33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

Real-time Neuronal Networks Reconstruction Using Hierarchical Systolic Arrays

Bo Yu, Terrence Mak, Yihe Sun and Chi-Sang Poon

Abstract— **The correlation network of neurons emerges as an important mathematical framework for a spectrum of applications including neural modeling, brain disease prediction and brain-machine interface. However, construction of correlation network is computationally expensive, especially when the number of neurons is large and this prohibits real-time applications. This paper proposes a hardware architecture using hierarchical systolic arrays to reconstruct the correlation network. Through mapping an efficient algorithm for cross-correlation onto a massively parallel structure, the hardware can accomplish the network construction with extremely small delay. The proposed structure is evaluated using Field Programmable Gate Array (FPGA). Results show that our method is three orders of magnitudes faster than the software approach using desktop computer. This new method enables real-time network construction and leads to future novel devices of real-time neuronal network monitoring and rehabilitation.**

I. INTRODUCTION

The emergence of network science provides a power mathematical tool to study the structural and functional networks of the brain. A range of novel models, both hierarchical and scale-free networks, have been proposed based on the inter-relationships between the brain regions, and also based on the recording of neuronal activities. This opens a new avenue to understand and interpret the complex brain, and also enables novel real-time biomedical engineering solution to brain diseases, such as epilepsy [1], Schizophrenia [2] and Alzheimer's disease [3].

Real-time constructing and tracking of the network of activities provides a new dimension of inspection and modeling. For example, epileptic propagation in brains can be tracked effectively by monitoring the functional network evolution. Real-time network analysis would enable a range of novel devices for brain disease monitoring and rehabilitation.

Currently, brain network analysis is performed off-line. This is mainly due to the complex computational requirement for the recorded data. The very first step of network analysis is to construct a relational network between all channels. The computational effort grows quadratic with the number of input channels. With the rapid advance of bio-sensing technology, including multi-electrode arrays (MEA) [4] and voltage-sensitive dye [5], the number of channel is increasing drastically. This exacerbates the computational effort for the network construction.

This paper presents a systolic array structure [6] for real-time correlation network construction. The cross-correlogram based method is employed to reduce hardware cost and provides effective correlation analysis for spike trains. A unit-dimension systolic array is proposed for single-pair spike trains, which is extended to a multi-dimensional systolic network for all pair-wise cross-correlation analysis. The massively parallel architecture significantly reduces the computational delay.

The paper is organized as follows. Section II introduces the correlation network and cross-correlogram. Section III illustrates the cross-correlogram based method for correlation network reconstruction. Section IV presents the structure of hierarchical systolic array. Section V discusses the evaluation results on the scalability, hardware cost and latency.

II. BACKGROUND

A. Correlation network

A correlation network, also called functional network [7], is method to express the correlation relationship among a group of neurons. In the network, edges between nodes represent the correlation between spike trains, which can be either a weighted value or binary value (1 and 0 represent correlated and non-correlated) [8]. Correlation networks has shown specific deviations in the neural network organizations in Schizophrenia [2], Alzheimer's [3] and epilepsy patients [1].

B. Cross-correlogram based correlation

The cross-correlogram is a classical method for finding the correlation between two spike trains. Fig. 1 shows the procedure of obtaining a cross-correlogram. Two spike trains, called target and reference spike train, are aligned and equally divided into a serial of bins in which '1' represents a spike. For each spike in the reference spike train, a window centered at the reference spike is applied on the target spike train. A sub-section of the target spike train is picked. A histogram, namely cross-correlogram, is obtained by accumulating all the sub-sections of the target spike train. If two spike trains are correlated, a peak will be shown in the cross-correlogram, as shown in Fig. 1(b). Otherwise, the histogram is more or less random.

Mathematically, the cross-correlogram is the cross-covariance between the two spike trains over a certain

This work is supported by EPSRC Knowledge Transfer Account.

Bo Yu and Yihe Sun are with Tsinghua National Laboratory for Information Science and Technology, Institute of Microelectronics, Tsinghua University, Beijing 100084, China.

Terrence Mak is with School of Electrical, Electronic and Computer Engineering, Newcastle upon Tyne, NE1 7RU, UK. He is also with the Institute of Neuroscience, Newcastle Biomedicine, at the same University.

Chi-Sang Poon is with Harvard-MIT Division of Health Sciences and Technology, MIT, Cambridge, MA 02139, USA.

978-1-4244-4122-8/11/$26.00 ©2011 IEEE

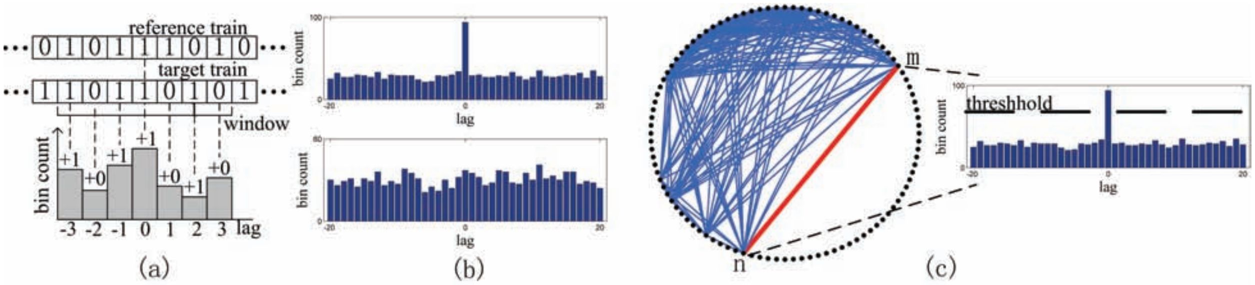

Fig. 1. (a) The procedure of cross-correlogram. (b) Cross-correlogram of spike trains with correlation (upper) and without correlation (lower). (c) An example of correlation network (left), the cross-correlogram between channel m and n (right).

window and can be formulated by Eq. 1,

$$\begin{aligned} CCgram &= \vec{x}^T \cdot \vec{y} \\ &= [cgm(-w), cgm(-w+1), \cdots, cgm(w)] \\ cgm(\tau) &= \sum_{i=1}^{N} x(i)y(i+\tau) \quad \tau \in [-w, w] \end{aligned} \tag{1}$$

where $N \times 1$ vectors, $\vec{x}$, $\vec{y}$, are two spike trains, w is the half window size, $\vec{x}^T \cdot \vec{y}$ represents the cross-correlogram of two spike trains, $cgm(\tau)$ represents the cross-correlogram with timing lag τ. Only multiplication and accumulation operation are required. Furthermore, binary multiplication is actually logic-and operation whose hardware implementation is simple. As a result, in terms of hardware cost cross-correlogram is much cheaper than the well known Pearson's correlation that involves division and square root operation and has been used in [8] for constructing correlation network.

A threshold based method is proposed to quantitatively determine whether two spike trains are correlated through the results of cross-correlogram. The threshold is set to $\frac{k \sum_{i=-w}^{w} cgm(i)}{2w+1}$, k is a constant empirical value. If the max value of cross-correlogram is larger than the threshold, the two channels are considered to be correlated.

A correlation network is always obtained by calculating all pair-wise correlations of the spike trains and exemplified by Fig. 1(c). Nodes and edges represent spike trains (neurons) and correlated relationship. Denote M spike trains as $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_M$, $\vec{x}_i$ is $N \times 1$ vector. The pair-wise correlation between all electrodes can be formulated by Eq. 2,

$$\begin{aligned} CCmtrix &= (\vec{x}_1 \vec{x}_2 \ldots \vec{x}_M)^T \cdot (\vec{x}_1 \vec{x}_2 \ldots \vec{x}_M) \\ &= \begin{pmatrix} \vec{x}_1^T \cdot \vec{x}_1 & \vec{x}_1^T \cdot \vec{x}_2 & \ldots & \vec{x}_1^T \cdot \vec{x}_M \\ \vec{x}_2^T \cdot \vec{x}_1 & \vec{x}_2^T \cdot \vec{x}_2 & \ldots & \vec{x}_2^T \cdot \vec{x}_M \\ \vdots & \vdots & \ddots & \vdots \\ \vec{x}_M^T \cdot \vec{x}_1 & \vec{x}_M^T \cdot \vec{x}_2 & \ldots & \vec{x}_M^T \cdot \vec{x}_M \end{pmatrix} \end{aligned} \tag{2}$$

where $CCmatrix$ is a matrix containing all pairs of correlations. Each entry in the matrix is a cross-correlogram between two spike trains.

C. Computational complexity analysis

The correlation between a signal and itself is not useful. Correlation between $\vec{x}_i$ and $\vec{x}_j$ is the same as correlation between $\vec{x}_j$ and $\vec{x}_i$. As a result, for the matrix in Eq. 2, only the components upper the diagonal or below the diagonal part are useful. When there are M spike trains, $\frac{M(M-1)}{2}$ cross-correlation calculations are required. The number of calculations of correlation is quadratic to the number of spike trains. Constructing the correlation network presents a large computational load under a large number of spike trains. A long period of computing latency is unavoidable if using traditional serial computing technique. In order to reduce computing latency, we propose to use a hierarchical systolic array structure to calculate correlation network. Our method can turn the quadratic relationship to a linear one.

III. Hardware Architecture

Systolic array is a specialized form of parallel computing. Identical processing units are organized in a regular network. Each processing unit only communicates with its neighbor units. Pipelines are inserted in the communication channel, which make data flow through the network rhythmically and regularly.

A. Systolic array for cross-correlogram computation

The structure of the correlation hardware is shown in Fig. 2. Spike trains, $\vec{x}$ and $\vec{y}$, are fed into two delay chains. Delay chains coordinate two spike trains and generate all the pairs of signals with certain timing lags for correlation analysis. One delayed spike signal, y_i, is broadcasted to each logic-and gate as one input. Another input of each logic-and gate is delayed $\vec{x}$ with a certain timing lag to y_i. Logic-and gates are used here to perform binary multiplication. Hardware adders accumulate the results of logic-and gate. Results of adders are stored in registers, 'R', for accumulating. The number of and-adder pairs equals to the window size, so cross-correlograms with different timing lags are calculated concurrently. The latency of cross-correlogram calculations using this structure depends on the length of spike trains and the window size. The latency is $(l + 2w) \times T_{clock}$, where l and w are the length of signal and the window size, T_{clock} is the clock cycle.

B. Multi-dimensional systolic arrays

As the number of spike trains increases, the computing delay will grow quadratically if using single correlation hardware. In this paper, a two-dimensional systolic array that

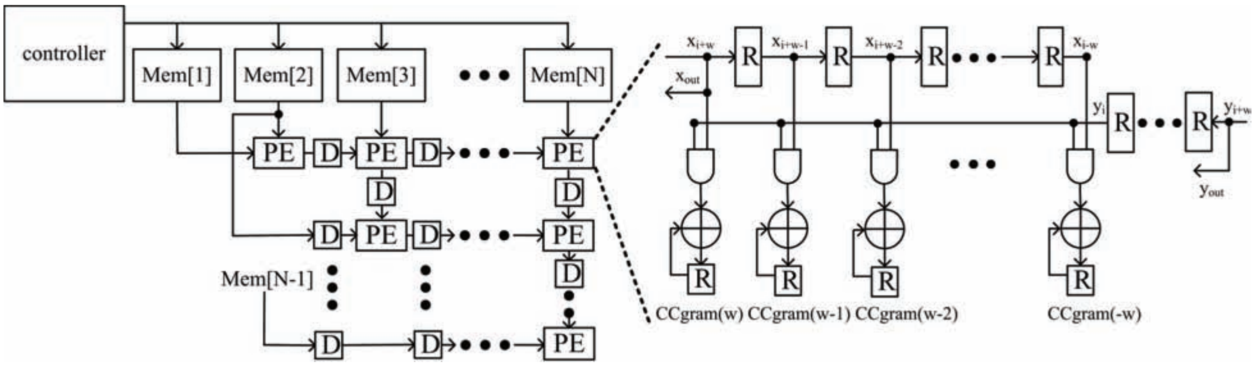

Fig. 2. Structure of one-dimensional array for single pair wise cross-correlogram (right) and two-dimensional array for calculating correlation network (left). In the right sub-figure, 'R' represents registers.

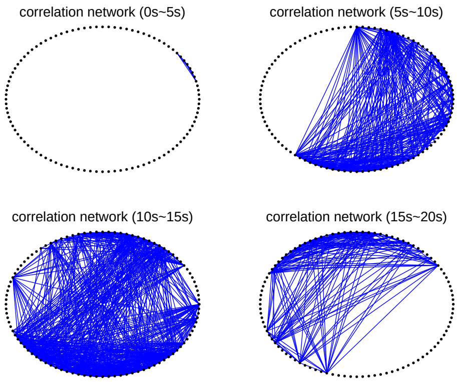

Fig. 3. Correlation networks across one hundred trains during a bursting event on the retina.

embeds much identical pair-wise cross-correlogram hardware is used to speed up the computation.

The structure of the array that calculates the correlation network of N spike trains is shown in Fig. 2. In the figure, $Mem[i], i \in [1, N]$ stores the ith spike train. The controller reads spike trains out of memories and pumps them into the array. Delay units are inserted between two neighboring processing elements (PEs) to form pipelined structure that enables high throughput. Each processing element (PE) implements the signal pair-wise cross-correlogram corresponding to an upper diagonal entry in Eq. 2. For example, the PE in row 1, column 2 corresponds to the entry in row 1, column3 of Eq. 2 and computes the correlation between the first and the third spike train. The number of PEs is equivalent to the number of upper diagonal entries in Eq. 2, so for N spike trains, $\frac{N(N-1)}{2}$ PEs are required. The latency of the structure depends on the length of a spike train, the number of spike trains and the window size. The latency is

$$latency = [l + 2(n-1) + w] \times T_{clock} \tag{3}$$

where l is the length of signal, n is the number of spike trains, w is the window size and T_{clock} is the clock cycle. The latency of the structure is linear to the number of channel.

C. Summary

The proposed hierarchical systolic array explores computing parallelism in the cross-correlogram calculation and correlation network construction. The paralleled structure speeds up the computing and makes the computing latency linear to the number of spike trains. The processing elements in the systolic array communicate regularly with their neighbor. Because of the regular topology, the array can be easily scaled up with the number of spike trains. However, the number of PE is square to the number of spike trains. As a result, the hardware cost will become large as the number of electrodes increases.

IV. Results

We apply cross-correlogram on the retinal data. The data obtained from CARMEN project was used for studying retinal bursts and waves [9]. The data are equally divided by bins (e.g. 40ms). The window size of cross-correlogram is 20 bins. If the maximum value of cross-correlogram in the window is three times larger than the average value, two spike trains are considered to be correlated. Fig. 3 shows the correlation network extract across 100 spike trains during a bursting event on the retina.

With massive hardware arithmetic units, logic and memory resources, modern FPGAs are well suit for implementing complex neural signal processing algorithms and large scale parallel structures. The reconfigurable ability makes FPGAs more flexible than specific hardware. FPGAs have been widely used in neural signal processing applications. In this paper, Xlinix FPGA is used to evaluate the proposed structure. The targeting device is Xilinx Virtex6 xc6vlx760. The design tool is Xilinx ISE.

In order to study the scalability of the structure, proposed arrays with different window size and channel number are implemented on the FPGA. Fig. 4 (a) and (b) shows the relationships between logic resources (in terms of the number of LUTs, namely look-up table, which is the basic logic resource of FPGA), storage resources (in terms of the number of BRAMs, namely block memory) and the number of spike trains. Controller and PE arrays are implemented by LUTs, and BRAMs are for storing spike trains. As the number of

TABLE I

COMPUTING DELAY COMPARISON BETWEEN PROPOSED ARRAY AND SOFTWARE[1]

N_{ch}	4	8	12	16	20	24	28	32
$Delay_{array}(\mu s)$	20.16	20.32	20.48	20.64	20.8	20.96	21.12	21.28
$Delay_{software}(\mu s)$	8.6×10^3	4.1×10^3	1.02×10^4	1.8×10^4	2.8×10^4	4.1×10^4	5.5×10^4	7.4×10^4

Note: 1. Software is a Matlab program implemented on Intel Core I5 650 (@3.2 GHz). 2. N_{ch}: the number of channel.

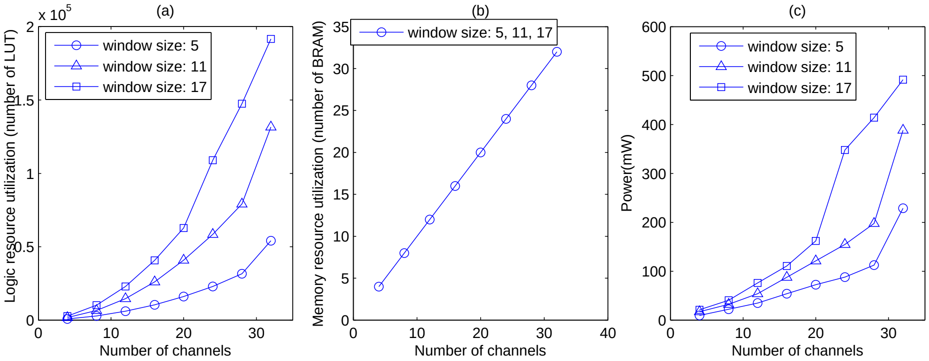

Fig. 4. (a) Logic resource utilization of arrays (in terms of the number of LUT). (b) Memory resource utilization of arrays (in terms of the number of BRAM). (c) Power consumption of arrays.

spike trains grows up, the logic resources increase quadratically. It is mainly because that PE arrays grows quadratically as the number of signals increase. The consumed memory resources are linear to the number of spike trains. The window size also has impact on the logic resources. At a certain number of channels, the LUT resources grow up as the window size increasing. The window size does not affect the memory resources. Fig. 4(c) shows the relationship between the number of spike train and power consumption of the array. The power consumption grows up as the number of channels increases. The array consumes more power with larger window size.

The computational delay of the array that is the time spent on computing all correlations between electrodes is determined by Eq. 3. In our experiment, spike trains contain 1000 bins and clock frequency is 50 MHz. The latency of arrays with different number of channels is shown in Table I. As the number of channels growing up, the computation delay increase linearly. We compare our structure with the high performance PC in terms of computational delay. The delay of PC is obtained by measuring the running time of a Matlab programme implementing the cross-correlogram based correlation network calculation on Intel Core I5 650. From Table I, we can see that the calculation delay of PC increase quadratically. The gap between the performance of the PC and the systolic array grows larger as the number of channels increases. When the number of channel is 32, the systolic array is almost 3500 times faster than PC.

V. CONCLUSION

In the paper, a novel hardware architecture utilizing hierarchical systolic array is proposed for re-constructing correlation network from multiple spike trains. The parallelism in the network construction algorithm is exploited and implemented using a hierarchical systolic array. Because of the massively parallel architecture, the correlation network construction is improved by three orders of magnitudes when compared with software. This novel hardware architecture design leads to future portable device for real-time brain monitoring and neuro-rehabilitations.

VI. ACKNOWLEDGMENTS

The authors gratefully acknowledge the inspiring discussion with Dr. Marcus Kaiser and Dr. Jennifer Simonotto.

REFERENCES

[1] S. Ponten, F. Bartolomei, and C. Stam, "Small-world networks and epilepsy: graph theoretical analysis of intracerebrally recorded mesial temporal lobe seizures," *Clinical Neurophysiology*, vol. 118, pp. 918–927, 2007.

[2] S. Micheloyannis, E. Pachou, C. J. Stam, M. Breakspear, P. Bitsios, M. Vourkas, S. Erimaki, and M. Zervakis, "Small-world networks and disturbed functional connectivity in schizophrenia," *Schizophrenia Research*, vol. 87, pp. 60–66, 2006.

[3] C. Stam, B. Jones, G. N. amd M Breakspear, and P. Scheltens, "Small-world networks and functional connectivity in alzheimers disease," *Cerebral Cortex*, vol. 17, pp. 92–99, 2007.

[4] M. Gandolfo, A. Maccione, M. Tedesco, S. Martinoia, and L. Berdondini, "Tracking burst patterns in hippocampal cultures with high-density cmos-meas," *Journal of Neural Engineering*, vol. 7, no. 5, 2010.

[5] C. Petersen, A. Grinvald, and B. Sakmann, "Spatiotemporal dynamics of sensory responses in layer 2/3 of rat barrel cortex measured in vivo by voltage-sensitive dye imaging combined with whole-cell voltage recordings and neuron reconstructions," *Journal of Neuroscience*, vol. 23, pp. 1298–1309, 2003.

[6] M. Gandolfo, A. Maccione, M. Tedesco, S. Martinoia, and L. Berdondini, "Why systolic architectures?," *Journal of Computer*, vol. 15, pp. 37–46, 1982.

[7] O. Sporns, D. Chialvo, M. Kaiser, and C. O. Hilgetag, "Development and function of complex brain networks," *Trends in Cognitive Science*, vol. 8, pp. 418–425, 2004.

[8] P. Ribeiro, J. Simonotto, M. Kaiser, and F. Silv, "Parallel calculation of multi-electrode array correlation networks," *Journal of Neuroscience Methods*, vol. 184, pp. 357–364, 2009.

[9] B. K. Stafford, A. Sher, A. M. Litke, and D. A. Feldheim1, "Spatial-temporal patterns of retinal waves underlying activity-dependent refinement of retinofugal projections," *Neuron*, vol. 64, pp. 200–212, 2009.