

Project 1: Information Transmission in Spiking Networks

Oliver Eberle Malte Esders Tabea Kossen

February 6, 2015

1 Introduction

How information is processed in the brain is one of the main questions in neuroscience. Classically, there are two main theories how information is encoded via action potentials - either by their firing rate or by their timing of occurrence.

Critics of the second theory claim that our brain is a too noisy *device* and thus, is not able to convey precise spike timings in a network. The possibility of synchronous spiking in a feed-forward network is examined in the following.

2 Methods

2.1 Leaky integrate and fire model

To model electric neural properties a conductance based leaky integrate and fire model was used:

$$\tau_m \frac{dV}{dt} = (-V + E_m) + R_m(-I_{K^+} - I_{syn} + I_{noise}) \quad (1)$$

If the membrane voltage V reaches the threshold V_{thresh} , V is being reset to the resting potential and a spike is counted. To model the decreased probability of another spike shortly after a previous one, an absolute refractory period was introduced, which deactivates incoming inputs from other neurons.

Additionally, a relative refractory period was used, which describes the increase of the K^+ conductance after an action potential and the subsequent exponential decay:

$$\frac{dg_K}{dt} = -\frac{g_K}{\tau_K} \quad (2)$$

$$I_K = g_K(V - E_K) \quad (3)$$

Hereby, the g_K conductance is manually increased after a postsynaptic spike emerged.

$$g_K \rightarrow g_K + \tilde{g}_K \quad (4)$$

Background activity was modelled by an Ornstein-Uhlenbeck process (OUP), which is described by a stochastic differential equation and is used to introduce background membrane potential activity.¹ Characteristic for the OUP is the so-called mean reversion, which expresses a non-constant drifting force towards the long-term mean value.

$$\frac{dI_{noise}}{dt} = \frac{I_\mu - I_{noise}}{\tau_{OUP}} + \sigma\eta \quad (5)$$

¹V. Giorno, S. Spina, On the return process with refractoriness for a non-homogeneous Ornstein-Uhlenbeck neuronal model, Mathematical Biosciences, Vol. 12, No. 2, April 2014 <https://aims sciences.org/journals/pdfs.jsp?paperID=9176&mode=full>

Assumptions and parameters

In the model it is assumed that all neurons have the same properties, including an uniform number of synapses per neuron. Only excitatory neurons are implemented, assuming that the noise acts in an inhibitory manner, helping to convey precise information via spike times.

The noise values I_μ and σ were adapted, so that it produces 2 spikes per second.

Post-synaptic potentials were determined so that they would peak after 1.4 ms with a voltage of 0.11mV (compared to resting potential).

Used values for the mentioned parameters are as follows:

| | | |
|---------------|---------------|-------------------------------------|
| Δt | 0.1 ms | Simulation time step |
| E_K | -77 mV | K^+ equilibrium potential |
| V_{thresh} | -55 mV | Spiking threshold |
| E_m | -70 mV | Membrane resting potential |
| \tilde{g}_K | 5 nS | Additive K-conductance |
| R_m | $10^7 \Omega$ | Membrane resistance |
| τ_{syn} | 0.335 ms | Synaptic time constant |
| τ_{OUP} | 5 ms | Noise time constant |
| σ | 10^{-7} | Standard deviation of noise current |
| I_μ | 0.886 pA | Mean noise current |

2.2 Implementation

The network is implemented using an object-oriented approach in Python 2.7. The model is working via three interactive classes: a model neuron class, a synapse class and a noise class. In order to be able to efficiently control the behavior of the model, most functionality is "hidden" within the classes, so that as little as possible code is needed to create new neuronal networks.

When a neuron object is created, it automatically creates a noise object from which it will draw its "personalised" noise. This is necessary because the noise is not simple gaussian noise, but follows the Ornstein-Uhlenbeck process (see below). Once created, the neuron objects need to be assigned other neurons that they are projecting onto. When this is done, the neurons will create as many synapse objects as needed to simulate synaptic transmission. Every synapse object has its own timing and will be activated by its host neuron after the corresponding presynaptic neuron fires. In this model, all synapses have the same strengths, and all are excitatory.

In the network, 1000 neurons are created, who are separated into 10 groups of equal size. The first group projects onto the second group, all neurons from the first group have a synapse with all neurons from the second, and so on, which results in 90.000 synapses total (not 100.000 since the last group does not project to another group). This results in a fully connected feed forward network (synfire chain).

3 Results

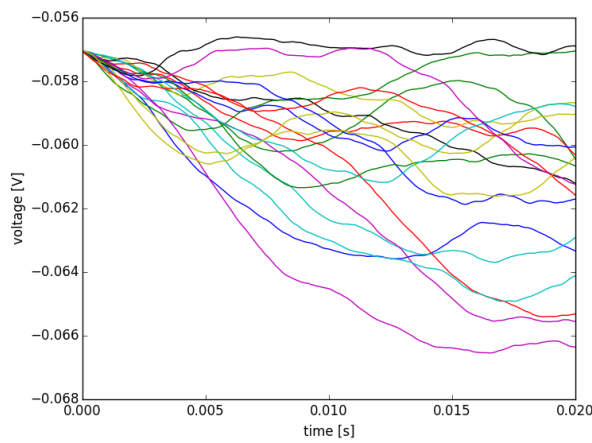


Figure 1

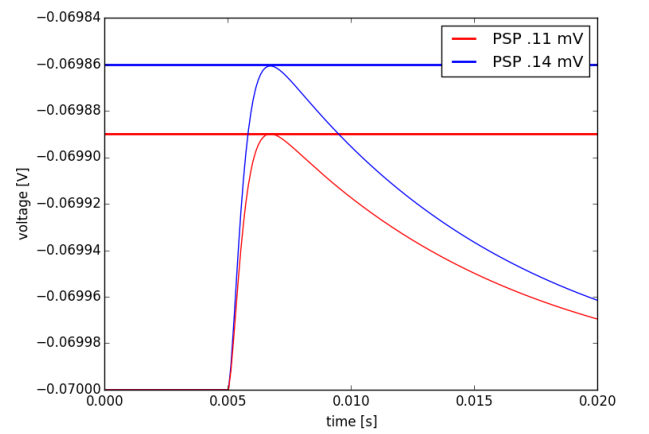


Figure 2

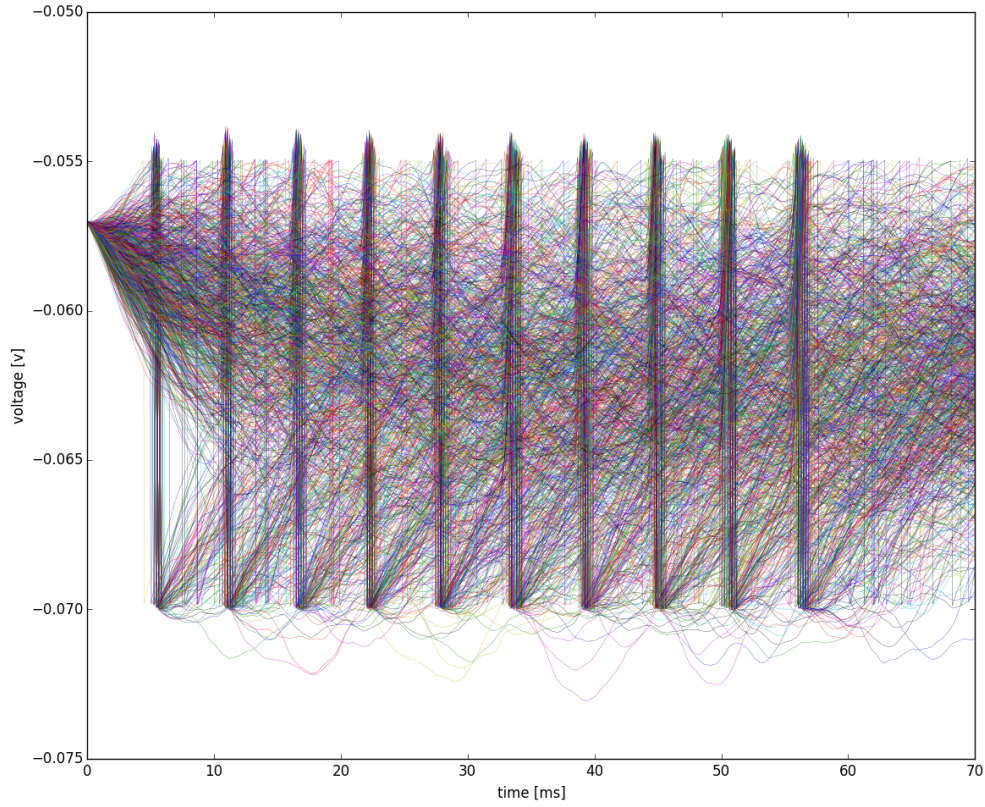


Figure 3: Voltage traces of all 100 neurons in all of the 10 groups. Clearly visible are the spiking ensembles of the groups and a moderate synchronization process throughout the network. After reaching the threshold at $V_{thresh} = -54.0mV$ the voltage is reset to the resting potential $V_{rest} = -70.0mV$.

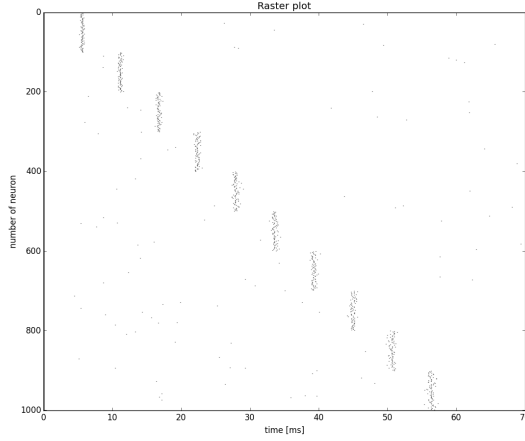


Figure 4: Synchronization

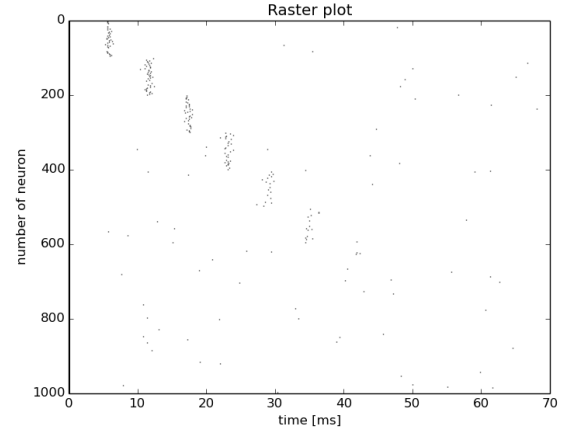


Figure 5: Asynchronization

4 Discussion

5 Conclusion

The main conclusion of this project is that information processing via spike timing is indeed possible in spite of the background noise. In most cases the noise does not interfere with the synchronization and even precise synchronous firing is feasible. This allows for an increased information capacity compared to frequency-coded information processing.

For this network which carries information by precise spike timing a relatively large group of neurons is needed. However, neurons can participate in several volleys simultaneously and the network only needs to be locally feedforward.