Quantitative evaluation of the performance of discrete-time reservoir computers in the forecasting, filtering, and reconstruction of stochastic stationary signals

Lyudmila Grigoryeva[1], Julie Henriques[2,3], and Juan-Pablo Ortega[3,*]

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.

Key Words: Reservoir computing, echo state networks, liquid state machines, time-delay reservoir, memory capacity, forecasting, filtering, stationary signals, separation property, fading memory property.

1 Introduction

Reservoir computing is a recent but already well established neural computing paradigm [Jaeg 01, Jaeg 04, Maas 02, Maas 11, Croo 07, Vers 07, Luko 09] that has shown a significant potential in overcoming some of the limitations inherent to more standard Turing-type machines. This computation approach, also referred to in the literature as **Echo State Networks** and **Liquid State Machines**, 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 [Jaeg 02, Whit 04, Gang 08, Herm 10, Damb 12] as well as the study of different signal treatment properties (see [Yild 12, Luko 09] 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 [Grig 15b, Grig 15a] in the context of RCs constructed via the sampling of the solutions of a time-delay differential equation.

[1]Department of Mathematics and Statistics. Universität Konstanz. Box 146. D-78457 Konstanz. Germany. Lyudmila.Grigoryeva@uni-konstanz.de

[2]Cegos Deployment. 11, rue Denis Papin. F-25000 Besançon. jhenriques@deployment.org

[3]Corresponding author. Centre National de la Recherche Scientifique, Laboratoire de Mathématiques de Besançon, UMR CNRS 6623, Université de Franche-Comté, UFR des Sciences et Techniques. 16, route de Gray. F-25030 Besançon cedex. France. Juan-Pablo.Ortega@univ-fcomte.fr

These RCs are usually referred to as time-delay reservoirs (TDRs). More specifically, in [Grig 15b] 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 [Grig 15a] in order to be able to handle multidimensional input signals and real-time multitasking [Maas 11], 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 [Orti 12, Grig 14a], 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 [Grig 15b] 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 will *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 will use the generalization of the model in [Grig 15b] 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 (see [Yild 12, Luko 09] and references therein).

We conclude the paper with a section in which we show that several memory, forecasting, and filtering problems that appear profusely in the context of parametric stochastic time series models can be readily formulated and tackled via RC which, as we show, outperforms in some instances standard techniques in that setup.

Notation: column vectors are denoted by bold lower or upper case symbol like $\mathbf{v}$ or $\mathbf{V}$. We write $\mathbf{v}^\top$ to indicate the transpose of $\mathbf{v}$. Given a vector $\mathbf{v} \in \mathbb{R}^n$, we denote its entries by v_i, with $i \in \{1, \ldots, n\}$; we also write $\mathbf{v} = (v_i)_{i\in\{1,\ldots,n\}}$. The symbols $\mathbf{i}_n$ and $\mathbf{0}_n$ stand for the vectors of length n consisting of ones and zeros, respectively. We denote by $\mathbb{M}_{n,m}$ the space of real $n \times m$ matrices with $m, n \in \mathbb{N}$. When $n = m$, we use the symbol $\mathbb{M}_n$ to refer to the space of square matrices of order n. Given a matrix $A \in \mathbb{M}_{n,m}$, we denote its components by A_{ij} and we write $A = (A_{ij})$, with $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots m\}$. We write $\mathbb{I}_n$ and $\mathbb{O}_n$ to denote the identity matrix and the zero matrix of dimension n, respectively. We use $\mathbb{S}_n$ to indicate the subspace $\mathbb{S}_n \subset \mathbb{M}_n$ of symmetric matrices, that is, $\mathbb{S}_n = \left\{A \in \mathbb{M}_n \mid A^\top = A\right\}$. Finally, the symbols $\mathrm{E}[\cdot]$, $\mathrm{var}(\cdot)$, and $\mathrm{Cov}(\cdot, \cdot)$ denote the mathematical expectation, the variance, and the covariance, respectively.

Acknowledgments: 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.

2 Reservoir computing with stationary input signals

We start by spelling out in detail the main dynamical property of the input signals that we consider in this work. Let $\{z(t)\}_{t\in\mathbb{Z}}$ be a one-dimensional stochastic time series, that is, for any time $t \in \mathbb{Z}$, the value $z(t) \in \mathbb{R}$ is the realization of a univariate random variable.

Definition 2.1 *The time series $\{z(t)\}_{t\in\mathbb{Z}}$ is said to be **strictly stationary** if the joint distributions of the multivariate random variables $(z(t_1),\ldots,z(t_k))^\top$ and $(z(t_1+h),\ldots,z(t_k+h))^\top$ are the same for all positive integers k and for all $t_1,\ldots,t_k,h \in \mathbb{Z}$.*

Definition 2.2 *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*

$$\mu_z^{r_1,\ldots,r_k}(t,h_2,\ldots,h_k) := \mathrm{E}\left[z(t)^{r_1}z(t+h_2)^{r_2}\cdots z(t+h_k)^{r_k}\right], \tag{2.1}$$

together with the convention

$$\mu_z^{r_1}(t) = \mathrm{E}\left[z(t)^{r_1}\right].$$

It is straightforward to show that the symbols in (2.1) satisfy the following two reduction properties:

- If $h_i = 0$ then:

$$\mu_z^{r_1,\ldots,r_i,\ldots,r_k}(t,h_2,\ldots,h_i,\ldots,h_k) = \mu_z^{r_1+r_i,r_2,\ldots,r_{i-1},r_{i+1},\ldots,r_k}(t,h_2,\ldots,h_{i-1},h_{i+1},\ldots,h_k).$$

- If $h_i = h_j \neq 0$ with $i<j$ then:

$$\mu_z^{r_1,\ldots,r_k}(t,h_2,\ldots,h_k) = \mu_z^{r_1,\ldots,r_{i-1},(r_i+r_j),r_{i+1},\ldots,r_{j-1},r_{j+1},\ldots,r_k}(t,h_2,\ldots,h_i,\ldots,h_{j-1},h_{j+1},\ldots,h_k).$$

The following proposition is a direct consequence of Definition 2.1.

Proposition 2.3 *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 (2.1) by*

$$\mu_z^{r_1,\ldots,r_k}(h_2,\ldots,h_k) := \mathrm{E}\left[z(t)^{r_1}z(t+h_2)^{r_2}\cdots z(t+h_k)^{r_k}\right] \quad \textit{for any} \quad t\in\mathbb{Z}. \tag{2.2}$$

Remark 2.4 We emphasize that strict stationarity is a significant generalization of the standard notion of stationarity formulated in terms of the time-independence of the order one and two automoments [Broc 02] (mean and autocovariance functions), also referred to as **second order stationarity**. A case in which both notions coincide is when the process $\{z(t)\}_{t\in\mathbb{Z}}$ is Gaussian, that is, when for any $t_1,\ldots,t_k \in \mathbb{Z}$ and $k \in \mathbb{N}$, the distribution function of $(z(t_1),\ldots,z(t_k))$ is a multivariate Gaussian. ■

2.1 The RC setup for signal forecasting, filtering, and reconstruction

2.1.1 The tasks

In this work we study the performance of reservoir computing in the handling of three different signal processing tasks, namely **forecasting, filtering**, and **reconstruction**, that we now describe in detail. 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.

If we use the mean square error $\mathrm{E}\left[(y(t)-\bar{y}(t))^2\right]$ as a loss function to measure the difference between the machine output $\bar{y}(t)$ and the actual value $y(t)$ of the teaching signal that we seek to reproduce, a general result shows (see for example Section 4.1 in [Hami 94]) that this machine learning task amounts to a nonparametric estimation of the conditional expectation $\mathrm{E}[y(t)\,|\mathcal{F}_t]$, where $\mathcal{F}_t$ is the information set generated by the input signal up to time t, that is, $\mathcal{F}_t = \sigma(z(t), z(t-1), \dots)$; the symbol $\sigma(z(t), z(t-1), \dots)$ denotes the sigma-algebra generated by the random variables $\{z(t), z(t-1), \dots\}$. We will distinguish three different, but possibly overlapping situations, that will be illustrated later on in Section 3:

(i) Forecasting and reconstruction: in this case the teaching signal is a function, in general nonlinear, of the input signal. More specifically, 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 $y(t) = H(z(t+f), \dots, z(t), \dots, z(t-h))$ that depends on the value of the input signal f lags into the future (forecasting part) and h lags into the past (reconstruction part). Examples of this task are presented in Subsections 3.1 and 3.2.

(ii) Filtering: it is a generalization of the previous case in which the input and teaching signal exhibit statistical dependence but do not necessarily have a deterministic functional dependence. An example of this task is presented in Subsection 3.3.

2.1.2 The reservoir computing setup and its capacity

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

$$\mathbf{x}(t) = F(\mathbf{x}(t-1), \mathbf{I}(t), \boldsymbol{\theta}), \quad \text{with} \quad t\in\mathbb{Z},\ \mathbf{x}(t),\ \mathbf{I}(t)\in\mathbb{R}^N,\ \text{and}\ \boldsymbol{\theta}\in\mathbb{R}^K. \tag{2.3}$$

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 $\mathbf{I}(t) := \mathbf{c}z(t)$.

Example 2.5 Time-delay reservoirs (TDRs): This RC setup [Roda 11, Guti 12] is based on the sampling of the solutions of a time-delay differential equation (TDDE) of the form

$$\dot{x}(t) = -x(t) + f(x(t-\tau), I(t), \boldsymbol{\theta}), \tag{2.4}$$

with **time-delay** τ and where $f : \mathbb{R}\times\mathbb{R}\times\mathbb{R}^K \to \mathbb{R}$ is called the **kernel map**. A reservoir map $F : \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}^K \longrightarrow \mathbb{R}^N$ like in (2.3) can be constructed in this case by considering the Euler time-discretization of (2.4) with integration step $d := \tau/N$, with N the number of neurons of the desired RC setup, namely,

$$\frac{x(t)-x(t-d)}{d} = -x(t) + f(x(t-\tau), I(t), \boldsymbol{\theta}). \tag{2.5}$$

Next, we use an input mask $\mathbf{c}\in\mathbb{R}^N$ to multiplex the input signal over the delay period by setting $\mathbf{I}(t) := \mathbf{c}z(t) \in \mathbb{R}^N$. We then organize it, as well as the solutions of (2.5), in neuron layers $\mathbf{x}(t)$ parametrized by a discretized time $t\in\mathbb{Z}$ which yields

$$x_i(t) := e^{-\xi}x_{i-1}(t) + (1-e^{-\xi})f(x_i(t-1), I_i(t), \boldsymbol{\theta}),\ \text{with}\ x_0(t) := x_N(t-1),\ \text{and}\ \xi := \log(1+d), \tag{2.6}$$

where $x_i(t)$ and $I_i(t)$ stand for the ith-components of the vectors $\mathbf{x}(t)$ and $\mathbf{I}(t)$, respectively. The value d is referred to as the **separation between neurons**. The reservoir map (2.3) is obtained by using (2.6) in order to write down the neuron values of the layer for time t in terms of those for time $t-1$ and the current input signal value. More specifically:

$$\left\{\begin{array}{rcl} x_1(t) & = & e^{-\xi}x_N(t-1)+(1-e^{-\xi})f(x_1(t-1),I_1(t),\boldsymbol{\theta}),\\ x_2(t) & = & e^{-2\xi}x_N(t-1)+(1-e^{-\xi})\left\{e^{-\xi}f(x_1(t-1),I_1(t),\boldsymbol{\theta})+f(x_2(t-1),I_2(t),\boldsymbol{\theta})\right\},\\ & \vdots & \\ x_N(t) & = & e^{-N\xi}x_N(t-1)+(1-e^{-\xi})\sum_{j=0}^{N-1}e^{-j\xi}f(x_{N-j}(t-1),I_{N-j}(t),\boldsymbol{\theta}),\end{array}\right. \tag{2.7}$$

which corresponds to a description of the form

$$\mathbf{x}(t)=F(\mathbf{x}(t-1),\mathbf{I}(t),\boldsymbol{\theta}), \tag{2.8}$$

that uniquely determines the reservoir map $F:\mathbb{R}^N\times\mathbb{R}^N\times\mathbb{R}^K\longrightarrow\mathbb{R}^N$. Physical implementations of this scheme carried out with dedicated hardware are already available and have shown excellent performances in the processing of empirical data: spoken digit recognition [Jaeg 07, Appe 11, Larg 12, Paqu 12, Brun 13], the NARMA model identification task [Atiy 00, Roda 11], continuation of chaotic time series, and volatility forecasting [Grig 14a]. ■

As we explained in the previous subsection, a 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

$$(\mathbf{W}_{\text{out}},a_{\text{out}}):=\underset{\mathbf{W}\in\mathbb{R}^N,a\in\mathbb{R}}{\arg\min}\left(\mathrm{E}\left[\left(\mathbf{W}^\top\mathbf{x}(t)+a-y(t)\right)^2\right]+\lambda\left\|\mathbf{W}\right\|^2\right),\quad \lambda\in\mathbb{R}, \tag{2.9}$$

whose solution is given by

$$\mathbf{W}_{\text{out}}=(\Gamma(0)+\lambda\mathbb{I}_N)^{-1}\mathrm{Cov}(y(t),\mathbf{x}(t)), \tag{2.10}$$

$$a_{\text{out}}=\mathrm{E}\left[y(t)\right]-\mathbf{W}_{\text{out}}^\top\boldsymbol{\mu}_x. \tag{2.11}$$

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

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

is the lag-zero auto covariance exhibited by the reservoir output. We show in the next subsection 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:

$$\begin{aligned}&\mathrm{E}\left[\left(\mathbf{W}_{\text{out}}^\top\cdot\mathbf{x}(t)+a_{\text{out}}-y(t)\right)^2\right]=\mathbf{W}_{\text{out}}^\top\Gamma(0)\mathbf{W}_{\text{out}}+\mathrm{var}\left(y(t)\right)-2\mathbf{W}_{\text{out}}^\top\mathrm{Cov}(y(t),\mathbf{x}(t))\\ &=\mathrm{var}\left(y(t)\right)-\mathrm{Cov}(y(t),\mathbf{x}(t))^\top(\Gamma(0)+\lambda\mathbb{I}_N)^{-1}(\Gamma(0)+2\lambda\mathbb{I}_N)(\Gamma(0)+\lambda\mathbb{I}_N)^{-1}\mathrm{Cov}(y(t),\mathbf{x}(t)).\end{aligned} \tag{2.12}$$

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

$$C(\boldsymbol{\theta},\mathbf{c},\lambda):=\frac{\mathrm{Cov}(y(t),\mathbf{x}(t))^\top(\Gamma(0)+\lambda\mathbb{I}_N)^{-1}(\Gamma(0)+2\lambda\mathbb{I}_N)(\Gamma(0)+\lambda\mathbb{I}_N)^{-1}\mathrm{Cov}(y(t),\mathbf{x}(t))}{\mathrm{var}\left(y(t)\right)}. \tag{2.13}$$

We emphasize that this capacity is a natural generalization to the context of non-independent stationary input signals of the notion introduced in [Jaeg 02, Whit 04, Gang 08, Herm 10, Damb 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 worth noting that since the normalized error coming from (2.12) is bounded between zero and one, it is clear that $0 \leq C(\boldsymbol{\theta}, \mathbf{c}, \lambda) \leq 1$.

2.2 The reservoir model

The capacities (2.13) for a reservoir of the form (2.3) are in general very difficult to compute analytically. The standard approach to determine them consists hence in fixing a triple $(\boldsymbol{\theta}, \mathbf{c}, \lambda)$ and in estimating the corresponding reservoir capacity $C(\boldsymbol{\theta}, \mathbf{c}, \lambda)$ via Monte Carlo simulations. This strategy makes the finding of the optimal parameters for a given task computationally very expensive.

In [Grig 15b] we introduced an approximate model for the time-delay reservoirs (TDRs) presented in Example 2.5 that made possible an analytic estimation of their capacities under a very strong independence hypothesis in the input signal; this condition was already present in the original definitions of this notion [Jaeg 02, Whit 04, Gang 08, Herm 10, Damb 12]. These developments provided a functional link between the TDR parameters and its performance with respect to a given reconstruction task which was used to accurately determine the optimal reservoir architecture by solving a well posed optimization problem.

In this section we extend that construction to more general RCs driven by strictly stationary input signals (see Definition 2.1). The approximate model of the RC in (2.3) is obtained, as in [Grig 15b], 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 (2.3). This condition means that 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 $\rho\left(A(\mathbf{x}_0, \boldsymbol{\theta})\right) < 1$, 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 [Grig 15b] 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 approximating the reservoir (2.3) by the series expansion:

$$\begin{aligned}\mathbf{x}(t) &= F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) + D_{\mathbf{x}} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta})(\mathbf{x}(t-1) - \mathbf{x}_0) + \sum_{i=1}^{R} \frac{1}{i!} D_{\mathbf{I}}^{(i)} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \overbrace{\mathbf{I}(t) \otimes \cdots \otimes \mathbf{I}(t)}^{i \text{ factors}} \\ &= \mathbf{x}_0 + A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t-1) - \mathbf{x}_0) + \boldsymbol{\varepsilon}(t), \end{aligned} \tag{2.14}$$

where $\boldsymbol{\varepsilon}(t) \in \mathbb{R}^N$ is a vector whose entries are polynomial functions of the input signal $z(t)$ at t. More specifically

$$\begin{aligned}\boldsymbol{\varepsilon}(t) &= \sum_{i=1}^{R} \frac{1}{i!} D_{\mathbf{I}}^{(i)} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \mathbf{I}(t) \otimes \cdots \otimes \mathbf{I}(t) = \sum_{i=1}^{R} \frac{1}{i!} D_{\mathbf{I}}^{(i)} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \left(\mathbf{c} z(t)\right) \otimes \cdots \otimes \left(\mathbf{c} z(t)\right) \\ &= \sum_{i=1}^{R} \frac{z(t)^i}{i!} D_{\mathbf{I}}^{(i)} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \underbrace{\left(\mathbf{c} \otimes \cdots \otimes \mathbf{c}\right)}_{i \text{ factors}} = \left(q_R^1\left(z(t), \mathbf{c}\right), \ldots, q_R^N\left(z(t), \mathbf{c}\right)\right)^{\top}, \end{aligned} \tag{2.15}$$

where $q_R^j(\cdot, \mathbf{c})$ is a polynomial of degree $R \in \mathbb{N}$ whose monomial of order i has as coefficient the value $\frac{1}{i!} D_{\mathbf{I}}^{(i)} F_j(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \underbrace{\left(\mathbf{c} \otimes \cdots \otimes \mathbf{c}\right)}_{i \text{ factors}}$ with F_j is the j-th component of the map $F := (F_1, \ldots, F_N)$ in (2.3).

In what follows, we will refer to the recursion

$$\mathbf{x}(t) = \mathbf{x}_0 + A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t-1) - \mathbf{x}_0) + \boldsymbol{\varepsilon}(t) \tag{2.16}$$

as the **approximate reservoir model** or just the **reservoir model**. We now notice that the strict stationarity of $\{z(t)\}_{t\in\mathbb{Z}}$ implies that of $\{\boldsymbol{\varepsilon}(t)\}_{t\in\mathbb{Z}}$. In particular,

$$\boldsymbol{\mu}_{\boldsymbol{\varepsilon}} := \mathrm{E}\left[\boldsymbol{\varepsilon}(t)\right] = \left(\left(q_R^1\left(x,\mathbf{c}\right)\right)\left(\mu_z^{\cdot}\right),\cdots,\left(q_R^N\left(x,\mathbf{c}\right)\right)\left(\mu_z^{\cdot}\right)\right)^{\top}, \tag{2.17}$$

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 $i,j\in\{1,\ldots,N\}$:

$$\left(\Gamma_{\boldsymbol{\varepsilon}}(h)\right)_{i,j} = \mathrm{E}\left[\boldsymbol{\varepsilon}^i(t)\boldsymbol{\varepsilon}^j(t+h)\right] - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^i\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^j = \left(q_R^i\left(x,\mathbf{c}\right)\bullet q_R^j\left(y,\mathbf{c}\right)\right)\left(\mu_z^{\cdot,\cdot}(h)\right) - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^i\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}^j, \tag{2.18}$$

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, whose proof can be found in the appendix, 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 (2.16).

Proposition 2.6 *Let $\{\mathbf{x}(t)\}_{t\in\mathbb{Z}}$ be the output of the reservoir model (2.16). 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 (2.16)). Under those hypotheses, the reservoir output $\{\mathbf{x}(t)\}_{t\in\mathbb{Z}}$ is second order stationary (see Remark 2.4) and the corresponding time-independent mean $\boldsymbol{\mu}_{\mathbf{x}}$ and autocovariances Γ are given by:*

$$\boldsymbol{\mu}_{\mathbf{x}} := \mathrm{E}\left[\mathbf{x}(t)\right] = \mathbf{x}_0 + \left(\mathbb{I}_N - A(\mathbf{x}_0,\boldsymbol{\theta})\right)^{-1}\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}, \tag{2.19}$$

$$\Gamma(h) := \mathrm{E}\left[\left(\mathbf{x}(t)-\boldsymbol{\mu}_{\mathbf{x}}\right)\left(\mathbf{x}(t+h)-\boldsymbol{\mu}_{\mathbf{x}}\right)^{\top}\right] = \sum_{j,k=0}^{\infty} A^j\Gamma_{\boldsymbol{\varepsilon}}\left(k-j-h\right)\left(A^k\right)^{\top}, \quad h\in\mathbb{Z}, \tag{2.20}$$

with $\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}$ and $\Gamma_{\boldsymbol{\varepsilon}}$ provided by (2.17) and (2.18), respectively. Under these hypotheses, the recursion (2.16) that determines the reservoir model can be rewritten as

$$\left(\mathbf{x}(t)-\boldsymbol{\mu}_{\mathbf{x}}\right) = A(\mathbf{x}_0,\boldsymbol{\theta})(\mathbf{x}(t-1)-\boldsymbol{\mu}_{\mathbf{x}}) + \left(\boldsymbol{\varepsilon}(t)-\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right), \tag{2.21}$$

and

$$\mathbf{x}(t) = \boldsymbol{\mu}_{\mathbf{x}} + \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j\left(\boldsymbol{\varepsilon}(t-j)-\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right), \tag{2.22}$$

is the unique stationary solution of (2.21).

2.3 Reservoir capacity estimations for signal forecasting, reconstruction, and filtering

We now use the reservoir model introduced in the previous section in order to provide estimates for the different information processing tasks that we described in Section 2.1.1. 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 2.6.

2.3.1 Linear and quadratic forecasting and reconstruction

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

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

with $\mathbf{L} \in \mathbb{R}^{f+h+1}$. We construct the teaching signal by setting $y(t) := \mathbf{L}^\top \mathbf{z}^{f,h}(t)$. Notice first that $\mu_y := \mathrm{E}[y(t)] = \mu_z \mathbf{L}^\top \mathbf{i}_{f+h+1}$. We now estimate the memory capacity $C_H(\boldsymbol{\theta}, \mathbf{c}, \lambda)$ associated to the task H and exhibited by the reservoir model (2.21). Notice that the evaluation of the expression (2.13) requires the computation of the lag-zero autocovariance $\Gamma(0)$ of the reservoir output in terms of the reservoir parameters, as well as $\mathrm{var}(y(t))$ and $\mathrm{Cov}(y(t), \mathbf{x}(t))$. The expression for $\Gamma(0)$ is explicitly provided by (2.20); regarding $\mathrm{var}(y(t))$ and $\mathrm{Cov}(y(t), \mathbf{x}(t))$ we have:

- $\mathrm{var}(y(t)) = \mathrm{E}\left[y(t)^2\right] - \mu_y^2 = \mathbf{L}^\top \mathrm{E}\left[\mathbf{z}^{f,h}(t)\mathbf{z}^{f,h}(t)^\top\right]\mathbf{L} - \mu_y^2 = \mathbf{L}^\top(\Gamma^z - \mu_z^2 \mathbf{i}_{f+h+1}\mathbf{i}_{f+h+1}^\top)\mathbf{L}$, with $\Gamma^z \in \mathbb{S}_{f+h+1}$ defined by

$$\Gamma^z_{i,j} = \mu_z^{1,1}(i-j), \quad \text{with} \quad i,j \in \{1, \ldots, f+h+1\}, \tag{2.23}$$

  and $\mu_z^{1,1}$ the second order automoment of the input signal.

- $\mathrm{Cov}(y(t), \mathbf{x}(t))$: consider the representation in (2.22) of the unique stationary solution of the reservoir model, that is,

$$(\mathbf{x}(t) - \boldsymbol{\mu}_{\mathbf{x}}) = \sum_{j=0}^{\infty} A(\mathbf{x}_0, \boldsymbol{\theta})^j \boldsymbol{\rho}(t-j), \tag{2.24}$$

  with $\boldsymbol{\rho}(t) := \boldsymbol{\varepsilon}(t) - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}$. Consequently

$$\begin{aligned}
\mathrm{Cov}(y(t), \mathbf{x}(t)) &= \sum_{j=1}^{f+h+1} \mathrm{Cov}\left(L_j z(t+f+1-j), \mathbf{x}(t)\right) \\
&= \sum_{j=1}^{f+h+1} \sum_{k=0}^{\infty} L_j A(\mathbf{x}_0, \boldsymbol{\theta})^k \mathrm{E}\left[\left(z(t+f+1-j) - \mu_z\right)\left(\boldsymbol{\varepsilon}(t-k) - \boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right)\right] \\
&= \sum_{j=1}^{f+h+1} \sum_{k=0}^{\infty} L_j A(\mathbf{x}_0, \boldsymbol{\theta})^k \left[\begin{pmatrix} \left(x \bullet q_R^1(y, \mathbf{c})\right)\left(\mu_z^{1,\cdot}(f+k+1-j)\right) \\ \vdots \\ \left(x \bullet q_R^N(y, \mathbf{c})\right)\left(\mu_z^{1,\cdot}(f+k+1-j)\right) \end{pmatrix} - \mu_z \boldsymbol{\mu}_{\boldsymbol{\varepsilon}} \right],
\end{aligned} \tag{2.25}$$

  where this expression has been written using the same convention as in (2.18).

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

$$y(t) := H\left(\mathbf{z}^{f,h}(t)\right) = \sum_{i,j=1}^{f+h+1} Q_{i,j} z(t+f+1-i) z(t+f+1-j). \tag{2.26}$$

This implies that

$$\mu_y := \mathrm{E}[y(t)] = \sum_{i,j=1}^{f+h+1} Q_{i,j} \mu_z^{1,1}(i-j). \tag{2.27}$$

At the same time

$$y(t)^2 = \sum_{i,j,k,l=1}^{f+h+1} Q_{i,j}Q_{k,l}z(t+f+1-i)z(t+f+1-j)z(t+f+1-k)z(t+f+1-l), \tag{2.28}$$

and hence

$$\mathrm{E}\left[y(t)^2\right] = \sum_{i,j,k,l=1}^{f+h+1} Q_{i,j}Q_{k,l}\mu_z^{1,1,1,1}(i-j,i-k,i-l). \tag{2.29}$$

Consequently, by (2.27) and (2.29),

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

In order to compute $\mathrm{Cov}\left(y(t),\mathbf{x}(t)\right)$, we use again the representation (2.24) and hence

$$\begin{aligned}
&\mathrm{Cov}\left(y(t),\mathbf{x}(t)\right) = \mathrm{Cov}\left(y(t),\mathbf{x}(t)-\boldsymbol{\mu}_{\mathbf{x}}\right) = \sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k \mathrm{Cov}\left(y(t),\boldsymbol{\rho}(t-k)\right)\\
&= \sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k \left[\mathrm{E}\left[y(t)\boldsymbol{\varepsilon}(t-k)\right] - \mu_y\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right]\\
&= \sum_{k=0}^{\infty}\sum_{i,j=1}^{f+h+1} A(\mathbf{x}_0,\boldsymbol{\theta})^k Q_{i,j}\mathrm{E}\left[z(t+f+1-i)z(t+f+1-j)\boldsymbol{\varepsilon}(t-k)\right] - \mu_y\sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\\
&= \sum_{k=0}^{\infty}\sum_{i,j=1}^{f+h+1} Q_{i,j}A(\mathbf{x}_0,\boldsymbol{\theta})^k \begin{pmatrix} \left(x\bullet y\bullet q_R^1\left(z,\mathbf{c}\right)\right)\left(\mu_z^{1,1,\cdot}(i-j,i-k-f-1)\right)\\ \vdots \\ \left(x\bullet y\bullet q_R^N\left(z,\mathbf{c}\right)\right)\left(\mu_z^{1,1,\cdot}(i-j,i-k-f-1)\right)\end{pmatrix} - \mu_y\sum_{k=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^k\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}.
\end{aligned} \tag{2.30}$$

2.3.2 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 $\mathrm{E}\left[y(t)\,|\mathcal{F}_t\right]$, where $\mathcal{F}_t$ is the information set generated by the input signal up to time t, that is, $\mathcal{F}_t = \sigma\left(z(t),z(t-1),\dots\right)$. As we already explained in Subsection 2.1.1, this conditional expectation minimizes the mean square error committed by the RC at the time of reproducing the teaching signal. We start by introducing the following definition.

Definition 2.7 *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*

$$\mu_{y,z}^r(t,h) := \mathrm{E}\left[y(t)z(t+h)^r\right]. \tag{2.31}$$

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*

$$\mu_{y,z}^r(h) := \mathrm{E}\left[y(t)z(t+h)^r\right], \quad \textit{for any } t\in\mathbb{Z}. \tag{2.32}$$

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 using (2.13) and by noting, first, that $\mathrm{var}(y(t))$ can be estimated out of the teaching signal and second, that by (2.24):

$$\begin{aligned}\mathrm{Cov}\left(y(t),\mathbf{x}(t)\right) &= \mathrm{Cov}\left(y(t),\mathbf{x}(t)-\boldsymbol{\mu}_{\mathbf{x}}\right) = \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j \mathrm{Cov}\left(y(t),\boldsymbol{\varepsilon}(t-j)-\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right)\\ &= \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j \mathrm{Cov}\left(y(t),\boldsymbol{\varepsilon}(t-j)\right) = \sum_{j=0}^{\infty} A(\mathbf{x}_0,\boldsymbol{\theta})^j \left[\begin{pmatrix}\left(x\bullet q_R^1(u,\mathbf{c})\right)\left(\mu_{y,z}^{\cdot}(-j)\right)\\ \vdots\\ \left(x\bullet q_R^N(u,\mathbf{c})\right)\left(\mu_{y,z}^{\cdot}(-j)\right)\end{pmatrix} - \mu_z\boldsymbol{\mu}_{\boldsymbol{\varepsilon}}\right],\end{aligned} \tag{2.33}$$

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 (2.33). When this expression, together with the estimate of $\mathrm{var}\left(y(t)\right)$ is substituted in (2.13), we obtain an estimate of the RC capacity for any value of its parameters $\boldsymbol{\theta}$ and the input mask $\mathbf{c}$.

2.4 The fading memory and the separation properties

The fading memory and the separation properties that we describe later on in this section have been identified in the context of reservoir computing to be in relation with good information processing performances (see [Yild 12, Luko 09] and references therein). The goal of the following paragraphs is showing that the reservoir model (2.16) for the discrete-time reservoir computer (2.3) exhibits these features under reasonable assumptions on the reservoir map.

Definition 2.8 *Consider the discrete-time reservoir map* (2.3).

(i) *We say that the reservoir map* (2.3) *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* $t\in\mathbb{Z}$.

(ii) *We say that* (2.3) *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 the appendices at the end of the paper.

Theorem 2.9 *Consider the reservoir model* (2.16) *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*

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

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 $\mathcal{B} = \{\mathbf{I} \in \mathbb{R}^N | \ \|\mathbf{I}\| < \|\mathbf{c}\|k\}$.*

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 (2.16) satisfies the uniform fading memory property.*

Remark 2.10 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 [Grig 15a] 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 2.9 in the particular case of the time-delay reservoirs introduced in Example 2.5. In that situation, some hypotheses are either automatically satisfied or can be formulated in a simplified manner.

Corollary 2.11 *Consider a time-delay reservoir of the type introduced in Example 2.5 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)**.

3 Examples

This section describes examples in connection with the three different tasks listed in Subsection 2.1.1, namely, forecasting, reconstruction, and filtering. The first two examples take place in the context of two parametric stochastic time series families (ARMA and GARCH). As we will see, forecasting in

those two setups comes down, in the terminology introduced in Section 2.3, to solving a linear and a quadratic forecasting task for which the performance of RC has been already empirically evaluated in [Grig 14a]. The last example falls in the category of pure filtering. In that case we will show how a specific type of RC is capable of outperforming two standard filtering techniques (Kalman filtering and the hierarchical-likelihood approach). Additionally, we will evaluate the accuracy of the capacity formulas introduced in Section 2.3 and based on the reservoir model (2.16) at the time of estimating the performance of the actual RC.

3.1 Multistep forecast of ARMA temporal aggregates

Consider the causal and invertible ARMA(p,q) specification (see [Box 76, Broc 06], and references therein for details) determined by the equivalent relations

$$\mathbf{\Phi}(L)z(t) = \mathbf{\Theta}(L)\zeta(t),\ z(t) = \sum_{i=0}^{\infty}\psi_i\zeta(t-i),\ \zeta(t) = \sum_{j=0}^{\infty}\pi_j z(t-j), \quad \{\zeta(t)\} \sim \mathrm{IID}(0,\sigma^2), \tag{3.1}$$

where L is the backward shift operator defined by $L(z(t)) := z(t-1)$, $\mathbf{\Phi}(z) := 1-\phi_1 z - \cdots - \phi_p z^p$, $\mathbf{\Theta}(z) := 1+\theta_1 z+\cdots+\theta_q z^q$, $\mathbf{\Psi}(z) := \mathbf{\Phi}(z)^{-1}\mathbf{\Theta}(z)$, $\mathbf{\Pi}(z) := \mathbf{\Theta}(z)^{-1}\mathbf{\Phi}(z)$, and the symbols $\mathbf{\Phi}(L)$ and $\mathbf{\Theta}(L)$ stand for the operators $\mathbf{\Phi}(L) := 1-\phi_1 L - \cdots - \phi_p L^p$ and $\mathbf{\Theta}(L) := 1+\theta_1 L + \cdots + \theta_q L^q$, respectively. Consider now an **aggregation vector** $\mathbf{w} \in \mathbb{R}^f$. It can be shown [Grig 14b, Grig 15c] that given a realization $\mathbf{z}_T$ of the process $\{z(t)\}_{t\in}$ up to time T, the best forecast $\widehat{z^{\mathbf{w}}}(T+f)$ (in the sense that it minimizes the mean square forecasting error) of the temporal aggregate $z^{\mathbf{w}}(T+f) = \sum_{i=1}^{f} w_{f-i+1}z(T+i)$ based in the information set generated by $\mathbf{z}_T$, is given by

$$\widehat{z^{\mathbf{w}}}(T+f) = \sum_{i=1}^{f}\sum_{j=1}^{T+i-1+r} w_{f-i+1}\psi_j\zeta(T+i-j) = \sum_{i=1}^{f}\sum_{j=1}^{T+i-1+r}\sum_{k=0}^{\infty} w_{f-i+1}\psi_j z(T+i-j-k), \tag{3.2}$$

with $r := \max\{p,q\}$. The mean square forecasting error MSFE $\left(\widehat{z^{\mathbf{w}}}(T+f)\right)$ associated to this forecast is [Grig 14b]:

$$\begin{aligned}\mathrm{MSFE}\left(\widehat{z^{\mathbf{w}}}(T+f)\right) &= E\left[\left(\widehat{z^{\mathbf{w}}}(T+f) - z^{\mathbf{w}}(T+f)\right)^2\right] \\ &= \sigma^2\left[\sum_{i=1}^{f} w_{f-i+1}^2\sum_{l=0}^{i-1}\psi_l^2 + 2\sum_{i=1}^{f-1}\sum_{j=i+1}^{f} w_{f-i+1}w_{f-j+1}\sum_{l=0}^{i-1}\psi_l\psi_{j-i+l}\right].\end{aligned}$$

This task can be solved via RC by using as input signal either the innovations $\zeta(t)$ or the time series values $z(t)$ up to time T and using as teaching signal the values $z^{\mathbf{w}}(t)$, also up to time T. In the terminology introduced in Section 2.3, in both cases this forecasting problem amounts to a linear task. For example if $z(t)$ is used as teaching signal, the linear task map $H : \mathbb{R}^f \to \mathbb{R}$ is given by $\mathbf{z}^f(t) = (z(t+f),\ldots,z(t+1)) \longmapsto \mathbf{w}^\top\mathbf{z}^f(t)$

The RC performance can be evaluated in this case by using the formula (2.13) and the elements introduced in Section 2.3.1, as long as the necessary automoments exist. That is the case whenever the innovations $\{\zeta(t)\}_{t\in\mathbb{Z}}$ are Gaussian because in that situation the input signal $\{z(t)\}_{t\in\mathbb{Z}}$ is a Gaussian process (see [Broc 06]) and hence all the automoments are finite and can be readily computed [Holm 88, Tria 03].

3.2 Multistep forecast of temporally aggregated GARCH volatilities

In this section we study the forecasting of the volatility associated to a flow-type aggregated sample generated by a GARCH(1,1) process [Engl 82, Boll 86] and we show that it can be encoded as a quadratic forecasting task of the type described in Section 2.3.1. More specifically, consider the process $\{z(t)\}_{t\in\mathbb{Z}}$ determined by

$$z(t) = \sigma(t)\zeta(t) \quad \text{with} \quad \zeta(t) \sim \text{IID}(0,\sigma^2) \quad \text{and} \quad \sigma^2(t) = \alpha_0 + \alpha_1 z(t-1)^2 + \beta\sigma(t-1)^2, \tag{3.3}$$

where $\sigma(t)$ stands for the positive square root of $\sigma^2(t)$. The constants α_0, α_1 and β are positive real numbers subjected to the constraints $\alpha_0 > 0$, $\alpha_1, \beta \geq 0$ and $\alpha_1 + \beta < 1$ that ensure the existence of a unique stationary solution and the positivity of the conditional variance process $\{\sigma(t)^2\}_{t\in\mathbb{Z}}$ that is by construction predictable. GARCH processes are profusely used in the financial econometrics literature in the modeling of the conditional volatility associated to the time evolution of asset returns.

A problem of much importance in financial risk and portfolio management applications is the forecasting of the variance $\text{var}_T\left(z_{T+f}[f]\right)$ of the flow aggregate $z_{T+f}[f] := z(T+1) + \cdots + z(T+f)$ based on the information set $\mathcal{F}_T$ generated by a realization $\mathbf{z}_T$ of the process $\{z(t)\}_{t\in\mathbb{Z}}$ up to time T and the conditional volatility $\sigma(T)$. It can be shown [Henr 14] that in the GARCH(1,1) context this forecast is given by

$$\text{var}_T\left(z_{T+f}[f]\right) = \text{E}\left[z_{T+f}[f]^2 \mid \mathcal{F}_T\right] \frac{f\alpha_0}{1-(\alpha_1+\beta)} + \left(\sigma(T+1)^2 - \frac{\alpha_0}{1-(\alpha_1+\beta)}\right) \frac{1-(\alpha_1+\beta)^f}{1-(\alpha_1+\beta)}, \tag{3.4}$$

where $\sigma(T+1) = \left(\alpha_0 + \alpha_1 z(T)^2 + \beta\sigma(T)^2\right)^{\frac{1}{2}}$. The estimation of this forecast can be solved via RC by using $\{z(t)\}_{t\in\mathbb{Z}}$ as input signal and corresponding teaching signal

$$y(t) := \left(z(t+1) + \cdots + z(t+f)\right)^2 = \sum_{k_1+k_2+\cdots+k_f=2} \binom{2}{k_1, k_2, \ldots, k_f} z(t+1)^{k_1} z(t+2)^{k_2} \cdots z(t+f)^{k_m}, \tag{3.5}$$

where the summation in the second equality is taken over all sequences of nonnegative integer indices $k_1, \ldots, k_f$, such that the sum $k_1 + \cdots + k_f = 2$. This can be encoded as a quadratic forecasting task with a task matrix $Q \in \mathbb{S}_f$ whose coefficients are given by the expression (3.5). The evaluation of the RC performance in this task using the formula (2.13) and Section 2.3.1 is only possible when the higher order automoments of the process (3.3) exist. This feature is by no means guaranteed in the GARCH context and imposes various semialgebric constraints on the coefficients α_0, α_1, β. See [Li 01, Ling 02] for a characterization of the higher order moments of the GARCH family.

Time-delay reservoir (TDR) computers have shown in [Grig 14a] excellent empirical performances at the time of carrying out this task when using as data generating process the VEC-GARCH family introduced in [Boll 88], which is a multivariate generalization of the GARCH family. In that work, TDRs were also shown to outperform standard parametric multivariate volatility models in the forecasting of actual market realized volatility.

3.3 Filtering of autoregressive stochastic volatilities

In this example we consider the autoregressive stochastic volatility (ARSV) model [Tayl 86] determined by the linear state-space prescription

$$\left\{ \begin{array}{rcll} z(t) & = & r + \sigma(t)\zeta(t), & \{\zeta(t)\}_{t\in\mathbb{Z}} \sim \text{IID}(0,1) \\ b(t) & = & \lambda + \alpha b(t-1) + w(t), & \{w(t)\}_{t\in\mathbb{Z}} \sim \text{IID}(0,\sigma_w^2) \end{array} \right. \tag{3.6}$$

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 (3.6) 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

$$\operatorname{var}(z(t)) = \mathrm{E}[\sigma(t)^2] = \exp\left[\frac{\lambda}{1-\alpha} + \frac{1}{2}\sigma_b^2\right], \quad \text{and} \quad \operatorname{kurtosis}(z(t)) = 3\exp\left(\sigma_b^2\right). \tag{3.7}$$

Moreover, it can be shown [Tayl 86] 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 main difference of the ARSV model (3.6) with respect to the GARCH family is that, in this case, 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 (3.6) 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 [Harv 94] 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 [Lee 96, Lee 06, Cast 08, Lim 11] (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 problem of the type characterized in Section 2.1.1 and for which the RC performance was studied in Section 2.3.2 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 2.3 and based on the reservoir model (2.16) at the time of estimating the performance of the actual RC.

We proceed by using a time-delay reservoir of the type described in the Example 2.5 constructed with the so-called Ikeda kernel map given by the expression:

$$f(x, I, \boldsymbol{\theta}) = \eta \sin^2(x + \gamma I + \phi), \quad \boldsymbol{\theta} := (\eta, \gamma, \phi) \in \mathbb{R}^3. \tag{3.8}$$

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 1 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.

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

Table 1: 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.

We finally evaluate in the context of this filtering task the accuracy of the capacity formulas introduced in Section 2.3. 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 1. 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 (3.8) and of the distance between neurons. The right panel was obtained by evaluating the formula (2.13) based on the reservoir model (2.16) with the help of the elements introduced in Section 2.3.2 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 [Grig 15b, Grig 15a] for independent signals, that the theoretical model can be efficiently used to determine the optimal reservoir architecture in the presence of strictly stationary inputs.

4 Appendices

4.1 Proof of Proposition 2.6

We start by emphasizing that the potential lack of independence in the input signal $\{z(t)\}_{t\in\mathbb{Z}}$ implies that $\{\boldsymbol{\varepsilon}(t)\}_{t\in\mathbb{Z}}$ is in general not a white noise. Consequently, unlike the situation encountered in [Grig 15b, Grig 15a], the recursion (2.16) does not determine a standard VAR(1) model. Nevertheless, since by hypothesis $\rho\left(A(\mathbf{x}_0, \boldsymbol{\theta})\right) < 1$, the proof of the existence of a unique stationary solution for a VAR(1) model can be mimicked in this case (see, for example, Section 2.1.1 and Proposition C.9 in [Lutk 05]) in order to show that the recursion

$$\mathbf{x}(t) - \mathbf{x}_0 = A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t-1) - \mathbf{x}_0) + \boldsymbol{\varepsilon}(t), \tag{4.1}$$

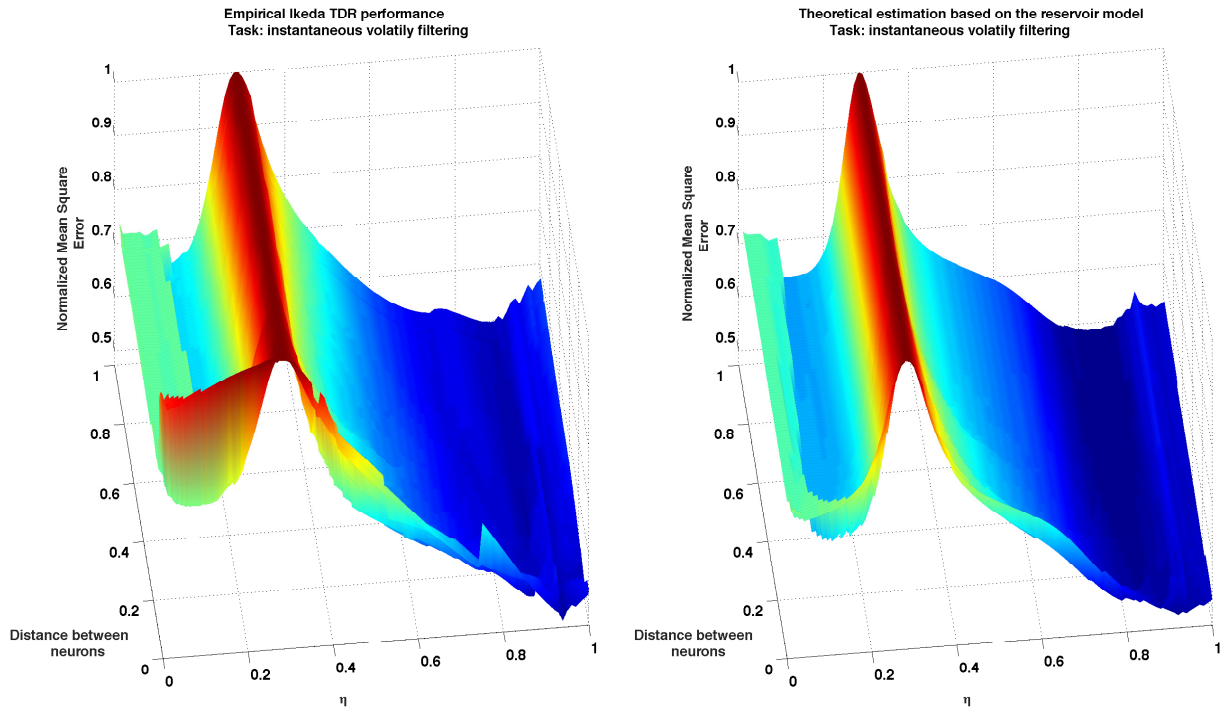

Figure 1: Error surfaces associated to the filtering of the instantaneous volatility $\sigma(t)$ an ARSV data generating process. The left panel has been computed using Monte Carlo simulations in order to empirically evaluate the filtering error of the Ikeda TDR as a function of η in (3.8) and of the distance between neurons. The right panel was obtained by evaluating the formula (2.13) based on the reservoir model (2.16) with a nonlinearity of order $R = 8$. The two surfaces have minima at virtually the same parameter values.

has a unique second order stationary solution given by

$$\mathbf{x}(t) = \mathbf{x}_0 + \sum_{i=0}^{\infty} A(\mathbf{x}_0, \boldsymbol{\theta})^i \boldsymbol{\varepsilon}(t-i). \tag{4.2}$$

Taking expectations in both sides of (4.2) and using the stationarity of $\boldsymbol{\varepsilon}(t)$ yields (2.19) (see also Proposition C.10 in [Lutk 05]). It is straightforward to verify using that expression that (4.1) is identical to (2.21) whose unique stationary solution is given by (2.22) and which hence coincides necessarily with (4.2). Finally, using the definition of Γ and (2.22), the expression (2.20) follows. ■

4.2 Proof of Theorem 2.9

Proof of part (i) We start by modifying the notation introduced in (2.15) in order to specifically indicate the dependence of $\boldsymbol{\varepsilon}(t)$ on the input signal and on the input mask. We set:

$$\boldsymbol{\varepsilon}(z, \mathbf{c}) := \sum_{i=1}^{R} \frac{z^i}{i!} D_{\mathbf{I}}^{(i)} F(\mathbf{x}_0, \mathbf{0}_N, \boldsymbol{\theta}) \underbrace{(\mathbf{c} \otimes \cdots \otimes \mathbf{c})}_{i \text{ factors}}. \tag{4.3}$$

Note now that the map $\boldsymbol{\varepsilon}(\cdot, \mathbf{c}) : \mathbb{R} \longrightarrow \mathbb{R}^N$ that assigns $\boldsymbol{\varepsilon}(z, C)$ to the input signal z is the composition of $z \mapsto \mathbf{c}z$ and the map $F_I^R(\cdot, \mathbf{x}_0, \boldsymbol{\theta}) : \mathbb{R}^N \to \mathbb{R}^N$ in the statement. Consequently, if $F_I^R(\cdot, \mathbf{x}_0, \boldsymbol{\theta})$ satisfies

any of the two injectivity hypotheses in **(a)** or **(b)** in the statement, then the map $\boldsymbol{\varepsilon}(\cdot, \mathbf{c})$ is necessarily injective, that is,

$$\boldsymbol{\varepsilon}(z, \mathbf{c}) \neq \boldsymbol{\varepsilon}(z', \mathbf{c}), \quad \text{whenever} \quad z \neq z'. \tag{4.4}$$

Let now $\{z(t)\}_{t\in\mathbb{Z}}$, $\{z'(t)\}_{t\in\mathbb{Z}}$ be two input signals that are identical except at $s \in \mathbb{N}$, that is, $z(s) \neq z'(s)$. We show by induction that the corresponding reservoir outputs $\mathbf{x}(t)$, $\mathbf{x}'(t)$ are such that $\mathbf{x}(t) \neq \mathbf{x}'(t)$ for any $t \geq s$. First, since $z(t) = z'(t)$ for any $t < s$, then $\mathbf{x}(t) = \mathbf{x}'(t)$ for any $t < s$ necessarily. Now by (2.16) and using the notation introduced in (4.3), we have:

$$\mathbf{x}(s) - \mathbf{x}'(s) = A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(s-1) - \mathbf{x}'(s-1)) + \boldsymbol{\varepsilon}(z(s), C) - \boldsymbol{\varepsilon}(z'(s), C) = \boldsymbol{\varepsilon}(z(s), C) - \boldsymbol{\varepsilon}(z'(s), C).$$

Given that $z(s) \neq z'(s)$, the hypotheses **(a)** or **(b)** in the statement of the theorem together with (4.4) imply that $\mathbf{x}(s) \neq \mathbf{x}'(s)$. We now establish the induction step by assuming that $\mathbf{x}(t) \neq \mathbf{x}'(t)$ for some $t \geq s$ and we show that $\mathbf{x}(t+1) \neq \mathbf{x}'(t+1)$. Indeed, in this case

$$\mathbf{x}(t+1) - \mathbf{x}'(t+1) = A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t) - \mathbf{x}'(t)). \tag{4.5}$$

Since by hypothesis zero is not an eigenvalue of $A(\mathbf{x}_0, \boldsymbol{\theta})$, we have that $\mathbf{x}(t+1) \neq \mathbf{x}'(t+1)$ necessarily because $\mathbf{x}(t) - \mathbf{x}'(t) \neq 0$ by the induction hypothesis.

Proof of part (ii) Let $\varepsilon_1 > 0$ and $h \in \mathbb{N}$. The continuity of the map $\boldsymbol{\varepsilon}(\cdot, \mathbf{c})$ defined in (4.3) implies that there exists $\delta(\varepsilon_1) > 0$ such that if two input signals $\{z(t)\}_{t\in\mathbb{Z}}$, $\{z'(t)\}_{t\in\mathbb{Z}}$ are such that $|z(s) - z'(s)| < \delta(\varepsilon_1)$ for all $s \in [t-h, t]$, then $\|\boldsymbol{\varepsilon}(z(s), \mathbf{c}) - \boldsymbol{\varepsilon}(z'(s), \mathbf{c})\| < \varepsilon_1$, for all $s \in [t-h, t]$. Additionally, since by hypotheses the input signals are bounded, then $\|\boldsymbol{\varepsilon}(z(t), \mathbf{c})\| < K_{\max}$ and $\|\boldsymbol{\varepsilon}(z'(t), \mathbf{c})\| < K_{\max}$, for all $t \in \mathbb{Z}$ and for some $K_{\max} \in \mathbb{R}$.

Let now $\mathbf{x}(t)$ and $\mathbf{x}'(t)$ be the reservoir outputs corresponding to $z(t)$ and $z'(t)$, respectively, then

$$\mathbf{x}(t) - \mathbf{x}'(t) = A(\mathbf{x}_0, \boldsymbol{\theta})(\mathbf{x}(t-1) - \mathbf{x}'(t-1)) + \boldsymbol{\varepsilon}(z(t), \mathbf{c}) - \boldsymbol{\varepsilon}(z'(t), \mathbf{c}).$$

Since by hypothesis $\|A(\mathbf{x}_0, \boldsymbol{\theta})\| < 1$ and the spectral radius $\rho(A(\mathbf{x}_0, \boldsymbol{\theta})$ satisfies that $\rho(A(\mathbf{x}_0, \boldsymbol{\theta}) < \|A(\mathbf{x}_0, \boldsymbol{\theta})\|$ for any matrix norm, we have that $\rho(A(\mathbf{x}_0, \boldsymbol{\theta}) < 1$. Additionally, since the input signals are strictly stationary with finite automoments up to order $2R$, we can use Proposition 2.6 to rewrite this expression as

$$\mathbf{x}(t) - \mathbf{x}'(t) = \sum_{i=0}^{\infty} A(\mathbf{x}_0, \boldsymbol{\theta})^i (\boldsymbol{\varepsilon}(z(t-i), \mathbf{c}) - \boldsymbol{\varepsilon}(z'(t-i), \mathbf{c})) \tag{4.6}$$

which implies that

$$\begin{aligned} ||\mathbf{x}(t) - \mathbf{x}'(t)|| \leq & \sum_{i=0}^{h} ||A(\mathbf{x}_0, \boldsymbol{\theta})||^i ||\boldsymbol{\varepsilon}(z(t-i), \mathbf{c}) - \boldsymbol{\varepsilon}(z'(t-i), \mathbf{c})|| + 2K_{\max} \sum_{i=h+1}^{\infty} ||A(\mathbf{x}_0, \boldsymbol{\theta})||^i \\ \leq & \varepsilon_1 \sum_{i=0}^{h} ||A(\mathbf{x}_0, \boldsymbol{\theta})||^i + 2K_{\max} \frac{||A(\mathbf{x}_0, \boldsymbol{\theta})||^{h+1}}{1 - ||A(\mathbf{x}_0, \boldsymbol{\theta})||} = \frac{\varepsilon_1 + (2K_{\max} - \varepsilon_1)||A(\mathbf{x}_0, \boldsymbol{\theta})||^{h+1}}{1 - ||A(\mathbf{x}_0, \boldsymbol{\theta})||} =: \varepsilon. \end{aligned}$$

Finally, the hypothesis $\|A(\mathbf{x}_0, \boldsymbol{\theta})\| < 1$ implies that $\lim_{h\to\infty} ||A(\mathbf{x}_0, \boldsymbol{\theta})||^{h+1} = 0$ and hence since $\varepsilon_1 > 0$ can be made as small as desired, so is ε, which proves the statement. ■

4.3 Proof of Corollary 2.11

Proof of part (i) In order to obtain this result as a corollary of part **(i)** in Theorem 2.9 we need to prove the following two statements. First, that the injectivity hypotheses on $f_I^R(\cdot, x_0, \boldsymbol{\theta})$ imply

those about the corresponding map $F_I^R(\cdot, \mathbf{x}_0, \boldsymbol{\theta})$ in Theorem 2.9. Second, in this case 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.

The first statement can be proved by noting that, as a consequence of (2.7):

$$F_I^R(\mathbf{I}, \mathbf{x}_0, \boldsymbol{\theta}) = (1 - e^{-\xi}) \begin{pmatrix} f_I^R(I_1, x_0, \boldsymbol{\theta}) \\ e^{-\xi} f_I^R(I_1, x_0, \boldsymbol{\theta}) + f_I^R(I_2, x_0, \boldsymbol{\theta}) \\ e^{-2\xi} f_I^R(I_1, x_0, \boldsymbol{\theta}) + e^{-\xi} f_I^R(I_2, x_0, \boldsymbol{\theta}) + f_I^R(I_3, x_0, \boldsymbol{\theta}) \\ \vdots \\ e^{-(N-1)\xi} f_I^R(I_1, x_0, \boldsymbol{\theta}) + e^{-(N-2)\xi} f_I^R(I_2, x_0, \boldsymbol{\theta}) + \cdots + f_I^R(I_N, x_0, \boldsymbol{\theta}) \end{pmatrix}. \tag{4.7}$$

This expression can be used to to easily show recursively that $F_I^R(\cdot, \mathbf{x}_0, \boldsymbol{\theta})$ is injective if $f_I^R(\cdot, x_0, \boldsymbol{\theta})$ is injective. Concerning the second point, Theorem D.11 in [Grig 15b] proves that zero is not an eigenvalue of $A(\mathbf{x}_0, \boldsymbol{\theta})$.

Proof of part (ii) It follows from part **(ii)** in Theorem 2.9 by noting that $|\partial_x f(x_0, 0, \boldsymbol{\theta})| < 1$ implies (see the proof of Theorem D.10 in Supplementary Material of [Grig 15b]) that

$$||A(\mathbf{x}_0, \boldsymbol{\theta})||_\infty < 1. \quad \blacksquare$$

References

[Appe 11] L. Appeltant, M. C. Soriano, G. Van der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. R. Mirasso, and I. Fischer. "Information processing using a single dynamical node as complex system". *Nature Communications*, Vol. 2, p. 468, Jan. 2011.

[Atiy 00] A. F. Atiya and A. G. Parlos. "New results on recurrent network training: unifying the algorithms and accelerating convergence". *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, Vol. 11, No. 3, pp. 697–709, Jan. 2000.

[Boll 86] T. Bollerslev. "Generalized autoregressive conditional heteroskedasticity". *Journal of Econometrics*, Vol. 31, No. 3, pp. 307–327, 1986.

[Boll 88] T. Bollerslev, R. F. Engle, and J. M. Wooldridge. "A capital asset pricing model with time varying covariances". *Journal of Political Economy*, Vol. 96, pp. 116–131, 1988.

[Box 76] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, 1976.

[Broc 02] P. J. Brockwell and R. A. Davis. *Introduction to Time Series and Forecasting*. Springer, 2002.

[Broc 06] P. J. Brockwell and R. A. Davis. *Time Series: Theory and Methods*. Springer-Verlag, 2006.

[Brun 13] D. Brunner, M. C. Soriano, C. R. Mirasso, and I. Fischer. "Parallel photonic information processing at gigabyte per second data rates using transient states". *Nature Communications*, Vol. 4, No. 1364, 2013.

[Cast 08] J. del Castillo and Y. Lee. "GLM-methods for volatility models". *Stat. Model.*, Vol. 8, No. 3, pp. 263–283, 2008.

[Croo 07] N. Crook. "Nonlinear transient computation". *Neurocomputing*, Vol. 70, pp. 1167–1176, 2007.

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

[Engl 82] R. F. Engle. "Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation". *Econometrica*, Vol. 50, No. 4, pp. 987–1007, 1982.

[Gang 08] 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. 18970–5, Dec. 2008.

[Grig 14a] 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.

[Grig 14b] L. Grigoryeva and J.-P. Ortega. "Hybrid forecasting with estimated temporally aggregated linear processes". *Journal of Forecasting*, Vol. 33, pp. 577–595, 2014.

[Grig 15a] L. Grigoryeva, J. Henriques, L. Larger, and J.-P. Ortega. "Nonlinear memory capacity of parallel time-delay reservoir computers in the processing of multidimensional signals.". *Preprint*, 2015.

[Grig 15b] 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.

[Grig 15c] L. Grigoryeva and J.-P. Ortega. "Asymptotic forecasting error evaluation for estimated temporally aggregated linear processes". *International Journal of Computational Economics and Econometrics*, Vol. 5, No. 3, pp. 289–318, 2015.

[Guti 12] J. M. Gutiérrez, D. San-Martín, S. Ortín, and L. Pesquera. "Simple reservoirs with chain topology based on a single time-delay nonlinear node". In: *20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 13–18, 2012.

[Hami 94] J. D. Hamilton. *Time series analysis*. Princeton University Press, Princeton, NJ, 1994.

[Harv 94] A. C. Harvey, E. Ruiz, and N. Shephard. "Multivariate stochastic variance models". *Review of Economic Studies*, Vol. 61, pp. 247–264, 1994.

[Henr 14] J. Henriques and J.-P. Ortega. "Construction, management, and performance of sparse Markowitz portfolios". *Studies in Nonlinear Dynamics and Econometrics*, Vol. 18, No. 4, pp. 383–402, 2014.

[Herm 10] 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.

[Holm 88] B. Holmquist. "Moments and cumulants of the multivariate normal distribution". *Stochastic Analysis and Applications*, Vol. 6, No. 3, pp. 273–278, Jan. 1988.

[Jaeg 01] H. Jaeger. "The 'echo state' approach to analysing and training recurrent neural networks". Tech. Rep., German National Research Center for Information Technology, 2001.

[Jaeg 02] H. Jaeger. "Short term memory in echo state networks". *Fraunhofer Institute for Autonomous Intelligent Systems. Technical Report.*, Vol. 152, 2002.

[Jaeg 04] 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.

[Jaeg 07] H. Jaeger, M. Lukoševičius, D. Popovici, and U. Siewert. "Optimization and applications of echo state networks with leaky-integrator neurons". *Neural Networks*, Vol. 20, No. 3, pp. 335–352, 2007.

[Larg 12] L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutierrez, L. Pesquera, C. R. Mirasso, and I. Fischer. "Photonic information processing beyond Turing: an optoelectronic implementation of reservoir computing". *Optics Express*, Vol. 20, No. 3, p. 3241, Jan. 2012.

[Lee 06] 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.

[Lee 96] Y. Lee and J. A. Nelder. "Hierarchical generalized linear models". *J. Roy. Statist. Soc. Ser. B*, Vol. 58, No. 4, pp. 619–678, 1996.

[Li 01] W. K. Li, S. Ling, and M. MacAleer. "A survey of recent theoretical results for time series models with GARCH errors". 2001.

[Lim 11] 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.

[Ling 02] S. Ling and M. McAleer. "Stationarity and the existence of moments of a family of GARCH processes". *Journal of Econometrics*, Vol. 106, No. 1, pp. 109–117, Jan. 2002.

[Luko 09] M. Lukoševičius and H. Jaeger. "Reservoir computing approaches to recurrent neural network training". *Computer Science Review*, Vol. 3, No. 3, pp. 127–149, 2009.

[Lutk 05] H. Lütkepohl. *New Introduction to Multiple Time Series Analysis*. Springer-Verlag, Berlin, 2005.

[Maas 02] 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.

[Maas 11] W. Maass. "Liquid state machines: motivation, theory, and applications". In: S. S. Barry Cooper and A. Sorbi, Eds., *Computability In Context: Computation and Logic in the Real World*, Chap. 8, pp. 275–296, 2011.

[Orti 12] 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*, pp. 895–899, 2012.

[Paqu 12] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, and S. Massar. "Optoelectronic reservoir computing". *Scientific reports*, Vol. 2, p. 287, Jan. 2012.

[Roda 11] A. Rodan and P. Tino. "Minimum complexity echo state network.". *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, Vol. 22, No. 1, pp. 131–44, Jan. 2011.

[Tayl 86] S. J. Taylor. *Modelling Financial Time Series*. John Wiley \& Sons, Chichester, 1986.

[Tria 03] K. Triantafyllopoulos. "On the central moments of the multidimensional Gaussian distribution". *The Mathematical Scientist*, Vol. 28, pp. 125–128, 2003.

[Vers 07] D. Verstraeten, B. Schrauwen, M. DHaene, and D. Stroobandt. "An experimental unification of reservoir computing methods". *Neural Networks*, Vol. 20, pp. 391–403, 2007.

[Whit 04] 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.

[Yild 12] 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.