Q3 Adaptive Exponential Integrate-and-Fire Model

Table of Contents

Q3 part-1 writing difference equations 1
Q3 part-a Steady state values 1
Q2 part-c Transient solution for RS type neuron 2
Q2 part-c Transient solution for IB type neuron 5
Q2 part-c Transient solution for CH type neuron 8

Q3 part-1 writing difference equations

```
display('difference equation for Izhikevich Model');
display('V(n+1)=1/C(#gL(V(n)#EL)+gL#Texp(# U(n)+Iapp(n)+V(n)');
display('and');
display('U(n+1)=1/#w(a[V(n)#EL]#U(n))+U(n)');
```

```
difference equation for Izhikevich Model
V(n+1)=1/C(#gL(V(n)#EL)+gL#Texp(# U(n)+Iapp(n)+V(n)
and
U(n+1)=1/#w(a[V(n)#EL]#U(n))+U(n)
```

Q3 part-a Steady state values

solving for steady state for RS neurons

```
[C,gL,EL,vT,del_T,a,tau_w,b,Vr]=neuron_data_q3(1);
syms V U
[sol_V,sol_U] = vpasolve([-gL*(V-EL)+(gL*del_T*exp((V-vT)/del_T)) ==
 U , a*(V-EL) == U], [V,U] );
sol_V= double(sol_V);
sol_U=double(sol_U);
V_steady_state = double(sol_V);
U_steady_state = double(sol_U);
display('for RS neurons');
display(strcat('steady state V= ',num2str(V_steady_state)));
display(strcat('steady state U= ',num2str(U_steady_state)));

% for IB neurons
[C,gL,EL,vT,del_T,a,tau_w,b,Vr]=neuron_data_q3(2);
syms V U
[sol_V,sol_U] = vpasolve([-gL*(V-EL)+(gL*del_T*exp((V-vT)/del_T)) ==
 U , a*(V-EL) == U], [V,U] );
sol_V= double(sol_V);
sol_U=double(sol_U);
V_steady_state = double(sol_V);
U_steady_state = double(sol_U);
display('for RS neurons');
```

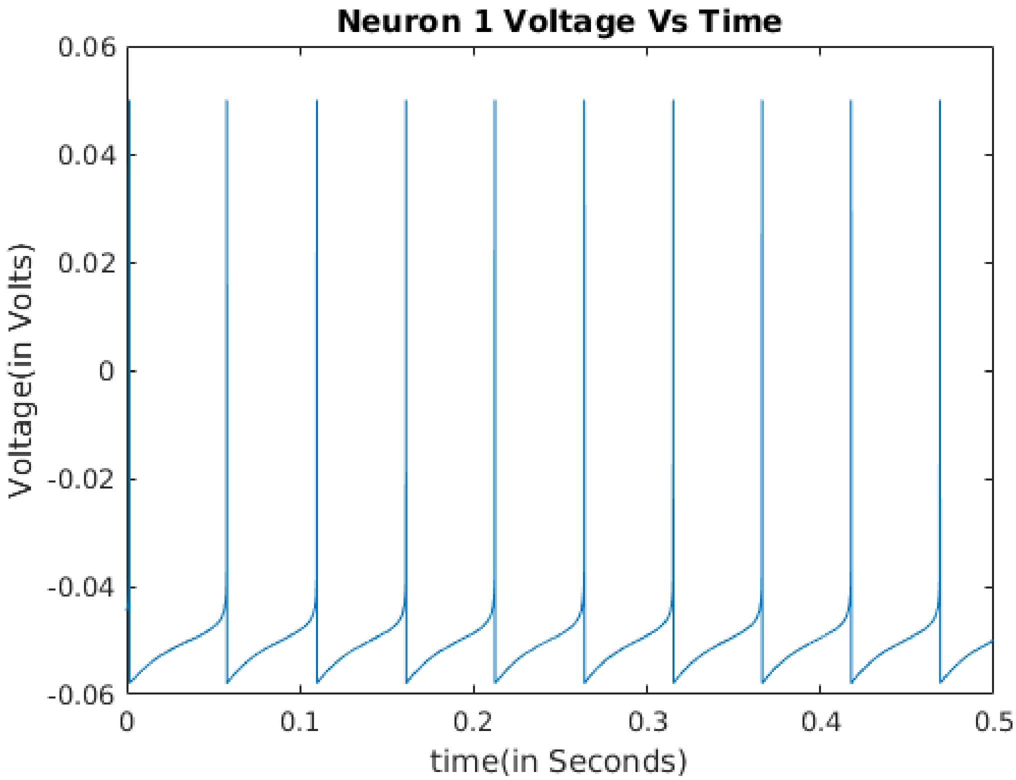
```
display(strcat('steady state V= ',num2str(V_steady_state)));
display(strcat('steady state U= ',num2str(U_steady_state)));

% for CH neurons
[C,gL,EL,vT,del_T,a,tau_w,b,Vr]=neuron_data_q3(3);
syms V U
[sol_V,sol_U] = vpasolve([-gL*(V-EL)+(gL*del_T*exp((V-vT)/del_T)) ==
 U , a*(V-EL) == U], [V,U] );
sol_V= double(sol_V);
sol_U=double(sol_U);
V_steady_state = double(sol_V);
U_steady_state = double(sol_U);
display('for RS neurons');
display(strcat('steady state V= ',num2str(V_steady_state)));
display(strcat('steady state U= ',num2str(U_steady_state)));
```

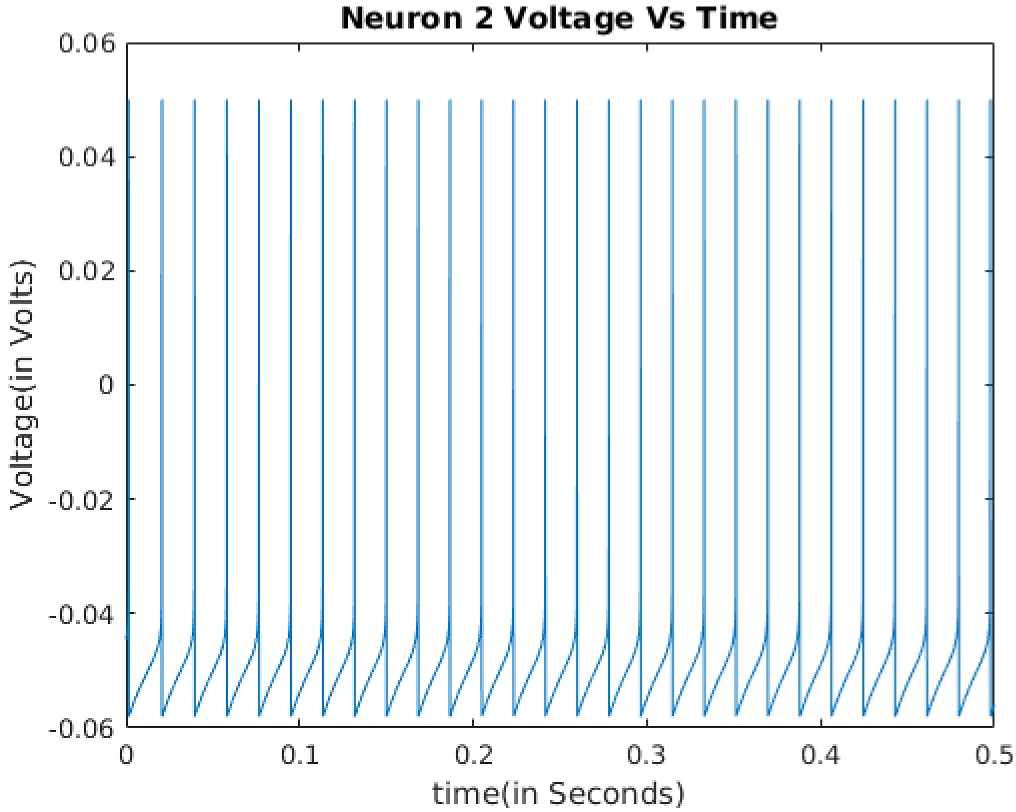
```
for RS neurons
steady state V=-0.044548
steady state U=5.0904e-11
for RS neurons
steady state V=-0.046018
steady state U=4.7927e-11
for RS neurons
steady state V=-0.046062
steady state U=2.3876e-11
```

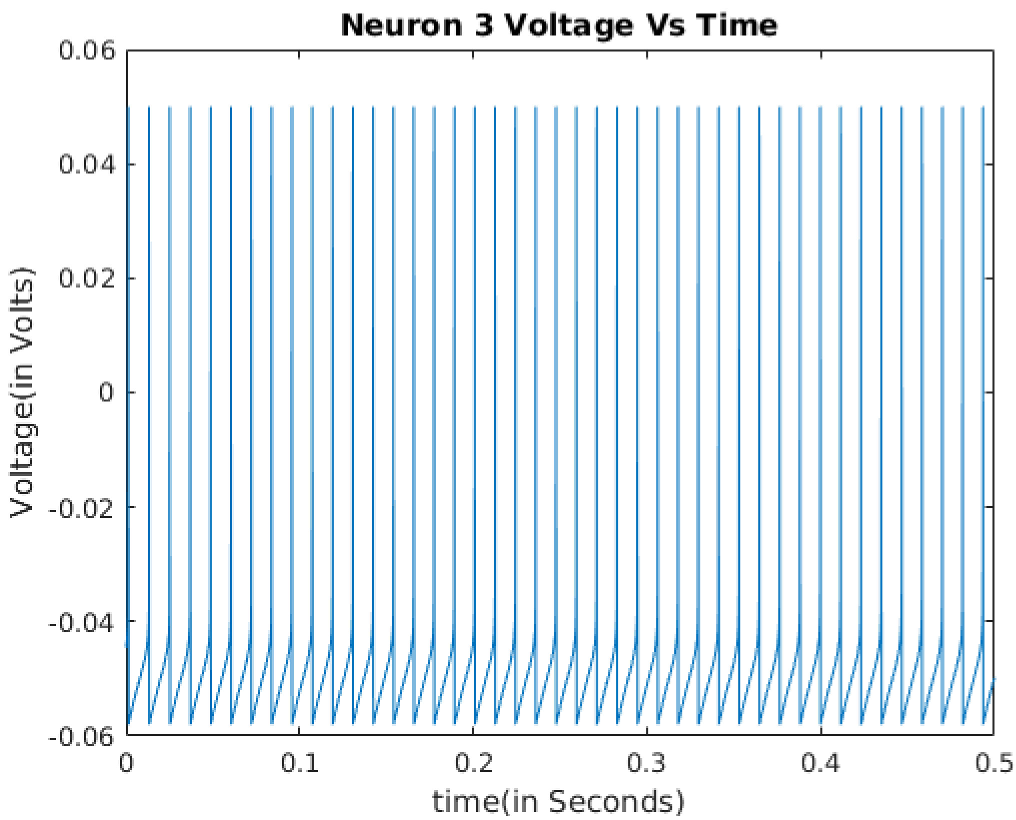
Q2 part-c Transient solution for RS type neuron

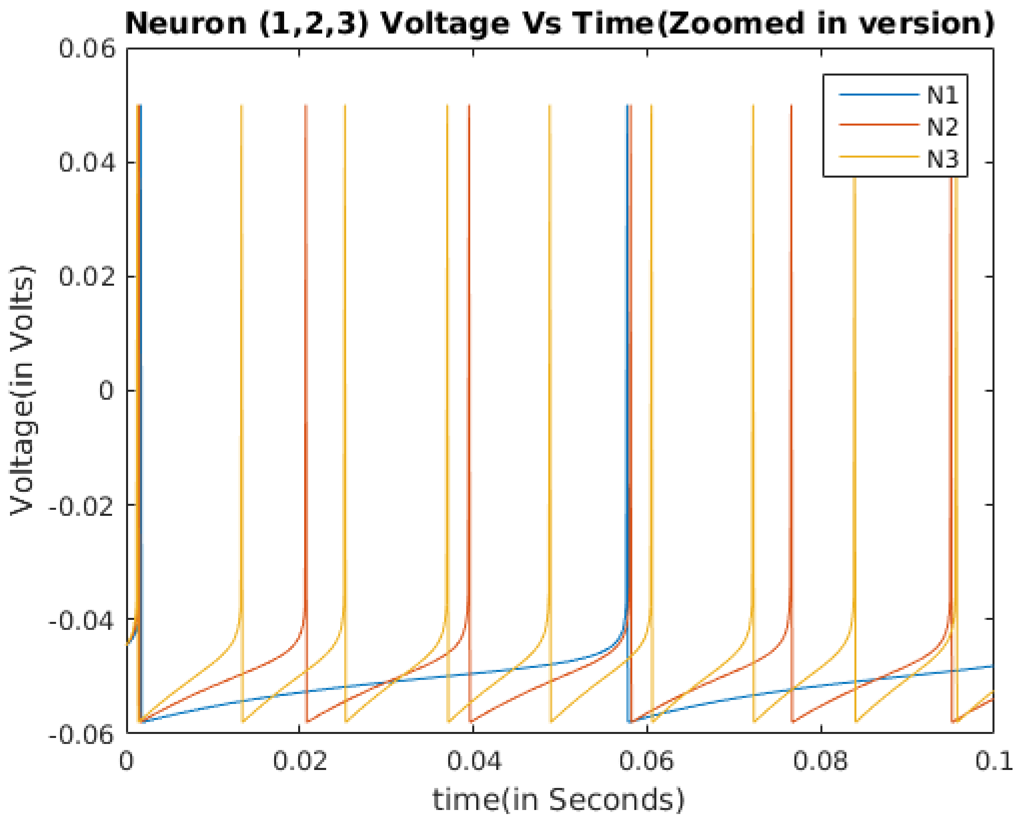
```
N=3;
T=0.500;
delta_t= 0.1 * 10^-3;
M=T/delta_t;
input=zeros(N,M);
for i=1:N
    input(i,:)=(1.5+i)*100*10^-12;
end

[y,z] = euler_q3(delta_t,T,input,1);
x = 0:delta_t:T;


figure()
plot(x,y(1,:));
title('Neuron 1 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x,y(2,:));
title('Neuron 2 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');
```

```
figure()
plot(x,y(3,:));
title('Neuron 3 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x(1:1000),y(1,1:1000),x(1:1000),y(2,1:1000),x(1:1000),y(3,1:1000));
title('Neuron (1,2,3) Voltage Vs Time(Zoomed in version)');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');
legend('N1','N2','N3');
```




Neuron 2 Voltage Vs Time
Voltage(in Volts)
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
time(in Seconds)
0
0.1
0.2
0.3
0.4
0.5

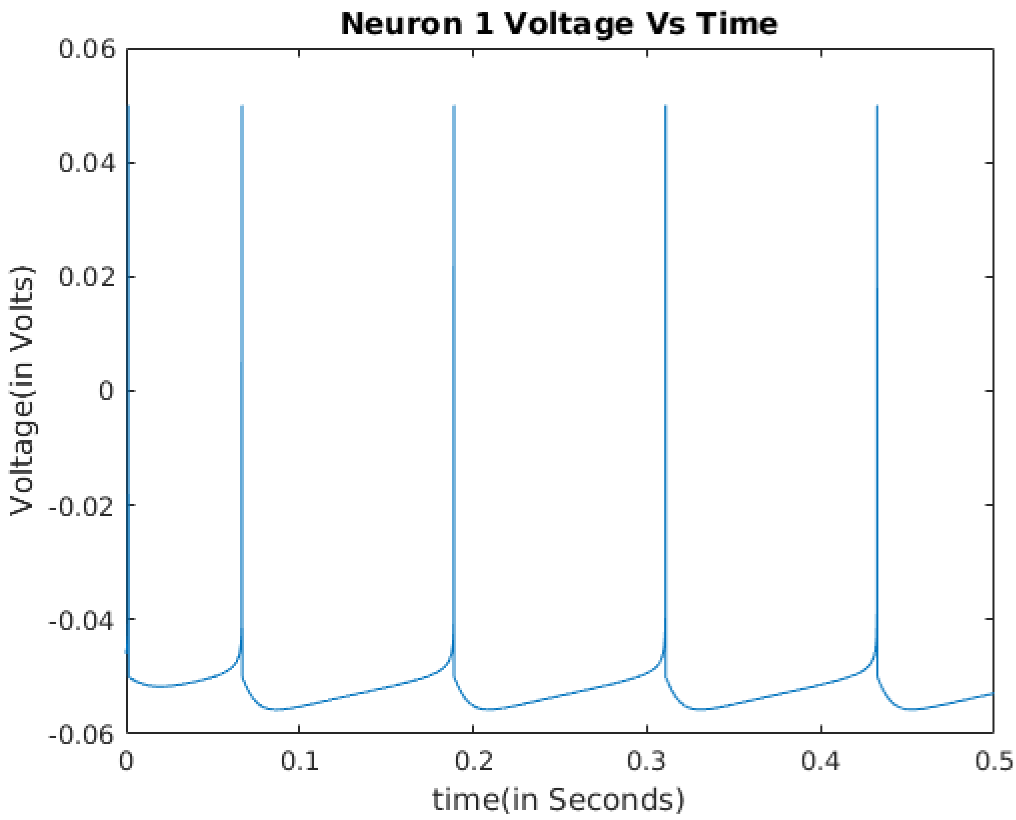

Neuron 3 Voltage Vs Time
Voltage(in Volts)
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
time(in Seconds)
0
0.1
0.2
0.3
0.4
0.5

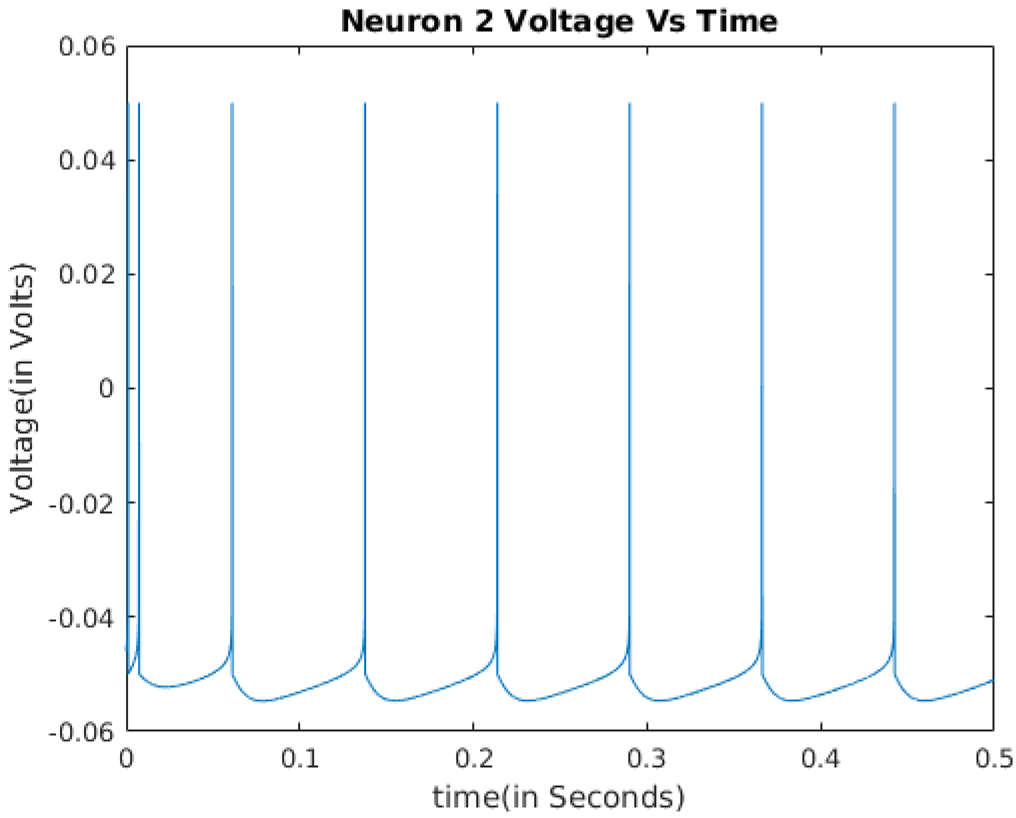


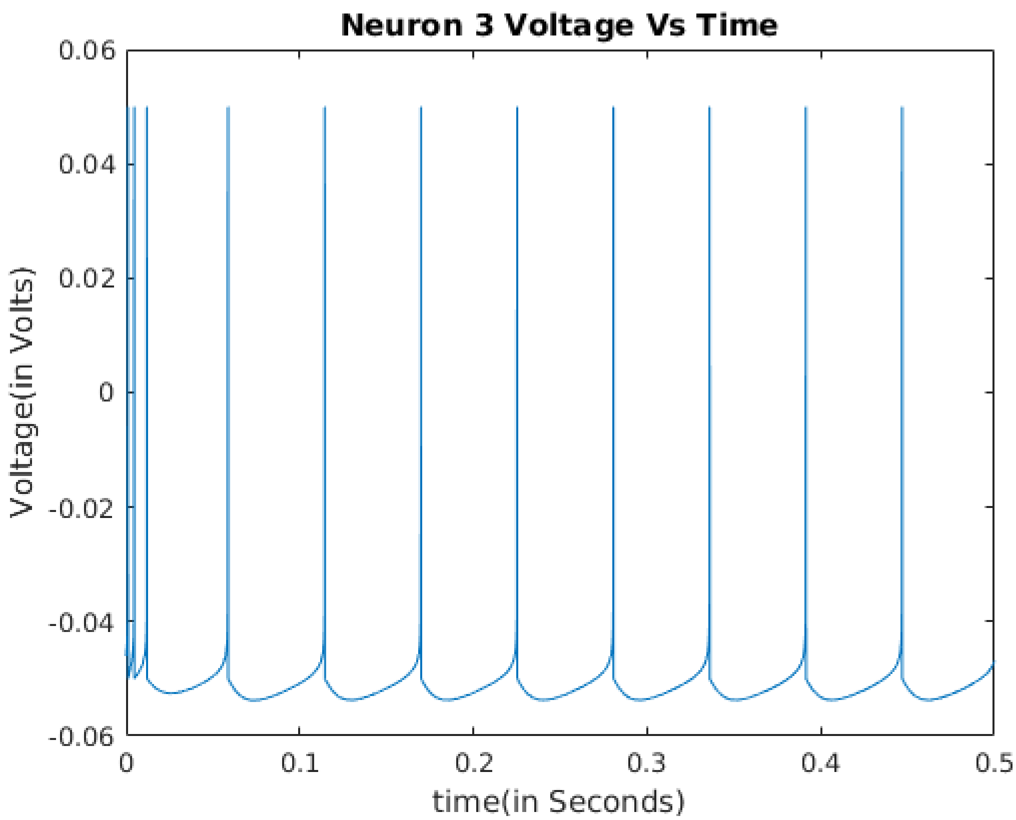
Q2 part-c Transient solution for IB type neuron

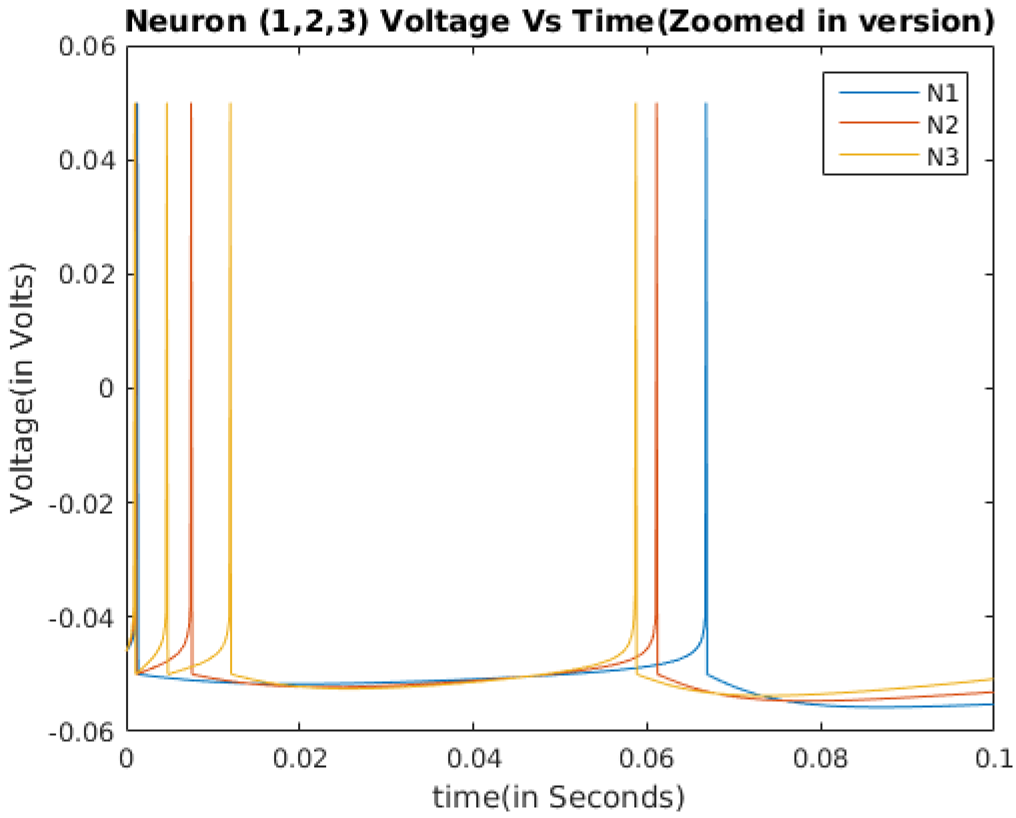
```
N=3;
T=0.500;
delta_t= 0.1 * 10^-3;
M=T/delta_t;
input=zeros(N,M);
for i=1:N
    input(i,:)=(1.5+i)*100*10^-12;
end

[y,z] = euler_q3(delta_t,T,input,2);
x = 0:delta_t:T;



figure()
plot(x,y(1,:));
title('Neuron 1 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x,y(2,:));
title('Neuron 2 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');
```

```
figure()
plot(x,y(3,:));
title('Neuron 3 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x(1:1000),y(1,1:1000),x(1:1000),y(2,1:1000),x(1:1000),y(3,1:1000));
title('Neuron (1,2,3) Voltage Vs Time(Zoomed in version)');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');
legend('N1','N2','N3');
```




Neuron 2 Voltage Vs Time
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
Voltage(in Volts)
0
0.1
0.2
0.3
0.4
0.5
time(in Seconds)

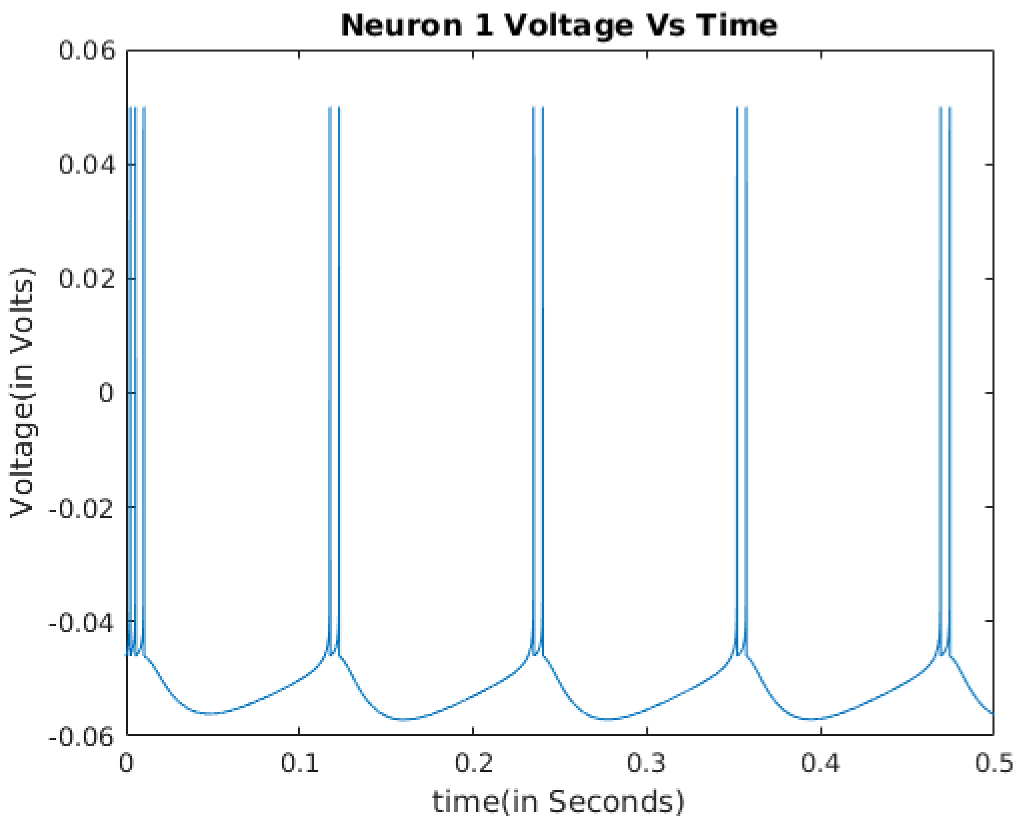

Neuron 3 Voltage Vs Time
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
Voltage(in Volts)
0
0.1
0.2
0.3
0.4
0.5
time(in Seconds)

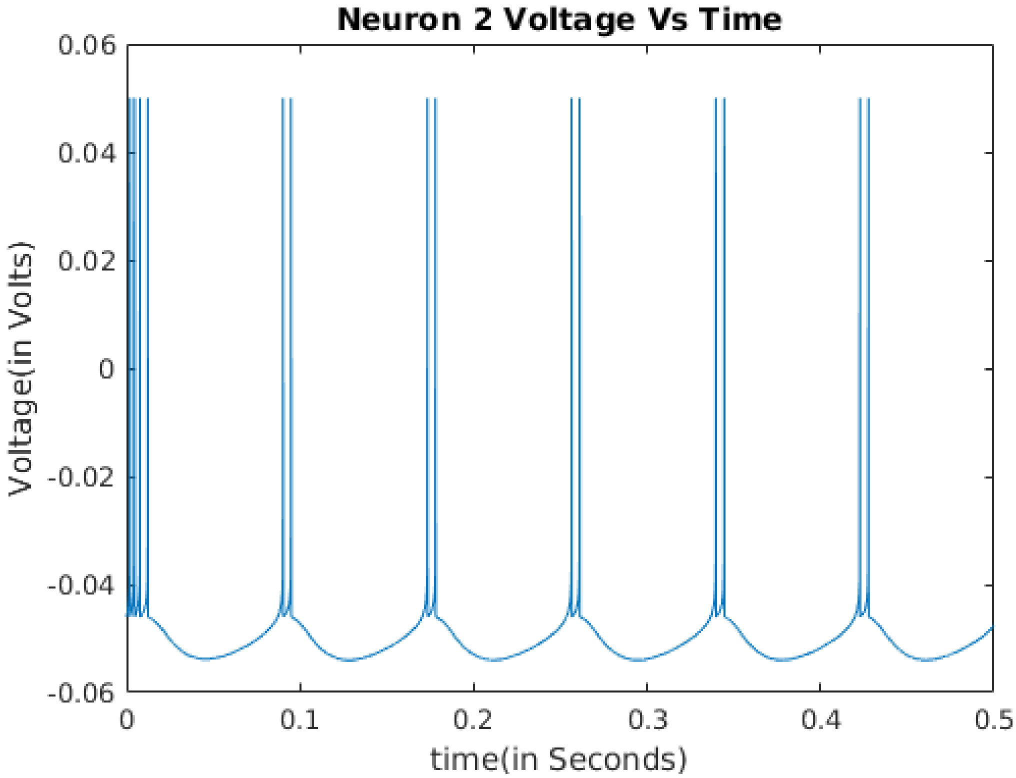


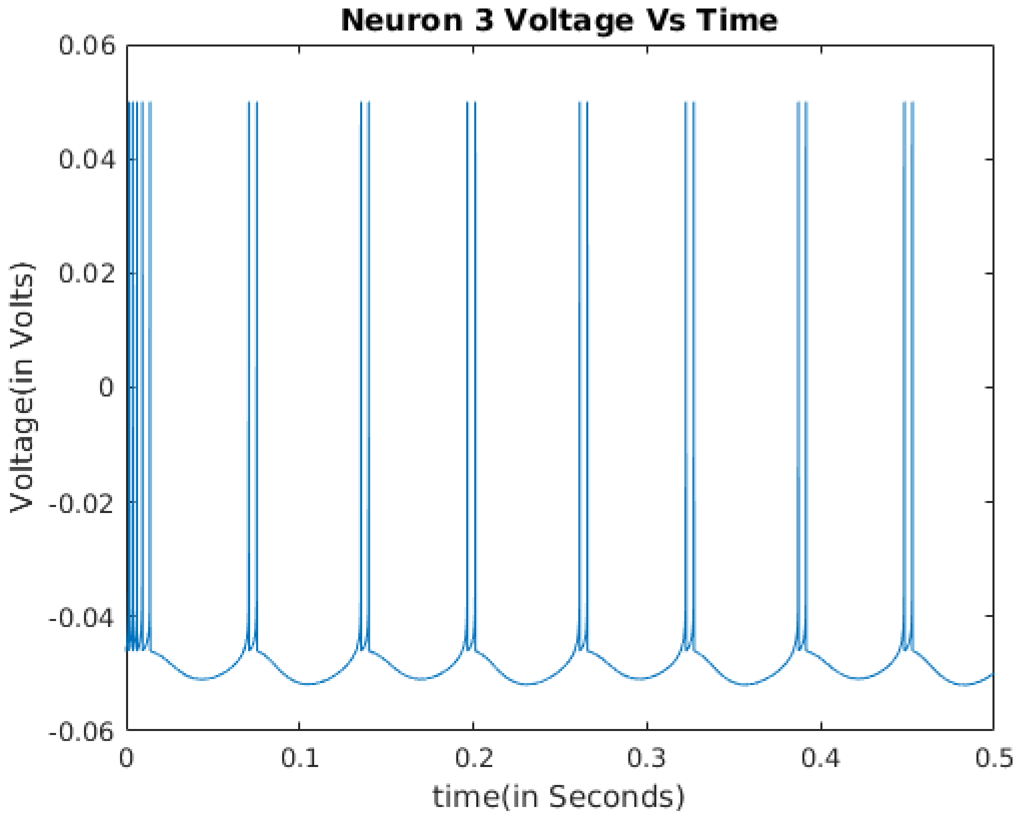
Q2 part-c Transient solution for CH type neuron

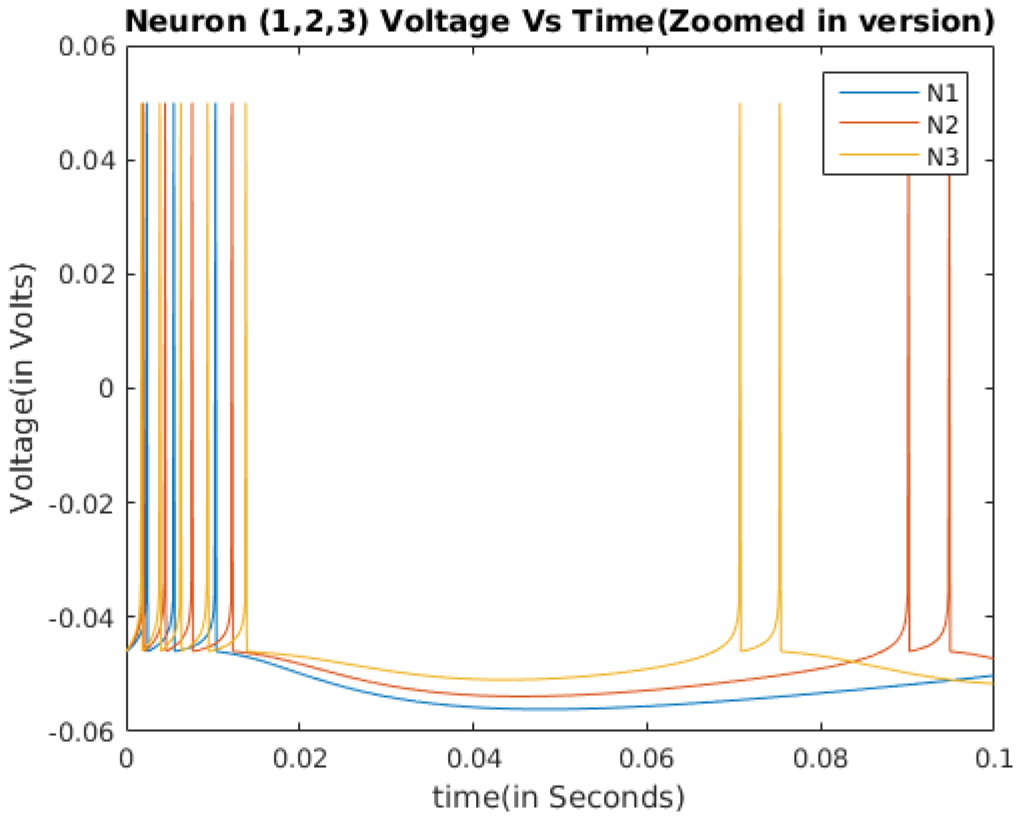
```
N=3;
T=0.500;
delta_t= 0.1 * 10^-3;
M=T/delta_t;
input=zeros(N,M);
for i=1:N
    input(i,:)=(1.5+i)*100*10^-12;
end

[y,z] = euler_q3(delta_t,T,input,3);
x = 0:delta_t:T;



figure()
plot(x,y(1,:));
title('Neuron 1 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x,y(2,:));
```

```
title('Neuron 2 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x,y(3,:));
title('Neuron 3 Voltage Vs Time');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');

figure()
plot(x(1:1000),y(1,1:1000),x(1:1000),y(2,1:1000),x(1:1000),y(3,1:1000));
title('Neuron (1,2,3) Voltage Vs Time(Zoomed in version)');
xlabel('time(in Seconds)');ylabel('Voltage(in Volts)');
legend('N1','N2','N3');
```




Neuron 2 Voltage Vs Time
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
Voltage(in Volts)
0
0.1
0.2
0.3
0.4
0.5
time(in Seconds)


Neuron 3 Voltage Vs Time
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
Voltage(in Volts)
0
0.1
0.2
0.3
0.4
0.5
time(in Seconds)



Published with MATLAB® R2015b