



**HAL**  
open science

# CHANI: Correlation-based Hawkes Aggregation of Neurons with bio-Inspiration

Sophie Jaffard, Samuel Vaiter, Patricia Reynaud-Bouret

► **To cite this version:**

Sophie Jaffard, Samuel Vaiter, Patricia Reynaud-Bouret. CHANI: Correlation-based Hawkes Aggregation of Neurons with bio-Inspiration. 2024. hal-04589554

**HAL Id: hal-04589554**

**<https://hal.science/hal-04589554v1>**

Preprint submitted on 28 May 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# CHANI: Correlation-based Hawkes Aggregation of Neurons with bio-Inspiration

**Sophie Jaffard**

*Laboratoire J. A. Dieudonné (LJAD)  
Université Côte d'Azur  
Nice, France*

SOPHIE.JAFFARD@UNIV-COTEDAZUR.FR

**Samuel Vaïter**

*Laboratoire J. A. Dieudonné (LJAD)  
CNRS, Université Côte d'Azur  
Nice, France*

SAMUEL.VAITER@UNIV-COTEDAZUR.FR

**Patricia Reynaud-Bouret**

*Laboratoire J. A. Dieudonné (LJAD)  
CNRS, Université Côte d'Azur  
Nice, France*

PATRICIA.REYNAUD-BOURET@UNIV-COTEDAZUR.FR

## Abstract

The present work aims at proving mathematically that a neural network inspired by biology can learn a classification task thanks to local transformations only. In this purpose, we propose a spiking neural network named CHANI (Correlation-based Hawkes Aggregation of Neurons with bio-Inspiration), whose neurons activity is modeled by Hawkes processes. Synaptic weights are updated thanks to an expert aggregation algorithm, providing a local and simple learning rule. We were able to prove that our network can learn on average and asymptotically. Moreover, we demonstrated that it automatically produces neuronal assemblies in the sense that the network can encode several classes and that a same neuron in the intermediate layers might be activated by more than one class, and we provided numerical simulations on synthetic dataset. This theoretical approach contrasts with the traditional empirical validation of biologically inspired networks and paves the way for understanding how local learning rules enable neurons to form assemblies able to represent complex concepts.

**Keywords:** Hawkes process, Local learning rule, Expert aggregation, Spiking neural network, Neuronal synchronization

## 1 Introduction

Biological neurons and their organization into networks served as the inspiration for the perceptron (Rosenblatt, 1957), which pioneered artificial neural networks (ANNs). This initial brain inspiration continued to drive the development of

more sophisticated models such as the neocognitron (Fukushima and Miyake, 1982) and later convolutional neural networks (LeCun et al., 1989), all drawing from the biological processes observed in the visual cortex by Hubel and Wiesel (1962). Conversely, machine learning techniques can shed light on brain learning processes. Spiking neural networks (SNNs) (Tavanaei et al., 2019) illustrate well these interactions between the two domains: these networks model neurons activity by sequences of spikes representing the electrical pulses of biological neurons. More relevant than ANNs if one wants to study the brain neural code, they also provide insights into the development of energy-efficient algorithms (Stone, 2018). To be realistic, SNNs are trained thanks to local learning rules. However, there is a gap between the empirical results provided by these rules and the underlying theory.

In the present study, as an initial stride towards demonstrating mathematically that local learning rules enable neurons to form assemblies and induce global learning, we propose a network of spiking neurons called CHANI for Correlation-based Hawkes Aggregation of Neurons with bio-Inspiration. The task is as follows: the network should learn to classify objects into one of several classes by identifying relevant feature correlations, and its learning process should involve local transformations only. The model involves hidden and output nodes as postsynaptic neurons that produce spikes as a multivariate discrete-time Hawkes process (Ost and Reynaud-Bouret, 2020), whose spiking probability is a function of the weighted sum of the activity of presynaptic neurons at the previous time step. The learning algorithm used to update synaptic weights comes from the expert aggregation field (Cesa-Bianchi and Lugosi, 2006) thanks to this local paradigm: presynaptic neurons can be seen as experts and the strength of the connections between them and the postsynaptic neuron varies based on gains derived from these connections. Hidden neurons are trained to identify neuronal synchronization among presynaptic neurons, and a pruning process is employed to retain only those neurons that encode meaningful correlations, whereas output neurons are trained to respond to the classes in which the objects are classified.

**Contributions.** We present the first biologically inspired network which provably learns a classification task thanks to a local learning rule. Furthermore, our algorithm inherently generates neuronal assemblies: the network can encode multiple classes, and a single neuron in the intermediate layers may be activated by several classes. Our contributions are multiple.

- Under very general assumptions, we derive regret bounds on the learning capabilities of hidden and output neurons (Propositions 6 and 11).
- In a more specific framework that we named CHANI EWA, we compute the explicit limit of the synaptic weights, and we provide rates of convergence (Theorems 18 and 23). We demonstrate that at the limit the hidden layers of CHANI encode feature correlations, and we establish that only pertinent feature correlations

persist following neuron selection (Proposition 19).

- Under the framework CHANI EWA, we prove that the network exhibits learning capabilities on average, and we interpret the limit output weights from a learning point of view (Corollary 20).
- When, in addition, classes are defined by feature correlations, we prove that CHANI succeeds the learning task asymptotically (Proposition 26) and we compute CHANI’s VC-dimension (Propositions 27).
- We illustrate our theoretical analysis with numerical results on the classification of handwritten digits (section 5).

**Related work.** In our prior work Jaffard et al. (2024), we introduced a network named HAN (Hawkes Aggregation of Neurons) and established theoretical foundations for its learning capabilities. However, within the framework outlined in Jaffard et al. (2024), HAN had a very simple structure, lacking hidden layers, and could only achieve very simple tasks. In the present work, we extend these results to CHANI, which can support an arbitrary number of hidden layers designed to detect neuronal synchronization, and provide novel findings described above.

Our network is inspired by the cognitive model Component-Cue (Gluck and Bower, 1988), which states that an individual learns to classify objects by finding combinations of features which describe well the several object categories. The article Mezzadri et al. (2022) compared this model to another one called ALCOVE (Kruschke, 2020), which stipulates that people classify new objects by comparing them to previously learned ones, in order to see which one is closer to human behavior. They showed that often Component-Cue is a best fit to human learning. Note that Component-Cue is a classic ANN in the sense that it does not involve spiking neurons nor local learning rule, and it does not comprise hidden layers.

In the brain, conceptual objects are represented by analyzing and representing relationships among incoming signals. While elementary concepts can be depicted by the responses of individual neurons, more intricate ones are represented by groups of interconnected neurons that work together: we talk about neuronal assemblies (Singer et al., 1997). A single neuron can contribute to multiple assemblies, indicating that it may participate in representing different concepts. A current measure of assembly organization is based on correlations of firing among neurons. It has been shown on recorded and simulated data that this synchronicity can be caused by dynamic changes of synaptic connection strength (Gerstein et al., 1989), and neuroscientists have identified several local learning rules explaining these changes. As well-known example, Hebbian learning (Hebb, 2005) says that neurons that repeatedly fire together tend to become associated.

Spike-timing-dependent plasticity (STDP) (Caporale and Dan, 2008), a refined form of Hebbian learning, states that a connection is strengthened if the presynaptic neuron spiked just before the postsynaptic neuron, and weakened otherwise; this rule has been used in order to simulate neuronal assemblies (Litwin-Kumar and

Doiron, 2014). However, there is no mathematical proof that these local rules enable to learn, nor that they produce neuronal assemblies. Establishing such theoretical guarantees would aid in comprehending how local mechanisms contribute to the formation of neuronal assemblies and how they function.

Enunciated in Legenstein et al. (2005), the Spiking Neuron Convergence Conjecture (SNCC) says that SNNs can learn to implement any achievable transformation. This conjecture is inspired by the perceptron convergence theorem (Rosenblatt et al., 1962), which asserts that as soon as the data to classify is linearly separable (*i.e.*, as soon as there exist weights with whom the data can be classified), then the weights given by the perceptron learning algorithm enable to correctly classify the data. This conjecture has been explored for networks using STDP as learning rule (Legenstein et al., 2005, 2008), with no conclusive theoretical result. In section 4.2.4, we prove a version of this conjecture in a very specific case for CHANI.

The multivariate Hawkes process (Hawkes, 1971) is a self-exciting and mutually exciting point process. Originally applied to the modelling of earthquake data (Türkyilmaz et al., 2013), its field of application is very broad: it is well-adapted to model networks of firing neurons (Hodara and Löcherbach, 2017), financial transactions (Hawkes, 2018; Bacry et al., 2015), health data (Bao et al., 2017), social networks (Zhou et al., 2013), or more generally, any sequences of events such that the occurrence of an event influences the probability of further events to occur. Also known as generalized linear models (GLMs) (Gerhard et al., 2017) in the literature, a lot of studies are about the estimation of its parameters (Reynaud-Bouret and Schbath, 2010; Kirchner, 2017), its mathematical properties (Brémaud and Massoulié, 1996) and its simulation in large networks (Bacry et al., 2017; Phi et al., 2020; Mascart et al., 2022, 2023). In neuroscience, it is used for instance to reconstruct functional connectivity (Reynaud-Bouret et al., 2013; Lambert et al., 2018), or to model networks of spiking neurons in order to study their mathematical properties (Galves and Löcherbach, 2016). However, a common assumption made when mathematically analyzing neural networks modeled by Hawkes processes is that their synaptic weights remain constant. This enables to achieve a stationary state, but prevents the modeling of learning behavior.

Hawkes processes have already been used to model events in a context of online learning: Chiang and Mohler (2020) proposes to solve spatio-temporal event forecasting and detection problems thanks to a multi-armed bandit algorithm, and Hall and Willett (2016) perform dynamic mirror descent to track how occurred events influence future events. However, none of these works use online learning algorithms to update the parameters of the Hawkes processes as we propose.

In the recent years, the emergence of transformers (Vaswani et al., 2017; Bietti et al., 2023) have revolutionized the field of deep learning, especially in natural language processing, but not only: for instance, the Transformer Hawkes Process (Zuo et al., 2020) incorporates attention modules into the formula of the conditional

intensity of the Hawkes process in order to learn event sequence data. Many works have been done to understand them better: for instance, Rohekar et al. (2023) provide a causal interpretation of self-attention, and Cordonnier et al. (2020) investigate the relationship between self-attention and convolutional layers. In section 2.8, we draw an analogy between transformers and our model to contribute in understanding the remarkable effectiveness of transformers.

## 2 Framework

In this section, we introduce our network architecture CHANI. All the notations are summarized in Appendix A. Our goal is to classify object natures belonging to a set  $\mathcal{O}$  into one of several classes belonging to a set  $K$ . In other words, we want to **learn a function from  $\mathcal{O}$  to  $K$** . CHANI should achieve this purpose thanks to local mechanisms.

For a quantity  $x_e$  indexed by  $e \in E$ , we use the following notations,  $x_E = (x_e)_{e \in E}$  and  $\langle x_e \rangle_{e \in E} = \frac{1}{|E|} \sum_{e \in E} x_e$ . These notations are also used if the index is above. For all  $n \in \mathbb{N}$ , we denote  $[n] = \{1, \dots, n\}$ . All the other notations are listed in Table 3.

### 2.1 Network architecture

The network is made of spiking neurons and aims at classifying objects, each having a nature  $o \in \mathcal{O}$ , into one of several classes. For instance, a nature of object can be a blue circle and the network may have to classify several objects having the same nature “blue circle”. The network is made of an input layer, a number  $L$  of hidden layers and an output layer. The input layer is a set of neurons  $I$  coding for features (for instance “blue”, or “circle”) meant to describe the object natures. The  $l^{\text{th}}$  hidden layer is a set  $J_l$  coding for feature tensors. It is included in a set  $I^{\vee l}$  that we define recursively as follows.

- $I^{\vee 0} := I$  and for  $i \in I$ , its support is  $S(i) := \{i\}$ .
- $I^{\vee l} := \{j \vee j' \text{ where } j, j' \in I^{\vee l-1} \text{ and } S(j) \cap S(j') = \emptyset\}$  and for  $j \vee j' \in I^{\vee l}$ ,  $S(j \vee j') := S(j) \cup S(j')$ . This set contains feature tensors of the previous  $I^{\vee l-1}$ .

Note that for  $j \in I^{\vee l}$ ,  $S(j)$  is a set of size  $2^l$  of elements belonging to  $I$ . Then we define by recursion the  $l^{\text{th}}$  hidden layer as a set  $J_l \subset I^{\vee l}$  as follows.

- $J_1 \subset I^{\vee 1}$
- $J_l \subset J_{l-1}^{\vee 1}$  where  $J_{l-1}^{\vee 1} := \{j \vee j' \text{ where } j, j' \in J_{l-1} \text{ and } S(j) \cap S(j') = \emptyset\}$ .

The output layer is a set  $K$  of neurons, each coding for a class. Hence  $K$  is also the set of classes.

**Sequential training.** CHANI’s training can be divided into  $L + 1$  sessions. For  $l \in [L]$ , the  $l^{\text{th}}$  session corresponds to the training of the  $l^{\text{th}}$  hidden layer. The session  $L + 1$  corresponds to the training of the output layer. The network’s

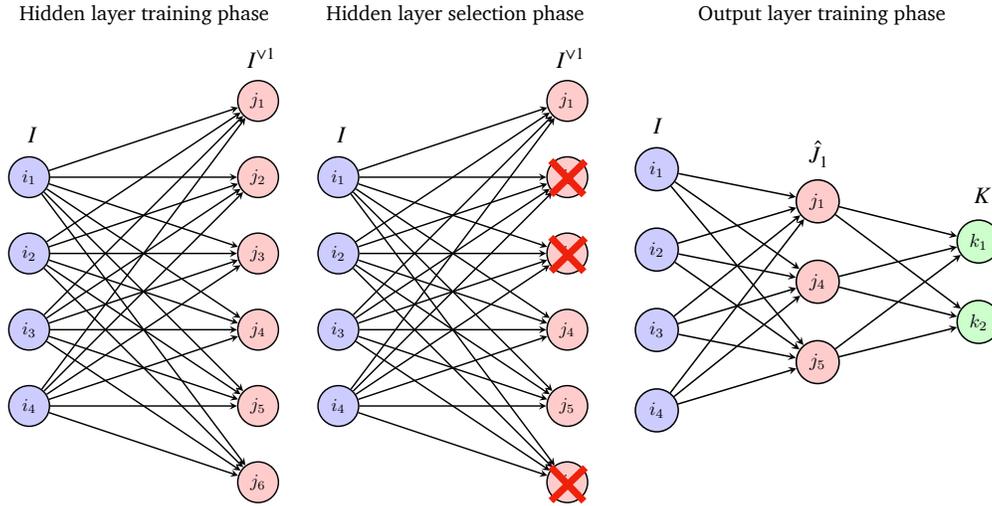


Figure 1: Illustrative example of CHANI structure during its training with depth  $L = 1$ .

structure is built sequentially. This can be biologically realistic: for instance, in the retina (Alexiades and Cepko, 1997) or in the cerebral cortex (Rash and Grove, 2006), cells are generated sequentially.

The training session of a hidden layer  $l$  is split in two phases. During the training phase, a number  $M$  of objects  $o_1^l, \dots, o_M^l \in \mathcal{O}$  are sequentially presented to the network to train the layer  $l$ , each for  $T$  time steps during which neurons can spike. Two objects can have the same nature: for instance, the first presented object  $o_1^l$  can be a blue circle, as well as the tenth presented object  $o_{10}^l$ . During the selection phase, the network is frozen in its actual state, and each nature of object  $o \in \mathcal{O}$  is presented once. This second phase aims at selecting neurons: among the set of trained neurons  $J_l$ , only a subset  $\hat{J}_l \subset J_l$  is selected to continue the network's training. The neurons belonging to the set  $J_l \setminus \hat{J}_l$  are then pruned from the network architecture. This pruning procedure can be compared to the pruning of dendritic spines pruning in biological neurons (Segal, 2005). By convention, we will denote  $\hat{J}_0 = I$  and  $J_{L+1} = K$ , and we say that an input neuron is a neuron of layer 0 and an output neuron is a neuron of layer  $L + 1$ .

The training session of the output layer  $K$  is as follows: a number  $N$  of objects  $o_1^{L+1}, \dots, o_N^{L+1} \in \mathcal{O}$  are sequentially presented to the network. There is no selection phase. For an illustration of CHANI structure during its training see Figure 1. See also Algorithm 1 for a summary of the network's training.

## 2.2 The network's activity

The network's activity is defined by random variables indicating the spike times of the neurons. The activity of neuron  $a$  at time step  $t \in [T]$  of object  $o_m^l$  during the

first phase of the training session of layer  $l \in [L + 1]$  is

$$X_{m,t}^{a,l} = \begin{cases} 1 & \text{if neuron } a \text{ spiked,} \\ 0 & \text{otherwise.} \end{cases} .$$

The activity of neuron  $a$  at time step  $t$  of object  $o$  during the selection phase of the training session of layer  $l \in [L]$  is

$$X_{o,t}^{a,l} = \begin{cases} 1 & \text{if neuron } a \text{ spiked,} \\ 0 & \text{otherwise.} \end{cases} .$$

For  $l \in [L + 1]$ , we denote by  $\mathcal{F}_{m,t}^l$  the  $\sigma$ -algebra generated by every event that happened until time step  $t$  of object  $m$  of the training phase of layer  $l$ , and for  $l \in [L]$ , we denote by  $\mathcal{F}_{o,t}^l$  the  $\sigma$ -algebra generated by every event that happened until time step  $t$  of object nature  $o$  of the selection phase of layer  $l$ .

**Definition 1 (Nature-only dependent)** *A neuron  $a$  is said to be **nature-only dependent** if there exist variables  $(Y_o^a)_{o \in \mathcal{O}}$  such that when presented with an object  $o$ , the activity of neuron  $a$  at any time step follows the same distribution as  $Y_o^a$ . In other words, the activity of neuron  $a$  depends solely on the nature of the presented object, and not on time intrinsically. We denote  $p_o^a := \mathbb{P}(Y_o^a = 1)$  its spiking probability at any time step when presented with an object of nature  $o \in \mathcal{O}$ .*

**Input layer activity.** The input neurons are nature-only dependent. During the presentation of an object, the activity of an input neuron  $i$  is an *i.i.d.* sequence of Bernoulli variables and is independent of its activity during the presentation of another object. We do not need to assume independence between input neurons, allowing for an iterative network structure.

**Hidden layers activity.** The activity of hidden neurons is non-linear discrete Hawkes processes. The conditional spiking probability of hidden neuron  $j \in J_{l'}$  at time step  $t$  of object  $o_m^l$  with  $l' \leq l$  during the training phase of layer  $l$  knowing the  $\sigma$ -algebra  $\mathcal{F}_{m,t-1}^l$  is

$$p_{m,t}^{j,l}(w_m^j) := (\psi_{m,t}^{j,l}(w_m^j))_+$$

where

$$\psi_{m,t}^{j,l}(w_m^j) := -\nu + w_m^j \cdot X_{m,t-1}^{\hat{J}_{l'-1},l}$$

where  $\nu \geq 0$ ,  $w_m^j = (w_m^{j' \rightarrow j})_{j' \in \hat{J}_{l'-1}}$  is the family of weights between  $j$  and its presynaptic neurons,  $X_{m,t-1}^{\hat{J}_{l'-1},l} := (X_{m,t-1}^{j',l})_{j' \in \hat{J}_{l'-1}}$  is the activity of neurons of layer  $l' - 1$  at the previous time step, and  $\cdot$  refers to the usual scalar product in  $\mathbb{R}^{|\hat{J}_{l'-1}|}$ .

The weights  $w_m^j$  belong to the simplex so  $p_{m,t}^{j,l}(w_m^j) \in [0, 1]$  a.s. The quantity  $-\nu$  acts as a threshold, providing non-linearity to our network. It can be biologically interpreted as a reset value of the neuron membrane potential. This non-linearity is crucial to enable the network to learn complex classes: indeed, with linear Hawkes processes instead, the network would not be able to learn to classify non-linearly separable data (such as the XOR). See Section 4.3 for a discussion about the role of  $\nu$ . When a neuron  $j$  is being trained, its synaptic weights  $w_m^j$  evolve after each presented object; hence during this phase,  $j$  is not nature-only dependent. Moreover, every neuron  $j$  ends its training phase with a final weight family  $w_{M+1}^j$ .

The conditional spiking probability of hidden neuron  $j \in J_l$  at time step  $t$  of object  $o$  during the selection phase of layer  $l'$  knowing the  $\sigma$ -algebra  $\mathcal{F}_{o,t-1}^{l'}$  is

$$p_{o,t}^j(w_{M+1}^j) := (-\nu + w_{M+1}^j \cdot X_{o,t-1}^{\hat{J}_{l-1,l}})_+.$$

Indeed, after a learning phase, synaptic weights are definitively frozen. Therefore, as long as the input neurons stay nature-only dependent, hidden neurons also become nature-only dependent during the selection phase: two objects having the same nature will elicit the same network activity on average.

**Output layer activity.** The activity of output neurons are linear discrete Hawkes processes. The conditional spiking probability of output neuron  $k \in K$  at time step  $t$  of object  $o_m^{L+1}$  knowing the  $\sigma$ -algebra  $\mathcal{F}_{m,t-1}^{L+1}$  is

$$p_{m,t}^k(w_m^k) := w_m^k \cdot X_{m,t-1}^{\hat{J}_L, L+1}. \quad (1)$$

The synaptic weights  $w_m^k$  form a probability distribution over the set of presynaptic neurons so  $p_{m,t}^k(w_m^k) \in [0, 1]$  a.s. It corresponds to the Solo case of (Jaffard et al., 2024).

### 2.3 Network structure and neuron selection

The set  $I$  is an arbitrary set of features describing the objects  $o \in \mathcal{O}$  and  $J_1 := I^{\vee 1}$ . For every  $l \in [L]$ , a threshold  $s_l \in (0, 1]$  is fixed. At the end of the selection phase of layer  $l$ , the set of selected neurons is

$$\hat{J}_l := \{j \in J_l \text{ s.t. } \exists o \in \mathcal{O}, \langle X_{o,t}^{j,l} \rangle_{t \in [T]} \geq s_l\}.$$

In other words, the selected neurons  $\hat{J}_l$  are the ones with an empirical spiking probability  $\langle X_{o,t}^{j,l} \rangle_{t \in [T]}$  larger than the threshold  $s_l$  for at least one object of the selection phase. This corresponds to step 11 of Algorithm 1. Then for  $l < L$ , the set of neurons used to train the next layer is  $J_{l+1} := \hat{J}_l^{\vee 1}$ . This selection encourages sparsity in the network in terms of the number of neurons.

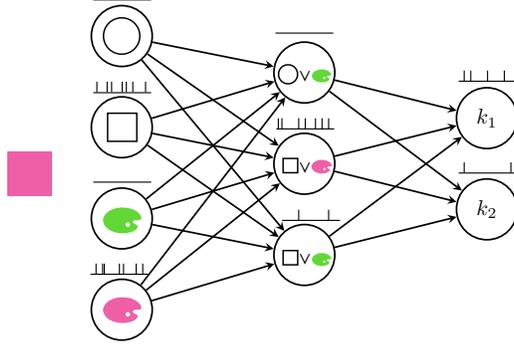


Figure 2: Illustrative example of CHANI with depth  $L = 1$  after neuron selection. The presented object excites input neurons coding for its features. Then it is classified in the class coded by the output neuron which spiked the most, here class  $k_1$ .

## 2.4 Network code and correlations

Hidden layer  $l$  aims at describing feature tensors: ideally, after its training phase, a neuron  $j = j_1 \vee j_2$  would be active when neurons  $j_1$  and  $j_2$  are correlated (*i.e.*, when they often spike simultaneously) and non-active otherwise. Besides, after the selection phase, only neurons coding for highly correlated pairs should be selected. In other words, layer  $l$  is trained to code for high correlations between neurons of layer  $l - 1$ .

The output layer aims at describing the several classes in which the objects are classified. Ideally, after its training session, a neuron  $k$  would be active when an object belonging to class  $k$  is presented to the network, and non-active otherwise. The eventual activity of neuron  $k$  when presented with an object which does not belong to its class is considered as noise.

The rule to classify objects is as follows: the object is classified in the class coded by the output neuron which spiked the most during the presentation of the object. Therefore, to correctly classify the object, the neuron coding for its class must overcome the noise coming from other neurons. For an illustration of CHANI classifying an object, see Figure 2.

We will need several notions of correlations. Let  $I' \subset I$  a subset of features,  $o \in \mathcal{O}$ . Then the *correlation of  $I'$  w.r.t. object  $o$*  is

$$\rho_o(I') := \mathbb{P}\left(\bigcap_{i \in I'} \{Y_o^i = 1\}\right),$$

the *average correlation of  $I'$*  is  $\rho(I') := \langle \rho_o(I') \rangle_{o \in \mathcal{O}}$ , and the *average correlation of  $I'$  w.r.t class  $k \in K$*  is  $\rho^k(I') := \langle \rho_o(I') \rangle_{o \in k}$ . Let  $l \in [L]$ ,  $m \in [M]$ ,  $J$  be a subset of neurons belonging to a layer with depth less than  $l$ . Then the *empirical correlation*

of  $J$  when presented with object  $o_m^l$  is

$$\hat{\rho}_m^l(J) := \left\langle \prod_{j \in J} X_{m,t}^{j,l} \right\rangle_{t \in [T]}.$$

## 2.5 Ideal network performance

We define below ideal layers representing ideal performances for CHANI.

### Definition 2 (Ideal hidden layer, ideal output layer, ideal network)

• Let  $l \in [L]$ ,  $J_l \subset I^{\vee l}$ . The spiking probabilities  $(p_o^j)_{o \in \mathcal{O}, j \in \tilde{J}_l}$  of the neurons of the set  $J_l$  are said to be ideal if there exists  $\gamma_l > 0$  such that

$$p_o^j = \gamma_l \rho_o(S(j)).$$

To simplify the exposition, we will say in this case that  $J_l$  is an ideal hidden layer of depth  $l$  and  $\gamma_l$  is its constant.

• The spiking probabilities  $(p_o^k)_{o \in \mathcal{O}, k \in K}$  of output neurons of the set  $K$  are said to be ideal if for every  $k \in K$ ,  $o \in \mathcal{O}$ ,

$$p_o^k > 0 \text{ if and only if } o \in k.$$

We will say in this case that  $K$  is an ideal output layer.

• The network is said to be ideal when all its hidden and output layers are ideal.

The family  $(p_o^j)_{o \in \mathcal{O}, j \in J_l}$  represents ideal spiking probabilities of hidden neurons  $j$  which would be nature-only dependent and would code exactly for correlations between neurons of the set  $S(j)$  with multiplicative constant  $\gamma_l$ . The family  $(p_o^k)_{o \in \mathcal{O}, k \in K}$  represents ideal spiking probabilities of output neurons  $k$  which would code exactly for classes  $k$  without any noise.

Note that this definition does not involve synaptic weights, and a priori we do not know if there exist weight families with which our network has ideal hidden and output layers. It is through the updating of its synaptic weights that our network evolves, so we cannot directly compare its performance to that of ideal layers, which do not use synaptic weights. We therefore need to define ideal weight families against which to compare the weights of our network.

**Definition 3 (Feasible weight family)** Let  $J_L \subset I^{\vee L}$ . We say that  $q^K \in (\mathcal{P}_{J_L})^{|K|}$ , where  $\mathcal{P}_{J_L}$  is the set of probability distributions over the set  $J_L$ , is a feasible weight family w.r.t  $J_L$  when

$$\text{Disc}^{\text{id}}(q^K) > 0 \quad \text{where} \quad \text{Disc}^{\text{id}}(q^K) := \langle p_o^{k,\text{id}}(q^k) - p_o^{k',\text{id}}(q^{k'}) \rangle_{\substack{k \in K \\ o \in k \\ k' \neq k}}$$

with for every  $k \in K$ ,  $p_o^{k,\text{id}}(q^k) := q^k \cdot (\rho_o(S(j)))_{j \in J_L}$ . The quantity  $\text{Disc}^{\text{id}}(q^K)$  is called the ideal discrepancy of the feasible weight family  $q^K$ : the greater the ideal discrepancy, the better the feasible weight family. We denote by  $\mathcal{Q}_{J_L}$  the set of feasible weight families w.r.t.  $J_L$ .

Note that according to (1), the quantity  $p_o^{k,\text{id}}(q^k)$  is the exact spiking probability of neuron  $k$  when connected to ideal hidden layer  $J_L$  with constant 1, with weights  $q^k$ . Therefore, the quantity  $\text{Disc}^{\text{id}}(q^K)$  represents the average difference between the activity of an output neuron  $k$  and the activity of other output neurons when presented with objects of class  $k$ : it measures how well the output layer classifies objects on average when connected to an ideal layer with weights  $q^k$ . If the output layer is connected to the ideal hidden layer  $\tilde{J}_L$  with an arbitrary constant  $\gamma_L$ , then by linearity the corresponding quantity is  $\gamma_L \text{Disc}^{\text{id}}(q^K)$ . See Figure 3.2 for an illustration of the notion of ideal discrepancy.

**Definition 4 (Strong feasible weight family)** Let  $J_L \subset I^{\vee L}$ . We say that  $q^K \in (\mathcal{P}_{J_L})^{|K|}$  is a strong feasible weight family w.r.t. the set  $J_L$  when

$$p_o^{k,\text{id}}(q^k) > 0 \quad \text{if, and only if,} \quad o \in k,$$

i.e., when the output layer  $K$ , connected to the ideal last hidden layer  $J_L$  with constant 1 and with weights  $q^K$ , is ideal.

By linearity, a family  $q^K$  is in fact a strong feasible weight family w.r.t. the set  $J_L$  when the output layer  $K$ , connected to the ideal last hidden layer  $J_L$  with any constant  $\gamma_L$  and with weights  $q^K$ , is ideal.

## 2.6 Learning rule

We extend the learning rule of Jaffard et al. (2024) to hidden layers: we use expert aggregation (Cesa-Bianchi and Lugosi, 2006) to update the synaptic weights.

### 2.6.1 EXPERT AGGREGATION

Here is a description of the expert aggregation problem. During  $M$  rounds, a forecaster can choose between several experts belonging to a set  $E$ . At each round  $m$ , each expert  $e \in E$  has an unknown gain  $g_m^e \in \mathbb{R}$ . Thanks to the past knowledge, the forecaster chooses a probability distribution  $p_m$  over the set of experts  $E$ , and then receives the aggregated gain  $g_m := p_m \cdot g_m^E$  where  $g_m^E := (g_m^e)_{e \in E}$ . This aggregated gain can be interpreted as the expectation of the gain that the forecaster would get by choosing one expert with probability  $p_m$ . As the rounds progress, experts accumulate cumulated gains  $G_m^E := (G_m^e)_{e \in E}$  where  $G_m^e := \sum_{m'=1}^m g_{m'}^e$ , as well as the forecaster which accumulates the cumulated gain  $G_m := \sum_{m'=1}^m g_{m'}$ .

The regret of the forecaster measures how good its strategy is. It is defined as

$$R_M := \max_{q \in \mathcal{P}_E} \sum_{m=1}^M q \cdot g_m^E - G_m.$$

It compares the best possible cumulated gain with a constant strategy to the actual cumulated gain of the forecaster. It quantifies how close the strategy of the forecaster is to the optimal one.

An expert aggregation algorithm is a function  $f : \mathbb{R} \times \mathbb{R}^{|E|} \mapsto [0, 1]^{|E|}$  that the forecaster uses to update its strategy: the probability distribution chosen by the forecaster for the next round is

$$p_{m+1} := f(G_m, G_m^E).$$

Expert aggregation algorithms are designed to achieve low regret bounds. We will use the following two:

- EWA (Exponentially Weighted Average) determines the probability of selecting expert  $e$  at round  $m$  as

$$p_{m+1}^e = \frac{\exp(\eta G_m^e)}{\sum_{e' \in E} \exp(\eta G_m^{e'})}.$$

The parameter  $\eta$  is the learning rate. It quantifies how fast the forecaster learns.

- PWA (Polynomially Weighted Average) determines the probability of selecting expert  $e$  at round  $m$  as

$$p_{m+1}^e = \frac{(G_m^e - G_m)_+^{b-1}}{\sum_{e' \in E} (G_m^{e'} - G_m)_+^{b-1}},$$

where  $b \geq 2$  is a parameter to choose.

These two expert aggregation algorithms employ different strategies: EWA ensures that each expert is assigned a strictly positive probability of selection, whereas PWA assigns a zero probability to experts whose cumulative gains are less than the forecaster's.

### 2.6.2 WEIGHTS UPDATE

Each neuron  $j$  of a layer  $l \in [L + 1]$  learns by updating its synaptic weights  $w_m^j = (w_m^{j' \rightarrow j})_{j' \in \hat{J}_{l-1}}$  during its training phase by running its own expert aggregation algorithm  $f^l$ . Each neuron  $j$  is a forecaster, each presented object is a round, and the corresponding set of experts is the set of presynaptic neurons  $\hat{J}_{l-1}$ . Therefore, each presynaptic neuron  $j'$  is attributed a gain  $g_m^{j' \rightarrow j}$  and the weights of  $j$  evolve according to the rule

$$w_{m+1}^j = f^l(G_m^j, (G_m^{j' \rightarrow j})_{j' \in \hat{J}_{l-1}})$$

where  $G_m^j := \sum_{m'=1}^m w_{m'}^j \cdot g_{m'}^j$  is the cumulated gain of postsynaptic neuron  $j$  and  $G_m^{j' \rightarrow j} := \sum_{m'=1}^m g_{m'}^{j' \rightarrow j}$  is the cumulated gain of presynaptic neuron  $j'$ . This corresponds to steps 7 and 18 of Algorithm 1.

Most of the theoretical findings regarding expert aggregation algorithms apply to arbitrary bounded sequences of gains. Therefore, we can choose any bounded gains suitable for the learning task. For a hidden neuron  $j = j_1 \vee j_2 \in \mathcal{J}_l$ , the gain of presynaptic neuron  $j'$  is defined as

$$g_m^{j' \rightarrow j} := \hat{\rho}_m^l(\{j_1, j_2, j'\}) \quad (2)$$

where  $\hat{\rho}_m^l(\{j_1, j_2, j'\}) = \langle X_{m,t}^{j_1,l} X_{m,t}^{j_2,l} X_{m,t}^{j',l} \rangle_{t \in [T]}$  is the empirical correlation between neurons  $j_1$ ,  $j_2$  and  $j'$ . This choice of gain is designed to achieve the objective of neuron  $j$  training, which is to detect correlations between neurons  $j_1$  and  $j_2$ .

For an output neuron  $k \in K$ , we extend the gains defined in Jaffard et al. (2024). The gain of presynaptic neuron  $j$  is

$$g_m^{j \rightarrow k} := \begin{cases} \langle X_{m,t}^{j,L+1} \rangle_{t \in [T]} \times \frac{N}{N^k} & \text{if } o_m^{L+1} \in k \\ -\langle X_{m,t}^{j,L+1} \rangle_{t \in [T]} \times \frac{N}{N^{k'}} \times \frac{1}{|K|-1} & \text{if } o_m^{L+1} \in k' \neq k \end{cases} \quad (3)$$

where  $N^{k'}$  is the number of objects belonging to class  $k'$  used to train the set of output neurons  $K$ . Therefore, when the presented object is in class  $k$ , *i.e.*, when neuron  $k$  should spike more than the other output neurons, it attributes positive gains to active presynaptic neurons. When the presented object is in another class, *i.e.*, when neuron  $k$  should stay silent, it attributes losses (negative gains) to active presynaptic neurons. During the training phase of the output layer, the hidden layer  $L$  is already trained so the synaptic weights of a neuron  $j \in \hat{\mathcal{J}}_L$  are  $w_{M+1}^j$ .

These choices of gains establish *local rules*: each neuron within a layer operates its own expert aggregation algorithm independently of other neurons in the same layer. The synaptic weights' evolution of a neuron is solely influenced by the spikes of its presynaptic neurons. There are no backward passes, and the information regarding the true class of the presented object only plays a role in the output layer learning rule.

## 2.7 CHANI algorithm

The overall CHANI algorithm is summarized in Algorithm 1. Its complexity is determined by the number of calls made to the pseudo-random generator for obtaining Bernoulli variables. We need to simulate  $O\left(MT\left(|I| + \sum_{l=1}^{L-1} |\hat{\mathcal{J}}_l|\right) + NT\left(|I| + \sum_{l=1}^L |\hat{\mathcal{J}}_l|\right)\right)$  random variables. Here the depth  $L$  is considered a constant.

---

**Algorithm 1: CHANI**

---

```

1 Initialization:  $\hat{J}_0 := I$ 
2 for  $l = 1$  to  $L$  do
3   Initialization:  $J_l := \hat{J}_{l-1}^{\vee 1}$ ,  $\hat{J}_l := \emptyset$ ,  $w^{j' \rightarrow j} := 1/|\hat{J}_{l-1}|$  for  $j \in J_l$ 
4   for  $m = 1$  to  $M$  do
5     Simulate  $(X_{m,t}^{I,l})_{t \in [T]}$ ,  $(X_{m,t}^{\hat{J}_1,l})_{t \in [T]}$ ,  $\dots$ ,  $(X_{m,t}^{\hat{J}_{l-1},l})_{t \in [T]}$  according to
       section 2.2.
6     Compute gains  $g_m^{J_l}$  according to (2).
7     for  $j \in J_l$  do
8        $w_{m+1}^j \leftarrow f^l(G_m^j, (G_m^{j' \rightarrow j})_{j' \in \hat{J}_{l-1}})$  // aggregate experts
9   for  $o \in \mathcal{O}$  do
10    Simulate  $(X_{o,t}^{I,l})_{t \in [T]}$ ,  $(X_{o,t}^{\hat{J}_1,l})_{t \in [T]}$ ,  $\dots$ ,  $(X_{o,t}^{\hat{J}_{l-1},l})_{t \in [T]}$ ,  $(X_{o,t}^{J_l,l})_{t \in [T]}$ 
       according to section 2.2.
11    for  $j \in J_l$  do
12      if  $\langle X_{o,t}^{j,l} \rangle_{t \in [T]} \geq s_l$  and  $j \notin \hat{J}_l$  then
13         $\left[ \right]$  add  $j$  to  $\hat{J}_l$  // select neuron  $j$ 
14 Initialization:  $w^{j \rightarrow k} := 1/|\hat{J}_L|$ 
15 for  $m = 1$  to  $N$  do
16   Simulate  $(X_{m,t}^{I,L+1})_{t \in [T]}$ ,  $(X_{m,t}^{\hat{J}_1,L+1})_{t \in [T]}$ ,  $\dots$ ,  $(X_{m,t}^{\hat{J}_L,L+1})_{t \in [T]}$  according to
       section 2.2.
17   Compute gains  $g_m^K$  according to (3).
18   for  $k \in K$  do
19      $w_{m+1}^k \leftarrow f^{L+1}(G_m^k, (G_m^{j \rightarrow k})_{j \in \hat{J}_L})$  // aggregate experts
Output:  $\hat{J}_1, \dots, \hat{J}_L, (w_{M+1}^j)_{j \in \hat{J}_1 \cup \dots \cup \hat{J}_L}, (w_{N+1}^k)_{k \in K}$  // selected
neurons and final weights

```

---

## 2.8 Analogy with Transformers

The key innovation of the transformer architecture (Vaswani et al., 2017) is the self-attention mechanism, which allows the model to weigh the importance of different input tokens when processing each token in a sequence. This mechanism enables the model to capture long-range dependencies more effectively than traditional recurrent or convolutional neural networks. When using the specific expert aggregation EWA, CHANI shares similarities with transformers.

**Input embedding.** Objects to be classified have features which are embedded into point processes by the input layer.

**Attention layers.** Let  $l \in [L]$ ,  $j = j_1 \vee j_2 \in J_l$  and  $m \in [M]$ . Then, the conditional spiking probability of neuron  $j$  during the presentation of object  $o_m^l$  of its learning phase reads

$$p_{m,t}^{j,l}(w_m^j) = \text{softmax}(\eta^j A_{m-1}^j \cdot B_{m-1}) \cdot X_{m,t-1}^{\hat{J}_{l-1}}$$

where  $X_{m,t-1}^{\hat{J}_{l-1}} = (X_{m,t-1}^{j'})_{j' \in \hat{J}_{l-1}}$  is the activity of the previous layer at the previous time step of current object  $o_m^l$ ,  $A_{m-1}^j := (X_{m',t}^{j_1} X_{m',t}^{j_2})_{1 \leq m' \leq m-1, 1 \leq t \leq T}$  is the vector of correlations between  $j_1$  and  $j_2$  until previous object  $o_{m-1}^l$ ,  $B_{m-1} := (X_{m',t}^{j'})_{j' \in \hat{J}_{l-1}, m' \in [m-1], t \in [T]}$  is the vector of previous layer activity until object  $o_{m-1}^l$ , and  $A_{m-1}^j \cdot B_{m-1} := \left( \sum_{t \in [T]} X_{m',t}^{j_1} X_{m',t}^{j_2} X_{m',t}^{j'} \right)_{j' \in \hat{J}_{l-1}}$  is the matrix of correlations between  $j_1, j_2$  and neurons of the previous layer until previous object  $o_{m-1}^l$ . Therefore, the hidden layer of depth  $l$  acts as an attention layer enlightening correlations between neurons of the previous layers. At the end of the learning phase, the resulting weight matrix  $w_{M+1}^{J_L}$  can be seen as a score matrix which is then used to select neurons coding for high correlations. The accumulation of hidden layers is analogous to a succession of attention layers.

**Final linear layer.** The output layer acts as a linear layer at the end of an attention module as the conditional spiking probability is linear in the activity of the last hidden layers.

## 3 Direct interpretation via regret bounds

In this section, we assume that the expert aggregation algorithms admit a regret bound. As each hidden and output neuron runs its own expert aggregation algorithm, it possesses its own regret. Through our choice of gains, we reinterpret these regrets, enabling us to gain meaningful insights into the network's learning.

### 3.1 Hidden layers

**Assumption 5 (Hidden neurons regret bound)** *There exists  $C > 0$  such that for every  $l \in [L]$ , the expert aggregation algorithm  $f^l$  used to train the hidden*

layer of depth  $l$  is such that for any set of experts  $E$  and any deterministic sequence of gains  $g_{1:M}^E := (g_m^e)_{e \in E, m \in [M]}$  taking value in  $[a, b]$  with  $a < b$ ,

$$R_M \leq C(b - a)\sqrt{\ln(|E|)M}.$$

Note that both EWA and PWA verify this bound for certain parameters (see Cesa-Bianchi and Lugosi (2006) for details). The interpretation of the regret of hidden neurons leads to the following result.

**Proposition 6 (Hidden neurons regret bound)** *Suppose Assumption 5 (hidden neurons regret bound) holds. Then a.s. for every  $l \in [L]$  and  $j = j_1 \vee j_2 \in J_l$  neuron of the  $l^{\text{th}}$  layer,*

$$\max_{q \in \mathcal{P}_{J_{l-1}}} \left\langle (\psi_{m,t}^{j,l}(q) - \psi_{m,t}^{j,l}(w_m^j)) X_{m,t}^{j_1} X_{m,t}^{j_2} \right\rangle_{\substack{t \in [T] \\ m \in [M]}} \leq C \sqrt{\frac{\ln(|\hat{J}_{l-1}|)}{M}}.$$

This regret bound can be interpreted as follows: since  $p_{m,t}^{j,l}(w_m^j) = (\psi_{m,t}^{j,l}(w_m^j))_+$ , on average, neuron  $j = j_1 \vee j_2$  will tend to spike as frequently as possible when neurons  $j_1$  and  $j_2$  are highly correlated, that is, when the product  $X_{m,t}^{j_1} X_{m,t}^{j_2}$  is often equal to 1. As HAN did not have hidden layers, this result represents a first step towards analyzing the contribution of hidden layers to CHANI.

### 3.2 Output layer

To interpret output neurons regret, we extend the notion of class discrepancy introduced in Jaffard et al. (2024) as follows.

**Definition 7 (Class discrepancy)** *Let  $k \in K$  an output neuron. Then the class discrepancy of neuron  $k$  with weights  $q$  is the quantity*

$$\text{Disc}_N^k(q) := \left\langle \hat{p}_m^k(q) - \hat{p}_m^{k'}(q) \right\rangle_{\substack{k' / k' \neq k \\ m / \sigma_m^{L+1} \in k}}$$

This new definition of class discrepancy generalizes the one of Jaffard et al. (2024) to a network with any hidden layers' activity. It compares the average activity of neuron  $k$  with the average activity of other neurons when presented with objects of class  $k$ . The larger the class discrepancy is, the more neuron  $k$  overcomes the noise in order to correctly classify the objects of its class. It is a **global notion** since it involves the activity of all the output neurons. The average class discrepancy of the network is called the network discrepancy, that we define as follows.

**Definition 8 (Network discrepancy)** *The network discrepancy of the network with output weights  $q^K = (q^k)_{k \in K} \in (\mathcal{P}_{\hat{J}_L})^{|K|}$  is the quantity*

$$\text{Disc}_N(q^K) := \langle \text{Disc}_N^k(q^k) \rangle_{k \in K}.$$

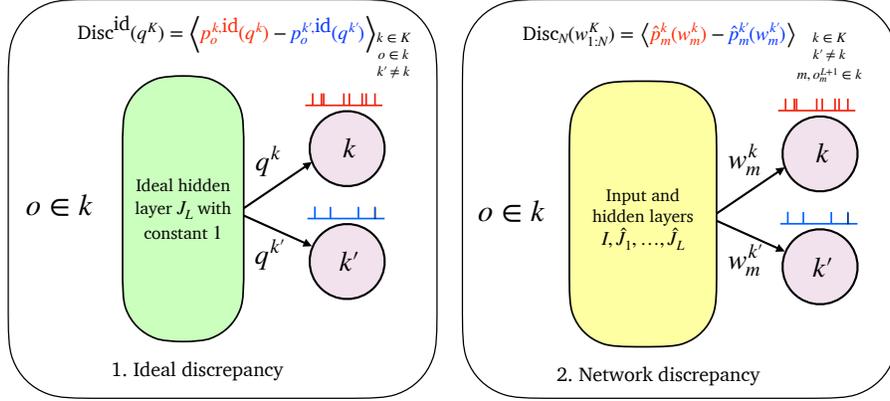


Figure 3: Illustration of the notions of ideal (1) and network (2) discrepancies.

For an illustration of the notion of network discrepancy, see Figure 3.

**Assumption 9 ( $\xi$ -balanced)** *There exists a constant  $\xi > 0$  independent of  $N$  such that for every  $k \in K$ ,  $N_k/N \geq \xi$ .*

This assumption means that during the training phase of the output layer, the proportion of presented objects of any class is at least  $\xi$ . This ensures that the network sees a significant number of objects of each class.

**Assumption 10 (Output neurons regret bound)** *The expert aggregation algorithm  $f^{L+1}$  used to train the output layer is such that for any set of experts  $E$  and any deterministic sequence of gains  $g_{1:N}^E := (g_m^e)_{\substack{e \in E \\ m \in [N]}}$  taking value in  $[a, b]$ ,*

$$R_M \leq C(b - a)\sqrt{\ln(|E|)M}$$

where  $C$  is a constant.

**Proposition 11** *Suppose Assumptions 9 and 10 hold. Then,*

$$\text{Disc}_N(w_{1:N}^K) \geq \max_{q^K \in (\mathcal{P}_{J_L})^{|K|}} \text{Disc}_N(q^K) - \frac{C|K|}{\xi(|K| - 1)} \sqrt{\frac{\ln(|\hat{J}_L|)}{N}} \quad a.s.$$

The proposition, established by combining local regret bounds of output neurons, ensures that the network discrepancy of HAN exceeds the maximum achievable with constant weights for the output layer, minus an error term which tends to zero. However, the quantity  $\text{Disc}_N(q^K)$  to which we compare CHANI's network discrepancy depends on the activity of CHANI's final hidden layer. In section 4, we introduce a more specific framework in which we compare CHANI's performance against that of ideal layers which operate independently of CHANI.

## 4 Theoretical results for CHANI EWA

### 4.1 CHANI EWA Framework

In this section, we wish to conduct a more thorough analysis of the network's performance. In this purpose, we need to work within a more precise framework that we call CHANI EWA, defined by the following assumptions.

**Assumption 12 (EWA for hidden layers)** *For any  $l \in [L]$ , the expert aggregation  $f^l$  used to train the hidden layer of depth  $l$  is EWA with learning rate  $\eta^l = \sqrt{\frac{8 \ln(|\tilde{J}_{l-1}|)}{M}}$ .*

Note that this choice of  $\eta^l$  maximizes the regret bound of EWA given in Cesa-Bianchi and Lugosi (2006). This assumption imposes no restriction on the expert aggregation algorithm  $f^{L+1}$  used to train the output layer.

**Assumption 13 (Balanced)** *During each training session of a layer, each nature of object  $o \in \mathcal{O}$  is presented the same amount of times to the network.*

This is a technical assumption that we make to facilitate the calculations.

**Assumption 14 (1/2 bias)** *All hidden neurons have bias  $\nu = \frac{1}{2}$ .*

This assumption is here to ensure that hidden neurons activity converge to ideal hidden layers. See discussion about the choice of  $\nu$  in section 4.3.

**Assumption 15 (Decreasing correlation)** *For all  $I' \subset I$  such that  $\rho(I') > 0$ , for all  $i \in I \setminus I'$ ,  $\rho(I' \cup \{i\}) < \rho(I')$ .*

This assumption means that adding a neuron to a set of correlated neurons results in a loss in correlation for at least one object. It is obviously verified when input neurons spike independently of one another. To ensure that input neurons activity verify this assumption, one can add a small independent perturbation to their activity.

**Assumption 16 (Max-correlated sets)** *There exist sets  $\tilde{J}_l$  for  $l \in [L]$  defined by induction as follows.*

- $\tilde{J}_1 \subset I^{V1}$  and there exists  $p_1 \in (0, 1]$  such that

$$\forall j \in \tilde{J}_1, \exists o \in \mathcal{O}, \rho_o(S(j)) \geq p_1 \text{ and } \forall j' \in I^{V1} \setminus \tilde{J}_1, \forall o \in \mathcal{O}, \rho_o(S(j')) < p_1.$$

- For  $l \in \{2, \dots, L\}$ ,  $\tilde{J}_l \subset \tilde{J}_{l-1}^{V1}$  and there exists  $p_l \in (0, 1]$  such that

$$\forall j \in \tilde{J}_l, \exists o \in \mathcal{O}, \rho_o(S(j)) \geq p_l \text{ and } \forall j' \in \tilde{J}_{l-1}^{V1} \setminus \tilde{J}_l, \forall o \in \mathcal{O}, \rho_o(S(j')) < p_l.$$

By convention, we denote  $\tilde{J}_0 := I$ .

This assumption states that there exist sets of feature tensors  $\tilde{J}_l$  coding for high feature correlations for at least one object. These sets represent important combination of features that are needed to describe the classes. They will be the sets of interest that we want to target during hidden layer neuron selection phases.

**Assumption 17 (Sparse features)** *Let  $l \in \{0, \dots, L\}$ . The set  $\tilde{J}_l$  defined in Assumption 16 is such that for all  $o \in \mathcal{O}$ ,  $|\{j \in \tilde{J}_l, \rho_o(S(j)) > 0\}| \leq \frac{|\tilde{J}_l|}{2}$ .*

This assumption means that for any object  $o \in \mathcal{O}$ , at most half the neurons of  $I$  and  $\tilde{J}_l$  for  $l \in [L]$  are active and spike together. This is a technical assumption that we need to study CHANI asymptotic behavior.

## 4.2 Limit behavior

### 4.2.1 HIDDEN LAYERS

In the framework CHANI EWA, we can explicitly compute the limit of hidden neurons weights.

**Theorem 18** *Suppose we are in the CHANI EWA framework. Let  $\alpha > 0$ . There exist constants  $C_1$  and  $C_2$  independent of  $M$  and  $T$  such that if  $M \geq C_1$  and  $T \geq C_2 M^{L-1}$ , then there exist thresholds  $s_1, \dots, s_L$  such that with probability  $1 - \alpha$ :*

- (i) *For all  $l \in [L]$ ,  $\hat{J}_l = \tilde{J}_l$ .*
- (ii) *Let  $l \in [L]$ . Then, for all  $j = j_1 \vee j_2 \in \tilde{J}_l$ , at the end of learning the synaptic weights are such that*

$$\|w_{M+1}^j - \frac{1}{2} \mathbb{1}_{\{j_1, j_2\}}\|_2 = O\left(\left(\mathcal{J}_{l-1} \ln(\mathcal{J}_{l-1})^{\frac{1}{2}} \ln(\mathcal{J}_{l-1} \alpha^{-1})^{\frac{1}{2}}\right)^{l-1} \left(\frac{M^{l-1}}{T}\right)^{1/2} + e^{-\square \min_{l' \leq l-1} \{2^{-2^{l'}} \rho_{l'} \ln(|\tilde{J}_{l'}|)^{1/2}\}} M^{1/2}\right)$$

where  $\mathbb{1}_{\{j_1, j_2\}} := (\mathbb{1}_{j' \in \{j_1, j_2\}})_{j' \in \tilde{J}_{l-1}}$ ,  $\mathcal{J}_{l-1} := \min_{l' \leq l-1} |\tilde{J}_{l'}|$ ,  $\rho_{l'} := \min_{j \in \tilde{J}_{l'}} \rho(S(j))$  is the minimum average correlation of the support of a hidden neuron of depth  $l'$  and  $\square$  is a constant independent of the network parameters.

This theorem states that the sets of interest  $\tilde{J}_l$  are selected with high probability. Besides, the weights of a hidden neuron  $j = j_1 \vee j_2$  converge to uniform distribution on neurons  $j_1$  and  $j_2$  with an error term which is small when  $M \gg 1$  and  $T \gg M^{L-1}$ . The dependency on the size of the selected sets  $|J_l|$  also shows that the fewer neurons selected, the better. When taking only into account the dependency on  $M$  and  $T$ , the error term of the weights of layer  $l$  is in  $O\left(\left(\frac{M^{l-1}}{T}\right)^{1/2} + e^{-C\sqrt{M}}\right)$  where  $C$  is a constant independent of  $M$  and  $T$ . The exact error term is given in the proof of Theorem 18 (see Appendix C.3). It can be decomposed in two terms:

the first one comes from the randomness induced by the spikes, as well as the accumulation of the errors coming from previous layers. The second one comes from the specific use of the expert aggregation algorithm EWA. This second term decreases as the average correlation of the support of hidden neurons grows: the more hidden neurons describe high correlations between input neurons the better.

**Proposition 19** *The hidden layers of the network with limit weights as defined in Theorem 18 are ideal.*

Combined with Theorem 18, this proposition ensures that with high probability, at the limit the hidden layers of our network code exactly for feature correlations. A hidden neuron is activated when presented with objects having a certain combination of features. If this combination can be found in several classes, then the same neuron will be activated, and can therefore help to represent more than one class: in this sense, CHANI produces neuronal assemblies. These results, valid for any number of hidden layers, are unprecedented since in our previous work Jaffard et al. (2024) HAN did not have hidden layers.

#### 4.2.2 AVERAGE LEARNING

Now that we studied the hidden layers behavior, we can analyze the term to which we compared the network discrepancy in Proposition 11 in order to get a comparison to a more sensible quantity: the largest ideal discrepancy of a feasible weight family. Let

$$E_{\text{hid}}(M, T, \alpha) := \max_{l \in [L], j = j_1 \vee j_2 \in \tilde{J}_L} \|w_{M+1}^j - \mathbb{1}_{\{j_1, j_2\}}\|_1$$

the maximum error term of a neuron of hidden layer, bounded in Theorem 18. By combining Theorem 18 and Proposition 11, we get the following corollary.

**Corollary 20** *Suppose we are in the CHANI EWA framework and Assumption 10 (output neurons regret bound) holds. Let  $\alpha > 0$ . Suppose  $M \geq C_1$  and  $T \geq C_2 M^{L-1}$  where  $C_1$  and  $C_2$  are the constants defined in Theorem 18,  $s_1, \dots, s_L$  are the thresholds given by Theorem 18 and  $\mathcal{Q}_{\tilde{J}_L}$ , the set of feasible weight families w.r.t. the ideal set of neurons  $\tilde{J}_L$ , is non-empty. Then with probability  $1 - \alpha$  the conclusions of Theorem 18 hold and there exists a constant  $\gamma_L > 0$  such that*

$$\begin{aligned} \text{Disc}_N(w_{1:N}^K) &\geq \gamma_L \max_{q^K \in \mathcal{Q}_{\tilde{J}_L}} \text{Disc}^{\text{id}}(q^K) \\ &\quad - O\left(\sqrt{\frac{\ln(|\tilde{J}_L| \alpha^{-1})}{NT}} + \sqrt{\frac{\ln(|\tilde{J}_L|)}{N}} + E_{\text{hid}}(M, T, \alpha)\right). \end{aligned}$$

This corollary establishes that CHANI’s network discrepancy exceeds the ideal discrepancy of the best feasible weight family possible multiplied by a constant  $\gamma_L$ , minus an error term which is small when  $M \gg 1$ ,  $T \gg M^{L-1}$  and  $N \gg 1$ , and increases with the number of selected neurons. This error term is threefold: the first part comes from the randomness of the spikes, the second one comes from the regret bound of the expert aggregation of the output layer  $f^{L+1}$ , and the third part comes from the approximation error between hidden layer weights and their limit. When taking only into account the dependency in  $T$ ,  $M$  and  $N$ , the overall error term is in

$$O\left(N^{-1/2} + \left(\frac{M^{L-1}}{T}\right)^{1/2} + e^{-C\sqrt{M}}\right)$$

where  $C > 0$  is a constant independent of  $M$ ,  $N$  and  $T$ . Note that this Theorem holds for any expert aggregation used to train the output layer, as long as it verifies Assumption 10 (output neurons regret bound).

In other words, on average, CHANI performs as well as it would if the output layer were connected to the ideal hidden layer  $\tilde{J}_L$  with constant  $\gamma_L$  with the best possible feasible weight family. Note that unlike Proposition 11, this Theorem compares CHANI’s network discrepancy to a deterministic quantity which does not depend on CHANI itself, and which represents an ideal performance where the hidden layers detect neuronal synchronization and the output layer succeeds to rewrite the classes in terms of combination of correlations between  $L$  features.

This theorem can be related to Theorem 3.3 of Jaffard et al. (2024), where HAN network discrepancy is compared to the best safety discrepancy of a feasible weight family, a quantity describing how well a feasible weight family classifies objects. However, this new result is much more general, for several reasons. Firstly, in the current work, CHANI can comprise any number of hidden layers, whereas in our previous work HAN is restricted to having solely an input and an output layer. Secondly, both of these theorems assume the existence of a feasible weight family. However, in Jaffard et al. (2024), this assumption is considerably more restrictive, as a feasible weight family can only exist if the classes can be described as simple combinations of features. In our current study, a feasible weight family is achievable when classes can be characterized as combinations of feature correlations, the complexity of which grows with the depth  $L$  (see conditions for the existence of strong feasible weights families in section 4.2.4). In particular, if a feasible weight family does not already exist, the addition of layers may facilitate its emergence.

### 4.2.3 OUTPUT LAYER

In this section, we work in the framework CHANI EWA, with the additional assumption that the expert aggregation used for the output layer is EWA as well. This allows us to end the asymptotic study started in section 4.2.1 by computing the limit output weights.

**Assumption 21 (EWA for the output layer)** *The expert aggregation  $f^{L+1}$  used to train the output layer is EWA with learning rate  $\eta^k = \frac{1}{|\mathcal{O}|} \sqrt{\frac{2 \ln(|\tilde{J}_L|)}{N}}$ .*

Note again that this choice of  $\eta^k$  maximizes the regret bound of EWA given in Cesa-Bianchi and Lugosi (2006).

We extend the notion of feature discrepancy defined in Jaffard et al. (2024) for an input neuron coding for a feature to a neuron of the last hidden layer coding for a feature tensor of depth  $L$ .

**Definition 22 (Feature discrepancy)** *Let  $j \in I^{\vee L}$ . The feature discrepancy of  $j$  w.r.t. class  $k \in K$  is the quantity*

$$\text{Disc}^{j \rightarrow k} := \rho^k(S(j)) - \langle \rho^{k'}(S(j)) \rangle_{k' \neq k}$$

The feature discrepancy of neuron  $j$  quantifies how much the feature correlation encoded by  $j$  is relevant to class  $k$ : it compares the average correlation of the support of  $j$  w.r.t. class  $k$  with its average correlation w.r.t. other classes. This new definition of feature discrepancy is much more complex than that of Jaffard et al. (2024). In our previous work, the feature discrepancy of an input neuron concerned only its individual activity, with no link to the activity of other input neurons whereas here it can capture complex patterns of input neurons activity.

**Theorem 23** *Suppose we are in the CHANI EWA framework and Assumption 21 holds. Let  $\alpha > 0$ . Suppose  $M \geq C_1$  and  $T \geq M^{L-1} C_2$  where  $C_1$  and  $C_2$  are the constants defined in Theorem 18 and  $s_1, \dots, s_L$  are the thresholds given by Theorem 18. Besides, let  $\tilde{J}_L^k := \arg \max_{j \in \tilde{J}_L^k} \text{Disc}^{j \rightarrow k}$  and  $\mathbb{1}_{\tilde{J}_L^k} := (\mathbb{1}_{j \in \tilde{J}_L^k})_{j \in \tilde{J}_L}$ . Then with probability  $1 - \alpha$ , the conclusions of Theorem 18 hold and for all  $k \in K$*

- if  $\tilde{J}_L^k = \tilde{J}_L$  then

$$\|w_{N+1}^k - \frac{1}{|\tilde{J}_L^k|} \mathbb{1}_{\tilde{J}_L^k}\|_2 = O\left(\sqrt{|\tilde{J}_L| \ln(|\tilde{J}_L|)} N E_{\text{hid}}(M, T, \alpha)\right).$$

- if  $\tilde{J}_L^k \subsetneq \tilde{J}_L$  let  $\Delta^k := \max_{j \in \tilde{J}_L^k} \text{Disc}^{j \rightarrow k} - \max_{j \in \tilde{J}_L \setminus \tilde{J}_L^k} \text{Disc}^{j \rightarrow k}$ . Then

$$\|w_{N+1}^k - \frac{1}{|\tilde{J}_L^k|} \mathbb{1}_{\tilde{J}_L^k}\|_2 = O\left(\sqrt{|\tilde{J}_L| \ln(|\tilde{J}_L|)} N E_{\text{hid}}(M, T, \alpha) + e^{-\square \frac{\Delta^k}{2^{2L}}} \sqrt{\ln(|\tilde{J}_L|)^N}\right).$$

Note that the error term  $E_{\text{hid}}(M, L, T)$  is independent of the size of the last hidden layer  $|\tilde{J}_L|$  and the number of presented objects for the training phase of the output layer  $N$ . The overall error is twofold: the first part comes from the approximation error between hidden layers weights and their limit, and the second

part comes from the use of the expert aggregation EWA for the output layer. When taking only into account the dependency on  $M$ ,  $N$  and  $T$ , the error is in

$$O\left(\left(\frac{NM^{L-1}}{T}\right)^{1/2} + N^{1/2}e^{-CM^{1/2}} + e^{-DN^{1/2}}\right)$$

where  $C$  and  $D$  are constants independent of  $M$ ,  $N$  and  $T$ : it is small when  $N \gg 1$ ,  $M \gg \ln(N)^2$ ,  $T \gg NM^{L-1}$  and increases with the number of selected neurons.

This theorem states that the weights of an output neuron  $k$  converge to the family which uniformly distributes weights on presynaptic neurons with maximal feature discrepancy. Therefore, at the limit, neuron  $k$  will be connected only to neurons coding for correlations which are sensible to class  $k$ . This result can be related to Theorem 3.4 of Jaffard et al. (2024), where we computed HAN limit weights with no hidden layers. Here, we generalize this asymptotic analysis to CHANI. Besides, thanks to the addition of hidden layers, CHANI's limit output weight family is more interesting than HAN's from a learning point of view since it decomposes classes in terms of feature correlations of depth  $L$  (against combinations of simple features). In a general case, we cannot say if this family is a strong feasible weight family, or even if it enables the network to correctly classify objects when noise is allowed. However, we can conclude in a more precise framework defined below where classes are defined by feature correlations.

#### 4.2.4 WHEN CLASSES ARE DEFINED BY FEATURE CORRELATIONS

First, let us define some assumptions that we will need to perform our study.

**Assumption 24 (Binary correlations)** *There exists  $p \in (0, 1]$  such that for all  $j \in \tilde{J}_L$ , for all  $o \in \mathcal{O}$ ,  $\rho_o(S(j)) \in \{0, p\}$ . If  $\rho_o(S(j)) = p$  then we say that object  $o$  has features  $S(j)$ . Besides, for  $j \in \tilde{J}_L$ , let  $\mathcal{O}^j := \{o \in \mathcal{O}, \rho_o(S(j)) = p\}$  the set of objects with features  $S(j)$ . Then all the  $\mathcal{O}^j$  have the same cardinality.*

The first part of this assumption is verified if for instance the input neurons spike independently of each other and have a fixed spiking probability when they are activated. The restriction on the cardinality of the sets  $\mathcal{O}^j$  is a technical assumption that we make in order to facilitate the computations.

**Assumption 25 (Class decomposition)** *For every  $k \in K$ , there exists a set  $E^k \subset \tilde{J}_L$  such that  $k = \bigcup_{j \in E^k} \mathcal{O}^j$ .*

This assumption means that the classes in which the objects are classified are defined by feature correlations. The complexity of these correlations increases with  $L$ : as the network's depth grows, a set  $\mathcal{O}^j$  captures more intricate patterns within the objects.

**Proposition 26** *Suppose Assumption 24 (binary correlations) holds. Then the following statements are equivalent.*

- (i) *Assumption 25 (class decomposition) is verified.*
- (ii) *There exists a strong feasible weight family w.r.t. the set  $\tilde{J}_L$ .*
- (iii) *The limit output weights family defined in Theorem 23 is a strong feasible weight family.*

*In particular, when Assumption 25 (class decomposition) is verified, the limit network is ideal.*

Combined with Theorems 18 and 23, this proposition asserts that at the limit the network is ideal when classes are defined by feature correlations of depth  $L$ . A neuron of a hidden layer  $j$  codes exactly for the correlations of the features belonging to its support, *i.e.*, when presented with an object  $o$  its spiking probability is proportional to  $\rho_o(j)$ . Besides, a neuron  $k$  of the output layer codes exactly for class  $k$ , *i.e.*, its spiking probability is strictly positive if and only if  $o \in k$ . Furthermore, the output layer does not produce any noise: an output neuron stays completely silent when presented with an object belonging to another class. We had to make some technical assumptions to achieve this result; however, our numerical studies of section 5 suggest that we do not need all these assumptions for CHANI to work empirically.

Besides, thanks to the equivalence between the existence of a strong feasible weight family and Assumption 25 (class decomposition), we know that **as soon as there exists a strong feasible weight family, the weights of the output layer of our network converge to one of these families with high probability.** We can see this statement as a version of the Spiking Neuron Convergence Conjecture (SNCC) which says that spiking neural networks can learn to implement any achievable transformation: this conjecture is true for CHANI in this simplified framework.

This result is unprecedented since in our study Jaffard et al. (2024), we had to assume that the output limit weights were a feasible weight family in order to draw conclusions (see Corollary 3.5), without providing any criteria ensuring it.

In this framework, we can compute the Vapnik-Chervonenkis dimension of our network. For a given classification algorithm, its VC-dimension corresponds to the size of the largest set of points that it can shatter. It measures the complexity of the class of functions that can be learned. In the literature, this dimension appears in generalization error bounds.

We consider any last hidden layer  $J_L \subset I^{\vee l}$ . We assimilate the objects to be classified as sequences  $o = (o_i)_{i \in I}$  where  $o_i = 1$  if  $o$  has feature  $i$  and  $o_i = 0$  otherwise: they live in a space of dimension  $|I|$ . Neuron  $j \in J_L$  is activated when  $o$  has all the features of the set  $S(j)$ . We consider a binary output: let the hypothesis set  $H := \{\mathbb{1}_{J'_L} \text{ such that } J'_L \subset J_L\}$ , where for an object  $o$ ,  $\mathbb{1}_{J'_L}(o) = 1$  if and only if

there exists  $j \in J'_L$  activated by  $o$ , and 0 otherwise. It corresponds to the functions that CHANI can learn. Then, we have the following result.

**Proposition 27** *The VC-dimension of the hypothesis set  $H$  is the size of the last hidden layer*

$$\text{VCdim}(H) = |J_L|.$$

Therefore, the complexity of the classes that CHANI can learn is directly related to the size of the last hidden layer, which contains in itself the complexity of all the previous layers. However, as the number of selected neurons increases, the errors in the results regarding the asymptotic behavior of CHANI also increase. Therefore, it's essential to strike a balance between accurately representing complex classes and learning efficiently.

### 4.3 Discussion about the choice of bias.

Assumption 14 (1/2 bias) could be relaxed to any  $\nu \in [1/2, 1)$ : Theorem 18 and Proposition 19 would still hold. Indeed, their key ingredient is that hidden neurons spiking probabilities converge to ideal spiking probabilities. The conditional spiking probability of hidden neuron  $j = j_1 \vee j_2$  with bias  $\nu$  and weights  $\mathbb{1}_{\{j_1, j_2\}}$  (which are the limit weights given by Theorem 18) is  $\left(-\nu + \frac{1}{2}(X_{m,t-1}^{j_1} + X_{m,t-1}^{j_2})\right)_+$ . For  $\nu \in [1/2, 1)$ , neuron  $j$  is active if and only if neurons  $j_1$  and  $j_2$  spike, and in this case its conditional spiking probability is  $1 - \nu$ . Therefore, the limit spiking probabilities would still be ideal. However, there is a decrease in firing rate from one hidden layer to the next and the choice 1/2 guarantees the lowest decrease. This decrease appears in the constant  $\gamma_L$  of Corollary 20: in fact,  $\gamma_L = (1 - \nu)^{2^L - 1}$  and a large  $\gamma_L$  guarantees a large class discrepancy for our network. Therefore, the choice  $\nu = 1/2$  is optimal.

For  $\nu < 1/2$ , neuron  $j$  is active even when only one of the two neurons  $j_1$  and  $j_2$  spikes, so this choice of bias does not provide ideal limit spiking probabilities and the extra activity of hidden neurons can be identified as noise.

Let us study a very simple example in the framework of section 4.2.4 where for any  $\nu < 1/2$ , the limit weights do not enable to correctly classify the objects.

We consider the set of object natures  $\mathcal{O} = \{\square, \square, \square, \circ, \circ, \circ, \nabla, \nabla, \nabla\}$ . The set of features and input neurons is  $I = \{\text{blue, red, green, square, circle, triangle}\}$ . We assume that input neurons  $i$  spike independently of each other with probability  $p'$  if and only if the presented object has feature  $i$ , and we choose depth  $L = 1$ . Then the set

$$\begin{aligned} \tilde{J}_1 := \{ & \text{blueVsquare, red} \vee \text{square, green} \vee \text{square, blue} \vee \text{circle, red} \vee \text{circle,} \\ & \text{green} \vee \text{circle, blue} \vee \text{triangle, red} \vee \text{triangle, green} \vee \text{triangle}\} \end{aligned}$$

verifies Assumption 16 (max-correlated sets) with  $p_1 = p'^2$  and can be identified with the set of object natures  $\mathcal{O}$ , and Assumption 24 (binary correlations) is verified. Therefore, any set of classes  $K$  verify Assumption 25 (class decomposition) since any class can be written as the union of its single elements. Let us choose  $K = \{k_1, k_2\}$  with  $k_1 = \{\square, \blacksquare, \circ, \nabla\}$  and  $k_2 = \{\square, \circ, \bigcirc, \nabla, \blacktriangledown\}$ . This classes are such that class  $k_1$  contains every item which is squared or blue except the blue square, and class  $k_2$  contains every other possible object.

**Proposition 28** *In this framework, for any  $\nu \in [0, 1/2)$ , the limit weights do not enable to correctly classify the  $\square$ .*

Therefore, even with  $\nu$  less but close to  $1/2$ , the limit network fails to correctly classify the objects, whereas for any  $\nu \geq 1/2$  the results of section 4.2.4 hold and the limit network succeeds to accomplish the task. This proposition enlightens that even a small noise can stop the network from working properly in a very simple case. The choice of bias allows us to control this phenomenon.

## 5 Numerical results

We trained CHANI on the `digits` dataset of `scikit-learn`. This dataset provides 1797 images of 64 pixels of handwritten digits which were randomly separated in a training set (80% of the images) and a testing set (20% of the images). For computational reasons, we decided to use this dataset rather than one with better resolution. The numerical results for CHANI with no hidden layer (which corresponds to the network of our previous work HAN) are visible in Table 1. Although CHANI with EWA performs much better than CHANI with PWA, its accuracy is only around 50%.

Expert Aggregation algorithm	Accuracy	Confidence interval of level 0.9
EWA	53.0	[50.8, 55.6]
PWA	14.8	[14.7, 15.0]

Table 1: Numerical results with no hidden layer. The parameters are  $T = 2000$ ,  $N = 1437$ ,  $\eta^k = 0.0005$ ,  $b = 2$ . A number 100 of realizations were made.

The results for CHANI with one hidden layer are visible in Figure 4. The blue curve (resp. the green curve) represents the percentage of correct classifications of the testing set of CHANI with the expert aggregation algorithm EWA (resp. PWA) in function of the number of selected hidden neurons. Before selection, there was 2016 neurons. We can see that no matter the number of selected neurons, CHANI with PWA performs badly and does not exceed a 40% correct classifications. Besides, its performance is unstable. On the other hand, the performance of

## CHANI

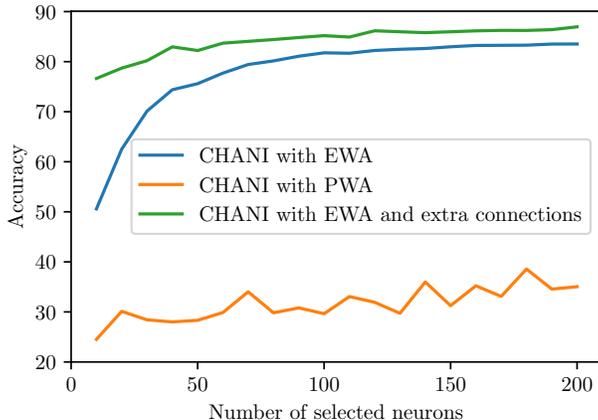


Figure 4: Numerical results with one hidden layer for CHANI with EWA, CHANI with PWA and CHANI with EWA and extra connections. The parameters  $T = 2000$ ,  $M = 40$ ,  $N = 1357$  and 20 realizations were made for each number of selected neurons. For CHANI with EWA:  $\eta_1 = 3$ ,  $\eta_2 = 0.002$ . For CHANI with PWA:  $b_1 = 2$ ,  $b_2 = 2$ . For CHANI with EWA and extra connections:  $\eta_1 = 3$ ,  $\eta_2 = 0.007$ ,  $\alpha = 0.7$  and  $\beta = 0.25$ .

CHANI with EWA increases steadily with the number of selected neurons, to station at around 83.5%, and exceeds 80% from 80 selected neurons.

The numerical results with two hidden layers and EWA can be seen in Table 2. Here, we fixed the number of selected neurons of the first hidden layer and varied the number of selected neurons of the second one. We can see that in all cases, the results, comparable to CHANI with no hidden layer, are far less good than with one hidden layer. There are several possible explanations. A good representation of classes could be combination of correlation between two features and not three (*i.e.*, a digit could be well represented by combinations of pixel tuples and not triplets). Numerical results could also be affected by the phenomenon of decreasing spiking probability from one layer to the next: there may not be enough spikes in the system in order to have a second hidden layer with meaningful cumulated gains. Another reason could be that this dataset does not respect all the assumptions that we made in order to have theoretical guarantees.

**Addition of connections.** In order to improve CHANI performance, we added connections between input and output neurons in the case  $L = 1$ . Indeed, a digit could be well represented by the combination of single pixels and pixel tuples. To do this, we extend the gains of hidden neurons w.r.t. output neurons defined by (3) to input neurons as well, and we add a multiplicative parameter  $\alpha > 0$  in order to find balance between hidden and input neurons gains. Then the gain of

Number of selected neurons on the second hidden layer	Accuracy
50	44.6
100	49.7
150	51.0

Table 2: Numerical results with two hidden layers for CHANI with EWA. The parameters are  $T = 2000$ ,  $M = 40$ ,  $N = 1277$ ,  $\eta^k = 0.0005$ . A number 100 of hidden neurons of the first hidden layer are selected and 10 of realizations were made.

an input neuron  $i$  w.r.t. output neuron  $k$  when presented with object  $o_m^{L+1}$  is

$$g_m^{j \rightarrow k} := \begin{cases} \alpha \langle X_{m,t}^{i,L+1} \rangle_{t \in [T]} \times \frac{N}{N^k} & \text{if } o(m) \in k \\ -\alpha \langle X_{m,t}^{i,L+1} \rangle_{t \in [T]} \times \frac{N}{N^{k'}} \times \frac{1}{|K|-1} & \text{if } o(m) \in k' \neq k \end{cases}.$$

Indeed, as the neurons spiking probabilities decreases from one layer to the next, the empirical spiking probabilities of neurons from distinct layers may have very different order of magnitude. For the same reason, we introduce a parameter  $\beta \in [0, 1]$  as well in the formula of the conditional spiking probability of an output neuron  $k$  when presented with object  $o_m^{L+1}$ :

$$p_{m,t}^k(w_m^k) = w_m^{\hat{j}_L \rightarrow k} \cdot X_{m,t-1}^{\hat{j}_1, L+1} + \beta w_m^{I \rightarrow k} \cdot X_{m,t-1}^{I, L+1}$$

where  $w_m^k := (w_m^{l \rightarrow k})_{l \in I \cup \hat{j}_L}$ ,  $w_m^{\hat{j}_L \rightarrow k} := (w_m^{j \rightarrow k})_{j \in \hat{j}_L}$  and  $w_m^{I \rightarrow k} := (w_m^{i \rightarrow k})_{i \in I}$ . This parameter  $\beta$  allows to reduce the impact of input neurons, which spike more frequently than hidden neurons, on the behavior of output neurons.

The numerical results with this configuration and the expert aggregation EWA correspond to the green curve of Figure 4. We can see that CHANI performs better with these new connections: with only 10 selected neurons, its accuracy on the testing set is already at 76.6% and it grows reach 87% for 200 selected neurons, and exceeds 84% from 70 selected neurons.

**Comparison with STDP:** The same dataset has been used to train spiking neural networks with STDP by Rybka et al. (2024); Sboev et al. (2022). Their results vary from 83% to 95% of accuracy depending on the network settings: in some of them they are comparable to CHANI’s performance. Note that these models are more complex and do not have any theoretical guarantees.

To conclude, CHANI with EWA succeeds the learning task with reasonable results. This is a huge improvement in comparison to our numerical results of Jaffard et al. (2024), where we trained HAN on a simulated dataset of 9 identically

repeating objects. By adding layers and taking neuronal synchronization into account, CHANI can now learn more realistic tasks.

## Conclusion

In this paper, we introduced our cognitive network, CHANI (Correlation-based Hawkes Aggregation of Neurons with bio-Inspiration), which provably learns to classify objects, and we presented theoretical insights into CHANI’s learning capabilities with any number of hidden layers. First, we interpreted regret bounds resulting from the use of expert aggregation algorithms as learning rule from a learning perspective. Subsequently, in the specific framework CHANI EWA, we demonstrated that at the limit hidden neurons encode feature correlations, and we proved that our neuron selection method prioritizes those encoding the most significant correlations. Furthermore, through an analysis of the asymptotic behavior of output neurons, we demonstrated CHANI’s ability to learn the correct classification of objects on average and even asymptotically when classes are defined by feature correlations. Lastly, we calculated the VC-dimension of our network. To our knowledge, this study is the first one to establish that local learning rules enable a biologically inspired network to learn to recognize complex concepts by forming neuronal assemblies, inducing global learning. However, several challenging research avenues remain unexplored. In order to make our model more realistic and computationally efficient, one interesting direction is investigating CHANI’s behavior when objects are not presented for a fixed duration, but only until one output neuron has spiked significantly more than the others. This would correspond to a reaction time and would have a cognitive interpretation. Another innovative line of research is extending these findings by establishing regret bounds for other local rules, such as STDP (Spike-Timing-Dependent Plasticity).

## Acknowledgments and Disclosure of Funding

This research was supported by the French government, through CNRS (eXplAIIn team), the UCA *Jedi* and 3iA Côte d’Azur Investissements d’Avenir managed by the National Research Agency (ANR-15 IDEX-01 and ANR-19-P3IA-0002), directly by the ANR project ChaMaNe (ANR-19-CE40-0024-02) and GraVa (ANR-18-CE40-0005), and finally by the interdisciplinary Institute for Modeling in Neuroscience and Cognition (NeuroMod).

All the numerical results can be reproduced thanks to the code available at <https://github.com/SophieJaffard/CHANI>.

## Appendix A. Notations

Table 3: Table of notations

Notation	Description
$\mathcal{O}$	set of objects
$o$	nature of an object
$M$	number of objects presented to the network to train a hidden layer
$N$	number of objects presented to the network to train the output layer
$m$	index of an object
$T$	number of time steps during which one object is presented
$I$	set of features and input neurons
$i$	index of an input neuron or a feature
$K$	set of classes and output neurons
$k$	index of an output neuron or a class
$L$	number of hidden layers
$l$	depth of a hidden layer
$I^{\vee l}$	set of every tensor of features of depth $l$
$J_l$	set of hidden neurons of depth $l$ before selection or arbitrary set
$\hat{J}_l$	set of selected hidden neurons of depth $l$
$\tilde{J}_l$	set of hidden neurons of interest of depth $l$
$j = j_1 \vee j_2$	index of a hidden neuron and tensor
$S(j)$	support of neuron $j$
$o_m^l$	nature of the $m^{\text{th}}$ object of the training phase of layer $l$
$\mathcal{P}_E$	set of probability distributions over the set $E$

## Appendix B. Proofs of section 3

### B.1 Proof of Proposition 6

Let  $l \in [L]$ ,  $j = j_1 \vee j_2 \in J_l$ . Then

$$\begin{aligned}
 \sum_{m=1}^M w_m^j \cdot g_m^j &= \sum_{m=1}^M w_m^j \cdot (\hat{\rho}_m^l(\{j_1, j_2, j'\})_{j' \in \hat{J}_{l-1}}) \\
 &= \sum_{m=1}^M w_m^j \cdot (\langle X_{m,t}^{j_1} X_{m,t}^{j_2} X_{m,t}^{j'} \rangle_{t \in [T]})_{j' \in \hat{J}_{l-1}} \\
 &= \sum_{m=1}^M \langle X_{m,t}^{j_1} X_{m,t}^{j_2} (w_m^j \cdot X_{m,t}^{\hat{J}_{l-1}}) \rangle_{t \in [T]}
 \end{aligned}$$

and

$$\sum_{m=1}^M w_m^j \cdot g_m^j + \sum_{m=1}^M \langle -\nu X_{m,t}^{j_1} X_{m,t}^{j_2} \rangle_{t \in [T]} = \sum_{m=1}^M \langle X_{m,t}^{j_1} X_{m,t}^{j_2} \psi_{m,t}(w_m^j) \rangle_{t \in [T]}.$$

Similarly, for  $q \in \mathcal{P}_{\hat{j}_{l-1}}$ ,

$$\sum_{m=1}^M q \cdot g_m^j + \sum_{m=1}^M \langle -\nu X_{m,t}^{j_1} X_{m,t}^{j_2} \rangle_{t \in [T]} = \sum_{m=1}^M \langle X_{m,t}^{j_1} X_{m,t}^{j_2} \psi_{m,t}(q) \rangle_{t \in [T]}.$$

Hence the regret of neuron  $j$  is  $R_M^j = \max_{q \in \mathcal{P}_{\hat{j}_{l-1}}} \sum_{m=1}^M \langle X_{m,t}^{j_1} X_{m,t}^{j_2} (\psi_{m,t}(q) - \psi_{m,t}(w_m^j)) \rangle_{t \in [T]}$ . The gains take value in  $[0, 1]$  so we get the bound by applying the regret bound of the expert aggregation given by Assumption 5.

## B.2 Proof of Proposition 11

Let  $k \in K$ ,  $j \in \hat{J}_L$ . Let  $\text{Disc}_N^k(w_{1:M}^k) := \frac{1}{N} \sum_{m=1}^N w_m^k \cdot g_m^k$ . Then

$$\begin{aligned} \text{Disc}_N^k(w_{1:M}^k) &= \frac{1}{N^k} \sum_{m, o_m^{L+1} \in k} w_m^k \cdot \langle X_{m,t}^{j, L+1} \rangle_{t \in [T]} \\ &\quad - \left\langle \frac{1}{N^{k'}} \sum_{m, o_m^{L+1} \in k'} w_m^k \cdot \langle X_{m,t}^{j, L+1} \rangle_{t \in [T]} \right\rangle_{k' / k' \neq k} \end{aligned}$$

and similarly for  $q^k \in \mathcal{P}_{\hat{j}_{L-1}}$ , we have  $\frac{1}{N} \sum_{m=1}^N q^k \cdot g_m^k = \text{Disc}_N^k(q^k)$ . So the regret of neuron  $k$  divided by  $N$  is

$$\frac{R_N^k}{N} = \max_{q^k \in \mathcal{P}_{\hat{j}_{L-1}}} \text{Disc}_N^k(q^k) - \text{Disc}_N^k(w_{1:M}^k) \leq \frac{C|K|}{\xi(|K|-1)} \sqrt{\frac{\ln(|\hat{J}_L|)}{N}} \quad (4)$$

according to Assumption 10, because the gains take value in  $[-\frac{1}{\xi(|K|-1)}, \frac{1}{\xi}]$ . Then

$$\begin{aligned} \text{Disc}_N(w_{1:N}^K) &= \left\langle \hat{p}_m^k(w_m^k) - \hat{p}_m^{k'}(w_m^{k'}) \right\rangle_{\substack{k \in K \\ k' \neq k \\ m, o_m^{L+1} \in k}} \\ &= \left\langle \hat{p}_m^k(w_m^k) \right\rangle_{\substack{k \in K \\ m, o_m^{L+1} \in k}} - \left\langle \hat{p}_m^{k'}(w_m^{k'}) \right\rangle_{\substack{k \in K \\ k' \neq k \\ m, o_m^{L+1} \in k}} \end{aligned}$$

Let us exchange the name of the indexes  $k$  and  $k'$  in the second term.

$$\text{Disc}_N(w_{1:N}^K) = \left\langle \hat{p}_m^k(w_m^k) \right\rangle_{\substack{k \in K \\ m, o_m^{L+1} \in k}} - \left\langle \hat{p}_m^k(w_m^k) \right\rangle_{\substack{k \in K \\ k' \neq k \\ m, o_m^{L+1} \in k'}}$$

$$= \left\langle \langle \hat{p}_m^k(w_m^k) \rangle_{m, o_m^{L+1} \in k} - \langle \hat{p}_m^k(w_m^k) \rangle_{\substack{k' \neq k \\ m, o_m^{L+1} \in k'}} \right\rangle_{k \in K} = \langle \text{Disc}_N^k(w_{1:N}^k) \rangle_{k \in K}.$$

and similarly  $\text{Disc}_N(q^K) = \langle \text{Disc}_N^k(q^k) \rangle_{k \in K}$ . Therefore, according to (4),

$$\text{Disc}_N(w_{1:N}^K) \geq \max_{q^K \in \mathcal{P}_{\hat{J}_L}^{|K|}} \text{Disc}_N(q^K) - \frac{C|K|}{\xi(|K| - 1)} \sqrt{\frac{\ln(|\hat{J}_L|)}{N}}.$$

## Appendix C. Proofs of section 4

### C.1 Preliminary propositions

**Proposition 29** *Let  $(\Omega, \mathcal{F}, \mathbb{P})$  be a probability space, let  $\mathcal{G} \subset \mathcal{F}$  be a  $\sigma$ -algebra, and let  $E, F$  be  $\mathcal{G}$ -measurable random finite sets.*

*Let  $A, B \in \mathbb{N}^*$ , and let  $(Z_{a,b}^{e \rightarrow f})_{e \in E, f \in F, 1 \leq a \leq A, 1 \leq b \leq B}$  be random variables bounded by 1 such that for every  $f \in F$ ,  $e \in E$  the variables  $(Z_{a,b}^{e \rightarrow f})_{1 \leq a \leq A, 1 \leq b \leq B}$  are independent knowing the  $\sigma$ -algebra  $\mathcal{G}$ . Let  $\alpha > 0$ . Then with probability  $1 - \alpha$ , for all  $e \in E, f \in F$*

$$\left| \sum_{a=1}^A \langle Z_{a,b}^{e \rightarrow f} \rangle_{b \in [B]} - \mathbb{E} \left[ \sum_{a=1}^A \langle Z_{a,b}^{e \rightarrow f} \rangle_{b \in [B]} \mid \mathcal{G} \right] \right| < \sqrt{\frac{A}{2B} \ln \left( \frac{2|E||F|}{\alpha} \right)}.$$

**Proof** Let  $\alpha > 0$ ,  $e \in E$ ,  $f \in F$ . The variables  $(Z_{a,b}^{e \rightarrow f})_{1 \leq a \leq A, 1 \leq b \leq B}$  are independent knowing the  $\sigma$ -algebra  $\mathcal{G}$  and bounded by 1 so according to Hoeffding's inequality, for  $\beta > 0$  and  $\mathcal{G}$ -measurable,

$$\mathbb{P} \left( \left| \sum_{a=1}^A \langle X_{a,b}^{e \rightarrow f} \rangle_{b \in [B]} - \mathbb{E} \left[ \sum_{a=1}^A \langle X_{a,b}^{e \rightarrow f} \rangle_{b=1, \dots, B} \mid \mathcal{G} \right] \right| \geq \beta \mid \mathcal{G} \right) \leq 2e^{-2\beta^2 \frac{B}{A}}.$$

Let

$$D^c := \left\{ \exists e \in E, f \in F, \left| \sum_{a=1}^A \langle X_{a,b}^{e \rightarrow f} \rangle_{b=1, \dots, B} - \mathbb{E} \left[ \sum_{a=1}^A \langle X_{a,b}^{e \rightarrow f} \rangle_{b \in [B]} \mid \mathcal{G} \right] \right| \geq \beta \right\}.$$

Then  $\mathbb{P}(D^c \mid \mathcal{G}) \leq |E||F|2e^{-2\beta^2 \frac{B}{A}}$ . Let us choose  $\beta = \sqrt{\frac{A}{2B} \ln \left( \frac{2|E||F|}{\alpha} \right)}$ . Then  $\mathbb{P}(D^c \mid \mathcal{G}) \leq \alpha$ . By integrating we get  $\mathbb{P}(D^c) \leq \alpha$ . ■

**Proposition 30** *Let  $d \in \mathbb{N}$ ,  $A^1, \dots, A^d \in \mathbb{R}$ . Let  $E$  be the subset of  $[d]$  defined by  $E := \arg \max_i A^i$ , and let  $w_i(M) := \frac{\exp(\sqrt{M}A^i)}{\sum_{l=1}^d \exp(\sqrt{M}A^l)}$ . Let  $A = \max_i A^i$ .*

- If  $E = [d]$  then for every  $i \in [d]$ ,  $w_i(M) = \frac{1}{d}$ .
- If  $E \subsetneq [d]$ , let  $\Delta := \max_i A^i - \max_{i \notin E} A^i$ . Then, for all  $i$ ,

$$\left| w_i(M) - \frac{1}{|E|} \mathbb{1}_{i \in E} \right| \leq \frac{1}{|E|} \max \left( 1, \frac{d - |E|}{|E|} \right) \exp(-\sqrt{M}\Delta).$$

**Proof** Let  $A_{\max} := \max_i A^i$ .

**Case**  $E = [d]$ . Then for all  $i \in [d]$

$$w_i(M) = \frac{\exp(\sqrt{M}A_{\max})}{\sum_{l=1}^d \exp(\sqrt{M}A_{\max})} = \frac{1}{d}.$$

**Case**  $E \subsetneq [d]$ . Let  $A_{\max \text{ bis}} := \max_{i \notin E} A^i$ . Then

$$w_i(M) \leq \frac{\exp(\sqrt{M}A^i)}{|E| \exp(\sqrt{M}A_{\max})} = \frac{1}{|E|} \exp(-\sqrt{M}(A_{\max} - A^i)) \quad (5)$$

Let  $i \in E$ .

$$\begin{aligned} w_i(M) &\geq \frac{\exp(\sqrt{M}A_{\max})}{|E| \exp(\sqrt{M}A_{\max}) + (d - |E|) \exp(\sqrt{M}A_{\max \text{ bis}})} \\ &= \frac{1}{|E| \left( 1 + \frac{d - |E|}{|E|} \exp(-\sqrt{M}(A_{\max} - A_{\max \text{ bis}})) \right)} \\ &\geq \frac{1}{|E|} \left( 1 - \frac{d - |E|}{|E|} \exp(-\sqrt{M}(A_{\max} - A_{\max \text{ bis}})) \right) \\ &= \frac{1}{|E|} - \frac{d - |E|}{|E|^2} \exp(-\sqrt{M}(A_{\max} - A_{\max \text{ bis}})) \end{aligned}$$

And thanks to (5),  $w_i(M) \leq \frac{1}{|E|}$ . Hence

$$\frac{1}{|E|} - \frac{d - |E|}{|E|^2} \exp(-\sqrt{M}(A_{\max} - A_{\max \text{ bis}})) \leq w_i(M) \leq \frac{1}{|E|}.$$

Let  $i \in [d] \setminus E$ . According to (5)

$$0 \leq w_i(M) \leq \frac{1}{|E|} \exp(-\sqrt{M}(A_{\max} - A_{\max \text{ bis}})).$$

■

As proved in (Gao and Pavel, 2018, Proposition 4), we have

**Proposition 31** *Let  $d \in \mathbb{N}$ ,  $\eta \in \mathbb{R}$ . Then the softmax function  $f$  with temperature  $\eta$  defined on  $\mathbb{R}^d$  by  $x \mapsto \text{softmax}(\eta x)$  is  $\eta$ -Lipschitz.*

### C.2 Definition of the network activity and coupled network activity

Let us define more precisely the variables  $X_{m,t}^{j,l}$  and  $X_{o,t}^{j,l}$ . For  $l \in [L]$ , let  $\mathcal{F}_l$  the  $\sigma$ -algebra generated by every event that happened until the end of the selection phase of layer  $l$ .

- For  $i \in I, m \in [M], l \in [L]$ , the variables  $(X_{m,t}^{i,l})_{1 \leq t \leq T}$  are i.i.d and follow a Bernoulli distribution with parameter  $p_o^i$  where  $o_m^l = o$ . Similarly, for  $i \in I, o \in \mathcal{O}, l \in [L]$ , the variables  $(X_{o,t}^{i,l})_{1 \leq t \leq T}$  are i.i.d and follow a Bernoulli distribution with parameter  $p_o^i$ .
- For  $l \in [L], j \in \hat{J}_1 \cup \dots \cup \hat{J}_{l-1} \cup J_l$ , we define conditionally to the past  $\mathcal{F}_{l-1}$  the variables  $(U_{m,t}^{j,l})_{1 \leq m \leq M, 1 \leq t \leq T}$  and  $(U_{o,t}^{j,l})_{o \in \mathcal{O}, 1 \leq t \leq T}$  which are i.i.d and follow a uniform distribution on  $[0, 1]$ . Then, if  $j \notin \tilde{J}_l$  we define its activity by  $X_{m,t}^{j,l} := \mathbb{1}_{p_{m,t}^{j,l}(w_{M+1}^j) \geq U_{m,t}^{j,l}}$ , if  $j \in \tilde{J}_l$  then  $X_{m,t}^{j,l} := \mathbb{1}_{p_{m,t}^{j,l}(w_m^j) \geq U_{m,t}^{j,l}}$ , and in any case  $X_{o,t}^{j,l} := \mathbb{1}_{p_{o,t}^{j,l}(w_{M+1}^j) \geq U_{o,t}^{j,l}}$ .

Let  $l \in [L], l' \leq l$  and  $w_{\text{bis}} := (w_{\text{bis}}^j)_{j \in \hat{J}_1 \cup \dots \cup \hat{J}_{l-1} \cup J_l}$  an  $\mathcal{F}_{l-1}$ -measurable weight family. Then we define the coupled network activity with weights  $w_{\text{bis}}$  as the variables  $Z_{m,t}^{j,l} := \mathbb{1}_{p_{m,t}^{j,l}(w_{\text{bis}}^j) \geq U_{m,t}^{j,l}}$  and  $Z_{o,t}^{j,l} := \mathbb{1}_{p_{o,t}^{j,l}(w_{\text{bis}}^j) \geq U_{o,t}^{j,l}}$ .

### C.3 Proof of Theorem 18

Let  $\alpha > 0$ . Let

$$E_{\text{EWA}}^{j,l}(M, |\tilde{J}_{l-1}|) := \frac{1}{2} \max(1, \frac{|\tilde{J}_{l-1}| - 2}{2}) \exp\left(-\rho(S(j))2^{2-2^l} \sqrt{2M \ln(|\tilde{J}_{l-1}|)}\right).$$

and for  $l \in [L], j = j_1 \vee j_2 \in \tilde{J}_l$ , let  $w_\infty^j := \mathbb{1}_{\{j_1, j_2\}}$ . Let us prove by induction the following proposition for every  $l \in [L]$ .

**Proposition 32** *There exist constants  $C_1^l$  and  $C_2^l$  and events  $(D^{l'})_{l' \leq l}$  and  $(D_{\text{sel}}^{l'})_{l' \leq l}$ , each of probability more than  $1 - \alpha$ , such that if  $T/M^{l-1} \geq C_1^l$  and  $M \geq C_2^l$  then on  $D^1 \cap D_{\text{sel}}^1 \cap \dots \cap D^{l-1} \cap D_{\text{sel}}^l$ , for every  $l' \leq l$ ,  $\hat{J}_{l'} = \tilde{J}_{l'}$  and for all  $j \in \tilde{J}_{l'}, j' \in \tilde{J}_{l'-1}$ ,*

$$\begin{aligned} |w_{M+1}^{j' \rightarrow j} - w_\infty^{j' \rightarrow j}| &\leq E_{\text{approx}}(T, |\tilde{J}_{l'-1}|, |\tilde{J}_{l'-1}^1|, \alpha) + E_{\text{EWA}}^{j,l}(M, |\tilde{J}_{l'-1}|) \\ &\quad + E_{\text{prec}}(M, |\tilde{J}_{l'-1}|, l' - 1) \end{aligned}$$

**Case l=1:** For  $i \in I, j = i_1 \vee i_2 \in J^1$ , we remind that the cumulated gain of neuron  $i$  w.r.t. neuron  $j$  is  $G_M^{i \rightarrow j} = \sum_{m=1}^M \langle X_{m,t}^{i,1} X_{m,t}^{i_1,1} X_{m,t}^{i_2,1} \rangle_{t \in [T]}$ . Let

$$\bar{G}_M^{i \rightarrow j} := \sum_{m=1}^M \mathbb{P}(X_{m,t}^{i,1} = 1, X_{m,t}^{i_1,1} = 1, X_{m,t}^{i_2,1} = 1).$$

The variables  $(X_{m,t}^{i,1} X_{m,t}^{i_1,1} X_{m,t}^{i_2,1})_{1 \leq m \leq M, 1 \leq t \leq T}$  are independent so according to Proposition 29 there exists an event  $D^1$  of probability more than  $1 - \alpha$  on which for all  $j \in J_1, i \in I$

$$|G_M^{i \rightarrow j} - \bar{G}_M^{i \rightarrow j}| \leq \sqrt{\frac{M}{2T} \ln \left( \frac{2|J_1||I|}{\alpha} \right)}.$$

Let  $\bar{w}_{M+1}^{i \rightarrow j} := \frac{\exp(\eta_1 \bar{G}_M^{i \rightarrow j})}{\sum_{i' \in I} \exp(\eta_1 \bar{G}_M^{i' \rightarrow j})}$ . Then according to (31),

$$\|w_{M+1}^j - \bar{w}_{M+1}^j\| \leq \eta_1 \sqrt{\sum_{i' \in I} |G_M^{i' \rightarrow j} - \bar{G}_M^{i' \rightarrow j}|}.$$

Hence on  $D^1$ , with  $\eta_1 = \sqrt{\frac{8 \ln(|I|)}{M}}$  we get

$$\|w_{M+1}^j - \bar{w}_{M+1}^j\|_2 \leq 2 \sqrt{\frac{|I| \ln(|I|)}{T} \ln \left( \frac{2|I||J_1|}{\alpha} \right)} := E_{\text{approx}}(T, |I|, |J_1|, \alpha). \quad (6)$$

Besides, for  $j = i_1 \vee i_2 \in J_1$  and  $i \in I$ ,  $\bar{G}_M^{i \rightarrow j} = \sum_{m=1}^M \rho_{o_m^1}(\{i, i_1, i_2\})$ . According to Assumption 13, each nature of object is presented the same amount of times so

$$\bar{G}_M^{i \rightarrow j} = \frac{M}{|\mathcal{O}|} \sum_{o \in \mathcal{O}} \rho_o(\{i, i_1, i_2\}) = M \rho(\{i, i_1, i_2\}).$$

and  $\eta_1 \bar{G}_M^{i \rightarrow j} = \sqrt{8 \ln(|I|) M} \rho(\{i, i_1, i_2\})$ . We can distinguish two cases.

1. If  $j = i_1 \vee i_2$  is such that  $\rho(\{i_1, i_2\}) = 0$  then for all  $i \in I, \rho(\{i, i_1, i_2\}) = 0$ . Hence according to Proposition 30, we have

$$\bar{w}_{M+1}^{i \rightarrow j} = w_\infty^{i \rightarrow j} \quad (7)$$

where  $w_\infty^{i \rightarrow j} := \frac{1}{|I|}$ . This definition of  $w_\infty^{i \rightarrow j}$  does not conflict with the one given at beginning of the proof because according to the definition of  $\tilde{J}_1$  given in Assumption 16,  $j \notin \tilde{J}_1$ .

2. If  $j = i_1 \vee i_2$  is such that  $\rho(\{i_1, i_2\}) > 0$  then according to Assumption 15,  $\arg \max_{i \in I} \rho(\{i, i_1, i_2\}) = \{i_1, i_2\}$ . Hence according to Proposition 30, we have

$$|\bar{w}_{M+1}^{i \rightarrow j} - w_\infty^{i \rightarrow j}| \leq E_{\text{EWA}}^{j,1}(M, |I|). \quad (8)$$

Hence at the end of the learning phase of layer 1, by combining (6) and (7) we have for all  $j$  such that  $\rho(S(j)) = 0$ :

$$\|w_{M+1}^j - w_\infty^j\|_2 \leq E_{\text{approx}}(T, |I|, |J_1|, \alpha) \quad (9)$$

and by combining (6) and (8) we have for all  $j$  such that  $\rho(S(j)) > 0$ :

$$\|w_{M+1}^j - w_\infty^j\|_2 \leq E_{\text{approx}}(T, |I|, |J_1|, \alpha) + \sqrt{|I|} E_{\text{EWA}}^{j,1}(M, |I|). \quad (10)$$

Now let us study the selection phase. Let  $\mathcal{F}_1^{\text{learn}}$  be the  $\sigma$ -algebra generated by every event that happened until the end of the learning phase of layer 1. Let  $j \in J_1$ . The variables  $(X_{o,t}^{j,1})_{o \in \mathcal{O}, 1 \leq t \leq T}$  are independent knowing the  $\sigma$ -algebra  $\mathcal{F}_1^{\text{learn}}$  since the weights  $w_{M+1}^j$  are frozen and are  $\mathcal{F}_1^{\text{learn}}$ -measurable. Hence according to Proposition 29, there exists a set  $D_{\text{sel}}^1$  of probability more than  $1 - \alpha$  on which for every  $j \in J_1, o \in \mathcal{O}$ ,

$$\left| \langle X_{o,t}^{j,1} \rangle_{t \in [T]} - \mathbb{E}[X_{o,1}^{j,1} \mid \mathcal{F}_1^{\text{learn}}] \right| \leq \sqrt{\frac{1}{2T} \ln \left( \frac{2|J||\mathcal{O}|}{\alpha} \right)}$$

since  $\mathbb{E}[X_{o,t}^{j,1} \mid \mathcal{F}_1^{\text{learn}}]$  does not depend on  $t$ .

Let  $(Z_{o,t}^{j,1})_{j \in J_1, o \in \mathcal{O}, 1 \leq t \leq T}$  be the coupled network activity with weights  $(w_{\infty}^j)_{j \in J_1}$  as defined in section C.2. Then for  $j \in J_1, o \in \mathcal{O}$ ,

$$\mathbb{E}[|X_{o,t}^{j,1} - Z_{o,t}^{j,1}| \mid \mathcal{F}_1^{\text{learn}}] \leq |p_{o,t}^{j,1}(w_{M+1}^j) - p_{o,t}^{j,1}(w_{\infty}^j)| \leq |(w_{M+1}^j - w_{\infty}^j) \cdot X_{o,t}^{j,1}|.$$

By Cauchy-Schwartz inequality and since the variables  $X_{o,t}^{j,1}$  are bounded by 1,

$$\mathbb{E}[|X_{o,t}^{j,1} - Z_{o,t}^{j,1}| \mid \mathcal{F}_1^{\text{learn}}] \leq \|w_{M+1}^j - w_{\infty}^j\|_1 \leq \sqrt{|I|} \|w_{M+1}^j - w_{\infty}^j\|_2 := E_w(1).$$

Therefore, on  $D_{\text{sel}}^1$ , for all  $j \in J_1$  and  $o \in \mathcal{O}$

$$\left| \langle X_{o,t}^{j,1} \rangle_{t \in [T]} - \mathbb{E}[Z_{o,1}^j \mid \mathcal{F}_1^{\text{learn}}] \right| \leq \sqrt{\frac{1}{2T} \ln \left( \frac{2|J||\mathcal{O}|}{\alpha} \right)} + E_w(1).$$

Let  $j = i_1 \vee i_2 \in J_1$ . Let us compute  $\mathbb{E}[Z_{o,1}^j \mid \mathcal{F}_1^{\text{learn}}]$ . If  $\rho(S(j)) > 0$  then

$$\mathbb{E}[Z_{o,1}^j \mid \mathcal{F}_1^{\text{learn}}] = \mathbb{E} \left[ \left( -\frac{1}{2} + \frac{1}{2}(X_{o,1}^{i_1,1} + X_{o,1}^{i_2,1}) \right)_+ \right] = \frac{1}{2} \rho_o(S(j)).$$

If  $\rho(S(j)) = 0$  then  $\mathbb{E}[Z_{o,1}^j \mid \mathcal{F}_1^{\text{learn}}] = \mathbb{E} \left[ \left( -\frac{1}{2} + \frac{1}{|I|} \sum_{i \in I} X_{o,1}^{i,1} \right)_+ \right] = 0 = \frac{1}{2} \rho_o(S(j))$  according to Assumption 17 because less than half the neurons of  $I$  are active when presented with  $o$  and because  $\rho_o(S(j)) = 0$ . Therefore on  $D_{\text{sel}}^1$

$$\left| \langle X_{o,t}^{j,1} \rangle_{t \in [T]} - \frac{1}{2} \rho_o(S(j)) \right| \leq \sqrt{\frac{1}{2T} \ln \left( \frac{2|J||\mathcal{O}|}{\alpha} \right)} + E_w(1).$$

Besides, according to Assumption 16, there exists  $p_1$  such that if  $j \in \tilde{J}_1$  then there exists  $o \in \mathcal{O}$  such that  $\rho_o(S(j)) \geq p_1$  and if  $j \notin \tilde{J}_1$  then for all  $o \in \mathcal{O}$ ,  $\rho_o(S(j)) < p_1$ . Let  $q_1 := \max_{j \in J_1 \setminus \tilde{J}_1, o \in \mathcal{O}} \rho_o(S(j))$ . Let us choose the threshold  $s_1 := \frac{1}{2}(\frac{1}{2}p_1 + \frac{1}{2}q_1)$ . Then, on  $D^1 \cap D_{\text{sel}}^1$ ,

$$\left| \langle X_{o,t}^{j,1} \rangle_{t \in [T]} - \frac{1}{2} \rho_o(S(j)) \right| = O\left(T^{-1/2} + e^{-c\sqrt{M}}\right)$$

so there exist constants  $C_1^1$  and  $C_2^1$  independent of  $M, N$  and  $T$  such that for  $T \geq C_1^1$  and  $M \geq C_2^1$  we have for all  $j \in \tilde{J}_1$ ,  $o \in \mathcal{O}$

$$\left| \langle X_{o,t}^{j,1} \rangle_{t \in [T]} - \frac{1}{2} \rho_o(S(j)) \right| < \frac{1}{2} \left( \frac{1}{2} p_1 - \frac{1}{2} q_1 \right).$$

so if  $j \in \tilde{J}_1$  then  $\langle X_{o,t}^{j,1} \rangle_{t \in [T]} \geq s_1$  and otherwise  $\langle X_{o,t}^{j,1} \rangle_{t \in [T]} < s_1$ . Hence with this choice of threshold, on  $D^1 \cap D_{\text{sel}}^1$  we have  $\hat{J}_1 = \tilde{J}_1$  and Proposition 32 is true for rank 1.

**Case  $l > 1$  :** Suppose Proposition 32 is true for rank  $l-1$ . Let  $D^1, D_{\text{sel}}^1, \dots, D^{l-1}, D_{\text{sel}}^{l-1}$  the events and  $C_1^{l-1}, C_2^{l-1}$  the constants given by the proposition at rank  $l-1$ . Suppose  $\frac{T}{M^{l-2}} \geq C_1^{l-1}$  and  $M \geq C_2^{l-1}$ . Then the conclusions of Proposition 32 hold for rank  $l-1$ .

We remind that for  $j = j_1 \vee j_2 \in J_l$ ,  $j' \in \hat{J}_{l-1}$ , the cumulated gain of neuron  $j'$  w.r.t neuron  $j$  is  $G_M^{j' \rightarrow j} = \sum_{m=1}^M \langle X_{m,t}^{j_1,l} X_{m,t}^{j_2,l} X_{m,t}^{j',l} \rangle_{t \in [T]}$ . Let

$$\mathring{G}_M^{j' \rightarrow j} := \sum_{m=1}^M \mathbb{E}[X_{m,1}^{j_1,l} X_{m,1}^{j_2,l} X_{m,1}^{j',l} \mid \mathcal{F}_{l-1}].$$

The variables  $(X_{m,t}^{j_1,l} X_{m,t}^{j_2,l} X_{m,t}^{j',l})_{1 \leq m \leq M, 1 \leq t \leq T}$  are independent knowing  $\mathcal{F}_{l-1}$ . According to Proposition 29, there exists a set  $D^l$  of probability greater than  $1 - \alpha$  on which for all  $j \in \tilde{J}_l, j' \in \hat{J}_l$ ,

$$|G_M^{j' \rightarrow j} - \mathring{G}_M^{j' \rightarrow j}| \leq \sqrt{\frac{M}{2T} \ln \left( \frac{2|\hat{J}_{l-1}||J_l|}{\alpha} \right)} \quad (11)$$

since  $\mathbb{E}[X_{m,t}^{j_1,l} X_{m,t}^{j_2,l} X_{m,t}^{j',l} \mid \mathcal{F}_{l-1}]$  does not depend on  $t$ .

Let  $(Z_{m,t}^{j,l})_{j \in \hat{J}_1 \cup \hat{J}_{l-1}, 1 \leq m \leq M, 1 \leq t \leq T}$ ,  $(Z_{o,t}^{j,l})_{j \in \hat{J}_{l-1}, o \in \mathcal{O}, 1 \leq t \leq T}$  be the coupled network activity with arbitrary  $\mathcal{F}_{l-1}$ -measurable weights  $(w_{\star}^j)_{j \in \hat{J}_1 \cup \dots \cup \hat{J}_{l-1}}$  as defined in section C.2. For  $j = j_1 \vee j_2 \in J_l$ ,  $j' \in \hat{J}_{l-1}$ , let  $\bar{G}_M^{j' \rightarrow j} := \sum_{m=1}^M \mathbb{E}[Z_{m,1}^{j_1,l} Z_{m,1}^{j_2,l} Z_{m,1}^{j',l} \mid \mathcal{F}_{l-1}]$ . Then since the variables are bounded by 1, we have

$$\begin{aligned} & |\mathring{G}_M^{j' \rightarrow j} - \bar{G}_M^{j' \rightarrow j}| \leq \\ & \sum_{m=1}^M \left( \mathbb{E}[|X_{m,t}^{j_1,l} - Z_{m,t}^{j_1,l}| \mid \mathcal{F}_{l-1}] + \mathbb{E}[|X_{m,t}^{j_2,l} - Z_{m,t}^{j_2,l}| \mid \mathcal{F}_{l-1}] + \mathbb{E}[|X_{m,t}^{j',l} - Z_{m,t}^{j',l}| \mid \mathcal{F}_{l-1}] \right). \end{aligned} \quad (12)$$

Let  $j'' \in \{j', j_1, j_2\}$ . Note that  $j'' \in \tilde{J}_{l-1}$ . We have

$$\mathbb{E}[|X_{m,t}^{j'',l} - Z_{m,t}^{j'',l}| \mid \mathcal{F}_{l-1}] \leq \mathbb{E}[|p_{m,t}^{j'',l}(w_{M+1}^{j''}) - p_{m,t}^{j'',l}(w_{\infty}^{j''})| \mid \mathcal{F}_{l-1}]$$

$$\begin{aligned}
 &\leq \mathbb{E}[\|w_{M+1}^{j''} \cdot X_{m,t-1}^{\tilde{J}_{l-1,l}} - w_{\star}^{j''} \cdot Z_{m,t-1}^{\tilde{J}_{l-1,l}}\| \mid \mathcal{F}_{l-1}] \\
 &\leq \mathbb{E}[\|w_{M+1}^{j''} \cdot X_{m,t-1}^{\tilde{J}_{l-1,l}} - w_{\star}^{j''} \cdot X_{m,t-1}^{\tilde{J}_{l-1,l}}\| \mid \mathcal{F}_{l-1}] + \mathbb{E}[\|w_{\star}^{j''} \cdot X_{m,t-1}^{\tilde{J}_{l-1,l}} - w_{\star}^{j''} \cdot Z_{m,t-1}^{\tilde{J}_{l-1,l}}\| \mid \mathcal{F}_{l-1}] \\
 &\leq \|w_{M+1}^{j''} - w_{\star}^{j''}\|_1 + \max_{a \in \tilde{J}_{l-2}} \mathbb{E}[\|X_{m,t}^{a,l} - Z_{m,t}^{a,l}\| \mid \mathcal{F}_{l-1}]
 \end{aligned}$$

because the variables  $X_{m,t-1}^{\tilde{J}_{l-1,l}} = (X_{m,t-1}^{a,l})_{a \in \tilde{J}_{l-1}}$  are bounded by 1 and the weights  $w_{\star}^{j''}$  and  $w_{M+1}^{j''}$  are  $\mathcal{F}_{l-1}$ -measurable and bounded by 1. By iteration we get

$$\mathbb{E}[\|X_{m,t}^{j'',l} - Z_{m,t}^{j'',l}\| \mid \mathcal{F}_{l-1}] \leq \sum_{l'=1}^{l-1} \max_{a \in \tilde{J}_{l'}} \|w_{M+1}^a - w_{\star}^a\|_1 \quad (13)$$

Let us now work on the event  $D^1 \cap D_{\text{sel}}^1 \cap \dots \cap D^{l-1} \cap D_{\text{sel}}^{l-1} \cap D^l$ . Then for every  $l' \leq l-1$ ,  $\tilde{J}_{l'} = \tilde{J}_{l'}$ . For every  $a \in \tilde{J}_{l'}$ , let us choose  $w_{\star}^a = w_{\infty}^a$ . Then

$$\mathbb{E}[\|X_{m,t}^{j'',l} - Z_{m,t}^{j'',l}\| \mid \mathcal{F}_{l-1}] \leq \sum_{l'=1}^{l-1} E_w(l') \quad (14)$$

where  $E_w(l') := \max_{a \in \tilde{J}_{l'}} \sqrt{\tilde{J}_{l'-1}} \|w_{M+1}^a - w_{\infty}^a\|_2$  and we can control the  $E_w(l')$ . Then by combining (12) and (14) we get

$$|\hat{G}_M^{j' \rightarrow j} - \tilde{G}_M^{j' \rightarrow j}| \leq 3M \sum_{l'=1}^{l-1} E_w(l'). \quad (15)$$

For  $j \in J_l$ ,  $j' \in \tilde{J}_{l-1}$ , let  $\tilde{w}_{M+1}^{j' \rightarrow j} := \frac{\exp(\eta^j \tilde{G}_M^{j' \rightarrow j})}{\sum_{j'' \in \tilde{J}_{l-1}} \exp(\eta^j \tilde{G}_M^{j'' \rightarrow j})}$ . According to Prop. 31,

$$\begin{aligned}
 \|w_{M+1}^j - \tilde{w}_{M+1}^j\|_2 &\leq \eta^j \|(\tilde{G}_M^{j' \rightarrow j})_{j' \in \tilde{J}_{l-1}} - (G_M^{j' \rightarrow j})_{j' \in \tilde{J}_{l-1}}\|_2 \\
 &\leq \eta^j \left( \|(\hat{G}_M^{j' \rightarrow j})_{j' \in \tilde{J}_{l-1}} - (G_M^{j' \rightarrow j})_{j' \in \tilde{J}_{l-1}}\|_2 + \|(\tilde{G}_M^{j' \rightarrow j})_{j' \in \tilde{J}_{l-1}} - (\hat{G}_M^{j' \rightarrow j})_{j' \in \tilde{J}_{l-1}}\|_2 \right)
 \end{aligned}$$

and by combining (11) and (15), with  $\eta^j = \sqrt{\frac{8 \ln(|\tilde{J}_{l-1}|)}{M}}$  we get

$$\|w_{M+1}^j - \tilde{w}_{M+1}^j\|_2 \leq E_{\text{prec}}(M, |\tilde{J}_{l-1}|, l-1) + E_{\text{approx}}(T, |\tilde{J}_{l-1}|, |\tilde{J}_{l-1}^{\vee 1}|, \alpha) \quad (16)$$

where

$$E_{\text{prec}}(M, |\tilde{J}_{l-1}|, l-1) := 6 \sqrt{2 \ln(|\tilde{J}_{l-1}|)} M \sum_{l'=1}^{l-1} E_w(l')$$

and

$$E_{\text{approx}}(T, |\tilde{J}_{l-1}|, |\tilde{J}_{l-1}^{\vee 1}|, \alpha) := 2 \sqrt{\frac{|\tilde{J}_{l-1}| \ln(|\tilde{J}_{l-1}|)}{T} \ln \left( \frac{2|\tilde{J}_{l-1}| |J_l|}{\alpha} \right)}.$$

Let us study  $\bar{G}_M^{j' \rightarrow j}$ . For  $j'' = a \vee b \in \{j', j_1, j_2\}$ , the conditional spiking probability of  $Z_{m,t}^{j'',l}$  is  $\left(-\frac{1}{2} + \frac{1}{2}(Z_{m,t-1}^{a,l} + Z_{m,t-1}^{b,l})\right)_+$ . Then  $j''$  spikes with probability  $\frac{1}{2}$  if and only if neurons  $a$  and  $b$  spiked before. Let  $j_1 = a_1 \vee a_2$ ,  $j_2 = b_1 \vee b_2$ ,  $j' = d_1 \vee d_2$ . Let  $c_{l'}$  is the number of distinct variables of layer  $l'$  which intervene in the tensors  $j_1$ ,  $j_2$  and  $j'$ .

$$\begin{aligned}
 \bar{G}_M^{j' \rightarrow j} &= \sum_{m=1}^M \mathbb{E}[Z_{m,t}^{j_1,l} Z_{m,t}^{j_2,l} Z_{m,t}^{j',l} \mid \mathcal{F}_{l-1}] \\
 &= \sum_{m=1}^M \mathbb{E}\left[\frac{1}{2^{c_{l-1}^{j' \rightarrow j}}} Z_{m,t}^{a_1,l} Z_{m,t}^{a_2,l} Z_{m,t}^{b_1,l} Z_{m,t}^{b_2,l} Z_{m,t}^{d_1,l} Z_{m,t}^{d_2,l} \mid \mathcal{F}_{l-1}\right] \\
 &\dots \\
 &= \sum_{m=1}^M 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \mathbb{E}\left[\prod_{i \in S(j_1) \cup S(j_2) \cup S(j)} X_{m,t}^{i,l} \mid \mathcal{F}_{l-1}\right] \\
 &= \sum_{m=1}^M 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \prod_{i \in S(j_1) \cup S(j_2) \cup S(j)} \mathbb{E}[X_{m,t}^{i,l} \mid \mathcal{F}_{l-1}] \\
 &= \sum_{m=1}^M 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \prod_{i \in S(j_1) \cup S(j_2) \cup S(j)} \mathbb{E}[X_{m,t}^{i,l}] \\
 &= \sum_{m=1}^M 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \rho_m(S(j_1) \cup S(j_2) \cup S(j'))
 \end{aligned}$$

because the variables  $X_{m,t}^{i,l}$  are mutually independent knowing  $\mathcal{F}_{l-1}$  and are independent of  $\mathcal{F}_{l-1}$ . Therefore, according to Assumption 13,

$$\bar{G}_M^{j' \rightarrow j} = 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \rho(S(j') \cup S(j))M.$$

There are two cases. If  $\rho(S(j)) = 0$  then for all  $j' \in \tilde{J}_{l-1}$ ,  $\bar{G}_M^{j' \rightarrow j} = 0$ . Let  $w_\infty^{j' \rightarrow j} := \frac{1}{|\tilde{J}_{l-1}|}$ . This definition does not conflict with the one given at the beginning of the proof because  $j \notin \tilde{J}_l$  according to  $\tilde{J}_l$  defined in Assumption 16.

Then for all  $j' \in \tilde{J}_{l-1}$ ,

$$\bar{w}_{M+1}^{j' \rightarrow j} = w_\infty^{j' \rightarrow j}. \quad (17)$$

If  $\rho(S(j)) > 0$  then according to the definition of  $J_l$ ,  $S(j_1) \cap S(j_2) = \emptyset$  so for  $j' \in \{j_1, j_2\}$ ,  $c_{l'}^{j' \rightarrow j} = 2^{l-l'}$  for any  $l' \leq l-1$ . So  $\bar{G}_M^{j' \rightarrow j} = 2^{-2^l+2} \rho(S(j))M$ . If  $j' \notin \{j_1, j_2\}$  then  $c_{l-1}^{j' \rightarrow j} = 3$ , otherwise  $c_{l-1}^{j' \rightarrow j} = 2$ . Besides, it is clear that if  $j' \in \{j_1, j_2\}$  and  $j'' \notin \{j_1, j_2\}$  then for every  $l' \leq l-2$ ,  $c_{l'}^{j' \rightarrow j} \leq c_{l'}^{j'' \rightarrow j}$ . In

addition,  $\rho(S(j)) \geq \rho(S(j') \cup S(j))$  for any  $j' \in \tilde{J}_{l-1}$ . Therefore for  $j' \neq j_1, j_2$ ,  $\bar{G}_M^{j' \rightarrow j} \leq 2^{-2^{l+1}} \rho(S(j))M$ . Hence

$$\arg \max_{j' \in \tilde{J}_{l-1}} 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \rho(S(j') \cup S(j))M = \{j_1, j_2\}$$

and

$$\begin{aligned} & \max_{j' \in \tilde{J}_{l-1}} 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \rho(S(j') \cup S(j)) \\ & - \max_{j' \neq j_1, j_2} 2^{-(c_{l-1}^{j' \rightarrow j} + c_{l-2}^{j' \rightarrow j} + \dots + c_1^{j' \rightarrow j})} \rho(S(j') \cup S(j)) \geq 2^{-2^{l+1}} \rho(S(j)). \end{aligned}$$

Let  $w_\infty^{j' \rightarrow j} := \mathbb{1}_{j' \in j_1 \vee j_2}$ . According to Proposition 30, with  $\eta^j = \sqrt{\frac{8 \ln(|\tilde{J}_{l-1}|)}{M}}$  we have for all  $j' \in \tilde{J}_{l-1}$

$$|\bar{w}_{M+1}^{j' \rightarrow j} - w_\infty^{j' \rightarrow j}| \leq E_{\text{EWA}}^{j,l}(M, |\tilde{J}_{l-1}|). \quad (18)$$

Therefore, for  $j = j_1 \vee j_2 \in J_l$ ,

- if  $\rho(S(j)) = 0$  then by combining (17) and (16) we get

$$\|w_{M+1}^j - w_\infty^j\|_2 \leq E_{\text{prec}}(M, |\tilde{J}_{l-1}|, l-1) + E_{\text{approx}}(T, |\tilde{J}_{l-1}|, |\tilde{J}_{l-1}^{\vee 1}|, \alpha).$$

- if  $\rho(S(j)) > 0$  then by combining (16) and (18) we get

$$\begin{aligned} \|w_{M+1}^j - w_\infty^j\|_2 & \leq E_{\text{prec}}(M, |\tilde{J}_{l-1}|, l-1) + E_{\text{approx}}(T, |\tilde{J}_{l-1}|, |\tilde{J}_{l-1}^{\vee 1}|, \alpha) + \\ & \sqrt{|\tilde{J}_{l-1}| E_{\text{EWA}}^{j,l}(M, |\tilde{J}_{l-1}|)}. \end{aligned}$$

Now let us study the selection phase. Let  $\mathcal{F}_l^{\text{learn}}$  be the  $\sigma$ -algebra generated by every event that happened until the end of the learning phase of layer  $l$ . Let  $j \in J_l$ . The variables  $(X_{o,t}^{j,l})_{o \in \mathcal{O}, 1 \leq t \leq T}$  are independent knowing the  $\sigma$ -algebra  $\mathcal{F}_l^{\text{learn}}$  since the weights  $w_{M+1}^j$  are frozen and are  $\mathcal{F}_l^{\text{learn}}$ -measurable. Hence according to Proposition 29, there exists a set  $D_{\text{sel}}^l$  of probability more than  $1 - \alpha$  on which for every  $j \in J_l, o \in \mathcal{O}$ ,

$$\left| \langle X_{o,t}^{j,l} \rangle_{t \in [T]} - \mathbb{E}[X_{o,1}^{j,l} \mid \mathcal{F}_1^{\text{learn}}] \right| \leq \sqrt{\frac{1}{2T} \ln \left( \frac{2|J_l||\mathcal{O}|}{\alpha} \right)}$$

since  $\mathbb{E}[X_{o,t}^{j,l} \mid \mathcal{F}_1^{\text{learn}}]$  does not depend on  $t$ .

Let  $(Z_{o,t}^{j,l})_{j \in \hat{J}_1 \cup \dots \cup \hat{J}_{l-1} \cup J_l, o \in \mathcal{O}, 1 \leq t \leq T}$  be the coupled network activity with any weights  $(w_\star^j)$  which are  $\mathcal{F}_l^{\text{learn}}$ -measurable as defined in section C.2. Then for  $j \in J_l, o \in \mathcal{O}$ , similarly as in (13) we get

$$\mathbb{E}[\|X_{o,1}^{j,l} - Z_{o,1}^{j,l}\| \mid \mathcal{F}_1^{\text{learn}}] \leq \sum_{l'=1}^{l-1} \max_{a \in J_{l'}} \|w_{M+1}^a - w_\star^a\|_1 + \max_{a \in J_l} \|w_{M+1}^a - w_\star^a\|_1.$$

Let us work on  $D^1 \cap D_{\text{sel}}^1 \cap \dots \cap D^l \cap D_{\text{sel}}^l$ . Let  $w_\star^j := w_\infty^j$  for  $j \in \tilde{J}_1 \cup \dots \cup \tilde{J}_{l-1} \cup J_l$ . Then we by Cauchy-Schwartz inequality we have

$$\begin{aligned} \left| \langle X_{o,t}^{j,l} \rangle_{t \in [T]} - \mathbb{E}[Z_{o,1}^{j,l} \mid \mathcal{F}_1^{\text{learn}}] \right| &\leq \sqrt{\frac{1}{2T} \ln \left( \frac{2|J_l||\mathcal{O}|}{\alpha} \right)} + \sum_{l'=1}^l E_w(l') + \\ &\quad \sqrt{|\hat{J}_{l-1}|} \max_{a \in J_l} \|w_{M+1}^a - w_\infty^a\|_2 \end{aligned}$$

where we control every term. Let us study  $\mathbb{E}[Z_{o,1}^{j,l} \mid \mathcal{F}_1^{\text{learn}}]$ . If  $\rho(S(j)) > 0$  then similarly as in the computation of  $\tilde{G}_M^{j' \rightarrow j}$  we get that  $\mathbb{E}[Z_{o,1}^{j,l} \mid \mathcal{F}_l^{\text{learn}}] = 2^{-2^l+1} \rho(S(j))$ . If  $\rho(S(j)) = 0$  then

$$\mathbb{E}[Z_{o,t}^{j,l} \mid \mathcal{F}_l^{\text{learn}}] = \mathbb{E} \left[ \left( -\frac{1}{2} + \frac{1}{|\tilde{J}_{l-1}|} \sum_{j' \in \tilde{J}_{l-1}} Z_{o,t-1}^{j',l} \right)_+ \mid \mathcal{F}_l^{\text{learn}} \right].$$

The computation in the case  $\rho(S(j)) > 0$  also holds for neurons of layer  $l-1$ , which are all in this case because  $\hat{J}_{l-1} = \tilde{J}_{l-1}$ . Therefore, according to Assumption 17, at most half the neurons of layer  $l-1$  are active when presented with  $o$ . Hence

$$\mathbb{E}[Z_{o,1}^{j,l} \mid \mathcal{F}_1^{\text{learn}}] = 0 = 2^{-2^l+1} \rho(S(j)).$$

Therefore for all  $j \in J_l, o \in \mathcal{O}$ ,

$$\begin{aligned} &\left| \langle X_{o,t}^{j,l} \rangle_{t \in [T]} - 2^{-2^l+1} \rho(S(j)) \right| \\ &\leq \sqrt{\frac{1}{2T} \ln \left( \frac{2|J_l||\mathcal{O}|}{\alpha} \right)} + \sum_{l'=1}^l E_w(l') + \sqrt{|\hat{J}_{l-1}|} \max_{a \in J_l} \|w_{M+1}^a - w_\infty^a\|_2 := E(l, M, T) \end{aligned}$$

Besides, according to Assumption 16, there exists  $p_l > 0$  such that if  $j \in \tilde{J}_l$  then there exists  $o \in \mathcal{O}$  such that  $\rho_o(S(j)) \geq p_l$  and if  $j \in J_l \setminus \tilde{J}_l$  then for all  $o \in \mathcal{O}$ ,  $\rho_o(S(j)) < p_l$ . Let  $q_l := \max_{j \in J_l \setminus \tilde{J}_l} \rho_o(S(j))$ . Let us choose the threshold  $s_l := \frac{1}{2}(2^{-2^l+1} p_l + 2^{-2^l+1} q_l)$ . Since  $E(l, M, T) = O\left(\left(\frac{M^{l-1}}{T}\right)^{1/2} + e^{-C\sqrt{M}}\right)$  then for  $\frac{T}{M^{l-1}}$  and  $M$  large enough,

$$E(l, M, T) < \frac{1}{2}(2^{-2^l+1} p_l - 2^{-2^l+1} q_l). \quad (19)$$

Hence there exists constants  $C_1^l$  and  $C_2^l$  such that for  $\frac{T}{M^{l-1}} \geq C_1^l$  and  $M \geq C_2^l$ , (19) is true. Besides,  $\frac{T}{M^{l-2}} \geq \frac{T}{M^{l-1}}$  so let us take  $C_1^l \geq C_1^{l-1}$  and  $C_2^l \geq C_2^{l-1}$  so that we still have  $\frac{T}{M^{l-2}} \geq C_1^{l-1}$  and  $M \geq C_2^{l-1}$ . Under this condition and this choice of threshold, if  $j \in \tilde{J}_l$  then  $\langle X_{o,t}^{j,l} \rangle_{t \in [T]} \geq s_l$  and otherwise  $\langle X_{o,t}^{j,l} \rangle_{t \in [T]} < s_l$  so  $\hat{J}_l = \tilde{J}_l$  and Proposition 32 is true for rank  $l$ .

Now that Proposition 32 is proved for every  $l \in [L]$ , we can directly deduce Theorem 18 by bounding  $|J_l|$  by  $|\tilde{J}_{l-1}|^2$ .

#### C.4 Proof of Proposition 19

Let  $l \in [L]$ ,  $j \in \tilde{J}_l$ ,  $o \in \mathcal{O}$ . With the same calculation as in the proof of Theorem 18 (see section C.3), we have  $\mathbb{E}[p_o^j(w_\infty^j)] = 2^{-2^l+1} \rho(S(j))$ . Therefore for every  $l \in [L]$  the limit layer is an ideal hidden layer with constant  $2^{-2^l+1}$ .

#### C.5 Proof of Corollary 20

Suppose Assumption 10 and the assumptions of Theorem 18 hold. Let  $\alpha > 0$ . Suppose  $T/M^{L-1} \geq C_1$  and  $M \geq C_2$  where  $C_1$  and  $C_2$  are the constants defined in Theorem 18,  $s_1, \dots, s_L$  are the thresholds given by Theorem 18 and  $\tilde{\mathcal{Q}}_L$  is non-empty. Then the conclusions of Theorem 18 hold. Let us use the notations introduced in the proof of Theorem 18.

Let  $k \in K, j \in \hat{J}_L$ . Let us define  $S_N^{j \rightarrow k} := \sum_{m, o_m^{L+1} \in k} \langle X_{m,t}^{j,L+1} \rangle_{t \in [T]}$  and  $\hat{S}_N^{j \rightarrow k} := \sum_{m, o_m^{L+1} \in k} \mathbb{E}[X_{m,1}^{j,L+1} | \mathcal{F}_L]$ . The variables  $(X_{m,t}^{j,L+1})_{1 \leq t \leq T, m \text{ s.t. } o_m^{L+1} \in k}$  are bounded by 1 and independent knowing  $\mathcal{F}_L$ . Hence according to Proposition 29, there exists an event  $D^{L+1}$  of probability more than  $1 - \alpha$  such that on  $D^{L+1}$ , for every  $k \in K, j \in \hat{J}_L$  we have

$$|S_N^{j \rightarrow k} - \hat{S}_N^{j \rightarrow k}| \leq \sqrt{\frac{N^k}{2T} \ln \left( \frac{2|\hat{J}_L||K|}{\alpha} \right)}.$$

Let  $(Z_{m,t}^{j,L+1})_{j \in \hat{J}_1 \cup \dots \cup \hat{J}_L, 1 \leq m \leq M, 1 \leq t \leq T}$  be the coupled network activity with arbitrary weights  $(w_{\star}^j)_{j \in \hat{J}_1 \cup \dots \cup \hat{J}_L}$  which are  $\mathcal{F}_L$ -measurable as defined in section 2.2. For  $k \in K, j \in \hat{J}_L$  let

$$\bar{S}_N^{j \rightarrow k} := \sum_{m, o_m^{L+1} \in k} \mathbb{E}[Z_{m,1}^{j,L+1} | \mathcal{F}_L].$$

Then,

$$|\bar{S}_N^{j \rightarrow k} - \hat{S}_N^{j \rightarrow k}| \leq \sum_{m, o_m^{L+1} \in k} \mathbb{E}[|X_{m,1}^{j,L+1} - Z_{m,1}^{j,L+1}| | \mathcal{F}_L]$$

and similarly as in the proof of Theorem 18 (see section C.3), we have

$$\begin{aligned} |\bar{S}_N^{j \rightarrow k} - \hat{S}_N^{j \rightarrow k}| &\leq \sum_{m, o_m^{L+1} \in k} \sum_{l=1}^L \sqrt{|\hat{J}_l|} \max_{a \in \hat{J}_l} \|w_{M+1}^a - w_\star^a\|_2 \\ &= N^k \sum_{l=1}^L \sqrt{|\hat{J}_l|} \max_{a \in \hat{J}_l} \|w_{M+1}^a - w_\star^a\|_2. \end{aligned}$$

Let us work on  $D := D^1 \cap D_{\text{sel}}^1 \cap \dots \cap D^L \cap D_{\text{sel}}^L \cap D^{L+1}$  where for  $l \in [L]$ ,  $D^l$  and  $D_{\text{sel}}^l$  are the events given in the proof of Theorem 18 (see section C.3). Then  $D$  is of probability greater than  $1 - 2(L+1)\alpha$ . Choose  $w_\star = w_\infty$ . Then, for  $j \in \tilde{J}_L$ ,  $k \in K$ ,

$$|S_N^{j \rightarrow k} - \bar{S}_N^{j \rightarrow k}| \leq \sqrt{\frac{N^k}{2T} \ln \left( \frac{2|\tilde{J}_L||K|}{\alpha} \right)} + N^k \sum_{l=1}^L E_w(l). \quad (20)$$

Here, according to Assumption 13, every nature of object is presented the same amount of time to the network during the learning phase of the output layer. Hence Assumption 9 holds for  $\xi = \frac{1}{|\mathcal{O}|}$ . Therefore, according to Proposition 11,

$$\text{Disc}_N(w_{1:N}^K) \geq \max_{q^K \in (\mathcal{P}_{\tilde{J}_L})^{|K|}} \text{Disc}_N(q^K) - \frac{C|\mathcal{O}||K|}{|K|-1} \sqrt{\frac{\ln(|\tilde{J}_L|)}{N}}.$$

Let  $q^K \in (\mathcal{P}_{\tilde{J}_L})^{|K|}$  be a feasible weight family.

$$\text{Disc}_N(q^K) = \left\langle \hat{p}_m^k(q^k) - \hat{p}_m^{k'}(q^{k'}) \right\rangle_{\substack{k \in K \\ k' \neq k \\ m, o_m^{L+1} \in k}} = \left\langle (q^k - q^{k'}) \cdot \left( \frac{1}{N^k} S_N^{j \rightarrow k} \right)_{j \in \tilde{J}_L} \right\rangle_{\substack{k \in K \\ k' \neq k}}$$

Besides, according to (21) we have  $\mathbb{E}[X_{m,1}^{j,L+1} | \mathcal{F}_L] = 2^{-2L+1} \rho_{o_m^{L+1}}(S(j))$  so

$$\begin{aligned} \left\langle (q^k - q^{k'}) \cdot \left( \frac{1}{N^k} \bar{S}_N^{j \rightarrow k} \right)_{j \in \tilde{J}_L} \right\rangle_{\substack{k \in K \\ k' \neq k}} &= 2^{-2L+1} \left\langle (q^k - q^{k'}) \cdot \rho_o(S(j)) \right\rangle_{\substack{k \in K \\ k' \neq k \\ o \in k}} \\ &= \gamma_L \text{Disc}^{\text{id}}(q^K) \end{aligned}$$

with  $\gamma_L = 2^{-2L+1}$ . Therefore, by lower bounding  $N^k$  by  $\frac{N}{|\mathcal{O}|}$  for every  $k$  we get

$$\text{Disc}_N(q^K) \geq \gamma_L \text{Disc}^{\text{id}}(q^K) - 2\sqrt{\frac{|\mathcal{O}|}{2TN} \ln \left( \frac{2|\tilde{J}_L||K|}{\alpha} \right)} - 2 \sum_{l=1}^L E_w(l).$$

Hence

$$\text{Disc}_N(w_{1:N}^K) \geq \gamma_L \text{Disc}^{\text{id}}(q^K) - 2\sqrt{\frac{|\mathcal{O}|}{2TN} \ln \left( \frac{2|\tilde{J}_L||K|}{\alpha} \right)} - 2 \sum_{l=1}^L E_w(l)$$

$$- \frac{C|\mathcal{O}||K|}{|K|-1} \sqrt{\frac{\ln(|\tilde{J}_L|)}{N}}.$$

### C.6 Proof of Theorem 23

Suppose Assumption 10 and the assumptions of Theorem 18 hold. Let  $\alpha > 0$ . Suppose  $T/M^{L-1} \geq C_1$  and  $M \geq C_2$  where  $C_1$  and  $C_2$  are the constants defined in Theorem 18,  $s_1, \dots, s_L$  are the thresholds given by Theorem 18 and  $\tilde{\mathcal{Q}}_L$  is non-empty. Then the conclusions of Theorem 18 hold. For  $k \in K$ , let  $(w_\infty^{j \rightarrow k})_{j \in \tilde{J}_L} := \frac{1}{|\tilde{J}_L^k|} \mathbb{1}_{|\tilde{J}_L^k|}$ . Let us work on the event  $D$  defined in the proof of Corollary 20 and let us use the notations introduced in section C.5. Then  $G_N^{j \rightarrow k} = \frac{N}{N^k} S_N^k - \left\langle \frac{N}{N^{k'}} S_N^{k'} \right\rangle_{k' \neq k}$ . Let  $\bar{G}_N^{j \rightarrow k} = \frac{N}{N^k} \bar{S}_N^k - \left\langle \frac{N}{N^{k'}} \bar{S}_N^{k'} \right\rangle_{k' \neq k}$ . Then according to (20),

$$\begin{aligned} |G_N^{j \rightarrow k} - \bar{G}_N^{j \rightarrow k}| &\leq N \sqrt{\frac{1}{2TN^k} \ln \left( \frac{2|\tilde{J}_L||K|}{\alpha} \right)} + \left\langle N \sqrt{\frac{1}{2TN^{k'}} \ln \left( \frac{2|\tilde{J}_L||K|}{\alpha} \right)} \right\rangle_{k' \neq k} \\ &\quad + 2N \sum_{l=1}^L E_w(l). \end{aligned}$$

Let us study  $\bar{G}_N^{j \rightarrow k}$ . Similarly as in the proof of Theorem 18 (see section C.3),

$$\mathbb{E}[Z_{m,1}^{j,L+1} | \mathcal{F}_L] = 2^{-2L+1} \rho_{o_m^{L+1}}(S(j)). \quad (21)$$

Then  $\bar{G}_N^{j \rightarrow k} = 2^{-2L+1} \text{Disc}^{j \rightarrow k}$ . Let  $\bar{w}_{N+1}^{j \rightarrow k} := \frac{\exp(\eta^k \bar{G}_N^{j \rightarrow k})}{\sum_{h \in \tilde{J}_L} \exp(\eta^k \bar{G}_N^{h \rightarrow k})}$ . Let  $k \in K$ ,  $E^k := \arg \max_{j \in \tilde{J}_L} \text{Disc}^{j \rightarrow k}$ . Then according to Proposition 30, if  $E^k = \tilde{J}_L$  then for every  $j \in \tilde{J}_L$  we have  $\bar{w}_{N+1}^{j \rightarrow k} = w_\infty^{j \rightarrow k}$ . Otherwise,

$$|\bar{w}_{N+1}^{j \rightarrow k} - w_\infty^{j \rightarrow k}| \leq E_{\text{EWA}}^k(N, |\tilde{J}_L|)$$

where

$$E_{\text{EWA}}^k(N, |\tilde{J}_L|) := \frac{1}{|\tilde{J}_L|} \max \left( 1, \frac{|\tilde{J}_L| - |\tilde{J}_L^k|}{|\tilde{J}_L^k|} \right) \exp \left( -2^{-2L+1} \Delta^k \sqrt{8N \ln(|\tilde{J}_L|)} \right).$$

Let  $\xi(N, |\tilde{J}_L|, |\mathcal{O}|, T, K) = \sqrt{\frac{\ln(|\tilde{J}_L|)}{|\mathcal{O}|T} \ln \left( \frac{2|\tilde{J}_L||K|}{\alpha} \right)} + \frac{1}{|\mathcal{O}|} \sqrt{8|\tilde{J}_L| \ln(|\tilde{J}_L|) N \sum_{l=1}^L E_w(l)}$ .

Besides, according to Proposition 31, with  $\eta^k = \frac{1}{|\mathcal{O}|} \sqrt{\frac{2 \ln(|\tilde{J}_L|)}{N}}$ , by bounding every  $\frac{N}{N^{k'}}$  by  $|\mathcal{O}|$  we get that  $\|w_{N+1}^k - \bar{w}_{N+1}^k\|_2 \leq \xi(N, |\tilde{J}_L|, |\mathcal{O}|, T, K)$ . Hence,

$$\|w_{N+1}^k - w_\infty^k\|_2 \leq \begin{cases} \xi(N, |\tilde{J}_L|, |\mathcal{O}|, T, K) & \text{if } E^k = \tilde{J}_L, \\ \xi(N, |\tilde{J}_L|, |\mathcal{O}|, T, K) + \sqrt{|\tilde{J}_L|} E_{\text{EWA}}^k(N, |\tilde{J}_L|) & \text{otherwise.} \end{cases}$$

### C.7 Proof of Proposition 26

Suppose Assumption 24 (binary correlations) is verified.

Proof of (i)  $\Rightarrow$  (iii). Suppose Assumption 25 (class decomposition) is verified. Let us compute the feature discrepancies of the neurons of  $\tilde{J}_L$  in order to compute the weights  $w_\infty^K$  given in Theorem 23. For every  $k \in K$ , we choose the set  $E^k$  given in Assumption 25 (class decomposition) with maximal size. Let  $j \in \tilde{J}_L$ .

$$\text{Disc}^{j \rightarrow k} = \rho^k(S(j)) - \langle \rho^{k'}(S(j)) \rangle_{k' \neq k} = \langle \rho_o(S(j)) \rangle_{o \in k} - \langle \rho_o(S(j)) \rangle_{k' \neq k, o \in k'}.$$

Then according to Assumption 24 (binary correlations),  $\rho_o(S(j)) = p$  if and only if  $o$  has features  $S(j)$ , *i.e.*, if and only if  $o \in \mathcal{O}^j$ , and  $\rho_o(S(j)) = 0$  otherwise.

Let  $j \in E^k$ . Let  $o \in k$ . We have  $\langle \rho_o(S(j)) \rangle_{o \in k} = \frac{|\mathcal{O}^j|}{n^k} p = \frac{Cp}{n^k}$ . Let  $o \in k' \neq k$ . If  $\rho_o(S(j)) > 0$ , it would mean that  $o$  has features  $S(j)$  so  $o$  would belong to  $\mathcal{O}^j$ , then  $o$  would belong to class  $k$ . This is impossible so  $\rho_o(S(j)) = 0$ . Therefore,  $\text{Disc}^{j \rightarrow k} = \frac{Cp}{n^k}$ . Let  $j \notin E^k$ . This means that there exists  $o \in \mathcal{O}^j$  such that  $o \notin k$ . Therefore  $\langle \rho_o(S(j)) \rangle_{o \in k} \leq \frac{(C-1)p}{n^k}$  and  $\text{Disc}^{j \rightarrow k} \leq \frac{(C-1)p}{n^k}$ . Hence the  $\tilde{J}^k = E^k$  (where  $\tilde{J}^k$  is the set defined in Theorem 23). So according to Theorem 23, for every  $k \in K$   $w_\infty^k = \frac{1}{|E^k|} \mathbb{1}_{j \in E^k}$ .

Let  $(\tilde{J}_L, (p_o^j)_{o \in \mathcal{O}, j \in \tilde{J}_L})$  an ideal hidden layer. Let  $\gamma_L$  its constant. Let  $k \in K$ ,  $o \in \mathcal{O}$ . The spiking probability  $p_o^k$  of neuron  $k$  when presented with object  $o$  is

$$p_o^k = w_\infty^k \cdot p_o^{\tilde{J}_L} = \gamma_L w_\infty^k \cdot (\rho_o(S(j)))_{j \in \tilde{J}_L} = \frac{\gamma_L}{|E^k|} \sum_{j \in E^k} \rho_o(S(j)) = \frac{p\gamma_L n_o^j}{|E^k|},$$

where  $n_o^j$  is the amount of sets  $\mathcal{O}^j$  containing  $o$ . If  $o \in k$ , according to Assumption 25 (class decomposition)  $n_o^j \geq 1$ . If  $o \notin k$  then  $n_o^j = 0$  (otherwise  $o$  would belong to a set  $\mathcal{O}^j$  in the composition of class  $k$  so  $o$  would belong to  $k$ ). Therefore,  $p_o^k > 0$  if, and only if,  $o \in k$  so  $w_\infty^K$  is a strong feasible weight family for the ideal layer  $\tilde{J}_L$ .

Proof of (ii)  $\Rightarrow$  (i). Suppose there exists a strong feasible weight family  $q^K$  w.r.t.  $\tilde{J}_L$ . Let  $k \in K$  and  $E^k := \{j \in \tilde{J}_L, q^{j \rightarrow k} > 0\}$ . Let us show that  $k = \bigcup_{j \in E^k} \mathcal{O}^j$ . The ideal spiking probability of neuron  $k$  when presented with  $o \in \mathcal{O}$  is  $p_o^{k, \text{id}} = \sum_{j \in \tilde{J}_L} q^{j \rightarrow k} \rho_o(S(j))$ . Let  $o \in k$ . Then  $p_o^{k, \text{id}} > 0$  so there exists  $j \in \tilde{J}_L$  such that  $q^{j \rightarrow k} > 0$  and  $\rho_o(S(j)) > 0$ . Hence  $j \in E^k$  and  $o$  has features  $S(j)$  so  $o \in \mathcal{O}^j$ . Therefore  $k \subset \bigcup_{j \in E^k} \mathcal{O}^j$ . Let  $o \in \bigcup_{j \in E^k} \mathcal{O}^j$ . There exists  $j \in E^k$  such that  $o \in \mathcal{O}^j$  and  $q^{j \rightarrow k} > 0$ . Then  $p_o^{k, \text{id}} > 0$  so  $o \in k$ . Therefore,  $\bigcup_{j \in E^k} \mathcal{O}^j \subset k$ . It is obvious that (iii)  $\Rightarrow$  (ii) so we can conclude.

### C.8 Proof of Proposition 27

Let  $j \in J_L$ . Let  $\sigma^j := (o^{i \rightarrow j})_{i \in I}$  where  $o^{i \rightarrow j} = 1$  if and only if  $i \in S(j)$ . Then  $j$  is the only neuron of layer  $J_L$  activated by  $\sigma^j$ . Indeed, all the sets  $S(j')$  for

$j' \in J_L$  have the same size and only a neuron  $j'$  with smaller  $|S(j')|$  can also be activated by  $o^j$ . Let  $S := \{o_j, j \in J_L\}$ . Then  $|S| = |J_L|$  because all the  $o^j$  are distinct. This set can be fully shattered by  $H$ : indeed, let  $S' \subset S$ , and  $J'_L := \{j \in J_L, \exists o \in S', o = o^j\}$ . Then  $S'$  and  $S \setminus S'$  are separated by  $\mathbb{1}_{J'_L}$ .

Let  $S$  be a set of objects that can be fully shattered by  $H$ . In particular, for each  $o \in S$ , there exists  $J'_L \subset J_L$  non empty such that such that  $o$  is the unique object in  $S$  which activates a neuron in  $J'_L$ . Let  $j_o$  be such a neuron. Then  $o \mapsto j_o$  is an injection so  $|S| \leq |J_L|$ .

### C.9 Proof of proposition 28

Let us use the notations of Theorem 23. Let  $j \in \tilde{J}_L$ ,  $k \in K$ . The limit cumulative gain in  $M$  and  $T$  of a hidden neuron is of the form

$$\bar{G}_N^{j \rightarrow k} := CN \left( \langle p_{o, \infty}^j \rangle_{o \in k} - \langle p_{o, \infty}^j \rangle_{k' / k' \neq k} \right)$$

where  $C > 0$  is a constant and  $p_{o, \infty}^j$  is the limit spiking probability of hidden neuron  $j$  when presented with object of nature  $o$ . Since limit hidden layers are not ideal layers, another quantity replaces the feature discrepancy: the quantity

$$c^{j \rightarrow k} := \langle p_{o, \infty}^j \rangle_{o \in k} - \langle p_{o, \infty}^j \rangle_{k' / k' \neq k}$$

and according to Proposition 30, the output weights of neuron  $k \in K$  converge to uniform distribution of weight on the set  $\arg \max_{j \in \tilde{J}_L} c^{j \rightarrow k}$ . In order to compute the limit weights of neuron  $k_2$ , let us compute the quantities  $c^{j \rightarrow k_2}$ . Let  $m \in [N]$ . The limit conditional spiking probability of a hidden neuron  $j = i_1 \vee i_2$  is  $\left( -\nu + \frac{1}{2}(X_{m,t}^{i_1} + X_{m,t}^{i_2}) \right)_+$ . Therefore, when only one of the two input neurons  $i_1$  and  $i_2$  is active, the spiking probability of  $j$  is  $q := \left( \frac{1}{2} - \nu \right) p'$  and when the two neurons  $i_1$  and  $i_2$  are active, the spiking probability of  $j$  is  $p := p'^2(1 - \nu) + 2p'(1 - p')\left(\frac{1}{2} - \nu\right)$ . By definition of the feature discrepancy we have

$$\begin{aligned} d^{\text{Blue}\vee\text{Square} \rightarrow k_2} &= \frac{1}{5}p - q \\ d^{\text{Red}\vee\text{Circle} \rightarrow k_2} &= \frac{1}{5}p + \frac{2}{5}q - \frac{1}{2}q = \frac{1}{5}p - \frac{1}{10}q \\ d^{\text{Blue}\vee\text{Circle} \rightarrow k_2} &= \frac{3}{5}q - \frac{1}{4}p - \frac{1}{4}q = \frac{7}{20}q - \frac{1}{4}p \end{aligned}$$

By symmetry, we know every  $d^{i \rightarrow k_2}$ . Since  $\nu < \frac{1}{2}$  we have  $q > 0$ . Then  $d^{\text{Blue}\vee\text{Square} \rightarrow k_2} < d^{\text{Red}\vee\text{Circle} \rightarrow k_2}$ . Besides,  $d^{\text{Red}\vee\text{Circle} \rightarrow k_2} - d^{\text{Blue}\vee\text{Circle} \rightarrow k_2} = \frac{9}{20}p - \frac{9}{20}q > 0$ , because  $p > q$ . Therefore, the limit weights of  $k_2$  converge to the family

putting weight  $1/4$  on Red $\vee$ Circle, Red $\vee$ Triangle, Green $\vee$ Circle and Green $\vee$ Triangle. Hence, the limit activity of  $k_2$  is equal to zero when presented with  $\square$ , so this objects is not well classified.

## References

- Macrene R Alexiades and Constance L Cepko. Subsets of retinal progenitors display temporally regulated and distinct biases in the fates of their progeny. *Development*, 124(6):1119–1131, 1997.
- Emmanuel Bacry, Iacopo Mastromatteo, and Jean-François Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1(01):1550005, 2015.
- Emmanuel Bacry, Martin Bompaire, Philip Deegan, Stéphane Gaïffas, and Søren V Poulsen. tick: A python library for statistical learning, with an emphasis on hawkes processes and time-dependent models. *JMLR*, 18(1):7937–7941, 2017.
- Yujia Bao, Zhaobin Kuang, Peggy Peissig, David Page, and Rebecca Willett. Hawkes process modeling of adverse drug reactions with longitudinal observational data. In *Proceedings of the 2nd Machine Learning for Healthcare Conference*, volume 68, pages 177–190, 2017.
- Alberto Bietti, Vivien Cabannes, Diane Bouchacourt, Herve Jegou, and Leon Bottou. Birth of a transformer: A memory viewpoint. In *NeurIPS*, 2023.
- Pierre Brémaud and Laurent Massoulié. Stability of nonlinear hawkes processes. *Ann. Prob.*, pages 1563–1588, 1996.
- Natalia Caporale and Yang Dan. Spike timing-dependent plasticity: a hebbian learning rule. *Annu. Rev. Neurosci.*, 31:25–46, 2008.
- Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.
- Wen-Hao Chiang and George Mohler. Hawkes process multi-armed bandits for disaster search and rescue. *arXiv preprint arXiv:2004.01580*, 2020.
- Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. On the relationship between self-attention and convolutional layers. In *ICLR*, 2020.
- Kunihiko Fukushima and Sei Miyake. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognit.*, 15(6):455–469, 1982.
- Antonio Galves and Eva Löcherbach. Modeling networks of spiking neurons as interacting processes with memory of variable length. *J. Soci. Fr. Stat.*, 157(1): 17–32, 2016.

- Bolin Gao and Lacra Pavel. On the properties of the softmax function with application in game theory and reinforcement learning, 2018.
- 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.
- G.L. Gerstein, P. Bedenbaugh, and A.M.H.J. Aertsen. Neuronal assemblies. *IEEE Trans. Biomed. Eng.*, 36(1):4–14, 1989. doi: 10.1109/10.16444.
- Mark A Gluck and Gordon H Bower. From conditioning to category learning: an adaptive network model. *J. Exp. Psychol.*, 117(3):227, 1988.
- Eric C Hall and Rebecca M Willett. Tracking dynamic point processes on networks. *IEEE Trans. Inf. Theory*, 62(7):4327–4346, 2016.
- Alan G Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.
- Alan G Hawkes. Hawkes processes and their applications to finance: a review. *Quantitative Finance*, 18(2):193–198, 2018.
- Donald Olding Hebb. *The organization of behavior: A neuropsychological theory*. Psychology press, 2005.
- Pierre Hodara and Eva Löcherbach. Hawkes processes with variable length memory and an infinite number of components. *Adv. Appl. Probab.*, 49(1):84–107, 2017.
- David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *J. Physiol.*, 160(1):106, 1962.
- Sophie Jaffard, Samuel Vaiter, Alexandre Muzy, and Patricia Reynaud-Bouret. Provable local learning rule by expert aggregation for a Hawkes network. In *AISTATS*, 2024.
- Matthias Kirchner. An estimation procedure for the hawkes process. *Quantitative Finance*, 17(4):571–595, 2017.
- John K Kruschke. Alcove: An exemplar-based connectionist model of category learning. In *Connectionist psychology: a text with readings*, pages 107–138. Psychology Press, 2020.
- Regis C Lambert, Christine Tuleau-Malot, Thomas Bessaih, Vincent Rivoirard, Yann Bouret, Nathalie Leresche, and Patricia Reynaud-Bouret. Reconstructing the functional connectivity of multiple spike trains using hawkes models. *J. Neurosci. Methods*, 297:9–21, 2018.

- Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. Handwritten digit recognition with a back-propagation network. In *NeurIPS*, 1989.
- Robert Legenstein, Christian Naeger, and Wolfgang Maass. What can a neuron learn with spike-timing-dependent plasticity? *Neural Comput.*, 17(11):2337–2382, 2005.
- Robert Legenstein, Dejan Pecevski, and Wolfgang Maass. A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLoS computational biology*, 4(10):e1000180, 2008.
- Ashok Litwin-Kumar and Brent Doiron. Formation and maintenance of neuronal assemblies through synaptic plasticity. *Nature communications*, 5(1):5319, 2014.
- C. Mascart, D. Hill, A. Muzy, and P. Reynaud-Bouret. Scalability of large neural network simulations via activity tracking with time asynchrony and procedural connectivity. *Neural Comput.*, 34(9):1915–1943, 2022.
- Cyrille Mascart, David Hill, Alexandre Muzy, and Patricia Reynaud-Bouret. Efficient simulation of sparse graphs of point processes. *ACM Transactions on Modeling and Computer Simulation*, 33(1-2):1–27, 2023.
- Giulia Mezzadri, Thomas Laloë, Fabien Mathy, and Patricia Reynaud-Bouret. Hold-out strategy for selecting learning models: Application to categorization subjected to presentation orders. *J. Math. Psychol.*, 109:102691, 2022.
- Guilherme Ost and Patricia Reynaud-Bouret. Sparse space–time models: Concentration inequalities and lasso. 2020.
- Tien Cuong Phi, Alexandre Muzy, and Patricia Reynaud-Bouret. Event-scheduling algorithms with kalikow decomposition for simulating potentially infinite neuronal networks. *SN Comput. Sci.*, 1(1):1–10, 2020.
- Brian G Rash and Elizabeth A Grove. Area and layer patterning in the developing cerebral cortex. *Current opinion in neurobiology*, 16(1):25–34, 2006.
- Patricia Reynaud-Bouret and Sophie Schbath. Adaptive estimation for hawkes processes; application to genome analysis. 2010.
- Patricia Reynaud-Bouret, Vincent Rivoirard, and Christine Tuleau-Malot. Inference of functional connectivity in neurosciences via hawkes processes. In *GlobalSIP*, pages 317–320, 2013.
- Raanan Y. Rohekar, Yaniv Gurwicz, and Shami Nisimov. Causal interpretation of self-attention in pre-trained transformers. In *NeurIPS*, 2023.

- F. Rosenblatt. *The Perceptron, a Perceiving and Recognizing Automaton Project Para*. Cornell Aeronautical Laboratory, 1957.
- Frank Rosenblatt et al. *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*, volume 55. Spartan books Washington, DC, 1962.
- Roman Rybka, Yury Davydov, Danila Vlasov, Alexey Serenko, Alexander Sboev, and Vyacheslav Ilyin. Comparison of bagging and sparsity methods for connectivity reduction in spiking neural networks with memristive plasticity. *Big Data and Cognitive Computing*, 8(3):22, 2024.
- Alexander Sboev, Yury Davydov, Roman Rybka, Danila Vlasov, and Alexey Serenko. A comparison of two variants of memristive plasticity for solving the classification problem of handwritten digits recognition. In *Biologically Inspired Cognitive Architectures 2021*, pages 438–446, 2022.
- Menahem Segal. Dendritic spines and long-term plasticity. *Nature Reviews Neuroscience*, 6(4):277–284, 2005.
- Wolf Singer, Andreas K Engel, Andreas K Kreiter, Matthias HJ Munk, Sergio Neuenschwander, and Pieter R Roelfsema. Neuronal assemblies: necessity, signature and detectability. *Trends in cognitive sciences*, 1(7):252–261, 1997.
- James V Stone. Principles of neural information theory. *Computational Neuroscience and Metabolic Efficiency*, 2018.
- Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. Deep learning in spiking neural networks. *Neural Netw.*, 111:47–63, 2019.
- K Türkyilmaz, Maria Nicolette Margaretha van Lieshout, and A Stein. Comparing the hawkes and trigger process models for aftershock sequences following the 2005 kashmir earthquake. *Mathematical geosciences*, 45:149–164, 2013.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017.
- Ke Zhou, Hongyuan Zha, and Le Song. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *AISTATS*, 2013.
- Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. Transformer Hawkes process. In *ICML*, 2020.