

5-2014

# How Imbalanced Excitation and Inhibition May Cause Low Entropy Brain Dynamics

Qusay Redwan Alfaori  
*University of Arkansas, Fayetteville*

Follow this and additional works at: <http://scholarworks.uark.edu/bmeguht>

---

## Recommended Citation

Alfaori, Qusay Redwan, "How Imbalanced Excitation and Inhibition May Cause Low Entropy Brain Dynamics" (2014). *Biomedical Engineering Undergraduate Honors Theses*. 8.  
<http://scholarworks.uark.edu/bmeguht/8>

This Thesis is brought to you for free and open access by the Biomedical Engineering at ScholarWorks@UARK. It has been accepted for inclusion in Biomedical Engineering Undergraduate Honors Theses by an authorized administrator of ScholarWorks@UARK. For more information, please contact [scholar@uark.edu](mailto:scholar@uark.edu).

This thesis is approved.

Thesis Advisor:

Woodrow Shew  
for [unclear]

Thesis Committee:

[Signature]

[Signature]

How Imbalanced Excitation and Inhibition May Cause Low Entropy Brain  
Dynamics

An Undergraduate Honors College Thesis

in the

Department of Biomedical Engineering  
College of Engineering  
University of Arkansas  
Fayetteville, AR

by

QUSAY REDWAN AWWAD ALFAORI

# Table of Contents

Introduction: .....3  
The Computer Model: .....6  
Summary and Conclusions: .....12  
Appendix A: Matlab Computer Model of Brain with Changing Inhibition .....13  
Works Cited .....23

## Figures

**FIGURE 1** – .....8  
**FIGURE 2** – .....9  
**FIGURE 3** – .....9  
**FIGURE 4** – .....10  
**FIGURE 5** – .....11

## Title: How Imbalanced Excitation and Inhibition May Cause Low Entropy Brain Dynamics

### Introduction:

In the United States of America, there has been an increase in the prevalence of Autism. Its prevalence was about 1 in 150 in the year 2000, but now it is roughly 1 in 88 [1]. Autism is a severe neurobehavioral syndrome caused by genetic disorders [2]. Autism is more common among boys. It affects 4 times as many boys as girls [2]. Autism causes disruption in behavior, emotion, and cognition. Autism is defined and diagnosed based on three behavioral characteristics: impaired language, abnormal social interactions, and restricted/repetitive behavior [3, 4]. In this project, we hypothesize that Autism might be associated with low entropy neural network dynamics. The basic logic underlying this hypothesis is as follows. First, behavior is generated by neural network dynamics. Therefore, restricted/repetitive autistic behavior is expected to be generated by restricted/repetitive neural network dynamics. By the definition of entropy, restricted/repetitive neural network dynamics have low entropy compared to less restricted, more diverse neural network dynamics.

Another important reason for our hypothesis stems from evidence that autism may be caused by an imbalance between two types of neurons in the brain: excitatory and inhibitory. Previous work has shown that such an imbalance can cause low entropy neural network dynamics as well [8]. Therefore, the mechanism which causes low entropy in autistic brain circuits may be imbalanced excitation and inhibition.

Before discussing evidence for such an imbalance in autism, I will first provide more background on what it means for excitation and inhibition to be balanced. Neurons are the basic units of neural networks and they transmit signals to each other via using neurotransmitters,

which are chemical messengers. The basic structure of neuron is composed of a cell body, dendrites, and an axon. The cell body contains the nucleus and receives the chemical message via input to the dendrites. Then the cell body converts the chemical message into an electrical signal. As a result, potential change in the cell's membrane potential will be generated. If the membrane potential goes above the threshold, then the neuron generates an action potential, which is an electrical signal that propagates down the axon and reaches the axon terminal. Then the neurotransmitters at the axon terminal will be released to the synapse. The synapse is the connection point between each neuron. Neurons are either excitatory or inhibitory. Both types generate action potentials, but the main difference between them is the neurotransmitter that each type uses. Excitatory neurons excite other neurons, i.e. make them more likely to fire an action potential, by releasing the neurotransmitter Glutamate, but inhibitory neurons inhibit or prevent firing of other neurons by releasing the neurotransmitter Gamma-Aminobutyric acid, also known as GABA. Within the brain there must exist a balance between excitatory and inhibitory neurons. In a normal cerebral cortex, about 80% of the neurons are excitatory and 20% of the neurons are inhibitory [2]. An increase in excitation over inhibition can lead to a hyper-excitable state, which causes epilepsy [2]. Thus, the balance of excitation and inhibition signals is important for normal functioning of the brain [6].

Next I will review theories and experimental evidence that autism is linked with imbalanced excitation and inhibition. Autistic individuals have abnormal perceptions due to defects in cortical networks [4]. One theory of autism suggests that hyper-functioning in the brain occurs when neuronal networks in a certain region of the brain process and store information excessively [5]. As a result, hyper-perception, hyper-attention, and hyper-memory are caused by

the defect in those certain brain regions [5]. Hyper-excitability can be the result of excitation inhibition imbalance [5].

The lack of balance between those two types of neurons can cause the brain to malfunction, such as inability to properly control the body or negatively affect cognition development, which autistic individuals suffer from. Excitation/inhibition circuit dynamics have an effect on critical stages of brain development [3]. Abnormal synapse, either weak or strong, can affect the balance of excitation/inhibition ratio [7]. Abnormal synapse strength can be caused by protein deficiencies or mutations which results in weak synapse input [7]. Reduced excitation, due to weak input at the synapse, may result in an increase in inhibition. As a result, an imbalance between excitation and inhibition will form [7]. Together, these previous studies suggest that an imbalance of excitation and inhibition may be an underlying cause of autism.

The goal of this project is to determine how brain entropy changes due to changes in excitation/inhibition balance in the brain using a computer model. Entropy in this context quantifies how many different patterns of activated neurons the brain can create. The computer model used in this project is more biologically realistic than previous models used in projects that are similar to this project. The goals of this project were the following:

1. Observe and determine the relationship between entropy and the number of observed neurons. This is important, because in experiments one can never measure the activity of all neurons in a network. We varied the size of the subset of neurons and quantified how entropy changes.
2. Determine the relationship between entropy and the percent of inhibitory neurons (I) in the network. In healthy mammals this number is about  $I=20\%$ , but in some autism

animal models, excess inhibitory neurons were found. We varied I from 0 to 40% and measured resulting changes in entropy.

3. Determine the relationship between the strength of inhibitory connections ( $I_m$ ) and entropy. Here, we kept the percentage I fixed and multiplied the inhibitory connections by a number between 0 and 2 and measured changes in entropy.

### The Computer Model:

The computer model in this project is composed of several parts that work together to function as a simple brain model. This brain model is not intended to include the complexity of an actual human being brain. The goal of this computer model is to provide insight on how changing the strength or numbers of inhibitory neurons changes entropy of neural network dynamics. This computer model code can be found in Appendix A.

We created our simple network model using Matlab. We set the total number of neurons, N, in our model equal to 500. Those 500 neurons may be considered to represent a small network within the brain, which contains approximately 10 billion neurons. In reality, neurons generate action potentials. As described previously, action potentials propagate down the axon of a neuron until it reaches the synapse and cause neurons to generate electrochemical signals, known as neurotransmitters. In this computer model, a neuron will be either active (analogous to firing an action potential) or in-active (analogous to a quiescent neuron), where an active neuron is assigned a value of 1 and an inactive neuron is assigned a value of 0. Within the neural network, the connection between neurons vary in strength, and neuron firing is determined by these connection strengths. For example, if the strength of connection between neuron A and neuron B is high and If neuron A generates an action potential, then neuron B will most likely generate an



action potential due to that high connection strength. In this computer model, we created a connection matrix,  $C$ , that represents the connection strengths among neurons. In matrix  $C$ , the connection between neuron A and neuron B is  $C_{AB}$ . We create the matrix by filling the whole matrix with zeros, i.e. no connections are present initially. Then, we modify 5% of the connections' strength to be non-zero. The connection strength is chosen randomly using a uniform distribution. To model inhibitory connections in our model, we set 20% of matrix  $C$  columns to be negative values. The last step to creating the connection matrix is to enforce the eigenvalue of the connection matrix. We set the largest eigenvalue in the matrix equal to 1. The reason we set it to 1 is that we want to prevent a growing or decaying number of active neurons and enforce a balanced excitation inhibition. This is the baseline connection matrix before changing any inhibitory connection strengths, i.e. before changing  $I_m$ . Next, we use the equation below in order to model the dynamics of our neural network.

$$s_i(t + 1) = \theta[\sum_{j=1}^N C_{ij}s_j(t) - \zeta_i(t)], \quad (1)$$

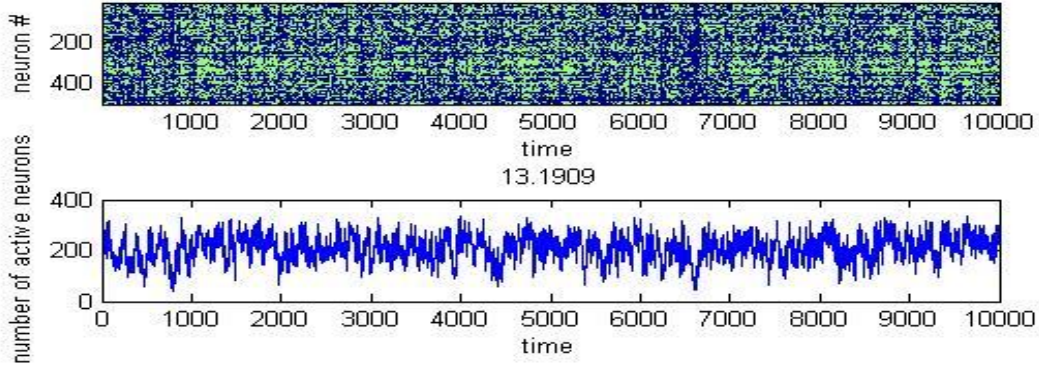
Where  $s_i(t)$  is the state of neuron  $i$  at time  $t$ ,  $\Theta[x]$  is a Heaviside step function ( $\theta[x] = 0$  for  $x \leq 0$ ,  $\theta[x] = 1$  for  $x > 0$ ),  $\zeta$  is a random number between 0 and 1 drawn every time a neuron is updated, and  $N$  is the total number of neurons in our system (500). The neural activity at time  $t+1$  is determined by the summation of the neural input at time  $t$ . real synapses unpredictably fail sometimes and that is the reason we use  $\zeta$  in this model.  $\zeta$ , is a random value, will represent the unpredictable nature of real synapses in our model.

Next, we want to determine the effect of the strength of inhibitory neurons on entropy using the model. In our model, we modify, increase or decrease, the strength of inhibition by multiplying all the inhibitory connections in the connection matrix,  $C$ , by a constant  $I_m$ .  $I_m$  ranges from 0 to 2,

where  $I_m = 1$  is considered to be a normal inhibition.  $I_m < 1$  represents reduced inhibition, and  $I_m > 1$  represents increased inhibition, relative to the balanced condition.

The activity of the neural network for  $I_m = 1$  and  $I = 20\%$  is shown in figure 1 below.

**FIGURE 1 – Activity of Neurons and Entropy.**



Shown is the number of active neurons vs. time, and the calculated entropy for the network series. Activity  $S(t)$ , and entropy  $H = 13.1909$  for the condition of  $I_m = 1$  modulated inhibition.

As mentioned previously, entropy quantifies how many different patterns of activated neurons the brain can create. The entropy of the brain can be calculated using the equation below.

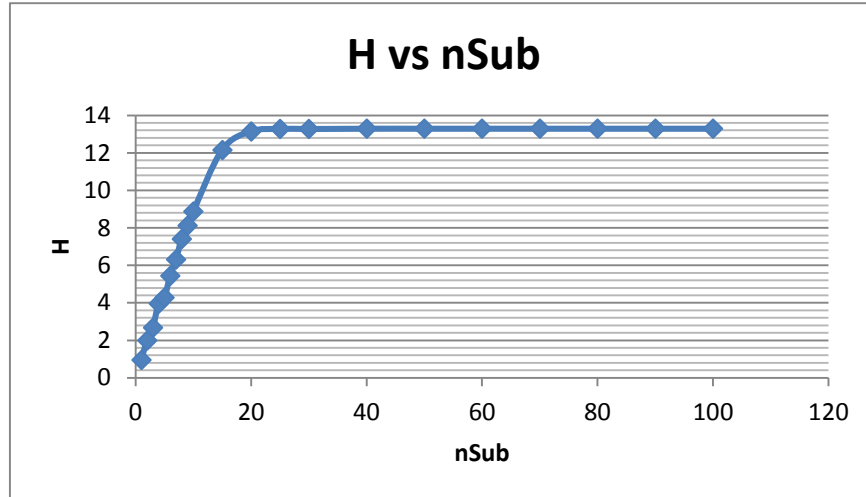
$$H = -\left[\sum_{j=1}^m P_i \log_2(P_i)\right] \quad (2)$$

Where,  $m$  = the number of unique patterns of activity and  $P_i$  = the probability of pattern  $i$ .

Before we start adjusting the strength of inhibition within the neural network, we decided to determine the number of subset of neurons that will result in the largest possible value of entropy. As shown in figure 2 below, the entropy maximizes at a subset of neurons that contains 25 neurons or more. This is because more than about 25 neurons every pattern is unique and entropy cannot exceed the value for all unique patterns. Therefore, for the remainder of this study,

we computed entropy from a subset of 10 neurons to avoid the regime with all unique patterns where entropy cannot change.

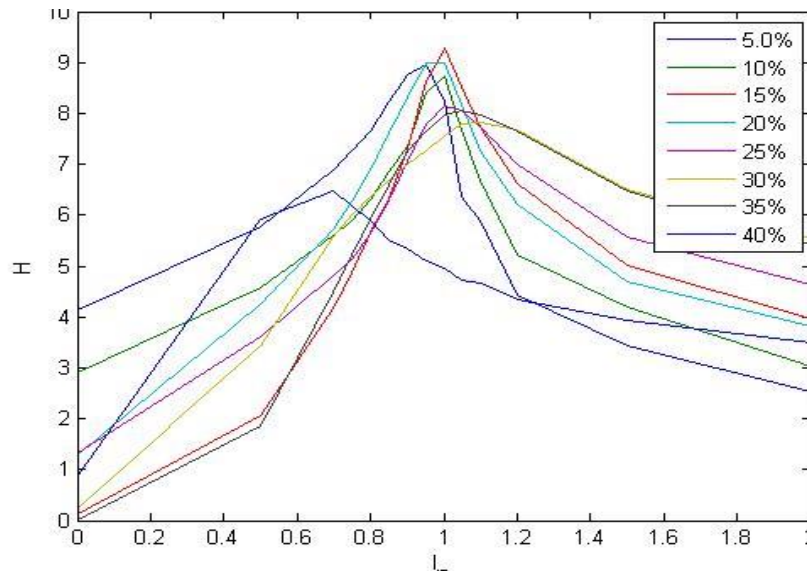
**FIGURE 2 – Effects of the number of subset of neurons on overall neural Entropy**



Shown is the number of subset of neurons vs. Entropy.

Next, we decided to observe the effect of percent of inhibitory neurons in the brain on entropy. The figure below shows the effect of different percent of inhibitory neurons, ranging from 5% to 40%, on entropy of the brain at  $I_m$  values ranging from 0 to 2.

**FIGURE 3 – Effects of changing percent of inhibitory neurons on Entropy at values of  $I_m$  ranging from 0 to 2.**

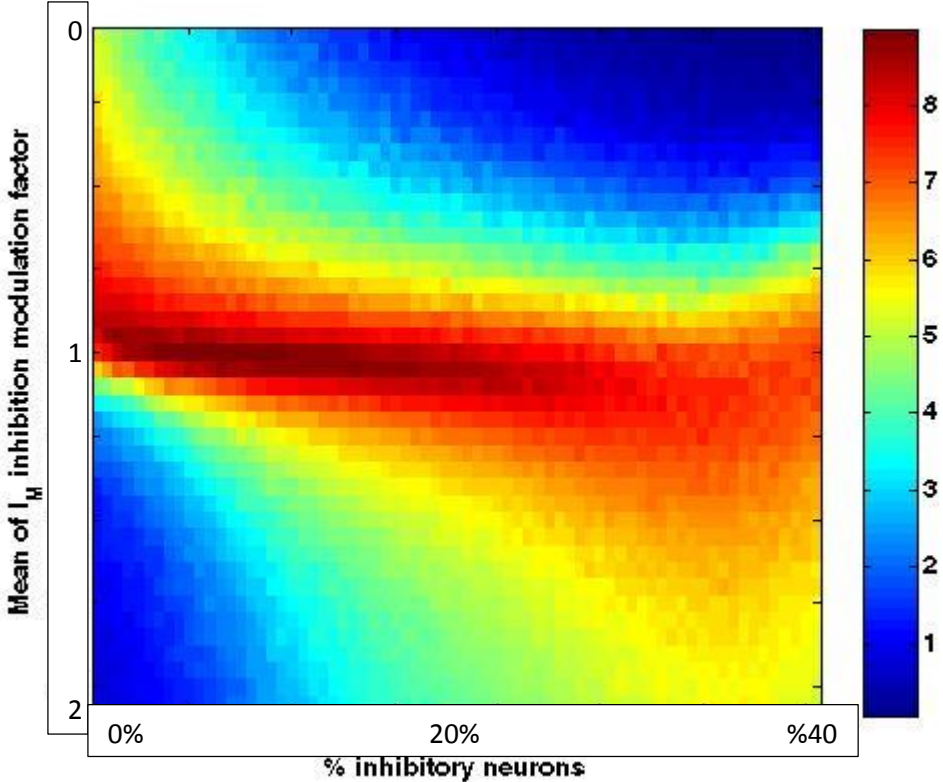


Shown is the percent of inhibitory neurons vs. calculated Entropy at various  $I_m$  values ranging from 0 to 2. Each curve represent  $I$  values ranging from 5% to 40% (as shown in the percentage box in the figure).

From figure 3 above, we concluded that brains with 25 percent or more inhibitor neurons will have lower entropy. Also, the figure shows that the ideal percentage of inhibitory neurons in the brain is approximately 20 percent, which is proven by other research.

Next, we decided to observe how consistent or variable brain entropy is across a single individual. The figure below shows average entropy vs  $I_m$  and  $I$  for many repetitions over 100 different brains.

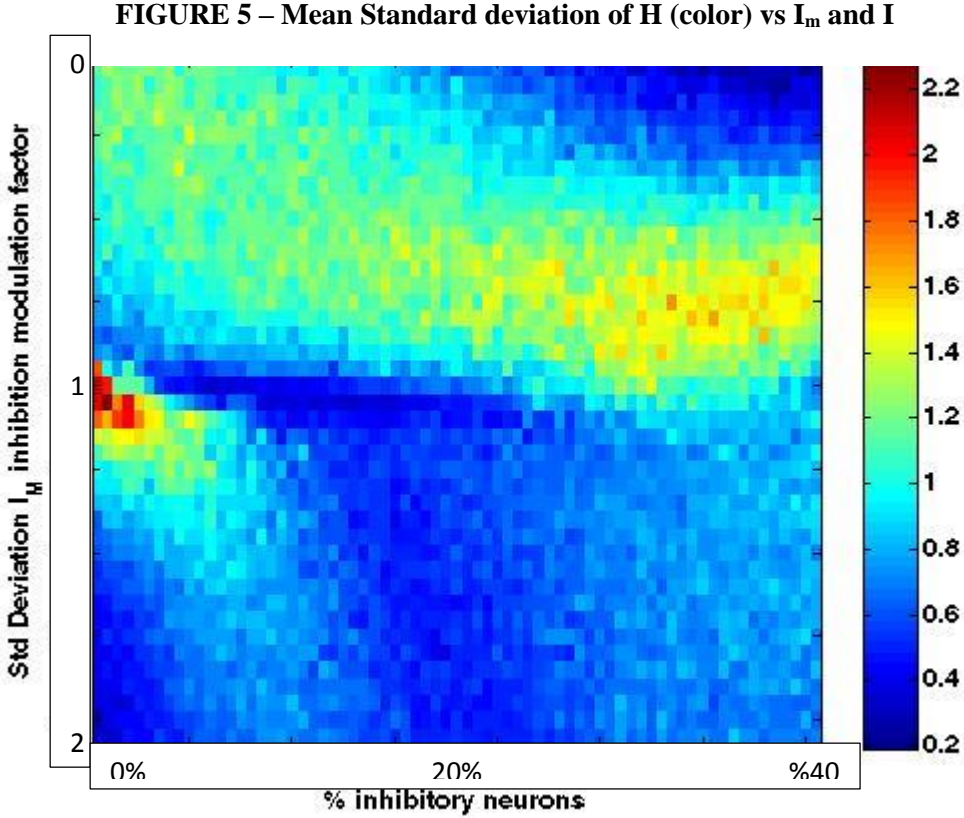
**FIGURE 4 – Average H (color) vs  $I_m$  and  $I$**



The numbers on the axes do not correspond to  $I_m$  and % $I$ . Rather, the horizontal axis ranges from  $I=0%$  to 40% in 72 steps. The vertical axis ranges from  $I_m=0$  to 2 in 43 steps.

Figure 4 above shows that the mean entropy over a 100 different brains is highest at approximately  $I=20%$  and  $I_m=1$ .

Figure 5 below shows the standard deviation of entropy vs.  $I$  and  $I_m$  for many repetitions of 100 different brains.



The numbers on the axes do not correspond to  $I_m$  and % $I$ . Rather, the horizontal axis ranges from  $I=0\%$  to  $40\%$  in 72 steps. The vertical axis ranges from  $I_m=0$  to 2 in 43 steps.

Figure 4 above shows that the mean standard deviation of entropy over a 100 different brains is lowest at approximately  $I=20\%$  and  $I_m=1$ .

## Summary and Conclusions:

In conclusion, we used our simple neural network model to observe and study the effects of varying inhibition and the strength of inhibitory connections. In similarity with what we expected, we found that entropy is highest under balanced neural network conditions which consist of approximately 80% excitatory neurons and a 20% inhibitory neurons at a strength of inhibitory connections with a value that is equal to 1, according to our model. In the context of entropy, high entropy implies the ability to store more information within the neural network, normal development of the brain, and a good ability to perceive a good range of stimuli from the surroundings. Considering high entropy to be in favor of animal survival, we meditate that evolution plays a role in the survival of those animals. Evolutionary pressures or Natural selection may result in brains with high entropy and balanced neural network. Finally, we predict if entropy is advantageous to us, then our results indicate that high entropy that a narrow range of  $I=20\%$  and  $I_m=1$  both express high entropy. Also, our low standard deviation plot for mean entropy implies high reliability for high entropy at  $I=20\%$  and  $I_m=1$ . Also, when considering Autism and its symptoms, our results show that an Imbalance neural network causes low entropy and unreliable brain. Thus, we predict that low entropy is one of the causes to the disorder.

## Appendix A: Matlab Computer Model of Brain with Changing Inhibition

### Matlab Computer Model Version 1:

```
%basic E/I model
clear

plotflag=1; %set to 1 to show the results in a plot (very slow, but
informative), otherwise set to 0

N=500; %total number of neurons
T=10000; %duration of simulation

%%%%%%%%% make the connection matrix %%%%%%%%%%%
C=rand(N);
imask=rand(1,N)<=0.2; %define 20% inhibitory neurons
C(:,imask)=-1*C(:,imask); %set outgoing connection from inhibitory neurons
to be negative
k=0.05; %mean degree N*k
C(rand(N)>k)=0; %set mean degree

% get largest eigenvalue
eigvec=ones(N,1);
NN=1000;
eigval=zeros(1,NN);
converg=zeros(1,NN);
for i=1:NN;
    eigvec2=C*eigvec;
    eigval(i)=mean(eigvec2./eigvec);
    converg(i)=sum(abs(eigvec2-eigvec*eigval(i)));
    eigvec=eigvec2./sum(abs(eigvec2))*N;
end

%enforce largest eigenvalue = 1
C=C/eigval(end);
%%%%%%%%% done making the connection matrix %%%%%%%%%%%

%IfacList=1;
IfacList=[0.5 0.7 0.9 0.95 1 1.05 1.10 1.2];
%IfacList=[0 0.5 0.7:0.05:1.1 1.2 1.5 2];
nifac=length(IfacList);

for i=1:nifac

    B=C;
    B(:,imask)=IfacList(i)*B(:,imask);

    nREP=10;
    H=zeros(1,nREP);
    for R=1:nREP
```

```

nev=false(N,T); %initialize matrix for storing activity of all
neurons during one simulation

%%%%%%%% compute the activity of the network %%%%%%%%%

%initial condition: activate Ni neurons in first timestep
nev(1:N/2,1)=1;

%evolve dynamics: probabilistic spike propagation
t=1;
while t<T %stop computing if we reach T steps

    nev(:, t+1) = B*nev(:,t)>rand(N,1); %determine which neurons fire
in the next time step

    %if the activity dies out, restart it
    if sum(nev(:,t+1))==0
        [~,rind]=sort(rand(1,N));
        nev(rind(1),t+1)=1;
        restart(t+1)=1;
        disp('restart')
    end

    t=t+1;
end
%%%%%%%% done computing the activity of the network %%%%%%%%%

sumn=sum(nev,1);

%compute entropy estimate H from a subset of neurons
nsub=20;
subset=nev(1:nsub,:);
[~,~,j]=unique(subset,'rows');
npat=max(j);
p=histc(j,0.5:npat+0.5)/T; p(end)=[];
H(R)=-sum(p.*log2(p));

%display results if plotflag is set to 1
if plotflag
    %show basic activity versus time
    figure(1)
    title('Activity vs Time')
    subplot(311)
    imagesc(nev,[0 2])
    ylabel('neuron #')
    xlabel('time')

    subplot(312)
    plot(sumn)
    ylabel('number of active neurons')
    xlabel('time')

    title(H(R))
end

```



```

        pause
    end

end

end

```

## Matlab Computer Model Version 2:

```

%basic E/I model
clear

plotflag=0; %set to 1 to show the results in a plot (very slow, but
informative), otherwise set to 0

N=500; %total number of neurons
T=10000; %duration of simulation

%%%%%%%%% make the connection matrix %%%%%%%%%%%
C=rand(N);
imask=rand(1,N)<=0.2; %define 20% inhibitory neurons
C(:,imask)=-1*C(:,imask); %set outgoing connection from inhibitory neurons
to be negative
k=0.05; %mean degree N*k
C(rand(N)>k)=0; %set mean degree

% get largest eigenvalue
eigvec=ones(N,1);
NN=1000;
eigval=zeros(1,NN);
converg=zeros(1,NN);
for i=1:NN;
    eigvec2=C*eigvec;
    eigval(i)=mean(eigvec2./eigvec);
    converg(i)=sum(abs(eigvec2-eigvec*eigval(i)));
    eigvec=eigvec2./sum(abs(eigvec2))*N;
end

%enforce largest eigenvalue = 1
C=C/eigval(end);
%%%%%%%%% done making the connection matrix %%%%%%%%%%%

IfacList=[0 0.5 0.7:0.05:1.1 1.2 1.5 2];
nifac=length(IfacList);

nREP=10; % with our model so far # of repetitions did not affect the out come
of H,
H=zeros(nifac,nREP);

for i=1:nifac

```

```

B=C;
B(:,imask)=IfacList(i)*B(:,imask);

for R=1:nREP
    disp(['rep=',num2str(R),'..Im=',num2str(i)])

    nev=false(N,T); %initialize matrix for storing activity of all
neurons during one simulation

    %%%%%%%%% compute the activity of the network %%%%%%%%%

    %initial condition: activate Ni neurons in first timestep
nev(1:N/2,1)=1;

    %evolve dynamics: probabilistic spike propagation
t=1;
while t<T %stop computing if we reach T steps

    nev(:, t+1) = B*nev(:,t)>rand(N,1); %determine which neurons fire
in the next time step

    %if the activity dies out, restart it
if sum(nev(:,t+1))==0
    [~,rind]=sort(rand(1,N));
    nev(rind(1),t+1)=1;
    restart(t+1)=1;
    disp('restart')
end

    t=t+1;
end
%%%%%%%% done computing the activity of the network %%%%%%%%%

sumn=sum(nev,1);

%compute entropy estimate H from a subset of neurons
nsub=10;
subset=nev(1:nsub,:);
[~,~,j]=unique(subset,'rows');
npat=max(j);
p=histc(j,0.5:npat+0.5)/T; p(end)=[];
H(i,R)=-sum(p.*log2(p));

%display results if plotflag is set to 1
if plotflag
    %show basic activity versus time
figure(1)
subplot(311)
imagesc(nev,[0 2])
ylabel('neuron #')
xlabel('time')

    subplot(312)
plot(sumn)

```

```

        ylabel('number of active neurons')
        xlabel('time')

        title(H(i,R))

        %pause
    end

end

end

```

```

%%
figure(2)
plot(IfacList,H)

```

### Matlab Computer Model Version 3:

```

%basic E/I model
clear

plotflag=0; %set to 1 to show the results in a plot (very slow, but
informative), otherwise set to 0

N=500; %total number of neurons
T=10000; %duration of simulation

%%%%%%%%% make the connection matrix %%%%%%%%%%%

brain=[ .05:.005:.25]

b=length(brain)

IfacList=[0 0.5 0.7:0.05:1.1 1.2 1.5 2];
nifac=length(IfacList);

% define H here
H=zeros(nifac,b)
%% start loop here
for e=1:b

C=rand(N);
imask=rand(1,N)<=brain(e); %define % of inhibitory neurons
C(:,imask)=-1*C(:,imask); %set outgoing connection from inhibitory neurons
to be negative
k=0.05; %mean degree N*k
C(rand(N)>k)=0; %set mean degree

% get largest eigenvalue
eigvec=ones(N,1);
NN=1000;

```

```

eigval=zeros(1,NN);
converg=zeros(1,NN);
for i=1:NN;
    eigvec2=C*eigvec;
    eigval(i)=mean(eigvec2./eigvec);
    converg(i)=sum(abs(eigvec2-eigvec*eigval(i)));
    eigvec=eigvec2./sum(abs(eigvec2))*N;
end

%enforce largest eigenvalue = 1
C=C/eigval(end);
%%%%%%%% done making the connection matrix %%%%%%%%%

IfacList=[0 0.5 0.7:0.05:1.1 1.2 1.5 2];
nifac=length(IfacList);

% change nrep to 1
nREP=1;

for i=1:nifac

    B=C;
    B(:,imask)=IfacList(i)*B(:,imask);

    for R=1:nREP
        disp(['brain=',num2str(e),'..Im=',num2str(i)])

        nev=false(N,T); %initialize matrix for storing activity of all
neurons during one simulation

        %%%%%%%%% compute the activity of the network %%%%%%%%%

        %initial condition: activate Ni neurons in first timestep
nev(1:N/2,1)=1;

        %evolve dynamics: probabilistic spike propagation
t=1;
while t<T %stop computing if we reach T steps

            nev(:, t+1) = B*nev(:,t)>rand(N,1); %determine which neurons fire
in the next time step

            %if the activity dies out, restart it
if sum(nev(:,t+1))==0
                [~,rind]=sort(rand(1,N));
                nev(rind(1),t+1)=1;
                restart(t+1)=1;
                %disp('restart')
            end

            t=t+1;
        end
        %%%%%%%%% done computing the activity of the network %%%%%%%%%
    end
end

```

```

sumn=sum(nev,1);

%compute entropy estimate H from a subset of neurons
nsub=10;
subset=nev(1:nsub,:);
[~,~,j]=unique(subset,'rows');
npat=max(j);
p=histc(j,0.5:npat+0.5)/T; p(end)=[];
H(i,e)=-sum(p.*log2(p));

%display results if plotflag is set to 1
if plotflag
    %show basic activity versus time
    figure(1)
    subplot(311)
    imagesc(nev,[0 2])
    ylabel('neuron #')
    xlabel('time')

    subplot(312)
    plot(sumn)
    ylabel('number of active neurons')
    xlabel('time')

    title(H(i,R))

    %pause
end

end

end

end

%% end it here
%%
figure(2)
plot(IfacList,H)
xlabel('I_m')
ylabel('H')
legend('5.0%','10%','15%','20%','25%','30%','35%','40%')

figure(3)
imagesc(H)
xlabel('% inhibitory neurons')
ylabel('I_M inhibition modulation factor')
colorbar

```

## Matlab Computer Model Version 4:

```
%basic E/I model
clear

plotflag=0; %set to 1 to show the results in a plot (very slow, but
informative), otherwise set to 0

N=500; %total number of neurons
T=5000; %duration of simulation

alphalist=[.05:.005:.40]; % Originally was [0.05:0.01:0.25]
b=length(alphalist);

IfacList=[0:0.05:2];%[0 0.5 0.7:0.05:1.1 1.2 1.5 2];
nifac=length(IfacList);

nREP=100;

sponr=1/N/100;

k=0.05; %mean degree N*k

H=zeros(nifac,b,nREP);

for R=1:nREP

    for alpha=1:b

        disp(['rep=',num2str(R), ' alpha=',num2str(alpha)])
        for i=1:nifac

            %make a new brain for every rep, every parameter set
            B=rand(N);
            imask=rand(1,N)<=alphalist(alpha); %define % of inhibitory
            neurons
            B(:,imask)=-1*B(:,imask); %set outgoing connection from
            inhibitory neurons to be negative
            B(rand(N)>k)=0; %set mean degree
            B=B/max(abs(eig(B))); %enforce largest eigenvalue = 1
            B(:,imask)=IfacList(i)*B(:,imask);

            nev=false(N,T); %initialize matrix for storing activity of all
            neurons during one simulation

            %%%%%%%%% compute the activity of the network %%%%%%%%%

            %initial condition: activate Ni neurons in first timestep
            nev(1:N/2,1)=1;

            %evolve dynamics: probabilistic spike propagation
            t=1;
            while t<T %stop computing if we reach T steps
```

```

nev(:, t+1) = B*nev(:,t)>rand(N,1); %determine which neurons
fire in the next time step

%random activation at rate of one spike among all neurons
every 100 timesteps
nev(rand(N,1)<sponr,t+1)=1;

t=t+1;
end
%%%%%% done computing the activity of the network %%%%%%%%%%

sumn=sum(nev,1);

%compute entropy estimate H from a subset of neurons
nsub=10;
subset=nev(1:nsub,:);
[~,~,j]=unique(subset,'rows');
npat=max(j);
p=histc(j,0.5:npat+0.5)/T; p(end)=[];
H(i,alpha,R)=-sum(p.*log2(p)); %%%check this for 3D

%display results if plotflag is set to 1
if plotflag
    %show basic activity versus time
    figure(1)
    subplot(311)
    imagesc(nev,[0 2])
    ylabel('neuron #')
    xlabel('time')

    subplot(312)
    plot(sumn)
    ylabel('number of active neurons')
    xlabel('time')

    title(H(i,alpha,R)) %%%%%%%%%%%%%%% ask about this one if
H(i,e,R) is correct?

    %pause
end
end
end
end
save('Hmany','H')

%%
mH=mean(H,3);
stdH=std(H,0,3);

figure(3)
imagesc(mH)
xlabel('% inhibitory neurons')

```

```
ylabel('Mean of I_M inhibition modulation factor')
colorbar

figure(4)
imagesc(stdH)
xlabel('% inhibitory neurons')
ylabel(' Std Deviation I_M inhibition modulation factor')
colorbar
```



## Works Cited

1. Hook, E. B., & Lindsjö, A. (1978). Down syndrome in live births by single year maternal age interval in Swedish study: comparison with results from a New York State study. *The American Journal of Human Genetics*, 30(1): 19–27.
2. Fernandez, F., & Garner, C. C. (2007). Over-inhibition: a model for developmental intellectual disability. *Trends in neurosciences*, 30(10), 497-503.
3. Fernandez, F., Morishita, W., Zuniga, E., Nguyen, J., Blank, M., Malenka, R. C., & Garner, C. C. (2007). Pharmacotherapy for cognitive impairment in a mouse model of Down syndrome. *Nature neuroscience*, 10(4), 411-3.
4. Wehr, M., & Zador, A. M. (2003). Balanced inhibition underlies tuning and sharpens spike timing in auditory cortex. *Nature*, 426(6965), 442-6.
5. Kinouchi, O., & Copelli, M. (2006). Optimal dynamical range of excitable networks at criticality. *Nature Physics*, 2(5), 348-351.
6. Larremore, D. B., Shew, W. L., & Restrepo, J. G. (2011). Predicting Criticality and Dynamic Range in Complex Networks: Effects of Topology. *Physical Review Letters*, 106(5), 1-4.
7. Shew, W. L., Yang, H., Petermann, T., Roy, R., & Plenz, D. (2009). Neuronal Avalanches Imply Maximum Dynamic Range in Cortical Networks at Criticality. *Journal of Neuroscience*, 29(49), 15595-15600.