MODELING OF THE NETWORKING AND ACTIVITY OF CULTURED MOUSE NEURONS FOR SIMULATED EXPERIMENTS

BY

ABRAHAM SHULTZ
B.S. WORCESTER POLYTECHNIC INSTITUTE (2004)

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF MASSACHUSETTS LOWELL

Signature of Author: _____ Date: 4/30/2013

Signature of Thesis Supervisor: _____
Holly Yanco, PhD

Signatures of Other Thesis Committee Members: _____
Thomas Shea, PhD

MODELING OF THE NETWORKING AND ACTIVITY OF CULTURED MOUSE NEURONS FOR SIMULATED EXPERIMENTS

BY

ABRAHAM SHULTZ

ABSTRACT OF A THESIS SUBMITTED TO THE FACULTY OF THE
DEPARTMENT OF COMPUTER SCIENCE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE
UNIVERSITY OF MASSACHUSETTS LOWELL
2013

Thesis Supervisor: Holly Yanco, PhD

Professor, Computer Science Department

Multi-Electrode Arrays (MEAs) are a useful tool for inspecting the activity of living, interacting cultures of neurons in a manner that is not intrusive to the cells. However, the limited lifespan of the cells *in vitro* and the impossibility of exactly duplicating any particular network of living cells are problems for research with MEAs. This thesis proposes a simulation method for modeling the layout and networking of the cells in a culture, and for simulating the activity of that network. The development of the simulator and the selection of its parameters to match those of biological networks are discussed. Results from the simulation are compared with results from biological networks to determine if the simulation is of sufficient fidelity. Once the simulation is completed, it will have sufficient predictive value for some aspects of MEA research to allow easier "rough drafts" of experiments. These rough drafts may allow researchers to determine if the effort of culturing a living neural network is warranted.

ACKNOWLEDGMENTS

Thank you to Holly Yanco, Thomas Shea, Sangmook Lee, Jill Zemianek, Mary Guaraldi, and Jonathan Hasenzahl.

This work was sponsored in part by Army Research Office grant W911NF-11-1-0125.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Cultured Neurons in Research

In order to gather information about about the behavior of neurons, neurobiology researchers grow cultures of neurons outside of the animals that produced them. These cultures of neurons allow the researchers to perform experiments that operate directly on the neurons, without the complications that may be caused by the interacting systems of a living organism. For example, tetrodotoxin (TTX) prevents sodium channels from acting, which prevents neurons from signaling to each other. In culture, the suppression of signaling does not kill the neurons, and can be reversed by removing the TTX. However, in organisms, the action of TTX paralyzes the respiratory muscles, which kills the animal.

Despite the advantages of cultures, they also have drawbacks. Neurons in an organism have the rest of the organism to protect them. The immune system of the organism provides protection from bacteria, and the bloodstream of the animal brings the food and oxygen that the neurons need to survive. In a culture, protection and sustenance for the neurons must be provided by the researcher, which makes the maintenance of cultures time-consuming and difficult. This thesis describes the development and validation of software to simulate a culture of neurons, with the intent to perform experiments in simulation before attempting them with biological cells.

1.1.1 Experimental Uses

A Multi-Electrode Array (MEA) is a type of culture dish that provides researchers with a way to monitor the electrical activity of neurons at or near the level of individual cells.

Because the cells are grown in a culture medium, chemicals can be added to or removed from their environment to modify their ability to signal. Research in Thomas Shea's lab (Shea 2009) has used this method to demonstrate that inhibitory connections are required for learning in cultured neurons. For the purposes of this thesis, the neurons under discussion are a culture of disassociated mouse neurons.

1.1.2 Construction

The MEA itself consists of a glass plate with an array of conductive pads laid out on it, as shown in Figure 1. Conductive traces extend from each pad to the edges of the plate. These conductive pads are used to detect the electrical activity of neurons cultured on the plate. When a neuron sends a signal, its electrical potential changes, and this change in potential is detected by sensitive amplifiers connected to the traces for pads near that neuron. The size of each pad is close to the size of a single neuron, so neuron firing can be localized to a single neuron or small group of neurons by determining from which pad the signal came.

To start a culture, the MEA glass is first prepared by coating it with laminin and other proteins that enable neurons to bind to the plate surface. The presence of this protein creates a surface that cells are able to stick to, but does not guarantee that a cell will adhere to any specific location and grow there. In order to acquire cells, mice must be bred and sacrificed, and the fetal mouse neural tissue must be surgically prepared and chemically treated before being plated on the MEA. Fetal mice are used as the cell source, because their neurons are still developing and forming connections.

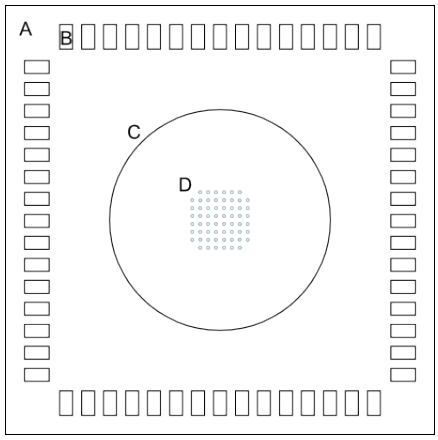

Figure 1: A. The glass base plate. B. Contacts for connection to amplifier. C. Culture media retaining ring. D. Grid of electrodes to detect neuronal signals. Note that this image is not to scale. The grid of electrodes, in particular, is magnified, as it would not otherwise be easily visible. The connections between B and D are not shown for clarity.

The chemical treatment uses enzymes to disassociate the individual neurons. The neurons are added as a suspension in liquid medium and given some time to bond, after which the culture medium is replaced, removing any unbonded cells with the old culture medium (Wagenaar, Pine, and Potter 2006). Typical cell suspension densities range from 300 to 2,000 cells per square millimeter, but can reach as high as 80,000 cells per square millimeter (Shea 2009; Ruaro, Bonifazi, and Torre 2005). Extremely sparse cultures tend to have high mortality rates and do not form sufficient connections to display mature signaling patterns (Shea 2009).

In addition to culture density, the distribution of cells on the plate can be controlled in other ways. One method is to apply the suspension of neurons to the desired regions using a micropipette, resulting in higher neuron density in areas where the drops of suspension were added. Another method is to apply the adhesive protein to the plate in a pattern. The patterned substrate is created by using a process such as microstamping to apply a pattern of binding proteins, rather than an even coating, to the culture area. The cells can only adhere to the areas where the protein

is applied, so the resulting plate has areas with high cell density and areas with few or no cells. The pattern of cells influences the connectivity of the culture (Sorkin, Gabay, Blinder, Baranes, Ben-Jacob, and Hanein 2006).

When the cells are initially added to the culture, they are not connected. For most of the first month in culture, the cells build new connections. Starting at around 7 days *in vitro* (DIV) and continuing to around 30 DIV, the connections are not complete, and signaling is dominated by constant, high-amplitude spiking (Warwick et al. 2010). This type of signaling is visualized in Figure 2. The resulting signals have been described as "epileptiform."

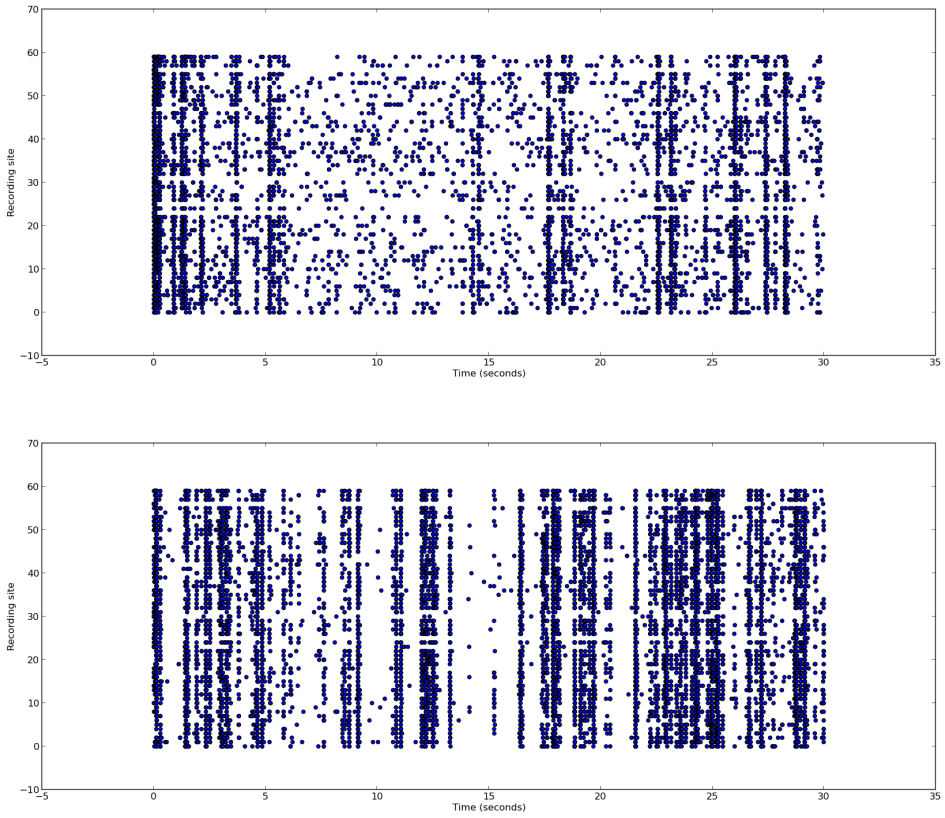

Figure 2: On the top is a graph of typical activity in a mature biological network. The vertical bands are culture-wide activity. The bottom graph shows an immature network. The inhibitory connections have not yet formed, so the there is substantially more activity, particularly in the form of culture-wide bursts.

After the initial period of epileptiform activity, the cells enter a "mature" phase, characterized by sparse bursts of spikes separated by quiet periods. The active bursts may be localized to one region, spread across the culture, or propagate from region to region. After 2-3 months of this sort of activity, the culture eventually becomes senescent, and only reacts to stimuli in simple, stereotyped ways (Warwick, Xydas, Nasuto, Becerra, Hammond, Downes, Marshall, Whalley, et al. 2010). The cells can continue to live for months or even years, assuming that equipment failure or bacterial infection does not kill them (Potter and DeMarse 2001).

1.2 Problem Statement

Real neurons in culture have a limited life span. In the young, epileptiform stage, it is impossible to isolate the neurons' response to stimulus from the constant spike activity, so experiments must be performed after the neuron network is finished developing. As a consequence, the window for research on an MEA only lasts as long as the period of mature, complex interaction. The period of mature signaling usually lasts 2-3 months, after which the culture enters a senescent stage, where the neuronal responses are minimal or stereotyped.

The cultures as a whole are difficult to maintain. After the culture is plated, the culture medium must be replaced regularly, and the culture as a whole must be maintained at tightly controlled temperature and humidity levels, usually in an automated incubator. All of this equipment and maintenance is expensive, and it is all required to prevent the cells dying due to bacterial infection or a hostile environment. In addition to the costs of keeping the culture alive, there is a significant investment in hardware to acquire the signal from the MEA, amplify it, and record it. All of this equipment is immobile, so physical access to the equipment is required to perform experiments.

Because the culture is applied to the plate as a suspension of neurons, there are only limited ways to control the eventual location and distribution of neurons. The formation of connections between cells in the culture is also stochastic, though some overall organization is emergent from "rules" within each cell. As a result, each biological culture is unique and cannot be exactly replaced when it dies.

There are computational methods for determining the approximate wiring of a developed culture, based on the propagation delay of a signal in the culture and the synchrony of activity between different sites in the culture (Erickson, Tooker, Tai, and Pine 2008; Esposti and Signorini 2008b). These methods offer some promise for mapping the connectivity of the dish, but they do not give a complete or fully-accurate map. Even if it was possible to completely map the connections of a culture, there is no way to duplicate it, as there is no way to control the growth of individual biological neurons and their axons.

1.2.1 Advantages of Simulation

One desirable aspect of simulated MEA cultures is that the design of the simulation may allow the elimination of aging as a confounding variable. It is actually simpler to produce a simulation in which the age of the cells is not taken into account than one in which it is. In such a simulation, the cells would be modeled as if they are always mature but not yet senescent. Assuming that the parameters that govern the behavior of cells in biological neural network at a particular point in time can be determined, those parameters can be held constant. As a consequence, all of the experiments performed in the simulation would be performed on a representation of mature population of cells, removing the possibility of the culture aging out of the period in which it supports experimentation.

Another convenient aspect of simulated MEA cultures is that they are cheap and portable. The proposed simulation would be a software environment and data for it, which can be trivially duplicated and transported as easily as e-mailing it from researcher to researcher. The simulation would run on an ordinary desktop computer, and so running a simulated experiment would not require specialized hardware. Designing the simulation to operate on commonly available hardware enables geographically distributed groups of researchers to test multiple possible array configurations, cell densities, and other parameters.

Assuming it is possible to map the connectivity of a culture and accurately model the activity of the cells of that culture, it may be possible to have a simulated culture model not only a common configuration of electrodes and cell types, but also the neuronal connectivity and behavior of a particular culture. Such a simulation would enable researchers to share a model of that specific culture, and its behavior, with colleagues in distant locations. It could even allow researchers to continue to experiment on a simulated version of a particular culture after the biological version has died.

If the simulation could be shown to have sufficient fidelity, these simulated experiments could also be used to test multiple avenues of research in parallel. The results of the simulation would be used to guide the selection of configurations most likely to be fruitful when applied to real cells, at a lower cost than experimenting with living cells, and in much less time.

1.2.2 The Need for Validation

Overuse of simulations presents a danger of the research becoming ungrounded from the behavior of actual neurons, and so researchers would do well to keep the limitations of simulation in mind. As long as there are unknown factors in inter-neuron

communication, no simulation will be perfect. In addition, limits to computational power and development time impose limits on the fidelity of any simulation.

Biological neurons are sufficiently complex that the most accurate simulations of a single cell are computationally intense to run. A simulation of a large population of cells must trade some degree of biological veracity for computational tractability. If the behavior being examined depends on a detail that the simulation elides to save on processor cycles, the behavior of real cells and the simulation will differ. For example, real neurons have separate channels for sodium and calcium ions. Some drugs, e.g. lithium salts, are believed to work by interacting with sodium channels. If a simulation does not model these channels, but simply treats all incoming charges identically, it will not accurately model the reaction of neurons to a drug that interacts with particular channels. For the purpose of suggesting potentially useful avenues of research with real cells, it is not required that the simulation be perfect, only good enough to match the observed behavior. In order to allow this development, the simulation must be validated against real biological networks to confirm that the products of the simulation display growth, connectivity, and behavior that are sufficiently similar to biological networks for the desired application. As long as the limitations of the simulation are kept in mind, it may serve as a useful tool for researchers.

1.2.3 Contributions

This thesis describes the creation and development of a program called Cultured Neuron Simulator (or CNS). CNS simulates the growth of neurons, in order to create a simulated network similar in topology to a biological network. CNS also simulates the behavior of the neurons, in order to create a record of activity similar to that of biological neurons. The simulator performs both of these operations more quickly

than growing the biological neurons, so it enables researchers to create experimental simulations quickly and easily.

Because the simulation runs on commodity PC hardware, it can be used by any researcher, without the investment in culture plates and support equipment needed to conduct biological experiments. It can also be run thousands of times, with slightly varied parameters, to explore a large space of possible experiment configurations. Any results that appear interesting can be used to guide exploration with biological cultures.

In order to assess the similarity of the results from CNS to biological networks, analysis tools were developed. These tools can also be used to analyze and visualize biological data.

CHAPTER 2

RELATED WORK

2.1 Frameworks for modeling neurons

The requirements for a framework that models a single neuron are different from the requirements for a framework that can model an MEA. Because an MEA has a very large number of neurons, the model used to simulate the activity of each neuron must be computationally lightweight enough that the simulation as a whole can run in a reasonable amount of time. The MEA also includes contact pads at fixed locations, so the simulation must account for the alterations of the electrical signal induced by imperfections in the connections between the pads and the adjacent neurons. Many neuronal modeling frameworks emphasize accurate electrophysiology over computational speed, or have no model of the spatial arrangement of neurons.

2.1.1 Requirements

In order to support a simulation of a MEA, the simulation software must be able to simulate a number of cells equal to the number in an MEA, and do so more quickly than simply plating the cells and observing them. A very dense culture might have 5,000 cells/mm^2, on the order of 400,000 cells in a 20mm diameter MEA (le Feber, Stegenga, and Rutten 2008; Vibert, Pakdaman, Boussard, and Av-Ron 1997). If the simulation environment cannot simulate that many cells, it cannot simulate the full MEA.

Because the MEA has electrodes in a specific spatial arrangement, and can only record from neurons near or on those electrodes, the simulator, and so the simulation framework, must have the ability to work with the physical location of neurons. Information about the relative locations of the cells is not generally an element of most neural network simulation environments, which favor a highly detailed spike-transfer model over concerns about cell placement and growth.

Further, it takes about two weeks for a culture to enter the mature phase of development. If the simulation cannot produce results in less than two weeks, it would be faster to just plate the cells, and perform the experiment on biological neurons. Of course, it would be preferable to have the simulation be much faster, both to facilitate development of the simulator and to ease the work of the simulation's users.

In order to be useful for this project, the software used must be under current development and open source. Choosing open source software allows other researchers to use the work, as well as developing and contributing to it. Proprietary software introduces a single point of failure, where the data produced by the software becomes useless if the software is no longer available or is obsolete. Software obsolesces because the company developing it goes out of business, changes their business plan, or releases updated versions that break backwards compatibility. For this reason, the simulation framework should produce and consume data in human-readable, open formats, and be able to be developed by anyone if the original developers abandon it.

2.1.2 Existing Frameworks

There is already software available for simulating networks of neurons. The predominant model for simulation of individual neurons is a compartment-based model, where the simulated neuron is divided into compartments. Each compartment receives inputs, performs calculations on those inputs, and produces outputs, which

are sent to other compartments. The inputs and outputs are usually representations of flows of ions and their attendant charges within the cell, so the models frequently include representations of different ion channels and receptor types. GENESIS, NEURON, SNNAP, NODUS, and XNBC are all compartment-based software packages for simulating neurons (Kroupina and Rojas 2004).

NEURON allows for a three-dimensional topology of the neurons, and supports modeling networks of neurons as well as individual neurons. The GUI for constructing networks would not be useful for building a network with hundreds of thousands of neurons, as it would require a large amount of interaction time from the user. To avoid using the GUI, automated tools could be used to generate a large network in NEURON's domain-specific language, HOC.

However, executing a very large network could become problematic, as NEURON uses highly detailed neuron models, and so must solve large sets of differential equations for each cell. The original paper describing NEURON does not describe any performance limitations, but makes some reference to the fact that NEURON parallelizes well onto Cray supercomputers (Hines 1993). A later paper characterizes the performance of NEURON on a variety of hardware, showing that multiple processors will usually result in a significant speedup, but sometimes at the cost of heavy investment in supercomputer hardware (Migliore, Cannia, Lytton, Markram, and Hines 2006). A 2,525 cell network takes around a half hour on one processor, but reduces to around 8 seconds when running on 256 processors of the IBM Blue Gene supercomputer. For a complex compartment-based simulation, this level of performance is quite good, but the number of cells involved is still much smaller than the number of cells that may be present in an MEA.

Performance is also a concern with the GENESIS simulation environment, but one that the authors have put significant work into ameliorating. Like NEURON,

GENESIS is a compartment-based simulator with a significant level of detail, down to simulating different types of ion channels. GENESIS includes several equation solvers, and is available in a parallelized version for execution on multiprocessor computers. Given that modern computers are frequently multiprocessor systems, such as Intel's "Core" series, parallelizing the execution of compute-bound aspects of the simulation is a good strategy.

Still, as of 2005, attempting to model more than 1000 neurons, with significant simplifications of the individual neuron models, was "too slow" (Vanier 2005).

SNNAP is largely written in Java, so it can run on any platform, but the DOS version is highlighted as being the higher-performance version. The initial paper describing SNNAP mentions supporting "up to 30 neurons", but the DOS version claims to support up to 10,000 cells, so it has clearly been improved since the first release (Ziv, Baxter, and Byrne 1994). Unfortunately, it is quite possible for a MEA to contain more cells in its active area than even the DOS version of SNNAP can support. As a consequence, there are configurations of cell density and MEA size that exceed the capacity of SNNAP to model. It would be preferable to have the capabilities of the simulator exceed the configurations it could be called upon to simulate, so this is a sufficient constraint to eliminate SNAPP from use in this work.

Neither NODUS nor XNBC appear to be in active development. NODUS was written for Apple Macintosh computers of sufficient age to be of interest primarily to historians, while XNBC has not had a Unix release within the last two years (De Schutter 1993). The most recent version of XNBC supports three cell models. These models appear to be highly customizable, but there is no mention of provisions for describing new models without significant programming (Vibert, Alvarez, and Kosmidis 2001). XNBC also only claims support for "thousands" of neurons as of version 8.

In all of these software suites, the emphasis appears to be on highly-detailed models of individual neurons, more than on the behavior of large groups of interacting neurons. Highly detailed models tend to be computationally intense, and so the math involved in simulating the neurons consumes a large amount of time when the networks become large. As a group, these simulators appear to be too detailed to support a very large network, because of the amount of computation required to update the network.

Recently, there have been two software frameworks, CX3D and NETMORPH, developed for modeling the morphological properties of developing neurons (Koene, Tijms, van Hees, Postma, de Ridder, Ramakers, van Pelt, and van Ooyen 2009). These modeling frameworks do not include the activity of the cells once they are grown, but instead aim to accurately simulate the forms of the cells as they develop (Acimovic et al. 2011). For this reason, they are not complete tools for MEA simulations, although the connectivity data they produce could be used together with an activity simulator such as NEURON (Koene et al. 2009). They also are too slow for use in simulating a complete MEA. On a modern PC with 4GB of RAM and a 3GHz processor, both simulators took times ranging from 2 hours to a week to perform simulations involving 100 neurons (Acimovic et al. 2011). A full MEA can have 10,000 to 100,000+ cells, so assuming a linear increase in runtime, the simulation could be expected to take anywhere from just over a week in the best case, to more than nine months in the worst case. A simulator that is slower than the real process is not useful for performing the sort of massively parallel simulated experiments that CNS is intended for. However, Acimovic et al. (2011) does provide methods of characterizing the connectivity of networks that will be useful as ways to analyze the networks generated by CNS.

The software framework selected for this thesis is Brian (Goodman and Brette 2008). Brian is a programming framework, written in Python, which provides methods to configure the model used for individual neurons, assign neurons to populations, and connect the neurons. Brian permits researchers to express the equations for the behavior of a neuron in standard time-derivative or algebraic formulations. The observed biological parameters of the neurons, such as refractory periods, leakage currents, and so forth, can be specified in the simulation as well. Combining this representation of the behavior of an individual neuron with connection model generated by simulator results in an executable simulation which can emulate the behavior of a large collection of neurons. Because Brian supports multiple different cell models, the model parameters may be changed to alter the behavior of the network.

Brian does not explicitly model the location and shape of cells in the way that CX3D or NETMORPH do. However, Brian does not need to be concerned with the shape or location of the cells as long as the rest of the simulation handles those aspects of the network that are affected by cell placement. Brian's implementation as a framework for a programming language, rather than a program of its own, permits the developer to bring the simulation of plating, cell growth, and execution together into one program, rather than using a different modeling program and simulation environment for each stage. Prior to developing CNS, the author of this thesis had written simulations in Brian that model cell locations as a grid, based on the neural networks used in Natschlaeger, Maas, and Markram (Natschlaeger et al. 2002). Using such a model allows for easy calculation of distance between cells and so their interaction.

Brian is also not tied to any specific model of the neuron, but allows the user to define the equations and functions that govern the behavior of the neuron. The ability to define new models easily allows the user to choose an appropriate trade-

off between computational speed and behavioral fidelity without having to switch simulation environments. It is possible to implement a compartmental model in Brian, or use a simpler model to save computation time.

2.2 Other MEA Plating Simulators

The MEA model created by Kahng et al. (2007) accounts for the distribution of cells based on a patterned substrate in the MEA. The patterned substrate provides areas where the neurons can stick to the plate, separated by areas where they cannot adhere. The simulation in Kahng et al. is intended to capture changes in the behavior of the network based on differences in the line widths of the pattern of cell-adhesive proteins in the culture dish.

Kahng et al. represent the location of the cells using a square grid, where each location on the grid may or may not have a cell on it. The pattern of cell-adhesive proteins can be expressed by indicating whether each grid location is coated with protein or not. The presence of adhesive protein in a grid cell affects the probability that a cell will be located there.

In the interest of performance, the firing model used by Kahng's group is a leaky integrate-and-fire (LIF) model with a Poisson-distribution spike generation process. Even with this simplified firing model, the simulation displays burst activity like that observed in real dishes, so it is possible that more complex models are not needed to achieve realistic behavior. However, while a LIF model produces activity with spike timing similar to a biological network, it does not produce output voltages like those observed in a biological network. More specifically, voltage at the membrane of each simulated cell does not have fluctuations under the firing threshold, which a real cell does have, and it does not produce an output spike like a real cell. Instead,

the voltage merely resets, and the cell is considered to have "fired" at that point in time.

Kahng et al. do not report whether varying the density of connections between the simulated cells varies the prevalence of epileptiform activity and transition to regulated firing as reported in real cells (Shea 2009).

SIMONE (Statistical sIMulation Of Neuronal networks Engine) is a model of both biological neurons and the electrodes used to sample data from them. SIMONE includes a more complex model of the activity of the neuron and the electrical characteristics of the contacts in the MEA than Kahng's paper (Escol, Pouzat, Chaffiol, Yvert, Magnin, and Guillemaud 2008). SIMONE uses integrate-and-fire neurons, like Kahng's team's simulator, but only for calculating spike timing. The actual spike waveforms are based on templates, in order to more accurately approach the output of real cells. The use of a model for the electrode's electrical characteristics and influence on the spikes allows SIMONE to be relatively computationally light-weight, while still producing electrically correct output.

SIMONE allows the user to provide statistical distributions to govern the placement of cells in the simulated MEA and the connectivity between them, as well as the distributions of many properties of the neurons themselves. However, SIMONE does not appear to provide any mechanism to modify the model used to calculate the spiking behavior of the cell. The model is leaky integrate-and-fire (LIF), and no provision is made for replacing it with another model. The LIF model was chosen because it provides similar spike timings to biological cells with minimal computational cost. Despite this decision, SIMONE's authors also call it out as unable to "quickly simulate large network synchronization." Given that computers increase in power (or decrease in cost for a given power) rapidly, it is useful to include the ability

to transition to a more complex model, if either the simulation calls for it or the available technology supports it.

The use of the Brian framework in CNS allows the user to modify the model used to simulate neurons. During the development of CNS, this ability was important. Initial versions of CNS used a LIF neuron model, which, as described above, produces realistic spike timing, but not realistic spike voltage. When it became apparent that a simple LIF model was not acceptable, the process of converting CNS to use a more sophisticated model did not require extensive modifications.

CHAPTER 3

METHODOLOGY

3.1 Development

The goal of this research was to create a software simulation of a MEA. In order to simulate a full MEA, the system must model the dispersal of cells over the surface of the MEA, the networking of those cells, and their activity. The first part of the simulation is deciding the distribution of the cells over an area according to the density of the desired culture and the surface area of the MEA plate. The process of determining the cell locations is called "plating." After the plating simulation has placed the cells, a growth simulation uses the locations of the cells to determine how the individual neurons are connected to form the network. In order to decide which neurons are connected, mathematical models based on the observed networking behavior of real neurons are used. The plating and growth simulation have been written and several iterations of testing and further development have been performed to bring the output of the simulation into line with the observed behavior of biological neural networks. The resulting body of software is called CNS, for "Cultured Neuron Simulator."

The output of the plating and growth simulations is the connectivity map of a network. In order to be useful for simulation of real networks, the connectivity of the simulated network must be "biologically plausible"; that is, it must match the parameters of biological networks as observed in culture.

In order to compare the output of a simulated network to that of a real network, automated tools were written to operate on the data created by the simulator. When the simulator is executed, it stores its output data in the same format used by the biology department to collect activity data from living cultures. Using the same data format allows the development of tools to compare the activity of two networks. Common formats allow the tools to work the same way on both simulated and biological data sets. The tools are used to confirm the utility of the model for simulating biological networks. Discrepancies between the results of applying analysis tools to data derived from simulated and biological networks indicate shortcomings in the simulation, and so ways that the model can be improved.

The automated tools indicated that the behavior of the simulation as it was initially written had substantial differences from the behavior of biological cultures. Once the cause of the difference was understood, development was performed in iterations. Each iteration consisted of analysis of the data from previous runs, modification of the simulation, debugging, and execution of the modified simulation to produce data for the next iteration. As the simulator was developed, the tools were re-run on each new version to determine if the most recent modifications had increased or decreased the ability of CNS to accurately model the behavior of biological cells. The first iteration used LIF neurons, but the data from that iteration indicated that the measured cell voltages were implausible. The next iteration used the Izhikevich neuron model, but the resulting simulation was too active, in comparison to real cultures. The final iteration modified the combination of neuron firing patterns in the simulation to match the expected behavior of the types in a biological cortical culture, and produced results in line with the results from biological neurons.

3.2 Neuron models

For initial development, the cell model used was a simple leaky integrate-and-fire (LIF) model. The LIF model was chosen because it is computationally lightweight and displays a sufficient degree of similarity to real neurons to be used in simulation. Both Kahng et al. (2007), and Jolivet, Lewis, and Gerstner (2004) indicate that a LIF model can approximate the spike timing of a living neural network or a more complex mathematical representation of a neuron, to a high degree of accuracy. The particular model may have to depend on the cell, but Ostojic, Brunel, and Hakim (2009) indicates that exponential integrate-and-fire models offer a good match for the behavior of pyramidal cells *in vitro*, so, again, integrate-and-fire models may be both sufficiently accurate and computationally tractable. The Jolivet et al. paper also calls out specific portions of the simulated cells' signaling behavior that the LIF model does not accurately capture, so effort in improving the model may be focused in these areas (Jolivet, Lewis, and Gerstner 2004).

However, it should be noted that accuracy of the output of a simulated neuron can be examined for accuracy from the standpoint of timing, or by comparison of the actual electrical output. From a timing point of view, a simulated neuron is an accurate representation if the spike output of the neuron matches the conditions and timing that would elicit spikes from a biological neuron. The electrical signal has elements, particularly low-amplitude variations, that are not duplicated by LIF models, but can be approximated by more complex models. In this case, not only does the timing of spikes have to be correct, but the voltage output of the neuron must match that of real neurons.

The LIF model is good for modeling the timing of signals between neurons, but it does not produce biologically plausible action potentials. In a biological neuron, the action potential is a sudden spike that appears when the neuron reaches its

firing threshold. When they reach their firing threshold, LIF neurons simply reset to their rest potential without producing a voltage spike. The point when they reset is regarded as a spike event by Brian, but is not a spike in the sense of the neuron producing an elevated voltage.

The result of this lack of spike voltage is that when analyzed by the same tools used to determine the inter-spike intervals in biological networks, the simulation data appears to have relatively few spikes. The lack of spikes results in the data used by one of the analysis scripts being very sparse, which in turn appears as unrealistically large values in the output of the analysis scripts, as detailed further in the results section of this thesis.

The lack of biologically plausible spike voltage in the signal is a problem with any simulation that uses LIF neurons. LIF neurons are useful, however, because they are not computationally intensive to simulate, which enables scaling to large networks. The SIMONE simulator uses LIF neurons, but gets around the lack of spike voltage by adding a biologically plausible voltage spike to the signal when the LIF neuron fires (Escol, Pouzat, Chaffiol, Yvert, Magnin, and Guillemaud 2008).

To obtain more realistic spike voltages, CNS was converted to use an Izhikevich 2-D integrate and fire model instead of basic LIF neurons (Izhikevich 2003). The Izhikevich model has parameters that can be configured to reproduce the behavior of many biological neurons. Intrinsically bursting neurons were selected, as that behavior is found on in the pyramidal neurons of rat visual cortices. Changing the behavior of the Izhikevich model neurons is a matter of altering four parameters. If it proved possible to quantify the proportion of each type of firing behavior present in the dish, the simulation could be changed to duplicate those proportions.

3.3 Plating simulation

In this work, the layout of the simulated cells is simplified into a planar grid. In a typical MEA, the cells are plated on glass prepared with binding proteins, allowed to bond, and then washed, so any cells that are not in contact with the glass are removed. As a result, all of the cells in the culture are in a single layer on the glass of the MEA. Each square of the simulated grid is approximately the size of a single neuron cell body (30μm), and the full grid is 2500μm square. These parameters are configurable in the simulation software, to support different types of cells or configurations of MEA.

Cells are distributed on the grid according to a midpoint displacement fractal algorithm (Fournier, Fussell, and Carpenter 1982). Kahng et al. (2007) does not provide details on their model of cell distribution beyond a description of the plating process. For CNS, a midpoint displacement fractal was chosen to set the distribution of cell adhesion probabilities because of the similarity of its results to turbulent flows. The uneven distribution of cells in dishes is supported by the uneven areal density of cultures as seen in Shea (2009). The plasma fractal provides a real-valued probability of each point on the dish being occupied by a cell.

As an alternative to the plasma fractal, CNS also allows the use of an image to specify the cell occupancy probabilities. The red channel of the image is mapped to the grid of points on the dish, with the saturation of color at each point used as the probability of that point containing a cell. The image can contain stripes or other patterns, which can be used to simulate micropatterning of the adhesive protein substrate of the dish, or micropipetting of the suspended neurons to specific locations in the dish.

After the probabilities are determined, using either an image or a plasma fractal, the plating simulation then marks each location as occupied or not, based

on the probabilities of a location having a cell and the density of cells in the plating solution. Those locations that are marked as occupied are treated as having a cell on them. The others are assumed to be empty space.

After the cell locations are determined, there are a series of pruning steps that are intended to simulate cell deaths in the culture. Not all of the cells from the initial plating survive to maturity. In biological cultures, 45-60% of the cells die before the network is done wiring itself, within approximately the first 17 DIV (Erickson, Tooker, Tai, and Pine 2008). Because so many of the cells die off, they do not need to be considered when the simulation begins to determine network connectivity. In order to model the early cell mortality, the locations that the simulator has marked as occupied are decimated based on the observed survival probabilities of cells in culture. The survival rate, expressed as a percentage, is a configurable parameter of the simulator. At present, the pruning function assumes that all cells are equally likely to die, but this function could be updated to bias cell survival rates in a number of ways, such as making cells that are near other cells more likely to survive. Obviously, such biases should be supported by observation of biological neuronal networks.

3.4 Connectivity and Growth

Axons may be of any length. In humans, the sciatic axon reaches from the base of the spine to the big toe, nearly a meter. Because the active area of an MEA is around 2mm^2, it is possible for any cell to be connected to any other cell, resulting in $N \times (N-1)$ possible connections among N cells. Typical cell suspension densities are in the range of 300-2000 cells/mm^2, resulting in 1200-8000 cells in the active area of the MEA and so millions of possible connections (Wagenaar, Pine, and Potter 2006). However, there are limits on cell growth and networking which make the computation of the network connectivity more tractable. Chemical interactions between cells restrict the

number of connections that should be considered when developing the connectivity of the dish.

Kahng, Nam, and Lee (2007) provides a model based on observation of chemotaxis in developing neurons, but simplified into a stochastic model. The growing end of an axon moves in a random walk on a grid. Each step may take it in any of 8 directions: up, down, left, right, or the four diagonals. If, after making a step, the walking point is with 20μm of a dendrite of another neuron, the two neurons are considered connected. The paper indicates that the probability of a connection between two cells is effectively a function of the distance between them which makes it unlikely a cell will connect to itself, but likely it will connect to neighbors, and unlikely that it will reach very far (Segev and Ben-Jacob 2000).

Gafarov (2006) suggests that the spiking activity of developing neurons also causes the release of the chemicals that will attract developing axons towards a cell. The effect of these chemicals on the developing axons is interesting because it proposes a function for epileptiform activity. The high frequency signaling in early development would cause higher release of attractive chemicals, and so drive the formation of early connectivity. Stimulation also increases activity of the stimulated neurons, and so the development of connectivity in the growing culture can be guided by stimulation. Specifically, a specific spatial pattern of stimulation will encourage growth around the neurons activated by the stimulus, thus driving an increase in connectivity to those neurons (Zemianek, Lee, Guaraldi, and Shea 2012).

CNS uses a Gaussian distribution to model the probability of a pair of cells connecting based on the straight-line distance between them, with parameters set to maximize connectivity around 200μm from the cell body. The Gaussian distribution also provides a limitation on the number of possible connections that must be considered by the program during the growth simulation. If the distance between two cells

is so large that the probability of a connection between them is vanishingly small, it may be disregarded when the network is being laid out, thus saving computation time.

In addition to the limits imposed by chemotaxis, observations of the connections in MEAs performed by confocal microscopy indicate that only 20-50% of the possible connections are made. Once some connectivity threshold has been reached, a neuron will not connect to any other neurons. The restriction on connectivity sets an upper limit on the out-degree of neurons, that is, the number of other neurons that they form connections to, considered from the perspective of the connecting neuron. A number of possible limitations on the out-degree of neurons have been found in the literature.

Segev et al. (2003) indicates that cultured neurons tend to send out approximately 10 neurites to connect to other cells. The number of outgoing connections provides a good upper bound on the number of connections made by simulated cells, but it seems unrealistic to assume that simply stopping at 10 connections will build a biologically plausible network.

Patel, Scott, and Meaney (2012) indicate that the out-degree of neurons can be modeled using a Poisson distribution with a mean of 22. Allowing such a stochastic distribution of connectivity will cause some neurons to be extremely well connected while others are less connected. Such patterns of connectivity are seen in biological cultures, so the Poisson distribution is used in the simulation to set the out-degree of the neurons. Once a neuron forms a number of connections equal to its selected out-degree, no further connections from that neuron are considered, although other neurons may still form connections to it.

One factor that is not considered by CNS is the ability of neurons to migrate. Individual neurons, at least during the first several DIV, are capable of moving around

the MEA to form clusters (Segev, Benveniste, Shapira, Ben-Jacob, et al. 2003). The formation of these clusters occurs at the same time as the transition of the culture to a pattern of activity dominated by synchronized bursts. The activity of the network is an emergent property of its connectivity (Fuchs, Ayali, Ben-Jacob, and Boccaletti 2009). It is likely that such a clustered network displays a small-world topology, with the neurons in each cluster densely connected, and only a few connections from cluster to cluster. The clusters form in cultures with greater than 10,000 cells/mm^2, which is an extremely high plating density. Clustering does not require support from glial cells or electrical signaling, and so is probably a chemically and mechanically mediated emergent property of neurons (Segev, Benveniste, Shapira, Ben-Jacob, et al. 2003). Segev et al. (2003) provides a model for the mechanical forces on migrating neurons, should this functionality be required for future simulation work.

Once the model is completed, it may be executed, and voltage and spike train data collected from it. Since each neuron is in a known location in the simulated culture, the simulation selects the neurons located on or near the conductive pads for a given MEA layout, and records data from those neurons. The Brian simulation software also supports logging of potentials of arbitrary individual neurons as well as collection of spike data from any neuron or set of neurons, including the the possibility of logging the membrane voltage of every neuron in the entire simulated population. Biological cultures do not support this level of logging detail.

3.5 Spike and Burst Characterization Tools

In order to assess the similarity of the simulated network to a real MEA, tools must be created to quantify the activity of both networks. When these tools are applied to both real and simulated networks, the results must be sufficiently similar. Activity is measured in terms of spike signals from cells, which may be observed as individual

spikes or grouped into bursts of spikes. To measure similarity between the output of CNS and real neurons, the output of the simulation is recorded and spikes and bursts are counted, as well as the timing of the intervals between spikes and bursts. These data are compared to recordings from live groups of neurons, using the same metrics. The recorded data is from MEA cultures of disassociated neurons, so it is the sort of behavior that the simulation is intended to capture. If the spike and burst data match, the simulation can be said to produce activity like that of a real neural network.

Kahng's et al.'s simulator has a notion of the location of cells, so by checking locations that conformed to the area of their electrodes, represented by circles 30μm in diameter on a 100μm grid, they could determine which neurons were on top of those electrodes, and so which neuronal signals the pad could detect (Kahng, Nam, and Lee 2007). Those neurons were the ones whose spike trains were analyzed. However, the paper does not discuss the actual spike train data produced from the simulated MEAs, beyond displaying histograms of firing rates and inter-spike intervals for the simulation. These data are not compared with similar statistics from biological simulations.

The SIMONE paper does compare the results of actual cell recordings from a cockroach and execution of SIMONE with the cell parameters configured based on what is known of the neurophysiology of the cockroach. The statistical tests show that the data sets arising from SIMONE and from the real cockroach are very similar (Escol, Pouzat, Chaffiol, Yvert, Magnin, and Guillemaud 2008). The most obvious difference is higher amplitude in the simulated sample, which can be eliminated by increasing the simulated tissue resistance or the simulated sensor input resistance. However, without further testing, it is not possible to determine if the resistances are actually higher than they were configured to be in the simulator, or if this is simply

a workaround that the simulator makes possible. When working with simulators, it is important to confirm that the results are supported by reality.

Wagenaar, DeMarse, and Potter (2005) provide an algorithm for spike detection, which is used in Serra et al. (2008) to count neuronal signals and divide them into spikes and bursts. A spike is is an individual signal of more than three times the root-mean-square (RMS) value of a 5ms window surrounding the signal, separated from other similar signals by 0.7 seconds or more. Bursts are clusters of at least three spikes with less than 0.7 seconds between them.

When analyzing recorded data, it is possible to make assessments based on the entirety of the data, rather than on a small window of it. In the analysis tools, a spike is defined as any signal whose absolute value is greater than three times the standard deviation of the entire recorded data set, once the mean has been subtracted. The Center for Cellular Neurobiology and Neurodegeneration Research (CCNNR) at the University of Massachusetts in Lowell, who provided the recorded data and live cultures for this work, use this criteria for spike detection, so the tools detect the same spikes that the CCNNR does, when used on the same data.

In addition to spikes and bursts, there are clustering algorithms and the Center of Activity Trajectory (CAT) metric as explained in Chao, Bakkum, and Potter (2007). CAT is particularly interesting because it captures the spatial character of the neurons, both in live cultures and simulation. CAT tracings of real neurons display complex, stable, recurring patterns of activity. The simulation should be expected to display similar spatiotemporal complexity. Chao, Bakkum, and Potter (2007) is of particular interest because it relates changes in spatiotemporal activity patterns to training, allowing visualization of the changes in the activity of the dish as it is trained.

Another characterization approach is the cross-correlation functions (CCFs) of the recording sites in the dish. The CCF of two neurons measures how the firing of one neuron affects the firing of the other over a given time lag. When the cells are directly connected, there is significant correlation, but the function is generally more complex than that, and includes data about the overall network connectivity. Beyond simple networks of a few neurons, "the relative impact of direct connections and collective dynamics on cross-correlations ... remains an open issue in realistic networks of spiking neurons," and so CCFs may not produce results that are useful for reading out the connectivity of a network (Ostojic, Brunel, and Hakim 2009). However, by treating each site of the MEA as a "neuron" for the purposes of cross-correlation, CCFs for each pair of recording sites in the MEA could be determined, and used to characterize that culture. A simulated dish that is claimed to be a functional duplicate of a specific culture should display the same CCFs as the culture, between the same pairs of recording sites. It may be possible to create a model that only generates the activity that would produce the appropriate CCF between the recording sites, but does not individually simulate all of the neurons that are not directly recorded. Such a model might run faster than one that simulates all the neurons, but would be difficult to derive, and may lack the explanatory power of a more detailed simulation.

CHAPTER 4

RESULTS

4.1 Simulation of Epilieptiform activity in Random Networks

Before developing CNS, some exploratory models were created to assess various approaches for simulation of large collections of neurons. One of these models was an attempt to simulate the action of bicuculline on biological networks. Bicuculline inhibits inhibitory signaling in biological networks, leaving only excitatory signaling. As a result, the level of activity in the network increases dramatically. Removing the inhibitory connections from a neural network simulated in Brian results in epileptiform behavior much like that of a cultured biological network before inhibitory connections form, or when inhibitory connections are disabled. These results seem to indicate that this form of large scale model is capable of mimicking the behavior of real neurons. The images in Figure 3 were generated using a random network, rather than the approach later developed for the plating simulator. Similarly increased activity is visible in biological cultures, as illustrated in Figure 2.

4.2 Activity Metrics

In order to characterize the activity of neurons near the recording sites, the time intervals between spikes were measured. The inter-spike timing information was used to create spike timing histograms by counting the number of inter-spike intervals of a certain length. The spike timing histograms measure the relationship between the length of an interval and its probability of occurring. Biological networks display

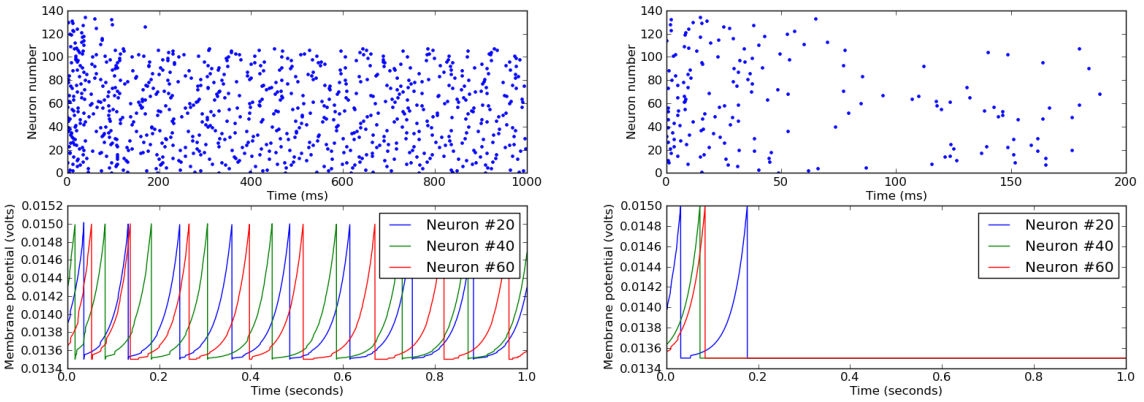

Figure 3: On the left is a graph of the activity of a small simulated network with no inhibitory signaling present. The upper graph is occurrences of signals at each neuron over time, the lower is the membrane voltage of three arbitrarily selected neurons. On the right is the same simulated network, but with inhibitory signaling present. The inhibitory signaling reduces the activity of the network considerably.

a power-law distribution of inter-spike interval lengths, with longer intervals being much rarer than shorter ones. The power-law distribution, and particularly the value of the exponent of the distribution, captures the spread of lengths of inter-spike intervals that the network displays. However, as a metric, the scaling exponent of the inter-spike interval distribution does not capture the frequency of spikes, just the relationship of frequency to occurrence. The same activity, stretched out over a year or compressed into a second, would show the same scaling exponent for its distribution.

The rate at which spikes occur, expressed as spikes over a given time, does not capture the spread of lengths of inter-spike intervals in the way that the scaling exponent of the power law does, but it does capture their frequency, which the power law ignores. Consequently, the combination of scaling exponent and spike rate form a combined metric that captures both aspects of the local, or single-site, spiking activity of the neural network.

In biological data, the scaling exponent varies with the activity level of the culture. Young cultures, displaying epileptiform activity, generally have a higher scaling exponent, which drops to a lower level as the culture matures. The addition of bicuculline to the culture causes a return to epileptiform activity, resulting in a higher scaling exponent.

4.2.1 Exponent of Interspike Interval Distribution

In order to determine the expected values of the scaling exponent and spike rate for cultured neurons, the data from two experiments in cultured neurons was examined. The cultures for each experiment were prepared identically, and used the same equipment to record culture activity. The data collected from the cultures was compared to data collected by running the simulation 50 times, to generate 50 different networks and collect data from each of them. The same script that was used to calculate the scaling exponent of the inter-spike interval was applied to the resulting output files.

4.2.2 Bicuculline Experiment

The first experiment was developed to determine the effect of the inhibitory antagonist bicuculline on developing cultures. Because the action of bicuculline mimics epilepsy, and so increases signaling, it could potentially drive synaptic growth, and so cause the culture to reach a mature state faster (Zemianek, Shultz, Lee, Guaraldi, Yanco, and Shea 2012). The mature state would be detected by examination of the recordings of each culture, produced when it was not under the influence of bicuculline.

The cultures used in this study were in four groups: Control Bicuculline, Control Epilepsy, Post-Epilepsy, and Pre-Epilepsy. The Control Bicuculline group were mature cultures which were exposed to bicuculline, to provide a baseline for the normal reaction of a culture to bicuculline. The Control Epilepsy group consisted

Condition	Mean	Std. Dev.
Control Bicuculline	2.8624	1.5504
Control Epilepsy	2.7772	1.3578
Post-Epilepsy	2.8377	1.5355
Pre-Epilepsy	2.9812	1.5401

Table 1: Mean and Standard Deviation of the inter-spike interval scaling exponent of biological cultures in the bicuculline experiment, grouped by experimental condition. The value is a unitless exponent.

of immature cultures which were allowed to develop normally, to provide a baseline for the normal transition of a culture out of the immature phase of its development. The Post-Epilepsy group were developing cultures that received bicuculline after the initial epileptic activity of the immature phase was complete. The Pre-Epilepsy group consisted of developing cultures that received bicuculline before and during the initial epileptic activity phase. Each group had three cultures in it, for a total of twelve cultures in the experiment.

4.2.3 Stimulation Experiment

The second experiment was intended to measure the long-term effect of stimulation on cultures (Zemianek, Lee, Guaraldi, and Shea 2012). The experiment had three conditions. The Control condition consisted of mature cultures which were not stimulated. The Continuous group received stimulation consisting of a recorded biological signal every thirty minutes, for eight hours a day, for 5 days. The Single group received a single stimulation each day, for five days. When the signals were read from the cultures, they were separated into two further conditions, Normal and Stimulation. In the Normal condition, the electrical activity of the culture is read, without stimulating it, to produce a recording of the spontaneous activity of the culture. In the Stimulation condition, the culture is stimulated, so the recording includes the

Condition	Normal		Stimulation	
	Mean	Std. Dev.	Mean	Std. Dev.
Control	6.7289	9.3068	6.2736	5.9524
Single	4.0961	6.0904	5.5239	4.6489
Continuous	3.4167	4.5725	5.3989	4.2063

Table 2: Mean and Standard Deviation of the inter-spike interval scaling exponent of biological cultures in the stimulation experiment, by experimental condition

stimulation and the resulting reaction from the culture. The data for this experiment is collected in Table 2.

4.2.4 Spike Rate Measurements

Spike rate data was also collected for each site of each culture in both of the biological experiments. The cumulative means and standard deviations of the spike rates over all the biological experimental cases were very similar, as shown in Table 3.

Condition	Mean	Std. Dev.
Control Bicuculline	4.2970	1.6069
Control Epilepsy	4.3432	1.4710
Post-Epilepsy	4.3477	1.7663
Pre-Epilepsy	4.0570	1.5872
Control	5.4279	2.0810
Single	5.3828	2.4163
Continuous	5.3976	2.5362

Table 3: Mean and Standard Deviation of spike rate across all conditions

The spike rates of the stimulation experiment in Table 3 are slightly inflated, as they include both the stimulated culture and the unstimulated culture. As a consequence, some of the recordings include the stimulation signal itself, which gets counted as spikes.

Separating the recordings taken around the stimulation from recordings of the culture's spontaneous activity, as in Table 4, reveals that the cultures which developed without stimulation displayed a higher overall mean spike rate, and higher standard

deviation in spike rate when unstimulated than when stimulated. The stimulation would lead to a large wave of mostly synchronous activity, which reduces the mean spike rate by causing many spikes close together, and reduces the standard deviation by causing those spikes to occur nearly synchronously.

Condition	Normal		Stimulation	
	Mean	Std. Dev.	Mean	Std. Dev.
Control	5.4514	2.1480	5.2477	1.4609
Single	6.1371	2.7850	4.5503	1.5504
Continuous	6.0873	2.9236	4.5757	1.6346

Table 4: Mean and Standard Deviation of spike rate in the stimulation experiment, by condition

The cultures which developed without stimulation were also slower to develop to full maturity, and so they may present a higher spike rate because they remained in the epileptiform signaling phase of their development longer than the other cultures.

4.3 First Development Iteration

The exponent of the inter-spike interval distribution of the simulated neural network (Table 5) is well within the range expected of a biological neural network, when analyzed with the same procedure as was used to analyze the biological cultures. The match between the exponents indicates that the likelihood of an inter-spike interval of a given length follows the same distribution, whether it is in the biological or simulated networks.

The spike rate of the simulated neurons, as shown in Table 6, was both higher than the observed spike rates of biological cultures in either experiment.

Condition	Mean	Std. Dev.
Simulation	3.7590	1.2391

Table 5: Mean and Standard Deviation of scaling exponent of the inter-spike interval of a simulated network.

Condition	Mean	Std. Dev.
Simulation	12.4878	4.9543

Table 6: Mean and Standard Deviation of spike rate in simulation

The difference in spike rates is an artifact of a difference between LIF neuron models and biological cells in the definition of the signal that constitutes a spike. In a LIF model, a neuron is regarded as having spiked when its membrane potential reaches a certain value. However, the model does not produce a biologically plausible output voltage. It merely records that the neuron has fired, and applies the required charge to the membrane of all the connected neurons. As a result, the timing of the spikes is correct, but their voltages are wrong.

4.4 Second Development Iteration

The failure of the LIF model to correctly simulate neuron output voltages is what motivated the transition to the Izhikevich neuron model, as described in the methodology section of this paper. The second iteration of development was the addition of this model, and testing to validate the changes.

In Izhikevich's paper, the user-tunable parameters are called a, b, c, and d. There is also the recovery variable u, and the membrane voltage v. The parameter a is the time scale of the recovery variable, u, which is intended to simulate the activation of potassium (K^+) channels and deactivation of sodium (Na^+) channels after the delivery of an action potential. The parameter b affects the coupling between the membrane potential and the recovery variable u. Higher values make the neuron more likely to have oscillations when v is below the firing threshold. The parameter c is the reset voltage of the neuron after firing, caused by the opening of K^+ channels. The d parameter controls the speed with which the recovery variable resets after firing.

For the second iteration of development, the neuron model used in CNS was initially configured as what Izhikevich refers to as an "Intrinsically Bursting (IB)" neuron. IB neurons produce bursts of spikes when first triggered, which would increase the spike rate by having multiple spikes in quick succession after the threshold is reached. In order to bring CNS into line with the observed biological spike rate, the model for the neurons was converted to "Regular Spiking (RS)", which only spikes once in response to reaching the threshold. Izhikevich's paper calls out RS neurons as being "the most typical neurons in the cortex," so it is reasonable to expect RS neuron signals to dominate in a cortical culture.

The conversion to RS rather than IB neurons was not sufficient to reduce the firing rate of the simulated culture. The resulting simulation had an average firing rate over all sites of 8.2 spikes per second, with a standard deviation of 4.8503. There is some reduction in the spike rate with the change of model, which can be attributed to the lower reset voltage of RS neurons causing them to take longer to return to threshold, and a higher d parameter which will initially slow recharging. The simulation model as a whole still displays a much higher standard deviation than the biological neurons.

4.5 Third Development Iteration

Izhikevich also indicates that while RS neurons likely dominate excitatory firing, inhibitory firing is usually provided by "Fast Spiking (FS)" or "Low Threshold Spiking (LTS)" neurons. Both of these neurons display high firing rates (quick recovery after firing), but since their action is inhibitory, converting the simulated inhibitory neurons to one of these models should result in an overall decrease in spike rates. Research performed by Izhikevich et al. supports this expectation (2004)

The third iteration of the development of CNS was to convert the simulated inhibitory population to FS rather than RS neurons, and confirm that the change resulted in an overall drop in signal rate.

After the inhibitory neuron model was corrected, the average spike rate per site dropped to 6.574 spikes per second, with a standard deviation of 4.3182. The mean of the modified simulation is within one standard deviation of the means of the stimulation experiment conditions, but the standard deviation of the simulation spike rate is still higher than the biological neurons.

The standard deviation of average spike rate would capture the breadth of spread in activity levels between individual culture sites. As expected, some sites show activity levels of around 14 spikes per second, while others have less than two spikes per second. The network does not display the spontaneous activity of a pure LIF network, and so must be driven by an input. In this case, as in Izhikevich, Gally, and Edelman (2004), the stimulation is a Poisson-patterned noisy input applied to all neurons. According to that paper, the frequency of the stimulation input can tune the average firing rate of the network, and the magnitude of the fluctuation in the average firing rate. With low-frequency stimulation, the network should be expected to display an alternation between quiet periods and culture-wide burst activity, leading to a high variance in spike rates. Stimulation at rates higher than 1Hz should result in asynchronous signaling with a near-Poisson distribution. Some of this signaling would be reaction to the input, but most of it would be a result of signals coming in from nearby neurons.

In CNS, varying the stimulation frequency affected the average spike rate, while leaving the standard deviation of the spike rate largely unaffected. However, the alteration of the stimulation frequency was done using a subthreshold input of 10mV. A superthreshold input is one sufficient to cause a spike when it reaches a cell, rather

than just influencing the state of the cell membrane charge. When superthreshold stimulation was used, it affected both the standard deviation of the spike firing rate, as well as the value of the mean spike firing rate.

Freq.	Voltage	Mean	Std. Dev.
0.5Hz	10mV	3.9954	3.1995
1Hz	10mV	6.6165	3.5776
5Hz	10mV	12.6314	3.4670
10Hz	10mV	12.8342	3.5377
0.5Hz	30mV	4.7848	2.9264
1Hz	30mV	6.3712	4.1304
5Hz	30mV	12.0730	7.5294
10Hz	30mV	16.3097	6.9309
0.5Hz	60mV	4.5510	2.4783
1Hz	60mV	6.4734	3.2452
5Hz	60mV	8.6803	7.1438
10Hz	60mV	11.4570	8.0073

Table 7: Mean and Standard Deviation of spike rate with varying stimulation frequency. With subthreshold stimulation, the mean spike rate is altered without affecting the spike rate standard deviation. As stimulation approaches the threshold, the standard deviation of the spike rate is affected.

A stimulation frequency of 0.5Hz and a voltage of 60mV results in a firing rate with a mean and standard deviation very close to that of biological cultures. These parameters were incorporated into CNS to provide a default configuration that closely mimics the firing of biological neurons.

4.6 Performance

The more cells there are, the longer the simulation takes to run. As of the writing of this paper, a simulation of 5,254 cells takes approximately 28 minutes to execute 30 seconds of simulated culture activity, or about one minute for each second of culture activity. One second of simulation per one minute of real time is obviously a much worse runtime than an actual culture, which operates in real time, but the simulation does not require the two weeks of growth time that a real culture would require, and

so is still more efficient. The simulation speed of CNS also compares favorably with the execution speeds of other simulators, as listed in the Related Work section.

The performance of CNS is highly dependent on the configuration of the simulation. The most important factors are the plating density and MEA size, which directly contribute to the number of cells in the recording area. As a result, the easiest way to reduce the runtime of the simulation would be to reduce the number of cells that it is required to simulate. The problem then becomes determining which cells contribute to the recorded activity of the dish, and which can be ignored without affecting the validity of the simulation.

Whether it is necessary to simulate the activity of cells outside of the recording range of the electrodes is still an open question. Cultures were prepared with neurons and adhesive protein only present in the area of the recording electrodes, as well as normal preparation, in order to determine if minimizing the culture area had a detectable effect on the behavior of the culture. If there is no detectable effect, then the area of the culture beyond the recording electrodes does not contribute to the behavior of the culture. Areas which do not contribute to the behavior of the culture can be eliminated from simulation, thus saving considerable processing time. At present, the data recorded from the minimal cultures has not been fully analyzed, so the validity of this approach is unknown.

4.7 Current State of the Simulation

As of this writing, the iterations of modification to the neuron model to bring it into line with the actual neurons present in cultures have produced a model with an inter-spike distribution scaling exponent and a spike rate that match the observed values in biological networks.

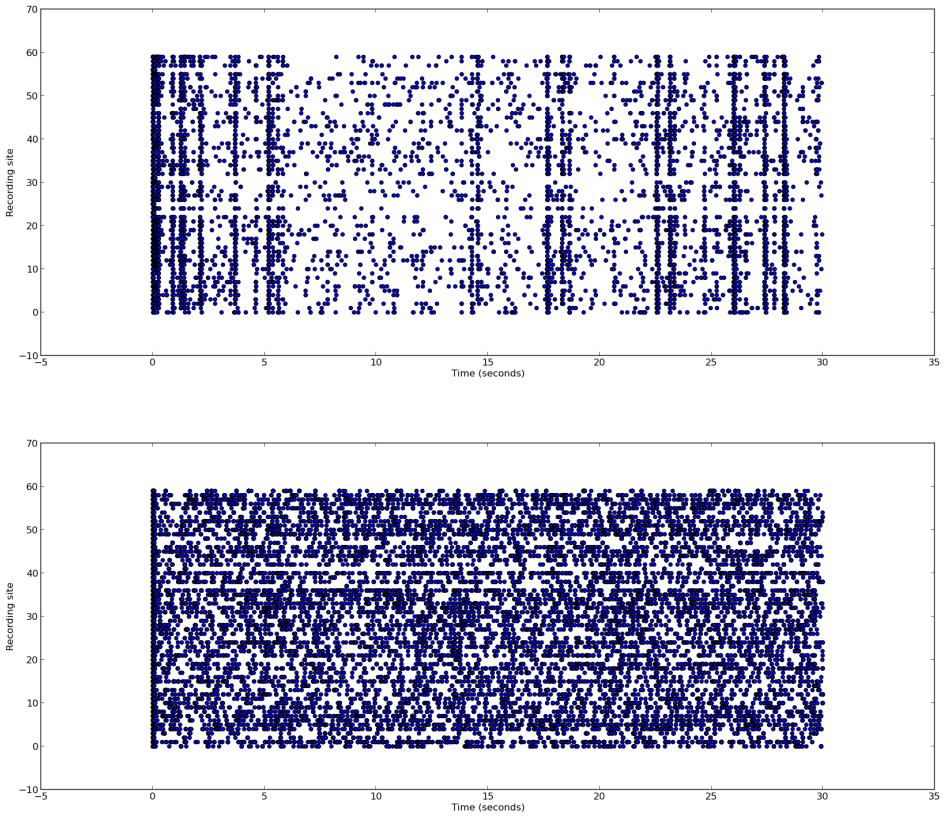

Figure 4: The upper image is electrical activity measured from a biological network. Each point is a spike. Note the presence of strong vertical bands, which indicate the near-simultaneous firing of many neurons. The lower image shows the output of the current simulator. Note the absence of strong vertical banding as was seen in the biological culture.

However, these metrics only capture the behavior of single points within the simulated culture. They do not adequately measure the relationship between sites within the culture, and this overall network effect is important for accurate simulation of the biological cultures.

Figure 4 shows the output of the simulator and a biological signal. The vertical bands which are present in the biological signal are absent in the simulated signal. There are two possible explanations for this difference.

The configuration of the simulator may have overly weak values for the synaptic strength of inter-neuron connections. If the cause of the problem is synaptic strength settings, the likely problem is that the current configured strength of a synapse is too low. With weak connections, each neuron does not exert enough influence on its neighbors to bring them into synchrony, and so the firing is out of synchrony. As connections get stronger, the tendency of neurons to fire together increases, which appears as vertical banding in the output. Once the connections become too strong, any neuron firing immediately triggers all the other neurons to fire, and so the culture is dominated by lock-step, culture-wide firing. In order to determine the appropriate strength, values from the literature on synaptic connections will be used to set the default strength, and the resulting output will be examined for banding.

The failure to display synchronous firing may be caused by the influence of noise on what the simulation considers a "spike.". That is to say, the pattern may be present, but masked by weaker signals that are erroneously regarded as spikes.

There is noise present in the apparatus used to collect data from the biological culture. Some of the noise is removed by a filter implemented in LabView. The filter is a Bessel bandpass filter with an upper cutoff of 200Hz and a lower cutoff of 0.5Hz. However, any noise in that band still gets through. A recording of a dish of buffer solution with no neurons in it produced a signal that consists entirely of this noise, with no neuronal signals.

The noise in the system increases the standard deviation of the voltage over the length of a recording. A spike is defined as a sample greater than three times the standard deviation of the signal, minus the mean of the signal. Since the standard deviation of the simulated signal does not include noise, it is quite low, and the

threshold that must be met for a sample to qualify as a spike is also low, and so any regular activity may be masked by a large number of spurious spikes.

There is also error present in the signal caused by the settling time of the analog circuits in response to transients. The initial 500μSec of the signal are offset by as much as a volt from the baseline of the signal due to the transient response of the amplifiers as they come online. This distortion can have a strong effect on the statistical properties of the signal unless the early samples are discarded.

In order to determine if the distortions inherent in the recording due to noise and transient response are affecting the simulation, an option will be added to the simulator to also simulate the transient response. The simulated transient response will be added to the simulated recording data, to imitate the effects of the recording hardware on the resulting recording.

CHAPTER 5

FUTURE WORK

5.1 Topological Characterization Tools

The metrics currently in use for assessing the similarity of the simulation produced by CNS to a biological culture are focused on spike and burst timing. Spike and burst timing will allow comparison of the simulated network to a real network in terms of activity, but not in terms of topology. In order to characterize the connective topology of the network, metrics from graph theory could be used, including some of those related to small-world networks.

A small-world network is one characterized by most of the inter-node connections being local, but some "shortcuts" also existing which link non-local nodes. The shortcuts result in a network where where any two nodes are unlikely to be directly connected, but highly likely to have a short hop distance between them. Determining the shortcut density of biological neural networks in comparison to simulated networks may help illuminate the sources of complex network activity. The simplest metric of shortcut density is measuring the ratio of the diameter of the network, that is, the greatest distance between any two points, and the number of nodes of the network. A network with many shortcuts will have a low diameter despite a large size, and so this ratio will be low. A shortcut-rich pattern of connections arises from the distance-based model of neuron connectivity, where local connections are favored, probabilistically, but longer connections are still possible. The longer connections act as shortcuts between the densely-connected regions created by local connections.

More specifically, small-world networks have a characteristic path length closer to that of a random network than a regularly-connected network. However, small-world networks also have a clustering coefficient, as defined by Watts and Strogatz (1998), closer to that of a regularly connected network than a random network. These networks are seen in the organization of biological neural networks and in cultured neural networks (Shefi, Golding, Segev, Ben-Jacob, and Ayali 2002; Esposti and Signorini 2008a; Roxin, Riecke, and Solla 2004).

In order to determine if the network connectivity of the simulated network matches that of real networks, a metric that can be used to compare them must be determined and applied. The density of shortcuts as a fraction of total connectivity has strong effects on the ability of the network to sustain periodic patterns of activity, so it would allow networks to be quantified with regard to their ability to support periodic patterns. Loops in small-world networks can result in patterns of activity that repeat over time, and can be elicited by stimulating a number of neurons that is relatively small compared to the total population (Roxin, Riecke, and Solla 2004). Similarly, a small number of inputs (relative to the number of cells in the network) can destabilize the looping pattern or change it. Periodicity of activity has been observed in biological neural networks, so it is likely that this metric supports a useful comparison (Rolston, Wagenaar, and Potter 2007).

Acimovic et al. (2011) characterizes the connectivity of networks using two other metrics. The first is prevalence of motifs, which are the possible connectivity patterns between a set of three neurons. The second is in-degree distribution, which is the distribution of neurons for each possible number of incoming connections from other neurons.

Motifs provide a way to distinguish the topology of a network from a random network. A random network would be expected to show a distribution of motifs

largely based on the ratio of connected edges to nodes, while a non-random network would show some other distribution. By producing random networks as well as non-random ones, the two distributions can be compared to determine how closely the non-random network resembles a random one.

In-degree distribution conveys information about network density as well as network topology. A very densely connected network will have an in-degree distribution with most of the neurons having a high in-degree, while a sparse network will have most of the neurons possessing a low in-degree. A small-world network, on the other hand, will show neither of these patterns. Because a small-world network has several very highly connected "hubs" providing short path length from any node to any other node, and many nodes with few connections, a small-world network would be expected to have a distribution of in-degree matching this configuration. That is, there would be relatively few nodes with a very high in-degree, while all the rest of the nodes would have a lower in-degree.

5.2 Difficulties with Topological Models

Kahng et al. (2007) highlights the difficulty of completely determining the total synaptic connectivity of the neurons in the array, a problem that simulations do not share. "Total" synaptic connectivity is different from local synaptic connectivity. Local connectivity measures include mapping all the connections of one cell to others, or characterizing the statistical distribution of connectivity patterns likely to occur among neurons. Total connectivity is the production of a map of every single neuron-to-neuron connection in the culture, also known as a "connectome," by analogy with "genome." For a simulation, this connectivity is simple to determine and analyze. For a biological culture, it is currently not possible to determine the total connectivity of the culture.

One method of determining the connectivity of a single neuron in culture is to infect the neuron with a transgenic rabies virus that carries a genetic code for red and green fluorescent proteins, rather than rabies viral coat proteins. As a result, the neuron does not produce more rabies viruses. Instead, the fluorescent proteins are expressed in the synapses of the infected neuron, with the red fluorescence appearing at inhibitory connections and green at excitatory connections. A confocal microscope can then be used to examine the tagged synapses. Since the neurons grow in a single layer as a result of the MEA plating process, it is easier to see the connections between them than it would be in a three-dimensional tissue mass. In order to automate the process of converting from images of fluorescing synapses to a connectivity map that can be used in simulation, computer vision techniques may be used to detect synapses and their connections to neurons. However, the transgenic rabies virus does not spread to all the neurons in the culture, so this approach is limited to imaging single neurons. Since versions of rabies that can spread do so by eventually destroying their host cell, a version of this process that uses viral spread to mark more neurons would be doomed by its own success.

Another approach to determining the connectivity of the neurons is the use of statistical measures on the movement of spike signals through the network. Work with caged-neuron MEAs has provided a statistical basis for determining whether the path taken by a signal traverses single or multiple neurons (Erickson, Tooker, Tai, and Pine 2008). Sorting connections by whether they traverse single or multiple neurons is not sufficient to determine the connectivity of the network, but it may be able to be used as a metric to verify overall statistical similarity. It may also be possible, with further research, to refine the measurement to more accurately determine the number of neurons in a path and their connectivity. Determining the connectivity of

existing biological neural networks can also validate the output of simulated plating and cell growth models, as described in Kahng et al. (2007).

Due to the problems with determining the total connectivity of a biological network to a degree of accuracy sufficient to make comparisons to a simulated network, topological analysis tools were not used in the initial development of the simulator. Instead, information about the known local topology of neurons in culture, such as the maximum connection distance, out-degree distribution, and so on, were used to guide the development of the simulator. In this way, the simulated culture can at least be known to conform to what can be known about the connectivity of biological cultures.

5.3 Duplicating Specific Networks

In normal use, the simulated networks generated by CNS are intended to have the same sort of connectivity, at a global level, as biological networks constructed using the same parameters, but do not match a specific existing network. If the designer of the network wishes to have complete control over the simulated network's connectivity, they may configure each individual connection. Configuring the network by specifying each connection is more labor-intensive for the programmer, but allows the flexibility to have any specific connectivity model which the user supplies. Furthermore, this mode of setting connections allows the user to attempt to duplicate the observed connectivity of a biological neural network, which is a first step towards duplicating the functionality of a specific neural network. Before a network can be duplicated, the connectivity of the biological network must be determined. As previously discussed, current methods of assessing the connectivity of a biological culture are insufficient to produce a complete map of the connections of the culture.

5.4 Hebbian Learning and Oja's rule

Hebbian learning is a mechanism for controlling the strength of connections between neurons based on whether activity from one neuron causes the other to fire. The colloquial statement of the rule is that "neurons that fire together wire together," that is, the synapses that connect them are strengthened. Hebbian learning does not put an upper bound on synaptic strength, so over extended periods, the connection weights grow very large. Oja's rule provides a method of countering this, so the network can "forget" associations and does not become over-trained.

Repeating patterns could be viewed as a form of active memory storage, where the network keeps a representation of some stimulation pattern in active processing, similar to trying to keep a list in mind by reciting it over and over (Rolston, Wagenaar, and Potter 2007). Similar looping patterns are seen in simple circuits of neurons called Central Pattern Generators, or CPGs. These networks are suggested as a source of timing signals for animal locomotion. By increasing the synaptic strength of the connections within a CPG, the resulting gait could become more stable, that is, resistant to perturbation into another gait. The patterns of repeated activity seen in Roxin, Riecke, and Solla (2004), when occurring in a network with variable connection weights under a learning rule such as Oja's rule, may serve well as a model of memorization based on long-term potentiation. The network can receive a stimulus that sets up a repeating pattern, and each repetition of the pattern acts as a "rehearsal," further strengthening the connections that support that pattern. These recurring patterns of activity may in turn have a part in learning and memory, as well as more abstract mental structures such as thoughts and beliefs (Madhavan, Chao, and Potter 2007; Sharp 2011). New stimuli may extinguish or modify the pattern, but the strengthened connections will remain, to make recognition of that pattern, or in the case of CPGs, adoption of that gait, more likely in the future. Oja's

rule can contribute to the weakening of the connections that form the pattern over time. Strengthening the connections that lead away from the neurons that support the pattern may also contribute to diminishing the probability of activity remaining in the pattern over long periods (Sharp 2011). In a simulated culture, the weights of individual connections can be tuned, so a ring or loop could be set up to act as a pattern generator, and then various methods of enhancing or diminishing that pattern could be tested.

Limited connectivity and inhibition are important to sustain repeating, patterned activity. In models with too many long-distance connections, the excess connectivity suppresses repeating signals, because the signal returns to the location it started from before those cells have time to rest and prepare to fire (Roxin, Riecke, and Solla 2004). If the cell energy reserves are exhausted by firing too often in a short period, the signal dies out. Inhibitory connections may also prevent excessive firing in the cells, allowing them to reserve their energy.

Chao, Bakkum, and Potter (Chao et al. 2008a) shows that some degree of random background noise is required to for a simulated MEA culture to display learning. It may be that this noise is required to prevent the increase in associative strength caused by Hebbian learning from resulting in attractors so strong that the organism develops irresistible automatic responses to a given input, rather than being able to shape its response to a stimuli. That is, without some degree of perturbation, a stimulus-response pair could become so ingrained as to make it impossible to react in any other way, or alter that response in the future. Perturbation from external sources may also play the role of afferent signals, that is, signals that would be coming into the brain from the nervous system throughout the rest of the body in an intact animal. Feedback-controlled stimulation at random sites on the MEA resulted in reduction of bursting across the entire culture, but did not prevent the culture from responding

to stimuli (Wagenaar, Madhavan, Pine, and Potter 2005). However, it may not be desirable to suppress bursts completely, as they may carry information or represent the activity of stable, information-storing attractors in the network (Madhavan, Chao, and Potter 2007). Combining the plasticity of the reaction to tetanus from Madhavan et al. (2007) with the burst-suppressing feedback system of Wagenaar et al. (2005) may present a way to "program" a culture to react to specific input with a specific form of burst activity. Such a system would use a combination of tetanizing input to shape burst patterns and a feedback-moderated stimulation system to eliminate undesired bursts.

A functional simulation of a MEA could be configured to demonstrate Hebbian learning, and the parameters of the learning and perturbing functions set to evoke different responses. Tweaking the parameters in this way could result not only in demonstrations of practical mechanisms for learning, but also suggestions of ways that learning can fail. For example, a system with insufficient perturbation might learn very quickly, at the expense of easily overtraining. In fact, the importance of continuous random perturbation to maintain plasticity, and so aid in learning has been demonstrated in simulation already (Chao, Bakkum, and Potter 2008b). Similarly, a system with a weak influence from Hebbian learning might have to continually discover new solutions to training problems, rather than retaining old ones.

CHAPTER 6

CONCLUSIONS

6.1 Plating and Growth Simulation

The primary contribution of this project is the development of CNS, the program which generates a connectivity map of a simulated culture and simulates the activity of that culture.

CNS has a wide array of configuration parameters, which are used to set the properties of the neurons, MEA plate, and the plating and growth of the neurons. The parameters are set based on observed properties of biological cultures and the measurements of the MEAs used by the Center for Cellular Neurobiology and Neurodegeneration Research.

The individual neuron parameters as listed in CNS's configuration file control the size of the neurons, the limits on axon length, and the in and out degrees of individual neurons. The spatial parameters are all expressed in micrometers. At present, the model used to simulate the activity of the neurons is part of the code of CNS, rather than an independent module. The code of CNS can be edited by anyone familiar with the Python programming language, but for future development, the parameters of the activity model will be moved to the configuration file.

Because Brian supports easy changes to the cell model used to determine the behavior of the network, modifications, such as adding or removing long-term potentiation or using a more complicated model of the behavior of the cell under stimulation, are fairly easy to accomplish. These changes do not alter the connectivity

of the network, so it should be possible to test which aspects of the behavior of the network arise from connectivity, and which require specific elements of the cell model to be in place to occur.

The parameters of the MEA are also configurable. The current configuration matches that of MultiChannel Systems MEA plates, which have an 8 by 8 grid of electrodes, each 20μm across and spaced 200μm apart. The corner electrodes are not present, leaving 60 contact points. In CNS's configuration file, the configuration is described by setting the number of rows and columns of pads to be 8, setting the spacing and pad diameter parameters to 200 and 20, respectively, and setting a boolean parameter to inform the simulator that the corner pads are not present. The pad parameters also include a limit for how close a neuron must be to a pad to be recorded by that pad. Increasing this parameter means the pad will receive more signals, but it also increases the time required to calculate the voltage at that pad, and so slows the simulation.

The only plating parameters are the density of the cells in solution, and an optional image depicting the desired cell adhesion probabilities. The density is expressed in cells per millimeter^2, as they are in the literature, so copying this parameter from a paper is straightforward. The image is resized so that each pixel corresponds to a potential neuron location, and the value of that pixel's red channel is used to set the probability of a neuron attaching at that location. Fully saturated red pixels get a 100% chance of having a cell attach, but the cell may be removed later when the cells are reduced to match the density of cells in solution.

The final set of parameters is the growth parameters. There is only one growth parameter: the percentage of the initially plated cells that survive to maturity. In culture, only about half of the cells survive from plating to the formation of the mature network. The remaining cells die off. CNS removes cells from the plating

simulation to match the configured survival rate. There was a growth parameter that governed the percentage of the possible connections between neurons that actually formed. This parameter was replaced by a stochastic limit on in- and out-degree of neurons, based on research by Patel, Ventre, and Meany (2012).

6.2 Electrical Modeling of MEA Sites

Because the simulation produced by CNS is intended to model the behavior of neurons in culture, the simulation also includes aspects of the culture dish and data acquisition equipment used to monitor biological cultures. The recording sites within the culture dish are in fixed positions, and can only monitor those neurons on or close to the recording sites. CNS models this by determining the weighted sum of the voltages for the neurons around a recording site. The weights are based on the distance of each neuron from the recording site, so those neurons closest to the site contribute the most to the voltage at that locations, while more distant neurons contribute less.

6.3 Analysis Tools

In order to compare the activity of the biological networks to the output of CNS, a set of visualization and analysis tools were created. The analysis tools consume data in the same format used by the biology lab, and so they are also useful for visualizing and processing data generated from biological cultures.

Many of the scripts simply graph the voltages or spike occurrences over time in a recording of a biological or simulated culture. The resulting images allow for a visual check of approximate similarity, by letting a user look at the resulting images to determine if they are similar. Visual overviews make it simple and quick to detect large errors in the configuration of the simulation, but do not constitute a rigorous test of similarity between the simulation and biological cells.

In order to obtain more useful values for comparison, a script to calculate the power law exponent and spike rate from recorded data was written. The spike rate is simply calculated by counting spikes and dividing by the length of the recording in seconds, to obtain the spike rate in spikes per second. The power law exponent of inter-spike intervals is estimated using a combination of maximum-likelihood fitting and goodness-of-fit tests (Clauset, Shalizi, and Newman 2009). Unfortunately, this method does not work well for extremely quiet cultures, because such a culture will not have a sufficient number of inter-spike intervals for a maximum-likelihood method to find a good fit. This is typically apparent in the results as a very high exponent, usually greater than 500, when the usual values from biological cultures are around 2-6.

Bibliography

Acimovic, J., T. Mäki-Marttunen, R. Havela, H. Teppola, and M. Linne (2011). Modeling of neuronal growth in vitro: comparison of simulation tools NETMORPH and CX3D.

Chao, Z., D. Bakkum, and S. Potter (2007). Region-specific network plasticity in simulated and living cortical networks: comparison of the center of activity trajectory (CAT) with other statistics. *Journal of Neural Engineering 4*, 294.

Chao, Z., D. Bakkum, and S. Potter (2008a). Shaping embodied neural networks for adaptive goal-directed behavior. *PLoS Computational Biology 4*(3), e1000042.

Chao, Z. C., D. J. Bakkum, and S. M. Potter (2008b, 03). Shaping embodied neural networks for adaptive goal-directed behavior. *PLoS Computational Biology 4*(3), e1000042.

Clauset, A., C. R. Shalizi, and M. E. Newman (2009). Power-law distributions in empirical data. *SIAM Review 51*(4), 661–703.

De Schutter, E. (1993). *Nodus, A User Friendly Neuron Simulator for Macintosh Computers*. Kluwer Academic.

Erickson, J., A. Tooker, Y. Tai, and J. Pine (2008). Caged neuron MEA: A system for long-term investigation of cultured neural network connectivity. *Journal of Neuroscience Methods 175*(1), 1–16.

Escol, R., C. Pouzat, A. Chaffiol, B. Yvert, I. E. Magnin, and R. Guillemaud (2008). SIMONE: A Realistic Neural Network Simulator to Reproduce MEA-Based Recordings. *IEEE Transactions on Neural Systems and Rehabilitation Engineering 16*(2).

Esposti, F. and M. Signorini (2008a). Synchronization of neurons in micro-electrode array cultures. *The European Physical Journal-Special Topics 165*(1), 129–135.

Esposti, F. and M. Signorini (2008b). The estimation of long-term memory characteristics in MEA neuronal culture recordings. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, pp. 1017–1020. IEEE.

Fournier, A., D. Fussell, and L. Carpenter (1982, June). Computer rendering of stochastic models. *Communications of the ACM 25*(6), 371–384.

Fuchs, E., A. Ayali, E. Ben-Jacob, and S. Boccaletti (2009). The formation of synchronization cliques during the development of modular neural networks. *Physical Biology 6*, 036018.

Gafarov, F. (2006). Self-wiring in neural nets of point-like cortical neurons fails to reproduce cytoarchitectural differences. *Arxiv preprint q-bio/0607002*.

Goodman, D. and R. Brette (2008). Brian: a simulator for spiking neural networks in Python. *Frontiers in Neuroinformatics 2*.

Hines, M. (1993). Neurona program for simulation of nerve equations. *Neural systems: Analysis and Modeling*, 127–136.

Izhikevich, E. (2003). Simple model of spiking neurons. *Neural Networks, IEEE Transactions on 14*(6), 1569–1572.

Izhikevich, E. M., J. A. Gally, and G. M. Edelman (2004). Spike-timing dynamics of neuronal groups. *Cerebral Cortex 14*(8), 933–944.

Jolivet, R., T. Lewis, and W. Gerstner (2004). Generalized integrate-and-fire models of neuronal activity approximate spike trains of a detailed model to a high degree of accuracy. *Journal of Neurophysiology 92*(2), 959.

Kahng, D., Y. Nam, and D. Lee (2007). Stochastic simulation model for patterned neural multi-electrode arrays. In *Bioinformatics and Bioengineering, 2007. BIBE 2007. Proceedings of the 7th IEEE International Conference on*, pp. 736–740. IEEE.

Koene, R., B. Tijms, P. van Hees, F. Postma, A. de Ridder, G. Ramakers, J. van Pelt, and A. van Ooyen (2009). Netmorph: a framework for the stochastic generation of large scale neuronal networks with realistic neuron morphologies. *Neuroinformatics 7*(3), 195–210.

Kroupina, O. and R. Rojas (2004). *A Survey of compartmental modelling packages*. Freie Univ., Fachbereich Mathematik und Informatik.

le Feber, J., J. Stegenga, and W. Rutten (2008). Do external stimuli, applied to train cultured cortical networks, disturb the balance between activity and connectivity? In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, pp. 5081–5084. IEEE.

Madhavan, R., Z. Chao, and S. Potter (2007). Plasticity of recurring spatiotemporal activity patterns in cortical networks. *Physical Biology 4*, 181.

Migliore, M., C. Cannia, W. Lytton, H. Markram, and M. Hines (2006). Parallel network simulations with neuron. *Journal of Computational Neuroscience 21*(2), 119–129.

Natschlaeger, T., W. Maass, and H. Markram (2002). The "Liquid computer: A novel strategy for real-time computing on time series. *Special Issue on Foundations of Information Processing of TELEMATIK 8*(1), 39–43.

Ostojic, S., N. Brunel, and V. Hakim (2009). How connectivity, background activity, and synaptic properties shape the cross-correlation between spike trains. *The Journal of Neuroscience 29*(33), 10234.

Patel, T. P., S. C. Ventre, and D. F. Meaney (2012). Dynamic changes in neural circuit topology following mild mechanical injury in vitro. *Annals of Biomedical Engineering 40*(1), 23–36.

Potter, S. and T. DeMarse (2001). A new approach to neural cell culture for long-term studies. *Journal of Neuroscience Methods 110*(1-2), 17–24.

Rolston, J., D. Wagenaar, and S. Potter (2007). Precisely timed spatiotemporal patterns of neural activity in dissociated cortical cultures. *Neuroscience 148*(1), 294–303.

Roxin, A., H. Riecke, and S. Solla (2004). Self-sustained activity in a small-world network of excitable neurons. *Physical Review Letters 92*(19), 198101.

Ruaro, M., P. Bonifazi, and V. Torre (2005). Toward the neurocomputer: Image processing and pattern recognition with neuronal cultures. *Biomedical Engineering, IEEE Transactions on 52*(3), 371–383.

Segev, R. and E. Ben-Jacob (2000). Generic modeling of chemotactic based self-wiring of neural networks. *Neural Networks 13*(2), 185 – 199.

Segev, R., M. Benveniste, Y. Shapira, E. Ben-Jacob, et al. (2003). Formation of electrically active clusterized neural networks. *Physical Review Letters 90*(16), 168101–168101.

Serra, M., A. Chan, M. Dubey, V. Gilman, and T. Shea (2008). Folate and s-adenosylmethionine modulate synaptic activity in cultured cortical neurons: acute differential impact on normal and apolipoprotein-deficient mice. *Physical Biology 5*, 044002.

Sharp, P. E. (2011). Buddhist Enlightenment and the Destruction of Attractor Networks: A Neuroscientific Speculation on the Buddhist Path from Everyday Consciousness to Buddha-Awakening. *Journal of Consciousness Studies 18*(3-4), 137–169.

Shea, T. (2009). Optimization of neuronal-computer interface. Technical report, DTIC Document.

Shefi, O., I. Golding, R. Segev, E. Ben-Jacob, and A. Ayali (2002). Morphological characterization of in vitro neuronal networks. *Physical Review E 66*(2), 021905.

Sorkin, R., T. Gabay, P. Blinder, D. Baranes, E. Ben-Jacob, and Y. Hanein (2006). Compact self-wiring in cultured neural networks. *Journal of Neural Engineering 3*, 95.

Vanier, M. (2005, Mar). Large-scale neural network models. In *In Advanced Tutorials at WAM-BAMM*05*, San Antonio, Texas, USA.

Vibert, J.-F., F. Alvarez, and E. K. Kosmidis (2001). XNBC V9: A user friendly simulation and analysis tool for neurobiologists. *Neurocomputing 3840*(0), 1715 – 1723.

Vibert, J.-F., K. Pakdaman, E. Boussard, and E. Av-Ron (1997). XNBC: a simulation tool. application to the study of neural coding using hybrid networks. *Biosystems 40*(12), 211 – 218.

Wagenaar, D., T. DeMarse, and S. Potter (2005). MEABench: A toolset for multi-electrode data acquisition and on-line analysis. In *2nd International IEEE EMBS Conference on Neural Engineering*, pp. 518–521. IEEE.

Wagenaar, D., R. Madhavan, J. Pine, and S. Potter (2005). Controlling bursting in cortical cultures with closed-loop multi-electrode stimulation. *The Journal of Neuroscience 25*(3), 680–688.

Wagenaar, D., J. Pine, and S. Potter (2006). An extremely rich repertoire of bursting patterns during the development of cortical cultures. *BMC Neuroscience 7*(1), 11.

Warwick, K., D. Xydas, S. Nasuto, V. Becerra, M. Hammond, J. Downes, S. Marshall, B. Whalley, et al. (2010). Controlling a mobile robot with a biological brain. *Defence Science Journal 60*(1), 5–14.

Watts, D. and S. Strogatz (1998). Collective dynamics of "small-world" networks. *Nature 393*(6684), 440–442.

Zemianek, J. M., S. Lee, M. Guaraldi, and T. B. Shea (2012). Accelerated establishment of mature signaling patterns following stimulation of developing neuronal networks:learning versus plasticity. *International Journal of Developmental Neuroscience*.

Zemianek, J. M., A. M. Shultz, S. Lee, M. Guaraldi, H. Yanco, and T. B. Shea (2012). Transient epileptiform signaling during neuronal network development: Regulation by external stimulation and bimodal gabaergic activity. *International Journal of Developmental Neuroscience*.

Ziv, I., D. Baxter, and J. Byrne (1994). Simulator for neural networks and action potentials: description and application. *Journal of Neurophysiology 71*(1), 294–308.