Topological and simplicial features in reservoir computing networks

James McAllister[1], John Wade[1], Conor Houghton[2], and Cian O'Donnell[1,2]

[1] Intelligent Systems Research Centre, Ulster University
mcallister-j23@ulster.ac.uk
c.odonnell2@ulster.ac.uk
[2] School of Engineering Mathematics and Technology, University of Bristol

Abstract. Reservoir computing is a framework which uses the non-linear internal dynamics of a recurrent neural network to perform complex non-linear transformations of the input. This enables reservoirs to carry out a variety of tasks involving the processing of time-dependent or sequential-based signals. Reservoirs are particularly suited for tasks that require memory or the handling of temporal sequences, common in areas such as speech recognition, time series prediction, and signal processing. Learning is restricted to the output layer and can be thought of as "reading out" or "selecting from" the states of the reservoir. With all but the output weights fixed they do not have the costly and difficult training associated with deep neural networks. However, while the reservoir computing framework shows a lot of promise in terms of efficiency and capability, it can be unreliable. Existing studies show that small changes in hyperparameters can markedly affect the network's performance. Here we studied the role of network topologies in reservoir computing in the carrying out of three conceptually different tasks: working memory, perceptual decision making, and chaotic time-series prediction. We implemented three different network topologies (ring, lattice, and random) and tested reservoir network performances on the tasks. We then used algebraic topological tools of directed simplicial cliques to study deeper connections between network topology and function, making comparisons across performance and linking with existing reservoir research.

Keywords: reservoirs, recurrent neural networks, network topologies, directed simplicial cliques

1 Introduction

The link between structure and function in networks of complex systems has been demonstrated theoretically and empirically, but a clear and comprehensive understanding of it is nevertheless elusive [2]. Networks have many applications, but wherever they are applied, their structure has important consequences for their function or behaviour. In reservoir computing [7], the structure and connectivity of the reservoir — a recurrent neural network — has received much interest, as it impacts upon task performance [4, 20, 11]. The parameters of the reservoir network components have vital implications for the performance of the reservoir. These parameters include input signal scaling factor, connection sparsity of the reservoir, spectral radius of the weight matrix (the maximum of the absolute values of its eigenvalues), and the choice of non-linear activation function in the network nodes. However, even very small changes in the hyperparameters can markedly affect the network's performance [9, 20]. The performance differences even with similar distributions of parameters indicate that the heart of the matter may lie in the topology of the network [4].

Answers to the structure-function question in reservoir networks could lead to increased computational ability and efficiency. Knowing *a priori* which network features might optimise functionality would reduce a large amount of the uncertain, inefficient, and costly "groping in the dark" for suitable networks, and *post hoc* adjustments to fine-tune the setup. A better knowledge of the structure-function relationship in reservoir computing would not only be of theoretical interest, but also of great potential practical use.

2 Reservoir Computing and Network Structure

In reservoir computing a recurrent neural network (RNN) is used as a fixed, random, large-scale dynamical system, called the reservoir, which is used to process time-dependent or sequential-based signals. It is particularly suited for tasks that require memory or the handling of temporal sequences, common in areas such as speech recognition, time series prediction, and signal processing [7, 17]. The key elements of the framework (see Fig. 1) are: the input layer with randomised fixed weights $\boldsymbol{W}_{\text{in}}$; the reservoir with fixed internal weights $\boldsymbol{W}$ of a desired sparsity and/or node degree, with weights often uniformly sampled between -1 and 1 and scaled by the spectral radius; and the output layer with weight matrix $\boldsymbol{W}_{\text{out}}$, which is typically trained by regularised linear regression. The recurrent neural networks in reservoirs have rich dynamics allowing complex non-linear transformations of their input. The dynamics are driven by the equation

$$\boldsymbol{x}(t+1) = (1-\alpha)\boldsymbol{x}(t) + \alpha f(\boldsymbol{W}\boldsymbol{x}(t) + \boldsymbol{W}_{\text{in}}\boldsymbol{i}(t))$$

where α is the leaky coefficient, f is the non-linear activation function, $\boldsymbol{x}$ is the state vector, and $\boldsymbol{i}$ is the input data. By non-linearly embedding the input into a higher dimensional feature space, the problem is more likely to be linearly separable and therefore solvable (Cover's theorem [3]). Since the training of the system is restricted to the output layer, reservoirs do not have the costly and challenging training associated with deep neural networks. This makes them particularly desirable from an efficiency point of view. However, as mentioned above, it is not fully known how the connectivity of nodes within the reservoir affect performance. There is a connection between reservoir performance and general architectural factors such as modularity [22], small-world qualities [11], scale-free characteristics [8], and even specific topologies [4]. However, even reservoirs with ostensibly equivalent characteristics can show drastically varied performance depending on their initialisations.

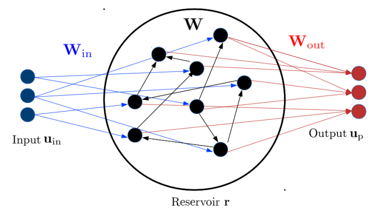

Fig. 1: Adapted [20] illustration of the setup of a reservoir, with input neurons and fixed input weights W_{in} on the left; fixed internal reservoir with weights W in the centre; and the trained output layer with weights W_{out} on the right.

Here we examine the roles of network connectivity and topology on the performance of reservoir networks. We identify topological fingerprints of reservoir networks in three conceptually and practically different tasks: working memory, perceptual decision making, and chaotic time-series prediction. The working memory task measures the degree to which the reservoir can reproduce sequential random input data after time delay; the perceptual decision making task tests the capacity to process motion data and discern direction coherence; and the time series prediction task measures how long the reservoir can accurately predict the Lorenz system. We first use reservoirs of differing explicit network topologies (ring, lattice, and random) and test their performances on the three tasks mentioned above. Some previous work has studied the general performance properties of several explicit topologies — including ring and lattice — for computing substrates using a CHARC (CHAracterisation of Reservoir Computers) framework [4]. However, they only investigated three generic properties of reservoirs in these topologies, namely, the separation property, the generalisation property, and the echo state property. These characteristics are important for reservoir dynamics, but do not fully capture all dynamical properties and functional capacities of a reservoir. We will test topology performance on *specific* tasks and functions. We then make use of concepts from algebraic topology to more deeply probe into the roles of higher-dimensional connectivity in reservoir function. This involves developing reservoirs with differing dimensions of directed simplicial cliques [1] and analysing resulting task-performances.

3 Methodology

Here we outline the specifics of the three tasks on which we test reservoir performance, followed by the various topological concepts and tools which will be implemented.

3.1 Reservoir Tasks

Working Memory This working memory task is inspired by other reservoir work [5, 10, 22]. In this task paradigm, a random input sequence $X(t)$ is presented to the network through an input neuron. The network independently learns delayed versions of the input, producing multiple outputs. Each output Y_τ predicts the input $X(t)$ delayed by τ time steps, i.e., $Y_\tau(t) = X(t-\tau)$. The input values are randomly drawn from a uniform distribution, $X(t) \sim \text{Uniform}(-0.5, 0.5)$. The networks are trained for 4000 time steps and tested on the subsequent 1000. Each output is trained independently, and the performance, referred to as Working Memory Capacity, is calculated as the cumulative squared Pearson correlation coefficient (ρ) across all outputs:

$$\text{Working Memory Capacity} = \sum_{\tau} \rho^2(y_i, \hat{y}_i)$$

where y_i and $\hat{y}_i$ denote the true and predicted values, respectively. Although such a simple input-output delay is a trivial task from an engineering point of view, we consider it a valuable benchmark task for reservoir networks, because any complicated task on time series data will need to be able to temporarily store information from the past in order to combine it with the present input.

Perceptual Decision Making The random dot motion task is a widely used paradigm in perceptual decision-making experiments involving human and animal participants [24, 23]. In this task, participants are presented with a display consisting of randomly moving dots on a screen. The dots move in different directions and speeds, creating a noisy visual stimulus. The task is typically to detect or discriminate the overall direction of motion among the dots, which may be either coherent (moving in the same direction) or incoherent (moving randomly). By varying the parameter of coherence level (the proportion of dots moving coherently), we manipulate the difficulty of the task. The random dot motion task provides insight into how systems integrate sensory information to make perceptual decisions amidst uncertainty, making it a valuable tool for studying visual perception and decision-making processes [12, 24]. In this task, we create dot motion data, where a specified number of dots move left or right with a certain degree of coherence. The reservoir must learn the coherence level and "decide" the correct direction of the dot motion, providing this as output. We use the reciprocal mean squared error as a measure of performance:

$$\text{Decision Making Performance} = \left(\frac{1}{N} \sum_{i=1}^{N} \frac{1}{(y_i - \hat{y}_i)^2} \right)^{-1}$$

where N is the number of data points, y_i the true value, and $\hat{y}_i$ predicted value.

Chaotic Time-Series Prediction The canonical example of a reservoir computing task is prediction of the chaotic time-series data from the Lorenz system [18, 4, 7]. The Lorenz system is notable for having chaotic solutions for certain parameter values and initial conditions. We train the reservoir on a set of Lorenz data, and then provide the system with a random point on the Lorenz trajectory, and measure the valid time for which the reservoir can accurately predict the true trajectory. We calculate the normalised mean squared error [18],

$$\mathrm{NMSE} = \frac{\|\boldsymbol{u}(t) - \tilde{\boldsymbol{u}}(t)\|}{\sqrt{\langle \|\boldsymbol{u}(t)\|^2 \rangle}}$$

where $\boldsymbol{u}(t)$ is the true value and $\tilde{\boldsymbol{u}}(t)$ is the predicted value. We measure, in Lyapunov time, the valid time of the predicted trajectory, that is, the point at which the predicted trajectory exceeds a specified NMSE threshold, set at 0.4 in this study.

3.2 Topological Tools

Explicit Network Topologies We studied networks with three distinct topologies: ring, lattice, and random (Fig. 2). The ring topology has the lowest degree of complexity. It connects each node to a fixed number of nearest neighbours in a circular arrangement, with a random subset of nodes also having recurrent connections to themselves. The lattice topology has a greater connection complexity. With this structure, we arrange nodes in a square grid where each node is connected to its immediate neighbouring nodes, with a randomised subset of nodes also having recurrent connections to themselves. These two networks, along with the random network, were all initialised with the same number of nodes, equivalent sparsity, and equal spectral radius. The three types of network structure were implemented in the reservoir framework on the three tasks defined above. The means and standard deviations of the performance on the three tasks using each type of network were used to compare between the three topologies' capacities. With three groups for comparison, we conducted a non-parametric Kruskal-Wallis rank sum test due to lack of homogeneity in data variance. This was to determine if there were statistically significant differences in the means of performances. Post-hoc pairwise tests were used to find pairwise comparisons in significantly different measures.

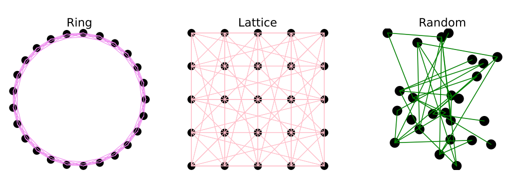

Fig. 2: Three topologies used in reservoir networks.

Directed Simplicial Cliques A simplex is the generalisation of a tetrahedron to higher dimensions [16]; in a directed simplex the direction of transmission is taken into account [1]. Directed simplices are network cliques containing a single **source** neuron and a single **sink** neuron (Fig. 3), reflecting a specific motif of connectivity [25, 19]. The presence of simplicial cliques and cavities correlates with functional roles of the network, such as the processing and integration of information in the brain [21]. The activity of nodes within the network is dependent on the number and dimension of the cliques that the nodes belongs to, and their specific positions in directed cliques. This concept has been useful in analysing and modelling networks [15, 21]; however, little has been done to study the implication these properties have for reservoir function. We constructed enhanced networks with higher-than-average numbers of high dimensional directed simplicial cliques. This process involved initialising a graph with a specified number nodes and creating dense, highly interconnected subgraphs (cliques) by systematically connecting the nodes in these subgraphs with structured edges. Subsequently, the graph is pruned to achieve a desired level of sparsity. Fig. 4 demonstrates the difference in numbers of directed simplicial cliques between a random network and our enhanced networks.

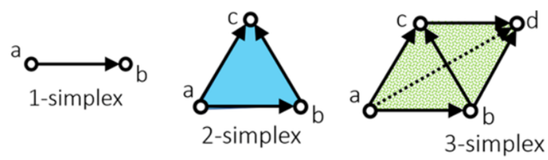

Fig. 3: Example directed simplices (with a source and a sink) of 3 different dimensions.

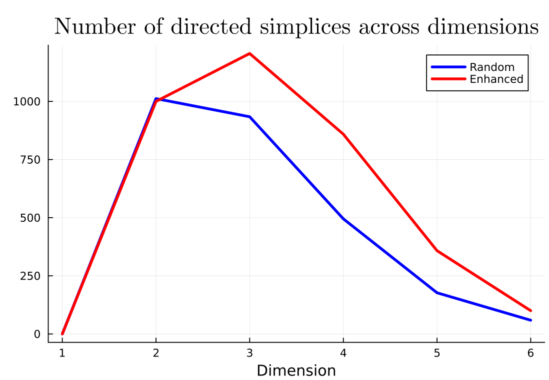

Fig. 4: Comparison of number of directed simplicial cliques of various dimensions in a random and enhanced network.

As in the case of the explicit topologies, we examined the differences in performance across the three tasks. In this case, we only had two groups for comparison, where data variance was non-homogenous. We therefore carried out a non-parametric Wilcoxon Signed-Rank test to determine if there was a statistically significant difference between the performances.

Eigenspectral Analyses In addition to the topological concepts employed in this work, we also studied the eigenspectra of the various networks used as reservoirs, calculating spectral measures such as spectral gap and participation ratio. The spectral gap is the difference between the moduli of the two largest eigenvalues. This measure gives information about a graph's potential for information propagation, resilience to disruptions, and overall efficiency in various network dynamics [6, 14]. A large spectral gap may be associated with high network performance, such as facilitated synchronisation and rapid convergence. The participation ratio formula used is:

$$\text{Participation ratio} = \frac{(\sum_i \lambda_i)^2}{\sum_i \lambda_i^2}$$

where λ_i are the eigenvalues of the weight matrix. Participation ratio is a measure of dimensionality; it provides a quantitative measure of how broadly or narrowly the network's connectivity or activity is distributed. It helps assess the extent to which certain nodes dominate or share influence within the network, offering insights into the heterogeneity of timescales in the network dynamics [13].

3.3 Parameters

Following parameter sweeps on baseline random reservoirs, we set the following parameters constant throughout the study: reservoir size = 400; spectral radius = 0.9; input scaling = 0.1; sparsity = 0.9; regularisation coefficient = 1×10^6.

4 Results

4.1 Ring, Lattice, and Random Topologies

The performance of the reservoirs with different topologies varied across the tasks (Fig. 5).

Interestingly, the only significant difference (at the $< 1\%$ confidence level) in performance is seen in the working memory task, where the ring topology was outperformed by both the lattice and random topologies. The other two tasks showed no statistically significant differences (at the $< 1\%$ confidence level). Post-hoc tests confirmed that the ring topology performance was significantly different to the other two, but that there was no significant difference between the lattice and randomised topology performances (not shown).

The unique feature of the working memory task is its requirement of a longer term memory. The decision making (dot motion) and time-series prediction (Lorenz) tasks do not *generally* require the ability to remember far into the past; they primarily require only current and very recent historical data to accurately compute the next output. The performance of the ring topology in the

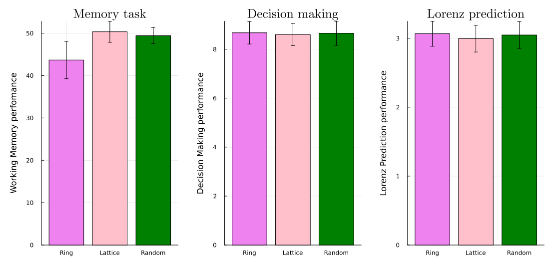

Fig. 5: Performances (means and standard deviations) of different topologies on 3 tasks.

working memory task indicates that the simplicity and naivety of its structure make it less suitable for long-term memory. For the other tasks, however, the simple ring topology appears to perform equally well.

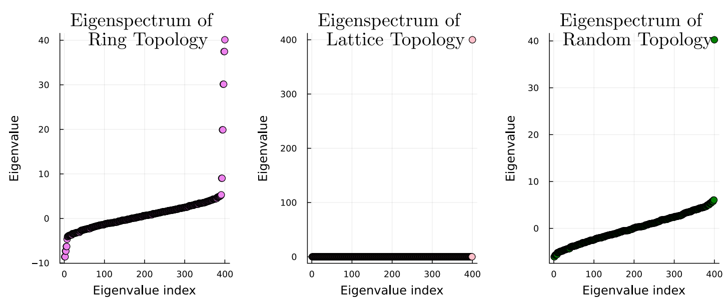

Fig. 6: Eigenspectra of different topologies.

	Spectral Gap	Participation Ratio
Ring	5.36	0.0229
Lattice	400.00	0.2506
Random	34.12	0.0033

Table 1: Spectral measures of networks with differing topologies.

Examining the eigenspectra (Fig. 6), along with the spectral gap and participation ratio measures (Table 1) of the different topologies gives a clearer picture and insight into their underlying differences. The ring topology stands out for its very small spectral gap, indicating a weaker graph modularity, and lower degree of expansion and information transmission. The other two topologies (especially the lattice) exhibit higher spectral gaps, indicating stronger modularity and concomitant network synchronisability. These are possibly important features when it comes to the ability to internalise longer term historical inputs, as in the case of the working memory task.

4.2 Directed Simplicial Clique Enhancement

In the resulting performances we found that across the three different tasks, as in the case of the three explicit topologies, a significant difference (at 1% level) was found in the memory task, where random reservoirs outperform those with higher degrees of simplicial cliques (Fig. 7).

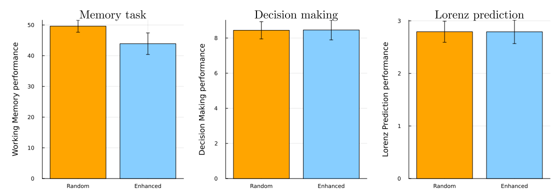

Fig. 7: Performances (means and standard deviations) of random and enhanced networks.

Examining the eigenspectra of the random and "enhanced" topologies reveals an interesting fact: the eigenspectrum of the enhanced topology closely resembles that of the ring topology in the preceding section (Fig. 8). It furthermore also demonstrates a smaller spectral gap (Table 2), indicating that the same potential issues inherent to the ring topology may be present with the simplicially enhanced networks.

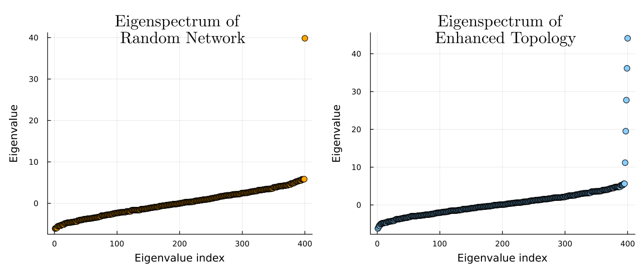

Fig. 8: Eigenspectra of networks with differing directed simplicial dimensions.

	Spectral gap	Participation ratio
Random networks	34.06	0.0028
Enhanced networks	8.83	0.0028

Table 2: Spectral measures across simplicially differing networks.

5 Discussion

We examined the role of explicitly defined network topologies and enhanced topological features on specific computational tasks in reservoir computing. Our primary finding is that reservoir performance can be surprisingly robust to topological differences in the reservoir connectivity.

Indeed, we found that the only statistically significant difference between the network architectures lies with the ring topology and in the memory task. The lack of significant difference between the lattice and random graph may simply be due to the fact that they both have higher degrees of recurrent connections, despite their eigenspectra being qualitatively different. The ring topology is highly regular and simple, with the correspondingly smallest spectral gap. As indicated previously, the decision making and time-series prediction require only a relatively little degree of memory, whereas the working memory task by definition requires longer term storage of historical input. This kind of task may place demands upon the internal recurrent dynamics of the reservoir, which are highly dependent on the network connectivity. The simple and regular connectivity of the ring topology perhaps make it unsuitable for producing the complex, nonlinear, recurrent dynamics required for internalising long-term memory. The ring topology has been previously identified as exhibiting more linear and ordered dynamics, and theoretically possessing limitations of memory [4]. Our work has corroborated this in a particular use-case.

In order to probe deeper into the question of network topology and specific task performance, we developed a framework for constructing "enhanced" networks with high-dimensional directed simplices. In comparison with randomly initialised networks, these enhanced networks possess greater amounts of high dimensional topological features: characteristics which have been observed in complex network data, and posited to be important for network function [21]. However, the findings of our research indicate that simple enhancement of higher dimensional directed simplices does not significantly improve the performance of reservoirs on the three specific computational tasks of interest to this study. The only statistically significant outcome was that the random reservoirs outperform the enhanced networks in the memory task. Once again, this can possibly be traced to the much larger spectral gap of the random reservoirs, indicating a higher degree of network synchronisation, and information transmission. This may be vital for the internal dynamics necessary for longer term memory.

Our work carries some limitations. First, we implemented the topological and simplicial features within a very narrow context. A greater investigation of how these network features vary with contexts is necessary, such as how their roles develop over different reservoir sizes, spectral radii, sparsities, etc. Second, we did not expand upon the possibilities of defining particular distributions of simplicial dimensions; the specificity and complexity of these directed simplicial cliques could be extended. Therefore how distinct higher simplicial dimensions and motifs impact performance and network efficiency is yet to be understood.

6 Conclusion

Here we tested the network-function relationship question using topology and reservoir computing. The promise of effective and efficient computation through reservoir networks is to be made more certain through a better understanding of the underlying topological features of recurrent neural networks internal to the reservoir. This work has studied the role of three explicit topologies on performance in three tasks. It demonstrated that the specific network topologies considered (ring and lattice) do not generally have significant impact on task performance in comparison with random networks. However, the ring topology significantly underperformed in the memory task. This indicates that certain topological features may underpin performance in tasks requiring longer-term memory. This work further developed a "directed simplicial clique" enhanced framework for implementing reservoirs with higher-dimensional connectivity, which also produced a significant difference from randomly initialised networks in the working memory task, but not in the other two. Overall this suggests that the reservoir computing framework may be highly robust to high-level topological alterations in the internal reservoir.

Acknowledgements: The first author would like to thank the Faculty of Engineering, University of Bristol, for a visiting scholarship and the Northern Ireland Department for the Economy for a PhD studentship.

References

1. R. Aharoni, E. Berger, and R. Meshulam, "Eigenvalues and homology of flag complexes and vector representations of graphs," GAFA, Geometric and Functional Analysis, vol. 15, no. 3, pp. 555–566, Jun. 2005, doi:10.1007/s00039-005-0516-9.
2. S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D.-U. Hwang, "Complex networks: Structure and dynamics", Physics Reports, vol. 424, 4–5, p. 175-308, 2006, https://doi.org/10.1016/j.physrep.2005.10.009.
3. T. M. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," IEEE Transactions on Electronic Computers, vol. EC-14, no. 3, pp. 326–334, Jun. 1965, doi:10.1109/PGEC.1965.264137.
4. M. Dale, S. O'Keefe, A. Sebald, S. Stepney, and M. A. Trefzer, "Reservoir computing quality: connectivity and topology," Natural Computing, vol. 20, no. 2, pp. 205–216, Jun. 2021, doi:10.1007/s11047-020-09823-1.
5. F. Damicelli, C. C. Hilgetag, and A. Goulas, "Brain connectivity meets reservoir computing," PLOS Computational Biology, vol. 18, no. 11, p. e1010639, Nov. 2022, doi:10.1371/journal.pcbi.1010639.
6. F. Chung, Spectral Graph Theory, vol. 92. in CBMS Regional Conference Series in Mathematics, vol. 92. Providence, Rhode Island: American Mathematical Society, 1996. doi:10.1090/cbms/092.
7. M. Cucchi, S. Abreu, G. Ciccone, D. Brunner, and H. Kleemann, "Hands-on reservoir computing: a tutorial for practical implementation," Neuromorphic Computing and Engineering, vol. 2, no. 3, p. 032002, Sep. 2022, doi:10.1088/2634-4386/ac7db7.
8. Z. Deng, and Y. Zhang. "Collective behavior of a small-world recurrent neural system with scale-free distribution", IEEE Transactions on Neural Networks, vol. 18, no. 5, pp. 1364–1375, 2007.
9. A. Haluszczynski and C. Räth, "Good and bad predictions: Assessing and improving the replication of chaotic attractors by means of reservoir computing," Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 29, no. 10, p. 103143, Oct. 2019, doi:10.1063/1.5118725.

10. H. Jaeger, “Short term memory in echo state networks”, 2001, doi:10.24406/PUBLICA-FHG-291107.
11. Y. Kawai, J. Park, and M. Asada, “A small-world topology enhances the echo state property and signal propagation in reservoir computing,” Neural Networks, vol. 112, pp. 15–23, Apr. 2019, doi:10.1016/j.neunet.2019.01.002.
12. A. Von Lautz, J. Herding, and F. Blankenburg, “Neuronal signatures of a random-dot motion comparison task,” NeuroImage, vol. 193, pp. 57–66, Jun. 2019, doi:10.1016/j.neuroimage.2019.02.071.
13. L. Lovász, Large Networks and Graph Limits, vol. 60. in Colloquium Publications, vol. 60. Providence, Rhode Island: American Mathematical Society, 2012. doi:10.1090/coll/060.
14. Mohar, B., Alavi, Y., Chartrand, G. and Oellermann, O., 1991. “The Laplacian spectrum of graphs”. Graph theory, combinatorics, and applications, vol. 2, pp. 871-898.
15. S. Momani, V.-T. Pham, and Z. Wei, “Directed simplicial complexes in brain real-world networks,” Eur. Phys. J. Spec. Top., vol. 233, no. 4, pp. 807–816, Jun. 2024, doi:10.1140/epjs/s11734-024-01159-6.
16. J. R. Munkres, Elements of Algebraic Topology, 1st ed. CRC Press, 2018. doi:10.1201/9780429493911.
17. M. Inubushi, K. Yoshimura, Y. Ikeda, and Y. Nagasawa, “On the Characteristics and Structures of Dynamical Systems Suitable for Reservoir Computing,” in Reservoir Computing, K. Nakajima and I. Fischer, Eds., in Natural Computing Series. Singapore: Springer Singapore, 2021, pp. 97–116. doi:10.1007/978.
18. J. Pathak et al., “Hybrid forecasting of chaotic processes: Using machine learning in conjunction with a knowledge-based model,” Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 28, no. 4, p. 041101, Apr. 2018, doi:10.1063/1.5028373.
19. R. Perin, T. K. Berger, and H. Markram, “A synaptic organizing principle for cortical neuronal groups,” Proceedings of the National Academy of Sciences USA, vol. 108, no. 13, pp. 5419–5424, Mar. 2011, doi:10.1073/pnas.1016051108.
20. A. Racca and L. Magri, “Robust optimization and validation of echo state networks for learning chaotic dynamics,” 2021, doi:10.48550/ARXIV.2103.03174.
21. M. W. Reimann et al., “Cliques of neurons bound into cavities provide a missing link between structure and function,” Frontiers of Computational Neuroscience., vol. 11, p. 48, Jun. 2017, doi:10.3389/fncom.2017.00048.
22. N. Rodriguez, E. Izquierdo, and Y.-Y. Ahn, “Optimal modularity and memory capacity of neural reservoirs,” Network Neuroscience, vol. 3, no. 2, pp. 551–566, Jan. 2019, doi:10.1162/netn_a_00082.
23. A. E. Rorie and W. T. Newsome, “A general mechanism for decision-making in the human brain?,” Trends in Cognitive Sciences, vol. 9, no. 2, pp. 41–43, Feb. 2005, doi:10.1016/j.tics.2004.12.007.
24. M. N. Shadlen and W. T. Newsome, “Neural Basis of a Perceptual Decision in the Parietal Cortex (Area LIP) of the Rhesus Monkey,” Journal of Neurophysiology, vol. 86, no. 4, pp. 1916–1936, Oct. 2001, doi:10.1152/jn.2001.86.4.1916.
25. S. Song, P. J. Sjöström, M. Reigl, S. Nelson, and D. B. Chklovskii, “Highly Nonrandom Features of Synaptic Connectivity in Local Cortical Circuits,” PLoS Biol, vol. 3, no. 3, p. e68, Mar. 2005, doi:10.1371/journal.pbio.0030068.