Journal of Physics: Photonics

OPEN ACCESS

RECEIVED
8 January 2021

REVISED
10 March 2021

ACCEPTED FOR PUBLICATION
25 March 2021

PUBLISHED
13 April 2021

Original Content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence.

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

PAPER

Insight into delay based reservoir computing via eigenvalue analysis

Felix Köster[1,*], **Serhiy Yanchuk**[1,2] **and Kathy Lüdge**[1]

[1] Institut for Theoretical Physics, Technische Universität Berlin, Berlin 10623, Germany
[2] Institut for Mathematics, Technische Universität Berlin, Berlin 10623, Germany
[*] Author to whom any correspondence should be addressed.

E-mail: f.koester@tu-berlin.de

Keywords: reservoir computing, laser physics, nonlinear dynamics

Abstract

In this paper we give a profound insight into the computation capability of delay based reservoir computing via an eigenvalue analysis. We concentrate on the task-independent memory capacity to quantify the reservoir performance and compare these with the eigenvalue spectrum of the dynamical system. We show that these two quantities are deeply connected, and thus the reservoir computing performance is predictable by analysing the small signal response of the reservoir. Our results suggest that any dynamical system used as a reservoir can be analysed in this way. We apply our method exemplarily to a photonic laser system with feedback and compare the numerically computed recall capabilities with the eigenvalue spectrum. Optimal performance is found for a system with the eigenvalues having real parts close to zero and off-resonant imaginary parts.

1. Introduction

Reservoir computing is a novel approach for time-dependent tasks in machine learning. First introduced by Jaeger [1] and inspired by the human brain [2], it utilizes the inherent computational capabilities of dynamical systems. Very recently the universal approximation property has also been shown for a wide range of reservoir computers, which solidifies the concept as a broad applicable scheme [3].

Hardware setups have shown the feasibility and wide range of realizations [4–6], while theoretical and numerical analysis show interesting advancements [7, 8] and pinpoint to easily implementable realizations [9, 10]. Different applications have been demonstrated [11–20]. Since speed is of essence in computation, optoelectronic [21, 22] and optical setups [9, 23–29] are frequently studied, which additionally come with the benefit of low energy consumption.

A new and sophisticated approach to the reservoir computing scheme was introduced by Appeltant *et al* in [30], where a single dynamical node under the influence of external feedback utilizes a time-multiplexed reservoir. The spatially extended network structure of classical reservoirs is no longer needed with this scheme, which reduces the complexity in reservoir hardware in exchange for processing speed. A schematic sketch is shown in figure 1. Realizations with a single delayed reservoir [31–36] give a first glimpse over the potential of this idea for, e.g. time-series predictions [28, 37], equalization tasks on nonlinearly distorted signals [38], and fast word recognition [39]. A general analysis, introduced by Dambre *et al* [40], was also used to quantify the task-independent computational capabilities of semiconductor lasers [41]. For an overview, we refer to [42–44].

A lot of research was already invested in order to develop a deeper understanding of reservoir computing systems, however, effective measures that allow to predict the performance are still missing. In this paper we want to fill this gap by providing a scheme that allows to predict general trends of the performance using the eigenvalue spectra of the dynamical system (the reservoir) without input. As an example reservoir, we chose a laser that is subjected to optical self-feedback. We use the Lang–Kobayashi system, which is an established model for a semiconductor laser with delayed external feedback. We calculate the total memory capacity as well as the linear and nonlinear contributions using the method derived in [40] and compare the results with

© 2021 The Author(s). Published by IOP Publishing Ltd

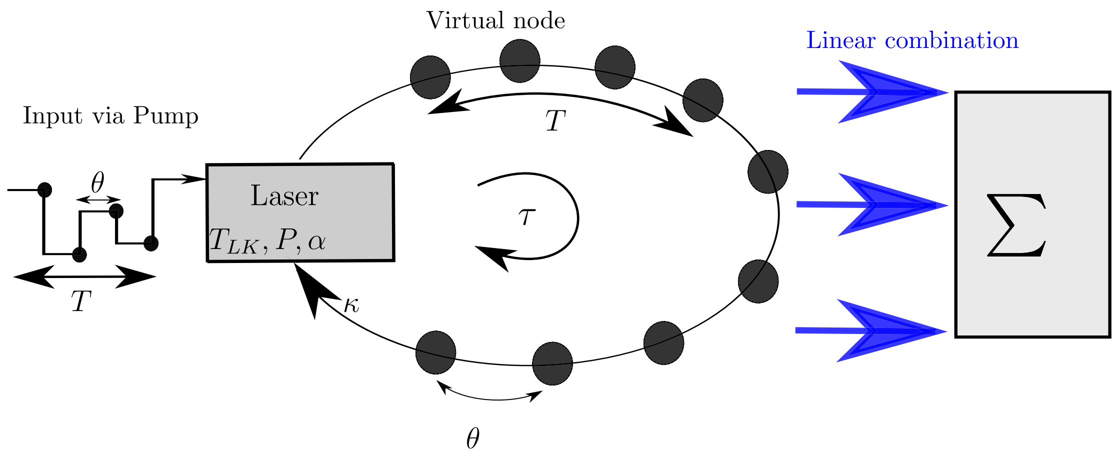

Figure 1. Scheme of time-multiplexed reservoir computing with a laser.

the computed eigenvalue spectrum of the system, where we discover a clear connection. In particular, a high linear memory capacity is found for systems, where a large number of eigenvalues are close to criticality (with small negative real parts) and non-resonant (with imaginary parts not-resonant to the input timescale).

The paper is structured as follows. First, we give an overview of the methods used for calculating the memory capacity and the eigenvalue spectrum in section 2. After that, we present our results and discuss the impact of the eigenvalues on the performance and different nonlinear recall contributions first for a reservoir formed by a solitary laser and then by a laser with external cavity.

2. Methods

The reservoir computing scheme employs the idea of a dynamical reservoir, which projects input information into a high dimensional phase space. The nonlinear response of the reservoir is then used by a linear readout to approximate a specific task depending on the input. Often the reservoir consists of many nodes with relatively simple dynamics (for example, tanh-function [1]) in which the input enters via a weighted matrix. Afterward, the response is read out and linearly combined to generate an output. The idea is to minimize the Euclidean distance between the generated output and the target. This approach is particularly resourceful for time-dependent tasks, because the dynamical system which forms the reservoir has memory and thus acts as a memory kernel.

The modified approach introduced by [30] uses a single node with delay as a reservoir, in which the output dimensions are distributed over time. A mask g is used to feed the input into the system, which produces a high dimensional response. These responses are saved over time and used for the linear readout approximation. A sketch of the setup is shown in figure 1. In the following, we will give a short overview of the quantities and notations used in this paper. We also refer to our previous works [45, 46], where a detailed explanation of how the reservoir setup is operated and task-independent memory capacities are computed is given.

2.1. Time-multiplexed reservoir computing

Let us briefly remind the main ingredients of the time-multiplexed reservoir computing scheme [30, 45, 46]. An input vector $\mathbf{u} \in \mathbb{R}^L$ enters the system componentwise at times $t_l = lT$, $l = 1, \ldots, L$, L being the number of sample points. The time between two inputs $t_{l+1} - t_l$ is called the clock cycle T and describes the period length in which one input u_l is applied to the system. Inside each interval of one clock cycle T, a T-periodic mask function g is applied on the inputs (see figure 1). The mask g is piecewise-constant on N_V intervals, each of length $\theta = T/N_V$ corresponding to N_V virtual nodes. The values of the mask function g play the same role as the input weights in spatially extended reservoirs, with the difference that the input weights are now distributed over time.

The system responses are collected in the state matrix $\mathsf{S} \in \mathbb{R}^L \times \mathbb{R}^{N_V}$, where N_V is the dimension of the measured system's state. More specifically, the elements of the state matrix are $\mathsf{S}_{ln} = s(lT + n\theta)$ with $n = 1, \ldots, N_V$, and $l = 1, \ldots, L$, where $s(t) \in \mathbb{R}$ is the state of the dynamical element of the reservoir at time t, e.g. a variable of the delay system in simulations, or laser intensity in an experimental realization.

A linear combination of the state matrix is given by $\mathsf{S}\mathbf{w}$, where $\mathbf{w} \in \mathbf{R}^M$ is a vector of weights. Such a combination is trained to find a least square approximation to some target vector $\hat{\mathbf{y}}$:

$$\arg\min_{\mathbf{w}} \left[||\mathsf{S}\mathbf{w} - \hat{\mathbf{y}}||_2^2 + \lambda_T ||\mathbf{w}||_2^2 \right],$$

where $\|\cdot\|_2$ is the Euclidean norm, and λ_T is a Tikhonov regularization parameter. A solution to this problem is known to satisfy:

$$\mathbf{w} = (\mathsf{S}^T\mathsf{S} + \lambda_T \mathrm{I})^{-1}\mathsf{S}^T\hat{\mathbf{y}}, \tag{1}$$

when $\mathsf{S}^T\mathsf{S} + \lambda_T\mathrm{I}$ is invertible. In the case of our Lang–Kobayashi model, since the physical system is intrinsically noisy, we used the state noise regularization [1, 47] and set $\lambda_T = 0$. This is reasonable, as noise dominates very small dependencies in the given training data set, which the linear readouts would otherwise try to fit. It also gives a more realistic threshold for the precision of the state readouts compared to numerical precision. Comparisons of simulations without noise and with Tikhonov regularization to a noisy system without Tikhonov regularization yielded similar results.

To quantify the system's performance, we use the normalized root mean square error (NRMSE) between the approximation $\mathbf{y} = \mathsf{S}\mathbf{w}$ and the target $\hat{\mathbf{y}}$:

$$\mathrm{NRMSE} = \sqrt{\frac{\sum_{l=1}^{L}(\hat{y}_l - y_l)^2}{N \cdot \mathrm{var}(\hat{\mathbf{y}})}}, \tag{2}$$

where $\mathrm{var}(\hat{\mathbf{y}})$ is the variance of the target values $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_L)$.

2.2. Memory capacity

Dambre *et al* have shown in [40] that the computational capability of a reservoir system can be quantified via an orthonormal set of basis functions on a sequence of inputs. Here we give a recap of the used quantities introduced in [45]. In particular, the capacity to fulfil a certain task is given by:

$$\mathrm{C}_{\hat{\mathbf{y}}} = \frac{\hat{\mathbf{y}}^T\mathsf{S}(\mathsf{S}^T\mathsf{S})^{-1}\mathsf{S}^T\hat{\mathbf{y}}}{||\hat{\mathbf{y}}||^2} = \frac{\hat{\mathbf{y}}^T\mathbf{y}}{||\hat{\mathbf{y}}||^2}. \tag{3}$$

The capacity equals 1 if $\mathbf{y} = \hat{\mathbf{y}}$ and the reservoir computer computes perfectly the task; $C = 0$ if it can not compute it at all, and in between 0 and 1 if it is partially capable to fulfil the task. In section appendix A, we explain how equation (3) follows from the corresponding expression in [40]. Further, following Dambre *et al* [40], we use finite products of normalized Legendre polynomials $\mathcal{P}_d$ as a basis of the Hilbert space of all possible transformations (thus tasks with targets $\hat{\mathbf{y}}$) on an input sequence $\{u\} = \{u_{-L}, \ldots, u_{-3}, u_{-2}, u_{-1}\}$. As inputs into the system, we use uniformly distributed random numbers u_l, which are independent and identically drawn in $[-1, 1]$. This yields uncorrelated inputs and thus uncorrelated memory capacities. After feeding the input sequence $\{u\}$ of random numbers into the system, it yields a reservoir response S. Formally, the memory capacity (equation (3)) is defined for an infinitely long sequence $L \to \infty$. To approximate it numerically, we use $L = 250\,000$.

In order to describe a task, the target vector $\hat{\mathbf{y}}$ is defined as:

$$\hat{\mathbf{y}}_{\{u\}} = \prod_i \mathcal{P}_{d_i}(u_{-i}), \tag{4}$$

where $\{d\} = \{d_1, \ldots, d_I\}$ is a sequence of degrees such that the Legendre polynomial $P_{d_i}(u_{-i})$ of degree d_i is applied to the input u_{-i}. The product of all such polynomials is used to generate the task (target vector $\hat{\mathbf{y}}$). The collection of all tasks (4) for any possible degree sequence $\{d\}$ is the Hilbert space of all possible transformations [40].

Further, to define the linear and nonlinear memory capacities, one uses special tasks, for which the sum of the degrees $\sum_i d_i$ is constant:

$$\hat{\mathbf{y}}^d_{\{u\}} = \prod_{d_1+d_2+\cdots=d} \mathcal{P}_{d_i}(u_{-i}). \tag{5}$$

Clearly, there are many such possible tasks for all sequences $\{d\}$ with $d = \sum_i d_i$. The memory capacity MC^d of degree d is defined as the sum of the capacities $\mathrm{C}_{\hat{\mathbf{y}}}$ computed using equation (3) for all tasks (5) of degree d:

$$MC^d = \sum_{\{d\}:\, d_1+d_2+\cdots=d} C_{\hat{\mathbf{y}}^d_{\{u\}}}. \tag{6}$$

The well known linear memory capacity corresponds to $d = 1$. The total memory capacity is then given by the memory capacities MC^d of all degrees d.

$$MC = \sum_{d>0} MC^d. \tag{7}$$

It was shown in [40] that MC is limited by the readout-dimension N_V, which equals the number of virtual nodes N_V. An intuitive explanation is the following. The linear readout Sw of the reservoir computing scheme can be considered a linear combination of the columns of the state matrix S. Thus the amount of dimensions this basis can approximate is given by the number of linearly independent readouts. If the systems states are linearly independent, it can at most approximate N_V different dimensions, which is in our case N_V different tasks constructed from equation (4). A more rigorous explanation is given by Dambre *et al* in [40].

2.3. NARMA10

In addition to memory capacities, we evaluate the NRMSE of the NARMA10 task. NARMA10 is an often used benchmark test that combines linear and nonlinear memory transformations. It is given by the following iterative formula:

$$A_{n+1} = 0.3A_n + 0.05A_n \left(\sum_{i=0}^{9} A_{n-i} \right) + 1.5u_{n-9}u_n + 0.1. \tag{8}$$

Here, A_n is an iteratively given number and u_n is an independent and identically drawn uniformly distributed random number in $[0, 0.5]$. The reservoir is fed with the random numbers u_n and has to predict the value of A_{n+1}.

2.4. Lang–Kobayashi model

We use the Lang–Kobayashi laser as an example reservoir. This is a model applicable for semiconductor lasers with external feedback operating with low feedback strength. The Lang–Kobayashi equations have been studied widely, modelling successfully semiconductor lasers [48, 49] exhibiting complex dynamics and bifurcation scenarios [49–51]. The dimensionless equations of motion are given by [52]:

$$\frac{dE(t)}{dt} = (1 + i\alpha)N(t)E(t) + \kappa e^{i\phi}E(t-\tau) + D_{\text{noise}}\xi(t), \tag{9}$$

$$\frac{dN(t)}{dt} = \frac{1}{T_{LK}}(P + \eta I(t)g(t) - N(t) - (2N(t) + 1)|E(t)|^2). \tag{10}$$

The parameters scaling was chosen as in [53] with a modification to allow for the information input. The system time is normalized to the photon lifetime. Here, E is the complex electric field, N is the charge carrier inversion, and ξ describes spontaneous emission modelled by Gaussian white noise, g is the masking function, I is the input, α the amplitude-phase coupling, κ is the feedback strength, ϕ the feedback phase, τ the delay time, D_{noise} the noise amplitude and $P + \eta I(t)g(t)$ is the pump current, composed of P a constant pump level and η the input strength of the information fed into the system via electric injection, which is small with respect to P. $I(t)g(t)$ is the piecewise constant input function, which contains the data set values multiplied with a mask function. T_{LK} is the time scale ratio, modelling class B laser behaviour for sufficiently large T_{LK} seen in conventional semiconductor lasers. $T_{LK} \ll 1$ models class A behaviour observed in quantum dot lasers [54–56] or quantum cascade lasers [57, 58]. Note that the threshold pump current for the solitary laser is at $P_{th} = 0$, while P_{th} changes with κ according to $P_{th} = -\kappa$.

2.5. Calculating eigenvalue spectrum

The eigenvalues of any dynamical system describe the dynamics for small perturbations around a linearized point. Because reservoir computers are often operated close to a stable fixpoint the eigenvalue spectrum of its linearized system can be analysed. The goal of this paper is to find a relation between the nonlinear memory recallability and the eigenvalue spectrum. The latter can be computed with much less numerical effort and could then be used to predict good parameter ranges for reservoir setups. It also gives insight into the timescales of the eigendirections of the system, which contain information on the memory kernel of the reservoir. To compute the eigenvalue spectrum, we used two methods: the first method is an analytical approximation in the long delay limit [59], while the second relies on numerical computation with the DDE-biftool software package [60–62].

To begin with, we give a short overview of the first method from [59], which provides an approximation of the spectrum of long-delay systems. In [53], it was applied to the Lang–Kobayashi system. As delay-based reservoir computing is mostly used with a long delay compared to the local dynamics, this is a valid approximation that gives a general tool to analyse the reservoirs of such type.

The characteristic equation for the eigenvalues is obtained through the linearization around a steady state x^*, and it reads as:

$$\det\left(-\lambda I + A + Be^{-\lambda\tau}\right) = 0, \tag{11}$$

with some constant matrices A and B, and I is the identity matrix. For large τ, its solutions can be decomposed into two parts, in which one scales as $\Re(\lambda) \sim 1/\tau$, also called the pseudocontinuous spectrum, and a strongly unstable spectrum with the scaling $\Re(\lambda) \sim 1$ with $\Re(\lambda) > 0$. The strongly unstable spectrum is absent for reservoir computing applications, since, otherwise, the reservoir's state is strongly unstable, and the echo state (or fading memory) property [1] is lost to a large extend. Formally, the condition for the absence of the strong unstable spectrum is the stability (all eigenvalues have negative real parts) of the linearization matrix A of instantaneous terms (see equation 11). That means that either the operating point of the reservoir shifts to another attractor or the reservoir is unstable. Hence, we focus on the pseudocontinuous spectrum, which can be obtained by introducing the Ansatz:

$$\lambda = \frac{\gamma}{\tau} + i\mu, \tag{12}$$

where γ and μ are two new real variables. Substituting equation (12) into (11) one gets in the leading order:

$$\det\left(-i\mu I + A + Be^{-\gamma - i\mu\tau}\right) = 0. \tag{13}$$

Equation (13) is a polynomial with respect to $e^{-\gamma-i\mu\tau}$. If $Y_j(\mu)$ are solutions of this polynomial, then:

$$e^{-\gamma - i\mu\tau} = Y_j(\mu), \tag{14}$$

and $\gamma_j(\mu) = -\ln|Y_j(\mu)|$ are the rescaled real parts of the eigenvalues from the pseudocontinuous spectrum. More exactly, the curves $\gamma_j(\mu)/\tau + i\mu$ are approximated with the eigenvalues for large τ.

In the case of the Lang–Kobayashi system, there are two solutions for the real parts of the pseudocontinuous eigenvalue spectrum, see the derivation in [53],

$$\gamma_1(\mu) = -\ln\left|1 + i\frac{\mu}{\kappa}\right|, \tag{15}$$

$$\gamma_2(\mu) = -\ln\left|1 + i\frac{\mu}{\kappa} + \frac{2\epsilon(P+\kappa)(\epsilon(1+2A^2) - i\mu)}{\kappa(\mu^2 + \epsilon^2(1+2A^2)^2)}\right|, \tag{16}$$

where $\epsilon = T_{LK}^{-1}$, and $A_l^2 = \frac{P-N^*}{2N^*+1}$ is the constant intensity, with the corresponding inversion N^* at the external cavity mode (ECM). ECMs are the solutions of the Lang–Kobayashi system of the form $E = A_l e^{i\omega t}, N = N^*$ with constant A_l and N^*, which play the role of the equilibria. Due to the S^1 symmetry of the system, each of these solutions can be transformed into an equilibrium with the corresponding characteristic equation (11). The Lang–Kobayahi system possesses many ECMs, however, for the case $\alpha = 0$, we consider the ECM with $N^* = -\kappa$, which is the most stable [53]. In general our method of linearization applies to all fixpoint solutions of any system, thus $\alpha \neq 0$ can be analysed as long as the system is in fixpoint. For $\alpha \neq 0$ the system often starts to jump between different solutions, where some are not of fixpoint nature. Thus comparisons between the different operating points is not as simple and the focus on the essential new method does get lost, why we choose $\alpha = 0$.

In order to approximate the imaginary parts μ of the pseudocontinuous spectrum, we consider the argument of (14) and obtain:

$$\mu_{j,k} = \frac{2\pi k}{\tau} - \frac{1}{\tau}\arg Y_j(\mu_{j,k}), \quad k \in \mathbb{N}, \tag{17}$$

where $\mu_{j,k}$ is the imaginary part of a kth eigenvalue on the jth branch. For the purpose of this paper, we need an approximation of the eigenvalues around the origin. As one can simply show using (17), for large τ, these eigenvalues (their imaginary parts) can be approximated as:

$$\mu_{j,k} = \frac{2\pi k}{\tau} - \frac{1}{\tau}\arg Y_j(0) + \mathcal{O}\left(\frac{1}{\tau^2}\right), \tag{18}$$

as soon as $k/\tau \ll 1$, see also [63, 64] for more detailed estimations. In the case of the Lang-Kobayashi system, the roots $Y_j(0)$ are real, hence, we have either $\arg Y_j(0) = 0$ or π. This leads to:

$$\mu_{j,k} \approx \frac{\pi}{\tau}(2k - \nu), \tag{19}$$

with $\nu = 0$ for $\arg Y_j(0) = 0$, and $\nu = 1$ for $\arg Y_j(0) = \pi$. Hence, all imaginary parts $\mu_{j,k}$ are integer multiples of π/τ. In particular, for any $T \approx j\tau$, $j \in \mathbb{N}$ ($T \approx j\tau/2$ for $\nu = 0$, respectively), the product $\mu_{j,k}T$ is proportional to an integer number of π. This kind of resonance occurs for all considered eigenvalues (independent of k), and it plays an important role in the total memory loss of the reservoir, which is discussed in section 3.2 below.

The second method for computing the eigenvalue spectrum is based on DDE-biftool [60, 61], which is a path-continuation package for Matlab capable of numerically computing the eigenvalues. In our case, we compute the first 100 eigenvalues with the highest real parts to compare these with results from the memory capacities.

We also consider the case of no feedback $\kappa = 0$. This yields a solitary semiconductor laser system that can be tuned from being an effectively 1-dimensional problem (Class A-like for $T_{LK} \ll 1$) to a 2D problem (Class B-like for $T_{LK} \gg 1$) [65]. We will use a linearization and numerical evaluation of the eigenvalue problem. Even though the laser system is 3-dimensional, it possesses the S^1 symmetry $E \to Ee^{i\phi}$ allowing to reduce the dimension by one.

We would like to emphasize that the eigenvalue method would also apply to a reservoir computer with a different form of information input, e.g. optical injection in the case of the Lang–Kobayashi system, since the analysis is performed without any reservoir computer input. As long as the input is a small perturbation to the system, the responses are fully described by the linearized system.

2.6. Simulation description

Simulations have been performed in C++ with standard libraries, except for linear algebra calculations, which were done via the linear algebra library 'Armadillo' [66]. A Runge–Kutta 4th order method was applied to integrate the delay-differential equation given by equations (9) and (10) numerically, with an integration step $\Delta t = 0.01$ in time units of the photon lifetime. The noise strength is $D_{\text{noise}} = 10^{-7}$ in all simulations. After simulating the system without reservoir inputs to let transients decay, a buffer time of 100 000 inputs was applied (this is excluded from the training process). In the training process, 250 000 inputs were used to have sufficient statistics ($N = 250\,000$). Afterwards, the memory capacities were calculated, whereby a testing phase is not necessary, because the independent and identically drawn uniformly distributed inputs u are statistically equal if drawn for training or testing phases. All possible combinations of the Legendre polynomials up to degree $D = 5$ and 500 input steps into the past were considered ($i = -500$). Capacities $C^d_{\hat{\mathbf{y}}\{u\}}$ below 0.001 were excluded because of finite statistics. For calculating the matrix inverse, the Moore–Penrose pseudoinverse from the C++ linear algebra library 'Armadillo' was used. In the case of the NARMA10 task, 25 000 inputs for training and testing were used. For the piecewise-constant T-periodic masking function g independent and identically distributed random numbers between [0, 1] were used.

For all simulations, the input strength η was fixed to 0.01. The small input strength was used to guarantee the linear answers of the reservoir and, hence, the relevance of the eigenvalue analysis.

2.7. Geometrical intuition

In this paper we will use two quantities Φ and Λ to approximate the memory capacity properties of the reservoir computer. For these two values we would like to give a geometrical intuition, shown figure 2.

The first value $\Phi = \Im(\lambda)T$ we call the relative angular distance between two inputs, where $\Im$ denotes the imaginary part of the eigenvalue λ. Here λ is a critical eigenvalue, i.e. one having its real part close to 0. Φ geometrically describes the angular distance between two distance vectors δs_1 and δs_2 of the system's state s_1 and s_2 at two instances in time separated by one clock cycle interval T. If this relative angular distance is a multiple of π the responses tend to overlap, reducing the separatibility of the inputs, thus degrading the reservoir computer performance.

The second quantity $\Lambda = e^{\Re(\lambda)T} = |\delta s_2|/|\delta s_1|$ describes the distance reduction between two perturbed states, where $\Re$ denotes the real part of the eigenvalue λ. Λ describes the contraction of the system's state towards a new fixpoint due to a new reservoir input. To distinguish two responses s_1 and s_2 for two different inputs u_1 and u_2, the distance $|\delta s_{1,2}|$ (see figure 2) between two responses should be large enough. On the other hand, if the reaction of the system is very fast, i.e. very negative eigenvalues, the system has a high echo state property and thus low memory capacities for any inputs longer than a few (in the worst case even longer than one) steps into the past. If the remaining information of the input nth steps back is degrading very fast

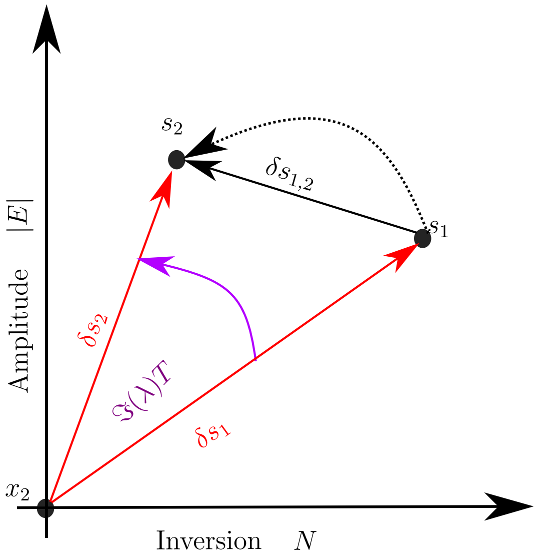

Figure 2. Sketch of the system response in phase space to a small input during the clock cycle T. The trajectory moves from a state s_1 to a new state s_2 (dotted black line). x_2 is a fixpoint of the system due to the new reservoir input u_2. The red vectors $\delta s_1 = s_2 - x_2$, $\delta s_2 = s_1 - x_2$ indicate the distances from this fixpoint for the two instances in time. The distance reduction $\Lambda = e^{\Re(\lambda)T} = |\delta s_2|/|\delta s_1|$ describes the relation of the magnitudes of the two vectors. The purple arrow describes the angular distance $\Phi = \Im(\lambda)T$ covered in one clock cycle interval T. For simplicity, we excluded trajectory responses for different virtual nodes θ.

(very negative eigenvalues), the systems capability to recall is lowered, and at some point reaches the level of the system noise. The distance reduction $\Lambda = e^{\Re(\lambda)T}$ gives a good estimation for both of these properties.

In this paper we will show that both quantities together pinpoint to well performing reservoir computer setups.

3. Results

The following section is structured as follows. First, we will discuss a Lang–Kobayashi laser with $\kappa = 0$, i.e. a solitary laser system, as a reservoir to simplify and depict general concepts. Afterward, we will activate the delay and look at the full Lang–Kobayashi system as a reservoir computer.

3.1. Laser without feedback

We first consider a solitary semiconductor laser system as a reservoir. One has to think of the virtual nodes not to be located on the delay line, but rather as time separated readouts of the system state, that are used in a linear combination. We set $\kappa = 0$ in equations (9), (10) and use 10 virtual nodes ($N_V = 10$). For the considered parameter values $P = 0.05$ and $\alpha = 0$ and without input and noise ($\eta = 0$, $D_{\text{noise}} = 0$), the system's solution converges to a single stable ECM, for which we compute the two eigenvalues. The two eigenvalues are plotted in figure 3(b) as a function of T_{LK}, which gives from left to right the transition from Class A to Class B laser.

To compare the two eigenvalues with the recall capability of the laser, we plot the computed linear, quadratic, cubic, and total memory capacities in figure 3(a). The memory capacities do not change significantly for $T_{LK} \lesssim 3$ where the system corresponds to a class A laser with an adiabatic approximation of the charge carriers. For these parameter values, as one can see from the real parts of the eigenvalues, one eigendirection is considerably faster than the other, and thus can be ignored. At $T_{LK} \approx 2$, the transition from a Class A laser to a class B laser appears, whose steady state solution is a focus. The additional degree of freedom of the charge carrier dynamics leads to an increase of the total memory capacity from about 5 to the theoretical maximum of 10.

Figure 3(c) shows the angular distance $\Phi = \Im(\lambda)T$ taken modulo π, which is based on the rotation $\Im(\lambda)T$ of a small perturbation vector in the 2D phase space during the evolution over the time-interval T (see figure 2). The discontinuities of Φ in figure 3(c) (indicated with vertical dashed purple lines in figure 3(a)) correspond to resonances, i.e. integer numbers of half-a-circle rotations. Comparing the memory capacity at these points in the class B regime, one observes dips in the linear memory capacity and slight changes in the higher-order memories. This effect is pronounced if at the same time real parts of the eigenvalues are close to 0. Since the degradation of the linear memory coincides with the discontinuities in $\Phi \mod \pi$, it can be

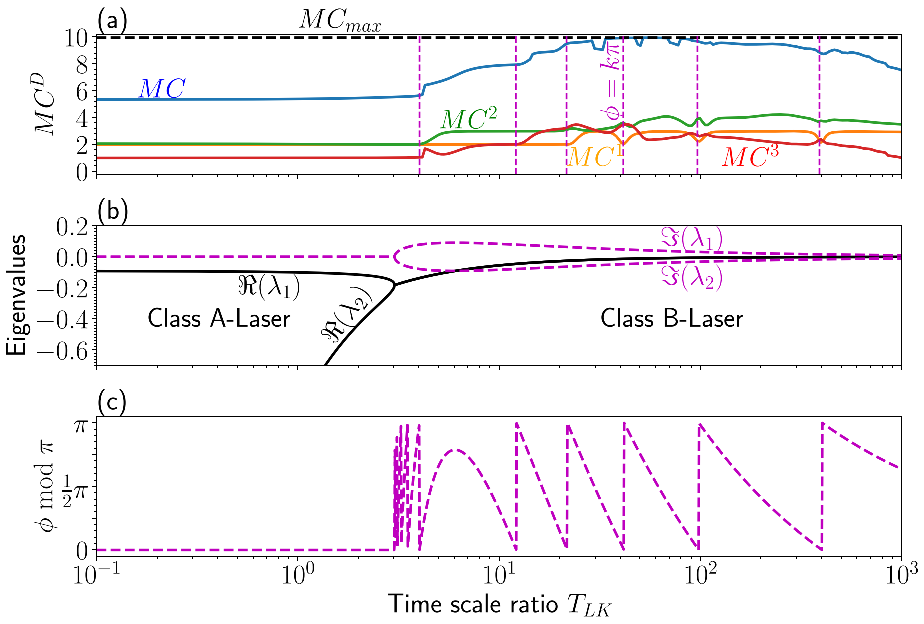

Figure 3. (a) Linear MC^1, quadratic MC^2, cubic MC^3, and total MC memory capacities plotted over the logarithmic lifetime scale ratio T_{LK}. (b) The real and imaginary parts of the eigenvalues showing the transition from class A to class B system. (c) The angular distance between two inputs $\Phi = \Im(\lambda)T$ taken modulo π. Results shown are for the semiconductor laser system with $\kappa = 0$, $P = 0.05$, $N_V = 10$ and $T = 220$ and a logarithmic scan for T_{LK}.

linked to an overlapping of the systems responses, and to a decreasing linear separatibility of the output. We would like to emphasize that even though the system has no optical feedback ($\kappa = 0$) the dynamical system still can act as a reservoir with very short-memory. This comes from the fact that the reaction of the system is not instantaneous yielding a memory kernel of a few inputs into the past. The memory of the reservoir is limited by the real part of the largest critical eigenvalue. This is due to the fact, that the real part of the largest eigenvalue yields the timescale on which small perturbations to the fixpoint exponentially decay.

For a larger picture of the resonance effects at $\Phi \approx k\pi$ ($k \in \mathbb{N}$), a 2D parameter scan was done as a function of the timescale ratio T_{LK} and the clock cycle T (shown in figure 4). The linear, quadratic, cubic, and total memory capacities are color-coded in panel (a)–(d). Bright regions in (a)–(d) correspond to high memory capacities, while dark blue to low memory capacities. The black dashed line shows the scan from figure 3. Purple solid lines show the parameter values where $\Phi \approx k\pi$. The influence of the angular distance Φ is most prominent in figure 4(b), where dips are visible in the linear memory capacity. Its influence on the higher-order memory capacity is also detectable, but harder to describe, as both quadratic and cubic memories either decrease or increase depending on the resonance line.

The solid red lines denote parameter values, where the distance reduction $\Lambda = e^{\Re(\lambda)T} = D_{\text{noise}} = 10^{-7}$ for the two eigenvalues $\lambda_{1,2}$ of the solitary laser system. The two red arrows indicate the direction in the parameter space for decreasing $\Re(\lambda)$ and thus decreasing Λ. The distance reduction shows a decrease in the memory capacities for a decrease in Λ. This rises from the fact that lower Λ correspond to faster eigendirections and thus faster echo state properties.

Combining the information about the two quantities Φ and Λ and comparing it with the memory capacity, we can pinpoint to well performing reservoir computers for the class B and class A laser system. Namely, the linear memory capacity has larger values in the absence of resonances $\Phi \approx k\pi$ and for values of Λ closer to 1. We now want to expand this knowledge to the case of a laser with external feedback.

3.2. Laser with feedback

We now expand our results to the infinite-dimensional phase space of a semiconductor system with delay, i.e. the Lang–Kobayashi system. In [9, 22, 26, 45, 46] it was shown that resonances between τ and T often decrease memory capacity and thus reservoir computing performance. Here we look at this phenomenon from another point of view, namely, as a resonance between T and the imaginary parts of the eigenvalues. We use the resonance property described in section 2.5: for certain resonant values of T, the product $\Im(\lambda)T$ is proportional to an integer number of π for all critical eigenvalues simultaneously. We computed the first 100 eigenvalues using DDE-biftool for the Lang–Kobayashi system. By superimposing all Φ_i, where $i \in 0, 1, 2..., N$

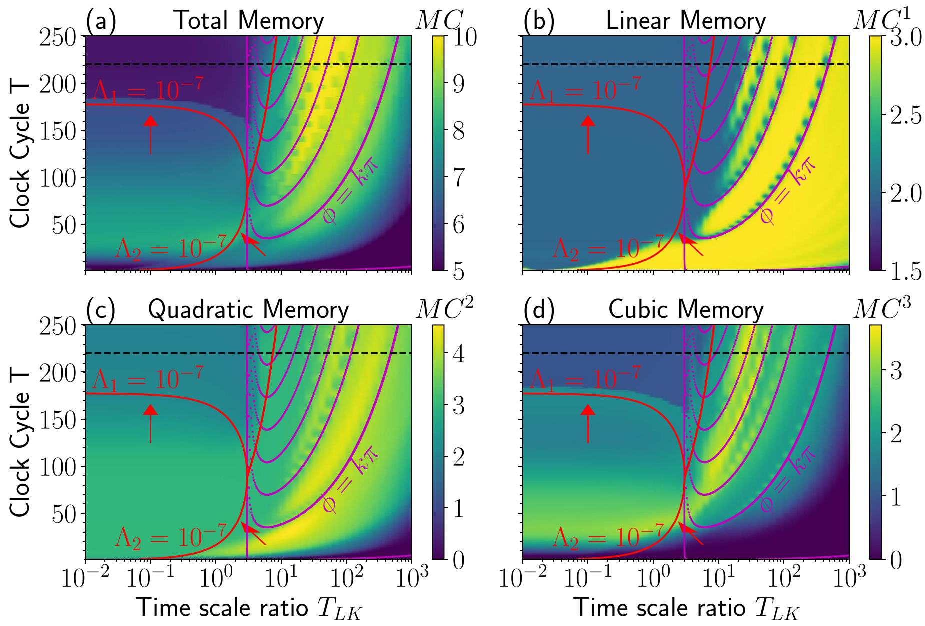

Figure 4. 2D parameter scan in the plane of clock cycle T and logarithmic life time scale ratio T_{LK} showing the total, linear, quadratic, and cubic memory capacities as a colour code in panel (a)–(d). The purple and two red solid lines show the parameter values where $\Im(\lambda)T \approx k\pi$, $k = 1, 2, \ldots$, and $\Lambda_{1,2} = e^{\Re(\lambda_{1,2})T} \approx 10^{-7}$, respectively. $\lambda_{1,2}$ are the two eigenvalues of the Class B system shown in figure 3. The dashed black line indicates the parameter scan used in figure 3. Other parameters as in figure 3.

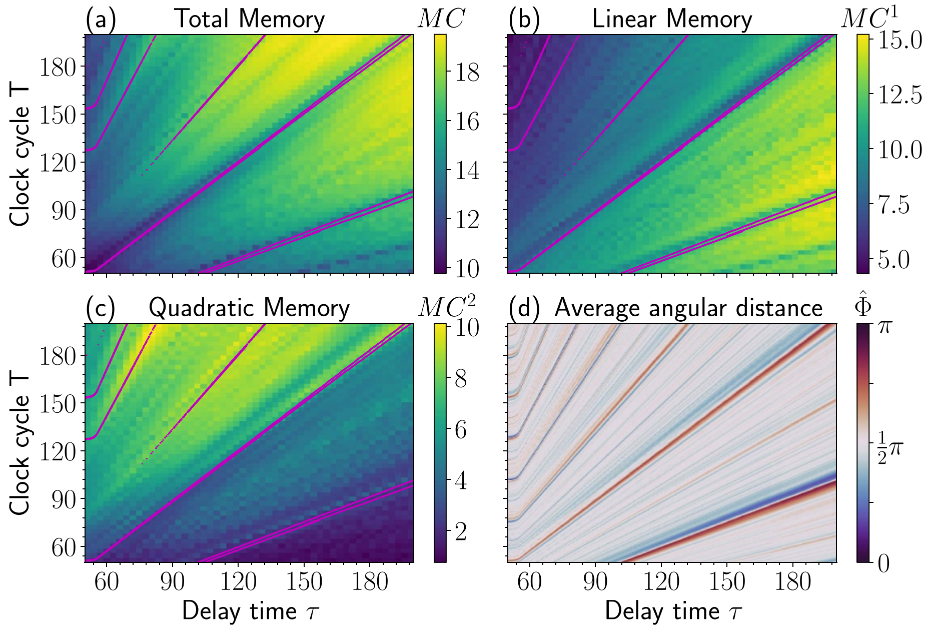

Figure 5. 2D parameter scan in the plane of clock cycle T and delay τ with $T_{LK} = 100$, $N_V = 20$, $P = 0.01$, $\kappa = 0.1$. Colour coded is (a) total, (b) linear, and (c) quadratic memory capacities. (d) Average angular distance $\hat{\Phi}$ given by equation (20) for the first 100 eigenvalues. Parameter values where $\hat{\Phi} \approx k\pi$ are shown by solid purple lines in panels (a)–(c).

is the index of the ith eigenvalue, we evaluate the resonance effects of the strongest N eigendirections by computing the average angular distance:

$$\hat{\Phi} = \frac{1}{N}\sum_{i=0}^{N}\Phi_i, \tag{20}$$

and compare the results with the linear, quadratic, and total memory capacities. The comparison of the memory capacities and $\hat{\Phi}$ is shown in figure 5, where a 2D parameter scan is plotted in the parameter plane of the delay time τ and the clock cycle T. Bright regions in (a–c) correspond to high memory capacities,

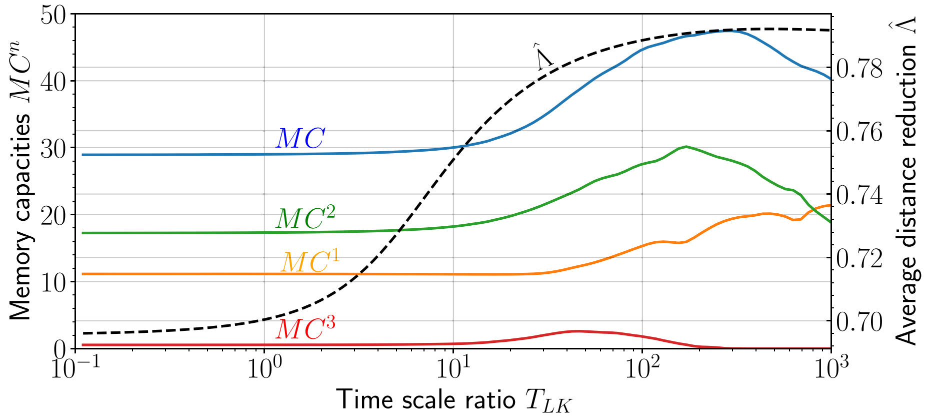

Figure 6. Linear, quadratic, cubic, and total memory capacities are shown as orange, green, red, and blue lines as a function of the lifetime scale ratio T_{LK}. The sum of the distance reductions of the first 100 eigenvalues $\hat{\Lambda} = \frac{1}{N}\sum_{i=0}^{100} e^{\Re(\lambda_i)T}$ is plotted as a dashed black line. The increase of $\hat{\Lambda}$ coincides with the linear memory capacity. Parameters are $P = 0.05$, $\kappa = 0.1$, $T = 410$, $\tau = 500$, $N_V = 50$.

while dark blue to low memory capacities. Panel (d) shows the results for $\hat{\Phi}$ for the first 100 eigenvalues. Values close to 0 or π indicate parameters where all leading eigendirections possess resonant eigenvalues, i.e. $\Im(\lambda)T \approx k\pi$, and perform an integer of half circle rotations during one input time T. The solid purple lines in figures 5(a)–(c) indicate the resonant values $\hat{\Phi} \approx k\pi$. A match with lower total memory capacities, especially for the linear memory is clear. For a reservoir to be applicable to many tasks, a higher total memory capacity is desirable, thus a decrease in the total memory capacity decreases the applicability of the reservoir to many different tasks. Our results support the fact that the clock cycle T should be chosen to be off resonant of the delay time τ. The eigenvalue analysis gives an additional explanation and intuition for why this is the case. Taking into account the resonance effect and our results from [45], we set the delay time $\tau \approx \sqrt{2}T$ for all following simulations.

As we have seen in section 3.1 for a 2D reservoir, the reservoir performance decreases when the real part of the eigenvalues becomes strongly negative. In such a case, the reservoir 'forgets' the input too fast. Here we extend this idea to the case of the infinite-dimensional reservoir.

As long as the perturbation from the information fed into the system is small enough, one can think of all eigenvalues and their corresponding eigendirections as the available phase space of the reservoir computer. Thus, a higher phase space volume can lead to a more promising reservoir computer. We introduce the average of the distance reduction Λ by:

$$\hat{\Lambda} = \frac{1}{N}\sum_{k=0}^{N} e^{\Re(\lambda_k)T}. \tag{21}$$

It describes the average distance reduction of the N slowest eigendirections. Since only a finite number of complex eigenvalues lie to the right of a line parallel to the imaginary axis, all eigendirections except a finite number are strongly contracting, i.e. possess strongly negative real parts [67]. This implies the possibility of considering only a finite number N of eigenvalues in equation (21).

Figure 6 depicts the memory capacities and the distance reduction $\hat{\Lambda}$ as a function of the timescale ratio T_{LK}, or in other words, the evolution of the memory capacities along the transition from a Class A to a Class B laser system with delayed feedback. Similarly to the case without feedback in figure 3, the memory capacity stays about constant for $T_{LK} \lesssim 2$, and increases when the additional dimensions become available by the reservoir for $T_{LK} \gtrsim 2$. The increase of $\hat{\Lambda}$ coincides with the increase of the linear memory. The higher orders show a similar trend, but are, in general, more involved and should be investigated more deeply. Thus, the knowledge of the eigenvalues provides a qualitative prediction of the linear memory capacity.

To give a broader overview, we perform a 2D parameter scan along the feedback strength κ and pump P (shown in figure 7) and plot the linear, quadratic, and total memories as a colour-code. Bright regions correspond to high memory capacities, and dark to low memory capacities. Additionally, in figure 7(d), the sum of the average distance reduction $\hat{\Lambda}$ is color-coded within the same 2D parameter plane spanned by κ and P. Comparing the three memory capacity scans with $\hat{\Lambda}$, we can see a close relationship between them. Thus, $\hat{\Lambda}$ is a very good indicator for choosing well-performing reservoir computers. This saves a lot of computational efforts, as the eigenvalues can be computed in a fraction of the time needed to compute the memory capacities.

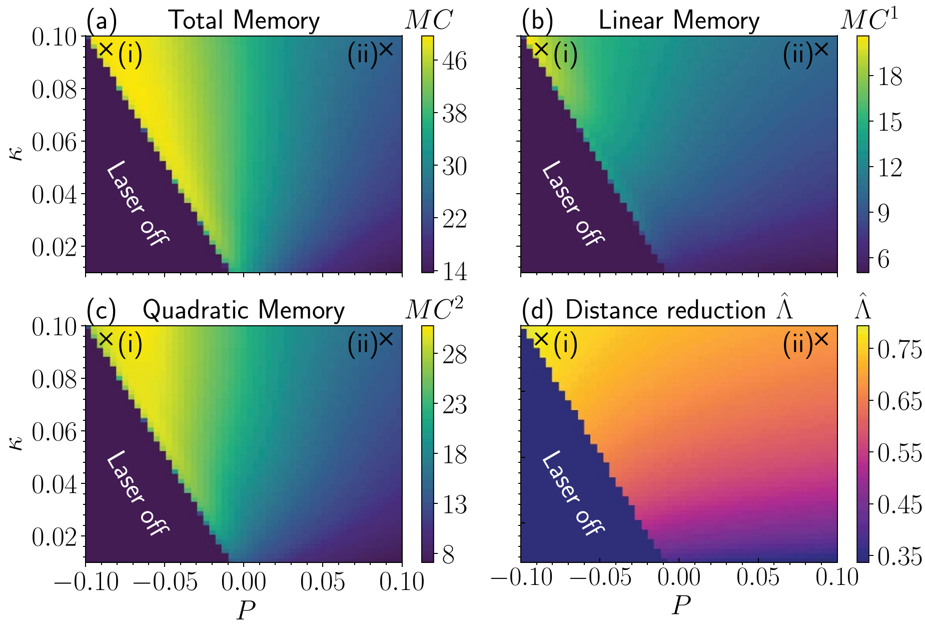

Figure 7. 2D parameter scan in the plane of feedback strength κ and pump P. Colour coded are the linear (a), quadratic (b), total memory (c) capacities, and the (d) average distance reduction $\hat{\Lambda} = \frac{1}{N}\sum_{i=0}^{N} e^{\Re(\lambda_i)T}$ for the first 100 eigenvalues $N = 100$. The two crosses indicate the parameter values used in figure 9. Parameters are $T_{LK} = 1.0$, $T = 350$ and $\tau = 500$, $N_V = 50$.

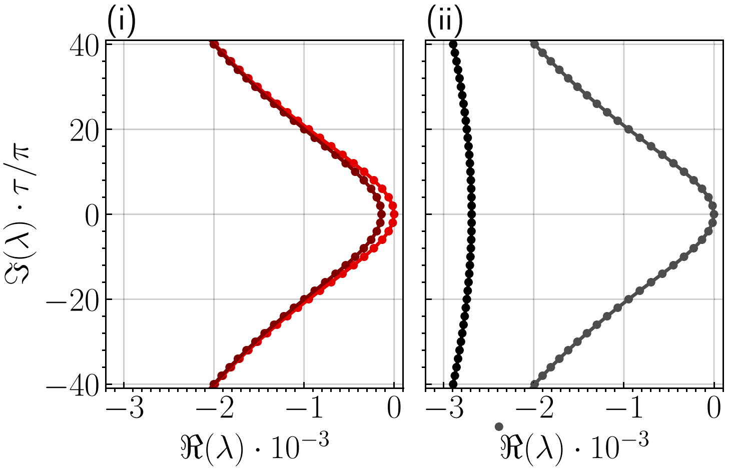

Figure 8. Pseudocontinuous eigenvalue spectra for two different parameter sets given by equation (15) (bright) and equation (16) (dark) plotted for (i) $P = -0.095$ and (ii) $P = 0.095$. $\kappa = 0.1$, $\tau = 500$, and $T_{LK} = 1.0$. The value of $T_{LK} = 1.0$ corresponds to a solitary laser operating between Class A and Class B. For pump values slightly above threshold $P_{th} = -\kappa$ (i), the eigenvalue spectrum has more eigenvalues with real parts close to 0.

To illustrate possible configurations of eigenvalues and their connection to $\hat{\Lambda}$, we chose two different parameter setups in figure 8: $P = -0.095$ (a) and $P = 0.095$ (b), with the other parameters fixed. The two parameter setups are marked as black crosses in figure 7, which correspond to parameters close to and well above threshold. The first parameter set (figure 8(i)) corresponds to an eigenvalue spectrum for an operation point close above the threshold with a low power output. Here the laser system possesses more eigenvalues with real parts close to 0, thus it has many slowly contracting eigendirections which means $\hat{\Lambda}$ is closer to 1. Calculating $\hat{\Lambda}$ for the first parameter set yields $\hat{\Lambda} = 0.75$. The second parameter set (figure 8(ii)) corresponds to a laser that is operated high above threshold. This laser has less slowly contracting eigendirections, i.e. $\hat{\Lambda}$ is closer to 0. Calculating $\hat{\Lambda}$ for the second parameter set yields $\hat{\Lambda} = 0.6$.

Now we use the insights gained from the distance reduction Λ and from the angular distance Φ and test the reservoir computer performances for the two parameter sets from figures 8(i) and (ii), marked as black crosses in figure 7. The performance is quantified by evaluating both the memory capacity and the prediction error (NRMSE) for the NARMA10 task shown in figures 9(b) and (a).

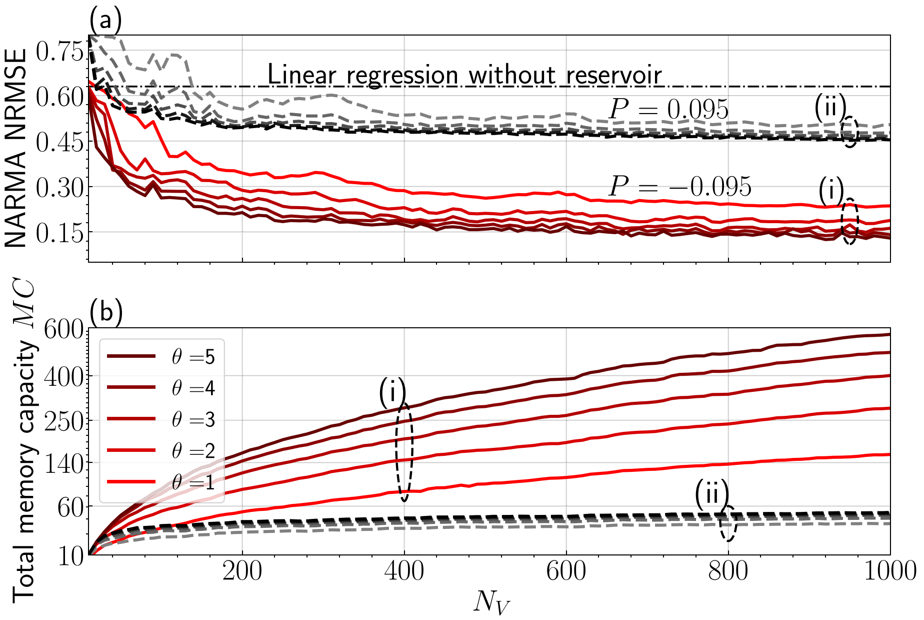

Figure 9. Computation error NARMA10 NRMSE (a) and the total memory capacity (b) for the two parameter values: $P = 0.095$ (dashed lines) and $P = -0.095$ (solid lines), see also crosses in figure 7. Other parameters: $\tau = 1.41T$, $\kappa = 0.1$, and $T_{LK} = 1.0$. On the x axis the number of virtual nodes N_V is shown. Different brightness correspond to different distances θ between the virtual nodes. The dashed black line shows the minimum reached with a linear regression without a reservoir.

On the horizontal axis, we change the number of virtual nodes N_V, i.e. we increase the number of readout dimensions, which should, naively thinking, increase the performance of the reservoir. While we do this, we keep the distance between the virtual nodes θ the same for 5 different cases of θ from $\theta = 1$ up to $\theta = 5$, shown in black and red with decreasing brightness for the optimized (figure 8(i)) and not-optimized (figure 8(ii)) point respectively. Increasing the virtual node distance θ should reduce the linear dependency of the nodes, as the time between two responses is increased. This is obviously dependent on the reaction time of the system, which is also given by the eigenvalues of the system. Thus the influence of increasing θ on the slowly contracting eigendirections (figure 8(i)) is pronounced compared to the one with fast eigendirection (figure 8(ii)). The increase of θ also effectively increases the clock cycle $T = N_V\theta$ and thus the delay time $\tau = 1.41T$. We want to emphasize, that this does not alter the general trend of $P = -0.095$ (figure 8(i)) having many slowly contracting eigendirections compared to $P = 0.095$ (figure 8(ii)).

The results indicate that even though the number of virtual nodes increases, the NARMA10 error for the case with the low distance reduction $\hat{\Lambda}$ (ii) does not go below 0.45. On the other hand the case with the high $\hat{\Lambda}$, our optimal case (i), reaches very small errors below 0.15, a factor 3 better than the low distance reduction $\hat{\Lambda}$ (ii) case. We also want to emphasize that the simulation was done for a high noise value of $D_{\text{noise}} = 10^{-7}$. Simulating the system without any noise $D_{noise} = 0$ NARMA10 errors (NRMSE) of down to 0.05 were reached. We conclude that a high distance reduction is very beneficial for the performance.

As a dashed black line in figure 9(a) we additionally show the minimum reached by a linear regression without a reservoir. Every reservoir setup with results below this line has higher performance and thus can be considered an improvement on the NARMA10 task. We included this here to emphasize the reduction of the NARMA10 error (NRMSE) by the eigenvalue analysis, for which an improvement of about a factor 4 is reached to the linear regression without reservoir.

The total memory capacities for the two cases (i) and (ii) are shown in figure 9(b). We can see the same trend: the memory capacity reaches a limit for the case with low $\hat{\Lambda}$, whereas the improved case with the highest $\hat{\Lambda}$ increases in its memory capacity further for higher N_V. The results suggest that operation points with high distance reduction $\hat{\Lambda}$ (solid lines in figure 9) pinpoint to well-performing reservoir computers.

4. Conclusion

We have shown that the eigenvalue spectrum analysis of a dynamical system used as a reservoir (e.g. a laser described by the Lang–Kobayashi system) is capable of predicting good reservoir computing operation points. Because of the available analytical and numerical tools for the description of the eigenvalue spectrum, such analysis can be readily applied for different dynamical systems, which are used as reservoirs with operating points close to an equilibrium. The eigenvalue method is of magnitudes faster to compute and could help in numerically predicting good reservoir computers for experimental setups.

Due to the relation between the eigenvalue spectrum and the performance of the delay-based reservoir computing, the central message of this paper is twofold: first, the eigenvalues must be off-resonant, where the resonance condition is given in terms of the imaginary parts of the eigenvalues. Namely, the product of the latter with the input clock-cycle should be away from values of $k\pi$. Importantly, such resonances appear for all critical eigenvalues at almost the same parameter values, due to general properties of the spectrum of delay systems with large delays [59]. Therefore, such an off-resonant condition plays an important role even when the reservoir's effective dimensionality is high.

The second conclusion is that, for optimal performance, the spectrum must be close to criticality. This closeness is measured by the real part of the eigenvalue spectrum, which should be close to zero and negative. In this paper, we propose the average distance reduction as a measure $\hat{\Lambda}$ for such closeness, which is given by equation (21).

For future work we would like to analyse clearer properties of the reservoir via the eigenvalue spectrum. We believe that the timescale analysis, i.e. eigenvalue analysis, has more advantages. The eigenvalues do not only yield the timescale on which the system forgets, but also the timescale on which the system learns (for accordingly small inputs). Thus, through the eigenvalues we can define a system which is either fast and has short term memory or we can define a system like we did in this paper, which is slow and has more memory capacity further into the past.

Appendix A. Memory capacity expression (3)

We show how the expression for the memory capacity from [40] can be rewritten in the form of equation (3). From Dambre *et al* [40], the capacity to approximate a target data $\hat{\mathbf{y}}$ is given by:

$$C_L[\mathsf{S},\hat{\mathbf{y}}] = \frac{\sum_{ij}\langle \hat{y}s_i\rangle_L \langle s_is_j\rangle_L^{-1}\langle s_j\hat{y}\rangle_L}{\langle \hat{y}^2\rangle_L}. \tag{A.1}$$

Here s_i is the ith readout of the M system responses for the lth input-target pair, $\langle \nu\rangle_L = \frac{1}{L}\sum_{l=1}^{L}\nu_l$ is the average over all input-output paris L and $\langle s_is_j\rangle_L^{-1}$ is the inverse of $\langle s_is_j\rangle_L$. One can insert the average over all input-output pairs yielding:

$$C_L[\mathsf{S},\hat{\mathbf{y}}] = \frac{\sum_{ij}\sum_{l=1}^{L}\hat{y}_l s_{i,l}(\sum_{l=1}^{L} s_{i,l}s_{j,l})^{-1}\sum_{l=1}^{L} s_{j,l}\hat{y}_l}{\sum_{l=1}^{L}\hat{y}_l^2}. \tag{A.2}$$

In the denominator $\sum_{l=1}^{L}\hat{y}_l^2$ can be substituted with the square norm of the target vector $\sum_{l=1}^{L}\hat{y}_l^2 = ||\hat{\mathbf{y}}||^2$. The first term $\hat{y}_l s_{i,l}$ is the ith system response to the lth input-output pair (ith column and lth row) multiplied with the lth target. Summing over all input–output pairs L, this is the same as the ith entry of the matrix product:

$$\sum_{l=1}^{L}\hat{y}_l s_{i,l} = (\hat{\mathbf{y}}^T\mathsf{S})_i. \tag{A.3}$$

The same reasoning applies to $(\sum_{l=1}^{L} s_{i,l}s_{j,l})$ yielding in:

$$\left(\sum_{l=1}^{L} s_{i,l}s_{j,l}\right)^{-1} = (\mathsf{S}^T\mathsf{S})_{i,j}^{-1} \tag{A.4}$$

$\sum_{l=1}^{L} s_{j,l}\hat{y}_l$ is just the transposed case of $\sum_{L=1}^{L}\hat{y}_l s_{i,l}$, thus $\sum_{l=1}^{L} s_{j,l}\hat{y}_l = (\mathsf{S}^T\hat{\mathbf{y}})_j$. Summing over all responses i and j is equivalent to the matrix product:

$$\sum_{ij}(\hat{\mathbf{y}}^T\mathsf{S})_i(\mathsf{S}^T\mathsf{S})_{i,j}^{-1}(\mathsf{S}^T\hat{\mathbf{y}})_j = \hat{\mathbf{y}}^T\mathsf{S}(\mathsf{S}^T\mathsf{S})^{-1}\mathsf{S}^T\hat{\mathbf{y}}, \tag{A.5}$$

with which we have reached equation (3):

$$\mathrm{C} = \frac{\hat{\mathbf{y}}^T\mathsf{S}(\mathsf{S}^T\mathsf{S})^{-1}\mathsf{S}^T\hat{\mathbf{y}}}{||\hat{\mathbf{y}}||^2}. \tag{A.6}$$

Data availability statement

The data generated and/or analysed during the current study are not publicly available for legal/ethical reasons but are available from the corresponding author on reasonable request.

Acknowledgments

The authors thank Florian Stelzer and Mirko Goldmann for fruitfull discussions. This study was funded by the 'Deutsche Forschungsgemeinschaft' (DFG) in the framework of SFB910 and Project No. 411803875.

ORCID iDs

Felix Köster https://orcid.org/0000-0002-3577-3690
Serhiy Yanchuk https://orcid.org/0000-0003-1628-9569
Kathy Lüdge https://orcid.org/0000-0002-4831-8910

References

[1] Jaeger H 2001 The 'echo state' approach to analysing and training recurrent neural networks GMD Report 148 GMD—German National Research Institute for Computer Science

[2] Maass W, Natschläger T and Markram H 2002 Real-time computing without stable states: a new framework for neural computation based on perturbations *Neural Comput.* **14** 2531

[3] Gonon L and Ortega J P 2020 Reservoir computing universality with stochastic inputs *IEEE Trans. Neural Netw. Learn. Syst.* **31** 100–12

[4] Fernando C and Sojakka S 2003 Pattern recognition in a bucket *Advances in Artificial Life* (Berlin: Springer) pp 588–97

[5] Antonik P, Duport F, Hermans M, Smerieri A, Haelterman M and Massar S 2016 Online training of an opto-electronic reservoir computer applied to real-time channel equalization *IEEE Trans. Neural Netw. Learn. Syst.* **28** 2686–98

[6] Dockendorf K, Park I, He P, Principe J C and DeMarse T B 2009 Liquid state machines and cultured cortical networks: the separation property *Biosystems* **95**

[7] Gallicchio C, Micheli A and Silvestri L 2018 Local Lyapunov exponents of deep echo state networks *Neurocomputing* **298** 34

[8] Gallicchio C and Micheli A 2019 Lecture Notes in Computer Science *Richness of Deep Echo State Network Dynamics* vol 11506 (Switzerland: Springer Nature)

[9] Röhm A and Lüdge K 2018 Multiplexed networks: reservoir computing with virtual and real nodes *J. Phys. Commun.* **2** 085007

[10] Goldmann M, Köster F, Lüdge K and Yanchuk S 2020 Deep time-delay reservoir computing: dynamics and memory capacity *Chaos* (arXiv:2006.06322v1) (submitted)

[11] Bauduin M, Smerieri A, Massar S and Horlin F 2015 Equalization of the non-linear satellite communication channel with an echo state network *2015 IEEE 81st Vehicular Conf. (VTC Spring)*

[12] Keuninckx L, Danckaert J and Van der Sande G 2017 Real-time audio processing with a cascade of discrete-time delay line-based reservoir computers *Cogn. Comput.* **9** 315–26

[13] Scardapane S and Uncini A 02 2017 Semi-supervised echo state networks for audio classification *Cogn. Comput.* **9** 125–35

[14] Argyris A, Bueno J, Soriano M C and Fischer I 2017 Improving detection in optical communications using all-optical reservoir computing *2017th Conf. Lasers and Electro-Optics Europe European Quantum Conf. (CLEO/Europe-EQEC)* p 1

[15] Argyris A, Bueno J and Fischer I 2018 Photonic machine learning implementation for signal recovery in optical communications *Sci. Rep.* **8** 1–13

[16] Amil P, Soriano M C and Masoller C 2019 Machine learning algorithms for predicting the amplitude of chaotic laser pulses *Chaos* **29** 113111

[17] Pathak J, Hunt B, Girvan M, Lu Z and Ott E 2018 Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach *Phys. Rev. Lett.* **120** 024102

[18] Pathak J, Wikner A, Fussell R, Chandra S, Hunt B, Girvan M and Ott E 2018 Hybrid forecasting of chaotic processes: using machine learning in conjunction with a knowledge-based model (arXiv:1803.04779)

[19] Cunillera A, Soriano M C and Fischer I 2019 Cross-predicting the dynamics of an optically injected single-mode semiconductor laser using reservoir computing *Chaos* **29** 113113

[20] Vandoorne K, Mechet P, Van Vaerenbergh T, Fiers M, Morthier G, Verstraeten D, Schrauwen B, Dambre J and Bienstman P 2014 Experimental demonstration of reservoir computing on a silicon photonics chip *Nat. Commun.* **5** 3541

[21] Larger L, Soriano M C, Brunner D, Appeltant L, Gutierrez J M, Pesquera L, Mirasso C R and Fischer I 2012 Photonic information processing beyond turing: an optoelectronic implementation of reservoir computing *Opt. Express* **20** 3241–9

[22] Paquot Y, Duport F, Smerieri A, Dambre J, Schrauwen B, Haelterman M and Massar S 2012 Optoelectronic reservoir computing *Sci. Rep.* **2** 287

[23] Brunner D, Soriano M C, Mirasso C R and Fischer I 2013 Parallel photonic information processing at gigabyte per second data rates using transient states *Nat. Commun.* **4** 1364

[24] Vinckier Q, Duport F, Smerieri A, Vandoorne K, Bienstman P, Haelterman M and Massar S 2015 High-performance photonic reservoir computer based on a coherently driven passive cavity *Optica* **2** 438–46

[25] Nguimdo R M, Lacot E, Jacquin O, Hugon O, Van der Sande G and de Chatellus H G 2017 Prediction performance of reservoir computing systems based on a diode-pumped erbium-doped microchip laser subject to optical feedback *Opt. Lett.* **42** 375–78

[26] Röhm A, Jaurigue L C and Lüdge K 2019 Reservoir computing using laser networks *IEEE J. Sel. Top. Quantum Electron.* **26** 7700108

[27] Bueno J, Maktoobi S, Froehly L, Fischer I, Jacquot M, Larger L and Brunner D 2018 Reinforcement learning in a large-scale photonic recurrent neural network *Optica* **5** 756–60

[28] Bueno J, Brunner D, Soriano M C and Fischer I 2017 Conditions for reservoir computing performance using semiconductor lasers with delayed optical feedback *Opt. Express* **25** 2401–12

[29] Nakayama J, Kanno K and Uchida A 2016 Laser dynamical reservoir computing with consistency: an approach of a chaos mask signal *Opt. Express* **24** 8679–92

[30] Appeltant L, Soriano M C, Van der Sande G, Danckaert J, Massar S, Dambre J, Schrauwen B, Mirasso C R and Fischer I 2011 Information processing using a single dynamical node as complex system *Nat. Commun.* **2** 468

[31] Ortin S and Pesquera L 2017 Reservoir computing with an ensemble of time-delay reservoirs *Cogn. Comput.* **9** 327–36

[32] Dion G, Mejaouri S and Sylvestre J 2018 Reservoir computing with a single delay-coupled non-linear mechanical oscillator *J. Appl. Phys.* **124** 152132

[33] Brunner D, Penkovsky B, Marquez B A, Jacquot M, Fischer I and Larger L 2018 Tutorial: photonic neural networks in delay systems *J. Appl. Phys.* **124** 152004

[34] Chen Y, Yi L, Ke J, Yang Z, Yang Y, Huang L, Zhuge Q and Hu W 2019 Reservoir computing system with double optoelectronic feedback loops *Opt. Express* **27** 27431–40

[35] Hou Y S, Xia G Q, Yang W Y, Wang D, Jayaprasath E, Jiang Z, Hu C X and Wu Z M 2018 Prediction performance of reservoir computing system based on a semiconductor laser subject to double optical feedback and optical injection *Opt. Express* **26** 10211–19

[36] Sugano C, Kanno K and Uchida A 2020 Reservoir computing using multiple lasers with feedback on a photonic integrated circuit *IEEE J. Sel. Top. Quantum Electron.* **26** 1500409

[37] Kuriki Y, Nakayama J, Takano K and Uchida A 2018 Impact of input mask signals on delay-based photonic reservoir computing with semiconductor lasers *Opt. Express* **26** 5777–88

[38] Argyris A, Cantero J, Galletero M, Pereda E, Mirasso C R, Fischer I and Soriano M C 2020 Comparison of photonic reservoir computing systems for fiber transmission equalization *IEEE J. Sel. Top. Quantum Electron.* **26** 5100309

[39] Larger L, Baylón-Fuentes A, Martinenghi R, Udaltsov V S, Chembo Y K and Jacquot M 2017 High-speed photonic reservoir computing using a time-delay-based architecture: million words per second classification *Phys. Rev.* X **7** 011015

[40] Dambre J, Verstraeten D, Schrauwen B and Massar S 2012 Information processing capacity of dynamical systems *Sci. Rep.* **2** 514

[41] Harkhoe K and Van der Sande G 2019 Task-independent computational abilities of semiconductor lasers with delayed optical feedback for reservoir computing *Photonics* **6** 124

[42] Brunner D, Soriano M, Van der Sande G, Dambre J, Bienstman P, Larger L, Pesquera L and Massar S 2019 *Photonic Reservoir Computing: Optical Recurrent Neural Networks* vol 07 (Berlin: De Gruyter) (https://doi.org/10.1515/9783110583496)

[43] Van der Sande G, Brunner D and Soriano M C 2017 Advances in photonic reservoir computing *Nanophotonics* **6** 561

[44] Tanaka G, Yamane T, Héroux J B, Nakane R, Kanazawa N, Takeda S, Numata H, Nakano D and Hirose A 2018 Recent advances in physical reservoir computing: a review *Neural Netw.* **115** 100–23

[45] Köster F, Ehlert D and Lüdge K 2020 Limitations of the recall capabilities in delay based reservoir computing systems *Cogn. Comput.* accepted (https://doi.org/10.1007/s12559-020-09733-5)

[46] Stelzer F, Röhm A, Lüdge K and Yanchuk S 2020 Performance boost of time-delay reservoir computing by non-resonant clock cycle *Neural Netw.* **124** 158–69

[47] Jaeger H, Lukoševičius M, Popovici D and Siewert U 2007 Optimization and applications of echo state networks with leaky-integrator neurons *Neural Netw.* **20** 335–52

[48] Alsing P M, Kovanis V, Gavrielides A and Erneux T 1996 Lang and Kobayashi phase equation *Phys. Rev.* A **53** 4429–34

[49] Heil T, Fischer I and Elsäßer W 1999 Influence of amplitude-phase coupling on the dynamics of semiconductor lasers subject to optical feedback *Phys. Rev.* A **60** 634

[50] Erneux T, van Tartwijk G H M, Lenstra D and Levine A M 1995 Determining lang and Kobayashi Hopf bifurcation points *Proc. SPIE* **2399** 170–81

[51] Rottschäfer V and Krauskopf B 2007 The ECM-backbone of the Lang–Kobayashi equations: a geometric picture *Int. J. Bifurcation Chaos* **17** 1575–88

[52] Lang R and Kobayashi K 1980 External optical feedback effects on semiconductor injection laser properties *IEEE J. Quantum Electron.* **16** 347–55

[53] Yanchuk S and Wolfrum M 2010 A multiple time scale approach to the stability of external cavity modes in the Lang–Kobayashi system using the limit of large delay *SIAM J. Appl. Dyn. Syst.* **9** 519

[54] Erneux T, Viktorov E A and Mandel P 2007 Time scales and relaxation dynamics in quantum-dot lasers *Phys. Rev.* A **76** 023819

[55] O'Brien D, Hegarty S P, Huyet G and Uskov A V 2004 Sensitivity of quantum-dot semiconductor lasers to optical feedback *Opt. Lett.* **29** 1072–4

[56] Lingnau B and Lüdge K 2015 Analytic characterization of the dynamic regimes of quantum-dot lasers *Photonics* **2** 402–13

[57] Wang C, Grillot F, Kovanis V and Even J 2013 Rate equation analysis of injection-locked quantum cascade lasers *J. Appl. Phys.* **113** 063104

[58] Columbo L and Brambilla M 2014 Multimode regimes in quantum cascade lasers with optical feedback *Opt. Express* **22** 10105–18

[59] Lichtner M, Wolfrum M and Yanchuk S 2011 The spectrum of delay differential equations with large delay *SIAM J. Math. Anal.* **43** 788–802

[60] Engelborghs K, Luzyanina T and Roose D 2002 Numerical bifurcation analysis of delay differential equations using DDE-Biftool *ACM Trans. Math. Softw.* **28** 1–21

[61] Sieber J, Engelborghs K, Luzyanina T, Samaey G and Roose D 2014 *DDE-BIFTOOL manual—bifurcation analysis of delay differential equations*

[62] Janssens S 2010 On a normalization technique for codimension two bifurcations of equilibria of delay differential equations PhD Thesis

[63] Yanchuk S 2005 Properties of stationary states of delay equations with large delay and applications to laser dynamics *Math. Methods Appl. Sci.* **28** 363–77

[64] Yanchuk S, Lücken L, Wolfrum M and Mielke A 2015 Spectrum and amplitude equations for scalar delay-differential equations with large delay *Discrete Cont. Dyn. Syst.* A **35** 537–53

[65] Arecchi F T, Lippi G L, Puccioni G P and Tredicce J R 1984 Deterministic chaos in laser with injected signal *Opt. Commun.* **51** 308–15

[66] Sanderson C and Curtin R 2016 Armadillo: a template-based C++ library for linear algebra *J. Open Source Softw.* **1** 26

[67] Hale J K and Verduyn Lunel S M 1993 *Introduction to Functional Differential Equations* (New York: Springer)