

THE EFFECT OF FIBER DEPTH ON THE ESTIMATION OF PERIPHERAL
NERVE FIBER DIAMETER USING GROUP DELAY AND SIMULATED
ANNEALING OPTIMIZATION

A Thesis

presented to

the Faculty of the Biomedical/General Engineering Department at

California Polytechnic State University, San Luis Obispo

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Engineering

by

Nam Tran

June 2014

© 2014

Nam Tran

ALL RIGHTS RESERVED

COMMITTEE MEMBERSHIP

TITLE: The effect of fiber depth on the estimation of peripheral nerve fiber diameter using group delay and simulated annealing optimization

AUTHOR: Nam Tran

DATE SUBMITTED: June 2014

COMMITTEE CHAIR: Dr. Robert Szlavik
Associate Professor of Biomedical Engineering

COMMITTEE MEMBER: Dr. Lily Laiho
Associate Professor of Biomedical Engineering

COMMITTEE MEMBER: Dr. David Clague
Associate Professor of Biomedical Engineer

ABSTRACT

The effect of fiber depth on the estimation of peripheral nerve fiber diameter
using group delay and simulated annealing optimization

Nam Tran

Peripheral neuropathy refers to diseases of or injuries to the peripheral nerves in the human body. The damage can interfere with the vital connection between the central nervous system and other parts of the body, and can significantly reduce the quality of life of those affected. In the US, approximately between 15 and 20 million people over the age of 40 have some forms of peripheral neuropathy. The diagnosis of peripheral neuropathy often requires an invasive operation such as a biopsy because different forms of peripheral neuropathy can affect different types of nerve fibers. There are non-invasive methods available to diagnose peripheral neuropathy such as the nerve conduction velocity test (NCV).

Although the NCV is useful to test the viability of an entire nerve trunk, it does not provide adequate information about the individual functioning nerve fibers in the nerve trunk to differentiate between the different forms of peripheral neuropathy. A novel technique was proposed to estimate the individual nerve fiber diameters using group delay and simulated annealing optimization. However, this technique assumed that the fiber depth is always constant at 1 mm and the fiber activation due to a stimulus is depth independent. This study aims

to incorporate the effect of fiber depth into the fiber diameter estimation technique and to make the simulation more realistic, as well as to move a step closer to making this technique a viable diagnostic tool.

From the simulation data, this study found that changing the assumption of the fiber depth significantly impacts the accuracy of the fiber diameter estimation. The results suggest that the accuracy of the fiber diameter estimation is dependent on whether the type of activation function is depth dependent or not, and whether the template fiber diameter distribution contains mostly large fibers or both small and large fibers, but not dependent on whether the fiber depth is constant or variable.

ACKNOWLEDGMENTS

This work is dedicated to my family, and all those who have helped me achieve so much over the years. My success thus far is not just the result of my personal abilities, but also the result of the love and support of my family, professors, advisors, and friends. I am thankful for having them in my life.

It was a privilege to learn from one of the best advisors, Dr. Robert Szlavik. His guidance and enthusiasm has helped me overcome so many obstacles to complete this work. He has pushed me hard but it was well worth it. I am also very honored to have Dr. David Clague and Dr. Lily Laiho evaluating my work and traveling the last milestone of my journey at Cal Poly with me.

And lastly, I would like to acknowledge and express my gratitude for the generous financial support that Cal Poly has given me to complete my graduate coursework.

TABLE OF CONTENTS

List of Tables	ix
List of Figures	x
Chapter 1. Introduction	1
1.1 Peripheral Neuropathies	3
1.2 Nerve Conduction Studies	5
Chapter 2. Methods	8
2.1 Study Overview	8
2.2 Assumptions	9
2.3 Generation of Template Nerve Fiber Diameters and Depths	10
2.4 Estimating the Activation of Nerve Fibers and Compound Action Potential	14
2.5 Estimating the Peripheral Nerve Fiber Diameter Distribution by Group Delay Measurements	17
2.6 Simulation	23
Chapter 3. Results	26
3.1 Depth Variability	27
3.2 Distribution Types	29
3.3 Activation Functions	31
Chapter 4. Discussion	34
Chapter 5. Insights about the Simulation and Future Work	42
5.1 Problem of Negative Group Delay Estimates	42

5.2 Poor Performance for High Fiber Depth Variability	46
5.3 Future Work.....	49
Chapter 6. Conclusion.....	51
Chapter 7. Literature cited	53
Appendix A - Minitab ANOVA Results for Chi-square Goodness of Fit Test	
Values for Fiber Diameter Estimation as a Function of Depth	
Variability, Template Fiber Diameter Distribution Types, and	
Activation Function Types	59
Appendix B - Minitab ANOVA Results for Number of Fibers Activated	
during Simulation of Fiber Diameter Estimation as a Function of	
Depth Variability, Template Fiber Diameter Distribution Types, and	
Activation Function Types	62
Appendix C – Data Collected from the Simulations	65
Appendix D – MATLAB Code Used in Simulation.....	69
Appendix E – MATLAB Code used for Debugging and Diagnostic.....	89

LIST OF TABLES

Table 1. Classification of peripheral nerves [2]	4
Table 2. Parameters used for generating a complete template fiber diameter distribution	10
Table 3. Parameters used to generate the template fiber diameter distribution containing large diameter fibers only	13
Table 4. The simulation groups conducted in this study, where the depth variability, diameter distribution types, and activation functions are factors that can affect the chi-square goodness of fit values for the estimated fiber diameters. Each simulation group has $n = 10$	24

LIST OF FIGURES

Figure 1. A typical nerve conduction velocity setup with a stimulus electrode and two recording electrodes at distances $d_1 = 3$ cm and $d_2 = 5$ cm away from the stimulus site, respectively	6
Figure 2. Structure of a large myelinated nerve fiber, where the nerve conduction jump from one node of Ranvier to another and which speeds up the conduction velocity.....	7
Figure 3. A sample of a complete template distribution of nerve fiber diameters generated using the input parameters in Table 2 in Matlab.....	11
Figure 4. A sampled population of nerve fiber depths generated from a uniform distribution using Matlab.....	12
Figure 5. A sample of a large distribution of nerve fiber diameters generated using the input parameters in Table 3 in Matlab.....	13
Figure 6. Flow chart of the group delay estimation from the compound evoked potentials, recorded at two recording sites, when stimulating a population of nerve fibers from 0 to 1 mA, with an incremental step of 5 μ A [3]	21
Figure 7. Flowchart of the simulated annealing optimization algorithm to compute an improved estimation of the nerve fiber diameter in the population set \hat{d} from the group delay estimated population set \tilde{d} [12]	22
Figure 8. Plot of the chi-square goodness of fit results as a function of the number of fibers activated during the simulation for all combinations of fiber depth variability, diameter distribution types, and activation	

functions (Table 4). There is a weak correlation observed,

$R^2 = 0.35704$27

Figure 9. Plot of mean chi-square goodness of fit results for different combinations of depth variability and template diameter distributions. There is no significant difference between the different depth variability, p-value = 0.399. There is a significant difference in the mean chi-square goodness of fit results for the different types of template fiber diameter distribution used, p-value ≈ 0.00 28

Figure 10. Plot of mean number of fibers activated during simulation for different combinations of depth variability and template fiber diameter distributions29

Figure 11. Plot of mean chi-square goodness of fit results for different combinations of template fiber diameter distribution and activation functions. There is a significant difference between the types of template fiber diameter distribution, p-value ≈ 0.00 , and between the types of activation functions, p-value ≈ 0.000 . And there is a significant interaction between the types of template fiber diameter distribution and the types of activation functions, p-value = 0.00730

Figure 12. Plot of mean number of fibers activated during simulation for different combinations of template fiber diameter distribution and activation functions31

Figure 13. Plot of mean chi-square goodness of fit results for different combinations of fiber depth variability and activation functions. There

is no significant difference between the different depth variability,
 p-value = 0.399. However, there is a significant difference in the types
 of activations used, p-value \approx 0.000. The combination of variable fiber
 depth and depth independent activation function was not considered
 because the fiber depth is assumed to have an effect on the
 activation of the nerve fibers when it is not constant32

Figure 14. Plot of mean number of fibers recruited during simulation for
 different combinations of fiber depth variability and activation functions..33

Figure 15. Histograms of a complete template nerve fiber diameter
 population d and the group delay estimated nerve fiber diameter
 population \tilde{d} for variable fiber depth, complete template fiber
 diameter distribution, and depth dependent activation function. The
 chi-square goodness of fit result for the two distributions is
 $Q(\chi^2|x) = 0.1086$ 38

Figure 16. Histograms of the template nerve fiber diameter population d
 generated from the large fiber distribution and the group delay
 estimated nerve fiber diameter population \tilde{d} for variable depth, large
 template fiber diameter distribution, and depth independent
 activation function. The chi-square goodness of fit result for the two
 distributions is $Q(\chi^2|x) = 0.1513$39

Figure 17. Plot of the template compound evoked potential and the group

delay estimated compound evoked potential generated from the simulation of variable fiber depth, complete template fiber diameter distribution, and depth dependent activation function	40
Figure 18. Plot of the template compound evoked potential and the group	
delay estimated compound evoked potential generated from the simulation of variable fiber depth, large template fiber diameter distribution, and depth dependent activation function	41
Figure 19. Plot of the phase response as a function of the frequency used in	
the group delay estimation. Each line represents the phase response in the group delay estimation of a pair of decomposed single fiber action potential waveforms. The span = 15×10^{-3} s	43
Figure 20. Plot of the template compound evoked potential and the group	
delay estimated compound evoked potential generated from the simulation with span = 15×10^{-3} s. The estimated compound evoked potential is the flat line at zero.	44
Figure 21. Plot of the phase delay as a function of the frequency used in the	
group delay estimation. Each line represents the phase delay in the group delay estimation of a pair of decomposed single fiber action potential waveforms. The depth variability is between 1 – 10 mm	46
Figure 22. Plot of the template compound evoked potential and the group	
delay estimated compound evoked potential simulated for fiber depth variability between 1 – 10 mm	48

1. Introduction

In the human body, the peripheral nerves connect the central nervous system to muscles, organs, and other body tissues [1]. Damage to the peripheral nerves, also known as peripheral neuropathy, can interfere with the vital connections between the central nervous system and other parts of the body to cause pain, numbness, muscle weakness, and reduce the quality of life [1]. In the United States, between 15 and 20 million people over the age of 40 have some form of peripheral neuropathy [2]. Currently, one of the most common non-invasive methods to diagnose peripheral neuropathy is the nerve conduction velocity test [3].

Although the nerve conduction velocity test is useful in determining the viability of peripheral nerves, it does not produce any information about the characteristics of the individual functioning nerve fibers because it only evaluates the gross conduction properties of the underlying nerve trunk [3]. In previous nerve conduction velocity studies, it was commonly assumed that fibers in the same velocity class have the same evoked potential waveforms for the estimated conduction velocity distribution (CVD) [4, 5, 6, 7]. The estimated CVD is the electrophysiological counterpart of the morphological fiber diameter distribution [8]. Additional information, such as the size of individual nerve fibers that contribute to the compound evoked potential, can help differentiate the clinical conditions because different types of nerve fibers are affected differently

depending on the underlying clinical conditions [3]. Differentiating surviving fibers based on their size can assist in the diagnosis between chronic inflammatory demyelinating polyneuropathy, which damages the larger myelinated nerve fibers, and early diabetic peripheral neuropathy, which damages smaller unmyelinated fibers [9, 10, 11].

A more robust method using group delay and simulated annealing optimization that can yield a size distribution of the underlying nerve fibers was first proposed by Szlavik [3, 12]. This method uses an estimation of the group delay between two recording electrodes, also known as the phase distortion of the signals travelling down the nerve trunk as a function of the frequency [12]. The group delay information is then used to estimate the diameters of the nerve fibers. However, this estimation technique did not take into consideration the effect of the depth of the nerve fibers (the perpendicular distance from the stimulating electrode to the fiber), and instead assumed it was always constant at 1 mm. This study aims to incorporate the effect of nerve fiber depth into the model and assess the performance of the fiber diameter estimation technique with the additional variable through simulation. The goal of this study is to gain additional insight about the effectiveness of this estimation technique for the purpose of developing it into a viable diagnostic tool.

1.1 Peripheral Neuropathies

Peripheral neuropathies describe the damage or impairment of peripheral nerves [13]. The most common causes for peripheral neuropathies are injuries, underlying diseases, infections, or hereditary conditions [1]. Injuries can cause nerve damage through physical contacts, such as severing or crushing the nerves. Nerve damage caused by injuries is easy to diagnose because the injuries are often visible and the cause is readily known. On the other hand, nerve damage caused by underlying diseases is often harder to diagnose because the underlying causes are often not visible and yet to be diagnosed.

One of the common underlying diseases that can cause peripheral neuropathy is diabetes [14]. The National Institutes of Health estimated that 60-70% of diabetic patients have some measurable form of neuropathy [15]. One form of diabetic neuropathy is chronic inflammatory demyelinating polyneuropathy (CIDP) that impairs the sensations and diminishes tendon reflexes [9]. CIDP affects 0.5 per 100,000 children and 1-2 per 100,000 adults [14, 16, 17]. According to the American Academy of Neurology, it is mandatory to obtain the cerebrospinal fluid, and a nerve biopsy specimen to make a definitive diagnosis of the disease [18]. The non-invasive nerve conduction velocity test can be used to diagnose CIDP, but it does not distinguish CIDP from other types of neuropathies that are not affected by the demyelinating process, which is a symptom of CIDP (Table 1) [2].

Type	Structure and functions
Large nerve fibers	Heavily myelinated A-alpha fibers mediate motor strength A-beta fibers mediate vibratory and touch sensation
Medium nerve fibers	Myelinated A-gamma fibers carry information to muscle spindles
Small nerve fibers	Unmyelinated C fibers and myelinated A-delta fibers mediate pain, thermal sensation and autonomic function

Table 1. Classification of peripheral nerves [2]

A common underlying disease that affects small unmyelinated fibers is early diabetic peripheral neuropathy (EDPN). Because the symptoms of EDPN are very similar to CIDP, it is difficult to accurately diagnose the condition without performing additional invasive tests, and the results from the nerve conduction test alone is not definitive enough [10, 11]. Many experts recommend a skin biopsy to evaluate the density of nerve fibers in the epidermis for small fiber neuropathies diagnoses [2]. Such invasive tests are costly and expose patients to risks. Therefore, the goal of developing a fiber diameter estimation technique using group delay measurements and simulated annealing optimization is to provide clinicians with a non-invasive diagnostic tool that has the same level of accuracy.

1.2 Nerve conduction studies

The conduction velocity distribution estimation method uses two compound action potentials, recorded at two recording electrodes, to estimate the conduction velocity distribution, using a least squares approach (Figure 1) [19, 10, 8]. There are various digital signal processing techniques used to estimate the conduction velocity distribution [8]. However, due to the limitations in stimulating and recording action potentials of the small myelinated and unmyelinated fibers, the available methods can only analyze the activity of the large myelinated fibers because the nerve conduction measurement is largely composed of the large myelinated diameter fibers of the nerves [8, 10].

In most common nerve conduction study setups and techniques, a stimulating electrode sends a stimulating pulse to the nerve trunk (Figure 1). Two recording electrodes, placed at known distances away, will pick up the signal and measure how long it takes for the signal to travel from the first recording electrode to the second recording electrode (Figure 1). Since the distance and time to travel between two recording electrodes are known, the nerve conduction velocity (NCV) is obtained by [20]:

$$\text{NCV (m/s)} = \frac{\text{Distance between the recording electrodes}}{\text{Conduction time}} \quad (1)$$

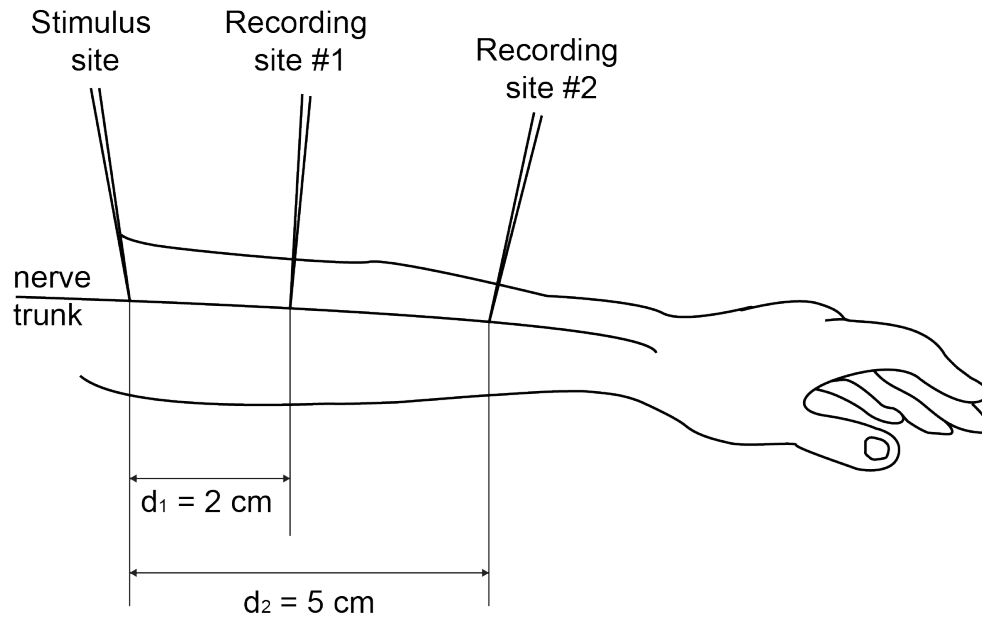


Figure 1. A typical nerve conduction velocity setup with a stimulus electrode and two recording electrodes at distances $d_1 = 3 \text{ cm}$ and $d_2 = 5 \text{ cm}$ away from the stimulus site, respectively.

The conduction velocity of a nerve is dependent on its size and myelination [20]. The large myelinated fibers conduct faster because they are insulated by the myelin sheath [20]. The gaps in the myelin sheath, also known as the Nodes of Ranvier that exist between the myelin sheath cells along the axon of the nerve fibers, enable an action potential traveling down the axon to jump from node to node and speed up the conduction velocity (Figure 2) [20]. This propagation is also known as saltatory conduction [20]. Thus, in patients with demyelinated nerve fibers, the NCV is much slower (approximately 30m/s) than the normal conduction velocity (approximately 45m/s) [20]. Since the

demyelinating process does not affect the small unmyelinated nerve fibers, the NCV does not slow down significantly (approximately 35m/s or more) [20].

Therefore, the nerve conduction test is sensitive only to the abnormalities in large nerve fibers. For example, a patient can have damage to small unmyelinated nerve fibers and the nerve trunk can still achieve 35m/s or more in conduction velocity as long as he or she has all the large myelinated nerve fibers functioning. In order to diagnose peripheral neuropathy more accurately and non-invasively, a more robust method that could objectively evaluate the function of all individual nerve fibers contributing to the overall compound action potential is needed.

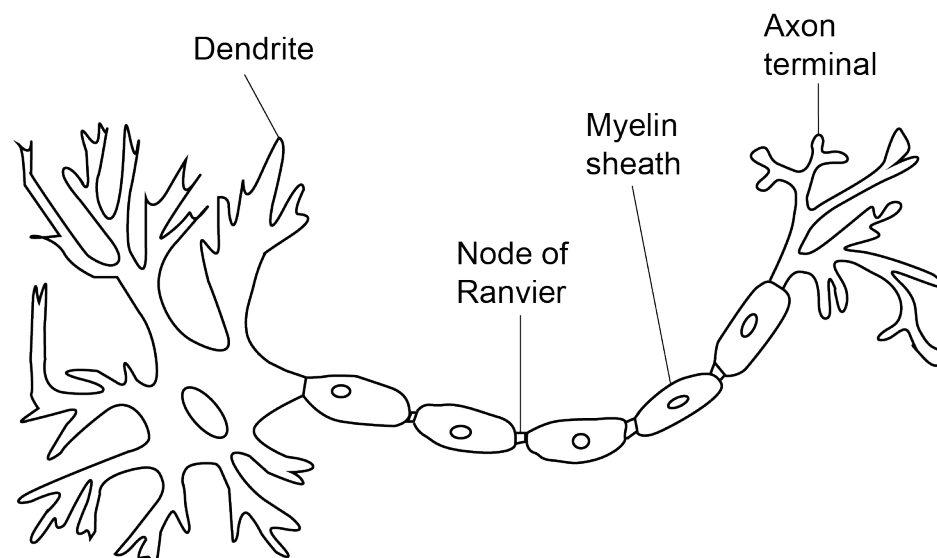


Figure 2. Structure of a large myelinated nerve fiber, where the nerve conduction jump from one node of Ranvier to another and which speeds up the conduction velocity.

2. Methods

2.1 Study Overview

This study will examine the effect that incorporating the nerve fiber depth has on the performance of the estimation of peripheral nerve fiber diameter using the group delay and simulated annealing optimization. In the previous simulation studies, the activation function of a nerve fiber was only dependent on its diameter [3; 12]. The estimation technique in this study deviates from previous studies by using a different activation function that depends on both the diameter and depth of the nerve fiber [3, 12, 21]. In addition, the depth of the nerve fiber is no longer constant as in the previous simulation for the purpose of calculating the fiber evoke potential [3; 12]. The depth of the nerve fiber will vary to mimic the actual human anatomy, thus making the simulation more realistic.

The technique of estimating peripheral nerve fiber diameter using group delay and simulated annealing optimization is still in the conceptual development stage and requires further validation. In addition, it is not guarantee that the available equipment used in the nerve conduction test are compatible with this estimation technique. Therefore, it is practical to study the modifications to the technique via simulation instead of an animal model. All simulations are performed with Matlab (version R2013a).

2.2 Assumptions

This study follows closely the assumptions made in the previous simulation studies to estimate the nerve fiber diameter using group delay and simulated annealing optimization [3; 12]:

- Once stimulated, each active nerve fiber will transmit the action potential at the same time from the same site as the stimulating electrode. In the human body, the difference in time and location could be very small, and is therefore negligible.
- There is a fixed value for the threshold current to excite a nerve fiber of a specific size and at a specific depth. In reality, the activation threshold fluctuates over a small range.
- The conduction velocity is constant along a nerve.
- The conduction velocity and fiber diameter are linearly related. This assumption is not valid if there are disruptions of the myelin, causing the relationship to become nonlinear. This is possible in patients with partially demyelinated fibers as a result of their underlying clinical conditions.
- An isotropic condition exists for the surrounding tissue, and minimizes the effect on the activation threshold of nerve fibers.
- The fiber depth is no longer assumed to be constant at 1mm. Instead, the fiber depth will vary between 1 – 4mm, following a uniform distribution.

2.3 Generation of Template Nerve Fiber Diameters and Depths

The generation of a template population of 100 nerve fibers follows an empirically determined nerve fiber diameter distribution and a technique to generate a distribution of fiber diameters with weighted modes [22, 23]:

$$p_d(d_k) = \sum_{h=1}^4 \frac{\beta_h}{\sigma_h \sqrt{2\pi}} \exp \left[-\frac{(d_k - \mu_h)^2}{2\sigma_h^2} \right] \quad (2)$$

Symbol	Quantity	Value
β_1	1 st mode scaling constant	0.05 m
σ_1	1 st mode standard deviation	0.1274 μm
μ_1	1 st mode mean	0.5 μm
β_2	2 nd mode scaling constant	0.25 m
σ_2	2 nd mode standard deviation	0.8493 μm
μ_2	2 nd mode mean	3 μm
β_3	3 rd mode scaling constant	0.3 m
σ_3	3 rd mode standard deviation	1.699 μm
μ_3	3 rd mode mean	7.5 μm
β_4	4 th mode scaling constant	0.4 m
σ_4	4 th mode standard deviation	1.699 μm
μ_4	4 th mode mean	13 μm

Table 2. Parameters used for generating a complete template fiber diameter distribution

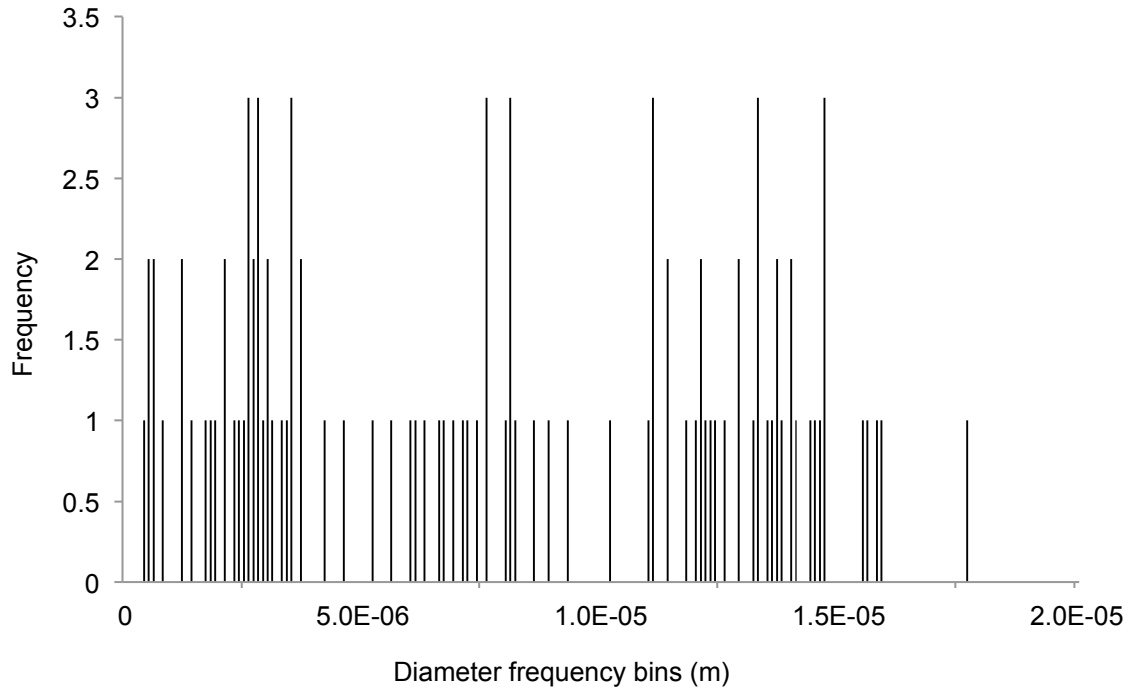


Figure 3. A sample of a complete template distribution of nerve fiber diameters generated using the input parameters in Table 2 in Matlab.

This study assumes that the nerve fiber depth has a uniform probability distribution. In other words, it is equally likely to find a fiber at any particular depth, regardless of its size. All template nerve fiber diameter and depth distributions are generated using Matlab.

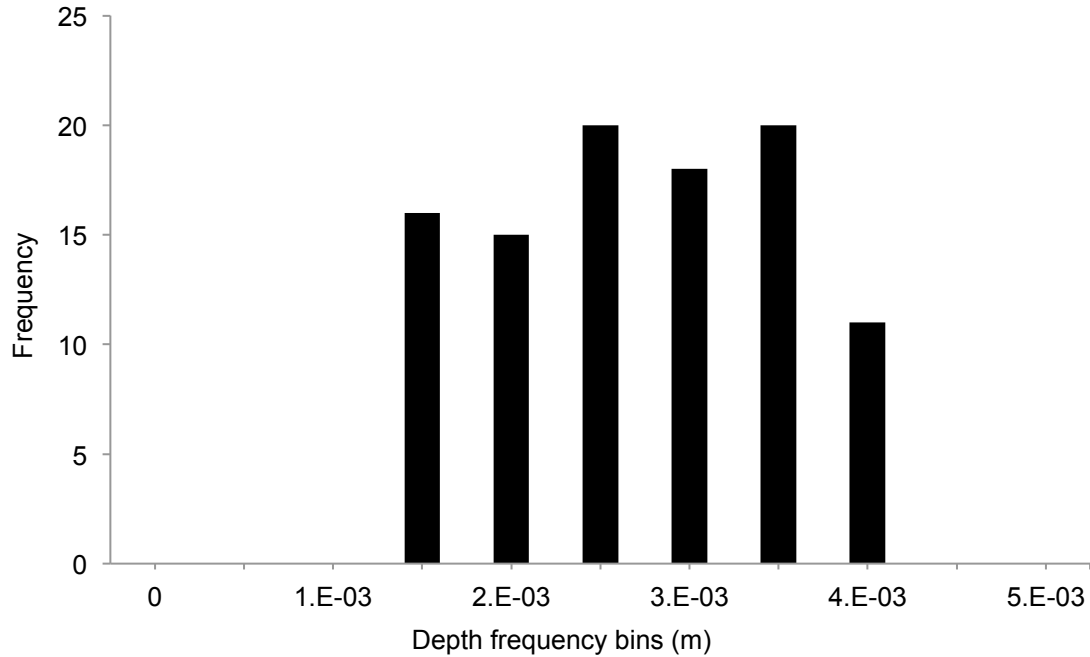


Figure 4. A sampled population of nerve fiber depths generated from a uniform distribution using Matlab.

In addition, this study will also look at the effectiveness of the estimation technique for a population consisting of mostly large nerve fibers (Table 3). In the previous simulation studies, there were two types of template diameter distributions used in the simulation: complete and large [3; 12]. Because this study aims to make improvements on the estimation technique, it is useful for comparisons to simulate similar conditions as in previous studies.

Symbol	Quantity	Value
β_1	1 st mode scaling constant	0.3 m
σ_1	1 st mode standard deviation	1.699 μm
μ_1	1 st mode mean	7.5 μm
β_2	2 nd mode scaling constant	0.7 m
σ_2	2 nd mode standard deviation	1.699 μm
μ_2	2 nd mode mean	13 μm

Table 3. Parameters used to generate the template fiber diameter distribution containing large diameter fibers only

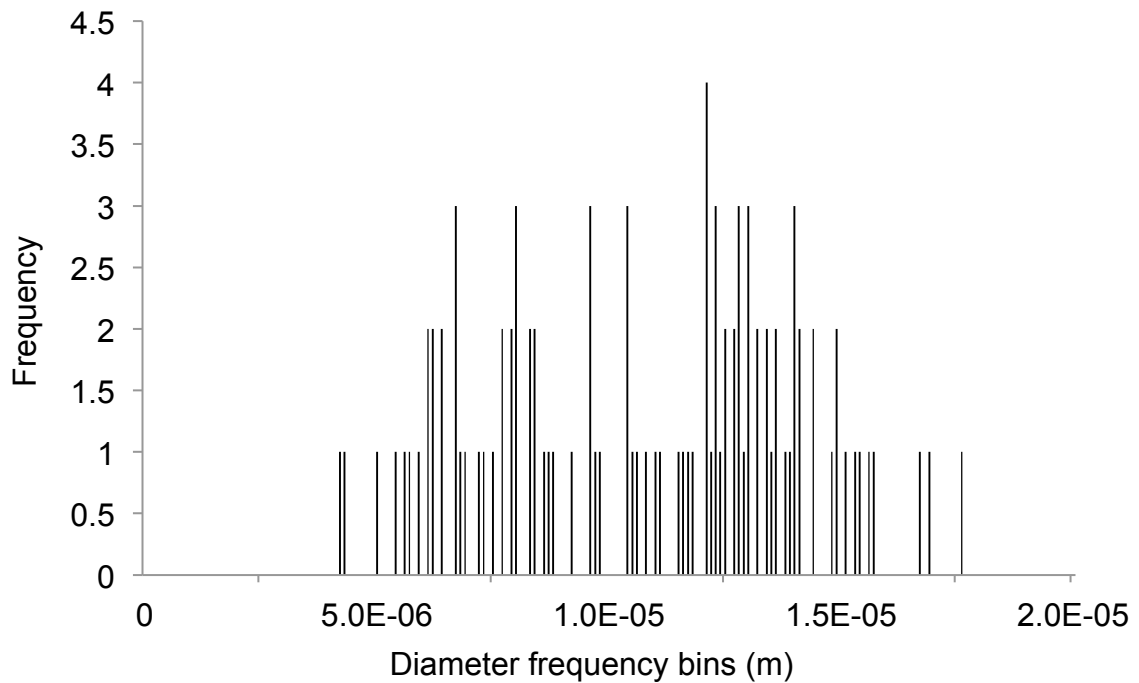


Figure 5. A sample of a large template distribution of nerve fiber diameters generated using the input parameters in Table 3 in Matlab.

2.4 Estimating the Activation of Nerve Fibers and Compound Action Potential

In the previous study, the activation of a nerve fiber was a function of the fiber diameter only [3]. This study aims to make the model more realistic by incorporating a previously determined activation function, $\xi(d)$, that included the effect of the fiber depth [21]:

$$\xi(d) = A(0.291r^2 + 13.566r - 1.083rd - 56.530d + 2.246d^2 + 305.315 + c_e) \quad (3)$$

r – the fiber depth (mm); d – the fiber diameter (μm)

A – the contact area between the stimulating electrode and the skin. In this study, the contact area is assumed to be $1 \times 10^{-4} \text{ m}^2$, and the dimensions of the stimulating electrode are 1 cm x 1 cm.

c_e – the correction constant equals to 53.355. The activation function proposed in the previous study is an estimate of a best-fit regression [21]. This estimation produced a function that has possible outputs that are less than zero.

Physiologically, a nerve fiber cannot be activated by a stimulus current less than zero. Therefore, in this study, a correction constant is added to make the output of the activation function valid for all inputs. Although, the activation threshold will be increased for all fiber diameters and depths, the overall shape and behavioral patterns of the activation function are the same so the correction constant will have minimal effect on the overall behavior of the estimation model.

The correction constant is equal to the absolute value of the minimum of the proposed activation function:

$$\min(0.2912r^2 + 13.5664r - 1.0835rd - 56.5306d + 2.2463d^2 + 305.3158) \approx -53.355$$

$$\text{at } (r = 0.209700303, d = 12.633622).$$

The stimulating electrode stimulates the nerve trunk starting at $I_0 = 0$ mA to $I_f = 1$ mA, with increment $\Delta I = 0.5$ μ A. For each stimulus level i , the simulation will compute a compound evoked potential for each recording site $n = 1, 2$ [3]:

$$\psi_i^{(n)}(t) = \sum_{k=1}^m u[\Omega_i - \xi(d_k)] G[v_k(t - \delta_k^{(n)}), \bar{r}] \quad (4)$$

t – time in seconds

v_k – the conduction velocity of the k^{th} fiber

$\delta_k^{(n)}$ – the propagation delay in seconds of the single fiber action potential from the stimulus electrode to the n^{th} recording site.

\bar{r} – the perpendicular depth between the center of the k^{th} fiber to the recording site.

$G[v_k(t - d_k^{(n)}), \bar{r}]$ – the single fiber action potential waveform contributing to the compound evoked potential when the step function, u , is positive.

G – the single fiber action potential model [27].

$$\begin{aligned}
G[v_k(t - d_k^{(n)}), \bar{r}] = & \\
& \frac{D_k^2}{4\pi\sigma_e a_k} \left[\alpha \exp\left\{-\left(\frac{D_k}{4}\right)^2 \left(\frac{v_k(t - \delta_k^{(n)}) + s_k}{a_k}\right)\right\} - \exp\left\{-\left(\frac{D_k}{4}\right)^2 \left(\frac{v_k(t - \delta_k^{(n)}) - s_k}{a_k}\right)\right\} + \right. \\
& \left. (1 - \alpha) \exp\left\{-\left(\frac{D_k}{4}\right)^2 \left(\frac{v_k(t - \delta_k^{(n)}) - u_k}{a_k}\right)^2\right\} \right] \quad (5)
\end{aligned}$$

The difference in the compound action potentials at steps Ω_i and Ω_{i-1} can be decomposed into a series of waveforms:

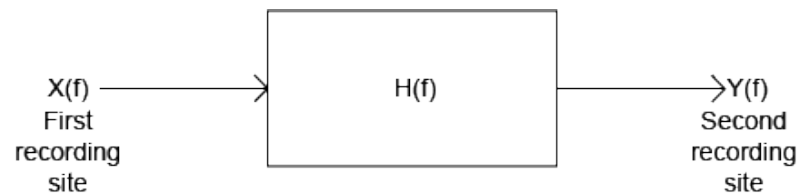
$$\Gamma_{i-1}^{(n)}(t) = \Psi_i^{(n)}(t) - \Psi_{i-1}^{(n)}(t) \quad \text{for } 2 \leq i \leq q+1 \quad (6)$$

If the incremental increase in stimulus current, ΔI , is small enough, the waveform, $\Gamma^{(n)}(t)$, can consist of either the single fiber action potentials associated with the most recently recruited fiber, or no waveform because the last increase in stimulus current does not recruit any additional fiber. Since the stimulus increases by a fixed amount at each step, a perfect decomposition is not always possible. It is possible that some of the non-zero $\Gamma^{(n)}(t)$ waveforms can contain more than one single fiber action potential because the incremental increase in stimulus current, ΔI , is large enough to recruit more than one additional fiber.

2.5 Estimating the Peripheral Nerve Fiber Diameter Distribution by Group Delay Measurements

The recent work by Szlavik outlined a technique to estimate the size distribution of contributing nerve fibers that is linearly related to the conduction velocity distribution [3]. This technique relies on an estimation of the group delay of the signal between the two recording electrodes that are arranged in a configuration analogous to the ones used in the nerve conduction velocity test (Figure 1) [3]. The group delay is a measure of the phase distortion of the signal traveling through a system as a function of the frequency.

Consider a system $H(f)$ with the input $x(f)$, and the output $y(f)$:



The group delay associated with each contributing nerve fiber can be estimated from the decomposed waveforms that nominally consist of the contributing single fiber action potentials $I^{(n)}(t)$ from the two recording sites. The frequency response of a given fiber $H_{i-1}(f)$ is equal to the Fourier transform of the single fiber evoked potential at the recording site 2 divided the Fourier transform of the single fiber evoked potential of the recording site 1:

$$H_{i-1}(f) = \frac{\mathbf{F}[\Gamma_{i-1}^{(2)}(t)]}{\mathbf{F}[\Gamma_{i-1}^{(1)}(t)]} \quad (7)$$

Each frequency response $H_{i-1}(f)$ has a magnitude response and a phase response, thus:

$$H(e^{j\omega}) = |H_{i-1}(f)| \angle \theta_{i-1}(f) \quad (8)$$

For each pair of non-zero decomposed waveforms $\Gamma_{i-1}^{(1)}(t)$ and $\Gamma_{i-1}^{(2)}(t)$, the group delay can be estimated by:

$$\tau_{i-1} = - \frac{1}{2\pi} \frac{d\theta_{i-1}(f)}{df} \quad (9)$$

In practice, to facilitate the estimation of the group delay τ_{i-1} for each pair of non-zero decomposed waveforms $\Gamma_{i-1}^{(1)}(t)$ and $\Gamma_{i-1}^{(2)}(t)$, a least squares line is fit to the phase response θ_{i-1} . The diameters of the fibers can be computed from the estimated group delay:

$$d_{i-1} = \frac{l}{c\tau_{i-1}} \quad (10)$$

where l is the distance between 2 recording electrodes (m) and the constant $c = 5.0 \times 10^5 \text{ s}^{-1}$. The diameter distribution computed by the group delay estimation is evaluated by the chi-square test against the template diameter distribution.

The estimated group delay and fiber diameter distribution data are then used in the simulated annealing optimization algorithm to improve the accuracy of the estimation of the fiber diameter distribution [12]. Finding an optimal solution for a problem with large number of possible solutions can be difficult, or impossible, within a reasonable length of time. For the purpose of diagnosing peripheral neuropathy, where time and user-friendliness are important, it is practical to use the simulated annealing optimization process because it uses less computing power and converges relatively quickly on an optimal solution.

The simulated annealing algorithm mirrors the annealing process in metal work [24]. The annealing process in metal consists of heating and controlled cooling a material to alter its physical properties by changing its internal structure. As the metal cools, its new structure gets locked in place and the metal retains the newly obtained properties. Ideally, if the rate of cooling can be controlled and lengthened, the metal can have a higher yield strength and tensile strength than when the metal is cooled rapidly. In simulated annealing, a variable called temperature is varied to simulate the heating process [24]. At high temperature, the algorithm is allowed to accept solutions that are worse than the current solution at a high frequency, allowing the algorithm to jump out of any local minimums it finds itself in initially [24]. As the temperature decreases, the chance of accepting worse solutions decreases allowing the algorithm to narrow down the search space, so that a solution close to the optimum can be found [24].

In this study, the simulated annealing method varies the fiber diameter and the time delay of the fiber evoked potential, for a randomly chosen fiber in the population. However, because the sample data used in this study is non-deterministic, it is necessary to have a suitable time reference or temporal marker δ_j that is inherent to sampled data [12]. A relevant temporal marker that can be used is the centroid $\Gamma_j^{(n)}(t)$ of the absolute value of the decomposed single fiber action potential [25]:

$$\delta_j = \frac{\int_{-\infty}^{\infty} |t \Gamma_j^{(n)}(t)| dt}{\int_{-\infty}^{\infty} |\Gamma_j^{(n)}(t)| dt} \quad (11)$$

The functions evaluated in the centroid expression must be greater than zero for all t [12]. The simulated annealing optimization algorithm is implemented according to the flowchart (Figure 7) [12]. The algorithm is applied only to the template maximal compound evoked potential at the first recording site in this study. In practice, either recording site can be used.

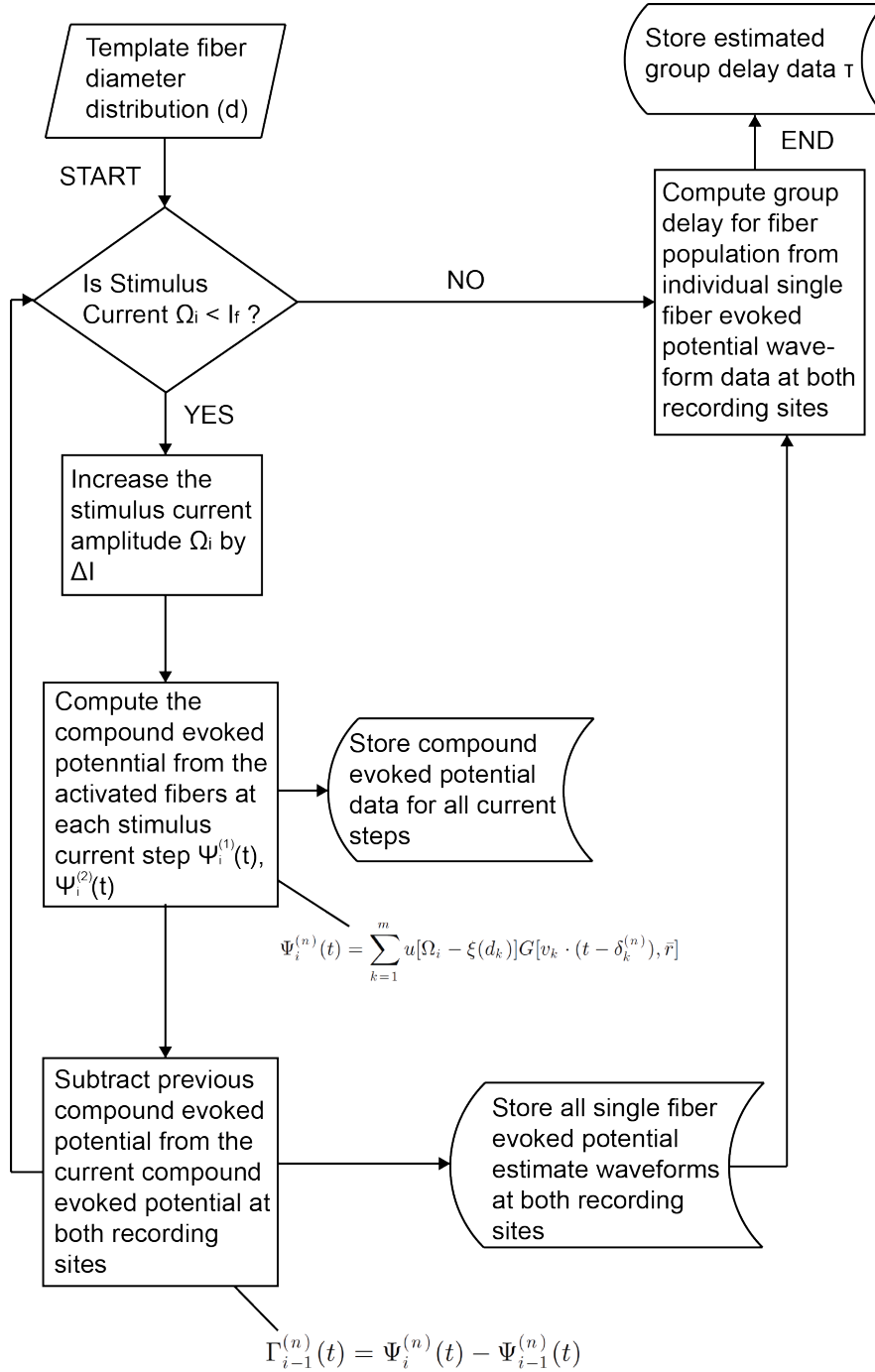


Figure 6. Flow chart of the group delay estimation from the compound evoked potentials, recorded at two recording sites, when stimulating a population of nerve fibers from 0 to 1 mA, with an incremental step of 5μA [3].

2.6 Simulation

This study develops a simulation of the estimation of nerve fiber diameter using group delay and simulated annealing optimization that investigates three characteristics: fiber depth variability, diameter distribution types, and activation functions (Table 4).

In order to explore the effect of fiber depth variability, both constant and variable fiber depth cases will be simulated (Table 4). This study specifies the depth variability to be either constant at 1mm, similar to the previous studies, or variable between 1 – 4mm. Following the previous studies, this study also examines at the effect of different types of template fiber diameter distributions, complete and large (Table 4) [3; 12].

Since this study no longer assumes that the fiber depth is always constant and has no effect on the activation threshold of the nerve fibers, this study will also examine the effect of the different types of activation functions (Table 4). Specifically, the effect of the diameter-and-depth dependent activation function versus the diameter-only dependent activation function will be examined. In addition, this study doesn't consider the combination of variable fiber depth and depth independent activation function because the fiber depth is assumed to have an effect on the activation of the nerve fibers if it is to vary, consistent with results of the previous study [21].

In the previous study, it was determined that the stimulus current required to activate a particular fiber increases as the depth increases [21]. Therefore, as the average depth of the fibers increases due to more variability in fiber depths, it is expected that fewer fibers get recruited during the simulation. Since the single fiber potential waveforms are important inputs for the fiber diameter estimation algorithm, a few number of fibers recruited could reduce the accuracy of the estimation.

Simulation groups	Fiber depth variability	Distribution types	Activation function
1	Constant at 1mm	Complete	Depth and diameter dependent
2	Constant at 1mm	Large	Depth and diameter dependent
3	Variable 1 - 4mm	Complete	Depth and diameter dependent
4	Variable 1 - 4mm	Large	Depth and diameter dependent
5	Constant at 1mm	Complete	Diameter dependent only, depth independent
6	Constant at 1mm	Large	Diameter dependent only, depth independent

Table 4. The simulation groups conducted in this study, where the depth variability, diameter distribution types, and activation functions are factors that can affect the chi-square goodness of fit values for the estimated fiber diameters. Each simulation group has $n = 10$.

The chi-square goodness of fit test will assess the performance of the fiber diameter estimation. This study uses the chi-square probability distribution function $Q(\chi^2|x)$ [26]. $Q(\chi^2|x)$ is the probability that the observed chi-square will exceed the value χ^2 [26]. For two identical distributions, $Q(\chi^2|x) = 1$ [26]. The chi-square goodness of fit test is calculated in Matlab using the same algorithm as in previous studies [3; 12].

The effects of the fiber depth variability, diameter distribution types, and activation functions on the accuracy of the fiber diameter estimation will be compared using the general linear model ANOVA in Minitab (version 16). The chi-square goodness of fit result, $Q(\chi^2|x)$, is the response variable and the fiber depth variability, diameter distribution types, and activation functions are predictor variables. The null hypothesis is that there is no difference in the means of the chi-square goodness of fit value for different combinations of fiber depth variability, diameter distribution types, and activation functions. The alternative hypothesis is that the means chi-square goodness of fit are not all the same for the different combinations of fiber depth variability, diameter distribution types, and activation functions. The general linear model ANOVA was also used to perform pairwise comparisons among the variables.

3. Results

The general linear model ANOVA results show that the different combinations of fiber depth variability, diameter distribution types, and activation functions do not yield the same mean chi-square goodness of fit results for fiber diameter estimation. The $R^2 = 0.3979$ suggests that 39.79% of the variation in the data is explained by the ANOVA model.

There is a trend that the more fibers recruited, the better the chi-square goodness of fit value for the fiber estimation (Figure 8). The $R^2 = 0.357$ suggested a weak correlation, but it is expected for a simulation using non-deterministic samples. It is observed that the results of groups 2, 4 and 6, which use the large template fiber diameter distributions, contain more variation in the number of fibers recruited than groups 1, 3, and 5, which use the complete template fiber diameter distributions (Figure 8). In addition, when considering the mean number of fibers recruited as a response variable of fiber depth variability, diameter distribution types, and activation functions, the ANOVA results show the same trend as for the mean chi-square goodness of fit results (Appendix B).

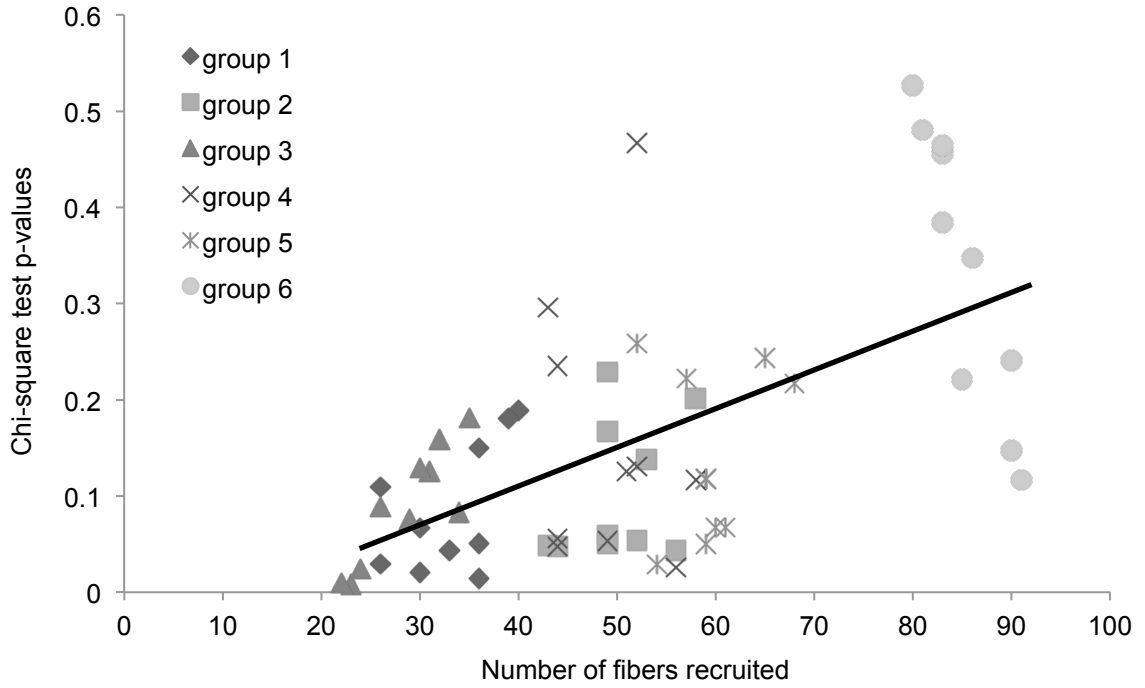


Figure 8. Plot of the chi-square goodness of fit results as a function of the number of fibers activated during the simulation for all combinations of fiber depth variability, diameter distribution types, and activation functions (Table 4). There is a weak positive correlation observed, $R^2 = 0.35704$.

3.1 Depth Variability

At a glance, the chi-square goodness of fit result suggests that as the depth variability increases and no longer stays constant at 1 mm, the accuracy of the fiber diameter estimation decreases (Appendix A) (Figure 9). This trend is also observed in the number of fibers recruited, as the depth variability increases, there are fewer fibers recruited (Appendix B) (Figure 10). However, the ANOVA result suggests that when taking into account the fiber diameter distribution

types, and activation function types, there is no significant difference in the chi-square goodness of fit value between the fiber depth constant at 1mm and fiber depth varies between 1 – 4mm (Appendix A) (p -value = 0.399).

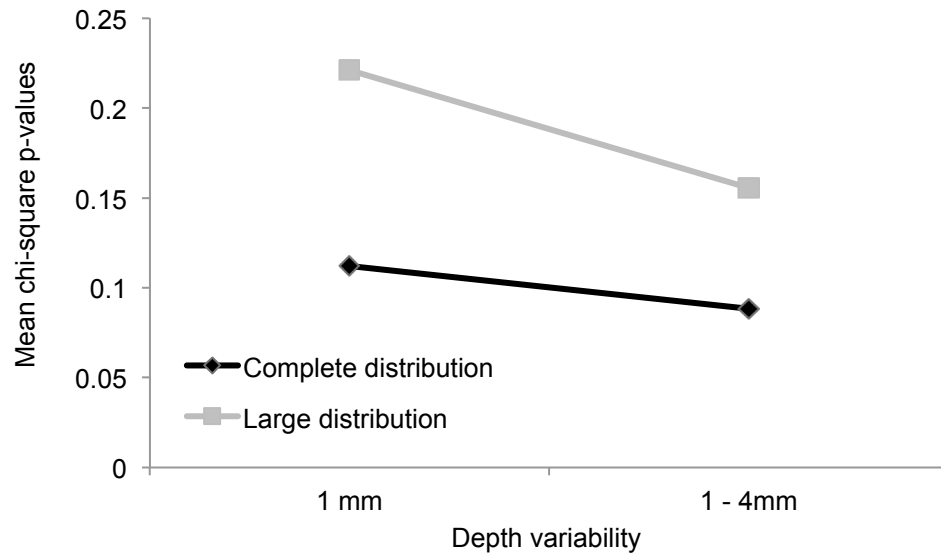


Figure 9. Plot of mean chi-square goodness of fit results for different combinations of depth variability and template diameter distributions. There is no significant difference between the different depth variability, p -value = 0.399. There is a significant difference in the mean chi-square goodness of fit results for the different types of template fiber diameter distribution used, p -value \approx 0.00.

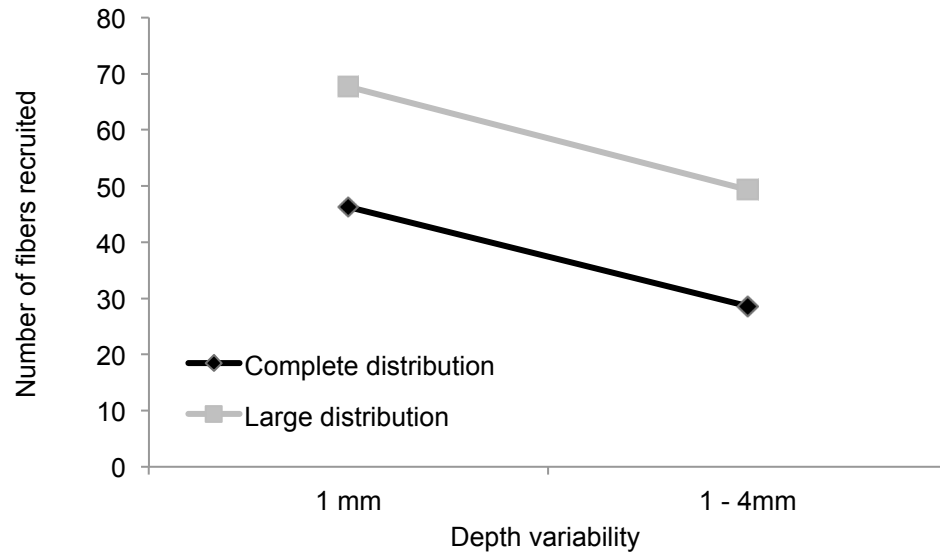


Figure 10. Plot of mean number of fibers activated during simulation for different combinations of depth variability and template fiber diameter distributions.

3.2 Distribution Types

The ANOVA results suggest that the large template fiber diameter distributions yield higher chi-square goodness of fit results than the complete template fiber diameter distributions, taking into account fiber depth variability, and activation functions ($p\text{-value} \approx 0.00$) (Figure 9 and 11). The same trend is also observed for the number of fibers recruited during simulation (Figure 10 and 12).

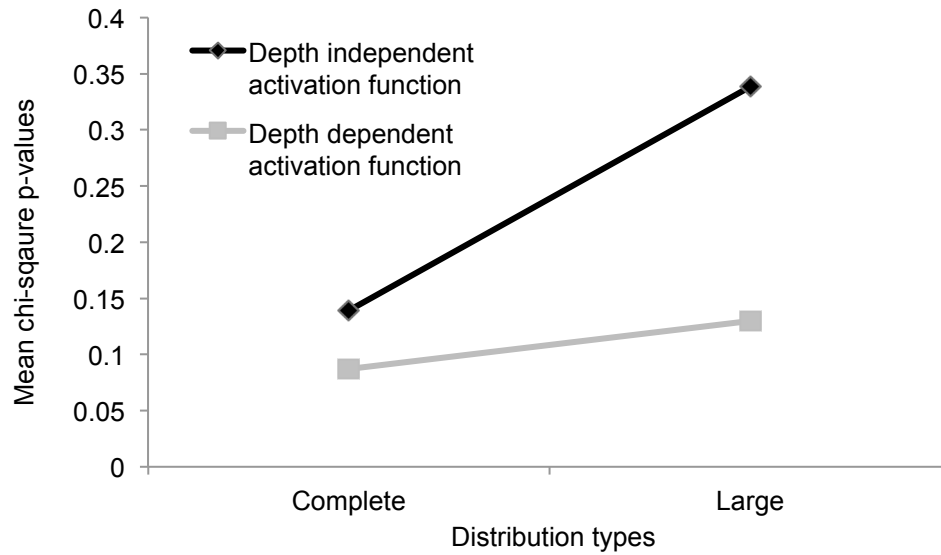


Figure 11. Plot of mean chi-square goodness of fit results for different combinations of template fiber diameter distribution and activation functions. There is a significant difference between the types of template fiber diameter distribution, $p\text{-value} \approx 0.00$, and between the types of activation functions, $p\text{-value} \approx 0.000$. And there is a significant interaction between the types of template fiber diameter distribution and the types of activation functions, $p\text{-value} = 0.007$.

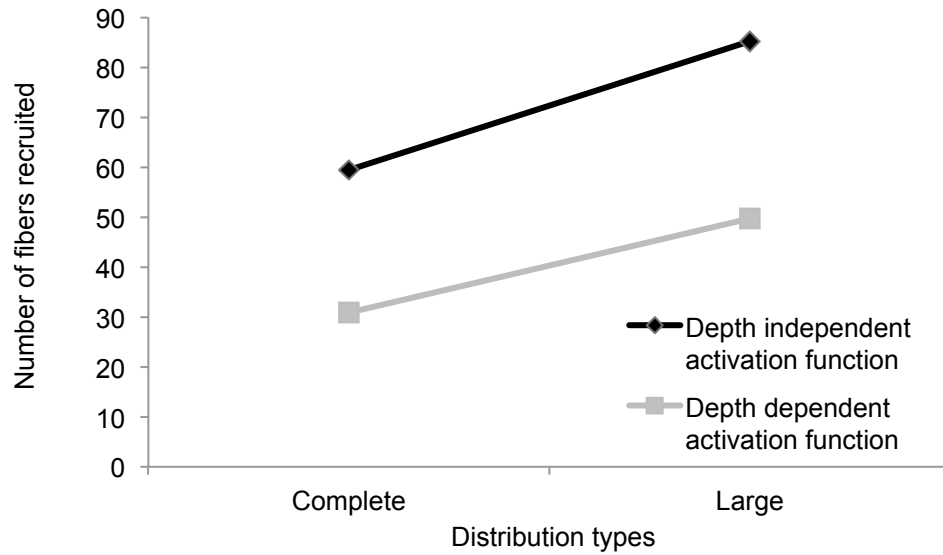


Figure 12. Plot of mean number of fibers activated during simulation for different combinations of template fiber diameter distribution and activation functions.

3.3 Activation Functions

The ANOVA results suggest that the depth independent activation function yields significantly higher chi-square goodness of fit values for fiber diameter estimation than the depth dependent activation function ($p\text{-value} \approx 0.000$) (Figure 11 and 13). Additionally, there is a significant interaction between the types of template fiber diameter distribution and the types of activation ($p\text{-value} = .007$). In other words, the depth independent activation and the large template fiber diameter distribution complement each other to increase the accuracy of the fiber diameter estimation non-additively.

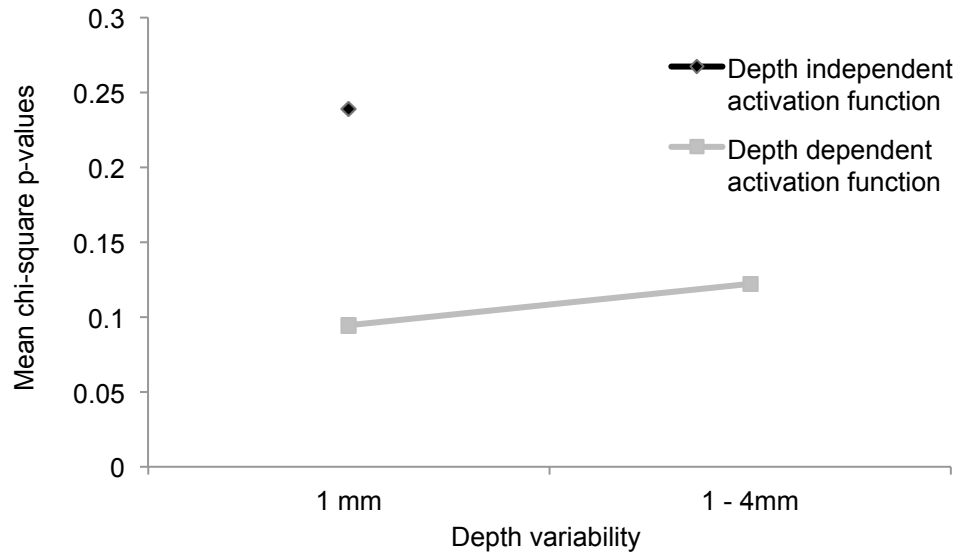


Figure 13. Plot of mean chi-square goodness of fit results for different combinations of fiber depth variability and activation functions. There is no significant difference between the different depth variability, $p\text{-value} = 0.399$. However, there is a significant difference in the types of activations used, $p\text{-value} \approx 0.000$. The combination of variable fiber depth and depth independent activation function was not considered because the fiber depth is assumed to have an effect on the activation of the nerve fibers when it is not constant.

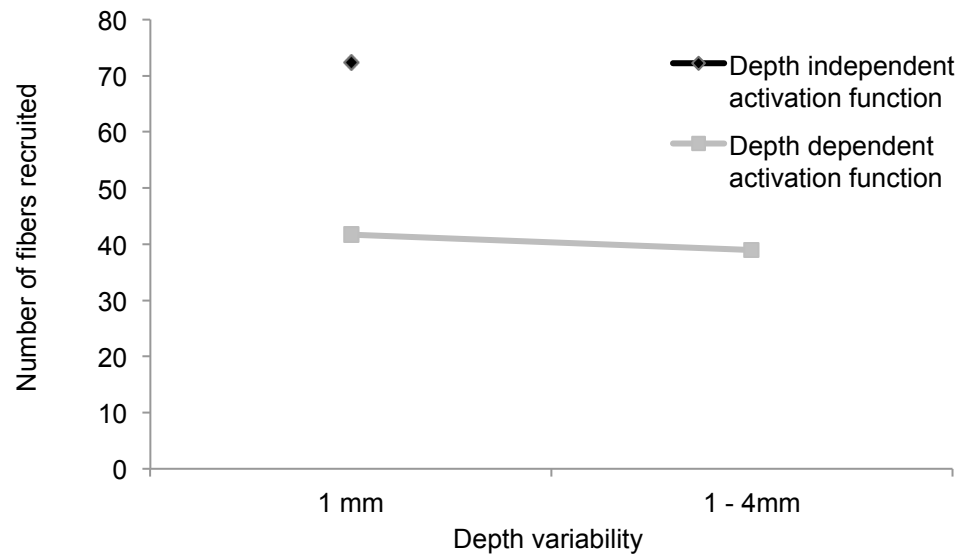


Figure 14. Plot of mean number of fibers recruited during simulation for different combinations of fiber depth variability and activation functions.

4. Discussion

The results from this study suggest that the major factors that influence the accuracy of the fiber diameter estimation in simulation are the types of template fiber diameter distributions and the activation functions used. Although the fiber depth variability, at a glance, seems to have an effect on the accuracy of fiber diameter estimation, it is not statistically significant from the sample data in this study. However, by changing the assumption about the effect that fiber depth has on the activation function of the nerve fibers, a significant difference in the accuracy of fiber diameter estimation is observed between the depth-and-diameter dependent activation function and the depth independent, diameter dependent activation function.

When the fiber depth varies from 1 – 4mm instead of being constant at 1mm, fewer fibers get activated because it takes higher stimulus currents, on average, to activate the same fibers of a particular diameter [21] (Figure 10). Since this study only simulated a small variation in fiber depth, between 1 – 4mm, a larger variation in fiber depth could yield a more significant effect of the fiber depth.

In the previous study, it was observed that the fiber depth is positively correlated to the required stimulus current, but the fiber diameter is negatively correlated to the required stimulus current [21]. Therefore, the depth-and-

diameter dependent activation requires higher stimulus current than the depth independent and diameter dependent activation function in order to activate a fiber of a particular diameter, for the range of fiber depth simulated in this study. Since the maximum stimulus current is set at 1 mA in this study, the depth-and-diameter dependent activation functions recruited fewer nerve fibers on average, and in turn, generated fewer single fiber evoked potential waveforms. Therefore, as a result, the fiber diameter estimation algorithm yielded less accurate fiber diameter estimations on average. In addition, the way that the depth dependent activation is modified in this study could also contribute to the low number of nerve fibers recruited. This behavior may be attributed to the constant that is added to the activation function to make its outputs valid for all inputs. Although the magnitude of the correctional constant is small in relation to the range of the function's outputs, it nonetheless has increased the activation threshold for all the nerve fibers.

There is a weak correlation between the number of fibers recruited during simulation and the chi-square goodness of fit result for the fiber diameter estimation, but the number of recruited fibers and the chi-square results show the same trends for all combinations of different fiber depth variability, diameter distribution types, and activation functions (Figure 9, 10, 11, 12, 13, and 14). These trends suggest that, despite the low correlation, the number of fibers activated during simulation could be an influential driver of the accuracy of the fiber diameter estimation. It is possible to increase the number of fibers recruited

by raising the maximum stimulus current allowed in the simulation, currently set to be 1 mA. The 1 mA current limit used in the simulations is consistent with the maximal output available from clinical EMG equipment.

Another possibility to increase the number of fibers activated is to reduce the incremental step, ΔI , between the stimulation steps. This method works well if each stimulation step recruits more than one fiber. However, in this study, recruitment of more than one fiber at a given stimulus amplitude is rarely observed. The source code to compute how many additional fibers are activated at each stimulus step is included (Appendix E).

When using a large template fiber diameter distribution, the chi-square goodness of fit result increases significantly (Figure 9 and 11). Since changing the template fiber diameter distribution has no direct effect on the estimation algorithm and its protocols, the results suggest that the estimation algorithm can only estimate the diameter for the larger fibers. From the simulation results, the estimation algorithm rarely yields an estimated fiber diameter distribution containing fibers with diameters of 10 μm or less (Figure 15). The reason could be that the maximum stimulus current is not high enough to activate the smaller fibers. If the smaller nerve fibers do not get activated, there will not be a single fiber action potential, which is necessary for the estimation algorithm to estimate the fiber diameter. In addition, there is no visible difference in the accuracy in

estimating the compound evoked potential for a complete template fiber diameter distribution versus a large template fiber diameter distribution (Figure 17 and 18).

Therefore, the significant difference found in the accuracy of the fiber estimation for the different types of template fiber diameter distribution studied is an artifact of the diameters of nerve fibers that get activated. Since mostly larger fibers are activated, the chi-square goodness of fit result will be better for a template fiber diameter distribution that contains mostly large fibers than for a template fiber diameter distribution that contains both small and large fibers (Figure 15 and 16). In addition to the bias towards large template fiber diameter distributions, the simulation returns a much better fiber diameter estimation when combined with the depth independent activation function. This effect is exaggerated when combined with the depth independent activation function because the number of fibers recruited is much higher. The higher number of fibers recruited provides more single fiber evoked potentials for the estimation algorithm.

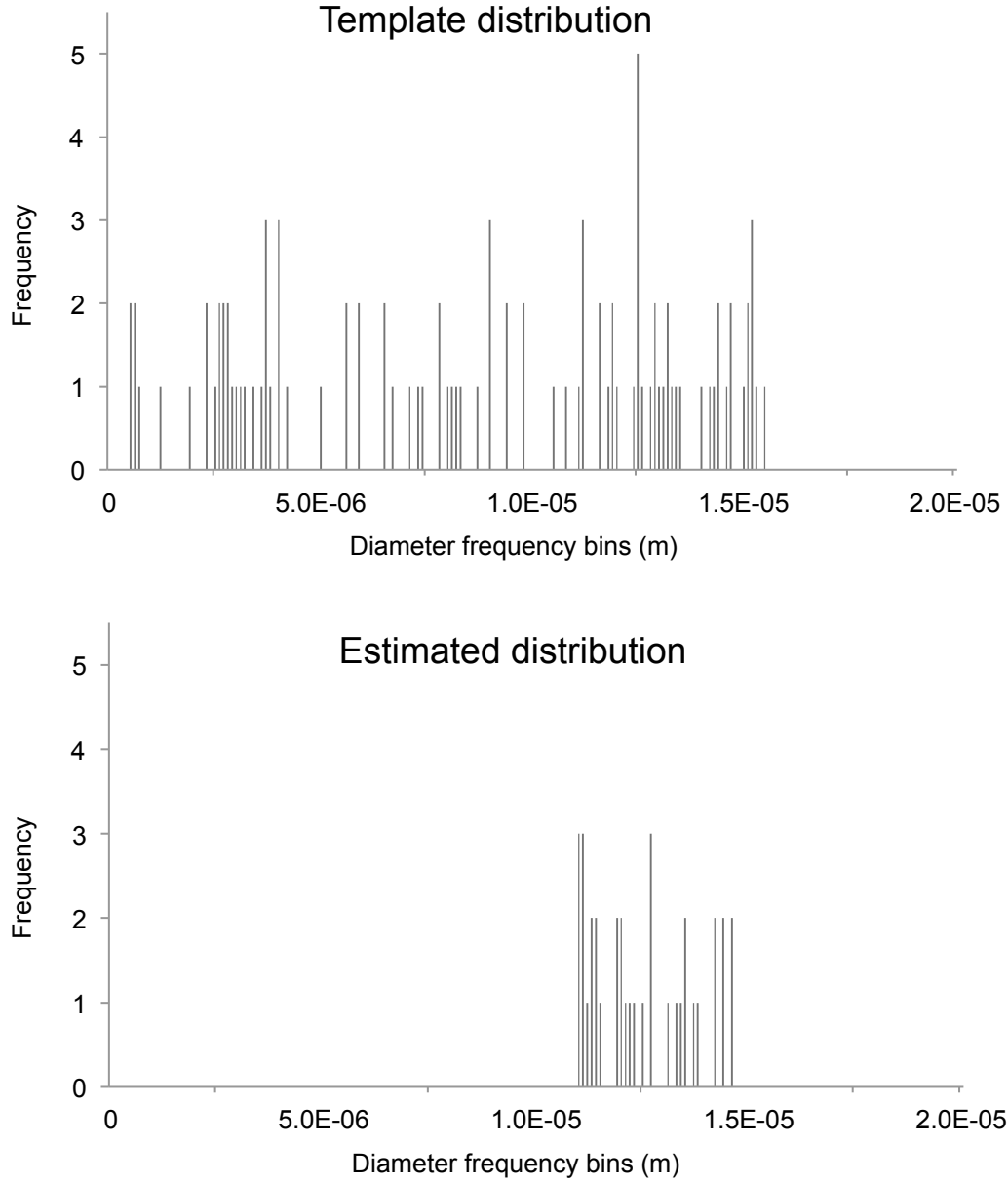


Figure 15. Histograms of a complete template nerve fiber diameter population d and the group delay estimated nerve fiber diameter population \tilde{d} for variable fiber depth, complete template fiber diameter distribution, and depth dependent activation function. The chi-square goodness of fit result for the two distributions is $Q(\chi^2|x) = 0.1086$.

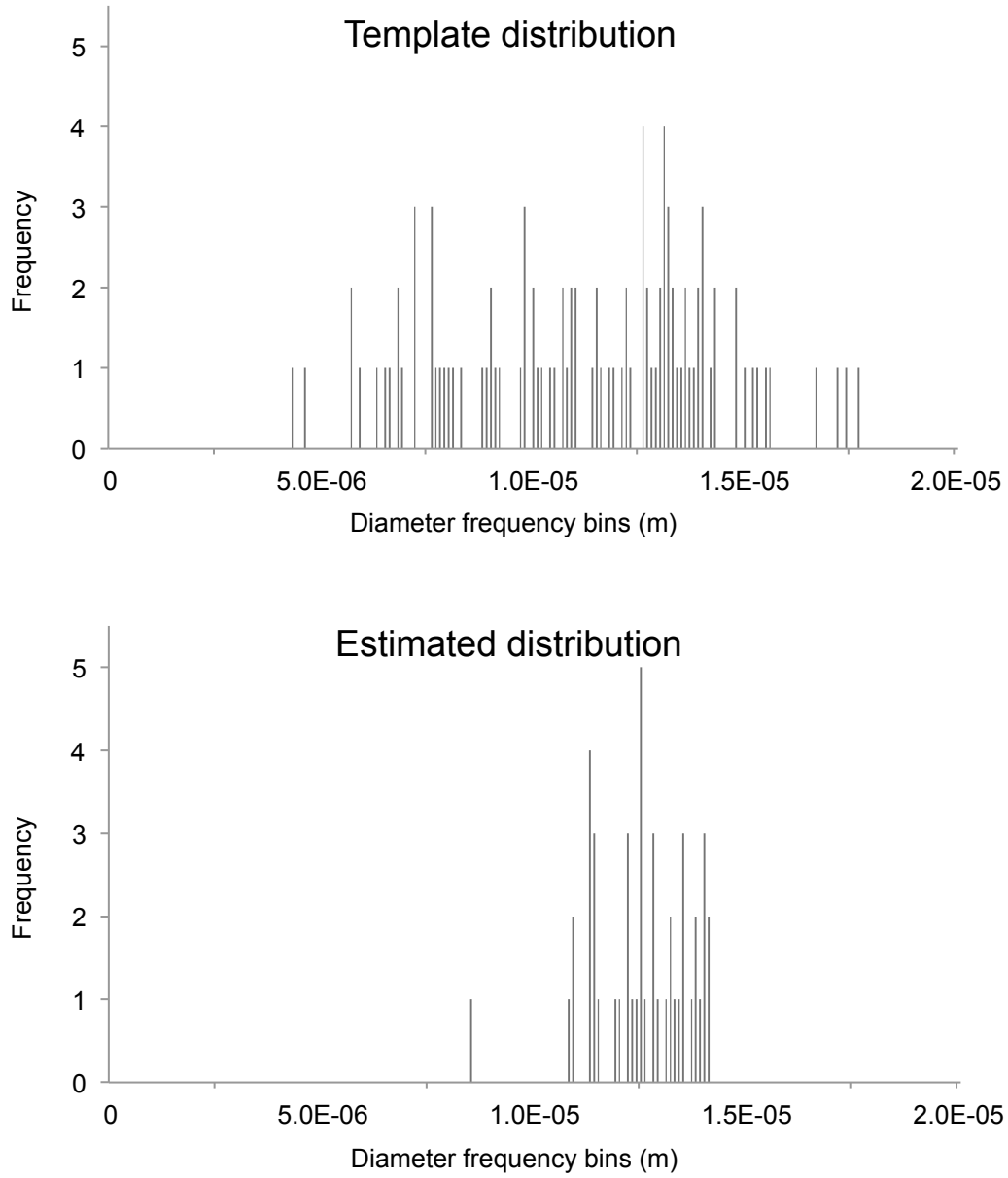


Figure 16. Histograms of the template nerve fiber diameter population d generated from the large fiber distribution and the group delay estimated nerve fiber diameter population \tilde{d} for variable depth, large template fiber diameter distribution, and depth independent activation function. The chi-square goodness of fit result for the two distributions is $Q(\chi^2|x) = 0.1513$.

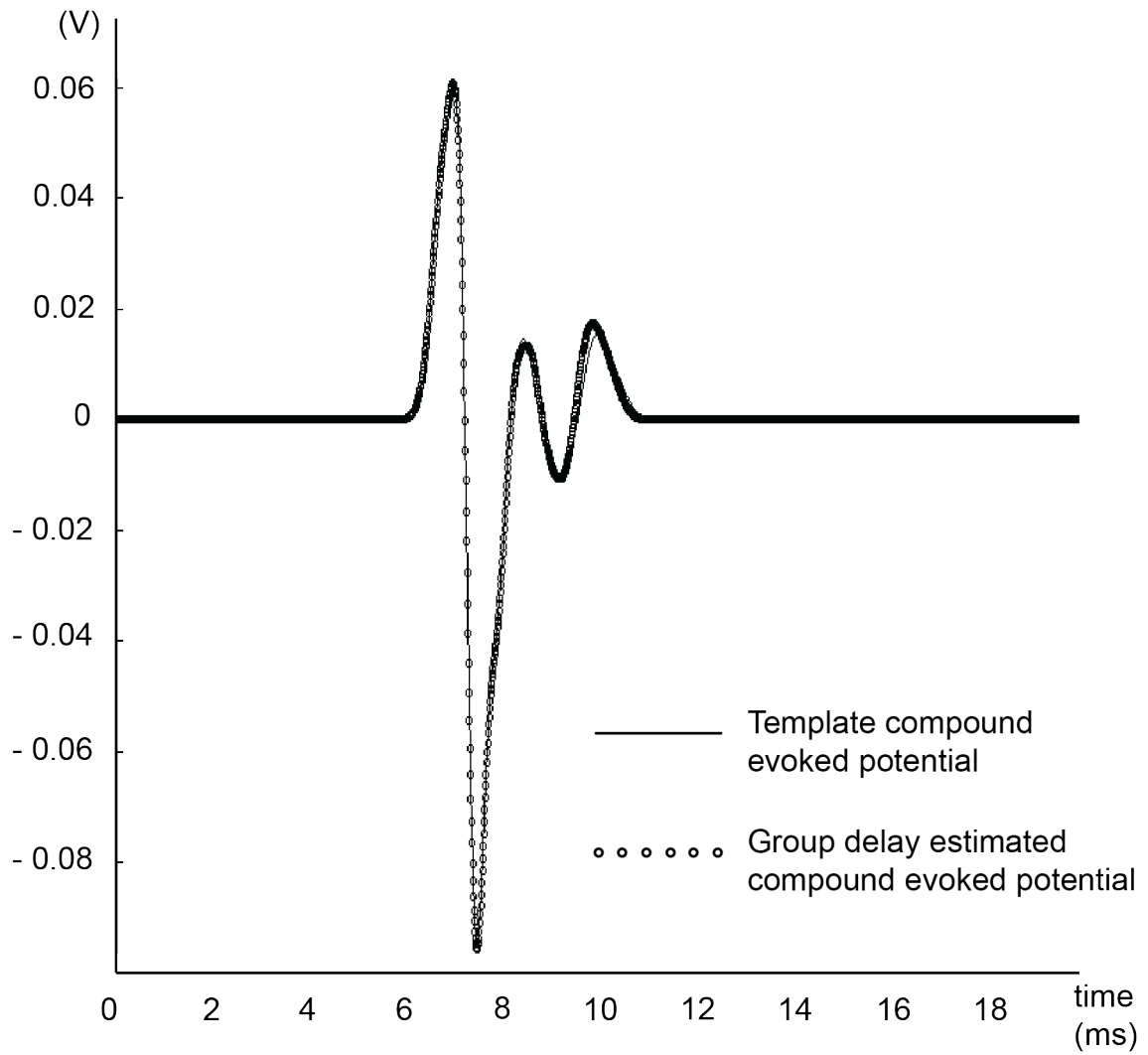


Figure 17. Plot of the template compound evoked potential and the group delay estimated compound evoked potential generated from the simulation of variable fiber depth, complete template fiber diameter distribution, and depth dependent activation function.

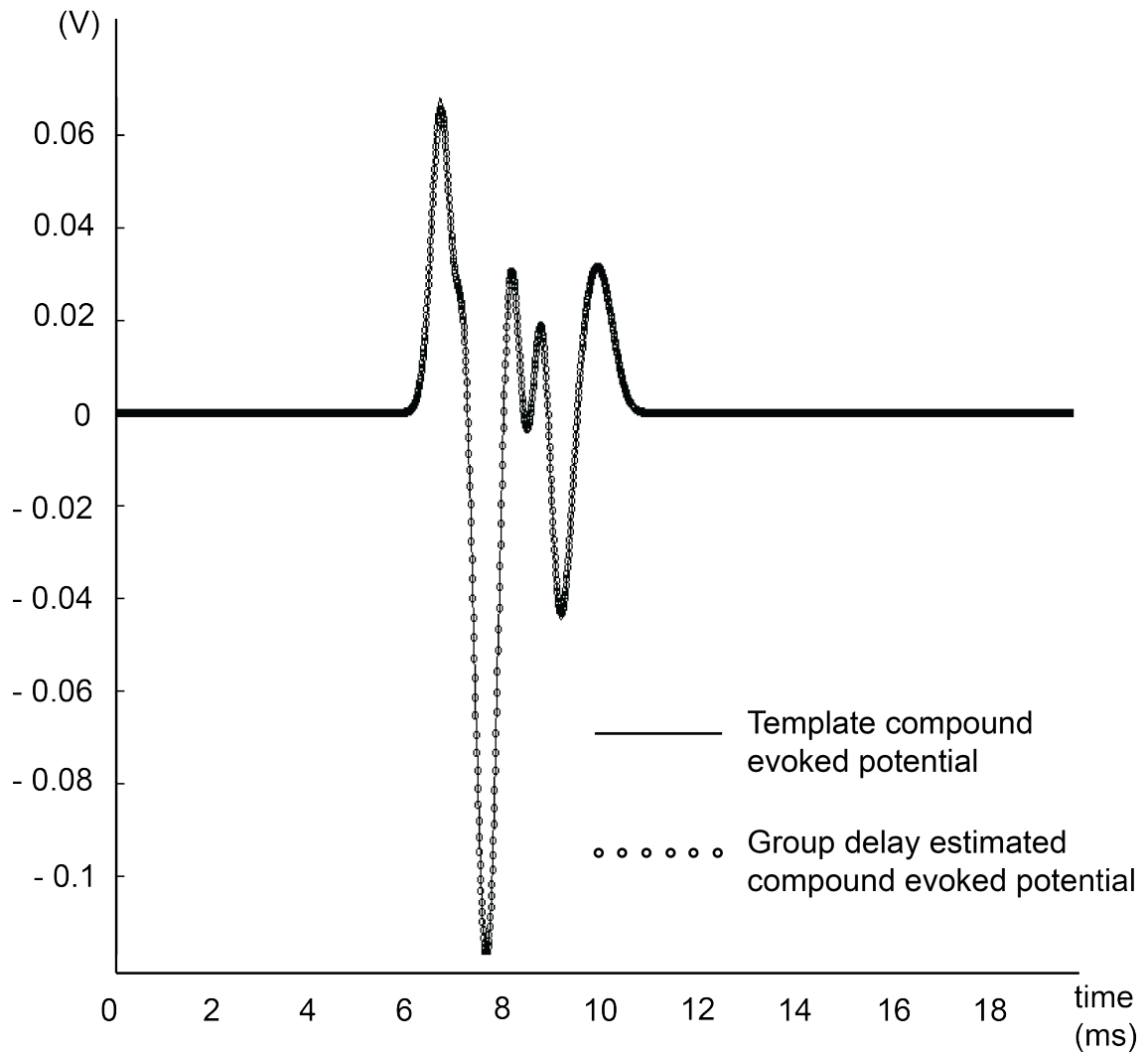


Figure 18. Plot of the template compound evoked potential and the group delay estimated compound evoked potential generated from the simulation of variable fiber depth, large template fiber diameter distribution, and depth dependent activation function.

5. Insights about the Simulation and Future Work

The performance of the simulation depends on many variables because the mathematical models and the code are complex (Appendix D). This study identifies some code issues and proposes countermeasures to relieve them.

5.1 Problem of Negative Group Delay Estimates

There are some cases where the simulation can estimate a positive phase response in the group delay estimation for the single fiber action potential between recording electrodes number one and two, resulting in a negative group delay (Figure 19). In other words, having a positive phase response means that the signal arrives at the second recording electrode before arriving at the first recording electrode. In reality, it is not possible to have a positive phase response under the assumptions of this study, where the first recording electrode is in between the stimulus and second recording electrode (Figure 1). As a consequence, the estimation and optimization of the compound evoke potential failed to produce expected results (Figure 20).

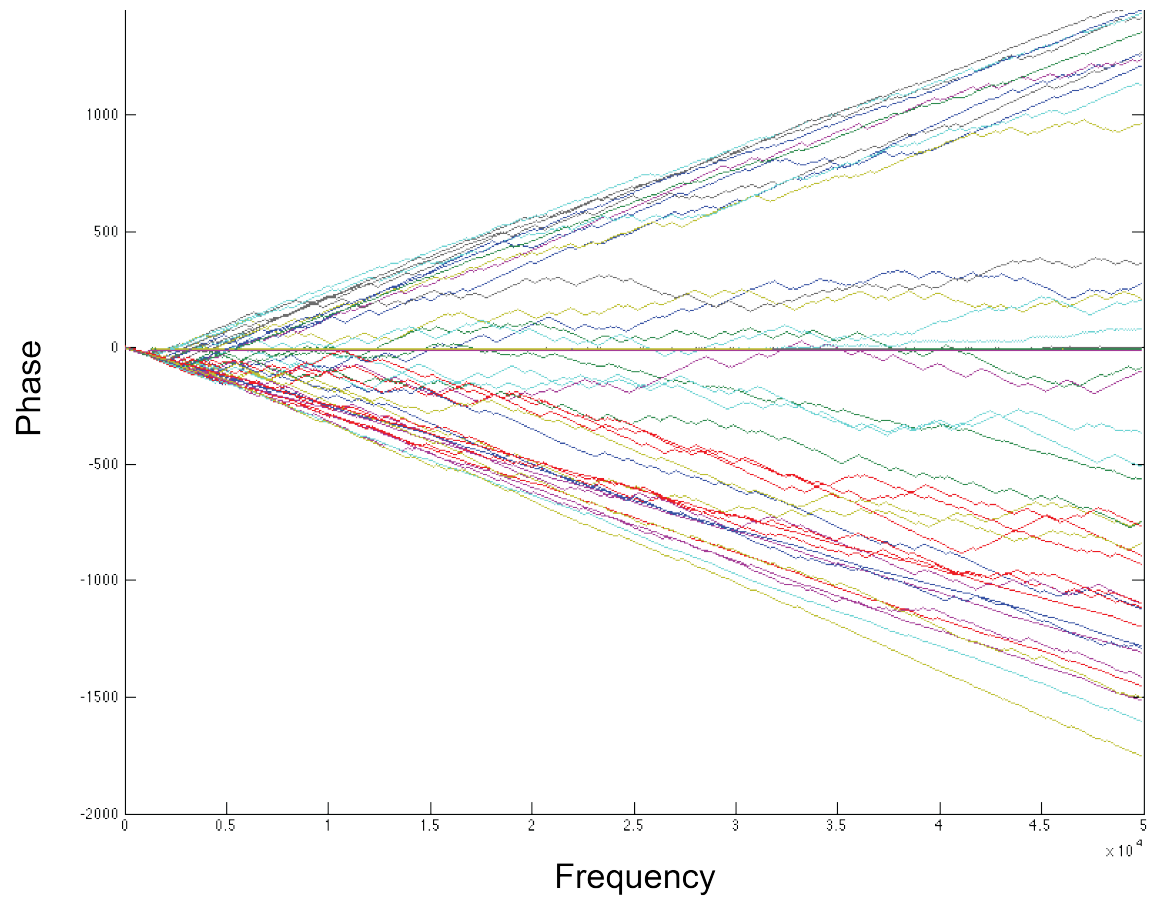


Figure 19. Plot of the phase response as a function of the frequency used in the group delay estimation. Each line represents the phase response in the group delay estimation of a pair of decomposed single fiber action potential waveforms. The span = 15×10^{-3} s.

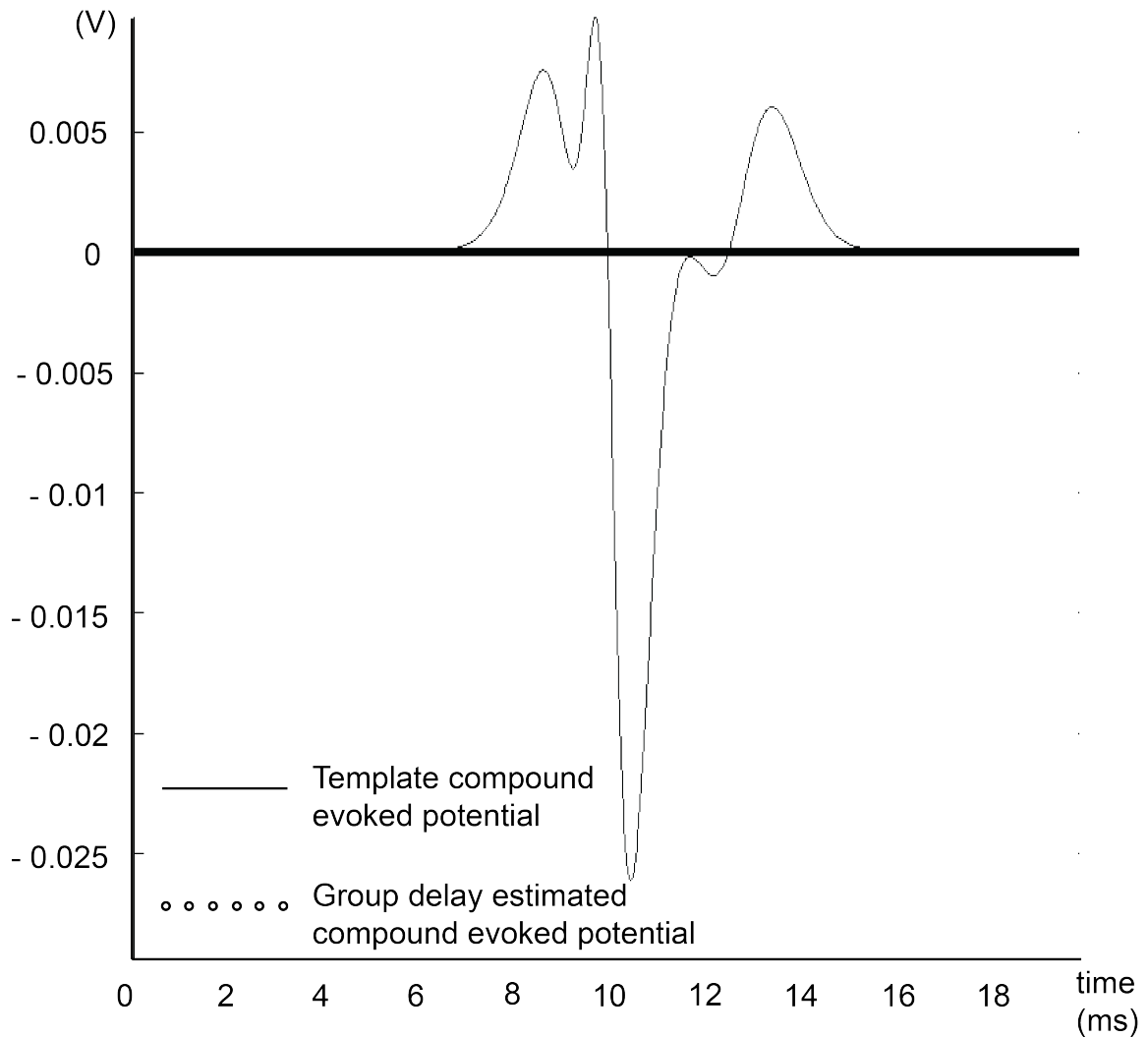


Figure 20. Plot of the template compound evoked potential and the group delay estimated compound evoked potential generated from the simulation with span = 15×10^{-3} s. The estimated compound evoked potential is the flat line at zero.

The underlying problem that caused a negative group delay result is associated with the input parameter that defines the time span of the simulation. The span is the time that the simulation will simulate a single fiber evoked potential waveform after stimulation. For example, if the span = 20×10^{-3} , the

simulation will simulate the fiber action potentials up to 20 milliseconds after the stimulus current is applied. However, if the fiber action potential waveform is longer than the specified span, the program will have an incomplete fiber action potential because the waveform is truncated. The observed truncated waveforms often occur at the second recording electrode site because it takes longer for the signal to travel to the second electrode from the stimulus electrode. The incomplete waveforms when used in the group delay estimation algorithm will produce erroneous results because the group delay estimation protocol essentially compares two unrelated waveforms. The same erroneous result is observed in all combinations of fiber depth variability, types of template fiber diameter distributions, and types of activation function when the span is not sufficiently long.

The solution is to increase the span to accommodate the longer waveforms. Some of the cases where increasing the span might be necessary include large maximum stimulus current amplitudes (> 1 mA), large populations of fibers (>100), shallow fibers (<1 mm deep), smaller fiber diameter (<3 μm), or long distance between the stimulus electrode and the second recording electrode (>5 cm).

5.2 Poor Performance for High Fiber Depth Variability

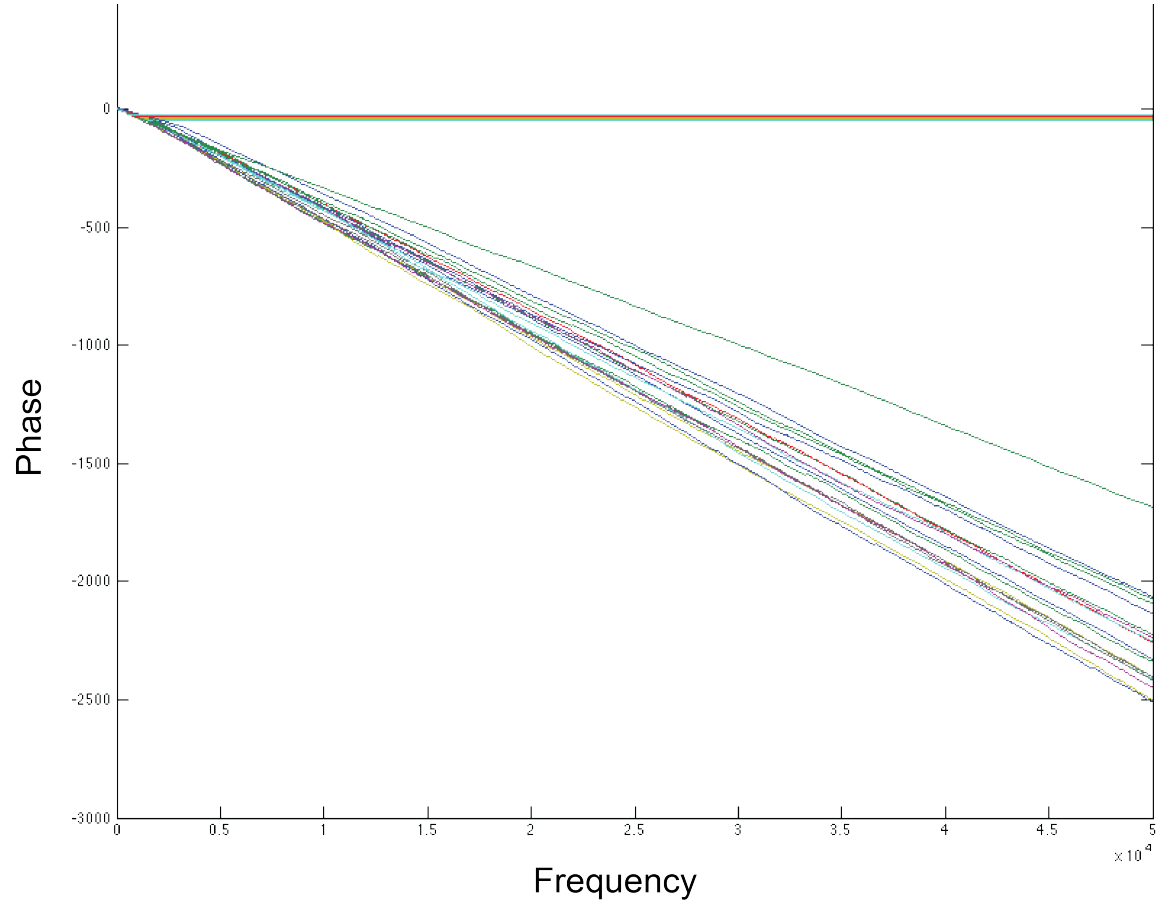


Figure 21. Plot of the phase delay as a function of the frequency used in the group delay estimation. Each line represents the phase delay in the group delay estimation of a pair of decomposed single fiber action potential waveforms. The depth variability is between 1 – 10 mm.

For the fiber depth variability between 1 – 10mm, the fiber diameter estimation has very poor performance. One sample simulation yields chi-square

goodness of fit p -value = 2.0774×10^{-5} . At a glance, the phase response of the group delay estimation yielded results very close to zero (Figure 21). In other words, for some fibers, there is virtually no delay of the signals as it passes through the nerve fibers. The same is observed in all combinations of types of template fiber diameter distribution, and types of activation functions. One possible explanation could be that, for some fibers, the single fiber evoke potentials are very close to zero, so when the decomposed waveforms are used in the group delay estimation, it computes a delay very close to zero. Further investigation is required to determine the underlying causes of this failure. Since the group delay results are not as expected, the estimation of the compound evoked potential also failed (Figure 22). This study restricted the fiber depth variability to 1 – 4 mm to alleviate the issue.

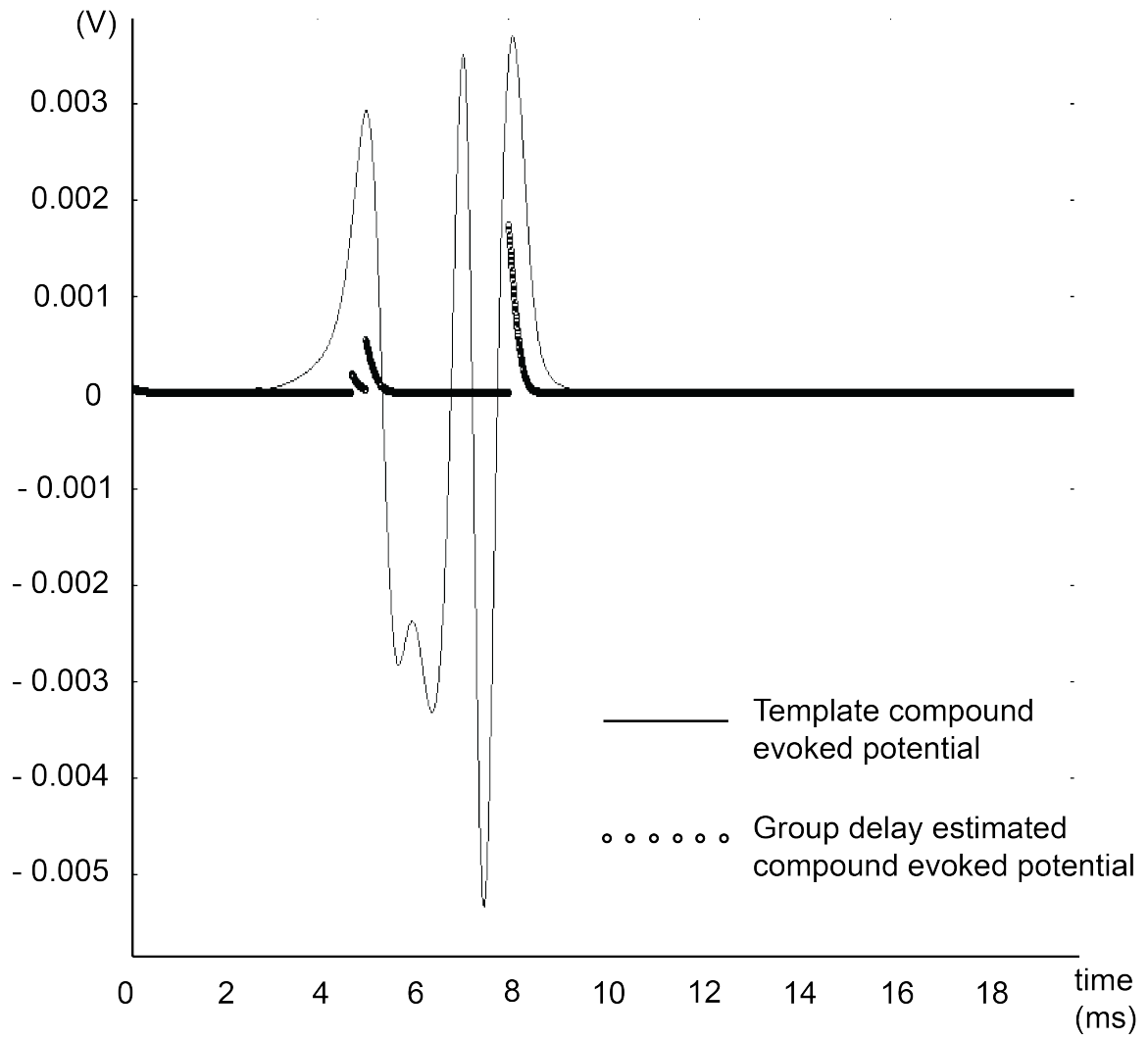


Figure 22. Plot of the template compound evoked potential and the group delay estimated compound evoked potential simulated for fiber depth variability between 1 – 10 mm.

5.3 Future Work

The results of this study show that the mathematical models used to simulate nerve fibers characteristics and behaviors have a tremendous influence on the accuracy of the fiber diameter estimation. Most notably, the types of activation function can significantly impact the accuracy of the nerve fiber diameter estimation. Future work should explore the possibility of creating a nerve fiber activation model based on an animal or human model. This type of activation model would be invaluable in the development of this fiber diameter estimation technique. Together with the current simulation protocol, a lot more insight will be gained about the feasibility of the estimation technique.

In this study, the number of fibers recruited and their single fiber action potential are the principal input data to the fiber diameter estimation algorithm. Therefore, increasing the number of fibers recruited, in theory, will increase the accuracy of the fiber diameter estimation. Future works should explore ways to increase the number of fibers recruited during stimulation. At a glance, increasing the maximum stimulus current can recruit more fibers.

In addition, the fiber depth variability between 1 – 10 mm had produced some possible erroneous results in this study. Further investigation is needed to determine whether the underlying cause is a code issue or an artifact of the mathematical models used in the simulation. It is possible that the available

single fiber evoked potential model is not effective for the range of fiber depth variability [27]. If that is the case, a different fiber evoked potential model that is more realistic will be needed to complete the development of this estimation technique.

6. Conclusion

By incorporating the effect of the nerve fiber depth into the fiber diameter estimation to make the simulation more realistic, this study has provided additional insight about the effect of fiber depth on the estimation technique. The results suggest that the accuracy of the fiber diameter estimation is dependent on the type of template fiber diameter distribution and the type of activation function, but not on the fiber depth variability. Although the effect of the fiber depth variability on the accuracy of fiber diameter estimation is not significant from the sample data, assuming that the fiber activation function is dependent on both the fiber depth and diameter significantly reduces the accuracy of the fiber diameter estimation. The underlying cause can be attributed to the trend that there are a fewer number of fibers recruited causing fewer single fiber evoked potential waveforms to be available for the estimation algorithm, thus limiting its accuracy. Similar to nerve conduction velocity, estimating fiber diameter using the group delay and simulated annealing optimization is limited by the difficulty in stimulating the smaller fibers.

This study has also identified some conditions where the simulation does not produce the expected results and has discussed how to modify the simulation to achieve the desired results. Specifically, the time span of the fiber evoked potential waveform simulation can significantly affect the simulation results by truncating the single fiber evoked potential waveform if it is not long enough. In

turn, a nonsensical estimation of the group delay is produced. Increasing the time span parameter will alleviate the issue. On the other hand, the potentially erroneous results caused by a large range of fiber depth variability requires further investigations. A possible area to investigate is the effectiveness of the mathematical models used in the simulation. Perhaps, using a model that better reflects human anatomy will improve the overall performance of the simulation.

7. Literature Cited

- [1] Torpy, J., Kincaid, J., & Glass, R. (2010). Peripheral neuropathy. *JAMA: The Journal of the American Medical Association*, 303(15), 1556.
- [2] Tavee J and Zhou L. (2009). Small fiber neuropathy: A burning problem. *Cleveland Clinic Journal of Medicine*. 76(5).
- [3] Szlavik, R. (2008). A novel method for characterization of peripheral nerve fiber size distributions by group delay. *IEEE Transactions on Biomedical Engineering*, 55(12), 2836-2840.
- [4] Tu, Y. X., Wernsdorfer, A., Honda, S., and Yomita, Y. (1997). Estimation of conduction velocity distribution by regularized-least-squares method, *IEEE Transactions on Biomedical Engineering*, 44(11), 1102-1106.
- [5] Gu, D., Gander, R. E., and Crichlow, E. C. (1996). Determination of nerve conduction velocity distribution from sampled compound action potential signals. *IEEE Transactions on Biomedical Engineering*, 43(8), 829-838.

- [6] Papadopoulou, F. A., and Panas, S. M. (1999). Estimation of the nerve conduction velocity distribution by peeling sampled compound action potentials. *IEEE Transactions on Magnetism*, 35(3), 1801-1804.
- [7] Gonzalez-Cueto, J. A., and Parker P. A. (2002). Deconvolution estimation of nerve conduction velocity distribution. *IEEE Transactions on Biomedical Engineering*, 49(2).
- [8] Dorfman, L. J. (1984). The distribution of conduction velocities (DCV) in peripheral nerves: a review. *Muscle & Nerve*, 7(1), 2 – 11.
- [9] Köller, H., Kieseier, B., Jander, S., & Hartung, H. (2005). Chronic inflammatory demyelinating polyneuropathy. *The New England Journal of Medicine*, 352(13), 1343-1356.
- [10] Dorfman, L. J., Cummins, K. L., Reaven, G. M., Ceranski, J., Greenfield, M. S., and Doberne, L. (1983). Studies of diabetic polyneuropathy using conduction velocity distribution (DCV) analysis. *Neurology*, 33(6): 773 – 779.

[11] Harati, Y. (1987). Diabetic peripheral neuropathies. *Annals of Internal Medicine*, 107, 546 – 559.

[12] Szlavik, R. (2009). A modification to the group delay and simulated annealing technique for characterization of peripheral nerve fiber size distributions for non-deterministic sampled data. *Conference Proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2009, 5002-5005.

[13] Tesfaye, S., Boulton, A. J. M., Dyck, P. J., Freeman, R., Horowitz, M., Kempler, P. (2010). Diabetic Neuropathies: Update on Definitions, Diagnostic Criteria, Estimation of Severity, and Treatments. *Diabetes Care*, 33(10), 2285-2293

[14] Connolly, A. M. (2001). Chronic inflammatory demyelinating polyneuropathy in childhood. *Pediatric Neurology*, 24, 177-82.

[15] National Institutes of Health. (2009). Diabetic Neuropathies: The Nerve Damage of Diabetes. *NIH Publication*, 09-3185.

[16] McLeod, J. G., Pollard, J. D., Macaskill, P., Mohamed, A., Spring, P., Khurana, V. (1999). Prevalence of chronic inflammatory demyelinating polyneuropathy in New South Wales, Australia. *Annals of Neurology*, 46, 910 - 913.

[17] Lunn, M. P., Manji, H., Choudhary, P. P., Hughes, R. A., Thomas, P. K. (1999). Chronic inflammatory demyelinating polyradiculoneuropathy: a prevalence study in southeast England. *Journal of Neurology, Neurosurgery & Psychiatry*, 66, 677 - 680.

[18] American Academy of Neurology. (1991). Research criteria for diagnosis of chronic inflammatory demyelinating polyneuropathy (CIDP): report from an ad hoc subcommittee of the American Academy of Neurology AIDS Task Force. *Neurology*, 41, 617 - 618.

[19] Cummins, K. L., Dorfman, L. J., and Perkel, D. H. (1979). Nerve fiber conduction-velocity distributions, II Estimation based on two compound action potentials. *Electroencephalography and Clinical Neurophysiology*, 46(6), 674-658.

[20] Oh, S. J. (2009). *Clinical electromyography: nerve conduction studies (3rd ed.)*. Philadelphia: Lippincott Williams & Wilkins.

[21] Soto, N. (2011). Characterizing nerve fiber activation by varying fiber diameter and depth within a conductive medium: a finite element approach (Master's Thesis). *California Polytechnic State University, Digital Commons*.

[22] Boyd, A., and Davey, M. R. (1968). *Composition of Peripheral Nerves*. Edinburgh, U.K.: E & S Livingstone.

[23] Szlavik, R. B., and deBruin, H. (1997). Simulating the distribution of axon size in nerves. Proceedings of the 23rd Canadian Medical and Biological Engineering Conference.

[24] Kirkpatrick S., Gellatt, C. D., Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598).

[25] Parker, P. A., and Kelly, P. (1982). Nerve conduction velocity measurement techniques. *Journal of Clinical Engineering*, 7(2), 153 – 158.

[26] Press, W., Flannery, B., Teukolsky, S., Vetterling, W. (1989). *Numerical Recipes The Art of Scientific Computing (FORTRAN version)*. Cambridge University Press.

[27] Fleisher, S., Studer, M., and Moschytz, G. (1984). Mathematical model of the single-fibre action potential. *Medical & Biological Engineering & Computing*, 22(5), 433-439.

Appendix A – Minitab ANOVA Results for Chi-square Goodness of Fit Test Values for Fiber Diameter Estimation as a Function of Depth Variability, Template Fiber Diameter Distribution Types, and Activation Function Types

General Linear Model: Chi-square t versus depth charac, distribution, ...

Factor	Type	Levels	Values
depth charactersitic	fixed	2	1, 2
distribution type	fixed	2	1, 2
activation function	fixed	2	1, 2

Analysis of Variance for Chi-square test values, using Adjusted SS for Tests

Source	DF	Seq SS	Adj SS	Adj MS	F	P
depth charactersitic	1	0.02666	0.00751	0.00751	0.72	0.399
distribution type	1	0.13554	0.17785	0.17785	17.08	0.000
activation function	1	0.20808	0.20808	0.20808	19.98	0.000
depth charactersitic* distribution type	1	0.00582	0.00594	0.00594	0.57	0.453
distribution type* activation function	1	0.08197	0.08197	0.08197	7.87	0.007
Error	54	0.56239	0.56239	0.01041		
Total	59	1.02047				

S = 0.102052 R-Sq = 44.89% R-Sq(adj) = 39.79%

Unusual Observations for Chi-square test values

Chi-square					
Obs	test values	Fit	SE Fit	Residual	St Resid
36	0.467500	0.155530	0.032272	0.311970	3.22 R
57	0.116500	0.338540	0.032272	-0.222040	-2.29 R

R denotes an observation with a large standardized residual.

Grouping Information Using Tukey Method and 95.0% Confidence

depth			
charactersitic	N	Mean	Grouping
2	20	0.2	A
1	40	0.2	A

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals
Response Variable Chi-square test values

All Pairwise Comparisons among Levels of depth charactersitic
depth charactersitic = 1 subtracted from:

depth			
charactersitic	Lower	Center	Upper
2	-0.03729	0.02741	0.09211

depth				
charactersitic	-+-----+-----+-----+-----			
2	(-----*-----)			
	-+-----+-----+-----+-----			
	-0.035 0.000 0.035 0.070			

Tukey Simultaneous Tests
Response Variable Chi-square test values
All Pairwise Comparisons among Levels of depth charactersitic
depth charactersitic = 1 subtracted from:

depth	Difference	SE of		
charactersitic	of Means	Difference	T-Value	Adjusted P-Value
2	0.02741	0.03227	0.8493	0.3994

Grouping Information Using Tukey Method and 95.0% Confidence

distribution				
type	N	Mean	Grouping	
2	30	0.2	A	
1	30	0.1	B	

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals
Response Variable Chi-square test values
All Pairwise Comparisons among Levels of distribution type
distribution type = 1 subtracted from:

distribution					
type	Lower	Center	Upper	+-----+-----+-----	
+-----					
2	0.06866	0.1334	0.1981	(-----*-----	
-----)					
				+-----+-----+-----	
+-----					
				0.070 0.105 0.140	
0.175					

Tukey Simultaneous Tests
Response Variable Chi-square test values
All Pairwise Comparisons among Levels of distribution type
distribution type = 1 subtracted from:

distribution	Difference	SE of		
type	of Means	Difference	T-Value	Adjusted P-Value

2	0.1334	0.03227	4.132	0.0001
---	--------	---------	-------	--------

Grouping Information Using Tukey Method and 95.0% Confidence

activation function	N	Mean	Grouping
1	20	0.3	A
2	40	0.1	B

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals

Response Variable Chi-square test values

All Pairwise Comparisons among Levels of activation function

activation function = 1 subtracted from:

activation function	Lower	Center	Upper	-----+-----+-----+-----
2	-0.2090	-0.1443	-0.07955	(-----*-----)
				-----+-----+-----+-----
				-----+-----
				-0.180 -0.120 -0.060
0.000				

Tukey Simultaneous Tests

Response Variable Chi-square test values

All Pairwise Comparisons among Levels of activation function

activation function = 1 subtracted from:

activation function	Difference of Means	SE of Difference	T-Value	Adjusted P-Value
2	-0.1443	0.03227	-4.470	0.0001

Appendix B – Minitab ANOVA Results for Number of Fibers Activated during Simulation of Fiber Diameter Estimation as a Function of Depth Variability, Template Fiber Diameter Distribution Types, and Activation Function Types.

General Linear Model: diff_count versus depth charac, distribution, ...

Factor	Type	Levels	Values
depth charactersitic	fixed	2	1, 2
distribution type	fixed	2	1, 2
activation function	fixed	2	1, 2

Analysis of Variance for diff_count, using Adjusted SS for Tests

Source	DF	Seq SS	Adj SS	Adj MS	F	P
depth charactersitic	1	4344.0	75.6	75.6	3.34	0.073
distribution type	1	6720.4	5405.6	5405.6	238.93	0.000
activation function	1	9363.6	9363.6	9363.6	413.88	0.000
depth charactersitic* distribution type	1	1.6	34.2	34.2	1.51	0.224
distribution type* activation function	1	193.6	193.6	193.6	8.56	0.005
Error	54	1221.7	1221.7	22.6		
Total	59	21845.0				

S = 4.75648 R-Sq = 94.41% R-Sq(adj) = 93.89%

Grouping Information Using Tukey Method and 95.0% Confidence

depth			
charactersitic	N	Mean	Grouping
1	40	57.0	A
2	20	54.3	A

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals

Response Variable diff_count

All Pairwise Comparisons among Levels of depth charactersitic

depth charactersitic = 1 subtracted from:

depth	charactersitic	Lower	Center	Upper	
1					-----+-----+-----+-
2		-5.766	-2.750	0.2656	(-----*-----)
					-----+-----+-----+-

-4.0 -2.0 0.0

Tukey Simultaneous Tests

Response Variable diff_count

All Pairwise Comparisons among Levels of depth charactersitic
depth charactersitic = 1 subtracted from:

depth charactersitic	Difference of Means	SE of Difference	T-Value	Adjusted P-Value
2	-2.750	1.504	-1.828	0.0730

Grouping Information Using Tukey Method and 95.0% Confidence

distribution

type	N	Mean	Grouping
2	30	67.3	A
1	30	44.0	B

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals

Response Variable diff_count

All Pairwise Comparisons among Levels of distribution type
distribution type = 1 subtracted from:

distribution type	Lower	Center	Upper	-----+-----+-----+-----
--				
2	20.23	23.25	26.27	(-----*-----)
--				-----+-----+-----+-----
				22.0 24.0 26.0

Tukey Simultaneous Tests

Response Variable diff_count

All Pairwise Comparisons among Levels of distribution type
distribution type = 1 subtracted from:

distribution type	Difference of Means	SE of Difference	T-Value	Adjusted P-Value
2	23.25	1.504	15.46	0.0000

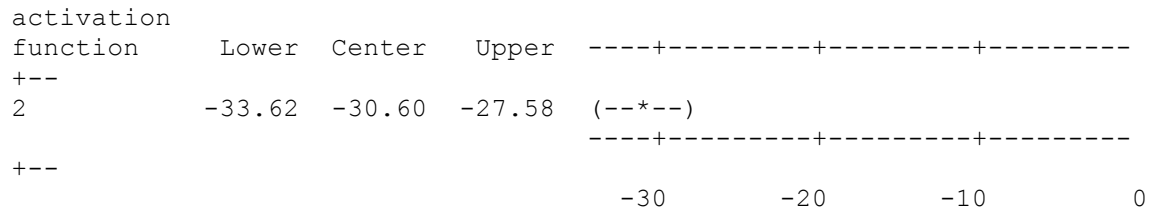
Grouping Information Using Tukey Method and 95.0% Confidence

activation

function	N	Mean	Grouping
1	20	70.9	A
2	40	40.3	B

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals
 Response Variable diff_count
 All Pairwise Comparisons among Levels of activation function
 activation function = 1 subtracted from:



Tukey Simultaneous Tests
 Response Variable diff_count
 All Pairwise Comparisons among Levels of activation function
 activation function = 1 subtracted from:

activation function	Difference of Means	SE of Difference	T-Value	Adjusted P-Value
2	-30.60	1.504	-20.34	0.0000

Appendix C – Data Collected from the Simulations

Numerical codes used for sample data:

Depth variability: 1 = constant at 1mm

2 = variable between 1 – 4mm

Template fiber diameter distribution

types: 1 = complete

2 = large

Activation function types: 1 = diameter-only dependent

activation function

2 = depth and diameter dependent

activation function

Data collected from simulations:

Fiber depth variability	Template distribution types	Activation function types	Chi- square test values
1	1	2	0.0506
1	1	2	0.0668
1	1	2	0.1891
1	1	2	0.0432
1	1	2	0.0201
1	1	2	0.0142
1	1	2	0.0295
1	1	2	0.1805
1	1	2	0.1498
1	1	2	0.1092
1	2	2	0.0434
1	2	2	0.0505
1	2	2	0.2013
1	2	2	0.2293
1	2	2	0.1670
1	2	2	0.0531
1	2	2	0.1378
1	2	2	0.0477
1	2	2	0.0585
1	2	2	0.0489

2	1	2	0.0756
2	1	2	0.1584
2	1	2	0.0085
2	1	2	0.0095
2	1	2	0.0243
2	1	2	0.0828
2	1	2	0.1256
2	1	2	0.181
2	1	2	0.1292
2	1	2	0.0885
2	2	2	0.1169
2	2	2	0.1256
2	2	2	0.0256
2	2	2	0.0561
2	2	2	0.2963
2	2	2	0.4675
2	2	2	0.0476
2	2	2	0.2353
2	2	2	0.0536
2	2	2	0.1308
1	1	1	0.0676
1	1	1	0.2581
1	1	1	0.2168

1	1	1	0.118
1	1	1	0.0676
1	1	1	0.2434
1	1	1	0.0286
1	1	1	0.0502
1	1	1	0.2221
1	1	1	0.1177
1	2	1	0.2412
1	2	1	0.1472
1	2	1	0.4558
1	2	1	0.5266
1	2	1	0.4808
1	2	1	0.384
1	2	1	0.1165
1	2	1	0.4644
1	2	1	0.3475
1	2	1	0.2214

Appendix D – MATLAB Code Used in Simulation

```
warning off
clear all
close all

dist_1 = 20.0E-3;
dist_2 = 50.0E-3;
I_o = 0.0;
I_f = 1.0E-3;
Delta_I = 0.5E-6;
mode = 1;
c = 5.0E5;
step = 10.0E-6;
span = 20.0E-3;

s_scale = 1;
I = 1;
sigma_e = 1;
alpha = 0.998;
psi_m = 10E-3;
psi_d = 3.5E5;

sa_mode = 1;
temp_start = 10;
temp_factor = 0.9;
temp_bound = 1.0E-5;
error_bound = 1.0E-4;
max_step = 1000;
fig_num = 8;
t_step = 10;

%num_modes = 4;
%p_mat = [0.05 0.5E-6 0.1274E-6; 0.25 3.0E-6 0.8493E-6; 0.30 7.5E-6
1.699E-6; 0.4 13E-6 1.699E-6];

num_modes = 2;
p_mat = [0.35 7.5E-6 1.699E-6; 0.65 13.0E-6 1.699E-6];

distrib_low_bound = 0;
distrib_high_bound = 20.0E-6;
distrib_step = 1.0E-7;
pop_size = 100;
bin_spacing = 1.0E-7;

line_limit_factor = 1;

[distrib_vect, distrib_function, cum_function, pop_diam_vect] =
Fiber_Distribution(num_modes, p_mat, distrib_low_bound,
distrib_high_bound, distrib_step, pop_size, bin_spacing);
```

```

figure(1)
plot(distrib_vect(:,1), distrib_vect(:,2));

%used for depth = 1mm only
%r = 1E-3 * ones(pop_size,1);

%used for a range of depth values, uniformly distributed
r = rand(pop_size,1)*3E-3 + 1E-3;

figure(10)
hist(r);

distrib_vect_original = distrib_vect;

save original_distribution.dat distrib_vect -ASCII -DOUBLE

time_count = span/step;

t = zeros(time_count, 1);

for i = 1:time_count;
    t(i) = (i-1)*step;
end

[cap_1, e_pot_1, act_1, fibers_1] = Compound_Action_Potential(dist_1,
I_o, I_f, Delta_I, c, step, span, r, s_scale, I, sigma_e, alpha, psi_m,
psi_d, pop_diam_vect);

figure(2)
plot(t, cap_1)

figure(3)
plot(t, e_pot_1)

[cap_2, e_pot_2, act_2, fibers_2] = Compound_Action_Potential(dist_2,
I_o, I_f, Delta_I, c, step, span, r, s_scale, I, sigma_e, alpha, psi_m,
psi_d, pop_diam_vect);

figure(4)
plot(t, cap_2)

figure(5)
plot(t, e_pot_2)

%figure(6)
%plot(t, e_pot_1)

diff_count = length(e_pot_2(1,:));

tau = zeros(diff_count,1);
H_Phase = zeros(time_count/2,diff_count);
H_P = zeros(time_count/2,1);
L_M = zeros(time_count/2,1);
Lines_Matrix = zeros(time_count/2,diff_count);
f = zeros(time_count/2,1);
delay_vect = zeros(diff_count,1);
fiber_pop = zeros(diff_count,1);

```

```

template = zeros(time_count,1);

for i = 1:time_count/(2*line_limit_factor)
    f(i) = ((i-1)*1/step)/time_count;
end

tic

for i = 1:diff_count
    [tau(i), H_P, L_M, est_diam, est_v]=Group_Delay(e_pot_1(:,i),
e_pot_2(:,i), step, span, line_limit_factor, dist_2-dist_1, c);
    H_Phase(:,i) = H_P;
    Lines_Matrix(:,i) = L_M;
    delay_vect(i)=dist_2/est_v;
    fiber_pop(i) = est_diam;
end

time = toc

figure(7)
plot(f, H_Phase)
hold on
plot(f, Lines_Matrix, 'r')

%Setup the histogram bin vector

count = (distrib_high_bound - distrib_low_bound)/bin_spacing;

distrib_vect = zeros(count,2);

for i = 1:count
    distrib_vect(i) = distrib_low_bound + (i-1)*bin_spacing;
end

distrib_vect(:,2)=hist(fiber_pop, distrib_vect(:,1));

distrib_vect_group_delay = distrib_vect;

save group_delay_estimated_distribution.dat distrib_vect -ASCII -DOUBLE

[est_size, est_comp, res_error, fin_temp] = Annealing(sa_mode,
fiber_pop, delay_vect, dist_2, cap_2(:,2000), e_pot_2, step, span, c,
temp_start, temp_factor, temp_bound, error_bound, max_step, r, s_scale,
I, sigma_e, alpha, fig_num, t_step);

%Setup the histogram bin vector

count = (distrib_high_bound - distrib_low_bound)/bin_spacing;

distrib_vect = zeros(count,2);

for i = 1:count
    distrib_vect(i) = distrib_low_bound + (i-1)*bin_spacing;
end

distrib_vect(:,2)=hist(est_size, distrib_vect(:,1));

distrib_vect_annealed = distrib_vect;

```

```

figure(9)
plot(distrib_vect(:,1), distrib_vect(:,2))

template_evoked_potential(:,1) = t(:,1);
template_evoked_potential(:,2) = cap_2(:,2000);

save template_evoked_potential.dat template_evoked_potential -ASCII -
DOUBLE
save optimized_distribution.dat distrib_vect -ASCII -DOUBLE
save optimized_evoked_potential.dat est_comp -ASCII -DOUBLE

[prob,df]=Chi_Square(distrib_vect_original, distrib_vect_original);
prob
df
[prob,df]=Chi_Square(distrib_vect_original, distrib_vect_group_delay);
prob
df
[prob,df]=Chi_Square(distrib_vect_original, distrib_vect_annealed);
prob
df

beep on
beep

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      Activation_Function_w_depth_and_diameter
%   Revision Date: 04/28/13
%   Author:       Ken Tran
%
%   Arguments:    r              = fiber depth in (m)
%                 pop_diam_vect  = fiber diameter in (m)
%
%   Returns:      act_function    = fiber activation current value
%                                   in (A)
%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function act_function = Activation_Function_w_depth_and_diameter(r,
pop_diam_vect)

depth_vector1= 1000*r;
pop_diam_vect1 = pop_diam_vect*10^6;

act_function = 10^-4 * ((0.2912)*depth_vector1.^2 +
(13.5664)*depth_vector1 - (1.0835)*depth_vector1.*pop_diam_vect1 -

```

```
(56.5306)*pop_diam_vect1 + (2.2463)*pop_diam_vect1.^2 + 305.3168 +
36318908217499/721258995000);
```

```
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      Fiber_Distribution
%   Revision Date: 02/27/09
%   Author:       Robert B. Szlavik
%
%   Arguments:  num_modes      =   Number of Gaussian modes in the
%                                   fiber diameter probability
density
%                                   function specified by n
%                                   p_mat      =   (n,3) matrix of mode parameters
%                                   where the columns are the
%                                   parameters psi, mu and sigma
for
%                                   the normalized Gaussian mode
%                                   defined below
%   Mode(x)      = (psi)/(sigma*sqrt(2*pi))*exp(-(x-mu)^2/(2*sigma^2))
%
%                                   where x is the diameter in (m)
%   distrib_low_bound = lower fiber diameter bound of
the
%                                   distribution in (m)
%   distrib_high_bound = upper fiber diameter bound of
the
%                                   distribution in (m)
%   step          = probability distribution step
size
%                                   in (m)
%   pop_size      = number of fibers in the
simulated
%                                   population.
%   bin_spacing   = bin spacing size in (m)
%
%   Returns:  distrib_vect(:,1) = vector of fiber diameters in
(m)
%            distrib_vect(:,2) = vector of fiber frequencies
for
%                                   plotting histogram
%   pop_diam_vect = vector of actual fiber
diameters in
%                                   (m)
%
%
%   Internal:  count          = number of points in vector
%                                   based
%                                   on bound span and step or
%                                   bin_size
%            distrib_function(:,1) = vector of fiber diameters
in
```

```

%
%          distrib_function(:,2)    =    (m)
%          cum_function(:,1)       =    probability density vector
%                                     =    vector of fiber diameters
in
%
%          cum_function(:,2)       =    (m)
%                                     =    cumulative distribution
vector
%          temp_function           =    vector used in computation
of
%                                     the cumulative distribution
%                                     function;
%          x                       =    vector of fiber diameter in
(m)
%          random_vect            =    uniformly distribution
random
%                                     vector used to generate
fiber
%                                     diameter distribution
%          pop_diam_vector         =    randomly generated fiber
%                                     diameter vector in (m)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [distrib_vect, distrib_function, cum_function, pop_diam_vect]
= Fiber_Distribution(num_modes, p_mat, distrib_low_bound,
distrib_high_bound, step, pop_size, bin_spacing)

```

```

    count = (distrib_high_bound-distrib_low_bound)/step;

    distrib_function = zeros(count,2);
    cum_function = zeros(count,2);
    temp_function = zeros(count,1);
    x = zeros(count,1);
    random_vect = zeros(pop_size,1);
    pop_diam_vect = zeros(pop_size,1);

%   Compute the probability distribution function.

    for i = 1:count
        distrib_function(i,1) = distrib_low_bound + (i-1)*step;
        x(i) = distrib_function(i,1);
        for j = 1:num_modes
            distrib_function(i,2) = distrib_function(i,2) +
((p_mat(j,1))/(p_mat(j,3)*sqrt(2*pi)))*exp(-(x(i)-
p_mat(j,2))^2/(2*p_mat(j,3)^2));
        end
    end

%   Compute the cumulative distribution function.

    for i = 1:count
        cum_function(i,1) = distrib_low_bound + (i-1)*step;
        for j = 1:i
            temp_function(j) = distrib_function(j,2);
        end
        cum_function(i,2) = step*trapz(temp_function);
    end
end

```

```

% Do the inverse mapping to compute the population of fiber diameters

    random_vect = rand(pop_size,1);

    for i = 1:pop_size
        pop_diam_vect(i) = interp1(cum_function(:,2),
cum_function(:,1), random_vect(i), 'spline');
    end

% Setup the histogram bin vector

    count = (distrib_high_bound - distrib_low_bound)/bin_spacing;

    distrib_vect = zeros(count,2);

    for i = 1:count
        distrib_vect(i,1) = distrib_low_bound + (i-1)*bin_spacing;
    end

    distrib_vect(:,2)=hist(pop_diam_vect, distrib_vect(:,1));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Function:      Compound_Action_Potential
% Revision Date: 09/30/13
% Author:       Robert B. Szlavik
% Modified by:  Ken Tran
%
%
% Computes an array of compound evoked potentials.
%
% USES FUNCTIONS:      Exponential_Activation_Function
%                      Fiber_Evoked_Potential
%
% Arguments:  dist      =  propagation distance in (m)
%              I_o      =  initial stimulus current value
in
%
%              I_f      =  (A)
%                      final stimulus current value in
%                      (A)
%              Delta_I  =  stimulus current increment in
(A)
%
%              c        =  Fiber_Evoked_Potential function
%                      velocity diameter constant
%                      (1/s)
%              step     =  time step in (s)
%              span     =  simulation time span in (s)
%              r        =  distance from recording point
to
%
%              s_scale =  fiber centerline (m)
%                      scaling factor for s variable
%                      (dimensionless) (s = s_scale*a)

```

```

%          I          = current through the second pole
(A)
%          sigma_e    = extracellular conductivity
(S/m)
%          alpha       = fraction of I distributed to
first      pole
%          psi_m       = current scaling factor in (A)
%          psi_d       = exponential scaling factor in
(1/m)
%          fiber_pop   = column vector of fiber
diameters in (m) where the dimensions are
%                  fiber_pop[# of fibers, 1]
%
% Returns: cap        = array of compound action
potentials           at each stimulus current level.
%                  cap[length(time),# of stim
%                  steps]
%          e_pot       = decomposed evoked potential
array               of non-zero potential
%                  increments
%                  adding into the compound evoked
%                  potential
%                  e_pot[length(time),# non-zero
%                  potential increments]
%          act         = array of all activated fibers
at                 a stimulus current step.
%                  1s in each column vector
Position of        to the activated fiber in the
%                  population vector
correspond         act[# in fiber_pop, # of stim
%                  steps]
fiber             = array of newly activated fibers
%                  a given stimulus current step.
%                  Position of 1s in each column
%                  vector correspond to the
activated         fiber in the fiber population
%                  vector.
%                  fibers[# in fiber_pop, # of
stim             steps]
%
% Internal: mode      = mode for Fiber_Invoked_Potential
%                  function. Set equal to 1 for
%                  time function.
%                  delta_fiber = fiber time delay vector in (s)
for input

```

```

%                               into Fiber_Invoked_Potential
%                               function.
%                               delta_fiber[# in fiber_pop]
%                               radius          = fiber radius vector in (m)
%                               radius[# in fiber_pop]
%                               stim_loop_count = number of stimulus current
steps
%                               as per
%                               (I_f - I_o)/Delta_I
%                               time_loop_count = number of time steps as per
%                               span/step
%                               fiber_potential = accumulated compound action
%                               potential vector
%                               fiber_potential[:,1] = time
points
%                               in (s).
%                               fiber_potential[:,2] =
accumulated
%                               potential values in (V).
%                               stim_val        = current value of the stimulus
%                               step           = current amplitude in (A).
%                               step           = unit step function value for
%                               determining if the current
fiber
%                               evoked potential is added to
the
%                               accumulating compound evoked
%                               potential.
%                               diff_count      = count variable for the number
of
%                               non zero and thus incremental
%                               fiber evoked potentials.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [cap, e_pot, act, fibers] = Compound_Action_Potential(dist,
I_o, I_f, Delta_I, c, step, span, r, s_scale, I, sigma_e, alpha, psi_m,
psi_d, fiber_pop)

```

```

mode = 1;
delta_fiber = zeros(length(fiber_pop));
radius = zeros(length(fiber_pop));

stim_loop_count = (I_f - I_o)/Delta_I;
time_loop_count = span/step;

fiber_potential = zeros(time_loop_count, 2);
cap = zeros(time_loop_count, stim_loop_count);
act = zeros(length(fiber_pop), stim_loop_count);
fibers = zeros(length(fiber_pop), stim_loop_count);

for i = 1:length(fiber_pop)

```

```

        delta_fiber(i) = dist/(c*fiber_pop(i));
        radius(i) = fiber_pop(i)/2;
    end

    stim_val = I_o;

    for i = 2:stim_loop_count

        for j = 1:length(fiber_pop)

            if stim_val >=
Activation_Function_w_depth_and_diameter(r(j), fiber_pop(j))
                Step = 1.0;
                act(j,i) = 1;
            elseif stim_val <
Activation_Function_w_depth_and_diameter(r(j), fiber_pop(j))
                Step = 0.0;
            end

            fiber_potential = Step*Fiber_Invoked_Potential(mode, c,
step, span, delta_fiber(j), r(j), radius(j), s_scale, I, sigma_e,
alpha);
            cap(:,i) = cap(:,i) + fiber_potential(:,2);

        end

        stim_val = stim_val + Delta_I;
        fiber_potential = zeros(time_loop_count, 2); % new change
    end

    diff_count = 0;

    fibers(:,1) = act(:,1);

    for i = 2:stim_loop_count

        fibers(:,i) = act(:,i)-act(:,i-1);

        if norm(act(:,i)-act(:,i-1)) ~= 0
            diff_count = diff_count + 1;
        end

    end

    e_pot = zeros(time_loop_count,diff_count);
    diff_count = 0;

    for i = 2:stim_loop_count

        if norm(act(:,i)-act(:,i-1)) ~= 0
            diff_count = diff_count + 1;
            e_pot(:,diff_count) = (cap(:,i)-cap(:,i-1));
        end

    end
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      Fiber_Invoked_Potential
%   Revision Date: 02/27/09
%   Author:       Robert B. Szlavik
%
%   Uses Fleisher's model to generate the time or position dependent
%   extracellular waveform of a single fiber invoked potential.
%
%   Arguments:  mode      =   potential as a function of time (mode = 1)
%                           =   potential as a function of distance (mode =
0)
%
%               c          =   velocity diameter constant (1/s)
%               step       =   time or distance step in (s) or (m)
%                           =   respectively
%               span       =   total time in (s) or total length in (m)
%               delta      =   time shift in (s) or space shift in (m)
%               r          =   distance from recording point to fiber
%                           =   centerline (m)
%               a          =   fiber radius (m)
%               s_scale    =   scaling factor for s variable
(dimensionless)
%                           =   (s = s_scale*a)
%               I          =   current through the second pole (A)
%               sigma_e    =   extracellular conductivity (S/m)
%               alpha      =   fraction of I distributed to first pole
%                           =   (dimensionless)
%
%   Returns:    v_vect     v_vect(:,1) = vector of time in (s) or
distance
%                           =   in (m)
%                           v_vect(:,2) = potential (V)
%
%   Internal:   count      =   number of points in vector (count =
span/delta)
%               diameter=   fiber diameter (diameter = 2*a)
%               s         =   distance from center 0 to first two poles
(m)
%               =   (s = s_scale*a) in (m)
%               D         =   parameter defined in Fleisher's paper
%                           =   (D = (a+s)/(r+s))
%               u         =   distance from the center 0 to the third
pole
%                           =   (u = (s)*(1+alpha)/(1-alpha) in (m)
%               z         =   current distance value in (m)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function v_vect = Fiber_Invoked_Potential(mode, c, step, span, delta, r,
a, s_scale, I, sigma_e, alpha)

    count = span/step;

```

```

v_vect = zeros(count,2);

diameter = 2*a;
s = s_scale*a;
D = (a+s)/(r+s);
u = (s)*((1+alpha)/(1-alpha));

if (mode == 1)
    for i = 1:count
        v_vect(i,1) = (i-1)*step;
        z = (v_vect(i,1)-delta)*c*diameter;
        v_vect(i,2) = (I*D^2)/(4*pi*a*sigma_e)*(alpha*exp(-
((D/4)^2)*((z+s)/a)^2))-exp(-((D/4)^2)*((z-s)/a)^2)+(1-alpha)*exp(-
((D/4)^2)*((z-u)/a)^2));
    end
end

if (mode == 0)
    for i = 1:count
        v_vect(i,1) = (i-1)*step;
        z = v_vect(i,1)-delta;
        v_vect(i,2) = (I*D^2)/(4*pi*a*sigma_e)*(alpha*exp(-
((D/4)^2)*((z+s)/a)^2))-exp(-((D/4)^2)*((z-s)/a)^2)+(1-alpha)*exp(-
((D/4)^2)*((z-u)/a)^2));
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      Group_Delay
%   Revision Date: 03/15/09
%   Author:        Robert B. Szlavik
%
%   Computes a group delay estimate of the propagation time between
%   two recording sites as well as a group delay estimate of the fiber
%   diameter and the fiber conduction velocity.
%
%   Arguments:  vwfe_1          =   single fiber potential waveform
at
%               the first recording electrode
in
%               (V) as a function of the time
%               vector with points sampled at
step
%               (s) for a total of span (s)
%               vwse_2          =   single fiber potential waveform
at
%               the second recording electrode
in
%               (V) as a function of the time
%               vector with points sampled at
step
%               (s) for a total of span (s)
%               step            =   time step in (s)
%               span            =   time span in (s)

```

```

%           line_limit_factor    =    constant factor that limits the
%           length of the phase vector used
%           in computing the least squares
%           estimate of the slope by
%           2*line_limit_factor
%           dist                 =    distance between recording
%           electrodes in (m)
%           c                    =    velocity diameter constant
(1/s)
%
%   Returns:   tau              =    group delay estimated
propagation
%           time between the two recording
%           sites (s)
%           H_Phase             =    phase spectrum vector (radians)
%           Lines_Vector        =    least squares estimate of
linear
%           phase response vector (radians)
%           est_diam            =    group delay estimated fiber
%           diameter (m)
%           est_v               =    group delay estimated fiber
%           propagation velocity (m/s)
%
%   Internal:   f               =    frequency vector (Hz)
%           H_of_f              =    fiber frequency response H(f)
%           vector
%           H_of_f_Phase        =    angle of the H(f) vector
(radians)
%           sum_numerator       =    variable used in computing the
%           least squares estimate of the
%           slope of the phase response
%           sum_denominator     =    variable used in computing the
%           least squares estimate of the
%           slope fo the phase response
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [tau, H_Phase, Lines_Vector, est_diam, est_v] =
Group_Delay(vwfe_1, vwse_2, step, span, line_limit_factor, dist, c)

count = span/step;

f = zeros(count,1);
Lines_Vector = zeros(count/2,1);
H_of_f = zeros(count,1);
H_Phase = zeros(count/2,1);

for i = 1:count
    f(i) = ((i-1)*1/step)/count;
end

S_fe = fft(vwfe_1);
S_se = fft(vwse_2);

for i = 1:count
    H_of_f(i) = S_se(i)/S_fe(i);

```

```

end

H_of_f_Phase = unwrap(angle(H_of_f));

for i = 1:count/2
    H_Phase(i,1) = H_of_f_Phase(i);
end;

sum_numerator = 0.0;
sum_denominator = 0.0;
for i = 1:count/(2*line_limit_factor)
    sum_numerator = sum_numerator + f(i)*H_of_f_Phase(i);
    sum_denominator = sum_denominator + f(i)^2;
end
Lines = sum_numerator/sum_denominator;

for i = 1:count/(2*line_limit_factor)
    Lines_Vector(i) = Lines*f(i);
end

tau = (-1/(2*pi))*Lines;
est_diam = dist/(c*tau);
est_v = dist/tau;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      Function_Centroid
%   Revision Date: 03/20/09
%   Author:       Robert B. Szlavik
%
%   Arguments:  v      =   potential and time vector
%                   (:,1) = vector of time points (s)
%                   (:,2) = potential values at time points (V)
%                   t_step =   point step value (speeds up integration)
%
%   Returns:    cent_val      =   centroid value in (s)
%                   centroid   =   computed function centroid
%                   (:,1)=time vector (s)
%                   (:,2)=centroid_top_function
%                   (:,3)=centroid_bottom_function
%
%   Internal:    centroid_top_function   =   array holder for t*f(t)
%                   centroid_bottom_function=   array holder for f(t)
function
%
%                   for centroid
%                   step           =   time step in (s)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [cent_val, centroid] = Function_Centroid(v, t_step)

    centroid = zeros(length(v(:,1))/t_step,3);

```

```

centroid_top_function = zeros(length(v(:,1))/t_step);
centroid_bottom_function = zeros(length(v(:,1))/t_step);

step = v(2,1)-v(1,1);

for i = 1:t_step:length(v(:,1))
    centroid_top_function(i) = abs(v(i,1)*v(i,2));
    centroid_bottom_function(i) = abs(v(i,2));
    centroid(i,1) = v(i,1);
    centroid(i,2) = centroid_top_function(i);
    centroid(i,3) = centroid_bottom_function(i);
end

cent_val =
(step*trapz(centroid_top_function))/(step*trapz(centroid_bottom_function));

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      Annealing
%   Revision Date: 03/25/09
%   Author:       Robert B. Szlavik
%
%   Implements the simulated annealing algorithm to determine the size
%   distribution of a population of fibers for which the summated
compound
%   action potential most close, in an optimized sense, resembles a
%   maximal evoked potential template.
%
%   USES FUNCTIONS:      Fiber_Invoked_Potential
%
%   Arguments:  vwfe_1      =   single fiber potential waveform
at
%                                   the first recording electrode
%
%                                   (V) as a function of the time
in
%                                   vector with points sampled at
%
%                                   (s) for a total of span (s)
step
%
%   Returns:      tau      =   group delay estimated
propagation
%                                   time between the two recording
%                                   sites (s)
%
%   Internal:      f      =   frequency vector (Hz)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [est_size, est_comp, res_error, fin_temp] = Annealing(sa_mode,
fiber_pop, delay_vect, dist, template, e_pot, step, span, c,
temp_start, temp_factor, temp_bound, error_bound, max_step, r, s_scale,
I, sigma_e, alpha, fig_num, t_step)

```

```

mode = 1;

temperature = temp_start;

count = span/step;

t = zeros(count,1);

for i = 1:count
    t(i) = (i-1)*step;
end

if (sa_mode == 1)

    for i = 1:length(fiber_pop)

        v_centroid = zeros(count,2);

        v_centroid(:,1) = t;
        v_centroid(:,2) = e_pot(:,i);

        [cent_val, centroid] = Function_Centroid(v_centroid, t_step);

        fiber_delay = delay_vect(i);

        if (fiber_delay > cent_val)

            delay_count = floor((fiber_delay - cent_val)/step);

            e_current = e_pot(:,i);
            e_pot(:,i) = zeros(count,1);

            for j = 1:(count-delay_count)
                e_pot(j+delay_count,i) = e_current(j);
            end

        end

        if (fiber_delay <= cent_val)

            delay_count = floor((cent_val - fiber_delay)/step);

            e_current = e_pot(:,i);
            e_pot(:,i) = zeros(count,1);

            for j = 1:(count-delay_count)
                e_pot(j,i) = e_current(j+delay_count);
            end

        end

    end

end

selection_vector = zeros(length(fiber_pop));

compound = zeros(count,1);

```

```

min_fib_diam = min(fiber_pop);
max_fib_diam = max(fiber_pop);

error_value = error_bound + 1;
res_error = error_value;
temp = temp_start;

while ((error_value > error_bound) && (temp > temp_bound))

    k = 0;
    oracle_count = 0;
    change_count = 0;
    total_change_count = 0;
    rejected_count = 0;

    while ((k < max_step) && (error_value > error_bound))

        k = k + 1;
        selection_vector = randperm(length(fiber_pop));
        chosen_fiber = selection_vector(1);

        new_fiber_diam = min_fib_diam + rand(1)*(max_fib_diam -
min_fib_diam);
        new_fiber_delay = dist/(c*new_fiber_diam);

        original_fiber_diam = fiber_pop(chosen_fiber);
        original_fiber_delay = delay_vect(chosen_fiber);

        fiber_pop(chosen_fiber) = new_fiber_diam;
        delay_vect(chosen_fiber) = new_fiber_delay;

        if (sa_mode == 0)

            v_vect = Fiber_Evoked_Potential(mode, c, step, span,
new_fiber_delay, r, new_fiber_diam/2, s_scale, I, sigma_e, alpha);
            e_pot(:,chosen_fiber) = v_vect(:,2);
        end

        if (sa_mode == 1)

            v_centroid = zeros(count,2);

            v_centroid(:,1) = t;
            v_centroid(:,2) = e_pot(:,chosen_fiber);

            [cent_val, centroid] = Function_Centroid(v_centroid,
t_step);

            if (new_fiber_delay > cent_val)

                delay_count = floor((new_fiber_delay - cent_val)/step);

                e_current = e_pot(:,chosen_fiber);
                e_pot(:,chosen_fiber) = zeros(count,1);

                for i = 1:(count-delay_count)
                    e_pot(i+delay_count,chosen_fiber) = e_current(i);

```

```

        end

    end

    if (new_fiber_delay <= cent_val)

        delay_count = floor((cent_val - new_fiber_delay)/step);

        e_current = e_pot(:,chosen_fiber);
        e_pot(:,chosen_fiber) = zeros(count,1);

        for i = 1:(count-delay_count)
            e_pot(i,chosen_fiber) = e_current(i+delay_count);
        end
    end

    end

    compound = zeros(count,1);

    for j = 1:length(fiber_pop)
        for i = 1:count
            compound(i) = compound(i) + e_pot(i,j);
        end
    end

    error_value = 0;

    for i = 1:count
        error_value = error_value + (template(i) - compound(i))^2;
    end

    error_value = sqrt(error_value);

    change = res_error - error_value;

    oracle_value = rand(1);

    if((oracle_value <= exp(-abs(change)/temp)) || (change >= 0))

        if (oracle_value <= exp(-abs(change)/temperature))
            oracle_count = oracle_count + 1;
        end

        if (change >= 0)
            change_count = change_count + 1;
        end

        total_change_count = total_change_count + 1;

        res_error = error_value;
    else

        fiber_pop(chosen_fiber) = original_fiber_diam;
        delay_vect(chosen_fiber) = original_fiber_delay;
    end
end

```

```

        if (sa_mode == 0)

            v_vect = Fiber_Invoked_Potential(mode, c, step, span,
original_fiber_delay, r, original_fiber_diam/2, s_scale, I, sigma_e,
alpha);
            e_pot(:,chosen_fiber) = v_vect(:,2);

        end

        if (sa_mode == 1)

            e_pot(:,chosen_fiber) = e_current;

        end

        rejected_count = rejected_count + 1;

    end

end

temp = temp*temp_factor;

compound = zeros(count,1);

for j = 1:length(fiber_pop)
    for i = 1:count
        compound(i) = compound(i) + e_pot(i,j);
    end
end

fprintf('Temp:  %3.2g, ', temp)
fprintf('\t Error:  %3.2g,', error_value)
fprintf('\t Tot Ch:  %d, ', total_change_count)
fprintf('\t Con Ch %d \n ', change_count)

figure(fig_num)
plot(t, template, 'k') %'g')
hold on
plot(t, compound, 'o k', 'MarkerSize',5) %'r')
hold off

end

est_size = fiber_pop;
est_comp = compound;
fin_temp = temperature;

% Chi Square Calculation

function [prob, df] = Chi_Square(first_distrib, second_distrib)

chi_square = 0;
df=length(first_distrib(:,1))-1;

```

```

sum1 = 0;
sum2 = 0;

for i = 1:length(first_distrib(:,1))
    sum1 = sum1+first_distrib(i,2);
    sum2 = sum2+second_distrib(i,2);
end

for i = 1:length(first_distrib(:,1))
    if ((first_distrib(i,2) ~= 0) || (second_distrib(i,2)~=0))
        chi_square = chi_square + ((first_distrib(i,2)-
second_distrib(i,2))^2)/(first_distrib(i,2)+second_distrib(i,2));
    end
    if ((first_distrib(i,2) == 0) && (second_distrib(i,2)==0))
        df=df-1;
    end
end

prob=1-gammainc(0.5*chi_square,0.5*df)';

```

Appendix E – MATLAB Code Used for Debugging and Diagnostic

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      diam_of_act_fibers
%   Revision Date: 04/28/13
%   Author:       Ken Tran
%
%
%   Description: diameters of the activated fibers
%   Run as it
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Which stimulus steps recruit an additional fiber
stim_loop_count = (I_f - I_o)/Delta_I;
stim_step = zeros(diff_count, 1);
j=1;

for i = 2:stim_loop_count

    if norm(act_1(:,i)-act_1(:,i-1)) ~= 0
        stim_step(j) = i;
        j = j+1;
    end

end

%stim_step

%looking for the latest activated fibers and their diameters

id_fiber_in_pop = zeros(diff_count,1);
i = 0;

for i = 1:diff_count
    id_fiber_in_pop(i,1) = find((act_1(:,stim_step(i)) -
act_1(:,(stim_step(i)-1))) == 1, 1, 'last' );

end

%id_fiber_in_pop
pop_diam_vect(id_fiber_in_pop)
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      group_delay_test
%   Revision Date: 04/28/13
%   Author:       Ken Tran
%
%
%   Description:  identify the fiber and its phase delay slope
%   Run as it
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Double checking the group delay plot

count = span/step;

f = zeros(count/2,1);
Lines_Vector = zeros(count/2,1);
H_of_f = zeros(count,1);
H_Phase_test = zeros(count/2,1);

%for i = 1:count
%   f(i) = ((i-1)*1/step)/count;
%end

for i = 1:count/(2*line_limit_factor)
    f(i) = ((i-1)*1/step)/count;
end

slope = zeros(diff_count,1);

for i = 1:diff_count
    fiber_number = i;
    bin1 = fft(e_pot_1(:,fiber_number));
    bin2 = fft(e_pot_2(:,fiber_number));

    for j = 1:count
        H_of_f(j) = bin2(j)/bin1(j);
    end

    H_of_f_Phase_test = unwrap(angle(H_of_f));

    for j = 1:count/2
        H_Phase_test(j,1) = H_of_f_Phase_test(j);
    end;

    figure(12)
    plot(f, H_Phase_test)
    hold on

    slope(i) = mean(diff(H_Phase_test)./diff(f));

```

```

end

[m,k] = max(slope)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   Function:      how_many_each_step
%   Revision Date: 04/28/13
%   Author:       Ken Tran
%
%
%   Description:   how many additional fibers recruited in a
particular
%                  step, since the last step
%   Run as it
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Which stimulus steps recruit an additional fiber
stim_loop_count = (I_f - I_o)/Delta_I;
stim_step = zeros(diff_count, 1);
j=1;

for i = 2:stim_loop_count

    if norm(act_1(:,i)-act_1(:,i-1)) ~= 0
        stim_step(j) = i;
        j = j+1;
    end

end

%how many fibers got recruited in each successful step
how_many = zeros(diff_count,1);
for i = 1:diff_count
    how_many(i,1) = sum((act_1(:,stim_step(i)) - act_1(:,(stim_step(i)-
1))));
end

how_many

```