

BioE 332 Assignment 3: Irregular spiking and high variability in balanced cortical networks with clustered connections

Kwabena Boahen, TA: Tatiana Engel

Assignment due: April 26, 2013

1 Goals

Cortical neurons *in vivo* exhibit highly irregular spiking patterns with statistics close to a Poisson process. The high variability of cortical responses seems inconsistent with the large number of synaptic connections received by each neuron. Indeed, consider a neuron receiving N independent synaptic inputs each with the strength $g \sim 1/N$. The mean input to this neuron gN is of the order 1, and the variance of the input is g^2N , which scales as $1/N$. This means that the standard deviation of the input decrease as $1/\sqrt{N}$, and should be very small for cortical neurons that receive $\sim 10^4$ synaptic inputs, suggesting that regular spiking should prevail in the cortex.

It turns out that the above argument does not hold in cortical circuits, where synapses appear to be very strong with $g \sim 1/\sqrt{N}$. The input variance g^2N is then of the order 1. Interestingly, the mean input can still be of the order 1 in such a network, if it operates in the balanced regime, where strong excitation and strong inhibition ($gN \sim \sqrt{N}$) nearly cancel each other. In the balanced state, spiking is driven by large fluctuations in the synaptic input and is nearly Poisson.

In this assignment, we will explore the balanced state in a network model with clustered connectivity. In such a network, firing is irregular and the activity pattern randomly wanders between multiple co-existing attractors. As a result, neurons exhibit the supra-Poisson spiking variability characteristic for spontaneous cortical activity. This model also captures reduction in spiking variability during stimulus-driven activity — a wide-spread cortical phenomenon. We will learn about:

Scientifically:

1. Recurrent networks with balanced excitation and inhibition.
2. Noise-driven transitions between co-existing attractors.
3. Statistical measures of spiking variability: Fano factor, coefficient of variation.

Technically:

1. Implementation of networks with heterogeneous connectivity in Brian.

Readings:

1. Self-sustaining balanced state in a recurrent neural network model [3].
2. Clustered connectivity in a balanced network generates slow dynamics and supra-Poisson variability [2].

2 Balance of excitation and inhibition in a recurrent network model

We begin with simulations of a homogeneous balanced network model that exhibits self-sustained irregular spiking activity. There are two requirements for the existence of a balanced state: strong ($\sim 1/\sqrt{N}$) recurrent synapses and sparse connectivity. The latter ensures that different neurons receive their input from largely non-overlapping subsets of pre-synaptic neurons in the network (i.e. inputs to different neurons are uncorrelated).

As a first step, we will implement the spiking network model from Ref. [2] with uniform connectivity ($p^{EE} = 0.2$), which mean that excitatory connections are randomly placed across the whole network with the probability $p^{EE} = 0.2$ (no clustering). In the next section we will add clustering of the excitatory connections to the model.

The implementation of the balanced network model from Ref. [2] is simple compared to more biophysically realistic networks from the first two assignments. In this model, both excitatory and inhibitory populations are described by the same equations with linear synapses. Brian is more efficient, if you define neuronal groups to be as large as possible, and then specify differential properties of sub-populations (e.g., of E and I cells) using subgroups. Therefore, we will generate a large `NeuronGroup` with $N^E + N^I$ neurons, and then split this group into excitatory and inhibitory subgroups. To implement the heterogeneity in the bias parameter μ and the difference in the membrane time constants τ of E and I cells, we initialize two parameter arrays:

```
import numpy as np
from numpy.random import *

# uniform between min and max
uniform = lambda min,max,N : rand(N)*(max-min)+min

mu = np.concatenate((uniform(mu_min_e, mu_max_e, NE), \
uniform(mu_min_i, mu_max_i, NI)))
tau = np.concatenate(([tau_e]*NE, [tau_i]*NI))
```

Now when we create a `NeuronGroup`, the parameters μ and τ will differ across neurons in this group. Do not forget to specify `method='Euler'` to use the Euler integration method, and specify the integration time step $dt = 0.1$ ms.

The differential equation for the membrane potential reads:

$$\frac{dV}{dt} = \frac{1}{\tau}(\mu - V) + I_{syn}. \quad (1)$$

The differential equations for the alpha-synapses with the filter-function $F^y(t)$ specified

in Ref. [2] read:

$$\frac{dI}{dt} = -\frac{1}{\tau_2}(I - x), \quad (2)$$

$$\frac{dx}{dt} = -\frac{1}{\tau_1}x + \sum \delta(t - t^i). \quad (3)$$

Note that synapses in the model are current-based, not conductance based as in the previous assignments. The current flowing through a current-based synapse does not depend on the membrane's voltage, and therefore excitatory and inhibitory synapses have to be determined through positive and negative weights. To keep the units consistent, so that both voltage and current are dimensionless quantities, you will have to scale I_{syn} in the voltage equation Eq. 1 by a scaling factor $\tau_{\text{scale}} = 1$ ms.

To implement sparse connectivity in Brian, specify the probability of connection as an argument when you create connections. For example:

```
Cii = Connection(Pi, Pi, 'x_i', sparseness=p_ii, weight=J_ii)
```

Simulate the balanced network model with uniform connectivity with parameter values from Ref. [2]. Generate a spike-raster plot for a subpopulation of 1600 excitatory neurons and observe irregular asynchronous firing [**Deliverable**]; an example is shown in Fig. 1. Plot the population firing rate sampled at 0.1 ms time window, and the population firing rate smoothed with 5 ms window [**Deliverable**], and notice that there is no obvious synchrony in firing across the population. For an example excitatory neuron in the network, plot the time course of the synaptic currents: excitation, inhibition and their sum [**Deliverable**], as well as the time course of the membrane potential [**Deliverable**]; an example shown in Fig. 1. Explain the irregularity of spiking based on the values of currents and voltage you measured; explain how these quantities are balanced [**Deliverable**].

3 Clustered balanced network

Variability of spontaneous activity in cortex is much higher than predicted by a Poisson process. Trial-to-trial rate fluctuations measured by a Fano factor exceed the Poisson value of 1 across many brain areas. Though the balance between excitation and inhibition can account for irregular asynchronous spiking, it cannot generate supra-Poisson variability — another slower time-scale in the dynamics is required. A hypothesis for the origin of slow rate fluctuations is that cortical circuits are organized in clusters of stronger connected neurons, which form local attractors. In the spontaneous state, activity wanders randomly from attractor to attractor driven by spiking noise, which results in rate fluctuations on slow time scale and, as a consequence, in trial-to-trial variability higher than Poisson. We will implement the clustered balanced network model from Ref. [2] and then compare the statistics of spiking in uniform and clustered networks.

To implement clustered connectivity, we first need to split the excitatory population into `numClust=50` clusters with `Nc=80` neurons in each. To this end, we will create a `subgroup` for each cluster, and since there is a large number of clusters, we will create a whole list of `subgroups`:

```
PeCluster = [ Pe[ i*Nc:( i+1)*Nc ] for i in range( numClust ) ]
```

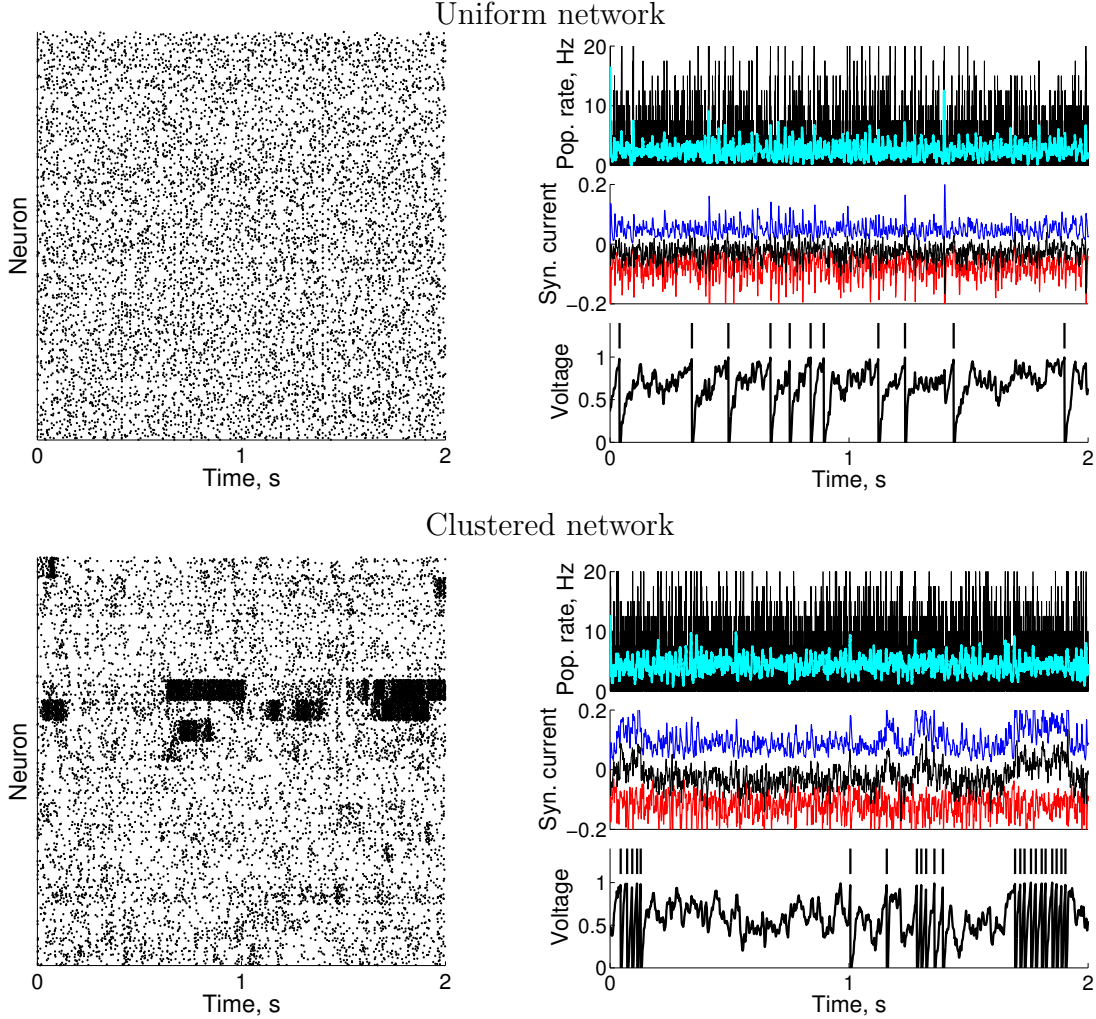


Figure 1: Spontaneous activity in uniform (upper panels) and clustered (lower panels) balanced networks. Left: Spike rasters for a subpopulation of 1600 excitatory neurons. Right: Time course of the population firing rate sampled at 0.1 ms time windows (black) and smoothed with 5 ms time window (cyan). Time course of the synaptic currents: excitation (blue), inhibition (red) and their sum (black), and membrane voltage for an example neuron. Replicated Fig. 1 from Ref. [2].

Note that here we used another way to specify a subgroup by using slicing operation of the group \mathbf{Pe} . Note also that \mathbf{Pe} itself was created as a subgroup of the larger group with $N^E + N^I$ neurons.

All connections involving inhibitory neurons are uniform, and therefore can be implemented exactly as in the previous section by referring to the whole subgroups of excitatory (\mathbf{Pe}) and inhibitory (\mathbf{Pi}) neurons. The excitatory to excitatory connections are clustered. The connection probabilities within ($p_{\text{in}}^{\text{EE}}$) and between ($p_{\text{out}}^{\text{EE}}$) clusters depend on the parameter $R^{\text{EE}} = p_{\text{in}}^{\text{EE}}/p_{\text{out}}^{\text{EE}}$ that controls neuronal clustering. The quantities $p_{\text{in}}^{\text{EE}}$ and $p_{\text{out}}^{\text{EE}}$

were chose so that the connection probability between excitatory neurons remained 0.2 when averaged across all pairs. Calculate how these connection probabilities depend on R^{EE} and on the fraction of neurons in a single cluster f **[Deliverable]**.

The clustered connectivity requires adding a large number of connections to the network, and therefore we will create lists for these connections. We create the list `CeeIn` of within-cluster excitatory connections:

```
net = Network()
...
CeeIn = [None]*numClust
for i in range(numClust):
    CeeIn[i] = Connection(PeCluster[i], PeCluster[i], \
        'x_e', sparseness=p_in, weight=J_ee*p_scale)
net.add( CeeIn[i] )
```

Here we first created a network object `net`, and then we added each connection in our list to this network. This is a good practice when your network model is complex, and you do not want to rely on Brian’s default “magic” functions to find all objects that should be included into the network. Now follow similar steps to add excitatory connections between different clusters of the network.

Remember, that when you create your own network, you should add all relevant objects to it, including neuronal groups, monitors and network operations. Therefore you should go back in your code and add all objects created earlier to the network `net`. To run this network later use the method

```
net.run(simtime)
```

To verify your implementation of clustered connectivity, you can use the python package NetworkX (networkx.lanl.gov) that provides handy tools for analysis and visualization of networks. Once you created a connection in Brian, you can access its properties through methods defined for the connectivity matrix `W` that is an attribute of the connection. For example, to get the array of within-cluster postsynaptic targets for neuron `preCell` in the cluster `iClust` use

```
targets = CeeIn[iClust].W.rowj[preCell]
```

This way you can access the values of synaptic connections created in Brian and use them to generate a NetworkX graph object with the same connectivity. Visualize clustered excitatory connections for a subset of three clusters for three values of parameter R^{EE} **[Extra credit]**. Increasing R^{EE} should result in visible segregation of discrete clusters, an example is shown in Fig. 2.

Run simulations of the clustered balanced network. Generate a raster plot of spontaneous activity for a subpopulation of 1600 excitatory neurons **[Deliverable]**. Record and plot the time course of population firing rate, and for an example neuron record and plot the time course of synaptic currents (excitation, inhibition and their sum) and membrane potential **[Deliverable]**; an example is shown in Fig. 1. Based on your measurements explain how supra-Poisson variability emerges in clustered balanced network **[Deliverable]**.

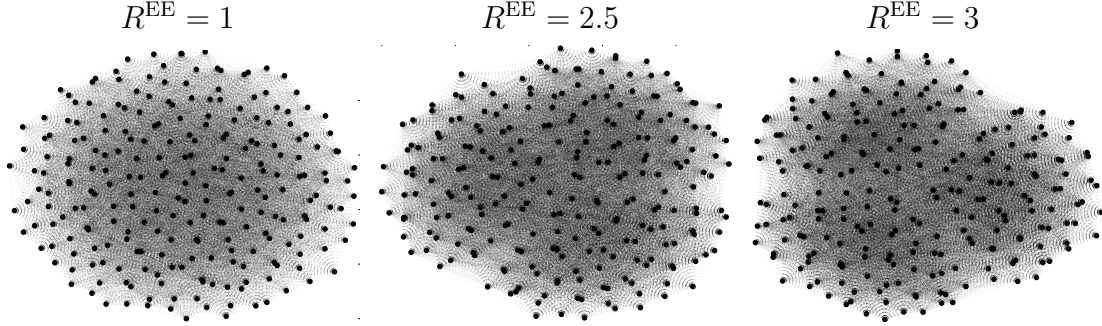


Figure 2: Visualization of connectivity for a subpopulation of 240 excitatory neurons in three clusters for different value of clustering parameter R^{EE} . Replicated Fig. 4 from Ref. [2].

4 Spiking statistics in uniform and clustered balanced networks

To quantify differences in spiking between uniform and clustered balanced networks, we will compute the following statistical measures for the population of excitatory neurons.

1. Distribution of the mean firing rate across the population.
2. Distribution of interspike intervals (ISI) across the population. ISI is the time interval between two consecutive spike times in a neuron: $\text{ISI}_i = t_{i+1} - t_i$.
3. Distribution across the population of the coefficient of variation (CV) of ISIs. The ISI CV for a neuron is defined as the ratio of the ISI standard deviation to the mean ISI:

$$\text{CV} = \frac{\sqrt{\langle \text{ISI}_i^2 \rangle - \langle \text{ISI}_i \rangle^2}}{\langle \text{ISI}_i \rangle} \quad (4)$$

4. Distribution across the population of the Fano factor (FF). FF is defined as the variance over the mean of the spike count N_T in a time window of length T :

$$\text{FF} = \frac{\langle N_{T,i}^2 \rangle - \langle N_{T,i} \rangle^2}{\langle N_{T,i} \rangle} \quad (5)$$

The averaging is performed over many non-overlapping time windows. Use $T = 100$ ms.

Compute and plot these distributions for a uniform and a clustered network [**Deliverable**]; an example is shown in Fig. 3. Discuss the differences in spiking statistics between two network types and explain their origin [**Deliverable**].

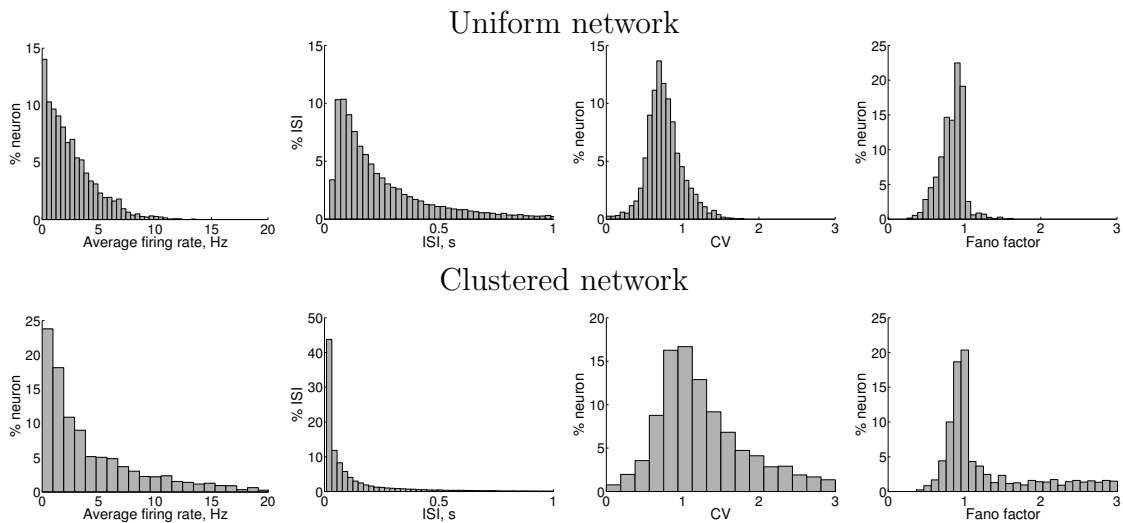


Figure 3: Statistics of spiking in uniform and clustered balanced networks. Replicated Fig. 2 from Ref. [2].

5 Effect of stimulation on spiking variability

Supra-Poisson variability of cortical spontaneous activity is greatly reduced during stimulation [1]. This phenomenon can be measured as a drop in the Fano factor locked on the stimulus onset, which has been reported across many cortical areas. The clustered balanced network model captures the stimulus-driven reduction in variability. Indeed, when a subset of clusters is driven by the stimulus, the activity pattern is biased towards the corresponding attractor. The “energy well” for this attractor becomes deeper and the activity is less likely to escape from this attractor. Therefore activity hops from attractor to attractor are less likely during the stimulation, and as a result variability decreases.

Perform simulations of uniform and clustered balanced networks with a stimulus applied to a subset of 5 clusters (400 neurons) for 400 ms. Repeat the simulation for 100 trials. Remember that when you run different trials, you have to make sure that the network parameters and connectivity are the same on each trial, but the random activity seed is different. Therefore use the same constant seed when you initialize the network parameters and connections, and then set different seeds for different trials to specify the initial conditions for the membrane potential. Plot spike rasters for an example trial for both networks **[Deliverable]**. Compute FF across trials in a sliding 100 ms window for each neuron in the network, and plot the FF averaged over all excitatory neurons as a function of time for uniform and clustered networks **[Deliverable]**; an example is shown in Fig. 4. Explain why you see strong differences in the mean activity of the stimulated clusters for uniform and clustered networks **[Deliverable]**.

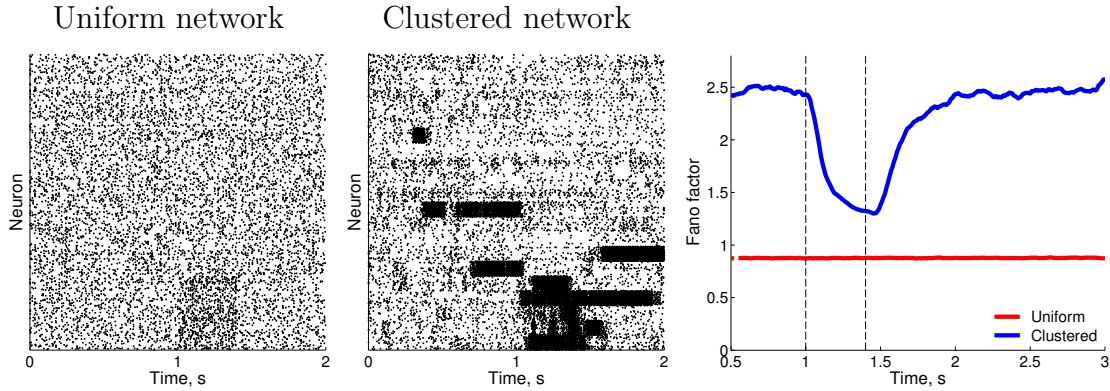


Figure 4: Stimulus quenches spiking variability in clustered but not in uniform networks. Replicated Fig. 7 from Ref. [2].

6 Extension question

Explore how the degree of clustering in the network and the cluster size affect the rate variability. Perform simulations of clustered networks with parameter R^{EE} ranging from 1 to 4. For each value of R^{EE} , estimate the rate variability by computing the population firing rate for the N_c neurons in the cluster over 2 s, in 20 windows of 100 ms each, and then averaging over all clusters and realizations. Plot the dependence of rate variability on the clustering parameter R^{EE} . Explain the non-monotonic dependence. Perform this analysis for different cluster sizes $N_c=60, 120, 140$. Explain the changes in rate variability with the cluster size.

7 List of model parameters

N_E	number of E neurons	4000
N_I	number of I neurons	1000
V_{th}	voltage threshold	1
V_{re}	reset voltage	0
t_{ref}	refractory time	5 ms
τ_e	membrane time constance for E neurons	15 ms
τ_i	membrane time constance for I neurons	10 ms
μ_e	bias current for E neurons	uniform in [1.1,1.2]
μ_i	bias current for I neurons	uniform in [1, 1.05]
$\tau_{2,e}$	decay time for E synapses	3 ms
$\tau_{2,i}$	decay time for I synapses	2 ms
τ_1	synaptic rise time	1 ms
p^{EI}	probability of connection I to E	0.5
p^{IE}	probability of connection E to I	0.5
p^{II}	probability of connection I to I	0.5
p^{EE}	probability of connection E to E	0.2 on average
p_{scale}	scaling factor for connection strength within a cluster	1.9
J^{EE}	synaptic strength E to E	0.024
J^{EI}	synaptic strength I to E	-0.045
J^{IE}	synaptic strength E to I	0.014
J^{II}	synaptic strength I to I	-0.057
dt	integration time step	0.1 ms
μ_{stim}	increase in bias current during stimulation	0.07

References

- [1] MM Churchland, BM Yu, JP Cunningham, LP Sugrue, MR Cohen, GS Corrado, WT Newsome, AM Clark, P Hosseini, BB Scott, DC Bradley, M a Smith, A Kohn, JA Movshon, KM Armstrong, T Moore, SW Chang, LH Snyder, SG Lisberger, NJ Priebe, IM Finn, D Ferster, SI Ryu, G Santhanam, M Sahani, and KV Shenoy. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nat Neurosci*, 13:369–78, 2010.
- [2] A Litwin-Kumar and B Doiron. Slow dynamics and high variability in balanced cortical networks with clustered connections. *Nature neuroscience*, 15(11), September 2012.
- [3] C van Vreeswijk and H Sompolinsky. Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science (New York, N. Y.)*, 274(5293):1724–6, December 1996.