2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

STOCHASTIC SPIKING NEURAL NETWORKS AT THE EDGE OF CHAOS

J.L. Rosselló[*], V. Canals, A. Oliver, A. Morro
Department of Physics
University of Balearic Islands
Palma of Majorca 07122, Spain
[*]e-mail: j.rossello@uib.es

Abstract— **In this work we show a study about which processes are related to chaotic and synchronized neural states based on the study of in-silico implementation of Stochastic Spiking Neural Networks (SSNN). Chaotic neural ensembles are excellent transmission and convolution systems. At the same time, synchronized cells (that can be understood as ordered states of the brain) are associated to more complex non-linear computations. We experimentally show that complex and quick pattern recognition processes arise when both synchronized and chaotic states are mixed. These measurements are in accordance with in-vivo observations related to the role of neural synchrony in pattern recognition and to the speed of the real biological process. The measurements obtained from the hardware implementation of different types of neural systems suggest that the brain processing can be governed by the superposition of these two complementary states with complementary functionalities (non-linear processing for synchronized states and information convolution and parallelization for chaotic).**

Keywords— spiking neural networks, chaotic circuits, image filtering, pattern recognition

I. INTRODUCTION

Understanding the brain and to reach the ability to reproduce its amazing processing capabilities is one of the most challenging purposes for science and technology. Neural processing capabilities are of different nature such as body control [1], high-speed signal processing and classification [2], memory [3], decision-making etc. Since these diverse processes depend on the neural behavior, it is crucial to know which type of connectivity can be associated to each process. Some researchers support the theory that the brain is naturally poised near criticality at the edge between ordered and chaotic states [4-5]. This duality implies that the brain can access to a large range of complementary behaviors related to these two opposite states, and this could be a plausible explanation of the brain processing diversity.

A feasible way to study the different brain capabilities is to artificially reproduce those systems by using biologically plausible neural implementations. In this sense, hardware solutions are much faster than software approaches since the intrinsic neural parallelism is maintained. Also, the stochastic processes that are present in the neural behavior are more naturally incorporated to the study using hardware than software models. From the different hardware solutions that can be considered, those based on the use of digital logic represent the more attractive way for studying neural systems [6] since they can be massively implemented in Field Programmable Gate Arrays (FPGAs) [7-8] where hundreds of coupled neurons can be configured in a single chip for its detailed study. FPGAs also present an appropriate configurability to quickly change and test different neural designs. In this study we use the circuit model of Stochastic Spiking Neurons inspired developed by the researchers in a previous work [9].

Understanding complex systems always entails choosing a level of description that retains key properties of the system while removing what is supposed to be nonessential. In this sense, spiking neural models [10-11] are bio-inspired descriptions that use delta functions to represent the Action Potentials of biological neurons. Implicitly this assumes that information is non-transmitted from cell to cell in the spike shape but on its timing characteristics. At the same time we assume that firing rate is the main code used in nervous systems [12], to what extent more complex timing codes exist remains a subject of considerable debate [13].

A key property of the spike trains measured in real neural systems is their seemingly stochastic or random nature [14]. In fact, the apparent lack of reproducible spike patterns has been one of the principal arguments in favor of firing rate coding in contrast to the more elaborated timing codes. The stochastic nature of spike trains is in part due to the mechanism of synaptic transmission since each synaptic vesicle releases its "quantum" of transmitter from the neuron presynaptic terminal with a given probability. This probability of synaptic release is subject to change and can be understood as a measurement of the connection weight. Due to this probabilistic nature for the synapsis, a basic feature of neuronal communication is the lack of transmission reliability [13-14]. Moreover, as the number of channels in the membrane is reduced, the axonal membrane

---

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness (MINECO) the Regional European Development Funds (FEDER) and the Comunitat Autònoma de les Illes Balears under grant contracts TEC2011-23113, AAEE018/2012 and BES-2012-053600.

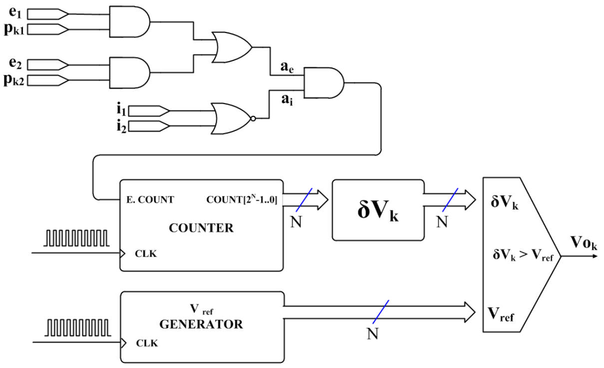


Fig. 1. Digital implementation of the Binding Neuron model. Simple digital blocks are considered as basic Boolean gates, a counter, a random number generator (for V_{ref}) and a comparator.

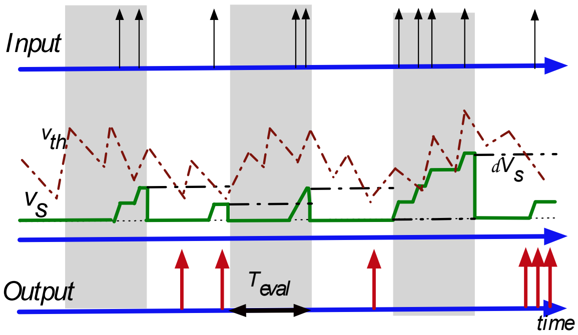


Fig. 2. Temporal evolution of membrane potential (vs) and the output (action potential) of the proposed stochastic digital neuron (Fig.1)

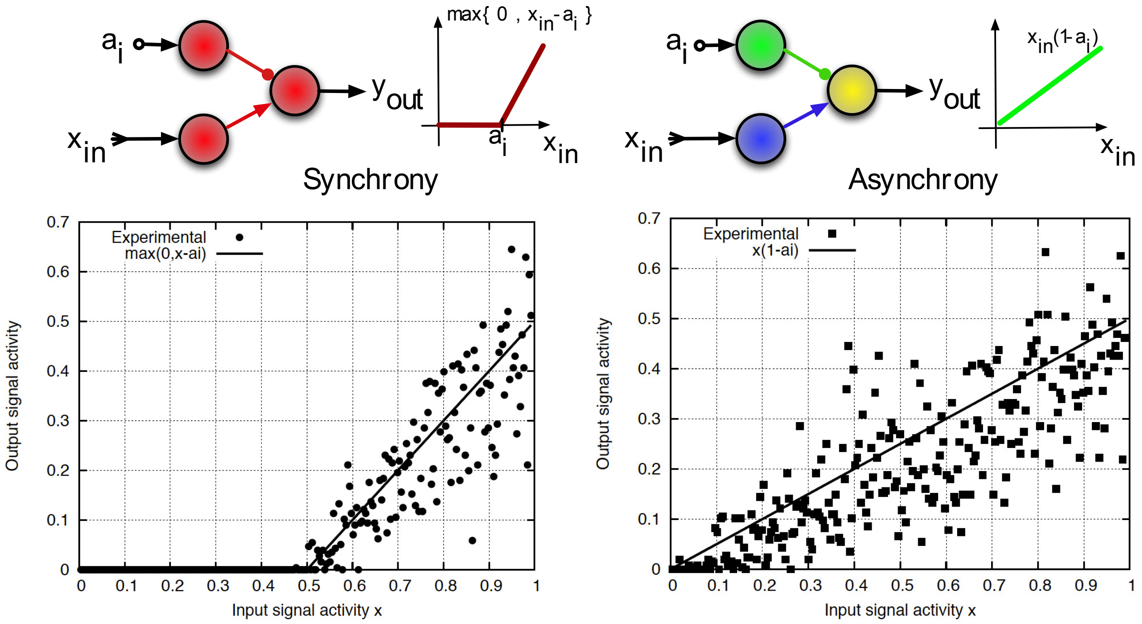
can spontaneously and stochastically reproduce action potentials [15], thus generating an autonomous internal activity. In general, depending on the underlying neural activity, spike trains may present a regular or a stochastic behavior [13].

The chaotic fluctuation of the resting potential of neurons [14] can provide a possible explanation for the different brain processing tasks. A synchrony (asynchrony) in the activity should create correlated (uncorrelated) random oscillations. The interaction between those different types of oscillations would lead to opposite behaviors.

In this paper we use a stochastic neural circuitry that implements a simple spiking neuron model [10] and also reproduces the probabilistic nature of synaptic transmissions [9]. In a previous work [9] we stated the basic principles of the hardware implementation of SNN by using digital circuitry. Spiking neural networks are the last generation of artificial neural models characterized by its bio-inspired nature [10]. Those models are used for medical [17-18] applications, prediction and forecasting [19], robotics [20], spatiotemporal information processing [21] or for the modeling of brain behavior [22-23]. For those reasons, spiking neural networks has been implemented in hardware in several studies for its faster processing [24-25]. From the hardware model proposed in reference [9] we study the different processing capabilities that can arise in neural systems, showing that neural ensembles present very different properties depending on their timing relationship. We study the effects of synchronized and chaotic signals and state that synchronized signals can be used for elaborated processing tasks such as pattern recognition and that chaotically related signals are optimum for information transmission and convolution and for parallelizing (and therefore speeding-up) different high-level tasks.

II. METHODS

The model in which the neural study is based is a variation of the binding neuron model [26-27]. The Binding Neuron (BN) model describes the neural functionality in terms of discrete events (digital pulses emulating the action potential of real neurons). Each input impulse is stored for a fixed time, similar to the tolerance interval proposed by MacKay [28]. Each neuron fires if the number of stored impulses (that represents a mean of the membrane potential) is equal or higher than a threshold value v_{th}. This model can be used for neural description and can be easily implemented using digital circuitry. The mean membrane potential is estimated each time T. Then, if and only if the mean membrane over-voltage ($\delta V_s \equiv v_s - v_{rest}$) during time T is greater than the oscillating threshold value v_{th}-v_{rest}, an action potential at the neuron output (v_o) is generated. The simple mechanism of the neural model can be reproduced digitally as shown in Fig. 1 (example for a two-input neuron scheme that can be generalized to the multiple input cases). In Fig. 2 it is showed the timing diagram of inputs, outputs and membrane potential v_s. During each time step T_{eval} ($T_{eval}=256$ in our experiments) the incoming pulses are summed up in the counter to estimate δV_s for the next period, then δV_s is compared with voltage v_{th}-v_{rest} (that varies chaotically between 0 and 255) to generate a spike signal at the output. All the incoming excitatory signals (e_i) are multiplied by its weight p_{kj} and joined using an OR gate (see Fig.1). Those spike trains increase the value of a digital counter and is captured by a register that provides at its output the number of incoming pulses during a certain period of time T_{eval} (representing an estimation of the mean incoming current to the neuron and therefore the membrane overvoltage δV_s). Signals p_{kj} represent the probability of signal transmission from the j_{th} to the k_{th} neuron, consisting of binary bits oscillating with a specific switching activity that is proportional to the probability of synaptic transmission (p_{kj}). At the same time, shunting inhibition signals (i_j) inhibits the action of excitatory pulses by using NOT gates. When the estimated mean overvoltage δV_s is over the reference voltage ($v_{ref} = v_{th}$-v_{rest}), different action potentials (represented by delta functions) are generated (see Fig. 2 for the illustration of this behavior). The mean membrane overvoltage (output binary number δV_k of the digital counter) is compared with the reference signal (v_{ref}) representing the difference between the threshold and the resting voltage ($v_{ref} = v_{th}$-v_{rest}). This reference signal oscillates randomly to accurately emulate the variation of v_{th} with respect to v_{rest} in real biological systems. Such random oscillation is assumed to be much faster than the typical spiking activity of neurons so that for each spike generation

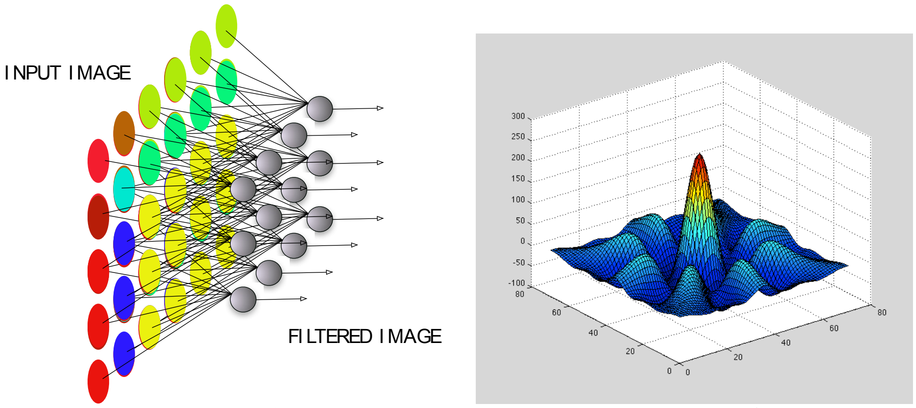


(a) (b)

Fig. 3. Basic measured differences between asynchronous and synchronous neural systems. Measurements show a non-linear relationship between output and input activity when the neural ensemble is synchronized in contrast to a linear relationship when the ensemble is not synchronized. In the figure we observe those relationships in the synchronous (a) and in the chaotic case (b). We show the cases in which there is one inhibitory and an excitatory signal. Since the behavior of the neural model is stochastic, the mathematical expressions can be easily obtained by applying basic probabilistic laws.

the reference voltage value is located at a completely new random value. To emulate such oscillations we implement digitally a random signal generator based on the use of three-coupled XNOR gates and a digital shift register [29-30]. The generation of stochastic signals p_{kj} that represents the probability of synaptic transmission between neurons is reproduced by digitally comparing the output of a random number generator and a specific binary value that has to be proportional to this probability. Signal p_{kj} therefore measures the strength of connection between the two neurons and its biophysical meaning is related to the probability of vesicle release in the synaptic connection. When operated with e_j through an AND gate (see Fig. 1) the result is a pulsed signal with a switching activity equal to the product of both signals ($p_{kj} \bullet e_j$). This is the way in which stochastic transmissions (and thus the weights of neural connections) are reproduced in the digital circuitry used to reproduce the neural model. The digital circuit used in this work was implemented using a Field-Programmable Gate Array model Cyclone III EP3C16F484C6 from ALTERA Corp., and the frequency of operation (that is fixing the minimum possible time interval between spikes) was 50MHz. In this work we study the behavior of spike trains presenting synchrony [16] or a stochastic behavior [13]. One of the key points of the stochastic neural model used is that the difference between the threshold and the resting potentials of neurons is variable and can be used to synchronize neurons. Synchronicity does not necessary imply a common periodic oscillation but a correlated oscillation (not necessary to be periodic). Detailed computational simulations of the default-mode brain network model shows that synchronized oscillation may be present even in distant brain regions [31]. In general, we say that two neurons are synchronized if they share the same reference voltage v_{ref}, otherwise they are chaotically related (de-synchronized).

To highlight the differences that can be observed we show the simple examples of Fig. 3. This figure illustrates four systems consisting in two input neurons with connection with a third one (the output). Those connections are of inhibitory and excitatory nature depending on the example. One of the two input activities to the circuit is fixed (a_i) while the other is varied (x_{in}). In the absence of synchrony between neurons we observe a nearly linear relationship between the output activity y_{out} and the two input activities (a_i and x_{in}). This can be appreciated in Fig. 3b. The final expression obtained is the product of the switching activity of x_{in} and the probability of no-inhibition $(1-a_i)$. The reason of this product is that it the neuron output is providing the collision probability between the two successes (activation of x_{in} and no inhibition). For the case of synchronized neurons and an inhibitory connection (Fig. 3a) we observe that when the input x_{in} is lower than the inhibition activity (a_i) the excitatory signal is unable to activate the output since it is correlated with the inhibitory signal. Only if the excitation exceeds the inhibition we get an appreciable output activity that can be approximated as being linear ($y_{out}=x_{in}-a_i$).



(a) (b)

Fig. 4. (a) Simple neural system used for image filtering, each neuron at the output layer is connected to 64 neurons of the input layer. (b) The weights used for the connection between the input and output layers corresponds to the delta response of the Gabor filter. Positive (negative) values represent excitatory (inhibitory) connections.

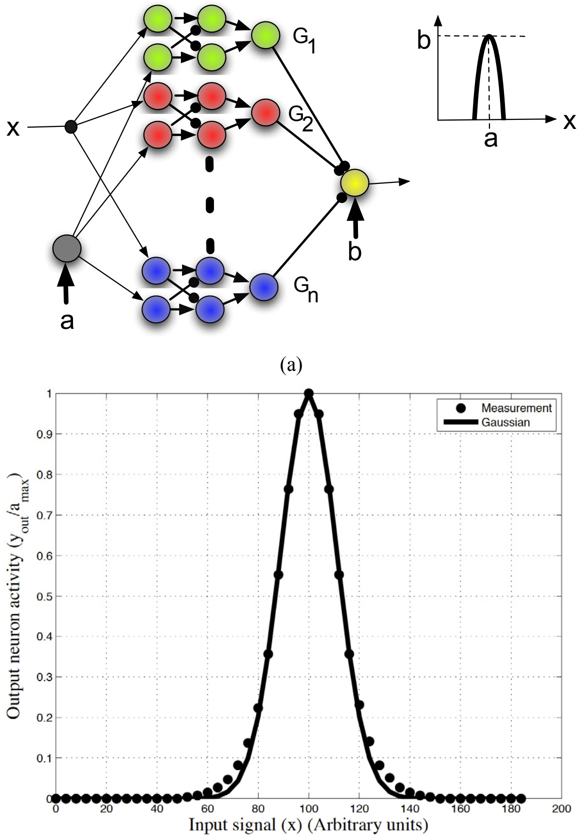
(a) (b) (b) (d) (e) (f)

Fig. 5. Switching activity of the two layers SSNN. (a) Original Lena image (input layer). (b) Output layer when λ=0.8 and b=1. (c) Output layer when λ =1 and b=0.5. (d) Output layer when λ =1 and b=1. (e) Output layer when λ =2 and b=0.5. (f) Output layer when λ =2 and b=1.

Therefore, in the synchronous case we observe an abrupt non-linear change between different linear states that is not present in the chaotic case. The dispersion observed in the measurements around the expected behavior are due to the intrinsic stochastic activity of neurons, where the firing rates are measured averaging in a given time period. The analytical expressions that are provided in Fig.3 for the expected neural behavior are extracted by applying basic probabilistic laws.

Therefore, the first and crucial observation that is derived from the digital stochastic neural model is that both chaotically related and synchronized neural systems present a linear and a non-linear behavior respectively. In case of synchrony we observe that the system selects its operation mode between different linear states while in the case of a chaotic behavior we observe a simple linear transformation.

The linear transformations minimize the information loss and are able to convolute chaotically-related incoming signals while the non-linear transformation is associated to more complex non-linear processing tasks like pattern discrimination, where the signal information is drastically reduced to the useful one.



(a)

(b)

Fig. 6. (a) A simple circuit scheme for pattern classification. (b) Measured activity of the proposed system when using 32 neural ensembles. Due to the effect of parallelization, the intrinsic variability of the stochastic process is minimized.

III. RESULTS

A. Chaotic ensembles of neurons as efficient information processing systems

As it has been shown, unsynchronized ensembles of neurons perform linear transformations to neural information. As a practical example, they can be used to implement the convolution to incoming sensory data.

An edge-detection filter based on the use of a Gabor filter can be implemented using a simple feed-forward SSNN with only two layers (see Fig.4). Frequency and orientation of Gabor filters are thought to be involved in the human visual system, specifically in the process of feature selection and discrimination. A Gabor filter is composed by the superposition of different Gaussian kernels modulated by plane waves. The main characteristic of each kernel is the orientation (θ), the bandwidth (b), the wavelength (λ), the aspect ratio (γ) and the phase (φ). The output response of the filter must be the convolution of the input image I(x,y) with the function:

$$f(x,y)=\sum e^{-(\hat{x}_i^2+\hat{y}_i^2\gamma^2)/2\sigma^2}\cos(2\pi\hat{x}_i/\lambda+\phi) \quad (1)$$

Where $\hat{x}_i$ and $\hat{y}_i$ are defined by the specific orientation θ_i as:

$$\begin{aligned}\hat{x}_i &= x\cdot\cos\theta_i + y\cdot\sin\theta_i\\ \hat{y}_i &= -x\cdot\sin\theta_i + y\cdot\cos\theta_i\end{aligned} \quad (2)$$

Parameters x and y are referred to the position in the x-y plane of the image. At the same time, the σ value is defined by the bandwidth b as:

$$\sigma=\lambda\sqrt{\ln 2/2}\left(2^b+1\right)/\left(\pi\left(2^b-1\right)\right) \quad (3)$$

For the implementation of an image filter we use a 2D SSNN with two layers (input neurons $\hat{I}_{kl}$ and output neurons $\hat{O}_{ij}$), where sub-indexes k,l and i,j are referred to their respective position in the image plane. The input layer provides the image information (the switching activity of neuron $\hat{I}_{kl}$ is proportional to the intensity of pixel p_{kl} of the image) to the output layer with a weight at the connection given by (1) as $\omega(\hat{O}_{ij};\hat{I}_{kl})=f(i-k,j-l)$. Those weights are implemented through the probability of synaptic transmission (p_{kj}) that we were talking in the previous section.

We implemented a neural system for the edge detection processing of a 64KB image. Each output neuron of the network is connected to sixty-four inputs of the image (pixels) following the weight distribution provided by (1) (see Fig. 4b) using a superposition of eight kernels with eight different orientations such that $\theta_i=i\pi/4$ (being *i an integer* ranging between *0* and *7*).

In Fig. 5 it is showed the switching activity of the two 2D layers of the network when processing the Lena's image. At the input layer we have the original image of Lena (Fig. 5a) and at the output layer we obtain the filtered image for different weight values associated to different parameters of the Gabor filter. We show the effect of five different filters. For each filter we use γ=0.5 and φ=0 and different values of λ and b (bandwidth). Exactly we use λ =0.8 and b=1 for Fig. 5b, λ=1 and b=0.5 for Fig. 5c, λ =1 and b=1 for Fig. 5d, λ =2 and b=0.5 for Fig. 5e and finally λ=2 and b=1 for Fig. 5f. We can appreciate the effect of changing λ and bandwidth through all the figures. From all the parameters, the one with λ=1 and b=1 (Fig. 5d) presents the better edge detection capability. For the implementation of this filter a total of 2^{22} synapses are used (more than four millions of synapses). Using similar methodologies, any kind of linear filtering can be built using a simple feed-forward neural network.

B. Synchronized systems are able to recognize patterns while chaos parallelize and speeds-up the process

Pattern recognition is a fundamental brain process consisting in a reaction to an external stimulus when it is within a relevant range. Some works suggest that this process is only explained using the action potential timing as codification methodology [32] since the analogue pattern match is done in a time scale of the order of dozens of milliseconds and biological neurons are oscillating at only 100Hz. These are the results observed in different studies related to visual pattern analysis and pattern classification carried out by macaque monkeys and fixing the time

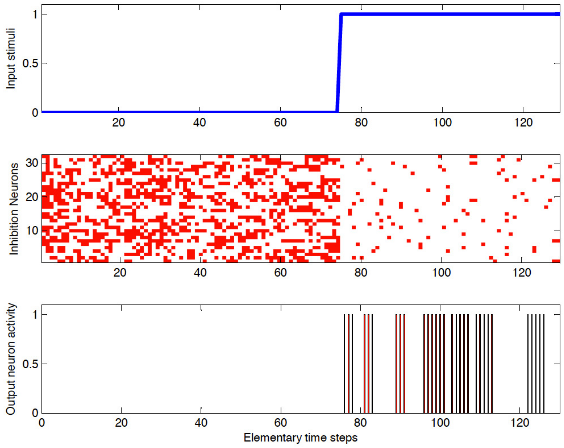


Fig. 7. Evaluation of the time response of the pattern recognition system, a low number of synaptic transmissions is needed to recognize if the incoming signal is within the desired range.

response in just 20-30ms [33-34]. Since the firing rate of neurons are usually below 100Hz, a coding of analogue variables by firing rates is traditionally considered to be dubious for pattern recognition. Although the timing codification proposed by Hopfield explains the fast pattern recognition developed by real neural systems, its physical implementation requires a high spatiotemporal precision with which neurons need to be wired up and the learning process of such coding could be quite complex.

Other studies demonstrate that when odor-evoked action potentials in honeybee antennal lobe neurons are pharmacologically desynchronized (without interrupting neural activity) fine olfactory discrimination is interrupted [35-36]. This empirical evidence links directly neuronal synchronization to pattern recognition. Here we suggest that synchronized systems (that presents the ability of switching between ON and OFF states), are responsible of the pattern recognition process by simply using a firing rate coding, and that this process is quick enough to explain the fast pattern recognition observed in the biological experiments [33-34].

In Fig. 6a we show a neural scheme where different synchronized neuron ensembles (G_1, G_2,…,G_n) are configured to recognize the point in which the input stimuli x is near a reference activity a. Each neural ensemble inhibits the activity of an output neuron that is stimulated with an external signal 'b'. The output activity of each synchronized ensemble vanishes only when $x=a$, thus allowing the output neuron to reach its maximum activity (b). If the stimuli (x) is not in the neighborhood of a then all the ensembles (that are not synchronized between them) inhibits the activity of the output neuron. The result is a Gaussian-like response of the output with respect the input (see the measurements shown in Fig. 6b). Note the differences in signal fluctuation between this figure and Fig 3 due to the difference in the neural network size (3 neurons vs. more than one hundred for the neural system used to generate Fig. 6b). The massive use of neural ensembles decrease the variability of the response as is shown in Fig.6b. The system response is quite fast since only three synaptic steps are involved in the process (thus being in accordance with the fast responses observed in real neural systems). The dispersion of the Gaussian function depends on the number of synchronized ensembles (the sigma value decreases as the number of ensembles increases). A measurement of the timing activity is shown in Fig. 7 where we represent the neural behavior before ($input=0$) and after ($input=1$) the sensed stimuli is within the desired range. A total of 32 neural ensembles are used in the experiment. Before the detection, the 32 ensembles present an appreciable activity and the output neuron is unable to switch. After the transition the ensembles activity practically vanishes and the output neuron immediately starts to fire about two time steps after the transition (just indicating the presence of the desired pattern). A time step represents the fastest neural oscillation period, therefore implying a recognition speed of the order of 20ms if the frequency of operation is of the order of 100Hz for real neural systems. This is in accordance with the in-vivo measurements [33-34].

The circuit of Fig.6a (adapted only for 1D signal and Gaussian discriminations) is easily generalized to higher dimensions and non-Gaussian patterns by superposing different Gaussian (kernels) functions.

The circuit shown in Fig. 6 is only an example of how a classifier can be implemented by combining synchronized and desynchronized systems. Probably there are many other configurations that can implement a similar computation but the concept is that, pattern classifiers with a few synaptic connections and with a simple training methodology (in this case, external signals are used to configure the network) can be implemented by using synchronized neural ensembles. Note that in Fig. 6a the different neural ensembles are not synchronized between them since they must completely inhibit the output neuron (a pure synchronized system would fail in a complete inhibition of the output neuron when the stimuli are out of the desired range of detection). Therefore, the existence of chaotically-related neural ensembles is necessary for a proper parallelization of the pattern recognition process.

IV. CONCLUSION

Using a digital stochastic neural model we implemented different neural systems incorporating hundreds of neurons for its study under different conditions of synchrony. The measurements show that neurons are unique processing elements that are able to change its functionality drastically depending on the correlations among spike trains (see results shown in Fig. 3). Simple processes such as image convolution are perfectly implemented by pure chaotic systems. In addition, complex processes such as pattern recognition can be associated to the relationship between both synchronized and de-synchronized groups of neurons. Synchronized ensembles are able to “choose” between different states due to its non-linear behavior (see results of Fig.3a and Fig.3c). The un-correlation between the synchronized neural ensembles seems to improve the complex processes by parallelizing them (and thus speeding up the process). As a consequence of that, it has been demonstrated that firing rate codes cannot be discarded to be involved in the pattern recognition process since the proposed mixed system (with both chaotic and ordered neural ensembles such as the one shown in Fig.6a) is able to recognize patterns using a low number of synaptic steps, thus explaining the fast pattern recognition process observed in the mammalian cortex.

The experimental implementations are in accordance with different in-vivo experiments in which pattern discrimination is interrupted when neurons are pharmacologically desynchronized [35-36]. In this work we also show a biologically realistic stochastic neural model that can be implemented in hardware by using digital gates. The proposed implementation has the advantage of operating with a high speed of operation (in the measurements shown the circuit was operating at 50MHz although much higher frequencies can be achieved by current digital technologies). As is shown, these systems can be used to experiment with large neural systems composed by thousands of neurons. This is in contrast with other FPGA implementation [38] where only a maximum of 20 Spiking Neurons can be implemented in a medium size FPGA.

REFERENCES

[1] J. D. Enderle and D. A. Sierra, "a new linear muscle fiber model for neural control of saccades," Int. J. Neur. Syst., vol. 23, pp. 1350002, 04/01; 2014/01, 2013

[2] M. A. H. Akhand and K. Murase, "Ensembles of neural networks based on the alteration of input feature values," Int. J. Neur. Syst., vol. 22, pp. 77-87, 02/01; 2014/01, 2012.

[3] G. Li, N. Ning, K. Ramanathan, W. He, L. I. Pan and L. Shi, "Behind the magical numbers: hierarchical chunking and the human working memory capacity," Int. J. Neur. Syst., vol. 23, pp. 1350019, 08/01; 2014/01, 2013.

[4] N. Bertschinger and T. Natschläger, "Real-Time Computation at the Edge of Chaos in Recurrent Neural Networks," Neural Comput., vol. 16, pp. 1413-1436, 07/01; 2014/01, 2004.

[5] D. R. Chialvo, "Emergent complex neural dynamics," Nat Phys, vol. 6, pp. 744-750, print, 2010.

[6] T. Hishiki and H. Torikai, "A Novel Rotate-and-Fire Digital Spiking Neuron and its Neuron-Like Bifurcations and Responses," Neural Networks, IEEE Transactions on, vol. 22, pp. 752-767, 2011.

[7] Y. Maeda and Y. Fukuda, "FPGA implementation of pulse density hopfield neural network," in Neural Networks, 2007. IJCNN 2007. International Joint Conference on, 2007, pp. 700-704.

[8] B. Belhadj, J. Tomas, O. Malot, G. N'Kaoua, Y. Bornat and S. Renaud, "FPGA-based architecture for real-time synaptic plasticity computation," in Electronics, Circuits and Systems, 2008. ICECS 2008. 15th IEEE International Conference on, 2008, pp. 93-96.

[9] J. L. Rosselló, V. Canals, A. Morro and A. Oliver, "Hardware implementation of stochastic spiking neural networks," Int. J. Neur. Syst., vol. 22, pp. 1250014, 08/01; 2014/01, 2012.

[10] S. Ghosh-Dastidar and H. Adeli, "Spiking neural networks," Int. J. Neur. Syst., vol. 19, pp. 295-308, 08/01; 2014/01, 2009.

[11] W. Maass, "Networks of spiking neurons: The third generation of neural network models," Neural Networks, vol. 10, pp. 1659-1671, 12, 1997.

[12] M. London, A. Roth, L. Beeren, M. Hausser and P. E. Latham, "Sensitivity to perturbations in vivo implies high noise and suggests rate coding in cortex," Nature, vol. 466, pp. 123-127, 07/01, 2010.

[13] C. Koch, Biophysics of Computation: Information Processing in Single Neurons (Computational Neuroscience). Oxford University Press, 1998.

[14] P. Steinmetz, A. Manwani, C. Koch, M. London and I. Segev, "Subthreshold Voltage Noise Due to Channel Fluctuations in Active Neuronal Membranes," J. Comput. Neurosci., vol. 9, pp. 133-148, 09/01, 2000.

[15] C. C. Chow and J. A. White, "Spontaneous action potentials due to channel fluctuations," Biophys. J., vol. 71, pp. 3013-3021, 12, 1996.

[16] A. Jahangiri and D. M. Durand, "Phase resetting analysis of high potassium epileptiform activity in ca3 region of the rat hippocampus," Int. J. Neur. Syst., vol. 21, pp. 127-138, 04/01; 2014/01, 2011.

[17] S. Ghosh-Dastidar and H. Adeli, "A new supervised learning algorithm for multiple spiking neural networks with application in epilepsy and seizure detection," Neural Networks, vol. 22, pp. 1419-1431, 12, 2009.

[18] S. Ghosh-Dastidar and H. Adeli, "Improved spiking neural networks for EEG classification and epilepsy and seizure detection," Integrated Computer-Aided Engineering, vol. 14, pp. 187-212, 01/01, 2007.

[19] S. Kulkarni, S. P. Simon and K. Sundareswaran, "A spiking neural network (SNN) forecast engine for short-term electrical load forecasting," Applied Soft Computing, vol. 13, pp. 3628-3635, 8, 2013.

[20] N. R. Luque, J. A. Garrido, R. R. Carrillo, S. Tolu and E. Ros, "Adaptive cerebellar spiking model embedded in the control loop: context switching and robustness against noise," Int. J. Neur. Syst., vol. 21, pp. 385-401, 10/01; 2014/01, 2011.

[21] A. Mohemmed, S. Schliebs, S. Matsuda and N. Kasabov, "Span: spike pattern association neuron for learning spatio-temporal spike patterns," Int. J. Neur. Syst., vol. 22, pp. 1250012, 08/01; 2014/01, 2012.

[22] R. Memmesheimer, "Quantitative prediction of intermittent high-frequency oscillations in neural networks with supralinear dendritic interactions," Proceedings of the National Academy of Sciences, vol. 107, pp. 11092-11097, 2010.

[23] A. Vidybida, "Testing of information condensation in a model reverberating spiking neural network," Int. J. Neur. Syst., vol. 21, pp. 187-198, 06/01; 2014/01, 2011.

[24] J. L. Rossello, V. Canals, A. Morro and J. Verd, "Chaos-based mixed signal implementation of spiking neurons," Int. J. Neur. Syst., vol. 19, pp. 465-471, 12/01; 2014/01, 2009.

[25] A. Basu, S. Shuo, H. Zhou, M. Hiot Lim and G. Huang, "Silicon spiking neurons for hardware implementation of extreme learning machines," Neurocomputing, vol. 102, pp. 125-134, 2/15, 2013.

[26] A. K. Vidybida, "Input–output relations in binding neuron," Biosystems, vol. 89, pp. 160-165, 0, 2007.

[27] A. K. Vidybida and K. G. Kravchuk, "Firing statistics of inhibitory neuron with delayed feedback. I. Output ISI probability density," Biosystems, vol. 112, pp. 224-232, 6, 2013.

[28] D. M. MacKay, Self-Organization in the Time Domain. 1962.

[29] R. Zhang, H. L. D. de S.Cavalcante, Z. Gao, D. J. Gauthier, J. E. S. Socolar, M. M. Adams and D. P. Lathrop, "Boolean chaos," Phys Rev E., vol. 80, pp. 045202, Oct, 2009.

[30] J. L. Rosselló, V. Canals, I. d. Paúl, S. Bota and A. Morro, "A simple CMOS chaotic integrated circuit," IEICE Electronics Express, vol. 5, pp. 1042-1048, 2008.

[31] T. Yamanishi, J. Liu and H. Nishimura, "Modeling fluctuations in default-mode brain network using a spiking neural network," Int. J. Neur. Syst., vol. 22, pp. 1250016, 08/01; 2014/01, 2012.

[32] F. Cona and M. Ursino, "A multi-layer neural-mass model for learning sequences using theta/gamma oscillations," Int. J. Neur. Syst., vol. 23, pp. 1250036, 06/01; 2014/01, 2013.

[33] J. J. Hopfield, "Pattern recognition computation using action potential timing for stimulus representation," Nature, vol. 376, pp. 33-36, 07/06, 1995.

[34] E. T. Rolls, "Brain mechanisms for invariant visual recognition and learning," Behav. Processes, vol. 33, pp. 113-138, 12, 1994.

[35] E. T. Rolls, M. J. Tovee, "Processing speed in the cerebral cortex and neurophysiology of visual masking," Proceedings of the Royal Society B: Biological Sciences, vol. 257, pp. 9-15, 1994.

[36] C. Linster and T. Cleland, "How spike synchronization among olfactory neurons can contribute to sensory discrimination," J. Comput. Neurosci., vol. 10, pp. 187-193, 03/01, 2001.

[37] M. Stopfer, S. Bhagavan, B. H. Smith and G. Laurent, "Impaired odour discrimination on desynchronization of odour-encoding neural assemblies," Nature, vol. 390, pp. 70-74, 11/06, 1997.

[38] A. Rosado, M. Bataller and J. Guerrero "FPGA implementation of spiking neural network" in 1st IFAC Conference on Embedded Systems, Computational intelligence and Telematics in Control - CESCIT'12, vol 1, 2012, pp. 139-144