A Modified Izhikevich Model For Circuit Implementation of Spiking Neural Networks

Arash Ahmadi and Mark Zwolinski
School of Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ, UK
Email:{aa5,mz}@ecs.soton.ac.uk

***Abstract*—The Izhikevich neuron model reproduces the spiking and bursting behaviour of certain types of cortical neurons. This model has a second order non-linearity that makes it difficult to implement in hardware. We propose a simplified version of the model that has a piecewise-linear relationship. This modification simplifies the hardware implementation but demonstrates similar dynamic behaviour.**

***Index Terms*—Neural networks, Neural network hardware.**

I. Introduction

Spiking neural networks have received much attention in the artificial neural network community during the past few years, due to their behavioural resemblance to real life neurons. Motivated by biological discoveries, pulse-coupled neural networks with spike-timing are considered an essential component in information processing by the brain. Accordingly, many different models have been presented for spiking neural networks to reproduce their dynamical behaviour. These models are based on a bio-chemical inspection of the neurons' structures and mostly are expressed in the form of differential equations. Although detailed neuron models, [1], can imitate most experimental measurements to a high degree of accuracy, these models are difficult to use in large scale artificial spiking neural networks, due to their complexity, [2]. Consequently, simplified models are highly popular for studies of neural information coding, memory and network dynamics. Lapicque, [3], proposed that spikes are generated when the integrated sensory or synaptic inputs to a neuron reach a threshold value; this model is called Integrate and Fire (IF). IF has become one of the most influential models in neurobiology, giving a simple mechanistic explanation for basic neural operations. Advances in experimental techniques have shown, however, that the IF model is far from accurate in describing real neurons. Izhikevich, [4], has developed a class of models of spiking neurons that balances the computational efficiency of IF models with the biological plausibility and versatility of Hodgkin-Huxley type models [1].

VLSI systems are considered to be strong choices for the direct implementation of neuro-inspired systems. In this approach, electronic components and circuits are utilised to mimic neurological dynamics. Because of the high performance and well developed technology, a VLSI implementation enables rapid prototyping of neural algorithms to test theories of neural computation, structure, network architecture, learning, and plasticity and also simulation of biologically inspired systems in real-time operation. This is of particular interest for sensory processing systems and biologically-inspired robotics.

A variety of analogue, digital and software-based implementations of spiking neural networks has been presented. While analogue implementations can replicate neural dynamics down to the ion channels in the neural membrane and are fast and efficient, they are inflexible and require a long development time [5], [6]. On the other hand, software-based systems implement neurobiological functions using standard microprocessors [7]. These systems are flexible and biologically realistic, but are often large, computationally slow, and have a high power consumption. Recently, as a midpoint in the design space, FPGAs have been used to build spiking neurons, [8]–[10]. This approach uses digital computation to emulate individual neuron behaviour, but a parallel and distributed network architecture to implement the system behaviour. Having a model which is implementable as well as accurate is the most essential criteria in all these approaches.

In this paper we offer a piece-wise linear modification of the Izhikevich model which is efficiently implementable in both analogue and digital schemes,

978-1-5090-2076-8/10/$31.00 ©2010 IEEE

yet accurate and with similar behaviour to the original model. This model uses the same approach as Izhikevich with a simple modification by which the squaring function is replaced with a "comparison" or "abstract value" both of which are far less expensive, compared to the square function, in either the analogue or digital implementations. The paper is organized as follows: section II provides a brief explanation of the problem background. Section III introduces the modified model while simulation results for single neuron behaviour are presented in section IV. The group dynamic of neurons is compared with the Izhikevich model in section V and the paper is concluded in section VI.

II. Background

By generating sequences of action potentials, nerons process data. Neurons encode computations into sequences of spikes which are biophysically determined by the cell's action-potential-generating mechanism. Izhikevich proposed a simplified model, [4], [11]. This model contains two coupled differential equations, but is able to reproduce complex neural behaviour. This model is based on the following ODEs:

$$\frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I \quad (1)$$

$$\frac{du}{dt} = a(bv - u) \quad (2)$$

with the auxiliary after-spike resting equations:

$$v \geq V_{th}\ then \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases}$$

Here, v represents the membrane potential of the neuron and u represents a membrane recovery variable, which accounts for the activation of K^+ ionic currents and inactivation of Na^+ ionic currents, and it provides negative feedback to v. After the spike reaches its apex (V_{th}), the membrane voltage and the recovery variable are reset according to the equations above. If v skips over V_{th}, then it first is reset to V_{th}, and then to c so that all spikes have equal magnitudes. The part $0.04v^2 + 5v + 140$ is chosen so that v is in the mV scale and time is in ms.

Rewriting the Izhikevich model as in [12], [13] we have:

$$\dot{u} = a(u - u_r)(u - u_t) + I. \quad (3)$$

If $u \geq u_{peak}$ then $u \leftarrow u_{reset}$. It can be simplified by applying a first-order Euler approximation to the quadratic model, resulting in:

$$u_{k+1} = u_k + I_k + a(u_k - u_r)(u_k - u_t). \quad (4)$$

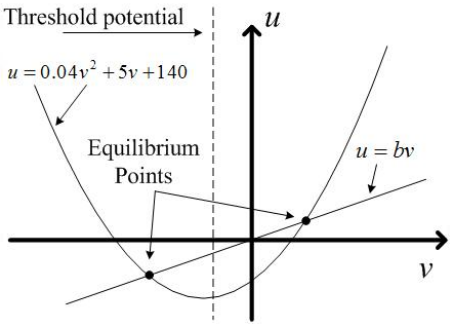

Fig. 1. Equilibrium locus of the Izhikevich model.

If $u_{k+1} \geq u_{peak}$ then $u_{k+1} \leftarrow u_{reset}$; where k is the step number. The last term can be approximated using two taps:

$$u_{k+1} = u_k + I_k + \underbrace{\lfloor \frac{u_k}{P_2} \rfloor}_{Tap1} + \underbrace{\lfloor \frac{u_k}{P_2} \rfloor}_{Tap2} \quad (5)$$

where $P_i = (-1)^{s_i} \cdot 2^{p_i}$ with p_i and s_i being the parameters of the i^{th} tap.

On the other hand, in equilibrium (without input), the Izhikevich model is:

$$u = 0.04v^2 + 5v + 140 \quad (6)$$

$$u = bv$$

These equations, from a geometrical viewpoint, represent a parabolic curve and a line, as depicted in Fig.1. The crossing points of these curves give the equilibrium points of the system (neuron). Different spiking patterns are produced by changing these crossing points and the threshold action potential.

The crossing points in Fig. 1 (equilibrium points) can be calculated as $E_1 = (e_{v1}, e_{u1})$ and $E_2 = (e_{v2}, e_{u2})$, where:

$$e_{v1,2} = 12.5\left((b-5) \pm \sqrt{b^2 - 10b + 2.6}\right) \quad (7)$$

$$e_{u1,2} = be_{v1,2} \quad (8)$$

Although this is known as the most practical, yet accurate, available model; still there are several challenges in realising the model on fixed-point machines or using analogue circuits. The difficulty of implementation arises from the quadratic part of the model, shown by the parabolic curve in Fig. 1.

III. Modified Neuron Model

To improve the computational efficiency of the model we propose a piecewise-linear replacement for the

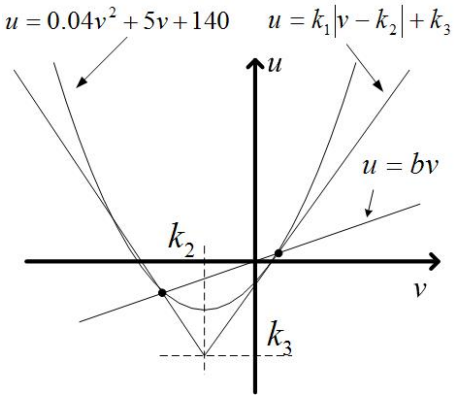

Fig. 2. Piecewise-linear approximation of the second order part of the model.

quadratic part in the equation. We propose the following function:

$$\begin{aligned} \frac{dv}{dt} &= k_1 |v - k_2| + k_3 - u + I \\ \frac{du}{dt} &= a(bv - u) \end{aligned} \tag{9}$$

where k_1, k_2 and k_3 are constant values, which can be pre-calculated by simulation. This new nonlinear function is depicted in Fig. 2.

As can be seen, this approximation provides three degree of freedom for achieving the closest behaviour to the original model. To find values of k_1, k_2 and k_3, three equations are required. The new equilibrium points in this approximation are:

$$\begin{cases} e_{v1} = \frac{k_3 - k_1 k_2}{b - k_1} \\ e_{v2} = \frac{k_3 + k_1 k_2}{b + k_1} \end{cases} \quad and \quad e_{u1,2} = b \cdot e_{v1,2}. \tag{10}$$

According to the values for E_1 and E_2 for the original model, parameters: k_1, k_2 and k_3 have to satisfy:

$$\begin{cases} 12.5\left((b-5) - \sqrt{b^2 - 10b + 2.6}\right) = \frac{k_3 - k_1 k_2}{b - k_1} \\ 12.5\left((b-5) + \sqrt{b^2 - 10b + 2.6}\right) = \frac{k_3 + k_1 k_2}{b + k_1} \end{cases} \tag{11}$$

while the third equation can be based on error minimisation in the model when compared to the original. To minimise this error we have to bear in mind that $v \geq V_{th}$, which means that v and u are limited and this approximation needs to be valid within these limits. Furthermore, to simplify the implementation, we can chose values for k_1 which can be multiplied such as powers of 2.

Since the model consists of simple arithmetic operations (addition/subtraction and shift) a large number of neurons can be implemented on one FPGA.

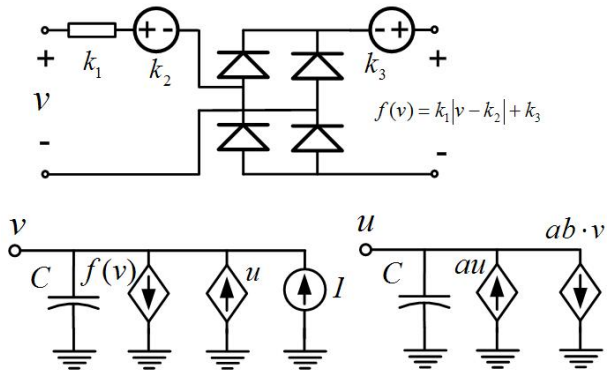

Fig. 3. Equivalent circuit model for neurons.

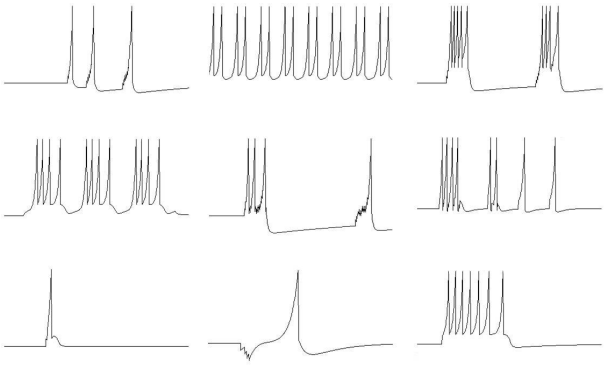

Fig. 4. Example MATLAB simulations showing spiking behaviours.

IV. Single Neuron Behaviour

It is also useful to be able to implement neurons using analogue circuits. Our new model is compatible with a simple analogue implementation. The piece-wise linear function can be implemented using a full-wave rectifier circuit, Fig. 3. Since this circuit has a very simple structure, it could be used in a large-scale implementation on an analogue fabric. Fig. 4 shows a few samples of the output spike patterns produced by this model.

V. Network Dynamics

To investigate the network dynamics of neurons with this model and compare it with the Izhikevich neuron model, the neural network of Fig. 5 is used. This network represents a simple creature's neural system with very simplified visual input, locomotion and decision sub-networks. It simply searches for food and moves toward it when in the vision range. The raster plots of the simulations are presented in Fig. 6. The network activities of the two models are very similar in structure, but differ in the precise details. Since the statistical nature of neural behaviour is generally of interest, these differences may not be significant.

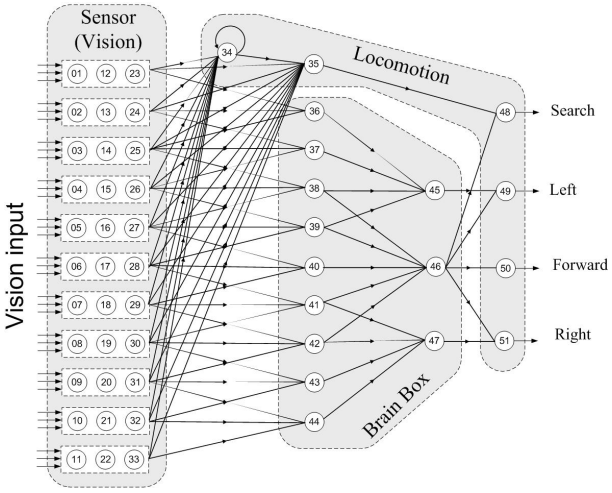

Fig. 5. Neural network for a simple creature with basic movements.

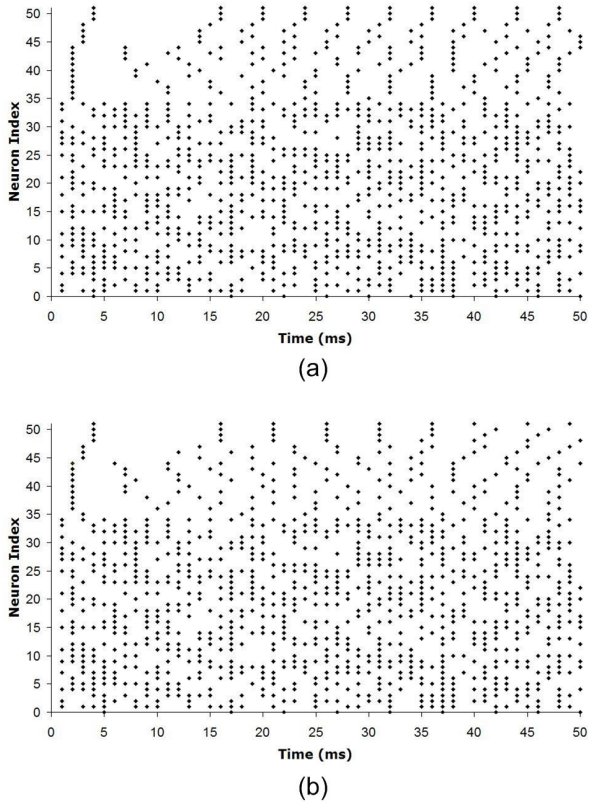

Fig. 6. Neuron activity chart a) Izhikevich b) proposed model.

VI. Conclusion

The Izhikevich neuron model is widely used for modelling spiking neural networks because of its simplicity. Nevertheless, it is based on a quadratic function, which may be expensive to implement in both digital and analogue hardware. We propose a modified version that is based on a piecewise-linear approximation. This model is relatively easy to implement. We have shown that the statistical behaviour of our model is compatible with that of the Izhikevich model.

Acknowledgment

This work has been supported by EPSRC grant EP/D079594/1.

References

[1] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *Journal of Physiology*, vol. 117, no. 4, p. 500544, August 1952.

[2] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*, 1st ed. Cambridge University Pres, August 2002.

[3] N. Brunel and M. C. W. van Rossum, "Lapicque's 1907 paper: from frogs to integrate-and-fire," *Biological Cybernetics*, vol. 97, no. 5-6, pp. 337–339, December 2007.

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

[5] S. Renaud, J. Tomas, Y. Bornat, A. Daouzli, and S. Saighi, "Neuromimetic ICs with analog cores: an alternative for simulating spiking neural networks," in *ISCAS'07: IEEE International Symposium on Circuits and Systems*, May 2007, pp. 3355–3358.

[6] J. H. B. Wijekoon and P. Dudek, "2008 special issue: Compact silicon neuron circuit with spiking and bursting behaviour," *Neural Networks*, vol. 21, no. 2-3, pp. 524–534, 2008.

[7] S. B. Furber, S. Temple, and A. D. Brown, "High-performance computing for systems of spiking neurons," in *Proc. AISB'06 workshop on GC5: Architecture of Brain and Mind*, vol. 2, 2006, pp. 29–36.

[8] M. Mokhtar, D. M. Halliday, and A. M. Tyrrell, "Hippocampus-inspired spiking neural network on FPGA," in *ICES '08: Proceedings of the 8th international conference on Evolvable Systems: From Biology to Hardware*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 362–371.

[9] B. Glackin, T. McGinnity, L. Maguire, Q. Wu, and A. Belatreche, "A novel approach for the implementation of large scale spiking neural networks on FPGA hardware," in *IWANN'05: Computational Intelligence and Bioinspired Systems*. Springer, June 2005, pp. 552–563.

[10] A. Cassidy, S. Denham, P. Kanold, and A. Andreou, "FPGA based silicon spiking neural array," in *BIOCAS'07:Biomedical Circuits and Systems Conference*, November 2007, pp. 75–78.

[11] E. M. Izhikevich, *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting (Computational Neuroscience)*. The MIT Press, November 2006.

[12] H. Shayani, P. J. Bentley, and A. Tyrrell, "Hardware implementation of a bio-plausible neuron model for evolution and growth of spiking neural networks on FPGA," in *16th European Symposium on Artificial Neural Networks, Advances in Computational Intelligence and Learning*, April 2008, pp. 197–202.

[13] ——, "An FPGA-based model suitable for evolution and development of spiking neural networks," in *NASA/ESA Conference on Adaptive Hardware and Systems*, 2008, pp. 236–243.