

---

## Part I

# Supplementary Section

### A SUPPLEMENTARY SECTION A

#### A.1 EXPERIMENTAL SETUP

##### A.1.1 HRSNN AND MRSNN

Table 1: Table showing the description of the models described in the paper

	LIF Neurons	STDP parameters
HRSNN	Heterogeneous	Heterogeneous
MRSNN	Homogeneous	Homogeneous

The models used in this paper were the Heterogeneous Recurrent SNN (HRSNN) and the Homogeneous Recurrent SNN (MRSNN). Both models use STDP as the learning method. For MRSNN, we use STDP with uniform parameters for all the synapses. However, for HRSNN, we use a distribution for each parameter to get a rich class of diverse LTP/LTD dynamics. But, at the core, all the training is done using STDP.

##### A.1.2 LIF NEURON NUMERICAL IMPLEMENTATION

To implement the LIF model, we discretize time into multiples of a small-time step  $\Delta t$  so that spikes can only happen at multiples of  $\Delta t$ . (Cramer et al., 2020; Perez-Nieves et al., 2021) Thus, we can approximately solve Eq. ?? as

$$v_i(t+\Delta t) = v_i[t+1] = \beta (v_i[t] - v_0) + v_0 + (1-\beta)I_i[t] - (v_{th} - v_r) S_i[t] \quad \text{s.t. } \beta = \exp(-\Delta t/\tau_m) \quad (1)$$

It is to be noted here that we use this approximation for numerically solving the LIF neurons. Hence, although we use continuous notations for the remainder of the paper, it is to be noted that we use the discrete form discussed here for numerical solutions.

##### A.1.3 SPIKE CODING

**Encoding:** For the RSN to process our time series, the signal must be represented as spikes. We use a temporal encoding technique for representing signals in this paper. The spikes are only generated whenever the signal changes in value. The implementation of the temporal encoding used in this research is based on the Step-Forward (SF) algorithm (Petro et al., 2019). The percentage of neurons to input the spikes to ( $\alpha$ ) is also chosen to provide good recurrent layer dynamics.

**Decoding:** To represent the recurrent state, we use an exponentially decreasing rate decoding strategy by taking the sum of all the spikes  $s$  over the last  $\tau$  timesteps into account as follows:

$$x_i^X(t) = \sum_{n=0}^{\tau} \gamma^n s_i(t-n) \quad \forall i \in E$$

where  $X$  denotes the model representation. The parameters  $\tau$  and  $\gamma$  are balanced to optimize the memory size of the stored data (e.g.,  $\tau \leq 50$ ) and its containment of information, which includes adjusting  $\tau$  to the pace at which the temporal data is presented and processed. The state of the recurrent layer will be only based on the output of excitatory neurons. Thus, it is crucial for the discount  $\gamma$  not to be too small, as it possibly flattens older values in the window to 0, making part of the sliding window unusable. Recent spikes hardly affect the recurrent layer state when setting  $\gamma$  too high in combination with a large window size. This causes the decoder to react too late to recent information provided by the recurrent layer and complicates the learning process of the readout layer.

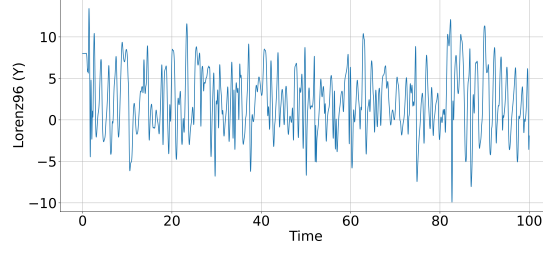


Figure 1: Figure showing a snippet for the Y dimension of the Lorenz96 time series used for the prediction problem

#### A.1.4 READOUT

After the initialization of the recurrent layer, the readout is the only component of the LSM with trainable parameters. It consists of a single fully connected layer for regression or classification. The readout does not have to be any deeper, as the output of the recurrent layer is already a high-dimensional representation of the processed input.  $x$  and  $y$  present the continuous signals of the time series  $T$  and model representation  $X$ .

$$x^T(t+k) = y^T(t) \approx \hat{y}^X(t) = f_\theta(x^X(t))$$

The mean squared error (MSE) is used as the loss function to train the readout, and the network was trained using the stochastic optimizer Adam (Kingma & Ba, 2014).

$$\mathcal{L}(y^T, \hat{y}^X) = \frac{1}{n} \sum_{i=0}^n (y_i^T - \hat{y}_i^X)^2$$

#### A.1.5 DATASETS

**Lorenz96: (Lorenz, 1996)** Our objective is more clearly demonstrated using the canonical chaotic system we will use as a test bed for the prediction capabilities of the HRSNN model. We use a multiscale Lorenz 96 system which is a set of coupled nonlinear ODEs and an extension of Lorenz’s original model (Thornes et al., 2017), (Chattopadhyay et al., 2020).

$$\begin{aligned} \frac{dX_k}{dt} &= X_{k-1} (X_{k+1} - X_{k-2}) + F - \frac{hc}{b} \Sigma_j Y_{j,k} \\ \frac{dY_{j,k}}{dt} &= -cbY_{j+1,k} (Y_{j+2,k} - Y_{j-1,k}) - cY_{j,k} + \frac{hc}{b} X_k - \frac{he}{d} \Sigma_i Z_{i,j,k} \\ \frac{dZ_{i,j,k}}{dt} &= edZ_{i-1,j,k} (Z_{i+1,j,k} - Z_{i-2,j,k}) - geZ_{i,j,k} + \frac{he}{d} Y_{j,k} \end{aligned} \quad (2)$$

This set of coupled nonlinear ordinary differential equations (ODEs) is a three-tier extension of Lorenz’s original model (Lorenz, 1963) and has been proposed by Thornes et al. (2017) as a fitting prototype for multiscale chaotic variability of the weather and climate system and a useful test bed for novel methods. In these equations,  $F = 20$  is a large-scale forcing that makes the system highly chaotic, and  $b = c = e = d = g = 10$  and  $h = 1$  are tuned to produce appropriate spatiotemporal variability. For this paper, we focus on predicting  $Y$  axes, which have relatively moderate amplitudes compared to  $X, Z$  and demonstrate high-frequency variability and intermittency, which makes the prediction problem difficult. It is to be noted here that the Lorenz 96 is a complex, difficult dataset for climate modeling. A snippet of the time series is shown in Fig. 1.

**SHD dataset:** We use the Spoken Heidelberg Digits spiking dataset to benchmark the HRSNN model with other standard spiking neural networks (Cramer et al., 2020). It was created based on the Heidelberg Digits (HD) audio dataset which comprises 20 classes of spoken digits from zero to nine in English and German, spoken by 12 individuals. For training and evaluation, the dataset (10420 samples) is split into a training set (8156 samples) and test set (2264 samples). To apply our RSNNs,

we converted all audio samples into 250- by-700 binary matrices. For this, all samples fit within a 1 the second window, shorter samples were padded with zeros, and longer samples were cut by removing the tail. Spikes were then binned in time bins, both of sizes 10ms and 4ms; for the RSNNs, the presence or non-presence of any spikes in the time bin is noted as a single binary event.

#### A.1.6 HYPERPARAMETERS

The hyperparameters used in this paper are summarized in Table 2

Table 2: Table showing the hyperparameters used in the experiments and their values

Parameter	Value	Description
$ E / N $	80%	Excitatory/inhibitory ratio
$\lambda$	2	Leak Exponent
$\tau$	50	Sliding window size
$\gamma^{\tau-1}$	0.02	Sliding Window Leak
SHD Parameters		
Parameter	Value	Description
$\tau$	17	Time delay
$a$	0.2	a parameter
$b$	0.1	b parameter
$n$	10	n parameter
$x_0$	1.2	Initial Condition
$h$	1.0	Time delta between two discrete timesteps
Lorenz-63 Parameters		
Parameter	Value	Description
$\rho$	28.0	$\rho$ -parameter
$\sigma$	10.0	$\sigma$ -parameter
$\beta$	8/3	$\beta$ -parameter
$x_0$	[1.0,1.0,1.0]	Initial Condition
$h$	0.03	Time delta between two discrete timesteps

#### A.2 MAXIMUM ENTROPY DISTRIBUTION

This subsection proves that the maximum entropy distribution with a fixed covariance matrix is Gaussian.

**Lemma:** Let  $q(\mathbf{r})$  be any density satisfying  $\int q(\mathbf{r})x_ix_jd\mathbf{r} = \Sigma_{ij}$ . Let  $p = \mathcal{N}(\mathbf{0}, \Sigma)$ . Then  $h(q) \leq h(p)$   
**Proof.**

$$\begin{aligned}
0 \leq \mathbb{KL}(q\|p) &= \int q(\mathbf{r}) \log \frac{q(\mathbf{r})}{p(\mathbf{r})} d\mathbf{r} \\
&= -h(q) - \int q(\mathbf{r}) \log p(\mathbf{r}) d\mathbf{r} \\
&= -h(q) - \int p(\mathbf{r}) \log p(\mathbf{r}) d\mathbf{r} \\
&= -h(q) + h(p)
\end{aligned}$$

since  $q$  and  $p$  yield the same moments for the quadratic form encoded by  $\log p(\mathbf{r})$ .

#### A.3 OPTIMAL HYPERPARAMETER SELECTION USING BAYESIAN OPTIMIZATION

Most recent research in Bayesian Optimization (BO) applications is limited to low-dimensional problems, as BO fails catastrophically when generalizing to high-dimensional problems (Frazier, 2018). However, in this paper, we aim to use BO to optimize the neuronal and synaptic parameters of a heterogeneous RSNN model. This BO problem thus entails a huge number of hyperparameters to be optimized; hence, using standard BO algorithms remains a significant challenge. Hence, to overcome this issue, we used a novel BO algorithm based on the assumption that our hyperparameters to be optimized are not completely random and uncorrelated but can be thought of as being drawn from a probability distribution as shown by Perez-Nieves et al. (2021). Thus, we use a modified BO to estimate *parameter distributions* for the LIF neurons and the STDP dynamics instead of searching for the individual parameters themselves. After learning the optimal distributions, we simply sample from the distribution to get the distribution of hyperparameters used in the model. To learn the probability distribution of the data, we modify the surrogate model and the acquisition function of the BO to treat the parameter distributions instead of individual variables. This makes our modified BO

Table 3: The list of parameter settings for the Bayesian Optimization-based hyperparameter search

Parameter	Initial Value	Range
$\eta$	10	(0,50)
$\gamma$	5	(0,10)
$\zeta$	2.5	(0,10)
$\eta^*$	1	(0,3)
$g$	2	(0,10)
$\omega$	0.5	(0,1)
$k$	50	(0,100)
$\lambda$ (SHD)	1	(0,2)
$\lambda$ (Lorenz)	1.5	(0,4)
$P_{IR}$	0.05	(0,0.1)
$\tau_{n-E}, \tau_{n-I}$ (SHD)	50ms	(0ms, 100ms)
$\tau_{n-E}, \tau_{n-I}$ (Lorenz)	100ms	(0ms, 300ms)
$A_{en-R}, A_{EE}, A_{EI}, A_{IE}, A_{II}$	30	(0,60)

highly scalable over all the variables (dimensions) used. The loss for the surrogate model’s update is calculated using the Wasserstein distance between the parameter distributions.

BO uses a Gaussian process to model the distribution of an objective function and an acquisition function to decide on points to evaluate. For data points in a target dataset  $x \in X$  and the corresponding label  $y \in Y$ , an SNN with network structure  $\mathcal{V}$  and neuron parameters  $\mathcal{W}$  acts as a function  $f_{\mathcal{V}, \mathcal{W}}(x)$  that maps input data  $x$  to predicted label  $\tilde{y}$ . The optimization problem in this work is defined as

$$\min_{\mathcal{V}, \mathcal{W}} \sum_{x \in X, y \in Y} \mathcal{L}(y, f_{\mathcal{V}, \mathcal{W}}(x)) \quad (3)$$

where  $\mathcal{V}$  is the set of hyperparameters of the neurons in  $\mathcal{R}$  (Details of hyperparameters given in the Supplementary) and  $\mathcal{W}$  is the multi-variate distribution constituting the distributions of (i) the membrane time constants  $\tau_{m-E}, \tau_{m-I}$  of the LIF neurons, (ii) the scaling function constants ( $A_+, A_-$ ) and (iii) the decay time constants  $\tau_+, \tau_-$  for the STDP learning rule in  $\mathcal{S}_{\mathcal{RR}}$ .

Again, BO needs a prior distribution of the objective function  $f(\vec{x})$  on the given data  $\mathcal{D}_{1:k} = \{\vec{x}_{1:k}, f(\vec{x}_{1:k})\}$ . In the Gaussian Process (GP)-based BO, we assume that the prior distribution of  $f(\vec{x}_{1:k})$  follows the multivariate Gaussian distribution, which follows a GP with mean  $\tilde{\mu}_{\mathcal{D}_{1:k}}$  and covariance  $\tilde{\Sigma}_{\mathcal{D}_{1:k}}$ . Thus, we estimate  $\tilde{\Sigma}_{\mathcal{D}_{1:k}}$  using the modified Matern kernel function. We use the loss function as  $d(x, x')$ , which is the Wasserstein distance between the multivariate distributions of the different parameters. That is, given two distributions of hyperparameters  $x_1, x_2$ , the distance between these two distributions (given as  $d(x_1, x_2)$ ) is used as the loss function in the Matern kernel for the modified BO. We want to learn the optimal distribution of hyperparameters  $x'$ , which maximizes the performance. It is to be noted here that for higher-dimensional metric spaces, we use the Sinkhorn distance as a regularized version of the Wasserstein distance to approximate the Wasserstein distance (Feydy et al., 2019).

$\mathcal{D}_{1:k}$  are the points evaluated by the objective function. The GP will estimate the mean  $\tilde{\mu}_{\mathcal{D}_{k:n}}$  and variance  $\tilde{\sigma}_{\mathcal{D}_{k:n}}$  for the rest unevaluated data  $\mathcal{D}_{k:n}$ . The acquisition function used in this work is the expected improvement (EI) of the prediction fitness as:

$$EI(\vec{x}_{k:n}) = (\tilde{\mu}_{\mathcal{D}_{k:n}} - f(x_{\text{best}})) \Phi(\vec{Z}) + \tilde{\sigma}_{\mathcal{D}_{k:n}} \phi(\vec{Z}) \quad (4)$$

where  $\Phi(\cdot)$  and  $\phi(\cdot)$  denote the probability distribution function and the cumulative distribution function of the prior distributions, respectively.  $f(x_{\text{best}}) = \max f(\vec{x}_{1:k})$  is the maximum value that has been evaluated by the original function  $f$  in all evaluated data  $\mathcal{D}_{1:k}$  and  $\vec{Z} = \frac{\tilde{\mu}_{\mathcal{D}_{k:n}} - f(x_{\text{best}})}{\tilde{\sigma}_{\mathcal{D}_{k:n}}}$ . BO will choose the data  $x_j = \operatorname{argmax} \{EI(\vec{x}_{k:n}); x_j \subseteq \vec{x}_{k:n}\}$  as the next point to be evaluated using the original objective function.

### A.3.1 OPTIMIZED HYPERPARAMETERS

The list of the hyperparameters optimized using the Bayesian Optimization technique is shown in Table 3. We also show the range of the hyperparameters used and the initial values. In addition to this, Table 4 enlist the final optimized distributions of the STDP and the LIF parameters obtained using BO.

Table 4: Table showing the average final distributions of the hyperparameters

	Parameter	Distribution	
4* STDP Parameter	$\tau_+$	Normal	$\bar{\mu} = 18.235$ $\bar{\sigma} = 1.522$
	$\tau_-$	Normal	$\bar{\mu} = 22.382$ $\bar{\sigma} = 1.768$
	$\eta_+$	Normal	$\bar{\mu} = 0.516$ $\bar{\sigma} = 0.0055$
	$\eta_-$	Normal	$\bar{\mu} = 0.448$ $\bar{\sigma} = 0.0057$
2* LIF Parameter	$\tau_m^{(e)}$	Gamma	$\bar{\alpha} = 2.89$ $1/\bar{\beta} = 0.248$
	$\tau_m^{(i)}$	Gamma	$\bar{\alpha} = 5.14$ $1/\bar{\beta} = 0.313$

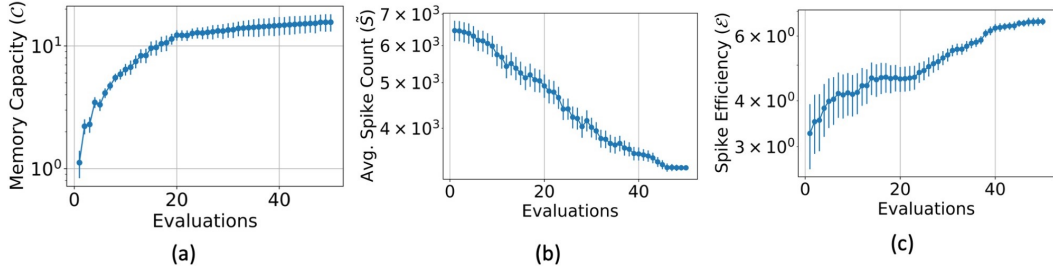


Figure 2: Figure Showing the convergence behaviors of the three types of BO described in the paper (a) BO optimizing the memory capacity  $\mathcal{C}$  (b) BO optimizing the average spike count  $\tilde{S}$  and (c) BO optimizing the spike efficiency  $\mathcal{E}$

#### A.3.2 CONVERGENCE ANALYSIS

We compare the convergence analysis of the three Bayesian Optimization techniques and the results are shown in Fig. 2. Each of the experiments was repeated five times and the mean and variance of the observations are shown in the Figure. It is to be noted here that since we define the BO as a minimization principle, we minimize  $\frac{1}{\mathcal{C}}$ ,  $\tilde{S}$  and  $\frac{1}{\mathcal{E}}$ .

#### A.4 COMPARING BAYESIAN OPTIMIZATION OBJECTIVE FUNCTIONS

We show the results of Bayesian Optimization results for the three cases we are considering in this paper for both the classification and prediction problems. The results for the classification problem are shown in Table 5. We tabulate the memory capacity, the average spike count and the observed accuracy for the three BO cases. Similarly, the results for the prediction problem are shown in Table 6. In that case, we tabulate the memory capacity, the average spike count and the observed NRMSE for the three BO cases. We rerun each of the experiments 5 times and report the mean and standard deviation of the results obtained.

#### A.5 COMPARING THE GENERALIZABILITY

We observed that increasing the neuronal heterogeneity increases the memory capacity of the network. However, this increment in the memory capacity might lead to a model which overfits the training data. However, the heterogeneous STDP model with varying synaptic dynamics gives rise to a heavy-tailed Feller process. Recent works by Simsekli et al. (2020) and Chakraborty & Mukhopadhyay (2021) show that the generalization error can be controlled by the Hausdorff dimension of the trajectories of the sample paths of the learning algorithm. This is intimately linked to the tail behavior of the driving process. The authors showed that heavier-tailed processes achieve better generalization. Thus, the tail index of the process can be used as a notion of capacity metric that estimates the generalization error, which does not necessarily grow with the number of parameters. The authors discuss that the stochastic process for the synaptic weights behaves like a Lévy motion around a local point. Because

Table 5: Table showing the performance of the Bayesian Optimization on the SHD Classification dataset for the three different cases where BO 1 optimizes  $\mathcal{C}$ , BO 2 optimizes  $\tilde{S}$  and BO 3 optimizes  $\mathcal{E}$

N_R	BO 1 Memory Capacity	BO 1 Average Spike Count	BO 1 Accuracy	BO 2 Memory Capacity	BO 2 Average Spike Count	BO 2 Accuracy	BO 3 Memory Capacity	BO 3 Average Spike Count	BO 3 Accuracy
100	5.31 ± 0.36	1399.14 ± 113.66	67.41 ± 4.86	5.22 ± 0.58	1189.37 ± 86.95	66.23 ± 5.74	5.12 ± 0.31	1268.86 ± 91.15	68.67 ± 4.97
200	5.45 ± 0.39	1479.86 ± 133.72	67.85 ± 4.16	5.38 ± 0.53	1233.19 ± 122.56	67.21 ± 5.49	5.28 ± 0.34	1328.18 ± 131.45	69.54 ± 4.27
300	6.72 ± 0.4	1541.14 ± 148.83	68.41 ± 4.87	6.42 ± 0.57	1325.25 ± 128.47	67.95 ± 5.11	6.54 ± 0.36	1391.68 ± 140.37	70.1 ± 4.69
400	7.21 ± 0.44	1682.68 ± 239.95	70.11 ± 4.33	6.91 ± 0.51	1425.69 ± 129.57	68.23 ± 5.27	7.01 ± 0.35	1511.79 ± 200.96	71.54 ± 4.06
500	8.59 ± 0.48	1768.24 ± 287.94	71.05 ± 3.98	7.69 ± 0.46	1555.29 ± 139.17	69.31 ± 5.03	8.63 ± 0.38	1621.8 ± 250.46	73.05 ± 3.87
1000	11.22 ± 0.46	2251.17 ± 319.75	72.93 ± 3.38	9.03 ± 0.44	2015.24 ± 147.44	70.89 ± 4.91	12.25 ± 0.39	2102.59 ± 279.86	75.32 ± 3.44
2000	13.3 ± 0.51	2566.21 ± 348.68	75.36 ± 3.29	9.89 ± 0.46	2314.59 ± 151.18	72.33 ± 4.88	13.95 ± 0.37	2410.08 ± 301.57	77.25 ± 3.17
3000	14.47 ± 0.53	2825.47 ± 355.87	77.14 ± 3.38	10.48 ± 0.41	2623.41 ± 177.94	74.63 ± 4.93	14.88 ± 0.42	2708.52 ± 315.34	78.21 ± 3.24
4000	15.17 ± 0.52	3551.07 ± 366.19	78.05 ± 3.25	11.57 ± 0.45	3045.28 ± 225.53	75.15 ± 4.85	15.87 ± 0.38	3218.42 ± 328.19	79.36 ± 3.13
5000	15.64 ± 0.57	4186.49 ± 383.09	78.92 ± 3.31	11.68 ± 0.48	3294.62 ± 241.14	75.87 ± 4.81	16.03 ± 0.41	3573.51 ± 331.18	80.49 ± 3.15

Table 6: Table showing the performance of the Bayesian Optimization on the Lorenz System Prediction dataset for the three different cases where BO 1 optimizes  $\mathcal{C}$ , BO 2 optimizes  $\tilde{S}$  and BO 3 optimizes  $\mathcal{E}$

N_R	BO 1 Memory Capacity	BO 1 Average Spike Count	BO 1 RMSE	BO 2 Memory Capacity	BO 2 Average Spike Count	BO 2 RMSE	BO 3 Memory Capacity	BO 3 Average Spike Count	BO 3 RMSE
100	3.56 ± 0.36	1354.35 ± 108.96	0.617 ± 0.019	3.02 ± 0.52	1101.25 ± 75.93	0.684 ± 0.026	3.45 ± 0.3	1207.52 ± 67.26	0.654 ± 0.0207
200	4.12 ± 0.39	1443.59 ± 127.23	0.587 ± 0.02	3.89 ± 0.48	1207.35 ± 118.57	0.639 ± 0.027	4.01 ± 0.33	1302.47 ± 96.05	0.613 ± 0.0218
300	5.26 ± 0.37	1499.62 ± 141.73	0.503 ± 0.027	4.44 ± 0.55	1257.26 ± 1257.26	0.558 ± 0.034	5.15 ± 0.34	1335.81 ± 168.02	0.531 ± 0.0287
400	6.37 ± 0.41	1528.73 ± 228.87	0.459 ± 0.033	5.87 ± 0.49	1304.35 ± 126.18	0.467 ± 0.04	6.05 ± 0.35	1415.32 ± 213.49	0.482 ± 0.0347
500	7.25 ± 0.45	1601.27 ± 277.97	0.389 ± 0.036	6.25 ± 0.45	1365.35 ± 137.04	0.421 ± 0.043	6.87 ± 0.36	1507.29 ± 219.58	0.411 ± 0.0377
1000	10.12 ± 0.43	1868.14 ± 301.17	0.316 ± 0.04	7.41 ± 0.51	1563.25 ± 146.76	0.396 ± 0.047	9.03 ± 0.37	1699.27 ± 275.79	0.332 ± 0.0417
2000	11.84 ± 0.48	2105.95 ± 331.54	0.293 ± 0.045	8.02 ± 0.5	1854.35 ± 150.28	0.351 ± 0.052	11.44 ± 0.39	2014.12 ± 280.03	0.301 ± 0.0467
3000	13.71 ± 0.51	2408.35 ± 348.26	0.258 ± 0.042	8.94 ± 0.53	2195.82 ± 179.75	0.335 ± 0.058	13.87 ± 0.4	2236.59 ± 281.05	0.241 ± 0.0482
4000	14.15 ± 0.52	2951.56 ± 352.66	0.242 ± 0.063	9.55 ± 0.55	2445.31 ± 217.73	0.326 ± 0.07	14.63 ± 0.41	2546.25 ± 289.81	0.227 ± 0.0649
5000	14.45 ± 0.54	3784.44 ± 353.51	0.203 ± 0.064	9.96 ± 0.58	2684.59 ± 234.63	0.302 ± 0.071	15.12 ± 0.42	2898.27 ± 307.14	0.195 ± 0.0655

of this locally regular behavior, the Hausdorff dimension can be bounded by the Blumenthal-Gettoor (BG) index (Blumenthal & Gettoor, 1960), which in turn depends on the tail behavior of the Lévy process. Thus, we can use the BG index as a bound for the Hausdorff dimension of the trajectories from the STDP learning process. Now, as the Hausdorff dimension is a measure of the generalization error and is also controlled by the tail behavior of the process, heavier tails imply less generalization error. In this paper, we empirically study the generalization ability of the HRSNN network using the BG index as a metric. We did the experiments on the 4 ablation study models for the classification task on the SHD dataset, and the results are reported in Table 7. From the table, we see that the heterogeneity in STDP improves the generalization error the most, while the heterogeneity in the LIF neurons increases the training and testing accuracies.

## A.6 RESULTS ON LIMITED TRAINING DATA

We have trained the models with limited training data. We observe that the HRSNN model with heterogeneous LIF and STDP dynamics not only has better testing accuracy but also shows better generalization behavior when compared to other homogeneous RSNN or the other ablation heterogeneous models (with heterogeneity in only one of them). Also, we see that the HRSNN model with heterogeneous STDP shows distinctly better generalization ability than the generalization ability of

Table 7: Table showing the Ablation Study for the comparison of the Generalizability of heterogeneous networks

	BG Index	Training Accuracy (A)	Testing Accuracy (B)	Generalization Error ( A-B )
Hom LIF Hom STDP	1.522	87.33	73.58	13.75
Hom LIF Het STDP	1.438	85.31	74.03	11.28
Het LIF Hom STDP	1.835	95.29	78.87	16.42
Het LIF Het STDP	1.711	94.32	80.49	13.83

Table 8: Table showing results with limited training data

Percentage Training Data	Train Accuracy (A)	Test Accuracy (B)	Generalization Error  A-B	Train Accuracy (A)	Test Accuracy (B)	Generalization Error  A-B
	<b>Heterogeneous LIF, Heterogeneous STDP</b>			<b>Homogeneous LIF, Homogeneous STDP</b>		
<b>100</b>	94.32	80.49	13.83	87.33	73.58	13.75
<b>90</b>	94.89	78.34	16.55	87.83	67.84	19.99
<b>80</b>	95.47	76.72	18.75	88.86	65.86	23
<b>70</b>	96.15	74.92	21.23	89.95	62.19	27.76
<b>60</b>	96.92	70.34	26.58	91.58	61.25	30.33
<b>50</b>	97.69	69.44	28.25	94.38	59.51	34.87
<b>40</b>	98.21	63.76	34.45	96.85	55.93	40.92
<b>30</b>	98.43	54.01	44.42	98.43	45.86	52.57
<b>20</b>	99.43	43.87	55.56	99.49	42.68	56.81
<b>10</b>	100	31.43	68.57	100	30.18	69.82
<b>5</b>	100	15.32	84.68	100	14.38	85.62
	<b>Heterogeneous LIF, Homogeneous STDP</b>			<b>Homogeneous LIF, Heterogeneous STDP</b>		
<b>100</b>	97.29	78.87	18.42	86.31	74.03	12.28
<b>90</b>	97.41	77.48	19.93	86.94	68.59	18.35
<b>80</b>	97.65	76.32	21.33	87.75	67.58	20.17
<b>70</b>	97.95	74.03	23.92	88.17	65.25	22.92
<b>60</b>	98.03	71.16	26.87	89.52	63.11	26.41
<b>50</b>	98.43	68.48	29.95	90.48	60.86	29.62
<b>40</b>	98.79	61.93	36.86	93.15	57.31	35.84
<b>30</b>	99.56	51.68	47.88	96.34	48.41	47.93
<b>20</b>	100	44.52	55.48	98.43	43.59	54.84
<b>10</b>	100	30.68	69.32	99.56	32.57	66.99
<b>5</b>	100	14.15	85.85	100	18.48	81.52

HRSNN with heterogeneous LIF neurons. On the other hand, the latter showcases significantly higher training and testing accuracy compared to the former model. This can be interpreted as follows: since heterogeneous LIF dynamics increase the memory capacity, it leads to an overfitting of the data. Heterogeneous STDP dynamics help in obtaining more generalizable solutions from this. Each has its own downsides; however, using HRSNN with both heterogeneous LIF and STDP dynamics shows better performance and generalization abilities, as seen from Table 8.

#### A.7 FURTHER EVALUATIONS

In Section B, we argued that as the heterogeneity in the neuronal parameters increases, the covariance decreases; hence the neurons become less correlated. In this section, we give empirical results to support the theory.

- **Impact of Heterogeneity on Covariance:** We plot the covariance matrices for different levels of heterogeneity  $\mathcal{J}$  (Eq. 40) for a small network with 50 neurons. The covariance matrix is calculated by taking the average neuronal states before the appearance of the first spike in the final layer. We see that as the heterogeneity in the neuronal parameters increases, the correlation between the neurons decreases. The results are shown in Fig. 3
- **Impact of Heterogeneity on Principal Components:** From the covariance plots, we see that increasing  $\mathcal{J}$  reduces the correlation between neurons. We also plot the probability density functions of the eigenvalues of the covariance matrix of the neurons with increasing heterogeneity in the neuronal parameters. We see that with higher heterogeneity in the neuronal parameters  $\mathcal{J}$ , the distribution of the eigenvalues of the covariance becomes flatter. This signifies that the covariance matrix has a lower variance for higher  $\mathcal{J}$ . A flatter distribution also indicates that a larger number of principal components are active. This supports our hypothesis that heterogeneity in the neuronal parameters increases the number

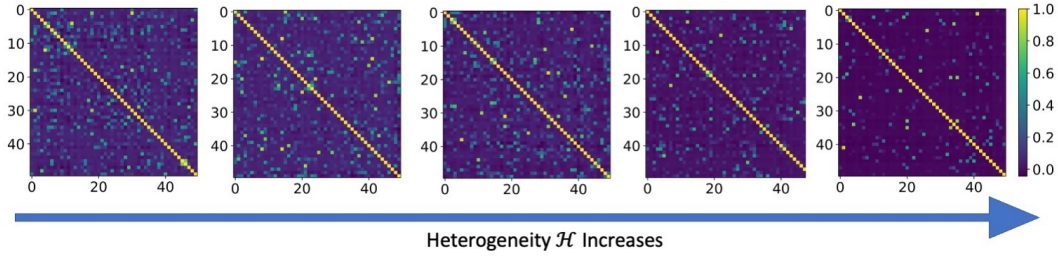


Figure 3: Figure showing as the heterogeneity in the neuronal parameters increases, the covariance between the neurons decreases

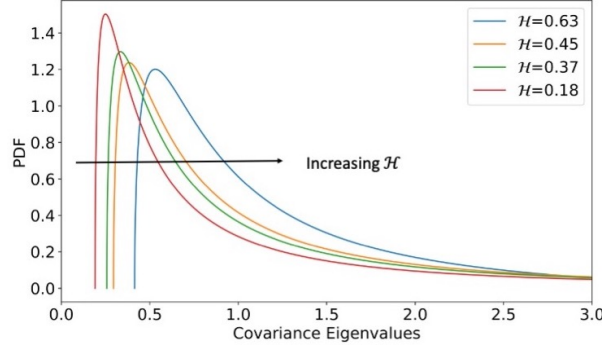


Figure 4: With higher heterogeneity in the neuronal parameters  $\mathcal{H}$ , the distribution of the eigenvalues of the covariance becomes flatter. This signifies that the covariance matrix has a lower variance for higher  $\mathcal{H}$ . A flatter distribution also signifies a greater number of principal components are active, which supports our hypothesis that heterogeneity in the neuronal parameters increases the number of principal components and helps in increasing the memory capacity of the model.

of principal components and helps increase the model’s memory capacity. The result is shown in this Fig. 4

- **Impact of Heterogeneity in STDP on Firing Rate:** We plot the mean firing rate of the neurons for the four types of HRSNNs and MRSNN with homogeneous LIF and STDP dynamics. We plot the results for a smaller network with 100 neurons and a Poisson input process. We see that the MRSNN model shows a much higher firing rate, especially at a higher frequency, demonstrating that MRSNN requires significantly more spikes than the HRSNN model. The result is shown in Fig. 5
- **Coupling Strength:** We note here that in this paper, we use (homogeneous or heterogeneous) STDP to learn the synaptic conductance connecting various neurons in the SNN. Therefore, we do not control the synaptic coupling strength as independent variables and hence, cannot perform control experiments with various extents of coupling strength. An interesting future extension of the results will be to quantify the coupling strength for HRSNN with heterogeneity in LIF and STDP dynamics. We can leverage McKenzie et al. (2021), where the authors proposed statistical tools to estimate synaptic coupling dynamics from spike-spike correlations.

## B SUPPLEMENTARY SECTION B

### B.1 APPROXIMATIONS AND ASSUMPTIONS

We make several approximations and assumptions for this section’s theoretical analysis of the heterogeneous RSN networks. Firstly, it must be noted that in this paper, the analytical relations are derived by taking the heterogeneity individually. i.e., when we consider heterogeneity in the neuronal parameters, we consider homogeneous STDP dynamics and vice-versa. In addition to this, we assume



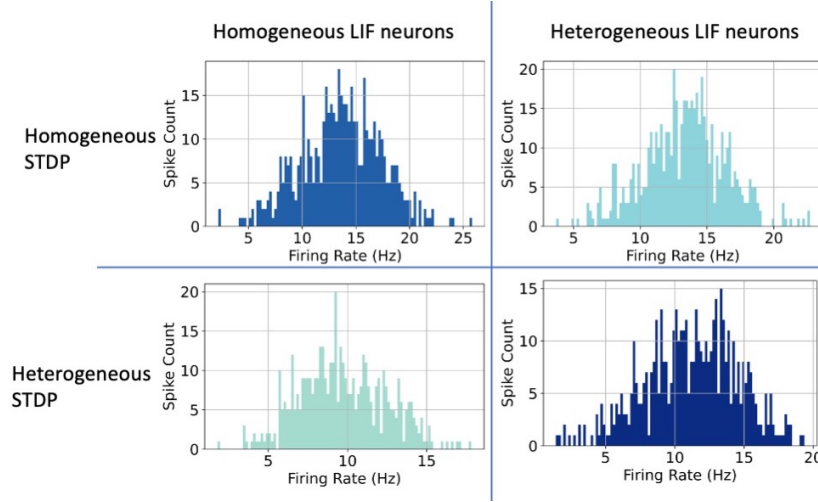


Figure 5: Figure showing the histograms of the firing rates for the four kinds of heterogeneous RSNN with homogeneous /heterogeneous neurons and synapses

*diffusion approximation.* That is, if a neuron receives Poissonian uncorrelated input spike trains and the contribution of a single synaptic connection is small compared to the distance between reset and threshold  $w \ll (V_\Theta - V_0)$ , the random input can be approximated by Gaussian white noise with mean  $\mu$  and noise intensity  $\sigma^2$ . This approximation does not hold if the network features highly correlated activity or receives strong external input common to many neurons. Also, we assume a fast/slow synaptic regime in which the synaptic time constant  $\tau_s$  is much shorter/longer than the membrane time constant  $\tau_m$ . In this work, we consider a mean-field approximation of the HRSNN network with heterogeneity in the parameters of the LIF neurons and the STDP dynamics independently.

## B.2 MEAN-FIELD REDUCTION MODEL OF HRSNN

In this section, we model the HRSNN network using heterogeneity in only the LIF neuron parameters. Following the works of Ly (2015), we can write the equations for the excitatory neurons indexed by  $j \in \{1, 2, \dots, N_e\}$  are:

$$\tau_m \frac{dv_j}{dt} = -v_j - g_{ie}(t)(v_j - \mathcal{E}_I) - g_{ee}(t)(v_j - \mathcal{E}_E) + \sigma_E \eta_j(t) \quad (5)$$

$$v_j(t^*) \geq \theta_j \text{ (refractory period)} \Rightarrow v_j(t^* + \tau_{ref}) = 0 \quad (6)$$

$$\tau_n \frac{d\eta_j}{dt} = -\eta_j + \sqrt{\tau_n} \xi_j(t) \quad (7)$$

$$g_{ee}(t) = q_j \frac{\gamma_{ee}}{p_{ee} N_e} \sum_{j' \in \{\text{presyn E cells}\}} G_{j'}(t) \quad (8)$$

$$g_{ei}(t) = \frac{\gamma_{ei}}{p_{ei} N_i} \sum_{k' \in \{\text{presyn I cells}\}} G_{k'}(t) \quad (9)$$

$$\tau_d \frac{dG_j}{dt} = -G_j + A_j \quad (10)$$

$$\tau_r \frac{dA_j}{dt} = -A_j + \tau_r \alpha \sum_l \delta(t - t_l) \quad (11)$$

where the inhibitory and excitatory reversal potentials are  $\mathcal{E}_I$ , and  $\mathcal{E}_E$ , respectively, with  $\mathcal{E}_I < 0 < \mathcal{E}_E$ .  $\xi_j(t)$  are uncorrelated white noise processes,  $p_{xy}$  is the proportion of neuron type  $y$  (randomly chosen) that provides presynaptic input to neuron type  $x$  ( $x, y \in \{e, i\}$ ). The second line in the equations describes the refractory period at spike time  $t^*$ . When the neuron's voltage crosses threshold  $\theta_j$ , the neuron goes into a refractory period for  $\tau_{ref}$  where the voltage is undefined, after which we set the

neuron's voltage to 0. In the last equation,  $t_l$  denotes the spike times of the  $j$  th excitatory neuron. Now, for the mean-field analysis, we use  $q_{ji}$  to model the synaptic heterogeneity between the pre- and post-synaptic neurons by modulating the synaptic conductance for both the excitatory and inhibitory neurons.

We note here the numerical assumptions for the mean-field analysis:

1. finite size effects are negligible ( $N_{e/i} \gg 1$ )
2. the firing rate of presynaptic neurons is governed by a Poisson process
3. the population firing rate averaged over  $q$  and  $\tau_m$  is a good approximation to the average presynaptic input rate and
4. a single p.d.f. function is sufficient to describe the population behavior) (finite  $N$ )

Similarly, for the inhibitory neurons indexed by  $k \in \{1, 2, \dots, N_i\}$ , the equations are:

$$\tau_m \frac{dv_k}{dt} = -v_k - g_{ii}(t)(v_k - \mathcal{E}_I) - g_{ei}(t)(v_k - \mathcal{E}_E) + \sigma_I \eta_k(t) \quad (12)$$

$$v_k(t^*) \geq 1 \text{ (refractory period)} \Rightarrow v_j(t^* + \tau_{ref}) = 0 \quad (13)$$

$$\tau_n \frac{d\eta_k}{dt} = -\eta_k + \sqrt{\tau_n \xi_k(t)} \quad (14)$$

$$g_{ie}(t) = q_j \frac{\gamma_{ie}}{p_{ie} N_e} \sum_{k' \in \{\text{presyn I cells}\}} G_{k'}(t) \quad (15)$$

$$g_{ii}(t) = \frac{\gamma_{ii}}{p_{ii} N_i} \sum_{k' \in \{\text{presyn I cells}\}} G_{k'}(t) \quad (16)$$

$$\tau_d \frac{dG_k}{dt} = -G_k + A_k \quad (17)$$

$$\tau_r \frac{dA_k}{dt} = -A_k + \tau_r \alpha \sum_l \delta(t - t_l) \quad (18)$$

For details regarding the equations, please refer to the paper by Ly (2015). Since the recurrent coupled stochastic network is difficult to describe theoretically, we use population density methods, where the probability of a neuron being in a particular state is determined by an equation. The variables in the populations are determined using distribution functions. The two forms of heterogeneity introduce a large number of dimensions. For simplicity, one can track a family of probability density functions for each  $(q_j, \tau_j)$  pair for each neuron. The subsequent equations are a good approximation to the HRSNN network with the following assumptions: (i) finite size effects are negligible ( $N_{e/i} \gg 1$ ) (ii) the firing rate of presynaptic neurons is governed by a Poisson process (iii) the population firing rate averaged over  $q$  and  $\tau_m$  is a good approximation to the average presynaptic input rate (iv) a single p.d.f. function is sufficient to describe the population behavior, and the heterogeneity is driven by  $(q_j, \tau_m, j)$  For each pair of values  $(q_j, \tau_m, j)$ , the probability density function  $\rho$  is defined by:

$$\int_{\Omega} \rho(v_E, \mathbf{w}_E, v_I, \mathbf{w}_I, t) dv_E d\mathbf{w}_E dv_I d\mathbf{w}_I = \Pr((v_E(t), \mathbf{w}_E(t), v_I(t), \mathbf{w}_I(t)) \in \Omega) \quad (19)$$

where  $\mathbf{w}_X$  denotes the other states variables of the corresponding neuron type  $X \in \{E, I\}$ , consisting of conductance, colored noise:  $\mathbf{w}_X = (g_X, a_X, \eta_X)$ . The evolution of the p.d.f.'s is governed by a continuity equation and boundary conditions:

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot \mathbf{J} \quad (20)$$

$$\mathbf{J} := (J_{v_E}, J_{g_E}, J_{a_E}, J_{\eta_E}, J_{v_I}, J_{g_I}, J_{a_I}, J_{\eta_I}) \quad (21)$$

$$J_{v_E} := -\frac{1}{\tau_m} [v_E + q\gamma_{ei}g_I(v_E - \mathcal{E}_I) + q\gamma_{ee}g_E(v_E - \mathcal{E}_E) + \sigma_E\eta_E] \rho \quad (22)$$

$$J_{v_I} := -\frac{1}{\tau_m} [v_I + \gamma_{ii}g_I(v_I - \mathcal{E}_I) + \gamma_{ie}g_E(v_I - \mathcal{E}_E) + \sigma_I\eta_I] \rho \quad (23)$$

$$J_{g_X} := -\frac{1}{\tau_d} [g_X - a_X] \rho \quad (24)$$

$$J_{a_X} := -\frac{a_X}{\tau_r} + v_X(t) \int_{a_X - \alpha_X}^{a_X} \rho(\dots, a'_X, \dots) da'_X \quad (25)$$

$$J_{\eta_X} := -\frac{1}{\tau_n} \eta_X \rho + \frac{1}{\tau_n} \frac{\partial^2 \rho}{\partial \eta_X^2} \quad (26)$$

$$v_X(t) := \iiint \frac{1}{\tau_m} J_{v_X} d\mathbf{w}_X dq d\tau_m \quad (27)$$

$$J_{\mathbf{w}_X} | \partial \mathbf{w}_X = 0 \quad (28)$$

The definitions of  $g_{XY}$  in the LIF neuron equations defined above result in a total conductance of  $\gamma_{XY}g_Y$  on average.

We describe an insightful analytic reduction that captures how the range of excitatory firing rates changes in different regimes. We focus on only the excitatory neurons, which have fewer state variables if the inhibitory population is ignored or assumed to be known.

Let us denote the approximate excitatory firing rate(s)  $v_E$  as  $r$ . The deterministic firing rate of the equation

$$\tau_m \frac{dv_E}{dt} = -v_E - q\tilde{g}_I(v_E - \mathcal{E}_I) - q\tilde{g}_E(v_E - \mathcal{E}_E) + \tilde{\eta}_E \quad (29)$$

is given by

$$r_0(q, \tau_m; \tilde{\mathbf{w}}_E) = \begin{cases} 0, & \text{if } \frac{q(\tilde{g}_E\mathcal{E}_E + \tilde{g}_I\mathcal{E}_I) + \tilde{\eta}_E}{1 + q(\tilde{g}_E + \tilde{g}_I)} \leq \theta \\ \frac{1 + q(\tilde{g}_E + \tilde{g}_I)}{\tau_m(\tilde{g}_E^*\mathcal{E}_E + \tilde{g}_I\mathcal{E}_I) + \tilde{\eta}_E}, & \text{if } \frac{q(\tilde{g}_E\mathcal{E}_E + \tilde{g}_I\mathcal{E}_I) + \tilde{\eta}_E}{1 + q(\tilde{g}_E + \tilde{g}_I)} > \theta \end{cases} \quad (30)$$

We define:  $\tilde{g}_E := \gamma_{ee}g_E, \tilde{g}_I := \gamma_{ei}g_I, \tilde{\eta}_E := \sigma_E\eta_E$ . Finally, the given state variables are integrated against their marginal density to get:

$$r(q, \theta) = \mathbb{E} \left[ \frac{r_0}{1 + r_0\tau_{ref}} \right] = \int \frac{r_0}{1 + r_0\tau_{ref}} \tilde{\rho}(\tilde{g}_E, \tilde{g}_I, \tilde{\eta}_E) d\tilde{\mathbf{w}}_E \quad (31)$$

There is a slight abuse of notation because the auxiliary variables  $a_X$  effect the conductances but are not written in the previous equation; the emphasis is on how  $(\tilde{g}_E, \tilde{g}_I, \tilde{\eta}_E)$  directly effects  $r$ . Since the external noise is applied indiscriminately,  $\tilde{\eta}_E$  is independent of the other variables and the marginal density factors into:

$$\tilde{\rho}(\tilde{g}_E, \tilde{g}_I, \tilde{\eta}_E) = \tilde{\rho}(\tilde{g}_E, \tilde{g}_I) \frac{e^{-(\tilde{\eta}_E/\sigma_E)^2}}{\sigma_E\sqrt{\pi}} \quad (32)$$

However,  $\tilde{\rho}(\tilde{g}_E, \tilde{g}_I)$  is still not analytically tractable, leading us to rely on Monte Carlo simulations to numerically estimate  $\tilde{\rho}(\tilde{g}_E, \tilde{g}_I)$ .

It must be noted here that this is a reduction model for the HRSNN network with many simplifying assumptions. It is not a complete mean-field derivation of the HRSNN model with heterogeneous LIF neurons, and heterogeneous STDP dynamics is a fascinating research question but beyond the scope of this paper.

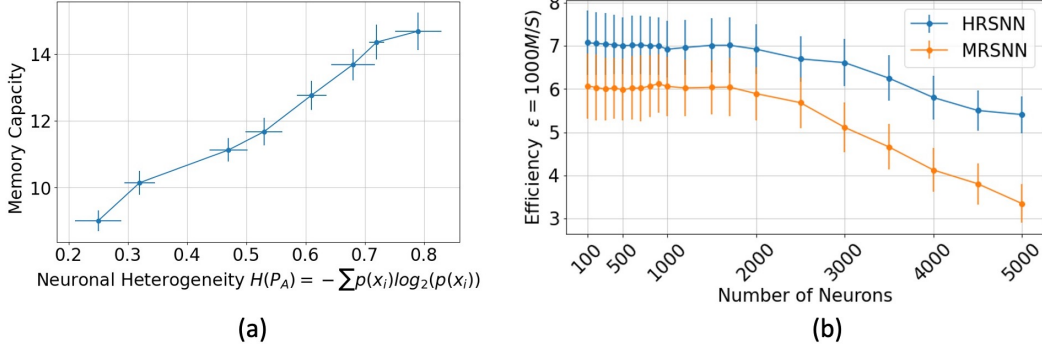


Figure 6: (a) Figure showing the variation of memory capacity with neuronal heterogeneity (b) Figure showing the variation of efficiency  $\mathcal{E}$  with number of neurons  $N_{\mathcal{R}}$

### B.3 ANALYTICAL RESULTS

**Analytical Results of Memory Capacity** Neuroscience networks of spiking neurons are increasingly used to understand mechanisms underlying phenomena observed in electrophysiological recordings. There are two complementary strategies for studying such a recurrent network of spiking neurons - (a) numerical simulations and (b) analytical methods using mean field models. With numerical simulations, we can simulate any network model without any approximation. However, this method typically works in high dimensional parameter space and is, thus, hard to interpret. Also, it is generally hard to characterize parameter regions where specific behaviors are found using numerical simulations. On the other hand, with analytical calculations, we obtain deeper insights into mechanisms underlying specific behaviors and can obtain critical parameters that control specific behaviors. So, now, we analytically study the variance of the estimated memory capacity with the change in the heterogeneity of neuronal parameters. We plot the change in the estimated memory capacity  $\mathcal{C}$ , calculated using Eq. ??, with respect to the neuronal heterogeneity  $\mathcal{H}$ , measured using the entropy of the neuronal parameters for the HRSNN model. The result is plotted in Fig. 6(a). We use a HRSNN model with  $N_{\mathcal{R}} = 1000$  and sequences of 4,000 random inputs chosen from  $\mathcal{U}[-1; 1]$ . We see that, as predicted, the memory capacity of the model increases linearly with the increase in heterogeneity within the limits of the application, as proved in Theorem 1. The error bars in Fig. 6(a) represent the standard deviation of the observations.

**Analytical Study of Spike Efficiency** We calculate the average firing rate of the heterogeneous spiking neural network for the prediction task during inference, and the results are shown in Fig 6(b). Using heterogeneity in the STDP parameters reduces the average number of spiking activations while keeping the memory capacity almost equal. This result shows that Heterogeneous STDP leads to sparse activation of neurons, as proved in Theorem 2.

**Comparison with Neuroscience Works:** We compare the analytical results obtained with some of the standard recurrent LIF network models in the literature. Brunel (2000) analytically study the dynamics of sparsely connected a network of sparsely connected excitatory and inhibitory integrate-and-fire neurons. The authors showed the existence of a diverse set of states, including synchronous states in which neurons fire regularly; asynchronous states with stationary global activity and very irregular individual cell activity; and states in which the global activity oscillates but individual cells fire irregularly, typically at rates lower than the global oscillation frequency. In this paper, we use heterogeneity in the LIF neurons. This leads to a diverse set of states for the neurons, which consequently helps orthogonalize the state space dynamics to increase the information stored in the memory of the network.

Denève & Machens (2016) discussed the inefficiency of irregular Poisson rate encoding in the brain. The authors argue that the Poisson point process, which we use to model the spike firing rate, is extremely inefficient as it exponentially increases the number of spikes required to convey information. The authors further discuss that in neuroscience, there exists a continuum between loosely balanced and tightly balanced spike-coding networks. Though loosely balanced networks are inefficient, they are cheap in terms of the number of connections per neuron and structure (Boerlin

et al., 2013; Boerlin & Denève, 2011; Bourdoukan et al., 2012). On the other hand, tightly-balanced spike-coding networks are highly efficient but extremely structured, dense connections that must constantly be maintained by STDP rules. For the HRSNN model, since we are engineering an artificial spiking neural network model, our network is highly structured and constantly updated using the heterogeneous STDP rules. Thus, we might say that the HRSNN model is a tightly-coupled network that helps in an efficient transfer of information. This hypothesis is supported by the results shown in Table ??, where the HRSNN model shows a higher performance using a lesser number of spikes.

#### B.4 MEMORY CAPACITY

Let  $x(t) \in U$  (where  $-\infty < t < +\infty$  and  $U \subset \mathbb{R}$  is a compact interval) be a single-channel stationary input signal. Assume that we have an RSNN, specified by its internal weight matrix  $\mathbf{W}$ , its input weight vector  $\mathbf{w}^{\text{in}}$  and the unit output functions  $\mathbf{f}, \mathbf{f}^{\text{out}}$ . The network receives  $x(t)$  at its input unit. For a given delay  $\tau$  and an output unit  $y_\tau$  with connection weight vector  $\mathbf{w}_\tau^{\text{out}}$  we consider the determination coefficient

$$\begin{aligned} d[\mathbf{w}_\tau^{\text{out}}](x(t-\tau), y_\tau(t)) &= \\ &= d\left(x(t-\tau), \mathbf{w}_\tau^{\text{out}} \begin{pmatrix} x(t) \\ \mathbf{r}(t) \end{pmatrix}\right) \\ &= \frac{\text{Cov}^2(x(t-\tau), y_\tau(t))}{\sigma^2(x(t))\sigma^2(y_\tau(t))} \end{aligned}$$

where Cov denotes covariance and  $\sigma^2$  variance. The  $\tau$ -delay Memory capacity of the network is defined by  $\mathcal{C}_\tau = \max_{\mathbf{w}_\tau^{\text{out}}} d[\mathbf{w}_\tau^{\text{out}}](x(t-\tau), y_\tau(t))$ . The Memory capacity of the network is  $\mathcal{C} = \sum_{\tau=1}^{\infty} \mathcal{C}_\tau$ .

The determination coefficient of two signals is the squared correlation coefficient. It ranges between 0 and 1 and represents the fraction of variance explainable in one signal by the other. Thus, the Memory capacity measures how much variance of the delayed input signal can be recovered from optimally trained output units, summed over all delays. Note that the output units do not interfere; arbitrarily, many output units  $y_\tau$  can be attached to the same network.

The performance of the heterogeneous network model derives from its ability to retain the memory of previous inputs. To quantify the relationship between the recurrent layer dynamics and the memory capacity, we note that the extraction of information from the recurrent layer is made through a linear combination of the neurons' states. Hence, more linearly independent neurons would offer more variable states and, thus, more extended memory.

For reservoir computing (RC), Jaeger (2002) shows that  $\mathcal{C}$  is bounded by the reservoir network size of the linear RC with the identity activation function and the independent and identically distributed (i.i.d.) model input. Memory capacity ( $\mathcal{C}$ ) is used to quantify the memory of RSNN. Such memory capacity measures the ability of RC to reconstruct precisely the past information of the model input. Also, the network's structural properties can greatly impact the  $\mathcal{C}$  of the linear RC. Now, the question arises what is the need to maximize the memory capacity of the network? The  $\mathcal{C}$  normally serves as a global index to quantify the memory property of the network. To comprehensively examine the memory property deeply, the local measurement of its memory property is indispensable. Thus, maximizing the  $\mathcal{C}$  acts as an estimator for better prediction results of the trained network.

Since the first-order approximation of the model is linear, the heterogeneity between state variables depends on all the eigenvalues of the adjacency matrix, with a larger mean eigenvalue meaning higher heterogeneity. Hence we can use the eigenvalues  $\{\lambda_i\}$  of the weight matrix  $W$  to quantify how fast the input decays in the recurrent layer approximately. In other words, the eigenvalues of  $W$  should be related to the memory capacity of the heterogeneous neural network model. Indeed, we find that the average eigenvalue modulus:  $\langle |\lambda| \rangle = 1/N_R \sum_{i=1}^{N_R} |\lambda_i|$  strongly correlates with  $\mathcal{H}$  and therefore with  $\mathcal{C}$  as well. Note that, as opposed to  $\mathcal{C}$  and  $\mathcal{H}$ ,  $\langle |\lambda| \rangle$  is much easier to compute and is solely determined by the recurrent layer network.

The memory capacity reflects the precision with which previous inputs can be recovered. The nonlinearity of the recurrent layer and other far-in-the-past inputs induce noise that complicates recovery. Thus, similar to the analysis done by Aceituno et al. (2020) for Echo state networks, the variance of the linear part of the recurrent layer is placed to maximize the recoverable information.

Thus, the inputs are projected into orthogonal directions of the recurrent layer state space to not add noise to each other. The variance spread across the different dimensions should be evenly distributed within those orthogonal directions, quantified by the neurons' covariance.

We start by noticing that the linear nature of the projection vector  $\mathbf{w}_{\text{out}}$  implies that we are treating the system as

$$\mathbf{r}(t) = \sum_{\tau=0}^{\infty} \mathbf{a}_{\tau} x(t - \tau) + \varepsilon(t) \quad (33)$$

where the vectors  $\mathbf{a}_{\tau} \in \mathbb{R}^{N_R}$  correspond to the linearly extractable effect of  $x(t - \tau)$  onto  $\mathbf{r}(t)$  and  $\varepsilon(t)$  is the nonlinear contribution of all the inputs onto the state of  $\mathbf{r}(t)$ .

Previous works have shown that linear recurrent layers have more extended memory, but nonlinearity is needed to perform interesting computations. Here we show that for a fixed ratio of the nonlinearity, greater heterogeneity leads to a lesser neuronal correlation, leading to a higher memory capacity.

To maintain this trade-off between linear and non-linear behavior, we will assume that linear and non-linear strengths distribution is fixed. This can be achieved if we impose that the probabilities of the neuron states do not change, meaning that the mean, variance, and other moments of the neuron outputs are unchanged; hence, the strength of the non-linear effects is unchanged. A first constraint can also be obtained from the maintained strength of the linear side of Eq.33

$$\text{Var} \left( \sum_{\tau=1}^{\infty} \mathbf{a}_{\tau} x(t - \tau) \right) = c \quad (34)$$

where  $c$  is a constant.

**Lemma 3.1.1:** *The state of the neuron can be written as follows:*

$$r_i(t) = \sum_{k=0}^{N_R} \sum_{n=1}^{N_R} \lambda_n^k \langle v_n^{-1}, \mathbf{w}^{\text{in}} \rangle (v_n)_i x(t - k) \quad (35)$$

where  $\mathbf{v}_n, \mathbf{v}_n^{-1} \in \mathbf{V}$  are, respectively, the left and right eigenvectors of  $\mathbf{W}$ , and  $\lambda_n^k \in \lambda$  belongs to the diagonal matrix containing the eigenvalues of  $\mathbf{W}$ ;  $\mathbf{a}_i = [a_{i,0}, a_{i,1}, \dots]$  represents the coefficients that the previous inputs  $\mathbf{x}_t = [x(t), x(t-1), \dots]$  have on  $r_i(t)$ .

**Proof:** We build on the work of Aceituno et al. (2020) where they showed that higher heterogeneity among the neuronal states implies higher memory capacity. Here we aim to show that as the number of neurons  $N_R$  in the recurrent layer decreases, heterogeneity increases the spectral radius. More formally, the spectral radius  $|\lambda_n|$  is directly proportional to  $\mathcal{H}$  as  $N_R$  decreases. We express the state of a neuron  $r_i(t)$  as

$$r_i(t) = \sum_{k=0}^{\infty} (W^k \mathbf{w}^{\text{in}})_i x(t - k) = \sum_{k=0}^{\infty} a_{i,k} x(t - k) = \langle \mathbf{a}_i, \mathbf{x}_t \rangle \quad (36)$$

where the vector  $\mathbf{a}_i = [a_{i,0}, a_{i,1}, \dots]$  represents the coefficients that the previous inputs  $\mathbf{x}_t = [x(t), x(t-1), \dots]$  have on  $r_i(t)$ . We can then plug this into the covariance between two neurons,

$$\begin{aligned} \text{Cov}(r_i, r_j) &= \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{q=t}^{t+T} \langle \mathbf{a}_i, \mathbf{x}_q \rangle \langle \mathbf{a}_j, \mathbf{x}_q \rangle \\ &= \langle \mathbf{a}_i, \mathbf{a}_j \rangle \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{q_i=0}^T \sum_{q_j=0}^T \langle \mathbf{x}_{q_i}, \mathbf{x}_{q_j} \rangle \\ &= \langle \mathbf{a}_i, \mathbf{a}_j \rangle \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{q=0}^T \langle \mathbf{x}_q, \mathbf{x}_q \rangle \\ &= \langle \mathbf{a}_i, \mathbf{a}_j \rangle \times \mathbb{E}[x^2(t)] \\ &= \langle \mathbf{a}_i, \mathbf{a}_j \rangle \end{aligned} \quad (37)$$

Now we write  $\mathbf{a}_i$  as a function of the eigenvalues of  $\mathbf{W}$ . Using the eigenvalue decomposition of the weight matrix  $\mathbf{W}$ , we rewrite the state of the neuron as follows:

$$r_i(t) = \sum_{k=0}^{N_R} \sum_{n=1}^{N_R} \lambda_n^k \langle \mathbf{v}_n^{-1}, \mathbf{w}^{\text{in}} \rangle (v_n)_i x(t-k) \quad (38)$$

where  $\mathbf{v}_n, \mathbf{v}_n^{-1} \in \mathbf{V}$  are, respectively, the left and right eigenvectors of  $\mathbf{W}$ , and  $\lambda_n^k \in \lambda$  belongs to the diagonal matrix containing the eigenvalues of  $\mathbf{W}$ ;  $\mathbf{a}_i = [a_{i,0}, a_{i,1}, \dots]$  represents the coefficients that the previous inputs  $\mathbf{x}_t = [x(t), x(t-1), \dots]$  have on  $r_i(t)$ . ■

**Theorem 1:** *If the memory capacity of the HRSNN and MRSNN networks are denoted by  $\mathcal{C}_H$  and  $\mathcal{C}_M$  respectively, then,  $\mathcal{C}_H \geq \mathcal{C}_M$ , where the heterogeneity in the neuronal parameters  $\mathcal{H}$  varies inversely to the correlation among the neuronal states measured as  $\sum_{n=1}^{N_R} \sum_{m=1}^{N_R} \text{Cov}^2(x_n(t), x_m(t))$  which in turn varies inversely with  $\mathcal{C}$ .*

**Proof:** As shown by Aceituno et al. (2020), the memory capacity increases when the variance along the projections of the input into the recurrent layer state has higher heterogeneity. This can be expressed in terms of the state space of the recurrent layer. Now, we aim to project the inputs into orthogonal directions of the network state space. Thus, we model the system as

$$\mathbf{r}(t) = \sum_{\tau=1}^{\infty} \mathbf{a}_{\tau} x(t-\tau) + \varepsilon(t) \quad (39)$$

where the vectors  $\mathbf{a}_{\tau} \in \mathbb{R}^N$  correspond to the linearly extractable effect of  $x(t-\tau)$  onto  $\mathbf{r}(t)$  and  $\varepsilon(t)$  is the nonlinear contribution of all the inputs onto the state of  $\mathbf{r}(t)$ .

Since our goal is to have a variance as homogeneous as possible along with the directions of  $\mathbf{a}_{\tau}$ , we need a variance that is as homogeneous along with orthogonal directions, where the vectors  $\mathbf{a}_{\tau} \in \mathbb{R}^N$  correspond to the linearly extractable effect of the input variable  $x(t)$  onto the states of the neurons ( $\mathbf{r}(t)$ ). Since the eigenvectors of  $\Sigma$  preserve orthogonality across the covariance matrix  $\Sigma$ , the new variances are given by the eigenvalues of the covariance matrix,  $\lambda_n(\Sigma)$ . Thus, we work on the distribution of the eigenvalues of the covariance matrix. Specifically, we want to show that increasing the heterogeneity in the neuronal membrane time constants decreases the correlation between the neuron states, which decreases the variance of the neuronal states of the eigenvalues, which would increase the memory capacity  $\mathcal{C}$ . We quantify the heterogeneity using the mean with respect to the square root of the raw variance of the eigenvalues of the covariance matrix given by

$$\mathcal{J} = \frac{\sum_{n=1}^{N_R} \lambda_n^2(\Sigma)}{\left(\sum_{n=1}^{N_R} \lambda_n(\Sigma)\right)^2} \quad (40)$$

where  $\lambda_n(\Sigma)$  is the  $n$ th eigenvalue of  $\Sigma$ . To get an intuition of how this metric reflects the heterogeneity in the neuronal parameters, consider the case of two eigenvalues  $\lambda_1, \lambda_2$ ; when  $\lambda_1 = \lambda_2$  - very homogeneous - then  $\mathcal{J} = \frac{1}{2}$ , but when  $\lambda_1 > 0, \lambda_2 = 0$  - heterogeneity is more and hence,  $\mathcal{J} = 1$ . The membrane time constant is given by the product of the membrane resistance  $R_m$  and membrane capacitance  $C_m$ , such that  $\tau_m = R_m C_m$ .  $R_m$  is the inverse of the permeability; the higher the permeability, the lower the resistance, and vice versa. Thus, the lower the time constant, the faster or more rapidly a membrane will respond to a stimulus. The effects of the time constant on propagation velocity will become clear below. Hence, variability in the membrane time constants will lead to variability in the propagation velocity of action potentials.

Now,

$$\left(\sum_{n=1}^{N_R} \lambda_n(\Sigma)\right)^2 = (\text{tr}[\Sigma])^2 = \left(\sum_{n=1}^{N_R} \text{Var}(r_n(t))\right)^2 \quad (41)$$

which is constant by the assumption that the probability distributions of the neuron activities are fixed. Hence we can focus on the value of  $\sum_{n=1}^{N_R} \lambda_n^2(\Sigma)$  which is true since

$$\Sigma^k \mathbf{e}_n(\Sigma) = \lambda_n(\Sigma) \Sigma^{k-1} \mathbf{e}_n(\Sigma) = \lambda_n^k(\Sigma) \mathbf{e}_n(\Sigma) \Rightarrow \sum_{n=1}^{N_R} \lambda_n^2(\Sigma) = \text{tr}[\Sigma^2] \quad (42)$$

where  $\mathbf{e}_n(\Sigma)$  and  $\lambda_n(\Sigma)$  are, resp. the  $n$ th eigenvector and eigenvalue of  $\Sigma$ . Hence, we can compute this by decomposing the square of the covariance matrix as follows:

$$\sum_{n=1}^{N_{\mathcal{R}}} \lambda_n^2(\Sigma) = \sum_{n=1}^{N_{\mathcal{R}}} \sum_{m=1}^{N_{\mathcal{R}}} \Sigma_{nm} \Sigma_{mn} = \sum_{n=1}^{N_{\mathcal{R}}} \sum_{m=1}^{N_{\mathcal{R}}} \text{Cov}^2(x_n(t), x_m(t)) \quad (43)$$

where  $\Sigma_{ij}$  are the factor matrices obtained using Cholesky decomposition of  $\Sigma$ . Thus,  $\sum_{n=1}^{N_{\mathcal{R}}} \lambda_n^2(\Sigma)$  increases as the neurons become more correlated; hence heterogeneity decreases.

Thus, from Eqs. 40, 43 we can write the heterogeneity as inversely proportional to  $\sum_{n=1}^{N_{\mathcal{R}}} \text{Cov}^2(x_n(t), x_m(t))$ . We see that increasing the correlations between neuronal states decreases the heterogeneity of the eigenvalues, which would reduce the memory capacity of the model. We show that the determinant of the covariance between neuronal parameters bounds the heterogeneity. Thus, as  $\mathcal{H}$  increases  $\rightarrow$  covariance decreases  $\rightarrow$  neurons become less correlated. Aceituno et al. (2020) proved that the neuronal state correlation is inversely related to the memory capacity of the network. Hence, we claim that as  $\mathcal{H}$  increases, the memory capacity  $\mathcal{C}$  also increases. Hence, for HRSNN, with  $\mathcal{H} > 0$ ,  $\mathcal{C}_H \geq \mathcal{C}_M$ . ■

## B.5 SPIKING EFFICIENCY

In this section, we model the spiking activity using a point process called the multivariate Point process model. A point process is a collection of random points on some underlying mathematical space, such as the real line, the Cartesian plane, or more abstract spaces.

The notion of using point process models, especially the interactive Hawkes processes, to model the spiking dynamics of LIF network dynamics has been studied in the literature previously (Löcherbach, 2017; Galves & Löcherbach, 2016; Mascart, 2021; Pfaffelhuber et al., 2022). We leverage these results to prove that heterogeneity in the synaptic dynamics can help reduce the spike count, as already discussed in the paper. We highlighted the key assumptions used in deriving the results in the Suppl. Sec. C. We apologize if there is still confusion, and we will add more in-depth discussion in the final manuscript as discussed below. In their paper, Löcherbach (2017) provide a survey of some aspects of the study of Hawkes processes in high dimensions to model biological neural systems and study their long-term behavior. Galves & Löcherbach (2016) provided an overview of point processes used as stochastic models for interacting neurons in discrete and continuous time. Similarly, Hawkes processes have met a recent interest in the mathematical neuroscience literature for their ability to model the dependence of a neuron’s activity in the network’s history (Mascart, 2021; Pfaffelhuber et al., 2022; Galves & Löcherbach, 2016; Gerhard et al., 2017; Zhou et al., 2020; Duval et al., 2022). Other works have also used a nonlinear interactive Hawkes process to model spiking neural networks with excitatory and inhibitory neurons (Chevallier et al., 2015; Chornoboy et al., 1988; Hansen et al., 2015; Reynaud-Bouret et al., 2014). Taking inspiration from these works, we use a microscopic model describing a large network of interacting neurons that can generate oscillations in a macroscopic frame. In the model, the activity of each neuron is represented by a point process indicating the successive times at which the neuron emits a spike, where each realization of this point process is the spike train. We take the spiking intensity of a neuron as the probability of emitting a spike during the next time instant, depending on the past history of the neuron and the activity of other neurons in the network. The neurons interact through their synapses. This means that a spike of a pre-synaptic neuron leads to an increase of the membrane potential of the post-synaptic neuron if the synapse is excitatory or to a decrease if the synapse is inhibitory, possibly after some delay, like the process of synaptic integration. When the membrane potential reaches a certain upper threshold, the neuron fires a spike. Thus, excitatory inputs from the neurons in the network increase the firing intensity, and inhibitory inputs decrease it. Hawkes processes provide good models of this synaptic integration phenomenon by the structure of their intensity processes. In this paper, we use a general class of mean-field interacting Hawkes processes, modeling the reciprocal interactions between a population of excitatory neurons and a population of inhibitory neurons.

Let us consider a subsection of the HRSNN network as shown in Fig. 7 denoted by  $N_x$ . We use the multivariate Point process model to create a probabilistic model that relates the inner structure of the sub-network and its spiking activity. In this model, each neuron  $i$  has a background spiking intensity  $\nu_i$  caused by neurons outside the network. We know that when a neuron spikes, it exerts an impact on its own spiking activity and the spiking activity of its output neurons. The impact of a neuron  $j$  on



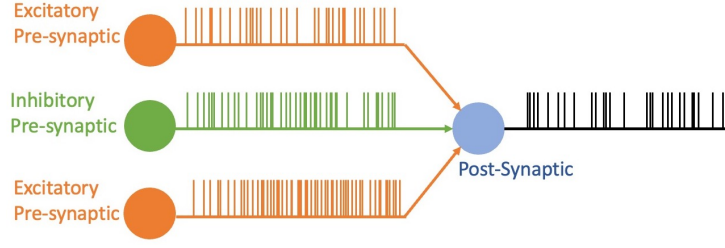


Figure 7: Figure showing the excitatory and inhibitory pre-synaptic neurons with excitatory and inhibitory spikes respectively incident on the post-synaptic neuron. We use this model to model the nonlinear interacting Hawkes process with inhibition.

neuron  $i$  is modeled by a real function  $h_{j \rightarrow i}(t)$ . This impact can be excitatory or inhibitory depending on whether the pre-synaptic neuron is excitatory or inhibitory, as shown in Fig. 7. While the spikes from excitatory neurons try to excite another spike, spikes originating from inhibitory neurons try to inhibit the spiking of the cascading neuron.

A Hawkes process is a point process in which each point is commonly associated with event occurrences in time, where every event time impacts the probability that other events will take place subsequently. These processes are characterized by the conditional intensity function, seen as an instantaneous measure of the probability of event occurrences. A Hawkes process is a point process in which each point is commonly associated with event occurrences. In this past-dependent model, every event time impacts the probability that other events take place subsequently. These processes are characterized by the conditional intensity function, seen as an instantaneous measure of the probability of event occurrences. Although the self-exciting Hawkes process remains widely studied, there has been a growing interest in modeling the opposite effect, known as inhibition, in which the probability of observing an event is lowered by the apparition of certain events. In practice, this amounts to considering negative kernel functions. To maintain the positivity of the intensity function, a non-linear operator is added to the expression, which in turn entails the loss of the cluster representation. This model is known as the non-linear Hawkes process, where the existence of such processes was proved via construction using bi-dimensional marked Poisson processes. The general Hawkes framework can be written as:

$$\lambda_t^i = \Phi_i \left( \sum_{j \in \mathcal{S}_{i,E}} \int_0^t h_{j \rightarrow i}(t-u) dZ_u^j \right), \quad (44)$$

where  $\lambda_t^i$  is the intensity of neuron  $i$ ,  $\Phi_i$  a positive function,  $Z_{j,t}$  is the counting process associated with neuron  $j$ ,  $h_{j \rightarrow i}(t)$  is the synaptic kernel associated with the synapse between neurons  $j$  and  $i$ .

To simplify the notation, we can rewrite Eq. 44 as

$$\lambda_i(t) = \Phi_i \left( \sum_{k \in I} \int_{(0,t)} h_{ki}(t-s) dZ_k(s) \right). \quad (45)$$

where  $h_{ik}(t-s)$  measures the influence of neuron  $k$  on neuron  $i$  and how this influence vanishes with the time. More precisely,  $h_{ik}(t-s)$  describes how a spike of neuron  $k$  lying back  $t-s$  time units in the past influences the present spiking rate at time  $t$ .

The goal of using heterogeneity in the STDP dynamics is to get better orthogonalization among the recurrent network states to lower higher-order correlations in spike trains. Studies have shown that the correlation of higher order progressively decreases the information available through neural population (Montani et al., 2009; Abbott & Dayan, 1999). Since we are trying to engineer a spike-efficient model, we leverage the heterogeneity in the STDP dynamics to reduce the higher-order correlations. The hypothesis is that using heterogeneity in STDP helps us orthogonalize the recurrent layer that can help us achieve an *efficient representation* of the input spike patterns with fewer spikes. This may be interpreted as the recurrent layer acting as an orthogonal bases function where inputs are projected onto these bases. Thus, having orthogonal bases can efficiently map inputs without much loss. While heterogeneous LIF neurons help us increase the number of principal components, thereby

enabling us to store a greater subclass of features, heterogeneous STDP helps us efficiently encode this orthogonalization of the recurrent layer, resulting in fewer spikes compared to a homogeneous RSNN. Thus, in effect, heterogeneous STDP parameters can learn the output more precisely, which is projected back into the recurrent network. One of the primary reasons why heterogeneous STDP helps project the input to orthogonal activations of the recurrent network can be attributed to the distribution of LTD dynamics, as this increases the competition and helps in distributing the projection of the inputs to multiple principal components. We discuss that the heterogeneous LTP/LTD dynamics in STDP lead to fewer spikes in the transmission of information.

**Lemma 3.2.1:** *If the neuronal firing rate of the HRSNN network with only heterogeneity in LTP/LTD dynamics of STDP is represented as  $\Phi_R$  and that of MRSNN represented as  $\Phi_M$ , then the HRSNN model promotes sparsity in the neural firing which can be represented as  $\Phi_R < \Phi_M$ .*

**Proof:** In this lemma, we show that the average firing rate of the model with heterogeneous STDP (LTP/LTD) dynamics (averaged over the population of neurons) is lesser than the corresponding average neuronal activation rate for a model with homogeneous STDP dynamics. We prove this by taking a sub-network of the HRSNN model as illustrated by Fig. 7. Now, we model the input spike trains of the pre-synaptic neurons using a multivariate interactive, nonlinear Hawkes process with multiplicative inhibition (Duval et al., 2022).

We consider a population of neurons of size  $N$  that is divided into population  $A$  (excitatory) with size  $N_A := \alpha N$  and a population  $B$  (inhibitory) with size  $N_B = (1 - \alpha)N$ . A particular instance of the model is then given in terms of a family of counting processes  $(Z_t^1, \dots, Z_t^{N_A})$  (population  $A$ ) and  $(Z_t^{N_A+1}, \dots, Z_t^N)$  (population  $B$ ) with coupled conditional stochastic intensities given respectively by  $\lambda^A$  and  $\lambda^B$ . Consider on a filtered probability space  $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t \geq 0}, \mathbf{P})$  an independent family of i.i.d. Poisson measures  $(\pi_i(ds, dz), i \in \{1, \dots, N\})$  with intensity measure  $ds \times dz$  on  $[0, \infty) \times [0, \infty)$ . Let  $(x, y) \mapsto F(x, y)$  and  $(x, y) \mapsto G(x, y)$  two nonnegative functions defined on  $(0, \infty)^2$ . We assume that  $F$  and  $G$  satisfy

$$F(x, y) = \Phi_A(x)\Phi_{B \rightarrow A}(y), G(x, y) = \Phi_B(x) + \Phi_{A \rightarrow B}(y),$$

where  $\Phi_A, \Phi_{B \rightarrow A}, \Phi_B$  and  $\Phi_{A \rightarrow B}$  are nonnegative functions, each of them globally Lipschitz with  $\Phi_{B \rightarrow A}$  bounded (and with no loss of generality we assume  $0 \leq \Phi_{B \rightarrow A} \leq 1$ ).

Let us consider the family of càdlàg  $(\mathcal{F}_t)_{t \geq 0}$  point processes  $(Z_t^i)_{t \geq 0, i=1, \dots, N}$  given by

$$Z_t^i = \int_0^t \int_0^\infty \mathbf{1}_{z \leq \lambda_s^i} \pi_i(ds, dz), i = 1, \dots, N,$$

where the intensity  $\lambda^i, i = 1, \dots, N$ , is given as:

$$\lambda_t^{A,N} := \Phi_A \left( \frac{1}{N} \sum_{j \in A} \int_0^{t^-} h_1(t-u) dZ_u^j \right) \Phi_{B \rightarrow A} \left( \frac{1}{N} \sum_{j \in B} \int_0^{t^-} h_2(t-u) dZ_u^j \right) \quad (46)$$

$$\lambda_t^{B,N} := \Phi_B \left( \frac{1}{N} \sum_{j \in B} \int_0^{t^-} h_3(t-u) dZ_u^j \right) + \Phi_{A \rightarrow B} \left( \frac{1}{N} \sum_{j \in A} \int_0^{t^-} h_4(t-u) dZ_u^j \right) \quad (47)$$

, where  $A \& B$  are the populations of the excitatory and inhibitory neurons, respectively.

The dynamics given by Eq. 47 is of Hawkes type: each particle's intensity depends on the whole system's history, through memory kernels  $h_i, i = 1, \dots, 4$  and firing rate functions  $\Phi_A$  and  $\Phi_B$ . The multiplicative influence of inhibitory population  $B$  onto population  $A$ , is represented using the inhibition kernel  $\Phi_{B \rightarrow A}$  which is a decreasing nonnegative function on  $[0, +\infty)$ , with  $\Phi_{B \rightarrow A}(0) = 1$  and  $\Phi_{B \rightarrow A}(x) \xrightarrow{x \rightarrow \infty} 0$  i.e., activity of population  $A$  should decrease as activity of population  $B$  rises. The model secondly incorporates retroaction from population  $A$  onto population  $B$ , which is supposed to be mostly additive, although possibly modulated by a nonlinear feedback kernel  $\Phi_{A \rightarrow B}$ .

Now, without loss of generality we assume that  $\Phi_A$  and  $\Phi_B$  are linear - i.e.,  $\Phi_A(x) = \mu_A + x, \Phi_B(x) = \mu_B + x, x \geq 0$ , where  $\mu_A, \mu_B \geq 0$ , and  $h_i \geq 0$  for  $i = 1, \dots, 4$ .

Hence, Eq. 47 becomes

$$\begin{cases} \lambda_t^A = \left( \mu_A + \alpha \int_0^t h_1(t-u) \lambda_u^A du \right) \Phi_{B \rightarrow A} \left( (1-\alpha) \int_0^t h_2(t-u) \lambda_u^B du \right), \\ \lambda_t^B = \mu_B + (1-\alpha) \int_0^t h_3(t-u) \lambda_u^B du + \Phi_{A \rightarrow B} \left( \alpha \int_0^t h_4(t-u) \lambda_u^A du \right). \end{cases} \quad (48)$$

For heterogeneous neuron populations, there exists an asymmetry of the weights. Based on balanced spiking neural networks with heterogeneous connection strengths, previous works have revealed that such heterogeneous networks possess heavy-tailed Lévy fluctuations (Shlesinger et al., 1987; Mantegna & Stanley, 1995; Cossell et al., 2015). The heterogeneous heavy-tailed distributions of synaptic weights have been fitted to lognormal distributions (Buzsáki & Mizuseki, 2014; Kuśmiercz et al., 2020). We model the inputs to neuron  $i \in E$  as:

$$l_i(t) = W_{EX} \tau_{mE} \sum_{j \in X} c_{ij} s_j(t) + W_{EE} \tau_{mE} \sum_{j \in E} c_{ij} s_j(t) - W_{EI} \tau_{mE} \sum_{j \in I} c_{ij} s_j(t) \quad (49)$$

$$= \mu_1 E + \Delta \mu_i + \eta_i(t) \quad (50)$$

where  $\mu$  denotes the mean inputs such that  $\mu_E = K \tau_{mE} (W_{EX} r_X + W_{EE} r_E - W_{EI} r_I)$ ;  $\Delta \mu_i$  = 'quenched' fluctuations (from neuron to neuron) with variance  $\langle \Delta \mu^2 \rangle_E = K \tau_{mE}^2 (W_{EX}^2 (r_X^2 + \Delta r_X^2) + W_{EE}^2 (r_E^2 + \Delta r_E^2) + W_{EI}^2 (r_I^2 + \Delta r_I^2))$  due to random connectivity. Finally,  $\eta_i$  denotes temporal fluctuations due to spiking activity. We assume that the pre-synaptic neurons fire as using the interactive Hawkes process described above.

Consider a case where  $\mu_A \gg 1, \mu_B = 0$  and  $h = 1_{[0, \theta]}$

$$\begin{cases} \lambda_t^A := \left( \mu_A + \alpha \int_0^t h_1(t-u) d\lambda_u^A \right) \Phi_{B \rightarrow A} \left( (1-\alpha) \int_0^t h_2(t-u) d\lambda_u^B \right), \\ \lambda_t^B := (1-\alpha) \int_0^t h_3(t-u) d\lambda_u^B + \alpha \int_0^t h_4(t-u) d\lambda_u^A. \end{cases} \quad (51)$$

In a normal case, the excitatory and inhibitory populations follow the following steps: (1)  $t \approx 0, \lambda_t^A \approx \mu_A$  is high and  $\lambda_t^B \approx 0$  is small (2) Feedback from  $A$  to  $B$ :  $\lambda_t^B$  increases (3) Inhibition of  $B$  to  $A$ : when  $\lambda_t^B$  gets high,  $\Phi_{B \rightarrow A}$  reduces  $\lambda_t^A$  (4)  $h_4$  has compact support: after a time  $\theta_4$ ,  $B$  no longer feels the influence of  $A$ : intensity of  $B$  is back to  $\mu_B \approx 0$  and  $A$  to its normal high activity  $\mu_A$  (State 1)

This leads to oscillations which lead to spikes. However, heterogeneity in the synaptic dynamics increases the stochasticity of the pre-synaptic spike arrival. Thus, due to the heterogeneity,  $\Phi_{B \rightarrow A}$  promotes the system in the inhibition state (state 3) and inhibits the system's movement to system 4 and system 1, thereby creating a spike. Hence,  $\Phi_R^A < \Phi_M^A$ . Similarly, for the inhibitory neurons, we can show that  $\Phi_R^B < \Phi_M^B$ . Thus, we get  $\Phi_R < \Phi_M$  ■

This lemma might be interpreted as the heterogeneous STDP dynamics increasing the synaptic noise, which reduces the number of spikes of the post-synaptic neuron. A heterogeneous STDP leads to a non-uniform scaling of correlated spike trains leading to de-correlation. Hence, we can say that heterogeneous STDP models have learned a better-orthogonalized subspace representation, leading to a better encoding of the input space with fewer spikes.

It is to be mentioned here that the synaptic noise might be thought of as analogous to the stochasticity in the gradient descent algorithm. As recently proved by Simsekli et al. (Simsekli et al., 2020; 2019), stochasticity plays an important role in the generalization ability of the model. We might interpret the synaptic noise in the heterogeneous STDP to play a similar role and helps in better generalizability of the HRSNN model. This hypothesis is empirically proven in Supplementary Section A. However, a detailed theoretical analysis would be a very interesting direction for future work.

**Theorem 2:** For a given number of neurons  $N_R$ , the spike efficiency of the model  $\mathcal{E} = \frac{\mathcal{C}(N_R)}{S}$  for HRSNN ( $\mathcal{E}_R$ ) is greater than MRSNN ( $\mathcal{E}_M$ ) i.e.,  $\mathcal{E}_R \geq \mathcal{E}_M$

**Proof:** To study the effect of the spike time when the weight  $w_k$  changes, we look into the expected value of the time difference in the post-synaptic spikes, which is given as:

$$\mathbb{E}[\Delta t_{\text{post}}] = \mathbb{E}[t_{\text{post}} - t_{\text{post}}] = (\mathbb{E}[t_{\text{post}}] - t_{\text{post}}) \Pr[s] \quad (52)$$

where  $\Pr[\exists s]$  is the probability of occurrence of the post-synaptic spike. Thus, the expected input to the neuron at time  $t(\mathbb{E}[i(t)])$ , which comprises of its excitatory and inhibitory components

$\mathbb{E}[i_e(t)], \mathbb{E}[i_i(t)]$  can be expressed as:

$$\mathbb{E}[\Delta i(t)] = \Delta \mathbb{E}[i_e(t)] - \Delta \mathbb{E}[i_i(t)] \quad \text{for } t < t_{\text{post}} \quad (53)$$

$$\text{where } \mathbb{E}[i_e(t)] = \rho_e \int_0^\infty \mu_{w_e}(w, t) dw \quad ; \quad \mathbb{E}[i_i(t)] = \rho_i \int_0^\infty \mu_{w_i}(w, t) dw \quad (54)$$

where  $\rho_e, \rho_i$  are the rates of incoming spikes and  $\mu_{w_e}(w, t), \mu_{w_i}(w, t)$  the probabilities of the weights associated to time  $t$ . Now, considering the case for RSNNs with homogeneous STDP ( $M$ ) and with heterogeneous STDP ( $R$ ), the difference in the variances of the two populations is given as:

$$\Delta \text{Var}[V_M] - \Delta \text{Var}[V_R] = \Delta \int_{-\infty}^t [\mathbb{E}[i_M^2(t)] - (\mathbb{E}[i_R^2(t)] - \mathbb{E}[i_R(t)]^2)] dt \quad (55)$$

Since  $t < t_{\text{post}}$ , STDP potentiates both inhibitory and excitatory synapses, so  $\Delta \mathbb{E}[i_i^2(t)] > 0, \Delta \mathbb{E}[i_e^2(t)] > 0$ . The term  $\mathbb{E}[i_M(t)]^2 = 0$  by the symmetry of the weights, and it is maintained at zero by the symmetry of the STDP. But for heterogeneous neuron populations, as described above, there exists an asymmetry of the weights. Based on balanced spiking neural networks with heterogeneous connection strengths, previous works have revealed that such heterogeneous networks possess heavy-tailed, Lévy fluctuations (Shlesinger et al., 1987; Mantegna & Stanley, 1995; Cossell et al., 2015). This implies  $\mathbb{E}[i_R(t)]^2 > 0 \Rightarrow \Delta \text{Var}[V_R] < \Delta \text{Var}[v(t)_M]$ . We calculate the number of post-synaptic spikes triggered when the stimulus is present. Now, representing the spike rate of the HRSNN and the MRSNN as  $\Phi_R, \Phi_M$  resp.,

$$\int_0^t \Phi_R(t) dt \leq \int_0^t \Phi_M(t) dt \Rightarrow S_R = N_{\mathcal{R}} \frac{T}{\hat{t}_{ISI}^R} \leq N_{\mathcal{R}} \frac{T}{\hat{t}_{ISI}^M} = S_M \quad (56)$$

Thus, spikes decrease when we use heterogeneity in the LTP/LTD Dynamics. Hence, we compare the efficiencies of the HRSNN with that of MRSNN as follows:

$$\frac{\mathcal{E}_R}{\mathcal{E}_M} = \frac{M_R(N_{\mathcal{R}}) \times S_M}{S_R \times M_M(N_{\mathcal{R}})} = \frac{\sum_{\tau=1}^{N_{\mathcal{R}}} \frac{\text{Cov}^2(x(t-\tau), \mathbf{a}_\tau^R \mathbf{r}_R(t))}{\text{Var}(\mathbf{a}_\tau^R \mathbf{r}_R(t))} \times \int_{t_{ref}}^\infty t \Phi_R dt}{\sum_{\tau=1}^{N_{\mathcal{R}}} \frac{\text{Cov}^2(x(t-\tau), \mathbf{a}_\tau^M \mathbf{r}_M(t))}{\text{Var}(\mathbf{a}_\tau^M \mathbf{r}_M(t))} \times \int_{t_{ref}}^\infty t \Phi_M dt} \quad (57)$$

Since  $S_R \leq S_M$  and also, the covariance increases when the neurons become correlated, and as neuronal correlation decreases,  $\mathcal{H}$  increases (Theorem 1), we see that  $\frac{\mathcal{E}_R}{\mathcal{E}_M} \geq 1 \Rightarrow \mathcal{E}_R \geq \mathcal{E}_M$  ■

## C SUPPLEMENTARY SECTION C

### C.1 HIGHER ORDER CORRELATION

In this paper, we took inspiration from results in reservoir computing, which show that we can maximize memory capacity using orthogonalization among reservoir states in the case of reservoir computers (Farkas & Gergel', 2017; Farkas et al., 2016). The goal of using heterogeneous STDP dynamics is to get better orthogonalized recurrent network states to achieve more efficient information transfer with lower higher-order correlations in spike trains. Recent studies (Montani et al., 2009; Abbott & Dayan, 1999) have shown that the correlation of higher order progressively decreases the information available through the neural population. The decrease in information becomes larger as the interaction order grows. Since we are trying to engineer a spike-efficient model, we leverage the heterogeneity in neuronal parameters to reduce the higher-order correlations. The hypothesis is that an orthogonal recurrent layer can help us efficiently represent the input spike patterns with fewer spikes. This may be interpreted as the recurrent layer acting as an orthogonal bases function where the inputs are projected onto these bases. Thus, having orthogonal bases can efficiently map the inputs without much loss. The heterogeneous STDP helps us efficiently achieve this orthogonalization of the recurrent layer, resulting in a lesser voltage variance across the neuron population. This leads to fewer spikes (since the mean is constant) compared to a homogeneous RSNN. Thus, in effect, heterogeneous STDP parameters can learn the output more precisely, which is projected back into the recurrent network. Hence, using heterogeneous STDP parameters leads to a better orthogonalization among the neuronal states and hence, a higher  $\mathcal{C}$ .

In this paper, we show that using a distribution of LTP/LTD dynamics in the STDP parameters helps us in mappings the input onto the orthogonal activations of the recurrent network to capture the

principal components of the input signal. The LTD dynamics play an important role in determining the orthogonality of neuronal activations. LTD windows of the STDP rules enable robust sequence learning amid background noise in cooperation with a large signal transmission delay between neurons and a theta rhythm (Hayashi & Igarashi, 2009). The LTD window in the range of positive spike-timing plays an important role in preventing noise influences with sequence learning. Oja (Oja, 1982; 1989) showed that the LIF neuron’s time constant is very fast compared to the time constant of learning in which the weights  $w_{ji}$  change. The learning is assumed to take place according to the STDP type conjunction of the inputs  $\xi_i$  and the integrated effect of the inputs,  $\nu_j$ , with an additional forgetting term attributed to the LTD dynamics:  $\frac{dw_{ji}}{dt} = \alpha \nu_j \xi_i - f(\nu_j, \xi_i, w_{ji})$ . In the case of homogeneous STDP,  $f(\cdot)$  is a constant; hence, the model can only efficiently learn the first principal component of the input. However, quite interesting functions emerge when considering STDP to have a distribution. This also helps us determine the next principal components other than the first one. Hence the diversity in the different LTD dynamics increases the competition and helps that not all inputs are mapped to the first principle component. Thus, the diversity in the LTD dynamics helps in projecting the input to orthogonal activations of the recurrent network.

Table 9: Table Showing the estimated highest order of correlation for HRSNN vs. MRSNN using CuBIC

	$\hat{\xi}$	p-value	Estimated Highest-order of correlation ( $\hat{\xi} + 1$ )
HRSNN	2	0.423	3
MRSNN	5	0.358	6

Now, for homogeneous RSNNs, several higher-order correlations, which according to our hypothesis, arise because of the poor orthogonalization among the network states. This results in the redundancies of spikes for encoding the same information. In this paper, we use heterogeneous STDP dynamics to learn an efficient orthogonal representation of the state space, which result in the network learning the same patterns but using fewer spikes. (theorem: 3) We also show that heterogeneity in the neuronal parameters decreases the neuronal correlation (theorem 1 And fig 2a). Thus, since heterogeneity results in better orthogonalization among the neuronal states, it results in fewer higher-order correlations. Moreover, recent studies have shown that the correlation of higher order progressively decreases the information available through the neural population, and the decrease in information becomes larger as the interaction order grows. Since we are trying to engineer an efficient model, we aim to reduce the higher-order correlations using heterogeneity in neuronal parameters (as shown in Theorem 1). In addition to this, to verify this, we used CuBIC (Staude et al., 2010), a cumulant-based inference of higher-order correlations in massively parallel spike trains. The details of the experimental methodology are given in Supplementary Section C. The outcome of CuBIC is a lower bound  $\hat{\xi}$  on the order of correlation in the spiking activity of large groups of simultaneously recorded neurons. CuBIC can provide statistical evidence for large correlated groups without the discouraging requirements on a sample size that direct tests for higher-order correlations have to meet. This is achieved by exploiting constraining relations among correlations of different orders. However, it must be noted that CuBIC is not designed to estimate the order of correlation directly; the inferred lower bound might not always correspond to the maximal order of correlation present in a given data set.

## REFERENCES

Larry F Abbott and Peter Dayan. The effect of correlated variability on the accuracy of a population code. *Neural computation*, 11(1):91–101, 1999.

2*	SHD			SSC			CIFAR10 DVS		
	Training Accuracy (A)	Testing Accuracy (B)	Generalization Error  A - B	Training Accuracy (A)	Testing Accuracy (B)	Generalization Error  A - B	Training Accuracy (A)	Testing Accuracy (B)	Generalization Error  A - B
Hom LIF Hom STDP	86.92 ± 1.35	72.89 ± 1.85	14.03 ± 1.67	74.69 ± 1.72	47.94 ± 1.94	26.75 ± 1.42	82.41 ± 1.8	65.33 ± 3.41	17.08 ± 1.35
Hom LIF Het STDP	85.76 ± 1.27	73.91 ± 1.49	11.85 ± 1.25	76.79 ± 1.58	52.96 ± 1.73	23.86 ± 1.29	83.48 ± 1.52	67.06 ± 2.97	16.42 ± 1.24
Het LIF Hom STDP	95.29 ± 1.16	78.36 ± 1.42	16.93 ± 1.13	84.26 ± 1.33	55.11 ± 1.65	29.15 ± 1.12	86.93 ± 1.79	68.37 ± 3.05	18.56 ± 1.42
Het LIF Het STDP	94.07 ± 1.03	80.01 ± 1.13	14.06 ± 1.02	86.41 ± 1.49	59.28 ± 1.35	27.13 ± 0.97	87.49 ± 1.76	70.54 ± 1.82	16.95 ± 1.38

- 
- Pau Vilimelis Aceituno, Gang Yan, and Yang-Yu Liu. Tailoring echo state networks for optimal learning. *iscience*, 23(9):101440, 2020.
- Robert M Blumenthal and Ronald K Getoor. Some theorems on stable processes. *Transactions of the American Mathematical Society*, 95(2):263–273, 1960.
- Martin Boerlin and Sophie Denève. Spike-based population coding and working memory. *PLoS computational biology*, 7(2):e1001080, 2011.
- Martin Boerlin, Christian K Machens, and Sophie Denève. Predictive coding of dynamical variables in balanced spiking networks. *PLoS computational biology*, 9(11):e1003258, 2013.
- Ralph Bourdoukan, David Barrett, Sophie Deneve, and Christian K Machens. Learning optimal spike-based representations. *Advances in neural information processing systems*, 25, 2012.
- Nicolas Brunel. Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of computational neuroscience*, 8(3):183–208, 2000.
- György Buzsáki and Kenji Mizuseki. The log-dynamic brain: how skewed distributions affect network operations. *Nature Reviews Neuroscience*, 15(4):264–278, 2014.
- Biswadeep Chakraborty and Saibal Mukhopadhyay. Characterization of generalizability of spike time dependent plasticity trained spiking neural networks. *arXiv preprint arXiv:2105.14677*, 2021.
- Ashesh Chattopadhyay, Pedram Hassanzadeh, and Devika Subramanian. Data-driven predictions of a multiscale lorenz 96 chaotic system using machine-learning methods: reservoir computing, artificial neural network, and long short-term memory network. *Nonlinear Processes in Geophysics*, 27(3):373–389, 2020.
- Julien Chevallier, María José Cáceres, Marie Doumic, and Patricia Reynaud-Bouret. Microscopic approach of a time elapsed neural model. *Mathematical Models and Methods in Applied Sciences*, 25(14):2669–2719, 2015.
- ES Chornoboy, LP Schramm, and AF Karr. Maximum likelihood identification of neural point process systems. *Biological cybernetics*, 59(4-5):265–275, 1988.
- Lee Cossell, Maria Florencia Iacaruso, Dylan R Muir, Rachael Houlton, Elie N Sader, Ho Ko, Sonja B Hofer, and Thomas D Mrsic-Flogel. Functional organization of excitatory synaptic strength in primary visual cortex. *Nature*, 518(7539):399–403, 2015.
- Benjamin Cramer, Yannik Stradmann, Johannes Schemmel, and Friedemann Zenke. The heidelberg spiking data sets for the systematic evaluation of spiking neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
- Sophie Denève and Christian K Machens. Efficient codes and balanced networks. *Nature neuroscience*, 19(3):375–382, 2016.
- Céline Duval, Eric Luçon, and Christophe Pouzat. Interacting hawkes processes with multiplicative inhibition. *Stochastic Processes and their Applications*, 148:180–226, 2022.
- Igor Farkaš and Peter Gergel’. Maximizing memory capacity of echo state networks with orthogonalized reservoirs. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2437–2442. IEEE, 2017.
- Igor Farkaš, Radomír Bosák, and Peter Gergel’. Computational analysis of memory capacity in echo state networks. *Neural Networks*, 83:109–120, 2016.
- Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trounev, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2681–2690. PMLR, 2019.
- Peter I Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

- 
- Antonio Galves and Eva Löcherbach. Modeling networks of spiking neurons as interacting processes with memory of variable length. *Journal de la Société Française de Statistique*, 157(1):17–32, 2016.
- Felipe Gerhard, Moritz Deger, and Wilson Truccolo. On the stability and dynamics of stochastic spiking neuron models: Nonlinear hawkes process and point process glms. *PLoS computational biology*, 13(2):e1005390, 2017.
- Niels Richard Hansen, Patricia Reynaud-Bouret, and Vincent Rivoirard. Lasso and probabilistic inequalities for multivariate point processes. 2015.
- Hatsuo Hayashi and Jun Igarashi. Ltd windows of the stdp learning rule and synaptic connections having a large transmission delay enable robust sequence learning amid background noise. *Cognitive neurodynamics*, 3(2):119–130, 2009.
- Herbert Jaeger. Short term memory in echo state networks. gmd-report 152. In *GMD-German National Research Institute for Computer Science (2002)*, <http://www.faculty.jacobs-university.de/hjaeger/pubs/STMEchoStatesTechRep.pdf>. Citeseer, 2002.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Łukasz Kuśmiercz, Shun Ogawa, and Taro Toyozumi. Edge of chaos and avalanches in neural networks with heavy-tailed synaptic weight distribution. *Physical Review Letters*, 125(2):028101, 2020.
- Eva Löcherbach. Spiking neurons: interacting hawkes processes, mean field limits and oscillations. *ESAIM: Proceedings and Surveys*, 60:90–103, 2017.
- Edward N Lorenz. Deterministic nonperiodic flow. *Journal of atmospheric sciences*, 20(2):130–141, 1963.
- Edward N Lorenz. Predictability: A problem partly solved. In *Proc. Seminar on predictability*, volume 1, 1996.
- Cheng Ly. Firing rate dynamics in recurrent spiking neural networks with intrinsic and network heterogeneity. *Journal of computational neuroscience*, 39(3):311–327, 2015.
- Rosario N Mantegna and H Eugene Stanley. Scaling behaviour in the dynamics of an economic index. *Nature*, 376(6535):46–49, 1995.
- Cyrille Mascart. *Efficient simulation of point processes with applications to neurosciences*. PhD thesis, Université Côte d’Azur, Nice, France, 2021.
- Sam McKenzie, Roman Huszár, Daniel F English, Kanghwan Kim, Fletcher Christensen, Euisik Yoon, and György Buzsáki. Preexisting hippocampal network dynamics constrain optogenetically induced place fields. *Neuron*, 109(6):1040–1054, 2021.
- Fernando Montani, Robin AA Ince, Riccardo Senatore, Ehsan Arabzadeh, Mathew E Diamond, and Stefano Panzeri. The impact of high-order interactions on the rate of synchronous discharge and information transmission in somatosensory cortex. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1901):3297–3310, 2009.
- Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3):267–273, 1982.
- Erkki Oja. Neural networks, principal components, and subspaces. *International journal of neural systems*, 1(01):61–68, 1989.
- Nicolas Perez-Nieves, Vincent CH Leung, Pier Luigi Dragotti, and Dan FM Goodman. Neural heterogeneity promotes robust learning. *Nature communications*, 12(1):1–9, 2021.
- Balint Petro, Nikola Kasabov, and Rita M Kiss. Selection and optimization of temporal spike encoding methods for spiking neural networks. *IEEE transactions on neural networks and learning systems*, 31(2):358–370, 2019.

- 
- Peter Pfaffelhuber, Stefan Rotter, and Jakob Stiefel. Mean-field limits for non-linear hawkes processes with excitation and inhibition. *Stochastic Processes and their Applications*, 153:57–78, 2022.
- Patricia Reynaud-Bouret, Vincent Rivoirard, Franck Grammont, and Christine Tuleau-Malot. Goodness-of-fit tests and nonparametric adaptive estimation for spike train analysis. *The Journal of Mathematical Neuroscience*, 4:1–41, 2014.
- Michael F Shlesinger, BJ West, and Joseph Klafter. Lévy dynamics of enhanced diffusion: Application to turbulence. *Physical Review Letters*, 58(11):1100, 1987.
- Umut Simsekli, Levent Sagun, and Mert Gurbuzbalaban. A tail-index analysis of stochastic gradient noise in deep neural networks. *arXiv preprint arXiv:1901.06053*, 2019.
- Umut Simsekli, Ozan Sener, George Deligiannidis, and Murat A Erdogdu. Hausdorff dimension, heavy tails, and generalization in neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.
- Benjamin Staude, Stefan Rotter, and Sonja Grün. Cubic: cumulant based inference of higher-order correlations in massively parallel spike trains. *Journal of computational neuroscience*, 29(1): 327–350, 2010.
- Tobias Thornes, Peter Düben, and Tim Palmer. On the use of scale-dependent precision in earth system modelling. *Quarterly Journal of the Royal Meteorological Society*, 143(703):897–908, 2017.
- Yan Zhou, Yaochu Jin, and Jinliang Ding. Surrogate-assisted evolutionary search of spiking neural architectures in liquid state machines. *Neurocomputing*, 406:12–23, 2020.