2008 International Symposium on Nonlinear Theory and its Applications
NOLTA'08, Budapest, Hungary, September 7-10, 2008

Modeling behavioral tasks to asses visual information integration and motor information encoding in parietal cortex

Alessandro Palladini†, Jugoslava Acimovic‡, Martin Hasler‡

†ARCES, University of Bologna, Italy
‡LANOS, School of Computer and Communication Sciences,
Ecole Polytechnique Federale de Lausanne, Switzerland
Email: apalladini@arces.unibo.it, jugoslava.acimovic@epfl.ch

Abstract—In this work, we present an example of bio-inspired dynamical network, developed to help in studies of motor-related processes in the brain. The presented model aims to mimic the functionality of a cell population in 7a parietal region. The obtained results help us in studying the experimental data set, described in [1]. This approach enables testing the influence of several relevant parameters of the data analysis procedure presented in [2].

1. Introduction

The neural network presented in this paper is a part of a larger study of hand-eyes coordination and motor intention in parietal cortex. The foundation for this work is described in [1]. The lab monkey is required to perform a complex set of behavioral tasks, while spike trains are collected from the 7a parietal area. In our previous work [2], we describe the machine learning based methods for detecting motor intention from the mean firing activity of simultaneously monitored cells. The corresponding spike trains are first converted into vectors of spike rates. In the first test, the mean activity is calculated for each cell and each epoch in the experiment, and the data label is assigned in accordance to that epoch. If hand or eye motion is planned, prepared, or executed in an epoch, the resulting spike rates vector is labeled as *motor intention.* Otherwise, when a movement is absent, we assign *no intention* label. For the second test, a sliding window is introduced for calculating spike rates. The set of allowed window positions spans the entire duration of a task. A spike rate is calculated for each position. The window length represents an additional parameter that influences the result. Labels are, again, assigned using the relative position of the window with respect to experimental epochs. Support vector machine classifier (SVM) is applied for discriminating between presence and absence of motor intention. Literature on brain-computer interface presents many methods for extracting various motor parameters, used to control a computer cursor or prosthetics [8]. In this work, we are focused on analyzing the processes in the recorded cortex region, rather than building an operating system, useful for making prosthetics. The results, presented in [2], show that the presence of motor intention in the studied region can be successfully identified, from a randomly chosen cell set. In average, for a randomly selected recording site, the classification precision remains between 25% and 30%. Although the analysis presented in [2] revealed interesting conclusions, experimentally imposed limitations disable testing several relevant parameters. First, the maximal number of simultaneously recorded cells is limited to fourteen. Also, the influence of window size cannot be properly investigated. These issues are assessed through the model described in this paper.
The presented artificial neuronal network relies on concepts of liquid state machines (LSN) [7] and echo state networks (ESN)[6]. Each cell in the network is represented using a leaky integrate and fire model complemented with the model for chemical synapses [5]. No special network structure is imposed, the connections are chosen randomly. The LSN is used in the network selection procedure: a large number of networks is generated, according to certain rules. The motor output layer is trained by linear regression over generated spike trains. The most successful networks are selected. Then, a small number of the states in the networks are chosen randomly, simulating the experimentally chosen neurons, and the corresponding spike trains are classified by SVM as in the experimental data analysis. We will refer to this result as 'classification outputs', in order to distinguish them from the motor output signals.

2. Behavioral tasks

The complex set of experiments, presented in [1], is replaced by only two tasks considered here. Since detection of motor intention is in the focus of this work, only one example of a task involving a movement, and a task with no movements are considered. Fig. 1 shows the tasks of interest, and the corresponding input signals for the model. Task on the top left figure requires movement of the hand toward the presented target. The epochs involved in preparation for the movement

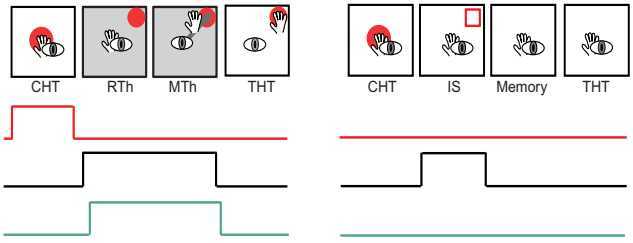

Figure 1: Abstract representation of behavioral tasks.

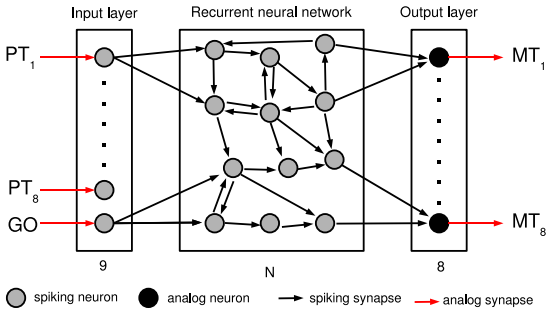

Figure 2: Proposed model architecture

and the movement itself are shaded. In these intervals we expect to observe motor intention. The three signals below are showing the two inputs and the desired output of the model. The first pulse is the instruction to move, the *GO* signal, and the second one corresponds to the presentation of the target on the screen. It is *ON* during the two motor-related epochs, and appears in one of eight inputs (corresponding to the eight target positions in the experiments described in [1]). The last pulse shows the ideal motor output of the network, the motor intention appears during the two movement related intervals. On the right side of the figure, the task with no movement required is described. Here the target is presented on the screen, i.e. the pulse appears on one of the eight inputs, but the instruction to move, i.e. the *GO* signal, is absent. Therefore, at the motor output, no activity is expected (as shown with the third signal).

3. Proposed Model

Network nodes are modeled as leaky integrate and fire (LIF) neurons [5], with the membrane potential of the neuron i, $u_i(t)$, given as,

$$\tau_i^m \frac{du_i}{dt} = -u_i(t) + R_i \cdot I_i(t) \qquad (1)$$

R_i is the cell membrane resistance, and $\tau_i^m = R_i \cdot C_i$ is the neuron time constant. The second term on the right side, the input current, is defined as

$$I_i(t) = J_i + I_i^e(t) + I_i^n(t) + \sum_j w_{ij} \sum_f \alpha_{ij}(t - t_j^f) \quad (2)$$

where J_i accounts for the internal constant current, $I_i^e(t)$ for the external current, $I_i^n(t)$ for the diffusive noise (modeled as the Gaussian zero mean white process, with the variance σ_i^n), and the last term describes the synaptic current (where $\alpha(\cdot)$ describes the shape of the post-synaptic current pulse, and is given in the literature [5]). Finally, the membrane potential resetting condition is given as

$$t^{(f)} : u_i(t^{(f)}) = \theta_i. \qquad (3)$$

where θ_i stands for the reseting threshold. Immediately after the spike event $t^{(f)}$, the membrane potential is reset to a new value $u_i^r < \theta_i$ and it stays in the same state during the refractory time τ_i^r. Therefore, the model for the i-th cell is defined with the listed equations and the set of parameters $\mathcal{C}_i = \{\tau_i^m, R_i, \theta_i, \tau_i^r, J_i, I_i^e(t), \sigma_i^n, \{w_{ij}\}_j, q_i, \Delta_{ij}, \tau_{ij}^s\}$, where w_{ij} is the synaptic weights, for the synapse connecting the neuron j with the neuron i. The last three parameters define the function $\alpha(\cdot)$, and represent the normalization factor (q_i), synaptic time constant (τ_{ij}^s), and synaptic delay (Δ_{ij}).

The network is shown in fig. 2; it is organized in the three layers: input, internal layer, and output. The input layer mimics some aspects of the experiments described in [1], precisely the activation of one of the eight targets ($PT_1, \ldots, PT_8$) and the *GO* signal. Therefore, it consists of nine inputs, connected to nine excitatory LIF neurons which convert pulses into spikes. Therefore, the input layer is completely defined by the set of parameters $\mathcal{L}^{in} = \{\mathcal{C}_i^{in}\}_{i=1\ldots9}$, i.e. the set of parameters for the nine input cells.
The internal layer is an approximative representation of the neural population in the 7a area of the parietal cortex, analyzed in details in [1]. The network is composed of N LIF neurons, N_E excitatory and N_I inhibitory cells, and is described by the set of parameters $\mathcal{L} = \{\mathcal{C}_i^E, \mathcal{C}_j^I\}_{i=1,\ldots,N_E,\ j=1,\ldots,N_I}$.
The motor output layer consists of eight analog linear neurons, each of them performing a linear combination of the outputs from the internal layer:

$$U_i^{out}(t) = R \sum_j w_{ij}^{out} \sum_f \alpha^{out}(t - t_j^{(f)}) \qquad (4)$$

The potential $U_i^{out}(t)$ corresponds to the output MT_i. Thus, output layer is defined by $\mathcal{L}^{out} = \{\mathcal{C}_i^{out}\}_{i=1..8}$. Since we rely on LSM paradigm, only w_{ij}^{out} are trained; all the other parameters are fixed in advance.

4. Networks generation and selection procedures

The procedure for constructing networks of desired properties is summarized in the following steps:

1. Randomly generate network parameters that should be fixed in advance.

2. Train the output layer coefficients, w_{ij}^{out}.

3. Evaluate outputs classification error.

4. If the number of outputs with classification error $\leq 10\%$ is > 6 the network is selected, otherwise discarded.

The neuron and synapse parameters are selected according to the approach proposed in [4]. The following parameters are kept constant for all the network layers: $\tau_i^m = 30$ ms, $R_i = 10\ M\Omega$, $\theta_i = 15$ mV, $\tau_i^r = 3$ ms for excitatory and $\tau_i^r = 2$ ms for inhibitory neurons, and $\sigma_i^n = 5$ mV. Also, $J_i \sim Unif[13.5,\ 14.5]nA$ and $u_i^r \sim Unif[13.5,\ 14.5]mV$ for all the layers, where $Unif[\cdot]$ denotes the uniform distribution over the interval given between brackets. The remaining model parameters are set for each layer separately.

The currents, leakage resistances and neuron thresholds of the input layer are scaled in order to ensure the correct conversion of input signals into spikes. Randomly selected 20% of all the possible synaptic connections between the input and the internal layer are set as: $w_{ij}^{in} = 10^{-2}$, $\tau_{ij}^{s,in} \sim Unif[2.5,\ 3.5]$, and $\Delta_{ij}^{in} \sim Unif[0.1,\ 1]$. The remaining connections are removed from the model.
The internal layer is critical for achieving the desired properties and performance. The main goal is to obtain as long memory as possible, while keeping the echo state property [6]. No general design criteria is available in the literature. We adopted the procedure presented in [4], setting 80% of the neurons as excitatory and 20% as inhibitory. Connections topology is random with only 20% of all the possible connections being non-zero. Synaptic weights for the excitatory cells are drawn from a Gaussian distribution with mean w_{mE} and standard deviation $w_{mE} \cdot 10^{-3}$, while for the inhibitory cells Gaussian distribution has mean equal $-2 \cdot w_{mE}$ and the same standard deviation. The other parameters are generated as: $\tau_{ij}^{s,E} \sim Unif[2.5,\ 3.5]ms$, $\tau_{ij}^{s,I} \sim Unif[5,\ 7]ms$, and $\Delta_{ij}^{E},\ \Delta_{ij}^{I} \sim Unif[0.1,\ 1]ms$. The difference in design strategy, with respect to [4] is that the network presented in this work is unstructured, with only the sparsity criteria imposed, and the average value for connection strength w_{mE} is the only parameter to be tuned. The parameters for the output layer are set as: $\tau_{ij}^{s,out} = 20$ ms, $\Delta_{ij}^{out} = 0$ ms.

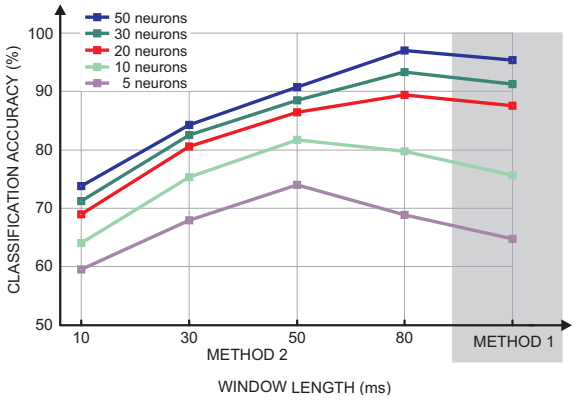

Figure 3: Average classification accuracy for different numbers of available neurons and W.

For networks generated according to the given procedure, the output coefficients w_{ij}^{out} are calculated through the linear regression training procedure. The energy of the output pulses is computed and compared with a fixed threshold value; all the output pulses whose energy is above threshold are considered as activation of the output, while the energy below the threshold means no activation. Finally, w_{mE} is tuned through a grid search optimization in order to maximize the number of selected networks. A network is selected if it has at least 6 operating outputs, i.e. outputs that have the classification error lower or equal to 10%. The procedure results in w_{mE} set to 10^{-3}.

5. SVM classifier training and testing

In order to repeat the analysis presented in [2], we substituted the motor output layer with a block of spike to rate conversion followed by an SVM classifier. The classifier is implemented using the publicly available software package [3]. The two parameters, the kernel width and the regularization coefficient, are chosen through the grid search procedure. In order to model the two different analysis procedures presented in [2] we used two different methods to compute spike rates. In the first method the whole epoch of interest is used to compute a rate for each neuron, and obtain the rate vector R. In the second method, the whole epoch of interest is divided into non overlapping windows of constant length, a spike rate is computed for each neuron, and the rate vector is obtained for each window position. The presented results are obtained for the window size of 10, 30, 50, 80 ms for the second method, and according to the first method. For each window choice, $M\%$ of all the neurons are selected (where $M = 50\%, 30\%, 20\%, 10\%, 5\%$), and their spike trains are used for calculating spike rate vectors. They are, then, used for training and testing the classifier. Its performance is described by classification accuracy.
The average performance for the method two, ob-

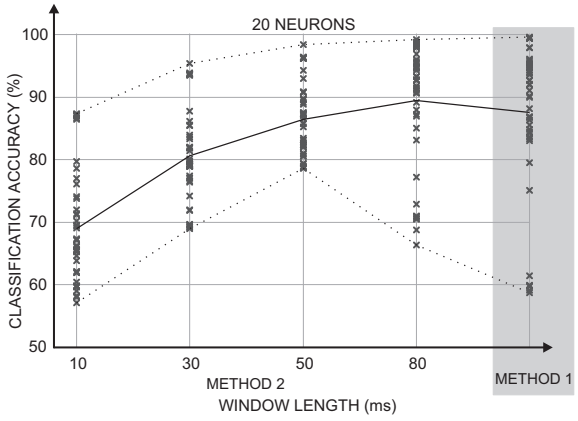

Figure 4: Average classification accuracy for different values of W and 20 neurons available.

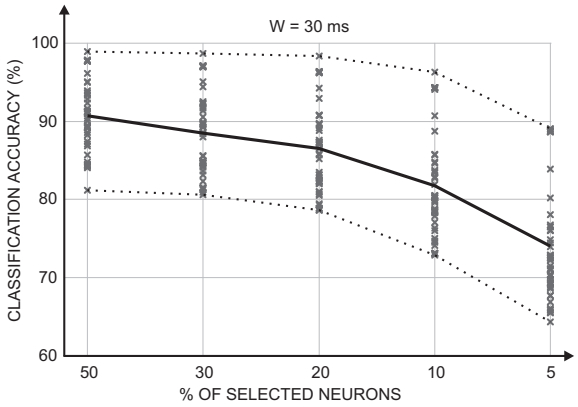

Figure 5: Average classification accuracy for different number of available neurons, with W=30 ms.

tained for different W and the number of available neurons, is shown in Fig. 3. These curves are obtained by averaging the performance of 200 networks of size 100. A drop of average performance is observed when the number of available neurons decreases. In Fig.4 the best, the average, the worst and individual values of a sample of 50 networks are shown: each scatter corresponds to the results obtained with 20 neurons available, averaged over 100 random selections. By increasing W, the average value increases, except when the whole epoch is used. While the average performance for most of the networks is improved, a consistent group of them is performing significantly worse, degradating the average value. This is due to the networks memory, which is not controlled in the generation procedure. In Fig.5 the individual performance for different numbers of available neurons is shown. The decrease in average performance, when reducing the number of available cells, is due to spread of performance distribution and a condensation around the lower bound. The analysis revealed a small group of networks with around 90 % accuracy when only 5 cells are available.

6. Conclusions

In this work we presented an example of bio-inspired network, based on LSM approach [7]. The model relies on LIF neurons and chemical synapses. The network generation is random, with biological motivated tunning of the main parameters. The proposed model is able to learn and solve, with a satisfactory level of accuracy, an approximate representation of the behavioral tasks presented in [1]. The obtained results highlighted the critical role of the number of simultaneously available cells and the length of the window used for the spike rates computation. The presented performance bounds are similar to those obtained by the analysis of the experimental data, given in [2].

Acknowledgments

The first and second author would like to acknowledge the internationalization funding provided by University of Ferrara, Italy and by European Doctorate on Information Technology (EDITH) program, respectively. Thanks are due to A. Battaglia-Mayer and R. Caminiti for letting us use their experimental data, and for many stimulating discussions.

References

[1] A. Battaglia-Mayer, M. Mascaro, R. Caminiti, "Temporal Evolution and Strength of Neural Activity in Parietal Cortex During Eye and Hand Movements", *Cerebral Cortex*, August 2006.

[2] J. Acimovic, A. Battaglia-Mayer, R. Caminiti, M. Hasler, "Automatic Methods for Motor Intention Recognition from Spike Rates", in *Proceedings of NOLTA*, Vancouver, Canada, September 16-19 2007.

[3] C. C. Chang, C. J. Lin, LIBSVM : a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[4] H. Burgsteiner, M. Kr´oll, A. Leopold, G. Steinbauer, "Movement prediction from real-world images using a liquid state machine", *Innovation in Applied Artificial Intelligence*, vol. 3533/2005. Springer Berlin/Heidelberg.

[5] W. Gerstner, W. Kistler, "Spiking neuron models", Cambridge University Press. 2002.

[6] H. Jaeger, M. Lukoevicius, D. Popovici, U. Siewert, "Optimization and applications of echo state networks with leaky-integrator neurons", *Neural Networks*, 20(3):335-352.

[7] W. Maas, T. Natschlager, H. Markram, "Real-time computing without stable states: a new framework for neural computation based on perturbation", *Neural Computation*, 14(11):2531-2560.

[8] J. M. Carmena, M. A. Lebedev, R. E. Crist, E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, M. A. L. Nicolelis, "Learning to Control a Brain-Machine Interface for Reaching and Grasping by Primates", *PLoS Biology*, 1:193-207.