A Model of Spike Neuron Oriented to Hardware Implementation

Andrey V.Gavrilov
Dept. of Computer Engineering
Novosibirsk State Technical University
Novosibirsk, Russia
neurotechlab@yandex.ru

Valeriy M.Kangler, Mikhail Katomin, Konstantin Panchenko
Motiv Ltd.
Novosibirsk, Russia
neurotechlab@yandex.ru

***Abstract*— One of most perspective and popular area of neural networks is neuromorphic computing dealing with development of brain-like spike neural networks oriented on hardware implementation. Such neural networks will provide replacement of computers with Von Neumann architecture in such fields as computer vision and control of autonomous robots. In this paper we suggest one model of spike integrate and fire neuron for development of such neural networks. This model is distinguished by simple arithmetic operations, providing, short-long controlled memory of integrated input signals and controlled refractory period for output. Proposed model will provide simple hardware implementation.**

Keywords— Neural Networks, Neuron, Machine Learning, Neuromorphic Computing

I. INTRODUCTION

Last years researchers and developers are interesting in neuromorphic technologies focusing on development of hardware spike neural networks with behaviour similar to real neocortex [1-6]. Most significant reason is rise of intelligent autonomous robots and visual systems demanding processing of visual information in real time. Besides neuromorphic networks aim to employ for processing of big data. Implementation of such processing in conventional computer with Von Neumann architecture is very ineffective, in particular, power-consuming. Besides autonomous robots demand low power consumption.

Known several powerfully supported programs and projects in this field:

- Project DARPA Synapse (DARPA's Systems of Neuromorphic Adaptive Plastic Scalable Electronics initiative) (http://www.artificialbrains.com/darpa-synapse-program) providing by IBM Research and HRL Laboratories with cooperation of some universities of USA. It was started in 2008 [7]. The goal of this project is to design a neuromorphic chip which is able to replicate a mammalian brain in size, functionality and power consumption: it should be able to recreate 10^{10} neurons with 10^{14} synapses consuming $1KW$ of electrical power and occupying $2dm^3$ (liters) of space.

- Project HBP (Human Brain Project) (https://www.humanbrainproject.eu/en_GB) of EU providing by different scientific centres and universities of Europe and including subprojects Neuromorphic Computing, Cognitive Architectures, Neurorobotics, Neuroinformatics and others.

- Project SpiNNaker (http://apt.cs.manchester.ac.uk /projects/SpiNNaker/project) of UK providing by University of Manchester, University of Southampton, University of Cambridge, University of Sheffield, ARM Ltd, Silistix Ltd, Thales. This project is a hardware-based, real-time, universal, neural network simulator following an event-driven computational approach [8]. It involves the design of a chip and the development of dedicated software to simulate neural networks.

- Project of company Qualcomm (http://www.slashgear.com/qualcomm-zeroth-processors-official-mimicking-human-brain-computing-14301263/).

- Project MONETA of Hewlett Packard [5]. This project aims to develop spike neural networks based on memristors.

- Blue Brain Project of EPFL (http://bluebrain.epfl.ch/) (from 2005) aims to provide a computational substrate for molecular-level simulations that present biological realism. The goal of this platform is to simulate the brains of mammalians with a high level of biological accuracy and, ultimately, to study the emergence of biological intelligence [9].

- Project Neurogrid of Group "Brains-In-Silicon" and Stanford University [10]. The core of this simulator is a neuromorphic analogue chip simulating leaky integrate-and-fire neurons in real-time [11]. This neuro chip is a hardware simulator running in real-time without learning capabilities.

- Project FACETS/BrainScales (http://facets.kip.uni-heidelberg.de/images/4/48/Public--FACETS_15879_ Summary-flyer.pdf) of some universities leading by Helderberg University. This project delivered wafer-scale integration of neuromorphic chips which simulate adaptive exponential leaky integrate-and-fire neurons [12].

Most of powerful and perspective of them are first three projects. The project SYNAPSE supports two ways of research: development of spike neuron networks in traditional digital VLSI technology [7], [13] and analog-digital technology based on memristors [14], [15], some novel devices appropriate for simulation of synapses of neurons.

Although usage of memristors seems as very perspective and widely developing approach, the digital approach is more realistic now. Technology of memristors will be usable just through about 7-10 years. So we focus on digital CMOS technology for implementation of spike neural networks.

Existing neurochips and models of neuron employed inside it have any disadvantages. For example, chip developed in 2014 by scientists of IBM Research TrueNorth [14], [16] with 5,4 billions of transistor supporting implementation of about 1 million neurons with 256 millions connections between them has not any capabilities to learn during exploitation. All learning to solve any task is provided by host computer with conventional architecture and special software tools previously. So that can not provide opportunity to adapt with changing environment during exploitation of neural network.

Besides in existing models of spike neurons after firing of neuron a level of stimulus (membrane potential) resets to zero. We offer another mechanism to implement a refractory period of neuron based on changing of threshold. We think that this approach provides information processes more flexible and similar to existing in biological neuron.

II. Proposed Model of Neuron

Analyzing problem of neural networks implementation in FPGA may be formulated follow requirements to model of neural network:

1) only integers or at least real numbers with fix point should be used,

2) only simple arithmetic operations such as adding and subtraction or at least minimization of usage of multiplication, should be used,

3) discrete time (clock rate) should be used,

4) ensuring demonstration of behaviour basically similar to behaviour of real biological neuron as we understand it in current time, i.e. must integrate in space and time input signals and generate output signals as frequency modulation of level of stimulus.

Most of existing models of spike integrate-and-fire neurons aim to similar biological neuron carefully as possible, for example, model of Hodgkin and Huxley [2] and of Izhikevich [17]. These models are not satisfied requirements described above.

Our proposed model of neuron satisfying upper requirements has structure shown in Figure 1.

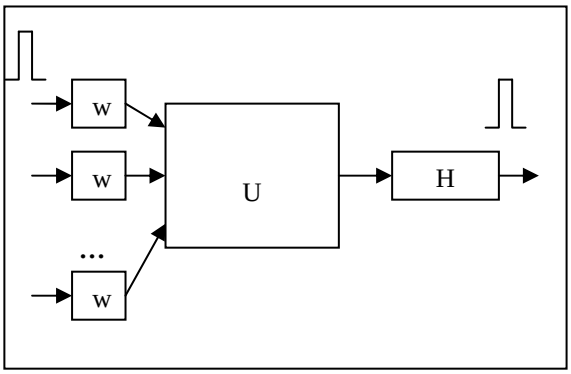


Fig. 1. Model of Spike Neuron.

Output signal of neuron on moment of time t is described as:

$$S(t) = \begin{cases} 1, U(t) > H \\ 0, U(t) \le H \end{cases}, \qquad (1)$$

where H – threshold of neuron,

$U(t)$ – level of stimulation of neuron in t,

$S(t)$ – output signal of neuron in t.

Level of stimulus of neuron is described as:

$$U(t) = U(t-1) + \sum_i w_i r S_i(t-1) - \Delta U, \qquad (2)$$

where r is rate of increasing of U by one input pulse. This parameter is constant for all neurons in contrast to w_i.

Level of stimulus of neuron depends on weighted stimulus of inputs in past time and decreases in every tact by ΔU. Every signal from any i-th input increases level of stimulus adding w_i to level of stimulus. Thus neuron remembers old signals in level of stimulus and little by little forgets them. In other words, more fresh input signals more influence on possibility of firing of neuron.

To provide frequency modulation of output signal the threshold of neuron is changing in accordance with following formulas:

$$H(t) = \begin{cases} H_{\max}, S(t) = 1 \\ H(t-1) - \Delta H, H(t-1) - \Delta H > H_{\min} \\ H_{\min}, \end{cases} \qquad (3)$$

Figure 2 illustrates behaviour of proposed model with assumption that the neuron has 3 inputs: from Sensor 1, Sensor 2 and another neuron. Sensor 1 generates pulse at decreasing of any measured value and Sensor 2 – at increasing of one. Figure 2 shows frequency modulated output signal depending of stimulus level of neuron and memorization and forgetting of input signals in neuron (its stimulus level).

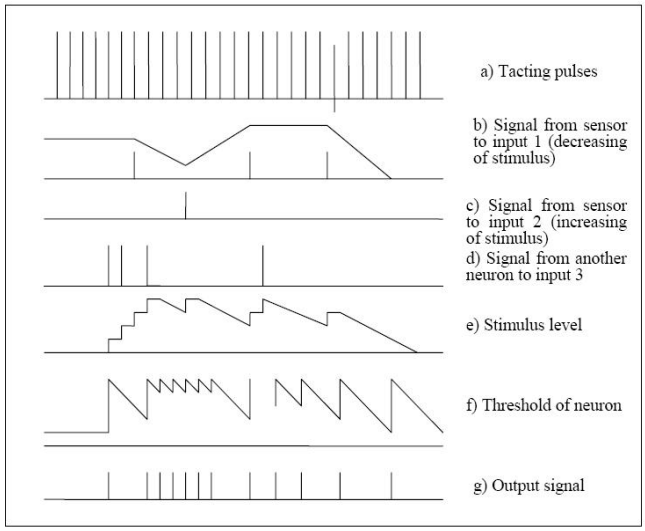


Fig. 2. Diagram of signals in proposed model of neuron.

Thus proposed model is characterized by following parameters:

1) Level of stimulus *U* (supposed integer number from 0 to 255);

2) Level of threshold *H* (supposed integer number from 0 to 255);

3) Value of weights *w* (supposed integer number from -127 to +127);

4) Rate of decreasing of threshold ΔH (supposed integer number from 0 to 255);

5) Rate of forgetting of input level of stimulus ΔU(supposed integer number from 0 to 255);

6) Rate of increasing of *U* by one input pulse;

7) Number of inputs (constrained by size of core and architecture of links between neurons in FPGA implementation).

These parameters can be setting and changing during training of neural network in host computer before exploitation. We suppose that parameter 3 (weights) and probably 6 can be changed during exploitation providing adaptation of neural network to dynamical environment and features of solving task.

Algorithm of Neuron in any moment *t* is following:

```
procedure Neuron;
new_U:=U;
new_S:=0;
for i from 1 to Ns
    if Si=1 then
        new_U:=new_U+wi;
if new_U>=Umax then
      U:=new_U
    else
      U:=Umax;
if U>H then
begin
    new_S:=1;
    H:=Hmax;
end
else
     H:=H-ΔH;
```

Proposed model of neuron provides two different regimes of usage depending on values of parameters:

- frequency modulation of stimulus, integrating in time from weighted inputs
- recognition of any pattern (event) consisting of single signals or series of signals with enough frequency from weighted inputs.

First regime may be used mostly in low layers of neural network to convert level of signal to series of pulses with corresponding frequency. Second regime may be employed to recognize any feature or fragment of image and image in whole.

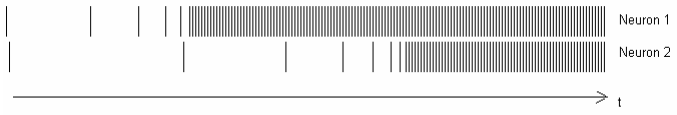
III. Simulation of Simple Neural Networks Based on Proposed Model

Some previous experiments with program model in C++ of this neuron were conducted. In all figures every line corresponds to activity of certain neuron (number of neuron rises from top to bottom of figure). Figure 3 shows any features of signal processing by simple neural networks from 2 neurons.

In Figure 3 a) first neuron has one synapse obtaining signal from sensor every tact with weight 1 and second neuron has one synapse connected with output of first one.

In Figure 3 b) first neuron has one synapse obtaining signal from sensor every tact with weight 1 and another synapse from output of neuron 2 with weight -1 (inhibitory synapse), second neuron has one synapse connected with output of first one. Thus this network has simple feedback. Another parameters of this simple network is following: $\Delta H=5$, $\Delta U=1$, $r=5$, $H_{max}=255$, $U_{max}=255$.

a)



b)

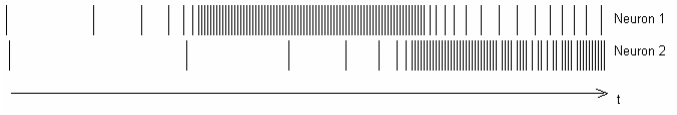


Fig. 3. Signals in simple 2-neurons network.

Figure 4 shows results of simulation of neural network consisting of 11 neurons: 10 of them are connecting by inputs with 10 different sensors with weights 20 and 11th neuron is

connecting with outputs of these 10 neurons with weights 5. Sensors in this experiment are without stimulus.

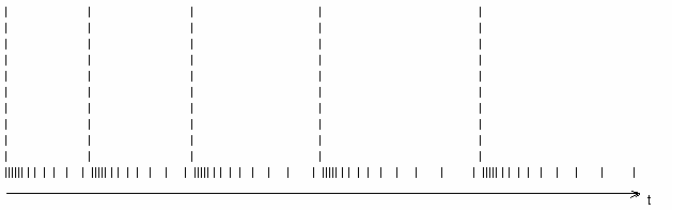


Fig. 4. Simulation of neural network from 11 neurons

Figure 5 shows results of simulation of same neural network, but sensors 1, 2 and 3 in series started generation of signals and then sets off in the reverse order.

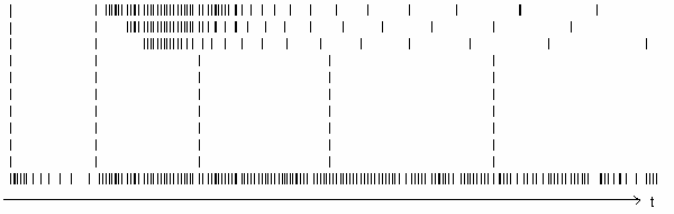


Fig. 5. Simulation of neural network from 11 neurons with activity of sensors 1, 2 and 3.

Experiments demonstrate that to obtain more appropriate frequency modulation it is necessary to use degenerative feedback. Neuron with such feedback with weight -20 is shown in figure 6 and simulation in figure 7 (compare with figure 2a without feedback).

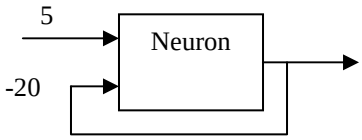


Fig. 6. Neuron with degenerative feedback.

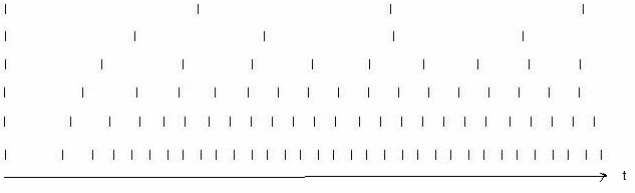


Fig.. 7. Frequency modulation of input signal by neuron with degenerative feedback (input signal is increasing from up to down).

Figure 8 shows similar experiment as in figure 5 but with degenerative feedback for input neurons and output neuron.

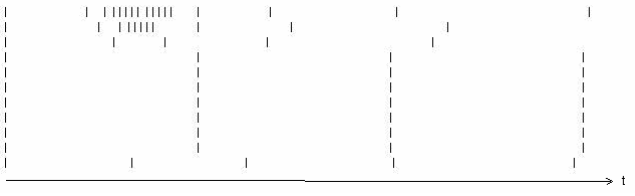


Fig. 8. . Simulation of neural network including 11 neurons with activity of sensors 1, 2 and 3 with degenerative feedback.

IV. On training of proposed model

The following list summarizes the desired properties of the learning method to be potentially suitable for the considered class of applications:

1) Ability to reconstruct precise timing of individual spikes in spike sequences;

2) On-line processing ability;

3) Stability of the optimal solution;

4) Locality.

To train this model may employ known algorithm STDP (Spike-Timing Dependent Plasticity) [18] as modified for spike networks Hebb's rule. For example, this method was employed for object recognition and motion anticipation in [19]. Popular for classical neural networks training algorithm EBP (Error Back Propagation) is not available for threshold based spike neurons. But it is possible to use Extreme Learning Machine (ELM) for training of classical feed forward neural network (or Multi Layer Perceptron) with one hidden layer (ELM) [20] or more (Hierarchical ELM) [21], [22] with threshold based neurons, and therefore to train spike neural network with hidden layers. Basic principle of ELM is training of only output layer and determination of weights in hidden layers randomly. If the learning of neural network is conducted basically previously in host computer it is possible to use genetic algorithms to train our network.

For our model we offer to employ some modified STDP for training of neuron described by follow:

$$w_i(t) = w_i(t-1) + kS(t-1)\sum_p f(p)I_i(p), \qquad (4)$$

where: k – learning rate,

$S(t)$ – output signal of neuron,

$f(p)$ – function of weights of past input pulses (decreasing in accordance with decreasing of time), $p=(t\text{-}1, \ldots, N_p)$;

$I_i(p)$ – input signal (0 or 1) in synapse I in past time p.

Moments of time p must be limited by enough few number N_p essentially if this learning is conducted during exploitation.

This modified STDP method provides more flexible capabilities to adapt weights.

Conclusions

Proposed in this paper model of spike neuron for hardware implementation is similar to suggested earlier leaky-integrate-and-fire (LIF) neuron model [16], [19]. Our model differs from it by another mechanism of refractory period based on changing of threshold instead membrane potential. This feature provides memory in neuron about past input pulses and so expands possible functionality of neuron and neural network in whole.

Previous experiments with software simulation show that proposed model may be used for different coding of

information: by frequency modulation (with degenerative feedback in neuron) and representation of any event by single pulse.

Proposed in this paper model of spike neuron and neural networks is implementing now by program simulation in C++ and in Python for large number of neurons. Usage and verification of it is planned in tasks of computer vision (object recognition) and navigation of mobile robots.

Further plans are dealing with implementation of proposed model of neuron and neural networks in FPGA. Hardware implementation will allows to solve in real time such complex problems with big data as services of data warehouse of environment and infectious diseases [23], geoinformation technology for assessment of ecological risks [24].

REFERENCES

[1] W. Maass, W., "Networks of Spiking Neurons: The Third Generation of Neural Network Models," NeuralNetworks, vol. 10, no. 9, pp. 1659-1671, 1997.

[2] W. Gerstner and W. M. Kistler, Spiking Neuron Models. Single Neurons, Populations, Plasticity, Cambridge University Press, 2002.

[3] Atsumi Masayasu, "Sequence Learning and Planning on Associative Spiking Neural Network," In Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN'02), 2002, pp.1649-1654.

[4] R. Brette and E. Guigon, "Reliability of spike timing is a general property of spiking model neurons," Neural Computation, 15(2), pp. 279-308, 2003.

[5] G. Snider et al, "From Synapses to Circuitry: Using Memristive Memory to Explore the Electronic Brain," IEEE Computer, vol. 44, iss. 2, Iss. 2, pp. 21–28, 2011.

[6] A. Gruening and S.M. Bohte, "Spiking Neural Networks: Principles and Challenges," In Proceeding of European Symposium on Neural Networks (ESANN 2014), Bruges (Belgium), pp 1-10, April 2014.

[7] A. Cassidy, A.G. Andreou and J. Georgiou, "Design of a one million neuron single FPGA neuromorphic system for real-time multimodal scene analysis," In Proc. of 45th Annual Conference on Information Sciences and Systems (CISS), Atlanta, GA, USA, pp. 1-6, 23 - 25 March, 2011.

[8] S.B. Furber, D.R. Lester, L.A. Plana, J.D. Garside, E. Painkras, S. Temple and A.D. Brown, "Overview of the SpiNNaker System Architecture," IEEE Transactions on Computers, vol. 62, issue 12, pp. 2454 – 2467, 2013.

[9] H. Markram, "The blue brain project," Nat Rev Neurosci, 7 (2), pp. 153–60, 2006.

[10] S. Choudhary, "Silicon Neurons that Compute," / S. Choudhary, S. Sloan, K. Boahen [et al.], In International Conf. on Artificial Neural Networks, LNCS, Springer, Heidelberg, vol. 7552, pp. 121–128, 2012.

[11] R. Silver, K. Boahen, S. Grillner, N. Kopell and K.L. Olsen, "Neurotech for Neuroscience: Unifying Concepts, Organizing Principles, and Emerging Tools," The Journal of Neuroscience, 27 (44), pp. 11807–11819, 2007.

[12] J. Schemmel, D. Bruderle, A. Grubl, M. Hock, K. Meier, S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," In: Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on. IEEE, pp. 1947–1950, 2010.

[13] A.S. Cassidy, J. Georgiou, A.G. Andreou, "Design of silicon brains in the nano-CMOS era: Spiking neurons, learning synapses and neural architecture optimization," Neural Networks, 45, pp. 4-26, 2013.

[14] Q. Wu, B. Liu, Y. Chen, H. Li, Q. Chen and Qiu Qinru, "Bio-Inspired Computing with Resistive Memories, Models, Architectures and Applications," In Proc. of International Symposium on Circuits and Systems (ISCAS), Melbourne< Australia, 2014.

[15] X. Wu, V. Saxena and K. Zhu, „A CMOS Spiking Neuron for Dense Memristor- Synapse Connectivity for Brain-Inspired Computing," In Proc. of International Joint Conference on Neural Networks (IJCNN) Vancouver, Canada, 2015.

[16] P.A. Merolla et al,"A million spiking-neuron integrated circuit with a scalable communication network and interface," Science, vol. 345, no. 6197, pp. 668–673, 2014.

[17] E. Izhikevich,"Simple model of spiking neurons," IEEE Transactions on Neural Networks, vol. 14, no. 6, pp. 1569–1572, 2003.

[18] W. Senn, M. Tsodyks and H. Markram,"An algorithm for synaptic modification based on exact timing of pre- and post-synaptic action potentials," in Artificial Neural Networks (ICANN'97), Lausanne, Switzerland, pp. 121-126, 1997.

[19] A. Nere, U. Olcese, D. Balduzzi and G. Tononi, "A Neuromorphic Architecture for Object Recognition and Motion Anticipation Using Burst-STDP," PLOS One, v. 7(5), pp. 1-17, 2012.

[20] Guang-Bin Huang, Qin-Yu Zhu, K. Z. Mao, Chee-Kheong Siew, P. Saratchandran and N. Sundararajan,"Can Threshold Networks be Trained Directly?," IEEE Trans. on Circuits and Systems—II: Express Briefs, vol. 53, no. 3, pp. 187-191, March 2006.

[21] Guang-Bin Huang, Zuo Bai, Lyanaarachchi Lekamalage Chamara Kasun and Chi Man Vong, "Local Receptive Fields Based Extreme Learning Machine," IEEE Computational intelligence magazine, vol. 10, no. 2, pp. 18-29, 2015.

[22] J. Tang, C. Deng and G.-B. Huang, "Extreme Learning Machine for Multilayer Perceptron," IEEE Transactions on Neural Networks and Learning Systems, vol PP, is. 99, May 2015.

[23] S. Yun, O. Alsova, N. Chistyakov et al, "Data Warehouse: environment and indections diseases," in Proceedings of 6th International Forum on Strategic Technology (IFOST-2011)", IEEE, Harbin, 22-24 Aug. 2011, pp. 792-795.

[24] V.V. Gubarev, M.S. Khairetdiniov, S.A. Avrorov, G.M. Voskoboinikova, A.A. Yakimenko, "Geoinformation technology for assesment of ecological risks," 12th International Conference on Actual Problems of Electronic Instrument Engineering (APEIE-2014), Novosibirsk, pp.38-43, 2014.