LETTER Communicated by Herbert Jaeger

Analysis and Design of Echo State Networks

Mustafa C. Ozturk
can@cnel.ufl.edu
Dongming Xu
dmxu@cnel.ufl.edu
José C. Príncipe
principe@cnel.ufl.edu
Computational NeuroEngineering Laboratory, Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, U.S.A.

The design of echo state network (ESN) parameters relies on the selection of the maximum eigenvalue of the linearized system around zero (spectral radius). However, this procedure does not quantify in a systematic manner the performance of the ESN in terms of approximation error. This article presents a functional space approximation framework to better understand the operation of ESNs and proposes an information-theoretic metric, the average entropy of echo states, to assess the richness of the ESN dynamics. Furthermore, it provides an interpretation of the ESN dynamics rooted in system theory as families of coupled linearized systems whose poles move according to the input signal dynamics. With this interpretation, a design methodology for functional approximation is put forward where ESNs are designed with uniform pole distributions covering the frequency spectrum to abide by the richness metric, irrespective of the spectral radius. A single bias parameter at the ESN input, adapted with the modeling error, configures the ESN spectral radius to the input-output joint space. Function approximation examples compare the proposed design methodology versus the conventional design.

1 Introduction

Dynamic computational models require the ability to store and access the time history of their inputs and outputs. The most common dynamic neural architecture is the time-delay neural network (TDNN) that couples delay lines with a nonlinear static architecture where all the parameters (weights) are adapted with the backpropagation algorithm. The conventional delay line utilizes ideal delay operators, but delay lines with local first-order recursive filters have been proposed by Werbos (1992) and extensively studied in the gamma model (de Vries, 1991; Principe, de Vries, & de Oliviera, 1993). Chains of first-order integrators are interesting because they effectively decrease the number of delays necessary to create time embeddings

(Principe, 2001). Recurrent neural networks (RNNs) implement a different type of embedding that is largely unexplored. RNNs are perhaps the most biologically plausible of the artificial neural network (ANN) models (Anderson, Silverstein, Ritz, & Jones, 1977; Hopfield, 1984; Elman, 1990), but are not well understood theoretically (Siegelmann & Sontag, 1991; Siegelmann, 1993; Kremer, 1995). One of the main practical problems with RNNs is the difficulty to adapt the system weights. Various algorithms, such as backpropagation through time (Werbos, 1990) and real-time recurrent learning (Williams & Zipser, 1989), have been proposed to train RNNs; however, these algorithms suffer from computational complexity, resulting in slow training, complex performance surfaces, the possibility of instability, and the decay of gradients through the topology and time (Haykin, 1998). The problem of decaying gradients has been addressed with special processing elements (PEs) (Hochreiter & Schmidhuber, 1997). Alternative second-order training methods based on extended Kalman filtering (Singhal & Wu, 1989; Puskorius & Feldkamp, 1994; Feldkamp, Prokhorov, Eagen, & Yuan, 1998) and the multistreaming training approach (Feldkamp et al., 1998) provide more reliable performance and have enabled practical applications in identification and control of dynamical systems (Kechriotis, Zervas, & Monolakos, 1994; Puskorius & Feldkamp, 1994; Delgado, Kambhampati, & Warwick, 1995).

Recently, two new recurrent network topologies have been proposed: the echo state network (ESN) by Jaeger (2001, 2002a; Jaeger & Hass, 2004) and the liquid state machine (LSM) by Maass (Maass, Natschläger, & Markram, 2002). ESNs possess a highly interconnected and recurrent topology of nonlinear PEs that constitutes a "reservoir of rich dynamics" (Jaeger, 2001) and contain information about the history of input and output patterns. The outputs of these internal PEs (echo states) are fed to a memoryless but adaptive readout network (generally linear) that produces the network output. The interesting property of ESN is that only the memoryless readout is trained, whereas the recurrent topology has fixed connection weights. This reduces the complexity of RNN training to simple linear regression while preserving a recurrent topology, but obviously places important constraints in the overall architecture that have not yet been fully studied. Similar ideas have been explored independently by Maass and formalized in the LSM architecture. LSMs, although formulated quite generally, are mostly implemented as neural microcircuits of spiking neurons (Maass et al., 2002), whereas ESNs are dynamical ANN models. Both attempt to model biological information processing using similar principles. We focus on the ESN formulation in this letter.

The echo state condition is defined in terms of the spectral radius (the largest among the absolute values of the eigenvalues of a matrix, denoted by $\|\cdot\|$) of the reservoir's weight matrix ($\|\mathbf{W}\| < 1$). This condition states that the dynamics of the ESN is uniquely controlled by the input, and the effect of the initial states vanishes. The current design of ESN parameters

relies on the selection of spectral radius. However, there are many possible weight matrices with the same spectral radius, and unfortunately they do not all perform at the same level of mean square error (MSE) for functional approximation. A similar problem exists in the design of the LSM. LSMs have been shown to possess universal approximation given the separation property (SP) for the liquid (reservoir in ESNs) and the approximation property (AP) for the readout (Maass et al., 2002). SP is quantified by a kernel-quality measure proposed in Maass, Legenstein, and Bertschinger (2005) that is based on the rank of a matrix formed by the system states corresponding to different input signals. The kernel quality is a measure for the complexity and diversity of nonlinear operations carried out by the liquid on its input stream in order to boost the classification power of a subsequent linear decision hyperplane (Maass et al., 2005). A variation of SP has been proposed in Bertschinger and Natschläger (2004), and it has been argued that complex calculations can be best carried out by networks on the boundary between ordered and chaotic dynamics.

In this letter, we are interested in studying the ESN for functional approximation (filters that map input functions $u(\cdot)$ of time on output functions $y(\cdot)$ of time). We see two major shortcomings with the current ESN approach that uses echo state condition as a design principle. First, the impact of fixed reservoir parameters for function approximation means that the information about the desired response is conveyed only to the output projection. This is not optimal, and strategies to select different reservoirs for different applications have not been devised. Second, imposing a constraint only on the spectral radius is a weak condition to properly set the parameters of the reservoir, as experiments show (different randomizations with the same spectral radius perform differently for the same problem; see Figure 2).

This letter aims to address these two problems by proposing a framework, a metric, and a design principle for ESNs. The framework is a signal processing interpretation of basis and projections in functional spaces to describe and understand the ESN architecture. According to this interpretation, the ESN states implement a set of basis functionals (representation space) constructed dynamically by the input, while the readout simply projects the desired response onto this representation space. The metric to describe the richness of the ESN dynamics is an information-theoretic quantity, the average state entropy (ASE). Entropy measures the amount of information contained in a given random variable (Shannon, 1948). Here, the random variable is the instantaneous echo state from which the entropy for the overall state (vector) is estimated. The probability density function (pdf) in a differential geometric framework should be thought of as a volume form; that is, in our case, the pdf of the state vector describes the metric of the state space manifold (Amari, 1990). Moreover, Cox (1946) established information as a coordinate free metric in the state manifold. Therefore, entropy becomes a global descriptor of information that quantifies the volume of the manifold defined by the random variable. Due to the

time dependency of the states, the state entropy averaged over time (ASE) is an appropriate estimate of the volume of the state manifold.

The design principle specifies that one should consider independently the correlation among the basis and the spectral radius. In the absence of any information about the desired response, the ESN states should be designed with the highest ASE, independent of the spectral radius. We interpret the ESN dynamics as a combination of time-varying linear systems obtained from the linearization of the ESN nonlinear PE in a small, local neighborhood of the current state. The design principle means that the poles of the linearized ESN reservoir should have uniform pole distributions to generate echo states with the most diverse pole locations (which correspond to the uniformity of time constants). Effectively, this will create the least correlated bases for a given spectral radius, which corresponds to the largest volume spanned by the basis set. When the designer has no other information about the desired response to set the basis, this principle distributes the system's degrees of freedom uniformly in space. It approximates for ESNs the well-known property of orthogonal basis. The unresolved issue that ASE does not quantify is how to set the spectral radius, which depends again on the desired mapping. The concept of memory depth as explained in Principe et al. (1993) and Jaeger (2002a) is helpful in understanding the issues associated with the spectral radius. The correlation time of the desired response (as estimated by the first zero of the autocorrelation function) gives an indication of the type of spectral radius required (long correlation time requires high spectral radius). Alternatively, a simple adaptive bias is added at the ESN input to control the spectral radius integrating the information from the input-output joint space in the ESN bases. For sigmoidal PEs, the bias adjusts the operating points of the reservoir PEs, which has the net effect of adjusting the volume of the state manifold as required to approximate the desired response with a small error. This letter shows that ESNs designed with this strategy obtain systematically better results in a set of experiments when compared with the conventional ESN design.

2 Analysis of Echo State Networks

2.1 Echo States as Bases and Projections. Let us consider the architecture and recursive update equation of a typical ESN more closely. Consider the recurrent discrete-time neural network given in Figure 1 with M input units, N internal PEs, and L output units. The value of the input unit at time n is $\mathbf{u}(n) = [u_1(n), u_2(n), \ldots, u_M(n)]^T$, of internal units are $\mathbf{x}(n) = [x_1(n), x_2(n), \ldots, x_N(n)]^T$, and of output units are $\mathbf{y}(n) = [y_1(n), y_2(n), \ldots, y_L(n)]^T$. The connection weights are given in an $N \times M$ weight matrix $\mathbf{W}^{in} = (w_{ij}^{in})$ for connections between the input and the internal PEs, in an $N \times N$ matrix $\mathbf{W} = (w_{ij})$ for connections between the internal PEs, in an $L \times N$ matrix $\mathbf{W}^{out} = (w_{ij}^{out})$ for connections from PEs to the

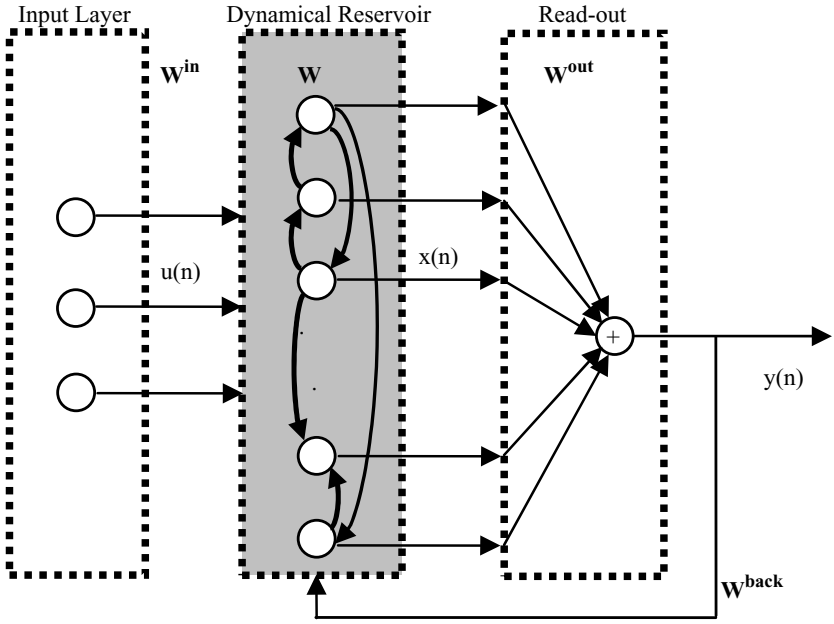


Figure 1: An echo state network (ESN). ESN is composed of two parts: a fixed-weight ($\|\mathbf{W}\| < 1$) recurrent network and a linear readout. The recurrent network is a reservoir of highly interconnected dynamical components, states of which are called echo states. The memoryless linear readout is trained to produce the output.

output units, and in an $N \times L$ matrix $\mathbf{W}^{back} = (w_{ij}^{back})$ for the connections that project back from the output to the internal PEs (Jaeger, 2001). The activation of the internal PEs (echo state) is updated according to

$$\mathbf{x}(n+1) = \mathbf{f}(\mathbf{W}^{in}\mathbf{u}(n+1) + \mathbf{W}\mathbf{x}(n) + \mathbf{W}^{back}\mathbf{y}(n)), \tag{2.1}$$

where $\mathbf{f} = (f_1, f_2, \ldots, f_N)$ are the internal PEs' activation functions. Here, all f_i's are hyperbolic tangent functions ($\frac{e^x - e^{-x}}{e^x + e^{-x}}$). The output from the readout network is computed according to

$$\mathbf{y}(n+1) = \mathbf{f}^{out}(\mathbf{W}^{out}\mathbf{x}(n+1)), \tag{2.2}$$

where $\mathbf{f}^{out} = (f_1^{out}, f_2^{out}, \ldots, f_L^{out})$ are the output unit's nonlinear functions (Jaeger, 2001, 2002a). Generally, the readout is linear so $\mathbf{f}^{out}$ is identity.

ESNs resemble the RNN architecture proposed in Puskorius and Feldkamp (1996) and also used by Sanchez (2004) in brain-machine

interfaces. The critical difference is the dimensionality of the hidden recurrent PE layer and the adaptation of the recurrent weights. We submit that the ideas of approximation theory in functional spaces (bases and projections), so useful in adaptive signal processing (Principe, 2001), should be utilized to understand the ESN architecture. Let $h(\mathbf{u}(t))$ be a real-valued function of a real-valued vector

$$\mathbf{u}(t) = [u_1(t), u_2(t), \ldots, u_M(t)]^{\mathrm{T}}.$$

In functional approximation, the goal is to estimate the behavior of $h(\mathbf{u}(t))$ as a combination of simpler functions $\varphi_i(t)$, called the basis functionals, such that its approximant, $\hat{h}(\mathbf{u}(t))$, is given by

$$\hat{h}(\mathbf{u}(t)) = \sum_{i=1}^{N} a_i \varphi_i(t).$$

Here, a_i's are the projections of $h(\mathbf{u}(t))$ onto each basis function. One of the central questions in practical functional approximation is how to choose the set of bases to approximate a given desired signal. In signal processing, the choice normally goes for a complete set of orthogonal basis, independent of the input. When the basis set is complete and can be made as large as required, fixed bases work wonders (e.g., Fourier decompositions). In neural computing, the basic idea is to derive the set of bases from the input signal through a multilayered architecture. For instance, consider a single hidden layer TDNN with N PEs and a linear output. The hidden-layer PE outputs can be considered a set of nonorthogonal basis functionals dependent on the input,

$$\varphi_i(\mathbf{u}(t)) = g\left(\sum_j b_{ij} u_j(t)\right).$$

b_{ij}'s are the input layer weights, and g is the PE nonlinearity. The approximation produced by the TDNN is then

$$\hat{h}(\mathbf{u}(t)) = \sum_{i=1}^{N} a_i \varphi_i(\mathbf{u}(t)), \tag{2.3}$$

where a_i's are the weights of the output layer. Notice that the b_{ij}'s adapt the bases and the a_i's adapt the projection in the projection space. Here the goal is to restrict the number of bases (number of hidden layer PEs) because their number is coupled with the number of parameters to adapt, which has an impact on generalization and training set size, for example. Usually,

since all of the parameters of the network are adapted, the best basis in the joint (input and desired signals) space as well as the best projection can be achieved and represents the optimal solution. The output of the TDNN is a linear combination of its internal representations, but to achieve a basis set (even if nonorthogonal), linear independence among the $\varphi_i(\mathbf{u}(t))$'s must be enforced. Ito, Shah and Pon, and others have shown that this is indeed the case (Ito, 1996; Shah & Poon, 1999), but a thorough discussion is outside the scope of this article.

The ESN (and the RNN) architecture can also be studied in this framework. The states of equation 2.1 correspond to the basis set, which are recursively computed from the input, output, and previous states through $\mathbf{W}^{in}$, $\mathbf{W}$, and $\mathbf{W}^{back}$. Notice, however, that none of these weight matrices is adapted, that is, the functional bases in the ESN are uniquely defined by the input and the initial selection of weights. In a sense, ESNs are trading the adaptive connections in the RNN hidden layer by a brute force approach of creating fixed diversified dynamics in the hidden layer.

For an ESN with a linear readout network, the output equation ($\mathbf{y}(n+1) = \mathbf{W}^{out}\mathbf{x}(n+1)$) has the same form of equation 2.3, where the φ_i's and a_i's are replaced by the echo states and the readout weights, respectively. The readout weights are adapted in the training data, which means that the ESN is able to find the optimal projection in the projection space, just like the RNN or the TDNN.

A similar perspective of basis and projections for information processing in biological networks has been proposed by Pouget and Sejnowski (1997). They explored the possibility that the response of neurons in parietal cortex serves as basis functions for the transformations from the sensory input to the motor responses. They proposed that "the role of spatial representations is to code the sensory inputs and posture signals in a format that simplifies subsequent computation, particularly in the generation of motor commands".

The central issue in ESN design is exactly the nonadaptive nature of the basis set. Parameter sets in the reservoir that provide linearly independent states and possess a given spectral radius may define drastically different projection spaces because the correlation among the bases is not constrained. A simple experiment was designed to demonstrate that the selection of the ESN parameters by constraining the spectral radius is not the most suitable for function approximation. Consider a 100-unit ESN where the input signal is $\sin(2\pi n/10\pi)$. Mimicking Jaeger (2001), the goal is to let the ESN generate the seventh power of the input signal. Different realizations of a randomly connected 100-unit ESN were constructed where the entries of $\mathbf{W}$ are set to 0.4, -0.4, and 0 with probabilities of 0.025, 0.025, and 0.95, respectively. This corresponds to a spectral radius of 0.88. Input weights are set to $+1$ or, -1 with equal probabilities, and $\mathbf{W}^{back}$ is set to zero. Input is applied for 300 time steps, and the echo states are calculated using equation 2.1. The next step is to train the linear readout. One method

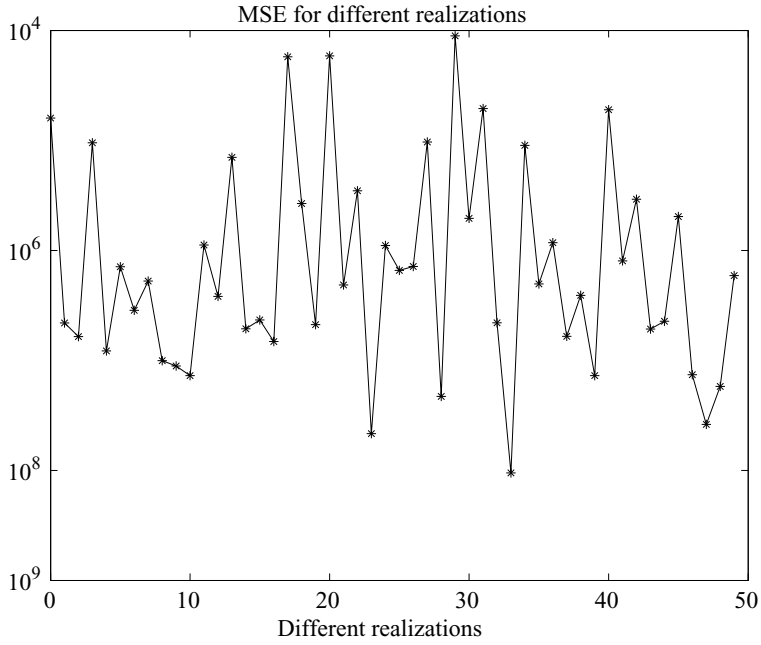


Figure 2: Performances of ESNs for different realizations of $\mathbf{W}$ with the same weight distribution. The weight values are set to 0.4, -0.4, and 0 with probabilities of 0.025, 0.025, and 0.95. All realizations have the same spectral radius of 0.88. In the 50 realizations, MSEs vary from 5.9×10^{-9} to 8.9×10^{-5}. Results show that for each set of random weights that provide the same spectral radius, the correlation or degree of redundancy among the bases will change, and different performances are encountered in practice.

to determine the optimal output weight matrix, $\mathbf{W}^{out}$, in the mean square error (MSE) sense (where MSE is defined by $O = \frac{1}{2}(\mathbf{d} - \mathbf{y})^{\mathrm{T}}(\mathbf{d} - \mathbf{y})$) is to use the Wiener solution given by Haykin (2001):

$$\mathbf{W}^{out} = E[\mathbf{x}\mathbf{x}^T]^{-1} E[\mathbf{x}\mathbf{d}] \cong \left(\frac{1}{N} \sum_n \mathbf{x}(n)\mathbf{x}(n)^T \right)^{-1} \left(\frac{1}{N} \sum_n \mathbf{x}(n)\mathbf{d}(n) \right). \quad (2.4)$$

Here, $E[.]$ denotes the expected value operator, and $\mathbf{d}$ denotes the desired signal. Figure 2 depicts the MSE values for 50 different realizations of the ESNs. As observed, even though each ESN has the same sparseness and spectral radius, the MSE values obtained vary greatly among different realizations. The minimum MSE value obtained among the 50 realizations is $5.9x10^{-9}$, whereas the maximum MSE is $8.9x10^{-5}$. This experiment

demonstrates that a design strategy that is based solely on the spectral radius is not sufficient to specify the system architecture for function approximation. This shows that for each set of random weights that provide the same spectral radius, the correlation or degree of redundancy among the bases will change, and different performances are encountered in practice.

2.2 ESN Dynamics as a Combination of Linear Systems. It is well known that the dynamics of a nonlinear system can be approximated by that of a linear system in a small neighborhood of an equilibrium point (Kuznetsov, Kuznetsov, & Marsden, 1998). Here, we perform the analysis with hyperbolic tangent nonlinearities and approximate the ESN dynamics by the dynamics of the linearized system in the neighborhood of the current system state. Hence, when the system operating point varies over time, the linear system approximating the ESN dynamics changes. We are particularly interested in the movement of the poles of the linearized ESN. Consider the update equation for the ESN without output feedback given by

$$\mathbf{x}(n+1) = \mathbf{f}(\mathbf{W}^{in}\mathbf{u}(n+1) + \mathbf{W}\mathbf{x}(n)).$$

Linearizing the system around the current state $\mathbf{x}(n)$, one obtains the Jacobian matrix, $\mathbf{J}(n+1)$, defined by

$$\begin{aligned}\mathbf{J}(n+1) &= \begin{bmatrix} \dot{f}(net_1(n))w_{11} & \dot{f}(net_1(n))w_{12} & \cdots & \dot{f}(net_1(n))w_{1N} \\ \dot{f}(net_2(n))w_{21} & \dot{f}(net_2(n))w_{22} & \cdots & \dot{f}(net_2(n))w_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ \dot{f}(net_N(n))w_{N1} & \dot{f}(net_N(n))w_{N2} & \cdots & \dot{f}(net_N(n))w_{NN} \end{bmatrix} \\ &= \begin{bmatrix} \dot{f}(net_1(n)) & 0 & \cdots & 0 \\ 0 & \dot{f}(net_2(n)) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \dot{f}(net_N(n)) \end{bmatrix} \cdot \mathbf{W} = \mathbf{F}(n) \cdot \mathbf{W}. \end{aligned} \tag{2.5}$$

Here, $net_i(n)$ is the ith entry of the vector $(\mathbf{W}^{in}\mathbf{u}(n+1) + \mathbf{W}\mathbf{x}(n))$, and w_{ij} denotes the (i, j)th entry of $\mathbf{W}$. The poles of the linearized system at time $n+1$ are given by the eigenvalues of the Jacobian matrix $\mathbf{J}(n+1)$.[1] As the amplitude of each PE changes, the local slope changes, and so the poles of

[1]The transfer function of a linear system $\mathbf{x}(n+1) = \mathbf{A}\mathbf{x}(n) + \mathbf{B}\mathbf{u}(n)$ is $\frac{\mathbf{X}(z)}{\mathbf{U}(z)} = (z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} = \frac{Adjoint(z\mathbf{I}-\mathbf{A})}{det(z\mathbf{I}-\mathbf{A})}\mathbf{B}$. The poles of the transfer function can be obtained by solving $det(z\mathbf{I} - \mathbf{A}) = 0$. The solution corresponds to the eigenvalues of $\mathbf{A}$.

the linearized system are time varying, although the parameters of ESN are fixed.

In order to visualize the movement of the poles, consider an ESN with 100 states. The entries of the internal weight matrix are chosen to be 0, 0.4 and -0.4 with probabilities 0.9, 0.05, and 0.05. $\mathbf{W}$ is scaled such that a spectral radius of 0.95 is obtained. Input weights are set to $+1$ or -1 with equal probabilities. A sinusoidal signal with a period of 100 is fed to the system, and the echo states are computed according to equation 2.1. Then the Jacobian matrix and the eigenvalues are calculated using equation 2.5. Figure 3 shows the pole tracks of the linearized ESN for different input values. A single ESN with fixed parameters implements a combination of many linear systems with varying pole locations, hence many different time constants that modulate the richness of the reservoir of dynamics as a function of input amplitude. Higher-amplitude portions of the signal tend to saturate the nonlinear function and cause the poles to shrink toward the origin of the z-plane (decreases the spectral radius), which results in a system with a large stability margin. When the input is close to zero, the poles of the linearized ESN are close to the maximal spectral radius chosen, decreasing the stability margin. When compared to their linear counterpart, an ESN with the same number of states results in a detailed coverage of the z-plane dynamics, which illustrates the power of nonlinear systems. Similar results can be obtained using signals of different shapes at the ESN input.

A key corollary of the above analysis is that the spectral radius of an ESN can be adjusted using a constant bias signal at the ESN input without changing the recurrent connection matrix, $\mathbf{W}$. The application of a nonzero constant bias will move the operating point to regions of the sigmoid function closer to saturation and always decrease the spectral radius due to the shape of the nonlinearity.[2] The relevance of bias in terms of overall system performance has also been discussed in Jaeger (2002b) and Bertschinger and Natschläger (2004), but here we approach it from a system theory perspective and explain its effect on reservoir dynamics.

3 Average State Entropy as a Measure of the Richness of ESN Reservoir

Previous research was aware of the influence of diversity of the recurrent layer outputs on the overall performance of ESNs and LSMs. Several metrics to quantify the diversity have been proposed (Jaeger, 2001; Maass, et al.,

[2] Assume $\mathbf{W}$ has nondegenerate eigenvalues and corresponding linearly independent eigenvectors. Then consider the eigendecomposition of $\mathbf{W}$, where $\mathbf{W} = \mathbf{PDP}^{-1}$, $\mathbf{P}$ is the eigenvector matrix and $\mathbf{D}$ is the diagonal matrix of eigenvalues ($\mathbf{D}_{ii}$) of $\mathbf{W}$. Since $\mathbf{F}(n)$ and $\mathbf{D}$ are diagonal, $\mathbf{J}(n+1) = \mathbf{F}(n)\mathbf{W} = \mathbf{F}(n)(\mathbf{PDP}^{-1}) = \mathbf{P}(\mathbf{F}(n)\mathbf{D})\mathbf{P}^{-1}$ is the eigendecomposition of $\mathbf{J}(n+1)$. Here, each entry of $\mathbf{F}(n)\mathbf{D}$, $f'(net(n))\mathbf{D}_{ii}$, is an eigenvalue of $\mathbf{J}$. Therefore, $|f'(net(n))\mathbf{D}_{ii}| \leq |\mathbf{D}_{ii}|$ since $f'(net_i) \leq f'(0)$.

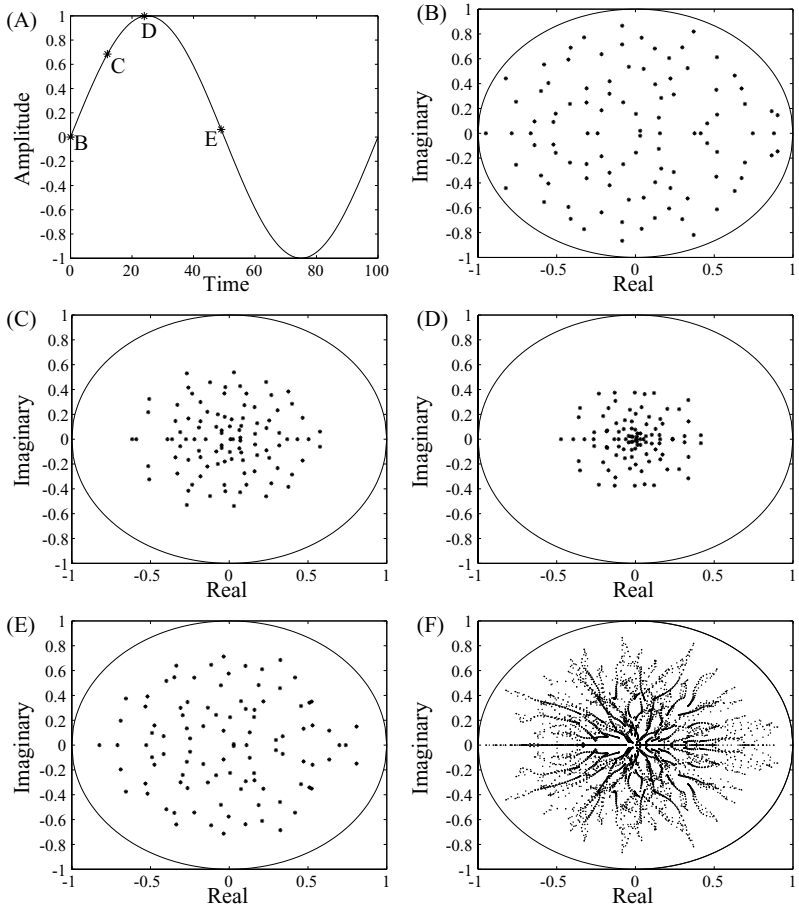


Figure 3: The pole tracks of the linearized ESN with 100 PEs when the input goes through a cycle. An ESN with fixed parameters implements a combination of linear systems with varying pole locations. (A) One cycle of sinusoidal signal with a period of 100. (B–E) The positions of poles of the linearized systems when the input values are at B, C, D, and E in Figure 5A. (F) The cumulative pole locations show the movement of the poles as the input changes. Due to the varying pole locations, different time constants modulate the richness of the reservoir of dynamics as a function of input amplitude. Higher-amplitude signals tend to saturate the nonlinear function and cause the poles to shrink toward the origin of the z-plane (decreases the spectral radius), which results in a system with a large stability margin. When the input is close to zero, the poles of the linearized ESN are close to the maximal spectral radius chosen, decreasing the stability margin. An ESN with more states results in a detailed coverage of the z-plane dynamics, which illustrates the power of nonlinear systems, when compared to their linear counterpart.

2005). Here, our approach of bases and projections leads to a new metric. We propose the instantaneous state entropy to quantify the distribution of instantaneous amplitudes across the ESN states. Entropy of the instantaneous ESN states is appropriate to quantify performance in function approximation because the ESN output is a mere weighted combination of the instantaneous value of the ESN states. If the echo state's instantaneous amplitudes are concentrated on only a few values across the ESN state dynamic range, the ability to approximate an arbitrary desired response by weighting the states is limited (and wasteful due to redundancy between the different states), and performance will suffer. On the other hand, if the ESN states provide a diversity of instantaneous amplitudes, it is much easier to achieve the desired mapping. Hence, the instantaneous entropy of the states appears as a good measure to quantify the richness of dynamics with instantaneous mappers. Due to the time structure of signals, the average state entropy (ASE), defined as the state entropy averaged over time, will be the parameter used to quantify the diversity in the dynamical reservoir of the ESN. Moreover, entropy has been proposed as an appropriate measure of the volume of the signal manifold (Cox, 1946; Amari, 1990). Here, ASE measures the volume of the echo state manifold spanned by trajectories.

Renyi's quadratic entropy is employed here because it is a global measure of information. In addition, an efficient nonparametric estimator of Renyi's entropy, which avoids explicit pdf estimation, has been developed (Principe, Xu, & Fisher, 2000). Renyi's entropy with parameter γ for a random variable X with a pdf $f_X(x)$ is given by Renyi (1970):

$$H_\gamma(X) = \frac{1}{1-\gamma} \log E[f_X^{\gamma-1}(X)].$$

Renyi's quadratic entropy is obtained for $\gamma = 2$ (for $\gamma \to 1$, Shannon's entropy is obtained). Given N samples $\{x_1, x_2, \ldots, x_N\}$ drawn from the unknown pdf to be estimated, Parzen windowing approximates the underlying pdf by

$$f_X(x) = \frac{1}{N} \sum_{i=1}^{N} K_\sigma(x - x_i),$$

where K_σ is the kernel function with the kernel size σ. Then the Renyi's quadratic entropy can be estimated by (Principe et al., 2000)

$$H_2(X) = -\log \left[\frac{1}{N^2} \sum_j \left(\sum_i K_\sigma(x_j - x_i) \right) \right]. \tag{3.1}$$

The instantaneous state entropy is estimated using equation 3.1 where the samples are the entries of the state vector $\mathbf{x}(n) = [x_1(n), x_2(n), \ldots, x_N(n)]^T$ of an ESN with N internal PEs. Results will be shown with a gaussian kernel with kernel size chosen to be 0.3 of the standard deviation of the entries of the state vector. We will show that ASE is a more sensitive parameter to quantify the approximation properties of ESNs by experimentally demonstrating that ESNs with different spectral radius and even with the same spectral radius display different ASEs.

Let us consider the same 100-unit ESN that we used in the previous section built with three different spectral radii 0.2, 0.5, 0.8 with an input signal of $\sin(2\pi n/20)$. Figure 4A depicts the echo states over 200 time ticks. The instantaneous state entropy is also calculated at each time step using equation 3.1 and plotted in Figure 4B. First, note that the instantaneous state entropy changes over time with the distribution of the echo states as we would expect, since state entropy is dependent on the input signal that also changes in this case. Second, as the spectral radius increases in the simulation, the diversity in the echo states increases. For the spectral radius of 0.2, echo state's instantaneous amplitudes are concentrated on only a few values, which is wasteful due to redundancy between different states. In practice, to quantify the overall representation ability over time, we will use ASE, which takes values -0.735, -0.007, and 0.335 for the spectral radii of 0.2, 0.5, and 0.8, respectively. Moreover, even for the same spectral radius, several ASEs are possible. Figure 4C shows ASEs from 50 different realizations of ESNs with the same spectral radius of 0.5, which means that ASE is a finer descriptor of the dynamics of the reservoir. Although we have presented an experiment with sinusoidal signal, similar results are obtained for other inputs as long as the input dynamic range is properly selected.

Maximizing ASE means that the diversity of the states over time is the largest and should provide a basis set that is as uncorrelated as possible. This condition is unfortunately not a guarantee that the ESN so designed will perform the best, because the basis set in ESNs is created independent of the desired response and the application may require a small spectral radius. However, we maintain that when the desired response is not accessible for the design of the ESN bases or when the same reservoir is to be used for a number of problems, the default strategy should be to maximize the ASE of the state vector. The following section addresses the design of ESNs with high ASE values and a simple mechanism to adjust the reservoir dynamics without changing the recurrent connection weights.

4 Designing Echo State Networks

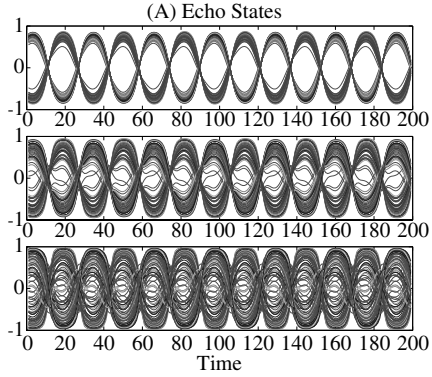
4.1 Design of the Echo State Recurrent Connections. According to the interpretation of ESNs as coupled linear systems, the design of the internal

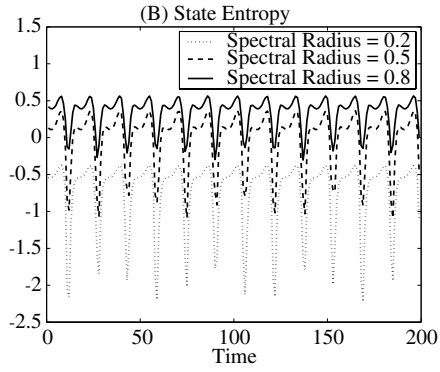
connection matrix, **W**, will be based on the distribution of the poles of the linearized system around zero state. Our proposal is to design the ESN such that the linearized system has uniform pole distribution inside the unit circle of the z-plane. With this design scenario, the system dynamics will include uniform coverage of time constants arising from the uniform distribution of the poles, which also decorrelates as much as possible the basis functionals. This principle was chosen by analogy to the identification of linear systems using Kautz filters (Kautz, 1954), which shows that the best approximation of a given transfer function by a linear system with finite order is achieved when poles are placed in the neighborhood of the spectral resonances. When no information is available about the desired response, we should uniformly spread the poles to anticipate good approximation to arbitrary mappings.

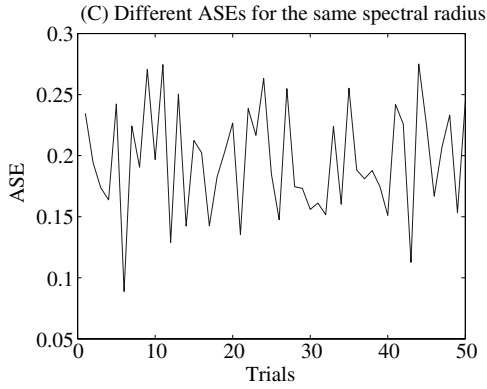
We again use a maximum entropy principle to distribute the poles inside the unit circle uniformly. The constraints of a circle as boundary conditions for discrete linear systems and complex conjugate locations are easy to include for the pole distribution (Thogula, 2003). The poles are first initialized at random locations; the quadratic Renyi's entropy is calculated by equation 3.1, and poles are moved such that the entropy of the new distribution is increased over iterations (Erdogmus & Principe, 2002). This method is efficient to find uniform coverage of the unit circle with an arbitrary number of poles. The system with the uniform pole locations can be interpreted using linear system theory. The poles that are close to the unit circle correspond to many sharp bandpass filters specializing in different frequency regions, whereas the inner poles realize filters of larger frequency support. Moreover, different orientations (angles) of the poles create filters of different center frequencies.

Now the problem is to construct an internal weight matrix from the pole locations (eigenvalues of **W**). In principle, we would like to create a sparse

Figure 4: Examples of echo states and instantaneous state entropy. (A) Outputs of echo states (100 PEs) produced by ESNs with spectral radius of 0.2, 0.5, and 0.8, from top to bottom, respectively. The diversity of echo states increases when the spectral radius increases. Within the dynamic range of the echo states, systems with smaller spectral radius can generate only uneven representations, while for $\|\mathbf{W}\| = 0.8$, outputs of echo states almost uniformly distribute within their dynamic range. (B) Instantaneous state entropy is calculated using equation 3.1. Information contained in the echo states is changing over time according to the input amplitude. Therefore, the richness of representation is controlled by the input amplitude. Moreover, the value of ASE increases with spectral radius. (C) ASEs from 50 different realizations of ESNs with the same spectral radius of 0.5. The plot shows that ASE is a finer descriptor of the dynamics of the reservoir than the spectral radius.


(A) Echo States
1
0
-1
0 20 40 60 80 100 120 140 160 180 200
Time


(B) State Entropy
Spectral Radius = 0.2
Spectral Radius = 0.5
Spectral Radius = 0.8
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0 50 100 150 200
Time


(C) Different ASEs for the same spectral radius
ASE
0.3
0.25
0.2
0.15
0.1
0.05
0 10 20 30 40 50
Trials

matrix, so we started with the sparsest matrix (with an inverse), which is the direct canonical structure given by (Kailath, 1980)

$$\mathbf{W} = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{N-1} & -a_N \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \tag{4.1}$$

The characteristic polynomial of $\mathbf{W}$ is

$$\begin{aligned} l(s) &= \det(s\mathbf{I} - \mathbf{W}) = s^N + a_1 s^{N-1} + a_2 s^{N-2} + a_N \\ &= (s - p_1)(s - p_2)\cdots(s - p_N), \end{aligned} \tag{4.2}$$

where p_i's are the eigenvalues and a_i's are the coefficients of the characteristic polynomial of $\mathbf{W}$. Here, we know the pole locations of the linear system obtained from the linearization of the ESN, so using equation 4.2, we can obtain the characteristic polynomial and construct $\mathbf{W}$ matrix in the canonical form using equation 4.1. We will call the ESN constructed based on the uniform pole principle ASE-ESN. All other possible solutions with the same eigenvalues can be obtained by $\mathbf{Q}^{-1}\mathbf{W}\mathbf{Q}$, where $\mathbf{Q}$ is any nonsingular matrix.

To corroborate our hypothesis, we would like to show that the linearized ESN designed with the recurrent weight matrix having the eigenvalues uniformly distributed inside the unit circle creates higher ASE values for a given spectral radius compared to other ESNs with random internal connection weight matrices. We will consider an ESN with 30 states and use our procedure to create the $\mathbf{W}$ matrix for ASE-ESN for different spectral radii between [0.1, 0.95]. Similarly, we constructed ESNs with sparse random $\mathbf{W}$ matrices with different sparseness constraints. This corresponds to a weight distribution having the values 0, c and $-c$ with probabilities p_1, $(1 - p_1)/2$, and $(1 - p_1)/2$, where p_1 defines the sparseness of $\mathbf{W}$ and c is a constant that takes a specific value depending on the spectral radius. We also created $\mathbf{W}$ matrices with values uniformly distributed between -1 and 1 (U-ESN) and scaled to obtain a given spectral radius (Jaeger & Hass, 2004). Then, for different $\mathbf{W}^{in}$ matrices, we run the ASE-ESNs with the sinusoidal input given in section 3 and calculate ASE. Figure 5 compares the ASE values averaged over 1000 realizations. As observed from the figure, the ASE-ESN with uniform pole distribution generates higher ASE on average for all spectral radii compared to ESNs with sparse and uniform random connections. This approach is indeed conceptually similar to Jeffreys' maximum entropy prior (Jeffreys, 1946): it will provide a consistently good response for the largest class of problems. Concentrating the poles of the linearized

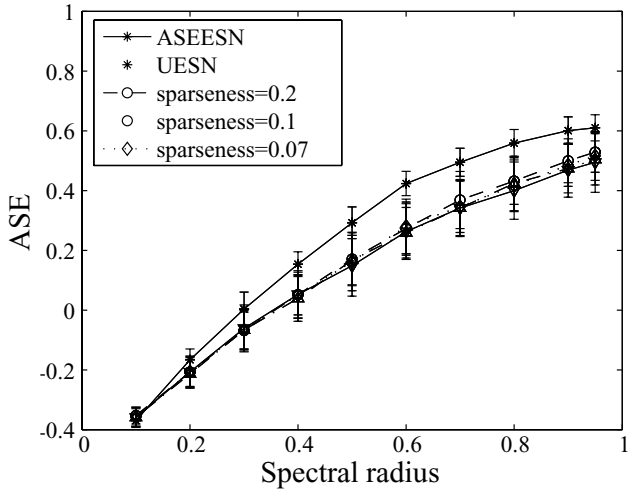


Figure 5: Comparison of ASE values obtained for ASE-ESN having **W** with uniform eigenvalue distribution, ESNs with random **W** matrix, and U-ESN with uniformly distributed weights between -1 and 1. Randomly generated weights have sparseness of 0.07, 0.1, and 0.2. ASE values are calculated for the networks with spectral radius from 0.1 to 0.95. The ASE-ESN with uniform pole distribution generates a higher ASE on average for all spectral radii compared to ESNs with random connections.

system in certain regions of the space provides good performance only if the desired response has energy in this part of the space, as is well known from the theory of Kautz filters (Kautz, 1954).

4.2 Design of the Adaptive Bias. In conventional ESNs, only the output weights are trained, optimizing the projections of the desired response onto the basis functions (echo states). Since the dynamical reservoir is fixed, the basis functions are only input dependent. However, since function approximation is a problem in the joint space of the input and desired signals, a penalty in performance will be incurred. From the linearization analysis that shows the crucial importance of the operating point of the PE nonlinearity in defining the echo state dynamics, we propose to use a single external adaptive bias to adjust the effective spectral radius of an ESN. Notice that according to linearization analysis, bias can reduce only spectral radius. The information for adaptation of bias is the MSE in training, which modulates the spectral radius of the system with the information derived from the approximation error. With this simple mechanism, some information from the input-output joint space is incorporated in the definition of the projection space of the ESN. The beauty of this method is that the spectral

radius can be adjusted by a single parameter that is external to the system without changing reservoir weights.

The training of bias can be easily accomplished. Indeed, since the parameter space is only one-dimensional, a simple line search method can be efficiently employed to optimize the bias. Among different line search algorithms, we will use a search that uses Fibonacci numbers in the selection of points to be evaluated (Wilde, 1964). The Fibonacci search method minimizes the maximum number of evaluations needed to reduce the interval of uncertainty to within the prescribed length. In our problem, a bias value is picked according to Fibonacci search. For each value of bias, training data are applied to the ESN, and the echo states are calculated. Then the corresponding optimal output weights and the objective function (MSE) are evaluated to pick the next bias value.

Alternatively, gradient-based methods can be utilized to optimize the bias, due to simplicity and low computational cost. System update equation with an external bias signal, b, is given by

$$\mathbf{x}(n+1) = \mathbf{f}(\mathbf{W}^{in}\mathbf{u}(n+1) + \mathbf{W}^{in}b + \mathbf{W}\mathbf{x}(n)).$$

The update equation for b is given by

$$\frac{\partial O(n+1)}{\partial b} = -\mathbf{e} \cdot \mathbf{W}^{out} \times \frac{\partial \mathbf{x}(n+1)}{\partial b} \tag{4.3}$$

$$= -e \cdot \mathbf{W}^{out} \times \left[\dot{\mathbf{f}}(net_{n+1}) \cdot \left(\mathbf{W} \times \frac{\partial \mathbf{x}(n)}{\partial b} + \mathbf{W}^{in} \right) \right]. \tag{4.4}$$

Here, O is the MSE defined previously. This algorithm may suffer from similar problems observed in gradient-based methods in recurrent networks training. However, we observed that the performance surface is rather simple. Moreover, since the search parameter is one-dimensional, the gradient vector can assume only one of the two directions. Hence, imprecision in the gradient estimation should affect the speed of convergence but normally not change the correct gradient direction.

5 Experiments

This section presents a variety of experiments in order to test the validity of the ESN design scheme proposed in the previous section.

5.1 Short-Term Memory Capacity. This experiment compares the short-term memory (STM) capacity of ESNs with the same spectral radius using the framework presented in Jaeger (2002a). Consider an ESN with a single input signal, $u(n)$, optimally trained with the desired signal $u(n-k)$, for a given delay k. Denoting the optimal output signal $y_k(n)$, the k-delay

STM capacity of a network, MC_k, is defined as a squared correlation coefficient between $u(n-k)$ and $y_k(n)$ (Jaeger, 2002a). The STM capacity, MC, of the network is defined as $\sum_{k=1}^{\infty} MC_k$. STM capacity measures how accurately the delayed versions of the input signal are recovered with optimally trained output units. Jaeger (2002a) has shown that the memory capacity for recalling an independent and identically distributed (i.i.d.) input by an N unit RNN with linear output units is bounded by N.

We use ESNs with 20 PEs and a single input unit. ESNs are driven by an i.i.d. random input signal, $u(n)$, that is uniformly distributed over $[-0.5, 0.5]$. The goal is to train the ESN to generate the delayed versions of the input, $u(n-1), \ldots, u(n-40)$. We used four different ESNs: R-ESN, U-ESN, ASE-ESN, and BASE-ESN. R-ESN is a randomly connected ESN used in Jaeger (2002a) where the entries of $\mathbf{W}$ matrix are set to 0, 0.47, -0.47 with probabilities 0.8, 0.1, 0.1, respectively. This corresponds to a sparse connectivity of 20% and a spectral radius of 0.9. The entries of $\mathbf{W}$ of U-ESN are uniformly distributed over $[-1, 1]$ and scaled to obtain the spectral radius of 0.9. ASE-ESN also has a spectral radius of 0.9 and is designed with uniform poles. BASE-ESN has the same recurrent weight matrix as ASE-ESN and an adaptive bias at its input. In each ESN, the input weights are set to 0.1 or -0.1 with equal probability, and direct connections from the input to the output are allowed, whereas $\mathbf{W}^{back}$ is set to $\mathbf{0}$ (Jaeger, 2002a). The echo states are calculated using equation 2.1 for 200 samples of the input signal, and the first 100 samples corresponding to initial transient are eliminated. Then the output weight matrix is calculated using equation 2.4. For the BASE-ESN, the bias is trained for each task. All networks are run with a test input signal, and the corresponding output and MC_k are calculated. Figure 6 shows the k-delay STM capacity (averaged over 100 trials) of each ESN for delays $1, \ldots, 40$ for the test signal. The STM capacities of R-ESN, U-ESN, ASE-ESN, and BASE-ESN are 13.09, 13.55, 16.70, and 16.90, respectively. First, ESNs with uniform pole distribution (ASE-ESN and BASE-ESN) have MCs that are much longer than the randomly generated ESN given in Jaeger (2002a) in spite of all having the same spectral radius. In fact, the STM capacity of ASE-ESN is close to the theoretical maximum value of $N = 20$. A closer look at the figure shows that R-ESN performs slightly better than ASE-ESN for delays less than 9. In fact, for small k, large ASE degrades the performance because the tasks do not need long memory depth. However, the drawback of high ASE for small k is recovered in BASE-ESN, which reduces the ASE to the appropriate level required for the task. Overall, the addition of the bias to the ASE-ESN increases the STM capacity from 16.70 to 16.90. On the other hand, U-ESN has slightly better STM compared to R-ESN with only three different weight values, although it has more distinct weight values compared to R-ESN. It is also significant to note that the MC will be very poor for an ESN with smaller spectral radius even with an adaptive bias, since the problem requires large ASE and bias can only reduce ASE. This experiment demonstrates the

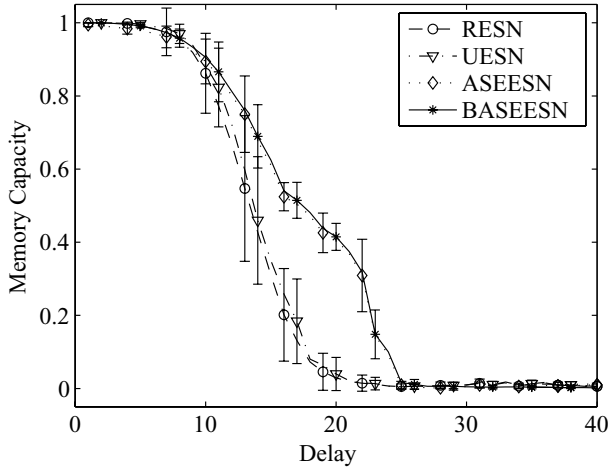


Figure 6: The k-delay STM capacity of each ESN for delays $1, \ldots, 40$ computed using the test signal. The results are averaged over 100 different realizations of each ESN type with the specifications given in the text for different $\mathbf{W}$ and $\mathbf{W}^{in}$ matrices. The STM capacities of R-ESN, U-ESN, ASE-ESN, and BASE-ESN are 13.09, 13.55, 16.70, and 16.90, respectively.

suitability of maximizing ASE in tasks that require a substantial memory length.

5.2 Binary Parity Check. The effect of the adaptive bias was marginal in the previous experiment since the nature of the problem required large ASE values. However, there are tasks in which the optimal solutions require smaller ASE values and smaller spectral radius. Those are the tasks where the adaptive bias becomes a crucial design parameter in our design methodology.

Consider an ESN with 100 internal units and a single input unit. ESN is driven by a binary input signal, $u(n)$, that assumes the values 0 or 1. The goal is to train an ESN to generate the m-bit parity corresponding to last m bits received, where m is $3, \ldots, 8$. Similar to the previous experiments, we used the R-ESN, ASE-ESN, and BASE-ESN topologies. R-ESN is a randomly connected ESN where the entries of $\mathbf{W}$ matrix are set to 0, 0.06, -0.06 with probabilities 0.8, 0.1, 0.1, respectively. This corresponds to a sparse connectivity of 20% and a spectral radius of 0.3. ASE-ESN and BASE-ESN are designed with a spectral radius of 0.9. The input weights are set to 1 or -1 with equal probability, and direct connections from the input to the output are allowed whereas $\mathbf{W}^{back}$ is set to 0. The echo states are calculated using equation 2.1 for 1000 samples of the input signal, and the first 100 samples corresponding to the initial transient are eliminated. Then the output weight

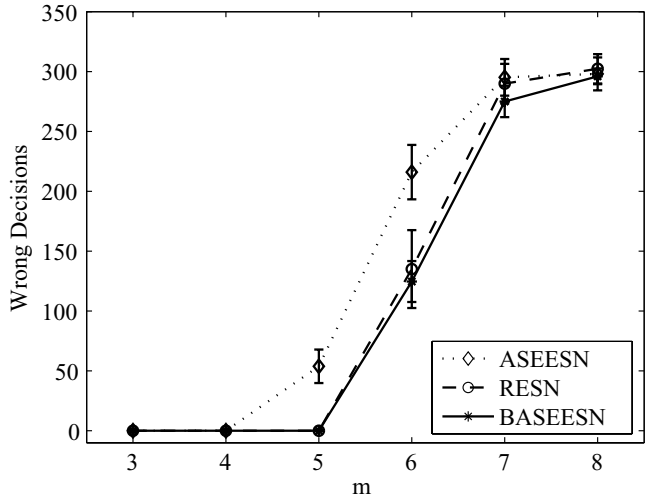


Figure 7: The number of wrong decisions made by each ESN for $m = 3, \ldots, 8$ in the binary parity check problem. The results are averaged over 100 different realizations of R-ESN, ASE-ESN, and BASE-ESN for different $\mathbf{W}$ and $\mathbf{W}^{in}$ matrices with the specifications given in the text. The total numbers of wrong decisions for $m = 3, \ldots, 8$ of R-ESN, ASE-ESN, and BASE-ESN are 722, 862, and 699.

matrix is calculated using equation 2.4. For ESN with adaptive bias, the bias is trained for each task. The binary decision is made by a threshold detector that compares the output of the ESN to 0.5. Figure 7 shows the number of wrong decisions (averaged over 100 different realizations) made by each ESN for $m = 3, \ldots, 8$.

The total numbers of wrong decisions for $m = 3, \ldots, 8$ of R-ESN, ASE-ESN, and BASE-ESN are 722, 862, and 699, respectively. ASE-ESN performs poorly since the nature of the problem requires a short time constant for fast response, but ASE-ESN has a large spectral radius. For 5-bit parity, the R-ESN has no wrong decisions, whereas ASE-ESN has 53 wrong decisions. BASE-ESN performs a lot better than ASE-ESN and slightly better than the R-ESN since the adaptive bias reduces the spectral radius effectively. Note that for $m = 7$ and 8, the ASE-ESN performs similar to the R-ESN, since the task requires access to longer input history, which compromises the need for fast response. Indeed, the bias in the BASE-ESN takes effect when there are errors ($m > 4$) and when the task benefits from smaller spectral radius. The optimal bias values are approximately 3.2, 2.8, 2.6, and 2.7 for $m = 3$, 4, 5, and 6, respectively. For $m = 7$ or 8, there is a wide range of bias values that result in similar MSE values (between 0 and 3). In

summary, this experiment clearly demonstrates the power of the bias signal to configure the ESN reservoir according to the mapping task.

5.3 System Identification. This section presents a function approximation task where the aim is to identify a nonlinear dynamical system. The unknown system is defined by the difference equation

$$y(n+1) = 0.3y(n) + 0.6y(n-1) + f(u(n)),$$

where

$$f(u) = 0.6\sin(\pi u) + 0.3\sin(3\pi u) + 0.1\sin(5\pi u).$$

The input to the system is chosen to be $\sin(2\pi n/25)$.

We used three different ESNs—R-ESN, ASE-ESN, and BASE-ESN—with 30 internal units and a single input unit. The $\mathbf{W}$ matrix of each ESN is scaled such that it has a spectral radius of 0.95. R-ESN is a randomly connected ESN where the entries of $\mathbf{W}$ matrix are set to 0, 0.35, -0.35 with probabilities 0.8, 0.1, 0.1, respectively. In each ESN, the input weights are set to 1 or -1 with equal probability, and direct connections from the input to the output are allowed, whereas $\mathbf{W}^{back}$ is set to $\mathbf{0}$. The optimal output weights are calculated using equation 2.4. The MSE values (averaged over 100 realizations) for R-ESN and ASE-ESN are $1.23x10^{-5}$ and $1.83x10^{-6}$, respectively. The addition of the adaptive bias to the ASE-ESN reduces the MSE value from $1.83x10^{-6}$ to $3.27x10^{-9}$.

6 Discussion

The great appeal of echo state networks (ESNs) and liquid state machine (LSM) is their ability to construct arbitrary mappings of signals with rich and time-varying temporal structures without requiring adaptation of the free parameters of the recurrent layer. The echo state condition allows the recurrent connections to be fixed with training limited to the linear output layer. However, the literature did not elucidate on how to properly choose the recurrent parameters for system identification applications. Here, we provide an alternate framework that interprets the echo states as a set of functional bases formed by fixed nonlinear combinations of the input. The linear readout at the output stage simply computes the projection of the desired output space onto this representation space. We further introduce an information-theoretic criterion, ASE, to better understand and evaluate the capability of a given ESN to construct such a representation layer. The average entropy of the distribution of the echo states quantifies the volume spanned by the bases. As such, this volume should be the largest to achieve the smallest correlation among the bases and be able to cope with

arbitrary mappings. However, not all function approximation problems require the same memory depth, which is coupled to the spectral radius. The effective spectral radius of an ESN can be optimized for the given problem with the help of an external bias signal that is adapted using the joint input-output space information. The interesting property of this method when applied to ESN built from sigmoidal nonlinearities is that it allows the fine tuning of the system dynamics for a given problem with a single external adaptive bias input and without changing internal system parameters. In our opinion, the combination of the largest possible ASE and the adaptation of the spectral radius by the bias produces the most parsimonious pole location of the linearized ESN when no knowledge about the mapping is available to optimally locate the bass functionals. Moreover, the bias can be easily trained with either a line search method or a gradient-based method since it is one-dimensional. We have illustrated experimentally that the design of the ESN using the maximization of ASE with the adaptation of the spectral radius by the bias has provided consistently better performance across tasks that require different memory depths. This means that these two parameters' design methodology is preferred to the spectral radius criterion proposed by Jaeger, and it is still easily incorporated in the ESN design.

Experiments demonstrate that the ASE for ESN with uniform linearized poles is maximized when the spectral radius of the recurrent weight matrix approaches one (instability). It is interesting to relate this observation with the computational properties found in dynamical systems "at the edge of chaos" (Packard, 1988; Langton, 1990; Mitchell, Hraber, & Crutchfield, 1993; Bertschinger & Natschläger, 2004). Langton stated that when cellular automata rules are evolved to perform a complex computation, evolution will tend to select rules with "critical" parameter values, which correlate with a phase transition between ordered and chaotic regimes. Recently, similar conclusions were suggested for LSMs (Bertschinger & Natschläger, 2004). Langton's interpretation of edge of chaos was questioned by Mitchell et al. (1993). Here, we provide a system-theoretic view and explain the computational behavior with the diversity of dynamics achieved with linearizations that have poles close to the unit circle. According to our results, the spectral radius of the optimal ESN in function approximation is problem dependent, and in general it is impossible to forecast the computational performance as the system approaches instability (the spectral radius of the recurrent weight matrix approaches one). However, allowing the system to modulate the spectral radius by either the output or internal biasing may allow a system close to instability to solve various problems requiring different spectral radii.

Our emphasis here is mostly on ESNs without output feedback connections. However, the proposed design methodology can also be applied to ESNs with output feedback. Both feedforward and feedback connections contribute to specify the bases to create the projection space. At the same

time, there are applications where the output feedback contributes to the system dynamics in a different fashion. For example, it has been shown that a fixed weight (fully trained) RNN with output feedback can implement a family of functions (meta-learners) (Prokhorov, Feldkamp, & Tyukin, 1992). In meta-learning, the role of output feedback in the network is to bias the system to different regions of dynamics, providing multiple input-output mappings required (Santiago & Lendaris, 2004). However, results could not be replicated with ESNs (Prokhorov, 2005). We believe that more work has to be done on output feedback in the context of ESNs but also suspect that the echo state condition may be a restriction on the system dynamics for this type of problem.

There are many interesting issues to be researched in this exciting new area. Besides an evaluation tool, ASE may also be utilized to train the ESN's representation layer in an unsupervised fashion. In fact, we can easily adapt with the SIG (stochastic information gradient) described in Erdogmus, Hild, and Principe (2003): extra weights linking the outputs of recurrent states to maximize output entropy. Output entropy maximization is a well-known metric to create independent components (Bell & Sejnowski, 1995), and here it means that the echo states will become as independent as possible. This would circumvent the linearization of the dynamical system to set the recurrent weights and would fine-tune continuously in an unsupervised manner the parameters of the ESN among different inputs. However, it goes against the idea of a fixed ESN reservoir.

The reservoir of recurrent PEs can be thought of as a new form of a time-to-space mapping. Unlike the delay line that forms an embedding (Takens, 1981), this mapping may have the advantage of filtering noise and produce representations with better SNRs to the peaks of the input, which is very appealing for signal processing and seems to be used in biology. However, further theoretical work is necessary in order to understand the embedding capabilities of ESNs. One of the disadvantages of the ESN correlated basis is in the design of the readout. Gradient-based algorithms will be very slow to converge (due to the large eigenvalue spread of modes), and even if recursive methods are used, their stability may be compromised by the condition number of the matrix. However, our recent results incorporating an L_1 norm penalty in the LMS (Rao et al., 2005) show great promise of solving this problem.

Finally we would like to briefly comment on the implications of these models to neurobiology and computational neuroscience. The work by Pouget and Sejnowski (1997) has shown that the available physiological data are consistent with the hypothesis that the response of a single neuron in the parietal cortex serves as a basis function generated by the sensory input in a nonlinear fashion. In other words, the neurons transform the sensory input into a format (representation space) such that the subsequent computation is simplified. Then, whenever a motor command (output of the biological system) needs to be generated, this simple computation to

read out the neuronal activity is done. There is an intriguing similarity between the interpretation of the neuronal activity by Pouget and Sejnowski and our interpretation of echo states in ESN. We believe that similar ideas can be applied to improve the design of microcircuit implementations of LSMs. First, the framework of functional space interpretation (bases and projections) is also applicable to microcircuits. Second, the ASE measure may be directly utilized for LSM states because the states are normally low-pass-filtered before the readout. However, the control of ASE by changing the liquid dynamics is unclear. Perhaps global control of thresholds or bias current will be able to accomplish bias control as in ESN with sigmoid PEs.

Acknowledgments

This work was partially supported by NSF ECS-0422718, NSF CNS-0540304, and ONR N00014-1-1-0405.

References

Amari, S.-I. (1990). *Differential-geometrical methods in statistics*. New York: Springer.

Anderson, J., Silverstein, J., Ritz, S., & Jones, R. (1977). Distinctive features, categorical perception, and probability learning: Some applications of a neural model. *Psychological Review, 84*, 413–451.

Bell, A. J., & Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural Computation, 7*(6), 1129–1159.

Bertschinger, N., & Natschläger, T. (2004). Real-time computation at the edge of chaos in recurrent neural networks. *Neural Computation, 16*(7), 1413–1436.

Cox, R. T. (1946). Probability, frequency, and reasonable expectation. *American Journal of Physics, 14*(1), 1–13.

de Vries, B. (1991). *Temporal processing with neural networks—the development of the gamma model*. Unpublished doctoral dissertation, University of Florida.

Delgado, A., Kambhampati, C., & Warwick, K. (1995). Dynamic recurrent neural network for system identification and control. *IEEE Proceedings of Control Theory and Applications, 142*(4), 307–314.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science, 14*(2), 179–211.

Erdogmus, D., Hild, K. E., & Principe, J. (2003). Online entropy manipulation: Stochastic information gradient. *Signal Processing Letters, 10*(8), 242–245.

Erdogmus, D., & Principe, J. (2002). Generalized information potential criterion for adaptive system training. *IEEE Transactions on Neural Networks, 13*(5), 1035–1044.

Feldkamp, L. A., Prokhorov, D. V., Eagen, C., & Yuan, F. (1998). Enhanced multistream Kalman filter training for recurrent networks. In J. Suykens, & J. Vandewalle (Eds.), *Nonlinear modeling: Advanced black-box techniques* (pp. 29–53). Dordrecht, Netherlands: Kluwer.

Haykin, S. (1998). *Neural networks: A comprehensive foundation* (2nd ed.). Upper Saddle River, NJ. Prentice Hall.

Haykin, S. (2001). *Adaptive filter theory* (4th ed.). Upper Saddle River, NJ: Prentice Hall.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Hopfield, J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences, 81*, 3088–3092.

Ito, Y. (1996). Nonlinearity creates linear independence. *Advances in Computer Mathematics, 5*(1), 189–203.

Jaeger, H. (2001). *The echo state approach to analyzing and training recurrent neural networks* (Tech. Rep. No. 148). Bremen: German National Research Center for Information Technology.

Jaeger, H. (2002a). *Short term memory in echo state networks* (Tech. Rep. No. 152). Bremen: German National Research Center for Information Technology.

Jaeger, H. (2002b). *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach* (Tech. Rep. No. 159). Bremen: German National Research Center for Information Technology.

Jaeger, H., & Hass, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science, 304*(5667), 78–80.

Jeffreys, H. (1946). An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London, A 196*, 453–461.

Kailath, T. (1980). *Linear systems*. Upper Saddle River, NJ: Prentice Hall.

Kautz, W. (1954). Transient synthesis in time domain. *IRE Transactions on Circuit Theory, 1*(3), 29–39.

Kechriotis, G., Zervas, E., & Manolakos, E. S. (1994). Using recurrent neural networks for adaptive communication channel equalization. *IEEE Transactions on Neural Networks, 5*(2), 267–278.

Kremer, S. C. (1995). On the computational power of Elman-style recurrent networks. *IEEE Transactions on Neural Networks, 6*(5), 1000–1004.

Kuznetsov, Y., Kuznetsov, L., & Marsden, J. (1998). *Elements of applied bifurcation theory* (2nd ed.). New York: Springer-Verlag.

Langton, C. G. (1990). Computation at the edge of chaos. *Physica D, 42*, 12–37.

Maass, W., Legenstein, R. A., & Bertschinger, N. (2005). Methods for estimating the computational power and generalization capability of neural microcircuits. In L. K. Saul, Y. Weiss, L. Bottou (Eds.), *Advances in neural information processing systems*, no. 17 (pp. 865–872). Cambridge, MA: MIT Press.

Maass, W., Natschläger, T., & Markram, H. (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation, 14*(11), 2531–2560.

Mitchell, M., Hraber, P., & Crutchfield, J. (1993). Revisiting the edge of chaos: Evolving cellular automata to perform computations. *Complex Systems, 7*, 89–130.

Packard, N. (1988). Adaptation towards the edge of chaos. In J. A. S. Kelso, A. J. Mandell, & M. F. Shlesinger (Eds.), *Dynamic patterns in complex systems* (pp. 293–301). Singapore: World Scientific.

Pouget, A., & Sejnowski, T. J. (1997). Spatial transformations in the parietal cortex using basis functions. *Journal of Cognitive Neuroscience, 9*(2), 222–237.

Principe, J. (2001). Dynamic neural networks and optimal signal processing. In Y. Hu & J. Hwang (Eds.), *Neural networks for signal processing* (Vol. 6-1, pp. 6–28). Boca Raton, FL: CRC Press.

Principe, J. C., de Vries, B., & de Oliviera, P. G. (1993). The gamma filter—a new class of adaptive IIR filters with restricted feedback. *IEEE Transactions on Signal Processing, 41*(2), 649–656.

Principe, J., Xu, D., & Fisher, J. (2000). Information theoretic learning. In S. Haykin (Ed.), *Unsupervised adaptive filtering* (pp. 265–319). Hoboken, NJ: Wiley.

Prokhorov, D. (2005). Echo state networks: Appeal and challenges. In *Proc. of International Joint Conference on Neural Networks* (pp. 1463–1466). Montreal, Canada.

Prokhorov, D., Feldkamp, L., & Tyukin, I. (1992). Adaptive behavior with fixed weights in recurrent neural networks: An overview. In *Proc. of International Joint Conference on Neural Networks* (pp. 2018–2022). Honolulu, Hawaii.

Puskorius, G. V., & Feldkamp, L. A. (1994). Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks. *IEEE Transactions on Neural Networks, 5*(2), 279–297.

Puskorius, G. V., & Feldkamp, L. A. (1996). Dynamic neural network methods applied to on-vehicle idle speed control. *Proceedings of IEEE, 84*(10), 1407–1420.

Rao, Y., Kim, S., Sanchez, J., Erdogmus, D., Principe, J. C., Carmena, J., Lebedev, M., & Nicolelis, M. (2005). Learning mappings in brain machine interfaces with echo state networks. In *2005 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Philadelphia.

Renyi, A. (1970). *Probability theory*. New York: Elsevier.

Sanchez, J. C. (2004). *From cortical neural spike trains to behavior: Modeling and analysis*. Unpublished doctoral dissertation, University of Florida.

Santiago, R. A., & Lendaris, G. G. (2004). Context discerning multifunction networks: Reformulating fixed weight neural networks. In *Proc. of International Joint Conference on Neural Networks* (pp. 189–194). Budapest, Hungary.

Shah, J. V., & Poon, C.-S. (1999). Linear independence of internal representations in multilayer perceptrons. *IEEE Transactions on Neural Networks, 10*(1), 10–18.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal, 27*, 623–656.

Siegelmann, H. T. (1993). *Foundations of recurrent neural networks*. Unpublished doctoral dissertation, Rutgers University.

Siegelmann, H. T., & Sontag, E. (1991). Turing computability with neural nets. *Applied Mathematics Letters, 4*(6), 77–80.

Singhal, S., & Wu, L. (1989). Training multilayer perceptrons with the extended Kalman algorithm. In D. S. Touretzky (Ed.), *Advances in neural information processing systems, 1* (pp. 133–140). San Mateo, CA: Morgan Kaufmann.

Takens, F. (1981). Detecting strange attractors in turbulence. In D. A. Rand & L.-S. Young (Eds.), *Dynamical systems and turbulence* (pp. 366–381). Berlin: Springer.

Thogula, R. (2003). *Information theoretic self-organization of multiple agents.* Unpublished master's thesis, University of Florida.

Werbos, P. (1990). Backpropagation through time: What it does and how to do it. *Proceedings of IEEE, 78*(10), 1550–1560.

Werbos, P. (1992). Neurocontrol and supervised learning: An overview and evaluation. In D. White & D. Sofge (Eds.), *Handbook of intelligent control* (pp. 65–89). New York: Van Nostrand Reinhold.

Wilde, D. J. (1964). *Optimum seeking methods*. Upper Saddle River, NJ: Prentice Hall.

Williams, R. J., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation, 1*, 270–280.

Received December 28, 2004; accepted June 1, 2006.