

Spiking Neurons and Networks

Karthi Srinivasan

IIT Madras

Contents

- Principles
 - Dynamics, Excitation, Inhibition, Spiking, Encoding, Decoding
- Models of artificial neurons
 - Hodgkin-Huxley, Izhikevich, Leaky Integrate-and-Fire
- Models of artificial synapses
 - Synaptic integration
 - LPF synapses of different kinds
- Learning in SNNs
 - Hebbian plasticity, STDP, Spatio-temporal backpropagation, Feedback alignment
- How to use SNNs
 - Neural engineering framework
 - Simulator tutorials (BRIAN, Nengo)

Basic Mechanisms - Dynamics

- Behavior of spiking neurons are modeled using differential equations
- Complexity of the model refers to complexity of these equations
- The principal variable is the membrane potential
- Complex models have other variables too
- Eg. Izhikevich model shown below:

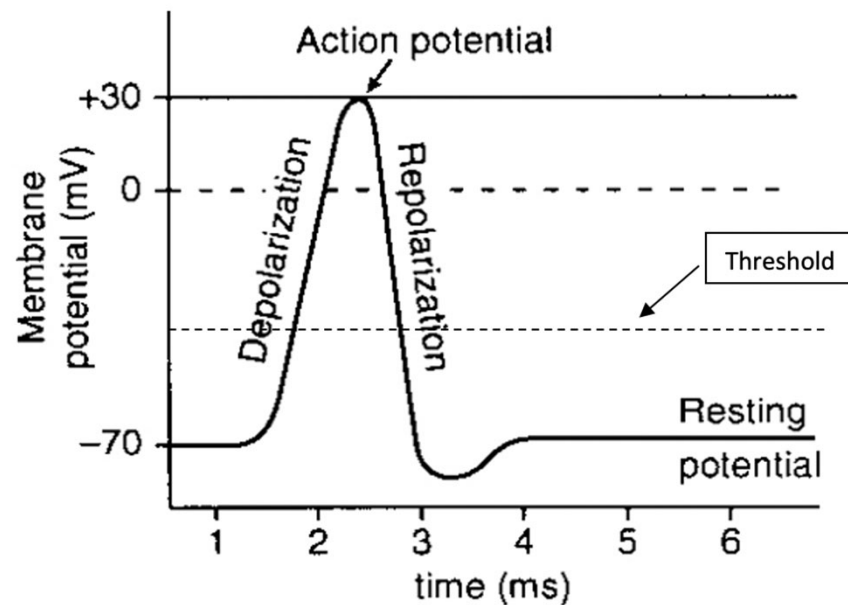
Differential equations of the Izhikevich model

$$\begin{aligned}v' &= 0.04v^2 + 5v + 140 - u + I \\u' &= a(bv - u)\end{aligned}$$

$$\begin{aligned}\text{If } v &\geq 30 \text{ mV} \\ \text{Then } v &= c, u = u + d\end{aligned}$$

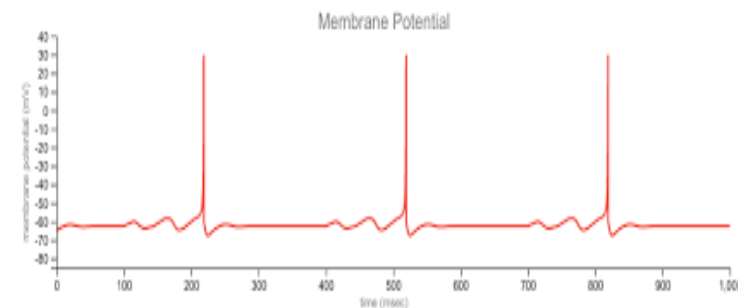
Basic Mechanisms – Excitation and Inhibition

- Excitation – depolarization of the membrane potential
- Inhibition – hyperpolarization of the membrane potential



Basic Mechanisms – Spiking

- A spike is a sudden increase and subsequent resetting of the membrane potential
- Triggered when membrane potential crosses some threshold
- Communicates information between neurons through neurotransmitters sent across the synapses
- Timing code
 - Value of the spike contains no information
 - Time of spike is relevant
 - In brains, spikes on average, carry approx. 3 bits/spike of information



Basic Mechanisms – Encoding

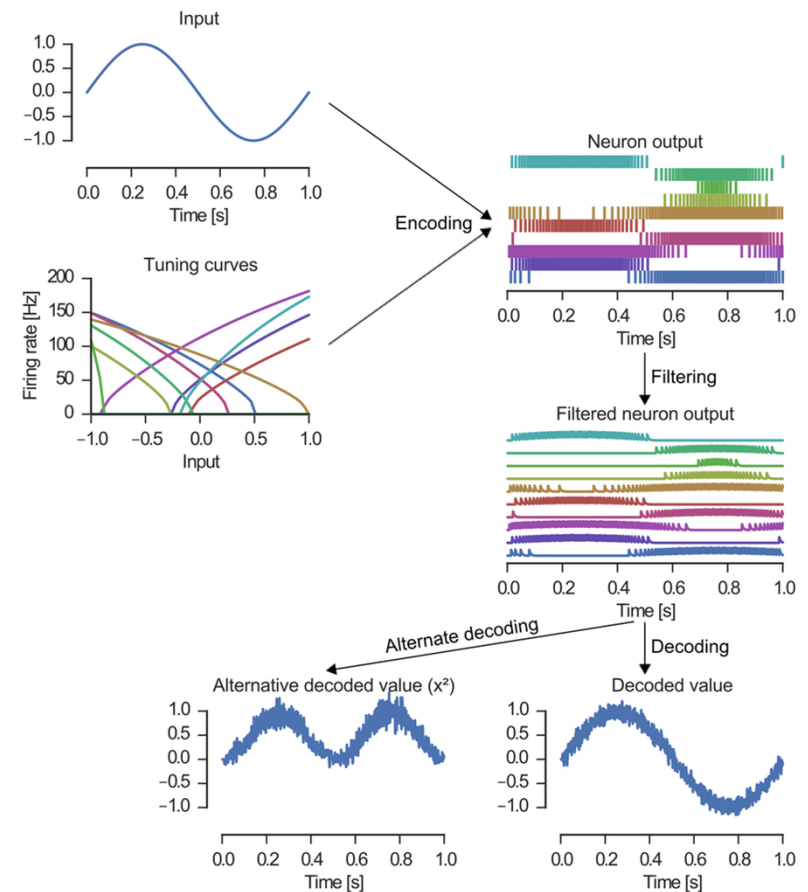
- Neurons communicate with each other using spikes
- But to process external signals, need a method for converting inputs to spikes
- Encoding is done by complex sensory neural networks in biological system
 - Eg. Cones, rods and many others cells cooperate to convert light into spikes in the retina

Basic Mechanisms – Encoding

- Mathematically, we represent this by an inner product between the input vector and the encoder vector, followed by feeding into a neuron:
 - $J_i(\bar{x}) = \alpha \langle \phi_i, \bar{x} \rangle + J_i^b$
 - $a(x) = G[J_i(\bar{x})]$
- $J_i(\bar{x})$ is the current fed into the neuron
- $a(x)$ is activity of the neuron i.e. the firing rate of the neuron
- α is a gain, J_i^b is the bias current
- $G[.]$ is the non-linearity of the neuron, which varies depending on the model being used

Basic Mechanisms – Decoding

- Typically done by low-pass filtering the output spike-train
- For populations, decoded output is linear combination of filtered spike trains
- Function which is being approximated can be changed by changing the decoder weights



Leaky Integrate-and-Fire Model

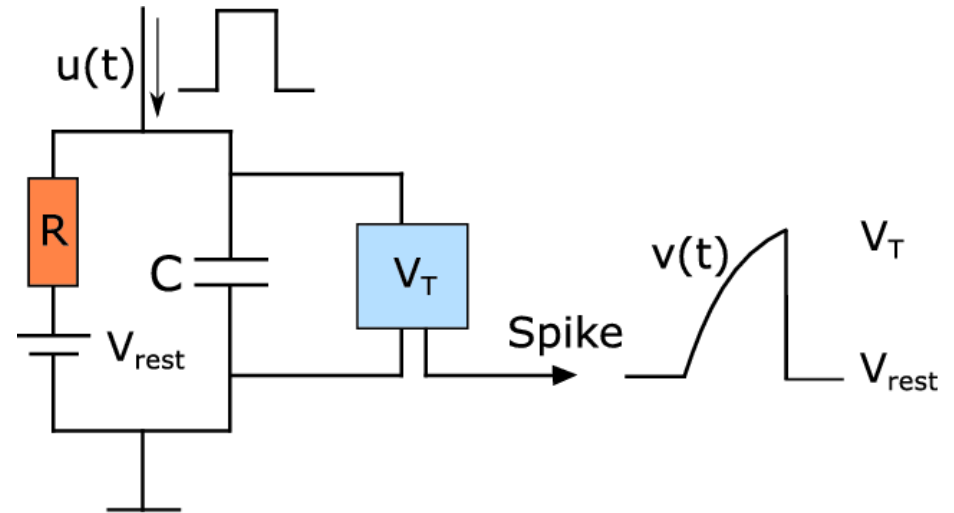
- Simplest Model
- Inexpensive to implement
- Can be used for excitatory/inhibitory neurons
- Does not contain adaptation

- Cannot describe many measured spiking behaviors

- Differential Equation:

$$C_m \frac{dV_m(t)}{dt} = I(t) - \frac{V_m(t)}{R_m}$$

- If $V_m > V_{th}$, then reset V_m to V_r



Hodgkin-Huxley Model

- Most biophysically realistic
 - Models ion channel conductances
- Has 4 state variables
 - V – membrane potential
 - n – potassium channel activation
 - m – sodium channel activation
 - h – sodium channel inactivation
- Difficult to implement

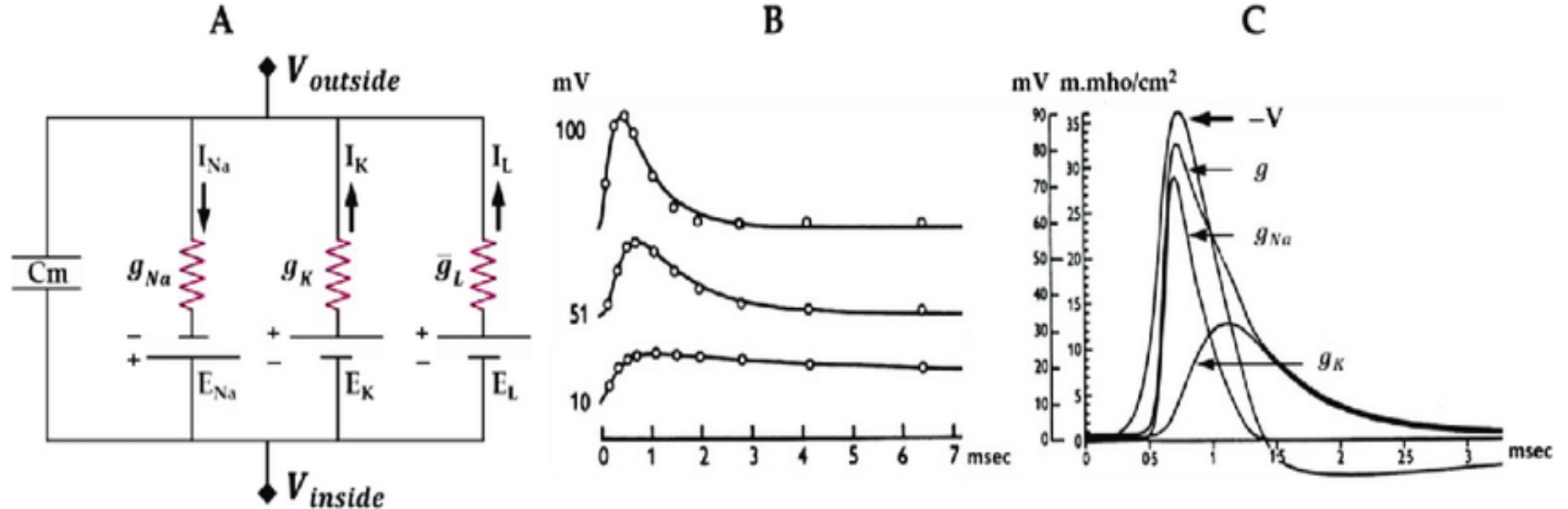
$$I = C_m \frac{dV_m}{dt} + \bar{g}_K n^4 (V_m - V_K) + \bar{g}_{Na} m^3 h (V_m - V_{Na}) + \bar{g}_l (V_m - V_l),$$

$$\frac{dn}{dt} = \alpha_n(V_m)(1 - n) - \beta_n(V_m)n$$

$$\frac{dm}{dt} = \alpha_m(V_m)(1 - m) - \beta_m(V_m)m$$

$$\frac{dh}{dt} = \alpha_h(V_m)(1 - h) - \beta_h(V_m)h$$

HH circuit model, response to incoming spike



Izhikevich Model

- Middle ground between LIF and HH
- 2 state variables
 - V – membrane potential
 - U – membrane recovery
- Can display 20 different types of spiking behaviors
 - Behavior controlled by varying a,b,c,d
- Easy to implement due to soft non-linearity in v equation

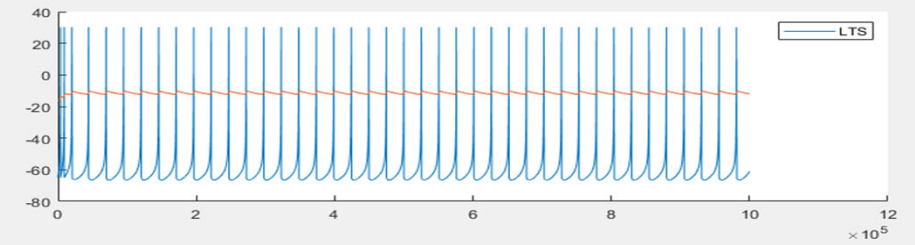
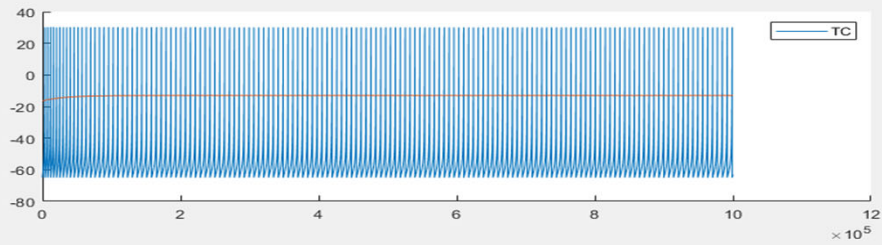
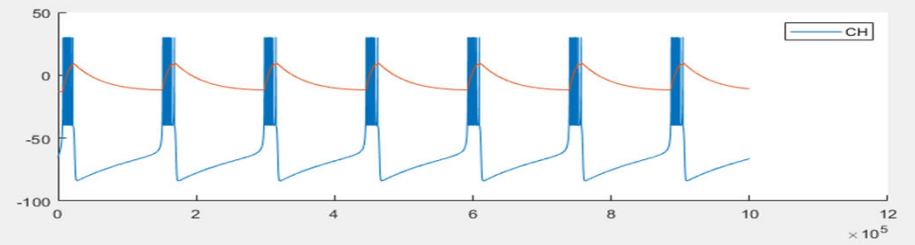
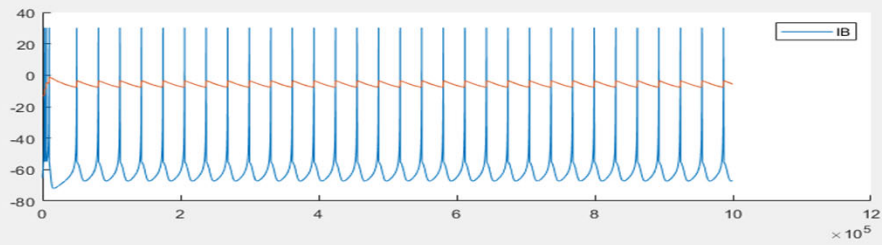
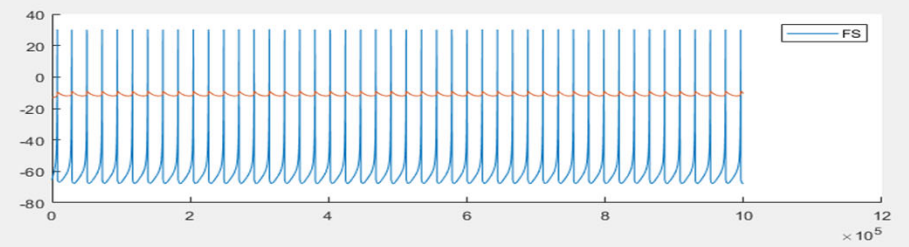
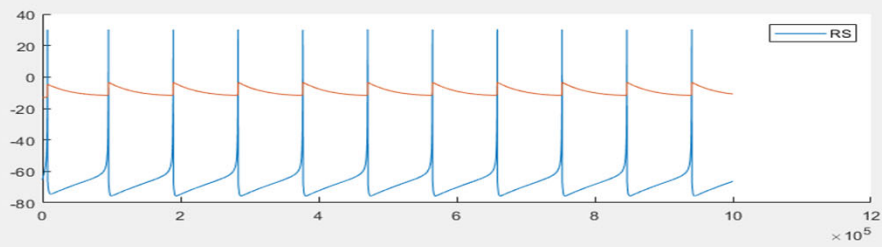
$$\frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I(t)$$

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

$$\text{if } v \geq 30, \quad v = c, \quad u = u + d$$

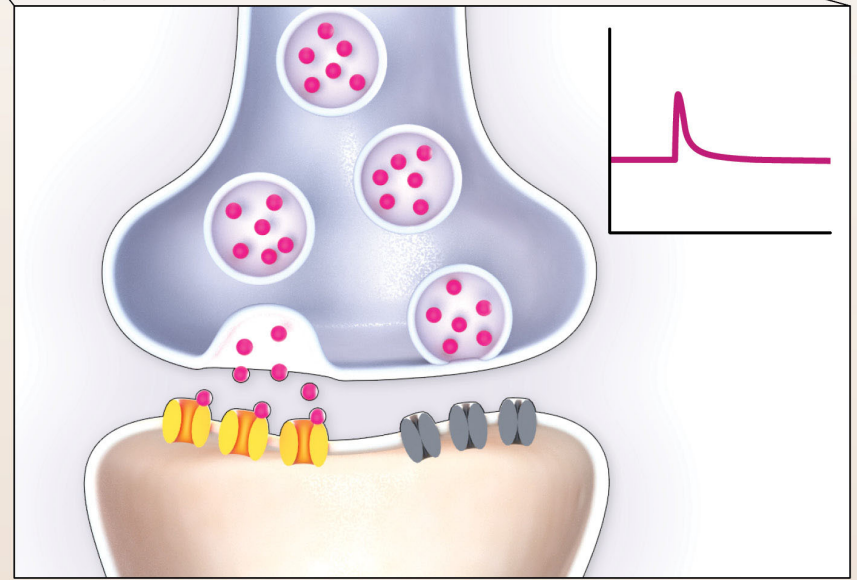
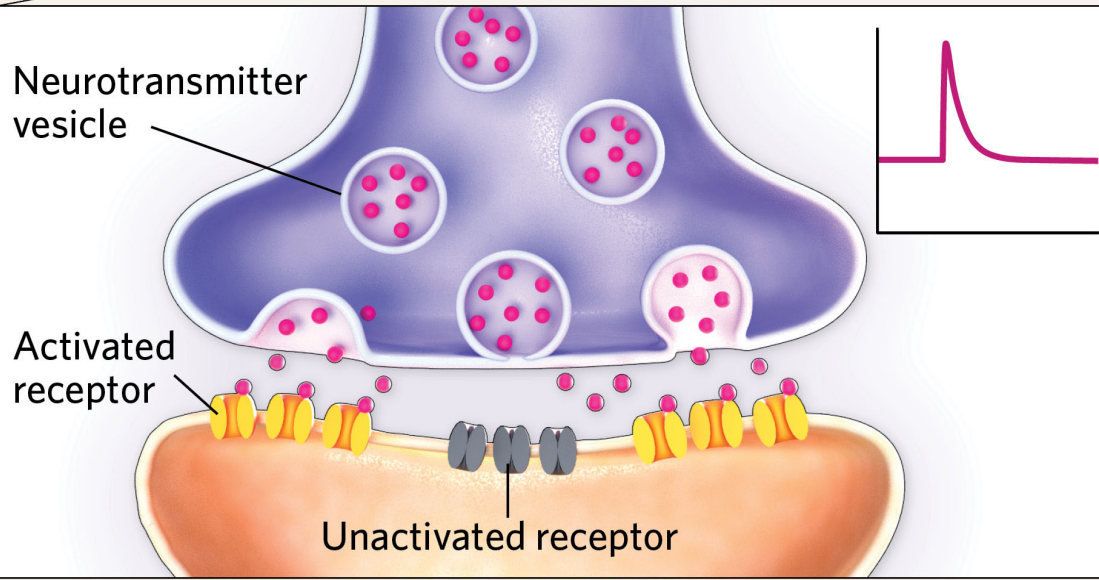
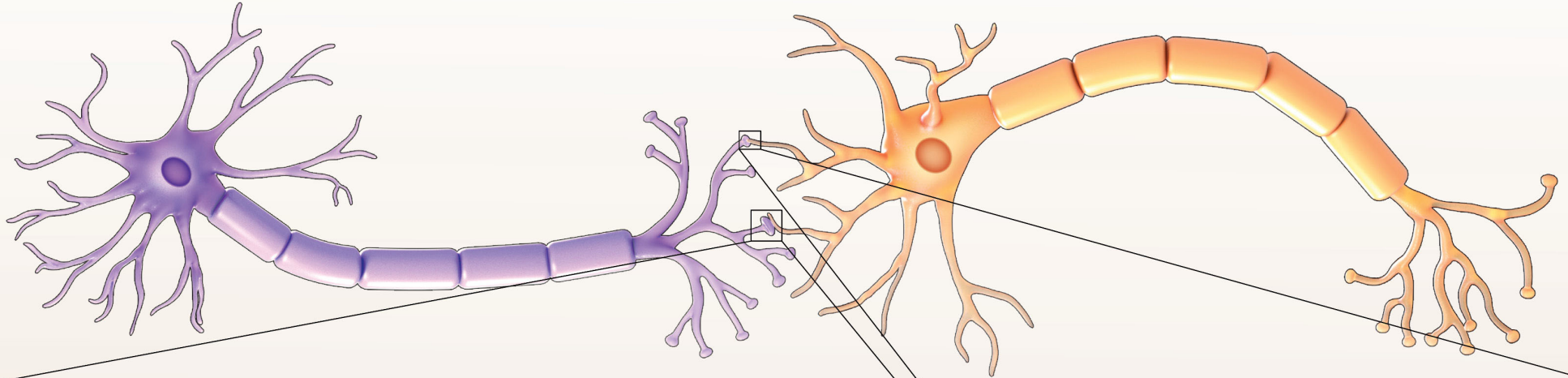
Simulations of the Izhikevich Model

- Blue – v (mV) Orange – u (mV)



Synapses

- Are used to communicate spikes between neurons
- Apply an LPF on spikes from pre-synaptic neuron to generate input current for post-synaptic neuron
- Also perform weighting of inputs
 - This weight is changed during training of the network
- Can undergo long-term potentiation/depression
 - Learning/memory
- Can be excitatory or inhibitory depending on type of neurotransmitter released
- RRAM devices are typically used to model synapses due to their variable resistance (weight)



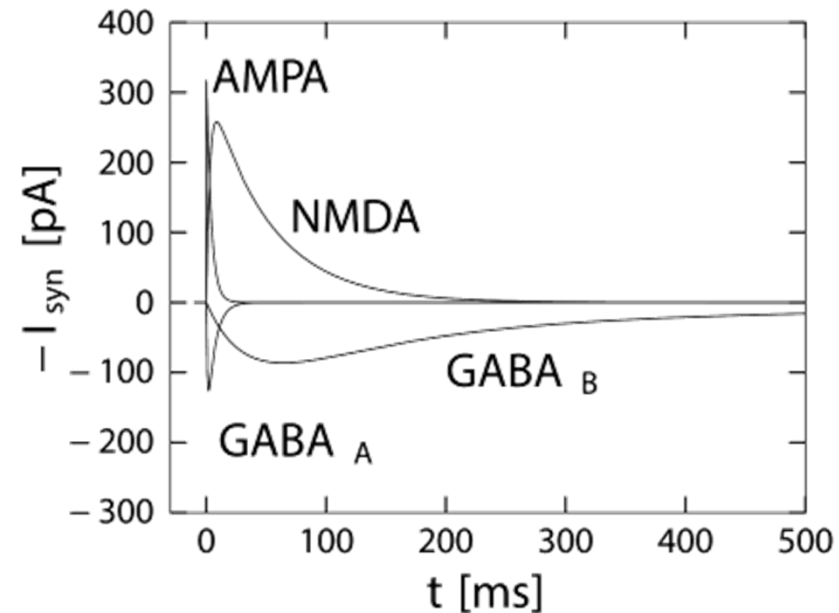
Synapse models

- Simplest synapse models are exponential decays

$$g_{\text{syn}}(t) = \sum_f \bar{g}_{\text{syn}} e^{-(t-t^{(f)})/\tau} \Theta(t - t^{(f)})$$

- More complex models have difference-of-exponentials

$$g_{\text{syn}}(t) = \sum_f \bar{g}_{\text{syn}} \left[1 - e^{-(t-t^{(f)})/\tau_{\text{rise}}} \right] \left[a e^{-(t-t^{(f)})/\tau_{\text{fast}}} + (1 - a) e^{-(t-t^{(f)})/\tau_{\text{slow}}} \right] \Theta(t - t^{(f)})$$



Learning

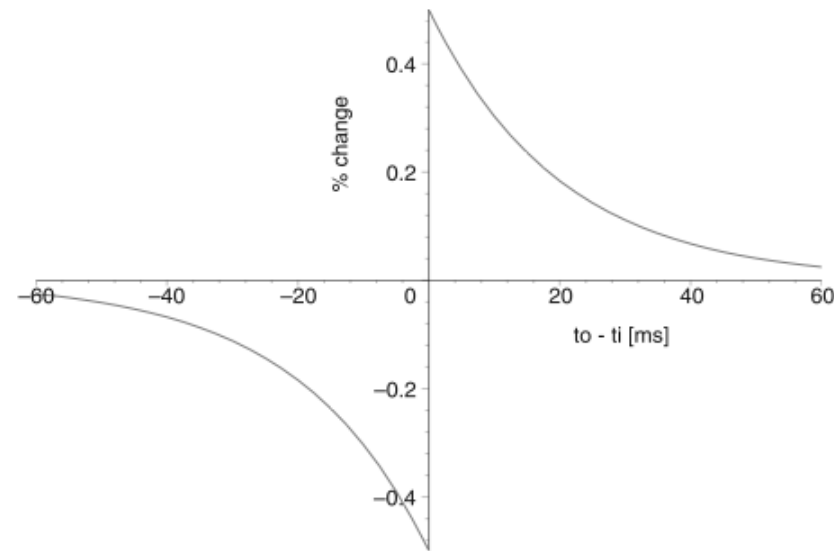
- Process of changing weights in the synapses of a network
- Typically a local process in SNNs, unlike in ANNs
- In biological neurons, weight is controlled by the amount of neurotransmitters released
- In SNNs, scaling gain of post-synaptic LPF is used to achieve weight change
- Adaptation in neurons can be thought of as learning too

Hebbian Learning

- Rule:
 - Let us assume that the persistence or repetition of a reverberatory activity (or "trace") tends to induce lasting cellular changes that add to its stability. ... When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.
- In short: neurons that fire together wire together; neurons that fire out of sync lose their link
- Shortcoming
 - Pre-synaptic neuron has to fire before post-synaptic neuron for causal influence to exist
- This was fixed in STDP

Spike-Timing-Dependent Plasticity (STDP)

- Weight change is inversely proportional to time difference between spikes
- If pre-synaptic spike occurs before post-synaptic spike, weight change is positive (potentiation)
- If post-synaptic spike occurs before pre-synaptic spike, weight change is negative (depression)
- Local learning rule



Neural Engineering Framework

- General framework for building spiking networks
- Can be used for non-spiking networks also
- Three Principles:
 - Non-linear encoding + weighted linear decoding
 - Transformations: functions of variables represented by the population
 - Dynamics: representations as state variables
- Can deal with noisy neurons

BRIAN2 Simulator - Installation

- Ensure you have python 3.5 (or higher) and pip installed
- Commands
 - `pip install brian2`

Nengo Simulator - Installation

- Ensure you have python 3.5 (or higher) and pip installed
- Works well with Windows/Ubuntu
- Commands:
 - `pip install nengo`
 - `pip install nengo_gui`
- Once installation is done, type “nengo” into your terminal
- If everything went well, a new browser tab with an editor and GUI should open automatically