Agent Based Model of Cavitation in Spinal Cord Injury

Rahma Ahmed
Bard College
Agent Based Model of Cavitation in Spinal Cord Injury

Senior Project Submitted to
The Division of Science, Mathematics, and Computing
of Bard College

by
Rahma Ahmed

Annandale-on-Hudson, New York
May 2019
Acknowledgements

I have to first thank Allah (SWT) for always being there for me throughout my life. He alone knows how much faith in him drives me every day. Without Islam, I would not be the woman I am today and I probably would not have made it through this senior project. To the people who have given me the most in this life, my parents: Mama and Baba, your love and support has meant more than words can describe and I am very grateful to have the both of you on my side. You have seen me through it all and taught me from a young age to be curious, ask questions, and never give up on what I put my mind to. Your unconditional encouragement helped me get through my classes and remember that everything will work out the way it is meant to. To my siblings: Abdelrahman, your very persuasive arguments are what sparked my interest in computer science. Without your consistent help through many bugs I have encountered, I probably would have thrown my laptop out the window months ago. Romiesa, my first best friend. Our hour long conversations and your random texts always put a smile on my face and help me stay calm when it seems like everything is falling down on top of me. Mawada, though you are younger, you sometimes make me feel like the younger sister with the way you support and encourage me. I do not think you know how much your small notes hidden in my bag and on my desk have put a smile on my face a bunch of times. Raheeq and Suhayl, you never fail to keep me busy and keep the house loud and interesting. My whole family has been my rock through the rollercoaster that is senior project and throughout my time away from home. To my senior project advisor Kerri-Ann Norton, you deserve a huge thank you for being patient with me and staying positive even when I was convinced nothing was working. Your constant feedback and optimism kept me going and this project would not have been possible without you. To my academic advisor Keith O’hara, I want to especially thank you for helping me realize that feeling like the smartest person ever and then the dumbest is part of the process for every CS project. The biggest thanks to my family and friends, my rock and support system through it all. To my friends at Bard, my family away from home whose company and laughter keeps my spirits high, thank you for helping me get through senior project. Thank you to Matt Carlberg who saw a talent in me and encouraged me to continue exploring CS. Thank you to Sven Anderson and Heather Bennett for serving on my board! There are many more people who I can thank and the list can go on but it seems I should probably get to the content of my SPROJ.
# Table of Contents

*Abstract .......................................................................................................................... 1

1 Introduction .................................................................................................................... 2
  1.1 Spinal Cord .................................................................................................................. 2
  1.2 Spinal Cord Injury ...................................................................................................... 3
  1.3 Neuroprotective treatments ....................................................................................... 6
  1.4 Regenerative treatments .......................................................................................... 8
  1.5 Research on cavitation .............................................................................................. 10
  1.6 Computational models ............................................................................................. 12
  1.7 Spinal Cord Injury Models ....................................................................................... 14

2 Methods .......................................................................................................................... 16
  2.1 Outline of Approach .................................................................................................. 16
  2.2 Explanation of simulation ......................................................................................... 21
  2.3 Image Processing ...................................................................................................... 25
  2.4 Statistics .................................................................................................................... 29

3 Results ............................................................................................................................. 31

4 Discussion ....................................................................................................................... 36
  4.1 Limitations ................................................................................................................ 37
  4.2 Treatment of SCI progression .................................................................................. 38
  4.3 Benefits of in silico models ...................................................................................... 39

5 References ....................................................................................................................... 41

6 Appendix .......................................................................................................................... 47
  6.1 Supplemental Figures ............................................................................................... 47
  6.2 Processing Code ........................................................................................................ 47
  6.3 R Code ....................................................................................................................... 52
Abstract

Annually, approximately 375,000 people suffer from spinal cord injury (SCI) worldwide and many SCI patients develop secondary health conditions such as respiratory, cardiovascular, and urinary/bowel complications which negatively impact their daily lives. SCI occurs when there is damage to the spinal cord resulting in decreased motor functions, decreased sensory functions, or paralysis. Days to weeks after initial impact, the lesion (area of injury) continues to increase in size in a process called progressive cavitation which demyelinates axons and inhibits effective axonal regeneration. In an in vitro model of progressive cavitation, Fitch et al. showed that activated macrophages cause cavities to form (areas devoid of cells) in astrocyte monolayers (Fitch et al. 1999). In this senior project, I developed an agent based model that replicates the process of cavitation as described in the in vitro experiment. My results showed that, similar to Fitch et al’s results, cavity size and astrocyte density increase with increasing cell speed. Due to the time, effort, ethics, and cost involved with in vivo studies, this model provides an alternative approach to predicting optimal treatments leading to a more guided exploration of treatments that target and reduce progressive cavitation.
1 Introduction

Approximately 375,000 people suffer from spinal cord injury (SCI) worldwide and 12,000 new SCIs occur in the United States annually (World Health Organization, 2013). Spinal cord injury (SCI) is characterized as damage to the spinal cord or nerves in the spinal canal (the cavity that contains the spinal cord). SCIs typically cause permanent damage to the body that results in a decline of body functions controlled by cells at and around the site of injury. Motor Vehicle Crashes account for 39.3% of SCIs, falls account for 31.8%, and violence (gunshot wounds), sports, and medical/surgical procedures combined account for less than 30% (National SCI Statistical Center, 2019). Up until World War II, serious SCIs usually resulted in death or a life with significant medical complications such as organ failure and blood clots. These complications also left most SCI patients confined to a wheelchair for the rest of their lives (Silver 2005). However, research has advanced treatments available for SCI patients including immobilization of the spine on the scene of the accident, surgical procedures, stem cell transplants, and many drug therapies are currently in clinical trials giving health care providers and their patients hope for recovery. Current treatments focus on helping people with SCI continue to be independent and productive despite their injury (Zimmer et al. 2007; Mayo Clinic Staff 2017). To advance the possible treatments and minimize the negative effects of SCI, research into understanding the progression of injury and possible treatment targets are two of the biggest focuses of much of today’s research into SCIs.

1.1 Spinal Cord

An understanding of the spinal cord is key to the study of SCI. The spinal cord is a long and thin cylindrical shaped structure within the central nervous system (CNS) located within the
vertebrae in the spine (Squire et al. 2008; Guertin 2012). It runs from the base of the skull to the start of the lumbar spine (lower back) and consists of many neurons and glial cells. Neurons are tasked with transferring signals to muscle and other neurons in the Peripheral Nervous System (PNS) and CNS. Glial cells are nonneuronal cells in the CNS with subtypes such as astrocytes, oligodendrocytes, and microglia with roles such as protecting neurons, forming myelin (sheaths of proteins that wrap around the axons of neurons to improve signal conduction down the axon), and removing dead neurons from tissue respectively. Signal conduction is how neurons pass information, as electrical impulses, to and from the CNS to other neurons and cells outside the CNS (in the PNS). Essentially, the spinal cord acts as a pathway to send sensory information from the body to the brain and motor information from the brain to the rest of the body.

1.2 Spinal Cord Injury

Spinal Cord Injury occurs when the above mentioned messenger pathway is disrupted resulting in motor and sensory deficiencies. Damage to the spinal cord occurs in two major steps: first the mechanical injury (also called the primary injury) then the chemical aftermath (or secondary injury). The primary lesion is the structural damage that results from the mechanical injury causing a compression or laceration in the spinal cord (Ahuja, Cadotte, and Fehlings, 2018). This primary lesion, which occurs up to hours after injury, causes axons and the cell membranes of neurons to be damaged resulting in swelling of the spinal cord. The structural devastation incurred from the mechanical injury induces a cascade of biochemical events which cause functional damage often referred to as the secondary injury (Leal-Filho 2011). The secondary injury often lasts days to weeks after the primary injury occurs and is characterized by demyelination of axons, death of neurons, and activation of the immune system. Swelling of the
spinal cord results in low blood supply (ischemia) (Figure 1) and causes less oxygen and nutrients to reach cells (which causes neuronal death due to lack of nutrients) and development of a larger lesion referred to as the secondary lesion. In adults, whereas damage to the PNS is typically reversed by the body’s repair mechanisms, the CNS does not experience the same restorative behavior making SCIs more detrimental than other types of injuries in the nervous system (McDonald, Becker, and Huettner, 2004).

Even though some cells die as a consequence of the mechanical (primary) injury, most of the cell death that occurs post SCI is a result of the secondary injury (Tator and Fehlings 1991). Death of oligodendrocytes elicits demyelination of axons because these glial cells can no longer produce the myelin sheaths necessary to keep axons myelinated. As part of the secondary injury, immune cells migrate to the injury site to clear away dead cells.

Di Giovanni et al. suggested that the neuronal death, apoptosis, that occurs during the secondary injury may be due to up-regulation of the cell-cycle (Di Giovanni et al. 2003). They showed that in a rat model of spinal cord injury, neurons showing signs of apoptosis express cell-cycle proteins suggesting these proteins are involved in neuronal damage and death. Since the cell cycle is usually down-regulated in non-mitotic cells, if cells enter the cell cycle, in this case it leads to apoptosis though the exact mechanism is not understood (Beattie et al. 2000). Another reason for neuronal death is damaged axons’ response to injury with neuronal degeneration (breaking down of damaged neurons) (Zheng et al. 2019).

Astrocytes, star shaped glial cells, enlarge, become reactive, and proliferate to form a protective layer on the outside of the lesion. This layer of glial cells is typically referred to as the glial scar (Figure 1) (Cregg et al. 2014). The glial scar serves as a barrier between the injured tissue
and healthy tissue and limits the size of the lesion during the first few weeks of injury. However, since the glial scar is an additional layer of cells, it causes nearby axons which were not demyelinated due to the death of oligodendrocytes to become demyelinated due to limited space and in some cases, the demyelinated cells die. Additionally, inflammatory cells such as macrophages are recruited to the lesion core as a response to the cell death that occurs (Fujiki, Kobayashi, and Isono, 2003). These macrophages remove dead cells and within a few weeks post injury, a fluid filled cavity is left behind in the space that cells once occupied. Various studies including Fujiki et al. found that cavities form in the presence of activated macrophages as part of the secondary lesion, but the exact mechanism is not well understood (Figure 1) (Fujiki et al., 2003).

Figure 1: Spinal Cord Injury Lesion. Various processes that occur post SCI. Cavitation occurs when cells die and are removed by immune cells leaving an area devoid of cells. The glial scar forms as a protective layer around the cavity (Mothe and Tator 2012).
al. 2005). This process is referred to as progressive cavitation because the scar is larger than its original size at the time of impact.

The cascade of events that occur during SCI includes severing of axons that are disrupted by the primary injury, demyelination of axons in both the primary (initial impact) and secondary injury (specifically the death of oligodendrocytes), apoptosis (cell death) and necrosis (tissue death) also occur in both steps of injury progression (Figure 1) (Leal-Filho 2011). Inflammation occurs in the secondary injury as the immune system responds to the injury, edema (swelling) occurs due to the inflammatory response, and excitotoxicity and oxidative damage to neurons occur due to increased release of glutamate (an excitatory neurotransmitter) resulting in overactivation of glutamate receptors on neurons which kills them. The primary injury may also rupture some blood vessels causing blood to leak out of them and cause hemorrhage (bleeding) which increases edema and causes vasospasm (spasms in arteries that lead to vessels constricting). Many of studies today focus either on neuroprotective measures to inhibit further damage that occurs in the spinal cord or on regenerative treatment to promote regeneration of myelin sheaths and severed axons and to improve survival of cells.

1.3 Neuroprotective treatments

Neuroprotective treatments are important in reducing the negative impact of the secondary injury (Martin et al. 2015; Ahuja et al. 2016; Ahuja and Fehlings 2016). Different treatments have been investigated for properties that induce neuroprotection in the spinal cord. A few such therapies that target different aspects of the secondary injury are induced hypothermia (low body temperature), Geranylgeranylace tone (GCA), Granulocyte colony-stimulating factor (G-CSF), and Flavopiridol.
Cooling of the body through induced hypothermia, in a controlled setting has shown to provide neuroprotection for cardiac arrest patients (Hypothermia after Cardiac Arrest Study Group 2002). In a rat model of SCI, Lo et al. demonstrated the positive effect of inducing hypothermia to minimize neuronal damage (Kwon et al. 2008; Lo et al. 2009). Though this type of treatment is practiced in clinical settings for cardiac arrest patients and has been tested on SCI patients, it has not yet been approved for treating SCI patients (Levi et al. 2009).

Geranylgeranyacetone (GCA) prevents increases of Tumor Necrosis Factor Alpha (TNF-α) (a protein involved in inflammation) and neutrophils (white blood cells in the immune system) in the spinal cord after SCI (Fujiki et al. 2005). This limits secondary injury, neuronal death, and progressive cavitation in rat models of spinal cord injury. However, as with hypothermia therapy, even though this drug has undergone some clinical trials, it is not yet approved for use in patients with SCI (Ahuja and Fehlings 2016).

Granulocyte colony-stimulating factor (G-CSF) has shown to have neuroprotective effects in rat models of spinal cord injury by stimulating angiogenesis (development of new blood vessels) which results in recovery of motor function (Kawabe et al. 2011). These findings were used to run clinical trials to test the use of G-CSF in human subjects (Kamiya et al. 2015). Kamiya et al. found that this drug is safe and effective in human subjects but again it is not yet used in clinical settings.

Flavopiridol, derived from an Indian plant, is an inhibitor of cyclin dependent kinases, a group of proteins that regulate the cell cycle (Byrnes et al. 2007). Byrnes et al. investigated the benefits of administering Flavopiridol to rats after inducing a contusion SCI. They found that this drug reduces the volume of the lesion produced post SCI. This is significant because the larger the lesion size, the more overall damage to the tissue resulting in decline of body functions and increased axonal regeneration. Using the Basso, Beattie, and Bresnahan (BBB) locomotor scale
(typically used to test function recovery and movement post SCI), Byrnes et al. observed improvement in functional recovery of rats treated with Flavopiridol. Since motor function is typically impacted by SCIs, finding ways to reduce the damage as much as possible can help patients get closer to their normal functions before their SCI. They also concluded that astrocytes become less reactive upon treatment resulting in reduced scarring around injury, an overall smaller lesion, and less negative impact on healthy tissue and cells. The last significant impact of Flavopiridol is reduction of cell cycle activation which resulted in less apoptosis. As Di Giovanni et al. found, inhibiting the activation of the cell cycle has neuroprotective benefits (Di Giovanni et al. 2005).

In a different model of glial scarring, Di Giovanni et al. studied the effects of Flavopiridol on rats after inducing a Traumatic brain injury (TBI). Similar to Brynes et al.’s study, these scientists found reduction of cell death, astrocyte proliferation, immune cell activation, lesion volume, and scar formation as well as increased functional recovery. These two studies support further exploration of flavopiridol as a treatment for CNS injury. However, though these studies are more than ten years old, this drug has only been used as a cancer drug (Tan and Swain 2002; Stephens et al. 2013; Orphan Drug Status for Alvocidib, 2014).

1.4 Regenerative treatments

In addition to neuroprotection, neuroregeneration is key to repair the damage done by the secondary injury to help patients with SCIs recover some of the functionality they lost. There is no consensus on whether it is better to administer these drugs right after a SCI occurs or if it is better to wait until the secondary injury has stopped or slowed down (Martin et al. 2015; Ahuja et al. 2016; Ahuja and Fehlings 2016). A few therapies including Chondroitinase ABC, stem cell
therapy, ibuprofen, and minocycline have been tested in animal models of SCI. Some have moved onto clinical trials but none of these are currently in clinical circulation. Chondroitinase ABC (ChABC), an enzyme derived from bacteria targets Chondroitin Sulphate Proteoglycans (CSPGs) which are proteins involved in glial scar formation and are critical in inhibiting axonal regeneration (Bradbury et al. 2002). Using an adult rat model of SCI, Bradbury et al. suggested the use of ChABC to upregulate proteins involved with neuronal regeneration and promote axonal regeneration. Neural stem cells (cells capable of differentiating into a number of cell types) have been used to promote functional recovery and inhibit glial scar formation (Teng et al. 2002). Though Teng et al. showed that this inhibition is temporary, they suggested use of stem cells soon after SCI to reduce the reactivity of astrocytes that form the glial scar. Ibuprofen has shown in vivo (inside of a living organism) and in vitro (outside of a living organism typically in a test tube or cell culture on a dish) to limit inhibition of axonal growth that occurs in the secondary injury (Wang et al. 2009). Wang et al. used a rat model of SCI to show that ibuprofen can protect tissue in the spinal cord and stimulate axonal regeneration though the exact mechanism is not well understood.

Minocycline is a clinically prescribed drug that has been in circulation for over 30 years to treat diseases related to inflammation and more recently for neuroprotection (Garrido-Mesa et al. 2013). It has been used in models of traumatic brain injury, and various neurodegenerative diseases to reduce apoptosis. Festoff et al. and Wells et al., showed the benefits of Minocycline in rat and mouse models of SCI (Wells et al. 2003; Festoff et al. 2006). By inducing contusion/compression SCI in rat and mouse models, Festoff et al., and Wells et al., respectively found Minocycline improves functional recovery (better BBB scores). They also found the drug reduces the lesion/cavity size and apoptosis. Results from a phase II studies lacked significance.
but the possibility of reducing the damage of the secondary injury warranted phase III trials to further investigate the benefits of using this drug in SCI patients (Casha et al. 2012). Though Minocycline is being used as an antibiotic to treat bacterial infections, there is no evidence that it is used to treat the secondary injury in SCI (Goff et al. 2014).

1.5 Research on cavitation

Glial scar formation and progressive cavitation are two of the least well understood aspects of SCI. Faulkner et al. measured the effects of removing the glial scar by ablating (surgically removing) dividing reactive astrocytes and inducing SCI (Faulkner et al., 2004). They found inflammation increased, characterized by the presence of more inflammatory cells, when reactive astrocytes were removed. This led to a fivefold larger secondary lesion compared to the lesion in the control mice (with reactive astrocytes as is typical post CNS injury). This study shows that completely removing the glial scar increases progressive cavitation leading to a larger lesion size which negatively impacts healing in SCI patients. This finding supports previous claims that the glial scar has a protective function (Faulkner et al. 2004; Sofroniew 2009).

In a model of glial scarring and cavitation, Fitch et al., used a series of in vivo and in vitro experiments to show that inflammation post primary CNS injury leads to more tissue damage (including cavitation and glial scarring) (Fitch et al. 1999). The in vivo studies modeled the secondary injury that occurs during SCI to investigate the effects of chemical inflammatory processes separately from the primary injury that occurs due to the physical tissue and cell damage. By injecting rats with compounds involved in the activation of macrophages and microglia, Fitch et al., concluded that continuous inflammation led to progressive cavitation and glial scarring.
Two in vitro models were then used to explore the effects of activated macrophages on neurons and astrocytes separately. Macrophages were isolated from adult rats and some were activated with Zymosan (a macrophage activator) while the control were not. Neurons were isolate from the lumbar and cervical spines of adult rats then plated in culture plates. Activated macrophages were then added for the experimental group and non-activated macrophages were used as controls. Fitch et al., found that activated macrophages and microglia decreased survival of neurons.

To investigate progressive cavitation, astrocytes were obtained and isolated from newborn rat cortices. These cultures were then seeded and left to settle for one to three days to form an astrocytic monolayer. Macrophages were used in two separate experiments as co-cultures with astrocytes or to make conditioned media (using three different macrophage concentrations). The macrophages and conditioned media were then added to astrocyte monolayers to replicate the chain of events that occur after CNS trauma. Fitch et al. calculated number of astrocytes, cell density, and size of cavities and found that activated macrophages both in co-culture or conditioned media caused progressive cavitation in the astrocyte monolayer. Although their model resulted in multiple smaller cavities, since the in vitro model uses cells without a physical injury, the process of cavitation in vitro leads to many cavities whereas in vivo there is typically one large cavity (Figure 1). They found increased astrocyte density, and suggested astrocyte migration as a possible mechanism for cavitation. Fitch et al. also demonstrated that using an anti-inflammatory agonist blocks activation of inflammation and reduces cavity formation. The in vitro model using conditioned media is the main basis for my in silico (on a computer) model and simulations.

Even though many studies have been done to help us understand SCIs better and find new treatment possibilities, patients with SCIs do not have many treatment options (Martin et al. 2015).
Other than surgery, physical therapy, and the few treatments out there, some patients may be eligible for clinical trials but most of the treatments available focus on preventing additional mechanical damage rather than stopping the process of the secondary injury as soon as possible.

1.6 Computational models

Computers have made study of complicated systems faster and more efficient allowing us to create computational models using computer science, math, and physics (Brodland 2015). These models are becoming an important complement to other types of studies, helping us visualize systems, understand how they work, and predict their behaviors and responses to changes in the environment. One of the most important aspects of a computer simulation is that it uses a pertinent model that reflects the system. The aspects included to make a model appropriate, that is including or excluding certain aspects of the system, depend heavily on the questions being asked. Once the question being asked is well defined, the type of model can be chosen and from there, the necessary parameters can be identified. Studies that use models for simulation have various applications such as evaluating performance, assessing and managing risks, and evaluating possible actions to be taken. Computational models have become especially useful in computational biology, for example, in the study of proteins, the immune system, and cancer (Krogh et al. 1993; Norton et al. 2010; Germain et al. 2011). To choose an appropriate model we must first examine the different kinds of computational models and their applications.

One of the ways in which computational and mathematical models are classified is discrete versus continuous models. A discrete model has variables (called state variables) that change at specific points in time when an event or change in state occurs (Birta and Arbez 2007). In simulations of these types of models, time changes in leaps (is not continuous). A continuous
model on the other hand, is one in which time is continual and variables change constantly based on the varying state of the system (Banks 1998). This type of model is used for example to measure concentration of cells or objects in an environment over time. However, when this type of model is simulated computationally, we do not have data points for every moment in time from the actual system so it is not actually completely continuous but the variables usually fall within a range of values. The more data points the model has, the more accurate it is because there are less gaps. There are also hybrid models which have elements from both discrete and continuous models (Martin and Raffo 2000; Bortolussi and Policriti 2008). These models use ideas from continuous modeling for dynamic variables/aspects of the model and ideas from discrete modeling for variables that change based on a specific timeline. These models can be described as discrete variables (that jump) in a continuous environment (that flows).

Discrete models typically come in two types: lattice and off-lattice. A Cellular Automaton (CA) is an example of an on lattice grid of cells (Wolfram 1984; Adamatzky 1994). Cellular automata have three major components: an array of cells, a neighborhood, and rules for interactions/transitions. Each cell in a CA can have a finite number of states that change based on the rules of transition at specific points in time depending on events that occur in its neighborhood. On the other hand, in off lattice (on a continuum of space) agent based models (ABMs), the system is typically one or a few autonomous agents that act and interact in a given environment (Bonabeau 2002). Each agent makes its own decisions which can impact and be impacted by the actions of other agents. This type of model allows for simulation of complex behaviors of systems that are not well understood because unlike most discrete models, it is not based on any equations (An et al. 2017). Since many biological systems have not yet been explained by equations, using ABMs allows one to give the model as much or as little information as is known about the system
It is important to investigate the pros and cons of simulations *in silico*. One of the biggest cons of simulation is accuracy (Banks 1998; Birta and Arbez 2007). Since knowledge of the processes that occur during SCI is limited, one may argue that this may result in an inaccurate model and simulation. To resolve this issue, we add in as much detail from the real world and try to fit our model to an *in vivo* or *in vitro* study by duplicating their results using our *in silico* model. The pros of simulation are vast. Whereas lab equipment to maintain model organisms, induce SCI, monitor changes on a cellular level, and other steps in the *in vivo* and *in vitro* process tend to be expensive, the price of computer simulation is close to nothing. Additionally, simulations are less invasive making experimentation with them more ethical and low stakes since it does not involve any living things. When simulations produce unexpected results, they force researchers to consider underlying reasons for the cause of these results (Brodland 2015). Simulations can also require less effort, technical limitations, and are less time consuming than working in a wet lab (Zhou and Kuhl 2011). Even though some simulations may require high amounts of processing power and may take hours to days to run, unlike *in vivo* and *in vitro* experiments, the experimenter can define simulation time. Having the power to define simulation time can allow researchers to run an experiment that takes days to run (*in vivo* or *in vitro*) in just a couple hours *in silico*.

### 1.7 Spinal Cord Injury Models

Within the last decade alone, there have been a multitude of *in silico* models of the spinal cord and spinal cord injury developed by scientists to increase our understanding of these two systems and investigate some of their processes. In an agent based model of pressure ulcers (tissue damage due to continued pressure), Ziraldo et al. used a model of tissue cells to simulate formation of pressure ulcers (PU) and test possible treatment options for reducing PU (Ziraldo et al. 2015).
They found that anti-inflammatory treatments could reduce the level of expansion of inflammation and tissue damage. They suggested use of ABMs to increase understanding of mechanisms involved with PU and investigate therapies to improve patient outcomes. Solovyev et al. developed a hybrid model to combine hyperemia (blood build up in vessels) and PU (Solovyev et al. 2013). Incorporating an Ordinary Differential Equation (ODE) model (which uses a set of ODEs) for hyperemia and an ABM for PU, they showed that individuals with SCI are more likely than those without SCI to develop ulcers as a response to pressure on their spine.

Finite element models (FEMs) are another type of model which has a set (finite) number of components (elements) that interact with one another at their points of connection (Zafarparandeh et al. 2014). Yuk created a finite element model of spinal cord tissue for use in understanding the mechanical injury of SCI (Yuk 2016). Another finite model of the spinal cord was developed by Zafarparandeh et al. to improve understanding of its biomechanics (structure and mechanism of movement) (Zafarparandeh et al. 2014). Maikos et al. developed a 3D FEM to simulate spinal cord injury in rat models (Maikos et al. 2008). Their model serves as yet another way to study the biomechanics of SCI especially in research involving model organisms. Though there are models of the spinal cord as well as spinal cord injury, there is little evidence of development of *in silico* models of progressive cavitation.
2 Methods

Since this *in silico* model aimed to replicate Fitch et al.’s *in vitro* study, with astrocytes interacting with one another and reacting to changes within their environment (treatment with active or nonactive macrophage conditioned media), an agent based model was used.

2.1 Outline of Approach

To accomplish the goal of creating a model that exemplifies astrocytes and consequences of SCI, the process of making a simulation as defined by Robert Shannon was used (Shannon 1998). The bullet points represent how each of Shannon’s steps was applied to this *in silico* model.

1. **Define problem.** What questions is this project trying to answer?
   - The goal of this project was to investigate (*in silico*) progressive cavitation which occurs during SCI. To do this, the *in vitro* results of Fitch et al. needed to be replicated then used to consider different treatment targets as possible ways to decrease cavitation during SCI (Fitch et al. 1999).

2. **Plan project** to ensure the necessary computational resources are available.
   - An agent based model of astrocytes was developed using Processing (version 3.3.6) which makes visualizing the cells simple.

3. **Define the system.** To create a model, the system being modeled must be clearly defined. A system is composed of a group of interacting factors that all work towards the same goal and are related to one another. This step includes understanding how the system works and deciding which parts to include and which to leave out.
   - To start off with a simpler scenario, the system did not include a macrophage coculture. This system is an *in vitro* cell culture of astrocytes whose movement is either affected by active or nonactive macrophage conditioned media (to replicate the
processes that occur during SCI). The astrocytes also experienced a drag force that represents their movement within the extracellular matrix or media and they interacted with each other through a spring force so as to limit the possibility that they completely cross one another.

4. **Formulate an idea of what the model** will look like. A model is a depiction or replication of a real world system which can then be used to predict something about the system.

   • This model included astrocytes depicted as blue ellipses placed in the middle of a 500x500 pixel screen where one pixel corresponds to one micron. It also had either activated or nonactivated macrophage conditioned media which was controlled by a variable that impacts astrocyte movement within the astrocytic monolayer and was either true for active or false for nonactivate (macrophage conditioned media).

5. **Preliminary design of the experiment.** This is where the factors being altered are determined and necessary data to support creation of the model is obtained. This step involves determining the plan for the experiment and deciding on measures of effectiveness.

   • In addition to the *in vitro* experiments from Fitch et al., parameters such as cell size and speed were necessary to replicate normal conditions of cells. Measure of effectiveness was whether or not increasing the speed of astrocytes in the activated macrophage media experiment also increased the astrocyte cavity area similar to Fitch et. al.’s *in vitro* cavitation assay.

6. **Preparing input data.** Finding and documenting data listed in step 5 for the model.
Details of parameters such as the average size of astrocytes, their movement speed, their interactions with one another, and how they are typically plated in vitro was gathered (see Table 1).

Table 1: List of parameter values. Range values for the three main variables in the model.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Astrocyte Size</td>
<td>20 – 25 microns (Bear et al. 2007)</td>
</tr>
<tr>
<td>Cell culture density</td>
<td>80% confluency, cells fill up 80% of the plate/well (Arora et al. 2008)</td>
</tr>
<tr>
<td>Astrocytic Speed</td>
<td><strong>Not active:</strong> 100 microns/24 hours (Emmett et al. 1991)</td>
</tr>
<tr>
<td></td>
<td><strong>Active or around active cells:</strong></td>
</tr>
<tr>
<td></td>
<td>220 microns/24 hours (Zhou et al. 1990)</td>
</tr>
<tr>
<td></td>
<td>481 microns/48 hours (Kuiper et al. 2009)</td>
</tr>
<tr>
<td></td>
<td>10 – 70 microns/hour (Környei et al. 2000)</td>
</tr>
</tbody>
</table>

7. **Translating model** to a simulation. A simulation is an emulation of real world systems over time using a model with data gathered from observing the system (Banks 1998). It shows the actions of the system and allows for experimentation to make inferences about future actions and characteristics of the system.

After gathering input data, the simulation was given rules for interaction (such as bouncing off of other cells in the case of a collision) and run for a total of 72 hours. Values from the literature (see table 1) were used to obtain a baseline for the astrocyte speed both in the nonactive state and when active macrophage conditioned media was added. Using the average active speed, simulation time was calculated as follows:

\[
\frac{480 \text{ microns}}{48 \text{ hours}} = 10 \text{ microns/1 hour} \\
(10 \text{ microns/1 hour}) \times (1 \text{ iteration/0.2 microns}) = 50 \text{ iterations/hour}
\]
For the first 2 days (48 hours) the cells moved at 0.084 microns as they settled. At the 48 hour mark, if the cells were activated in the simulation (the equivalent of being treated with activated macrophage conditioned media), the speed changed to 0.2 which corresponds to 10 microns/hour. Cavity size (white space in the model) was calculated at the end of the settling period and again at the end of the simulation (hours 48 and 72) (See 2.3 Image Processing). Hour 48 of simulation corresponds to the start of the treatment with either activated or nonactivated macrophage conditioned media (also called the 0 point of treatment or 0 hours into day 3). Hour 72 corresponds to the end of the simulation which is also 24 hours since treatment. The cavity size of the control (nonactivated macrophage conditioned media) at 24 hours is normalized to 1.

8. **Validating the model.** Affirm the validity of the model based on the real system’s results and analyze the expected operation and trends in the models as compared to the system.

   ◆ Preliminary results were compared to those of the *in vitro* experiments described by Fitch et al. (Fitch et al. 1999). Observations showed that the activated condition does not correspond to the value in Fitch et al.’s experiment so other values within the range (10-70 microns/hour) described by Környei et al. were used (Környei, et al. 2000). Based on the preliminary data with sampling of different speeds, there were speed values that resulted in cavity sizes that correspond to the cavity sizes explained by Fitch et. al.

9. **Final design of the experiment.** Planning any changes to original experiment and considering which statistical tests will be used.
• Though the model was validated, the preliminary results only included one trial per speed. More trials were needed to obtain a larger more representative sample size. Once the data was collected, t-tests were important to show the active treatment is statistically different than the treatment with nonactive macrophage conditioned media.

10. **Experiment.** This involves deciding on sample size, and running the simulation to gather data.

• Thirty separate simulations were run to increase sample size for each value and gather the data from the start (48 hours) and end (72 hours) of the treatment period (day three at 0 hours and 24 hours) and images of the cells were saved at 6 hour intervals beginning at time 0 of treatment (48 hours from plating).

• Additional experimentation using different values for the spring constant that affects the way cells bounce off one another was done to show preliminarily the effects that changing this variable has on cavity area.

11. **Analysis and Interpretation** of the data gathered. Deciding which statistical tests make the most sense based on the data. Interpreting what the results mean and the implications they have.

• Used image processing data to analyze differences in white space between the start and end of day 3 for each of the thirty simulations to determine delta (change) in cavity size. Ran statistics on the change in cavity size data (see section 2.4 statistics) to confirm validity of data collected and made conclusions based on the results.
12. **Documenting results and implementing** them.

- This senior project is a documentation of the *in silico* model’s results. Simulation with a range of values for parameters such as speed, spring constant for bounce, or adding macrophages as a co-culture can be used to show possible treatment targets to decrease cavitation post SCI. This data can eventually be used for *in vitro* studies though that is out of scope for this project.

2.2 **Explanation of simulation**

Due to the progression of the lesion during SCI and the autonomous nature of astrocytes in the spinal cord (they act upon and interact with their environment and one another), an agent based model of spinal cord injury was developed to simulate progressive cavitation in SCI. The model used for this simulation is based on Fitch et al.’s *in vitro* model of progressive cavitation. In this *in silico* model, a screen that is 500x500 pixels was used to correspond to a 500x500 micron well, with 1 pixel = 1 micron. Astrocytes, ellipses 20 microns in diameter, were placed on the screen in 80% confluency to simulate plating of cells and allowed to settle for 48 hours (2 days) (Környei, et al. 2000). After 2 days, the cells either changed their speed (by changing stepSize) if they were treated with the activated macrophage conditioned media or stay the same if they were treated with nonactivated macrophage conditioned media. Cells were then allowed to move around randomly for another 24 hours. The data was analyzed at hour 0 of day three (after cells have settled) and then again when the simulation ended at the 24 hour mark of day 3.
Table 2: Classes in simulation. All the classes, their attributes and methods.

<table>
<thead>
<tr>
<th>Class (objects)</th>
<th>Attributes</th>
<th>Methods (What the objects can do)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell (parent class)</td>
<td>Float size: <em>size of the cell</em></td>
<td>display(): <em>displays cell</em></td>
</tr>
<tr>
<td></td>
<td>Pvector location: <em>the location of the cell</em></td>
<td>update(): <em>updates the location of the cell based on the acceleration and velocity</em></td>
</tr>
<tr>
<td></td>
<td>Pvector velocity: <em>the cell’s speed</em></td>
<td>applyForce(): <em>applies a force by adding it to the acceleration</em></td>
</tr>
<tr>
<td></td>
<td>Pvector acceleration: <em>how quickly the cell’s speed increase</em></td>
<td></td>
</tr>
<tr>
<td>Astrocyte (extends Cell)</td>
<td>float stepSize: <em>number of microns astrocytes move per iteration</em></td>
<td>display(): <em>displays the cell as a blue ellipse</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>randStep(): <em>moves the cell by randomly selecting a value: -1, 0, or 1 then multiplying it by stepSize and adding the total value to the location</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>checkEdges(): <em>constrains the cells to the screen’s limits so they cannot go off the screen</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>bounce(): <em>given another cell and a spring force, allows cells to spring slightly when they come in contact so they do not overlap uses Hooke’s law</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Force = -kx, k = spring constant and x = difference between desired distance and actual different between cells</td>
</tr>
</tbody>
</table>
Table 3: Main Method. All the functions and parameters for the main sketch.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Astrocyte[] astrocytes: array of 500 astrocytes</td>
<td>setup(): sets all the original values of the attributes, the background color, display size, sets file to “NonActivated,” and creates the output file.</td>
</tr>
<tr>
<td>int state: a simulation is either in the 0 state opulating or the 1 state where cells are moving around</td>
<td>draw(): run simulations starting with the non-activated condition.</td>
</tr>
<tr>
<td>int cellSize: size of astrocytes – 20 microns</td>
<td>• Place cells on screen in a grid like space with a randomness of +/- 2 pixels</td>
</tr>
<tr>
<td>int iter: keeps track of how many draws have occurred to track simulation time</td>
<td>• Change state to 1 so they start moving. Allow cells to bounce off each other to limit overlap</td>
</tr>
<tr>
<td>int simNum: simulation number to allow multiple simulations to be with the same conditions (active/nonactive treatment and step size combination)</td>
<td>• If cells have settled for 48 hours (iter = 2400), analyze image and start saving images every six hours. If the cells are in active conditioned media, change the stepSize of the cells.</td>
</tr>
<tr>
<td>boolean active: whether the media is conditioned with activated (true) or nonactivated (false) macrophages</td>
<td>• Once the simulation reaches 24 hours (3600 iterations), analyze the image then switch to the “Activated” simulation and repeat the process. If both activated and non-activated simulations have run for a given simulation number (simNum), go to the next simulation.</td>
</tr>
<tr>
<td>int hour: hour within the simulation</td>
<td>• Stop simulations once simNum is greater than 30</td>
</tr>
<tr>
<td>float activeStep: step size (speed) for activated cells</td>
<td>• Close the output file so it can be viewed by the user.</td>
</tr>
<tr>
<td>float springConstant: the (k) spring constant of the cells</td>
<td>analyzeImage(): analyze the screen to determine how many white pixels there are.</td>
</tr>
<tr>
<td>int trialNum: trial number to help separate trials (with different activeSpeed) into different folders</td>
<td>bounceCells(Astrocyte[]astrocytes): check each cell against all other cells in an array excluding itself to determine which ones are overlapping. If any cells are overlapping, the force from bounce() is applied to one cell and the negative of the force is applied to the other cell. Once all cells are cross</td>
</tr>
</tbody>
</table>
and text files (for bookkeeping and ease of analysis)

**FOR ANALYSIS**

PrintWriter output: *output file*

int whitePixel: *number of white pixels on the screen*

String filetype: *output file name*

String filename: *activated or non-activated*

checked against one another, a drag force (that mimics movement of cells on a petri-dish) is applied to each cell and its location is updated based on all the forces on it.

The simulation of progressive cavitation post SCI using the *in silico* astrocyte model runs 30 simulations for each treatment type (active/nonactive macrophage conditioned media). Each simulation starts off at the initial state where the cells are placed on the screen in confluency similar to cells being plated (leftmost image of Figure 2). After settling for 48 hours (middle images in Figure 2), the image is analyzed. The astrocytes’ stepSize changes if the treatment is active macrophage conditioned media or stays the same if the treatment is nonactive macrophage conditioned media. The cells are allowed to move for another 24 hours and the final image is analyzed (rightmost images in Figure 2).
Figure 2: Schematic for methods. The leftmost image is a representation of when the cells are placed on the screen. The middle image shows the cells after settling (at 48 hours) and the rightmost image shows cells 24 hours after treatment (72 hours since plating).

2.3 Image Processing

Images are analyzed using the function analyzeImage() in the main sketch. Pixels are loaded and whitePixels, a variable that counts the amount of empty space, is set to zero. The white pixels represent the background and cavitation (since at plating, as seen in Figure 2, there are some white pixels in the background). Two for loops are used to go through every column and every row of pixels. If the pixel is white, it is turned into a purple pixel (for visual ease of catching incorrectly colored pixels) and the variable whitePixels increases by one. The pixels are then updated using the updatePixels() function to show the purple pixels (see Figure 3).
The accuracy of analyzeImage() function was tested using all the images shown in Figure 3. A, B, C, and G all show the original images that test the accuracy of analyzeImage() in differentiating between white pixels and non-white pixels. These images were also tested using the same blue/green color of the astrocytes in the simulation instead of the black and yielded the same results. Images D, E, F, and H show the white pixels changed to purple pixels. Table 4 is a quantified version of image 3. It shows that A, B, and G all had the correct number of white pixels accounted for while C had 316 white pixels not accounted for the reason for this discrepancy is unknown. However, compared to the overall number of pixels, 316/250000 = 0.002528. Less than 1% of the pixel values are incorrect.
Figure 3. Images used to test analyzeImage(). A, B, C, and G show original images that are used to test the image analysis function. A, B, and G are half white half black while C is a quarter black and three quarters white. Images D, E, F and H are the recolored version of A, B, C, and G. Each purple pixel was once a white pixel which the analyzeImage() function converted to a purple one.
Table 4: Image processing test results. The number of white pixels in A, B, and G were correctly calculated white 316 white pixels were missed in the calculation for image C.

<table>
<thead>
<tr>
<th>Image letter (see Figure 3)</th>
<th>White pixels calculated by analyzeImage() function</th>
<th>Total pixels in image</th>
<th>Actual number of white pixels</th>
<th>Percent error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A (half white)</td>
<td>125000</td>
<td>250000</td>
<td>125000</td>
<td>0</td>
</tr>
<tr>
<td>B (quarter white)</td>
<td>187500</td>
<td>250000</td>
<td>187500</td>
<td>0</td>
</tr>
<tr>
<td>C (half white)</td>
<td>124684</td>
<td>250000</td>
<td>125000</td>
<td>0.2528</td>
</tr>
<tr>
<td>G (half white)</td>
<td>125000</td>
<td>250000</td>
<td>125000</td>
<td>0</td>
</tr>
</tbody>
</table>

The text file produced for each set of 60 simulations (30 for control and 30 for experimental) was created using image processing of each pixel on the display screen. Based on the 50 iterations/hour calculation for simulation time, when a simulation has reached 2400 iterations, that corresponds to 48 hours or start of treatment and the image is analyzed. For the astrocytes, column of pixels is 0-499 and row of pixels is 50-449 to analyze the space where the cells are placed (plated in the middle of the screen; Figure 4). The cells are analyzed again at 72 hours (end of simulation). Cavity area is calculated as (white pixels at time x in active treatment)/(white pixels at time x in nonactive treatment) to give a value for how many folds larger the cavity treated with activated macrophage conditioned media is (x is either 48 or 72 for start and end respectively).

Once the raw data was collected, it was analyzed in excel to obtain the average white space for activated and nonactivated treatments. The average increase in area with control normalized to 1 was calculated for each “active” treatment.

\[
\text{Average increase in area} = \frac{\text{average white pixels in “active” at end of treatment}}{\text{average white pixels in “nonactive” at end of treatment}}
\]
Since white space in the nonactive state is the background and white space in the active state is background + cavity, cavity formation was calculated based on the change in whitespace from the start to end of each treatment. (Cavity formation = white pixels at end – white pixels at start).

![Figure 4. Example of astrocytes image analysis.](image)

**Figure 4. Example of astrocytes image analysis.** The leftmost image is the astrocyte monolayer at 48 hours, directly before the activated macrophage conditioned media is added. The middle image is at 72 hours, 24 hours after activation. The rightmost image shows all the white pixels turned purple. The black box represents the area where the white pixels were evaluated.

### 2.4 Statistics

Statistical analysis was done using R comparing between different aspects of the control astrocyte cultures with nonactive macrophage condition media and the experimental astrocyte cultures with active macrophage conditioned media.

The hypotheses for the data are:

- **Null hypothesis 1 (H₀₁):** There is no difference between the white space at the start points (hour 48) for activated and nonactivated macrophage conditioned media treatments.
- **Alternative hypothesis 1 (Hₐ₁):** There is a difference between the white space at the start points (hour 48) for activated and nonactivated macrophage conditioned media treatments.
- **H₀₂:** There is no difference in cavity area at end point (72 hours) between “active” and “nonactive” treatments.
- **Hₐ₂:** There is a difference in cavity area at end point (72 hours) between “active” and “nonactive” treatments.
- $H_0$: There is no difference between each “active” treatment condition in Fitch et al.’s *in vitro* model and its corresponding “active” treatment condition *in silico*.
- $H_A$: There is a difference between each “active” treatment condition in Fitch et al.’s *in vitro* model and its corresponding “active” treatment condition *in silico*.

To test the first set of hypotheses, $t$-tests with equal variance were run for each of the critical values $1.6, 1.8, \text{and} \ 2.4$ (these correspond with Fitch et al.’s results) for increase in cavity size. Since the purpose of this experiment is to show that the cavity size increases when there is a spinal cord injury (represented by activated macrophage conditioned media), for the second set of hypothesis, the $t$-tests with equal variance were run on the change in cavity size from hour 0 of day three to hour 24 of day three. To account for multiplicity of $p$-values (running three $t$-tests), the Bonferroni correction was applied to the critical $p$-value ($\alpha = 0.05$): $0.05/3 = 0.0167 = \alpha$ (new critical $p$-value). The model’s assumptions were conducted using R version 1.1.463 (See Appendix). Fitch et al. only included the summary of their data. Therefore, to check if increase in cavity size for different levels of activation *in vitro* corresponds with the increase in cavity size of the *in silico* model, a $z$-test was used with their data treated as the population. Average of the population from the *in vitro* model versus the data from the *in silico* model was used to test the third set of hypotheses.
3 Results

I developed an agent based model of progressive cavitation and used it to show that cavity area increases as astrocyte speed increases. A series of 30 simulations for each “active” treatment (activated macrophage conditioned media) was used to establish similarity between the outcomes of my in silico model and Fitch et al’s in vitro model. The control “nonactive” treatment did not cause cavities to form in the astrocyte monolayer. These simulations showed that at 24 hours post “active” treatment, cavities formed (Figure 5). The white space (background) in the control is significantly different than the white space (background + cavities) in the “active” treatments. The agent based model of cavitation closely replicates the findings of Fitch et al.’s in vitro cavitation model. Additionally, though I did not intend to include astrocyte density as a factor in my simulation, the model showed areas where astrocytes are more closely packed together and these increased as astrocyte speed increased.

Qualitatively, my in silico model of progressive cavitation (Figure 5B) compares to Fitch et al’s in vitro model of progressive cavitation (Figure 5A). Figure 5A shows the nuclei of the astrocytes which are closely packed meaning the astrocytes are closely packed. The formation of cavities (identified with asterisks) causes astrocytes to move close together resulting in an increased astrocyte density in some areas (shown by arrows) and no astrocytes in other areas (cavities) within the astrocyte monolayer. Figure 5B is a snapshot of my in silico model at the end of a simulation showing an example of entire astrocytes which received the “active” treatment. Similar to the in vitro model, these astrocytes form cavities which also causes the astrocytes to be closely packed. Although the astrocyte densities in my in silico model were not quantified, Figure 5B shows there are areas where astrocytes are more closely packed together. Though at first glance Fitch et al.’s cavities seem to be larger than the cavities formed by my model it is important to take
into account the scale differences between the two images as well as the distinction between imaging the cell nuclei versus the entire cell.

**Figure 5:** *In vitro vs. in silico cavitation model comparison.* A. shows cavities formed *in vitro* from Fitch et al.’s study. The white dots are the astrocytes’ nuclei (Fitch et al. 1999). Scale bar, 250 μm. B. shows the cavities formed from my simulation. The white circles are the astrocytes (the entire cell). Scale bar, 50 μm. The arrows in both models show areas with increased astrocyte density and the asterisks show cavities formed due to the active macrophages conditioned media.

The values used to test the effect of different speeds on astrocyte cavity area are presented in Figure 6. As speed of astrocytes with the active treatment increased, so did the cavity area. The cavity area consistently increases but once speed is large (around the 0.8 – 0.9 microns/iteration range which is about 40 - 45 microns/hour) the area starts to plateau. Since the cells have a limited area (500x500 display screen) and their overlap is limited, it is expected that at some point increasing the speed will not cause an increase in cavity area.
**Figure 6: Effects of astrocyte speed on increase in cavity area.** As the speed of astrocytes in the activation treatment increases, cavity increases. The cavity formation starts to plateau at 0.8 microns/iteration).

Quantitatively, my in silico model of progressive cavitation (Figure 7B) compares to Fitch et al’s in vitro model of progressive cavitation (Figure 7A). Each of the “active” treatment levels is significantly different than the control “nonactive” treatment. Based on the Bonferroni correction, the critical p-value, $\alpha$, is 0.0167. For all three cavity sizes, we reject the null hypothesis ($H_{02}$) therefore we accept the alternative hypothesis ($H_{A2}$). Increase in cavity size is explained by activation of macrophages for 1.6, 1.8, and 2.4 fold increases in cavitation. 1.6 ($t[58] = -95.321; P < 0.0001$), 1.8 ($t[58] = -140.97; P < 0.0001$), and 2.4 ($t[58] = -139.19; P < 0.0001$). Additionally, for the comparison of white pixels at the start point, we fail to reject the null hypothesis ($H_{01}$). There is no difference between the white space at the start point (0 hours into treatment) for activated and the white space at the start point for nonactivated macrophage conditioned media treatments for all “active” treatment levels. 0.25 microns/iteration ($t[58] = 0.37388; P = 0.7099$), 0.3 microns/iteration ($t[58] = -1.2877; P = 0.203$), and 1.4 microns/iteration ($t[58] = -0.91903; P = 0.3619$).
With each cell density used for the conditioned media *in vitro* there is a corresponding speed that gives a similar cavity area *in silico*. 5 *10^6 macrophages/ mL corresponds to a speed of 0.25 microns/iteration and a 1.6 fold increase in area, 10 *10^6 macrophages/ mL corresponds to a speed of 0.3 microns/iteration and a 1.8 fold increase in area, and 12 *10^6 macrophages/ mL corresponds to a speed of 1.4 microns/iteration and a 2.4 fold increase in area (Appendix - Supplementary Figure S1). Based on the average of the population from the *in vitro* experiment, z-test showed no significant difference between each pair of corresponding values *in vitro* and *in silico*. 0.25 microns/iteration (μ = 1.6, σ = 0.2), z(N=2)= 0.05010038, p = 0.96; 0.3 microns/iteration (μ = 1.8, σ = 0.15), z(N=2)= -0.0962963, p = 0.92; 1.4 microns/iteration (μ = 2.4, σ = 0.3), z(N=2)= 0.08463757, p = 0.92. Preliminary results of changing the spring constant for cells showed that as spring constant increases, the astrocyte cavity area decreases (Figure 8).
**Figure 7: In vitro vs. in silico increase of cavity area.** The *in silico* model (B) closely matches the *in vitro* model (A) described by Fitch et al., providing a measure of success. A. *In vitro* model from Fitch et al.’s experiment. The values next to each activated condition is on the scale of $10^6$ macrophages/mL. Increasing density results in an increase in cavity area. B. *In silico* model with different levels of “active” treatment. Increasing speed of astrocytes results in an increase in cavity area. In panels A and B, all the “active” treatments are significantly different than the control. (\* p<0.05; ** p<0.01; *** p<0.001)
4 Discussion

My model and simulation demonstrated the successful use of an *in silico* model to replicate the studies of an *in vitro* model of progressive cavitation. This novel agent based model used parameters based on *in vivo* and *in vitro* experiments (Table 1) and showed the increasing cavity area was not significantly different to Fitch et al.’s *in vitro* model. The different “active” treatments in the *in silico* model are statistically different than the control “nonactive” treatment (Figures 5 and 7). Preliminary results showed that increasing the spring constant decreases the cavity area in “active” treatments and further investigation is necessary to support this finding. This model provides the possibility of exploring targets for decreasing, and possibly eliminating the process.
of progressive cavitation during the secondary injury in SCI. Once these targets are identified, treatments that attack these targets can be tested in vitro and in vivo to help speed up the pipeline from identifying targets to drug discovery and use in human subjects. With the increasing number of SCI patients and the detrimental effects SCI has on the daily lives of those living with the injury, finding effective treatments for all levels of injury is key. Much of today’s research and treatments focus on preventing further damage to the spinal cord and despite the multitude of drugs and targets that have been tested and continue to be tested, little to none have made it to circulation for SCI patients. These findings emphasize the benefits of in silico modeling to simulate real systems (such as astrocytes’ response to active macrophages) in a controlled setting.

4.1 Limitations

With the use of computer modeling and simulation comes limitation, most of which are unavoidable but can be improved based on new data from the system being modeled. Unlike Fitch et al.’s in vitro model which uses live cells with complex behaviors and compositions, the in silico model described here only has the input given to it. This means that my model exhibits behaviors based on the input data which can be lacking. However, if additional aspects such as new cells and interactions are added, they make for a less intuitive cellular response and a closer replication of the actions and interactions of live cells. Additionally, comparing the results of my model to that of the in vitro model is not a completely equivalent comparison because the in silico definition of levels of activation is based on density of macrophages used to create the conditioned media whereas the in silico levels of activation are based on speed of astrocytes. An assumption I make is that more activation is equivalent to higher speeds. The differences between my model’s “active” and “nonactive” treatments is the speed and because I do not have a way of increasing macrophage
density in the conditioned media in my *in silico* model. I assume that more activation corresponds to faster cells.

### 4.2 Treatment of SCI progression

Understanding the gaps that exist in the field of SCI can be done by following the progression of treatment of SCI in a car accident victim. A car accident occurs and an individual sustains an injury to their spine. The spinal cord that lies inside the spine is compressed and cellular death occurs due to the mechanical injury (as explained in section 1.2 *Spinal Cord Injury*). The spine is then stabilized by first responders to limit further damage to the spinal cord. First responders focus on managing life threats through conducting a quick assessment of the patient’s airway, breathing, and circulation (Martin, Aleksanderek, and Fehlings 2015; Resnick 2013). During this time, though further mechanical injury is not likely to occur, the secondary (chemical) injury is beginning to affect the spinal cord.

When the patient arrives at the hospital, care is transferred to a trauma team which again obtains an overall impression of the patient while managing and presenting life threats. Once the trauma team has stabilized the patient, they are taken for imaging. Up until now, if a patient has sustained a SCI, their body continues experiencing the effects of the secondary injury. Though, the primary injury is not further affecting the patient at this stage, they are already sustaining more damage due to the secondary injury. Additionally, the most commonly used imaging technique, MRI, does not clearly identify what kind of damage is occurring in the spinal cord (Bozzo et al. 2011). It cannot differentiate between cavitation, cell death, and inflammation among other processes, making it difficult to identify what exactly is happening in the spinal cord using an MRI.
In SCI patients, time between injury and interventions is the difference between varying levels of recovery (Dvorak et al. 2015). Depending on the imaging results, the patient may or may not be taken to surgery. Typically if the patient does not need emergency surgery, they are monitored in an Intensive Care Unit (ICU). Sometimes, steroids, such as methylprednisolone (MSSP), which have anti-inflammatory effects, are administered to patients to reduce cell mortality (Bracken et al. 1997). However MSSP is typically only a feasible treatment option within eight hours of initial impact (Ahuja et al. 2016). Again, all throughout this time, the spinal cord is experiencing the detrimental effects of the secondary injury. This is a significant gap that exists in the treatment of SCI patients.

4.3 Benefits of in silico models

 Though many studies have been done on neuroprotective and regenerative drugs, none are currently prescribed to patients with SCI (Ahuja et al. 2016). Most of them have undergone or are in the process of undergoing clinical trials. After being released from the hospital, the patient will typically undergo months of physical therapy to help them regain some of their motor function but most patients never fully recover their typical motor and sensory functions. Further research into understanding how the body reacts to SCI and possible pathways that can be targeted to reduce the impact and magnitude of the secondary injury is key. Additionally, since there exists a multitude of studies and target pathways, a missing component is combination therapy (use of multiple therapies) to focus on both neuroprotection and regeneration post SCI.

*In silico* models, such as the one presented here, provide the possibility of streamlining the exploration process and simulating possible targets to provide a more guided exploration of treatments. This model can be improved to involve other complex aspects of the system (such as
adding macrophages in co-culture, or adding neurons and other glial cells) then used to test the effects of combining therapies and observing how they interact with one another. This also has implications beyond the spinal cord because the *in silico* model has proven to closely replicate the *in vitro* model, these methods of data gathering, model creation, and data input to create simulations can be applied to a wide range of other biological questions. Starting off with one accurate model of a biological system (such as the astrocytes in the spinal cord in this case) can lead to building upon that model and adding levels of complexity to then observe possible scenarios. My model supports the possibility of using *in silico* models to more easily run experiments based on data from the literature and to hypothesize how a model will react based on internal changes or changes within its environment. If *in silico* models are used more often to replicate biological systems, they can help us find new ways of treating patients living with a multitude of diseases and injuries.
5 References


32.


Yuk JCC. 2016. Spinal Cord Modelling for Understanding and Preventing Injury. University of Western Australia Supervisor:


6 Appendix

6.1 Supplemental Figures

Figure S1: In vitro vs. in silico cavity area comparison. The in silico model (in blue) closely matches the in vitro model (in black) described by Fitch et al., providing a measure of success.

6.2 Processing Code

Cell class:

```cpp
/* Cell is the parent class for all types of cells: neurons, astrocytes, and immune cells*/
class Cell {
    float size;
    PVector location;
    PVector velocity;
    PVector acceleration;

    Cell() {
        size = random(10, 30);
        location = new PVector(random(500), random(500));
        velocity = new PVector(0, 0);
        acceleration = new PVector(0, 0);
    }
}
```
//displays the cell as an ellipse if it is alive
void display() {
    fill(1);
    ellipse(location.x, location.y, size, size);
}

//updates the location of the cell based on velocity
void update() {
    velocity.add(acceleration);
    location.add(velocity);
    acceleration.mult(0); // set acceleration to zero so it does not continue speeding up
}

//applies a force to change the acceleration
void applyForce(PVector force) {
    acceleration.add(force);
}

Astrocyte class:

//Astrocytes. The cell body of an astrocyte spans 10-20 microns and
//its processes radiate out for another 20-30 microns, forming the
//stellate glial cell.

class Astrocyte extends Cell {
    float stepSize;

    Astrocyte() {
        size = 20;
        location = new PVector(random(500), random(500));
        velocity = new PVector(0, 0);
        acceleration = new PVector(0, 0);
        //stepSize = 0.2;
        stepSize = 0.0084;
    }

    void display() {
        fill(51, 204, 204);
        stroke(1);
        ellipse(location.x, location.y, size, size);
    }

    //From my lab 2 programming nature course
    void randStep() {
        int xRand = int(random(3))-1; // -1, 0 , or 1
        int yRand = int (random(3)) -1 ; // -1, 0 , or 1
        location.x += stepSize * xRand;
        location.y += stepSize * yRand;
    }

    void checkEdges() {
        location.x = constrain(location.x, size/2, width - size/2);
49
35. location.y = constrain(location.y, size/2, height - size/2);
36. }
37.
38. //based off Nature of Code chapter 3.10
39. //Spring forces
40. PVector bounce(Astrocyte other, float springForce) {
41. float k = springForce;
42. PVector dist = PVector.sub(other.location, location);//how far away the cells currently are
43. float distWant = other.size/2 + size/2; //dist should be at least sum of radii ("rest length" in spring forces)
44. float curLen = dist.mag();
45. 
46. float x = distWant - curLen;
47. dist.normalize();
48. 
49. PVector force = dist.mult(-1*k*x);
50. return force; //return the force PVector so we can add them all
51. }
52. }

Main Sketch:

1. /*
2. migration of astrocytes in the presence of activated macrophages
3. random movement that results in "pockets"
4. cells are "plated" in confluency ==> 80% of the plate is covered
5. */
6.
7. //if size = 20-30, area = 400 -900 so 200,000/400 = 500
8. Astrocyte[] astrocytes = new Astrocyte[500];
9.
10. int state; //if cells are beng plated =0, if cells are moving =1
11. int cellSize = 20;
12. int iter; //number of draw iterations so far
13. int simNum; //simulation number
14. boolean active;
15. int hour;
16. float activeStep; //how much cells moves in randStep if macrophages are activated
17. float springConstant; //the spring constant for the cells
18.
19. int trialNum; //helps keep images folder and textfile connected
20. //ANALYSIS STUFF
21. PrintWriter output; //output file
22. int whitePixels; //number of white pixels in the image
23.
24. String fileType; //activated or non activated...
25. String fileName;
26.
27. void setup() {
28. background(255);
29. size(500, 500); // in microns!!
30. state = 0;
31. simNum = 1;
32. active = false;
33. activeStep = 0.25;
34. springConstant = 0.0001;
void draw() {
    // RUN SIMULATION FOR ASTROCYTES */
    while (simNum <=30) { //go through 30 simulations
        if (state == 0) {
            iter = 0; //set to zero at the start of each new simulation
            for (int i =0; i< astrocytes.length; i++) {
                astrocytes[i] = new Astrocyte();
                int randX = 2*int((random(3))-1); // -1, 0, 1
                int randY = 2*int (random(3)) -1 ; // -1, 0, or 1
                astrocytes[i].location.x = (((i * cellSize)* cellSize/2) % width ) + randX;
                astrocytes[i].location.y = (int(i/(width/cellSize)) * cellSize + cellSize/2) +
                randY + 50;
                astrocytes[i].display();
            }
            state = 1;
        } else {
            background(255);
            bounceCells(astrocytes);
            if (iter>=2400) { //48 hours -> start treatment
                if (active) { //if in active state, change stepSize
                    for (int i =0; i< astrocytes.length; i++) {
                        astrocytes[i].stepSize = activeStep;
                    }
                }
                if (iter%300 == 0) {
                    hour = ((iter/300) * 6) - 48;
                    save("images_trial" +trialNum+ "/" +simNum+ "_" +fileName+ "]Astrocytes" +hour+
                    "hrs.jpeg");
                }
                if (hour == 0) {
                    analyzeImage(); //ANALYSIS
                    output.println(fileName +"_hr_"+ hour + "]" + whitePixels + "]" + pixels.length);
                }
            }
            //for (int i =0; i< astrocytes.length; i++) {
            //    astrocytes[i].stepSize = activeStep;
            //}
        } //next simulation
    } //end simulation
}
91.     // output.close();
92.     //noLoop();
93. } else if (active) {
94.     active = true;
95.     state = 0; //simulation for new type, start at plating again
96.     fileName = "Activated";
97.     //loop();
98. }
99. }
100. }
101. }
102. }
103. //end of while loop
104. output.close();
105. }
106. }
107. void analyzeImage() {
108.     loadPixels();
109.     whitePixels = 0;
110.     for (int x = 0; x<width; x++) {
111.         for (int y = 50; y<450; y++) {
112.             int pixelLoc = x + y * width;
113.             if (pixels[pixelLoc] >= color(150)) {
114.                 whitePixels ++;
115.             } else {
116.                 pixels[pixelLoc] = color(200, 100, 200);
117.             }
118.         }
119.     }
120.     updatePixels();
121. }
122. }
123. void bounceCells(Astrocyte[] astrocytes) {
124.     for (int i =0; i < astrocytes.length; i++) {
125.         astrocytes[i].randStep();
126.         for (int j = i+1; j < astrocytes.length; j++) {
127.             float r1 = astrocytes[i].size/2;
128.             float r2 = astrocytes[j].size/2;
129.             float dist = astrocytes[i].location.dist(astrocytes[j].location);
130.             float distWant = r1 + r2;
131.             if (dist < distWant) {
132.                 PVector force = astrocytes[i].bounce(astrocytes[j], springConstant);
133.                 astrocytes[i].applyForce(force);
134.                 //if one cell got positive force, the other needs to get the negative force...
135.                 astrocytes[j].applyForce(force.mult(-1));
136.             }
137.         }
138.     }
139.     for (int i =0; i < astrocytes.length; i++) {
140.         float c = 0.1; //friction from nature of code 2.7
141.         PVector friction = astrocytes[i].velocity.copy();
142.         friction.mult(-1);
143.         friction.normalize();
144.         friction.mult(c);
145.         astrocytes[i].applyForce(friction);
146.         astrocytes[i].update(); //adds acceleration to velocity and vel to location
147.         astrocytes[i].checkEdges();
6.3 R Code

```r
knitr::opts_chunk$set(echo = TRUE)
library(ggplot2)
library(gridExtra) # used to create a table

# t-test data
call_change <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/all_change.csv")

data for testing assumptions
call_change_anova <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/all_change_anova.csv")

# data for combined plot
call_data <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/plot_data.csv")

# data for separate plots
plot_data_sep <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/plot_data_separate.csv")

# z-test data
call_area <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/ratio_area_critical_vals.csv")

data for t-test on start point for active and nonactive treatments
call_start <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/all_start.csv")

testing assumptions of start data -- NOT NECESSARY THOUGH!!!
call_start_anova <- read.csv(file="~/Desktop/SPROJ/Take2/Analysis/Excel_csv_for_stats/all_start_anova.csv")

# t-tests hypotheses 1

t.test(all_start$start_nonactive_0.25, all_start$start_active_0.25, var.equal = TRUE)

t.test(all_start$start_nonactive_0.3, all_start$start_active_0.3, var.equal = TRUE)

t.test(all_start$start_nonactive_1.4, all_start$start_active_1.4, var.equal = TRUE)
```
#t-tests hypotheses 2

```r
t.test(all_change$change_nonactive_0.25, all_change$change_active_0.25, var.equal = TRUE)
```

```r
model_0.25 <- lm(all_change_anova$whitePixels_0.25 ~ all_change_anova$type_0.25)
anova(model_0.25)
```

```r
t.test(all_change$change_nonactive_0.3, all_change$change_active_0.3, var.equal = TRUE)
```

```r
model_0.3 <- lm(all_change_anova$whitePixels_0.3 ~ all_change_anova$type_0.3)
anova(model_0.3)
```

```r
t.test(all_change$change_nonactive_1.4, all_change$change_active_1.4, var.equal = TRUE)
```

```r
model_1.4 <- lm(all_change_anova$whitePixels_1.4 ~ all_change_anova$type_1.4)
anova(model_1.4)
```

#z-tests hypotheses 3

#this z.test function is a combination from Gabriel Perron's Code and code from the statistical analysis with r for dummies book: https://www.dummies.com/education/math/statistics/z-testing-r/

```r
z.test = function(a, mu, SD) {
#create function and define inputs to provide
# z.statistic = (mean(a) - mu) / SD #compute the function using inputs provided
  #return(z.statistic) #print value computed above#

  z.stat <- (mean(a) - mu) / SD #compute z_score/z_test using inputs provided
  one.tail.prob <- NULL #probability that the z_score is greater than x
  two.tail.prob <- NULL #probability that the z_score is greater than x and less than (-) x

  z.stat <- (mean(a) - mu) / SD #compute z_score/z_test using inputs provided
  one.tail.prob <- pnorm(abs(z.stat), lower.tail = FALSE)
  two.tail.prob <- 2 * one.tail.prob

  cat(" z =", z.stat, "\n", "two-tailed probability =", two.tail.prob, "\n")
  #"one-tailed probability =", one.tail.prob,"\n",
  # cat("two-tailed probability =", two.tail.prob, "\n") #"one-tailed probability =", one.tail.prob,"\n",

```

53
#return(z.stat) #print z_stat computed above

z.test(ratio_area$area_0.25, 1.6, 0.2) #activated - 5 --> area 1.6 speed 0.25
z.test(ratio_area$area_0.3, 1.8, 0.15) #activated - 10 --> area 1.8 speed 0.3
z.test(ratio_area$area_1.4, 2.4, 0.3) #activated - 12 --> area 2.4 speed 1.4

# In Vitro
tickMarks3 <- c("Control", "Activated - 5", "Activated -10","Activated - 12"

ggplot(plot_data_sep, aes(x = Condition_inVitro, y=Area_inVitro)) +
  geom_bar(stat = "identity", fill = "black") +
  xlab("") +
  ylab("Astrocyte Cavity Area") +
  scale_x_discrete(labels = tickMarks3, limits = plot_data_sep$Condition_inVitro)+
  scale_y_continuous(expand = c(0,0), breaks=seq(0,3.5,0.2), limits = c(0,3))
  + #y from 0 - 3 increasing by 0.2
  scale_colour_manual(values=c("black")) + #CHANGE COLORS (USE RGB COLOR PICKER)
  geom_errorbar(aes(ymin=Area_inVitro-StandardError_inVitro, ymax=Area_inVitro+StandardError_inVitro), width=.2,position=position_dodge(.9)) +
  annotate("text", x = c("Activated - 5","Activated - 10", "Activated - 12"),
  y = c(2.2,2.2,2.8), label = c("","",""))
  # annotate("text", x = c("Control", "Activated - 5","Activated - 10", "Activated - 12"), y = c(2.8, 2.2,2.2,2.8), label = c("A", ",","","")) +
  # coord_cartesian(ylim = (0, 3))+
  theme_classic()

#In Silico
tickMarks2 <- c(expression("Control 0.084"*mu"m"),
  expression("Activated 0.25"*mu"m"),
  expression("Activated 0.3"*mu"m"),
  expression("Activated 1.4"*mu"m"))

ggplot(plot_data_sep, aes(x = Condition_inSilico, y=Area_inSilico)) +
  geom_bar(stat = "identity", fill = 
#56B4E9") +
  xlab("") +
  ylab("Astrocyte Cavity Area") +
  annotate("text", x= c("Activated 0.25", "Activated 0.3", "Activated 1.4"),
  y = c(2, 2.2, 2.6), label = c("","",""))
  #annotate("text",x = c("Control 0.084","Activated 0.25", "Activated 0.3", 
"A
activated 1.4"), y = c(2.7, 2.2, 2.6), label = c("B", "***", "***", "***") +
  scale_x_discrete(labels = tickMarks2, limits = plot_data_sep$Condition_inSilico) +
  scale_y_continuous(expand = c(0,0), breaks=seq(0,2.7,0.2), limits = c(0,2.8)) +
  geom_errorbar(aes(ymin=Area_inSilico-StandardError_inSilico, ymax=Area_inSilico+StandardError_inSilico), width=.2,position=position_dodge(.9)) +
  theme_classic()

#COMBINED
 tickMarks <- c("Control",expression("Control 0.084\*\mu\"m\""), "Activated - 5", expression("Activated 0.25\*\mu\"m\""), "Activated -10", expression("Activated 0.3\*\mu\"m\""), "Activated -12", expression("Activated 1.4\*\mu\"m\""))

#legend inside - BEST ONE
 ggplot(plot_data, aes(x = Condition, y=Area, fill=Type)) +
  geom_bar(stat = "identity") +
  xlab("") +
  ylab("Astrocyte Cavity Area") +
  scale_x_discrete(labels = tickMarks, limits = plot_data$Condition)+
  scale_y_continuous(expand = c(0,0), breaks=seq(0,3,0.2))+
  scale_fill_manual(values=c("#56B4E9", "black"))+
  theme_classic()+
  theme(legend.position = c(0.3, 0.9), legend.direction = "horizontal", legend.text=element_text(face = "italic"), axis.text.x = element_text(angle=90))+
  labs(fill = "Model") #change legend title

#TESTING ASSUMPTIONS (repeated for all three speeds)

#get unstandardized predicted and residual values
 unstandardizedPredicted <- predict(model_0.25)
 unstandardizedResiduals <- resid(model_0.25)

#get standardized values
#This will allow you to look at the residuals for any analysis on common axis with a mean of zero.
 standardizedPredicted <- (unstandardizedPredicted - mean(unstandardizedPredicted)) / sd(unstandardizedPredicted)
 standardizedResiduals <- (unstandardizedResiduals - mean(unstandardizedResiduals)) / sd(unstandardizedResiduals)

#create residuals histogram
 hist(standardizedResiduals, freq = FALSE)
#add normal curve
curve(dnorm, add = TRUE)

par(mfrow=c(1,1))
plot(model_0.25)