Controlling Slow Observables of Recurrent Neural Networks

May 20, 2012

JACOBS UNIVERSITY

Dev Bahadur Poudel
MS Smart Systems
Jacobs University Bremen
School of Engineering and Science
Campus Ring 1, D-28759 Bremen, Germany
d.poudel@jacobs-university.de

Supervised By:
Prof. Dr. Herbert Jaeger
Professor, Computational Science
Jacobs University Bremen
School of Engineering and Science
Campus Ring 1, D-28759 Bremen, Germany
h.jaeger@jacobs-university.de
http: // www. faculty. jacobs-university. de/ hjaeger/

Contents

List of Figures

Abstract

This paper elucidates the technique to remove the slow distortion from the stimulated reservoir of Recurrent Neural Network(RNN). When input is given to RNN, the neurons are invigorated by the input data. If the input data is distorted by slow noise, then the reservoir are also induced by noise which will adversely affect the output of the reservoir. Hence, it is essential to eliminate the disturbances of the reservoir.

The basic idea is to build a controller that is capable to suppress the slow varying noise from the reservoir dynamics of Echo State Network(ESN),which is RNN where hidden layers are sparsely connected. The network is driven with distorted signal in unsupervised way and smoothed observables are obtained from reservoir. The Principal Components of the smoothed observables are used to train and design the controller such that they can dampen the high amplitude of PCs. The controller is turned on during payload task to suppress the slow noise in the reservoir dynamics. As a result, during the learning of the output weights with the controller, the induced noise in reservoir dynamics is removed by the controller.This technique inclines the ESN to a homeostatic system. In this experiment, firstly the network is driven by undistorted input and the performance is measured. Subsequently, our objective is to drive the same ESN with distorted signal without the controller. Without controller, the signal in reservoir dynamics is also found to be distorted. However, while learning in controller mode, the reservoir's status and performance have similar qualities as in when network is fed without distortion.

1 Introduction

Recurrent Neural Network[2] oriented systems require training data in large amount. In some cases, the raw training data may exhibit distorting variation on longtime scale, where the variation is a disturbance for processing task. This means distortion changes on a slower timescales than natural timescales of original data. If such data carrying slow distortion, are used for signal processing and temporal pattern recognition task then system becomes skeptical in nature. It is because the system learns the slow distortions along with the important data. The reflected output after training of such systems that learns the noise is not considered to be accurate. Apparently, for the system to be accurate it should learn only important signals and should have no effect of such distortions. For example the handwriting recognition system should be able to recognize the bold, slanting and large alphabets too. In this example the distortion changes on a slower timescale than the natural timescales of processing task. Similarly a humanoid robot walking with controller should be robust to walk on different ground slopes. Here the ground slopes changes on longer scales than step length.[4]

The heart of this project architecture is Echo State Network which is special kind of Recurrent Neural Network. ESNs have been developed from a mathematical and engineering perspective and it exhibit typical features of biological recurrent neural network.[5] RNN creates an internal state of the network which exhibit dynamic temporal behavior. Recurrent Neural Network can use their internal memory to process arbitrary sequences of inputs. Echo state network is a recurrent neural network with sparsely connected hidden layer. The connectivity and weights of hidden neurons are randomly assigned and are fixed i.e. they are not change during the training. The weights of the output neurons can be learned so that the network can reproduce specific patterns.In this experiment RNN is driven with the distorted input data with slow distortions. The weights of the reservoirs are not modified during the training. Only the output weights are modified during the payload task.[3]

This project illustrates the concept of how a feedback controller can be implemented to eliminate the induced noises in the reservoir dynamics. The important signal of our interest is modulated with very slow noise. The distorted signal drives the echo state network. The pay load task of the network is to predict the output signal by one time step ahead. Our objective is to implement a controller that tries to reduce the induced slow noise inside the reservoir of the echo state network. When the reservoir gets stimulated with the input signal, exhibiting slow distortions then the slow noise is also induced in the reservoir. The smoothed internal states i.e. observables are tracked from the pool of data inside the excited reservoirs. Then the few first principal components from this observables are computed. We select first four principal components. These four principal components have vital role to influence slow noises in the reservoir. Now we are destined to reduce the amplitude of these principal components to stabilize the reservoir. Using these PCs a controller is trained so that it can track slow observables and attempt to cancel them. When the network is used for payload task the controller is installed. The controller is intended to suppress the slow noises vigorously inside the reservoir. The controller has intrinsic relation with the noise to suppress the amplitudes of PCs. In principle RNN can learn to immunized itself against slow variations in input characteristics. The concept of self-regulation leads to a homeostatic self-stabilization of reservoir dynamics. [4] The concept of homeostatic system is derived from the biological homeostasis which means maintenance of physical equilibrium within an animal by a tendency to compensate for disrupting changes in the environment. We can summarize the whole procedure through the block diagram as shown in figure 1

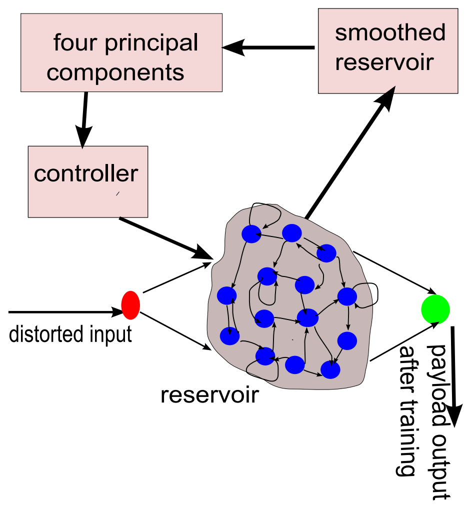


Figure 1: Block diagram

2 Methods

2.1 Echo State Network Setup

In order to create and train ESN, I have used "very simple toolbox for echo state network" in MAT LAB which was developed by Dr. Herbert Jaeger, Professor of Computational Science. ESN has N reservoirs units interconnected with each others. ESN can have many input units and output units.The adhoc settings for the network are explained below.

In this experiment the number of input neuron is one and number of output neuron is also one. The time stamp signal $u(n)$ is given as input from input neuron to stimulate the reservoirs of ESN. The number of neurons in this model is 50. The setting of number of neurons depend upon the hand of the designer. The number of neurons on other hand determine the strength and capacity of the network. The current state of neurons depend upon the previous input and previous state of the neurons. It can be expressed mathematically as

$$\mathbf{x}(n+1) = tanh\left(\mathbf{W}\mathbf{x}(n) + \mathbf{W}^{in}u(n)\right) \tag{1}$$

where $\mathbf{W}$ is the random reservoir weight matrix of size $(N*N)$ i.e 50*50 in my case, $\mathbf{W}^{in}$is the input weight matrix of size $1*N$.

The output of the network depends on the reservoir states. The output is computed as shown in equation 2.

$$y(n) = \mathbf{W}^{out}\mathbf{x}(n) \tag{2}$$

where $\mathbf{W}^{out}$ is the output weight matrix of size $N*1$.

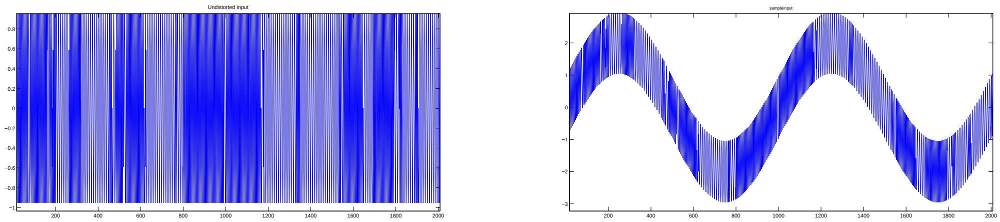
The other parameters of echo state network in this experiment i.e. spectral radius is of 0.5, connectivity is 30% and number of neurons is fixed to 50 as mentioned already. The payload task assigned to the network is one step ahead prediction of the input signal.[4]

2.2 Generating Signal

In this project I have used stochastic signal as the input to the network. The stochastic signal switch between two sine wave signals of different frequencies. The first sine wave has frequency of one and another have frequency of two. The switching of one sine wave signal to another sine wave is made randomly with a probability of 0.02. As the payload task is one step time ahead prediction the teacher signal is generated just by removing the first one time step signal from input signal.

$teacher = input(:,2:Length)$

where $teacher$ and $input$ are the matrix variables. The slow distortion is a sine wave of 1000 time steps with amplitude of 12. The original input signal was modulated with this slow distortion. The undistorted and distorted signal are as shown in Figure 2.



(a) Undistorted Input

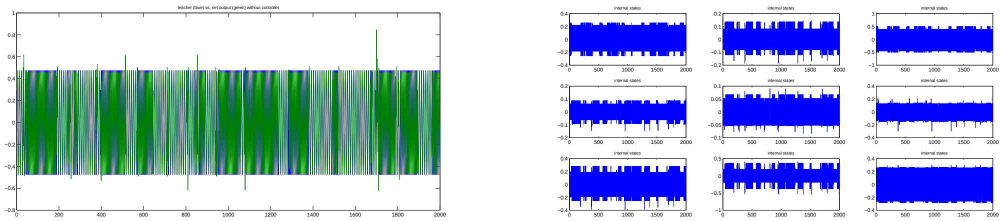
(b) Distorted Input after the modulation

Figure 2: Input signals

2.3 Baseline Performance

2.3.1 Learning and Testing Undistorted Signal

One of the first step in the project was to train the network with unmodulated data to obtain the baseline performance. The network was driven with 7000 time steps unmodulated signal dismissing first 1000 time steps for initial wash out. The output weights were learned for this signal. The result obtained was train root mean square of 0.21531 and test root mean square of 0.2501 and with average output weight of 105.03 The output and the internal states of network can be observed in figure 3.



(a) Teacher Vs Output of trained network

(b) Internal States of some neurons

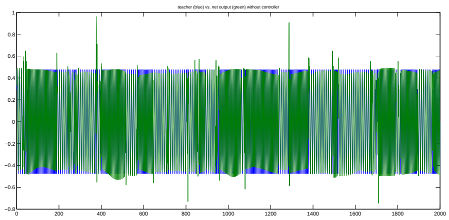
Figure 3: Baseline performance when network is driven with undistorted signal

We obtain smoothed internal states from the pool of excited reservoir. The four principal components were obtained from smoothed internal states . Since the input does not contain any slow distortion the reservoir of the network do not exhibit any distortion. The figure 3b shows the internal states are identical to the input data. The output of the network reflects the reservoir after learning the output weights. Hence teacher and produced output have no big differences. This can be inspected from figure 3a.

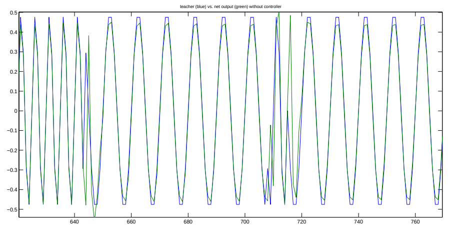
2.3.2 Learning and and Testing Distorted Signal

In the next part of the experiment the modulated signal with slow noise was given as input to the network. The training error obtained in this case was 0.2823 and the testing error of 0.31.The average output weight was 52.3.This means that due to the distortion the payload task is severely affected by the slow noise. The reservoir also exhibit the distortions from the input. During the learning process, the output weights are affected by the distracting noises in the reservoir. This will increase the error of the predicted output because of the task-irrelevant source introduced in the reservoir.

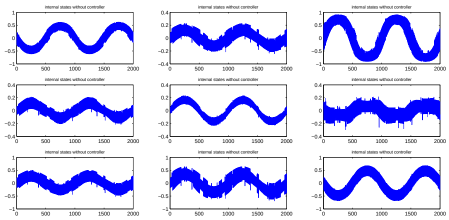
Figure 4a shows the plot of Teacher versus Output of the trained network in distorted case. Figure 4c shows the internal states of the echo state network when it is driven with the distorted signal. We can clearly observe that the reservoir is influenced by the slow noise and exhibit distracting signals.



(a) Teacher Vs output

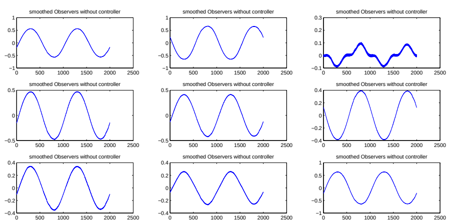


(b) Close Look of the figure a.

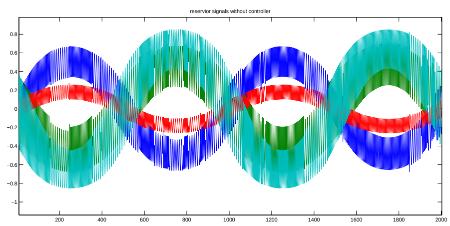


(c) Internal States

Figure 4: Payload Output and and the internal states without controller

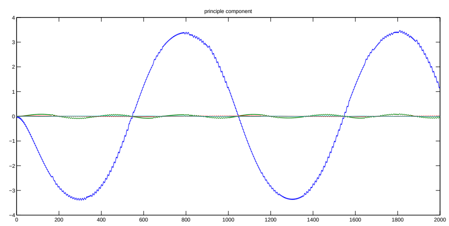


(a) Smoothed observables

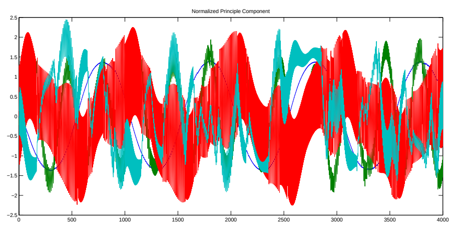


(b) Some internal states

Figure 5: Reservoir states and the smoothed observables without controller



(a) Four principal components



(b) Normalized Principal Components

Figure 6: Principal components in distorted signal without controller

Figure 5b shows some random reservoir. The reservoir exhibits a slow distortion signal. Similarly figure 5a shows the observables which are obtained from the reservoirs. We can find the principal components from the observables. Due to the induced noise in the reservoir the amplitude of first four Principal components of the smoothed observables have high amplitudes. The measured principal components can be seen as shown in figure 6.

2.4 Training and Testing Controller

Observables of a system models a real system in order to provide an estimate of its internal state of the real system. The observables play important role in the control theory. We have to obtain smoothed internal states i.e. observables $\mathbf{o}(n)$ from the stimulated reservoir. The observables are designed such that the resulting slow signal smooths out most of fast input signal.[4] The observers is formulated mathematically as in equation 3.

$$\mathbf{o}(n+1) = 0.98 * \mathbf{o}(n) + 0.02 * \mathbf{x}(n+1) \tag{3}$$

After driving the network with the distorted signal smoothed observables are obtained after the dismissing 1000 time steps for initial washout. The observables $\mathbf{O}(n)$ is centered to zero mean by subtracting the temporal means μ .

$\bar{\mathbf{o}}(n) = \mathbf{o}(n) - \mu$.

The correlation matrix $C = \overline{\mathbf{O}}\,\overline{\mathbf{O}}'$is computed where $'$denotes the transpose of the matrix.

Then we compute singular value decomposition (SVD) of the the correlation matrix C using mat-lab function $\mathbf{U} = svd(C)$

where $\mathbf{U}$ contains the directions for principal components.

Now we choose first four main directions$\mathbf{U}(:,1:4)$ and obtain the first four principal components from the centered observables $\overline{\mathbf{O}}$. The number of PCs to select depends upon the hand of the designer.

$P_0 = \mathbf{U}(:,1:4)'\overline{\mathbf{O}}$. P_0contains the first four PCs in its four rows.

Now we normalize the rows of $P_0$to unit variance by using $P = diag(\sigma^{-1})P_0$ where σ represents the vector of standard deviations of the rows of P_0.

Now our aim is to find four control vectors $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4$of size N such that it suppress the amplitude of main PCs to some degree during the payload task. The controller is obtained by finding correlation coefficient of each $p_i$with the centered reservoir states. The controller will have the intrinsic relation to dampen the noise. The four control vector is obtained mathematically as in equation 4.

$$\mathbf{c}_i = E[p_i(n)\bar{\mathbf{x}}(n)] \tag{4}$$

where $\bar{\mathbf{x}}(n)$ represents the centered reservoirs and $E[.]$denotes the expectation.[4]

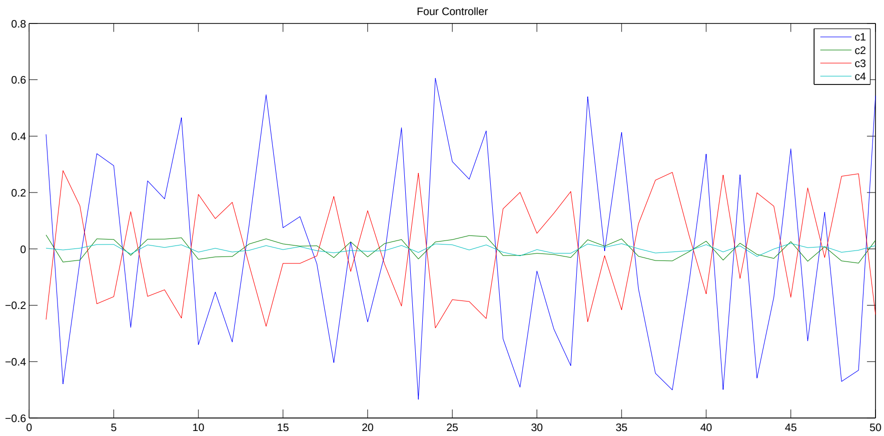


Figure 7: Four Controllers

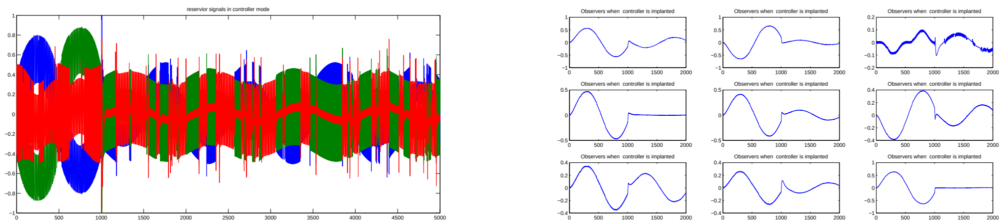
Figure 7 shows the four controllers which are responsible to suppress the distracting noise from the reservoir.

Now the network is again driven with distorted signal with controller switched on, using the control law as in equation 5. We generate online principal component as $P(n) = diag(\sigma^{-1})\mathbf{U}(:,1:4)'(\mathbf{o}(n) - \mu)$ from the stimulated reservoirs.[4]

Now the reservoirs of the network is controlled installing feedback proportional controller as shown in equation below.

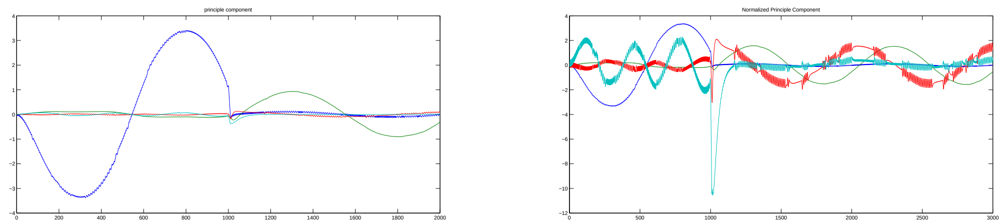
$$\mathbf{x}(n+1) = tanh(\mathbf{W}\mathbf{x}(n) + \mathbf{W}^{in}u(n) - \sum_{i=1}^{4} \gamma_i p_i(n)\mathbf{c}_i) \tag{5}$$

$\gamma_1, \gamma_2, \gamma_3, \gamma_4$ are the four gains for the controllers. The gains scales the controller to compensate the amplitude of the principal components. This leads to the concept of proportional controller to control the network. The values for four gains are assigned by tuning the values during running the network such that the gains can sufficiently suppress the amplitude of the principal components. In my case the first principal component have very high amplitude and other three principal components have negligible amplitudes. So we can only tune the first gain to scale the value of the controller.



(a) Internal states (b) Smoothed observables

Figure 8: Controlled Internal States and observables when controller is on after 1000 time stamp



(a) Four Principal Components (b) Normalized principal components

Figure 9: Principal Components when controller is on after 1000 time steps

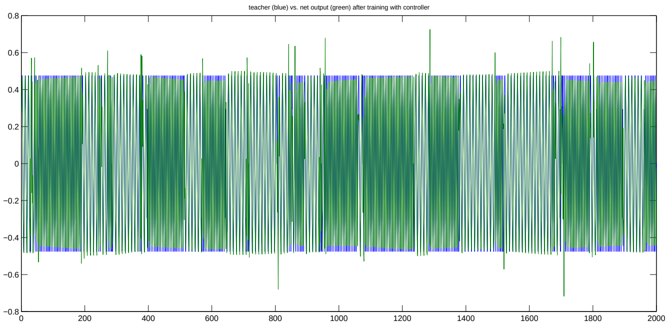
The gain should be tuned such that the controller can behave as suppressor for the leading PCs. The result after choosing appropriate values of the gain can be seen in table 1. The drastic differences in behavior of stimulated reservoir with and without controller and the PCs is demonstrated in the figures from 5a to 9

In figure 8b the observables becomes negligible after 1000 time steps. This is because my MAT LAB code installs the controller only after 1000 time steps so that we can clearly inspect the difference with and without controller. Before 1000 time steps the reservoir exhibits the distortion but after 1000 steps when the controller starts suppressing the PCs then the reservoir is excluded from the distracting noise.

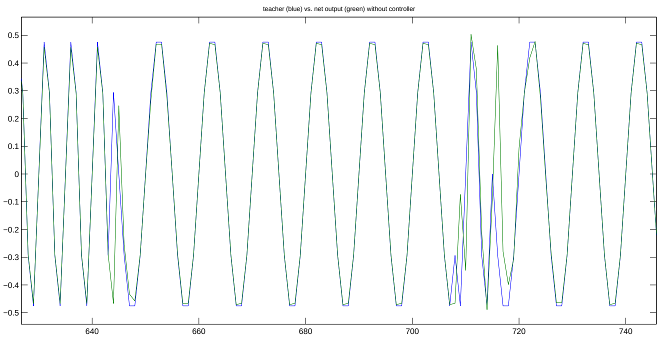
In figure 9 we can inspect that after 1000 time steps the blue line i.e. the principal component of maximum amplitude is suppressed after installing the controller. Before 1000 time steps the blue line has a high amplitude. This component induces the noise in the reservoir. Similarly the internal states in figure 10c reveals that after 1000 time step the reservoir are controlled. This illustration demonstrates that the controller works in principle as suppressor to the leading PCs. Now we can tune the gain values for more convincing output.

When we compare the internal states in figure 4c with internal states in figure 10c then we can see in figure 4c slow distortion is reflected in the reservoir where as in figure 10c the reservoir tries to stabilize themselves against the noises. This stabilized reservoir will facilitate the payload task learning. The figure 10a shows the payload output of the network after training with controller.

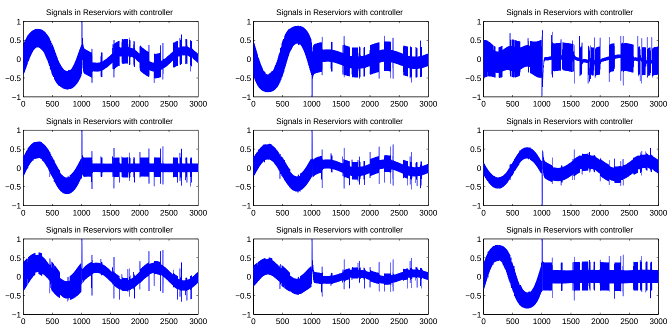
Now we can manifest that successful implementation of the controller tends to incline the network towards the homeostatic system. Homeostatic system is influenced from homeostasis in biological cells.[6] Homeostasis can be defined as the tendency of an organism or cell to regulate its internal conditions, such as the chemical composition of its body fluids, so as to maintain health and functioning, regardless of outside conditions. The organism or cell maintains homeostasis by monitoring its internal conditions and responding appropriately when these conditions deviate from their optimal state. The maintenance of a steady body temperature in warm-blooded animals is an example of homeostasis. In human beings, the homeostatic regulation of body temperature involves such mechanisms as sweating when the internal temperature becomes excessive and shivering to produce heat, as well as the generation of heat through metabolic processes when the internal temperature falls too low. These physiological manifestations reflects the efforts of the body to maintain its internal equilibrium. In homeostatic systems the internal functions are rigorously controlled by regulation mechanism that depend upon the external environment. Such a system responds and reacts to every disturbances and try to make modifications in its internal functions to maintain the internal balances. It is necessary for the system to maintain stability and to survive. Ecological,biological and social systems are homeostatic. If the system does not succeed in re-establishing its internal equilibrium it enters into another mode of behavior. The mode can ultimately lead to the destruction of the system if disturbances persist.[6] In our case the dynamics of reservoir after implementing the controller, tends to change for adaptability. The network now reacts to the disturbances imposed on it. This adaptability of reservoir against the distracting noise resembles the adaptive dynamics of living organism against external environment. We can manifest that controller reservoir dynamics in some sense can now immunize against the slow distortion to stabilize the reservoir states. However the system can be further extended to new version to immunize for any sort of changes in the environment.



(a) Teacher versus output



(b) Teacher Vs output in close view



(c) Internal States

Figure 10: Payload output and the internal states when controller is on after 1000 time steps

3 Results

When the controller is on and the network is driven with distorted input gain value should be tuned to suppress the noise. The amplitude of first principal component has maximum value among other three PCs as shown in figure 6. So I tuned the first gain to obtain best result. The table 1 shows the training root mean square and testing root mean square error at different values of gain.

γ_1	γ_2	γ_3	γ_4	Training error	Testing error
-90	0	0	0	0.227	0.248
-85	0	0	-0.5	0.231	0.254
-75	0	0	-0.5	0.219	0.237
-65	-0.5	-0.2	-0.1	0.229	0.256

Table 1: Result at different gain values

The average output weights was between 13.4 and 15 in all the above cases. The result shows that there is drastic improvement in the training error. If we recall error without controller it was 0.2823 but when controller is installed the training error reduces to same value when system is driven without distorted input. Hence, we can say the controller helps the ESN to resist against the slow varying noise. Since the input signal is completely stochastic the result may vary when the network is run next time however, it should be able to resist the noise with controller. When all the gain values are assigned to zero then network behaves in same way, when it is driven with distorted signal without the controller. This is because all the controller values are factored to zeros and there is no suppressing.

4 Further Works

4.1 Equilibration of Reservoirs

In this project only the output weights are recomputed in controller mode to compensate the slow varying noise.The weights of interconnected neurons $\mathbf{W}$ and weights from input to neuron $\mathbf{W}^{in}$ are constant. We can think of an approach to operate network without the controller for the distorted signal and achieve the performance like in case of undistorted case. The basic idea is to update the $\mathbf{W}$ and $\mathbf{W}^{in}$ and obtain new values that can neglect the noises. The weights are computed while running the network with distorted signal with controller is on. When the network now runs with recomputed new weights without controller, it should perform identically as with old weights with controller.This process of obtaining $\mathbf{W}_{eq}$and $\mathbf{W}^{in}$is called as equilibration of the reservoir.

I have left this task of equilibrating of reservoir as further work and research part of this project. However, I would like to include in this paper the technique of recomputing the internal weights. Let $\mathbf{x}(n)$ be the reservoir state signal obtained from equation 1 when driving the network with distorted signal and with the controller. We can recompute the input weights and internal weights such that they optimally reproduce $\mathbf{x}(n+1)$ from $u(n+1)$ and $\mathbf{x}(n)$ in a least mean square approach.[4]

$$\mathbf{W}_{eq}, \mathbf{W}_{eq}^{in} = argmin_{\tilde{\mathbf{w}},\widetilde{\mathbf{w}}} E\left[\left(\tanh^{-1}(\mathbf{x}(n+1)) - (\widetilde{\mathbf{W}}\mathbf{x}(n) + \widetilde{\mathbf{W}}^{in}u(n+1))\right)^2\right] \tag{6}$$

Then, the network can be run with updated weights with distorted input. These new weights obtained will help the network to self immunize against the slow varying noises. This concept of updating internal weights inside reservoir also manifest the concept of homeostatic system i.e now the system is capable to immunize against the external distortion. This concept of homeostatic was inherited from biological homeostasis behavior of cell where organism have tendency to maintain internal equilibrium by adjusting its physiological processes.

5 Discussion

Input signal modulated by slow distortions, when drives the ESN the slow variation input will be also induced in the reservoir dynamics. This was clearly inspected from the figures 4c. We can trace this slow variation in reservoir by finding the first few leading principal components from the smoothed reservoir data. Once these PCs are traced we can think of suppressing the amplitude of these PCs to stabilize the reservoir. If slow noise exists in the reservoir during learning, the output weights had to compensate the noise and system may become unreliable. There can be many techniques to suppress these PCs which is responsible to induce disturbances in the reservoir. In this paper simple approach of proportional controller is implemented. Control vectors are trained in such a way that they attempt to reduce the noise inside the reservoir. The controller was obtained from the coefficient correlation of one principal component with centered reservoir states. On implementing this controller during learning of payload task we observed that the amplitude of principal component is suppressed to certain degree as shown in figure 9. The reservoir dynamics is controlled by the feedback controller. The controller is called feedback controller because it gets feedback of PCs from reservoir during learning and implements the controller to compensate the high amplitudes of PCs. When network is trained in controller mode, the payload task performance is similar to the case when it was trained with undistorted signal. The controller has now the tendency to stabilize the reservoir against the distracting behavior of the input. This behavior of controller mimics the homeostasis behavior of biological cells. The network has now tendency to adapt for external undesired changes for slow distortions. This concept of artificial adaptability implemented in this project does not completely resembles biologically inspired neurons. Ashby completely defined homeostasis as the ability to adapt to a continuously changing and unpredictable environment. During the adaptive process some parameters need to be kept within pre-determined boundaries, either by evolutionary changes, physiological reactions, sensorial adjustment or by simply learning novel behaviors.We can combine biological inspirations and Ashby's motivated applications for the synthesis of more autonomous network that can respond to any kind of external disturbances.[1, 6]

Besides, to measure the consistency of the controller we can vary the modulation of the signal in different ways. We can make frequency modulation for distortion and inspect how the controller behaves for the adaptability of reservoir in such situation. The task for the network may become more complicated if there is an extreme distortion in the input signal. Further we can increase the difficulty of payload task. As in our experiment the task is to predict output by one time step ahead of input, the task can be made two or three time step ahead prediction. In this experiment the controller is simple proportional controller. It can be modified and imprroved to proportional integrator controller by adding integral terms to the error obtained. As already mentioned in section 4.1 the neurons weight can be recomputed to handle the distorted signal such that network can perform in accurate manner without the controller. Such network will have tendency to behave like a real homeostasis cell.

References

[1] Iizuka H., Di Paolo E.A., and SAB 2008 10th International Conference on Simulation of Adaptive Behavior. Extended homeostatic adaptation: Improving the link between internal and behavioural stability. *Lect.Notes Comput.Sci.Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5040 LNAI:1–11, 2008. ID: 283730574.

[2] Herbert Jaeger. *The "echo state" : approach to analysing and training recurrent neural networks*. GMD-Forschungszentrum Informationstechnik, Sankt Augustin, 2001. ID: 76383651.

[3] Herbert Jaeger. *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the "echo state network" approach*. GMD-Forschungszentrum Informationstechnik, Sankt Augustin, 2002. ID: 76475401.

[4] Herbert Jaeger. *Reservoir Self-Control for Achieving Invariance Against Slow Input Distortions* technical report no. 23 http://minds.jacobs-university.de/teaching/mlsemspring12mar12qa. Technical report, 2010.

[5] Herbert Jaeger and Harald Haas. *Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication* http://www.faculty.jacobs-university.de/hjaeger/pubs/esnscience04.pdf. 2004.

[6] R. C. Moioli, P. A. Vargas, F. J. Von Zuben, and P. Husbands. Towards the evolution of an artificial homeostatic system. In *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, pages 4023–4030, 2008. ID: 1.