Optimized Temporal Multiplexing for Reservoir Computing with a Single Delay-Coupled Node

Hazem Toutounji, Johannes Schumacher, Gordon Pipa

Department of Neuroinformatics, Institute of Cognitive Science
University of Osnabrück, Osnabrück, Germany
Email: {htoutounji,joschuma,gpipa}@uni-osnabrueck.de

Abstract—The computational performance of reservoir computers based on a single delay-coupled node is critically dependent on the temporal multiplexing of input to the reservoir. Here we present an optimization of the temporal multiplexing by means of optimizing virtual node distance to maximize the response of the delay-coupled system to stimulation. After demonstrating the analytical approach, we discuss how the optimization has a single optimum (concave problem), and illustrate the improvement of the reservoir computer's performance. To this end we predict a NARMA-10 time series and show that optimizing temporal multiplexing reduces the normalized root mean squared error by $\sim 8\%$.

1. Introduction

The dynamics of complex systems have successfully been used in recent years to implement a novel computational paradigm called reservoir computing [1]. This paradigm uses the nonlinearity and fading memory of the dynamics to nonlinearly mix past and recent activity, thereby allowing for nonlinear computation, such as time series prediction or classification. To achieve a high flexibility of the reservoir computer, many different kinds of such nonlinear mixing are usually necessary. To this end, classical reservoir computing utilizes networks of nonlinear elements, with each node combining past and recent states differently [2, 3].

Recently, it was demonstrated that the same reservoir computing concept can also be realized with a single delay-coupled nonlinear node [4]. Compared to classical reservoir computing using a network, the single delay-coupled node reservoir computer is especially appealing for technical implementation, i.e. electronic or optical, since it requires only two elements, first, a single nonlinear node and, second, a delay loop. To access different nonlinear mixing of signals from the past and the present, the single-node delay-coupled reservoir computer uses virtual nodes (Fig.1). Every virtual node has the analog role to the nodes of a recurrently connected network. While nodes of a network are mixing signals via their network coupling defined by the network topology, the virtual nodes of a delay-coupled reservoir are mixing signals via temporal correlation of the dynamics of the delay-coupled system. Therefore, the virtual node distance is made shorter than the characteristic time scale of the nonlinear node, which in turn renders the virtual node activity dependent, i.e. nonlinearly dependent. Thus, the temporal correlations of the virtual node states and their dependence on the relative position, i.e. delay between nodes, are analogous to the connections and topology of a recurrent network, or in other words, a network coupling used by classical reservoir computing is temporally multiplexed.

To process information, an external signal is applied to the dynamical system and thereby perturbing the reservoir dynamics. Here, we operate the delay-coupled node in an asymptotically-stable fixed-point state. To render the response of the delay-coupled system transient, i.e. reflecting nonlinear combinations of past and recent states, the reservoir is perturbed by a stimulus that is transient itself. To ensure such transient stimulus input, a masking of the stimulus with an alternating sequence was proposed [4]. The positions of that masking random sequence matches the positions of the virtual nodes.

The first time such single-node reservoir computers had been proposed, the delays between virtual nodes had been chosen equidistantly. However, given the fact that these relative delays directly influence the correlation of the corresponding virtual node states, and therefore also the nonlinear mixing of the signals, it is immediately evident that the node distance and position are important parameters that may significantly influence the performance of the reservoir computer. In this paper we present and study the optimization of this relative node spacing using an objective function that maximizes the virtual nodes' responsiveness to input.

2. Methods

2.1. Reservoir activity as a function of node delays

The goal is to optimize the computational properties of a delay coupled node, a reservoir, given a vector of temporal spacings, $\Theta = (\theta_1, \cdots, \theta_i, \cdots, \theta_n)$, of its n virtual nodes. To that end, we first need to define the activity $x(t)$ of the reservoir given θ_i.

The reservoir is a delay-coupled input-driven dynamical system that is defined by the forced Delay Differential Equation (DDE):

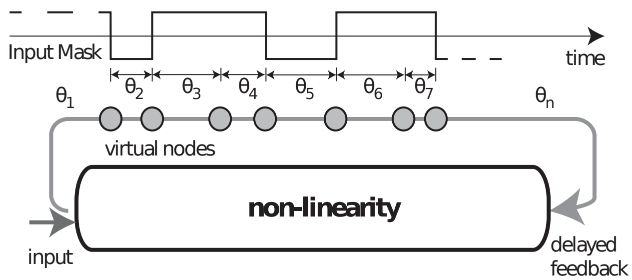

Figure 1: Schema of a single delay-coupled node as a reservoir computer. Input is multiplexed in time across a delay line of length τ by using a random binary mask of n bits. Each mask bit M_i is held constant for a short delay θ_i such that the sum of these delays is the length of the delay line τ. The masked input is then transfered by a nonlinear node and mixed nonlinearly with past inputs through feedback to the nonlinearity. At the end of each delay θ_i resides a virtual node from which linear readouts learn to extract information and perform online computations through linear regression.

$$\dot{x}(t) = -x(t) + f\Big(x(t-\tau), M \cdot u(t)\Big) \tag{1}$$

τ being the delay time, M the mask, and $u(t)$ the input which is held constant across τ. At a particular virtual node i, and given time in discrete units of τ, activity at i can be written as:

$$\dot{x}_i^{(t)} = -x_i^{(t)} + f_i^{(t)} \tag{2}$$

where $f_i^{(t)} = f\Big(x_i^{(t-1)}, M_i \cdot u^{(t)}\Big)$. Since $\theta_i \ll \tau$, $f_i^{(t)}$ can be assumed piecewise constant at each θ_i, the DDE for each virtual node can be approximated by a linear ODE with solution:

$$x_i^{(t)} = e^{-\theta_i} x_{i-1}^{(t)} + (1 - e^{-\theta_i}) f_i^{(t)} \tag{3}$$

By iteratively substituting $x_{i-1}^{(t)}$ by the approximate solution at node $i-1$ from (3), we can rewrite the reservoir activity at a node i as a function of $\{\theta_1, \cdots, \theta_i\}$:

$$x_i^{(t)} = e^{-\sum_{j=1}^{i}\theta_j} x_n^{(t-1)} + \sum_{j=1}^{i}(1 - e^{-\theta_j}) e^{-\sum_{k=j+1}^{i}\theta_k} \cdot f_j^{(t)} \tag{4}$$

2.2. Optimizing the reservoir's responsiveness to input

An important role of the randomly alternating mask M is to prevent network dynamics from saturating to a stable fixed point where the reservoir becomes input insensitive. However, the random choice of the mask values and the equal spacing of virtual nodes do not guarantee an optimal choice of masking in terms of non-saturating activity. The simplest of such cases is a sequence of few equal valued mask bits that can still lead to saturation. An optimal choice of the spacing between virtual nodes can then be built as to maximize the responsiveness of these nodes. A suitable proxy of responsiveness is the slope of reservoir activity at the readout points, i.e. the end points of the θ_i intervals. The bigger the slope, the further away is the activity from saturation. For that reason, we consider the slope a measure of the sensitivity of the node and the objective would be to maximize the overall sensitivity of the reservoir.

From (2) and (3), we can define the sensitivity of a node i as a function of θ_i:

$$S_i^{(t)} = \dot{x}_i^{(t)} = (-x_{i-1}^{(t)} + f_i^{(t)}) e^{-\theta_i} \tag{5}$$

Similar to the iterative procedure that led to (4), we can show that the sensitivity of a node d depends on the spacing of all the preceding nodes θ_i for $i \leq d$:

$$S_d^{(t)} = e^{-\sum_{j=i+1}^{d}\theta_j} \cdot S_i^{(t)} + \Gamma(\theta_{i+1}, \cdots, \theta_d)$$

where Γ is a term independent of θ_i. However, since the term $e^{-\sum_{j=i+1}^{d}\theta_j}$ decays exponentially the further node d is from i, one can ignore the contribution of θ_i to the sensitivity of node d for $d > i$.

From (5) we define a sensitivity vector $\mathbf{S} \in \mathbb{R}^n$. To optimize the overall sensitivity of the reservoir, we define an objective function:

$$\mathcal{O}(\Theta) = \langle \|\mathbf{S}\|_2^2 \rangle_t \quad \textit{subject to} \ \sum_{i=1}^{n} \theta_i = \tau \tag{6}$$

Being a sum of exponentials, as seen from (5), $\mathcal{O}$ is a concave function since the sum of a concave function is also concave, and a global maximum of $\mathcal{O}$ exists and can be found using gradient ascent. To find the vector Θ that maximizes (6), we follow the direction of the steepest ascent which is the gradient of $\mathcal{O}$. For this we need to compute the partial derivatives of S_i^2 with respect to θ_i:

$$\frac{\partial S_i^2}{\partial \theta_i} = -2(x_{(i-1)}^{(t)} - f_i^{(t)}) e^{-2\theta_i} \tag{7}$$

and these lead to the update rule of the vector $\Theta = (\theta_1, \cdots, \theta_n)$:

$$\Theta \leftarrow \Theta + \alpha \cdot P \cdot \mathcal{J}_{\mathcal{O}}(\Theta) \tag{8}$$

where α is a learning rate, P a projection matrix that assures that Θ satisfies the constraint in (6), and $\mathcal{J}_{\mathcal{O}}$ the Jacobian matrix of $\mathcal{O}$ with respect to Θ.

While the objective in (6) assures maximal responsiveness of the reservoir, it tends to prefer small values of θ_i many of which go to 0. This leads to reducing the reservoir's dimensionality which reflects negatively on its computational power. To avoid that effect, we introduce a constrained measure of virtual node sensitivity that scales linearly with θ_i:

$$\tilde{S}_i^{(t)} = \theta_i \cdot \dot{x}_i^{(t)} = \theta_i(-x_{i-1}^{(t)} + f_i^{(t)}) e^{-\theta_i} \tag{9}$$

which reserves the concavity of the problem and we apply the same procedure as above to maximize the objective:

$$O(\Theta) = \langle ||\tilde{\mathbf{S}}||_2^2 \rangle_t \quad \textit{subject to} \sum_{i=1}^{n} \theta_i = \tau \tag{10}$$

through computing:

$$\frac{\partial \tilde{S}_i^2}{\partial \theta_i} = \theta_i^2 \frac{\partial S_i^2}{\partial \theta_i} + 2\theta_i S_i^2 \tag{11}$$

3. Results

For all results, our model of choice for the nonlinear delay-coupled reservoir is an input driven Mackey-Glass equation [5] working, when not driven by input, at a fixed point regime:

$$\dot{x}(t) = -x(t) + \frac{\eta(x(t-\tau) + \gamma J(t))}{1 + (x(t-\tau) + \gamma J(t))} \tag{12}$$

where η and γ are model parameters, τ the delay length, and $J(t)$ is a temporally stretched input along τ and multiplexed with a binary mask, M. Simulations are carried on reservoirs with 100 virtual nodes. The task is to predict NARMA-10 time series that is a nonlinear function $y(t) = f(y(t-10), \ldots, y(t-1), u(t-10), u(t-1))$. $u(t)$ is the input to the reservoir and is drawn from a uniform distribution at the interval $[0, 0.5]$. Simulation starts with a short initial period for stabilizing the dynamics, followed by a 1000 time steps of optimization with each time step corresponding to one τ. Afterwards, readouts are trained on 3000 samples for both original and optimized relative delays, and validated on another 1000 time steps.

We compared the performance of 20 reservoirs before and after optimizing with (10). Initially, virtual nodes were equally spaced and due to the optimization procedure, a unimodal distribution of relative delays has formed with a maximum frequency around the original equidistant delay value (Fig.2A). Only few delays decayed to values close to 0 as a consequence of the modified objective function. The objective increased steadily through the optimization stage (results not shown) until reaching global maximum.

Most importantly, 18 out of the 20 trials showed improvement in performance. The median relative improvement is 8% from the average 0.22 NRMSE before optimization (Fig.2B).

Modifying relative delays also has an effect on the connectivity of the reservoir estimated from (4). Virtual nodes with shorter delays have stronger self-coupling and coupling to the successive nodes. This corresponds to higher influence on the overall dynamics (Fig.2D), upper node of inset **d3**). Longer delays, on the other hand, reflects week coupling and smaller contribution to the dynamics (Fig.2D, third node down the diagonal of inset **d2**).

To further illustrate the effect of the chosen objective function, we show the dynamics of the reservoir in relation

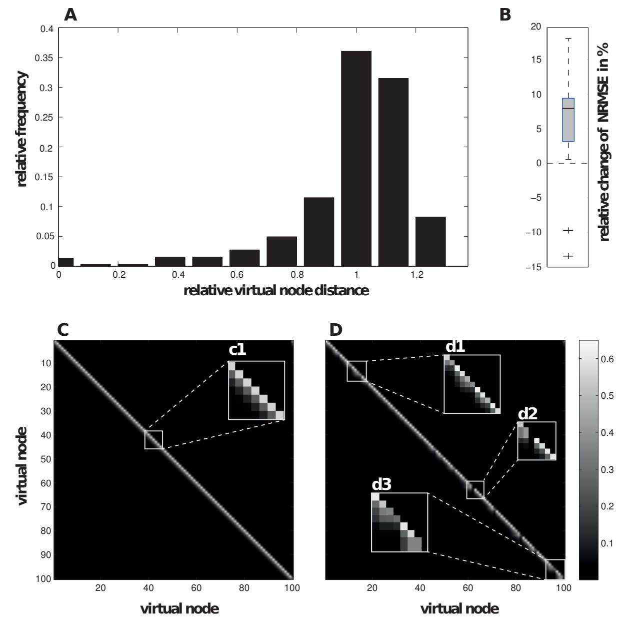

Figure 2: Optimization of the relative delays between virtual nodes. (**A**) Node distance after optimization (1 corresponds to the original equidistant node spacing). (**B**) Relative improvement in percent of the NRMSE for predicting a NARMA-10 time series as a result of optimizing the virtual node spacing (average NRMSE before optimization is 0.22). Virtual node connectivity (**C**) before and (**D**) after optimizing the delays. Since nodes are equally spaced in **C**, the relative self connectivity and connectivity to subsequent nodes is the same for all nodes. In contrast, the connectivity in **D** is optimized given the input and the mask leading to a different network motif of every virtual node.

to relative delays and the mask before and after optimization (Fig.3). Interestingly, in cases where a succession of virtual nodes had the same mask value, optimization led later nodes down the succession to reduce their relative delays, which results in less saturation of the reservoir's activity, and hence increasing sensitivity to input.

4. Discussion

We demonstrate that adapting the temporal multiplexing by optimizing the virtual node distance, and in turn the temporal profile of the mask, optimizes the performance of a single-node delay-coupled reservoir computer. We show that this improved performance is explained by the fact that the optimized choice of Θ enhances the virtual nodes' contribution to computation. The role of the optimization procedure discussed here is similar in effect to Intrinsic Plasticity (IP) known from neurobiology [6]. Both mecahnisms maximize the response to input of a nonlinear system, i.e. either a driven Mackey-Glass equation [5, 4] or a neuron in brain tissue [7, 8].

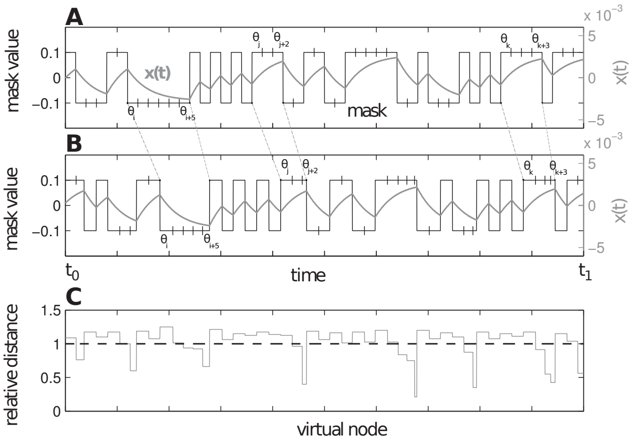

Figure 3: Optimization of the relative delays between virtual nodes. Reservoir activity superimposed on the corresponding mask (**A**) before and (**B**) after optimizing for sensitivity. (**A**) Relative delays are initially equal and network activity faces the risk of saturation, specially at parts of the mask with the same bit values for few successive virtual nodes. (**B**) Optimizing for sensitivity leads the delays of virtual nodes at the end of such mask parts to reduce in length which increases the responsiveness of the system. (**C**) Relative delays after optimization varies around the original value and adapts to the alterations of the mask.

The full potential of delay-coupled dynamical systems as computational frameworks is yet to be uncovered. We therefor plan in the future to extend the current work by expanding the objective so it incorporates further terms (e.g. decorrelating the response of virtual nodes), including multiple delay loops, and considering different and more complicated coupling topologies than what is presented here. Also, sensitivity to noise is still a limitation of the reservoir computer based on the selected Mackey-Glass node, even after maximizing its responsiveness. Adapting the system to become noise robust is a further point for future inquiry. It is also important to note, that optimization of such high number of parameters, i.e. 100 virtual node distances here, is usually not a trivial problem. Here we approached this issue by constructing a strictly concave objective function which makes the numerical optimization extremely efficient, and the solution unique.

Acknowledgments

We would like to thank the members of the PHOCUS consortium for fruitful discussions. The project PHOCUS acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 240763.

References

[1] D. V. Buonomano and W. Maass, “State-dependent computations: spatiotemporal processing in cortical networks,” *Nat. Rev. Neurosci.*, vol.10, no. 2, 113–25, 2009.

[2] H. Jaeger, “The echo state approach to analysing and training recurrent neural networks,” *Tech. report*, 2001.

[3] W. Maass, T. Natschläger, and H. Markram, “Real-time computing without stable states: a new framework for neural computation based on perturbations,” *Neural Comput.*, vol.14, no. 11, 2531–60, 2002.

[4] 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,” *Nat. Commun.*, vol.2, 468+, 2011.

[5] M. Mackey, L. Glass, “Mackey-Glass equation,” *Scholarpedia*, 4, no. 12, 6908, 2009.

[6] G. Daoudal and D. Debanne, “Long-term plasticity of intrinsic excitability : learning rules and mechanisms,” *Learn Mem.*, Nov-Dec ;10(6):456-65, 2003.

[7] A. Lazar, G. Pipa, and J. Triesch, “SORN: a self-organizing recurrent neural network,” *Front. Comput. Neurosci.*, vol.3, no. October, 23, 2009.

[8] H. Toutounji and G. Pipa, “Neuronal plasticity makes recurrent networks noise robust by generating intrinsic noise,” in prep.