**Project Progress Report: 2. Impact of Spiral Ganglion Cell Changes on Excitability**

Matt Nagle, Daria Nesterovich, Kristie Yang

**Objective**

This project will model how morphological changes in human Spiral Ganglion Cells (SGCs) following deafness affect cell excitability. Our goal is to determine the effects of degeneration and demyelination in SGC cell bodies and peripheral processes on SGC activation. Using the NEURON simulation environment, we plan to quantitatively model the response of a normal SGC, a deafened SGC morphology, three tonotopically organized SGCs, and a rotationally symmetric expansion of three SGC model. With deafness, there is an associated degeneration of the peripheral process, manifested as a decrease in length, as well as an increase in demyelination along the length of the central process and the cell body of the SGC.

**Initial Developments**

Following the preliminary research completed in conjunction with the proposal phase of the project, the initial task was to find literature values for membrane parameters associated with a normal human SGC cell. This task proved more challenging than anticipated, as there were no previous empirical studies that characterized the membrane parameters in human SGCs . As a result, the membrane parameter Cm= 2 F/cm2was chosen based on the parameter value measured in amphibian neurons. Additionally, the value of extracellular resistivity e=70 cmwas also chosen based on the amphibian parameters [1]. Although this proves a limitation to the current working model, sensitivity analysis will be conducted to examine the effects of varying membrane parameters on degenerated SGC excitation.

The current model is composed of a single healthy myelinated axon, consisting of 20 nodes, representing the peripheral process of the SGC. Each node consisted of 15 segments. The diameter of the fiber was set as 1.4 m,which was measured as the mean diameter of a dendrite in the basal turn of a human SGC [2]. The inner diameter and internodal length were calculated using the assumptions d = 0.7\*D and L = 100\*D, respectively. Assuming the myelin sheath acts as a perfect insulator allowed for a simplification in building the peripheral process model. The simplified model consists of nodes connected with resistors, where the axial resistance representing the myelinated sections is incorporated into the axial resistance of the nodal sections using the equation:



Although this does not accurately represent the geometry of a physical neuron, it allows for the use of the NEURON simulation environment. In order to integrate spatial geometry into the model of the peripheral process, we created a vector calculating nodal position in space. This vector, containing spatial coordinates of the nodal positions, was subsequently used to calculate extracellular voltages at the node locations. A single point source electrode was placed at 100 um from the 11th node of the peripheral process. The potential field due to a monopolar point source was calculated using the equation:



where *r* represents the distance between the node and the electrode. **Figure 1** displays the extracellular potential field in space due to a monopolar electrode. Figure 2 shows the successful initiation of an action potential due to the extracellular source.

The peripheral process was activated using a suprathreshold stimulus and the resulting membrane voltages were calculated as a function of space at multiple time points. **Figure 3** illustrates the initiation and propagation of an action potential at node 11, the central node lying directly beneath the point source electrode. The shape of the membrane voltage envelope is consistent with change in membrane potential under cathodic stimulation. The multiple spikes overlaying the envelope are a result of the influence of the extracellular potential on Vm at the nodes. At the moment, extracellular stimulus remains ON for the duration of the recording. In the future we plan to implement extracellular stimulations for variable durations.

A subsequent step in the modeling process will be to incorporate both myelinated and unmyelinated SGCs. Approximately 5% of the SGCs in a healthy cat remain unmyelinated [3]. Because this percentage is so small, we have decided that complete myelination of a single axon can be used to model a healthy patient for our initial testing. Upon further testing, we may populate the model with enough SGCs to recreate this percentage. It has been found that after four years of induced deafness, the unmyelinated percentage increases to 42.4% [3]. This has been hypothesized to be caused by decreased fitness of myelinated SGCs and demyelination of previously myelinated SGCs. This conversion or death of myelinated SGCs occurs via demyelination increments, where Schwann cells detach in patches randomly, in what can be characterized as an all or none process [4]. In order to model this demyelination, we have created a process that will randomly choose whether the internodal sections will be myelinated--and therefore modelled by a resistor--or unmyelinated--and therefore modelled by a “long node.” The probability of these events will be changed in the future, but right now it is set at 50%, a value chosen to test the setup and model a midway point in the progression of the onset of deafness. **FIGURE 4** shows the topology output of the NEURON code for the randomized SGC, with patchy sections of demyelination. Further modification and verification is needed to test this new SGC, as varying the length values of the node-resistor-node connections and node-unmyelinated-node segments has proved to be difficult.

**Ongoing and Next Steps**

Ongoing investigations include the addition of a soma to the single SGC model, along with variation in the nseg value of a peripheral process. The latter models the decreasing peripheral processes associated with deafness [5]. Initial testing of these changes has not yielded the expected results, and debugging has been inconclusive.

The group then plans to see the effects of moving the electrode closer and farther to the implemented soma. This can be verified with the papers and findings discussed in lecture, where action potentials start in the axon, and oscillation minimums in threshold occur at the nodes. From this result, we will be confident enough to simulate basal, middle, and apical turns of the cochlea with three SGCs.  Then we will move away from a single point source to a more appropriate distribution. The final steps will be to Expand the model to multiple SGCs per cochlear turn at the appropriate densities in a radial symmetric geometry. From these results and sensitivity analysis, we should be able to provide an insightful recommendation for improving the efficacy of cochlear stimulation in deafened individuals.

**References**

[1] Cartee, Lianne A. "Spiral ganglion cell site of excitation II: numerical model analysis." *Hearing research* 215.1 (2006): 22-30.

[2] Nadol, J. B., Jr. (1990) Degeneration of cochlear neurons as seen in the spiral ganglion of man. Hear Res 49: 141-154.

[3] Leake et al (1987) Cochlear pathology of long-term neomycin induced deafness in cats. Hear Res 1988  33(1):11-33.

[4] Cohen, G. M., Park, J. C., & Grasso, J. S. (1990). Comparison of demyelination and neural degeneration in spiral and Scarpa's ganglia of C57BL/6 mice. *Journal of electron microscopy technique*, *15*(2), 165-172.

[5] Briaire, Jeroen J., and Johan HM Frijns. "The consequences of neural degeneration regarding optimal cochlear implant position in scala tympani: a model approach." *Hearing research* 214.1 (2006): 17-27.

*Figure 1*. Plots of Extracellular Potential Induced by Point Source Extracellular Electrode (Cathodic Source). Subplots include Extracellular Potential in the extracellular space and as seen by each node along the length of the Peripheral Process.

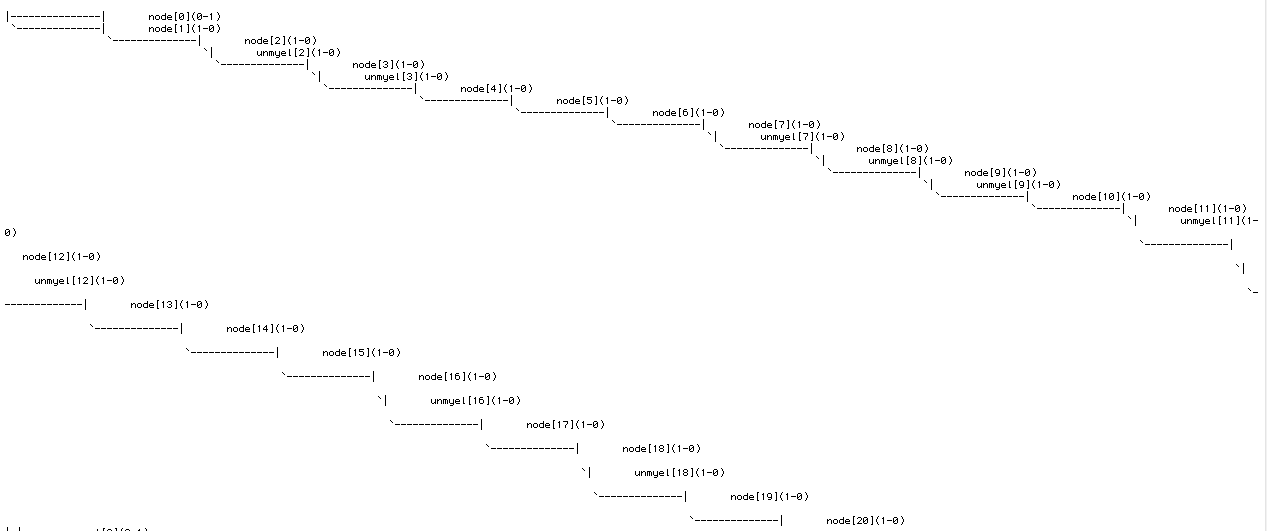


*Figure 2*. Plots of Extracellular Potential, Second Difference, and Induced Action Potential. Action Potential data were collected at Node 11 (loc: 1419 µm along 2680 µm process)





*Figure 3*. Plots of Membrane Potential Along the Length of the Peripheral Process. Each subplot displays a different snapshot in time to show action potential propagation at a single node (node 11). Note, time = 0 is the beginning of the extracellular stimulus.



*Figure 4*. Cell topology of a cell who’s sections are randomized for normal myelination and demyelination. The probability of demyelination in this example is 50%.

**Appendix**

**1. MyelinatedAxonSimpleExtra.hoc**

//Myelinated Axon, Perfect Insulator, With Extra Stim

//based on Modeling extracellular stimulation and recording with NEURON

//by Ted

//<http://www.neuron.yale.edu/ftp/ted/neuron/extracellular_stim_and_rec.zip>

load\_file("nrngui.hoc")

load\_proc("nrnmainmenu")

create node[1]

objref nodes

vinit = -65 //mV

// buildaxon creates an axon that has 21 compartments, 20 nodes

proc buildaxon() {local k

nnode = $1

create node[nnode]

nodes = new SectionList()

forsec "node" nodes.append

access node[0]

for k = 0, nnode-2{

connect node[k+1](1),node[k](0)

}

}

proc physprop(){

forsec nodes{

diam = 0.7\*1.4//um

l = 1//µm

L = 100\*1.4 //µm

nseg = 15

define\_shape()

}

}

proc membprop(){

forsec nodes{

insert hh

insert extracellular

Rm = 1000

Ra = 100\*((l+L)/l)/1e4

cm = 2

}

}

proc make() {

buildaxon($1)

physprop()

membprop()

}

make(21) //create 21 nodes and 20 internodes

objref pos\_nx, pos\_ny, pos\_nz

pos\_nx = new Vector(nnode-1)

pos\_ny = new Vector(nnode-1)

pos\_nz = new Vector(nnode-1)

for i= 0, nnode-2 {

pos\_nx.set(i, l\*(i+1)+i\*L)

pos\_ny.set(i, 0)

pos\_nz.set(i, 0)

}

// set electrode stim \*\*\*\*\*

AMP = -.1 //mA

DUR = 1

DEL = 1

dt = 0.01

tstop = 10

objref Istep

Istep = new Vector(tstop/dt)

for i = 0, tstop/dt-1 {

if (i <= (DEL+DUR)/dt && i >= DEL/dt){

Istep.set(i,AMP)

}else {

Istep.set(i,0)

}

}

// end set electrode stim \*\*\*\*\*

// set electrode pos \*\*\*\*\*\*\*

rho = 70 //Ohm cm //ex\_resistivity

XE = 1410 // µm

YE = 100 // µm

ZE = 0 // µm

objref dist\_r, xtra\_pot

dist\_r = new Vector(nnode-1)

xtra\_pot = new Vector(nnode-1)

for k = 0, nnode-2 {

dist = sqrt((pos\_nx.get(k) - XE)^2 + (pos\_ny.get(k) - YE)^2 + (pos\_ny.get(k) - ZE)^2)

dist\_r.set(k,dist)

ve = (AMP \* rho / 4 / PI)\*(1/dist)\*10^4

xtra\_pot.set(k,ve)

}

// end set electrode pos \*\*\*\*\*\*

for j = 0, nnode-2 {

access node[j]

node[j].e\_extracellular(1) = xtra\_pot.get(j)

print e\_extracellular

}

objref vec, g2

g2 = new Graph()

g2.size(0, 2680, -70,10)

vec = xtra\_pot

vec.plot(g2, pos\_nx, 3,2)

objref p, vnodes

p = new Graph()

p.size(0,tstop,-90,100)

vnodes = new Vector()

vnodes.record(&node[11].v(1))

run()

vnodes.plot(p,dt,3,2)

objectvar rvp, g

rvp = new RangeVarPlot("v")

node[0] rvp.begin(0)

node[20] rvp.end(1)

g = new Graph()

g.addobject(rvp)

g.size(rvp.left(), rvp.right(),-90,50)

**2. MyelinatedAxonSimple\_IStep.hoc**

//Myelinated Axon, Perfect Insulator, With Extra Stim and Additional Plots

//based on Modeling extracellular stimulation and recording with NEURON

//by Ted

//<http://www.neuron.yale.edu/ftp/ted/neuron/extracellular_stim_and_rec.zip>

load\_file("nrngui.hoc")

load\_proc("nrnmainmenu")

create node[1]

objref nodes

vinit = -65 //mV

// buildaxon creates an axon that has 21 compartments, 20 nodes

proc buildaxon() {local k

nnode = $1

create node[nnode]

nodes = new SectionList()

forsec "node" nodes.append

access node[0]

for k = 0, nnode-2{

connect node[k+1](1),node[k](0)

}

}

proc physprop(){

forsec nodes{

diam = 0.7\*1.4//um

l = 1//µm

L = 100\*1.4 //µm

nseg = 15

define\_shape()

}

}

proc membprop(){

forsec nodes{

insert hh

insert extracellular

Rm = 1000

Ra = 10000\*((l+L)/l)/1e4

cm = 2

}

}

proc make() {

buildaxon($1)

physprop()

membprop()

}

make(21) //create 21 nodes and 20 internodes

objref pos\_nx, pos\_ny, pos\_nz

pos\_nx = new Vector(nnode-1)

pos\_ny = new Vector(nnode-1)

pos\_nz = new Vector(nnode-1)

for i= 0, nnode-2 {

pos\_nx.set(i, l\*(i+1)+i\*L)

pos\_ny.set(i, 0)

pos\_nz.set(i, 0)

}

// set electrode stim \*\*\*\*\*

AMP = -.1 //mA

DUR = 3

DEL = 1

dtime = 0.025

ttot = 20

tstop = ttot

objref Istep

Istep = new Vector(tstop/dtime)

for i = 0, tstop/dtime-1 {

if (i <= (DEL+DUR)/dtime && i >= DEL/dtime){

Istep.set(i,AMP)

}else {

Istep.set(i,0)

}

}

// end set electrode stim \*\*\*\*\*

// set electrode pos \*\*\*\*\*\*\*

rho = 70 //Ohm cm //ex\_resistivity

XE = 1410 // µm

YE = 100 // µm

ZE = 0 // µm

objref dist\_r, xtra\_pot

dist\_r = new Vector(nnode-1)

xtra\_pot = new Vector(nnode-1)

// Point in time over Space

for k = 0, nnode-2 {

dist = sqrt((pos\_nx.get(k) - XE)^2 + (pos\_ny.get(k) - YE)^2 + (pos\_ny.get(k) - ZE)^2)

dist\_r.set(k,dist)

ve = (AMP \* rho / 4 / PI)\*(1/dist)\*10^4

xtra\_pot.set(k,ve)

print xtra\_pot.get(k)

}

// end set electrode pos \*\*\*\*\*\*

// Point in Space over time \*\*\*\*

objref dist\_at\_node

dist\_at\_node = new Vector(nnode-1)

objref xtra\_pot\_at\_node

xtra\_pot\_at\_node = new Matrix(tstop/dtime,nnode-1)

for j = 0, nnode-2 {

node\_num = j

dist\_at\_node.set(j,sqrt((pos\_nx.get(node\_num) - XE)^2 + (pos\_ny.get(node\_num) - YE)^2 + (pos\_ny.get(node\_num) - ZE)^2))

for k = 0, tstop/dtime-1 {

ve = Istep.get(k) \* (rho / 4 / PI)\*(1/dist\_at\_node.get(j))\*10^4

//ve = AMP \* (rho / 4 / PI)\*(1/dist\_at\_node.get(j))\*10^4

xtra\_pot\_at\_node.setval(k,j,ve)

}

}

objref f

f=new File()

f.wopen("Ve\_AtNode\_OverTime.txt")

xtra\_pot\_at\_node.fprint(f)

f.close()

// end point in Space over time \*\*\*\*

// plot extracellular pot at each node

/\*

for j = 0, nnode-2 {

access node[j]

time\_point = (DUR+DEL)/dtime-1

//tstop = time\_point

node[j].e\_extracellular(0) = xtra\_pot\_at\_node.getval(time\_point,j)

}

\*/

// Matt's plot

objref vec, g2

g2 = new Graph()

g2.size(0, 2680, -70,10)

vec = xtra\_pot

vec.plot(g2, pos\_nx, 3,2)

// end Matt's plot

// end plot extracellular pot at each node

objref p, vnodes

p = new Graph()

p.size(0,tstop,-90,100)

vnodes = new Vector()

vnodes.record(&node[11].v(1))

run()

vnodes.plot(p,dtime,3,2)

print "size:"

print vnodes.size()

// show how node changes over time

/\*

objref pos\_vec

pos\_vec = new Vector(pos\_nx.get(nnode-2))

for i=0, pos\_vec.size()-1 {

pos\_vec.set(i,i)

}

objref rvp2, vspace\_mat, vspace\_vec, g3,vnode2

vspace\_mat = new Matrix(int(((DUR+9)/dtime)/50)+2,pos\_vec.size()-1)

vspace\_vec = new Vector(pos\_vec.size()-1)

vnode2 = new Vector()

node\_num = 11

access node[node\_num]

count = 0

for(k = DEL/dtime; k < (DEL+DUR+9)/dtime-1; k = k + 50) {

tstop = (k)\*dtime

if (k < (DEL+DUR)/dtime){

for i=0,nnode-2 {

node[i].e\_extracellular(1) = xtra\_pot\_at\_node.getval(k,i)

}

}

run()

rvp2 = new RangeVarPlot("v")

node[0] rvp2.begin(1)

node[20] rvp2.end(0)

g3 = new Graph()

g3.addobject(rvp2)

g3.size(rvp2.left(),rvp2.right(),-90,50)

rvp2.to\_vector(vspace\_vec)

vspace\_mat.setrow(count,vspace\_vec)

count = count + 1

}

objref f

f=new File()

f.wopen("Vm\_AtONENode\_OverALLTime.txt")

vspace\_mat.fprint(f)

f.close()

\*/

// end show how node changes over time

// write Vm matrix --> matlab

/\*

objref vnode, vnode\_mat

vnode\_mat = new Matrix(tstop/dtime+1,nnode-1)

vnode = new Vector(tstop/dtime+1)

for j = 0, nnode-2 {

for k = 0, tstop/dtime-1 {

access node[j]

node[j].e\_extracellular(1) = xtra\_pot\_at\_node.getval(k,j)

}

vnode.record(&node[j].v(.5))

run()

vnode\_mat.setcol(j,vnode)

}

objref f

f=new File()

f.wopen("Vm\_AtNode\_OverTime.txt")

vnode\_mat.fprint(f)

f.close()

\*/

// end write Vm matrix --> matlab

objectvar rvp, g

rvp = new RangeVarPlot("v")

node[0] rvp.begin(0)

node[20] rvp.end(1)

g = new Graph()

g.addobject(rvp)

g.size(rvp.left(),rvp.right(),-90,50)