





RESEARCH ARTICLE | SEPTEMBER 12 2025

Echo state and band-pass networks with aqueous memristors: Leaky reservoir computing with a leaky substrate

Special Collection: [Nonlinear Dynamics of Reservoir Computing: Theory, Realization and Application](#)

T. M. Kamsma  ; J. J. Teijema  ; R. van Roij  ; C. Spitoni 



Chaos 35, 093133 (2025)

<https://doi.org/10.1063/5.0273574>



Articles You May Be Interested In

Iontronics: Aqueous ion-based engineering for bioinspired functionalities and applications

Chem. Phys. Rev. (September 2022)

III-V semiconductor nanostructures and iontronics: InAs nanowire-based electric double layer field effect transistors

AIP Conf. Proc. (August 2019)

MXene-TiO₂ heterostructured iontronic neural devices based on ion-dynamic capacitance enabling optoelectronic modulation

Appl. Phys. Rev. (November 2024)



Chaos

Special Topics Open for Submissions

[Learn More](#)

Echo state and band-pass networks with aqueous memristors: Leaky reservoir computing with a leaky substrate

Cite as: Chaos 35, 093133 (2025); doi: 10.1063/5.0273574

Submitted: 31 March 2025 · Accepted: 27 August 2025 ·

Published Online: 12 September 2025



View Online



Export Citation



CrossMark

T. M. Kamsma,^{1,2,a} J. J. Teijema,³ R. van Roij,² and C. Spitoni¹

AFFILIATIONS

¹Mathematical Institute, Utrecht University, Budapestlaan 6, Utrecht 3584 CD, The Netherlands

²Institute for Theoretical Physics, Utrecht University, Princetonplein 5, Utrecht 3584 CC, The Netherlands

³Department of Methodology and Statistics, Utrecht University, Padualaan 14, Utrecht 3584 CH, The Netherlands

Note: This paper is part of the Special Topic on Nonlinear Dynamics of Reservoir Computing: Theory, Application and Realization.

^a**Author to whom correspondence should be addressed:** t.m.kamsma@uu.nl

ABSTRACT

Recurrent Neural Networks (RNNs) are extensively employed for processing sequential data such as time series. Reservoir computing (RC) has drawn attention as an RNN framework due to its fixed network that does not require training, making it an attractive platform for hardware-based machine learning. We establish an explicit correspondence between the well-established mathematical RC implementations of echo state networks and band-pass networks with leaky integrator nodes on the one hand and a physical circuit containing iontronic simple volatile memristors on the other. These aqueous iontronic devices employ ion transport through water as signal carriers and feature a voltage-dependent (memory) conductance. The activation function and the dynamics of the leaky integrator nodes naturally materialize as the (dynamic) conductance properties of iontronic memristors, while a simple fixed local current-to-voltage update rule at the memristor terminals facilitates the relevant matrix coupling between nodes. We process various time series, including pressure data from simulated airways during breathing that can be directly fed into the network due to the intrinsic responsiveness of iontronic devices to applied pressures. We accomplish this by employing established physical equations of motion of iontronic memristors for the internal dynamics of the circuit.

© 2025 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0273574>

Reservoir computing (RC) is a proven method for processing temporal data and has drawn more recent attention as a suitable framework for hardware-based machine learning. Echo state and band-pass networks are extensively studied implementations of RC. We propose a novel physical circuit design, based on fluidic iontronic memristors, that provides a one-to-one correspondence with the mathematical descriptions of these RC paradigms. Using the underlying equations of motion of these fluidic devices, we process several time series, including simulated respiratory pressure waveforms, exploiting iontronics' intrinsic sensitivity to applied pressures. Our direct physical (iontronic) realization of these established RC implementations offers a blueprint for physically embedded temporal processing with an emerging substrate.

I. INTRODUCTION

Reservoir computing (RC) has gained significant attention as a recurrent neural network paradigm for processing temporal data.¹ RC employs a fixed high-dimensional reservoir (i.e., a dynamical system with many internal states) whose dynamics are driven by input signals, with the benefit that only a simple readout function requires training for classification tasks. Although the rise of computational capacity for training in the past few years has somewhat mitigated this benefit, new attention has recently been drawn to RC for hardware-based implementations as the fixed nature of the reservoir circumvents complicated internal tuning of the RC circuit.² Recent research, for instance, has explored the use of physical systems such as electronic, electrochemical, optical, and mechanical devices.^{2,3} However, although the use of physical substrates is

informed by the established mathematical frameworks for software RC, establishing a deeper physical equivalence is challenging.

In this work, we establish an explicit one-to-one correspondence between the physical equations of iontronic memristors placed within a peripheral circuit on the one hand and the governing equations of echo state networks (ESNs) and band-pass networks (BPNs) with leaky integrator nodes (LI-ESNs and LI-BPNs, respectively) on the other. Iontronics exploit aqueous ionic and molecular transport, akin to the brain's medium, and can, therefore, provide striking similarities with the brain in neuromorphic computing implementations,^{4–11} including RC.^{12,13} Additionally, the easily tunable memory timescales of iontronic platforms^{12,14–17} naturally match the relatively slow timescales found in natural or biological signals, something that is challenging within conventional fast solid-state devices.¹⁸ Moreover, we show how the pressure-dependence of these fluidic systems enables direct conversion of a biological pressure signal to circuit input without any intervention or interaction required from outside the network.

In this work, we (i) propose a circuit based on the emerging “leaky” substrate of aqueous iontronics that would be capable of advanced RC applications, (ii) propose a one-to-one correspondence between this physical (iontronic) circuit and the well-established mathematical LI-ESN and LI-BPN descriptions, (iii) leverage the unique property of slow easily tunable memory timescales of iontronic memristors, and (iv) exploit the intrinsic pressure responsiveness of iontronics to directly convert pressure signal inputs on-chip. Due to the equivalence between our proposed physical device and the abstract mathematical framework, the (extensive) theoretical results previously derived for ESNs and BPNs^{19–27} can be directly translated to the proposed physical circuit, without requiring us to reinvent the wheel for RC on a physical substrate. All code used for our results is available online at Ref. 28. Our results not only advance the theoretical understanding of RC in physical systems but also provide a pathway for the development of new (iontronic) hardware-based RC implementations.

II. ECHO STATE AND BAND-PASS NETWORKS

We consider LI-ESNs that converts a K -dimensional input $\mathbf{u}(t)$ with N reservoir neurons at state $\mathbf{x}(t)$ to an L -dimensional output $\mathbf{y}(t)$ at time t , here without output feedback and without direct input-to-output coupling. Such an LI-ESN is governed by¹⁹

$$\dot{\mathbf{x}} = \frac{1}{c} (-a\mathbf{x} + f(\mathbf{W}^{\text{in}}\mathbf{u} + \mathbf{W}\mathbf{x})), \quad (2.1)$$

$$\mathbf{y} = \mathbf{W}^{\text{out}}\mathbf{x}. \quad (2.2)$$

Here, $c \in \mathbb{R}$ is a global relaxation time parameter, $a \in \mathbb{R}$ is the leaking rate, $\mathbf{u} = \mathbf{u}(t) \in \mathbb{R}^K$ is the input, $\mathbf{x} = \mathbf{x}(t) \in \mathbb{R}^N$ is the state of the reservoir neurons, $f: \mathbb{R} \rightarrow \mathbb{R}$ is a sigmoidal activation function that is applied element-wise to its input, and $\mathbf{y} = \mathbf{y}(t) \in \mathbb{R}^L$ is the output vector. Here, we will set $f(x) = \tanh(x)$, a standard choice,¹⁹ and g will be the identity function such that $\mathbf{y} = \mathbf{W}^{\text{out}}\mathbf{x}$. The input and reservoir states are coupled through the input matrix $\mathbf{W}^{\text{in}} \in \mathbb{R}^{N \times K}$, the internal matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$, and the output matrix $\mathbf{W}^{\text{out}} \in \mathbb{R}^{L \times N}$. Notably, only \mathbf{W}^{out} needs to be found through training, which we

here perform via ridge regression,²⁹ with the other matrices being initialized randomly.

In its most general form, there is also an output-feedback term $\mathbf{W}^{\text{fb}}\mathbf{y}$ in the argument of f , which is not of relevance for the input processing we focus on here.¹⁹ Additionally, the output can be transformed by a function $g: \mathbb{R} \rightarrow \mathbb{R}$ that is applied element-wise to $\mathbf{W}^{\text{out}}\mathbf{x}$ such that $\mathbf{y} = g(\mathbf{W}^{\text{out}}\mathbf{x})$. Lastly, in general, one can also directly couple the input to the output according to $\mathbf{y} = g(\mathbf{W}^{\text{out}}[\mathbf{x}; \mathbf{u}])$ with $[\cdot; \cdot]$ denoting vector concatenation (in this case, $\mathbf{W}^{\text{out}} \in \mathbb{R}^{L \times (K+N)}$). These last two generalizations are omitted here to simplify the physical circuit realization.

The physical circuit we will introduce in Sec. III A is in principle described by continuous equations as Eq. (2.1). However, inputs \mathbf{u} are often discrete-time sampled. Additionally, here we simulate the circuit using physical equations of the internally used iontronic memristors, but this will also require discretization of the underlying equations. Using Euler discretization with (time) stepsize δ and $t = n\delta$ with $n \in \mathbb{N}$, we see that Eq. (2.1) for $n \geq 0$ becomes¹⁹

$$\mathbf{x}(n+1) = \left(1 - \frac{a\delta}{c}\right)\mathbf{x}(n) + \frac{\delta}{c}f(\mathbf{W}^{\text{in}}\mathbf{u}(n\delta) + \mathbf{W}\mathbf{x}(n)).$$

A. Echo state property

A key stability property of ESNs is the *echo state property* (ESP), which is defined by¹⁹

Definition 1. An ESN with reservoir states $\mathbf{x}(n)$ has the echo state property if for any compact $C \subset \mathbb{R}^K$ and any two starting states $\mathbf{x}(0)$ and $\mathbf{x}'(0)$, there exists a sequence $(\delta_h)_{h=0,1,2,\dots}$ that converges to 0 such that for any input sequence $(\mathbf{u}(n))_{n=0,1,2,\dots} \subseteq C$ it holds that $\|\mathbf{x}(h) - \mathbf{x}'(h)\| \leq \delta_h$.

Heuristically, Definition 1 tells us that an ESN with the ESP “forgets” its initial state at a rate independent from the input sequence or the precise initial state.

There are various constraints that guarantee the echo state property in leaky integrator ESNs.^{19,22} The condition we will use here is that the spectral radius $\rho(\mathbf{M})$ of $\mathbf{M} = (\delta/c)|\mathbf{W}| + (1 - a\delta/c)\mathbf{I}$ satisfies $\rho(\mathbf{M}) < 1$. A simple algorithm for constructing an internal weight matrix that guarantees the echo state property is given by²²

1. Generate a random matrix \mathbf{W} with only non-negative elements $w_{ij} \geq 0$.
2. Rescale \mathbf{W} such that the spectral radius $\rho(\mathbf{M})$ of the matrix $\mathbf{M} = (\delta/c)\mathbf{W} + (1 - a\delta/c)\mathbf{I}$ satisfies $\rho(\mathbf{M}) < 1$.
3. Change the sign of a desired number of elements w_{ij} .

Internal weight matrices \mathbf{W} with spectral radii smaller than 1 often also display the ESP, but this is not a guarantee.²² Alternatively, one could generate a random \mathbf{W} and then check if $\rho(\mathbf{M}) < 1$ afterward. The available code²⁸ provides aforementioned algorithm as an optional setting to guarantee the ESP. Lastly, $a_c^{\frac{\delta}{c}} \leq 1$ is a natural constraint.

B. Band-pass network

A LI-BPN is similar to a LI-ESN as in Eqs. (2.1) and (2.2), with one key difference. Each node can individually be designed to be sensitive to certain frequencies, which we implement here by providing each node with its own characteristic relaxation timescale c_i ,^{27,30}

as opposed to a single global timescale c for the entire network. So, the parameter $c \in \mathbb{R}$ is replaced by a vector $\mathbf{c} \in \mathbb{R}^N$ and Eq. (2.1) naturally becomes

$$\dot{\mathbf{x}} = (-\mathbf{a}\mathbf{x} + f(\mathbf{W}^{\text{in}}\mathbf{u} + \mathbf{W}\mathbf{x})) \oslash \mathbf{c}, \quad (2.3)$$

with \oslash being element-wise (Hadamard) division.

III. PHYSICAL LI-ESN WITH IONTRONIC MEMRISTORS

Consider the circuit schematically drawn in Fig. 1(a) containing (iontronic) memristors (blue), which we will describe in more detail in Sec. III A 1. The voltages at the terminals (dashed ellipses) obey a fixed local current-to-voltage update rule, which we will describe in detail in Sec. III A 2. We will show in Sec. III B that the physical circuit design in Fig. 1(a) is equivalent to the general mathematical LI-ESN description as in Eq. (2.1). Lastly, in Sec. III C, we will extend this equivalence to LI-BPNs as in Eq. (2.3).

A. Physical ESN circuit

1. (Iontronic) simple volatile memristors

Memristors, characterized by their history-dependent conductance, have drawn major interest as fundamental devices for neuromorphic computing architectures.³² Consequently, many different types of memristors with various conductance memory features exist.^{33–35} Inspired by the brain's aqueous medium and ionic signal carriers, iontronics that rely on ionic transport in an aqueous environment are emerging as a substrate for neuromorphic computing implementations.^{4–8} Of importance to this work is that various iontronic devices also feature a coupling between their electric properties and applied pressures,^{36–38} where, e.g., an applied pressure can drive a so-called electric *streaming current*.^{39,40}

The (iontronic) memristors schematically drawn in blue in Fig. 1(a) are *Simple Volatile Memristors* (SVMs).⁴¹ It has been demonstrated theoretically^{14,31} and experimentally^{12,15} that various fluidic iontronic memristors behave as SVMs,⁴¹ of which the electric conductance $g_i(t)$ is time-dependent and obeys the equation of motion (EOM),

$$\frac{dg_i}{dt} = \frac{g_{i,\infty}(V_i(t)) - g_i(t)}{\tau_i}. \quad (3.1)$$

Here, $g_{i,\infty}(V_i)$ is the steady-state conductance for a given voltage V_i , which is typically a sigmoidal function around the equilibrium conductance $g_{i,0} = g_{i,\infty}(0)$. It has been theoretically derived^{14,31} and experimentally observed^{12,15,17} that the intrinsic memory timescale τ_i of various iontronic SVMs scales quadratically with the device length L_i according to

$$\tau_i \propto \frac{L_i^2}{D}, \quad (3.2)$$

where D is the ionic diffusion coefficient, assumed equal for all ionic species of the aqueous electrolyte involved. Due to its dependence on L_i , the timescale τ_i can be individually chosen for each SVM across a wide range, which we will use in Sec. III C to implement the individual relaxation times in LI-BPNs. A variety of different iontronic SVMs are candidates for the circuit we propose here, including channels where the conductance memory is geometry-based,^{14,15} surface charge-based,³¹ colloid-based,¹² and polyelectrolyte-based,¹⁷ as schematically depicted in Fig. 1(c). Additionally, the power consumption of iontronic memristive devices can be extremely small, as low as order 10 fW per channel⁴² (assuming order 1 V driving force).

For our network demonstrations here, we chose to consider conical channel SVMs,¹⁴ but the results are representative of any SVM with a sigmoidal steady-state conductance. Specifically, we consider microfluidic channels as drawn in the top left of Fig. 1(c)

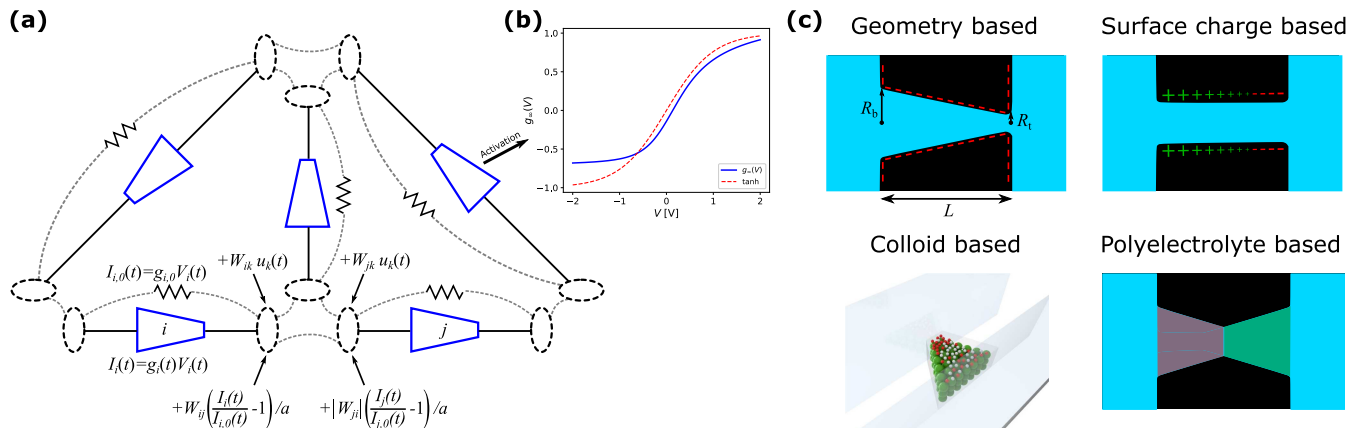


FIG. 1. (a) Schematic of a physical leaky integrator echo state or band-pass network circuit containing (iontronic) simple volatile memristors. Resistors are connected in parallel to the memristors with conductances equal to the equilibrium conductance of their respective parallel memristors. The terminals at either end of the memristors convert the incoming currents of neighboring memristor-resistor pairs to voltages of the next time step. For simplicity, only the inputs to and coupling between i and j memristors are included. The connections in this schematic are for illustrative purposes only and do not represent a specific circuit topology used in this work. (b) Dimensionless conductance g (blue), acting as a physical activation function in the circuit, compared to the standard ESN tanh activation function¹⁹ (red). (c) Several candidate iontronic simple volatile memristors where the conductance memory is geometry-based (top left),^{14,15} surface charge-based (top right),^{16,31} colloid-based¹² (bottom left), and polyelectrolyte-based (bottom right).¹⁷ All results in this work are from geometry-based iontronic memristors.¹⁴

with a base radius $R_b = 200$ nm, a tip radius $R_t = 50$ nm, a charge on the channel's surface of -2.4×10^{-22} C/nm², filled with an aqueous 1:1 electrolyte with equilibrium ion concentrations of 0.1 mM for both the positive and negative ions. The conductance of the channel is voltage-dependent and shown in Fig. 1(b) in blue (normalized and centered around 0). The proportionality constant in Eq. (3.2) can vary between iontronic devices, but the channels¹⁴ feature the relation $\tau_i = \frac{L_i^2}{12D}$, with lengths that can be fabricated from nm lengths^{43,44} all the way to mm length⁴⁵ scales, theoretically corresponding to a broad timescale range from $\sim 10^{-9}$ s up to $\sim 10^3$ s domains. While experimental evidence for the full range is still limited, the order $\sim 0.1 - 1$ s timescales that we will use here have been observed experimentally.^{12,15,46} The lengths L_i of the channels vary between different network applications, and even within individual networks for BPNs, to implement the different timescales τ_i as per Eq. (3.2). The full detailed physics and remaining parameters are described in the Appendix.

2. Current-to-voltage update rule

Consider the circuit schematically drawn in Fig. 1(a), where voltage terminals (dashed ellipses) connect parallel pairs of an Ohmic resistor and an SVM, here in the form of cone-shaped iontronic microfluidic channels¹⁴ (blue). Memristors are two-terminal devices with V_{it} and V_{ib} the voltages at the tip and base terminal, respectively, defined such that the voltage $V_i = V_{it} - V_{ib}$ over the SVM increases the conductance for positive V_i . Between the terminal pairs, two currents flow in parallel, a current $I_i = g_i(t)V_i(t)$ through the SVM and a current $I_{i,0} = g_{i,0}V_i(t)$ through the resistor with a fixed conductance $g_{i,0}$. The terminals obey the same update rule for the tip and base voltages V_{it} and V_{ib} , which depends on the currents I_j and $I_{j,0}$ of the neighboring terminals according to

$$V_{it} = \sum_{j: W_{ij} > 0} W_{ij} \left(\frac{I_j}{I_{j,0}} - 1 \right) a^{-1} + \sum_{j: W_{ij}^{\text{in}} > 0} W_{ij}^{\text{in}} u_j(t), \quad (3.3)$$

$$V_{ib} = \sum_{j: W_{ij} < 0} |W_{ij}| \left(\frac{I_j}{I_{j,0}} - 1 \right) a^{-1} + \sum_{j: W_{ij}^{\text{in}} < 0} |W_{ij}^{\text{in}}| u_j(t). \quad (3.4)$$

Here, W_{ij} , W_{ij}^{in} , and a are fixed and known *a priori*, and $I_j(t)$ and $I_{j,0}(t)$ are physical currents that only need to be measured locally. Lastly, $u_j(t)$ is the dynamic input. Depending on the input type, the input can feature an additional scaling factor $s^{\text{in}} \in \mathbb{R}$ such that $u_j(t) = s^{\text{in}} \tilde{u}_j(t)$. This factor can fix the units and ensure the input stays within a reasonable $\sim \pm 1$ V voltage regime. Notably, in Sec. IV C, $u_j(t)$ will receive its own local (pressure-to-)current-to-voltage update rule, where we analyze biological pressure signals as inputs by placing additional microfluidic channels between the pressure source and the SVM terminals. In such microfluidic channels, pressures are known to drive (electrical) *streaming currents* through the channels,^{39,40} which can then be converted according to a current-to-voltage update rule similar to Eq. (3.3), thereby providing a direct physical conversion between a biological signal and the ESN or BPN input without any intervention or interaction required from outside the network.

Although some functionality is assumed for the peripheral circuits at the terminals, this concerns only a straightforward conversion of locally measured currents to voltages with some *a priori* known fixed parameters. The use of peripheral circuitry for current-to-voltage conversions is relatively standard within neuromorphics, e.g., in the common neuromorphic circuits of coupled crossbar arrays that emulate artificial neural network current-to-voltage converters are employed to transform one array's current outputs to another array's voltage inputs.⁴⁷

B. Physical circuit and LI-ESN equivalence

For simplicity, let us initially consider all SVMs have equal length $L_i = L$ and, therefore, equal timescales $\tau_i = \tau$. In Sec. III C, we will make the straightforward extension to a range of timescales $\tau \in \mathbb{R}^N$, equivalent to a band-pass network. We will show that the physical circuit design in Fig. 1(a) is equivalent to the general mathematical LI-ESN description as in Eq. (2.1). The circuit will thereby be endowed with all its relevant derived properties, capabilities, and understandings,^{19–26} while the actual dynamics emerge from the intrinsic physics of the circuit, rather than numerically solving Eq. (2.1) in software. Specifically, the conductance EOM, Eq. (3.1), and steady-state conductance $g_{i,\infty}(V_i)$ naturally assume the role of the ESN dynamics and the activation function. Furthermore, the relative polarity of the voltage depending on the orientation of the SVMs provides a natural method to encode either positive or negative (adjacency) weight elements.

We introduce a straightforward conversion to a dimensionless conductance $g_i(t)$ normalized by $g_{i,0} = g_{i,\infty}(0)$, the equilibrium conductance

$$g_i(t) = \frac{g_i(t) - g_{i,0}}{ag_{i,0}}, \quad (3.5)$$

$$g_{i,\infty}(V_i) = \frac{g_{i,\infty}(V_i) - g_{i,0}}{g_{i,0}} \approx \tanh(V_i). \quad (3.6)$$

We stress that all results presented in this work exclusively use the physical function $g_{i,\infty}(V_i)$ ¹⁴ for the activation function $g_{i,\infty}(V_i)$, shown in blue in Fig. 1(b), alongside the function $\tanh(V_i)$ (red). The abovementioned similarity $g_{i,\infty}(V_i) \approx \tanh(V_i)$ only serves to support the equivalence to LI-ESNs.

The EOM of $g_i(t)$ is straightforwardly found through Eq. (3.1) as follows:

$$\begin{aligned} \frac{dg_i}{dt} &= \frac{1}{ag_{i,0}} \frac{(g_{i,\infty}(V_i) - g_{i,0}) - (g_i(t) - g_{i,0})}{\tau} \\ &= \frac{g_{i,\infty}(V_i) - ag_i(t)}{a\tau}. \end{aligned}$$

In vector notation, this becomes

$$\frac{d\mathbf{g}}{dt} = \frac{g_{\infty}(\mathbf{V}) - a\mathbf{g}(t)}{a\tau}, \quad (3.7)$$

where $g_{\infty}(\mathbf{V})$ is applied element-wise to \mathbf{V} .

Memristors are two-terminal devices with voltages V_{it} and V_{ib} at either terminal, respectively, here defined such that the voltage $V_i = V_{it} - V_{ib}$ over the SVM increases the conductance for positive

voltages. Both terminals have identical voltage update rules, which we now show are coupled to g as follows:

$$\begin{aligned} V_{i,t} &= \sum_{j:W_{ij}>0} W_{ij} \left(\frac{I_j}{I_{j,0}} - 1 \right) a^{-1} + \sum_{j:W_{ij}^{\text{in}}>0} W_{ij}^{\text{in}} u_j(t) \\ &= \sum_{j:W_{ij}>0} W_{ij} \frac{g_j(t) V_j - g_{j,0} V_j}{a g_{j,0} V_j} + \sum_{j:W_{ij}^{\text{in}}>0} W_{ij}^{\text{in}} u_j(t) \\ &= \sum_{j:W_{ij}>0} W_{ij} g_j + \sum_{j:W_{ij}^{\text{in}}>0} W_{ij}^{\text{in}} u_j. \end{aligned}$$

Similarly,

$$V_{i,b} = \sum_{j:W_{ij}<0} |W_{ij}| g_j(t) + \sum_{j:W_{ij}^{\text{in}}<0} |W_{ij}^{\text{in}}| u_j(t).$$

Therefore, the voltage over the SVM is given by

$$\begin{aligned} V_i &= V_{i,t} - V_{i,b} = \sum_{j:W_{ij}>0} W_{ij} g_j - \sum_{j:W_{ij}<0} |W_{ij}| g_j \\ &\quad + \sum_{j:W_{ij}^{\text{in}}>0} W_{ij}^{\text{in}} u_j - \sum_{j:W_{ij}^{\text{in}}<0} |W_{ij}^{\text{in}}| u_j, \end{aligned} \quad (3.8)$$

where we now see that negative weights are naturally encoded through the voltage sign reversal. Moreover, we see that the current-to-voltage conversion rule in Eq. (3.3) is equivalent to a matrix multiplication with the dimensionless conductances $g_i(t)$.

Compactly, Eq. (3.8) can be written in matrix vector notation as

$$\mathbf{V} = \mathbf{W}\mathbf{g} + \mathbf{W}^{\text{in}}\mathbf{u} \quad (3.9)$$

such that \mathbf{g} evolves according to

$$\frac{d\mathbf{g}}{dt} = \frac{g_\infty(\mathbf{W}^{\text{in}}\mathbf{u} + \mathbf{W}\mathbf{g}) - a\mathbf{g}(t)}{a\tau} \quad (3.10)$$

which we recognize as (the arguments inside) the activation function f in Eq. (2.1). Therefore, the straightforward current-to-voltage update rule described in Eq. (3.3) facilitates the matrix coupling between the nodes. This does require that the voltages can be adjusted quasi-instantaneously compared to the timescale τ of the SVM. To complete the equivalence to Eq. (2.1), let us consider the identifications,

$$\begin{aligned} g_\infty(x) &\approx \tanh(x) \leftrightarrow f(x), \\ \mathbf{g}(t) &\leftrightarrow \mathbf{x}(t), \\ a\tau &\leftrightarrow c. \end{aligned}$$

We now see that Eq. (3.10) is identical to Eq. (2.1), while being completely physically facilitated in the circuit shown in Fig. 1(a).

Above we described how the dynamics of the circuit depicted in Fig. 1(a) are equivalent to the ESN dynamics as per Eq. (2.1). Moreover, since applying the input and reading the output are performed by standard matrix multiplications $\mathbf{W}^{\text{in}}\mathbf{u}$ and $\mathbf{W}^{\text{out}}\mathbf{x}$, respectively, these actions too can be physically realized using crossbar arrays, which too could be implemented using ionic devices.^{48–54} Therefore,

excitingly, the full (ionic) hardware implementation of our LI-ESN circuit should be directly physically possible.

C. Physical LI-BPN

Because each SVM can straightforwardly be designed to feature its own timescale $\tau_i \propto L_i^2/D$ by varying the length L_i of the individual devices, we can easily go beyond ESNs to BPNs, which are known to perform considerably better on input tasks that feature components that span multiple frequencies.²⁷

Varying the lengths between the different SVMs corresponds to converting τ to a vector $\tau \rightarrow \boldsymbol{\tau} \in \mathbb{R}^N$ such that Eq. (3.10) becomes

$$\frac{d\mathbf{g}}{dt} = \frac{g_\infty(\mathbf{W}^{\text{in}}\mathbf{u} + \mathbf{W}\mathbf{g}) - a\mathbf{g}(t)}{a} \oslash \boldsymbol{\tau}, \quad (3.11)$$

with $\oslash \boldsymbol{\tau}$ being element-wise (Hadamard) division. We note that the dimensionless $g_{i,\infty}$ (V) is independent from L_i , so the extension from Eq. (3.10) remains valid. With the same identification steps as in Sec. III B, we see that Eq. (3.11) is equivalent to the mathematical description of LI-BPNs as in Eq. (2.3). Therefore, the circuit depicted in Fig. 1(a) can be designed to be either an ESN or BPN, depending on whether the channel lengths vary.

IV. TIME SERIES ANALYSIS TASKS

A. Mackey–Glass time series

To reproduce some of the known capabilities of LI-ESNs, and to compare to the time series prediction performance of other methods, we use our iontronic SVM based circuit to predict the synthetic Mackey–Glass⁵⁵ time series $P(t)$. This time series is one of the most common generated data sets to test ESNs on,⁵⁶ generated by

$$\frac{dP(t)}{dt} = \frac{\beta P(t - t_{\text{delay}})}{\theta + P(t - t_{\text{delay}})^n} - \gamma P(t), \quad (4.1)$$

where we use $\beta = 0.2$, $\theta = 1$, $\gamma = 0.1$, $n = 10$, and $t_{\text{delay}} = 17$. The first 17 time steps are randomly generated values in the range $[-1, 1]$. For the aforementioned parameters, Eq. (4.1) is known to feature a chaotic attractor.⁵⁷ As in Ref. 21, Eq. (4.1) is then rescaled $P(t) \mapsto \tanh(P(t) - 1)$ such that $P(t) \in [-1, 1]$ for all t .

A reservoir was used with parameters inspired by Ref. 21 of $K = 1$, $N = 400$, and $L = 1$, a network sparsity of 0.75, spectral radius of 0.95, $c = 2.27$ s, $\delta = 1$ s, a leaking rate of $a = 0.95$, and an input scaling of $s^{\text{in}} = 0.45$ V. The output matrix \mathbf{W}^{out} was trained on a test set of length 3000 s using ridge regression²⁹ discarding the first 100 s as a washout. Testing was done on a newly generated Mackey–Glass series with the same parameters, but a different random initialization of the first 17 steps. We distinguish between a washout period, where only the next time (1 s) step needs is predicted with the true signal as input, and free-running classification, where the network receives its own output as input for the next time step while receiving no information from the true signal. The resulting washout period ($t < t_{\text{free}} = 1000$ s) and free-running ($t > t_{\text{free}} = 1000$ s) predictions are shown in Fig. 2(a).

In Fig. 2(a), we see that the network with $N = 400$ is able to accurately predict the Mackey–Glass series for several 100 s. To quantify this performance and to compare it to previous results, we calculate the normalized root mean squared error (NRMSE)

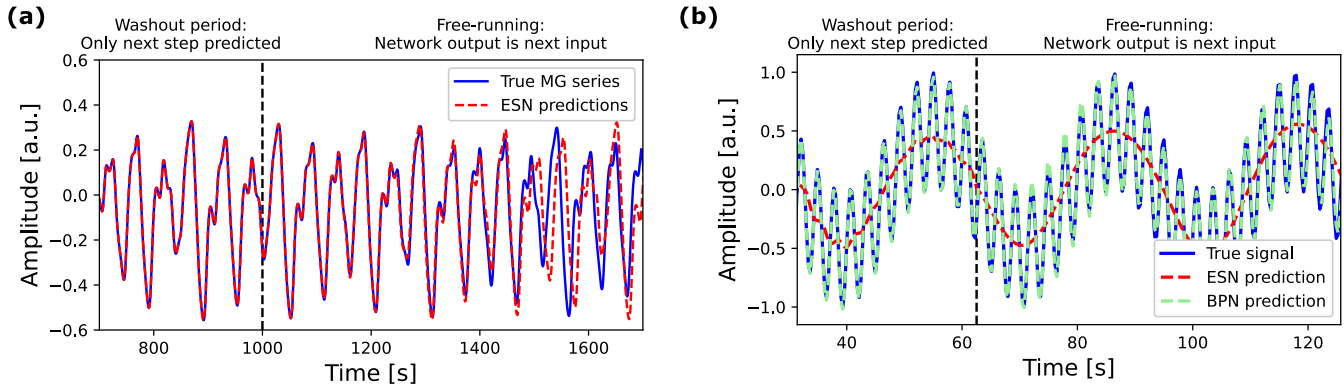


FIG. 2. (a) Mackey–Glass series predictions from a physical LI-ESN circuit containing iontronic conical channel memristors.¹⁴ For $t < 1000$, only the next step is predicted (with time stepsize of $\delta = 1$ s), for $t > 1000$, the network receives no further input from the true time series and uses its own predicted output as input for the next step. Predicting 84 steps ahead yielded a RMSE of $\text{RMSE}_{84} \approx 0.001$, averaged over 20 network initializations, comparable to earlier results using LI-ESNs²¹ and outperforming several neural network based approaches.^{58,59} (b) Harmonic time series predictions of an LI-ESN (red) and LI-BPN (green) containing 12 iontronic SVMs, showing that LI-ESNs struggle with the variations in signal timescales.

of the prediction $\hat{P}_i(t_{\text{free}} + 84)$ with the true value $P_i(t_{\text{free}} + 84)$. This entails that we compare the output of the network after it received its own prediction as input for 84 steps (i.e., 84 s into free-running mode), which we average over $T = 20$ different random initializations of the LI-ESN, according to

$$\text{NRMSE}_{84} = \sqrt{\frac{\sum_{i=1}^{T=20} \left(\hat{P}_i(t_{\text{free}} + 84) - P_i(t_{\text{free}} + 84) \right)^2}{\sigma^2 T}},$$

with $\sigma^2 \approx 0.05$ being the variance of the input data.

With our fully physically realizable circuit, we find $\text{NRMSE}_{84} \approx 0.008$ on the test series. This outperforms other approaches, such as self-organizing feature map models reaching $\text{NRMSE}_{84} \approx 0.022$ for a comparable training set size⁶⁰ (which is the best performance in survey Ref. 61), and various neural network based approaches^{58,59,62} that achieve $\text{NRMSE}_{84} \gtrsim 0.1$ (converting from RMSE to NRMSE assuming similar σ^2). This performance in and of itself is not surprising, as LI-ESNs with 400 nodes have long been shown to be capable of this.²¹ Therefore, recreating this using the physical conical channel SVM equations supports our claim of this work that our *physical* circuit is equivalent to these ESNs and thereby to all their capabilities.

Notably, in both predictions shown in Fig. 2, we use the physical steady-state conductance g_{∞} (V) as a activation function.¹⁴ Moreover, we are able to translate the parameters a and c to the physical length of the memristors, which for the results of Fig. 2(a), with diffusion coefficient $D = 1 \mu\text{m}^2 \text{ms}^{-1}$, would be $L = 169 \mu\text{m}$. Otherwise, the remaining parameters (e.g., channel radius, salt concentration, surface charge, etc.) are as briefly reported in Sec. III A 1 and in full detail in the Appendix.

B. Band-pass network for multi-frequency signals

All nodes in LI-ESNs feature the same universal relaxation timescale c . Therefore, LI-ESNs can struggle with inputs that incorporate signal timescales of different magnitudes,²⁷ whereas LI-BPNs have an inductive bias toward multi-frequency signals due to their heterogeneous timescales.^{27,30} This increase in model capacity can straightforwardly be physically embedded by varying the length of the iontronic SVMs, as discussed in Sec. III A 1. To demonstrate this, we compare small $N = 12$ LI-ESNs and LI-BPNs. The hyperparameters of both are optimized with the *optuna* framework (version 4.4.0)⁶³ using the default tree-structured Parzen estimator algorithm, optimization code with all details is available online.²⁸ The optimizations were for predicting a simple harmonic function that features oscillations on a ~ 1 and ~ 10 s scale given by

$$y_{\text{har}}(t) = \sin(t) \cos(1.2t),$$

which is discretized with time step sizes of $\delta = \pi/10$ s. The networks were trained on a domain of $[0, 80\pi]$. First, we consider a small LI-ESN of $K = 1$, $N = 12$, and $L = 1$, a network sparsity of 0.67, a timescale of $c = 1.87$ s, a leaking rate of $a = 0.44$, a spectral radius of 0.32, and an input scaling of $s^{\text{in}} = 0.26$ V. The output matrix is again trained using ridge regression²⁹ with a regularization of 4×10^{-5} . All results presented below did not appear to be sensitive to the precise hyperparameters.

The resulting ESN predictions can be seen in the red curve of Fig. 2(b), where it is clear that the LI-ESN is not able to capture the shorter timescale oscillations. Shortening the timescale c of the ESN did allow it to also predict also these shorter oscillations during the washout period (i.e., with the true signal as input), but then the network quickly diverged in free-running mode, leading to a worse overall performance. Since we only model 12 SVM nodes with weights that are randomly generated, there is significant variability between different initializations. However, these observations were robust and the LI-ESN consistently performed poorly on predicting the time series, as we will quantify below.

Converting our LI-ESN to a similarly sized LI-BPN significantly improves the performance. In this case, timescales are drawn from a normal distribution $C_i \sim N(\mu_c, \sigma_c^2)$, with $\mu_c = 2.79$ s and $\sigma_c = 9.9$ s, such that the timescale of each SVM node is set by $c_i = \max(\mu_c/5, C_i)$ to ensure all timescales are positive. More sophisticated methods of choosing the timescales can be considered, but for now, this simple approach is sufficient to demonstrate the benefits that LI-BPNs provide. Furthermore, the BPN features a network sparsity of 0.35, a leaking rate of $a = 0.86$, a spectral radius of 0.76, an input scaling of $s^{\text{in}} = 0.21$ V, and a regularization for ridge regression of 1×10^{-6} .

As shown in the green curve of Fig. 2(b), the resulting LI-BPN can successfully predict the time series with only 12 nodes. Again, there is some variability between different initiations of the small network, but the improved performance was consistent. The RMSE of predicting the free-running domain shown in Fig. 2(b) (for $[20, \pi, 40\pi]$), averaged over 100 different ESN and BPN initiations, shows that the LI-BPN features a factor 3 lower RMSE (≈ 0.04) than the LI-ESN (≈ 0.13). Quantitatively this seems like a somewhat marginal difference, but this is indicative of the qualitative observation that the higher frequency oscillations are not captured by the ESN circuit. We stress that physically, this conversion from an ESN to BPN is straightforward, as the variability in timescales can be realized on-chip through varying the channel lengths.

C. Airway pressure as direct physical input

Thus far, we have provided demonstrations of analyzing some synthetic time series with LI-ESNs and LI-BPNs containing iontronic SVMs. Here, we will consider measurements of ventilator pressures that were designed to accurately mimic the airway pressures present in lungs during breathing.⁵⁴ For this task, we will leverage two useful properties of iontronic SVMs: (i) their tunable timescales that coincide with timescales of natural or biological

origin and (ii) the intrinsic responsiveness of iontronic systems to pressure inputs.

As schematically shown in the blue curve of Fig. 3(a), the input pressure is applied at one end of a cylindrical microfluidic channel, Fig. 3a, (bottom), of length $L = 200$ μm and radius $R = 25$ μm , carrying a typical surface potential of $\psi_0 = -40$ mV. These channels feature a coupling between pressure and electric (ionic) current, specifically the resulting pressure drop $\Delta p(t)$ drives a so-called electric streaming current [red curve of Fig. 3(a)] given by⁴⁰

$$I_p(t) = \pi R^2 \frac{\epsilon \psi_0}{\eta} \frac{\Delta p(t)}{L},$$

with η and ϵ being the shear viscosity and electric permittivity of water. The length L and radius R of the channel are chosen such that for the typical biological pressure signal amplitude of ~ 10 mbar, the streaming current $I_p(t)$ will be of order ~ 1 nA. This is a current strength that can be reliably measured for single channels driven by pressure.^{36,39} The ionic streaming current becomes the input $u(t) = s^{\text{in}} I_p(t)$. Similar to how the ionic currents through the SVMs are converted to voltage contribution updates within the terminals, the streaming current can now be directly coupled to the SVMs such that the voltage contribution $s^{\text{in}} I_p(t) W_i^{\text{in}}$ of the inputs is as per Eq. (3.3).

Using the ventilator pressure data,⁶⁴ an example of which is shown (blue) in Fig. 3(a), we perform two tasks. First, the easier task of classifying whether the expiratory valve (i.e., the valve that lets air out) is open or closed, and then the harder task of predicting three steps (≈ 0.1 s) ahead, i.e., pressure is applied as input and the goal is to predict step $n + 3$ when at step n . The data were split into two parts for training and testing, an 80 000 step (i.e., 2700 s) segment for training, and a different 20 000 step (i.e., 680 s) segment for testing (in both instances discarding the first 1000 steps as washout). Hyperparameters were optimized for analyzing ventilator pressure data again with optuna framework (version 4.4.0),⁶³ using the default

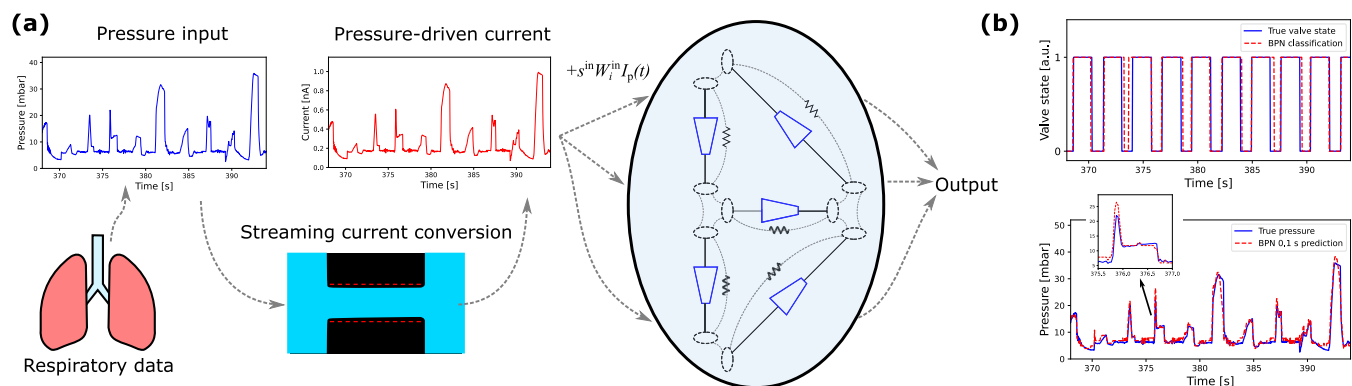


FIG. 3. (a) Schematic depiction of how the applied (airway) pressure (blue graph) drives an electric streaming current (red graph) through a cylindrical microfluidic channel (bottom).^{39,40} This ionic current is then converted to voltage updates in the terminals of the iontronic SVMs, similar to how the ionic currents through the memristors are coupled to the neighboring nodes. The connections in this schematic are for illustrative purposes only and do not represent a specific circuit topology used in this work. (b) Results of a classification task (top graph) of whether the expiratory valve (i.e., the valve that lets air out) is open or closed, using a small LI-BPN consisting of seven iontronic SVMs. Additionally, a prediction task (bottom graph) of what the pressure will be 0.1 s in the future using a LI-BPN containing 200 iontronic SVMs. The inset shows that <0.1 s variations in the pressure are predicted by the network.

tree-structured Parzen estimator algorithm, optimization code with all details is available online.²⁸ The resulting LI-BPNs feature a network sparsity of 0.017, time step $\delta = 0.034\,035\text{ s}$ (matching with the data), timescale parameters $\mu_c = 0.27\text{ s}$ and $\sigma_c = 1.89\text{ s}$, input scaling $s^{\text{in}} = 0.11\text{ V/nA}$, a leaking rate of $a = 0.98$, and a regularization of 1.69×10^{-6} .

In the top graph of Fig. 3(b), we show the valve classifications in red, compared to the true value in blue, corresponding to the input depicted in Fig. 3(a). For these classifications, a small LI-BPN circuit was modeled with only $N = 7$ iontronic SVMs, achieving $\sim 91\%$ on the test data accuracy averaged over 20 initializations. For comparison, we carried out the same task with a linear autoregression model of order 7 such that the number of fit-parameters matches the LI-BPN, achieving a lower accuracy of $\sim 82\%$. This is partially because such a model only uses a slim window of the last seven time steps for its classification. Spreading out this window through subsampling by using the values 8 m steps back, with $m \in \{1, 2, \dots, 7\}$, improves accuracy to 90%, almost matching the LI-BPN. However, it is not directly clear how a simple hardware implementation of such a (pressure-driven) autoregression model would be achieved, especially in the more complicated subsampling approach.

In the bottom graph of Fig. 3(b), we show the pressure predictions using a larger LI-BPN circuit containing $N = 200$ SVMs (otherwise the same parameters). Each point in the red graph is a prediction of 0.1 s ahead, achieving an RMSE of $\approx 3.0\text{ mbar}$ (measured over the full test data length). This is a slight improvement over the RMSE of $\approx 3.2\text{ mbar}$ achieved with a linear autoregression model of order 200, where we again note that it is not directly clear how such a (pressure-driven) autoregression model can be straightforwardly directly physically implemented like the LI-BPN. Although most individual pressure waves are longer than the prediction window of 0.1 s, there are certainly $< 0.1\text{ s}$ features within each pressure wave that are still correctly predicted by the network, as shown in the inset in the bottom graph of Fig. 3(b). We note that this task is especially difficult because normally multiple parameters accompany each individual waveform,⁶⁴ whereas here we solely provide the pressure as input.

V. DISCUSSION AND CONCLUSION

In summary, we proposed a physical circuit design that exhibits a one-to-one correspondence to the well-established mathematical description of the reservoir computing frameworks of leaky integrator echo state and band-pass networks.^{19–27} This circuit incorporates fluidic iontronic memristors,^{4–8} whose voltage-dependent conductance and conductance memory facilitate the activation function and dynamics of the nodes, respectively. The terminals at either end of the memristors feature fixed peripheral circuits that convert locally measured currents to voltages, forming a physical realization of the matrix coupling between nodes. By solving the physical equations of microfluidic conical channel memristors¹⁴ within such a circuit, we successfully analyze several signals such as the Mackey–Glass time series, supporting our claim that such a circuit is a physical manifestation of LI-ESNs and LI-BPNs.

A desirably property of these iontronic memristors is the dependence of their conductance memory timescale on the channel

length, i.e., each memristor can individually be designed to feature a certain relaxation time. This corresponds to going from a LI-ESN to a LI-BPN, which are known to perform significantly better on inputs that feature multiple timescales,²⁷ demonstrated here by showing a circuit with channels of differing lengths, i.e., a physical LI-BPN circuit, performs significantly better on predicting a harmonic time series with features across different timescales than a physical LI-ESN.

Lastly, we leverage another unique property of microfluidic devices, their intrinsic responsiveness to applied pressures. Applied pressures can drive electric currents through microfluidic channels, allowing a pressure signal to be converted to an ionic current. By coupling this to the existing current-to-voltage conversion in the terminals through an additional input channel, the pressure is converted to the network's input without any intervention or interaction required from outside the network. We demonstrated this by classifying and predicting features of simulated biophysically realistic data of airway pressures during breathing.⁶⁴

Some functionality of our proposed circuit design lies in the peripheral circuitry that connect the terminals of the memristors, namely, converting currents of neighboring memristors to updated voltages. Current-to-voltage converting peripheral circuits are common within neuromorphics,⁴⁷ supporting that such circuitry can efficiently be implemented. We ignored device noise in this study, assuming that the currents through the memristors can be accurately and reliably measured. Interestingly, LI-ESNs have also been theoretically studied with noise,^{19,20,26} so both from a theoretical and from a physical implementation perspective, this is an interesting direction to study next. Additionally, analyzing more real-world data, such as other biological (pressure) sources or chemical signals,^{65–67} would be a relevant expansion. Lastly, optimizing the circuit design for specific tasks could be of interest, as our focus here was primarily on establishing the correspondence between our physical circuit design and LI-ESNs or LI-BPNs, not on optimization.

With power consumptions of iontronic memristive devices as low as order 10 fW per channel,⁴² the overall circuit could potentially operate at very low power, where we leave a more thorough estimate for future work. The energy usage will also depend on the peripheral circuit, which can be a significant contributor to power consumption,⁶⁸ and on its ability to integrate such low power (sub)-nanoscale devices. Input and output actions, performed by standard matrix multiplications, could also be physically realized using cross-bar arrays implemented with (ionic) devices,^{48–54} further supporting our design as a fully physically realizable low power echo state or band-pass network circuit.

In conclusion, our proposed iontronic memristor based physical circuit design is theoretically equivalent to the well-established reservoir computing methods of leaky integrator echo state and band-pass networks. This is supported by performing several time series prediction and classification tasks. The fluidic devices do not necessarily need to be integrated in a fully fluidic circuit, thereby circumventing existing challenges of manufacturing integrated fully fluidic chips.⁶⁹ Notably, airway pressure signals were used as inputs, leveraging iontronics' intrinsic pressure responsiveness and natural timescales matching biology, with pressure classification tasks achieved using as few as seven memristors. Therefore, this work provides a design that can advance the field of iontronic

(neuromorphic) computing, while exploiting some of iontronics' intrinsic properties.

ACKNOWLEDGMENTS

We thank Dr. Jaehyun Kim and Professor Jungyul Park for providing us with the Colloid Based channel schematic shown in Fig. 1(c).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

T. M. Kamsma: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Project administration (equal); Software (equal); Visualization (equal); Writing – original draft (equal). **J. J. Teijema:** Data curation (equal); Investigation (equal); Project administration (equal); Software (equal). **R. van Roij:** Supervision (equal); Writing – original draft (equal). **C. Spitoni:** Supervision (equal); Writing – original draft (equal).

DATA AVAILABILITY

The data that support the findings of this study are available within the article.

APPENDIX: CONICAL CHANNEL CONDUCTANCE

This appendix is a summary of the results described in Ref. 14. We consider an azimuthally symmetric single conical channel, schematically depicted in Fig. 1c, (top left), of length L (given for each specific network in the main text) with the central axis at radial coordinate $r = 0$ and a radius described by $R(x) = R_b - x\Delta R/L$ for $x \in [0, L]$, where $R_b = 200$ nm is the base radius at $x = 0$, and $R_t = R_b - \Delta R = 50$ nm, the tip radius at $x = L \gg R_b$. The channel connects two bulk reservoirs of an incompressible aqueous 1:1 electrolyte with viscosity $\eta = 1.01$ mPa s, mass density $\rho_m = 10^3$ kg m $^{-3}$, and electric permittivity $\epsilon = 0.71$ nF m $^{-1}$, at the far side of both reservoirs we impose a fixed pressure $P = P_0$ and fixed ion concentrations $\rho_{\pm} = \rho_b = 0.1$ mM. The channel wall carries a uniform surface charge density $e\sigma = -0.0015$ enm $^{-2}$, screened by an electric double layer with Debye length $\lambda_D \approx 10$ nm, resulting in an electric surface potential of $\psi_0 \approx -10$ mV. The ions have concentrations $\rho_{\pm}(x, r)$, diffusion coefficients $D_{\pm} = D = 1$ μ m 2 ms $^{-1}$, and charge $\pm e$ with e being the proton charge. Over the channel, we impose an electric potential $V(t)$, defined as the voltage $V_t(t)$ in the tip reservoir minus the voltage $V_b(t)$ in the base reservoir.

The steady-state conductance of a conical channel depends on the voltage-dependent radially averaged salt concentration profile $\bar{\rho}_s(x, V) = 2 \int_0^{R(x)} r(\rho_+(x, r) + \rho_-(x, r))dr/R(x)^2$ that exhibits salt concentration polarization upon an applied voltage. The consequent voltage-dependent steady-state channel conductance is

described by³⁷

$$\frac{g_{\infty}(V)}{g_0} = \int_0^L \bar{\rho}_s(x, V) dx / (2\rho_b L) \\ = 1 + \Delta g \int_0^L \left[\frac{x}{L} \frac{R_t}{R(x)} - \frac{e^{\frac{Pe(V)}{L} \frac{R_t^2}{R_b R(x)}} - 1}{e^{\frac{Pe(V)}{L} \frac{R_t}{R_b}} - 1} \right] dx / L, \quad (A1)$$

where $g_0 = (\pi R_t R_b / L)(2\rho_b e^2 D / k_B T)$, $Pe(V) = Q(V)L / (D\pi R_t^2)$ the Péclet number at the narrow end, $Q(V) = -\pi R_t R_b \epsilon \psi_0 / (\eta L) V$ the volumetric fluid flow through the channel, and $\Delta g \equiv -e \Delta R \eta \sigma D / (\rho_b R_b R_t \epsilon \psi_0 k_B T)$.

The dynamic (dimensional) conductance $g(t)$ is well described by¹⁴

$$\frac{dg(t)}{dt} = \frac{g_{\infty}(V(t)) - g(t)}{\tau}, \quad (A2)$$

with $\tau = L^2 / 12D$ being the typical conductance memory time of the channel.

REFERENCES

- ¹K. Nakajima and I. Fischer, "Natural computing series reservoir computing theory, physical implementations, and applications," Technical Report (2021).
- ²M. Cuccchi, S. Abreu, G. Ciccone, D. Brunner, and H. Kleemann, "Hands-on reservoir computing: A tutorial for practical implementation," *Neuromorph. Comput. Eng.* **2**, 032002 (2022).
- ³X. Liang, J. Tang, Y. Zhong, B. Gao, H. Qian, and H. Wu, "Physical reservoir computing with emerging electronics," *Nat. Electron.* **7**, 193–206 (2024).
- ⁴A. Noy, Z. Li, and S. B. Darling, "Fluid learning: Mimicking brain computing with neuromorphic nanofluidic devices," *Nano Today* **53**, 102043 (2023).
- ⁵A. Noy and S. B. Darling, "Nanofluidic computing makes a splash," *Science* **379**, 143–144 (2023).
- ⁶Y. Hou, Y. Ling, Y. Wang, M. Wang, Y. Chen, X. Li, and X. Hou, "Learning from the brain: Bioinspired nanofluidics," *J. Phys. Chem. Lett.* **14**, 2891–2900 (2023).
- ⁷L. Yu, X. Li, C. Luo, Z. Lei, Y. Wang, Y. Hou, M. Wang, and X. Hou, "Bioinspired nanofluidic iontronics for brain-like computing," *Nano Res.* **17**, 503–514 (2023).
- ⁸B. Xie, T. Xiong, W. Li, T. Gao, J. Zong, Y. Liu, and P. Yu, "Perspective on nanofluidic memristors: From mechanism to application," *Chem. Asian J.* **17**, e202200682 (2022).
- ⁹Q. Fan, J. Shang, X. Yuan, Z. Zhang, and J. Sha, "Emerging liquid-based memristive devices for neuromorphic computation," *Small Methods* **9**, 2402218 (2025).
- ¹⁰C. Suwen Law, J. Wang, K. Nielsch, A. D. Abell, J. Bisquert, A. Santos, and J. Appl Phys, "Recent advances in fluidic neuromorphic computing," *Appl. Phys. Rev.* **12**, 21309 (2025).
- ¹¹H. Lv, R. Liu, and Y. Zhang, "Advancements in fluidic ionic devices: Implications for neuromorphic integrated circuit design," *ACS Sens.* **10**, 4804–4824 (2025).
- ¹²T. M. Kamsma, J. Kim, K. Kim, W. Q. Boon, C. Spitoni, J. Park, and R. van Roij, "Brain-inspired computing with fluidic iontronic nanochannels," *Proc. Natl. Acad. Sci. U. S. A.* **121**, e2320242121 (2024).
- ¹³S. Portillo, P. Ramirez, S. Mafe, and J. Cervera, "Neuromorphic reservoir computing with memristive nanofluidic diodes," *Nano Lett.* **25**, 9928–9934 (2025).
- ¹⁴T. M. Kamsma, W. Q. Boon, T. Ter Rele, C. Spitoni, and R. Van Roij, "Iontronic neuromorphic signaling with conical microfluidic memristors," *Phys. Rev. Lett.* **130**, 268401 (2023).
- ¹⁵J. Cervera, S. Portillo, P. Ramirez, and S. Mafe, "Modeling of memory effects in nanofluidic diodes," *Phys. Fluids* **36**, 47129 (2024).
- ¹⁶T. M. Kamsma, E. A. Rossing, C. Spitoni, and R. V. Roij, "Advanced iontronic spiking modes with multiscale diffusive dynamics in a fluidic circuit," *Neuromorph. Comput. Eng.* **4**, 024003 (2024).

- ¹⁷Z. Zhang, B. Sabbagh, Y. Chen, and G. Yossifon, “Geometrically scalable iontronic memristors: Employing bipolar polyelectrolyte gels for neuromorphic systems,” *ACS Nano* **18**, 15025–15034 (2024).
- ¹⁸E. Chicca and G. Indiveri, “A recipe for creating ideal hybrid memristive-CMOS neuromorphic processing systems,” *Appl. Phys. Lett.* **116**, 120501 (2020).
- ¹⁹H. Jaeger, M. Lukoševičius, D. Popovici, and U. Siewert, “Optimization and applications of echo state networks with leaky-integrator neurons,” *Neural Networks* **20**, 335–352 (2007).
- ²⁰H. Jaeger, “Adaptive nonlinear system identification with echo state networks,” *Adv. Neural Inf. Process. Syst.* **15**, 609–616 (2002).
- ²¹H. Jaeger, “The ‘echo state’ approach to analysing and training recurrent neural networks—With an erratum note 1,” *GMD Rep.* **148**, 13 (2010).
- ²²I. B. Yildiz, H. Jaeger, and S. J. Kiebel, “Re-visiting the echo state property,” *Neural Networks* **35**, 1–9 (2012).
- ²³M. Lukoševičius, *A Practical Guide to Applying Echo State Networks*, Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Lecture No. 7700 (Springer, 2012), pp. 659–686.
- ²⁴B. Whiteaker and P. Gerstoft, “Leaky integrator dynamical systems and reachable sets,” in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing—Proceedings 2021—June* (IEEE, 2021), pp. 4025–4029.
- ²⁵J. Lin, F. L. Chung, and S. Wang, “A fast parametric and structural transfer leaky integrator echo state network for reservoir computing,” *IEEE Trans. Syst. Man Cybern. Syst.* **54**, 3257–3269 (2024).
- ²⁶C. Sun, M. Song, D. Cai, B. Zhang, S. Hong, and H. Li, “A systematic review of echo state networks from design to application,” *IEEE Trans. Artif. Intell.* **5**, 23–37 (2024).
- ²⁷F. Wyffels, B. Schrauwen, D. Verstraeten, and D. Stroobandt, “Band-pass reservoir computing,” in *Proceedings of the International Joint Conference on Neural Networks* (IEEE, 2008), pp. 3204–3209.
- ²⁸T. M. Kamsma and J. J. Teijema (2025). “Code Repository for: Pyontronics (v0.2.0),” Zenodo. <https://doi.org/10.5281/zenodo.17076466>.
- ²⁹A. E. Hoerl and R. W. Kennard, “Ridge regression: Applications to nonorthogonal problems,” *Technometrics* **12**, 69–82 (1970).
- ³⁰G. Holzmann and H. Hauser, “Echo state networks with filter neurons and a delay & sum readout,” *Neural Networks* **23**, 244–256 (2010).
- ³¹T. M. Kamsma, W. Q. Boon, C. Spitoni, and R. van Roij, “Unveiling the capabilities of bipolar conical channels in neuromorphic iontronics,” *Faraday Discuss.* **246**, 125–140 (2023).
- ³²C. D. Schuman, S. R. Kulkarni, M. Parsa, J. P. Mitchell, P. Date, and B. Kay, “Opportunities for neuromorphic computing algorithms and applications,” *Nat. Comput. Sci.* **2**, 10–19 (2022).
- ³³V. K. Sangwan and M. C. Hersam, “Neuromorphic nanoelectronic materials,” *Nat. Nanotechnol.* **15**, 517–528 (2020).
- ³⁴J. Zhu, T. Zhang, Y. Yang, and R. Huang, “A comprehensive review on emerging artificial neuromorphic devices,” *Appl. Phys. Rev.* **7**, 011312 (2020).
- ³⁵C. D. Schuman, T. E. Potok, R. M. Patton, J. D. Birdwell, M. E. Dean, G. S. Rose, and J. S. Plank, “A survey of neuromorphic computing and neural networks in hardware,” *arXiv:1705.06963* (2017).
- ³⁶L. Jubin, A. Poggioli, A. Siria, and L. Bocquet, “Dramatic pressure-sensitive ion conduction in conical nanopores,” *Proc. Natl. Acad. Sci. U. S. A.* **115**, 4063–4068 (2018).
- ³⁷W. Q. Boon, T. E. Veenstra, M. Dijkstra, and R. Van Roij, “Pressure-sensitive ion conduction in a conical channel: Optimal pressure and geometry,” *Phys. Fluids* **34**, 101701 (2022).
- ³⁸A. Barnaveli, T. M. Kamsma, W. Q. Boon, and R. van Roij, “Pressure-gated microfluidic memristor for pulsatile information processing,” *Phys. Rev. Appl.* **22**, 054057 (2024).
- ³⁹F. H. J. Van Der Heyden, D. Stein, and C. Dekker, “Streaming currents in a single nanofluidic channel,” *Phys. Rev. Lett.* **95**, 116104 (2005).
- ⁴⁰B. L. Werkhoven and R. Van Roij, “Coupled water, charge and salt transport in heterogeneous nano-fluidic systems,” *Soft Matter* **16**, 1527–1537 (2020).
- ⁴¹T. M. Kamsma, R. van Roij, and C. Spitoni, “A simple mathematical theory for simple volatile memristors and their spiking circuits,” *Chaos Solitons Fractals* **186**, 115320 (2024).
- ⁴²D. Shi, W. Wang, Y. Liang, L. Duan, G. Du, and Y. Xie, “Ultralow energy consumption angstrom-fluidic memristor,” *Nano Lett.* **23**, 11662–11668 (2023).
- ⁴³A. J. Storm, J. H. Chen, X. S. Ling, H. W. Zandbergen, and C. Dekker, “Fabrication of solid-state nanopores with single-nanometre precision,” *Nat. Mater.* **2**, 537–540 (2003).
- ⁴⁴C. Yang, P. Hinkle, J. Menestrina, I. V. Vlassiouk, and Z. S. Siwy, “Polarization of gold in nanopores leads to ion current rectification,” *J. Phys. Chem. Lett.* **7**, 4152–4158 (2016).
- ⁴⁵L. Zhou, A. Eden, K. H. Chou, D. E. Huber, and S. Pennathur, “Nanofluidic diodes based on asymmetric bio-inspired surface coatings in straight glass nanochannels,” *Faraday Discuss.* **246**, 356–369 (2023).
- ⁴⁶D. Wang, M. Kvetny, J. Liu, W. Brown, Y. Li, and G. Wang, “Transmembrane potential across single conical nanopores and resulting memristive and memcapacitive ion transport,” *J. Am. Chem. Soc.* **134**, 3651–3654 (2012).
- ⁴⁷C. Li, D. Belkin, Y. Li, P. Yan, M. Hu, N. Ge, H. Jiang, E. Montgomery, P. Lin, Z. Wang, W. Song, J. P. Strachan, M. Barnell, Q. Wu, R. S. Williams, J. J. Yang, and Q. Xia, “Efficient and self-adaptive in-situ learning in multilayer memristor neural networks,” *Nat. Commun.* **9**, 1–8 (2018).
- ⁴⁸K. Liu, Y. Wang, M. Sun, J. Lu, D. Shi, and Y. Xie, “Resistance-restorable nanofluidic memristor and neuromorphic chip,” *Nano Lett.* **25**, 6530–6538 (2025).
- ⁴⁹Y. Van De Burgt, A. Melianas, S. T. Keene, G. Malliaras, and A. Salleo, “Organic electronics for neuromorphic computing,” *Nat. Electron.* **1**, 386–397 (2018).
- ⁵⁰S. Kazemzadeh, T. Stevens, and Y. van de Burgt, “All organic fully integrated neuromorphic crossbar array,” *Adv. Electron. Mater.* 2500054 (published online 2025).
- ⁵¹G. Xu, H. Cui, L. Wang, M. Zhang, W. Liu, T. Mei, B. Wu, C. Wan, and K. Xiao, “Ångström-scale-channel iontronic memristors for neuromorphic computing,” *ACS Appl. Mater. Interfaces* **17**, 34659–34668 (2025).
- ⁵²Y. Zhang, E. R. W. van Doremale, G. Ye, T. Stevens, J. Song, R. C. Chiechi, and Y. van de Burgt, “Adaptive biosensing and neuromorphic classification based on an ambipolar organic mixed ionic–electronic conductor,” *Adv. Mater.* **34**, 2200393 (2022).
- ⁵³L. S. Hu, M. Fattori, W. Schilp, R. Verbeek, S. Kazemzadeh, Y. van de Burgt, A. J. Kronemeijer, G. Gelinck, and E. Cantatore, “An energy-efficient solid-state organic device array for neuromorphic computing,” *IEEE Trans. Electron Devices* **70**, 6520–6525 (2023).
- ⁵⁴Y. Van De Burgt, E. Lubberman, E. J. Fuller, S. T. Keene, G. C. Faria, S. Agarwal, M. J. Marinella, A. Alec Talin, and A. Salleo, “A non-volatile organic electrochemical device as a low-voltage artificial synapse for neuromorphic computing,” *Nat. Mater.* **16**, 414–418 (2017).
- ⁵⁵M. C. Mackey and L. Glass, “Oscillation and chaos in physiological control systems,” *Science* **197**, 287–289 (1977).
- ⁵⁶R. Soltani, E. Benmohamed, and H. Ltifi, “Echo state network optimization: A systematic literature review,” *Neural Process. Lett.* **55**, 10251–10285 (2023).
- ⁵⁷J. Doynne Farmer, “Chaotic attractors of an infinite-dimensional dynamical system,” *Physica D* **4**, 366–393 (1982).
- ⁵⁸C. H. López-Caraballo, I. Salfate, J. A. Lazzús, P. Rojas, M. Rivera, and L. Palma-Chilla, “Mackey–Glass noisy chaotic time series prediction by a swarm-optimized neural network,” *J. Phys.: Conf. Ser.* **720**, 012002 (2016).
- ⁵⁹Y. Chen, B. Yang, and J. Dong, “Time-series prediction using a local linear wavelet neural network,” *Neurocomputing* **69**, 449–465 (2006).
- ⁶⁰J. Vesanto, “Using the SOM and local models in time-series prediction using the SOM and local models in time-series prediction,” in *Proceedings of Workshop on Self-Organizing Maps, WSOM’97* (1997), pp. 209–214.
- ⁶¹F. A. Gers, D. Eck, and J. Schmidhuber, *Applying LSTM to Time Series Predictable Through Time-Window Approaches*, Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Springer, 2001), Vol. 2130, pp. 669–676.
- ⁶²B. Ustundag and A. Kulaglic, “High-performance time series prediction with predictive error compensated wavelet neural networks,” *IEEE Access* **8**, 210532–210541 (2020).
- ⁶³T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, 2019), pp. 2623–2631.

⁶⁴A. Howard, Alexjyu, D. Suo, and W. Cukierski, *Google Brain—Ventilator Pressure Prediction* (Kaggle, 2021), see <https://kaggle.com/competitions/ventilator-pressure-prediction>.

⁶⁵T. Xiong, C. Li, X. He, B. Xie, J. Zong, Y. Jiang, W. Ma, F. Wu, J. Fei, P. Yu, and L. Mao, “Neuromorphic functions with a polyelectrolyte-confined fluidic memristor,” *Science* **379**, 156–161 (2023).

⁶⁶Y. Ling, L. Yu, Z. Guo, F. Bian, Y. Wang, X. Wang, Y. Hou, and X. Hou, “Single-pore nanofluidic logic memristor with reconfigurable synaptic functions and designable combinations,” *J. Am. Chem. Soc.* **146**, 14558–14565 (2024).

⁶⁷T. M. Kamsma, M. S. Klop, W. Q. Boon, C. Spitoni, B. Rueckauer, and R. Van Roij, “Chemically regulated conical channel synapse for neuromorphic and sensing applications,” *Phys. Rev. Res.* **7**, 013328 (2025).

⁶⁸M. Zahedi, M. Mayahinia, M. A. Lebdeh, S. Wong, and S. Hamdioui, “Efficient organization of digital periphery to support integer datatype for memristor-based CIM,” in *Proceedings of IEEE Computer Society Annual Symposium on VLSI, ISVLSI 2020—July* (IEEE, 2020), pp. 216–221.

⁶⁹N. E. Fraiman, B. Sabbagh, G. Yossifon, and A. Fish, “Toward an ion-based large-scale integrated circuit: Circuit level design, simulation, and integration of iontronic components,” *arXiv:2412.07784* (2024).