Predicting Performance using Approximate State Space Model for Liquid State Machines

A. Gorad, V. Saraswat, U. Ganguly
Indian Institute of Technology, Bombay, India, Phone +91 22 2576 7698
ajinkyagorad@ee.iitb.ac.in

Abstract—Liquid State Machine (LSM) is a brain-inspired architecture used for solving problems like speech recognition and time series prediction. LSM comprises of a randomly connected recurrent network of spiking neurons. This network propagates the non-linear neuronal and synaptic dynamics. Maass et al. have argued that the non-linear dynamics of LSMs is essential for its performance as a universal computer. Lyapunov exponent ($\mu$), used to characterize the non-linearity of the network, correlates well with LSM performance. We propose a complementary approach of approximating the LSM dynamics with a linear state space representation. The spike rates from this model are well correlated to the spike rates from LSM. Such equivalence allows the extraction of a memory metric ($\tau_M$) from the state transition matrix. $\tau_M$ displays high correlation with performance. Further, high $\tau_M$ system require lesser epochs to achieve a given accuracy. Being computationally cheap (1800× time efficient compared to LSM), the $\tau_M$ metric enables exploration of the vast parameter design space. We observe that the performance correlation of the $\tau_M$ surpasses the Lyapunov exponent ($\mu$), (2 – 4× improvement) in the high-performance regime over multiple datasets. In fact, while $\mu$ increases monotonically with network activity, the performance reaches a maxima at a specific activity described in literature as the edge of chaos. On the other hand, $\tau_M$ remains correlated with LSM performance even as $\mu$ increases monotonically. Hence, $\tau_M$ captures the useful memory of network activity that enables LSM performance. It also enables rapid design space exploration and fine-tuning of LSM parameters for high performance.

Index Terms—LSM, SNN, State Space model, performance prediction, dynamics, neural networks

I. INTRODUCTION

The brain inspired computational framework of Liquid State Machines (LSMs) was introduced by Maass in 2002 [1]. LSM consists of a large recurrent network of randomly connected spiking neurons called the reservoir. The components of this network, namely the neurons and synapses, all follow highly non-linear dynamics. Depending on the extent and strength of connectivity, the reservoir propagates this non-linear dynamics in a recurrent manner. Maass et al. have argued that these non-linear operations performed by LSMs allow it to display high performance and universal computational capabilities [1]. Many applications are based on the belief that non linearity of the LSMs enable powerful data processing [2]–[4]. In [3] non-linear computations were performed on time series data. In [4], LSM was used for movement prediction task and was shown to perform a non-linear technique of Kernal Principal Component Analysis. Speech recognition, time series prediction, and robot control are few of the many versatile applications for which LSM has demonstrated excellent performance [2], [5]–[7].

LSMs have a vast set of design parameters with no direct relation with the performance. It is also known that, depending on the reservoir connectivity, an LSM may function in the region of low propagation i.e. input spikes may cause a chain reaction in neurons in the LSM to propagate spikes in the network for a short time. Higher propagation may ultimately lead to chaotic dynamics [8]. However, in order to achieve the best performance, the network has to function at an optimal level of activity or, as the literature cites it, at the “edge of
A. Speech preprocessing

Speech pre-processing stage for an LSM consists of a human ear-like model, namely Lyon’s Auditory Cochlear model [21]. It consists of a cascade of second order filters to produce a response for each channel. The output of each second order filter is rectified and low-pass filtered to get smooth signal. The shape of the second order filters is determined by the quality factor $Q$ and its step size $\Delta_{\text{step}}$ [19]. Human ears have a large dynamic range (60-80 dB), thus, an Automatic Gain Control (AGC) stage with particular target amplitude $A_m$ and timescale $\tau_{\text{AGC}}$ is incorporated in the Lyon’s model. Since the samples in TI-46 dataset have the sampling rate of 12.5 KHz, the output of this AGC, called cochleogram was decimated in time by decimation factor $df$ of 12 for simulation purposes. This was done with the help of Auditory Toolbox in MATLAB [19]. This decimated output is converted into spikes trains using BSA algorithm [20]. A second order filter $h_{\text{BSA}}$ was used for this encoding scheme with the time constants of $\tau_b1$ & $\tau_b2$. Finally, 77 spike trains were generated for a corresponding input speech sample in the preprocessing stage.

Relevant parameters for this stage are given in Table I.

$$h_{\text{BSA}} = (e^{-\frac{t}{\tau_b1}} - e^{-\frac{t}{\tau_b2}})H(t)$$  \hspace{1cm} (1)

| Parameter | Value | Parameter | Value |
|-----------|-------|-----------|-------|
| $Q$       | 8     | $df$      | 12    |
| $\Delta_{\text{step}}$ | 0.25 | $\tau_b1$, $\tau_b2$ | 4, 1 ms |
| $A_m$     | 0.0004 | $\tau_{\text{AGC}}$ | 32 ms |

B. Liquid Reservoir

Liquid Reservoir is taken to be a 5x5x5 3D grid of Leaky Integrate and Fire (LIF) neurons with fraction $f_+$ as excitatory and the rest as inhibitory neurons in the simulations. Neural dynamics are given by [22], where $V$ is the membrane potential with time constant $\tau_{\text{Neu}}$, refractory period $T_{rp}$, threshold voltage $V_{th}$ and elicited spike time $t_s$, $v$ is the synaptic input to the neuron. $w_{ij}$ is the connection strength from $N_i$ to $N_j$.

$$\frac{\partial V}{\partial t} = -\frac{V}{\tau_{\text{Neu}}} + \sum_i \sum_j w_{ij}v_j$$  \hspace{1cm} (2)

$$V > V_{th} \rightarrow V = 0 \forall \ t_s < t < t_s + T_{rp}$$  \hspace{1cm} (3)
Neurons are connected with synapses of specific weights by the probabilistic rule given in (4), where $D(N_1, N_2)$ is the distance between neurons $N_1$ and $N_2$ in the reservoir, $\lambda$ is the effective synaptic distance, $K$ is the connection probability, and it can take values as $K_{EE}, K_{EI}, K_{IE}, K_{II}$. For e.g. $K_{EI}$ corresponds to the probability of connection from excitatory (E) to inhibitory (I) neuron along with the radial dependence (4).

$$P(N_1, N_2) = K \cdot e^{-\frac{D^2(N_1, N_2)}{\lambda^2}}$$  \hspace{1cm} (4)$$

Similarly, synaptic weights $w_{ij}$ can take values $W_{EE}, W_{EI}, W_{IE}, W_{II}$ in the liquid (where subscripts denote E: Excitatory, and I: Inhibitory e.g. $W_{IE}$ denotes weight of the inhibitory to excitatory connection) and are constant in time. Synaptic delays are fixed to $d_{ij}$ for every synapse. Both the excitatory and inhibitory synapses are second order and follow the timescales of ($\tau_{1E}, \tau_{2E}$) for excitatory and ($\tau_{1I}, \tau_{2I}$) for inhibitory connections in the reservoir. These timescales correspond to $\tau_1$ and $\tau_2$ for synaptic dynamics given by (5), where $H$ is the unit step function.

$$v = \frac{1}{\tau_1 - \tau_2} \left( e^{-\frac{t - t_s}{\tau_1}} - e^{-\frac{t - t_s}{\tau_2}} \right) H(t - t_s)$$  \hspace{1cm} (5)$$

Each input spike train from the preprocessing stage is given to randomly chosen $F_{in}$ neurons in the reservoir with uniformly distributed synaptic weights of $\pm W_{in}$. For these synapses, only excitatory timescales were used. Default network parameters for simulations on TI-46 dataset are given in Table. II which are also mentioned in [15].

### C. Classifier

To recognize the class of the input from the multidimensional liquid response, generally, a fully connected layer of spiking readout neurons is used. Excitatory timescales were used for both excitatory and inhibitory synapses connecting the reservoir to the classifier. Since the only weight update in an LSM are to happen here, it needs a learning rule which learns the weights to extract useful features from the reservoir. After training, the class corresponding to the most spiked neuron is regarded as the classification decision of the LSM for the given input.

We briefly describe the biologically inspired learning rule proposed in [15]. It uses calcium concentration, $c$ for depicting the activity of the neuron in the classifier to enable selective weight change during a training phase. The calcium concentration for a neuron is defined by a first order equation (6) with timescale $\tau_c$. Steady state concentration $c_s$ is approximately given by (7) and is the indicator of spike rate ($f_r$) of the neuron. Supervised learning with large forcing current $I_{teach}$ is used depending on the activity and desirability (or undesirability) of the neuron. The desired (or undesired) neuron is supplied $I_{\infty}$ (or $-I_{\infty}$) current to spike more (or less) for the present training input. For each output neuron, this rule uses a probabilistic weight update according to (9), (10) for the corresponding synapse whenever pre-neuron in the reservoir spikes. A constant weight update $\Delta w$ is added or subtracted, with probability $p^+$ or $p^-$ respectively, to the present synaptic weight depending on the input class and the limit on the activity (decided by $c_d$ & $\Delta c$). Hence, the weight is increased/decreased if the pre-neuron spikes and the classifier neuron is the desired/undesired neuron (9), (10). This weight is artificially limited by value $W_{lim}$. The default parameters are mentioned in Table. III.

$$\frac{dc}{dt} = -\frac{c}{\tau_c} + \delta(t - t_s)$$  \hspace{1cm} (6)$$

$$c_s = \frac{1}{e^{\frac{1}{\tau_c}} - 1} \approx \tau_c f_r$$  \hspace{1cm} (7)$$

$$I_{teach} = \begin{cases} +I_{\infty} \cdot H ((c_d + \delta c) - c) , & \text{if desired} \\ -I_{\infty} \cdot H ((c_d - \delta c) - c) , & \text{if undesired} \end{cases}$$  \hspace{1cm} (8)$$

$$w \xleftarrow{p^+} w + \Delta w; \ c_d < c < c_d + \Delta c$$  \hspace{1cm} (9)$$

$$w \xleftarrow{p^-} w - \Delta w; \ c_d - \Delta c < c < c_d$$  \hspace{1cm} (10)$$

| Parameter | Value | Parameter | Value | Parameter | Value |
|-----------|-------|-----------|-------|-----------|-------|
| $W_{EE}$  | 3     | $K_{EE}$  | 0.45  | $\tau_{1E}$ | 8 ms |
| $W_{EI}$  | 6     | $K_{EI}$  | 0.3   | $\tau_{2E}$ | 4 ms |
| $W_{IE}$  | -2    | $K_{IE}$  | 0.6   | $\tau_{1I}$ | 4 ms |
| $W_{II}$  | -2    | $K_{II}$  | 0.15  | $\tau_{2I}$ | 2 ms |
| $F_{in}$  | $\pm 8$ | $F_{in}$  | 0.85  | $\lambda$ | 2    |
| $d_{in}$  | 4     | $d_{in}$  | 1 ms  |           |       |

### D. Performance

We replicated the setup from [15], and matched the state-of-the-art performance for the chosen reservoir size of 5x5x5 for TI46 speech dataset (Table. IV). System is trained and for 200 epochs on 500 TI-46 spoken digit samples. Performance is evaluated using 5 fold testing where accuracy is averaged over the last 20 epochs [15]. Samples used in training and testing consisted of a uniform distribution of 50 samples for each digit ‘0-9’ and 100 samples for each speaker, among 5 female speakers. Time step of 1ms was used in all the simulations. Simulation was done in Matlab and took a wall-clock time of 14 hours for each run on a Intel Xeon processor running at 2.4 GHz.

### E. Reservoir-less Network

We use a feed-forward fully-connected reservoir-less network to benchmark against the LSM. We train the same...
TABLE IV

| Work                  | Dataset | Accuracy (%) |
|-----------------------|---------|--------------|
| Our                   | TI-46   | 99.09        |
| Zhang et al. [15]     | TI-46   | 99.10        |
| Verstraeten et al. [4] | TI-46   | 99.5         |
| Wade et al. [24]      | TI-46   | 95.25        |
| Dibazar et al. [23]   | TIDigits| 85.5         |
| Tavanaei et al. [22]  | Aurora  | 91           |

preprocessed input on the same classifier. The difference between the performance by this method and the performance by the LSM is the gain/loss in the performance by introducing the reservoir. Generally, LSMs will benefit from the higher dimensional mapping and the recurrent dynamics of the network.

III. METHODOLOGY

This section describes how the discrete state space approximation was developed for the LSM dynamics described in the background and the resulting memory metric.

A. State Space Approximation

To study the dynamics of the spiking trajectories of LSM, we consider the spike rate column vectors for input ($U^k$), reservoir ($X^k$) and readout ($R_o^k$), where each row of the vector corresponds to an instantaneous spike rate at time $k$ for a neuron. Spike rate activity calculated as the average number of spikes in a rectangular window of 50 ms, and is a column vector where each row corresponds to a neuron. This spike rate activity spans a trajectory in a multi-dimensional space (each dimension is represents a different neurons) with time.

The input activity $U^k$ gets mapped to the higher dimensional space of the reservoir. The future activity or state of the liquid $X^{k+1}$ can be written as a function $f$ of its present input $U^k$ and the current state $X^k$.

\[
X^{k+1} = f(X^k, U^k)
\]

Reservoir and readout neurons, we get size of $A$ to be $N \times N$, $B$ to be $N \times M$ and $W$ to be $L \times N$. Concatenation of matrices $X$ and $U$ is represented as $[X|U]$.

\[
X_{i+1} = [A|B] \cdot [X|U]^T
\]

\[
\Rightarrow [A|B] = X_{i+1} \cdot \text{pinv}([X|U]^T)
\]

\[
R_o = W \cdot X \Rightarrow [A|B] = R_o \cdot \text{pinv}(X)
\]

Once $A$, $B$ and $W$ are determined, Knowing only the input $U$, we estimate the $\hat{X}$ and, finally, $\hat{R}_o$ from the estimated $\hat{X}$. To evaluate the effectiveness of state space modelling of an LSM, we find the correlation coefficient of the actual response $X$ with predicted response $\hat{X}$ knowing the input ($U \rightarrow X$). We also evaluate other combinations of prediction to see how well this approximation holds. Forward combinations include $U \rightarrow \hat{X}$, $X \rightarrow \hat{R}_o$ and $U \rightarrow \hat{X} \rightarrow \hat{R}_o$. Reverse combinations include $R_o \rightarrow \hat{X}$, $X \rightarrow \hat{U}$ and $R_o \rightarrow \hat{X} \rightarrow \hat{U}$. This correlation coefficient qualifies the ability of state space model LSM (discussed in Section IV).

B. Concept of memory

For an $N$ dimensional state space represented by $X$ given by $[X|$ having $N$ time constants $\tau_i$ Memory of such a system can be defined as the mean of the time constants $\tau_i$.

\[
\hat{X} = \text{diag}(\frac{1}{\tau_1}, \frac{1}{\tau_2}, ..., \frac{1}{\tau_N}) \cdot X
\]

\[
\tau_M = \frac{1}{N} \sum_{i=1}^{N} \tau_i
\]

A discrete first order system with time constant $\tau_i$ is represented as using Euler method. Here, $h$ is the time step of the discretized system.

\[
x_{i+1} = (1 - \frac{h}{\tau_i}) x_i
\]

For a system matrix $A$ of size $N \times N$ in a discrete state space system which defines the time dynamics, we get its diagonal entries in vector $a$. From this, we propose the memory metric $\tau_M$ for an approximate model of the reservoir to be.

\[
a = \text{diag}(A)
\]

\[
\tau_M = \frac{1}{N} \sum_{i=1}^{N} \frac{h}{1 - |a_i|}
\]

This memory metric is calculated and and its relation with performance is explored (presented in Section IV). It is also compared to the previously identified state of the art performance prediction metric Lyapunov exponent ($\mu$) Intuitively, Lyapunov exponent simply characterizes the chaotic behaviour of the network by measuring the difference in output for two almost identical inputs as the measure of the network’s ability to resolve small differences in inputs. As such, it is not an equivalent model for LSM. It is calculated.
as the average over the scaled exponents ($\mu_i$) for classes $i = 1, 2, ..., 10$, where for each class $i$, $\mu_i$ is calculated from two samples $u_{1i}$, $u_{2i}$ and their reservoir response $x_{1i}$, $x_{2i}$ using (23).

$$\mu_i = \ln \frac{||x_{1i}(t) - x_{2i}(t)||}{||u_{1i}(t) - u_{2i}(t)||}$$  \hspace{1cm} (23)

C. Simulation Methodology

The reservoir in an LSM has a large number of parameters defining its design space. Study by [17] identified few key parameters, which include synaptic scaling $\alpha_w$ and effective connectivity distance $\lambda$. We vary the given synaptic weights by a constant factor $\alpha_w$. We simulate the LSM for speech recognition task using TI-46 dataset and evaluate the performance over 12 different $\alpha_w$, each comprising of 4 randomly generated structures for the same parameter settings.

IV. RESULTS AND DISCUSSION

A. Similarity to State Space

We estimate the reservoir spikes $\hat{X}$ and find them to be well matched with the actual spike rate $X$ for the transformation $U \rightarrow \hat{X}$ using the state space approximation with Pearson Correlation Coefficient (PCC) of 0.92. Figure 2 shows the visual representation of actual reservoir spike rate and the estimated spike rate for each neuron in the reservoir in the high performance region of operation.

Further, correlation coefficients for all model estimated spike rates with exact spike rates are evaluated as a function of performance for both the forward and reverse pathways (Fig. 3). Forward estimation gave higher correlation values than reverse estimation. Further, when estimating the reservoir spikes from the input spikes ($U \rightarrow \hat{X}$), we find that this model is a good fit when error is small. In other words, the mapping of LSM to state space is more accurate when the output performance of the network is high.

We obtain the state space model estimation of the readout $f_w$ by $W$ with correlation greater than 0.85 for all ranges of error ($X \rightarrow \hat{R}_o$). The overall transition from $U \rightarrow \hat{X} \rightarrow \hat{R}_o$ is the combination of $U \rightarrow X \& X \rightarrow \hat{R}_o$ and hence, the correlation between $R_o$ and $\hat{R}_o$ given the input $U$ is lower than both.

The mapping from readout to reservoir ($R_o \rightarrow \hat{X}$) gives weak correlation coefficients in reverse estimation. This is intuitively expected as the readout need not represent all the information in the liquid mainly because the classifier has a lower dimensionality of 10 compared to 125 in the reservoir. In comparison, when we estimate the input (of dimensionality 77 in these simulations) from the reservoir $X \rightarrow \hat{U}$, the correlation is close to 0.8, and overall estimation correlation $R_o \rightarrow \hat{X} \rightarrow \hat{U}$ is less than both $R_o \rightarrow \hat{X} \& X \rightarrow \hat{U}$.

B. Performance as a function of Network Activity

We calculate the performance and the memory metric in different regimes of the LSM operation. For different reservoir weight scaling $\alpha_w$ (also called synaptic weight scaling) we obtain low-propagation (sparse activity $\alpha_w = 0.5$), significant-propagation (normal activity $\alpha_w = 0.8$), in-chaos (high activity $\alpha_w = 2$) and saturation (all neurons spiking $\alpha_w = 5$), shown in Fig 4 (a). Performance of the LSM for various synaptic weight scaling is found (Fig. 4(b)) and the memory metric $\tau_M$ of the corresponding state space modelled system is found using (22), and its variation against synaptic weight scaling is shown in Fig. 4(c).

For low-propagation in the reservoir where $\alpha_w$ is 0.5, there is mainly higher dimensional mapping of the input and limited dynamics take place in the reservoir. This higher dimensional mapping gives a boost of 5% to the accuracy 93% of reservoir-

Fig. 2. Well correlated (Pearson Correlation Coefficient i.e. PCC = 0.92) reservoir spike rates for LSM X and state space estimation $\hat{X}$ for each neuron in the reservoir as a function of time for a speech sample from TI-46 dataset.

Fig. 3. Average correlation coefficient between LSM and State Space Model spike rates vs. error (log scale) for all the possible transformations on all 500 samples of TI-46 dataset. Forward correlation are stronger than backward correlations. In the forward transformations, correlations improve in low error (high accuracy) regime.
Fig. 4. (a) Spike rate in the reservoir vs. synaptic scaling ($\alpha_w$) for 40 neurons resulting from a TI-46 speech sample. At the optimal value of $\alpha_w = 0.8$, the spike propagation is optimal which related to high accuracy. At lower $\alpha_w (< 0.8)$, the spike propagation is weak and reduces accuracy. For higher $\alpha_w (> 0.8)$, the spike propagation is excessive, leading to degraded performance. Higher $\alpha_w = 5$, leads to relentless spiking. (b) Error vs. synaptic weight scaling and (c) Memory metric $\tau_M$ vs. synaptic scaling both show non-monotonic but correlated dependence. One sigma variation over different runs is also plotted in grey in (b) and (c). The performance of a reservoir-less network, where the input is directly connected to the classifier, is shown in comparison to an LSM.

less approach (Fig. 4). Increasing the weights ($\alpha_w = 0.8$) allows input spikes to propagate through the reservoir and hence allows memory dynamics, which contribute to the performance. This results in the reduction of error to 1.5%. The size of the LSM and activity together maximize the accuracy which corresponds to increase in the memory metric.

However, increasing activity does not always correspond to good performance, and hence should also not contribute to the memory of useful information present in the system. Our proposed memory metric decreases because of the increasing activity, as the disorder increases and the LSM enters chaos. In saturation regime ($\alpha_w = 5$), it is evident that there is not much useful information as the pattern is lost permanently.

We think that the state space approach helps in characterizing the memory of LSM, because, useful information is related in time by its past activity and the current input. This approximation of trajectories provides an estimate of how the system evolves with the addition of new input. As the LSM enters chaotic regimes, the information is actually destroyed due to disturbance of this trajectory, this is highlighted by the breaks in the trajectory and cannot be captured by the smooth state space transition.

C. Comparison with Lyapunov Exponent

Following from the above discussion, the key property of the proposed memory ($\tau_M$) metric that emerged on varying the synaptic weight scaling ($\alpha_w$) is that $\tau_M$ has very high correlation with the recognition performance, i.e. large memory results in higher accuracy (Fig. 5 (a)). A more significant trend is observed when we focus on the high performance region. The PCC of $\tau_M$ with accuracy was found to be 0.93 over all possible error values and it increased to 0.95 for points in close proximity to small error (Fig. 5 (b)). As a comparison benchmark, we also calculated the absolute PCC with accuracy of the previously identified [10] state-of-the-art performance prediction metric, Lyapunov exponent ($\mu$) as 0.95 which rapidly dropped down as we focused on the high performance region of operation (Fig. 5 (b)). In other words, the overall performance correlation for both the metrics is at par, however, $\tau_M$ performs better than $\mu$ for low error regions. This is due to the monotonic nature of Lyapunov exponent with respect to synaptic scaling. In general the accuracy of LSMs increase as the activity increases but then reduces with the onset of chaos. $\tau_M$ captures this optimal activity threshold precisely increasing its utility over the Lyapunov exponent.

In addition, this behavior of $\tau_M$ is general and can be
Fig. 5. (a) Correlation of $\tau_M$ and $\mu$ with error for TI-46 speech dataset. (b) For overall error range, the correlation values are comparable for $\tau_M$ and $\mu$ (0.96). Correlation of $\tau_M$ (0.64) with accuracy is $2\times$ higher than that of $\mu$ (0.31) at lower errors is also seen by the low error expanded view behaviors in (c) and (d) of the encircled regions in (a).

extended to other datasets. We adopt a well known strategy for generating another test dataset [17], [25]. We construct 10 input spike train templates of 40 Hz poisson distributed, each comprising 10 spike channels with sample length of 200 ms. From each of them we generate 50 spike patterns, where spike of each template is jittered temporally with standard deviation of 16 ms. Any resulting spikes outside the window of sample are removed. We train this dataset for 100 epochs on the LSM with a different set of reservoir parameters as mentioned in [17] and vary the synaptic scaling from 0.1 to 4.

With this dataset, we found the PCC of memory metric to the performance to be 0.87 (Fig. 6 (a)), which is significantly greater than PCC of 0.18 (Fig. 6 (b)) for the Lyapunov exponent. This suggests memory metric $\tau_M$ is a better measure for performance with increased generality across datasets.

Another attractive property of $\tau_M$ is the associated time efficiency of the system to achieve a given accuracy. Fig. 7 (a) shows a large density of high $\tau_M$ systems achieving greater accuracy more quickly (in lower number of epochs). For performance to fall below a specified error, we get an exponential relationship between the number of epochs required and the associated memory metric of the system (Fig. 7 (b)). In other words, the memory metric has a direct impact on the learning rate of the classifier with faster learning being enabled in high $\tau_M$ systems.

D. Computational Efficiency

One simulation of exact LSM takes 1.5 hrs on average, if the system is trained for only 20 epochs. In comparison, one $\tau_M$ extraction from the approximate state space modelling takes only 3 seconds which is an $1800\times$ speed up. Further, $\tau_M$ calculation is $2\times$ faster than the calculation for the Lyapunov exponent.

E. Design Space Search

Given the performance predictive properties of $\tau_M$, the design space exploration is greatly simplified for LSMs. As highlighted in the simulation methodology, LSMs can be tuned by varying a large number of parameters. To highlight the utility of $\tau_M$ in this regard, we use previously identified synaptic weight scaling ($\alpha_w$) and effective connectivity $\lambda$ as the key performance parameters [17] to explore the accuracy achieved. We compare the accuracy with the calculated $\tau_M$ and $\mu$ over the same parameter space. The accuracy shows a region of maxima (dark green - Fig. 8 (a)) and falls off on either side over the parameter space. This behavior can also be seen for the $\tau_M$ obtained over the same parameter space (Fig. 8 (b)). Hence, calculation of $\tau_M$ from state space approximation provides an efficient method to identify the correct parameter values for high performing LSM. Again, as discussed earlier in Fig. 5 the Lyapunov exponent does not capture the performance optima unlike $\tau_M$ and is monotonic in
nature with network activity. Correlation was found to be 0.90 for the memory metric and 0.18 for the Lyapunov exponent in the high performance region (Accuracy > 85%) for the design space search conducted over $\alpha_w$ and $\lambda$ (Fig. 8).

It is widely believed that recurrent network of non-linear elements producing highly non-linear information processing to enable high performance in recognition tasks. In this paper, we present an alternative interpretation of LSM where we approximate it with a linear state space model - a well-established mathematical framework. We demonstrate high correlation of the model response with the LSTM. Equivalence with state space allows the definition of a memory metric which accounts for the additional performance enhancement in LSMS over and above the higher dimensional mapping. The utility of $\tau_M$ as a performance predictor is general across datasets and is also responsible for time efficient learning capabilities of the classier at the output. We compare and highlight the advantages of $\tau_M$ over the existing state-of-the-art performance predictor metric Lyapunov exponent $\mu$, where a 2-4× improved correlation of accuracy is observed for $\tau_M$. $\tau_M$ captures the maximum performance for optimum parameters in the design space where the network has optimal activity is at the “edge of the chaos”. In contrast, Lyapunov exponent is monotonic with network activity - resulting in poor performance prediction at high performance. Further, $\tau_M$ is monotonic with network activity - resulting in poor performance prediction at high performance. Further, $\tau_M$ is monotonic with network activity - resulting in poor performance prediction at high performance.

V. CONCLUSION & FUTURE WORK

We present an alternative interpretation of LSM where we highlight the advantages of $\tau_M$ in LSMs over and above the higher dimensional mapping. The utility of $\tau_M$ as a performance predictor is general across datasets and is also responsible for time efficient learning capabilities of the classifier at the output. We compare and highlight the advantages of $\tau_M$ over the existing state-of-the-art performance predictor metric Lyapunov exponent $\mu$, where a 2-4× improved correlation of accuracy is observed for $\tau_M$. $\tau_M$ captures the maximum performance for optimum parameters in the design space where the network has optimal activity is at the “edge of the chaos”. In contrast, Lyapunov exponent is monotonic with network activity - resulting in poor performance prediction at high performance. Further, the computational efficiency of the state space model (1800× compared to LSM) to compute $\tau_M$ enables rapid design space exploration and parameter tuning.

REFERENCES

[1] Maass, W., Natschlger, 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.

[2] Verstraeten, D., Schrauwen, B., Stroobandt, D., & Van Campenhout, J. (2005). Isolated word recognition with the liquid state machine: a case study. Information Processing Letters, 95(6), 521-528.

[3] Maass, W., Natschlger, T., & Markram, H. (2004). Computational models for generic cortical microcircuits. Computational neuroscience: A comprehensive approach, 18, 575.

[4] Burgstheiner, H., Kuhl, M., Leopold, A., & Steinbauer, G. (2007). Movement prediction from real-word images using a liquid state machine. Applied Intelligence, 26(2), 99-109.

[5] Rossell, J. L., Alomar, M. L., Morro, A., Oliver, A., & Canals, V. (2016). High-density liquid-state machine circuitry for time-series forecasting. International journal of neural systems, 26(05), 1550036.

[6] de Azambuja, R., Klein, F. B., Adams, S. V., Stoelen, M. F., & Cangelosi, A. (2017, May). Short-term plasticity in a liquid state machine biomimetic robot arm controller. In 2017 International Joint Conference on Neural Networks (IJCNN) (pp. 3399-3408). IEEE.

[7] Das, A., Pradhanap, P., Groenendaal, W., Adiraju, P., Rajan, R. T., Canhoo, F., ... & Van Hoof, C. (2018). Unsupervised heart-rate estimation in wearables with Liquid states and a probabilistic readout. Neural Networks, 99, 134-147.

[8] Schrauwen, B., Bsing, L., & Legenstein, R. A. (2009). On computational power and the order-chaos phase transition in reservoir computing. In Advances in Neural Information Processing Systems (pp. 1425-1432).

[9] Legenstein, R., & Maass, W. (2007). What makes a dynamical system computationally powerful. New directions in statistical signal processing: From systems to brain, 127-154.

[10] Chrol-Cannon, J., & Jin, Y. (2014). On the correlation between reservoir metrics and performance for time series classification under the influence of synaptic plasticity. Pho'c, 9(7), e101792.

[11] Goodman, E., & Ventura, D. (2006, July). Spatiotemporal pattern recognition via liquid state machines. In Neural Networks. 2006. IJCNN’06. International Joint Conference on (pp. 3848-3853). IEEE.

[12] Norton, D., & Ventura, D. (2010). Improving liquid state machines through iterative refinement of the reservoir. Neurocomputing, 73(16-18), 2893-2904.

[13] Maass, W., Legenstein, R. A., & Bertschinger, N. (2005). Methods for estimating the computational power and generalization capability of neural microcircuits. In Advances in neural information processing systems (pp. 865-872).

[14] Nise, N. S. (2007). CONTROL SYSTEMS ENGINEERING, John Wiley & Sons.

[15] Zhang, Y., Li, P., Jin, Y., & Choe, Y. (2015). A digital liquid state machine with biologically inspired learning and its application to speech recognition. IEEE transactions on neural networks and learning systems, 26(11), 2635-2649.

[16] Smith, M. R., Hill, A. J., Carlson, K. D., Vineyard, C. M., Donaldson, J., Follett, D. R., ... & Aimone, J. B. (2017, May). A novel digital neuromorphic architecture efficiently facilitating complex synaptic response functions applied to liquid state machines. In Neural Networks (IJCNN), 2017 International Joint Conference on (pp. 2421-2428). IEEE.

[17] Tu, H., Xu, J. X., Chong, E., & VanDongen, A. M. (2013). Effects of synaptic connectivity on liquid state machine performance. Neural Networks, 38, 39-51.

[18] Buonomano, D. V., & Maass, W. (2009). State-dependent computations: spatiotemporal processing in cortical networks. Nature Reviews Neuroscience, 10(2), 113.

[19] Slaney, M. (1998). Auditory toolbox. Interval Research Corporation, Tech. Rep, 10, 1998.

[20] Schrauwen, B., & Van Campenhout, J. (2003, July). BSA, a fast and accurate spike train encoding scheme. In Proceedings of the international joint conference on neural networks (Vol. 4, No. 4, pp. 2825-2830). Piscataway, NJ: IEEE.

[21] Lyon, R. (1982, May). A computational model of filtering, detection, and compression in the cochlea. In Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP’82. (Vol. 7, pp. 1282-1285). IEEE.

[22] Tavanaei, A., & Maida, A. S. (2017). A novel digital neuromorphic hardware platform for estimating the computational power and generalization capability of neural microcircuits. In Advances in neural information processing systems (pp. 865-872).

[23] Slaney, M. (1998). Auditory toolbox. Interval Research Corporation, Tech. Rep, 10, 1998.

[24] Schrauwen, B., & Van Campenhout, J. (2003, July). BSA, a fast and accurate spike train encoding scheme. In Proceedings of the international joint conference on neural networks (Vol. 4, No. 4, pp. 2825-2830). Piscataway, NJ: IEEE.

[25] Lyon, R. (1982, May). A computational model of filtering, detection, and compression in the cochlea. In Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP’82. (Vol. 7, pp. 1282-1285). IEEE.