ANNEXE 1:

Quelques jeux de paramètres:

%Paramètres du modèle "regular spiking"

a = 0.02;

b = 0.2 ;

c=-65;

d= 8;

a1=0.04;

a2=5;

C=140;

Ts=3000;

N=10000;

Te=Ts/N;

I=8;

(par la suite, quand I n'est pas précisé, on prendra I=8 par défaut)

Paramètres du modèle "intrinsically bursting"

a = 0.02;

b = 0.2 ;

c=-55;

d= 4;

I=10;

Paramètres du modèle "chattering"

a = 0.02;

b = 0.2 ;

c=-50;

d= 2;

I=10;

ANNEXE 2

schéma simulink du neurone

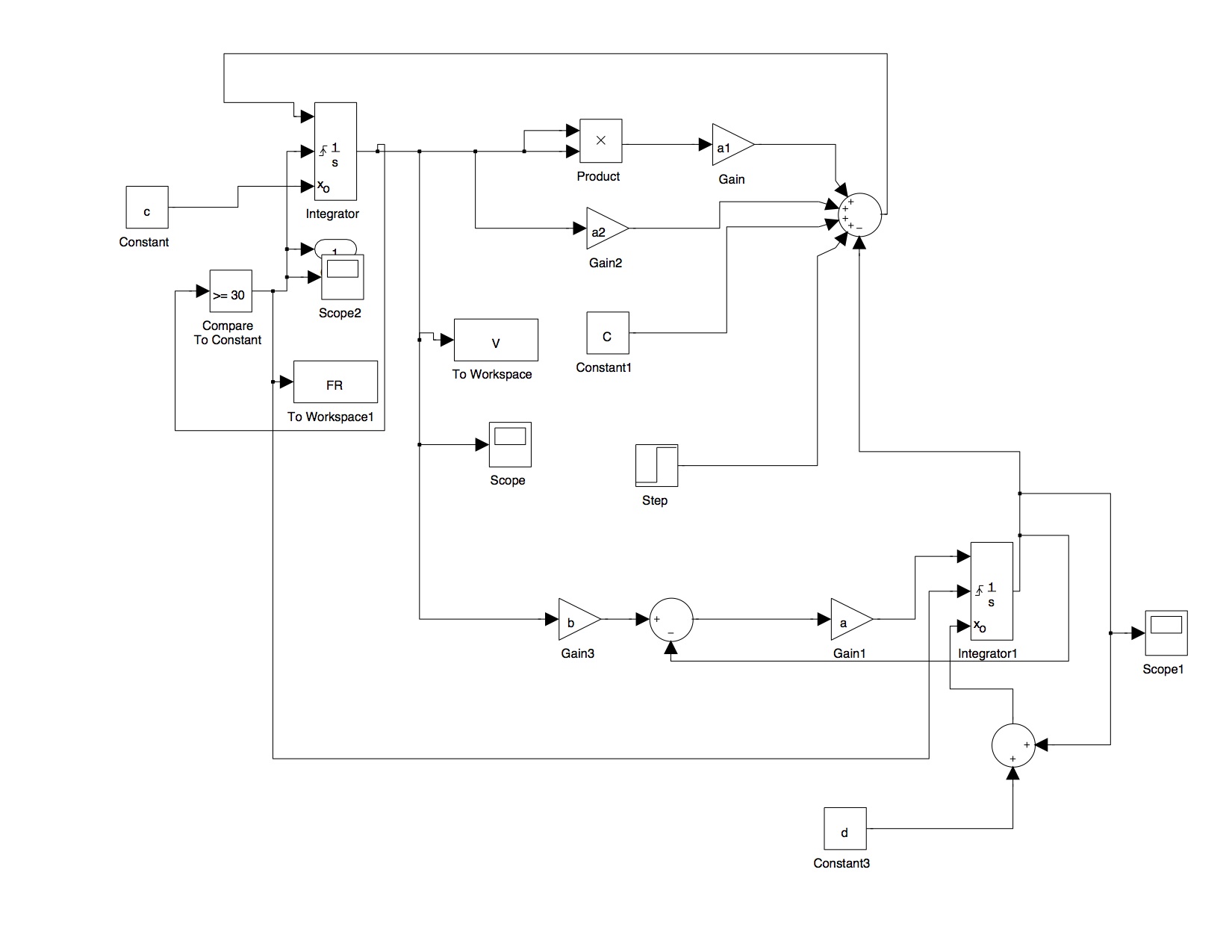
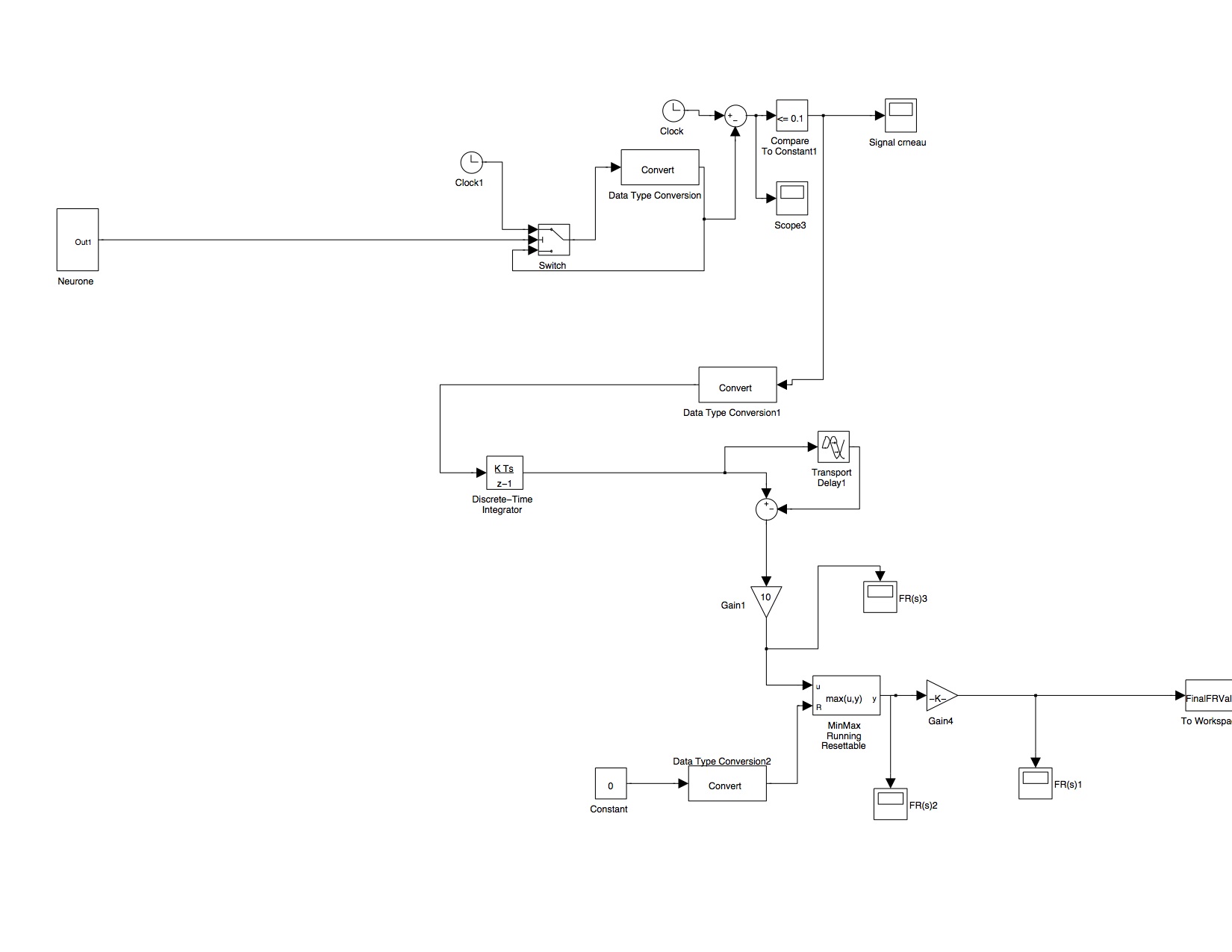


schéma simulink du dispositif de comptage

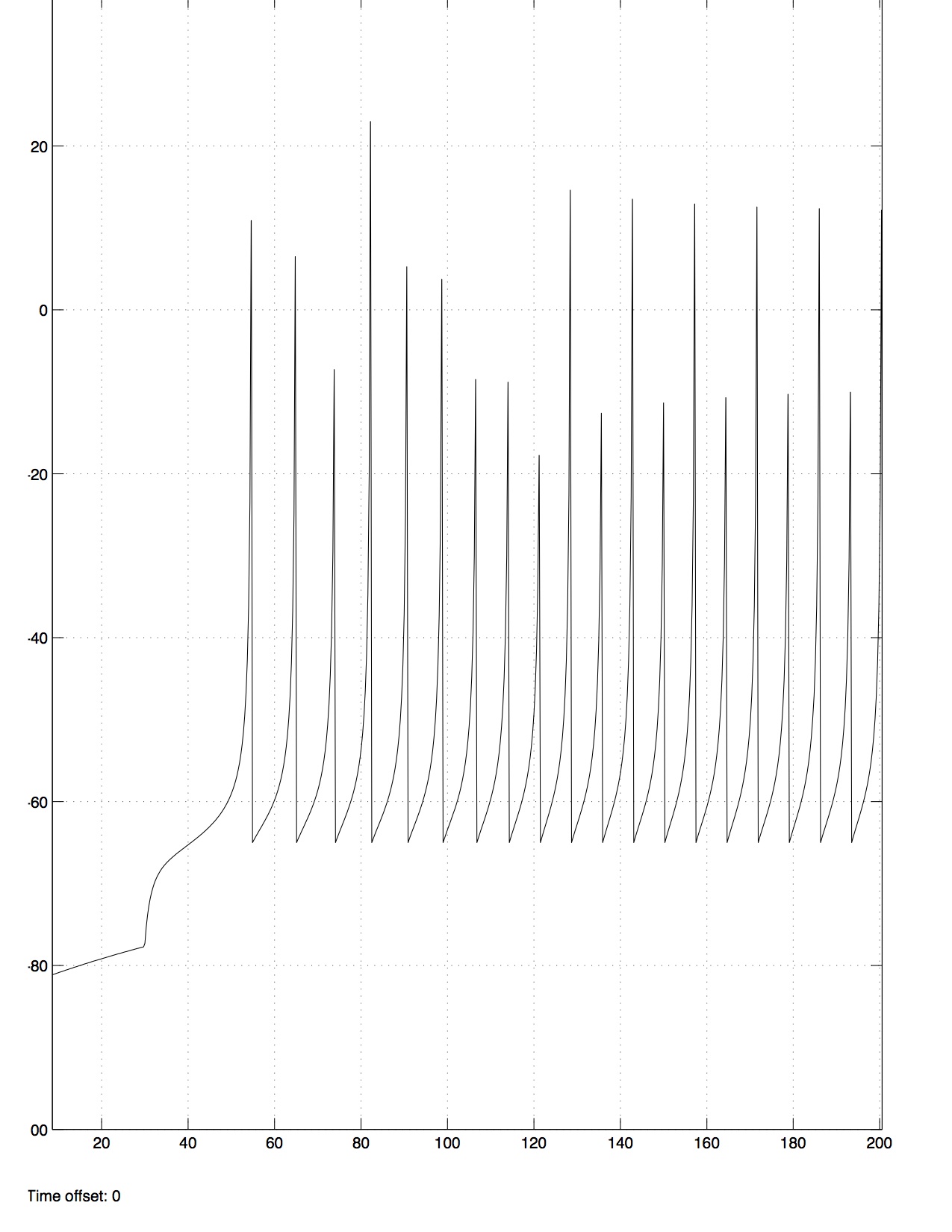


courbes obtenues pour chacun des modèles: V=f(t), V en mV, t en ms

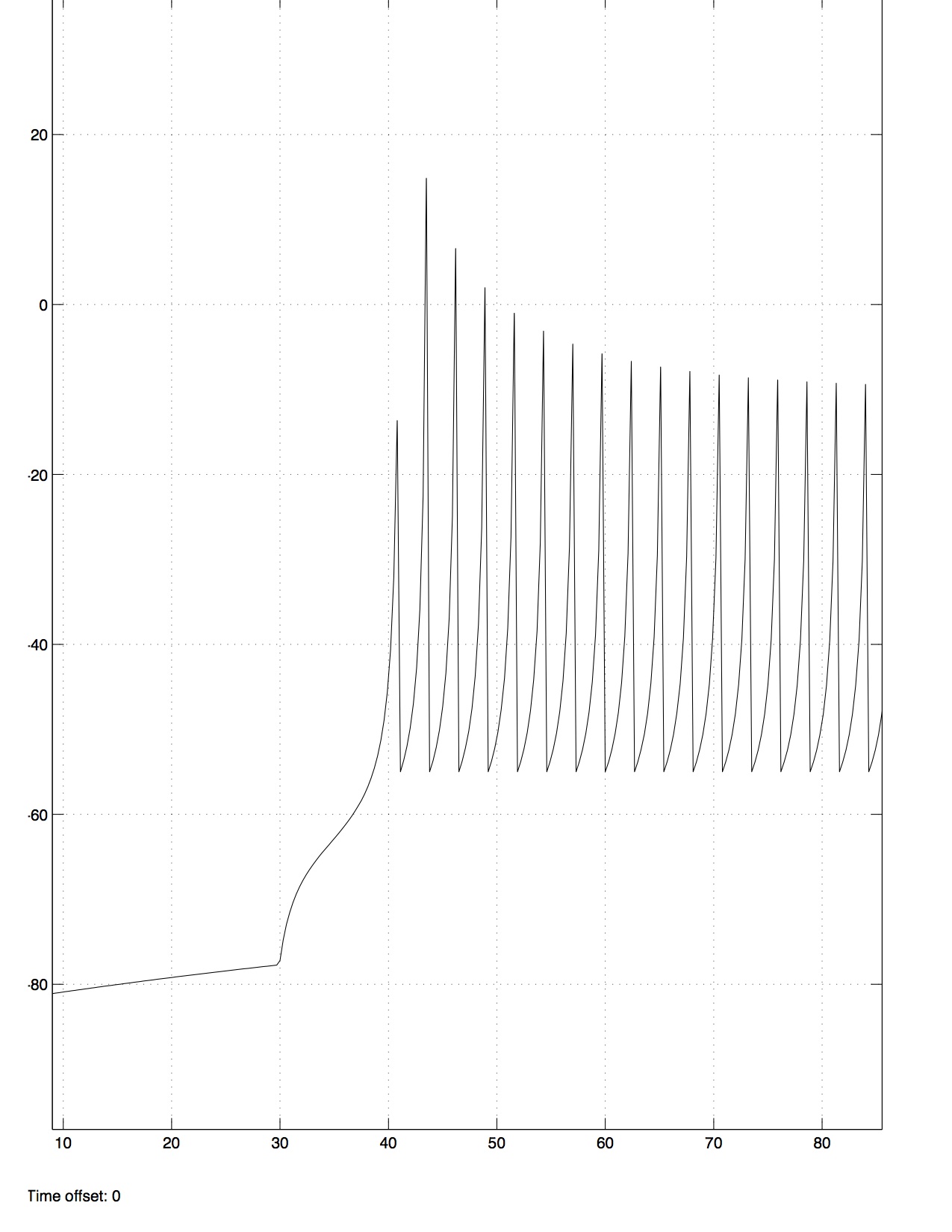
temps de départ de l'échelon de I =30 ms

10000points;

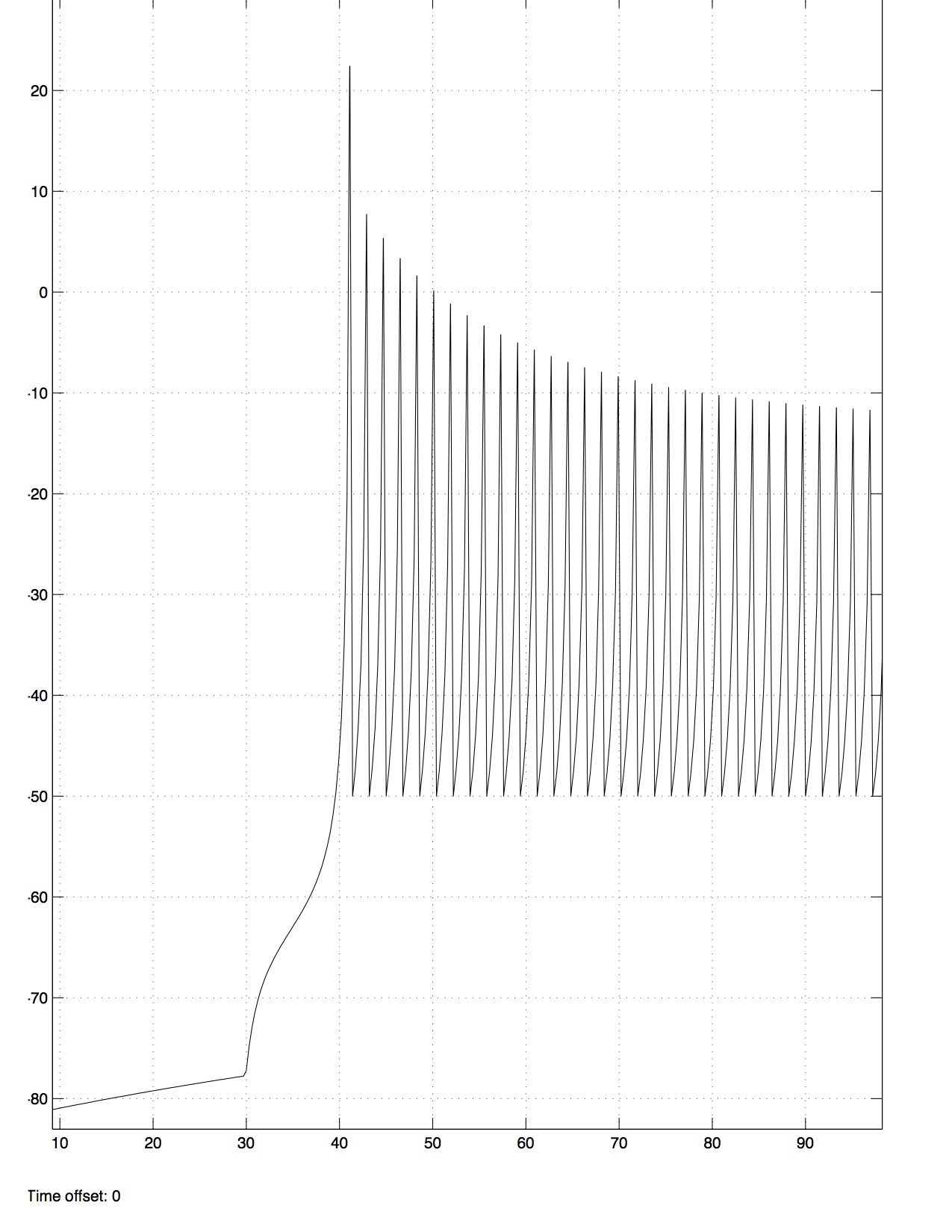
T simulation =150ms



**regular spiking**



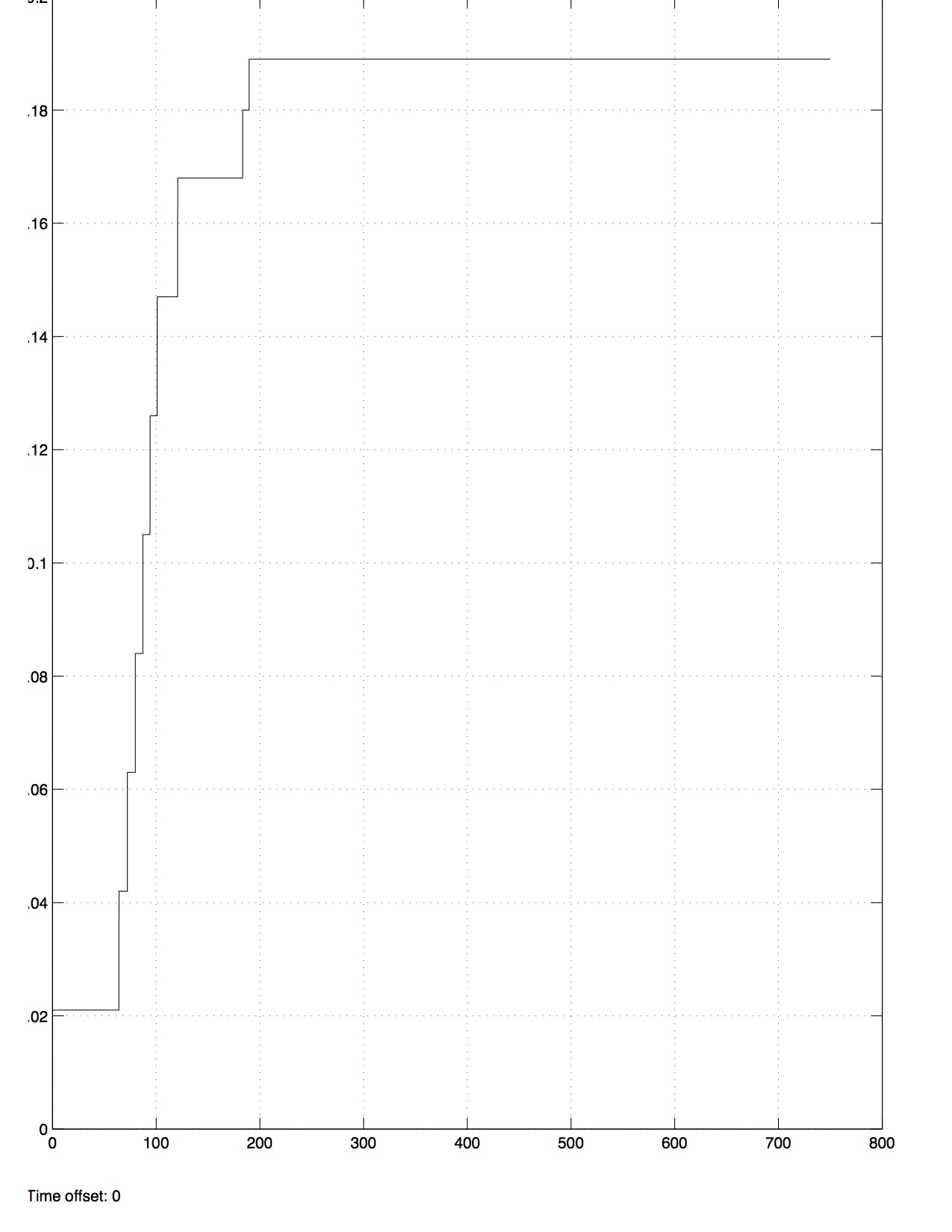
**intrinsically bursting**



**Chattering**

résultat du Firing Rate

pour une simulation de 50 000pts, 150ms, fenêtre 50ms, avec le max en temps réel du FR, pour le regular spiking



ANNEXE 3:

%Obtention de la sigmoïde

VI=(0:1:100);

VFR=eye(1,length(VI));

for i = 1 : length(VI)

I=i;

sim('projet\_neurones\_simulation1');

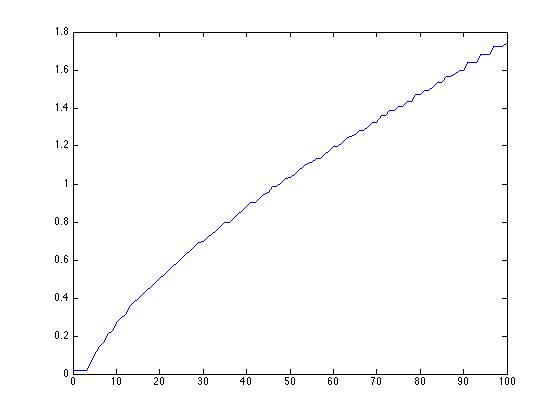
VFR(i)=FinalFRValue(length(FinalFRValue));

end

plot(VI,VFR)

courbe obtenue: Firing Rate = f(I), I en mA et Firing Rate en pics/milisecondes

pour des simulations de 150ms, fenêtre de 50ms, courbe finale de 101 points (101 simulations au total)



ANNEXE 4:

Programme Matlab de calcul de V

function [V,A] = lastchancefaux(a,b,c,d,i,tstep,Te,Ts)

N=Ts/Te;

A=zeros(1,N);

T=[0:Te:(N-1)\*Te];

I=[];

Tstep=Te\*floor(tstep/Te);

for k=1:Tstep/Te

I(k)=0;

end

for k=Tstep/Te+1:N

I(k)=i;

end

V=[-65];

U=[b\*V(1)];

for k=1:N-1

V(k+1)=V(k)+Te\*(0.04\*V(k)^2+5\*V(k)+140-U(k)+I(k));

U(k+1)=U(k)+Te\*a\*(b\*V(k)-U(k));

if V(k+1)>=30

V(k+1)=c;

U(k+1)=U(k+1)+d;

A(k+1)=1;

end

end

% plot(T,V,'magenta')

% hold on

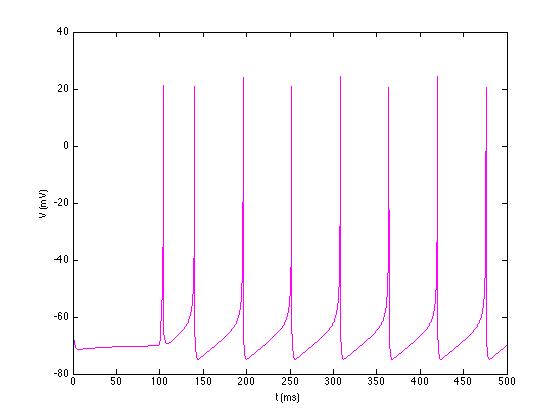
% plot(T,U,'red')

**Résultats pour les 3 modèles**

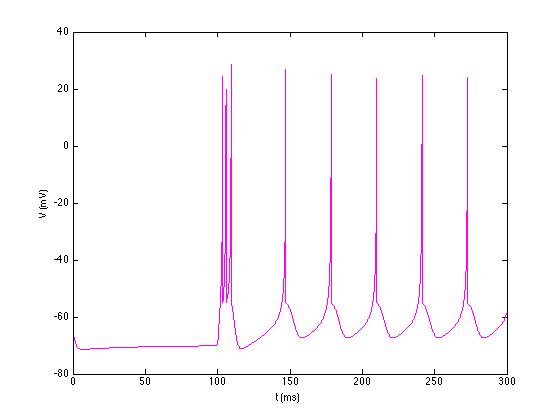
Ts variable selon les courbes

Te=0.05ms

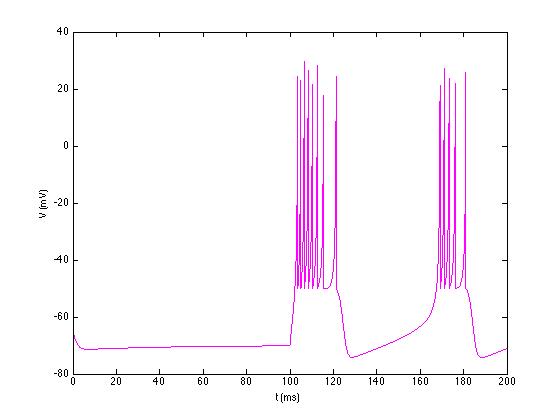
temps de début de l'échelon de I=100ms



**regular spiking**



**intrinsically bursting**



**chattering**

Programmes pour le calcul du Firing Rate

%firingRatelastchance

function FR=firingRatelastchance2(A,Te,T,Tf)

L=length(T);

FR=zeros(1,L-floor(Tf/Te));

for n=1:L-floor(Tf/Te)

q=0;

for i=n:n+floor(Tf/Te)

if (A(i)==1)

q=q+1;

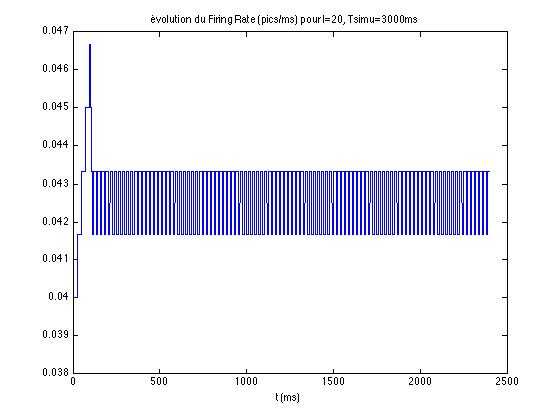
end

end

FR(n)=q/(Tf);

end;

courbe de FR=f(t) pour Tsimulation = 3000ms, Te=0.05ms, et fenêtre: 500ms, (résultat pour regular spiking)



%Obtention de plusieurs sigmoïdes

B=[100:100:600];

initneuro;

L=length(B);

for k=1:length(B)

VI=(0:2:100);

VFR=eye(1,length(VI));

for i = 1 : length(VI)

I=i;

T=[0:Te:(N-1)\*Te];

L=length(T);

Tf=B(k);

[V,A]=lastchancefaux(a,b,c,d,i,100,Te,Ts);

FR=firingRatelastchance2(A,Te,T,Tf);

T1=[0:Te:((L-floor(Tf/Te))-1)\*Te];

VFR(i)=max(FR);

end

plot(VI,VFR,'Color',[1 1-1/sqrt(k) 1/sqrt(k)])

hold on

end

Résultats pour le modèle de Regular Spinking en fonction de la taille de la fenêtre (51 points pour chaque courbe)