nature electronics

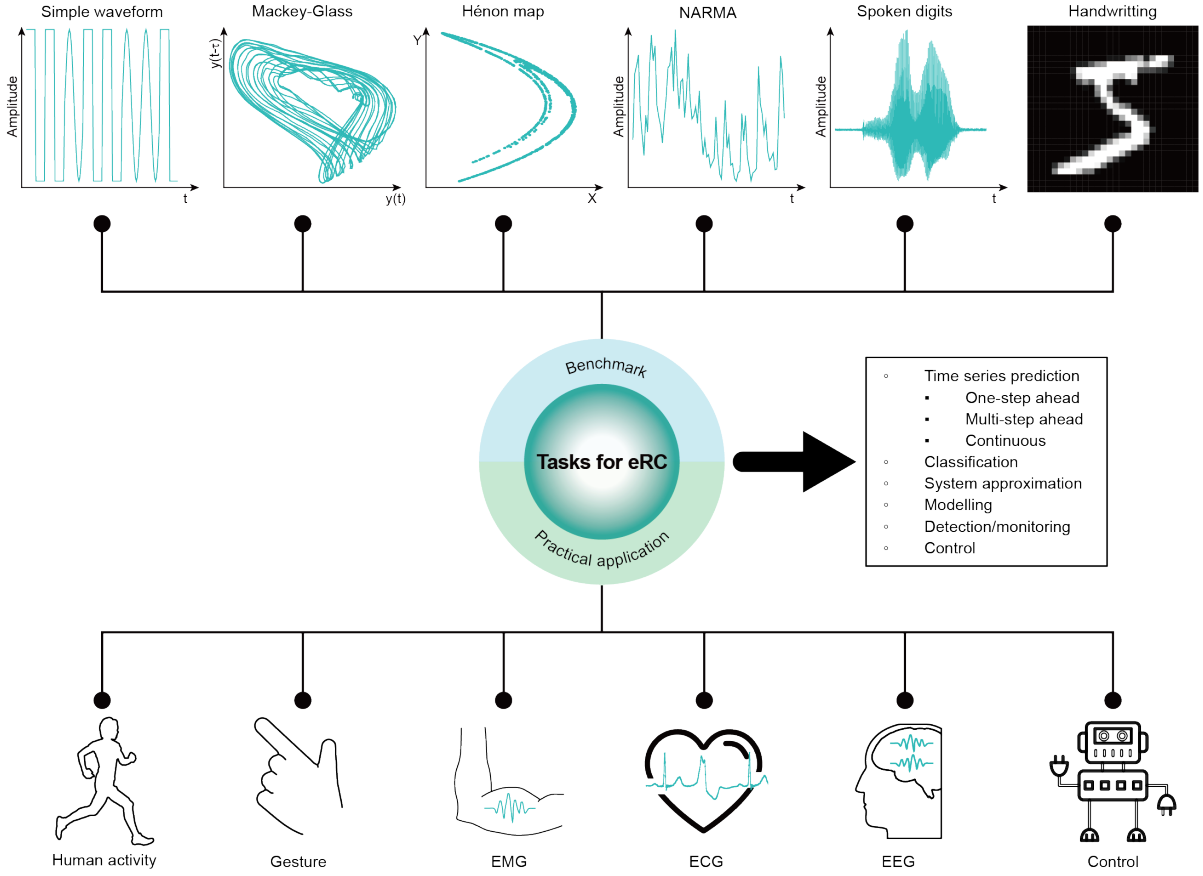
Review article

https://doi.org/10.1038/s41928-024-01133-z

Physical reservoir computing with emerging electronics

In the format provided by the authors and unedited

Supplementary Figure



Supplementary Figure 1. Representative tasks implemented by eRC.

Supplementary Note 1: eRC simulation model for optimisation

It is recommended to develop a simulation model for optimisation before hardware implementations as prior representative eRC works did[1-11]. The reasons are:

1) Modelling can help the developer to fully capture the mechanism of how their devices behave under an eRC architecture.

2) Modelling before the hardware experiment can find a properly working condition in the first place, while tunning parameters purely in the experiment may be extremely time-consuming.

3) A simulation model can estimate the required reservoir size for a given machine learning task.

4) Simulation model can also estimate the potentials of the proposed method (e.g., reservoir size can be arbitrarily expanded to explore the performance limits).

5) Simulation models can also be published online[6-11] in case the followers want to access the details.

Given the advantages of doing simulation for eRC optimisation, however, one shall keep the following hardware-software differences in mind to set up a more practical and reasonable optimisation. Firstly, the value scanning on hardware could be affected by the cycle-to-cycle variation and device reliability degradation over time, while the pure software simulations may not be able to accurately capture the complex dynamics of electronic devices. Secondly, the constraints in hardware could invalidate the parameter scanning results for maximizing the performance. For instance, the optimal input range from 3V to 5V could result in ~10× more power consumption and degraded device endurance than that in range of 0.3V to 0.5V. Thirdly, the analogue signals in eRC suffer from inevitable noise that may affect the system performance optimisation.

Supplementary Note 2: Supplementary discussion for memory capacity

Memory capacity (MC) indicates the reservoir's capability to retain the historical information within the state vector, which is an important task-independent network property. In this review article, the main text introduces the main idea of testing MC in the **Performance benchmark** section, and the mechanism of MC in each architecture is also briefly introduced in the **Architectures of eRC** section. We also summarized the recently reported MC evaluations in **Supplementary Table 2**. To facilitate future studies that analyze the MC in their reservoir systems, the following subtle but critical aspects deserve further attentions:

1) The use of intermediate and virtual nodes becomes a critical issue. As mentioned in the main text, the intermediate and virtual nodes require serial operation and additional memory to store the information, which however is against the intention of MC measurement — testing the reservoir layer's short-term memory capacity in preserving historical information in a fading manner. More specifically, since MC is quantified by a sequence recalling task, the sequence could be permanently stored to yield a high MC if non-volatile memory exists in the system. In delay-coupled RC, the time-multiplexing is an essential operation so that the cost of virtual nodes has to be taken into account in the system development. For other architectures, the virtual node should be avoided when testing MC in order to allow the measured MC value to reflect the real intrinsic short-term memory generated by the dynamics of the reservoir layer.
2) Given the same architecture and physical node, the MC value is also significantly affected by the reservoir configuration. For example, the input range determines to what extent the nonlinearity of the physical nodes distorts the states. MC would be much higher if the physical nodes worked in the linear range. Therefore, it is suggested that future studies should also discuss the input and nonlinear ranges together with their MC results.
3) It is found in **Supplementary Table 2** that the MC values calculated by the state vector collected in experiments are usually lower than that in simulations. Unlike simulation, the hardware experiments usually suffer from nonideal factors such as the ambient noise, quantization noise, etc., which lower the MC. However, hardware measurement is still preferred in testing MC since it is closer to the real properties claimed by a PRC.

In addition, it is recommended to refer to the original paper that defines the short-term memory measurement for RC[12]. To correlate the MC value with actual RC performance, it is recommended to refer to the thesis[13] that investigated the effect of MC and other task-independent properties concerning NARMA10 results.

Supplementary Table 1. Terminology definition

Terminology	Definition
Memory capacity (MC)	A quantification of RC's capacity in retaining historical information in the reservoir state
State richness	The number of linearly uncoupled dynamics of reservoir channels
Time-multiplexing	A signal pre-processing technique to increase the number of reservoir states
Physical node	The interconnected nonlinear components in the reservoir layer
Virtual node state	The node states generated by time-multiplexing operation
Information processing capacity (IPC)	A quantification of RC's capacity in both memory and nonlinear fitting
Kernel Quality (KQ)	A quantification of RC's capacity in high-dimensional kernel mapping
Generalization Rank (GR)	A quantification of RC's capacity in forgetting information from a given period of time

Supplementary Table 2. Comparison of existing eRCs' MC

Ref	Architecture	Processing core	# of total states	MC	Virtual node
Appeltant 2012	Delay-coupled	Nonlinearity circuit	400	75	Yes
Nakane et al. 2023	In materia	Spin-wave device	360	38	No
Liang et al. 2022	Rotating neurons	ReLU circuit	100	20	No
Soriano et al. 2015	Delay-coupled	Nonlinearity circuit	400	18	Yes
Zheng et al. 2021	Delay-coupled	MEMS	\	13.34	Yes
Zhu et al. 2021	In materia	Nanowire networks	100	11.9	No
Vidamour et al. 2022	Rotating neurons + in materia	Magnetic metamaterial	\	11.5	Yes
Furtuta et al. 2018	Dynamic devices	Magnetic Tunnel Junctions	50	7.8	Yes
Daniels et al. 2022	In materia	3D nanowire networks	24	5	No
Sun et al. 2022	Dynamic devices	3D memristor array	\	3.5	Yes
Tsunegi et al. 2019	Dynamic devices	Magnetic tunnel junction	250	3.2	Yes
Tang et al. 2022	Dynamic devices	Fe-FET	\	3.1	Yes
Liu et al. 2022	In materia	Ferroelectric □-In2Se3 devices	20	2.7	Yes
Toprasertpong et al. 2022	In materia	Fe-FET	600	2.35	Yes
Nako et al. 2020	In materia	Fe-FET	100	2.2	Yes
Sakai et al. 2021	Dynamic devices	Au nanogap	10	1.07	Yes

Supplementary Table 3. Comparison of representative eRC implementations.

Architecture	Refs.	Physical node	# of physical nodes	# of virtual or intermediate nodes	# of total states	Pre-processing	Output layer	Task	Result
Delay-coupled RC	Appeltant et al., 2011	Nonlinear circuit (Mackey Glass type)	1	400	400	Time-multiplexing	400 × 1	NARMA10*	NRMSE=0.15
						Cochlear ear model; Time-multiplexing	400 × 10	Spoken digit recognition	WER=0.2%
	Torrejon et al., 2017	Spintronic oscillators	1	400	400	Cochlear ear model	400 × 10	Spoken digit recognition	Recognition rates=99.6%
						Time-multiplexing	400 × 1	Waveform classification	RMSE = 1%
	Brunner et al., 2013	Semiconductor laser diode	1	388	388	Cochlear ear model; Time-multiplexing	388 × 10	Spoken digit recognition	classification error = 0.014%
						Time-multiplexing	388 × 1	Santa Fe prediction	prediction error = 10.6% at a prediction rate of 1.3×10^7 points per second
	Zhu et al., 2020	$CsPbI_3$-based memristor	1	30	31	Spike encoding	CNN + 5 × 5 perceptron	Neural activity analysis	Accuracy = 87.0%
	Barazani et al., 2020	MEMS accelerometer	1	100	100	Raw signal	100 × 1	From NARMA2 to NARMA 20	NRMSE = 0.2 (NARMA2) to 0.8 (NARMA20)
Dynamic devices RC	Sun et al., 2021	SnS-based memristor	14	5	71	Spike encoding	71 × 5	Korean sentences recognition†	Accuracy = 91%
	Moon et al., 2019	WO_x memristors	50	8	400	Lyon's passive ear model; Spike encoding	400 × 10	Spoken digit recognition	Recognition rates=99.2%

Architecture	Refs.	Physical node	# of physical nodes	# of virtual or intermediate nodes	# of total states	Pre-processing	Output layer	Task	Result
Dynamic devices RC			20	50	1000	Raw signal	1000 × 1	Mackey–Glass time series prediction	Reliable forecasting can be obtained, aided by periodic updates
	Du et al., 2017	WO_x memristors	5	1	5	Spike encoding	5 × 10	Digit recognition†	Successful
			88	2	176	Spike encoding	176 × 10	Handwritten digit classification	Accuracy = 91.1%
			90	1	90	Raw signal	90 × 1	Second-order nonlinear dynamic	NMSE = 3.61×10^{-3}
	Wang et al., 2021	WO_3 memristors	16	10	161	Spike encoding	161×51×4‡	Artificial olfactory inference†	Accuracy = 95%
	Zhong et al., 2021	TiO_x/TaO_y-based dynamic memristor	40	10	400	Lyon's passive ear model; Time-multiplexing	400 × 10	Spoken-digit recognition	Recognition rate = 99.6%
			25	4	100	Time-multiplexing	100 × 1	Hénon map time-series prediction	NRMSE = 0.046
			10	4	40	Time-multiplexing	40 × 1	Waveform classification	NRMSE = 0.14
	Sun et al., 2021	Micromechanical resonator	1	440	440	Image chopping/merging	440 × 10	Handwritten digit classification	Accuracy = 93%
			1	N/A	N/A	Raw signal	N/A	NARMA1	MNSE = 0.051
			1	N/A	N/A	Filtering; Image chopping	N/A	Gesture classification (6-axis IMU) †	Accuracy = 97.17±1%
	Yu et al., 2021	Ferroelectric tunneling junctions	28	7	196	Image chopping; Spike encoding	196 × 64 × 256 × 10‡	Handwritten digit classification	Accuracy = 92.3%
	Zhong et al., 2022	TiO_x-based memristor	24	5	120	Time-multiplexing	120 × 1	Arrhythmia detection	Accuracy = 96.6%

Architecture	Refs.	Physical node	# of physical nodes	# of virtual or intermediate nodes	# of total states	Pre-processing	Output layer	Task	Result
Dynamic devices RC			24	5	120	Time-multiplexing	121 × 1	Dynamic gesture recognition†	Accuracy = 97.9%
	Yang et al., 2023	IGZO artificial synapse	784	50	39200	Time-multiplexing; Image chopping/merging	39200 × 10	Handwritten digit classification*	Accuracy = 90.86%
			64	20	1280	Time-multiplexing; Frequencies filtering	1280 × 10	Spoken-digit classification*	Accuracy = 100%
	Jaafar et al., 2022	Mesoporous silica memristors	5	5	25	Spike encoding	25 × 1	5 × 5 Letter recognition	Accuracy = 100%
	Nishioka et al., 2022	Li^+ electrolyte/ diamond interface	8	10	80	Spike encoding	80 × 1	Second-order nonlinear system	Predicted error = 2.08 × 10-4
			8	10	80	Spike encoding	80 × 1	NARMA2	NMSE = 0.02
			196	1	196	Spike encoding	196 × 10	Handwritten digit classification	Accuracy = 88.8%
	Liang et al., 2022	electrolyte-gated transistors	88	1	88	Spike encoding	88 × 10	Handwritten digit classification*	Accuracy = 87.1%
	Sun et al., 2022	3D memristor array	N/A	N/A	N/A	Spike encoding (2 bit + 1 bit)	N/A	Dynamic Gesture Recognition†	Accuracy = 90%
	Zhang et al., 2022	GaO_x photo-synapses	40	1	40	Image chopping/reshaping; Spike encoding	40 × 5‡	Latent fingerprint recognition	Accuracy = 100%
	Liu et al., 2022	α-In_2Se_3 based optoelectronic synapse	100	1	100	Spike encoding; Image chopping/merging	100 × 10	Handwritten digit classification†	Accuracy = 86.1%
			200	1	200	Image chopping/rotating; Spike encoding	200 × 10	QR code recognition†	Accuracy = 98.6%

Architecture	Refs.	Physical node	# of physical nodes	# of virtual or intermediate nodes	# of total states	Pre-processing	Output layer	Task	Result
			42	4	168	Raw signal	168 × 1	MSO time-series data	NRMSE = 0.105
In materia RC	Milano et al., 2021	Ag nanowire networks	1	1	4	Spike encoding	4 × 4‡	4 × 4 pattern recognition	Accuracy = 90%
			1	1	196	Image chopping/merging, Spike encoding	196 × 10	Handwritten digit classification	Accuracy = 90.4%
			1	20	N/A	Time-multiplexing	N/A	Mackey–Glass time series prediction	N\A
	Cucchi et al., 2021	Organic electrochemical networks	1	N/A	N/A	Time-multiplexing	N/A	MIT-BIH dataset	Accuracy = 88%
			1	N/A	N/A	Time-multiplexing, Frequency encoding	N/A	Flower classification	Accuracy = 96%
	Usami et al., 2021	Sulfonated Polyaniline Network	1	1	15	Raw signal	15 × 1	Waveform generation	Accuracy = 88%-99%
			1	1	12	Lyon's auditory model filtering	12 × 10	Spoken-digit classification	Accuracy = 60%
	Hochstetter et al., 2021	Ag-PVP-Ag nanowire networks	1	1	261	Raw signal	261 × 1	Waveform generation	Accuracy = 98% for sine, 81% for square, 56% for phase shifted and 75% for double frequency
	Toprasertpong et al., 2022	HfO_2-based FeFET	1	600	600	Time-multiplexing	600 × 1	NARMA2	NMSE = 7.3 × 10−4

Architecture	Refs.	Physical node	# of physical nodes	# of virtual or intermediate nodes	# of total states	Pre-processing	Output layer	Task	Result
In materia RC	Liu et al., 2022	ferroelectric α-In_2Se_3 devices	1	40	40	Time-multiplexing	40 × 1	Prediction of Brazilian electrical energy consumption	Mean absolute percentage error of 4.67%
			1	80	80	Time-multiplexing	80 × 1	Waveform Classification	NRMSE = 0.2
Rotating neurons RC	Liang et al., 2022	Integration-ReLU-Leakage circuit	400	1	400	Raw signal	400 × 1	NARMA10*	NRMSE=0.078
			64	1	64	Raw signal	64 × 1	Mackey–Glass time series prediction	NRMSE=0.03
			64	1	64	Raw signal	64 × 1‡	Handwritten vowel recognition†	Accuracy = 94.0±0.8%

*Simulation only

‡Memristor-based output layer

†Custom task

# of physical nodes: number of physical nodes in parallel being used in the reservoir layer.

# of virtual or intermediate nodes: a single physical node subjected to serial operations can generate more than one state value, that is, the virtual nodes generated by the time-multiplexing, and intermediate nodes as a result of the storage of historical node states. Both virtual and intermediate nodes require serial operation and external memory.

# of total states: number of total states being sent to the output layer. In most cases, it is equal to the number of physical nodes times the number of virtual or intermediate nodes. The exception could be found in the systems that use other sources, e.g. input signal, as extra state channels.

Output layer: the design of the output layer presented in the form of $a \times b$, where a denotes the number of the state vector, and b denotes the number of the final output. For classification tasks, b should be the number of classes in the result, while b is one for the prediction or time-series processing tasks.

Reference

1 Appeltant, L. *et al.* Information processing using a single dynamical node as complex system. *Nat. Commun.* **2**, 468, (2011).

2 Nakajima, M. *et al.* Physical deep learning with biologically inspired training method: gradient-free approach for physical hardware. *Nat. Commun.* **13**, 7847, (2022).

3 Moon, J. *et al.* Temporal data classification and forecasting using a memristor-based reservoir computing system. *Nat. Electron.* **2**, 480-487, (2019).

4 Kulkarni, M. S. & Teuscher, C. in *Proceedings of the 2012 IEEE/ACM International Symposium on Nanoscale Architectures* 226–232 (Association for Computing Machinery, Amsterdam, The Netherlands, 2012).

5 Daniels, R. K. *et al.* Reservoir computing with 3D nanowire networks. *Neural Netw* **154**, 122-130, (2022).

6 Liang, X. *et al.* Rotating neurons for all-analog implementation of cyclic reservoir computing. *Nat. Commun.* **13**, 1549, (2022).

7 Zhong, Y. N. *et al.* A memristor-based analogue reservoir computing system for real-time and power-efficient signal processing. *Nat. Electron.* **5**, 672-681, (2022).

8 Zhong, Y. *et al.* Dynamic memristor-based reservoir computing for high-efficiency temporal signal processing. *Nat. Commun.* **12**, 408, (2021).

9 Milano, G., Montano, K. & Ricciardi, C. In materia implementation strategies of physical reservoir computing with memristive nanonetworks. *Journal of Physics D-Applied Physics* **56**, 084005, (2023).

10 Milano, G. *et al.* In materia reservoir computing with a fully memristive architecture based on self-organizing nanowire networks. *Nat. Mater.* **21**, 195-202, (2022).

11 Hochstetter, J. *et al.* Avalanches and edge-of-chaos learning in neuromorphic nanowire networks. *Nat. Commun.* **12**, 4008, (2021).

12 Jaeger, H. *Short term memory in echo state networks*. (GMD Forschungszentrum Informationstechnik, 2002).

13 Appeltant, L. *Reservoir computing based on delay-dynamical systems* Doctoral thesis, (2012).