Reservoir Computing: Quo Vadis?

Alireza Goudarzi
Department of Computer Science
University of New Mexico
1 University Blvd
Albuquerque, NM 87131
alirezag@cs.unm.edu

Christof Teuscher
Department of Electrical and Computer
Engineering
Portland State University
1900 SW 4th Ave
Portland, OR 97201
teuscher@pdx.edu

ABSTRACT

Reservoir Computing (RC) is an umbrella term for adaptive computational paradigms that rely on an excitable dynamical system, also called the “reservoir.” The paradigms have been shown to be particularly promising for temporal signal processing. RC was also explored as a potential candidate for emerging nanoscale architectures. In this article we reflect on the current state of RC and muse about its future. In particular, we propose a set of open problems that we think need to be addressed in order to make RC more mainstream.

CCS Concepts

•General and reference → Surveys and overviews;

Keywords

reservoir, computing, future, problems, questions

1. INTRODUCTION

Reservoir Computing (RC) refers to a number of different machine learning techniques that use the high-dimensional transient dynamics of an excitable system, the *reservoir*, to process spatio-temporal input signals and transform them into useful outputs by means of a linear readout layer. The proper weights of the linear readout layer can either be trained by a gradient descent method, closed-form linear regression, or determined analytically in certain cases [23]. The emphasis on the dynamical regime of the reservoir makes its microstructure irrelevant to the information processing as long as the reservoir produces a fading memory of the input. As a consequence, RC has gained popularity in recent years among researchers that aim to build computing systems from unconventional physical substrates and devices, such as photonic systems [66], self-assembled nanowires [56], molecular, and chemical systems [22, 49].

RC is a special instance of *intrinsic computation*. Intrinsic computation is commonly defined as making use of the spa-

tial and temporal dynamics a system has “intrinsically.” It is important to note that “[i]ntrinsic computation makes no reference to utility” [13]. On the other end of the spectrum we have “designed computation,” that makes information processing in a specific system “useful.” In traditional CMOS technology, one relies on designed computation only. With the advent of emerging devices, such as memristors [61], that are expected to behave in time-dependent nonlinear ways beyond a simple switching behavior and will exhibit significant physical variation, we cannot rely on that principle anymore. It is generally believed that the technological promise of harnessing intrinsic computation beyond the digital realm has enormous potential for cheaper, faster, more robust and more energy-efficient information processing technology, as for example our own studies have confirmed [8].

The theoretical basis of how and why RC works are well understood [19, 20, 23, 67]. RC works on two operating principles:

1. a conversion of spatio-temporal information into a purely spatial representation: the history of the spatio-temporal inputs will be encoded in the instantaneous state of the reservoir, which therefore has a spatial representation;
2. short-term memory: the memory of the inputs will fade away from the reservoir state over time.

An ideal reservoir has to satisfy two conditions for these principles to be in place. First, the reservoir degrees of freedom should be as close as possible to orthogonal. In other words, the states of the reservoir should be linearly independent. Failure to satisfy this condition causes reservoir dimensions to collapse, which means the reservoir effectively operates in a lower dimension than the number of its degrees of freedom [14, 19]. The second condition is that the reservoir must operate near a suitable dynamical regime, which is sometimes called the *edge-of-chaos* [37]. In simple terms, this is a regime where perturbations stay constant in size as they spread throughout the reservoir, i.e., they neither die out nor get amplified. If this condition is not met, the information in the reservoir either decays too quickly for the readout layer to be used or it grows and interferes destructively, which results in decoherent reservoir states that make signal extraction difficult [19].

Interest in RC has grown over recent years in three main research areas: (1) neuroscience and cognitive science [7], (2) machine learning [39], and (3) unconventional computing [21]. In its earliest form, RC was proposed as a simplified model of corticostriatal plasticity [17]. More recently RC has been used to explain higher-order cognitive functions as

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

NanoCom’16 Sep 28–30, 2016, New York, NY, USA

© 2016 ACM. ISBN 978-1-4503-4061-8/16/09...\$15.00

DOI: <http://dx.doi.org/10.1145/2967446.2967448>

well [27]. In neuroscience, RC is frequently used to explain short-term memory and its interaction with various cognitive processes. RC is also used as a model for cortical microcircuits [40–42]. In machine learning, it is used to lower the training complexity of recurrent neural networks. Despite application of RC to many engineering problems [39], its current limitation is the lack of scalability to solve larger problems, i.e., problems with high dimensions and large amounts of data, such as real-time video processing. Finally, in unconventional computing, RC is used as a computational approach for physical substrates where the precise structure is hard to control, or cannot be controlled at all. Such substrates may, for example, be fabricated through nanoscale or molecular self-assembly [16].

To allow a more concise discussion in this paper, we adopt the following abstract representation of an RC device operating in discrete time:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t), \text{ and} \quad (1)$$

$$\mathbf{y}_t = g(\mathbf{x}_t), \quad (2)$$

where $\mathbf{u}_t$ and $\mathbf{x}_t$ are multidimensional vectors of the input and reservoir state at time t , f is an operator that governs the evolution of the reservoir state based on its history and input(s). To keep the model general, we use f to encapsulate both the structure and the dynamics of the reservoir. The multi-dimensional output at time t is denoted by $\mathbf{y}_t$. g is the output function that takes in the observable reservoir degrees-of-freedom and generates the desired output(s). Note that f is fixed and only g is adaptive. In general, the number of observable states is not the same as the dimensionality of $\mathbf{x}$. This definition carries over to continuous time as well.

Theoretical research in RC aims to shed light on the computational properties with respect to the choices of f and g . Experimental research, on the other hand, aims to find what choice of f and g are suitable for a given application. Despite significant progress in RC research, a number of important theoretical and experimental questions remain unaddressed. In the following, we will list some key questions that we believe would present fruitful future research directions to further advance the field.

2. THEORETICAL CHALLENGES

2.1 Computational Power and Complexity

From a computational perspective, an important question is the computational power of RC, i.e., what kind of algorithms can be expressed. That includes the question of universality.

- **Q1: What classes of problems can RC solve efficiently?** *Recurrent Neural Networks* (RNNs) are universal models that in principle can achieve everything RC does. However, for certain tasks, such as chaotic time series prediction, RC achieves good performance with less training than a regular RNN [33]. But for other tasks, such as large-scale language modeling, RC is not competitive with state-of-the-art machine learning techniques [46, 64]. And depending on the size and the parameters of the network, certain tasks may not be solvable at all. RC lacks a formal framework that would help to describe the computational expressiveness and complexity as a function of

the common parameters. In other words, what types of networks can recognize regular languages, context-free languages, etc. And what is the computational and the learning (time and space) complexity of these networks? Having answers to these questions would allow the community to better describe what classes of problems RC is good for. Currently, these decisions are mostly made on an trial and error basis.

More specifically, compared to RNN and *Long Short-Term Memory* (LSTM) networks [29] trained with *Back-propagation Through Time* (BPTT) [3], Hessian-free optimization [45], or linear approximations [51], what is the nature of the task-dependent information represented in the error signal that is sent back to the network that a memoryless readout layer in RC cannot provide? Can this limitation be overcome by approaches that transfer learning from readout connections to the reservoir [59, 62]? Or by methods that adjust macroscopic properties of the reservoir to match the memory requirements of the task [15, 50, 63]? And if such approaches are effective in improving RC, will they suffer from similar challenges, such as vanishing or exploding gradients [52], as the other training algorithms do?

- **Q2: Can a specific RC perform universal computation?** While universal computation is interesting from a theoretical point of view, it is mostly irrelevant for real applications, where universal computation is rarely required. Siegelmann and Sontag proved the computational universality of recurrent neural networks [55]. It was later shown that even a system with a single feedback connection is universal [30]. However, both of these systems require a precise design of the network structure. RCs with feedback [33] have been studied since the introduction of *Echo State Networks* (ESN) [31], yet their computational power is not yet understood. In particular, could a certain type of RC compute recursively enumerable languages, or in other words, could they become a universal computer? In addition, if such a system is universal, how would one come up with a generic training algorithm that would allow to effectively train the readout layer for any given task, specifically for tasks with arbitrary memory depth?

2.2 Reservoir Structure and Function

Another important area in RC that needs significant research is the relationship between the structure and the function of the reservoir. While previous work has investigated some aspects (e.g., [1, 6, 10, 11, 37, 57]) of the influence of different reservoir and readout layer architectures on task performance, there does not exist a systematic framework.

- **Q3: Can RC benefit from more complex readout layers?** Traditionally, RC uses a linear readout layer trained with linear regression or stochastic gradient descent to minimize a squared-error objective function. Other types of readout layers, such as Boltzmann machines [53] and Volterra filters [6] have been used to extend the power of RC. Yet, from a theoretical standpoint, it is unclear how big the impact is on task performance. A more systematic investigation with comprehensive comparisons is needed. In addition, using more

complex readout layers increases the training complexity, which takes away one of the benefits of RC.

- **Q4: Can we find better measures of the memory and information processing capabilities?** It is necessary to have reliable and robust measures to quantify RC performance on a given or an entire class of tasks. Today, the most commonly used measures include the memory capacity [32, 67], the nonlinear memory capacity [14], and measures of the dynamical regime [10] of a reservoir. While these measures shed light on some of the reservoir properties, it is nontrivial to relate them to the expected performance. For example, for linear networks, the notion of memory capacity has recently been extended to predict the expected error on a task and to provide a worst-case error bound [23]. However, the presented framework only applies to linear reservoirs. Without a suitable performance measure and a way for tuning the reservoir for tasks, parameter optimization can simply be too computationally expensive.
- **Q5: What is the role of heterogeneous or disordered structure in RC?** *Echo State Networks* (ESN) [31] and *Liquid State Machines* (LSM) [43] are the earliest incarnations of RC, both of which feature randomly generated recurrent neural networks that make up the reservoir. Subsequent theoretical analysis has shown that the same performance can be obtained with reservoirs that have a regular structure. Thus, the benefit of random or “complex” reservoir structures, if there is any at all, remains an open question. The role of complex RC structure is of particular interest to the unconventional computing community that tries to use RC to build computational devices from unstructured networks of simple elements, for example obtained through self-assembling nanowires or memristors [16, 36].
- **Q6: Can hierarchical RCs help us to solve more complex tasks?** Digital systems build on the modularity principle. A set of transistors forms a logic gate, a set of logic gates forms a circuit, and circuits are used to build architectures. Most RC systems are monolithic, which limits their computational capabilities because such networks are not easily scalable by simply increasing the number of nodes within a given reservoir. The idea of hierarchical reservoirs is not new. Triefenbach *et al.* [65], for example, used hierarchical reservoirs for phoneme recognition. More recently, Burger *et al.* [9] demonstrated that hierarchical reservoirs can outperform monolithic reservoir systems. For certain tasks, hierarchical reservoirs outperformed single reservoirs by at least 20%.

Composability and hierarchy are key to building large(er)-scale intrinsic computing systems that scale to tasks of significant complexity. How can we build hierarchical and scalable RC systems, similar to what we do with CMOS circuits? How could we ensure composability of the building blocks? How would we train such systems efficiently? And what network topologies/hierarchies would lead to optimal performance?

3. PHYSICAL REALIZATION

RC can be implemented using any physical system that offers interesting enough dynamics for information processing. One of the earliest demonstrations of this was voice recognition using wave patterns on the surface of a water tank [18]. Recent advances in materials science and nanotechnology have made it possible to create computational substrates out of various unconventional devices [8, 22, 25, 34, 48, 56, 66]. Many such emerging devices, however, cannot be fabricated as precisely and reliably as one wishes. The resulting structures may be partially or completely random, heterogeneous, and unreliable [4, 5, 16, 28, 60, 68], which typically benefits RC because such systems offer richer dynamics. Another motivation for doing signal processing directly with physical devices is the potentially simplified and more natural interfacing with such systems [54].

- **Q7: What are the limitations and benefits of using a given physical system for RC?** In a physical RC implementation, the dynamics as well as the RC performance are governed by the physics of the actual substrate used. In other words, some physical substrates may be better for certain tasks than others. A comprehensive study and framework of the relationship between the physical properties of substrates and the reservoir performance is needed. The study should consider the analysis of physical implementations using both conventional and unconventional substrates.
- **Q8: What are the benefits of a physical implementation of RC?** Another important question is to determine what the benefits of a physical RC implementation really are. Computing with a bucket of water may neither be fast nor particularly useful. And with today’s high performance computers and custom architectures geared for neural network simulations, it is often faster to simply run a RC system in simulation than using an actual physical implementation. On the other hand, as pointed out above, a physical RC can be more natural to interface if the signals are already in the right form. For example, an optical RC would be easy to interface with optical signals, as opposed to electrical signals. Also, physical implementations could be more energy efficient and more robust against noise.

4. APPLICATIONS

For low-dimensional problems, such as chaotic time-series prediction, anomaly detection, control problems, and human action recognition, RC is able to compete with many classical machine learning techniques. Hierarchies of RCs have been used to achieve competitive results in voice recognition [65]. In light of the recent advances in deep learning, there is a growing trend in the machine learning community to perform sensory processing on high-dimensional raw inputs, such as raw pixel values of photos and videos [47]. It is known that the performance of an RC system typically suffers in the presence of high-dimensional inputs [14, 26].

- **Q9: Can RC improve real-time processing of video signals?** Video processing for autonomous cars, embedded systems, the internet of things, robotics, action-recognition, and scene comprehension is a growing application area for neural networks. Classical neu-

ral networks can achieve good and reasonable fast performance thanks to the power of modern GPUs, compute clusters, and the cloud. Finding ways for RC to contribute to this area could significantly lower the cost and the power consumption of real-time video processing. This would be particularly important for embedded applications, such as mobile platforms, where battery life is a concern.

- **Q10: What are promising approaches to process high-dimensional inputs using RC?** Currently, high-dimensional RC input signals have to go through several steps of low-dimensional feature extraction pre-processing. PCA is one of the most popular techniques [38]. However, complex high-dimensional visual data cannot be effectively processed using PCA [69]. Some form of hybrid architecture that combines RC with richer forms of pre-processing could solve that challenge.
- **Q11: Can hybrid models based on deep learning and RC benefit real-time video processing?** The main short-term memory in the brain, the prefrontal cortex, is heavily interconnected with other parts of the brain [12,44,58]. Visual information passes through several processing stages before reaching the prefrontal cortex [2,24,35]. Could brain-inspired architectures use a mix of deep learning and RC to efficiently process high-dimensional information and then pass it to a RC for further processing?

5. CONCLUSIONS

RC is a powerful machine learning technique for temporal information processing. A key benefit of RC is the ability to use the transient dynamics of a simulated or physical system for information processing. This removes the necessity of adaptation and learning in the computational core, i.e., the reservoir. As a consequence, RC can be used for almost any physical system that provides sufficiently rich dynamics, and, if perturbed, allows interpreting these dynamics with a linear output layer.

We have proposed a set of questions that we believe the community should attempt to answer in order to bring RC to solve more complex real-world problems, such as real-time video processing, real-time control problems, implementing cyber-physical and embedded systems, and designing cognitive systems. It is our hope that these questions will influence future research directions in that exciting area.

6. ACKNOWLEDGMENTS

We thank Darko Stefanovic, Mathew R. Lakin, Mattia Rigotti, Jens Bürger, and David Sussilo for their fruitful comments on this work. This work was supported by the National Science Foundation under awards # 1028378 and #1518833 and by the Defense Advanced Research Projects Agency (DARPA) under award # HR0011-13-2-0015. The views expressed are those of the author(s) and do not reflect the official policy or position of the Department of Defense or the U.S. Government. Approved for Public Release, Distribution Unlimited.

7. REFERENCES

- [1] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer. Information processing using a single dynamical node as complex system. *Nat Commun*, 2:468, 2011.
- [2] C. L. Asplund, J. J. Todd, A. P. Snyder, and R. Marois. A central role for the lateral prefrontal cortex in goal-directed and stimulus-driven attention. *Nat Neurosci*, 13(4):507–512, 2010.
- [3] A. Atiya and A. Parlos. New results on recurrent network training: unifying the algorithms and accelerating convergence. *Neural Networks, IEEE Transactions on*, 11(3):697–709, 2000.
- [4] A. V. Avizienis, C. Martin-Olmos, H. O. Sillin, M. Aono, J. K. Gimzewski, and A. Z. Stieg. Morphological transitions from dendrites to nanowires in the electroless deposition of silver. *Crystal Growth & Design*, 13(2):465–469, 2013.
- [5] A. V. Avizienis, H. O. Sillin, C. Martin-Olmos, H. H. Shieh, M. Aono, A. Z. Stieg, and J. K. Gimzewski. Neuromorphic atomic switch networks. *PLoS ONE*, 7(8):e42772, 2012.
- [6] L. Boccato, A. Lopes, R. Attux, and F. J. Von Zuben. An extended echo state network using volterra filtering and principal component analysis. *Neural Networks*, 32(0):292–302, 2012.
- [7] D. V. Buonomano and W. Maass. State-dependent computations: spatiotemporal processing in cortical networks. *Nat Rev Neurosci*, 10(2):113–125, 2009.
- [8] J. Bürger, A. Goudarzi, D. Stefanovic, and C. Teuscher. Computational capacity and energy consumption of complex resistive switch networks. *AIMS Materials Science*, 2(4):530–545, 2015.
- [9] J. Bürger, A. Goudarzi, D. Stefanovic, and C. Teuscher. Hierarchical composition of memristive networks for real-time computing. In *Proceedings of the 2015 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, pages 33–38. IEEE Press, 2015.
- [10] L. Büsing, B. Schrauwen, and R. Legenstein. Connectivity, dynamics, and memory in reservoir computing with binary and analog neurons. *Neural Computation*, 22(5):1272–1311, 2010.
- [11] J. Butcher, D. Verstraeten, B. Schrauwen, C. Day, and P. Haycock. Reservoir computing and extreme learning machines for non-linear time-series data analysis. *Neural Networks*, 38(0):76–89, 2013.
- [12] J. D. Cohen, W. M. Perlstein, T. S. Braver, L. E. Nystrom, D. C. Noll, J. Jonides, and E. E. Smith. Temporal dynamics of brain activation during a working memory task. *Nature*, 386(6625):604–608, 1997.
- [13] J. P. Crutchfield, W. L. Ditto, and S. Sinha. Introduction to focus issue: Intrinsic and designed computation: Information processing in dynamical systems—beyond the digital hegemony. *Chaos*, 20:037101, 2010.
- [14] J. Dambre, D. Verstraeten, B. Schrauwen, and S. Massar. Information processing capacity of dynamical systems. *Sci. Rep.*, 2:514, 2012.
- [15] S. Dasgupta, F. Wörgötter, and P. Manoonpong. Information theoretic self-organised adaptation in

- reservoirs for temporal memory tasks. In C. Jayne, S. Yue, and L. Iliadis, editors, *Engineering Applications of Neural Networks*, volume 311 of *Communications in Computer and Information Science*, pages 31–40. Springer Berlin Heidelberg, 2012.
- [16] E. C. Demis, R. Aguilera, H. O. Sillin, K. Scharnhorst, E. J. Sandouk, M. Aono, A. Z. Stieg, and J. K. Gimzewski. Atomic switch networks-nanoarchitectonic design of a complex system for natural computing. *Nanotechnology*, 26(20):204003, 2015.
- [17] P. Dominey, M. Arbib, and J.-P. Joseph. A model of corticostriatal plasticity for learning oculomotor associations and sequences. *Journal of Cognitive Neuroscience*, 7(3):311–336, 1995.
- [18] C. Fernando and S. Sojakka. Pattern recognition in a bucket. In W. Banzhaf, J. Ziegler, T. Christaller, P. Dittrich, and J. Kim, editors, *Advances in Artificial Life*, volume 2801 of *Lecture Notes in Computer Science*, pages 588–597. Springer Berlin Heidelberg, 2003.
- [19] S. Ganguli, D. Huh, and H. Sompolinsky. Memory traces in dynamical systems. *Proceedings of the National Academy of Sciences*, 105(48):18970–18975, 2008.
- [20] S. Ganguli and H. Sompolinsky. Short-term memory in neuronal networks through dynamical compressed sensing. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 667–675. Curran Associates, Inc., 2010.
- [21] A. Goudarzi, M. Lakin, and D. Stefanovic. Reservoir computing approach to robust computation using unreliable nanoscale networks. In O. H. Ibarra, L. Kari, and S. Kopecki, editors, *Unconventional Computation and Natural Computation*, Lecture Notes in Computer Science, pages 164–176. Springer International Publishing, 2014.
- [22] A. Goudarzi, M. R. Lakin, and D. Stefanovic. DNA reservoir computing: A novel molecular computing approach. In D. Soloveichik and B. Yurke, editors, *DNA Computing and Molecular Programming*, volume 8141 of *Lecture Notes in Computer Science*, pages 76–89. Springer International Publishing, 2013.
- [23] A. Goudarzi, S. Marzen, P. Banda, G. Feldman, C. Teuscher, and D. Stefanovic. Memory and information processing in recurrent neural networks. *arXiv:1604.06929v1 [cs.NE]*, 2016.
- [24] G. G. Gregoriou, S. J. Gotts, H. Zhou, and R. Desimone. High-frequency, long-range coupling between prefrontal and visual cortex during attention. *Science*, 324(5931):1207–1210, 2009.
- [25] N. D. Haynes, M. C. Soriano, D. P. Rosin, I. Fischer, and D. J. Gauthier. Reservoir computing with a single time-delay autonomous boolean node. *Phys. Rev. E*, 91:020801, 2015.
- [26] M. Hermans and B. Schrauwen. Memory in reservoirs for high dimensional input. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–7, 2010.
- [27] X. Hinault and P. F. Dominey. Real-time parallel processing of grammatical structure in the fronto-striatal system: A recurrent network simulation study using reservoir computing. *PLoS ONE*, 8(2):e52946, 2013.
- [28] R. Ho, K. Mai, and M. Horowitz. The future of wires. *Proceedings of the IEEE*, 89(4):490–504, 2001.
- [29] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [30] B. Horne, H. Siegelmann, and C. Giles. What narx networks can compute. In M. Bartosek, J. Staudek, and J. Wiedermann, editors, *SOFSEM '95: Theory and Practice of Informatics*, volume 1012 of *Lecture Notes in Computer Science*, pages 95–102. Springer Berlin Heidelberg, 1995.
- [31] H. Jaeger. The “echo state” approach to analysing and training recurrent neural networks. Technical Report GMD Rep. 148, St. Augustin: German National Research Center for Information Technology, 2001.
- [32] H. Jaeger. Short term memory in echo state networks. Technical Report GMD Report 152, GMD-Forschungszentrum Informationstechnik, 2002.
- [33] H. Jaeger and H. Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78–80, 2004.
- [34] Y. Katayama, T. Yamane, D. Nakano, R. Nakane, and G. Tanaka. Wave-based device scaling concept for brain-like energy efficiency and integration. In *Nanoscale Architectures (NANOARCH), 2015 IEEE/ACM International Symposium on*, pages 23–24, 2015.
- [35] F. Katsuki and C. Constantinidis. Early involvement of prefrontal cortex in visual bottom-up attention. *Nat Neurosci*, 15(8):1160–1166, 2012.
- [36] M. Kulkarni and C. Teuscher. Memristor-based reservoir computing. In *Proceedings of the IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH'12)*, pages 226–232, 2012.
- [37] R. Legenstein and W. Maass. Edge of chaos and prediction of computational performance for neural circuit models. *Neural Networks*, 20(3):323–334, 2007.
- [38] M. Lukoševičius. A practical guide to applying echo state networks. In G. Montavon, G. B. Orr, and K.-R. Müller, editors, *Neural Networks: Tricks of the Trade: Second Edition*, pages 659–686. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [39] M. Lukoševičius, H. Jaeger, and B. Schrauwen. Reservoir computing trends. *KI - Künstliche Intelligenz*, 26(4):365–371, 2012.
- [40] W. Maass. Motivation, theory, and applications of liquid state machines. In B. Cooper and A. Sorbi, editors, *Computability in Context: Computation and Logic in the Real World*. Imperial College Press, 2011.
- [41] W. Maass, P. Joshi, and E. D. Sontag. Principles of real-time computing with feedback applied to cortical microcircuit models. In *Advances in Neural Information Processing Systems 18*, pages 835–842, Cambridge, MA, 2006. MIT Press.
- [42] W. Maass, T. Natschläger, and H. Markram. Computational models for generic cortical microcircuits. In J. Feng, editor, *Computational Neuroscience: a Comprehensive Approach*, chapter 18, pages 575–605. CRC Press, 2002.

- [43] W. Maass, T. Natschläger, and H. Markram. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531–60, 2002.
- [44] V. Mante, D. Sussillo, K. V. Shenoy, and W. T. Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, 2013.
- [45] J. Mertens and I. Sutskever. Learning recurrent neural networks with hessian-free optimization. In *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.
- [46] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur. Recurrent neural network based language model. In *INTERSPEECH*, pages 1045–1048, 2010.
- [47] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
- [48] K. Nakajima, T. Li, H. Hauser, and R. Pfeifer. Exploiting short-term memory in soft body dynamics as a computational resource. *Journal of The Royal Society Interface*, 11(100):20140437, 2014.
- [49] O. Obst, A. Trinchì, S. G. Hardin, M. Chadwick, I. Cole, T. H. Muster, N. Hoschke, D. Ostry, D. Price, K. N. Pham, and T. Wark. Nano-scale reservoir computing. *Nano Communication Networks*, 4(4):189–196, 2013.
- [50] M. C. Ozturk, D. Xu, and J. C. Principe. Analysis and design of echo state networks. *Neural Computation*, 19(1):111–138, 2007.
- [51] L. Pasa and A. Sperduti. Pre-training of recurrent neural networks via linear autoencoders. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3572–3580, 2014.
- [52] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. *JMLR W & CP*, 28:1310–1318, 2013.
- [53] B. Schrauwen and L. Büesing. A hierarchy of recurrent networks for speech recognition. In *Advances in Neural Information Processing Systems (NIPS) Workshops*, 2009.
- [54] J. Shi, J. Wan, H. Yan, and H. Suo. A survey of cyber-physical systems. In *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*, pages 1–6, 2011.
- [55] H. T. Siegelmann and E. D. Sontag. Turing computability with neural nets. *Applied Mathematics Letters*, 4(6):77–80, 1991.
- [56] H. O. Sillin, R. Aguilera, H.-H. Shieh, A. V. Avizienis, M. Aono, A. Z. Stieg, and J. K. Gimzewski. A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. *Nanotechnology*, 24(38):384004, 2013.
- [57] Q. Song and Z. Feng. Effects of connectivity structure of complex echo state network on its prediction performance for nonlinear time series. *Neurocomputing*, 73(10-12):2177 – 2185, 2010.
- [58] T. Spellman, M. Rigotti, S. E. Ahmari, S. Fusi, J. A. Gogos, and J. A. Gordon. Hippocampal-prefrontal input supports spatial encoding in working memory. *Nature*, 522(7556):309–314, 2015.
- [59] J. J. Steil. Online reservoir adaptation by intrinsic plasticity for backpropagation decorrelation and echo state learning. *Neural Networks*, 20(3):353–364, 2007.
- [60] A. Z. Stieg, A. V. Avizienis, H. O. Sillin, C. Martin-Olmos, M. Aono, and J. K. Gimzewski. Emergent criticality in complex Turing B-type atomic switch networks. *Advanced Materials*, 24(2):286–293, 2012.
- [61] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams. The missing memristor found. *Nature*, 453(7191):80–83, 2008.
- [62] D. Sussillo and L. Abbott. Transferring learning from external to internal weights in echo-state networks with sparse connectivity. *PLoS ONE*, 7(5):1–8, 2012.
- [63] J.-P. Thivierge and P. Cisek. Nonperiodic synchronization in heterogeneous networks of spiking neurons. *Journal of Neuroscience*, 28(32):7968–7978, 2008.
- [64] M. H. Tong, A. D. Bickett, E. M. Christiansen, and G. W. Cottrell. Learning grammatical structure with echo state networks. *Neural Networks*, 20(3):424–432, 2007.
- [65] F. Triefenbach, A. Jalalvand, B. Schrauwen, and J.-P. Martens. Phoneme recognition with large hierarchical reservoirs. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 2307–2315. Curran Associates, Inc., 2010.
- [66] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman. Experimental demonstration of reservoir computing on a silicon photonics chip. *Nat Commun*, 5, 2014.
- [67] O. L. White, D. D. Lee, and H. Sompolinsky. Short-term memory in orthogonal neural networks. *Phys. Rev. Lett.*, 92:148102, 2004.
- [68] P. Xu, S.-H. Jeon, H.-T. Chen, H. Luo, G. Zou, Q. Jia, M. Anghel, C. Teuscher, D. J. Williams, B. Zhang, X. Han, and H.-L. Wang. Facile synthesis and electrical properties of silver wires through chemical reduction by polyaniline. *The Journal of Physical Chemistry C*, 114(50):22147–22154, 2010.
- [69] P. V. Yoshua Bengio, Aaron Courville. Representation learning: A review and new perspectives. *arXiv:1206.5538 [cs.LG]*, 2012.