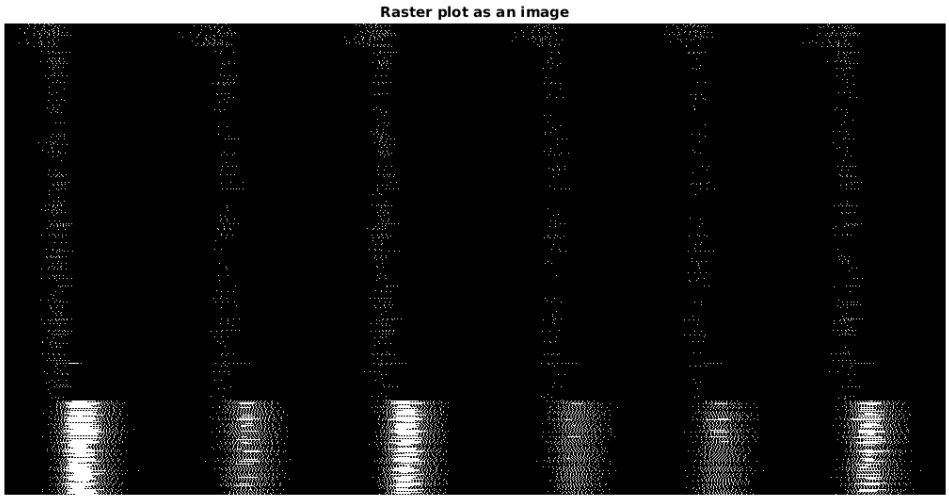
Q2 Dynamical Random network

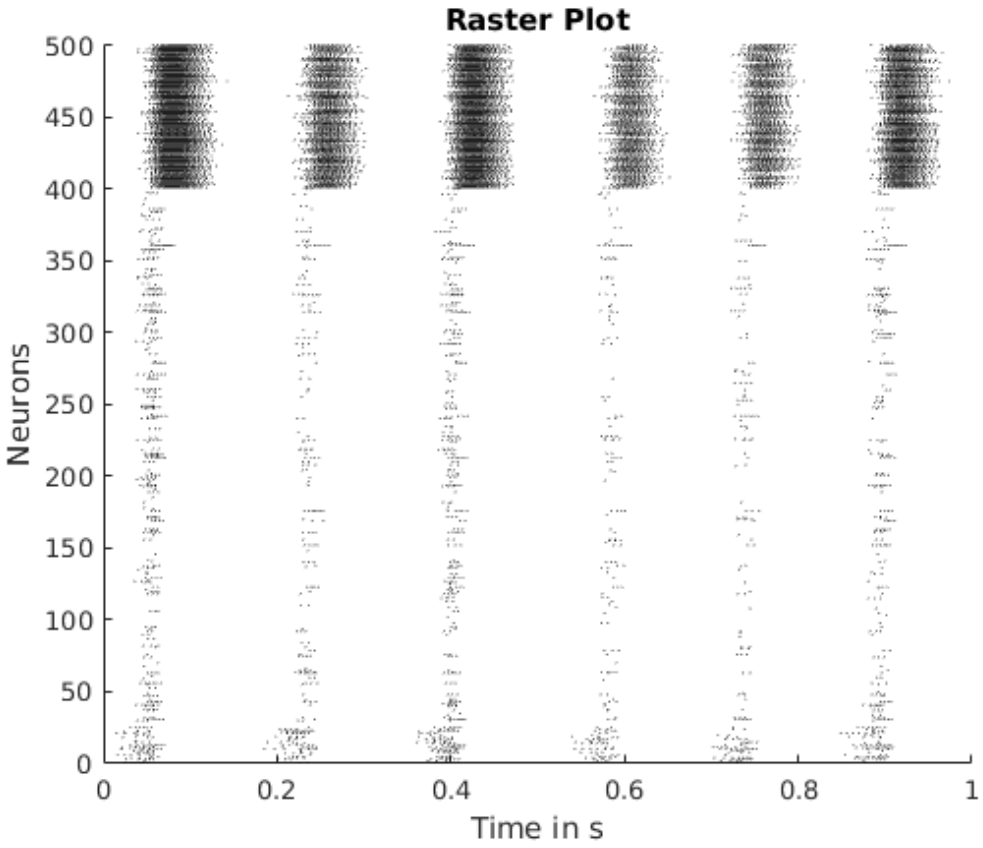
Table of Contents

Part A

```
seed=200;
rng(seed,'twister');

ms=1E-3;
N=500;
fanout_ratio=N/10;
Ne=N*0.80;
Ni=N-Ne;
%  creating network
% 1-400 neurons= exitatory
% 400*500 neurons= inhibitory
fanin=cell(1,N);
fanout_matrix=zeros(N,fanout_ratio);
for i=1:Ne
    fanout_matrix(i,:)=randperm(N,fanout_ratio);
end
for i=Ne+1:N
    fanout_matrix(i,:)=randperm(Ne,fanout_ratio);
end



Weights_matrix=3000*ones(N,fanout_ratio);
Weights_matrix(Ne+1:end,:)=-3000;

delay_matrix=randi([1,20],[N,fanout_ratio])*ms;
delay_matrix(round(N*0.8)+1:end,:)=1*ms;

% constants
delta_t=1*ms;
T=1000*ms;
t=linspace(0,T,T/delta_t);
Io=1E-12;
tau=15*ms;
tau_s=tau/4;
EL=-70*1E-3;
gL=30*1E-9;
Vt=20*1E-3;
```

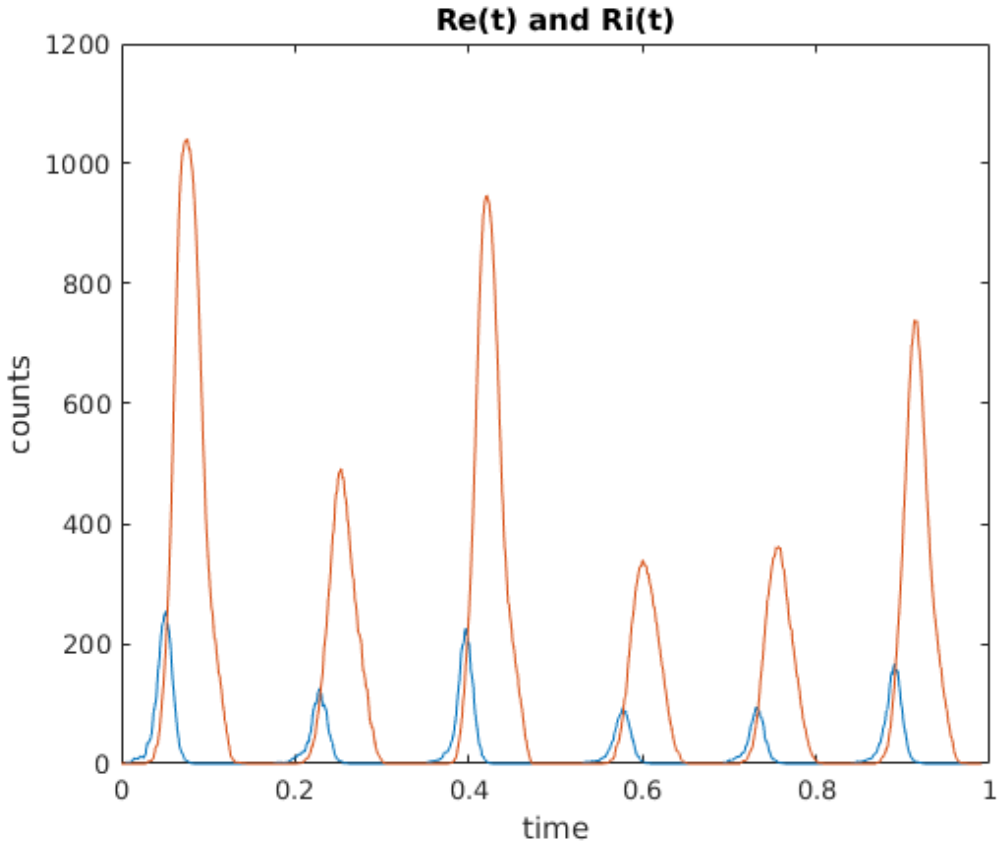
```
C=300*1E-12;
Rp=2*ms;
ws=3000;

% % forming Iext matrix

lambda=100;
myPoissonSpikeTrain = rand(25, T/delta_t) < lambda*delta_t;
Iext_t= @(ts,t) Io*ws*(exp(-(t-ts)/tau)-exp(-(t-ts)/tau_s)).*(t>ts);
Iext=zeros(25,T/delta_t);
for i=1:25
    ts=find(myPoissonSpikeTrain(i,:)==1)*delta_t;
    for k=1:size(ts,2)
        Iext(i,:)=Iext(i,:)+Iext_t(ts(k),t);
    end

end


[V,t,spikes,none]=LIF_dynamic( delta_t,T,N,fanout_matrix,Weights_matrix,delay_matr

figure()
imshow(spikes*255);
title('Raster plot as an image');
plotRaster(spikes,t);

MEvent. CASE!
MEvent. CASE!
```





Part B

```
Re_temp=sum(spikes(1:round(N*0.8),:),1);
Ri_temp=sum(spikes(round(N*0.8)+1:end,:),1);
Re=zeros(1,T/delta_t-10*ms/delta_t);
Ri=zeros(1,T/delta_t-10*ms/delta_t);
for i=1:T/delta_t-10*ms/delta_t
    Re(i)=sum(Re_temp(i:i+10*ms/delta_t));
    Ri(i)=sum(Ri_temp(i:i+10*ms/delta_t));
end

figure();
plot(t(1:T/delta_t-10*ms/delta_t),Re,t(1:T/delta_t-10*ms/delta_t),Ri);
title('Re(t) and Ri(t)')
xlabel('time');ylabel('counts');
```



Part C

Spikes happens after approximately regular intervals because of random nature of input.Neural Network adaptes to random input and starts giving spikes

Published with MATLAB® R2015b