Frontiers in Reservoir Computing

Claudio Gallicchio[1], Mantas Lukoševičius[2], Simone Scardapane[3]

1 - University of Pisa, Italy
2 - Kaunas University of Technology, Lithuania
3 - Sapienza University of Rome, Italy

Abstract. Reservoir computing (RC) studies the properties of large recurrent networks of artificial neurons, with either fixed or random connectivity. Over the last years, reservoirs have become a key tool for pattern recognition and neuroscience problems, being able to develop a rich representation of the temporal information even if left untrained. The common paradigm has been instantiated into several models, among which the Echo State Network and the Liquid State Machine represent the most widely known ones. Nowadays, RC represents the de facto state-of-the-art approach for efficient learning in the temporal domain. Besides, theoretical studies in RC area can contribute to the broader field of Recurrent Neural Networks research by enabling a deeper understanding of the fundamental capabilities of dynamical recurrent models, even in the absence of training of the recurrent connections. RC paradigm also allows using different dynamical systems, including hardware, for computation.

This paper is intended to give an overview on the RC research field, highlighting major frontiers in its development and finally introducing the contributed papers to the ESANN 2020 special session.

1 Introduction

The success of deep learning can be attributed to two factors. Firstly, the architecture of modern deep networks, which allows them to encode important biases on the data processing, e.g., convolutional filtering operations for images, attention for sequences. Secondly, their training mechanisms, providing a simple and effective way of adapting most internal parameters through gradient descent.

Reservoir computing (RC) [1, 2], and Echo State Networks (ESNs) [3, 4] in particular, were originally proposed as a way to overcome the training difficulties of classical recurrent neural networks (RNNs) [5, 6, 7]. They do this by considering only fixed (i.e., non-trainable) recurrent components, allowing to reduce the training process to a linear regression. Because ESNs lend themselves easily to formal characterizations, there is a vast literature underscoring the dynamics and memory capacity of this class of networks.

Despite the training simplification, RC remains today, even after years of important breakthroughs in fully-trainable neural networks, an essential research field with many important open questions and frontiers of development. Among other things, it sheds light on the relative importance of model structure versus training algorithms of different shallow or deep architectures. Without the pretension of being exhaustive, this tutorial paper touches on some of the recent lines of RC research that are particularly appealing, focusing on mathematical foundations, deep models and RC for structured data. It also introduces the papers presented at the *Frontiers in Reservoir Computing* special session of the ESANN 2020 conference.

This paper is organized as follows. We first provide an introduction to the basics of RC in Section 2. Then, we briefly discuss some major recent research directions in RC in Section 3. Finally, the papers presented at the special session are introduced in Section 4.

2 Reservoir Computing

The field of RC [1, 2] originated in 2001 with Liquid State Machines (LSMs) [8] from a computational neuroscience side and ESNs [3, 4] from a machine learning side as a simplified way of training Recurrent Neural Networks (RNNs), where the recurrent part does not need to be adapted as long as it is generated following certain rules.

A typical ESN update equation is [9]

$$\mathbf{x}(n) = (1-\alpha)\mathbf{x}(n-1) + \alpha \tanh\left(\mathbf{W}^{\text{in}}[1;\mathbf{u}(n)] + \mathbf{W}\mathbf{x}(n-1)\right), \tag{1}$$

where $\alpha \in (0,1]$ is the leaking rate, $\mathbf{x}(n) \in \mathbb{R}^{N_\text{x}}$ are reservoir activations, $\mathbf{u}(n) \in \mathbb{R}^{N_\text{u}}$ is the input, both at time step n, $\tanh(\cdot)$ is applied element-wise, $[\cdot;\cdot]$ stands for vector concatenation, $\mathbf{W}^{\text{in}} \in \mathbb{R}^{N_\text{x}\times(1+N_\text{u})}$ and $\mathbf{W} \in \mathbb{R}^{N_\text{x}\times N_\text{x}}$ are the input and recurrent weight matrices respectively. The recurrent part (1) is called the reservoir (hence "reservoir computing"). An ESN typically has a linear readout layer from the reservoir

$$\mathbf{y}(n) = \mathbf{W}^{\text{out}}[1;\mathbf{u}(n);\mathbf{x}(n)], \tag{2}$$

where $\mathbf{y}(n) \in \mathbb{R}^{N_\text{y}}$ is the network output and $\mathbf{W}^{\text{out}} \in \mathbb{R}^{N_\text{y}\times(1+N_\text{u}+N_\text{x})}$ is the output weight matrix. A graphical depiction of the ESN structure and training is presented in Figure 1.

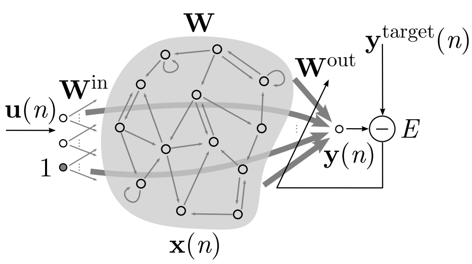

Fig. 1: An echo state network [9].

Typically $\mathbf{W}^{\text{in}}$ and $\mathbf{W}$ weights are generated randomly using some heuristic rules [9] based on conditions for asymptotic stability of reservoir dynamics, commonly known under the name of the echo state property [4, 10]. The elements of $\mathbf{W}^{\text{out}}$ are the only weights trained. Since there is no recurrence in the trained part, $\mathbf{W}^{\text{out}}$ can be computed in one go using linear regression after running the reservoir with the data $\mathbf{u}(n)$ and collecting $[1; \mathbf{u}(n); \mathbf{x}(n)]$ with corresponding target outputs $\mathbf{y}^{\text{target}}(n)$ for all training time n [9].

In subsequent years the RC framework for RNN training, that conceptually separated the recurrent reservoir from the readout, became a fruitful platform for analysing RNNs and proposing numerous modifications and extensions to the original RC methods [2, 11].

3 Frontiers

In this section we briefly discuss some potentially groundbreaking recent developments of RC research, warning the reader that the touched list of topics is far from being exhaustive for obvious reasons of brevity.

A large amount of research work in RC is building on its efficiency and providing even more efficient implementations. Since the reservoir can be random, it is susceptible to implementations in exotic hardware dynamical systems that are not necessarily programmable in the classical digital sense, including analog electronic, photonic, mechanical, biological, etc. Hardware RC implementations have lately become a very active interdisciplinary research area that is beyond the scope of this short overview. For this we refer the readers to a recent review [12]. At the same time, the efficient training of the readouts has been recently further improved to add a time series cross-validation with minimal additional computational costs [13]. This enables cross-validation to became a standard practice in RC for a more reliable model and hyper-parameter selection.

A major strand of research in the field is focused on the mathematical foundations of RC networks. This kind of studies is enlightened by the fact that learning is restricted to a particularly simple readout part, and the emerging properties of the dynamical recurrent layer can be more easily studied and highlighted exploiting, e.g., the theory of filters or the theory of dynamical systems. A fundamental question is that of universality, i.e., the identification of the class of transformations that can be accurately approximated by RC networks. Previous works in this direction focused on continuous time cases in the formalism of LSMs [14], exploiting the properties of filters with fading memory. Recently, a number of fundamental results have been introduced in the literature for to the case of discrete time systems described by the ESN-style formalism. In [15] it was proven that ESNs can approximate any fading memory filter. This result was later extended by the same authors in [16], where they showed analogous properties for several classes of RC models. This includes non-linear reservoirs parametrized as trigonometric state affine systems with linear readout, and, interestingly, even linear reservoirs provided that the output is computed by either a polynomial readout or by a multi-layer perceptron. Relevantly, these studies attempt to give a unified view over some of the most fundamental properties of reservoir systems: the fading memory property [17], the pairwise separation property [18], and the echo state property [4, 10].

Another line of theoretical studies focuses on conditions that would ensure stability of the reservoir dynamics in presence of external driving inputs. Here the goal is to analyze the quality of the reservoir dynamics studied as a non-linear and non-autonomous dynamical system [19, 20, 21]. Interesting results in this regard were recently introduced in [22], were the concept of echo state property was found to have profound implications in terms of robustness to inputs more in general for RNNs. The analysis of (parameters) and input perturbations has in this context an important role, which can be used to identify the boundaries of echo state property validity (as a function of the specific input). From a practical perspective, the work in [23] introduced an index of asymptotic synchronization of the reservoir state trajectories. More recently, in [24] a further index of reservoir stability has been proposed, with the aim of measuring the number of local point attractors of dynamics.

Deep RC defines another increasingly popular line of research. Early works in this direction [25, 26, 27, 28] already showed the merits of hierarchically combining multiple ESN architectural components, somehow anticipating the surge of works on deep recurrent neural architectures [29, 30, 31]. A great appeal of deep RNNs is that they enable to treat multiple time-scales (and, in general, multiple temporal views at different granularities) in a natural fashion. The recent results on Deep Echo State Networks (DeepESNs) [32], actually showed that such ability is indeed intrinsic to the architectural design of stacked RNNs, i.e., it is a *bias* of deep recurrent neural systems. Properly designed deep RNNs even without (or prior to) learning of recurrent connections are already able to outperform state of the art methodologies in complex tasks on time series, such as polyphonic music and speech processing [33]. Compared to the case of standard shallow reservoirs of ESNs described in (1), a deep reservoir can be described as a pipeline of non-linear systems, where the state update equation in each layer l is given by

$$\mathbf{x}^{[l]}(n) = (1-\alpha^{[l]})\mathbf{x}^{[l]}(n-1)+\alpha^{[l]} \tanh\left(\mathbf{W}^{\mathrm{in}[l]}[1;\mathbf{u}^{[l]}(n)] + \mathbf{W}^{[l]}\mathbf{x}^{[l]}(n-1)\right), \quad (3)$$

where the superscript $[l]$ indicates that the quantity is referred to the l-th layer. Crucially, for $l = 1$ the driving input is the external signal, i.e. $\mathbf{u}^{[l]}(n) = \mathbf{u}(n)$, while each successive layer is driven by the activation of the previous one, i.e., $\mathbf{u}^{[l]}(n) = \mathbf{x}^{[l-1]}(n)$ for $l > 1$. Conditions for the valid initialization of the set of input and recurrence matrices are given in [34]. Further RC approaches to the design of deep models for time-series processing have been introduced in [35, 36, 37].

Finally, dynamical recurrent models based on RC can find applications also in more general contexts where the nature of the data is more complex than time-series or sequences. A domain of particular interest in the current development of machine learning is that of graph data, where the information to be processed is represented in terms of entities and relations among them. The interested reader can find a primer introduction to the field of deep learning for graphs in [38]. In this context, reservoir systems can be used as alternative solutions to commonly adopted convolutional neural networks for graphs, with the striking advantage of a parsimonious training algorithm [39]. The basic idea is to extend the operation of the standard reservoir system to operate on discrete graph structures, where the role played by time-steps is now taken by the vertices of the input structure, and the relation of "previous time-step" is now generalized to the concept of neighborhood:

$$\mathbf{x}(v) = \tanh\left(\mathbf{W}^{\mathrm{in}}[1;\mathbf{u}(v)] + \sum_{v' \in \mathcal{N}(v)} \mathbf{W}\mathbf{x}(v')\right). \tag{4}$$

In (4), v denotes a vertex in the input graph, $\mathbf{u}(v)$ is a vector of features associated to it, and $\mathcal{N}(v)$ is the neighborhood of v (i.e., the set of vertices adjacent to v). In this way, it is possible to encode each input graph as the stationary state of a reservoir dynamical system. Recently, an extension of the above described approach in the direction of deep RC has been presented in [40], where deep reservoir for graphs enable the design of *fast and deep* graph neural networks achieving state of the art performance on graph classification problems with small training costs.

4 Special Session Papers

This section introduces the papers presented at the ESANN special session, contributing to the advancements of RC research under several perspectives.

Reservoir memory machines. The authors of [41] study RC systems in the context of memory augmented neural networks, by proposing a Neural Turing Machine (NTM) [42, 43] architecture in which the learned RNN controller is substituted by an ESN. The contribution can be seen as both an NTM alternative that is faster to be trained, and as an extension of the standard ESN, augmented with an external memory unit. The preliminary experimental analysis presented in the paper shows that Reservoir Memory Machines are able to solve some simple tasks typical of NTM, going beyond the performance achievable by simple ESNs.

Pyramidal Graph Echo State Networks. In [44] reservoirs are considered in the context of deep learning for graphs. In particular, the authors propose a deep architecture comprising multiple graph reservoir layers interleaved by pooling operations, exploring different graph coarsening approaches. Interestingly, the paper introduces the concept of pooling in fully untrained neural architectures for graphs. The results show that the proposed approach offers a particularly advantageous trade-off, allowing to further reduce the burden of training (with respect to the already efficient deep RC for graphs) at the cost of a small decrease in classification accuracy.

Simplifying Deep Reservoir Architectures. The paper [45] is positioned in the research line on deep RC. Specifically, the authors investigate progressively simplifications to the construction of a DeepESN, where each reservoir layer is constrained to a ring topology, and the connections from the external input, and those in between consecutive reservoir layers, are constructed in a deterministic fashion. The resulting RC model transfers the idea of minimal ESNs [46] to the case of deep architectures. The experimental results presented in the paper indicate the practical advantages of the introduced minimal deep RC design.

Self-organized dynamic attractors in recurrent neural networks. In [47], the authors study the properties of the attractors of reservoir dynamical systems in an unsupervised learning setting. In their analysis, they consider an RC system with additional reservoir recurrence connections that are adapted with differential Hebbian learning [48]. Experiments in an example setup show the emergence of different kinds of dynamics, with a prevalence of periodic and quasi-periodic attractors. The study is proposed in the perspective of introducing mechanisms based on self-organization for developing persistent memory in physical RC systems [49, 50].

Self-Organizing Kernel-based Convolutional Echo State Network for Human Actions Recognition. The authors of [51] propose a hierarchical RC architecture based on convolutional ESN, where the reservoir states over time are processed by a convolution level followed by a max over time pooling and by an output level. Interestingly, the reservoir weight matrices $\mathbf{W}^{\mathrm{in}}$ and $\mathbf{W}$ are determined by a preliminary phase of unsupervised adaptation using a self-organizing algorithm. The resulting approach is demonstrated on problems in the area of skeleton-based human action recognition.

References

[1] David Verstraeten, Benjamin Schrauwen, Michiel d'Haene, and Dirk Stroobandt. An experimental unification of reservoir computing methods. *Neural networks*, 20(3):391–403, 2007.

[2] Mantas Lukoševičius and Herbert Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149, 2009.

[3] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *science*, 304(5667):78–80, 2004.

[4] Herbert Jaeger. The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148(34):13, 2001.

[5] Kenji Doya. Bifurcations in the learning of recurrent neural networks. In *Proceedings of IEEE International Symposium on Circuits and Systems 1992*, volume 6, pages 2777–2780, 1992.

[6] Danilo P Mandic and Jonathon Chambers. *Recurrent neural networks for prediction: learning algorithms, architectures and stability.* John Wiley & Sons, Inc., 2001.

[7] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318, 2013.

[8] Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531–2560, 2002.

[9] Mantas Lukoševičius. A practical guide to applying echo state networks. In Grégoire Montavon, Geneviève B. Orr, and Klaus-Robert Müller, editors, *Neural Networks: Tricks of the Trade, 2nd Edition*, volume 7700 of *LNCS*, pages 659–686. Springer, 2012.

[10] Izzet B Yildiz, Herbert Jaeger, and Stefan J Kiebel. Re-visiting the echo state property. *Neural networks*, 35:1–9, 2012.

[11] Mantas Lukoševičius, Herbert Jaeger, and Benjamin Schrauwen. Reservoir computing trends. *KI - Künstliche Intelligenz*, 26(4):365–371, 2012.

[12] Gouhei Tanaka, Toshiyuki Yamane, Jean Benoit HÃ©roux, Ryosho Nakane, Naoki Kanazawa, Seiji Takeda, Hidetoshi Numata, Daiju Nakano, and Akira Hirose. Recent advances in physical reservoir computing: A review. *Neural Networks*, 115:100 – 123, 2019.

[13] Mantas Lukoševičius and Arnas Uselis. Efficient cross-validation of echo state networks. In *Artificial Neural Networks and Machine Learning –ICANN 2019: Workshop and Special Sessions. ICANN 2019.*, volume 11731 of *Lecture Notes in Computer Science*, pages 121–133. Springer, Cham, 2019.

[14] Wolfgang Maass. Liquid state machines: motivation, theory, and applications. In *Computability in context: computation and logic in the real world*, pages 275–296. World Scientific, 2011.

[15] Lyudmila Grigoryeva and Juan-Pablo Ortega. Echo state networks are universal. *Neural Networks*, 108:495–508, 2018.

[16] Lukas Gonon and Juan-Pablo Ortega. Reservoir computing universality with stochastic inputs. *IEEE transactions on neural networks and learning systems*, 2019.

[17] Stephen Boyd and Leon Chua. Fading memory and the problem of approximating nonlinear operators with volterra series. *IEEE Transactions on circuits and systems*, 32(11):1150–1161, 1985.

[18] Robert Legenstein and Wolfgang Maass. What makes a dynamical system computationally powerful. *New directions in statistical signal processing: From systems to brain*, pages 127–154, 2007.

[19] Gandhi Manjunath and Herbert Jaeger. Echo state property linked to an input: Exploring a fundamental characteristic of recurrent neural networks. *Neural computation*, 25(3):671–696, 2013.

[20] Filippo Maria Bianchi, Lorenzo Livi, and Cesare Alippi. Investigating echo-state networks dynamics by means of recurrence analysis. *IEEE transactions on neural networks and learning systems*, 29(2):427–439, 2016.

[21] Lorenzo Livi, Filippo Maria Bianchi, and Cesare Alippi. Determination of the edge of criticality in echo state networks through fisher information maximization. *IEEE transactions on neural networks and learning systems*, 29(3):706–717, 2017.

[22] Gandhi Manjunath. Memory-loss is fundamental for stability and distinguishes the echo state property threshold in reservoir computing & beyond. *arXiv preprint arXiv:2001.00766*, 2020.

[23] Claudio Gallicchio. Chasing the echo state property. *Proceedings of ESANN*, 2019. arXiv preprint arXiv:1811.10892.

[24] Andrea Ceni, Peter Ashwin, Lorenzo Livi, and Claire Postlethwaite. The echo index and multistability in input-driven recurrent neural networks. *arXiv preprint arXiv:2001.07694*, 2020.

[25] Herbert Jaeger. Discovering multiscale dynamical features with hierarchical echo state networks. Technical report, Jacobs University Bremen, 2007.

[26] Mantas Lukoševičius. Self-organized reservoirs and their hierarchies. In *International Conference on Artificial Neural Networks*, pages 587–595. Springer, 2012.

[27] Fabian Triefenbach, Azarakhsh Jalalvand, Benjamin Schrauwen, and Jean-Pierre Martens. Phoneme recognition with large hierarchical reservoirs. In *Advances in neural information processing systems*, pages 2307–2315, 2010.

[28] Fabian Triefenbach, Azarakhsh Jalalvand, Kris Demuynck, and Jean-Pierre Martens. Acoustic modeling with hierarchical reservoirs. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(11):2439–2450, 2013.

[29] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

[30] Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. How to construct deep recurrent neural networks. *arXiv preprint arXiv:1312.6026*, 2013.

[31] Michiel Hermans and Benjamin Schrauwen. Training and analysing deep recurrent neural networks. In *Advances in neural information processing systems*, pages 190–198, 2013.

[32] Claudio Gallicchio, Alessio Micheli, and Luca Pedrelli. Deep reservoir computing: A critical experimental analysis. *Neurocomputing*, 268:87–99, 2017.

[33] Claudio Gallicchio, Alessio Micheli, and Luca Pedrelli. Design of deep echo state networks. *Neural Networks*, 108:33–47, 2018.

[34] Claudio Gallicchio and Alessio Micheli. Echo state property of deep reservoir computing networks. *Cognitive Computation*, 9(3):337–350, 2017.

[35] Filippo Maria Bianchi, Simone Scardapane, Sigurd Løkse, and Robert Jenssen. Bidirectional deep-readout echo state networks. *Proceedings of ESANN*, 2018. arXiv preprint arXiv:1711.06509.

[36] Stefano Nichele and Andreas Molund. Deep learning with cellular automaton-based reservoir computing. 2017.

[37] Xiaochuan Sun, Tao Li, Qun Li, Yue Huang, and Yingqi Li. Deep belief echo-state network and its application to time series prediction. *Knowledge-Based Systems*, 130:17–29, 2017.

[38] Davide Bacciu, Federico Errica, Alessio Micheli, and Marco Podda. A gentle introduction to deep learning for graphs. *arXiv preprint arXiv:1912.12693*, 2019.

[39] Claudio Gallicchio and Alessio Micheli. Graph echo state networks. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2010.

[40] Claudio Gallicchio and Alessio Micheli. Fast and deep graph neural networks. In *Proceedings of AAAI 2020*, 2020. arXiv preprint arXiv:1911.08941.

[41] Benjamin Paassen and Alexander Schulz. Reservoir memory machines. In *Proceedings of ESANN*, 2020.

[42] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

[43] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476, 2016.

[44] Filippo Maria Bianchi, Claudio Gallicchio, and Alessio Micheli. Pyramidal graph echo state networks. In *Proceedings of ESANN*, 2020.

[45] Claudio Gallicchio, Alessio Micheli, and Antonio Sisbarra. Simplifying deep reservoir architectures. In *Proceedings of ESANN*, 2020.

[46] Ali Rodan and Peter Tino. Minimum complexity echo state network. *IEEE transactions on neural networks*, 22(1):131–144, 2010.

[47] Benedikt Vettelschoss, Matthias Freiberger, and Joni Dambre. Self-organized dynamic attractors in recurrent neural networks. In *Proceedings of ESANN*, 2020.

[48] Bart Kosko. Differential hebbian learning. In *AIP Conference proceedings*, volume 151, pages 277–282. American Institute of Physics, 1986.

[49] Daniel Brunner, Miguel C Soriano, and Guy Van der Sande. *Photonic Reservoir Computing: Optical Recurrent Neural Networks*. Walter de Gruyter GmbH & Co KG, 2019.

[50] Gouhei Tanaka, Toshiyuki Yamane, Jean Benoit Héroux, Ryosho Nakane, Naoki Kanazawa, Seiji Takeda, Hidetoshi Numata, Daiju Nakano, and Akira Hirose. Recent advances in physical reservoir computing: A review. *Neural Networks*, 2019.

[51] Gin Chong Lee, Chu Kiong Loo, Wei Shiung Liew, and Stefan Wermter. Self-organizing kernel-based convolutional echo state network for human actions recognition. In *Proceedings of ESANN*, 2020.