# Simulating the Izhikevich spiking neuron model using the Brian2 software

## 1 Description

The study of spike-timing dynamics in the brain is of interest to neuroscientists and the artificial network community. Investigation of the relative timing of spikes of multiple neurons [1] and the role they play in temporal coding in the brain is an important issue issue. Similarly, spiking networks can serve as powerful supervised learning and memory representation tools with several potential applications in the machine learning domain [2].

The Izhikevich spiking neuron model [3, 4] is a two-dimensional system of ordinary differential equations. It is summarized in Figure 1[1]. In the equations shown in Figure 1, variable $v$ represents the membrane potential of the neuron and $u$ represents a membrane recovery variable, which accounts for the activation of $K^+$ ionic currents and the inactivation of *Na* ionic currents, and provides negative feedback to $v$.

Parameter $a$ describes the time scale of the recovery variable $u$. Smaller values result in slower recovery. Parameter $b$ describes the sensitivity of the recovery variable $u$ to the subthreshold fluctuations of the membrane potential $v$. Parameter $c$ describes the after-spike reset value of the membrane potential $v$ caused by the fast high-threshold $K^+$ conductances. Parameter $d$ describes after-spike reset of the recovery variable $u$ by slow high-threshold $Na^+$ and $K^+$ conductances. Typical values of the parameters are: $a = 0.02$, $b = 0.2$, $c = -65mV$, and $d = 2$.

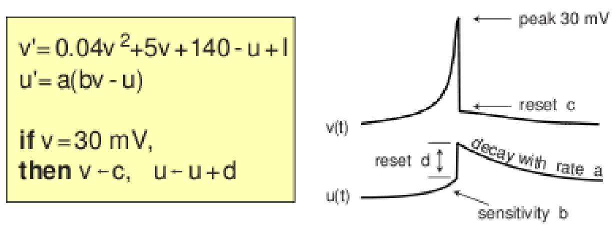


Figure 1: Izhikevich's spiking neuron model.

## 2 Objectives

The goal of the project is to implement the Izhikevich's model using the Brian2 Python library `https://brian2.readthedocs.io/en/stable/`. This library should allow a fast and extremely compact implementation of the model.

The student should: 1) Understand how the Izhikevich's model works. 2) Implement the Izhikevich's model using Brian2. 3) Test that for different parameters of the model it is possible to reproduce the spiking patterns described in [3, 4]. 4) Illustrate the results of the simulations with figures.

[1] Electronic version of the figure and reproduction permissions are freely available at `http://www.izhikevich.com`

As in other projects, a report should describe the characteristics of the design, implementation, and results. A Jupyter notebook should include calls to the implemented function that illustrate the way it works.

## 3 Suggestions

- Complete the two Jupyter Python notebooks available from Brian2 `https://brian2.readthedocs.io/en/stable/resources/tutorials/index.html`, they should be sufficient to implement the model.
- See Izhikevich's paper describing the model.
- Implement the Izhikevich's model as a Python function that receives as parameters the parameters of the model.
- Implementations can use any other Python library.

## References


[1] E. M. Izhikevich and G. M. Edelman. Large-scale model of mammalian thalamocortical systems. *Proceedings of the National Academy of Sciences (PNAS)*, 105(9):3593–3598, 2008.

[2] Hélène Paugam-Moisy, Régis Martinez, and Samy Bengio. Delay learning and polychronization for reservoir computing. *Neurocomputing*, 71(7-9):1143–1158, 2008.

[3] E. M. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14(6):1569–1572, 2003.

[4] E. M. Izhikevich. Which model to use for cortical spiking neurons? *IEEE Transactions on Neural Networks*, 15(5):1063–1070, 2004.