

Potential implementation of reservoir computing models based on magnetic skyrmions

Cite as: AIP Advances 8, 055602 (2018); <https://doi.org/10.1063/1.5006918>

Submitted: 28 September 2017 . Accepted: 12 January 2018 . Published Online: 25 January 2018

George Bourianoff, Daniele Pinna,  Matthias Sitte, and Karin Everschor-Sitte

COLLECTIONS

Paper published as part of the special topic on [62nd Annual Conference on Magnetism and Magnetic Materials, Chemical Physics, Energy, Fluids and Plasmas, Materials Science and Mathematical Physics](#)



View Online



Export Citation



CrossMark

ARTICLES YOU MAY BE INTERESTED IN

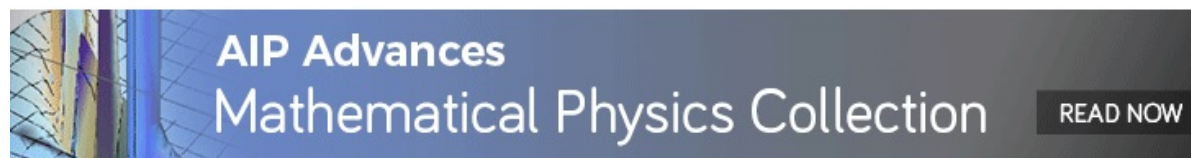
[Perspective: Magnetic skyrmions—Overview of recent progress in an active research field](#)
Journal of Applied Physics **124**, 240901 (2018); <https://doi.org/10.1063/1.5048972>

[The design and verification of MuMax3](#)

AIP Advances **4**, 107133 (2014); <https://doi.org/10.1063/1.4899186>

[Direct writing of room temperature and zero field skyrmion lattices by a scanning local magnetic field](#)

Applied Physics Letters **112**, 132405 (2018); <https://doi.org/10.1063/1.5021172>



Potential implementation of reservoir computing models based on magnetic skyrmions

George Bourianoff,^{1,a} Daniele Pinna,^{2,b} Matthias Sitte,²
 and Karin Everschor-Sitte^{2,c}

¹Intel Corporation, 1300 S. MoPac Exp, Austin, TX 78746, U.S.

²Institute of Physics, Johannes Gutenberg University Mainz, 55099 Mainz, Germany

(Presented 9 November 2017; received 28 September 2017; accepted 12 January 2018;
 published online 25 January 2018)

Reservoir Computing is a type of recursive neural network commonly used for recognizing and predicting spatio-temporal events relying on a complex hierarchy of nested feedback loops to generate a memory functionality. The Reservoir Computing paradigm does not require any knowledge of the reservoir topology or node weights for training purposes and can therefore utilize naturally existing networks formed by a wide variety of physical processes. Most efforts to implement reservoir computing prior to this have focused on utilizing memristor techniques to implement recursive neural networks. This paper examines the potential of magnetic skyrmion fabrics and the complex current patterns which form in them as an attractive physical instantiation for Reservoir Computing. We argue that their nonlinear dynamical interplay resulting from anisotropic magnetoresistance and spin-torque effects allows for an effective and energy efficient nonlinear processing of spatial temporal events with the aim of event recognition and prediction. © 2018 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/1.5006918>

I. INTRODUCTION

A great deal has been written about the end of CMOS scaling, continuation of Moore's Law and the need for alternative models of computing and related technologies. One of the most authoritative discussions on Moore's Law can be found in the "final" International Technology Roadmap for Semiconductors (ITRS)¹ published in 2015 and which had been continuously published since 1991. It predicted that CMOS transistors would quit shrinking in 2021 with the 5 nm node and that a great many technical challenges would need to be met for the 5 nm node to be economically viable. Those technical challenges were primarily related to controlling the economic costs associated with lithography, packaging, testing and the process technology itself. The technical challenges of gate leakage, interconnect power losses and material integration were all considered daunting and drove the economic issues to the point where further scaling was possible but not economically viable. The inescapable conclusion of the final ITRS document is that the scientific research community in collaboration with industry must investigate alternative models of computing including those which appear to be quite radical.

The universe of alternative models of computing is enormous and growing. A nonexclusive list includes² membrane computing, DNA computing, immune computing, quantum computing, neuromorphic computing, in-materio computing, swarm computing, analog computing, chaos/edge-of-chaos computing, computational aspects of dynamics of complex systems, self-organizing systems (e.g. multiagent systems, cellular automata, artificial life), and many others. In evaluating this list

^aGeorge Bourianoff is the corresponding author for discussions related to Reservoir Computing: george.bourianoff@intel.com

^bDaniele Pinna is the corresponding author for discussions related to device physics and numerical implementation: dapinna@uni-mainz.de

^cKarin Everschor-Sitte is the corresponding author for discussions related to skyrmion physics: kaeversc@uni-mainz.de

of alternative computational paradigms, one is reminded of Kroemer's Law³ which states that "the principal applications of any new and innovative technology always have been and will continue to be created by that new technology". One should therefore not only judge new technologies by how they fit in with present applications but specifically by their potential to create innovative applications themselves. Optimally, however, they will satisfy both criteria.

Predicted⁴ and subsequently discovered⁵ over the past two decades, skyrmions are widely regarded as promising candidates for spintronic applications due to their room-temperature stability^{6–11} and mobility at ultra-low current densities.^{12,13} Based on these alone, skyrmions have been proposed for enhancing a spectrum of existing technologies such as racetrack memories,^{14–17} transistors¹⁸ and logic gates.^{19,20} However, the integration of their intrinsic two-dimensional nature also enables their use for radically new technologies which have not been addressed until now.^{21–25}

In this paper we focus on Reservoir Computing (RC) models^{26–31} implemented with self-organizing neural networks in complex magnetic textures.²² The nodes are represented by magnetic skyrmions and the random connectivity by low magnetoresistive pathways in the material. Specifically, we will consider the effects of anisotropic magnetoresistance (AMR) on the conductivity pathways in systems with broken bulk and surface inversion symmetry. For purposes of placement in the taxonomy above, RC models are one category of neuromorphic computing and utilizing complex magnetic systems is an example of in-materio computing.³²

The paper is organised as follows. Sec. II contains a selective review of RC focusing on the requirements placed on the physical implementation system. In Sec. III we first briefly review the skyrmion and micromagnetic literature relevant to RC implementation and then present new simulation results to address the suitability of magnetic skyrmion networks for RC. In particular, we address the complex current patterns which form in skyrmion fabrics resulting from the interplay of anisotropic magnetoresistance and spin-torque effects and comment on optimally tuning the applied voltages to maximize the fabric's susceptibility without actually displacing its magnetic structures. In Sec. IV we conclude how well the capabilities of magnetic substrates meet the needs of RC and delineate areas of future research.

II. RESERVOIR COMPUTING

A Recursive Neural Network (RNN) is a network of nonlinear processing units (similar to neurons in the brain) with weighted connections between them (synapses) characterized by a flow of information that feeds back in loops. These loops imbue the system with a memory functionality derived from the "echo" of previous input sequences persisting over time. Previously, RNN's have been trained to perform specific tasks by algorithmically tuning the weights of each connection in the network. However, the presence of feedback loops makes RNN extremely difficult to train because of bifurcation points and the tendency to exhibit chaotic solutions. As such, the same property responsible for their power is also responsible for their limited applications to date.

Reservoir Computing (RC) models address this problem by treating the recurrent part (the reservoir) differently than both the read-in and read-outs from it.²⁷ This completely eliminates the need to train the large complex reservoir and only train the output weights. Because the latter are taken to not have any loops, they can be trained by straightforward linear regression techniques, i.e. least square algorithms. The usefulness of this framework derives from the reservoir's capacity to project different spatial-temporal events into a sparsely populated high dimensional space where they become easier to recognize and categorize. The ultimate separation of such events is a complex function of the network topology, synaptic delays and the variation in the node response functions including but not limited to their nonlinear properties. This has been shown with mathematical rigor,^{29–31} allowing RC to be successfully demonstrated for spatial temporal event recognition and prediction.

Unlike designed computation, where each device has a specific role, computation in RC networks does not rely on specific devices in specific roles, but is encoded in the collective nonlinear dynamics excited by an applied input signal. This most attractive aspect of RC results from the fact it is unnecessary to have any knowledge related to the structure of the reservoir itself. It is not necessary to know the reservoir's structure, individual node connections, weights, nor any of their nonlinear

characteristics. For this reason RC methods may use “found” networks that may be self-assembled autonomously by a wide variety of physical processes.

Reservoir Computing is typically further divided into two categories, Echo State Networks (ESNs) and Liquid State Networks (LSNs).²⁹ LSNs are characterized by nodes whose state is considered a continuous-time binary value (on/off) whose spiking frequency is determined by the activity of neighboring nodes. Even though they are not actively considered in this work, we note in passing that they are considered more reminiscent of bio-inspired neuromorphic models since they directly emulate the spiking potential generation and propagation observed in biological systems. On the other hand, ESNs are characterized by nodes whose state is defined by continuous values to be updated in discrete time steps depending on the state of nearby nodes. The ESN and LSN approaches alike both provide a similar function – they project the input spatio-temporal events into a high-dimensional space defined by the *state* of the entire reservoir.

A. Echo state networks

A generic ESN is shown schematically in Fig. 1. It consists of an input layer, the reservoir network, and the output layer. The individual nodes in the reservoir network are indicated by light blue dots and the nodes of the input (output) layer are represented by light green (red) dots. The arrows indicate connection paths. The input and output nodes represent a feedforward network (black arrows) whereas the reservoir nodes are in general bidirectional forming both single node and multimode loops (pink arrows). The combination of feedforward and feedback connections results in recursive operation.

Following standard nomenclature,³³ the input signal $u(n)$ is a discrete function of time, $x(n)$ represents the state of the reservoir, e.g. the electric potential at every node, and $y(n)$ represents the output signal. The corresponding weighting functions are denoted as \hat{W}^{in} , \hat{W} and \hat{W}^{out} for the input, the reservoir, and the output. In general, the weights of the reservoir and the input, i.e. \hat{W}^{in} and \hat{W} are time-independent. As already stated, the reservoir weighting function \hat{W} can be unknown. This allows random networks of self assembled reservoirs produced by a wide variety of physical processes to be used (e.g. networks formed by the skyrmion fabrics discussed in this paper).

The evolution of the reservoir state at step n in reaction to its input $u(n)$ and the reservoir state at time step $n - 1$ is defined as:

$$\tilde{x}(n) = \text{sig}(\hat{W}^{\text{in}} \cdot u(n)) + \hat{W} \cdot x(n-1), \quad (1)$$

$$x(n) = \lambda x(n-1) + (1 - \lambda) \tilde{x}(n), \quad (2)$$

$$y(n) = \hat{W}^{\text{out}} \cdot x(n), \quad (3)$$

where \tilde{x} is a provisional state variable, $\text{sig}(\cdot)$ is generically a sigmoidal function biasing the input signal so that the reservoir is excited but not saturated, and λ is a *leakage* parameter characterizing the lossiness of the network’s memory. Note, that for systems without leakage ($\lambda = 0$), one has $x(n) = \tilde{x}(n)$.

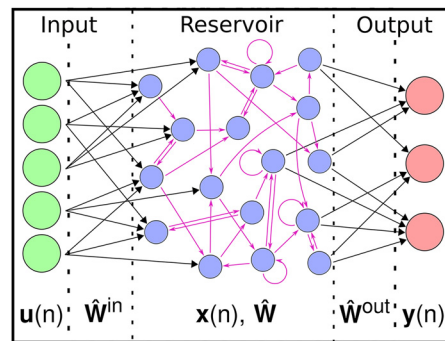


FIG. 1. Sketch of a generic echo state network. The nodes of the network are represented by dots where the color indicates their functionality as input (green) output (red) or internal network nodes (blue). The arrows represent the network connectivity, where black arrows are feedforward only and pink arrows might occur in bidirectionally.

In a natural reservoir system, the specific form of the sigmoidal function is determined by the system's physical properties.

The ultimate goal of the ESN is to classify similar input signals into identical outputs. This is achieved by first training the network on a sample set of pre-classified inputs $\mathbf{u}^{\text{train}}(n)$. Denoting by $\mathbf{y}^{\text{train}}(n)$ and $\mathbf{y}^{\text{target}}(n)$ the system's response to the training inputs and the desired target output respectively, we can calculate the error E between them, averaged over all N_y output nodes and T time steps:

$$E(\mathbf{y}^{\text{train}}, \mathbf{y}^{\text{target}}) = \frac{1}{N_y} \sum_{i=1}^{N_y} \sqrt{\frac{1}{T} \sum_{n=1}^T (y_i^{\text{train}}(n) - y_i^{\text{target}}(n))^2}. \quad (4)$$

The training operation is defined by finding the one-dimensional scalar array of output weights $\hat{\mathbf{W}}^{\text{out}}$ that minimizes the error function E . While the specific algorithms implementing the minimization task are beyond the scope of this paper, we would like to emphasize again that Eq. (4) only requires the output weights to be modified. This is inherently more efficient and robust than the methods required for training full RNNs. Furthermore, since training of the output weights does not modify the reservoir in any way, different features of the reservoir can be searched for simultaneously by setting multiple output arrays in parallel. This makes RC well suited for sensor fusion type applications.^{34,35} It should be noted that since the training involves solving a straightforward linear regression equation, the overhead associated with training each feature can be very small relative to the amount of processing that can be done with the system.

To summarize, the reservoir requires a few qualitative key characteristics to properly function in an Echo State computing system:

- It must have a short term memory, i.e. be recursively connected and/or use nodes with internal memory. As discussed, this guarantees the reservoir's sensitivity to the input's temporal correlations.
- The dimensionality of the reservoir's state space must be much larger than the input array. This corresponds basically to the number of nodes in the reservoir. The larger the reservoir, the greater the separation and probability of the linear classifier being able to successfully recognise specific events.
- The response of the reservoir must be a nonlinear function of its inputs and previous states. The stronger the nonlinearity effects the larger the separation of events in the reservoir's phase space facilitating classification during training.
- The reservoir's echo state time, defined as the timescale beyond which the reservoir dynamics effectively lose all initialization information, must be much larger than the largest relevant temporal correlations in the input signals. The reservoir's echo state time can be tuned by varying the leakage parameter λ .

These reservoir properties (and the parameters controlling them) determine the performance of the recursive network for spatial temporal event recognition and prediction. For computational materials like the magnetic textures considered in this paper, these (or an equivalent set) must be ultimately deduced by experimental measurements and tuned for optimal performance by leveraging physical insight over the materials being used. In the following section, we will proceed by reviewing the literature of magnetic skyrmions relevant to a potential RC implementation.

III. MAGNETIC SKYRMIONS AND "SKYRMION FABRICS"

Magnetic skyrmions are nontrivial topological magnetic textures that were predicted more than two decades ago.⁴ Experimentally, they were first discovered in the form of a skyrmion lattice in 2009⁵ and later also as isolated magnetic textures.³⁶ Their presence has been observed in many device-relevant materials and their properties have been extensively summarized in several reviews.^{37–40} Skyrmions are regarded as promising candidates for spintronic applications due to their mobility when driven by ultra-low currents^{12,13} and their room temperature stability.^{6–11} Particularly, the skyrmion racetrack memory has been a significant driver for intensively studying individual skyrmions.^{14–17}

Not much attention has, however, been given to applications involving intermediate skyrmion phases known as “skyrmion fabrics”.²² These are phases that interpolate between single skyrmions, skyrmion crystals and magnetic domain walls⁴¹ (examples shown in the top panels of Fig. 2). In the past, skyrmion fabrics have been studied only to observe how the different phases contribute to transitions between them.⁴²

We claim that skyrmion fabrics can provide a good basis for RC reservoirs in light of their random phase structure. The input signals can be realized via voltage patterns applied directly to the magnetic texture through various nanocontacts. Magnetoresistive effects^{43–45} such as the anisotropic magnetoresistance (AMR) will then guarantee that a certain magnetic texture will result in a unique corresponding current pattern throughout the reservoir (shown in middle panels of Fig. 2). As the input signal is varied, the magnetic texture will deform due to the interplay of AMR-mediated spin-torques and local pinning. These deformations will take place on a timescale that is slower than that taken by the current density pattern to adjust to the magnetic texture. The delay between the two will guarantee both the nonlinear resistance response of the magnetic material as well as the echoing of the spatial-temporal correlations in the dynamical response of the magnetic texture. Note that, for this to work properly, the skyrmion fabric must relax back as close as possible to its initial state upon switching off the voltage inputs. This in turn will guarantee that the reservoir operates reproducibly over separate runs. As such, optimal operation of the reservoir will require to tune voltage intensities and input signal frequencies with the objective to maximize the fabric’s susceptibility without actually displacing its constituting skyrmions and domain walls. Overall, the random skyrmion structure and corresponding current pattern will effectively model the reservoir node and weight structure discussed in the previous section.

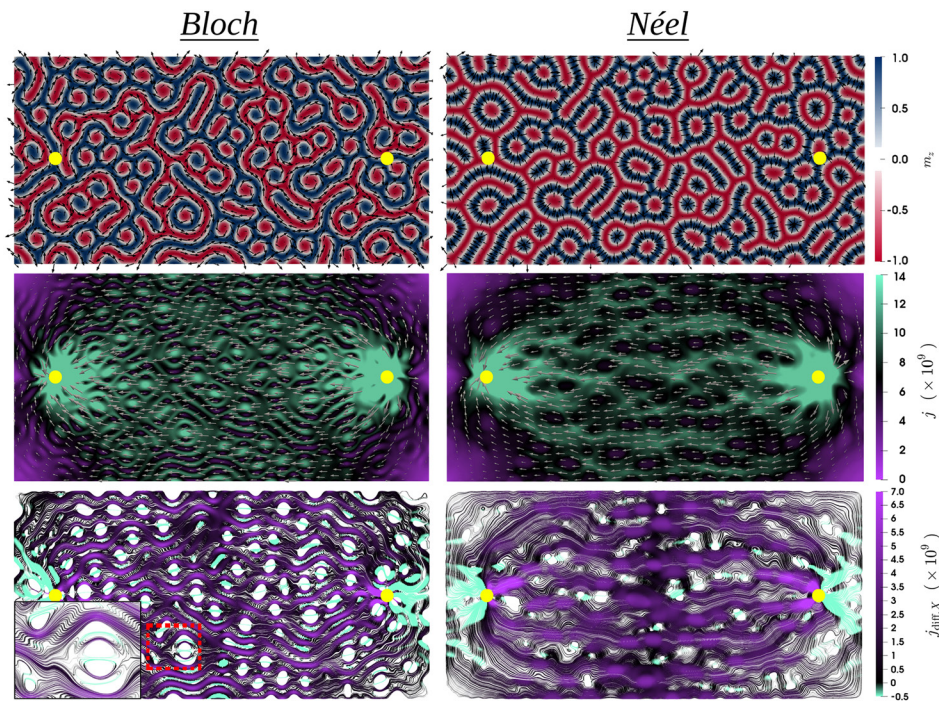


FIG. 2. Bloch (left column) and Néel (right column) skyrmion fabrics, where a voltage is applied in-between the contacts (yellow dots). Top row: Magnetization profiles. Color code and black arrows denote the out-of-plane and in-plane components respectively. Middle row: Current pathways through corresponding skyrmion fabrics. Color code and gray arrows denote current magnitude and direction respectively. Bottom row: Differential current flow lines showing the skyrmion-mediated AMR effects obtained by subtracting the trivial out-of-plane ferromagnetic current flow from that of the skyrmion fabric. Color code highlights regions where skyrmions enhance (green) and reduce (purple) flow along the negative x direction revealing current backflows in the AMR-dominated regime. The inset shows a close-up view of the area enclosed by the red dashed rectangle. The parameters used for these simulations are: $\alpha = 0.5$, $M_s = 4.9 \times 10^5 \text{ A m}^{-1}$, $\sigma_0 = 5 \times 10^6 \text{ S m}^{-1}$, $a = 1$, $U = 1 \times 10^{-3} \text{ V}$, $A_{ex} = 6 \times 10^{-12} \text{ J m}^{-1}$, $K_u = 1.3 \times 10^6 \text{ J m}^{-3}$, and $D_{B/N} = 3 \times 10^{-3} \text{ J m}^{-2}$.

The magnetization profiles and the current paths shown in Fig. 2 have been obtained by micromagnetic simulations using Micromagnum⁴⁶ and selfwritten software extensions where the magnetization dynamics and the current paths have been computed selfconsistently.²² The magnetization dynamics are given by the Landau-Lifshitz-Gilbert (LLG) equation for the magnetization direction $\mathbf{m} = \mathbf{M}/M_s$ with spin-transfer-torque effects:^{47,48}

$$(\partial_t + \xi \mathbf{j}[U, \mathbf{m}] \cdot \nabla) \mathbf{m} = -\gamma \mathbf{m} \times \mathbf{B}_{\text{eff}} + \alpha \mathbf{m} \times (\partial_t + \frac{\beta}{\alpha} \xi \mathbf{j}[U, \mathbf{m}] \cdot \nabla) \mathbf{m}. \quad (5)$$

Here, M_s is the saturation magnetization, the factor $\xi = P\mu_B/(eM_s)$ contains the polarization P the electron charge e and the Bohr magneton μ_B . The effective magnetic field is given by $\mathbf{B}_{\text{eff}} = -M_s^{-1}(\delta F[\mathbf{m}]/\delta \mathbf{m})$, where the micromagnetic free energy comprising exchange, anisotropy and dipolar interactions is:

$$F = \int \left(A_{\text{ex}}(\nabla \mathbf{m})^2 + K_u(1 - m_z^2) - \frac{\mu_0}{2} M_s \mathbf{m} \cdot \mathbf{H}_d(\mathbf{m}) \right) dV + F_{\text{DMI}}^{\text{B/N}}[\mathbf{m}], \quad (6)$$

and $F_{\text{DMI}}^{\text{B}} = \int D_B \mathbf{m} \cdot (\nabla \times \mathbf{m}) dV$ describes Bloch and $F_{\text{DMI}}^{\text{N}} = \int D_N \mathbf{m} \cdot [(\mathbf{z} \times \nabla) \times \mathbf{m}] dV$ Néel DMI.^{49–51} Note that the current density \mathbf{j} is a function of the applied voltage and the local magnetization. Since current density relaxation happens on a much faster time scale than the magnetization dynamics, it can be calculated self-consistently based on the AMR effect⁵² through $\mathbf{j}[U, \mathbf{m}] = -\boldsymbol{\sigma}[\mathbf{m}] \cdot \mathbf{E}[U]$. Here the electric field induced by the applied voltage is calculated by solving the Poisson equation $\mathbf{E} = -\nabla \Phi$ with boundary conditions $\Phi|_{c1} = -\Phi|_{c2} = U$ at the two contacts, and the conductivity tensor $\boldsymbol{\sigma}[\mathbf{m}] = \frac{1}{\rho_{\perp}} \mathbb{1} + (\frac{1}{\rho_{\parallel}} - \frac{1}{\rho_{\perp}}) \mathbf{m} \otimes \mathbf{m}$ varies with the local magnetization. We denote by ρ_{\perp} (ρ_{\parallel}) the current resistivities for flows perpendicular (parallel) to the magnetization direction. Based on these we define $\sigma_0 = (1/\rho_{\parallel} + 2/\rho_{\perp})/3$, and the AMR ratio $a = \frac{2(\rho_{\parallel} - \rho_{\perp})}{\rho_{\parallel} + \rho_{\perp}}$.^{22,52}

Previous work has focused on details of current paths through magnetic nano-ribbons in the presence of single magnetic Bloch and Néel skyrmions.²² As a result of the AMR effect, Néel skyrmions show a tendency to deflect current flow lines tangentially around their centers while

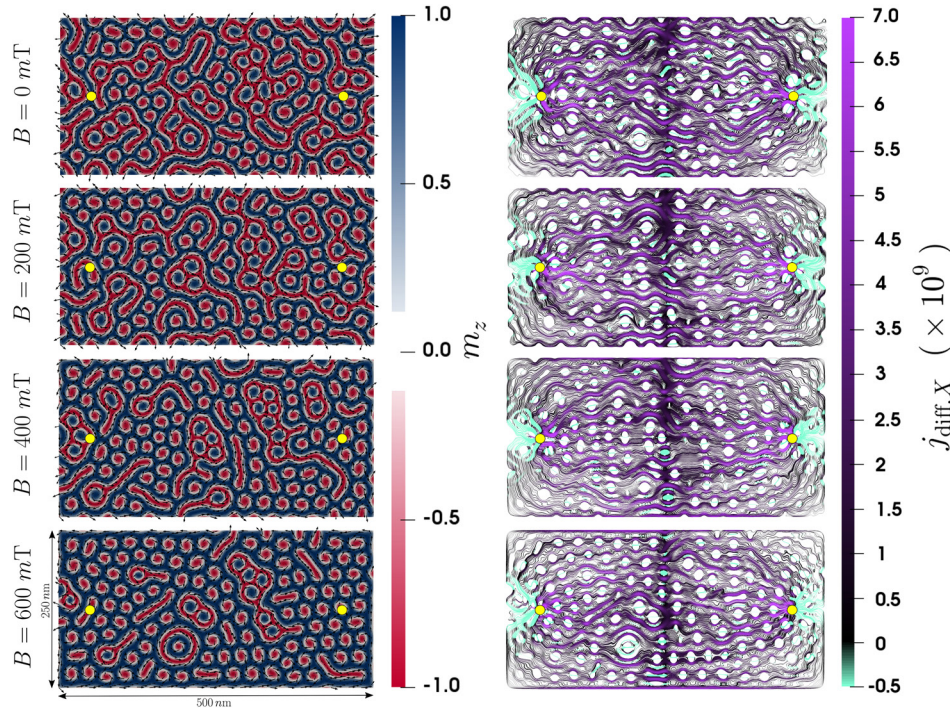


FIG. 3. Magnetic texture as a function of applied out-of-plane magnetic field intensity (left column) and corresponding differential current flows (right column).

Bloch skyrmions favor current flows through their centers. Furthermore, single skyrmions were shown to exhibit non-linear current-voltage characteristics due to the interplay of magnetoresistive effects and pinning.

This work expands on previous results by exploring the effect of Néel and Bloch skyrmion fabrics on their resulting AMR-mediated current flow. In Fig. 2, we show examples of Bloch (left column) and Néel (right column) skyrmion fabrics. Their complex magnetic texture (top row) reflects on the total current flow between two contacts where a voltage difference is applied (middle row). To isolate the pure AMR effect of the skyrmion fabric we subtract the current response of a trivial out-of-plane ferromagnetic state and plot the resulting differential current flow (bottom row). We find that this artificially constructed AMR-dominated regime exhibits current backflows reminiscent of the recursive connectivity required in a RC network.

Due to the tunable properties of magnetic skyrmions, the skyrmion fabric can be tweaked by application of static magnetic fields. These alter the density of skyrmions, thus tuning the effective node density throughout the sample. Figure 3 demonstrates this by showing how a Bloch magnetic texture and its resulting differential current flows both respond to variations of out-of-plane applied magnetic fields.

IV. CONCLUSIONS

We have argued that skyrmion fabrics embedded in broken inversion symmetry magnetic substrates are potentially attractive options for implementing Echo State (ES) recognition and prediction systems. The principle result from this paper shows how skyrmion fabrics induce a strongly perturbed current flow through the magnetic texture as compared to the expected current flow through the homogeneously magnetized state. We further show that such a perturbed current flow creates conduction paths similar to those required for Echo State Networks. Leveraging results from previous work on nonlinear I-V characteristics of individual skyrmions,²² we argue that skyrmion fabrics appear to meet the basic requirements of an Echo State Reservoir computing system. It should be pointed out that the functionality of the proposed system is not directly related to the magnitude of the nonlinear effect. Nonetheless, the total intensity of the AMR effect scales with the net magnetic distortion present in the magnetic sample. As such, complex random fabrics will allow for a more robust operating signal than the single isolated skyrmions previously studied.²² We would like to emphasize that the information processing capabilities inherent to such magnetic materials derives from the dynamic interplay of current flow with the complex magnetic texture.⁵³

The present calculation does not address the influence of thermal noise or potential grain structures in the magnetic material. Whereas our study is based solely on the anisotropic magneto-resistive (AMR) effect, other similar magnetization-modulated resistance effects could be included to tune and enhance our results. Furthermore, a properly functioning reservoir must guarantee a basis state across operational cycles. This implies that the skyrmion fabric must restore to its initial state whenever the driving voltages are switched off. Due to the dynamical character of magnetic skyrmions and domain walls, this may not be guaranteed if applied voltage intensities go beyond the threshold voltage for motion. Future work will focus on exploring how material inhomogeneities and impurities can be leveraged to tune for this threshold voltage deterministically.

The size scales of skyrmion fabrics are orders of magnitude smaller than other proposed implementations of reservoirs for ES networks, like memristor or optical networks. Furthermore, the characteristics of the skyrmion fabric (skyrmion density/size, domain wall width, etc ...) can be altered by tuning the material properties and/or applying magnetic fields. These effectively tune both the dimensionality and net nonlinearity of the reservoir's dynamics, thus determining its performance.

ACKNOWLEDGMENTS

We acknowledge discussions with Kai Litzius, Diana Prychynenko and Jairo Sinova. G. B. appreciates support and useful discussions with Narayan Srinivasa of Intel. We acknowledge the funding from the German Research Foundation (DFG) under the Project No. EV 196/2-1, the Alexander von Humboldt Foundation and the Transregional Collaborative Research Center (SFB/TRR) 173 Spin+X.

- ¹ ITRS, Int. Technol. roadmap Semicond. 79 (2015).
- ² A. Adamatzky, *Advances in Unconventional Computing*, 23 (Springer International Publishing, 2017).
- ³ C. K. Maiti, *Selected Works of Professor Herbert Kroemer, P97 and P103* (World Scientific, 2008).
- ⁴ A. N. Bogdanov and D. Yablonskii, Zh. Eksp. Teor. Fiz. **95**, 178 (1989).
- ⁵ S. Mühlbauer, B. Binz, F. Jonietz, C. Pfleiderer, A. Rosch, A. Neubauer, R. Georgii, and P. Böni, *Science* **323**, 1381 (2011).
- ⁶ X. Z. Yu, N. Kanazawa, Y. Onose, K. Kimoto, W. Z. Zhang, S. Ishiwata, Y. Matsui, and Y. Tokura, *Nat. Mater.* **10**, 106 (2011).
- ⁷ X. Yu, N. Kanazawa, W. Zhang, T. Nagai, T. Hara, K. Kimoto, Y. Matsui, Y. Onose, and Y. Tokura, *Nat. Commun.* **3**, 988 (2012).
- ⁸ D. A. Gilbert, B. B. Maranville, A. L. Balk, B. J. Kirby, P. Fischer, D. T. Pierce, J. Unguris, J. A. Borchers, and K. Liu, *Nat. Commun.* **6**, 8462 (2015).
- ⁹ S. Woo, K. Litzius, B. Krüger, M.-Y. Im, L. Caretta, K. Richter, M. Mann, A. Krone, R. M. Reeve, M. Weigand, P. Agrawal, I. Lemeshe, M.-A. Mawass, P. Fischer, M. Kläui, and G. S. D. Beach, *Nat. Mater.* **15**, 501 (2016).
- ¹⁰ O. Boulle, J. Vogel, H. Yang, S. Pizzini, D. de Souza Chaves, A. Locatelli, T. O. Menteş, A. Sala, L. D. Buda-Prejbeanu, O. Klein, M. Belmeguenai, Y. Roussigné, A. Stashkevich, S. M. Chérif, L. Aballe, M. Foerster, M. Chshiev, S. Auffret, I. M. Miron, and G. Gaudin, *Nat. Nanotechnol.* **11**, 449 (2016).
- ¹¹ R. Tomasello, M. Ricci, P. Burrascano, V. Puliafito, M. Carpentieri, and G. Finocchio, *AIP Adv.* **7**, 056022 (2017).
- ¹² F. Jonietz, S. Mühlbauer, C. Pfleiderer, A. Neubauer, W. Münzer, A. Bauer, T. Adams, R. Georgii, P. Böni, R. A. Duine, K. Everschor, M. Garst, and A. Rosch, *Science* **330**, 1648 (2010).
- ¹³ T. Schulz, R. Ritz, A. Bauer, M. Halder, M. Wagner, C. Franz, C. Pfleiderer, K. Everschor, M. Garst, and A. Rosch, *Nat. Phys.* **8**, 301 (2012).
- ¹⁴ A. Fert, V. Cros, and J. Sampaio, *Nat. Nanotechnol.* **8**, 152 (2013).
- ¹⁵ R. Tomasello, E. Martinez, R. Zivieri, L. Torres, M. Carpentieri, and G. Finocchio, *Sci. Rep.* **4**, 6784 (2014).
- ¹⁶ X. Zhang, G. P. Zhao, H. Fangohr, J. P. Liu, W. X. Xia, J. Xia, and F. J. Morvan, *Sci. Rep.* **5**, 7643 (2015).
- ¹⁷ J. Müller, *New J. Phys.* **19**, 025002 (2017).
- ¹⁸ X. Zhang, Y. Zhou, M. Ezawa, G. P. Zhao, and W. Zhao, *Sci. Rep.* **5**, 11369 (2015).
- ¹⁹ Y. Zhou and M. Ezawa, *Nat. Commun.* **5**, 4652 (2014).
- ²⁰ X. Zhang, M. Ezawa, and Y. Zhou, *Sci. Rep.* **5**, 9400 (2015).
- ²¹ D. Pinna, F. A. Araujo, J.-V. Kim, V. Cros, D. Querlioz, P. Bessiere, J. Droulez, and J. Grollier (2017), [arXiv:1701.07750](https://arxiv.org/abs/1701.07750).
- ²² D. Prychynenko, M. Sitté, K. Litzius, B. Krüger, G. Burianoff, M. Kläui, J. Sinova, and K. Everschor-Sitté (2017), [arXiv:1702.04298](https://arxiv.org/abs/1702.04298).
- ²³ Z. He and D. Fan (2017), [arXiv:1705.02995](https://arxiv.org/abs/1705.02995).
- ²⁴ Z. He and D. Fan, in *Des. Autom. Test Eur. Conf. Exhib. (DATE)*, 2017 (IEEE, 2017) pp. 350–355.
- ²⁵ S. Li, W. Zhang, Y. Huang, X. Zhang, Y. Zhou, and W. Zhao, *Nanotechnology* **28**, 31LT01 (2017).
- ²⁶ M. Lukosevicius, H. Jaeger, and M. Luko, *Comput. Sci. Rev.* **3**, 127 (2009).
- ²⁷ M. Lukosevicius, H. Jaeger, and B. Schrauwen, *KI - Künstliche Intelligenz* **26**, 365 (2012).
- ²⁸ J. Burger, A. Goudarzi, D. Stefanovic, and C. Teuscher, *AIMS Mater. Sci.* **2**, 530 (2015).
- ²⁹ M. Lukosevicius, PhD thesis, 136 (2012).
- ³⁰ D. Verstraeten, B. Schrauwen, M. D’Haene, and D. Stroobandt, *Neural Networks* **20**, 391 (2007).
- ³¹ W. Maass, T. Natschlager, and H. Markram, *Neural Comput.* **14**, 2531 (2002).
- ³² M. Dale, S. Miller, F. Julian, and Stepney, “Reservoir computing as a model for in-materio computing,” in *Advances in Unconventional Computing: Volume 1: Theory*, edited by A. Adamatzky (Springer International Publishing, Cham, 2017), pp. 533–571.
- ³³ M. Lukosevicius, “A practical guide to applying echo state networks,” in *Neural Networks: Tricks of the Trade: Second Edition*, edited by G. Montavon, K.-R. Orr, B. Genevieve, and Müller (Springer Berlin Heidelberg, Berlin, Heidelberg, 2012) pp. 659–686.
- ³⁴ F. Palumbo, P. Barsocchi, C. Gallicchio, S. Chessa, and A. Micheli, “Multisensor data fusion for activity recognition based on reservoir computing,” in *Evaluating AAL Systems Through Competitive Benchmarking: International Competitions and Final Workshop, EvAAL 2013, July and September 2013. Proceedings*, edited by J. A. Botía, J. A. Álvarez-García, K. Fujinami, P. Barsocchi, and T. Riedel (Springer Berlin Heidelberg, Berlin, Heidelberg, 2013) pp. 24–35.
- ³⁵ Z. Konkoli, *Int. J. Parallel, Emergent Distrib. Syst.* **5760**, 1 (2016).
- ³⁶ N. Romming, C. Hanneken, M. Menzel, J. E. Bickel, B. Wolter, K. von Bergmann, A. Kubetzka, and R. Wiesendanger, *Science* **341**, 636 (2013).
- ³⁷ N. Nagaosa and Y. Tokura, *Nat. Nanotechnol.* **8**, 899 (2013).
- ³⁸ G. Finocchio, F. Büttner, R. Tomasello, M. Carpentieri, and M. Kläui, *J. Phys. D: Appl. Phys.* **49**, 423001 (2016).
- ³⁹ A. Fert, N. Reyren, and V. Cros, *Nat. Rev. Mater.* **2**, 17031 (2017).
- ⁴⁰ W. Jiang, G. Chen, K. Liu, J. Zang, S. G. te Velthuis, and A. Hoffmann, *Phys. Rep.* (2017).
- ⁴¹ C. Y. You and N. H. Kim, *Curr. Appl. Phys.* **15**, 298 (2015).
- ⁴² P. Milde, D. Köhler, J. Seidel, L. M. Eng, A. Bauer, A. Chacon, J. Kindervater, S. Mühlbauer, C. Pfleiderer, S. Buhrandt, C. Schütte, and A. Rosch, *Science* **340**, 1076 (2013).
- ⁴³ T. McGuire and R. Potter, *IEEE Trans. Magn.* **11**, 1018 (1975).
- ⁴⁴ C. Hanneken, F. Otte, A. Kubetzka, B. Dupé, N. Romming, K. von Bergmann, R. Wiesendanger, and S. Heinze, *Nat. Nanotechnol.* **10**, 1039 (2015).
- ⁴⁵ A. Kubetzka, C. Hanneken, R. Wiesendanger, and K. von Bergmann, *Phys. Rev. B* **95**, 104433 (2017).
- ⁴⁶ “MicroMagnum—Fast Micromagnetic Simulator for Computations on CPU and GPU,” available online at micromagnum.informatik.uni-hamburg.de.
- ⁴⁷ L. Berger, *Phys. Rev. B* **54**, 9353 (1996).
- ⁴⁸ J. Slonczewski, *J. Magn. Magn. Mater.* **159**, L1 (1996).

⁴⁹ I. Dzyaloshinsky, *J. Phys. Chem. Solids* **4**, 241 (1958).

⁵⁰ T. Moriya, *Phys. Rev.* **120**, 91 (1960).

⁵¹ A. Thiaville, S. Rohart, É. Jué, V. Cros, and A. Fert, *EPL (Europhysics Lett.)* **100**, 57002 (2012).

⁵² B. Krüger, *Current-Driven Magnetization Dynamics: Analytical Modeling and Numerical Simulation*, Ph.D. thesis (2011).

⁵³ A. Smolensky, *Information Processing in Dynamical Systems: Foundations of Harmony Theory, Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, Volume I, Chapter 6 (MIT Press/Bradford Books).