Notes on Networks Impacted by Axonal Swelling

Brian Frost-LaPlante

Summer 2018

Abstract

We explore the effects of inhibition, feedback and network size on networks of integrate-and-fire neurons with damaged axons. The damage is implemented through a basic statistical algorithm, and is based on a physical model for the transmission of electrical signals through swollen axons with continuous swelling. We find that inhibition serves to make networks more resilient to damage of this type, while feedback can make networks either more sensitive or more robust to this damage depending on the type of neuron it originates from. The effects on larger networks have not yet been studied.

Introduction

This document details the simulation of damaged integrate-and-fire networks, most of which contain either two or three cells. The damage applied to the networks is a special case of an axonal swelling model presented by Maia and Kutz (2014), in which spiketrains are nonlinearly filtered upon transmission through an axon. The development of the simulation method is presented, followed by a comprehensive set of comparisons of damaged and undamaged two-cell networks' responses to Poisson input trains. A selection of three-cell network results are then presented in a similar fashion, followed by a small number of larger network simulation results.

Damage Model

The axonal swelling model presented by Maia and Kutz is given by altering the axon diameter in an active cable model of the axon. The active cable is described by a system of partial differential equations which determine the evolution of a voltage variable (V) and several recovery variables (here, a single variable R) in continuous space and time. This system is as follows:

$$\frac{\partial V}{\partial x} = \frac{D}{a(x)} \frac{\partial}{\partial x} \left(\frac{a^2(x)}{r_L(x)} \frac{\partial V}{\partial x} \right) + V(V-a)(V-1) - R \tag{1}$$

$$\frac{\partial R}{\partial t} = bV - cR \tag{2}$$

Here a, b, c and D are physical constants, r_L is the axonal resistance and $a(x)$ is the axonal diameter. We take r_L to be constant in these simulations, and we implement damage by varying $a(x)$.

We consider a continuous swelling of the axon diameter. This is to say that the axon diameter changes from an initial δ_B to a final, larger δ_A over a distance of δ_T (all in units cm) as described by the following continuous growth:

$$a(x) = \begin{cases} \delta_B, & \text{for } x \leq 0 \\ \tilde{a}(x), & \text{for } 0 \leq x \leq \delta_T \\ \delta_A, & \text{for } x \geq \delta_T \end{cases} \tag{3}$$

where

$$\tilde{a}(x) = \delta_B + (\delta_A - \delta_B)\left(10\frac{x^3}{\delta_T^3} - 15\frac{x^4}{\delta_T^4} + 6\frac{x^5}{\delta_T^5}\right)$$

The system of PDEs with this axonal diameter is solved using the pseudo-spectral method, in which each PDE is converted to a system of ODEs in the Fourier coefficients, and the nonlinearities are re-computed at each timestep. This method is described in more detail in the Appendix.

Effects of Damage on Spiketrains

In the paper by Maia and Kutz in which this damage model was presented, it was noted that single spikes propagating through the axon could be affected in one of three ways: they could be blocked from transmitting through the enlarged axon, they could be reflected so as to transmit in both the positive and negative x direction, and they could simply transmit as in an undamaged axon. Parameter ranges for these effects were studied in some detail for single spikes in this paper, but the situation in which several spikes are sent in sequence was not detailed. We thereby looked to characterize the impact of a damaged axon on spiketrains in correspondence with these parameter regimes.

When in the parameter space corresponding to blockage of a single spike, any spiketrain is entirely blocked, as expected. In the case of reflection, spikes which are sufficiently close in time will collide with reflected spikes, and as such, about one half of all spikes are properly transmitted. In the parameter regime where single spikes are tansmitted, the transmission of spikes is one of perfect or *filtered*, meaning that some (but not all) spikes are blocked from transmitting across the damaged axon. This filtering is, in part, dependent on how near spikes are to one another, but is too complex to be described entirely in this way; in some cases, spikes are deleted only if preceded by several closely spaced spikes, for example.

Interpretation as Bit-String

For the purpose of simulation, it is easiest to consider spikes as identical voltage pulses which are finite support in both time and space. Spiking neurons in reality have refractory periods, meaning that after a neuron has spiked, there is a minimum time T_R that must pass before that neuron will fire again. As such, all spikes must be at least T_R apart, and it is possible to consider a time-binning scheme of bins size T_R or less where no bin can contain more than one spike. With this time-binning, a spiketrain at some point in space x_0 can be discretely interpreted as a bitstring, in which a 1 represents that a spike was present in that time-bin and a 0 represents that no spike was present then. A program was written which generated continuous-time spiketrains according to an input bitstring at the smaller end of the damaged axon, and output the spiketrain observed at the other end as a bitstring. This interpretation of spiketrains as bitstrings is used exclusively hereafter.

Estimation Algorithm

Due to the computational complexity of solving this system of PDEs at each timestep in a network simulation, it was determined that network-level simulations must use an estimated output of the PDE solver, rather than an actual one. In the case of blockage, such an algorithm is unnecessary, as the output of the axon is always 0, In the case of reflection, the algorithm is slightly more complicated although still simple; one must simply determine how close incoming spikes must be to imply collision, and delete each second spike in such proximity at the output.

The case of transmission is significantly more challenging, as different axon damages yield very different "rules" for how spiketrains are filtered. In this regime, spikes are blocked based on their proximity to other spikes in time in some non-linear way, and it is not the case that a spike is affected only by the nearest spike before it. For example, in some cases tested experimentally, spike transmission can be blocked if preceded by 4 sufficiently-close spikes, but will not be affected by only three preceding spikes spaced in the same way. It is assumed, however, that the affect of a spike has some limited range in time, T_B. That is to say that the blockage of a spike for a given axonal damage will be determined only by spikes in the past T_B milliseconds.

If one has an estimate for T_B, the algorithm can be implemented as follows: consider all 2^{T_B} possible bitstrings of length T_B, and run the PDE solver on each of these bitstrings to obtain outputs for each such string. Now suppose you have a string which is longer than T_B bits in length, and you wish to estimate the output of the PDE solver, and use this data to create a lookup table for T_B length strings. Look at each consecutive string of T_B bits and determine via the lookup table what output this string would map to. Except for $T_B - 1$ bits at the beginning and end of the longer strings, you then have T_B guesses for what each bit will map to. Simply take the mean of these guesses and round to obtain a final guess. T_B depends on the kind of damage as well as the parameters of the active cable equation, so it must first be guessed via experimentation before implementing the algortihm.

To illustrate the power of this algorithm, we look at the example of $\delta_A = 4$mm, $\delta_B = 3$mm and $\delta_T = 0.25$mm. In this case we use $T_B = 9$ bins, and determine a lookup table of outputs from all possible 512-bit inputs. We then generate 1000 99-bit random spiketrains, where each bit is equally likely to be a 1 or 0 independently. We run the PDE solver on these 99-bit strings as well as the estimation algorithm described above, and determine that the bit-error-rate of the algorithm is on the order of a tenth of a percent. Furthermore, the estimation algorithm is 5000 times faster than the PDE solver on the hardware used, attesting to its desirability in larger network simulations.

Network Model

We look now to apply this estimator for axonal damage of a specific type to a small network of spiking neurons to see how the presence of damaged axons can affect network-level behavior. We choose an integrate-and-fire model for the spiking neuron, and look at networks containing both excitatory and inhibitory neurons. Spikes are output by neurons according to the integrate-and-fire model, and either a damaged or undamaged version of this output is seen as the input to connected cells depending on which axons are damaged. The damage model used through all of these experiments is that described above, with $\delta_A = 4$mm, $\delta_B = 3$mm and $\delta_T = 0.25$mm.

Integrate-and-Fire Neurons

The model used is a simple integrate-and-fire model in which both excitation and inhibition factor linearly into the voltage equation. In this model, each cell has three state variables: a voltage V, an excitatory synaptic current J_E and an inhibitory synaptic current J_I. At every timestep, it is checked if V is greater than a threshold voltage V_T. If it is and the cell's refractory period has passed since the last spike was fired, the voltage at that cell is reset to 0 and a spike is fired from that cell. When an inhibitory/excitatory spike is seen at the input of a cell, the corresponding J variable increases by some number w. The differential equations between spike arrivals and departures are as follows:

$$\begin{aligned}\tau_V \frac{dV}{dt} &= -V + J_E - J_I \\ \tau_J \frac{dJ_E}{dt} &= -J_E \\ \tau_J \frac{dJ_I}{dt} &= -J_I\end{aligned}$$

The network is stimulated by an excitatory global input, which is also a spiketrain, and is handled by each cell in the same manner as spikes spiketrain inputs from other cells. This global input is generated as a Poisson process with 0.1 ms timebins, and has finite duration, lasting exactly half of the duration of the simulation. For example, if a simulation's duration is 100 ms, a Poisson spiketrain formed by 500 timebins is fed as a global input for the first 50 ms, after which the network is let to run without global stimulus. Below is the update algorithm for the network simulations.

Algorithm 1 Network Loop

```
for cell in network do
    for input in cell.inputs do
        if input == 1 then
            J_E = J_E + w
        else if input == −1 then
            J_E = J_I + w
    update cell via the solution to the ODEs
    if cell.V ≥ V_T then
        cell.V = 0
        if cell.type = excitatory then
            cell.outputs = 1
        else if cell.type = inhibitory then
            cell.outputs = −1
```

There are several models for integrate-and-fire neurons, however this one was decided on because it can be easily solved explicitly between the arrival or departure of spikes. That is to say that the use of an ODE solver in software is entirely avoided in the implementation of this kind of integrate-and-fire neuron. A second model wherein the voltage equation is replaced by $\dot{V} = -V + J_E(V - V_E) + J_I(V - V_I)$, $V_I > V_T$, $V_E < 0$ was tested against the model described above. It produced results which were quantitatively slightly different, but qualitatively identical, and was slower in comparison.

Output Comparison

To evaluate the effect damage has on a network, we look at an undamaged network and a damaged network's response to the same set of N Poisson input trains and the same N random initial voltages for each cell. In these experiments, we take $N = 100$. In comparing the outputs, we care about differences in both inter-spike intervals (which are important in the context of rate encoding) and specific spike arrival time (which is important in the case of spike-time encoding). For the former, we look at a histogram of inter-spike intervals for each of the undamaged and damaged network, as well as the sample mean and variance of the inter-spike intervals. This helps to compare shifts in frequency in the output of a network due to damage. As for spike arrival time, we use a distance proposed by van Rossum which induces a metric space on the space of spiketrains. Suppose one has a spiketrain with M spikes at t_i, where i is between 1 and M. Define f as

$$f(t) = \sum_{i=1}^{M} H(t - t_i) e^{-(t-t_i)/t_c}$$

where H is the Heaviside step function and t_c is any positive real number. For a second spiketrain, suppose g is defined in a similar way. Then the distance D^2 between these two spiketrains is defined by

$$D^2(f, g) = \frac{1}{t_c} \int_0^\infty \big(f(t) - g(t)\big)^2 \, dt$$

Clearly D^2 depends on t_c, with smaller choices of t_c leading to a metric more sensitive to spike arrival time differences, and with larger t_c yielding a metric more sensitive to deletions (as seen in van Rossum's paper). As such, we choose a small t_c of 1ms to better illustrate differences in spike arrival time between damaged and undamaged networks. If two spiketrains, u and v are to be compared, we write $D^2(u, v)$ with the understanding that one must first form the functions f and g from u and v, respectively, before computing the D^2 distance as $D^2(f, g)$.

Suppose u_i is the spiketrain generated at the output of a network which is undamaged for input and initial conditions i (i is between 1 to N). For a damaged network of the same architecture, say the output measured at the same cell in response to the same input and initial conditions is v_i. In comparing the output of damaged networks to undamaged networks by means of the D^2 distance, one takes

$$D^2_{ave} = \sum_{i=1}^{N} D^2(u_i, v_i)$$

We report this as the D^2 distance for a given damage. This does assume a stationarity of the D^2 distance, however this assumption has been tested extensively and appears to be justified for the purposes of these experiments.

One-Cell Control

We start with a single cell as a control, receiving only stimulation from a global input. As the output of this cell is not fed into any other cells, it is unimportant whether this cell is excitatory or inhibitory. We observe its output, both damaged and undamaged, and compare these damages as described above. The undamaged and damaged histograms are shown below.

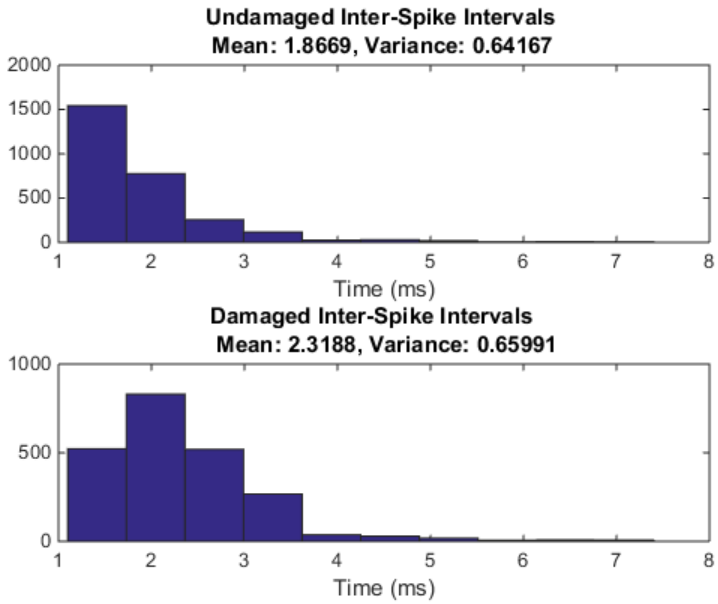

Figure 1: One-Cell Control Inter-Spike Interval Histograms

The mean D^2 distance across the 100 trials is 5.6233. The lower bound on the distribution at an inter-spike interval of 1ms is due to the 1ms refractory period of the cells. It can be seen that damage incites an increase in the mean of the distribution. This is expectable, as we know that damage deletes spikes fired at high frequencies, which corresponds to deletion of spikes which correspond to low inter-spike intervals. We will use these data as a control, to see how the magnitude of D^2 distances and inter-spike interval distributions in two-cell networks compare to those of this one-cell network. That is to say, we can see if certain two-cell networks are more or less resilient to damage than a single cell, and if so, to what extent.

Two-Cell Networks

We will look at all possible network architectures of two cells, consisting of inhibitory or excitatory cells with a global input. We will also consider all possible damages to these networks. We first describe, exhaustively, these networks and damages, and then analyze the statistics of the output for each network.

Network Architectures

The figure below shows all seven possible architectures of a two-cell network, where an 'E' represents an excitatory cell and an 'I' represents an inhibitory cell. A network can either contain two of the same type of cell or one of each type of cell. Furthermore, the connectivity can be either unidirectional or bidirectional. In networks with unidirectional coupling, only one kind of damage can occur; damage to the single axon in the network. In a network with bidirectional coupling, the number of possible damages is higher. If the two cells in the network are of the same type, one can damage either a single axon or both axons. In the network containing

bidirectional coupling between a single inhibitory neuron and a single excitatory neuron, there are two distinct ways to damage a single axon, and of course one may also damage both axons. In this case, thereby, there are three possible damages.

It should be noted that in the unidirectional case, the 'target neuron', i.e. the neuron at which the axon terminates, does not impact any cell. For this reason, it does not matter whether or not this target cell is inhibitory or excitatory. As such, we can look only at 5 architectures rather than 7.

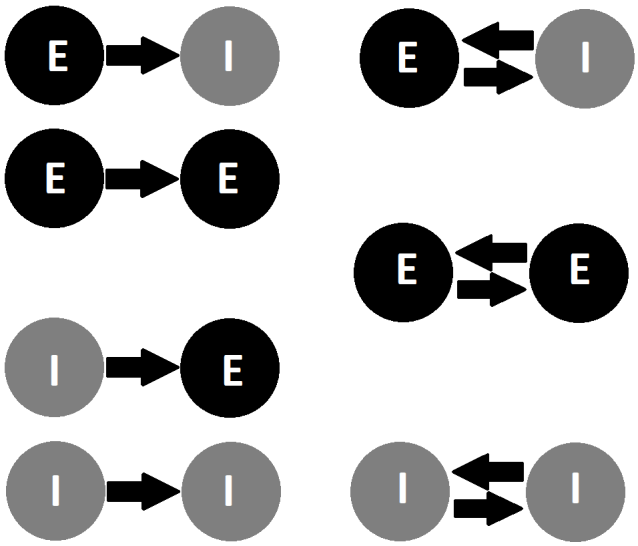

Figure 2: Two-Cell Network Architectures. The architectures on the left-hand side can be grouped into pairs, as the type of the second cell does not impact the output of the network.

Excitatory to Inhibitory with Bidirectional Coupling

In this network, three separate damages are possible. We look at the output of each of the excitatory cell and inhibitory cell with respect to each of these damages.

We look first at the excitatory neuron, whose histograms can be seen in Figure 3. The mean D^2 distance in the case in which the excitatory-to-inhibitory connection is damaged is 0.1507. This is an order of magnitude smaller than the one-cell control's D^2 distance, likely because the excitatory cell is kept from firing at a sufficiently high rate due to the inhibition of the second cell. The mean D^2 distance in the case in which the inhibitory-to-excitatory connection is damaged is 5.2018, which is quite similar to the single-cell case. Lastly the mean D^2 distance in the case in which both axons are damaged is 5.2109, slightly higher than this second case.

We look next at the inhibitory neuron, whose histograms can be seen in Figure 4. The mean D^2 distance in the case in which the excitatory-to-inhibitory connection is damaged is 0.5323. This is an order of magnitude smaller than the one-cell control's D^2 distance, likely because the excitatory cell is kept from firing at a sufficiently high rate due to the inhibition of the second cell. The mean D^2 distance in the case in which the inhibitory-to-excitatory connection is damaged is 7.5835, which is quite similar to the single-cell case. Lastly the mean D^2 distance

in the case in which both axons are damaged is 7.5089, nearly the same as in the second case.

The D^2 distances observed in this network, as well as the corresponding shifts in mean, show that this network is more resilient in all senses with respect to damage to the excitatory-to-inhibitory connection than to other kinds of damage. This is likely due to the lower firing rate of the excitatory cell, caused by inhibition within the network.

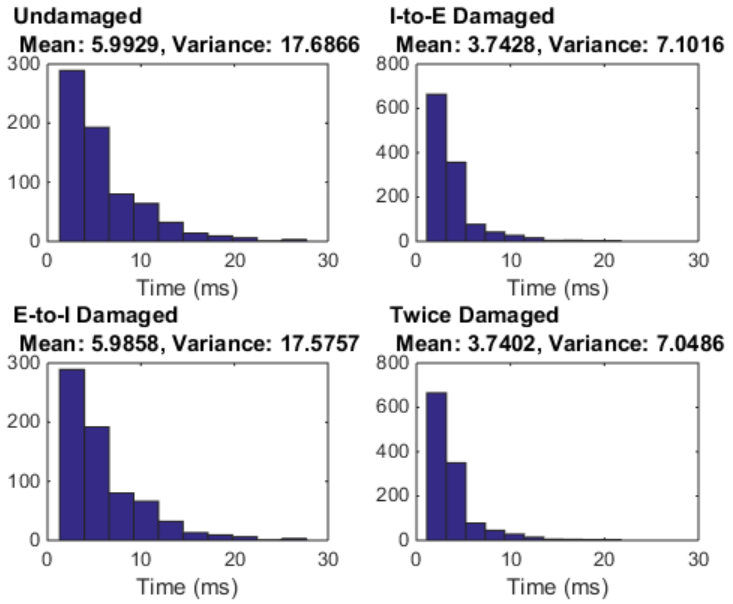

Figure 3: Excitatory Cell Inter-Spike Interval Histograms

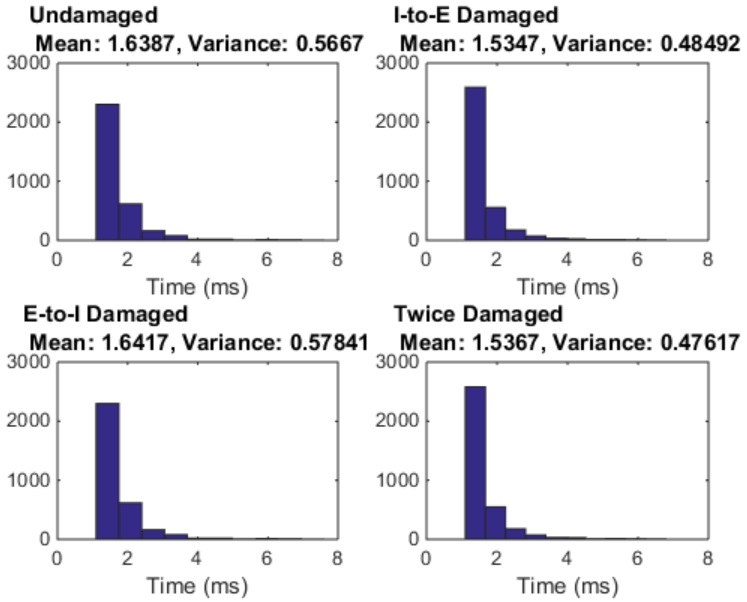

Figure 4: Inhibitory Cell Inter-Spike Interval Histograms

Excitatory to Excitatory with Unidirectional Coupling

In this case, a single damage is possible. The cell at which the axon oiginates is unaffected by damage, so only the outputs of the second cell (at which the axon terminates) is observed and compared. The average D^2 distance between the undamaged and damaged spiketrains is 7.4607. This second cell is excited by both the global input and the other excitatory neuron. When damage is present, it is excited less, but as is shown in the histograms, it still tends to fire at near the maximum rate. This yields a relatively small upward change in the mean, meaning that the network is not particularly harmed by damage in terms of firing rate. However, the spiketrain distance is larger than in the case of the single-cell control, meaning that the network is in fact not resilient with respect to spike arrival time.

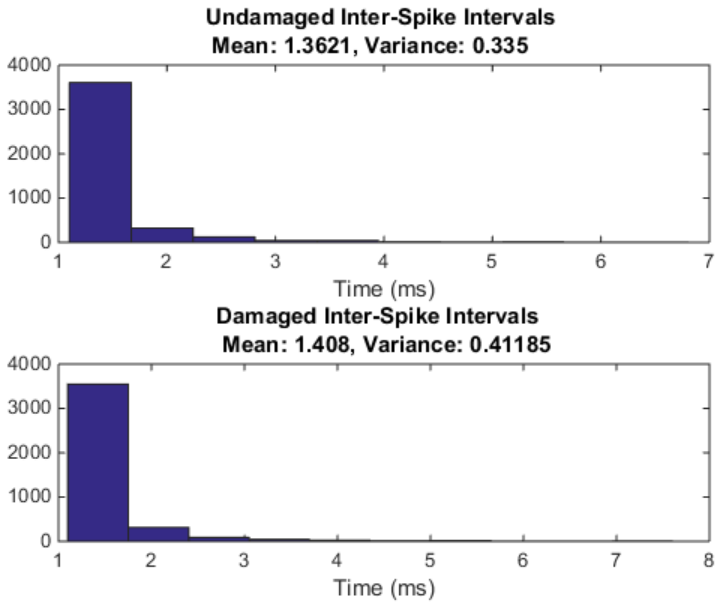

Figure 5: Excitatory-to-Excitatory Inter-Spike Interval Histograms

Excitatory to Excitatory with Bidirectional Coupling

In this case we can look at the output of each cell separately when considering the impact of the damage on the network when a single axon is damaged. When both axons are damaged, the outputs of either cell is identical, so we consider the output of a single arbitrary cell in this case. We look first at the case where one axon is damaged, and observe the output of the cell at which the damaged axon terminates. These histograms can be seen in Figure 6, and the average D^2 distance for this neuron is 12.2695. As an excitatory signal to this neuron is damaged, this neuron ought to fire at a lower rate in the damaged case. This is seen in the histograms, although the scale of the shift in mean is much smaller than that in the one-cell control. This is probably because the cell is often firing at its highest possible firing rate. The D^2 mean is large in this case, however.

We now look at the neuron from which the damaged axon originated. The corresponding histograms can be seen in Figure 7 and the average D^2 distance is 8.3485. The mean hardly changes, again likely due to how often the neurons are firing at their maximum frequency.

Lastly we look at the difference between the output of one cell when no damage is present and when both axons are damaged. The corresponding histograms can be seen in Figure 8 and the average D^2 distance is 12.6306. Unsurprisingly, this distance is higher than in the case of a single damage, but the order of magnitude is still the same. It is clear that all of these changes to the mean are of a smaller magnitude than in the one-cell control, showing that this network is more resilient to damage than a single cell in terms of mean firing rate. Hoever, in terms of spike arrival time, this network shows much higher D^2 distances between undamaged and damaged spiketrains.

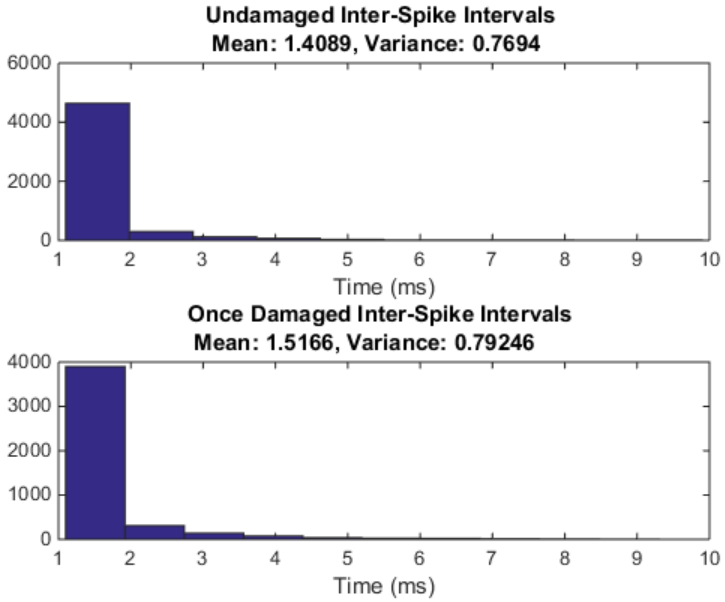

Figure 6: Target of Single Damaged Axon Histograms

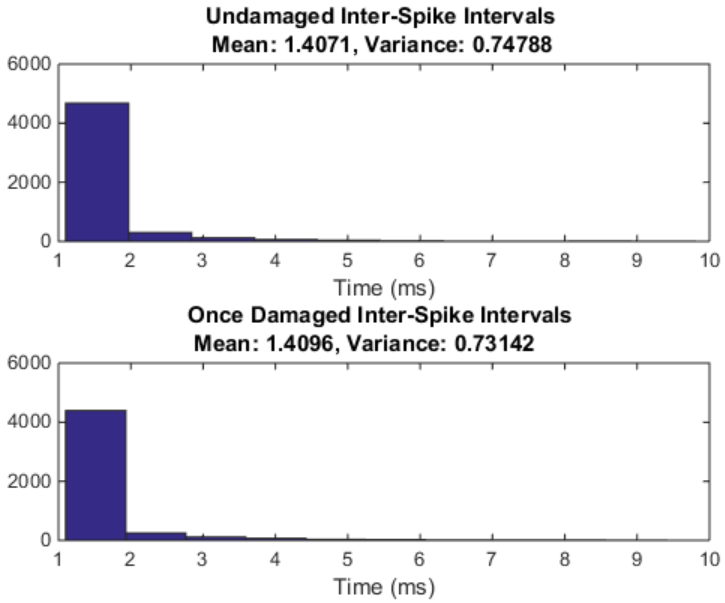

Figure 7: Origin of Single Damaged Axon Histograms

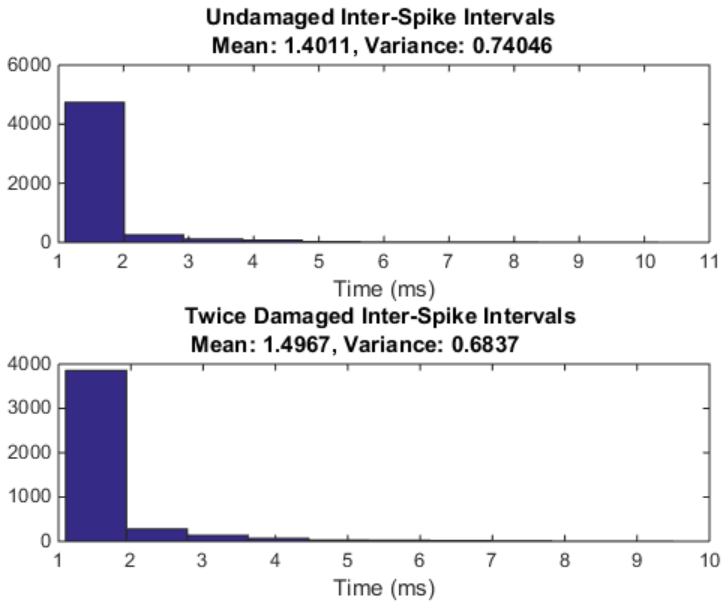

Figure 8: Two Damaged Axons Histograms

Inhibitory to Inhibitory with Bidirectional Coupling

In this case we can look at the output of each cell separately when considering the impact of the damage on the network when a single axon is damaged. When both axons are damaged, the outputs of either cell is identical, so we consider the output of a single arbitrary cell in this case. We look first at the case where one axon is damaged, and observe the output of the cell at which the damaged axon terminates. These histograms can be seen in Figure 9, and the average D^2 distance for this neuron is 1.2890. As an inhibitory signal to this neuron is damaged, this neuron ought to fire at a higher rate in the damaged case. This is seen in the histograms, although the scale of the shift in mean is small when compared to the one-cell control. The D^2 mean is small in this case as well. This is likely because the inhibition of all cells in the network leads to a slower firing network altogether, meaning that damage deletes spikes traveling through the network only rarely.

We now look at the neuron from which the damaged axon originated. The corresponding histograms can be seen in Figure 10 and the average D^2 distance is 1.6605. The effects here are of a similar magnitude as those seen in the target neuron, however the mean has shifted up in this cell. This is somewhat unsurprising, as the target cell was shown to be firing at a slightly higher rate, and is thus inhibiting this origin cell more in the damaged case. This extra inhibition leads to a larger average interspike interval.

Lastly we look at the difference between the output of one cell when no damage is present and when both axons are damaged. The corresponding histograms can be seen in Figure 11 and the average D^2 distance is 2.2756. Unsurprisingly, this distance is higher than in the case of a single damage, but the order of magnitude is still the same. It is clear that all of these changes are of a smaller magnitude than in the one-cell control, showing that this network is

more resilient to damage than a single cell. In terms of changes to the mean and variance, this network is incredibly resilient, with the changes being two orders of magnitude below that of the single-cell case.

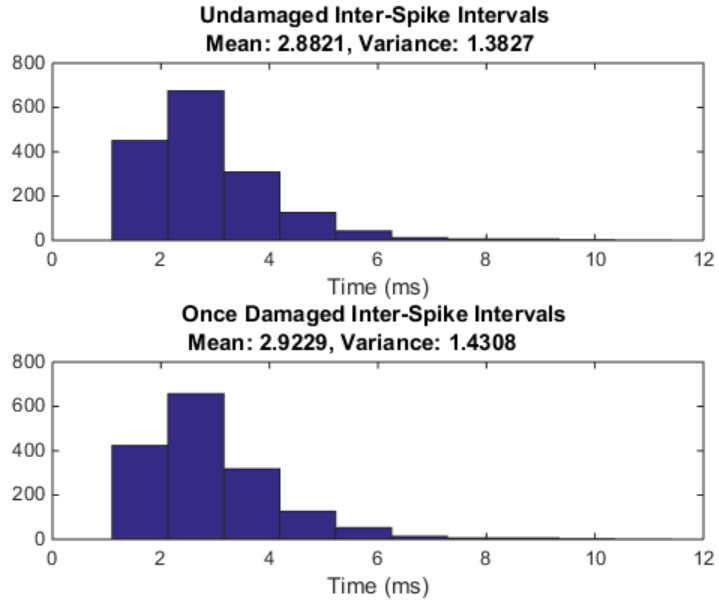

Figure 9: Target of Single Damaged Axon Histograms

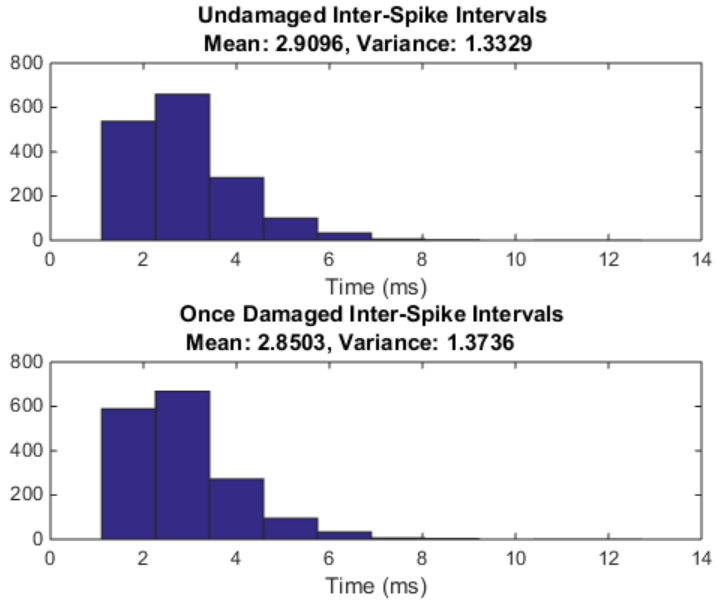

Figure 10: Origin of Single Damaged Axon Histograms

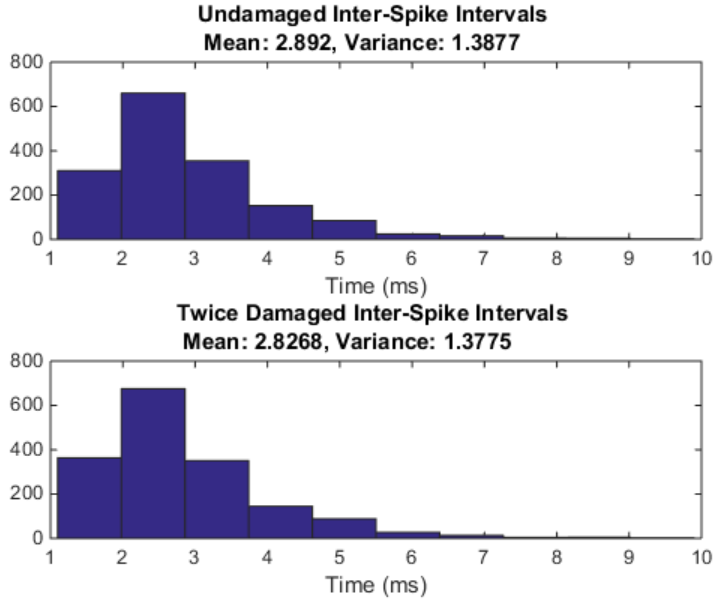

Figure 11: Two Damaged Axons Histograms

Inhibitory to Inhibitory with Unidirectional Coupling

In this case, a single damage is possible. The cell at which the axon oiginates is unaffected by damage, so only the outputs of the second cell (at which the axon terminates) is observed and compared. The average D^2 distance between the undamaged and damaged spiketrains is 4.9773. Damaging the inhibition seen by this observed cell ought to leave the cell more excited, allowing it to fire at a higher frequency. This is exactly what is seen in the histogram below, with the mean of the distribution dropping significantly when damage is present. The variance also decreases, likely as the neuron is firing more frequenctly at its maximum rate (the refractory period), leading to a spiketrain with less variable inter-spike intervals. No obvious recovey property due to the network may be interpreted here, as the change in the mean is substantial and the D^2 distance is of the same order of magnitude as the control.

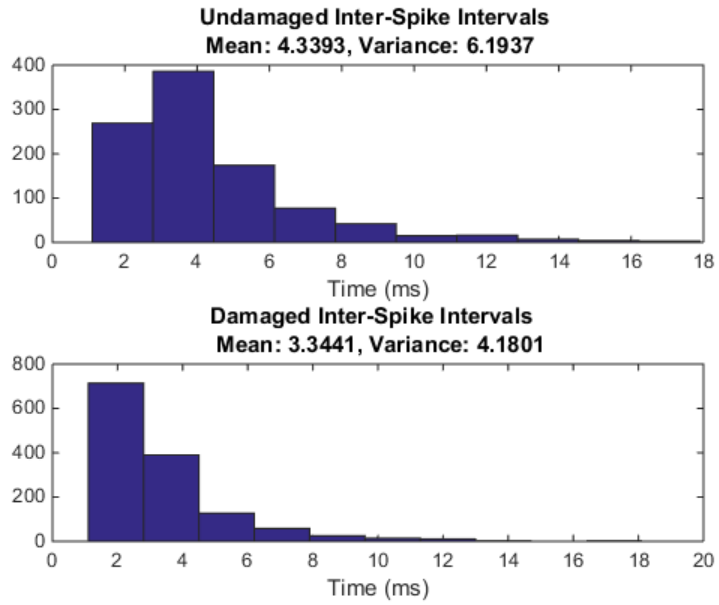

Figure 12: Inhibitory-to-Inhibitory Inter-Spike Interval Histograms

Discussion of Two-Cell Network Results

From this exhaustive two-cell network simulation, it can be seen that damage in the double-excitatory network with bidirectional coupling leads to the largest D^2 distance of all possible architectures and damages. However, it also sees fairly small changes in its statistical properties. It seems that the existence of inhibition in networks often leads to more resilience in the D^2 sense, as well as in the statistical inter-spike interval sense. This makes sense, as inhibition lowers the firing rates of connected cells, in turn making it less likely for a deletion to occur through damage.

With two cells of different types, however, there are also more complex effects at play that can further increase or decrease the D^2 distance between spiketrains. For example, one can

compare the excitatory-inhibitory and inhibitory-inhibitory networks, each with bidirectional coupling. In the inhibitory-excitatory network, the smallest order of magnitude distance is observed between all networks and damages when the excitatory-to-inhibitory connection is damaged. This means that the excitatory cell must not fire particularly high frequency spike-trains very often, else damage would be more effective. This low firing rate across time can be explained by inhibition from the connected inhibitory cell. Yet if we look at the case of the inhibitory-inhibitory network, a similar effect is present but a larger D^2 distance is seen. This is probably because the inhibitory cell in the excitatory-inhibitory network sees two excitatory sources: the global source and the excitatory neuron. It can thereby fire more inhibitory pulses and slow down the excitatory cell. It fires, thereby, at a relatively high rate, so damaging the inhibitory-to-excitatory axon has a much larger effect on the system. In the inhibitory-inhibitory case, each cell receives only a single excitation and inhibition, so this positive feedback effect is not present. This network is still relatively resilient in terms of mean firing rate, however, as the presence of inhibition in a network generally lowers the average fire rate and thereby lowers the impact of damage on the mean firing rate. Table 1 summarizes some of the results across different networks, from which one can see some of these aforementioned properties.

Architecture	Damage	Output Cell	Mean	Variance	D^2
E-E Bidirectional	Undamaged	Either	1.4	0.74	–
E-E Bidirectional	Both Axons Damaged	Either	1.5	0.68	12.6
E-I Bidirectional	Undamaged	Excitatory	6.0	17.7	–
E-I Bidirectional	E-to-I Damaged	Excitatory	6.0	17.6	0.15
E-I Bidirectional	Both Axons Damaged	Excitatory	3.7	7.0	5.2109
I-I Bidirectional	Undamaged	Either	2.9	1.4	–
I-I Bidirectional	One Axon Damaged	"Target"	2.9	1.4	1.3
I-I Bidirectional	Both Axons Damaged	Either	2.8	1.4	2.2756

Table 1: A Summary of Some Two-Cell Network Results

Three-Cell Networks

When three cells are present, the possible number of networks and damages becomes significantly larger. Thanks to the speediness of the damage-approximating algorithm, however, it is tractable to exhaustively simulate all possible networks architectures and damages in the same sense as was done in the two-cell case. This was done, with the same observables such as mean and variance of interspike intervals and average D^2 distance being computed for each damage paradigm. Unlike in the two-cell case, it is overwhelming to observe all of this data at once, and as such, the following sections will simply display the results of certain architecture classes. We begin with three-cell feedforward structures, which present some properties which are not particularly apparent in the two-cell case. We then look at the impact of adding certain single connections to these structures.

Feedforward Networks

Three-cell feedforward networks contain two axons and three indexed cells, Cell 1, Cell 2 and Cell 3. In this architecture, Axon 1 originates at Cell 1 and ends at Cell 2 and Axon 2 originates

at Cell 2 and ends at Cell 3. This is shown in Figure 13. We consider the output of this network to be the output of Cell 3. As the output of Cell 3 does not impact the rest of the network, it does not matter whether this cell is inhibitory or excitatory. As such, there are four unique feedforward network architectures determined by the first two cells, and there are four possible damage scenarios (as there are two possible axons which can be either undamaged or damaged) for each network. We present some results regarding D^2 damage to these networks in Table 2.

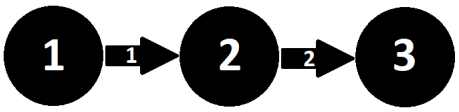

Figure 13: Feedforward Architecture with Cell and Axon Indices

Cell 1	Cell 2	Axon 1	Axon 2	D^2
E	E	Undamaged	Damaged	9.59
E	E	Damaged	Damaged	9.60
E	I	Undamaged	Damaged	5.81
E	I	Damaged	Damaged	5.90
I	E	Undamaged	Damaged	0.20
I	E	Damaged	Damaged	6.23
I	I	Undamaged	Damaged	0.58
I	I	Damaged	Damaged	5.37

Table 2: A Summary of Some Three-Cell Feedforward Network Results

From the above table, we can begin to see some resilience properties that certain networks have to damage. For example, there are two damage paradigms present for which a smaller order of magnitude of D^2 distance is seen than in the other presented networks. These are cases in which Axon 2 is damaged but Axon 1 is left undamaged, and Cell 1 is an inhibitory cell. This is to say that the cell whose output is being damaged is itself inhibited. In this case, the smallest effect of damage is seen. This is likely because Cell 2 in these cases is made to fire slower on account of inhibition from Cell 1, causing the damage on Axon 2 to have less of a severe impact on the output signal. This is true in the opposite sense as well; higher damages are seen when Cell 2 is being excited by an excitatory cell.

This can inform further insights into the two-cell network case. In the two-cell networks, there are no feedforward networks which exhibit effects like this, as only one axon is present. However, in cases with bidirectional feedback, it is possible to damage axons which carry the output of an inhibited or excited cell. For example, in the case of a network containing one excitatory and one inhibitory cell with bidirectional coupling, damaging the axon which originates at the excitatory cell is to damage the output of an inhibited cell. As can be seen in the two-cell network results above, this is a case in which damage has a smaller order of magnitude impact, exactly as in the feedforward case with three cells. Similarly, damage to the other axon in this network is damage to the output of an excited neuron, which has a significantly larger effect than the aforementioned damage.

Skew Product Networks

A skew-product network is a network where the cells are indexed as Cell 1, Cell 2 and Cell 3, in which axons connect cells of lower index to cells of higher index only. These networks contain 3 axons: Axon 1 originates at Cell 1 and terminates at Cell 2, Axon 2 originates at Cell 2 and terminates at Cell 3 and Axon 3 originates at Cell 1 and Terminates at Cell 3. In other words, these networks are like feedforward networks with a single added axon from Cell 1 to Cell 3, as can be seen in Figure 14. Once again, the output of Cell 3 is considered the output of the network. It's output is, again, irrelevant to the operation of the rest of the network, and so the network architecture is specified entirely by the first two cells. There are now three axons which can be damaged, yielding eight possible damages for each network. We present some examples of D^2 data for these networks in Table 3.

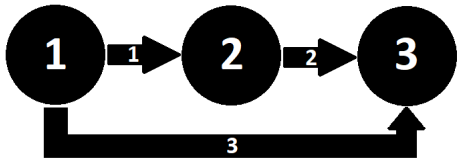

Figure 14: Skew Product Nework with Cell and Axon Indices

Cell 1	Cell 2	Axon 1	Axon 2	Axon 3	D^2
E	E	Undamaged	Damaged	Undamaged	5.81
E	E	Undamaged	Undamaged	Damaged	1.33
E	I	Undamaged	Damaged	Undamaged	9.46
E	I	Undamaged	Undamaged	Damaged	5.81
I	E	Undamaged	Damaged	Undamaged	0.30
I	E	Undamaged	Undamaged	Damaged	5.88
I	I	Undamaged	Damaged	Undamaged	0.09
I	I	Undamaged	Undamaged	Damaged	2.46

Table 3: A Summary of Some Three-Cell Skew Product Network Results

This data further shows the property described in the section on feedforward networks; damaging he output of inhibited cells shows a smaller effect on network output, and damaging the output of excited cells shows a larger effect on network output. Look first at the case in which Cell 1 and Cell 2 are each inhibitory. Each of Axon 2 and Axon 3 feed directly into Cell 3, however Axon 2 originates at an excited cell (Cell 2), while Axon 3 originates at Cell 1 which receives only global input. Damage to Axon 2 shows a significantly larger effect on the D^2 distance, as expected if one believes the above property. This effect is similarly visible in the case where Cell 1 is excitatory and Cell 2 is inhibitory. The symmetric effect on inhibited cells is very apparent in either case where Cell 1 is inhibitory.

Addition of a Feedback Path

Starting with the feedforward structure, three feedback paths can be added: one from Cell 2 to Cell 1, one from Cell 3 to Cell 2 and one from Cell 3 to Cell 1. We consider first the case in which the first of these feedback paths is added, labeling Cells 1, 2 and 3 and Axons 1 and 2 as we did in the feedforward case, and titling this new feedback path Axon 3. This can be seen in Figure 15. We present some results in Table 4.

Figure 15: Single-Feedback-Path Nework with Cell and Axon Indices

Cell 1	Cell 2	Axon 1	Axon 2	Axon 3	D^2
E	I	Undamaged	Unamaged	Damaged	2.46
E	I	Damaged	Undmaged	Unamaged	0.07
E	I	Undamaged	Damaged	Unamaged	5.47
I	E	Undamaged	Unamaged	Damaged	0.11
I	E	Damaged	Undmaged	Unamaged	7.44
I	E	Undamaged	Damaged	Unamaged	0.37

Table 4: A Summary of Some Single-Feedback-Path Network Results

While the feedback makes the mechanisms by which cells communicate far more complex, the same principles that were apparent in the above sections (on feedforward and skew product networks) can be seen here as well. For example, look first at the network where Cell 1 is excitatory and Cell 2 is inhibitory. Cell 1 is inhibited, as it is communicated to by Cell 2. Our understanding from earlier sections would lead us to believe that damaging its output would thereby have a smaller effect on the network output, and this is very clearly the case; in fact, the effect on the D^2 distance is two orders of magnitude smaller than some other damages. On the other hand, Cell 2 is excited by Cell 1, and so one would expect damaging its output to have a relatively large effect on the output of the network. This damage, a damage to Axon 2, has the largest effect of any single-axon damage. A symmetric set of effects can be seen in looking at the case in which Cell 1 is inhibitory and Cell 2 is excitatory, where Axons 2 and 3 come from inhibited cells and Axon 1 comes from an excited cell.

More generally, it is the case that when one constructs a network of three cells by adding an undamaged feedback connection to a damaged feedforward network, the effects of damage will be mitigated in the case that the feedback originates from an inhibitory cell or heightened in the case that the feedback originates from an excitatory cell.

There are many other three-cell networks whose properties have not been discussed here, however nearly all thee-cell networks exhibit these discussed properties; outgoing connections from excited cells more severely impact the network output when damaged, and outgoing connections from inhibited cells less severely impact the network output when damaged.

Population Networks

One can consider networks of similar structure to those studied in the two- or three-cell case in which each cell is replaced instead with a population of cells of one type. This is a natural way to increase network size while fixing some architectural properties that could be of interest. Through observing networks such as this, we gain some insight into the effects of network size on the resilience of networks to damage.

A Feedforward Network with Two-Cell Populations

As a first example of this, we consider a feedforward network in which there are three populations containing two cells each. The first population contains inhibitory cells and the second and third populations contain excitatory cells. Within populations, the cells are bidirectionally connected, and each cell feeds into both cells in the subsequent population. We recall from the feedforward networks the network in which Cell 1 was inhibitory and Cell 2 was excitatory; a network which is analogous to the six-cell network in question here. That network showed some relative resilience in the D^2 sense when Axon 2 was damaged, giving a D^2 distance of 0.20. Here, there are four axons between population 2 and population 3, which are analogous to this axon in the three-cell case. There are two cells in the final population, so when observing the results we look at the D^2 distances they present separately. We tabulate the effects of damaging one, two, three or four of these axons below, using an enumeration such that the first damaged axon feeds into Cell 1, the second damaged axon feeds into Cell 2 and he third damaged axon feeds into Cell 3.

Number of Damaged Axons	D^2 for Cell 1	D^2 for Cell 2
1	0.09	0.06
2	0.20	0.20
3	0.20	0.22
4	0.38	0.38

Table 5: A Summary of Some Population Network Results

When only one axon is damaged, each cell experiences a damage that is an order of magnitude smaller than in the three-cell case. For all other damages, the damage is comparable to the damage in the three-cell case.

Summary

In studying two- and three-cell networks, the properties most easily seen to affect robustness are second-order effects of inhibition and feedback. The former is most easily summarized by saying that if a cell A has only inputs from inhibitory cells, axons from A have a smaller effect on network output when damaged than those axons coming from cells that do not have this property. Conversely, if cell B has only inputs from excitatory neurons, axons from B have a larger effect on the network output when damaged than those axons coming from cells which do not have this property.

The effects of feedback are simply summarized by saying that if a damaged feedforward network is equipped with an undamaged feedback path, the effects of damage on the network output are

amplified in the case that the feedback path originated at an excitatory cells. Conversely the effects of damage are attenuated in the case that the feedback path originates at an inhibitory cell.

Appendix – Pseudo-Spectral Method

The Spectral Method

The pseudo-spectral method is a modification of the simpler spectral method. To illustrate this method, we look at a simplified version of the first PDE given by

$$\frac{\partial V}{\partial t} = \frac{D}{d(x)} \frac{\partial}{\partial x} \left(\frac{d^2(x)}{r_L(x)} \frac{\partial V}{\partial x} \right) \tag{4}$$

The first step in the method is to decide upon some discretization of space, so that at each time V is represented by an N-point vector. Let the set $\{\psi_n\}_{n=0}^{N-1}$ be orthonormal basis functions, such as the Fourier basis on the space of discrete-space N-point functions. Assume that V is separable and can thus be represented in the following way:

$$V(x,t) = \sum_{n=0}^{N-1} c_n(t)\psi_n(x)$$

for some c_n which, importantly, do not depend on the spatial coordinate x. Using the standard Fourier basis vectors for discrete-space functions with finite domain, the c_n coefficients are exactly those determined by the discrete Fourier transform of the function in the variable x, efficiently computed via use of the fast Fourier transform (FFT) algorithm. The left-hand side of the equation can thus be written as

$$\frac{\partial V}{\partial t} = \sum_{n=0}^{N-1} c_n'(t)\psi_n(x)$$

Using properties of the Fourier transform, x-derivatives can be taken in the transform domain via multiplication by ik_n, where i is the imaginary unit and k_n is the coefficient in the exponent of the n^{th} Fourier basis function, giving

$$\frac{\partial V}{\partial x} = \sum_{n=0}^{N-1} ik_n c_n(t)\psi_n(x)$$

The functions d and r_L are entirely functions of space, and assuming they have been discretized in the same way, they can similarly be represented as superpositions of the same Fourier basis functions. Their Fourier coefficients do not depend on time, and as such the only time-dependence in the right-hand side's total Fourier expansion is a result of the Fourier expansion of V alone. Multiplication in real space (the x-variable space rather than Fourier space) is given by convolution of Fourier coefficients in Fourier space. As the Fourier coefficients of r_L and d are simply real numbers, multiplication by these functions in real space simply yields coefficients of the right-hand side of the equation which are linear combinations of the original $c_n(t)$ coefficients. That is to say that the entire PDE can be cast as:

$$\sum_{n=0}^{N-1} c_n'(t)\psi_n(x) = \sum_{n=0}^{N-1} \left(\sum_{m=0}^{N-1} a_{n,m} c_m(t) \right) \psi_n(x)$$

where the $a_{n,m}$ coefficients are complex numbers which do not depend on t or x. The a coefficients are determined by:

1. Taking the initial FFT of V
2. Multiplying by ik_n to obtain the first space derivative of V in the transform domain

3. Taking the inverse FFT (IFFT) and multiplying pointwise by $d^2(x)/r_L(x)$
4. Taking the FFT of this expression, multiplying once again by ik_n to take another derivative
5. Taking another IFFT and multiplying pointwise by $D/d(x)$
6. Taking the FFT of this final expression in real space to obtain the entire right-hand side in Fourier space

As such, we have phrased the PDE as a system of N coupled linear ordinary differential equations, each of the form:

$$c_n'(t) = \sum_{m=0}^{N-1} a_{n,m} c_m(t)$$

A system like this can be solved with any numerical ODE solver, such as Runge-Kutta. One can iterate through any number of time steps entirely in the Fourier domain, and simply IFFT the result to obtain V at different time coordinates.

The Pseudo-Spectral Method

In the presence of nonlinear terms, the spectral method is no longer viable. Consider the V terms from the voltage equation from the FitzHugh-Nagumo model:

$$\frac{\partial V}{\partial t} = \frac{D}{d(x)} \frac{\partial}{\partial x} \left(\frac{d^2(x)}{r_L(x)} \frac{\partial V}{\partial x} \right) + V(V-a)(V-1)$$

The first term on the right-hand side was seen to be handled by the spectral method as described above, but trouble is run into with the introduction of the term which is cubic in V. The formulation of a system of N linear ODEs is not possible here, and simply iterating through time with the spectral representation of the system will not represent the nonlinear ways in which the cubic term changes in time. The pseudo-spectral method makes a rather simple alteration to the original method to handle these sorts of nonlinearities by assuming that for each time-step in your ODE algorithm of choice, the nonlinearity can be considered as approximately linear. After each iteration of the algorithm, the IFFT of the data is taken so that the nonlinearity can be once again computed in real space. The FFT of the right-hand side with this updated nonlinearity is then computed, and another iteration of the algorithm can be run. That is to say that each iteration of the ODE algorithm comprises the following steps:

1. Take the IFFT of the current V Fourier coefficients
2. Compute the nonlinear (in this case, cubic) term in real space
3. Take the FFT of the new linear term
4. Compute the linear term in the same method as described in the section on the spectral method
5. Add the linear and nonlinear Fourier coefficients and then iterate the ODE algorithm

The extension of this method to a system of PDEs is as one would expect; apply this same FFT, IFFT and ODE algorithm to multiple discrete-space functions with the same spatial discretization, so as to convert the system of n PDEs of length N discrete-space signals into a system of nN ODEs. This is shown in the worked out example in the next section.

Application to the FitzHugh-Nagumo Model

Recall that the system of PDEs in question is given by

$$\frac{\partial V}{\partial t} = \frac{D}{d(x)}\frac{\partial}{\partial x}\left(\frac{d^2(x)}{r_L(x)}\frac{\partial V}{\partial x}\right) + V(V-a)(V-1) - R$$
$$\frac{\partial R}{\partial t} = bV - cR$$

Assume that a, b, c, D, $d(x)$, $r_L(x)$ are provided, as well as initial conditions for V and R. First, one must decide on a spatial discretization. The system naturally has some finite length L, so take $\vec{x}$ to be a vector of N evenly spaced points between $-L/2$ and $L/2$. N is normally chosen to be some power of 2, as it will also be used as the number of Fourier modes for each discrete-space function. Take $d(x)$, $r_L(x)$, $V(x,0)$ and $R(x,0)$ at the points in $\vec{x}$ to get length N vectors d,r,$V(0)$ and $R(0)$.

Recall that the Fourier basis $\{\psi_n\}_{n=-\frac{N}{2}}^{\frac{N}{2}-1}$ is given by $\psi_n = e^{ik_n x}$, where $k_n = \frac{2\pi n}{L}$. This gives a length N vector $\vec{k} = \begin{pmatrix} k_0 \\ k_1 \\ \vdots \\ k_{(N/2)-1} \\ k_{-N/2} \\ k_{(-N/2)+1} \\ \vdots \\ k_{-1} \end{pmatrix}$.

We can now begin to solve the system of differential equations numerically. First, we apply the FFT to both $V(0)$ and $R(0)$, and concatenate the two vectors to obtain a vector of length $2N$. The first N elements of this vector are $\hat{V}(0)$, the discrete Fourier transform of $V(0)$, while the last N elements are $\hat{R}(0)$, the discrete Fourier transform of $R(0)$. Recall that in the pseudo-spectral method, we are looking to solve $2N$ ODEs in the Fourier coefficients of V and R; this vector will be our input initial conditions for an ODE solver which will do just that. Naturally, ODE solvers take initial conditions, timesteps, and a single function as inputs. One can decide on timesteps as they may please, however, the function representing the right-hand side of the equation is of the utmost importance here.

For each iteration of the function at time t, the right-hand side must carry out the following operations:

1. Extract $\hat{V}(t)$ and $\hat{R}(t)$ from the length $2N$ input vector
2. Take the IFFT of $\hat{V}(t)$ to obtain $V(t)$
3. Use this real-space $V(t)$ to compute $V(V-1)(V-a)$
4. Take the FFT of this nonlinear term and save it for later use
5. Begin to compute the linear term by first taking the point-wise product of $i\vec{k}$ and $\hat{V}(t)$
6. Take the IFFT of this expression and multiply it pointwise by d^2/r
7. Take the FFT of this expression and multiply it point-wise by $i\vec{k}$
8. Take the IFFT of this expression of this expression and multiply it by D/d
9. Take the FFT of this expression, add the nonlinear term and subtract $\hat{R}(t)$ to obtain the first N entries of the output

10. Take $b\hat{V}(t) - c\hat{R}(t)$ to obtain the last N entries of the output
11. Return these two expressions concatenated as a single $2N$-length vector

Note that the nonlinearity is recomputed each iteration, as explained in the previous section. The output of the ODE solver after n timesteps will be an $n \times 2N$ matrix. To obtain real-space data for V and R in time, separate the first N and second N columns of this matrix and IFFT each column of these two new matrices.

To-Do List

- Run statistical significance studies on DD^2 distance
- observe effects of larger feedforward networks
- Observe more population networks
- Consider implementation with blockage or reflection