2016 IEEE International Conference on Computational Science and Engineering, IEEE International Conference on Embedded and Ubiquitous Computing, and International Symposium on Distributed Computing and Applications to Business, Engineering and Science

Reservoir computing: information processing of stationary signals

Lyudmila Grigoryeva*, Julie Henriques† and Juan-Pablo Ortega‡§
*Department of Mathematics and Statistics, Universität Konstanz, Box 146. D-78457 Konstanz. Germany.
†Methodology and Quality of Life in Oncology Unit, CHRU Besançon. F-25000 Besançon. France.
‡Faculty of Mathematics and Statistics, Universität Sankt Gallen, Bodanstrasse 6, CH-9000 Sankt Gallen. Switzerland.
§Centre National de la Recherche Scientifique (CNRS). France

***Abstract*—This paper extends the notion of information processing capacity for non-independent input signals in the context of reservoir computing (RC). The presence of input autocorrelation makes worthwhile the treatment of forecasting and filtering problems for which we explicitly compute this generalized capacity as a function of the reservoir parameter values using a streamlined model. The reservoir model leading to these developments is used to show that, whenever that approximation is valid, this computational paradigm satisfies the so called separation and fading memory properties that are usually associated with good information processing performances. We show that several standard memory, forecasting, and filtering problems that appear in the parametric stochastic time series context can be readily formulated and tackled via RC which, as we show, significantly outperforms standard techniques in some instances.**

***Keywords*-Reservoir computing, echo state networks, liquid state machines, time-delay reservoir, memory capacity, forecasting, stationary signals.**

I. Introduction

Reservoir computing is a recent but already well established neural computing paradigm [1], [2], [3], [4], [5], [6], [7] that has shown a significant potential in overcoming some of the limitations inherent to more standard Turing-type machines. This computation approach is characterized by a simple and convenient supervised learning scheme, even though its performance presents as a weak side a substantial sensitivity to architecture parameters. This feature explains the development in the literature of various linear and nonlinear memory capacity measures [8], [9], [10], [11], [12] as well as the study of different signal treatment properties (see [13], [7] and references therein) that are used to characterize and measure the information processing abilities of these devices in order to be able to optimize them.

We have proposed several contributions in this direction in our previous works [14], [15] in the context of RCs constructed via the sampling of the solutions of a time-delay differential equation. These RCs are usually referred to as time-delay reservoirs (TDRs). More specifically, in [14] we constructed a simplified model for those specific RCs that allowed us to provide a functional link between the RC parameters and its performance with respect to a given memory task and which can be used to accurately determine the optimal reservoir architecture by solving a well structured optimization problem. The availability of this tool simplifies enormously the implementation effort and sheds new light on the mechanisms that govern this information processing technique. This approach was extended in [15] in order to be able to handle multidimensional input signals and real-time multitasking [4], that is, the simultaneous execution of several memory tasks. Additionally, we used this approach to estimate the memory capacity of parallel arrays of reservoir computers. This reservoir architecture had been introduced in [16], [17], where it was empirically shown to exhibit various improved robustness properties.

The notion of capacity is defined using independent input signals, which immediately limits its practical functionality in several aspects. Indeed, the use of independent inputs makes empty of content the treatment of forecasting problems. Additionally, most input signals that need to be processed in specific tasks exhibit sizable autocorrelation, which automatically precludes independence. Finally, simple numerical experiments show that optimal reservoir architectures with respect to a given memory task lose that optimality as soon as the input signal ceases to be independent.

All these facts call for a generalization of the notion of capacity suitable for correlated signals and for techniques to compute it. This is the main goal of this work. More specifically, we use an extension of the RC model introduced in [14] in order to generalize the memory capacity formulas that were introduced in that paper to non-independent strictly stationary signals. Moreover, the presence of input autocorrelation makes worthwhile the treatment of forecasting and filtering problems for which we extend the notion of capacity and that we will explicitly compute as a function of the reservoir parameter values. These results can be readily used in the execution of specific tasks since the expressions that we obtain are written in terms of various statistical features of the input and the teaching signal that can be simply estimated out of the training sample.

The results in this paper are formulated for general discrete-time RCs that are not necessarily TDRs. We use the generalization of the model in [14] to this context in order to show that, for that approximation, RCs satisfy the so called fading memory and separation properties that are typically associated to good information processing performances

978-1-5090-3593-9/16 $31.00 © 2016 IEEE
DOI 10.1109/CSE-EUC-DCABES.2016.231

(see [13], [7] and references therein).

II. Reservoir computing and memory capacities

The reservoir computing construction that we consider is based on the choice of a nonautonomous discrete-time dynamical system of the form:

$$\mathbf{x}(t) = F(\mathbf{x}(t-1), \mathbf{I}(t), \boldsymbol{\theta}), \tag{1}$$

with $t \in \mathbb{Z}$, $\mathbf{x}(t)$, $\mathbf{I}(t) \in \mathbb{R}^N$, and $\boldsymbol{\theta} \in \mathbb{R}^K$. The map $F : \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^K \longrightarrow \mathbb{R}^N$ is called the **reservoir map**. The vector $\mathbf{x}(t)$ is referred to as the **neuron layer** at time t and each of its components $x_i(t)$ are its **neuron values**. The vector $\boldsymbol{\theta} \in \mathbb{R}^K$ contains the set of parameters that the reservoir map depends on. The vector $\mathbf{I}(t) \in \mathbb{R}^N$ is the **input forcing** of the reservoir that is constructed out of the **input signal** $\{z(t)\}_{t\in\mathbb{Z}}$, $z(t) \in \mathbb{R}$, by using an **input mask** $\mathbf{c} \in \mathbb{R}^N$ and by setting $\mathbf{I}(t) := \mathbf{c}z(t)$.

A given task is assigned to the RC by fixing a teaching signal $\{y(t)\}_{t\in\mathbb{Z}}$ and by minimizing the mean square error committed at the time of reproducing it with an affine combination of the reservoir output $\mathbf{x}(t)$ of the form $\mathbf{W}^\top \mathbf{x}(t) + a$, with $a \in \mathbb{R}$ and $\mathbf{W} \in \mathbb{R}^N$. The optimal pair $(\mathbf{W}_{\text{out}}, a_{\text{out}})$ is referred to as the **readout layer** and is obtained by solving the ridge (or Tikhonov) regularized regression problem

$$(\mathbf{W}_{\text{out}}, a_{\text{out}}) := \underset{\mathbf{W}\in\mathbb{R}^N, a\in\mathbb{R}}{\arg\min} \Big(\mathrm{E}\left[\left(\mathbf{W}^\top \mathbf{x}(t) + a - y(t) \right)^2 \right] + \lambda \left\| \mathbf{W} \right\|^2 \Big), \quad \lambda \in \mathbb{R}, \tag{2}$$

whose solution is given by

$$\mathbf{W}_{\text{out}} = (\Gamma(0) + \lambda \mathbb{I}_N)^{-1} \mathrm{Cov}(y(t), \mathbf{x}(t)), \tag{3}$$

$$a_{\text{out}} = \mathrm{E}[y(t)] - \mathbf{W}_{\text{out}}^\top \boldsymbol{\mu}_x. \tag{4}$$

In this expression, $\boldsymbol{\mu}_x := \mathrm{E}[\mathbf{x}(t)]$ is the expectation of the reservoir output and

$$\Gamma(0) := \mathrm{E}\left[(\mathbf{x}(t) - \boldsymbol{\mu}_x)(\mathbf{x}(t) - \boldsymbol{\mu}_x)^\top \right]$$

is the lag-zero autocovariance exhibited by the reservoir output. We show later on that if the input signal is strictly stationary then the two moments $\boldsymbol{\mu}_x$ and $\Gamma(0)$ are time-independent. The mean square error committed by the reservoir when using the optimal readout is:

$$\begin{aligned} \mathrm{E}\left[\left(\mathbf{W}_{\text{out}}^\top \cdot \mathbf{x}(t) + a_{\text{out}} - y(t) \right)^2 \right] &= \mathbf{W}_{\text{out}}^\top \Gamma(0) \mathbf{W}_{\text{out}} \\ + \mathrm{var}(y(t)) - 2\mathbf{W}_{\text{out}}^\top \mathrm{Cov}(y(t), \mathbf{x}(t)) &= \mathrm{var}(y(t)) \\ - \mathrm{Cov}(y(t), \mathbf{x}(t))^\top (\Gamma(0) + \lambda\mathbb{I}_N)^{-1} &(\Gamma(0) + 2\lambda\mathbb{I}_N) \\ \times (\Gamma(0) + \lambda\mathbb{I}_N)^{-1} \mathrm{Cov}(y(t), \mathbf{x}(t)). \end{aligned} \tag{5}$$

The **reservoir capacity** $C(\boldsymbol{\theta}, \mathbf{c}, \lambda)$ is defined as one minus the mean square error that we just computed, normalized with the variance of the teaching signal, that is,

$$C(\boldsymbol{\theta}, \mathbf{c}, \lambda) := \Big(\mathrm{Cov}(y(t), \mathbf{x}(t))^\top (\Gamma(0) + \lambda\mathbb{I}_N)^{-1} (\Gamma(0) + 2\lambda\mathbb{I}_N)(\Gamma(0) + \lambda\mathbb{I}_N)^{-1} \mathrm{Cov}(y(t), \mathbf{x}(t)) \Big) / \mathrm{var}(y(t)). \tag{6}$$

We emphasize that this capacity is a natural generalization to the context of non-independent stationary input signals of the notion introduced in [8], [9], [10], [11], [12]. Additionally, we point out that $C(\boldsymbol{\theta}, \mathbf{c}, \lambda)$ depends on, apart from the reservoir parameters $\boldsymbol{\theta}$, the input mask $\mathbf{c}$, and the regularization constant λ, also on the task determined by the teaching signal $\{y(t)\}_{t\in\mathbb{Z}}$. It is clear that $0 \leq C(\boldsymbol{\theta}, \mathbf{c}, \lambda) \leq 1$.

III. The reservoir model

The capacities for a reservoir of the form (1) are in general very difficult to compute analytically. In [14] we introduced an approximate model for TDRs that made possible an analytic estimation of their capacities under a strong independence hypothesis in the input signal; this condition was already present in the original definitions of this notion [8], [9], [10], [11], [12]. We now extend that construction to more general RCs driven by strictly stationary input signals. Additional details and proofs can be found in [18].

Definition 3.1: The time series $\{z(t)\}_{t\in\mathbb{Z}}$ is said to be **strictly stationary** if the joint distributions of $(z(t_1), \ldots, z(t_k))^\top$ and $(z(t_1+h), \ldots, z(t_k+h))^\top$ are the same for all $k \in \mathbb{N}$ and for all $t_1, \ldots, t_k, h \in \mathbb{Z}$.

Definition 3.2: Given a time series $\{z(t)\}_{t\in\mathbb{Z}}$, $r_1, \ldots, r_k$, $k \in \mathbb{N}$ and $t, h_2, \ldots, h_k \in \mathbb{Z}$ we define the corresponding **higher order automoment** $\mu_z^{r_1,\ldots,r_k}(t, h_2, \ldots, h_k)$ as

$$\mu_z^{r_1,\ldots,r_k}(t, h_2, \ldots, h_k) := \mathrm{E}\left[z(t)^{r_1} z(t+h_2)^{r_2} \cdots z(t+h_k)^{r_k} \right]. \tag{7}$$

together with the convention $\mu_z^{r_1}(t) = \mathrm{E}[z(t)^{r_1})]$.

Proposition 3.3: Let $\{z(t)\}_{t\in\mathbb{Z}}$ be a stochastic time series whose higher order automoments exist. If $\{z(t)\}_{t\in\mathbb{Z}}$ is strictly stationary then its higher order automoments are time-independent. In that case, we replace the notation in (7) by

$$\mu_z^{r_1,\ldots,r_k}(h_2, \ldots, h_k) := \mathrm{E}\left[z(t)^{r_1} z(t+h_2)^{r_2} \cdots z(t+h_k)^{r_k} \right], \tag{8}$$

for any $t \in \mathbb{Z}$.

The approximate model of the RC in (1) is obtained, as in [14], by partially linearizing F with respect to the self-delay at a stable fixed point $\mathbf{x}_0 \in \mathbb{R}^N$ of the autonomous system associated to (1). The point $\mathbf{x}_0 \in \mathbb{R}^N$ is chosen so that $F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) = \mathbf{x}_0$, $\boldsymbol{\theta} \in \mathbb{R}^K$, and for which the spectral radius $\rho(A(\mathbf{x}_0, \boldsymbol{\theta})) < 1$, with $A(\mathbf{x}_0, \boldsymbol{\theta}) := D_{\mathbf{x}} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta})$, in order to ensure stability. In [14] we provided both theoretical and empirical evidence that suggests that optimal reservoir performance can be achieved when working in a statistically stationary regime around a stable equilibrium. The stability of the point $\mathbf{x}_0$ implies, in passing, that the reservoir states $\mathbf{x}(t)$ remain close to $\mathbf{x}_0$, and hence justifies the approximate reservoir model:

$$\mathbf{x}(t) = \mathbf{x}_0 + A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t-1) - \mathbf{x}_0) + \boldsymbol{\varepsilon}(t), \tag{9}$$

where $\boldsymbol{\varepsilon}(t) = \left(q_R^1\left(z(t), \mathbf{c}\right), \ldots, q_R^N\left(z(t), \mathbf{c}\right)\right)^\top$, with $q_R^j(\cdot, \mathbf{c})$ a polynomial of degree $R \in \mathbb{N}$ whose monomial of order i has as coefficient the value $\frac{1}{i!} D_{\mathbf{I}}^{(i)} F_j(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \underbrace{(\mathbf{c} \otimes \cdots \otimes \mathbf{c})}_{i \text{ factors}}$ with F_j is the j-th component of the map $F := (F_1, \ldots, F_N)$ in (1). The strict stationarity of $\{z(t)\}_{t\in\mathbb{Z}}$ implies that of $\{\boldsymbol{\varepsilon}(t)\}_{t\in\mathbb{Z}}$. In particular,

$$\boldsymbol{\mu}_{\boldsymbol{\varepsilon}} := \mathrm{E}\left[\boldsymbol{\varepsilon}(t)\right] = \left(\left(q_R^1\left(x, \mathbf{c}\right)\right)\left(\mu_z^{\cdot}\right), \cdots, \left(q_R^N\left(x, \mathbf{c}\right)\right)\left(\mu_z^{\cdot}\right)\right)^\top, \tag{10}$$

where the symbol $\left(q_R^i\left(x, \mathbf{c}\right)\right)\left(\mu_z^{\cdot}\right)$ stands for the evaluation of the polynomial $q_R^N\left(x, \mathbf{c}\right)$ according to the following convention: any monomial of the form $a_r x^r$ is replaced by $a_r \mu_z^r$. A similar convention can be used to write down the autocovariance $\Gamma_{\boldsymbol{\varepsilon}}(h)$, $h \in \mathbb{Z}$ of $\{\boldsymbol{\varepsilon}(t)\}_{t\in\mathbb{Z}}$. Indeed, for any $i, j \in \{1, \ldots, N\}$:

$$\begin{aligned}(\Gamma_{\boldsymbol{\varepsilon}}(h))_{i,j} &= \mathrm{E}\left[\varepsilon^i(t)\varepsilon^j(t+h)\right] - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^i \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^j \\ &= \left(q_R^i\left(x, \mathbf{c}\right) \bullet q_R^j\left(y, \mathbf{c}\right)\right)\left(\mu_z^{\cdot\cdot}(h)\right) - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^i \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^j, \end{aligned} \tag{11}$$

where the symbol $\bullet$ denotes polynomial multiplication and the first summand stands for the evaluation of the bivariate polynomial $q_R^i\left(x, \mathbf{c}\right) \bullet q_R^j\left(y, \mathbf{c}\right)$ according to the following convention: any monomial of the form $a_{r,s} x^r y^s$ is replaced by $a_{r,s} \mu_z^{r,s}(h)$, with $\mu_z^{r,s}$ the second order automoment of $\{z(t)\}_{t\in\mathbb{Z}}$.

The following proposition shows that the strict stationarity of the input signal implies the second order stationarity of the output $\{\mathbf{x}(t)\}_{t\in\mathbb{Z}}$ of the approximate reservoir (9).

Proposition 3.4: Let $\{\mathbf{x}(t)\}_{t\in\mathbb{Z}}$ be the output of the reservoir model (9). Suppose that the spectral radius $\rho\left(A(\mathbf{x}_0, \boldsymbol{\theta})\right) < 1$ and that the input signal $\{z(t)\}_{t\in\mathbb{Z}}$ is strictly stationary and has finite automoments up to order $2R$ (R is the order of the expansion that defines the reservoir model (9)). Under those hypotheses, the reservoir output $\{\mathbf{x}(t)\}_{t\in\mathbb{Z}}$ is second order stationary with the first two moments given by:

$$\boldsymbol{\mu}_{\mathbf{x}} = \mathbf{x}_0 + \left(\mathbb{I}_N - A(\mathbf{x}_0, \boldsymbol{\theta})\right)^{-1} \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}, \text{ and}$$

$$\Gamma(h) = \sum_{j,k=0}^{\infty} A^j \Gamma_{\boldsymbol{\varepsilon}}\left(k - j - h\right)\left(A^k\right)^\top, \quad h \in \mathbb{Z}, \tag{12}$$

where $\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}$ and $\Gamma_{\boldsymbol{\varepsilon}}$ provided by (10) and (11), respectively. Under these hypotheses, the recursion (9) that determines the reservoir model can be rewritten as

$$(\mathbf{x}(t) - \boldsymbol{\mu}_{\mathbf{x}}) = A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t-1) - \boldsymbol{\mu}_{\mathbf{x}}) + (\boldsymbol{\varepsilon}(t) - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}), \tag{13}$$

that has as unique stationary solution

$$\mathbf{x}(t) = \boldsymbol{\mu}_{\mathbf{x}} + \sum_{j=0}^{\infty} A(\mathbf{x}_0, \boldsymbol{\theta})^j \left(\boldsymbol{\varepsilon}(t-j) - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right). \tag{14}$$

IV. Reservoir capacity estimations for signal forecasting, reconstruction, and filtering

We now use the reservoir model introduced in the previous section in order to provide capacity estimates for different information processing tasks. All along this section, we assume that the input signal $\{z(t)\}_{t\in\mathbb{Z}}$ is strictly stationary and has finite automoments up to order $2R$ so that we can use the results contained in Proposition 3.4. We refer the reader to [18] for additional details and other examples.

Let $\{z(t)\}_{t\in\mathbb{Z}}$ and $\{y(t)\}_{t\in\mathbb{Z}}$ be two one-dimensional stochastic time series that will be called in what follows the input and teaching signals, respectively. The goal of any machine learning based signal treatment strategy consists of using finite size realizations $\mathbf{z}_T := \{z(1), \ldots, z(T)\}$ and $\mathbf{y}_T := \{y(1), \ldots, y(T)\}$ of $\{z(t)\}_{t\in\mathbb{Z}}$ and $\{y(t)\}_{t\in\mathbb{Z}}$, respectively, in order to train a device that is capable of reproducing out-of-sample realizations $\mathbf{y}'_{T'}$ of the teaching signal out of a corresponding realization of the input signal $\mathbf{x}'_{T'}$. The pairs $(\mathbf{z}_T, \mathbf{y}_T)$ and $(\mathbf{z}'_{T'}, \mathbf{y}'_{T'})$ are referred to as training and testing samples, respectively.

A. Forecasting and reconstruction

We define a (f, h)-lag forecasting/reconstruction task as a function $H : \mathbb{R}^{f+h+1} \to \mathbb{R}$ that is used to generate a one-dimensional signal $y(t) = H\left(z(t+f), \ldots, z(t), \ldots, z(t-h)\right)$ that depends on the value of the input signal f lags into the future (forecasting part) and h lags into the past (reconstruction part).

The linear case: Consider the linear forecasting/reconstruction task $H : \mathbb{R}^{f+h+1} \to \mathbb{R}$ determined by the assignment

$$\begin{aligned}\mathbf{z}^{f,h}(t) = (z(t+f), \ldots, z(t), \ldots, z(t-h)) \in \mathbb{R}^{f+h+1} \\ \longmapsto \mathbf{L}^\top \mathbf{z}^{f,h}(t),\end{aligned}$$

with $\mathbf{L} \in \mathbb{R}^{f+h+1}$. The teaching signal is constructed by setting $y(t) := \mathbf{L}^\top \mathbf{z}^{f,h}(t)$. We now estimate the memory capacity $C_H(\boldsymbol{\theta}, \mathbf{c}, \lambda)$ associated to the task H and exhibited by the reservoir model (13). Notice that the evaluation of the capacity requires the computation of the lag-zero autocovariance $\Gamma(0)$ of the reservoir output in terms of the reservoir parameters, as well as $\mathrm{var}\left(y(t)\right)$ and $\mathrm{Cov}(y(t), \mathbf{x}(t))$. The expression for $\Gamma(0)$ is explicitly provided by (12); regarding $\mathrm{var}\left(y(t)\right)$ and $\mathrm{Cov}(y(t), \mathbf{x}(t))$ we have:

- $\operatorname{var}(y(t)) = \mathbf{L}^\top(\Gamma^z - \mu_z^2 \mathbf{i}_{f+h+1}\mathbf{i}_{f+h+1}^\top)\mathbf{L}$, with $\Gamma^z \in \mathbb{S}_{f+h+1}$ defined by $\Gamma^z_{i,j} = \mu_z^{1,1}(i-j)$, with $i,j \in \{1,\ldots,f+h+1\}$, and $\mu_z^{1,1}$ the second order automoment of the input signal.
- Using the unique stationary solution of the reservoir model in (13) it is possible to show that

$$\operatorname{Cov}(y(t),\mathbf{x}(t)) = \sum_{j=1}^{f+h+1}\sum_{k=0}^{\infty} L_j A(\mathbf{x}_0,\boldsymbol{\theta})^k \times \left[\begin{pmatrix} \left(x \bullet q_R^1(y,\mathbf{c})\right)\left(\mu_z^{1,\cdot}(f+k+1-j)\right) \\ \vdots \\ \left(x \bullet q_R^N(y,\mathbf{c})\right)\left(\mu_z^{1,\cdot}(f+k+1-j)\right)\end{pmatrix} - \mu_z \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right],$$

where this expression has been written using the same convention as in (11).

The quadratic case: Consider the quadratic forecasting/reconstruction task $H : \mathbb{R}^{f+h+1} \to \mathbb{R}$ defined by the assignment $\mathbf{z}^{f,h}(t) \longrightarrow \mathbf{z}^{f,h}(t)^\top Q \mathbf{z}^{f,h}(t)$, with $Q \in \mathbb{S}_{f+h+1}$. We then define the teaching signal

$$y(t) := H\left(\mathbf{z}^{f,h}(t)\right) = \sum_{i,j=1}^{f+h+1} Q_{i,j} z(t+f+1-i) z(t+f+1-j). \tag{15}$$

This implies that

$$\mu_y := \operatorname{E}[y(t)] = \sum_{i,j=1}^{f+h+1} Q_{i,j}\mu_z^{1,1}(i-j). \tag{16}$$

At the same time

$$y(t)^2 = \sum_{i,j,k,l=1}^{f+h+1} Q_{i,j}Q_{k,l} z(t+f+1-i)z(t+f+1-j) \times z(t+f+1-k)z(t+f+1-l),$$

and hence

$$\operatorname{E}\left[y(t)^2\right] = \sum_{i,j,k,l=1}^{f+h+1} Q_{i,j}Q_{k,l}\mu_z^{1,1,1,1}(i-j,i-k,i-l). \tag{17}$$

Consequently, by (16) and (17),

$$\operatorname{var}(y(t)) = \sum_{i,j,k,l=1}^{f+h+1} Q_{i,j}Q_{k,l}\mu_z^{1,1,1,1}(i-j,i-k,i-l) - \left(\sum_{i,j=1}^{f+h+1} Q_{i,j}\mu_z^{1,1}(i-j)\right)^2.$$

In order to compute $\operatorname{Cov}(y(t),\mathbf{x}(t))$, we use again the representation (13) and hence

$$\begin{aligned}
&\operatorname{Cov}(y(t),\mathbf{x}(t)) = \operatorname{Cov}(y(t),\mathbf{x}(t)-\boldsymbol{\mu}_{\mathbf{x}}) = \sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k \\
&\times \operatorname{Cov}(y(t),\boldsymbol{\rho}(t-k)) = \sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k \\
&\times \left[\operatorname{E}[y(t)\boldsymbol{\varepsilon}(t-k)] - \mu_y\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right] = \sum_{k=0}^{\infty}\sum_{i,j=1}^{f+h+1} A(\mathbf{x}_0,\boldsymbol{\theta})^k Q_{i,j} \\
&\times \operatorname{E}\left[z(t+f+1-i)z(t+f+1-j)\boldsymbol{\varepsilon}(t-k)\right] \\
&- \mu_y \sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k \boldsymbol{\mu}_{\boldsymbol{\varepsilon}} = \sum_{k=0}^{\infty}\sum_{i,j=1}^{f+h+1} Q_{i,j}A(\mathbf{x}_0,\boldsymbol{\theta})^k \\
&\times \begin{pmatrix} \left(x \bullet y \bullet q_R^1(z,\mathbf{c})\right)\left(\mu_z^{1,1,\cdot}(i-j,i-k-f-1)\right) \\ \vdots \\ \left(x \bullet y \bullet q_R^N(z,\mathbf{c})\right)\left(\mu_z^{1,1,\cdot}(i-j,i-k-f-1)\right)\end{pmatrix} \\
&- \mu_y \sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}.
\end{aligned}$$

B. *Filtering of stochastic costationary signals*

This case is a generalization of the previous one in which the input and teaching signal exhibit statistical dependence, even though they do not necessarily have a deterministic functional link. This statistical relation is used by the RC in order to construct a nonparametric estimation of the conditional expectation $\operatorname{E}[y(t)\,|\mathcal{F}_t]$, where $\mathcal{F}_t$ is the information set generated by the input signal up to time t, that is, $\mathcal{F}_t = \sigma(z(t), z(t-1), \ldots)$. This conditional expectation minimizes the mean square error committed by the RC at the time of reproducing the teaching signal.

Definition 4.1: Let $\{z(t)\}_{t\in\mathbb{Z}}$ and $\{y(t)\}_{t\in\mathbb{Z}}$ be two one-dimensional stochastic time series. Given $r \in \mathbb{N}$ and $h \in \mathbb{Z}$ we define the **higher order comoment** as

$$\mu_{y,z}^r(t,h) := \operatorname{E}\left[y(t)z(t+h)^r\right]. \tag{18}$$

If the higher-order comoments up to order r exist and are time-independent, we say that $\{y(t)\}_{t\in\mathbb{Z}}$ and $\{z(t)\}_{t\in\mathbb{Z}}$ are **rth-order costationary** and we note

$$\mu_{y,z}^r(h) := \operatorname{E}\left[y(t)z(t+h)^r\right], \quad \text{for any } t \in \mathbb{Z}. \tag{19}$$

Suppose now that $\{z(t)\}_{t\in\mathbb{Z}}$ is the input of the RC and $\{y(t)\}_{t\in\mathbb{Z}}$ is a teaching signal defining a specific filtering task. As we did all along this section, we assume that the input signal is strictly stationary and has finite automoments up to order $2R$; additionally we suppose that $\{z(t)\}_{t\in\mathbb{Z}}$ and $\{y(t)\}_{t\in\mathbb{Z}}$ are costationary of order R.

With these assumptions, we can explicitly spell out the performance of the RC in the filtering task by noting, first, that $\operatorname{var}(y(t))$ can be estimated out of the teaching signal and second, that by (14):

$$
\begin{aligned}
\mathrm{Cov}\left(y(t),\mathbf{x}(t)\right) &= \mathrm{Cov}\left(y(t),\mathbf{x}(t)-\boldsymbol{\mu}_{\mathbf{x}}\right)\\
&= \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j \mathrm{Cov}\left(y(t),\boldsymbol{\varepsilon}(t-j)-\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right)\\
&= \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j \mathrm{Cov}\left(y(t),\boldsymbol{\varepsilon}(t-j)\right)
\end{aligned}
$$

$$
= \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j \left[\begin{pmatrix} \left(x\bullet q_R^1(u,\mathbf{c})\right)\left(\mu_{y,z}^{\cdot}(-j)\right)\\ \vdots \\ \left(x\bullet q_R^N(u,\mathbf{c})\right)\left(\mu_{y,z}^{\cdot}(-j)\right)\end{pmatrix} - \mu_z \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right],
$$

where the expression $\left(x\bullet q_R^i(u,\mathbf{c})\right)\left(\mu_{y,z}^{\cdot}(-j)\right)$ stands for the evaluation of the polynomial $x\bullet q_R^i(u,\mathbf{c})$ on the variables x and u, according to the following convention: each monomial of the form axu^r is replaced by $a\mu_{y,z}^r(-j)$.

We emphasize that given the input and teaching signals $\{z(t)\}_{t\in\mathbb{Z}}$ and $\{y(t)\}_{t\in\mathbb{Z}}$, respectively, the higher order comoments can be estimated out of the training sample and inserted in the equation above. These elements provide an estimate of the RC capacity for any value of its parameters $\boldsymbol{\theta}$ and the input mask $\mathbf{c}$.

V. The Fading Memory and the Separation Properties

The fading memory and the separation properties have been identified in the context of reservoir computing to be in relation with good information processing performances (see [13], [7] and references therein). The goal of the following paragraphs is showing that the reservoir model (9) for the discrete-time reservoir computer (1) exhibits these features under reasonable assumptions on the reservoir map.

Definition 5.1: Consider the discrete-time reservoir map (1). We say that the reservoir map (1) satisfies the ***uniform fading memory property (UFMP)*** whenever for any $\varepsilon>0$ there exist $\delta_\varepsilon>0$ and $h_\varepsilon\in\mathbb{N}$ such that if for any two input signals $\{z(t)\}_{t\in\mathbb{Z}}$, $\{z'(t)\}_{t\in\mathbb{Z}}$ the relation $|z(s)-z'(s)|<\delta_\varepsilon$ holds for all $s\in[t-h_\varepsilon,t]$, $t\in\mathbb{Z}$, then the corresponding outputs $\mathbf{x}(t)$, $\mathbf{x}'(t)$ are such that $||\mathbf{x}(t)-\mathbf{x}'(t)||<\varepsilon$. The values $\delta_\varepsilon>0$ and $h_\varepsilon\in\mathbb{N}$ corresponding to a given $\varepsilon>0$ are the same for any $t\in\mathbb{Z}$.

We say that (1) satisfies the ***separation property (SP)*** if for two input signals $\{z(t)\}_{t\in\mathbb{Z}}$, $\{z'(t)\}_{t\in\mathbb{Z}}$ that differ only at some time point $s\in\mathbb{Z}$, that is, $z(s)\neq z'(s)$, the corresponding outputs satisfy that $\mathbf{x}(t)\neq\mathbf{x}'(t)$ for any $t\geq s$.

The proof of the following two results can be found in [18].

Theorem 5.2: Consider the reservoir model (9) driven by the real valued and non-necessarily stationary input signal $\{z(t)\}_{t\in\mathbb{Z}}$.

(i) Let $\mathbf{c}\in\mathbb{R}^N$ be an input mask and $\mathbf{I}(t):=\mathbf{c}z(t)$ the corresponding input forcing. Let

$$
F_I^R(\mathbf{I}(t),\mathbf{x}_0,\boldsymbol{\theta}) := \sum_{i=1}^{R}\frac{1}{i!}D_{\mathbf{I}}^{(i)}F(\mathbf{x}_0,\mathbf{0}_N,\boldsymbol{\theta})\overbrace{\mathbf{I}(t)\otimes\cdots\otimes\mathbf{I}(t)}^{i\text{ factors}}
$$

be the Rth-order Taylor series expansion of the reservoir map F at the point $(\mathbf{x}_0,\mathbf{0}_N,\boldsymbol{\theta})$ with respect to the input forcing $\mathbf{I}(t)$. Assume that one of the following conditions holds:

- **(a)** The map $F_I^R(\cdot,\mathbf{x}_0,\boldsymbol{\theta}):\mathbb{R}^N\to\mathbb{R}^N$ is injective.
- **(b)** The input signal is bounded, that is, there exists $k\in\mathbb{R}^+$ such that $|z(t)|<k$, for all $t\in\mathbb{Z}$, and the map F_I^R is injective in the set $\mathcal{B}=\left\{\mathbf{I}\in\mathbb{R}^N \mid \ \|\mathbf{I}\|<\|\mathbf{c}\|k\right\}$.

If additionally, the linear map $A(\mathbf{x}_0,\boldsymbol{\theta}) := D_{\mathbf{x}}F(\mathbf{x}_0,\mathbf{0}_N,\boldsymbol{\theta}):\mathbb{R}^N\to\mathbb{R}^N$ has no zero eigenvalues, then the reservoir model satisfies the separation property.

(ii) Suppose that the input signal $\{z(t)\}_{t\in\mathbb{Z}}$ is strictly stationary with finite automoments up to order $2R$ and that it is bounded, that is, there exists $k\in\mathbb{R}^+$ such that $|z(t)|<k$ for all $t\in\mathbb{Z}$. If additionally the linear map $A(\mathbf{x}_0,\boldsymbol{\theta})$ is such that $\|A(\mathbf{x}_0,\boldsymbol{\theta})\|<1$, with $\|\cdot\|$ some matrix norm induced from $\mathbb{R}^N$, then the reservoir model (9) satisfies the uniform fading memory property.

This result can be easily extended to multidimensional input signals, that is, $\{\mathbf{z}(t)\}_{t\in\mathbb{Z}}$, $\mathbf{z}(t)\in\mathbb{R}^n$. In that case (see [15] for the details) the RC is constructed by using an input mask $\mathbf{c}\in\mathbb{M}_{N,n}$ that takes care not only of the temporal, but also of the dimensional multiplexing by setting $\mathbf{I}(t):=\mathbf{c}\mathbf{z}(t)$. The only additional hypothesis needed in that situation is that the rank of $\mathbf{c}$ has to equal n in order to conclude part **(ii)** of the theorem.

The following result contains a statement analogous to that of Theorem 5.2 in the particular case of the time-delay reservoirs (see [14]). In that situation, some hypotheses are either automatically satisfied or can be formulated in a simplified manner.

Corollary 5.3: Consider a time-delay reservoir of the type considered in [14] with nonlinear kernel $f:\mathbb{R}\times\mathbb{R}\times\mathbb{R}^K\to\mathbb{R}$, parameters $\boldsymbol{\theta}\in\mathbb{R}^K$, and a non-necessarily stationary input signal $\{z(t)\}_{t\in\mathbb{Z}}$, $z(t)\in\mathbb{R}$.

(i) Let $\mathbf{c}\in\mathbb{R}^N$ be an input mask and $\mathbf{I}(t):=\mathbf{c}z(t)$ the corresponding input forcing. Let $f_I^R(I(t),x_0,\boldsymbol{\theta}) := \sum_{i=1}^{R}\frac{1}{i!}(\partial_I^{(i)}f)(x_0,0,\boldsymbol{\theta})I(t)^i$ be the Rth-order Taylor series expansion of the kernel map f at the point $(x_0,0,\boldsymbol{\theta})$ with respect to the input forcing $I(t)$. Assume that one of the following conditions holds:

- **(a)** The map $f_I^R(\cdot,x_0,\boldsymbol{\theta}):\mathbb{R}\to\mathbb{R}$ is injective;
- **(b)** The input signal is bounded, that is, there exists $k\in\mathbb{R}^+$ such that $|z(t)|<k$, for all $t\in\mathbb{Z}$, and the map $f_I^R(\cdot,x_0,\boldsymbol{\theta})$ is injective in the set $\mathcal{B}=\{I\in\mathbb{R}\mid|I|<\|C\|k\}$.

Then the corresponding TDR model satisfies the **(SP)**.

(ii) Suppose that the input signal $\{z(t)\}_{t\in\mathbb{Z}}$ is strictly stationary with finite automoments up to order $2R$ and that it is bounded, that is, there exists $k \in \mathbb{R}^+$ such that $|z(t)| < k$, for all $t \in \mathbb{Z}$. If the partial derivative $\partial_x f(x_0, 0, \boldsymbol{\theta})$ of the nonlinear kernel f evaluated at the point $(x_0, 0, \boldsymbol{\theta})$ satisfies the condition $|\partial_x f(x_0, 0, \boldsymbol{\theta})| < 1$, then the TDR model satisfies the **(UFMP)**.

VI. Example: Filtering of Autoregressive Stochastic Volatilities

In this example we consider the autoregressive stochastic volatility (ARSV) model [19] determined by the linear state-space prescription

$$\begin{cases} z(t) &= r + \sigma(t)\zeta(t), \qquad \{\zeta(t)\}_{t\in\mathbb{Z}} \sim \mathrm{IID}(0,1) \\ b(t) &= \lambda + \alpha b(t-1) + w(t), \; \{w(t)\}_{t\in\mathbb{Z}} \sim \mathrm{IID}(0, \sigma_w^2) \end{cases} \tag{20}$$

where $b(t) := \log(\sigma(t)^2)$, λ is a real parameter, and $\alpha \in (-1, 1)$. We will additionally assume that the innovations $\{\zeta(t)\}_{t\in\mathbb{Z}}$ and $\{w(t)\}_{t\in\mathbb{Z}}$ are independent. It is easy to prove that the unique stationary process $\{z(t)\}_{t\in\mathbb{Z}}$ induced by (20) and available in the presence of the constraint $\alpha \in (-1, 1)$ is a white noise (the returns have no autocorrelation) with finite moments of arbitrary order. Moreover, the unconditional variance σ_b^2 of the stationary process $\{b(t)\}$ is given by

$$\sigma_b^2 = \frac{\sigma_w^2}{1-\alpha^2},$$

and if the innovations $\{\zeta(t)\}$ and $\{w(t)\}$ are Gaussian, then the unconditional variance and kurtosis of the process $\{y(t)\}$ are given by

$$\mathrm{var}(z(t)) = \mathrm{E}[\sigma(t)^2] = \exp\left[\frac{\lambda}{1-\alpha} + \frac{1}{2}\sigma_b^2\right],$$

and

$$\mathrm{kurtosis}\,(z(t)) = 3\exp\left(\sigma_b^2\right).$$

Moreover, it can be shown [19] that whenever σ_b^2 is small and/or α is close to one then the autocorrelation $\gamma(h)$ of the squared returns at lag h can be approximated by

$$\gamma(h) \simeq \frac{\exp(\sigma_b^2) - 1}{3\exp(\sigma_b^2) - 1}\alpha^h.$$

The volatility process $\{\sigma(t)\}_{t\in\mathbb{Z}}$ is a non-observable, non-predictable stochastic latent variable that cannot be written as a function of previous realizations of the observable variable $z(t)$ and the volatilities $\sigma(t)$. Many procedures have been developed over the years to go around this difficulty whose solution is needed, in particular, to estimate the model parameters. In this section we will focus in only two them that are profusely used in the literature. First, the specific form of the prescription (20) corresponds to a state-space model in which the observation equation is the one that yields $\{z(t)\}_{t\in\mathbb{Z}}$ and the state equation rules the time evolution of $b(t) := \log(\sigma(t)^2)$. This observation makes appropriate the use of the Kalman filter [20] to obtain estimations of the conditional log-variances $b(t)$ based on the observed values $z(t)$. The other method that we will use as a benchmark is the hierarchical-likelihood method [21], [22], [23] (abbreviated in what follows as h-likelihood) that incorporates the unobserved volatilities as an unknown variable at the time of writing a likelihood that is optimized and that takes into account the observed time series values $z(t)$.

In the RC context, the problem of estimating the unobserved volatility $\sigma(t)$ out of the observed values of $z(t)$ up to time t, can be easily encoded as a filtering problemfor which the RC performance was studied in Section IV-B by using the reservoir model. Indeed, it suffices to take $z(t)$ as input signal and as teaching signal $y(t)$ the functional form of the volatility that we are interested in. Both the Kalman filter and the h-likelihood methods are designed to produced optimal (linear in the case of Kalman) estimations of the the log-variance $\log(\sigma(t)^2)$, which is a limitation to which RC is not exposed.

In the paragraphs that follow we carry out an empirical exercise in this context in order to compare the performance of the RC with that of Kalman and h-likelihood, and also to evaluate the accuracy of the capacity formulas introduced in Section IV and based on the reservoir model (9) at the time of estimating the performance of the actual RC.

We proceed by using a time-delay reservoir of the type considered in [14] constructed with the so-called Ikeda kernel map given by the expression:

$$f(x, I, \boldsymbol{\theta}) = \eta \sin^2\left(x + \gamma I + \phi\right), \quad \boldsymbol{\theta} := (\eta, \gamma, \phi) \in \mathbb{R}^3. \tag{21}$$

The architecture of the reservoir chosen contains 40 neurons and an input mask $\mathbf{c} \in \mathbb{R}^N$ that was randomly constructed with values uniformly distributed in the interval $[-1, 1]$.

We present to this TDR the filtering tasks consisting on estimating four different functions of the volatility $\sigma(t)$ generated by an ARSV model with parameters $r = 3.9 \cdot 10^{-4}$, $\sigma_w = 0.675$, $\lambda = -0.821$, and $\alpha = 0.9$. The four different teaching signals used are $y_1(t) := \sigma(t)$, $y_2(t) := \sigma(t)^2$, $y_3(t) := \log(\sigma(t))$, and $y_4(t) := \log(\sigma(t)^2)$. Given a fixed input mask $\mathbf{c}$, the reservoir parameters $\boldsymbol{\theta}$ are optimized with respect to each of these four filtering tasks. In this case, the optimal parameters were the same for the four cases, namely, $\gamma = 2.866$, $\phi = 1.124$, $\eta = 0.461$, and $d = 0.839$; we recall that $d := \tau/N$ is the separation between neurons. Table I presents the performances (in terms of the normalized mean square error (NMSE)) exhibited by the TDR in the execution of the four filtering tasks and compares them with those attained using the Kalman filter and the h-likelihood approaches. The figures in the table show that these two benchmarks outperform the RC at the time of filtering the functions of the volatility (logarithm) that they have been designed for but when it comes to providing the values of the actual instantaneous volatility or variance, it is the RC that performs the best.

Table I
PERFORMANCES (IN TERMS OF THE NORMALIZED MEAN SQUARE ERROR (NMSE)) EXHIBITED BY THE TDR IN THE EXECUTION OF FOUR VOLATILITY FILTERING TASKS COMPARED WITH THOSE ATTAINED USING THE KALMAN FILTER AND THE H-LIKELIHOOD APPROACHES.

Stochastic volatility filtering performance (NMSE)					
		Teaching signal proposed/Task solved			
		Instantaneous volatility	Instantaneous variance	log of Instantaneous volatility	log of Instantaneous variance
Filtering Method	h-likelihood	0.476	0.730	**0.411**	**0.411**
	Kalman	0.536	0.812	0.429	0.429
Reservoir Method	Reservoir computer (TDR)	**0.437**	**0.594**	0.655	0.655
	Reservoir model	0.453	0.661	0.652	0.601

We finally evaluate in the context of this filtering task the accuracy of the capacity formulas introduced in Section IV. Figure 1 depicts the error surfaces associated to the filtering of the instantaneous volatility $\sigma(t)$ of the same ARSV data generating process that we considered in the construction of Table I. The left panel has been computed using Monte Carlo simulations in order to empirically evaluate the filtering error of the Ikeda TDR as a function of the parameter η in (21) and of the distance between neurons. The right panel was obtained by evaluating the capacity formula introduced in Section II and based on the reservoir model (9) with the help of the elements introduced in Section IV-B and a nonlinearity of order $R = 8$. The two surfaces clearly resemble each other and, more importantly, exhibit their minima at virtually the same parameter values. This proves that, as it was already shown in [14], [15] for independent signals, that the theoretical model can be efficiently used to determine the optimal reservoir architecture in the presence of strictly stationary inputs.

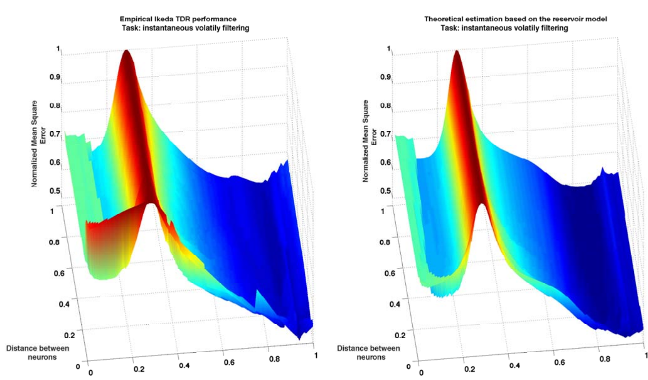

Figure 1. Error surfaces associated to the filtering of the instantaneous volatility $\sigma(t)$ an ARSV data generating process. The left panel has been computed using Monte Carlo simulations in order to empirically evaluate the filtering error of the Ikeda TDR as a function of η in (21) and of the distance between neurons. The right panel was obtained by evaluating the capacity formula based on the reservoir model (9) with a nonlinearity of order $R = 8$. The two surfaces have minima at virtually the same parameter values.

ACKNOWLEDGMENT

We acknowledge partial financial support of the Région de Franche-Comté (Convention 2013C-5493), the ANR "BIPHOPROC" project (ANR-14-OHRI-0002-02), and Deployment S.L. LG acknowledges financial support from the Faculty for the Future Program of the Schlumberger Foundation.

REFERENCES

[1] H. Jaeger, "The 'echo state' approach to analysing and training recurrent neural networks," *German National Research Center for Information Technology*, 2001.

[2] H. Jaeger and H. Haas, "Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.

[3] W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: a new framework for neural computation based on perturbations," *Neural Computation*, vol. 14, pp. 2531–2560, 2002.

[4] W. Maass, "Liquid state machines: motivation, theory, and applications," in *Computability In Context: Computation and Logic in the Real World*, S. S. Barry Cooper and A. Sorbi, Eds., 2011, ch. 8, pp. 275–296.

[5] N. Crook, "Nonlinear transient computation," *Neurocomputing*, vol. 70, pp. 1167–1176, 2007.

[6] D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt, "An experimental unification of reservoir computing methods," *Neural Networks*, vol. 20, pp. 391–403, 2007.

[7] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, vol. 3, no. 3, pp. 127–149, 2009.

[8] H. Jaeger, "Short term memory in echo state networks," *Fraunhofer Institute for Autonomous Intelligent Systems. Technical Report.*, vol. 152, 2002.

[9] O. White, D. Lee, and H. Sompolinsky, "Short-Term Memory in Orthogonal Neural Networks," *Physical Review Letters*, vol. 92, no. 14, p. 148102, apr 2004.

[10] S. Ganguli, D. Huh, and H. Sompolinsky, "Memory traces in dynamical systems." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 48, pp. 18 970–5, dec 2008.

[11] M. Hermans and B. Schrauwen, "Memory in linear recurrent neural networks in continuous time." *Neural networks : the official journal of the International Neural Network Society*, vol. 23, no. 3, pp. 341–55, apr 2010.

[12] J. Dambre, D. Verstraeten, B. Schrauwen, and S. Massar, "Information processing capacity of dynamical systems," *Scientific reports*, vol. 2, no. 514, 2012.

[13] I. B. Yildiz, H. Jaeger, and S. J. Kiebel, "Re-visiting the echo state property." *Neural networks : the official journal of the International Neural Network Society*, vol. 35, pp. 1–9, nov 2012.

[14] L. Grigoryeva, J. Henriques, L. Larger, and J.-P. Ortega, "Optimal nonlinear information processing capacity in delay-based reservoir computers," *Scientific Reports*, vol. 5, no. 12858, pp. 1–11, 2015.

[15] ——, "Nonlinear memory capacity of parallel time-delay reservoir computers in the processing of multidimensional signals," *Neural Computation*, vol. 28, pp. 1411–1451, 2016. [Online]. Available: http://arxiv.org/abs/1510.03891

[16] S. Ortin, L. Pesquera, and J. M. Gutiérrez, "Memory and nonlinear mapping in reservoir computing with two uncoupled nonlinear delay nodes," in *Proceedings of the European Conference on Complex Systems*, T. Gilbert, M. Kirkilionis, and G. Nicolis, Eds. Springer International Publishing Switzerland, 2012, pp. 895–899.

[17] L. Grigoryeva, J. Henriques, L. Larger, and J.-P. Ortega, "Stochastic time series forecasting using time-delay reservoir computers: performance and universality," *Neural Networks*, vol. 55, pp. 59–71, 2014.

[18] L. Grigoryeva, J. Henriques, and J.-P. Ortega, "Quantitative evaluation of the performance of discrete-time reservoir computers in the forecasting, filtering, and reconstruction of stochastic stationary signals," *Preprint*, 2015.

[19] S. J. Taylor, *Modelling Financial Time Series*. Chichester: John Wiley & Sons, 1986.

[20] A. C. Harvey, E. Ruiz, and N. Shephard, "Multivariate stochastic variance models," *Review of Economic Studies*, vol. 61, pp. 247–264, 1994.

[21] Y. Lee and J. A. Nelder, "Double hierarchical generalized linear models," *J. Roy. Statist. Soc. Ser. C*, vol. 55, no. 2, pp. 139–185, 2006. [Online]. Available: http://dx.doi.org/10.1111/j.1467-9876.2006.00538.x

[22] J. del Castillo and Y. Lee, "GLM-methods for volatility models," *Stat. Model.*, vol. 8, no. 3, pp. 263–283, 2008.

[23] J. Lim, L. Woojoo, Y. Lee, and J. del Castillo, "The hierarchical-likelihood approach to autoregressive stochastic volatility models," *Computational Statistics and Data Analysis*, vol. 55, no. 55, pp. 248–260, 2011.