 \times (29 - \kappa)$ salient features are obtained from each electrode, giving a total of $2\kappa^2(29 - \kappa)$ features per each MNIST image. Linear discriminant analysis (LDA) is then employed to classify the digits based on these features.

5.2.2 Information Dynamics

In addition to demonstrating the system's learning capability via MNIST classification, its performance can be related to an intrinsic information processing capability. We focus on the information theoretic measures of mutual information (MI), transfer entropy (TE) and active information storage (AIS), which provide insights into the intrinsic information transfer and storage of complex networks [19–22], and a way to analyse the dynamics of a network (e.g. recurrent neural network [23]) during learning.

MI estimates the relative information between system variables [24, 25]. In this work, features at different stages are binarised and thresholded to maximise MI. MI is estimated at different stages of the algorithm (i.e. MNIST pixel values, kernel scanned data, features extracted from the network) to determine the influence of each step (details can be found in previous work [25]). TE quantifies the amount of information provided by the past activity of a source component in predicting the target component's state, and captures

directed information flow between these components [19, 22]. In the context of networks, information transfer is calculated between connected nodes [21, 26]. Transfer entropy from node i to node j is calculated as the mutual information from the past of voltage time series i , $\mathbf{V}_i(n)$, to that of j , $\mathbf{V}_j(n+1)$, conditioned on the past k values of j , $\mathbf{V}_j^k(n)$: $\mathbf{TE}_{i,j} = I(\mathbf{V}_i(n); \mathbf{V}_j(n+1) | \mathbf{V}_j^k(n))$. For the edge e connecting nodes i and j , TE is calculated in both directions: $\mathbf{TE}_e = \mathbf{TE}_{i,j} + \mathbf{TE}_{j,i}$. AIS quantifies the amount of information provided by a component's past activity about its own state update, capturing the active memory for computing its next value [20]. In NWNs, memory is incorporated by the conductance on junctions (edges). Active information storage is thereby calculated based on the conductance time-series $\mathbf{G}_e(n)$ of junctions e [21, 26], as the mutual information from the past k values of the conductance $\mathbf{G}_e^k(n)$ to the next value $G_e(n+1)$: $\mathbf{AIS}_e = I(\mathbf{G}_e^k(n); G_e(n+1))$. In this study, TE and AIS are estimated (using the JIDT software [21]) for each Δt period with correspondence to the kernel scans described in fig. 5.1 and averaged across the network.

5.3 Results

5.3.1 MNIST classification

Figure 5.3 (a) shows MNIST classification accuracy with respect to the number of samples in the training group, using a kernel with $\kappa = 5$. Classification accuracy increases as more training samples are used, until saturating at $\approx 96.2\%$. Fig. 5.3 (b) shows the accuracy as a function of κ . Here, both training and testing groups are composed of 10,000 MNIST images. This figure only includes results for $3 \leq \kappa \leq 13$ due to the limitation of network size. When $\kappa = 1$, the kernel delivers data to the network one pixel at a time from the original image, without generating richer features from combinations of neighboring pixels, and there are insufficient features ($2\kappa^2(29 - \kappa) = 56$) to perform classification with high accuracy. On the other hand, when $\kappa = 28$, the kernel delivers the whole image into the system at once. While this process results in many features, the entire image is delivered over Δt , resulting in loss of detail due to the short duration. This makes it impossible to identify the digit. Fig. 5.3 (b) demonstrates that the accuracy saturates to $\approx 97.8\%$ as κ increases, with $\kappa = 9$ being the optimal size of the kernel. At this size, enough features are extracted without losing too many details over the learning period.

The MNIST classification task has been a common benchmark for all types of neural networks and learning algorithms. CNNs have demonstrated an unparalleled accuracy of over 99% while other ANNs are able to achieve an accuracy ranging between 95.3% to

98.4% [15]. These methods require multiple fully connected layers and numerous learning epochs and are thus computationally intensive. Here, a comparable accuracy is achieved by employing a recurrent network with sparse connectivity under the reservoir computing framework. More importantly, these simulation results suggest that NWNs can achieve high classification accuracy using only $\approx 10\%$ of the MNIST training dataset. Furthermore, as all the learning can be accomplished in a single epoch using only CPUs, NWNs are also less computationally intensive than ANNs.

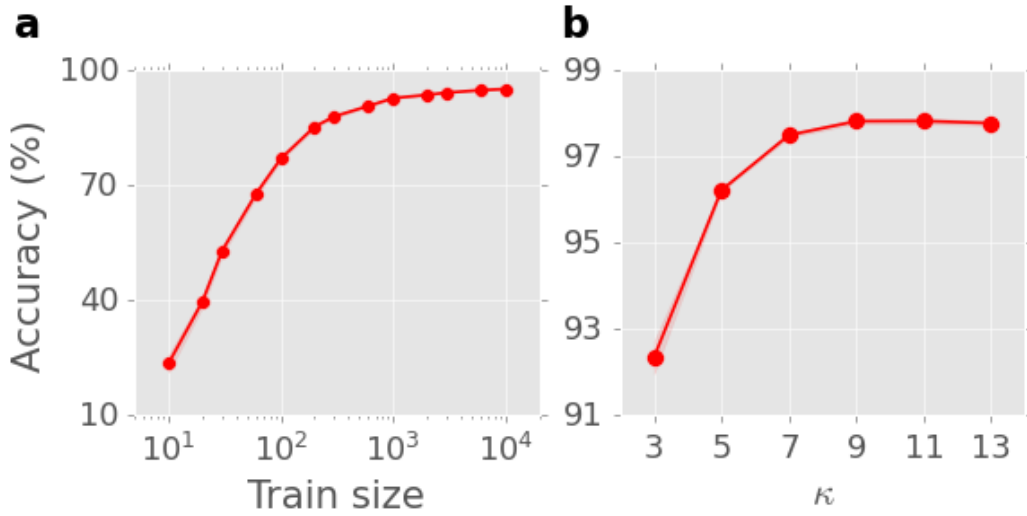


Fig. 5.3 (a) Classification accuracy as a function of number of MNIST images used for training for a kernel size $\kappa = 5$. (b) Classification accuracy as a function of kernel size.

5.3.2 Information dynamics

Table 5.1 lists the MI (natural logarithm) between each MNIST digit class (in 10,000 samples) and the data at different stages. MI increases with each stage of the algorithm and exhibits a marked increase when the most salient features are extracted from the network readouts by Eq. 5.1, indicating significant learning progress. Applying the same feature selection to the kernel scans (without network processing) results in substantially lower MI of 0.122 nats. Fig. 5.4 shows examples of the dynamical information theoretic measures within the neuromorphic system when learning the digits “0” (Fig. 5.4 (a)-(c)) and “9” (Fig. 5.4 (d)-(f)). Fig. 5.4 (a),(d) show the mean of input kernels across different instances of “0” and “9”. These are obtained by selecting out each instance of each digit from the whole time series and then averaging the pixel values within the kernel scans (cf. fig. 5.1 (b)), and then across

instances. Intrinsic information dynamics is then estimated at each Δt with correspondence to kernel scans.

Table 5.1 Mutual Information (nats) with Class

Raw	Scanned	Selected Input	Readout	Features
0.0877	0.115	0.122	0.123	0.163

Fig. 5.4 (b),(e) show TE averaged across instances of “0” and “9” and re-arranged into the same shape as fig. 5.4 (a). These results suggest that information transfer is strongest when the kernel traverses across the horizontal edge of a digit (i.e. TE acts like an edge detector). Input signals delivered from these areas tend to be substantially different from their values in the previous kernel scan, and so drive more complex transient dynamics in the system, hence inducing stronger information flow between the nodes in the network. Similarly, fig. 5.4 (c),(f) show AIS when learning “0” and “9”. In contrast to TE, these results indicate that the network exhibits stronger AIS when the kernel traverses areas with lower pixel values. The network tends to remain in similar stationary states at these points, where its memory of past states dominates the dynamics.

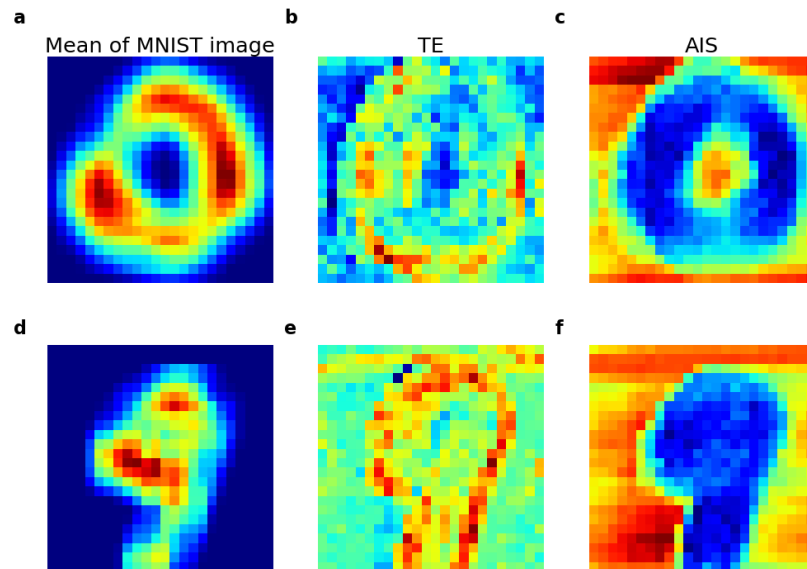


Fig. 5.4 Information dynamics of learning the digits “0” and “9”. (a) Average pixel values within the kernel for digit “0”; (b) Corresponding average TE across the network; (c) Corresponding average AIS across the network. (d)-(f) Same as (a)-(c) except for the digit “9”. For each digit, all three figures are averaged across different learning periods.

5.4 Conclusions

Neuromorphic nanowire networks represent a novel approach for neuro-inspired computing and information processing. In this study, the CNN-inspired kernel method applied to pre-process the MNIST dataset enabled NWNs to achieve an overall high performance on this classification task when implemented in a reservoir computing framework. An accuracy comparable to ANNs can be achieved using less training data and less computing resources. Additionally, information theoretic analysis of NWNs during MNIST learning demonstrated substantially increased mutual information between the extracted features and the classes, compared to raw data and classes. Furthermore, information transfer is maximised near the digit boundaries, while information storage is maximised in regions of lower pixel values. These proof-of-principle simulation results demonstrate the potential of nanowire networks for neuromorphic information processing and provide a useful scheme for hardware implementation of the MNIST classification task as well as more complex dynamical classification tasks such as human action recognition.

References

- [1] Kuncic, Z. & Nakayama, T. Neuromorphic nanowire networks: Principles, progress and future prospects for neuro-inspired information processing. *Advances in Physics: X* **6** (1), 1894234 (2021). doi:[10.1080/23746149.2021.1894234](https://doi.org/10.1080/23746149.2021.1894234) .
- [2] Stieg, A. Z. *et al.* Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials* **24** (2), 286–293 (2012). doi:[10.1002/adma.201103053](https://doi.org/10.1002/adma.201103053) .
- [3] Bellew, A. T., Bell, A. P., McCarthy, E. K., Fairfield, J. A. & Boland, J. J. Programmability of nanowire networks. *Nanoscale* **6** (16), 9632–9639 (2014). doi:[10.1039/C4NR02338B](https://doi.org/10.1039/C4NR02338B) .
- [4] Loeffler, A. *et al.* Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* **14**, 184 (2020). doi:[10.3389/fnins.2020.00184](https://doi.org/10.3389/fnins.2020.00184) .
- [5] Avizienis, A. V. *et al.* Neuromorphic Atomic Switch Networks. *PLoS ONE* **7** (8), e42772 (2012). doi:[10.1371/journal.pone.0042772](https://doi.org/10.1371/journal.pone.0042772) .
- [6] Manning, H. G. *et al.* Emergence of winner-takes-all connectivity paths in random nanowire networks. *Nature Communications* **9** (1), 3219 (2018). doi:[10.1038/s41467-018-05517-6](https://doi.org/10.1038/s41467-018-05517-6) .
- [7] O’Callaghan, C. *et al.* Collective capacitive and memristive responses in random nanowire networks: Emergence of critical connectivity pathways. *Journal of Applied Physics* **124** (15), 152118 (2018). doi:[10.1063/1.5037817](https://doi.org/10.1063/1.5037817) .
- [8] Diaz-Alvarez, A. *et al.* Emergent dynamics of neuromorphic nanowire networks. *Scientific Reports* **9** (1), 14920 (2019). doi:[10.1038/s41598-019-51330-6](https://doi.org/10.1038/s41598-019-51330-6) .
- [9] Milano, G. *et al.* Brain-Inspired Structural Plasticity through Reweighting and Rewiring in Multi-Terminal Self-Organizing Memristive Nanowire Networks. *Advanced Intelligent Systems* **2** (8), 2000096 (2020). doi:[10.1002/aisy.202000096](https://doi.org/10.1002/aisy.202000096) .
- [10] Li, Q. *et al.* Dynamic Electrical Pathway Tuning in Neuromorphic Nanowire Networks. *Advanced Functional Materials* **30** (43), 2003679 (2020). doi:[10.1002/adfm.202003679](https://doi.org/10.1002/adfm.202003679) .

- [11] Diaz-Alvarez, A., Higuchi, R., Li, Q., Shingaya, Y. & Nakayama, T. Associative routing through neuromorphic nanowire networks. *AIP Advances* **10** (2), 025134 (2020). doi:[10.1063/1.5140579](https://doi.org/10.1063/1.5140579) .
- [12] Kuncic, Z. *et al.* *Neuromorphic Information Processing with Nanowire Networks*, 1–5 (2020).
- [13] Zhu, R. *et al.* *Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning*, 102–106 (2020).
- [14] Fu, K. *et al.* *Reservoir Computing with Neuromemristive Nanowire Networks*, 1–8 (2020).
- [15] Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86** (11), 2278–2324 (1998). doi:[10.1109/5.726791](https://doi.org/10.1109/5.726791) .
- [16] Kuncic, Z. *et al.* *Emergent brain-like complexity from nanowire atomic switch networks: Towards neuromorphic synthetic intelligence*, 1–3 (2018).
- [17] Hochstetter, J. *et al.* Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nature Communications* **12** (1), 1–13 (2021). doi:[10.1038/s41467-021-24260-z](https://doi.org/10.1038/s41467-021-24260-z) .
- [18] Fukushima, K. & Miyake, S. in *Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition* 267–285 (Springer, 1982).
- [19] Schreiber, T. Measuring Information Transfer. *Physical Review Letters* **85** (2), 461–464 (2000). doi:[10.1103/PhysRevLett.85.461](https://doi.org/10.1103/PhysRevLett.85.461) .
- [20] Lizier, J. T., Prokopenko, M. & Zomaya, A. Y. Local measures of information storage in complex distributed computation. *Information Sciences* **208**, 39–54 (2012). doi:[10.1016/j.ins.2012.04.016](https://doi.org/10.1016/j.ins.2012.04.016) .
- [21] Lizier, J. T. JIDT: An Information-Theoretic Toolkit for Studying the Dynamics of Complex Systems. *Frontiers in Robotics and AI* **1** (2014). doi:[10.3389/frobt.2014.00011](https://doi.org/10.3389/frobt.2014.00011) .
- [22] Bossomaier, T., Barnett, L., Harré, M. & Lizier, J. T. *An Introduction to Transfer Entropy: Information Flow in Complex Systems* (Springer International Publishing, 2016).

-
- [23] Boedecker, J., Obst, O., Lizier, J. T., Mayer, N. & Asada, M. Information processing in echo state networks at the edge of chaos. *Theory in Biosciences* **131** (3), 205–213 (2012). doi:[10.1007/s12064-011-0146-8](https://doi.org/10.1007/s12064-011-0146-8) .
 - [24] Hjelm, R. D. *et al.* Learning deep representations by mutual information estimation and maximization (2019). [arXiv:1808.06670](https://arxiv.org/abs/1808.06670).
 - [25] Shine, J. M., Li, M., Koyejo, O., Fulcher, B. & Lizier, J. T. Topological Augmentation of Latent Information Streams in Feed-Forward Neural Networks. Preprint (2020).
 - [26] Zhu, R. *et al.* Information dynamics in neuromorphic nanowire networks. *Scientific Reports* **11** (1), 13047 (2021). doi:[10.1038/s41598-021-92170-7](https://doi.org/10.1038/s41598-021-92170-7) .

Chapter 6

Memristive Reservoirs Learn to Learn

“Learning never exhausts the mind.”

Leonardo da Vinci

This chapter is published as:

Zhu, R., Eshraghian, J., Kuncic, Z., 2023. Memristive Reservoirs Learn to Learn, in: Proceedings of the 2023 International Conference on Neuromorphic Systems (ICONS 2023), ACM, Santa Fe NM USA, pp. 1–7. <https://doi.org/10.1145/3589737.3605989>

Abstract

Memristive reservoirs draw inspiration from a novel class of neuromorphic hardware known as nanowire networks. These systems display emergent brain-like dynamics, with optimal performance demonstrated at dynamical phase transitions. In these networks, a limited number of electrodes are available to modulate system dynamics, in contrast to the global controllability offered by neuromorphic hardware through random access memories. We demonstrate that the learn-to-learn framework can effectively address this challenge in the context of optimization. Using the framework, we successfully identify the optimal hyperparameters for the reservoir. This finding aligns with previous research, which suggests that the optimal performance of a memristive reservoir occurs at the ‘edge of formation’ of a conductive pathway. Furthermore, our results show that these systems can mimic membrane potential behaviour observed in spiking neurons, and may serve as an interface between spike-based and continuous processes.

6.1 Introduction

Nanowire networks are a novel class of neuromorphic devices that demonstrate potential for brain-inspired computing and information processing [1–13]. By embedding memristive switching dynamics into naturally-arising neuromorphic connectivity structures [14, 15], these networks display emergent brain-like dynamics [16], including dynamical phase transitions and avalanche criticality [9, 17].

The synaptic sites of nanowire networks are not directly accessible, in contrast to random access memories (RAM) [18–20], where each memory cell is addressable and programmable. The lack of controllability is compensated for by the dynamic nature of nanowire networks, which is a key feature that enables them to adapt to evolving input signals. Nevertheless, it is worth investigating how these neuromorphic systems can be optimised for information processing tasks. For example, previous studies have shown that in a physical reservoir computing framework, nanowire networks can achieve superior learning performance when operating near a dynamical phase transition [8, 9]. Rather than manually exploring the optimal region of operation, an alternative and more effective way to optimise the parameters that can be physically adjusted (namely, inputs and outputs) is the learn-to-learn (L2L) framework, commonly known as meta-learning.

The L2L approach is a scheme for optimizing learning capacity from prior experiences [21]. Recent advances of L2L algorithms are built upon the premise that the learning system – typically an artificial neural network – is differentiable, so that the gradients can be propagated through the network to adjust the hyperparameters (e.g. learning rate) [22–24]. It has also been shown that biologically or physically inspired systems are suitable candidates as learning agents in the L2L framework [25, 26]. In particular, Bohnstingl et al. [27] showed that non-differentiable systems could be optimised using non-gradient-based optimization schemes.

This study demonstrates how learning is achieved using the collective dynamics of a system abstracted from physical nanowire networks (see details in [8, 9]) under a physical reservoir computing (RC) framework [28, 29], namely a memristive reservoir, where training is restricted to the readout layer to circumvent the computation burden of training traditional deep artificial network architectures [30]. The synaptic sites (recurrent weights) in the reservoir are not individually programmable due to the inherent physical properties of the hardware, but this may be offset by the highly rich set of dynamics available in the memristive substrate as a response to external stimuli. The L2L approach is applied to the memristive reservoir and we show that this system is able to learn the dynamics of a

family of nonlinearly filtered signals. Furthermore, we also demonstrate that the memristive reservoir system is able to generate dynamics that resemble the membrane potentials of spiking neural networks (SNNs) directly from continuous inputs, which implies its potential to bridge the gap between continuous signals and spike-based computing paradigms. By learning across various membrane potential dynamics, meta-learning can identify common principles or shared features that govern the dynamics of both spiking neurons and nanowire networks, which may offer deeper insights into the underlying ionic mechanisms that pervade both biology and emerging memory technologies. In particular, a nested-loop structure is implemented as following:

- L2L is used in the outer loop to determine the optimal hyperparameters of the memristive reservoir;
- Linear regression is employed in the inner loop to optimise the stimulus/readout protocol to achieve the task objectives.

Demonstrating how L2L is compatible with systems grounded in physics, such as memristive reservoirs, opens up the potential to optimise learning using adaptive networks where individual internal weights are not trained, but rather allowed to self-adjust in response to dynamical inputs, in a manner similar to the brain's neural network. This has implications beyond memristive reservoirs and could lead to more efficient and effective optimization schemes for learning complex physical phenomena.

6.2 Research Methods

As continuous temporal input voltage signals are delivered to a memristive reservoir via dedicated input nodes, the network autonomously adjusts its internal state. Voltages of a subset of the remaining nodes can be read out and used to train weights in the fully-connected linear output layer. Two L2L task families are studied: learning nonlinear Volterra dynamics and learning membrane dynamics from SNNs.

6.2.1 The L2L Framework

Fig. 7.1 illustrates the L2L framework, where two iterative loops work together to optimise the system's learning performance for a chosen family of tasks $\mathcal{F}$. In this work, the task families are split into two subsets, one used for meta-training and the other for meta-testing.

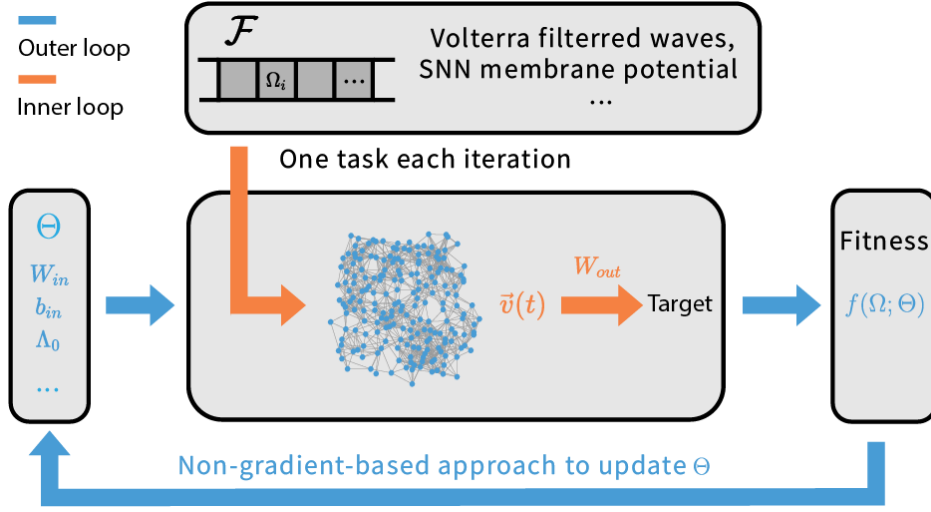


Fig. 6.1 Schematic diagram of the L2L framework. At each iteration of the outer loop, the hyperparameter set $\Theta = \{W_{in}, b_{in}, \Lambda_0, \dots\}$ is perturbed in multiple directions. These perturbed hyperparameters are passed to the inner loop and learning task Ω randomly selected from $\mathcal{F}$ are performed. Θ is then optimised in the outer loop based on the returned fitness of different tasks. The inner loop operations are indicated by orange arrows and the outer loop operations by blue arrows.

The objective of the outer loop (orange in Fig. 7.1) is to optimise the reservoir hyperparameters Θ based on the fitness in the inner loop. In each iteration of the inner loop (blue in Fig. 7.1), the system learns from a specific task Ω_i within the meta-training phase of $\mathcal{F}$ and the fitness $f(\Omega_i; \Theta)$ is evaluated. Non-gradient-based techniques, namely simulated annealing (SA) and evolutionary strategies (ES), are applied to the fitness to determine the optimal hyperparameter set Θ' (or range), thus achieving the best collective learning outcome for $\mathcal{F}$:

$$\Theta' = \underset{\Theta}{\operatorname{argmax}}(f(\Omega; \Theta)), \quad \Omega \in \mathcal{F}. \quad (6.1)$$

After 100 generations of outer-loop training, the performance of the baseline (without meta-learning) and meta-learned reservoirs are evaluated using tasks from the meta-testing set to evaluate the influence of the L2L algorithm on the reservoir's learning performance.

Simulated Annealing

Simulated annealing optimises the objective function by mimicking the physical annealing process [31]. The algorithm is parameterised by a decaying 'temperature' T . At each generation of the outer loop, the hyperparameters are perturbed in multiple directions by

a step size ε drawn from a normal distribution parameterised by T . For each perturbed hyperparameter $\tilde{\Theta}$, the change in fitness ($\Delta f = f(\Omega, \tilde{\Theta}) - f(\Omega; \Theta)$) is estimated. $\tilde{\Theta}$ is accepted if the corresponding fitness improves, otherwise the probability (P) of accepting $\tilde{\Theta}$ is determined by an exponential distribution:

$$P = \begin{cases} 1, & \Delta f \geq 0, \\ e^{\Delta f/T}, & \Delta f < 0. \end{cases} \quad (6.2)$$

When T is high, some sub-optimal solutions can be accepted and the algorithm explores a broader parameter space. As it ‘cools down’, the probability of accepting worse solutions decreases and the solution eventually converges if a global optimum exists.

Evolution Strategies

Evolution strategies are inspired by natural selection and evolution [32, 33]. A population of candidate solutions are generated and evolved based on weighted fitness and a pre-determined learning rate η [34, 35]. At generation k , each candidate $\Theta_{k,i}$ in the population is perturbed by a step size ε_i drawn from a normal distribution $(0, \mathbb{I})$, with standard deviation σ

After evaluating the fitness $f(\Omega_i; \Theta_{k,i} + \sigma \varepsilon_i)$ for all candidates in the same generation, $\Theta_{k,i}$ is evolved by:

$$\Theta_{k+1,i} = \Theta_{k,i} + \frac{\eta}{n\sigma} \sum_{i=1}^n f(\Omega_i; \Theta_{k,i} + \sigma \varepsilon_i) \varepsilon_i, \quad (6.3)$$

where n is the size of the population. The statistical setup enables ES to deal with high-dimensional problems, making it useful for optimization problems with many parameters.

6.2.2 Learning Volterra dynamics

Consider a two-terminal configuration of a memristive reservoir with one source node and one drain node (see [8] for details). Fig. 6.2 shows how the memristive reservoir is used to learn the nonlinear Volterra dynamics (Task 1). As the green box and arrows indicate, the input signal $x(t)$ is delivered to the source node, while the drain node is grounded and an external fully connected output layer linearly combines the voltage read outs $\vec{v}(t)$ of 64 other nodes to regress to each target signal $u(t)$. The objective of the L2L process here is to fine-tune the hyperparameters to achieve optimal regression for the nonlinear time-delayed target signal. The input weight (W_{in}), input bias (b_{in}), and initial reservoir state (Λ_0) are

optimised by L2L via simulated annealing. Λ_0 is parameterised by a pulse of 1 V DC with varying width T_0 applied to the input node prior to the task, as described in [8].

The input signal $x(t)$ is generated by combining two sine waves:

$$x(t) = A_1 \sin(2\pi \frac{t}{T_1} + \phi_1) + A_2 \sin(2\pi \frac{t}{T_2} + \phi_2), \quad (6.4)$$

where $T_1 = 0.323$ s and $T_2 = 0.5$ s, and where $A_1, A_2 \in [0.5, 1]$ and $\phi_1, \phi_2 \in [0, \frac{\pi}{2}]$ are chosen randomly. The target signal $u(t)$ is generated using a second-order Volterra filter:

$$u(t) = \int_{\tau} k_{\Omega}^1(\tau) x(t - \tau) d\tau + \int_{\tau_1} \int_{\tau_2} k_{\Omega}^2(\tau_1, \tau_2) x(t - \tau_1) x(t - \tau_2) d\tau_1 d\tau_2, \quad (6.5)$$

in which $\tau, \tau_1, \tau_2 \in [0, 0.5]$ denote the time delay of the signals, while k_{Ω}^1 and k_{Ω}^2 are task-specific random Volterra kernels (see [26] for further information on generating the kernels). 100 target signals are generated using different Volterra kernels to comprise the family $\mathcal{F}$. The green box in Fig. 6.2 shows an example of one input signal (black) and Volterra-filtered target signals (coloured) with different random kernels.

The readout weights W_{out} are trained using least square to minimise the loss:

$$\mathcal{L} = \|U - W^T V\|_2, \quad (6.6)$$

where $\|\cdot\|_2$ represents the L-2 norm, and $U = [u(1), u(2), \dots, u(t)]$ and $V = [\vec{v}(1), \vec{v}(2), \dots, \vec{v}(t)]$ are the stacked target and readout of the reservoir. The learning results are estimated by normalised root mean squared error (NRMSE):

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{T} \sum_{t=1}^T (W_{out}^T \vec{v}(t) - u(t))^2}}{u_{max} - u_{min}}, \quad (6.7)$$

where T is the length of the signal.

6.2.3 Learning membrane dynamics of SNNs

The blue arrows in Fig. 6.2 illustrate the scheme for this task family. A set of 20 randomly generated Volterra filter signals $\vec{u}(t)$ are utilised as input and converted to spike trains $\vec{u}'(t)$ using delta modulation. A fully connected SNN ($\mathcal{S}$, developed using [36]) with 20 input neurons, 1,000 hidden neurons, and 5 output neurons receives $\vec{u}'(t)$ as input. For each task,

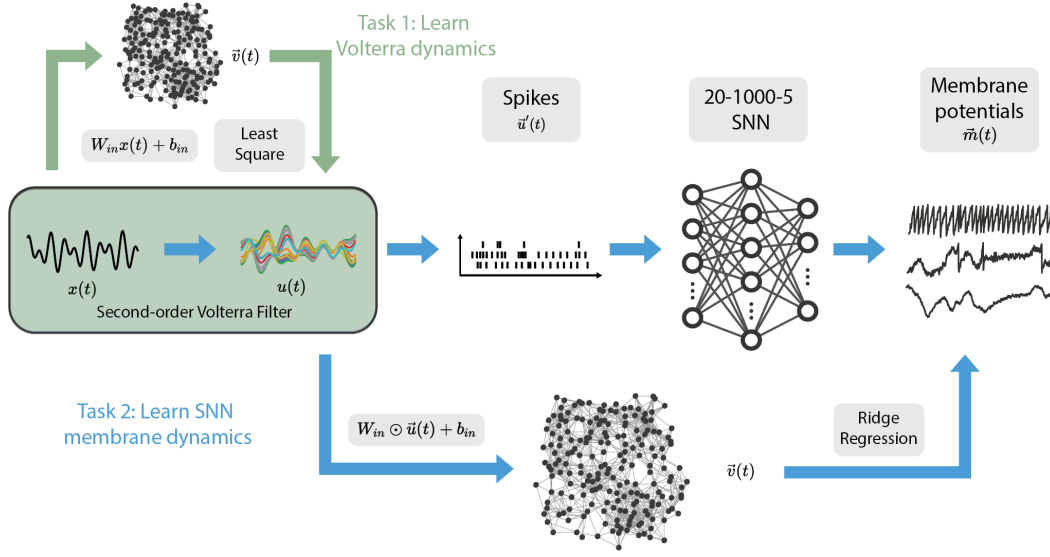


Fig. 6.2 Schematic diagram for the learning tasks employed in this study. The green box and arrows summarise the learning Volterra dynamics task family (Task 1), where the original signal $x(t)$ is the input and the Volterra-filtered signal $u(t)$ is the target for the reservoir. The blue arrows illustrate the flow of the learning SNN dynamics task family (Task 2), in which a set of 20 Volterra-filtered signals $\vec{u}(t)$ is converted to spikes $\vec{u}'(t)$ and delivered to a fully connected SNN. The membrane potentials $\vec{m}(t)$ of the readout neurons are used as target while the set of continuous Volterra-filtered signals are delivered directly to the reservoir as input.

the internal weights $W_{\mathcal{S}}$ of the SNN are randomly generated, and the membrane potentials of the 5 output neurons, $\vec{m}(t)$ are employed as the target signals of the learning task:

$$\vec{m}(t) = \mathcal{S}(\vec{u}'(t); W_{\mathcal{S}}). \quad (6.8)$$

The continuous Volterra signals $\vec{u}(t)$ are delivered to 20 input nodes in the memristive reservoir ($\mathcal{R}$), and 64 nodes are used as voltage readouts:

$$\vec{v}(t) = \mathcal{R}(W_{in} \odot \vec{u}(t) + \vec{b}_{in}; \Lambda_0). \quad (6.9)$$

Notice that W_{in} and $\vec{u}(t)$ have the same dimensions and $\odot$ represents element-wise multiplication.

For each task, the entire data stream is divided into three parts: the first 2000 data points are considered as a transient phase [37]; the subsequent 6000 data points (from 2000th to 8000th) employed as the support set (inner-loop training), and the last 1000 data points (from 8000th to 9000th) comprise the query set (inner-loop testing). The readout weights W_{out} are

trained for each task using ridge regression to minimise the loss function:

$$\mathcal{L} = \|M - W_{out}^T V\|_2^2 + \alpha \|W_{out}\|_2^2, \quad (6.10)$$

where α is a regularization term ranging between 10^{-4} and 1, with $M = [\vec{m}(1), \vec{m}(2), \dots, \vec{m}(t)]$ representing the stacked target membrane potentials. A 5-fold cross validation scheme is employed to avoid overfitting [38]. NRMSE of the query regime is reported as result.

A family $\mathcal{F}$ of 150 tasks is created by utilizing SNNs with different internal weights. The goal of L2L in this context is to adapt the reservoir to new tasks by extrapolating information from past learning experiences, allowing it to generate desired membrane potentials. Simulated annealing and evolution strategies are used separately to optimise the input weight (W_{in}) and the input bias ($\vec{b}_{in}$) of each signal. Λ_0 is fixed to the phase transition regime found in task 1.

6.3 Results

6.3.1 L2L Volterra dynamics

Table 6.1 NRMSE for Task 1 with and without meta-learning

	Meta-train	Meta-test
w/o meta-learning	0.209 ± 0.003	0.211 ± 0.014
Meta-learned (SA)	0.168 ± 0.008	0.164 ± 0.009

Fig. 6.3(a) shows the reservoir's conductance (blue curve) in response to a 1 V DC pulse of varying width T_0 . The shaded region represents the general phase transition regime, identified from previous studies [8], and the dashed line at $T_0 = 2.3$ s indicates the optimal initial reservoir state Λ_0 found by the SA optimization scheme, as shown in Fig. 6.3(b). T_0 (orange) converges toward $\simeq 2.3$ s as NRMSE (green) converges to a minimum after approximately 60 generations. Fig. 6.3(c) compares the system's learning outcomes for a specific task in the Volterra family before (blue) and after (red) the outer loop optimization (cf. Fig. 7.1) and Table. 6.1 shows the corresponding NRMSE of the whole meta-testing set. A notable improvement in regression to the target signal is evident.

These results demonstrate that the L2L framework is successful in fine-tuning the reservoir's initial state Λ_0 . Remarkably, the optimization scheme finds the optimal Λ_0 coinciding with the dynamical phase transition region identified independently in previous studies [8] as producing optimal task performance from memristive reservoirs.

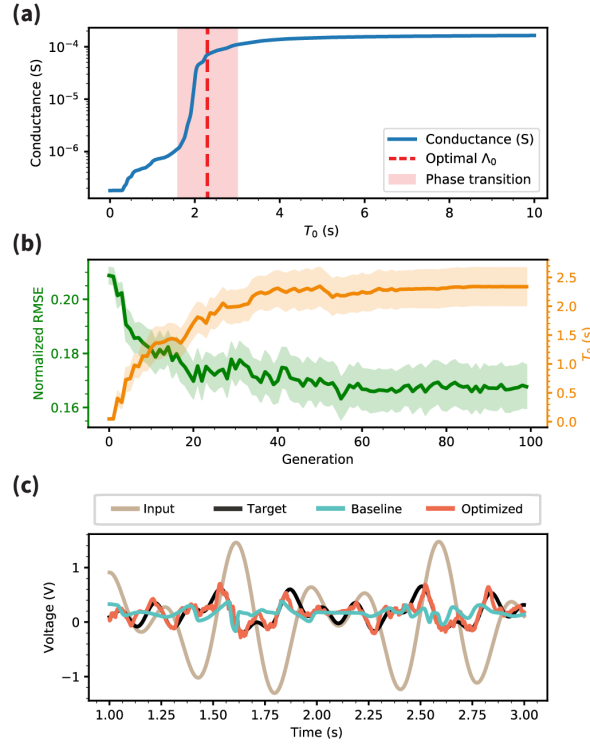


Fig. 6.3 L2L Volterra dynamics. (a) Memristive reservoir conductance response (blue) to a varying 1 V DC pulse width T_0 , showing a characteristic phase transition regime (shaded). The dashed red line represents the reservoir’s optimal initial state for learning Volterra dynamics, as found by the L2L scheme. (b) Normalised RMSE of the learning tasks and optimal T_0 with respect to outer loop generation number. (c) Input signal, corresponding target, and the resulting learning outcomes for one representative task in the family (baseline is before optimization).

To gain deeper insight into the internal dynamics of memristive reservoirs, Fig. 6.4 shows visualisation snapshots of network graphs of the reservoir for different T_0 . When the reservoir is under-activated ($T_0 < 2$ s), most memristive components are inactive and not enough information can be extracted to perform learning. On the other hand, when the reservoir is over-activated ($T_0 > 8$ s), the internal dynamics saturates. The initial reservoir state at $T_0 \approx 2.17$ s results in the best task performance and qualitatively, it is evident from Fig. 6.4 that this corresponds to an intermediate state, where conductance paths first span the network. At this ‘edge of formation’, the internal state of the reservoir produces voltage readout features that are more diverse than at later activation times, as shown in the corresponding node voltage distributions in Fig. 6.5.

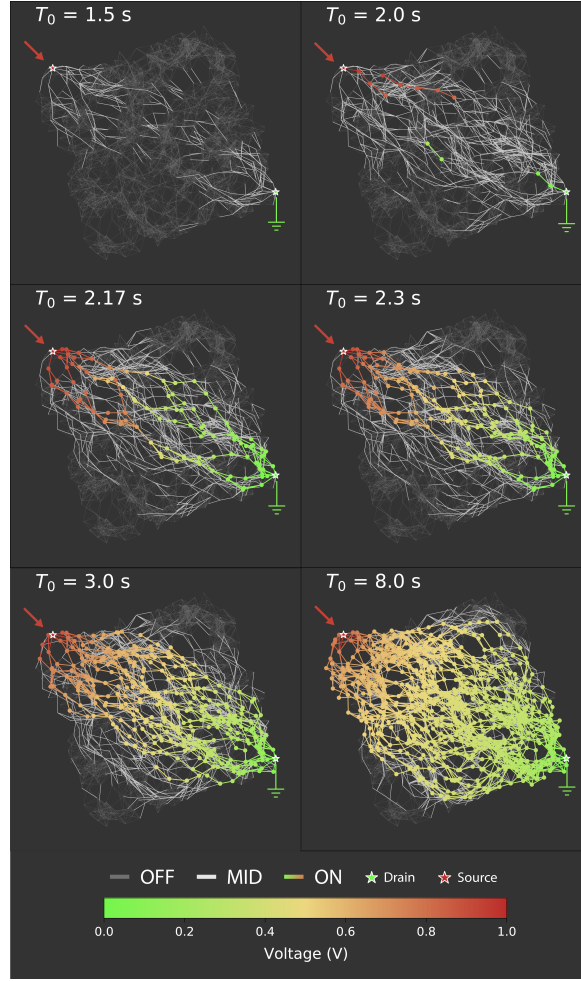


Fig. 6.4 Reservoir activity with different pre-initialization pulse widths. A 1 V DC signal of varying width is delivered to the reservoir in prior to the task. Nodes on the conductance path from source to drain are coloured based on their voltages. Edges are catagorised as high (ON), intermediate (MID) and low (OFF) conductance levels and coloured respectively.

6.3.2 L2L SNN dynamics

Fig. 6.6(a) shows the NRMSE of the query part (task-specific inner loop testing) for Task 2 during outer loop training for the two gradient-free optimization strategies considered. Similar to the previous task family, it can be observed that learning outcome improves with number of training generations. Fig. 6.6(b) compares the readouts generated by a baseline reservoir and an ES-optimised reservoir to the target curves (SNN membrane potentials) in a single meta-testing task. As is evident from the query period, the optimised system is considerably better in reproducing the fluctuating dynamics of the membrane potential, which governs the spiking behaviours of the SNN readout neurons.

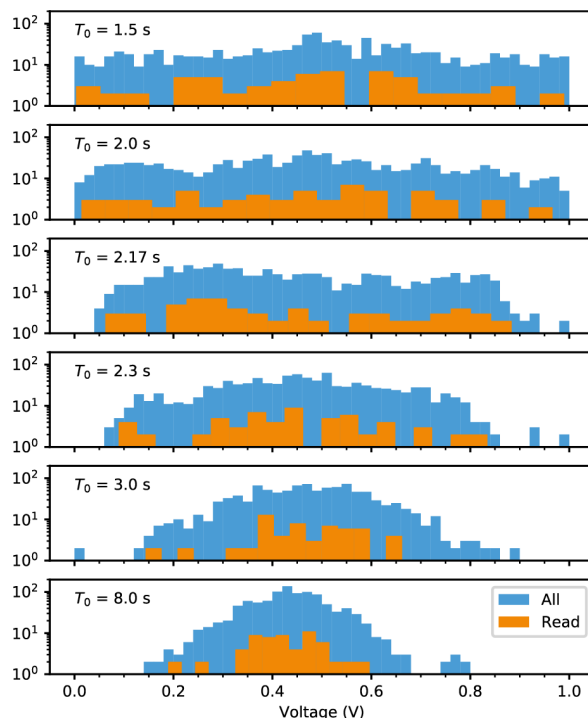


Fig. 6.5 Reservoir node voltage histograms for different pre-initialization pulse widths T_0 . Blue indicates all nodes, orange indicates nodes used as readouts for the learning tasks.

Table 6.2 NRMSE for Task 2 with and without meta-learning

	Meta-train	Meta-test
w/o meta-learning	0.212 ± 0.041	0.235 ± 0.036
Meta-learned (SA)	0.139 ± 0.011	0.164 ± 0.012
Meta-learned (ES)	0.109 ± 0.011	0.128 ± 0.004

To assess the learning gain from the L2L process, Table 6.2 compares the NRMSE for Task 2 with and without meta-learning and shows substantial improvement with optimised reservoirs, up to $\simeq 50\%$ in the case of ES optimization. Table 6.2 and Fig. 6.6(a) also indicate that the learning outcome converges to a better result with ES compared to SA optimization. This is because this task family (Task 2) employs more hyperparameters than the Volterra task family (Task 1) and ES typically performs better for a larger parameter space while SA is more suitable for stepping out of local minima.

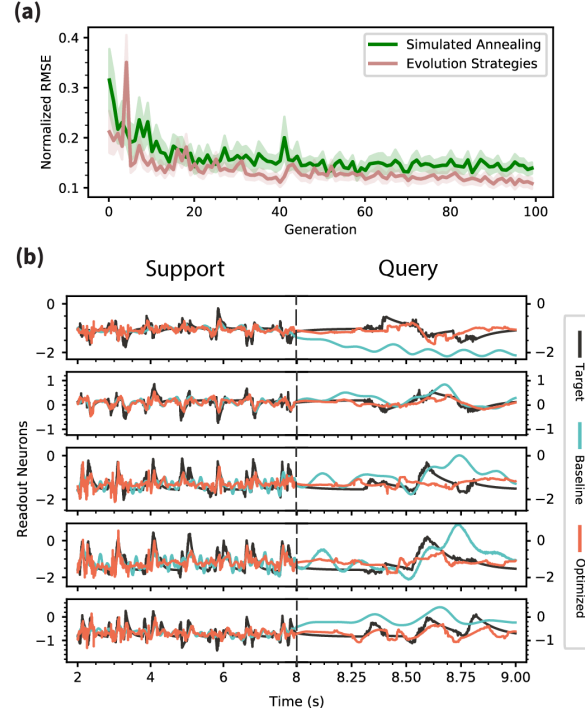


Fig. 6.6 L2L SNN membrane dynamics. (a) Normalised RMSE of the learning tasks as a function of generations in the outer loop. (b) Membrane potentials of 5 readout neurons from the SNN (black) overlaid by the learned dynamics from the baseline (blue) and the optimised (orange) memristive reservoirs for one representative meta-testing task in the task family. The left and right columns of panels show results for the support and query periods of the task, respectively. Note: a different timescale is used during the query phase for better visibility at test time.

6.4 Discussion

Previous studies have demonstrated the memory capacity of memristive reservoirs as well as their capability to generate dynamical features [2, 8, 39, 40]. The results here suggest that memristive reservoirs are able to combine these properties together to enable learning of the rich, nonlinear time-delayed dynamics embedded by the Volterra filter. Furthermore, the L2L framework consistently determined that the optimal initial state for the memristive reservoir was at what can be described as the ‘edge of formation’. This point exists between where the memristive components first become activated (cf. Fig. 6.4, $T_0=2.0$ s), and before an exponential cascade of parallel paths form (cf. Fig. 6.4, $T_0=2.3$ s). The conductance exponentially ramps up at the time of initial formation, and saturates as more parallel pathways form. The intermediate internal state, i.e., the ‘edge of formation’ was meta-learned as the optimal starting point prior to training in the outer loop.

This result opens up deeper insights to how memristive reservoirs can be optimally used in computation. It is somewhat intuitive that the reservoir does not perform at the time of initial formation, because the exponentially ramping conductance increase is highly unstable and challenging to control. The opposite problem exists after conductive pathways are formed, where switching has a negligible impact on the dynamics of the nanowire network. The ‘edge of formation’ can be thought of as a linear region in small-signal analysis, that fosters a controllable learning environment optimal for learning the rich dynamics of higher-order systems.

Additionally, this work also demonstrates that memristive reservoirs can learn the fluctuating dynamics of SNN membrane potentials. This is possible because, as shown in a previous study [9], memristive switching in a heterogeneous, recurrent network produces fluctuating dynamics that resemble action potentials which, when thresholded, generate spikes. In other words, the internal dynamics of memristive reservoirs effectively encode spike-like features into continuous input signals.

An immediate implication of this result is that memristive reservoirs have the potential to serve as an interface between continuous data streams and spike-based computing paradigms, which could substantially improve the workflow for SNN applications. The use of trainable spike-based embeddings has offered an alternative approach to classical rate and temporal encoders in many recent SNN works, but relies on gradient-based optimization to compress data into more efficient spike-based representations [41–43]. These results demonstrate how native internal dynamics, constrained by physics, naturally give rise to embeddings that can ultimately reconstruct signals in the context of gradient-free meta-learning.

6.5 Conclusion

This study shows that the L2L framework can effectively adjust the hyperparameters of memristive reservoirs, attaining a similar optimal regime as found by a manual search in previous studies. Moreover, we demonstrated that the learning capability of the system can be extended and optimised for a family of related tasks, rather than being limited to a single task. This approach could pave the way for highly adaptive learning tasks based on real-world settings, similar to how biological brains can learn quickly from limited examples. Our finding that memristive reservoirs have the ability to reproduce SNN membrane potentials is significant because of the potential to use them in place of spike-based embeddings or encoders. In particular, membrane potential dynamics in complex tasks may be encoded using memristive reservoirs as a more efficient mode of compression.

The memristive reservoirs studied here are motivated by physical reservoirs using self-assembled nanowire networks. The neuromorphic properties of these networks are influenced by various factors such as nanowire density, diameter, average length and amount of dielectric. Similar to how evolutionary optimization schemes can be utilised to design and train neuromorphic systems [44], it may be possible in the future to exploit the L2L framework to find the optimal nanowire networks for specific task families, effectively realizing a ‘learn-to-build’ approach.

References

- [1] Stieg, A. Z. *et al.* Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials* **24** (2), 286–293 (2012). doi:[10.1002/adma.201103053](https://doi.org/10.1002/adma.201103053) .
- [2] Sillin, H. O. *et al.* A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology* **24** (38), 384004 (2013). doi:[10.1088/0957-4484/24/38/384004](https://doi.org/10.1088/0957-4484/24/38/384004) .
- [3] Demis, E. C. *et al.* Atomic switch networks—nanoarchitectonic design of a complex system for natural computing. *Nanotechnology* **26** (20), 204003 (2015). doi:[10.1088/0957-4484/26/20/204003](https://doi.org/10.1088/0957-4484/26/20/204003) .
- [4] Kuncic, Z. *et al.* *Neuromorphic Information Processing with Nanowire Networks*, 1–5 (2020).
- [5] Zhu, R. *et al.* *Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning*, 102–106 (2020).
- [6] Lilak, S. *et al.* Spoken Digit Classification by In-Materio Reservoir Computing With Neuromorphic Atomic Switch Networks. *Frontiers in Nanotechnology* **3** (2021). doi:[10.3389/fnano.2021.675792](https://doi.org/10.3389/fnano.2021.675792) .
- [7] Loeffler, A. *et al.* Modularity and multitasking in neuro-memristive reservoir networks. *Neuromorphic Computing and Engineering* **1** (1), 014003 (2021). doi:[10.1088/2634-4386/ac156f](https://doi.org/10.1088/2634-4386/ac156f) .
- [8] Zhu, R. *et al.* Information dynamics in neuromorphic nanowire networks. *Scientific Reports* **11** (1), 13047 (2021). doi:[10.1038/s41598-021-92170-7](https://doi.org/10.1038/s41598-021-92170-7) .
- [9] Hochstetter, J. *et al.* Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nature Communications* **12** (1), 4008 (2021). doi:[10.1038/s41467-021-24260-z](https://doi.org/10.1038/s41467-021-24260-z) .
- [10] Zhu, R. *et al.* *MNIST classification using Neuromorphic Nanowire Networks*, ICONS 2021, 1–4 (Association for Computing Machinery, New York, NY, USA, 2021).
- [11] Kuncic, Z. & Nakayama, T. Neuromorphic nanowire networks: Principles, progress and future prospects for neuro-inspired information processing. *Advances in Physics: X* **6** (1), 1894234 (2021). doi:[10.1080/23746149.2021.1894234](https://doi.org/10.1080/23746149.2021.1894234) .

- [12] Milano, G. *et al.* In materia reservoir computing with a fully memristive architecture based on self-organizing nanowire networks. *Nature Materials* (2021). doi:[10.1038/s41563-021-01099-9](https://doi.org/10.1038/s41563-021-01099-9) .
- [13] Loeffler, A. *et al.* Neuromorphic learning, working memory, and metaplasticity in nanowire networks. *SCIENCE ADVANCES* (2023). doi:[10.1126/sciadv.adg3289](https://doi.org/10.1126/sciadv.adg3289) .
- [14] Loeffler, A. *et al.* Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* **14**, 184 (2020). doi:[10.3389/fnins.2020.00184](https://doi.org/10.3389/fnins.2020.00184) .
- [15] Milano, G., Miranda, E. & Ricciardi, C. Connectome of memristive nanowire networks through graph theory. *Neural Networks* **150**, 137–148 (2022). doi:[10.1016/j.neunet.2022.02.022](https://doi.org/10.1016/j.neunet.2022.02.022) .
- [16] Diaz-Alvarez, A. *et al.* Emergent dynamics of neuromorphic nanowire networks. *Scientific Reports* **9** (1), 14920 (2019). doi:[10.1038/s41598-019-51330-6](https://doi.org/10.1038/s41598-019-51330-6) .
- [17] Dunham, C. S. *et al.* Nanoscale neuromorphic networks and criticality: A perspective. *Journal of Physics: Complexity* **2** (4), 042001 (2021). doi:[10.1088/2632-072X/ac3ad3](https://doi.org/10.1088/2632-072X/ac3ad3) .
- [18] Chang, T.-C., Chang, K.-C., Tsai, T.-M., Chu, T.-J. & Sze, S. M. Resistance random access memory. *Materials Today* **19** (5), 254–264 (2016). doi:[10.1016/j.mattod.2015.11.009](https://doi.org/10.1016/j.mattod.2015.11.009) .
- [19] Eshraghian, J. K., Wang, X. & Lu, W. D. Memristor-based binarized spiking neural networks: Challenges and applications. *IEEE Nanotechnology Magazine* **16** (2), 14–23 (2022) .
- [20] Ielmini, D. & Wong, H.-S. P. In-memory computing with resistive switching devices. *Nature electronics* **1** (6), 333–343 (2018) .
- [21] Hospedales, T., Antoniou, A., Micaelli, P. & Storkey, A. Meta-Learning in Neural Networks: A Survey (2020). [2004.05439](https://arxiv.org/abs/2004.05439).
- [22] Hochreiter, S., Younger, A. S. & Conwell, P. R. Dorffner, G., Bischof, H. & Hornik, K. (eds) *Learning to Learn Using Gradient Descent*. (eds Dorffner, G., Bischof, H. & Hornik, K.) *Artificial Neural Networks — ICANN 2001*, Lecture Notes in Computer Science, 87–94 (Springer, Berlin, Heidelberg, 2001).
- [23] Andrychowicz, M. *et al.* Learning to learn by gradient descent by gradient descent (2016). [arXiv:1606.04474](https://arxiv.org/abs/1606.04474).

- [24] Finn, C., Abbeel, P. & Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks (2017). [arXiv:1703.03400](#).
- [25] Bellec, G., Salaj, D., Subramoney, A., Legenstein, R. & Maass, W. Long short-term memory and learning-to-learn in networks of spiking neurons (2018). [arXiv:1803.09574](#).
- [26] Subramoney, A., Scherr, F. & Maass, W. in *Reservoirs learn to learn* 59–76 (2021). [1909.07486](#).
- [27] Bohnstingl, T., Scherr, F., Pehle, C., Meier, K. & Maass, W. Neuromorphic Hardware Learns to Learn. *Frontiers in Neuroscience* **13** (2019) .
- [28] Jaeger, H. The “echo state” approach to analysing and training recurrent neural networks – with an Erratum note 47 (2001) .
- [29] Maass, W., Natschläger, T. & Markram, H. Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. *Neural Computation* **14** (11), 2531–2560 (2002). doi:[10.1162/089976602760407955](#) .
- [30] Lukoševičius, M. & Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review* **3** (3), 127–149 (2009). doi:[10.1016/j.cosrev.2009.03.005](#) .
- [31] Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by Simulated Annealing. *Science* **220** (4598), 671–680 (1983). [1690046](#) .
- [32] Rechenberg, I. Evolutionsstrategie. *Optimierung technischer Systeme nach Prinzipien der biologischen Evolution* (1973) .
- [33] Wierstra, D., Schaul, T., Peters, J. & Schmidhuber, J. *Natural Evolution Strategies*, 3381–3387 (IEEE, Hong Kong, China, 2008).
- [34] Sehnke, F. *et al.* Parameter-exploring policy gradients. *Neural Networks* **23** (4), 551–559 (2010). doi:[10.1016/j.neunet.2009.12.004](#) .
- [35] Salimans, T., Ho, J., Chen, X., Sidor, S. & Sutskever, I. Evolution Strategies as a Scalable Alternative to Reinforcement Learning (2017). [arXiv:1703.03864](#).
- [36] Eshraghian, J. K. *et al.* Training Spiking Neural Networks Using Lessons From Deep Learning (2021). doi:[10.48550/arXiv.2109.12894](#) .

- [37] Jaeger, H. A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach (2002) .
- [38] Hastie, T., Tibshirani, R., Friedman, J. H. & Friedman, J. H. The elements of statistical learning: data mining, inference, and prediction (2009).
- [39] Du, C. *et al.* Reservoir computing using dynamic memristors for temporal information processing. *Nature Communications* **8** (1), 2204 (2017). doi:[10.1038/s41467-017-02337-y](https://doi.org/10.1038/s41467-017-02337-y) .
- [40] Fu, K. *et al.* *Reservoir Computing with Neuromemristive Nanowire Networks*, 1–8 (2020).
- [41] Zhu, R.-J., Zhao, Q. & Eshraghian, J. K. Spikegpt: Generative pre-trained language model with spiking neural networks. *arXiv preprint arXiv:2302.13939* (2023) .
- [42] Dold, D. *Relational representation learning with spike trains*, 1–8 (IEEE, 2022).
- [43] Zhang, T. *et al.* Sscae: A neuromorphic snn autoencoder for sc-rna-seq dimensionality reduction (2023) .
- [44] Schuman, C. D., Mitchell, J. P., Patton, R. M., Potok, T. E. & Plank, J. S. *Evolutionary Optimization for Neuromorphic Systems*, 1–9 (ACM, Heidelberg Germany, 2020).

Chapter 7

Online Dynamical Learning and Sequence Memory in Neuromorphic Nanowire Networks

“Without the fear of heights, there can be no appreciation for the beauty of high places.”

The Dark Forest - Cixin Liu

This chapter has been accepted for publications as:

Zhu, R. , Lilak, S., Loeffler, A., Lizier, J., Stieg, A., Gimzewski, J., Kuncic, Z., 2023. Online dynamical learning and sequence memory in neuromorphic nanowire networks, Nature Communications 14, 6697. <https://doi.org/10.1038/s41467-023-42470-5>

Abstract

Nanowire Networks (NWNs) belong to an emerging class of neuromorphic systems that exploit the unique physical properties of nanostructured materials. In addition to their neural network-like physical structure, NWNs also exhibit resistive memory switching in response to electrical inputs, due to synapse-like changes in conductance at nanowire-nanowire cross-point junctions. Previous studies have demonstrated how the neuromorphic dynamics generated by NWNs can be harnessed for temporal learning tasks. This study extends these findings further by demonstrating online learning from spatiotemporal dynamical features using image classification and sequence memory recall tasks implemented on an NWN device. Applied to the MNIST handwritten digit classification task, online dynamical learning with the NWN device achieves an overall accuracy of 93.4%. Additionally, we find a correlation between classification accuracy of individual digit classes and mutual information. The sequence memory task reveals how memory patterns embedded in the dynamical features enable online learning and recall of a spatiotemporal sequence pattern. Overall, these results provide proof-of-concept of online learning from spatiotemporal dynamics using NWNs and further elucidate how memory can enhance learning.

7.1 Introduction

Neuromorphic devices offer the potential for a fundamentally new computing paradigm, one based on a brain-inspired architecture that promises enormous efficiency gains over conventional computing architectures [1–11]. A particularly successful neuromorphic computing approach is the implementation of spike-based neural network algorithms in CMOS-based neuromorphic hardware [2, 12–17]. An alternate neuromorphic computing approach is to exploit brain-like physical properties exhibited by novel nano-scale materials and structures [18–22], including in particular the synapse-like dynamics of resistive memory (memristive) switching [4, 23–31].

This study focuses on a class of neuromorphic devices based on memristive nanowire networks (NWNs) [32, 33]. NWNs are comprised of metal-based nanowires that form a heterogeneous network structure similar to a biological neural network [34–38]. Additionally, nanowire-nanowire cross-point junctions exhibit memristive switching attributed to the evolution of a metallic nano-filament due to electro-chemical metallisation [39–43]. Typically, each NWN contains thousands of nanowires and an even greater number of junctions. In response to electrical input signals, NWNs also exhibit brain-like collective dynamics (e.g. phase transitions, switch synchronisation, avalanche criticality), resulting from the interplay between memristive switching and their recurrent network structure [34, 37, 44–50].

Recurrent, sparse networks can transform temporal signals into a higher dimensional dynamical feature space [51, 52], which is advantageous for machine learning applications involving dynamically evolving data [53]. Furthermore, the computational burden of training network weights can be circumvented altogether by leveraging Reservoir Computing (RC), which restricts training to a linear output layer, in which only linear weights need to be learned using the rich dynamical features generated by the recurrent network reservoir [54–56]. Physical systems are particularly useful as reservoirs, due to their self-regulating dynamics and physical constraints imposed by conservation laws (e.g. Kirchoff’s laws), in contrast to algorithmic RC, which typically uses a random network with fixed weights and requires manual hyper-parameter optimisation [57]. Previous experimental [36, 58, 59] and simulation [36, 58, 60–65] studies have demonstrated NWNs exhibit fading memory and can effectively project input signals to a higher dimensional feature space, thus enabling their use as physical reservoirs in an RC approach to machine learning.

In previous physical RC studies, learning is achieved by training the readout weights after the entire input stream is delivered to the physical system [66], while the real-time response from the network is not fully reflected in the learning outcome. While such batch-

based approaches can be practically limited by memory availability when working with large datasets, an arguably more important consideration is the need to re-train weights when feature distributions evolve [67]. An alternate approach is online training, which has the potential to enhance dynamical learning by allowing the readout weights to adapt to non-stationary dynamical features incrementally [68, 69]. As is the case for conventional machine learning, online learning approaches are necessary for scaling up neuromorphic computing and ultimately achieving the goal of continual learning [70–72].

In this study, we use an NWN device to demonstrate *online dynamical learning*, i.e. learning incrementally from continuous streams of dynamical features. We implement an online training algorithm within an RC framework and use the MNIST handwritten digit database to deliver a stream of spatiotemporal patterns to the NWN device. Dynamical features in the device readouts are then used to train a linear classifier in an online manner, sample by sample, and information theoretic measures are used to analyse the online learning process. By constructing a numerical sequence pattern using the MNIST database, we then develop and implement a novel sequence memory task that demonstrates the NWN’s ability to generate spatiotemporal memory patterns in a similar manner to the brain, using attractor dynamics. We show how these sequence memory patterns can also be learned in an online manner and then used to recall a target digit presented earlier in the sequence. By comparing recall performance with and without the memory patterns, we demonstrate how memory enhances learning.

7.2 Results

The first task we performed to test online dynamical learning is the MNIST handwritten digit classification task, which has not previously been experimentally implemented on an NWN device (but has been implemented in NWN simulations [62, 65]). A schematic illustration of the experimental setup for online classification of MNIST handwritten digits using an NWN multi-electrode array (MEA) device is shown in Fig. 7.1. MNIST digit images [73] are converted to 1-D temporal voltage pulse streams and delivered consecutively to one electrode. The network’s real-time response is read out from other electrode channels and classification is performed in an external (digital) fully-connected output layer. The weights are learned from the dynamical features and updated after each digit sample using an online iterative algorithm based on recursive least squares (RLS). See [Methods](#) for full details.

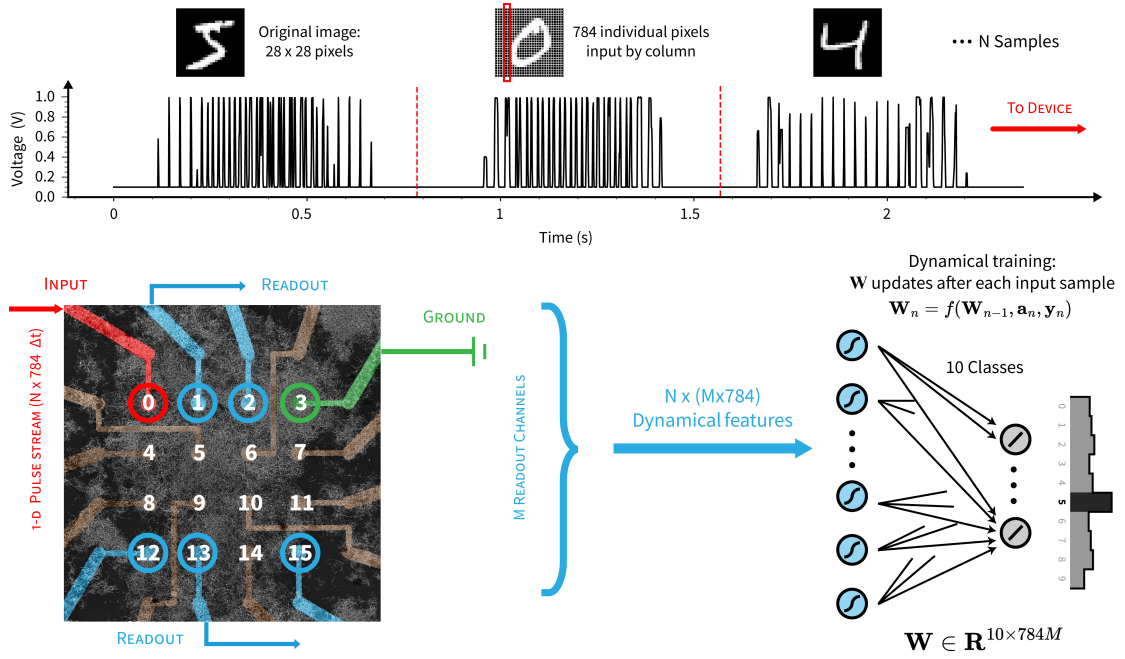


Fig. 7.1 Schematic illustration of nanowire network device setup for MNIST digit classification demonstrating online dynamical learning. Top: MNIST handwritten digit samples (N samples $\times$ 784 pixel features) are normalised and converted to 1-D temporal voltage pulse streams (each pixel occupies $\Delta t = 0.001$ s) and delivered consecutively to the nanowire multi-electrode device. Bottom left: scanning electron micrograph image of the 16-electrode device, showing source electrode (channel 0, red), drain electrode (channel 3, green), readout electrodes (channel 1, 2, 12, 13, 15, blue) and other electrodes not used (brown). Bottom right: readout voltages (i.e. $N \times M \times 784$ dynamical features) are input into an external linear classifier in which the weight matrix $\mathbf{W}_n$ for the $M \times 784$ features per digit sample is updated after each sample $\mathbf{a}_n$, with corresponding class $\mathbf{y}_n$ as the target output (digit ‘5’ shown as an example of classification result).

7.2.1 Dynamical feature generation

Fig. 7.2 shows examples of handwritten digit image samples converted to 1-D voltage pulse streams delivered to the allocated source electrode (channel 0) and the corresponding voltage streams read out from other channels (1, 2, 12, 13 and 15) for the setup shown in Fig. 7.1 (readouts for other channels and digits are shown in Supplementary Figures S8 and S9).

Each row in Fig. 7.2 shows the averaged image, input and readout data for 100 MNIST samples randomly selected from the training set for the corresponding digit class.

For each class, the readout voltages from each channel (columns 3-7, blue) are distinctly different from the corresponding input voltages and exhibit diverse characteristics across the readout channels. This demonstrates that the NWN nonlinearly maps the input signals

into a higher-dimensional space. Rich and diverse dynamical features are embedded into the channel readouts from the spatially distributed electrodes, which are in contact with different parts of the network (see Supplementary Figure S10 for additional non-equilibrium dynamics under non-adiabatic conditions). We show below how the inter-class distinctiveness of these dynamical features, as well as their intra-class diversity, can be harnessed to perform online classification of the MNIST digits.

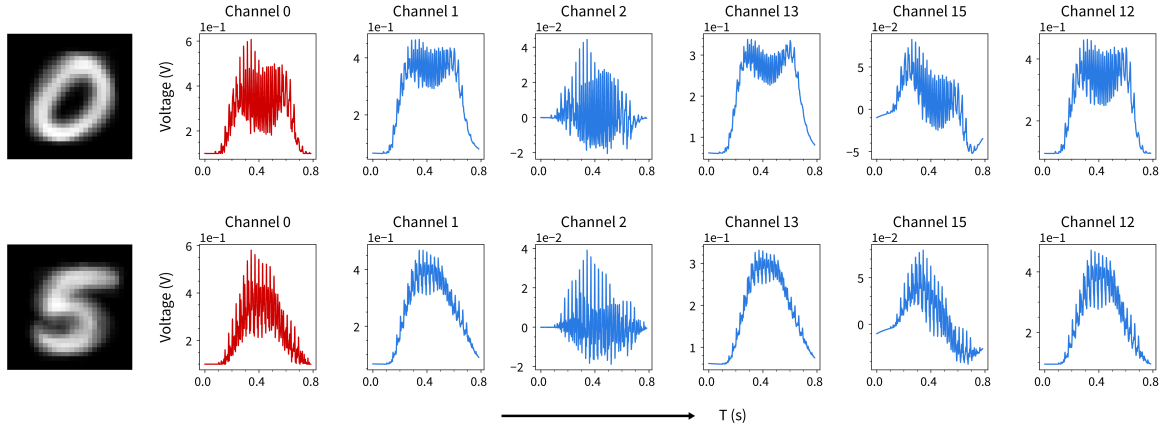


Fig. 7.2 **Input–output mapping of MNIST digits in a nanowire network device.** Images of MNIST digits ‘0’ and ‘5’ averaged across 100 samples randomly selected from the training set (column 1), their corresponding input voltage streams (column 2, red) and readout voltages from multiple channels (columns 3-7, blue).

7.2.2 Online learning

Table 7.1 presents the MNIST handwritten digit classification results using the online method (external weights trained by an RLS algorithm). Results are shown for one and five readout channels. For comparison, also shown are the corresponding classification results using the offline batch method (external weights trained by backpropagation with gradient descent). Both classifiers learn from the dynamical features extracted from the NWN, with readouts delivered to the two classifiers separately. For both classifiers, accuracies increase with the number of readout channels, demonstrating the non-linearity embedded by the network in the readout data. For the same number of channels, however, the online method outperforms the batch method. In addition to achieving a higher classification accuracy, the online classifier W requires only a single epoch of 50,000 training samples, compared to 100 training epochs for the batch method using 500 mini-batches of 100 samples and a learning rate $\eta = 0.1$. Accuracy of the online classifier becomes comparable to that of the batch classifier when active error correction is not used in the RLS algorithm (see Supplementary Table 1). A key

Readout Data	Classifier	Accuracy
784 x 1	online	91.6%
784 x 5	online	93.4%
784 x 1	batch	86.6%
784 x 5	batch	91.4%

Table 7.1 **MNIST digit classification using an NWN device with one and five readout channels for online and batch (offline) linear classifiers.** Each classification accuracy is calculated for 10,000 testing samples and averaged across 5 iterations. The online classifier is trained for a single epoch of 50,000 training samples, while the batch-based classifier is trained and optimised using gradient descent for 100 epochs with 500 mini-batches of size 100.

advantage of the online method is that continuous learning from the streaming input data enables relatively rapid convergence, as shown next.

To better understand how learning is achieved with the NWN device, we investigated further the dependence of classification accuracy on the number of training samples and the device readouts. Fig. 7.3(a) shows classification accuracy as a function of number of digits presented to the classifier during training (See Supplementary Figure S11 for classification results using different electrode combinations for input/drain/readouts and different voltage ranges). The classification accuracy consistently increases as more readout samples are presented to the classifier to update $\mathbf{W}$ and plateaus at $\simeq 92\%$ after $\simeq 10,000$ samples. Classification accuracy also increases with number of readout channels, corresponding to an increase in the number of dynamical features (i.e. 5×784 features per digit for 5 channel readouts, the channels are added following the order 1,2,13,15,12) that become sufficiently distinguishable to improve classification. However, as shown in Fig. 7.3(b), this increase is not linear, with the largest improvements observed from 1 to 2 channels. Fig. 7.3(c) shows the confusion matrix for the classification result using 5 readout channels after learning from 50,000 digit samples. The classification results for 8 digits lie within 1.5σ (where s.d. is $\sigma = 3\%$) from the average (93.4%). Digit ‘1’ demonstrates significantly higher accuracy since it has a simpler structure and ‘5’ is an outlier because of the irregular variances of handwriting and low pixel resolution (See Supplementary Figure S12 for examples of misclassified digits).

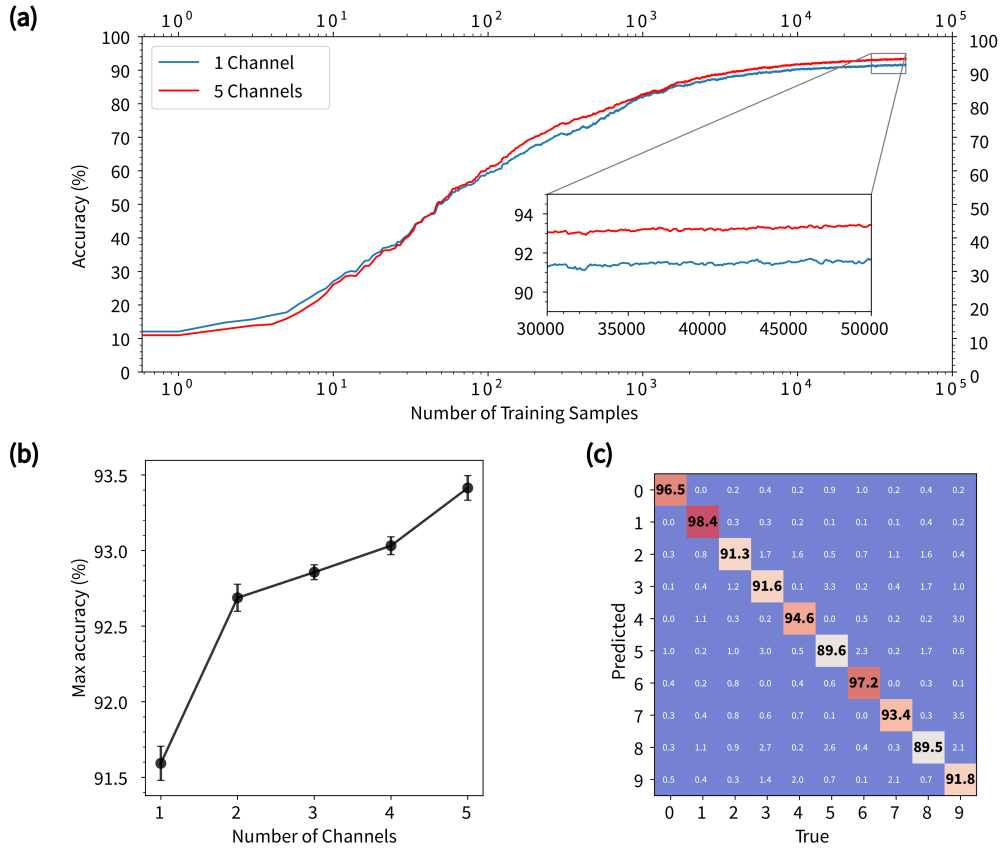


Fig. 7.3 **Online learning of MNIST digits.** (a) Testing accuracy as a function of number of training samples read out from one and five channels. Inset shows a zoom-in of the converged region of the curve. (b) Maximum testing accuracy achieved after 50,000 training samples with respect to number of readout channels used by the online linear classifier. Error bars indicate the standard error of the mean of 5 measurements with shuffled training samples. (c) Confusion matrix for online classification using 5 readout channels.

7.2.3 Mutual information

Mutual information (MI) is an information-theoretic metric that can help uncover the inherent information content within a system and provide a means to assess learning progress during training. Fig. 7.4(a) shows the learning curve of the classifier, represented by the mean of the magnitude of the change in the weight matrix, $|\Delta\mathbf{W}|$, as a function of number of sample readouts for 5 channels. Learning peaks at $\simeq 10^2 - 10^3$ samples, after which it declines rapidly and becomes negligible by 10^4 samples. This is reflected in the online classification accuracy (cf. Fig. 7.3(a)), which begins to saturate by $\sim 10^4$ samples. The rise and fall of the learning rate profile can be interpreted in terms of maximal dynamical information being extracted by the network. This is indicated by Fig. 7.4(b), which presents mutual

information (MI) between the 10 MNIST digit classes and each of the NWN device readouts used for online classification (cf. Fig. 7.3). The MI values for each channel are calculated by averaging the values across the 784 pixel positions. The coincidence of the saturation in MI with the peak in $|\overline{\Delta\mathbf{W}}|$ between $10^2 - 10^3$ samples demonstrates learning is associated with information dynamics. Note that by $\simeq 10^2$ samples, the network has received approximately 10 samples for each digit class (on average). It is noteworthy too that MI for the input channel is substantially smaller.

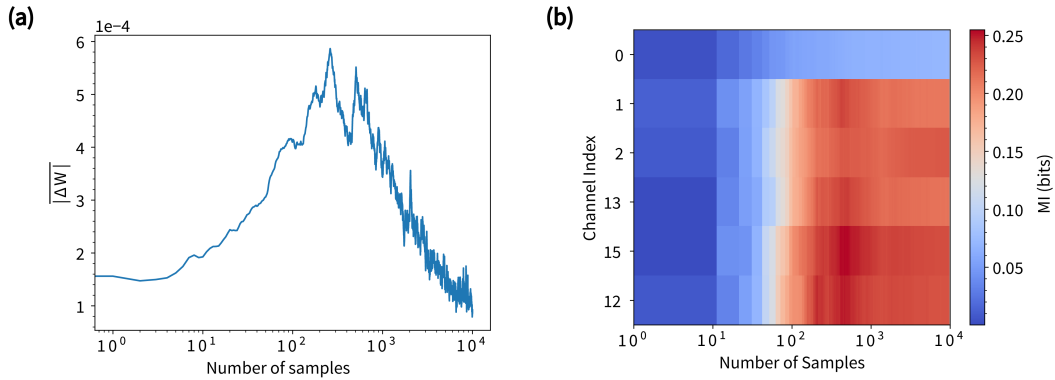


Fig. 7.4 Learning rate and mutual information. (a) Mean of the magnitude of changes in the linear weight matrix, $|\overline{\Delta\mathbf{W}}|$, as a function of number of samples learned by the network. (b) Corresponding Mutual Information (MI) for each of the 5 channels used for online classification (cf. Fig. 7.3) and for input channel 0.

Fig. 7.5 shows MI estimated in a static way, combining all the samples after the whole training dataset is presented to the network. The MI maps are arranged according to the digit classes and averaged within each class. The maps suggest that distinctive information content is extracted when digit samples from different classes are streamed into the network. This is particularly evident when comparing the summed maps for each of the digits (bottom row of Fig. 7.5). Additionally, comparison with the classification confusion matrix shown in Fig. 7.3(c) reveals that the class with the highest total MI value ('1') exhibits the highest classification accuracy (98.4%), while the lowest MI classes ('5' and '8') exhibit the lowest accuracies (89.6% and 89.5%), although the trend is less evident for intermediate MI values.

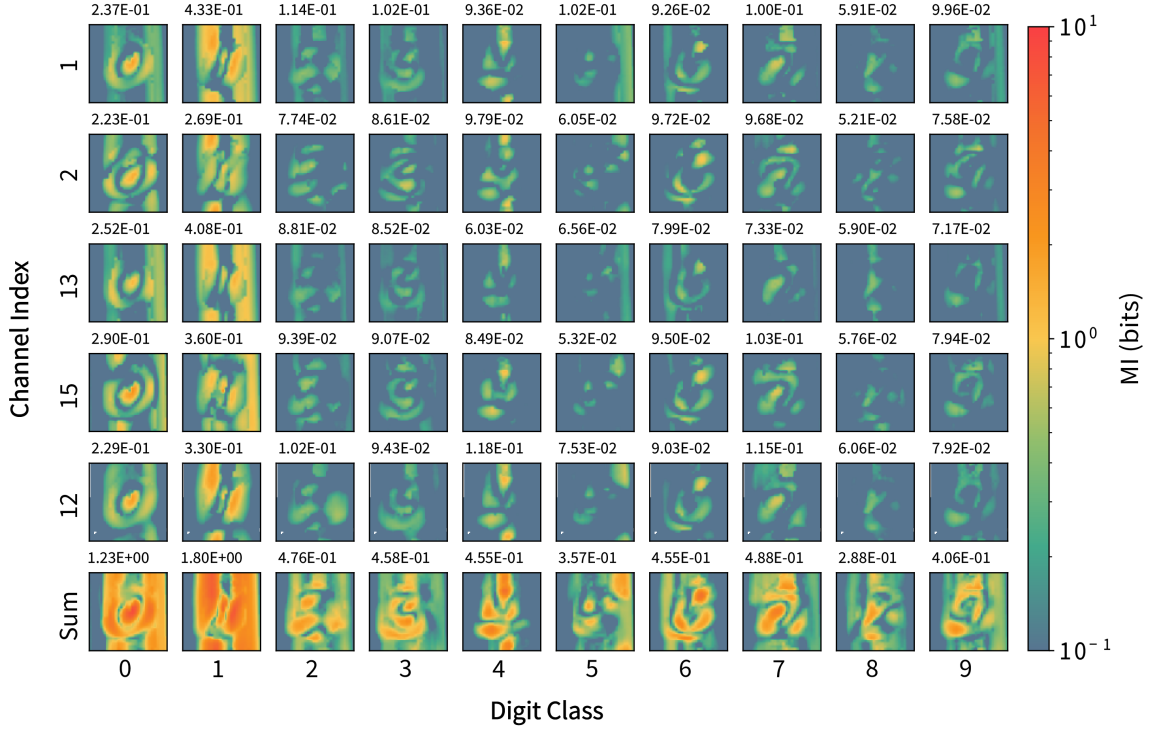


Fig. 7.5 **Spatial distribution of mutual information (MI) for readout channels (rows), sorted by digit class (columns).** MI maps summed over all channels are shown in the bottom row and mean MI is shown above each map.

7.2.4 Sequence memory task

As mentioned earlier, RC is most suitable for time-dependent information processing. Here, an RC framework with online learning is used to demonstrate the capacity of NWNs to recall a target digit in a temporal digit sequence constructed from the MNIST database. The sequence memory task is summarised in Fig. 7.6. A semi-repetitive sequence of 8 handwritten digits is delivered consecutively into the network in the same way as individual digits were delivered for the MNIST classification task. In addition to readout voltages, the network conductance is calculated from the output current. Using a sliding memory window, the earliest (first) digit is reconstructed from the memory features embedded in the conductance readout of subsequent digits. Fig. 7.6 shows digit ‘7’ reconstructed using the readout features from the network corresponding to the following 3 digits, ‘5’, ‘1’ and ‘4’. See [Methods](#) for details.

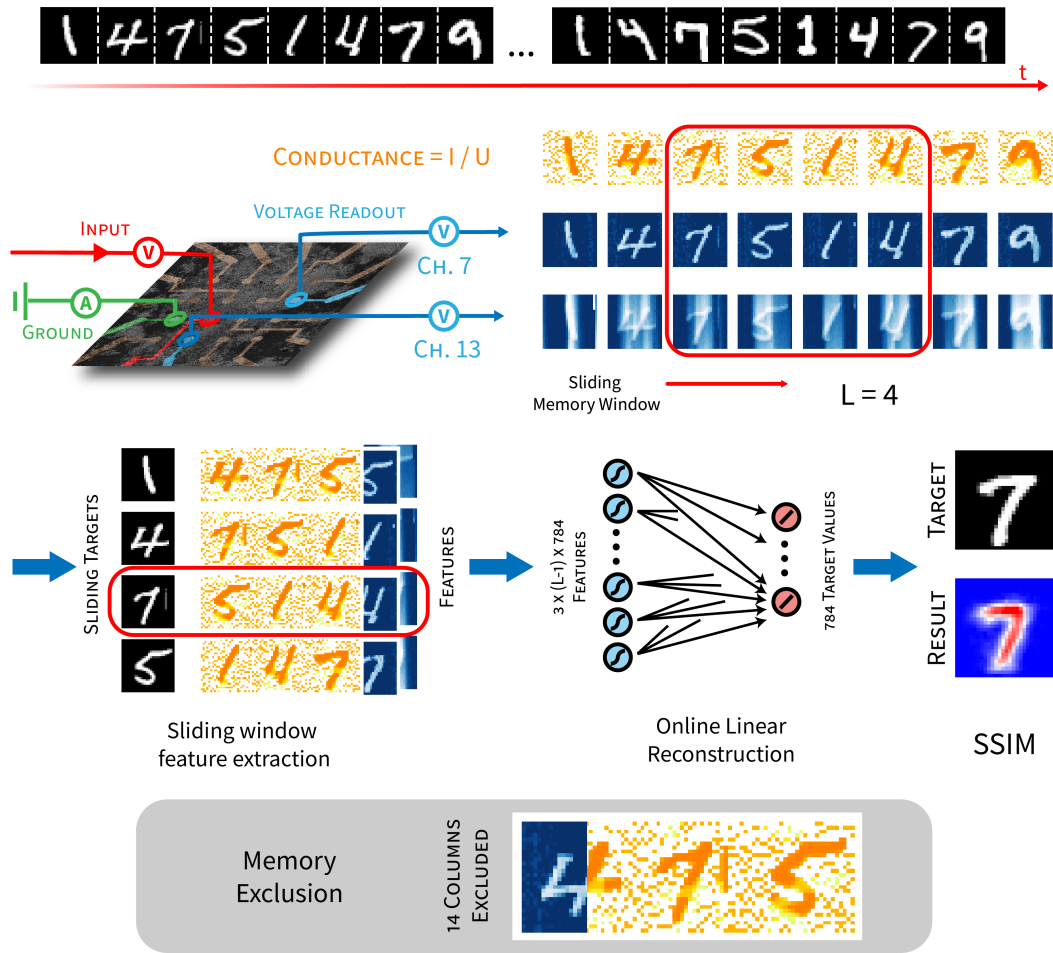


Fig. 7.6 **Schematic outline of the sequence memory task.** Samples of a semi-repetitive 8-digit sequence (14751479) constructed from the MNIST dataset are temporally streamed into the NWN device through one input channel. A memory window of length L ($L = 4$ shown as an example) slides through each digit of the readouts from 2 channels (7 and 13) as well as the network conductance. In each sliding window, the first (earliest) digit is selected for recall as its image is reconstructed from the voltage of one readout channel (channel 7) and memory features embedded in the conductance time series of later $L - 1$ digits in the memory window. Linear reconstruction weights are trained by the online learning method and reconstruction quality is quantified using the structural similarity index measure (SSIM). The shaded grey box shows an example of the memory exclusion process, in which columns of conductance memory features are replaced by voltage readouts from another channel (channel 13) to demonstrate the memory contribution to image reconstruction (recall) of the target digit.

Fig. 7.7(a) shows the network conductance time series and readout voltages for one of the digit sequence samples. The readout voltages exhibit near-instantaneous responses to high pixel intensity inputs, with dynamic ranges that vary distinctively among different

channels. The conductance time series also exhibits a large dynamic range (at least 2 orders of magnitude) and additionally, delayed dynamics. This can be attributed to recurrent loops (i.e. delay lines) in the NWN and to memristive dynamics, determined by nano-scale electro-ionic transport. The delay dynamics demonstrate that NWNs retain memory of previous inputs (see Supplementary Figure S13 for an example showing the fading memory property of the NWN reservoir). Fig. 7.7(b) shows the respective digit images and $I - V$ curves for the sequence sample. The NWN is driven to different internal states as different digits are delivered to the network in sequence. While the dynamics corresponding to digits from the same class show some similar characteristics in the $I - V$ phase space (e.g. digit '1'), generally they exhibit distinctive characteristics due to their sequence position. For example, the first instance of '4' exhibits dynamics that explore more of the phase space than the second instance of '4'. This may be attributed to differences in the embedded memory patterns, with the first '4' being preceded by '91' while the second '4' is preceded by '51' and both '9' and '5' have distinctively different phase space characteristics, which are also influenced by their sequence position as well as their uniqueness.

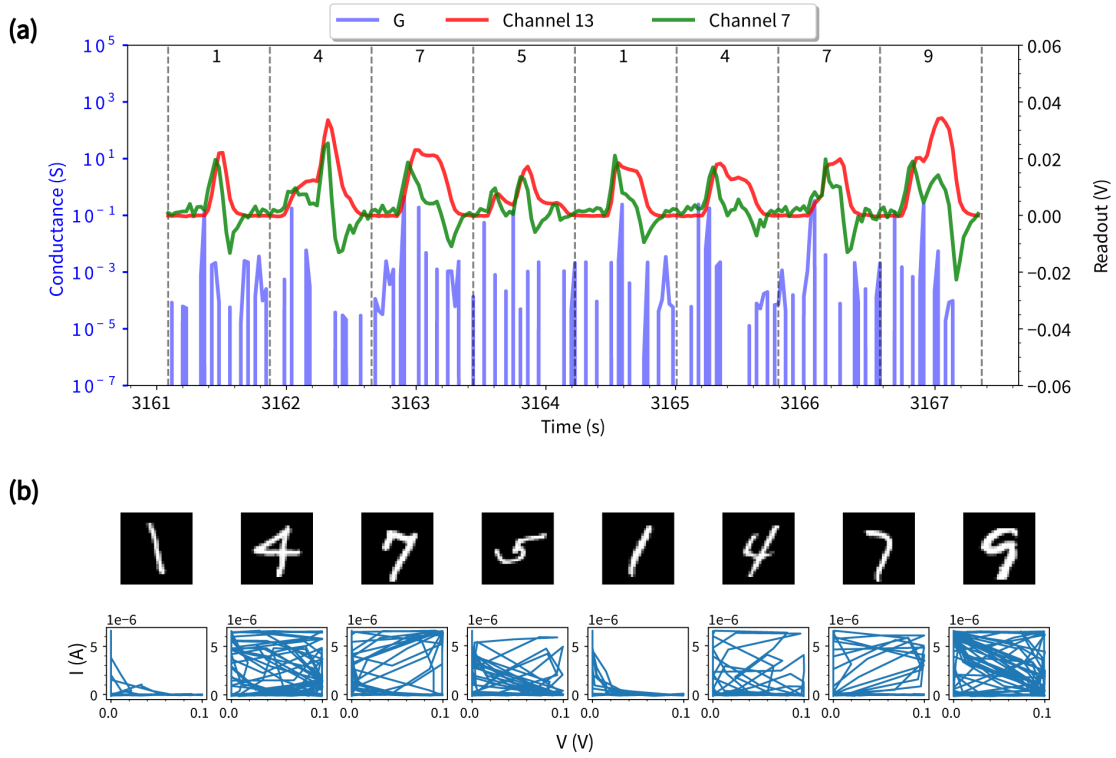


Fig. 7.7 **NWN sequence memory.** (a) Conductance time series (G) and readout voltages for one full sequence cycle during the sequence memory task. For better visualisation, voltage readout curves are smoothed by averaging over a moving window of length 0.05 s and values for channel 7 are magnified by $\times 10$. (b) Corresponding digit images and memory patterns in $I - V$ phase space.

Fig. 7.8(a) shows the image reconstruction quality for each digit in the sequence as a function of memory window length. Structural similarity (SSIM) is calculated using a testing group of 500 sets and the maximum values achieved after learning from 7000 training sets are presented (see Supplementary Figure S14 for the learning curving for $L = 4$ and Supplementary Figure S15 for average SSIM across all digits). The best reconstruction results are achieved for digits ‘1’ and ‘7’, which are repeated digits with relatively simple structures. In contrast, digit ‘4’, which is also repeated, but has a less simple structure, is reconstructed less faithfully. This indicates that the repeat digits produce memory traces that are not completely forgotten before each repetition (i.e. nano-filaments in memristive junctions do not completely decay). On average, the linear reconstructor is able to recall these digits better than the non-repeat digits. For the non-repeat digits (‘5’ and ‘9’) the reconstruction results are more interesting: digit ‘5’ is consistently reconstructed with the lowest SSIM, which correlates with its low classification accuracy (cf. Fig. 7.3(c)), while ‘9’ exhibits a distinctive jump from $L = 4$ to $L = 5$ (see also Fig. 7.8(b)). This reflects the

contextual information used in the reconstruction: for $L = 4$, ‘9’ is reconstructed from the sub-sequence ‘147’, which is the same sub-sequence for ‘5’, but for $L = 5$, ‘9’ is uniquely reconstructed from sub-sequence ‘1475’, with a corresponding increase in SSIM. This is not observed for digit ‘5’; upon closer inspection, it appears that the reconstruction of ‘5’ suffers interference from ‘9’ (see Supplementary Figure S16), due to the common sub-sequence ‘147’ and to the larger variance of ‘5’ in the MNIST dataset (which also contributes to its misclassification). A similar jump in SSIM is evident for the repeat digit ‘7’ from $L = 2$ to $L = 3$. For $L = 3$, the first instance of ‘7’ (green curve) is reconstructed from ‘51’, while the second instance (pink curve) is reconstructed from ‘91’, so the jump in SSIM from $L = 2$ may be attributed to digit ‘7’ leaving more memory traces in digit ‘1’, which has a simpler structure than either ‘9’ or ‘5’.

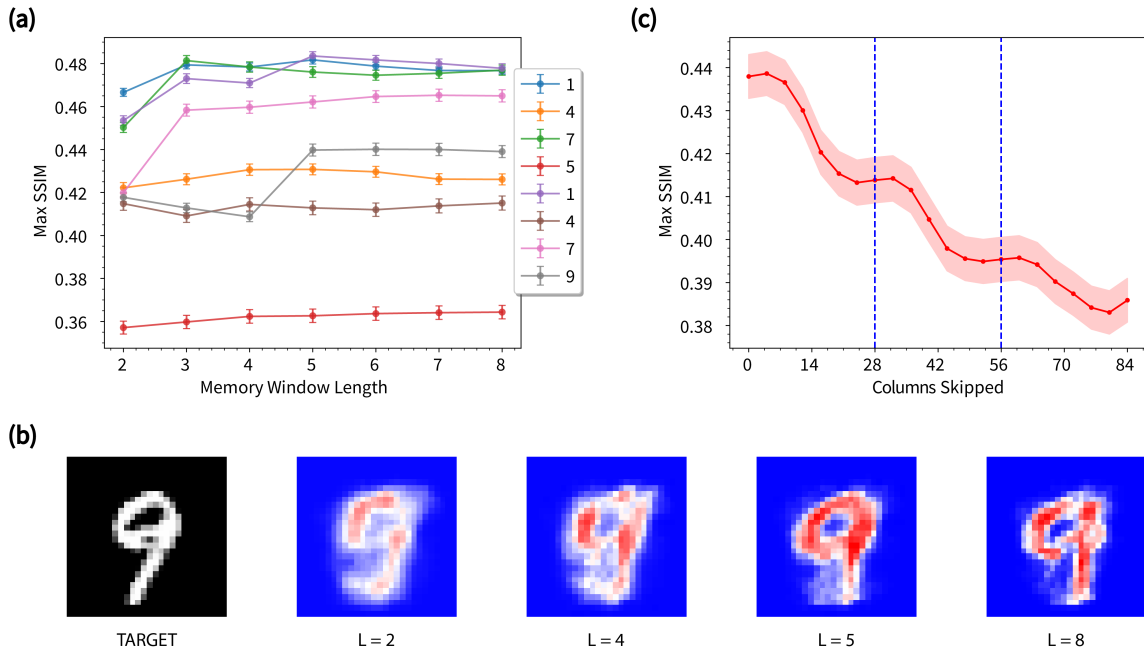


Fig. 7.8 Digit reconstruction in sequence memory task using the online-learning algorithm. (a) Maximum SSIM for each digit in the sequence as a function of the memory window length after the network learned from 7000 training sets. The testing set is comprised of 500 sequence samples. Errorbars indicate the standard error of the mean across the samples within each digit class. (b) An example of a target digit from the MNIST dataset and the reconstructed digits using memory windows of different lengths. (c) Maximum SSIM with respect to number of columns excluded in the memory feature space. The results are averaged across all testing digits using $L = 4$, and the standard error of the mean is indicated by the shading. Dashed blue lines indicate when whole digits are excluded.

While the SSIM curves for each individual digit in the sequence increase only gradually with memory window length, their average (shown in Supplementary Figure S15) shows

an increase up to $L = 5$, followed by saturation. This reflects the repetition length of the sequence.

Fig. 7.8(c) shows the maximum SSIM, averaged over all reconstructed digits using $L = 4$, when memory is increasingly excluded from the online reconstruction. SSIM decreases as more columns of conductance features are excluded (replaced with memoryless voltage features). This demonstrates that the memory embedded in the conductance features enhances online learning by the reconstructor. In particular, the maximum SSIM plateaus when ~ 28 and ~ 56 columns (corresponding to whole digits) are excluded and decreases significantly when the number of columns excluded is approximately 14, 42 or 70, indicating most of the memory traces are embedded in the central image pixels.

7.3 Discussion

This study is the first to perform the MNIST handwritten digit classification benchmark task using an NWN device. In a previous study, Milano et al. [65] simulated an NWN device and mapped the readouts to a ReRAM cross-point array to perform *in materia* classification (with a 1-layer neural network) of the MNIST digits, achieving an accuracy of 90.4%. While our experimental implementation is different, readouts from their simulated NWN device also exhibited diverse dynamics and distinct states in response to different digit inputs, similar to that observed in this study. Other studies using memristor cross-bar arrays as physical reservoirs achieved lower MNIST classification accuracies [74, 75]. In contrast, NWN simulation studies achieved higher classification accuracies of $\simeq 98\%$ by either pre-processing the MNIST digits with a convolutional kernel [62] or placing the networks into a deep learning architecture [76].

In this study, the relatively high classification accuracy achieved with online learning (93.4%) can be largely attributed to the iterative algorithm, which is based on recursive least squares (RLS). Previous RC studies by Jaeger et al. [77, 78] suggested that RLS converges faster than least mean squares (similar to gradient-based batch methods), which tends to suffer more from numerical roundoff error accumulation, whereas RLS converges in a finite number of steps and uses the remaining training samples for fine-tuning [69]. This is evident in our results showing incremental learning of the weight matrix and is also corroborated by our mutual information analysis. While we performed online classification in an external digital layer, it may be possible to implement the online learning scheme in hardware using, for example, a cross-point array of regular resistors, which exhibit a linear (i.e. Ohmic) response. Such a system would then represent an end-to-end analogue hardware solution for

efficient online dynamical learning in edge applications [29, 79]. An all-analogue RC system was recently demonstrated by Zhong et al. [30] using dynamic resistors as a reservoir and an array of non-volatile memristors in the readout module.

Other studies have exploited the structure of memristor cross-bar arrays to execute matrix–vector multiplication used in conventional machine learning algorithms for MNIST classification, both in experiment [80, 81] and simulation [82, 83], although crosstalk in memristor cross-bars limits the accuracy of classification implemented in this type of hardware [80].

Beyond physical RC, unconventional physical systems like NWNs could potentially be trained with backpropagation to realise more energy efficient machine learning than is currently possible with existing software and hardware accelerator approaches [84]. Furthermore, a related study by Loeffler et al. [85] (see also [86, 87]) demonstrates how the internal states of NWNs can be controlled by external feedback to harness NWN working memory capacity and enable cognitive tasks to be performed.

Information-theoretic measures like mutual information (MI) have been widely used to assess the intrinsic dynamics in random Boolean networks [88, 89], Ising models [90], and the learning process of echo state networks [91] as well as artificial neural networks (ANNs) [92]. In a previous simulation study [62, 64], we found that transfer entropy and active information storage in NWNs reveal that specific parts of the network exhibit richer information dynamics during learning tasks and we proposed a scheme for optimising task performance accordingly. However, such element–wise calculations are not feasible for physical NWN hardware devices because the number of readouts from the system is limited by the size of the MEA. In this study, we applied a similar approach to that used in Shine et al. [92] to estimate the information content of ANNs at different stages during the MNIST classification task. They found unequal credit assignment, with some image pixels, as well as specific neurons and weights in the ANN, contributing more to learning than others. In our case, by investigating the information content embedded in the NWN readouts, we found that the learning process synchronises with the information provided by the dataset in the temporal domain, while each readout channel provides distinct information about different classes. Interestingly, we also observed some indication of channel preference for a specific digit class, which could potentially be further exploited for channel–wise tuning for other learning tasks.

The sequence memory task introduced in this study is novel and demonstrates both online learning and sequence memory recall from the memory patterns embedded in NWN dynamics. In the brain, memory patterns are linked with network attractor states [93]. The

brain's neural network is able to remember sequence inputs by evolving the internal states to fixed points that define the memory pattern for the sequence [94]. In this study, we also found basins of attraction for the individual digits in the sequence, which allowed us to reconstruct the target digit image as a way of recalling the associated memory pattern. Delayed dynamics similar to that observed in the conductance time-series of NWNs were also utilised by Voelker et al. [95] to build spiking recurrent neural networks [96] and implement memory-related tasks. In their studies, the delayed dynamics and memory are implemented in software-based learning algorithms, while NWNs are able to retain memory in hardware due to the memristive junctions and recurrent structure [33]. A similar study by Payvand et al. [97] demonstrated sequence learning using spiking recurrent neural networks implemented in ReRAM to exploit the memory property of this resistive memory hardware. Although their sequence was more repetitive than ours and task performance is measured differently, they demonstrated improved performance when network weights were allowed to self-organise and adapt to changing input, similar to physical NWNs. Future potential applications like natural language processing and image analysis may be envisaged with NWN devices that exploit their capability of learning and memorising dynamic sequences. Future computational applications of NWNs may be realised under new computing paradigms grounded in observations and measurements of physical systems beyond the Turing Machine concept [98].

In conclusion, we have demonstrated how neuromorphic nanowire network devices can be used to perform tasks in an online manner, learning from the rich spatiotemporal dynamics generated by the physical neural-like network. This is fundamentally different from data-driven statistical machine learning using artificial neural network algorithms. Additionally, our results demonstrate how online learning and recall of streamed sequence patterns are linked to the associated memory patterns embedded in the spatiotemporal dynamics.

7.4 Methods

7.4.1 Experimental setup

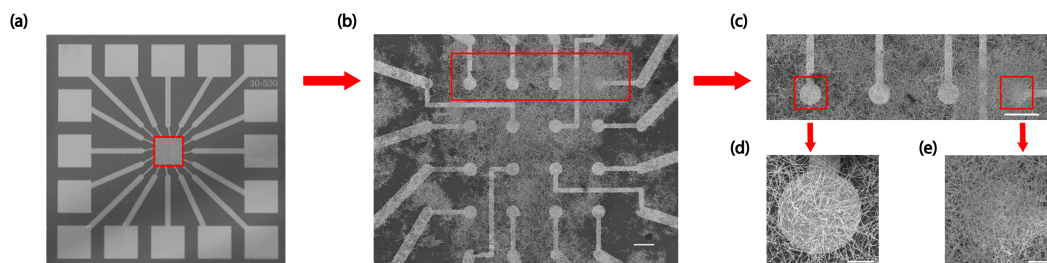


Fig. 7.9 **Images of NWN device.** (a) Optical image of the multi-electrode array. Input/output are enabled by the outer electrodes. (b) Scanning electron microscopy (SEM) image of the Ag_2Se Network. 16 inner electrodes are fabricated as a 4×4 grid and the nanowires are drop-casted on top of them. Scale bar: $100 \mu\text{m}$. (c) Zoom-in of the SEM image for electrodes 0-3. Scale bar: $100 \mu\text{m}$. (d) Zoom-in for electrode 0. Scale bar: $20 \mu\text{m}$. (e) Zoom-in for electrode 3. Scale bar: $20 \mu\text{m}$.

An NWN device was fabricated and characterised following the procedure developed in our previous studies (e.g. [34–36, 49, 59]). Briefly, a multi-electrode array (MEA) device with 16 electrodes (4×4 grid) was fabricated as the substrate of the device using photolithographically patterned Cr/Ti (5 nm) and Pt (150 nm). Selenium nanowires were first formed by hydrothermal reduction of sodium selenite. Ag_2Se nanowires were then synthesised by redispersing Se nanowires in a solution of silver nitrate (AgNO_3). The resulting nanowire solution was drop-casted over the inner electrodes of the MEA to synthesise the nanowire network (See Supplementary Figure S5 for SEM images of the NWN without electrodes and Supplementary Figure S6 for a simulated NWN as well as its corresponding graph representation). A data acquisition device (PXI-6368) was employed to deliver electric signals to the network and simultaneously read out the voltage time-series from all electrodes. A source measurement unit (PXI-4141) was used to collect the current time-series through the grounded electrode. A switch matrix (TB-2642) was used to modularly route tall signals through the network as desired. All the equipment listed above was from National Instruments and controlled by a custom-made LabView package designed for these applications [49, 59, 99]. The readout voltage data exhibited non-uniform phase shifts of $10 - 100\Delta t$ compared to the input stream, so a phase correction method was applied to prepare the readout data for further usage (see details in the following section).

7.4.2 Online learning

Learning tasks were performed under a reservoir computing (RC) framework [58, 59, 62, 99]. With N digit samples used for training, the respective pixel intensities were normalised to $[0.1, 1]$ V as input voltage values and denoted by $\mathbf{U} \in \mathbf{R}^{N \times 784}$ for future reference. $\mathbf{U}$ was then converted to a 1-D temporal voltage pulse stream and delivered to an input channel while another channel was grounded. Each voltage pulse occupied $\Delta t = 0.001$ s in the stream. Voltage features were read simultaneously from M other channels on the device (see Supplementary Figure S7 for device setups). These temporal readout features were normalised and re-arranged to a 3-D array, $\mathbf{V} \in \mathbf{R}^{N \times M \times 784}$.

The phase of the readout voltage data (V) was adjusted per instance in the dataset based on the corresponding input using cross-correlation [100]. For the n -th digit sample, the respective segment in the input pulse stream was denoted as $\mathbf{u}_n \in \mathbf{R}^{784 \times 1}$, and the corresponding dynamical features from M readout channels were represented by $[\mathbf{v}_{n,1}, \mathbf{v}_{n,2}, \dots, \mathbf{v}_{n,m}]$, where $\mathbf{v}_{n,m} \in \mathbf{R}^{784 \times 1}$. The cross-correlation of $\mathbf{u}_n$ and $\mathbf{v}_{n,m}$ is calculated as:

$$C_{n,m}(\tau) = (\mathbf{u}_n * \mathbf{v}_{n,m})(\tau) = \sum_{t=1}^{784} \mathbf{u}_n(t) \mathbf{v}_{n,m}(t - \tau), \quad (7.1)$$

for $\tau = -783, -782, \dots, 0, \dots, 783$. The phase difference ϕ is determined by:

$$\phi_{n,m} = \underset{\tau}{\operatorname{argmax}}(C_{n,m}(\tau)). \quad (7.2)$$

The 1-D phase adjustment was applied to the readout feature $\mathbf{v}_{n,m}$ of the instance based on the phase difference $\phi_{n,m}$.

The NWN device readouts embed dynamical features that are linearly separable, so classification can be performed in a linear output layer:

$$\mathbf{W}\mathbf{A} = \mathbf{Y}, \quad (7.3)$$

where $\mathbf{W}$ is the weight matrix (i.e. classifier), $\mathbf{A}$ is the readout feature space and $\mathbf{Y}$ contains the sample classes. An online method was implemented based on Greville's iterative algorithm for computing the pseudoinverse of linear systems [101]. This method is also a special case of the recursive least square (RLS) algorithm [69], using a uniform sample weighting factor of $\lambda = 1$.

The sample feature space was denoted by $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$, $\mathbf{A} \in \mathbf{R}^{K \times N}$, in which each column ($\mathbf{a}_n$) represented one sample and every sample was composed of K features ($K =$

784M). The corresponding classes for each sample were $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]$, $\mathbf{Y} \in \mathbf{R}^{10 \times N}$. The order of the columns in $\mathbf{A}$ and $\mathbf{Y}$ are randomly shuffled to smooth the learning curve. During training, a new feature vector of the n -th digit sample $\mathbf{a}_n$ and its corresponding class vector $\mathbf{y}_n$ were appended to the right of respective matrices $\mathbf{A}_n$ and $\mathbf{Y}_n$ as columns, and the algorithm solved eqn. 7.3 for $\mathbf{W}$ ($\mathbf{W} \in \mathbf{R}^{10 \times K}$) incrementally. The difference between the target $\mathbf{y}_n$ and the projected result using the previous weight matrix $\mathbf{W}_{n-1}$ was described by:

$$\mathbf{e}_n = \mathbf{y}_n - \text{softmax}(\mathbf{W}_{n-1} \mathbf{a}_n) \quad . \quad (7.4)$$

When $\mathbf{e}_n$ was below a preset threshold $e' = 0.1$, $\mathbf{W}$ was updated by:

$$\mathbf{W}_n = \mathbf{W}_{n-1} + \mathbf{e}_n \mathbf{b}_n^T \quad , \quad (7.5)$$

where

$$\mathbf{b}_n = \frac{\boldsymbol{\theta}_{n-1} \mathbf{a}_n}{\mathbf{1} + \mathbf{a}_n^T \boldsymbol{\theta}_{n-1} \mathbf{a}_n} \quad , \quad (7.6)$$

with

$$\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} - \boldsymbol{\theta}_{n-1} \mathbf{a}_n \mathbf{b}_n^T. \quad (7.7)$$

For the cases when $\mathbf{e}_n$ was above the threshold, an error-correction scheme was applied to optimise the result [68]. In addition, $\mathbf{A}$, $\mathbf{Y}$ and $\boldsymbol{\theta}$ were initialised at $n = 0$ by:

$$\mathbf{A}_0 = \varepsilon \mathbf{I}, \quad \mathbf{Y}_0 = \mathbf{0}, \quad \text{and} \quad \boldsymbol{\theta}_0 = \frac{1}{\varepsilon^2} \mathbf{I}, \quad (7.8)$$

with $\varepsilon = \overline{|\mathbf{A}|}$.

7.4.3 Mutual information

To gain deeper insight into the network's behaviour and attribute real-time learning to its dynamics, mutual information (MI) between the dynamical features and the corresponding classes was calculated to estimate the information content, in a way similar to a previous study on ANNs [92]. All MI results were calculated using the Java Information Dynamics Toolkit (JIDT) [102]. MI was estimated spatially based on the pixel positions from different readout channels, and temporally as the feature space expanded when more samples were learned. Among the N digit samples delivered to the network, an ensemble was created using the readout data from channel m at the i -th pixel position: $\mathbf{V}_{m,i} = [v_{1,m,i}, v_{2,m,i}, \dots, v_{N,m,i}]$, $\mathbf{V}_{m,i} \in \mathbf{R}^{1 \times N}$. Another class vector $\mathbf{P} \in \mathbf{R}^{1 \times N}$ was

created and mutual information was estimated accordingly by:

$$\mathcal{M} = \Omega_{\text{MI}}[\mathbf{V}_{m,i}, \mathbf{P}] \quad , \quad (7.9)$$

where Ω_{MI} stands for the mutual information operator, where the Kraskov estimator was employed [103].

A 3-D matrix $\mathcal{M} \in \mathbf{R}^{N \times M \times 784}$ was generated after calculating spatial-temporally throughout $\mathbf{V}$. $\mathcal{M}$ was averaged across the pixel axis (third) to obtain the temporal mutual information per channel. The spatial analysis of mutual information was based on the calculation result for the whole dataset. The class-wise interpretation of $\mathcal{M}$ was generated by averaging across samples corresponding to each digit class.

7.4.4 Sequence memory task

A sequence-based memory task was developed to investigate sequence memory and recall. Samples of an 8-digit sequence with a semi-repetitive pattern (14751479) were constructed by randomly sampling the respective digits from the MNIST dataset. Input pixel intensities were normalised to the range $[0, 0.1]$ V and the samples were streamed into and read out from the NWN in the same way as the classification task, using channels 9, 8 and 7 for input, ground and readout, respectively. In addition to dynamical features from the voltage readouts, memory features were used from the network conductance, calculated pixel-wise by

$$G_i = \frac{\mathcal{J}_i}{U_i} \quad , \quad (7.10)$$

where $\mathcal{J}$ is the current leaving the ground channel and $\mathbf{U}$ is the input voltage.

To test recall, a digit from the sequence was selected and its image was reconstructed from voltage readouts and memory features in the conductance time series corresponding to digits later in the sequence. A variable memory window of length $L \in [2, 8]$ determines the sequence portion used to reconstruct a previous digit image, i.e. from $L - 1$ subsequent digits. For example, a moving window of length $L = 4$ reconstructs the first (target) digit from the conductance memory features in the subsequent 3 digits (cf. Fig. 7.6). By placing the target digits and memory features into ensembles, a dataset of 7000 training samples and 500 testing samples was composed using the sliding windows.

To reconstruct each target digit image, the same linear online learning algorithm used for MNIST classification was applied. In this case, $\mathbf{Y}$ in eqn. 7.3 was composed as $\mathbf{Y} =$

$[\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n]$, with $\mathbf{Y} \in \mathbf{R}^{784 \times N}$, and softmax in eqn. 7.4 was no longer used. Structural similarity index measure (SSIM) [104] was employed to quantify the reconstruction quality.

To further test that image reconstruction exploits memory features and not just dynamical features associated with the spatial pattern of the sequence (i.e. sequence classification), a memory exclusion test was developed as follows. The conductance features corresponding to a specified number of columns of inputs were replaced by voltage features from channel 13 (voltages are adjusted to the same scale as conductance), so that the memory in conductance is excluded without losing the non-linear features in the readout data (cf. Fig. 7.6). The target digit was then reconstructed for a varying number of columns with memory exclusion.

References

- [1] Mead, C. Neuromorphic electronic systems. *Proceedings of the IEEE* **78** (10), 1629–1636 (1990). doi:[10.1109/5.58356](https://doi.org/10.1109/5.58356) .
- [2] Indiveri, G. *et al.* Neuromorphic Silicon Neuron Circuits. *Frontiers in Neuroscience* **5** (2011). doi:[10.3389/fnins.2011.00073](https://doi.org/10.3389/fnins.2011.00073) .
- [3] Schuman, C. D. *et al.* A survey of neuromorphic computing and neural networks in hardware. *arXiv* (2017). doi:[10.48550/ARXIV.1705.06963](https://doi.org/10.48550/ARXIV.1705.06963) .
- [4] Ielmini, D. & Wong, H.-S. P. In-memory computing with resistive switching devices. *Nature Electronics* **1** (6), 333–343 (2018). doi:[10.1038/s41928-018-0092-2](https://doi.org/10.1038/s41928-018-0092-2) .
- [5] Kendall, J. D. & Kumar, S. The building blocks of a brain-inspired computer. *Applied Physics Reviews* **7** (1), 011305 (2020). doi:[10.1063/1.5129306](https://doi.org/10.1063/1.5129306) .
- [6] Mehonic, A. *et al.* Memristors—From In-Memory Computing, Deep Learning Acceleration, and Spiking Neural Networks to the Future of Neuromorphic and Bio-Inspired Computing. *Advanced Intelligent Systems* **2** (11), 2000085 (2020). doi:[10.1002/aisy.202000085](https://doi.org/10.1002/aisy.202000085) .
- [7] Sebastian, A., Le Gallo, M., Khaddam-Aljameh, R. & Eleftheriou, E. Memory devices and applications for in-memory computing. *Nature Nanotechnology* **15** (7), 529–544 (2020). doi:[10.1038/s41565-020-0655-z](https://doi.org/10.1038/s41565-020-0655-z) .
- [8] Zhang, W. *et al.* Neuro-inspired computing chips. *Nature Electronics* **3** (7), 371–382 (2020). doi:[10.1038/s41928-020-0435-7](https://doi.org/10.1038/s41928-020-0435-7) .
- [9] Zhu, J., Zhang, T., Yang, Y. & Huang, R. A comprehensive review on emerging artificial neuromorphic devices. *Applied Physics Reviews* **7** (1), 011312 (2020). doi:[10.1063/1.5118217](https://doi.org/10.1063/1.5118217) .
- [10] Mehonic, A. & Kenyon, A. J. Brain-inspired computing needs a master plan. *Nature* **604** (7905), 255–260 (2022). doi:[10.1038/s41586-021-04362-w](https://doi.org/10.1038/s41586-021-04362-w) .
- [11] Christensen, D. V. *et al.* 2022 roadmap on neuromorphic computing and engineering. *Neuromorphic Computing and Engineering* **2** (2), 022501 (2022). doi:[10.1088/2634-4386/ac4a83](https://doi.org/10.1088/2634-4386/ac4a83) .

- [12] Pfeil, T. *et al.* Six Networks on a Universal Neuromorphic Computing Substrate. *Frontiers in Neuroscience* **7** (2013). doi:[10.3389/fnins.2013.00011](https://doi.org/10.3389/fnins.2013.00011) .
- [13] Merolla, P. A. *et al.* A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345** (6197), 668–673 (2014). doi:[10.1126/science.1254642](https://doi.org/10.1126/science.1254642) .
- [14] Thakur, C. S. *et al.* Large-Scale Neuromorphic Spiking Array Processors: A Quest to Mimic the Brain. *Frontiers in Neuroscience* **12** (2018). doi:[10.3389/fnins.2018.00891](https://doi.org/10.3389/fnins.2018.00891) .
- [15] Bouvier, M. *et al.* Spiking Neural Networks Hardware Implementations and Challenges: A Survey. *ACM Journal on Emerging Technologies in Computing Systems* **15** (2), 22:1–22:35 (2019). doi:[10.1145/3304103](https://doi.org/10.1145/3304103) .
- [16] Roy, K., Jaiswal, A. & Panda, P. Towards spike-based machine intelligence with neuromorphic computing. *Nature* **575** (7784), 607–617 (2019). doi:[10.1038/s41586-019-1677-2](https://doi.org/10.1038/s41586-019-1677-2) .
- [17] Eshraghian, J. K., Wang, X. & Lu, W. D. Memristor-Based Binarized Spiking Neural Networks: Challenges and applications. *IEEE Nanotechnology Magazine* **16** (2), 14–23 (2022). doi:[10.1109/MNANO.2022.3141443](https://doi.org/10.1109/MNANO.2022.3141443) .
- [18] Bose, S. K. *et al.* Evolution of a designless nanoparticle network into reconfigurable Boolean logic. *Nature Nanotechnology* **10** (12), 1048–1052 (2015). doi:[10.1038/nnano.2015.207](https://doi.org/10.1038/nnano.2015.207) .
- [19] Grollier, J. *et al.* Neuromorphic spintronics. *Nature Electronics* **3** (7), 360–370 (2020). doi:[10.1038/s41928-019-0360-9](https://doi.org/10.1038/s41928-019-0360-9) .
- [20] Sangwan, V. K. & Hersam, M. C. Neuromorphic nanoelectronic materials. *Nature Nanotechnology* **15** (7), 517–528 (2020). doi:[10.1038/s41565-020-0647-z](https://doi.org/10.1038/s41565-020-0647-z) .
- [21] Tanaka, H. *et al.* In-materio computing in random networks of carbon nanotubes complexed with chemically dynamic molecules: A review. *Neuromorphic Computing and Engineering* **2** (2), 022002 (2022). doi:[10.1088/2634-4386/ac676a](https://doi.org/10.1088/2634-4386/ac676a) .
- [22] Kuncic, Z., Nakayama, T. & Gimzewski, J. Editorial: Focus on disordered, self-assembled neuromorphic systems. *Neuromorphic Computing and Engineering* **2** (4), 040201 (2022). doi:[10.1088/2634-4386/ac91a0](https://doi.org/10.1088/2634-4386/ac91a0) .

- [23] Waser, R. & Aono, M. Nanoionics-based resistive switching memories. *Nature Materials* **6** (11), 833–840 (2007). doi:[10.1038/nmat2023](https://doi.org/10.1038/nmat2023) .
- [24] Ohno, T. *et al.* Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials* **10** (8), 591–595 (2011). doi:[10.1038/nmat3054](https://doi.org/10.1038/nmat3054) .
- [25] Yang, J. J., Strukov, D. B. & Stewart, D. R. Memristive devices for computing. *Nature Nanotechnology* **8** (1), 13–24 (2013). doi:[10.1038/nnano.2012.240](https://doi.org/10.1038/nnano.2012.240) .
- [26] La Barbera, S., Vuillaume, D. & Alibart, F. Filamentary Switching: Synaptic Plasticity through Device Volatility. *ACS Nano* **9** (1), 941–949 (2015). doi:[10.1021/nn506735m](https://doi.org/10.1021/nn506735m) .
- [27] Wang, Z. *et al.* Resistive switching materials for information processing. *Nature Reviews Materials* **5** (3), 173–195 (2020). doi:[10.1038/s41578-019-0159-3](https://doi.org/10.1038/s41578-019-0159-3) .
- [28] Diaz Schneider, J. I. *et al.* Resistive Switching of Self-Assembled Silver Nanowire Networks Governed by Environmental Conditions. *Advanced Electronic Materials* **8** (11), 2200631 (2022). doi:[10.1002/aelm.202200631](https://doi.org/10.1002/aelm.202200631) .
- [29] Kumar, P. *et al.* Hybrid architecture based on two-dimensional memristor cross-bar array and CMOS integrated circuit for edge computing. *npj 2D Materials and Applications* **6** (1), 1–10 (2022). doi:[10.1038/s41699-021-00284-3](https://doi.org/10.1038/s41699-021-00284-3) .
- [30] Zhong, Y. *et al.* A memristor-based analogue reservoir computing system for real-time and power-efficient signal processing. *Nature Electronics* **5** (10), 672–681 (2022). doi:[10.1038/s41928-022-00838-3](https://doi.org/10.1038/s41928-022-00838-3) .
- [31] Kotooka, T., Tanaka, Y., Tamukoh, H., Usami, Y. & Tanaka, H. Random network device fabricated using Ag₂Se nanowires for data augmentation with binarized convolutional neural network. *Applied Physics Express* **16** (1), 014002 (2023). doi:[10.35848/1882-0786/aca6a](https://doi.org/10.35848/1882-0786/aca6a) .
- [32] Milano, G., Porro, S., Valov, I. & Ricciardi, C. Recent Developments and Perspectives for Memristive Devices Based on Metal Oxide Nanowires. *Advanced Electronic Materials* **5** (9), 1800909 (2019). doi:[10.1002/aelm.201800909](https://doi.org/10.1002/aelm.201800909) .
- [33] Kuncic, Z. & Nakayama, T. Neuromorphic nanowire networks: Principles, progress and future prospects for neuro-inspired information processing. *Advances in Physics: X* **6** (1), 1894234 (2021). doi:[10.1080/23746149.2021.1894234](https://doi.org/10.1080/23746149.2021.1894234) .

- [34] Stieg, A. Z. *et al.* Emergent Criticality in Complex Turing B-Type Atomic Switch Networks. *Advanced Materials* **24** (2), 286–293 (2012). doi:[10.1002/adma.201103053](https://doi.org/10.1002/adma.201103053) .
- [35] Avizienis, A. V. *et al.* Neuromorphic Atomic Switch Networks. *PLoS ONE* **7** (8), e42772 (2012). doi:[10.1371/journal.pone.0042772](https://doi.org/10.1371/journal.pone.0042772) .
- [36] Demis, E. C. *et al.* Atomic switch networks—nanoarchitectonic design of a complex system for natural computing. *Nanotechnology* **26** (20), 204003 (2015). doi:[10.1088/0957-4484/26/20/204003](https://doi.org/10.1088/0957-4484/26/20/204003) .
- [37] Diaz-Alvarez, A. *et al.* Emergent dynamics of neuromorphic nanowire networks. *Scientific Reports* **9** (1), 14920 (2019). doi:[10.1038/s41598-019-51330-6](https://doi.org/10.1038/s41598-019-51330-6) .
- [38] Loeffler, A. *et al.* Topological Properties of Neuromorphic Nanowire Networks. *Frontiers in Neuroscience* **14**, 184 (2020). doi:[10.3389/fnins.2020.00184](https://doi.org/10.3389/fnins.2020.00184) .
- [39] Terabe, K., Nakayama, T., Hasegawa, T. & Aono, M. Formation and disappearance of a nanoscale silver cluster realized by solid electrochemical reaction. *Journal of Applied Physics* **91** (12), 10110–10114 (2002). doi:[10.1063/1.1481775](https://doi.org/10.1063/1.1481775) .
- [40] Terabe, K., Hasegawa, T., Nakayama, T. & Aono, M. Quantized conductance atomic switch. *Nature* **433** (7021), 47–50 (2005). doi:[10.1038/nature03190](https://doi.org/10.1038/nature03190) .
- [41] Schoen, D. T., Xie, C. & Cui, Y. Electrical Switching and Phase Transformation in Silver Selenide Nanowires. *Journal of the American Chemical Society* **129** (14), 4116–4117 (2007). doi:[10.1021/ja068365s](https://doi.org/10.1021/ja068365s) .
- [42] Menzel, S., Tappertzhofen, S., Waser, R. & Valov, I. Switching kinetics of electrochemical metallization memory cells. *Physical Chemistry Chemical Physics* **15** (18), 6945 (2013). doi:[10.1039/c3cp50738f](https://doi.org/10.1039/c3cp50738f) .
- [43] Kozicki, M. N., Mitkova, M. & Valov, I. in *Electrochemical Metallization Memories* Ch. 17, 483–514 (John Wiley & Sons, Ltd, 2016).
- [44] Bellew, A. T., Manning, H. G., Gomes da Rocha, C., Ferreira, M. S. & Boland, J. J. Resistance of Single Ag Nanowire Junctions and Their Role in the Conductivity of Nanowire Networks. *ACS Nano* **9** (11), 11422–11429 (2015). doi:[10.1021/acs.nano.5b05469](https://doi.org/10.1021/acs.nano.5b05469) .

- [45] Manning, H. G. *et al.* Emergence of winner-takes-all connectivity paths in random nanowire networks. *Nature Communications* **9** (1), 3219 (2018). doi:[10.1038/s41467-018-05517-6](https://doi.org/10.1038/s41467-018-05517-6) .
- [46] Milano, G. *et al.* Brain-inspired structural plasticity through reweighting and rewiring in multi-terminal self-organizing memristive nanowire networks. *Advanced Intelligent Systems* **2** (8), 2000096 (2020). doi:<https://doi.org/10.1002/aisy.202000096> .
- [47] Mallinson, J. B. *et al.* Avalanches and criticality in self-organized nanoscale networks. *Science Advances* **5** (11), eaaw8438 (2019). doi:[10.1126/sciadv.aaw8438](https://doi.org/10.1126/sciadv.aaw8438) .
- [48] Hochstetter, J. *et al.* Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nature Communications* **12** (1), 4008 (2021). doi:[10.1038/s41467-021-24260-z](https://doi.org/10.1038/s41467-021-24260-z) .
- [49] Dunham, C. S. *et al.* Nanoscale neuromorphic networks and criticality: A perspective. *Journal of Physics: Complexity* **2** (4), 042001 (2021). doi:[10.1088/2632-072X/ac3ad3](https://doi.org/10.1088/2632-072X/ac3ad3) .
- [50] Milano, G., Cultrera, A., Boarino, L., Callegaro, L. & Ricciardi, C. Tomography of memory engrams in self-organizing nanowire connectomes. *Nature Communications* **14** (1), 5723 (2023). doi:[10.1038/s41467-023-40939-x](https://doi.org/10.1038/s41467-023-40939-x) .
- [51] Medsker, L. & Jain, L. C. *Recurrent Neural Networks: Design and Applications* (CRC Press, 1999).
- [52] Lipton, Z. C., Berkowitz, J. & Elkan, C. A Critical Review of Recurrent Neural Networks for Sequence Learning 38 (2015) .
- [53] Shen, S. *et al.* Reservoir transformers. *arXiv* (2020). doi:[10.48550/ARXIV.2012.15045](https://doi.org/10.48550/ARXIV.2012.15045) .
- [54] Lukoševičius, M. & Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Science Review* **3** (3), 127–149 (2009). doi:[10.1016/j.cosrev.2009.03.005](https://doi.org/10.1016/j.cosrev.2009.03.005) .
- [55] Klos, C., Kalle Kossio, Y. F., Goedeke, S., Gilra, A. & Memmesheimer, R.-M. Dynamical Learning of Dynamics. *Physical Review Letters* **125** (8), 088103 (2020). doi:[10.1103/PhysRevLett.125.088103](https://doi.org/10.1103/PhysRevLett.125.088103) .

- [56] Gauthier, D. J., Bollt, E., Griffith, A. & Barbosa, W. A. S. Next generation reservoir computing. *Nature Communications* **12** (1), 5564 (2021). doi:[10.1038/s41467-021-25801-2](https://doi.org/10.1038/s41467-021-25801-2) .
- [57] Tanaka, G. *et al.* Recent advances in physical reservoir computing: A review. *Neural Networks* **115**, 100–123 (2019). doi:[10.1016/j.neunet.2019.03.005](https://doi.org/10.1016/j.neunet.2019.03.005) .
- [58] Sillin, H. O. *et al.* A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology* **24** (38), 384004 (2013). doi:[10.1088/0957-4484/24/38/384004](https://doi.org/10.1088/0957-4484/24/38/384004) .
- [59] Lilak, S. *et al.* Spoken Digit Classification by In-Materio Reservoir Computing With Neuromorphic Atomic Switch Networks. *Frontiers in Nanotechnology* **3** (2021). doi:[10.3389/fnano.2021.675792](https://doi.org/10.3389/fnano.2021.675792) .
- [60] Fu, K. *et al.* *Reservoir Computing with Neuromemristive Nanowire Networks*, 1–8 (2020).
- [61] Zhu, R. *et al.* *Harnessing adaptive dynamics in neuro-memristive nanowire networks for transfer learning*, 102–106 (2020).
- [62] Zhu, R. *et al.* *MNIST classification using Neuromorphic Nanowire Networks*, ICONS 2021, 1–4 (Association for Computing Machinery, New York, NY, USA, 2021).
- [63] Loeffler, A. *et al.* Modularity and multitasking in neuro-memristive reservoir networks. *Neuromorphic Computing and Engineering* **1** (1), 014003 (2021). doi:[10.1088/2634-4386/ac156f](https://doi.org/10.1088/2634-4386/ac156f) .
- [64] Zhu, R. *et al.* Information dynamics in neuromorphic nanowire networks. *Scientific Reports* **11** (1), 13047 (2021). doi:[10.1038/s41598-021-92170-7](https://doi.org/10.1038/s41598-021-92170-7) .
- [65] Milano, G. *et al.* In materia reservoir computing with a fully memristive architecture based on self-organizing nanowire networks. *Nature Materials* (2021). doi:[10.1038/s41563-021-01099-9](https://doi.org/10.1038/s41563-021-01099-9) .
- [66] Cucchi, M., Abreu, S., Ciccone, G., Brunner, D. & Kleemann, H. Hands-on reservoir computing: A tutorial for practical implementation. *Neuromorphic Computing and Engineering* **2** (3), 032002 (2022). doi:[10.1088/2634-4386/ac7db7](https://doi.org/10.1088/2634-4386/ac7db7) .
- [67] Hoi, S. C. H., Sahoo, D., Lu, J. & Zhao, P. Online learning: A comprehensive survey. *Neurocomputing* **459**, 249–289 (2021). doi:[10.1016/j.neucom.2021.04.112](https://doi.org/10.1016/j.neucom.2021.04.112) .

- [68] Tapson, J. & van Schaik, A. Learning the pseudoinverse solution to network weights. *Neural Networks* **45**, 94–100 (2013). doi:[10.1016/j.neunet.2013.02.008](https://doi.org/10.1016/j.neunet.2013.02.008) .
- [69] Farhang-Boroujeny, B. *Adaptive Filters: Theory and Applications* Second edn (John Wiley & Sons, 2013).
- [70] Fontenla-Romero, Ó., Guijarro-Berdiñas, B., Martinez-Rego, D., Pérez-Sánchez, B. & Peteiro-Barral, D. in *Online machine learning* 27–54 (IGI global, 2013).
- [71] Gomes, H. M., Read, J., Bifet, A., Barddal, J. P. & Gama, J. Machine learning for streaming data: State of the art, challenges, and opportunities. *ACM SIGKDD Explorations Newsletter* **21** (2), 6–22 (2019). doi:[10.1145/3373464.3373470](https://doi.org/10.1145/3373464.3373470) .
- [72] Kudithipudi, D. *et al.* Biological underpinnings for lifelong learning machines. *Nature Machine Intelligence* **4** (3), 196–210 (2022). doi:[10.1038/s42256-022-00452-0](https://doi.org/10.1038/s42256-022-00452-0) .
- [73] Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86** (11), 2278–2324 (1998). doi:[10.1109/5.726791](https://doi.org/10.1109/5.726791) .
- [74] Du, C. *et al.* Reservoir computing using dynamic memristors for temporal information processing. *Nature Communications* **8** (1), 2204 (2017). doi:[10.1038/s41467-017-02337-y](https://doi.org/10.1038/s41467-017-02337-y) .
- [75] Midya, R. *et al.* Reservoir Computing Using Diffusive Memristors. *Advanced Intelligent Systems* **1** (7), 1900084 (2019). doi:[10.1002/aisy.201900084](https://doi.org/10.1002/aisy.201900084) .
- [76] Kendall, J. D., Pantone, R. D. & Nino, J. C. Deep Learning in Memristive Nanowire Networks. *arXiv* (arXiv:2003.02642) (2020). [arXiv:2003.02642](https://arxiv.org/abs/2003.02642) [cs].
- [77] Jaeger, H. The “echo state” approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report* **148** (34), 13 (2001) .
- [78] Jaeger, H. Adaptive Nonlinear System Identification with Echo State Networks. *NIPS* (2003) .
- [79] Alaba, P. A. *et al.* Towards a more efficient and cost-sensitive extreme learning machine: A state-of-the-art review of recent trend. *Neurocomputing* **350**, 70–90 (2019). doi:[10.1016/j.neucom.2019.03.086](https://doi.org/10.1016/j.neucom.2019.03.086) .

- [80] Choi, S. *et al.* A self-rectifying TaOy/nanoporous TaOx memristor synaptic array for learning and energy-efficient neuromorphic systems. *NPG Asia Materials* **10** (12), 1097–1106 (2018). doi:[10.1038/s41427-018-0101-y](https://doi.org/10.1038/s41427-018-0101-y) .
- [81] Yao, P. *et al.* Fully hardware-implemented memristor convolutional neural network. *Nature* **577** (7792), 641–646 (2020). doi:[10.1038/s41586-020-1942-4](https://doi.org/10.1038/s41586-020-1942-4) .
- [82] Liu, X. & Zeng, Z. Memristor crossbar architectures for implementing deep neural networks. *Complex & Intelligent Systems* **8** (2), 787–802 (2022). doi:[10.1007/s40747-021-00282-4](https://doi.org/10.1007/s40747-021-00282-4) .
- [83] Mao, R., Wen, B., Jiang, M., Chen, J. & Li, C. Experimentally-Validated Crossbar Model for Defect-Aware Training of Neural Networks. *IEEE Transactions on Circuits and Systems II: Express Briefs* **69** (5), 2468–2472 (2022). doi:[10.1109/TCSII.2022.3160591](https://doi.org/10.1109/TCSII.2022.3160591) .
- [84] Wright, L. G. *et al.* Deep physical neural networks trained with backpropagation. *Nature* **601** (7894), 549–555 (2022). doi:[10.1038/s41586-021-04223-6](https://doi.org/10.1038/s41586-021-04223-6) .
- [85] Loeffler, A. *et al.* Neuromorphic learning, working memory, and metaplasticity in nanowire networks. *SCIENCE ADVANCES* (2023). doi:[10.1126/sciadv.adg3289](https://doi.org/10.1126/sciadv.adg3289) .
- [86] Diaz-Alvarez, A., Higuchi, R., Li, Q., Shingaya, Y. & Nakayama, T. Associative routing through neuromorphic nanowire networks. *AIP Advances* **10** (2), 025134 (2020). doi:[10.1063/1.5140579](https://doi.org/10.1063/1.5140579), <https://doi.org/10.1063/1.5140579> .
- [87] Li, Q. *et al.* Dynamic electrical pathway tuning in neuromorphic nanowire networks. *Advanced Functional Materials* **30** (43), 2003679 (2020). doi:<https://doi.org/10.1002/adfm.202003679>, <https://onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.202003679> .
- [88] Lizier, J., Prokopenko, M. & Zomaya, A. The Information Dynamics of Phase Transitions in Random Boolean Networks. *Proc. Eleventh International Conference on the Simulation and Synthesis of Living Systems (ALife XI)* (2008) .
- [89] Lizier, J. T., Pritam, S. & Prokopenko, M. Information Dynamics in Small-World Boolean Networks. *Artificial Life* **17** (4), 293–314 (2011). doi:[10.1162/artl_a_00040](https://doi.org/10.1162/artl_a_00040) .
- [90] Barnett, L., Lizier, J. T., Harré, M., Seth, A. K. & Bossomaier, T. Information Flow in a Kinetic Ising Model Peaks in the Disordered Phase. *Physical Review Letters* **111** (17), 177203 (2013). doi:[10.1103/PhysRevLett.111.177203](https://doi.org/10.1103/PhysRevLett.111.177203) .

- [91] Boedecker, J., Obst, O., Lizier, J. T., Mayer, N. M. & Asada, M. Information processing in echo state networks at the edge of chaos. *Theory in Biosciences* **131** (3), 205–213 (2012). doi:[10.1007/s12064-011-0146-8](https://doi.org/10.1007/s12064-011-0146-8) .
- [92] Shine, J. M., Li, M., Koyejo, O., Fulcher, B. & Lizier, J. T. Nonlinear Reconfiguration of Network Edges, Topology and Information Content During an Artificial Learning Task (2021).
- [93] Khona, M. & Fiete, I. R. Attractor and integrator networks in the brain. *Nature Reviews Neuroscience* **23** (12), 744–766 (2022). doi:[10.1038/s41583-022-00642-0](https://doi.org/10.1038/s41583-022-00642-0) .
- [94] Dayan, P. & Abbott, L. F. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems* Computational Neuroscience (Massachusetts Institute of Technology Press, Cambridge, Mass, 2001).
- [95] Voelker, A., Kajić, I. & Eliasmith, C. *Legendre Memory Units: Continuous-Time Representation in Recurrent Neural Networks*, Vol. 32 (Curran Associates, Inc., 2019).
- [96] Voelker, A. R. & Eliasmith, C. Improving Spiking Dynamical Networks: Accurate Delays, Higher-Order Synapses, and Time Cells. *Neural Computation* **30** (3), 569–609 (2018). doi:[10.1162/neco_a_01046](https://doi.org/10.1162/neco_a_01046) .
- [97] Payvand, M. *et al.* Self-organization of an inhomogeneous memristive hardware for sequence learning. *Nature Communications* **13** (1), 5793 (2022). doi:[10.1038/s41467-022-33476-6](https://doi.org/10.1038/s41467-022-33476-6) .
- [98] Jaeger, H., Noheda, B. & van der Wiel, W. G. Toward a formal theory for computing machines made out of whatever physics offers. *Nature Communications* **14** (1), 4911 (2023). doi:[10.1038/s41467-023-40533-1](https://doi.org/10.1038/s41467-023-40533-1) .
- [99] Kotooka, T. *et al.* Ag2Se Nanowire Network as an Effective In-Materio Reservoir Computing Device. Preprint (2021).
- [100] Ianniello, J. Time delay estimation via cross-correlation in the presence of large estimation errors. *IEEE Transactions on Acoustics, Speech, and Signal Processing* **30** (6), 998–1003 (1982). doi:[10.1109/TASSP.1982.1163992](https://doi.org/10.1109/TASSP.1982.1163992) .
- [101] Greville, T. N. E. Some Applications of the Pseudoinverse of a Matrix. *SIAM Review* **2** (1), 15–22 (1960). doi:[10.1137/1002004](https://doi.org/10.1137/1002004) .

-
- [102] Lizier, J. T. JIDT: An Information-Theoretic Toolkit for Studying the Dynamics of Complex Systems. *Frontiers in Robotics and AI* **1** (2014). doi:[10.3389/frobt.2014.00011](https://doi.org/10.3389/frobt.2014.00011) .
- [103] Kraskov, A., Stögbauer, H. & Grassberger, P. Estimating mutual information. *Physical Review E* **69** (6), 066138 (2004). doi:[10.1103/PhysRevE.69.066138](https://doi.org/10.1103/PhysRevE.69.066138) .
- [104] Wang, Z., Bovik, A., Sheikh, H. & Simoncelli, E. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* **13** (4), 600–612 (2004). doi:[10.1109/TIP.2003.819861](https://doi.org/10.1109/TIP.2003.819861) .

Chapter 8

Concluding Remarks

“路漫漫其修远兮，
吾将上下而求索。”

*“The road ahead is long and winding,
yet I will search high and low.”*

离骚 Li Sao - 屈原 Qu Yuan

8.1 Summary of Results

The results presented in this thesis reveal novel insights into the field of neuromorphic computing and engineering and a significant step forward in demonstrating learning applications of novel nanomaterial neuromorphic systems in particular.

Throughout this journey with neuromorphic nanowire networks, several pivotal findings have emerged that enrich our understanding of their learning capabilities, demonstrated by using NWNs as reservoirs in a reservoir computing framework. In Chapter 3, by investigating the information dynamics of NWNs during learning tasks. I uncovered a synchronicity between optimal task performance and specific information-theoretic measures (active information storage and transfer entropy). Furthermore, this optimal performance was found to coincide with the characteristic conductance transition, when the network is on the verge of activation. In Chapter 4, I demonstrated the machine learning concept of transfer learning through the Mackey–Glass chaotic time series prediction task. I showed how priming the network on highly chaotic signals improved prediction of less chaotic signals. The simulation results of Chapter 5 are the first to show one possible implementation of the MNIST handwritten digit classification task using NWNs, leveraging a convolutional input kernel to pre-process the spatial data.

In Chapter 6, I showed how the machine learning concept of meta-learning can address the challenge of optimising NWN reservoirs in a systematic way, avoiding ad-hoc trial and error. Interestingly, the optimal initial conditions identified in this chapter align with those highlighted in chapter 3 using information-theoretic measures.

While the findings from the preceding chapters were based on simulated NWNs, in Chapter 7 I translated this accumulated knowledge and insight to a hardware-based NWN reservoir system comprised of a Ag_2Se NWN and a 16-electrode array. The MNIST digit classification task was implemented by delivering the data as a continuous stream and using the recursive least square method, which enabled online learning of the dynamical features of each digit. I devised a novel sequence memory task employing the MNIST digit images and showed how the network's long-term memory embedded in its conductance enhances learning the digit sequence.

The original research results in this thesis offer a comprehensive exploration into NWNs and their learning capabilities, both in simulation and in hardware. These findings underscore the potential of NWNs as a pivotal tool in neuromorphic computing, signifying a promising direction for future explorations in the domain of brain-inspired systems. Moreover,

the adaptability and efficiency exhibited by NWNs in our experiments pave the way for innovations that could bridge the gap between artificial and biological neural processes.

8.2 Future Work and Outlook

The findings presented in this thesis reflect my very first endeavour in crafting brain-inspired learning systems. This journey has also sparked a myriad of intriguing ideas.

8.2.1 Equilibrium Propagation for NWNs

Related studies have successfully demonstrated that current pathways in NWNs can be adeptly adjusted to particular electrodes through associative learning [1, 2]. Loeffler et al. extended these findings by employing AEprop to direct the current pathway for cognitive tasks [3]. For broader learning tasks, achieving control over the localised or global dynamics within NWNs could be explored. Considering the inherent challenge that individual junctions are not readily accessible, equilibrium propagation emerges as a promising solution [4]. This technique offers the ability to nudge the internal dynamics of the system towards a desired direction, even when faced with a restricted number of inputs. The potential of this approach not only broadens the horizon for optimising NWNs but also lays the groundwork for more nuanced control mechanisms in future research.

8.2.2 Interfacing NWNs with other spike-based systems

In recent years, a variety of spike-driven chips have been developed and applied in numerous scenarios. Bridging NWNs with other spike-based systems holds the potential to open the door for a wider range of research opportunities, particularly for edge applications. This endeavour could branch out in multiple directions. For instance, the most intuitive approach is to deliver spikes into NWNs. Given the volatile nature of NWN memristive junctions, the encoded information within spikes can be retained only temporarily within these junctions. Over time, critical information can be reinforced while less crucial data naturally diminishes. Equally fascinating is the potential to employ NWNs as spike generators. In Chapter 6, we utilised NWNs to emulate the membrane potential of spiking neurons, which enables the generation of desired spike patterns. This avenue of exploration can be extended even further, offering exciting prospects for transferring continuous processes into spike-based systems.

8.2.3 Connect multiple NWNs as different brain regions

Another interesting idea is to connect various NWNs together, effectively configuring them to function as distinct brain regions. Much like the human brain possesses dedicated regions for processing different sensory data, such as the visual cortex for vision and the auditory cortex for hearing, the prospect of configuring interconnected NWNs to emulate these specialised functions holds significant promise. For instance, we could allocate one NWN for processing visual input and another for auditory information, while additional NWNs could serve as central information processing hubs. This approach leverages the inherent scalability of NWNs and distributes computational workloads across distinct networks. Such an approach holds great potential for mimicking the multifunctional capabilities of the brain.

8.3 Conclusion and Personal Remarks

In conclusion, this thesis has journeyed through the intricate landscape of NWNs and their potential in neuromorphic learning. This exploration has not only shed light on the inner workings of NWNs but also unveiled a roadmap for the future of brain-inspired learning systems. As the field of neuromorphic computing continues its dynamic evolution, my aspiration is to remain engaged in related endeavours. In my view, the next step forward may require the development of a more systematic and foundational framework, akin to Alan Turing's seminal contributions to modern digital computing. Additionally, I am keen to deepen my understanding of neuroscience, with the aim of achieving more faithful emulation of the brain's intricacies. Furthermore, I intend to explore the realm of devices and chips, seeking hands-on experiences that will further empower my research efforts for hardware-algorithm co-design.

Replicating intelligence in machines may evoke a cyberpunk aura, yet it carries a poetic essence that propels us forward. Let's continue this journey with unwavering momentum...

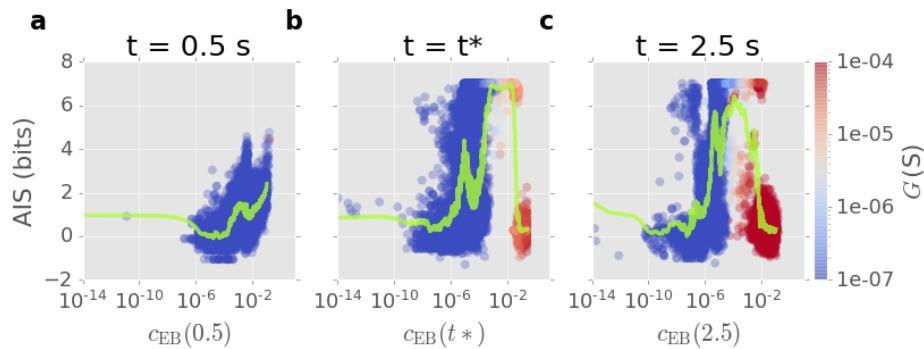
References

- [1] Diaz-Alvarez, A., Higuchi, R., Li, Q., Shingaya, Y. & Nakayama, T. Associative routing through neuromorphic nanowire networks. *AIP Advances* **10** (2), 025134 (2020). doi:[10.1063/1.5140579](https://doi.org/10.1063/1.5140579) .
- [2] Li, Q. *et al.* Dynamic Electrical Pathway Tuning in Neuromorphic Nanowire Networks. *Advanced Functional Materials* **30** (43), 2003679 (2020). doi:[10.1002/adfm.202003679](https://doi.org/10.1002/adfm.202003679) .
- [3] Loeffler, A. *et al.* Neuromorphic learning, working memory, and metaplasticity in nanowire networks. *SCIENCE ADVANCES* (2023). doi:[10.1126/sciadv.adg3289](https://doi.org/10.1126/sciadv.adg3289) .
- [4] Scellier, B. & Bengio, Y. Equilibrium Propagation: Bridging the Gap between Energy-Based Models and Backpropagation. *Frontiers in Computational Neuroscience* **11** (2017). doi:[10.3389/fncom.2017.00024](https://doi.org/10.3389/fncom.2017.00024) .

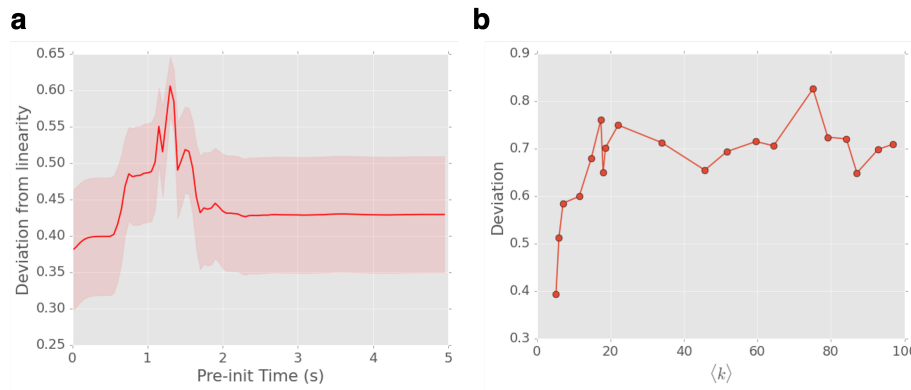
Appendix A

Supplementary materials for original research papers

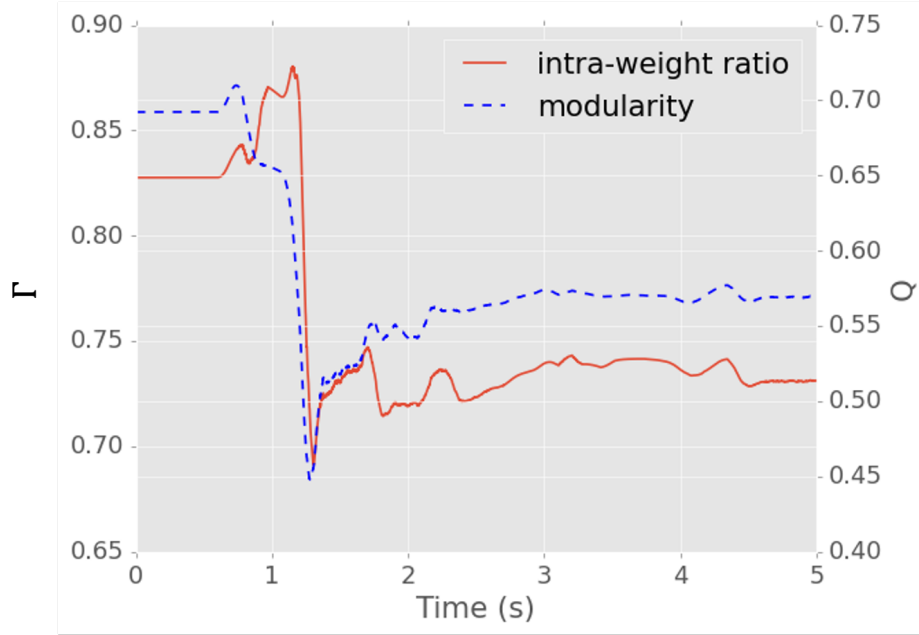
A.1 Supplementary information for Chapter 3



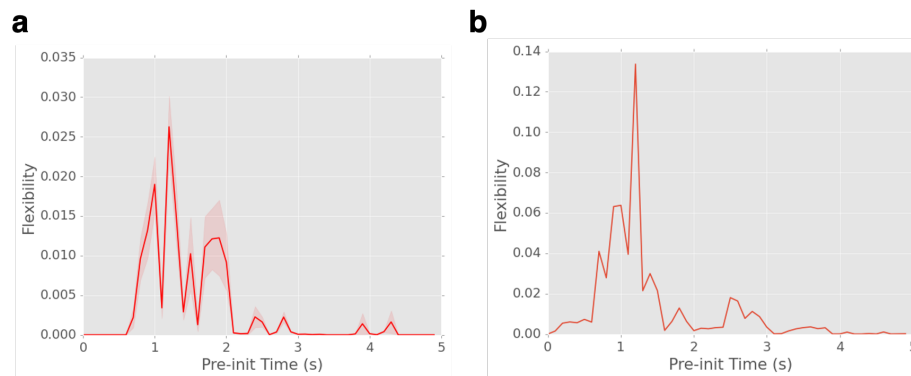
Supplementary Figure S1. Junction active information storage (AIS) as a function of edge betweenness centrality c_{EB} calculated for 50 simulations of a 100-node network with varying source-drain locations (fixed distance) at three different time points: (a) $t = 0.5$ s; (b) $t = t^*$, where $t^* = 1.22$ s is network activation time; and (c) $t = 2.5$ s. Each data point represents one junction at the specified time. The colourbar represents the corresponding junction conductance. The yellow curve is a moving window average (size 0.1 s) of the scatter points calculated by sorting the scatter points based on their c_{EB} values and averaging c_{EB} , AIS values within each window.



Supplementary Figure S2. Deviation from linearity of NWNs. (a) Deviation from linearity as a function of pre-initialisation time for a 100 nanowire/ 261 junction NWN realisation. (b) Deviation from linearity of NWNs as a function of mean connectivity $\langle k \rangle$.

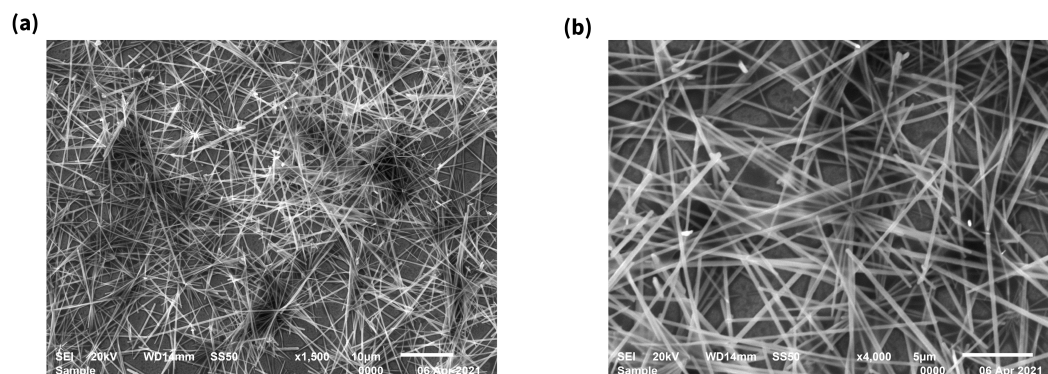


Supplementary Figure S3. Intra-module weight ratio Γ and modularity Q as a function of time. Red curve is the time-series of the sum of weights on intra-module edges normalised by the total weights in the network ($\Gamma = \frac{\sum_{m_i=m_j, i, j \in N} W_{i,j}}{\sum_{i, j \in N} W_{i,j}}$). Dashed blue curve represents the modularity time-series.

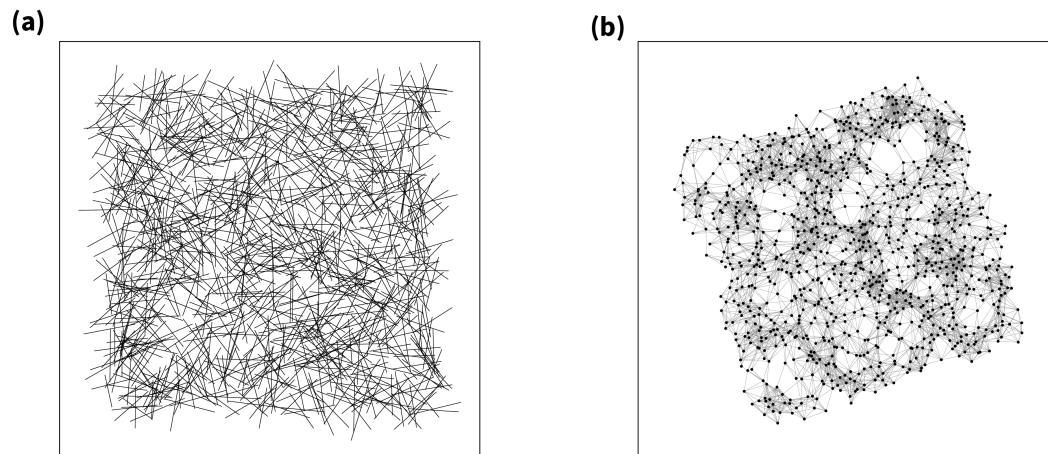


Supplementary Figure S4. Flexibility of NWNs during learning tasks. (a) Flexibility of NWN as a function of pre-initialisation time during the memory capacity (MC) task. (b) Flexibility of NWN as a function of pre-initialisation time during the non-linear transformation (NLT) task.

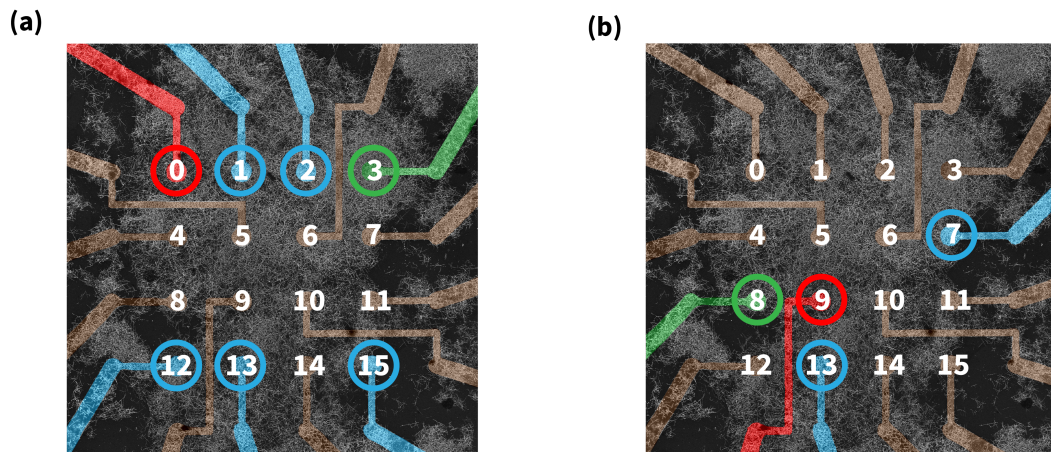
A.2 Supplementary information for Chapter 7



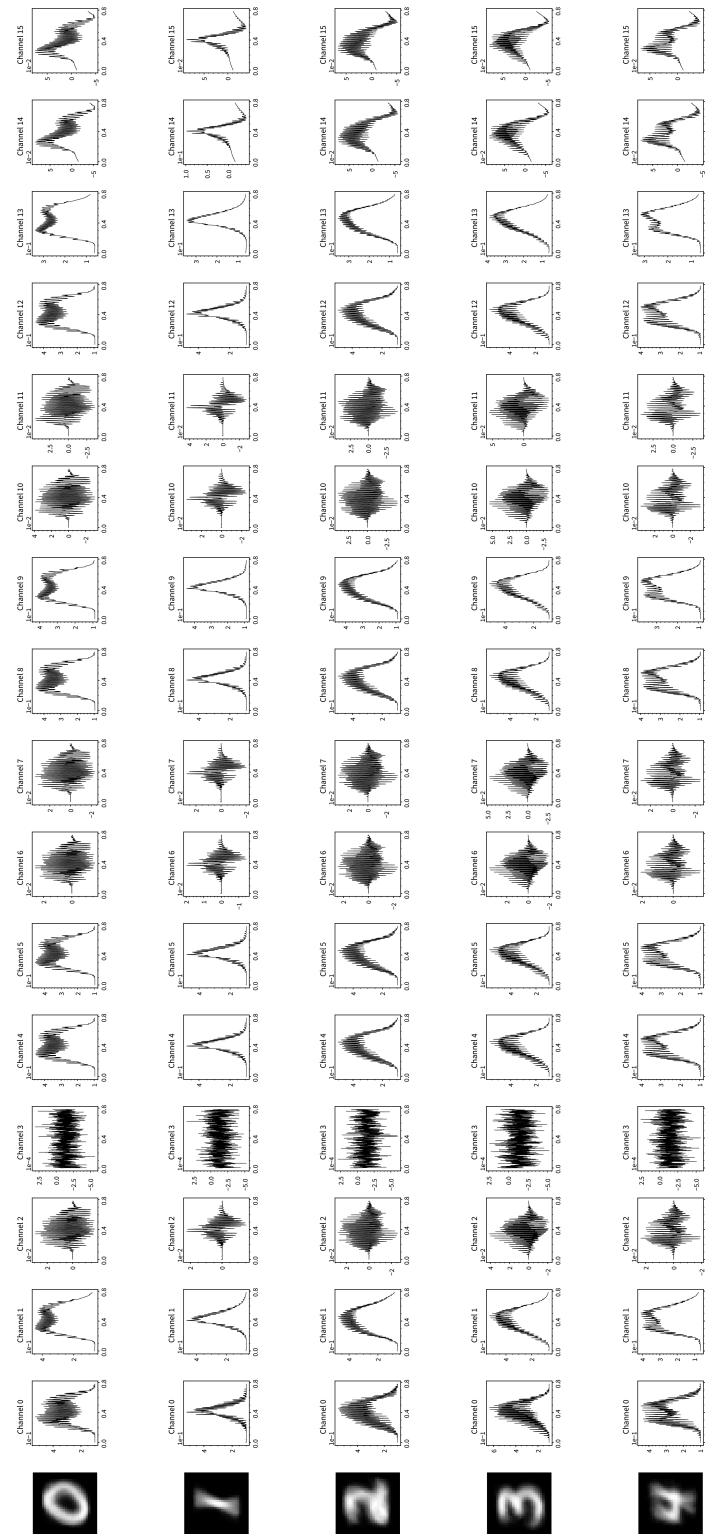
Supplementary Figure S5. SEM images of the nanowire network without electrodes. Scale bars are: (a) $10\mu\text{m}$; and (b) $5\mu\text{m}$.



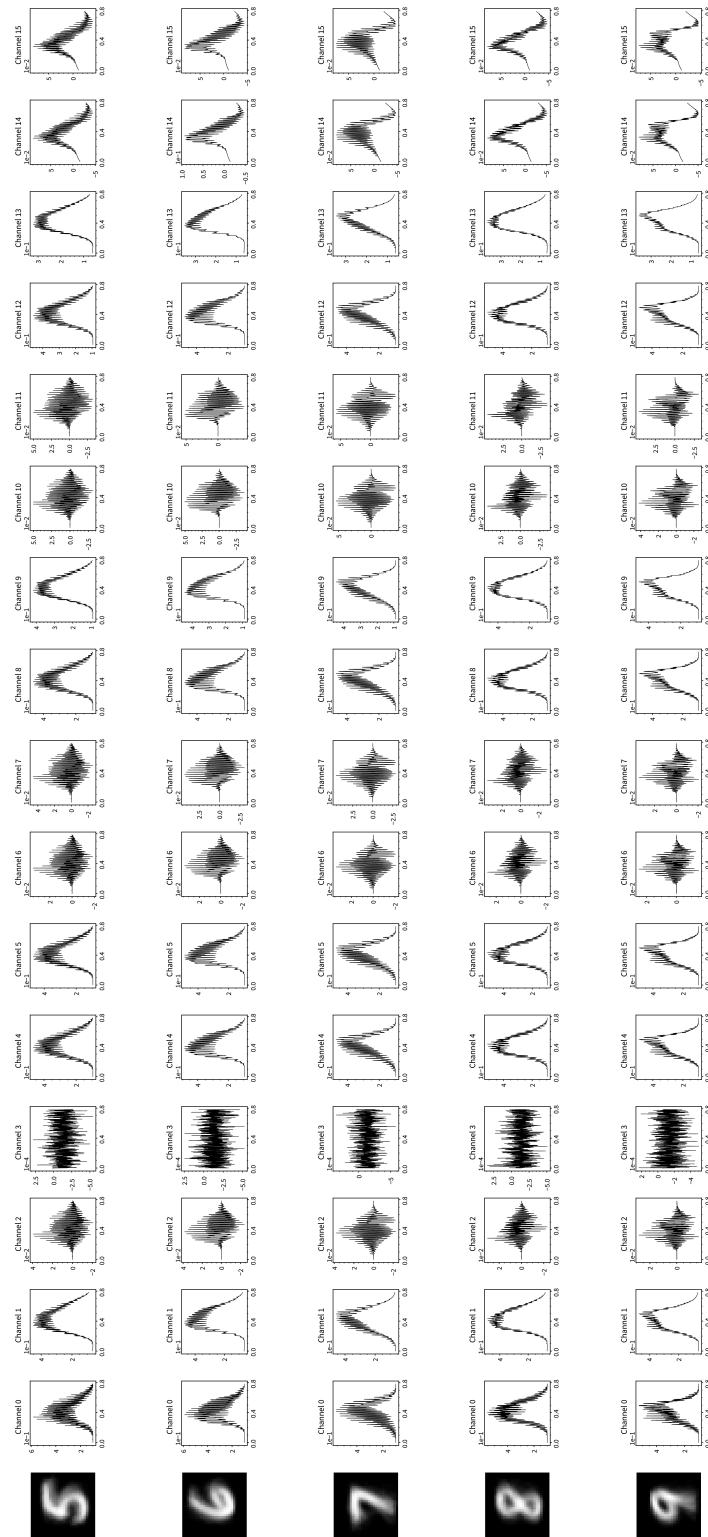
Supplementary Figure S6. Simulated nanowire network. (a) Network of 1024 nanowires with centres and lengths drawn from a uniform and Gamma distribution, respectively. (b) Corresponding graph representation, with nanowires represented by nodes and memristive cross-points between nanowires represented by edges (6877 total edges). Simulations are based on a physically-motivated model from previous studies [1–4].



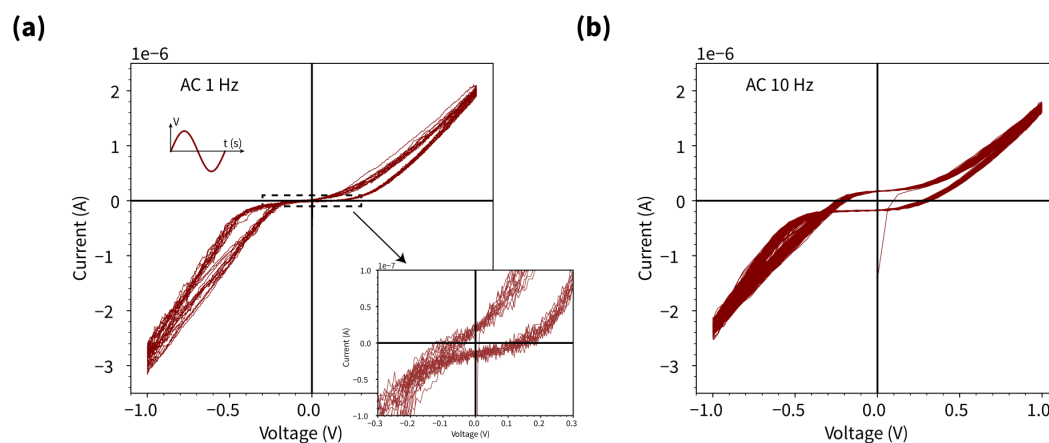
Supplementary Figure S7. Different device electrode setups. (a) Using electrode 0 as input (red) and electrode 3 as drain (green), while reading out from electrodes 1, 2, 12, 13 and 15 (blue). This is the setup used for the online MNIST classification task presented in main Results. (b) An alternate setup using electrode 9 as input and electrode 8 as drain, with readouts from electrodes 7 and 13. This is the setup used for the online image reconstruction task presented in main Results. During experiments, the voltage of every electrode and the drain current are measured continuously.



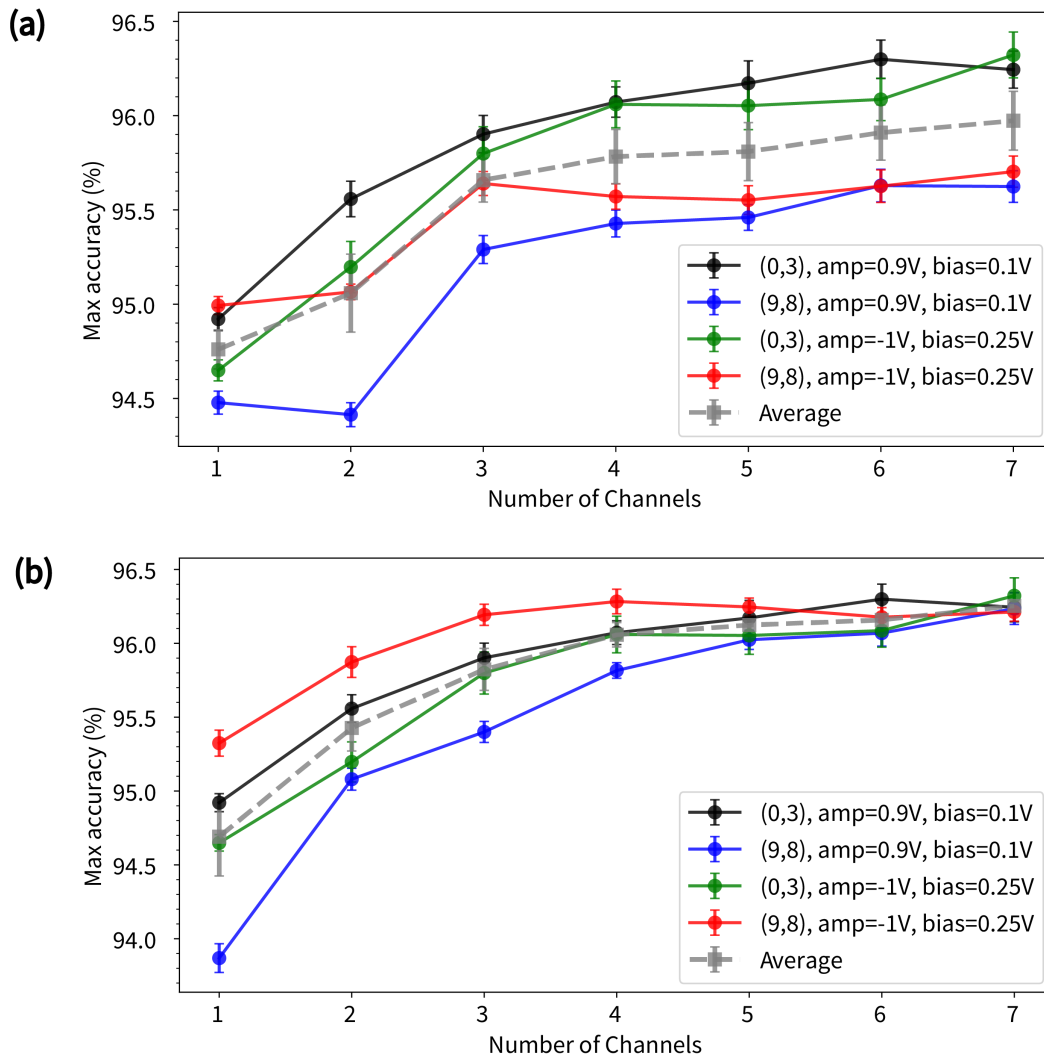
Supplementary Figure S8. Readout voltages for digits ‘0’ - ‘4’ from all 16 channels. Channel 0 and 3 are used for input and drain respectively. For each digit class, readouts correspond to 100 samples randomly selected from the training set and averaged for each channel.



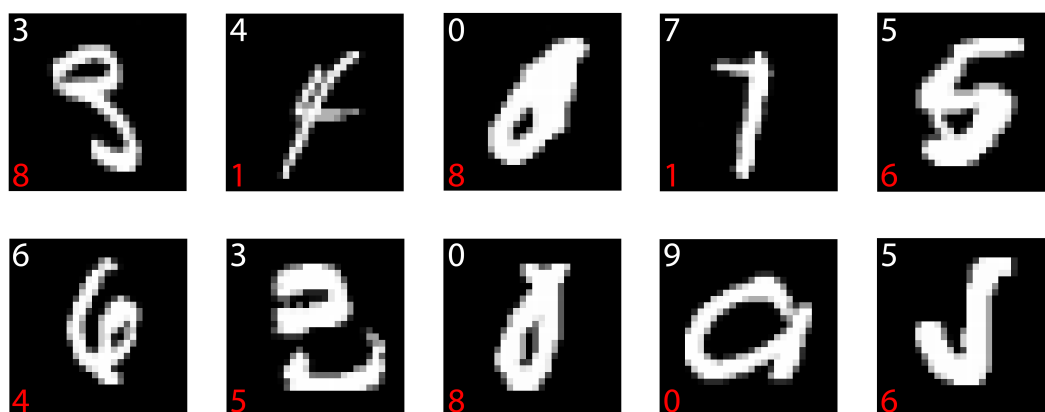
Supplementary Figure S9. Readout voltages for digits ‘5’ - ‘9’ from all 16 channels. Channel 0 and 3 are used for input and drain respectively. For each digit class, readouts correspond to 100 samples randomly selected from the training set and averaged for each channel.



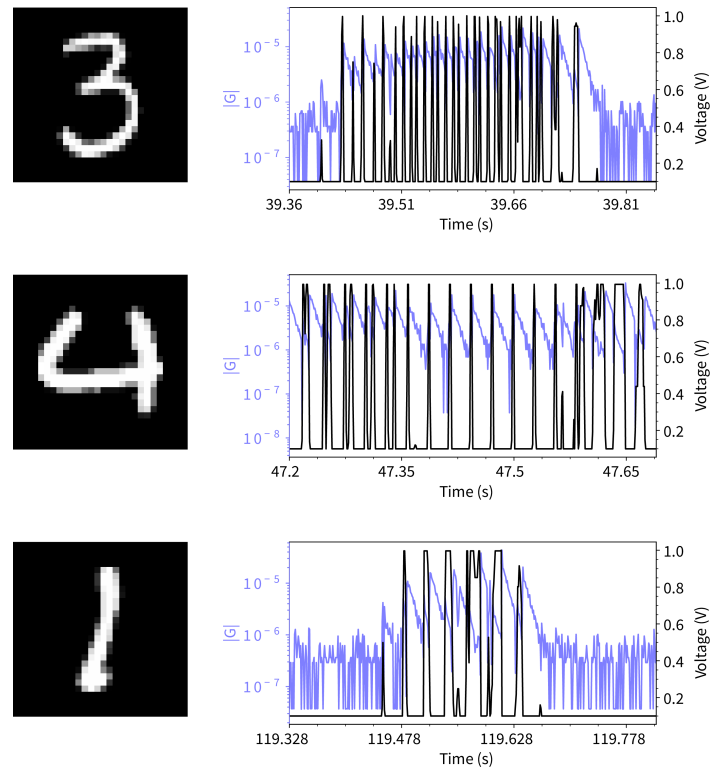
Supplementary Figure S10. $I - V$ curves under AC sweeps. (a) Response to 1 Hz sine wave. Inset is a zoom-in near the origin, showing a non-zero crossing. (b) Response to 10 Hz sine wave, showing curves driven further away from zero. See Supplementary Note 1 for further information.



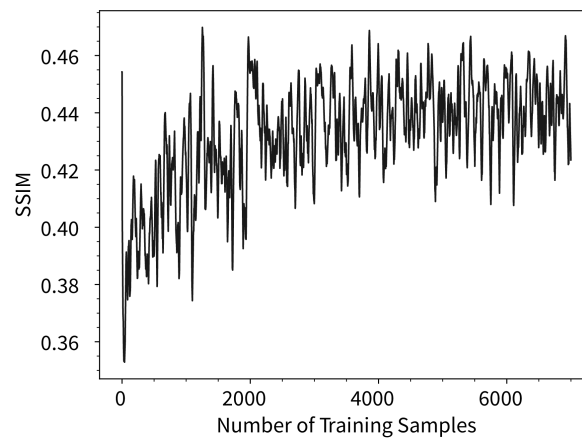
Supplementary Figure S11. MNIST online classification accuracy as a function of number of channels using different electrodes for (input,drain) and different voltage amplitude and bias. Maximum accuracy for 1,000 testing samples (after training on 50,000 samples) with respect to number of readout channels used by the online linear classifier. Error bars indicate the standard error of the mean of 5 measurements with different sequences of the training samples. (a) Source/drain electrodes (0,3), with channels 1,2,13,15,12,9,14 used for readout to the classifier; and source/drain electrodes (9,8), with channels 1,0,13,3,12,6,15 used for readout. (b) Same as (a), but with channels 12,7,14,1,0,13,3 used for (9,8). For all cases, readouts are streamed to the classifier in the specified order.



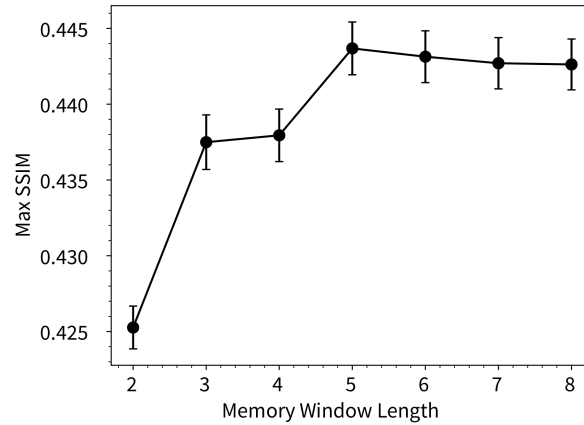
Supplementary Figure S12. Misclassified digits in the online MNIST classification task. The white digits at the top left corners are the true classes and the red digits at the bottom left are misclassified labels. Samples with unclear writings could be recognised incorrect digits of similar shape (i.e. 3 and 8, 9 and 0).



Supplementary Figure S13. Input voltage and network conductance for different MNIST digit inputs. Network conductance gradually decays after high intensity pixels are delivered, demonstrating the fading memory property.



Supplementary Figure S14. SSIM as a function of number of training samples used to train the online reconstructor. As the image reconstructor learns from more samples, SSIM increases rapidly initially then more gradually. The SSIM reported in the main results is the value after training with 7000 samples. A memory window of length $L = 4$ is shown here as an example.



Supplementary Figure S15. Maximum SSIM averaged across all digits as a function of memory window length. Max SSIM improves with memory window length, as more features are used to train the online image reconstructor. Max SSIM plateaus after the length of the memory window is greater than 4 because the reconstructor is able to use half of the semi-repetitive sequence. Errorbars indicate the standard error of the mean across all digits for reconstructions.



Supplementary Figure S16. Reconstruction of digit 5 in the sequence memory task. Reconstruction of digit 5 shows interference from digit 9, which shares the same memory features from the same sub-sequence 147. A memory window of length $L = 4$ is used here.

	Online (w/o active error correction)		Batch	
Readout data	784 x 1	784 x 5	784 x 1	784 x 5
1,000 testing	91.0%	94.1%	92.2%	93.8%
10,000 testing	85.9%	90.5%	86.6%	91.4%

Supplementary Table A.1. MNIST classification accuracies without active error correction in the online algorithm. Without error correction, accuracy of the online method is comparable to the batch method. For comparison, results are shown for 1,000 and 10,000 testing samples (for 50,000 training samples). Main results are for 10,000 testing samples. See Supplementary Figure S7 for results comparing different electrode combinations and number of channels using 1,000 testing samples.

Supplementary Note 1 – Memristive switching in Ag₂Se nanowire networks

Threshold switching in Ag₂Se nanowires was first reported by Schoen et al. [5], who found that switching from low to high conducting states is concomitant with (although not caused by) a structural phase transition. In the low conducting (cubic α) phase, Ag⁺ cations form an amorphous sublattice in which the high resistance may be attributed to localised trap states close to the conduction band edge. Under an electrical bias, a conducting Ag nano-filament forms across an individual nanowire. In Ag₂Se nanowire networks, Ag₂Se nanowires are synthesised in their orthorhombic β phase and volatile threshold switching is observed and attributed to Ag nano-filament formation at the cross-point junctions [6]. The memristive behaviour reported here, in particular the non-zero crossing in the $I - V$ curves (cf. Supplementary Figure S3), suggests that in addition to electro-chemical metallisation, other nanoscale processes may be operating at the cross-point junctions. Valov et al. [7] show that in redox-based memristive units, an electromotive force (effectively, a nanobattery) can be induced by strong spatial gradients in the distribution of Ag⁺ ions and their mobilities. This results in a non-equilibrium diffusion potential that manifests as a non-zero $I - V$ crossing. Our findings are consistent with this picture, showing that the effect becomes increasingly noticeable under non-adiabatic conditions, when the nanowire network is driven under higher frequency AC. Similar frequency-dependent behaviour has also been observed in individual TiO₂ devices and attributed to memcapacitive and meminductive properties [8].

References

- [1] Kuncic, Z. *et al.* *Emergent brain-like complexity from nanowire atomic switch networks: Towards neuromorphic synthetic intelligence*, 1–3 (2018).
- [2] Kuncic, Z. *et al.* *Neuromorphic Information Processing with Nanowire Networks*, 1–5 (2020).
- [3] Hochstetter, J. *et al.* Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nature Communications* **12** (1), 4008 (2021). doi:[10.1038/s41467-021-24260-z](https://doi.org/10.1038/s41467-021-24260-z).
- [4] Zhu, R. *et al.* Information dynamics in neuromorphic nanowire networks. *Scientific Reports* **11** (1), 13047 (2021). doi:[10.1038/s41598-021-92170-7](https://doi.org/10.1038/s41598-021-92170-7).
- [5] Schoen, D. T., Xie, C. & Cui, Y. Electrical Switching and Phase Transformation in Silver Selenide Nanowires. *Journal of the American Chemical Society* **129** (14), 4116–4117 (2007). doi:[10.1021/ja068365s](https://doi.org/10.1021/ja068365s).
- [6] Kotooka, T. *et al.* Ag₂Se Nanowire Network as an Effective In-Materio Reservoir Computing Device. Preprint (2021).
- [7] Valov, I. *et al.* Nanobatteries in redox-based resistive switches require extension of memristor theory. *Nature Communications* **4** (1), 1771 (2013). doi:[10.1038/ncomms2784](https://doi.org/10.1038/ncomms2784).
- [8] Qingjiang, L. *et al.* Memory Impedance in TiO₂ based Metal-Insulator-Metal Devices. *Scientific Reports* **4** (1), 4522 (2014). doi:[10.1038/srep04522](https://doi.org/10.1038/srep04522).