# Consolidated Hebbian learning and parsimonious energy consumption resulting in large capacity associative memories

Eliott Coyac, Telecom Bretagne, eliott.coyac@telecom-bretagne.eu
Vincent Gripon, Telecom Bretagne, vincent.gripon@telecom-bretagne.eu
Charlotte Langlais, Telecom Bretagne, charlotte.langlais@telecom-bretagne.eu
Claude Berrou, Telecom Bretagne, claude.berrou@telecom-bretagne.eu

We show how a neural network with modifiable recurrent connections can result in large capacity associative memories. This network relies on the use of consolidated Hebbian learning and parsimonious energy consumption. It is inspired by [GB11] in which pieces of information are embodied as cliques in the network. Our purpose is to obtain the same network, but with a single set of rules used for both learning and decoding in the fashion of [GFB12].

The proposed network is made of $n$ nodes, that correspond to mesoscopic coherent groups of neurons. We use the following formalism. The connection weight at time $t$ between any two nodes $i$ and $j$ is denoted by $W_{ij}(t)$ and initialized at 0, the activity of the node $i$ is denoted by $V_i(t)$ and $I_i(t)$ is the external input to the node $i$. The network dynamics is driven by two main rules: a) consolidated Hebbian learning and b) parsimonious energy consumption.

For rule a), we use classical Hebbian learning with an added sigmoid $s$. The sigmoid is chosen such that it has two fixed points $s(0) = 0$ and $s(1) = 1$. As a result, in the absence of external inputs, connection weights tend to 0 or 1. This idea of weight consolidation in order to introduce long-term stability has already been studied [FAB+00]. We use $\frac{dW}{dt} = \alpha(s(\epsilon \cdot VV^\top + W) - W)$, where $\alpha$ depends on the unit of time and $\epsilon$ is the Hebbian learning rate. We fix the unit of time to $\alpha = 1$. A message, represented as a set of synchronously active nodes, is learned in the network after the corresponding nodes are stimulated for several successive iterations. The required number of iterations depends on $\epsilon$ and $s$: if the number of iterations is not enough, the connection weight will eventually stabilize to 0. Therefore, any fortuitous stimulation of two nodes at the same time during the processing of a given input will likely not create lasting connections.

For rule b), we limit the number of active nodes at the same time. This can be considered as implementing inhibition and limited energy consumption concerns. Basically we propose to limit the number of activated nodes in the network at each time step by selecting the most promising ones. There are two ways to implement this: i) the most promising nodes are chosen in the whole network without locality and ii) the network is split into clusters, mimicking the clustered minicolumn structure of the neocortex, and only the most promising node in each cluster is activated [Rin10]. The latter gives better results. In both cases we introduce a corresponding nonlinear function $h_c : \mathbb{R}^d \to \mathbb{R}^d$, $d \in \mathbb{N}$ as follows:

$$h_c(X_i) \leftarrow \begin{cases} 1 & \text{if } X_i \in \max_c(X), \\ 0 & \text{otherwise} \end{cases}$$

where $\max_c(X)$ is the set of the $c$ highest values in $X$. In case i) $h_c$ is used as is on the $n$ nodes. In case ii) $h_1$ is used independently on each cluster. Thus the node activity can be described with this equation: $\frac{dV}{dt} = -V + h(W \cdot V + I)$, and the neural network by the following set of equations:

$$\begin{cases} \dfrac{dV}{dt} & = -V + h(W \cdot V + I) \\ \dfrac{dW}{dt} & = s(\epsilon \cdot VV^\top + W) - W \end{cases}$$

where $h$ depends on whether case i) or ii) is used.

Finally, we obtain a neural network based associative memory with boundedness, a short incremental learning period, long-term stability and a large storage capacity. New data can be learned at any time, with no need to cycle through previous inputs. The network can be easily binarized thanks to weight consolidation. As for performance, we show that an example network of 2048 nodes storing messages of 8 bytes each can store 15000 messages and retrieve them perfectly from an half-erased version of them with an error rate lesser than 2%. This network, when binarized, provides a storage efficiency of 46% compared to raw binary storage, using roughly only twice the storage space.

## References

[FAB+00] Stefano Fusi, Mario Annunziato, Davide Badoni, Andrea Salamon, and Daniel J Amit. Spike-driven synaptic plasticity: theory, simulation, vlsi implementation. *Neural Computation*, 12(10):2227–2258, 2000.

[GB11] Vincent Gripon and Claude Berrou. Sparse neural networks with large learning diversity. *Neural Networks, IEEE Transactions on*, 22(7):1087–1096, 2011.

[GFB12] Mathieu N Galtier, Olivier D Faugeras, and Paul C Bressloff. Hebbian learning of recurrent connections: a geometrical perspective. *Neural computation*, 24(9):2346–2383, 2012.

[Rin10] Gerard J Rinkus. A cortical sparse distributed coding model linking mini-and macrocolumn-scale functionality. *Frontiers in neuroanatomy*, 4, 2010.