

# Problem Set #3: Spikes

Hsieh Yu-Guan

April 24, 2017

## Introduction

Neurons in our brain need to fire signals to communicate with each other. These signals, electrochemical in nature, are referred to as spikes, or action potentials. Here we'll look at several different aspects of this essential element in our nervous system.

## 1 Spike trains

### 1.1 Poisson Model

Before discussing how spikes are produced, we'll first work on the statistical description of spike trains (i.e. a sequence of spikes and silences from a single neuron). As a first approximation, the generation of a random spike train can be simulated by a Poisson process. We assume that individual spikes are generated mutually independently with some probability that can be deduced from the instantaneous firing rate.

Since the computer is a discrete system, a spike train will just be modeled as an array of 0s and 1s. For example, we create a vector of 1000 elements such that each element of the vector has 25% to be 1.



FIGURE 1: A Bernoulli process of 1000 trials with  $p = 0.25$

Next we introduce time units, every 0 or 1 is associated with a time bin of length  $\Delta t$  ms. Here we choose  $\Delta t = 2$  ms and generate a spike train of length 1 sec with the firing rate 25 spikes/sec.

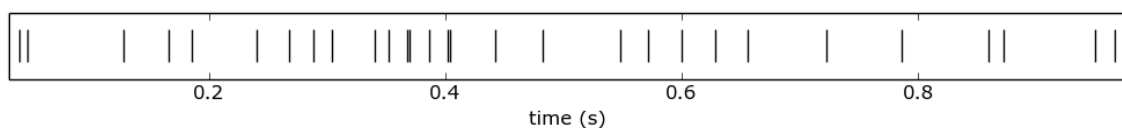


FIGURE 2: A Poisson spike train with an average rate of 25 spikes/sec

In the above figure, there are in effect 29 spikes that are generated. We may be interested in the distribution of the total number of spikes in each simulation that we refer to as total spike count here. Thus we'll generate 50 spike trains with the same parameters.

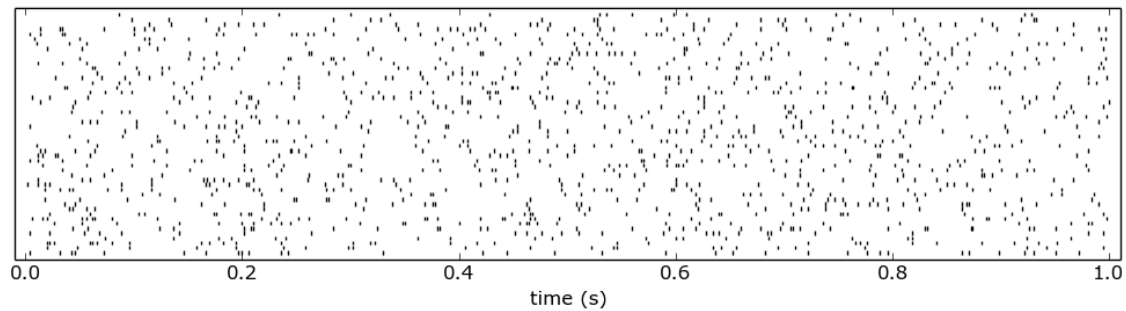


FIGURE 3: 50 Poisson spike trains with firing rate 25 spikes/sec

Then we plot the histogram of total spike counts.

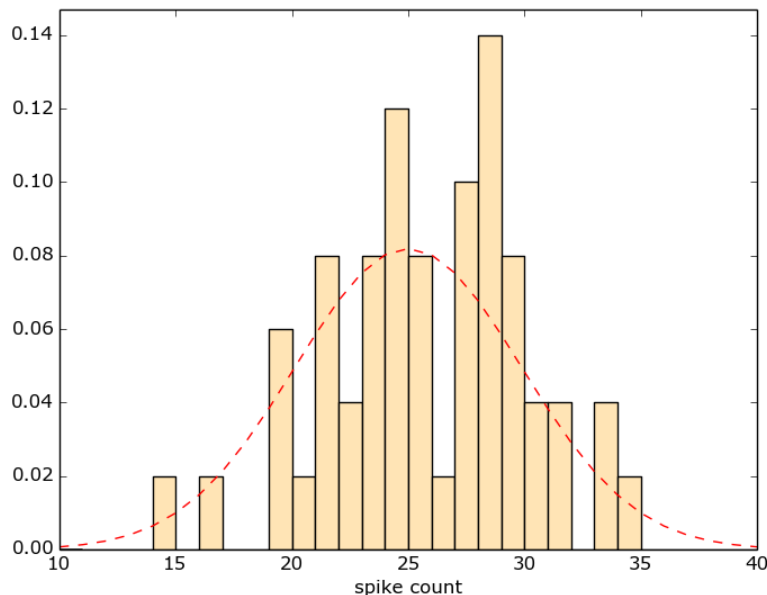


FIGURE 4: Histogram of total spike counts for 50 simulations

According to the central limit theorem, the distribution of spike counts here (which is in fact the binomial distribution  $B(500, 0.05)$ ) can be approximated by the normal distribution  $\mathcal{N}(np, np(1-p))$  with  $n = 500$  and  $p = 0.05$  (the red dashed line in the figure). This can be more or less seen above. However, the theoretical line doesn't fit yet very well the simulation results. It's simply due to the fact that we have too few samples here to describe the distribution, but as we can see later the approximation itself works indeed pretty well.

We also plot the histogram of interspike intervals for the same set of spike trains. This time the histogram follows an exponential distribution, as one might expect (it's a property of the Poisson process).

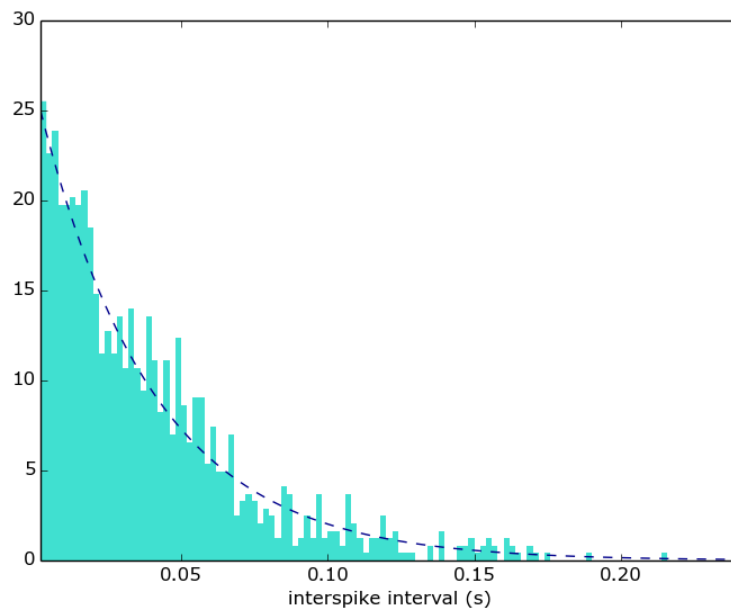


FIGURE 5: Histogram of interspike intervals counts for 50 simulations

We redo the same plots but with now 500 simulated spike trains.

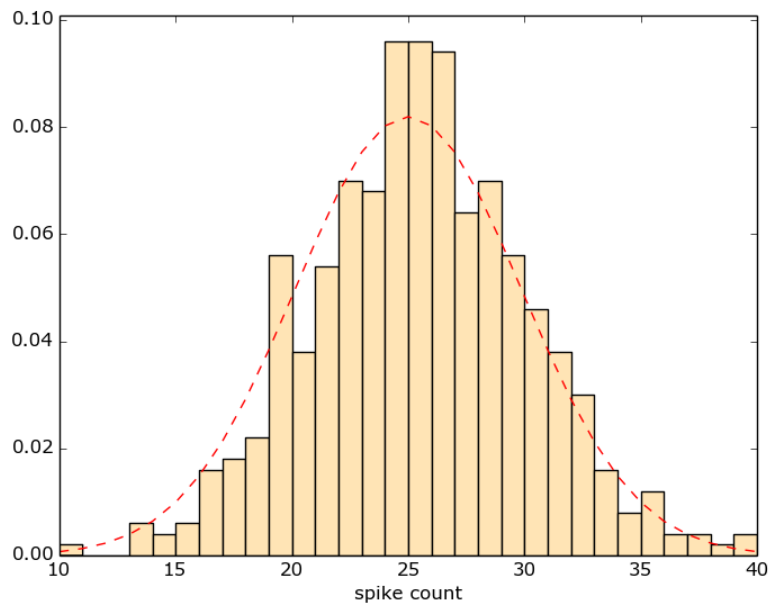


FIGURE 6: Histogram of total spike counts for 500 simulations

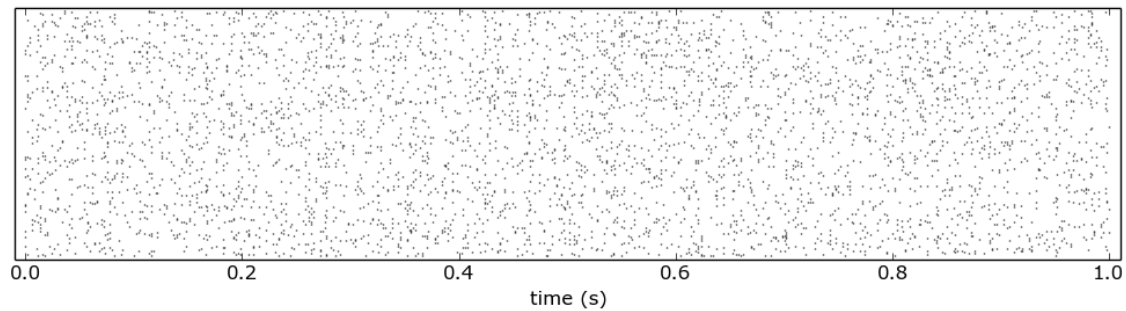


FIGURE 7: 500 Poisson spike trains with firing rate 25 spikes/sec

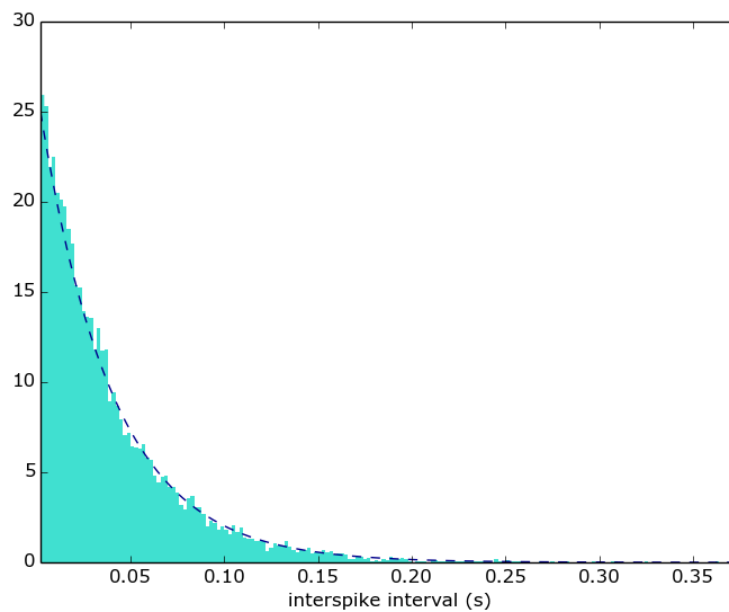
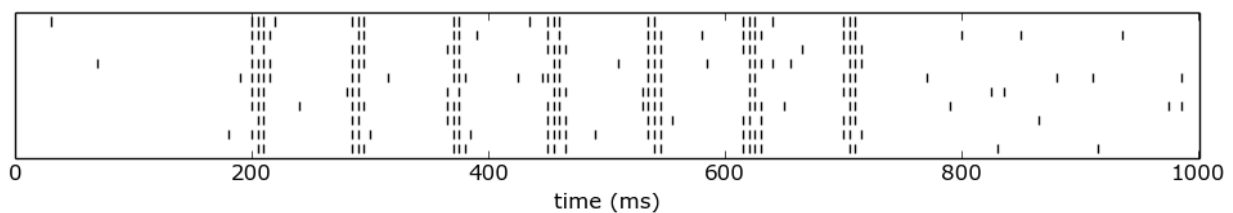


FIGURE 8: Histogram of interspike intervals counts for 500 simulations

We observe that theoretical results fit much better.

## 1.2 Analysis of spike trains

Besides modeling the spike train generations, we'd also like to do some simple analysis of real spike trains. We use thus the experimental data recorded from a single neuron in the primary somatosensory cortex of a monkey that was experiencing a vibratory stimulus. First we plot the spike trains for the stimulus  $f = 8.4$  Hz into the graph below.

FIGURE 9: Real spike trains recorded from a neuron in the primary somatosensory cortex of a monkey that was experiencing a vibratory stimulus with  $f = 8.4$  z

Here we don't observe anymore the poisson process. Instead, we tend to see more spikes at some specific moments that are separated by some fixed length time intervals. We now plot all the recorded spike trains into the same graph. Alternate backgroud colors are meant to indicate different stimuli.

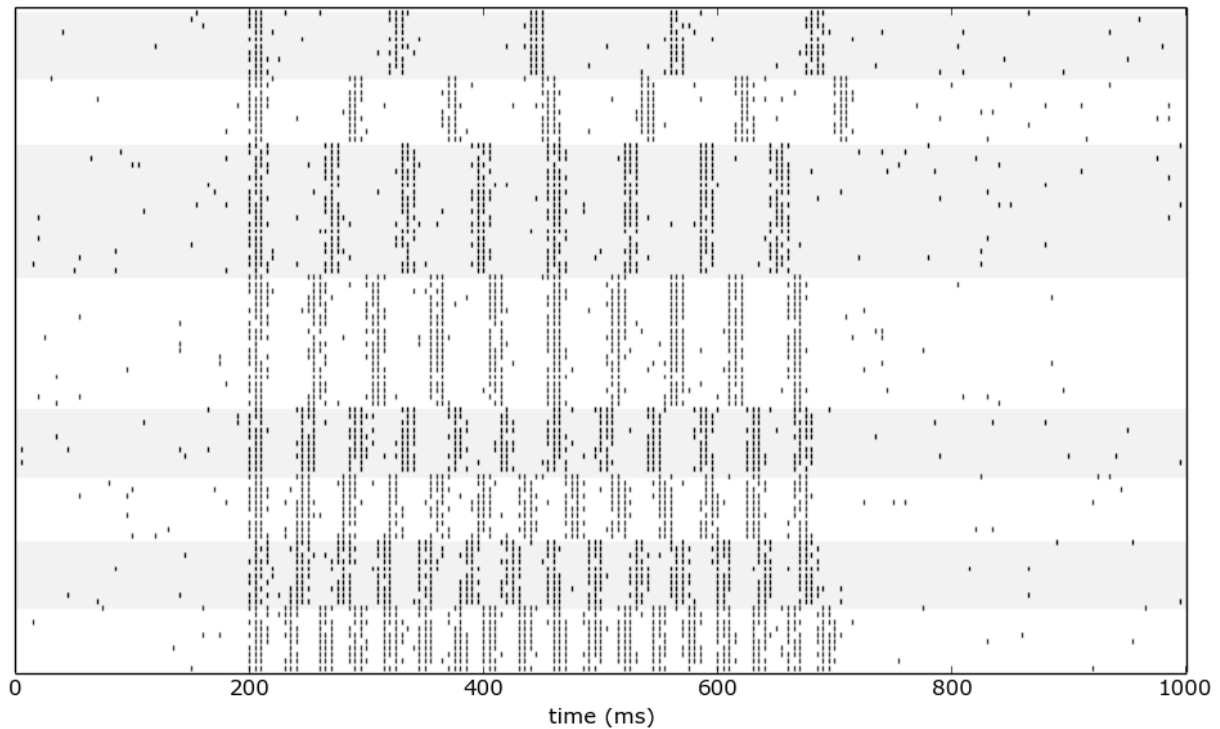


FIGURE 10: Real spike trains recorded from a nueron in the primary somatosensory cortex of a monkey that was experiencing some vibratory stimulus with different frequencies

The corresponding frequency for each dataset is not shown in the graph. In fact from top to bottom the frequency increases and we see that the seperating time intervals also become shorter and the spike count augments. This shall be even clearer if we give the exact numbers. The simulus is only present between  $t = 200$  ms and  $t = 700$  ms. We compute the average spike count and the standard deviation of spike counts for each stimulus.

TABLE 1: Mean values and standard deviations of spike counts for different stimuli

<i>Frequency (Hz)</i>	8.4	12	15.7	19.6	23.6	25.9	27.7	35
<i>Average spike count <math>m</math></i>	16.5	19.2	23.6	29.9	35.6	39.5	41.8	52.3
<i>Standard deviation <math>\sigma</math></i>	1.80	1.47	1.96	1.58	2.50	5.94	1.89	3.26

Sure when the frequency gets higher, the average spike count increase as well. We plot also the tuing curve of the neuron.

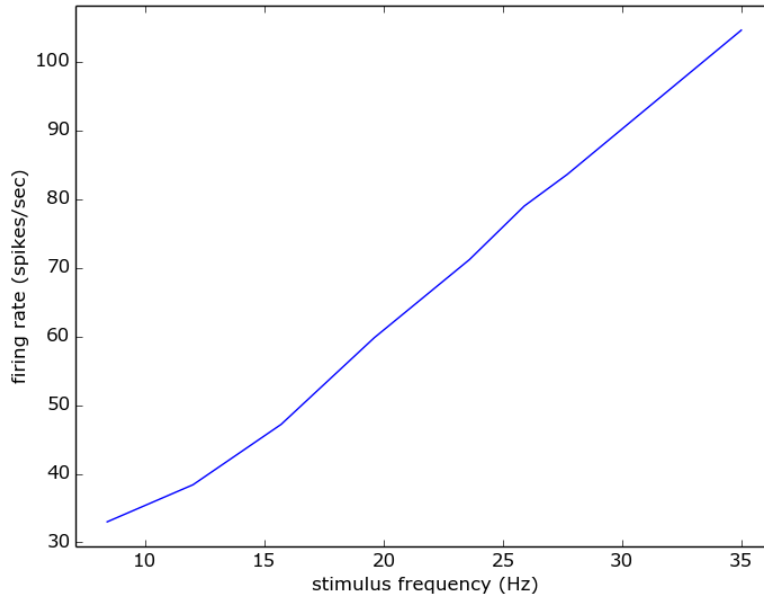


FIGURE 11: Tuning curve of the neuron

The relation between the average firing rate and the stimulus frequency is almost linear. We may also want to show that the mean value and the standard deviation of spike counts are positively correlated and even try to find an explicit relation between these two quantities (for example in the case when the spike count is sampled from some poisson distributions we have  $\mu = \sigma^2$ ). However, with the values given in [Table 1](#), we can not easily draw a conclusion.

## 2 Leaky Integrate-And-Fire Model

### 2.1 Model description

Let's look more closely at how spikes are generated in a cell from a biophysical point of view. In a biologically detailed model, we might need to take into account the topology of the neuronal tree and at the same time establish a model for each basic component of the neuron. Finally, the interactions between different components should also be simulated. It can be very complicated and thus, we'll consider a much more simplified model here that however still fits quite well experimental data.

In the leaky integrate-and-fire (LIF) model, the whole neuron is collapsed to a single point. The cell membrane acts like a RC circuit and the relationship between the output voltage  $V(t)$  and the input current  $I(t)$  is therefore given by

$$C \frac{dV(t)}{dt} = g_L(E_L - V(t)) + I(t) \quad (1)$$

where  $C$  is the membrane capacitance,  $g_L = 1/R$  is the conductance of the membrane that contributes to the leakage term and  $E_L$  is the reversal potential. This equation can be solved numerically using the Euler method, that is

$$V(t + \Delta t) = V(t) + \frac{dV(t)}{dt} \Delta t \quad (2)$$

for a small  $\Delta t$ . Otherwise, we can also solve the equation analytically and get

$$V(t) = (V(0) - E_L) \exp\left(-\frac{t}{\tau_m}\right) + E_L + \frac{1}{C} \int_0^t \exp\left(-\frac{s}{\tau_m}\right) I(t-s) ds \quad (3)$$

where  $\tau_m = RC$  is the membrane time constant. The analytic solution is useful when the integral has an explicit expression since the result is generally of better precision and the computation gets also faster.

The spiking events are then characterized by a firing time  $t$  that is defined by a threshold criterion. In other words, every time when the membrane voltage  $V$  reaches a certain threshold  $V_{th}$ , the neuron emits an action potential and  $V$  is reset to  $E_L$ .

## 2.2 constant stimulation

Let us start by studying a simple example here. Suppose that the integrate-and-fire neuron is stimulated by a constant input current  $I(t) = I_0$ . We compute first the solution of (1) using the Euler method. For the parameters, we fix  $C = 1\text{ nF}$ ,  $g_L = 0.1\mu\text{S}$ ,  $E_L = -70\text{ mV}$ ,  $V(0) = E_L$ ,  $I_0 = 1\text{ nA}$  and  $\Delta t = 1\text{ ms}$ . We run the simulation until  $t = 100\text{ ms}$ .

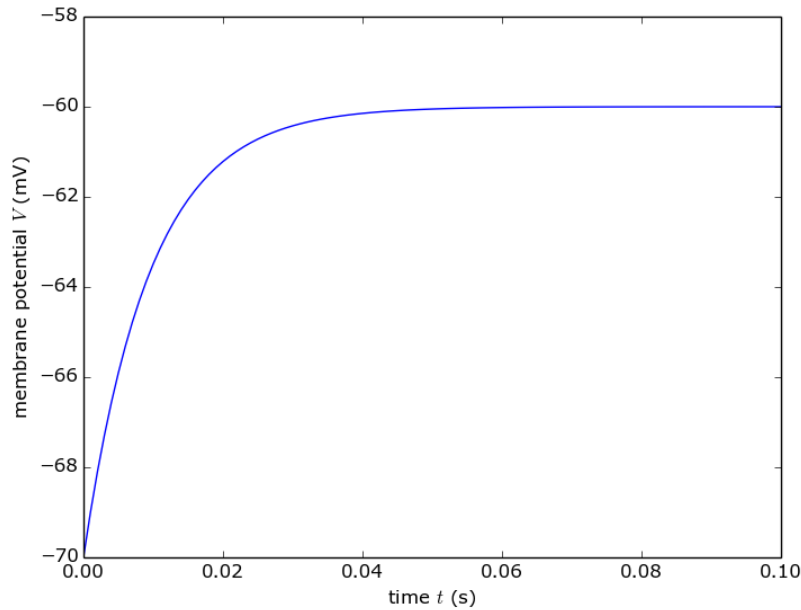


FIGURE 12: LIF model without spiking mechanism with constant input current  $I = 1\text{ nA}$

We observe an exponential rise to a limit value  $V_\infty = E_L + RI_0$ . In Figure 13, we compare the results of several different input currents. When  $I$  is higher,  $V$  climbs faster at the beginning and the final value is as well higher, though, the characteristic time doesn't change.

We may also be interested in the effect of  $\Delta t$ . From a mathematical point of view, the smaller is  $\Delta t$ , the better. In fact, we can see in Figure 14 that when the stepwidth increases, the numerical error with the real value of  $V$  gets equally larger (though sure, we can never represent the “real” value in a figure, but say, intuitively). However, more computations are also required when  $\Delta t$  gets smaller, a trade-off needs to be found. It's also shown in the figure that the two curves  $\Delta t = 1\text{ ms}$  and  $\Delta t = 0.1\text{ ms}$  are close, which means that  $\Delta t = 1\text{ ms}$  is already quite a sensible choice that allows us to have a good precision of  $V$ .

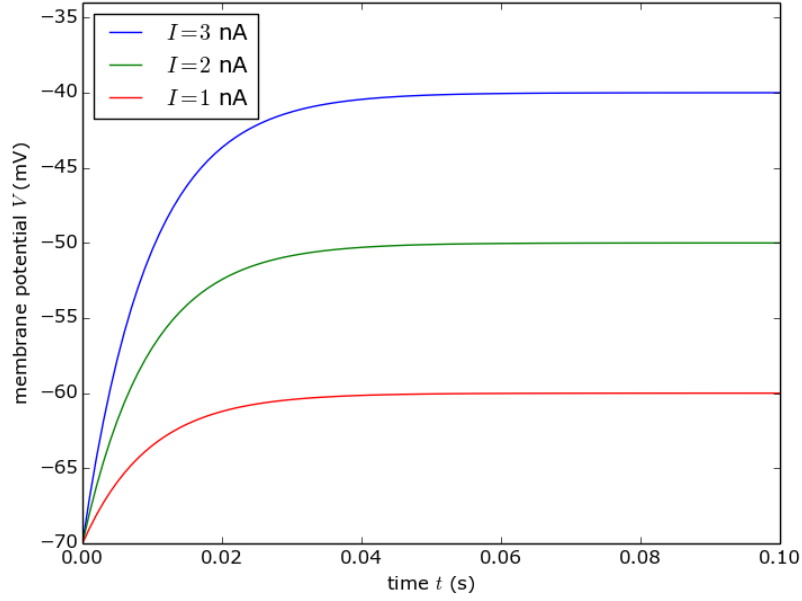
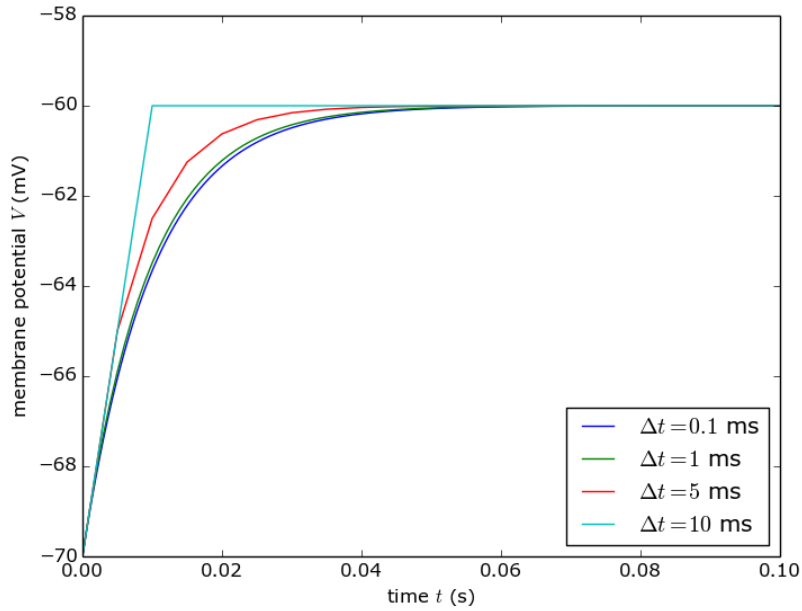


FIGURE 13: Different constant input currents

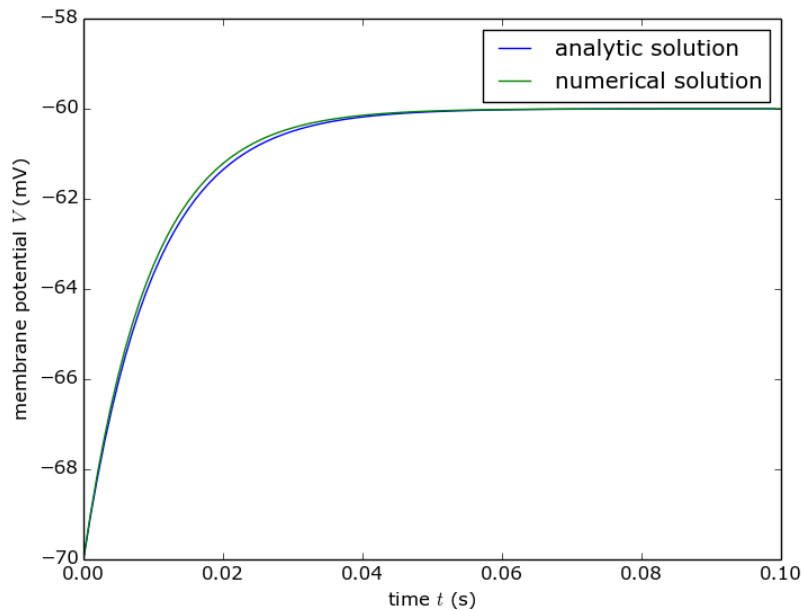
FIGURE 14: Effect of  $\Delta t$  ( $I = 1$  nA)

By using the equation (3) we know that in this case the analytic solution is given by

$$V(t) = V_{\infty} + (V(0) - V_{\infty}) \exp\left(-\frac{t}{\tau_m}\right). \quad (4)$$

The explicit expression allows us to verify the validity of the numerical approach. We compare thereby the plots of the two solutions under the condition  $I = 1$  nA. For the numerical part, we use  $\Delta t = 1$  ms. As shown in Figure 15, the two curves are similar and with what is mentioned above, if we decrease  $\Delta t$ , we can approach the analytic solution. (This may be confusing, but the plot of the analytic solution in the graph isn't either the true value of  $V$  since time is not continue in a computer. In effect, the plot is done by using a time bin of 1 ms.)



FIGURE 15: Comparison of analytic and numerical solutions ( $I = 1$  nA)

We now equip the cell with the simple spiking mechanism as described above. We choose  $V_{th} = -63$  mV and vary the value of input current. The results are plotted separately in order to get a better view. Here we use  $\Delta t = 0.1$  ms. We can imagine that with a larger stepwidth, we're not able to have a good precision of spike moments.

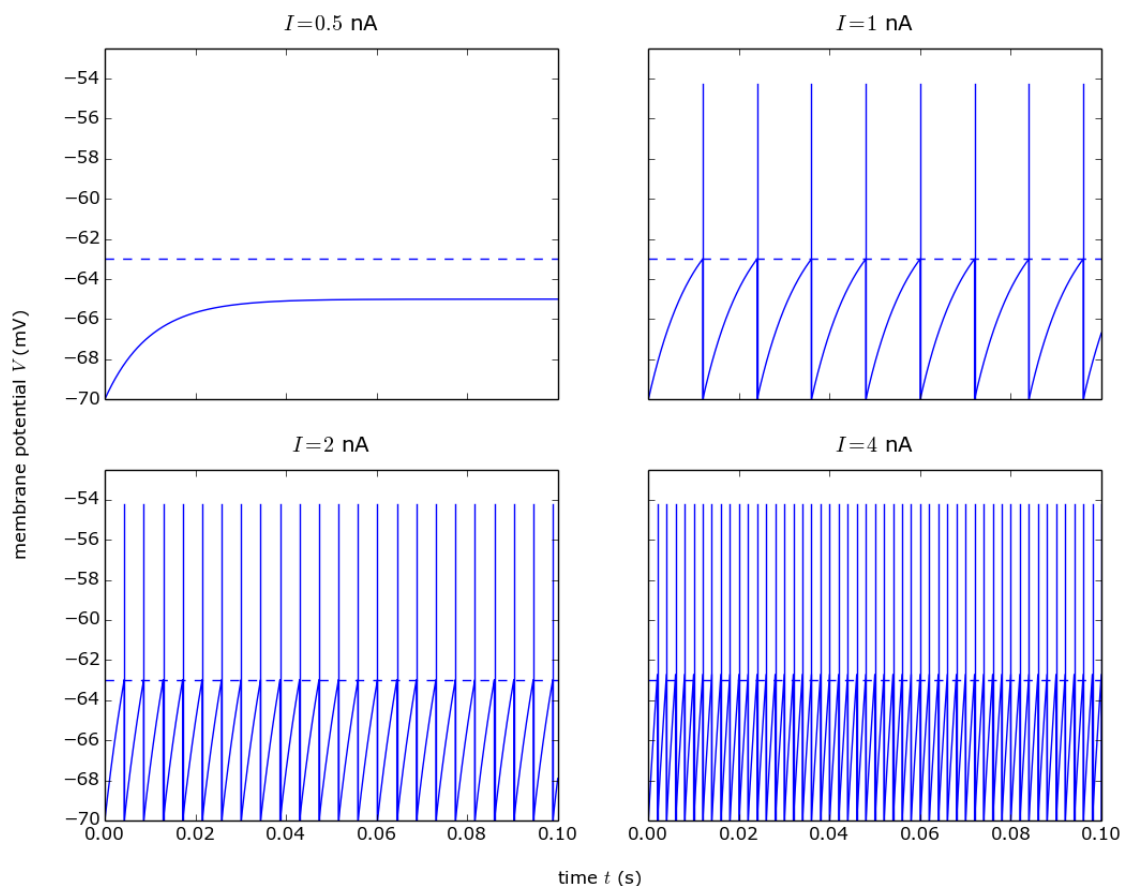


FIGURE 16: LIF model with constant input current

We get more spikes within the first  $t = 100$  ms when  $I$  augments, which means that the firing rate increases. Nevertheless, smaller  $\Delta t$  is needed to capture a higher firing rate. This can be more or less seen from the case  $I = 4$  nA where we have the feeling that spikes are not immediately generated after the threshold is reached. On the other hand, when  $I = 0.5$  nA, no spikes are emitted. As a matter of fact, we need to have  $V_\infty > V_{th}$  for the neuron to fire. It means that the condition for  $I$  is  $I > g_L(V_{th} - E_L)$ . The exact firing rate can also be computed under this hypothesis, we first compute the time  $T$  that it takes for  $V$  to reach the threshold  $V_{th}$  from  $E_L$ , and then we have  $f_{firing} = T^{-1}$ . The explicit formula is

$$f_{firing} = (\tau_m \log(\frac{E_L - V_\infty}{V_{th} - V_\infty}))^{-1} \quad (5)$$

In the figure below we plot the tuning curve of the neuron, i.e. the number of spikes within 100 ms as a function of the input current  $I$ . The two curves are acquired respectively by using (5) and by simulating directly the integrate-and-fire mechanism.

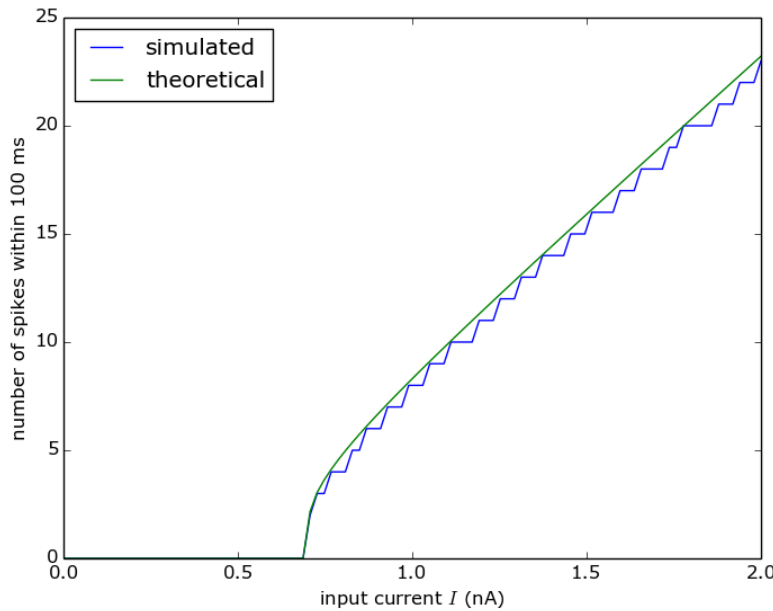
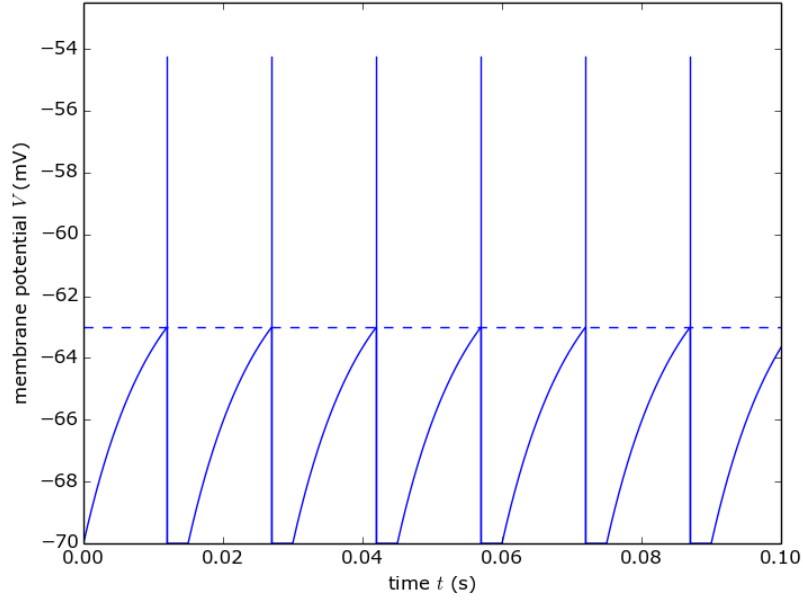


FIGURE 17: Tuning curve of the LIF neuron when  $I$  is constant

With the given parameter,  $g_L(V_{th} - E_L) = 0.7$  nA. We observe in the graph the neuron starts firing indeed at about  $I = 0.7$  nA. The curve obtained by the simulation approximates quite well the theoretical curve. The jadded shape comes from the fact that in the simulation,  $V$  is reset for every  $k$  time bins for some integer  $k$ , so the number of spikes within 100 ms is then  $\lfloor 1000/k \rfloor$  (with  $\Delta t = 0.1$  ms).

## 2.3 Refractory period and noise term

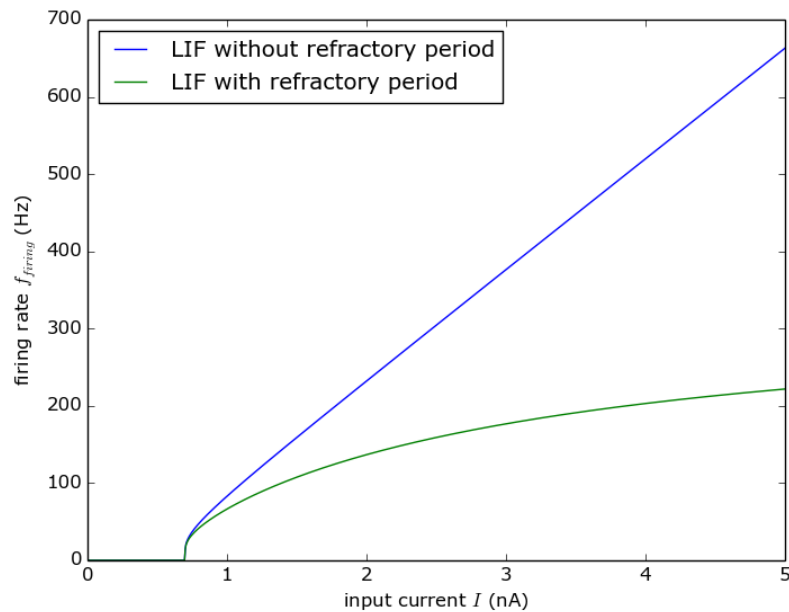
For the moment being the stimulus is always a constant current, but we'd like to add some new elements in the model to make it more realistic. The first step is to introduce a refractory period. A simple idea is to fix  $V(t) = E_L$  for a small amount of time  $\Delta$  after each spike, and the differential equation comes into play only after this period. Under this new hypothesis, we plot again the evolution of the membrane voltage  $V$ .

FIGURE 18: LIF neuron with refractory period ( $I$  constant)

The new firing rate becomes

$$f_{firing} = (\Delta + \tau_m \log(\frac{E_L - V_\infty}{V_{th} - V_\infty}))^{-1}. \quad (6)$$

It's more realistic since it admits a supremum. We compare the tuning curve of the original model and the model with the refractory period. Due to the reason that is explained before (some innate constraints of the simulated result), we plot the curves using the analytic formulas (5) and (6), which allows us to have a greater variation of  $I$  and the difference between two models becomes also clearer.

FIGURE 19: Tuning curve of the LIF model with and without refractory period ( $I$  constant)