Synaptic reliability

Balazs B Ujfalussy

06/11/2017

This is a demo for illustrating the response of integrate and fire neurons to synaptic input with different release probabilities. We have a single postsynaptic neuron, and $n = \{64, 16384\}$ presynaptic cells, all firing with $r = 2$ Hz and having release probability $p = 0.2$. The strength of the synapses is scaled with the number of neurons, so the mean synaptic drive is kept constant.

We first simulate the net presynaptic spike train.

```r
source('demos/IF_sim.R', chdir=T)
```

```r
p.release <- 0.2 # set the release probability
n.cells <- 2^seq(6,14) # number of cells is changed between 64 and 16000

n <- n.cells[4]
rate.cell <- 2 # Hz - mean firing rate of a single neuron
rate.population <- rate.cell * n # population firing rate
dt <- 1/1000 # s, simulation time step
Tmax <- 1 # s, duration of simulation
L <- Tmax / dt
t <- seq(dt, Tmax, by=dt)
# plot(t, sin(t*(2*pi) * 10)+1, t='l')
# the true pre pattern, sinusoidally modulated Poisson spike train
spikes <- rpois(L, rate.population*dt * (sin(t*(2*pi) * 10)+1))
```

Next, set up the synaptic filter - a simple exponential kernel to model the synaptic potentials

```r
## we filter the spikes to get the EPSPs - this is not a conductance based synapse!
tau <- 5 # ms, synaptic time constant
Nfilt <- 50 # length of the synaptif filter
t.filt <- seq(1, Nfilt)
filt.syn <- exp(-t.filt/tau) # simple exponential synaptic filters - rise is infinitely fast
filt.syn <- filt.syn * 300000 / sum(filt.syn) / n ## filter integralis proportional to 1/n
## 1/300000: scale factor to convert input to nano-Amper
```

Finally we simulate the stochastic synaptic transmission (using the R function *rbinom()*), and generate the postsynaptic response for 10 different trials using the same presynaptic spike train. The postsynaptic cell's firing threshold is set to suppress spiking so that we can focus on the variance of the membrane potential.

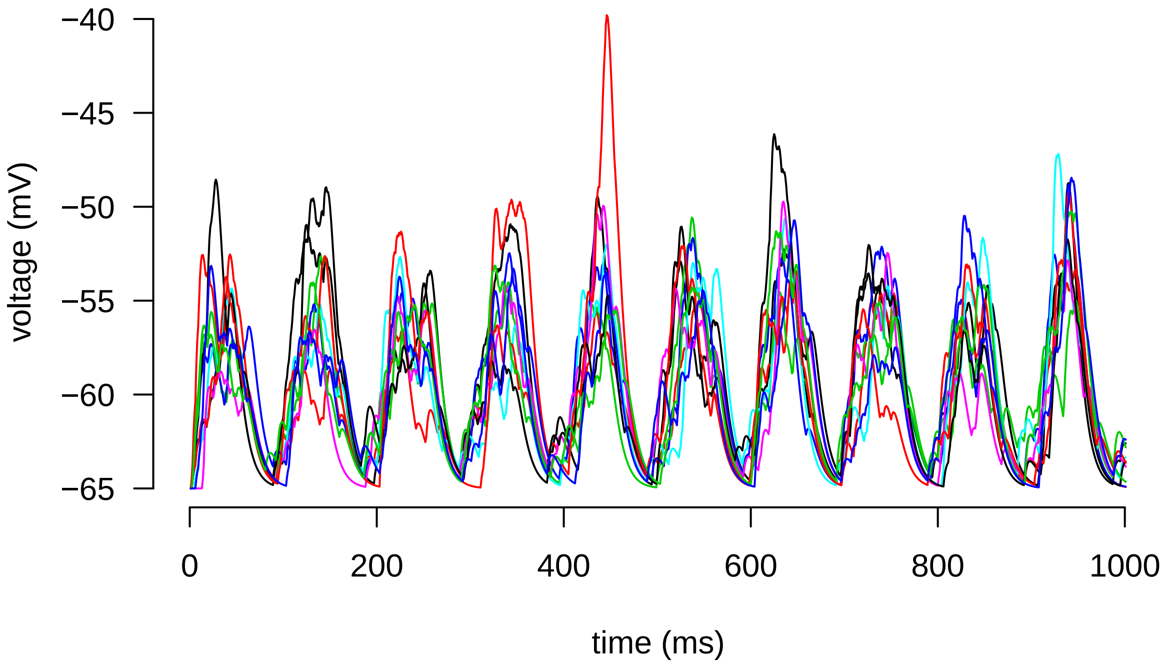
```r
Nrep <- 10 # we will simulate 10 repetitions
vpost <- matrix(NA, L+1, Nrep) # 10 postsynaptic response

for (irep in 1:Nrep){
  release <- rbinom(L, spikes, p.release) # release events

  release <- c(rep(0, Nfilt), release)
  input <- filter(release, filt.syn, method='convolution')
  input <- input[(Nfilt/2):(L+Nfilt/2)]

  v.IF <- sim.IF(I=input, v.rest=-65, Rm=0.045, tau=10, v.threshold=-10, v.reset=-55)
  vpost[,irep] <- v.IF
}
```

```
sds <- apply(vpost, 1, sd)
matplot(vpost, t="l", lty=1, axes=F, xlab="time (ms)", ylab="voltage (mV)"); axis(1); axis(2, las=2)
```



The task is to calculate how the variance of the membrane potential (across trials) depends on the number of neurons. Don't forget to interpret the results!