5-1-2024

Extracting Tissue Optical Porperties for Photon Distribution Time-of-Flight (PDTOF)

Lucas Allen Spears
University of Alabama in Huntsville

Extracting Tissue Optical Properties for Photon Distribution Time-Of-Flight (PDTOF)

by

Lucas Allen Spears

An Honors Capstone
submitted in partial fulfillment of the requirements
for the Honors Diploma
to
The Honors College
of
The University of Alabama in Huntsville
05/01/2024
Honors Capstone Project Director: Dr. Vinh Du Le

Lucas Spears April 29, 2024

Student (signature) **Date**

Vinh Nguyen Du Le 04/26/2024

Project Director (signature) **Date**

Department Chair (signature) **Date**

Digitally signed by Sean M Lane
Date: 2024.04.30 11:31:48 -05'00'

Honors College Dean (signature) Date

Honors College

Frank Franz Hall

+1 (256) 824-6450 (voice)

+1 (256) 824-7339 (fax)

honors@uah.edu

Honors Thesis Copyright Permission

This form must be signed by the student and submitted with the final manuscript.

In presenting this thesis in partial fulfillment of the requirements for Honors Diploma or Certificate from The University of Alabama in Huntsville, I agree that the Library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by my advisor or, in his/her absence, by the Chair of the Department, Director of the Program, or the Dean of the Honors College. It is also understood that due recognition shall be given to me and to The University of Alabama in Huntsville in any scholarly use which may be made of any material in this thesis.

___ Lucas Spears _____
Student Name (printed)

Student Signature

April 29, 2024

Date

Table of Contents

Abstract:

Mapping cerebral blood flow requires an understanding of optical properties of the brain tissue. While standard continuous-wave near-infrared spectroscopy techniques (CW-NIRS) can measure these properties, they cannot quantify absolute values for each tissue layer. We will use time-resolved near-infrared spectroscopy to extract these values in layered-tissue phantoms through rigorous data analysis of the photon distribution time-of-flight (PDTOF). The method of spectroscopy used for producing the PDTOF is called time correlated single photon counting (TCSPC). After data acquisition, diffusion theories will be used to extract the tissue response function from the measured response function which is contaminated with the laser response function. These response functions, once deconvolved, allow us to further analyze the tissue response function alone. Through fitting the tissue response function to equations for reflectance, we determine the optical properties of the tissue sampled. Though our experimentation was inconclusive in regards to the optical properties of the tissue, we have identified possible origins of error and layout future work required to improve these current techniques.

Background:

Time Correlated Single Photon Counting (TCSPC):

TCSPC is traditionally used in fluorescence measurements, where it measures the time elapsed between a laser's excitation pulse and the fluorescence photon emitted. However, in our case we are not interested in measuring fluorescence. We are interested in measuring the scattering of photons back towards the detector. Therefore the photon reaching the detector in our case will not be a fluorescence photon emitted from this laser excitation, it will instead be a photon from our incident laser which has been scattered back from the sample into our detector. The time measured is the difference between the time of a photon's arrival to the detector and the time of the laser pulse emission. Once this measurement is taken for some number of laser pulses, the photons signaling the detector will be added up and plotted over their time-of-flight to form the data for the sample which they were scattered from (Figure 1) [1].

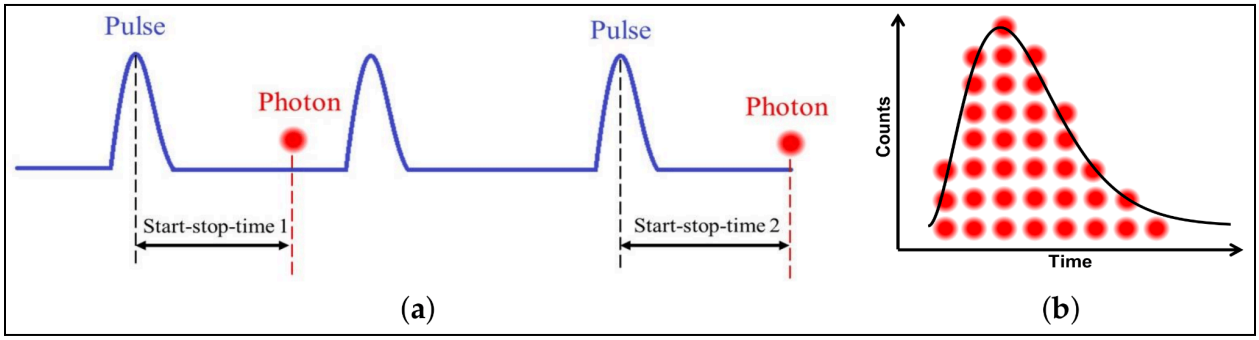


Figure 1: (a) Graphical representation of the difference between the laser pulse and the photon incident on the detector, the time difference between these two points is noted as that photon's

arrival time. (b) Once this measurement is completed for multiple laser pulsations over time, the arrival time is collected for a number of photons and photons with the same arrival time will increase the intensity on the vertical axis, this is represented graphically as the time-of-flight data.

Response Functions:

When we measure something with a detector, there is some implicit and expected distortion involved in the results. This is a consequence of the instrument response function, in our case this is due explicitly to the incident laser's instrument response function. The expected signal we will receive from such a measurement will be a convolution integral; in other words the measured signal is expressed as an integral of the product of the laser instrument response function and the actual undistorted signal. In order to obtain the valuable data we are after, namely the true undistorted signal, we will design a method for fitting a convolution integral to the measured signal function in order to determine the parameters of our actual undistorted signal. In order to do this, we will need data of the observed signal as well as data of the laser's instrument response function to restrain our fitting to only one unknown function.

Optical Properties:

The reduced scattering coefficient (μ_s') and the absorption coefficient (μ_a) are defined as optical properties. The reduced scattering coefficient is dependent upon the scattering coefficient, μ_s, and the anisotropy factor, g, (1).

$$\mu_s' = \mu_s(1 - g) \qquad (1)$$

The anisotropy factor is determined by scattering anisotropy and depends upon the cosine of the angle of deflection of the scattered photon. The scattering and absorption coefficients are considered constants dependent upon the sample. These coefficients are considered optical properties and are physically meaningful to us. The scattering coefficient is a measure of the sample's ability to scatter photons, while the absorption coefficient is a measure of the sample's ability to absorb photons of an incident beam of light; they each have units of inverse length. Through diffusion theory equations, we can fit the measured reflectance with theoretical reflectance dependent on these coefficients, and thus determine these coefficients for whatever sample the beam is incident on.

Methods:

Experimental Setup:

In order to measure the scattering response function of our sample, which is a mixture of intralipid and water to represent human tissue, we have designed a setup in which we can measure the scattering at different path lengths (1cm, 2cm, and 3cm). This path length is the distance between the emitter and detector; with a 3D printed piece of equipment (Figure 2) we have designed a way to change this path length as an independent variable with ease.

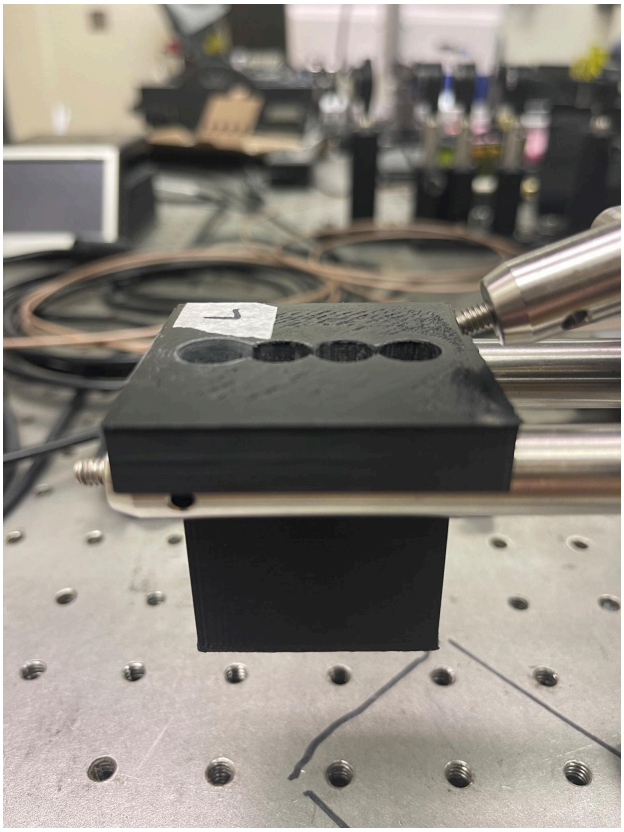


Figure 2: 3D printed emitter/detector holder and stabilizer, the emitted goes into the leftmost opening on the end and every opening to the right of the emitter is increasingly separated by one centimeter.

Using this holder for our laser/detector fibers we can situate the laser to hover above the sample and it will be stable. Our setup consists of this holder, the laser fiber, the detector fiber, the sample, the single photon avalanche diode detector, and a PC which collects the data into time-of-flight bins for TCSPC (Figure 3).

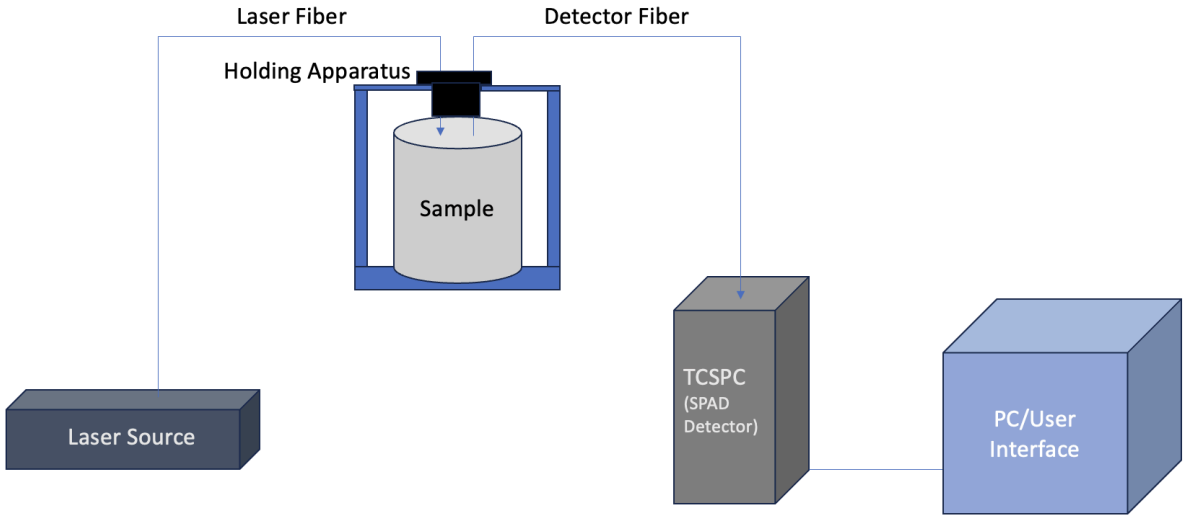


Figure 3: Experimental setup consisting of the laser source, laser fiber, the intralipid sample, the detector fiber, the holding apparatus for both the fibers, single photon avalanche diode detector, and PC for collecting data.

This setup was used for collecting scattering data for the intralipid phantom sample, but our end goal is to use this form of analysis for mapping cerebral blood flow which will occur with a detector placed directly onto the body. In order to better simulate this experiment, we also took measurements by pressing the laser/detector holder against an artery in the wrist (Figure 4).

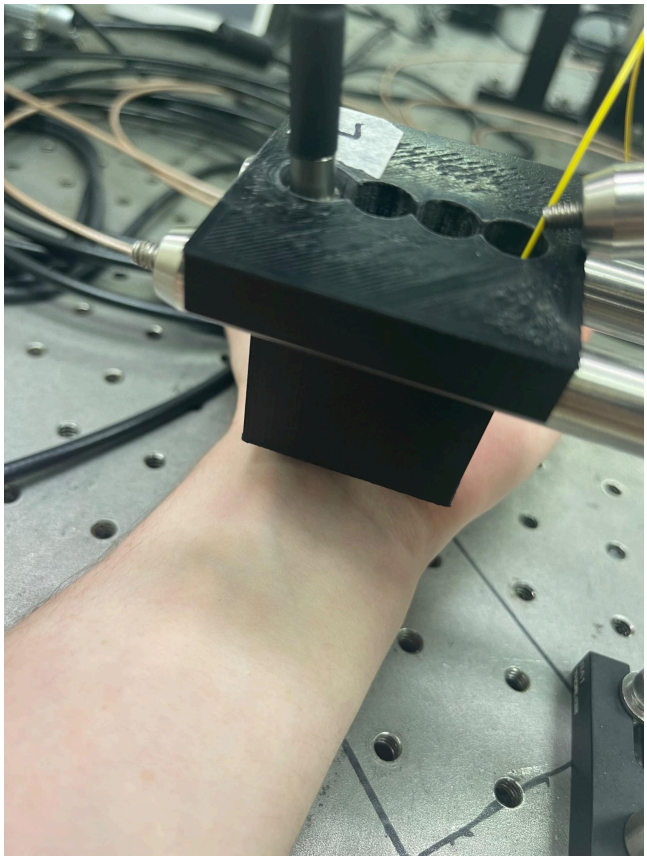


Figure 4: Experimental setup for measuring scattering from a blood artery in the wrist, better simulates how a laser interacts directly on the body and skin.

This experimental setup is essentially the same as for the intralipid sample, except this iteration uses the human wrist as the sample and adds some different aspects to the data such as pulse/heartbeat. Again, for this experimental setup you may arrange the detector fiber to be either 1 centimeter, 2 centimeters, or 3 centimeters away from the laser emitting fiber. This change in path length does directly affect the reflectance expected as well as the data collected, which we will discuss in the diffusion theories section of this paper.

For the last experimental setup, in order to measure the laser response function without the sample involved at all we must design a setup that allows the incident laser to signal the

detector without saturating the SPAD detector. This saturation arises when the laser reaches the detector with too intense of a power (Figure 5), but we mitigate this issue by feeding the laser through an array of neutral density (ND) filters which lower the overall power of the laser before it reaches the detector (Figure 6).

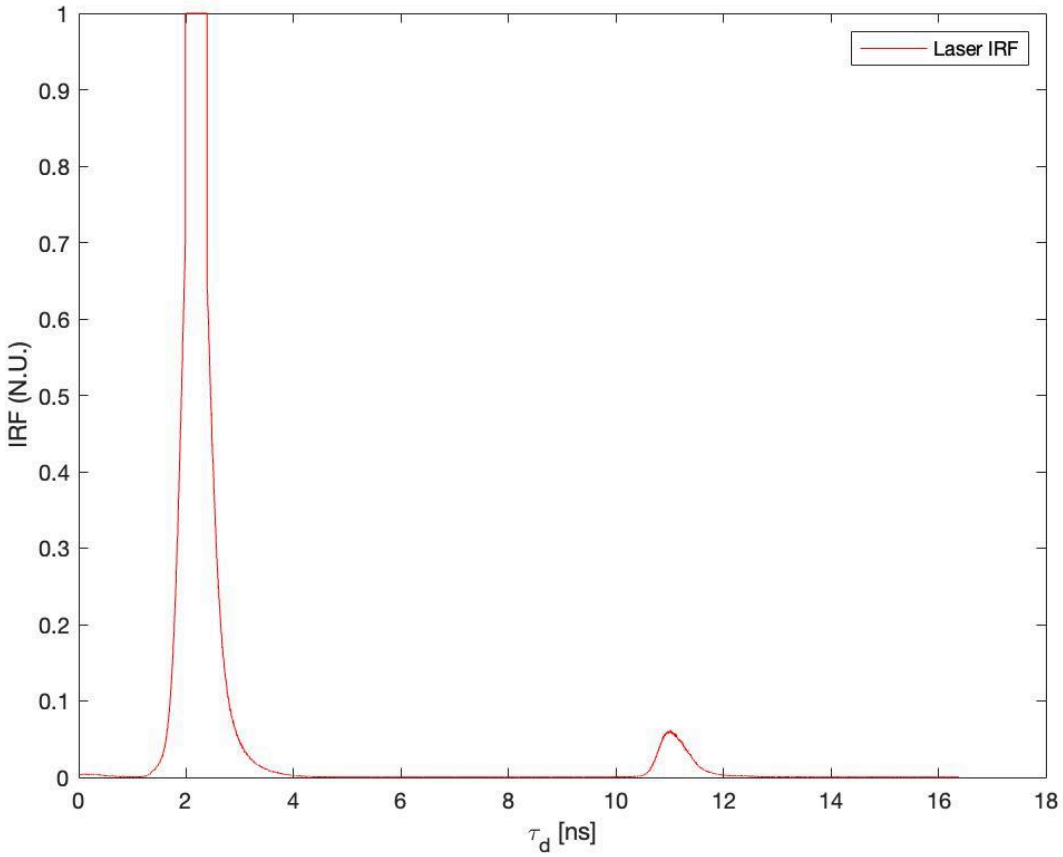


Figure 5: Simulated example of data collected when the SPAD detector is saturated with too intense of laser power. The peak of our data is essentially "cut off" and cannot be fully captured when the detector is saturated, leaving us unable to differentiate between the incline and decline of the peak near its maximum.

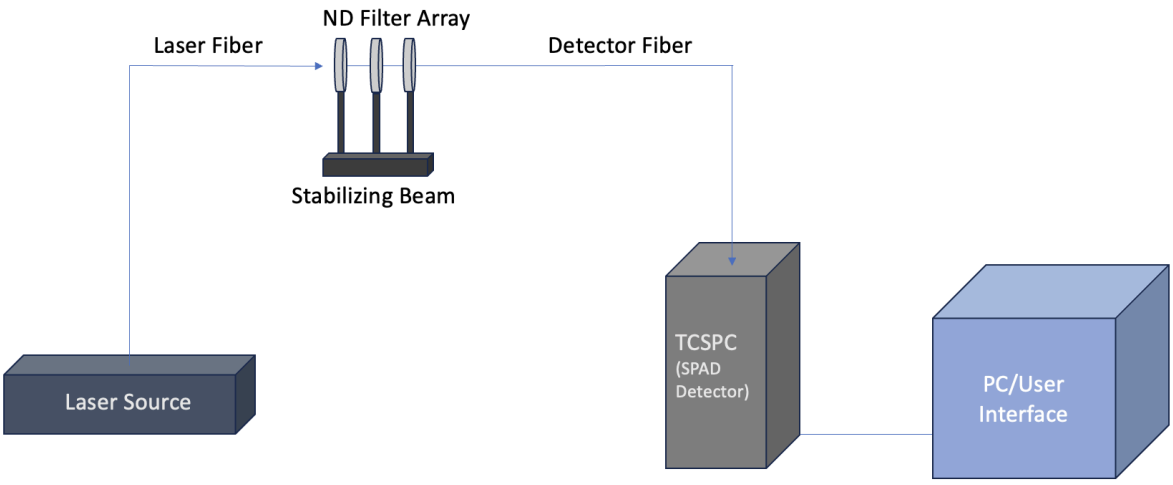


Figure 6: Setup for collecting the laser response function; includes a neutral density filter array of 3 ND filters as well as a stabilizing beam holding them in place to ensure the light is not refracted away from the detector.

The ND filters used in this array (ND = 4.0, 1.0, and 0.3) are important because each filter can reduce the laser power by a different amount, when calculating the laser power that will reach the detector each ND filter's respective percent transmittance is utilized as follows:

$$x_n = \%\ transmittance\ of\ n^{th}\ ND\ filter\ in\ array$$

$$\%P_0 = initial\ laser\ power\ percentage = 50\%$$

$$P = P_0 \prod_{n=1}^{n} x_n$$

For the three ND filters utilized in our experiment (Figure 7), they are 25 mm ThorLabs Absorptive Neutral Density Filters of which the transmission specs at 780 nm are known. Calculating using their percent transmittance, we have:

$$\%P = (0.50) \times (0.51) \times (0.17) \times (0.0018) = 0.00008$$

Therefore, the laser power that reaches the SPAD detector will be roughly 0.008 % power and will likely not saturate our detector because the laser's intensity is much too low to have this effect.

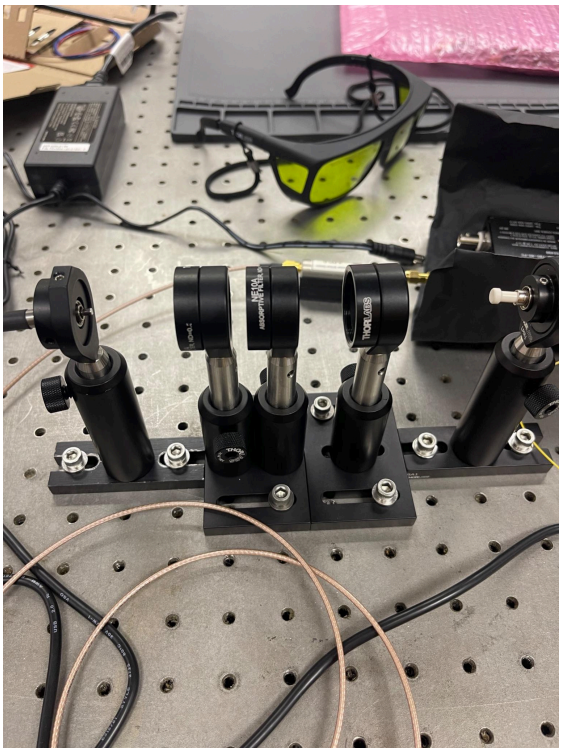


Figure 7: ND filter array, with the left side being the laser emitter fiber, and the right side is where the detector fiber is placed.

Diffusion Theories:

In order to interpret any data collected about the scattering from our samples, we must employ diffusion theories to explain this phenomenon. As previously mentioned, when measuring scattering time-of-flight data from the intralipid sample we observe a signal that is convoluted with a signal from the laser response function as well. This convolution causes us to measure a signal that is implicitly distorted to some degree by the laser electronics used to measure the signal. This distorted observed signal likely resides somewhere in between the two

shapes of the laser response function and the sample's true response function. This convolution is the reason why it is essential to measure the laser response function in addition to the signal from our sample. If we have data from both of these signals, then according to Lakowicz, we can model the convolution as an integral of these two functions [2]:

$$N(t) = \int_0^t L(t - \mu) I(\mu)\, d\mu \qquad (2)$$

Where N represents the observed convoluted signal, L represents the laser response function, and I is the true response function of the sample. With two of these functions known, we have narrowed our unknowns down to one function. In cases like this we can fit the measured signal with knowledge of the laser function in order to find the best fitting function for the true sample signal; this is done by guessing the form of this function and allowing for a variable parameter to find which parameter supplies the best fit, we then can assume this is the true nature of the signal response. Once we have acquired this true signal response function with a fitting in the form of eq. (2), we may apply the diffusion theories of reflectance to this calculated function.

According to Kienle and Patterson, reflectance can be represented by a set of diffusion equations including the fluence and diffuse reflectance [3]. These equations are approximations and require sets of conditions. In the diffusion theory utilized for our analysis, there are three different approximations: zero-boundary condition, extrapolated-boundary condition, and partial-current boundary condition. These boundary conditions require different sets of equations to represent the expected reflectance. We plan to fit these reflectance equations against our calculated true response function, by allowing the scattering coefficient to be a variable parameter. In doing this, we will find the best fit by fitting with the calculated true response function and determine the scattering coefficient of our sample. This scattering coefficient tells

us about the optical properties of our sample. In actuality, this optical property tells us the amount of photons our sample is scattering. Thus, when the sample is an artery this optical property will be measured in order to characterize the blood flow within the artery.

Fitting Analysis:

In fitting the measured convoluted signal function to an integral of the product of the laser's response function with the sample's true response function, we must be able to fit multiple varying parameters rapidly in order to find the most accurate fit to our observed plot. This was done in Matlab and the code utilizes most importantly a built-in Matlab function known as "fminsearch" [5]. This function allows the program to iterate through values for any amount of varying parameters and compare the output with some known constant value you choose to fit against. This comparison involves a square error calculation, and thus will recalculate over and over until the square error between the calculated function and the fitted function is minimized. When this difference is minimized, the value of the variables at this point are produced. In our case, we modeled the laser response function and the sample's true response function as having the form of an exponential decay function. In doing this, we are allowing for the exponential term to be multiplied by some varying coefficient as well as the time variable in the exponent to be multiplied by a varying coefficient. Thus, our eq. (2) looks more like:

$$N(t) = \int_0^t a e^{b(t-\mu)} \times \alpha e^{\beta\mu} dt \qquad (3)$$

Where a and b are constant coefficients for the laser response function, which are known simply with an exponential fitting to the collected laser response function data. And α and β are the unknown constant coefficients within the sample's true response function; μ is essentially a

dummy variable of integration and will disappear in the actual fitting equation. For our purposes in Matlab we represent these two convoluted functions as a simple product, seeing as the integral is not possible to fit with in this scenario; if we have the plot of N(t) from our observed signal of the sample, then we fit the product of these two exponential functions to this known data, with the only unknown variables being α and β. After fitting for these parameters, we have found our sample's true response function and the data has been deconvolved.

Once deconvolved, we plan to fit for the reflectance of our sample in a similar way utilizing Matlab's "fminsearch" function. However this function is much more complex as the reflectance equations are quite long and there are three different conditions to fit for. The reflectance equations for all three of the different boundary conditions depend upon $\mu_a,\ \mu_s',\ p,\ c,\ t,\ R_{eff},\ and\ n$. Where $\mu_a\ and\ \mu_s'$ are the absorption and reduced scattering coefficient, as discussed previously. p is the path length between the source and detector, c is the effective speed of light, t is the time, R_{eff} is the fraction of photons reflected at the boundary, and n is the index of refraction for the material. Since the absorption coefficient is material dependent, the path length is independent of the sample, the refractive index is material dependent, and c, t, and R_{eff} are all constants then we see that the reduced scattering coefficient is the only unknown variable which changes with changes within the sample. This allows us to characterize the blood flow in a sampled artery because we know that the absorption coefficient along with all other variables are unchanged by an increase in the blood flow. However, the reduced scattering coefficient should change with a change in blood flow as more photons should be scattered by more red blood cells traveling through the artery. With this in mind, we need only to let the reduced scattering coefficient (μ_s') be an adjustable parameter during our fitting to the

sample's true response function. If we apply the same technique of fitting, but instead using the reflectance equations as defined by Kienle and Patterson for each condition then we can approximate the reduced scattering coefficient of our sample's true response signal.

Trials:

To test our experimental setup and its applicability to human tissue, we used a sample of intralipid diluted with water. For the first experiment, we measured a mixture composed of 10% intralipid and 90% water (10% Lipid Solution). This solution's data was collected at intervals of 1, 2, and 3 centimeters of path length. The data was then analyzed with the response function fitting and reflectance fitting, where only μ_s' was allowed to vary, as discussed above. Next, in order to better test the fitting analysis of the reflectance fitting we measured a solution of 5% intralipid and 95% water in iterations of many trials, but with the path length held constant at 1 centimeter. In these 5% lipid solution trials we first measured it plainly, but the next measurement we added a small amount of ink $(100\ \mu L)$ and measured its signal again. We continued adding this amount of ink until we reached $300\ \mu L$ of ink mixed into our solution. Adding ink into the lipid solution is expected to affect the absorption coefficient due to an addition of a highly absorptive material (ink), but should not affect the reduced scattering coefficient because the amount of lipid in solution is remaining relatively unchanged. Consequently, we will need to slightly alter our method of fitting for this trial. The fitting for the true response function will be unchanged, but for reflectance we will need to now allow both μ_a and μ_s' to vary when fitting for this data. Our results are expected to show a relatively unchanged reduced scattering coefficient for the different solution regardless of amount of ink present, but

the absorption coefficient should theoretically increase with additions of ink. This experiment will allow us to check the validity of our reflectance fitting method as well as the measuring methods utilized for the response functions.

Results:

Trial #1 - 10% Lipid Solution:

From measuring the 10% lipid solution and the laser response function for this trial, we observed two plots with similar shape (Figure 8) for each path length interval. This is concurrent with the expectation, because the observed lipid signal is a convolution of the laser signal with another unknown signal; therefore the two signals should exhibit a similar shape.

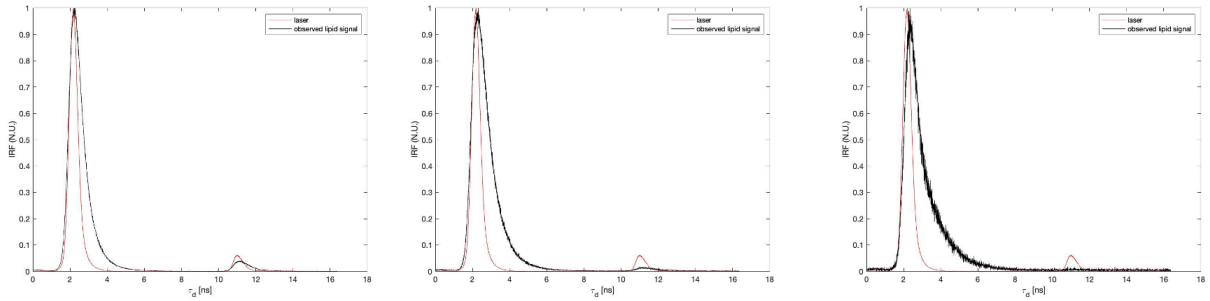


Figure 8: Raw data plots of the laser response function (red) compared to the observed lipid signal (black) for path length intervals of (a) 1cm, (b) 2 cm, and (c) 3cm.

Focusing on the fitting of the data collected at a path length of 1 centimeter (Figure 8a), we must first trim this data to only the right side of the peak and exclude peaks other than the maximum.

We trim this plot such that our data takes the shape of an exponential decay because this is the part of the data that will be important for us when fitting for reflectance (Figure 9).

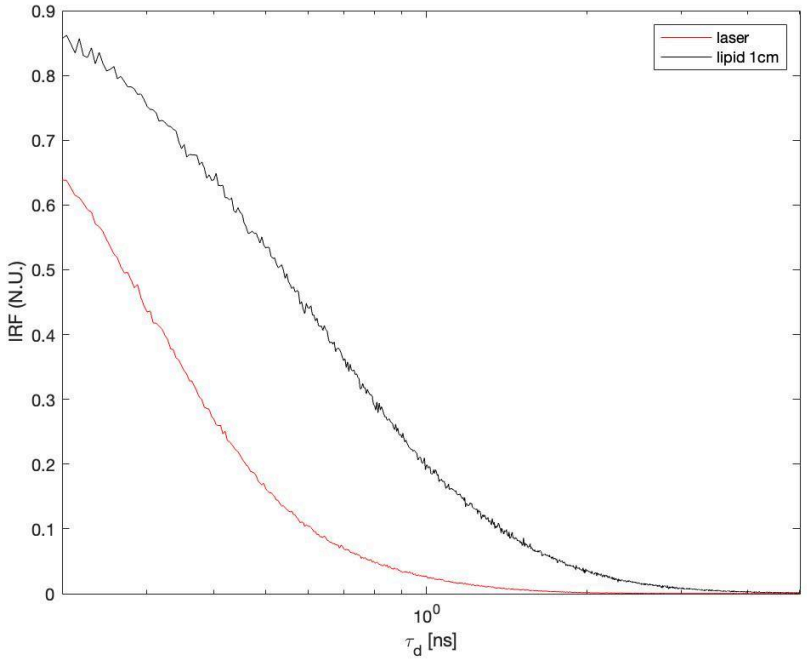


Figure 9: Trimmed data plot of the laser response function (red) and observed lipid response (black) for the 10% lipid sample at path length of 1 centimeter.

After this trimming of the data, we apply a simple exponential decay fitting to the laser response function, this is only for the purpose of characterizing our laser's signal into a function that we may express mathematically rather than visually. This fitting results in a relatively close approximation as we should expect because the laser response signal seems to have the shape of an exponential decay (Figure 10).

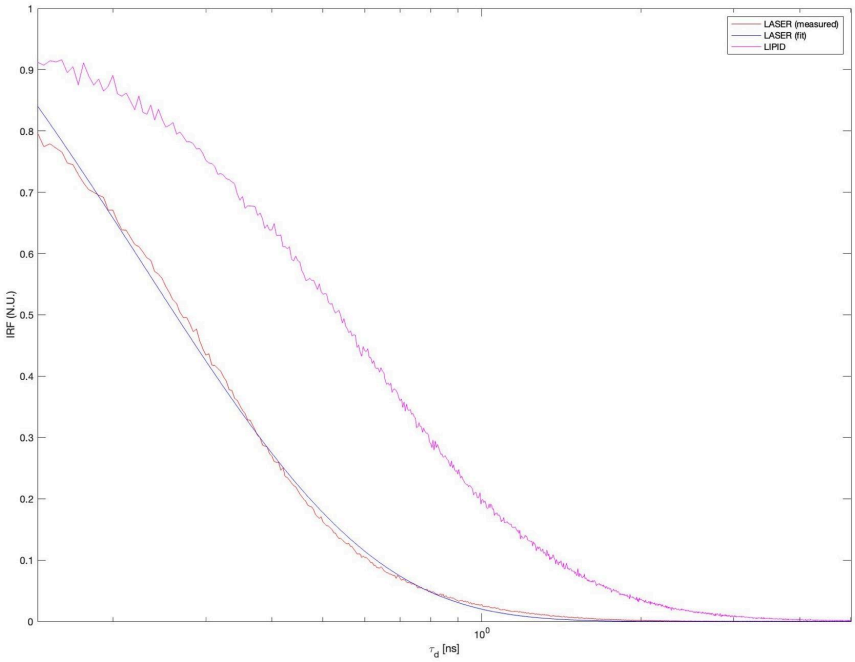


Figure 10: Plot of the measured laser response function (red), observed lipid signal (violet), and the fitted exponential decay function for the laser response (blue); all data is from the measurement of the 10% lipid sample at 1 cm path length.

Once we have fit for the laser's response function, we may know fit for the deconvoluted lipid response function by using the "fminsearch" Matlab function and the fact that we know the convolution is a product of the laser's response with the lipid's true response. This fitting takes place like a "trial and error" method in which we tell our program to start at some point for the unknown coefficients and then let it iterate over and over again until a non-linear least squares fitting is acquired that achieves the least amount of difference from the observed lipid function (Figure 11).

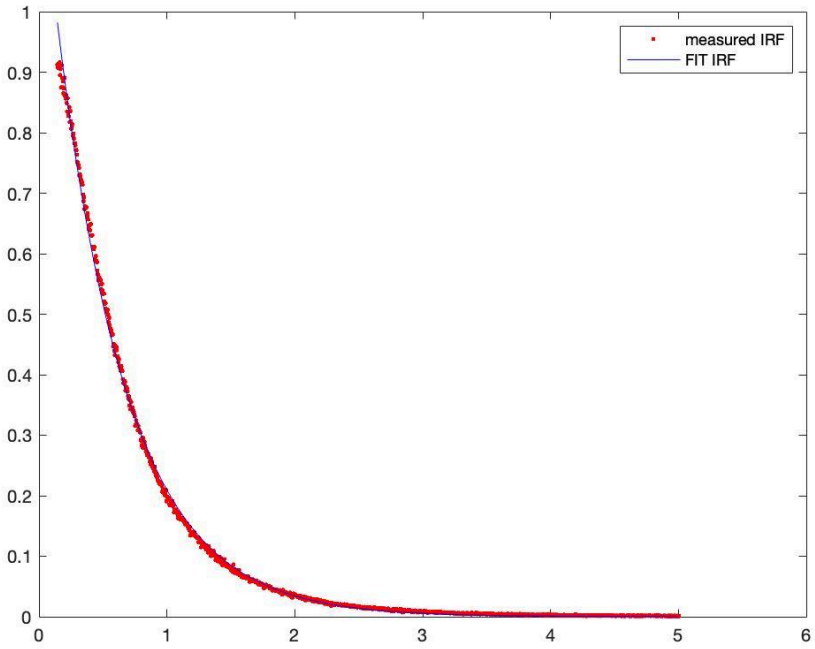


Figure 11: Result of the fitting for the true response function, plot of the observed lipid signal (red dotted line) compared to the product of the laser response function with the estimated true lipid response function (solid blue line) [4].

As you can see, the fitted convolution seemed to very closely resemble that of the actual observed convolution. Now, we may assume that the fitted coefficients for the lipid's true response function which is utilized to create this fitted convolution plot is an accurate representation of the lipid's true response function which we cannot directly observe. However, when plotting the response functions altogether, we notice a peculiarity (Figure 12).

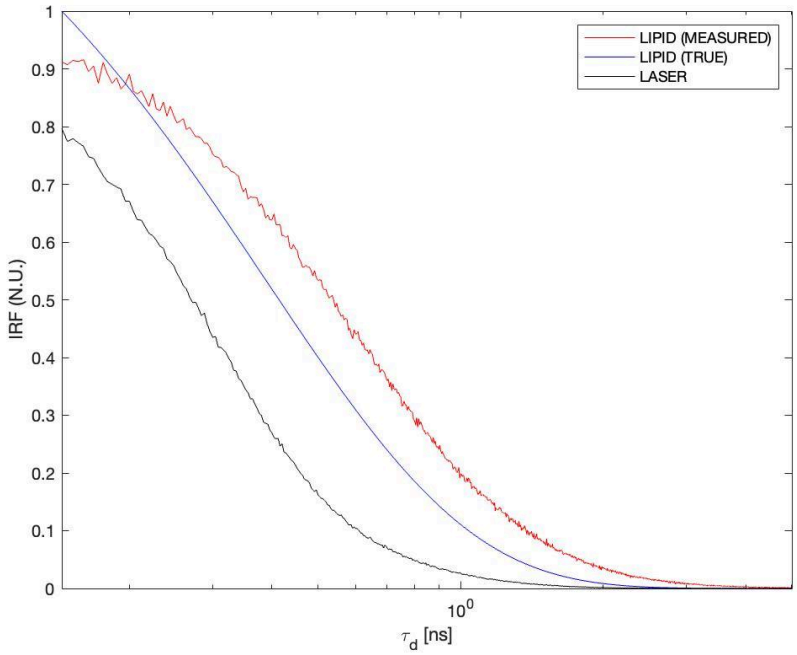
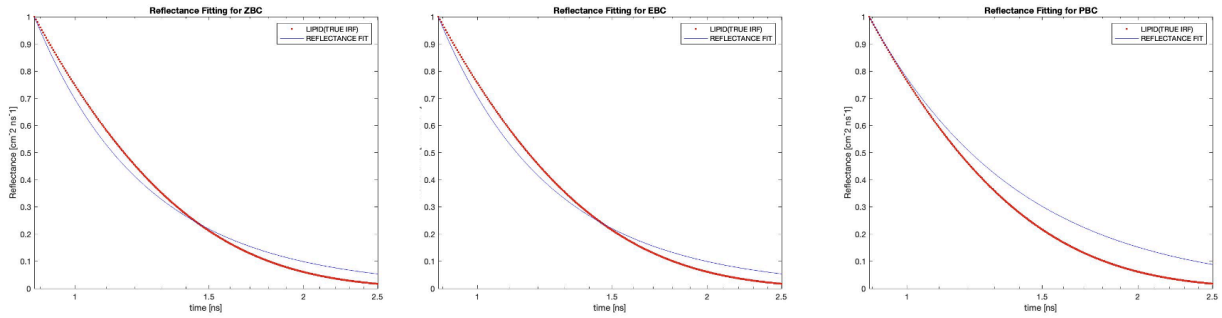


Figure 12: Plot of the observed convoluted lipid response function (red) compared to the fitted laser response function (black) and the fitted deconvolved true lipid response function (blue).

Notice that the convolution of the lipid's true signal with the laser's signal is higher than either the lipid's true deconvolved signal and the laser's fitted response function. This is incongruent with expectation, as we expect that convolution of a broader curve with a narrower curve to take the shape of a curve somewhere between the two as convolution is a contamination of these two signals. In other words, we expect our plot to display a true deconvolved lipid signal that is slightly broader than the convolved observed lipid signal, which is then also slightly broader than the laser's response signal. Thus, fitting this deconvolved signal to the theoretical reflectance equations for each different boundary condition will likely be inaccurate. Following through with the fitting for this trial, we observe a very unlikely result for the reduced scattering coefficient of our lipid signal (Figure 13).



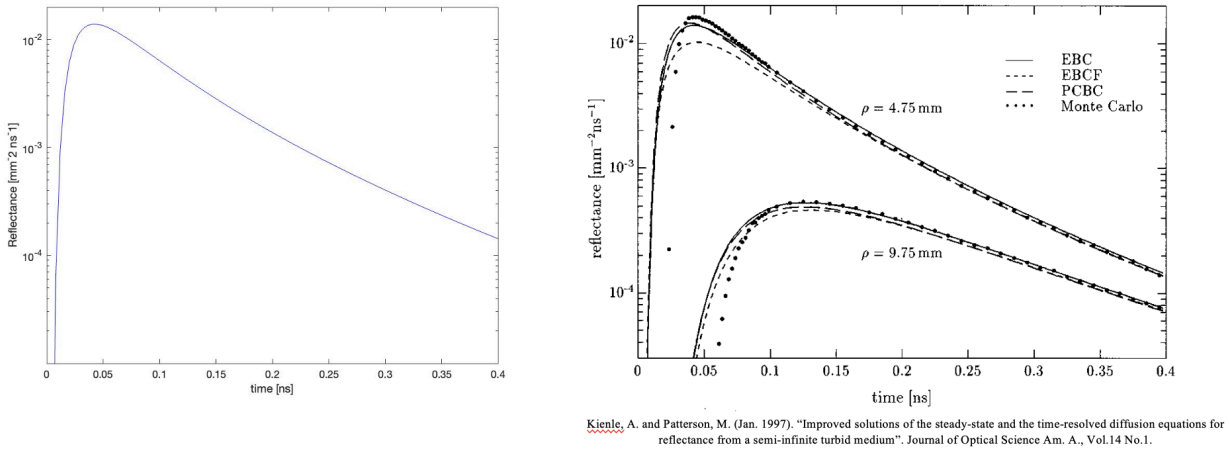
$\mu_s'(ZBC) = -11.74$ (a)

$\mu_s'(EBC) = -11.13$ (b)

$\mu_s'(PBC) = 3.48$ (c)

Figure 13: Reflectance fitting of the 10% lipid solution at 1 cm path length for boundary conditions (a) ZBC, (b) EBC, and (c) PBC. With the reduced scattering coefficient results listed below the respective boundary condition fitting method. Plot includes the deconvolved true lipid response function (red) compared to the fitted reflectance equations with varying μ_s' (blue) [6].

We observe from our fitting with the calculated deconvolved lipid signal to the reflectance equations mentioned previously, that the fitted function does not very closely resemble the shape of our deconvolved lipid signal. This results in reduced scattering coefficient values that do not make any physical sense, we expect the μ_s' to be positive and relatively congruent across these three boundary conditions. Thus, some error has taken place; in order to ensure that the error does not reside within the reflectance fitting code I have included an example calculation done in comparison to Kienle and Patterson's first figure [3]. Using the same parameters as Kienle and Patterson, we reproduced a plot included in the literature and thus proved that error did not originate from the reflectance equations being translated into the Matlab program (Figure 14).



(a) (b)

Figure 14: Comparison of plotting reflectance equations using (a) the Matlab code and with (b) the figure from literature. The Matlab code plot was calculated using the EBC only.

With this comparison, it is clear that the translation of reflectance equations from diffusion theory into the Matlab program are capable of reproducing the literature's plot and therefore are not the root of the error in the first trial.

Trial #2 - 5% Lipid Solution:

In order to further verify the validity of our fitting program calculations, we designed an experiment in which we will add ink to the solution but keep a constant path length. In this experiment, by allowing both the reduced scattering coefficient and the absorption coefficient to be adjustable parameters for the reflectance fitting we will verify whether the error originates from the fitting iterations or not. We begin in an identical way to the first experiment where we measure the laser's response function as well as the observed lipid's response function; here, we

increased the laser power when conducting this experiment to see if this would change the results in any way (Figure 15).

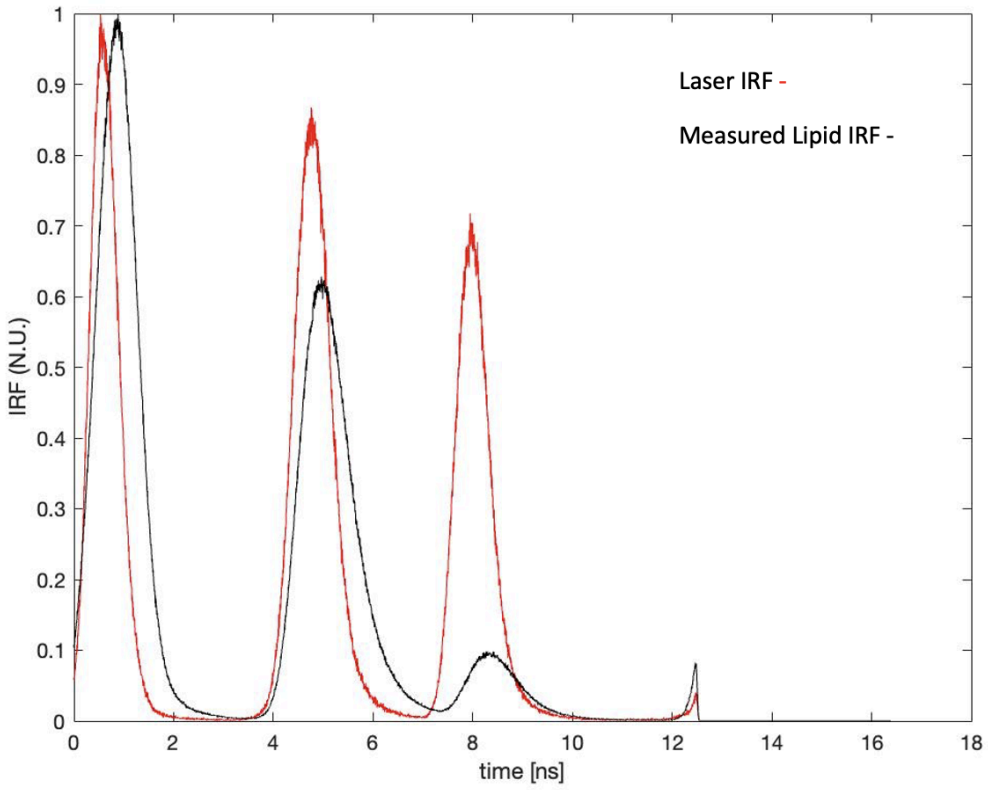


Figure 15: Comparison of the laser response function's normalized raw data (red) with the observed lipid signal's normalized raw data (black), both collected with the laser at 50% power.

Observably, we notice a very different shape than previously in the 10% lipid sample experiment. We can identify a "sawtooth" signal for both the laser response function and the observed lipid's response function; this can likely be attributed to an increase in the laser power which has now caused an afterpulse to be much more prevalent than before. The afterpulse exhibited in these signals is characteristic of SPAD detectors due to the extremely short time-of-flight detection on the order of a couple nanoseconds. The first and most intense peak is characteristic of the first scattered event received by the SPAD detector; however this signal is

produced by an absorption of the photons by the detector which enters a p-n junction (Figure 16). In this junction electrons from the n-region are excited into holes of the p-region. After some time, the excited electrons avalanche back into the n-region producing an electric signal due to the potential energy difference across the junction. Unfortunately, some of these excited electrons remain trapped in the p-region after the first avalanche and cannot return to the n-region until after this signal is received. These trapped electrons finally returning to the initial potential energy after the first peak are responsible for the afterpulse(s). This is the reason for the "staircase" shape of our response functions because the number of trapped electrons from one scattering event will decrease and decrease until there are none remaining. If our time-of-flight detection resolution were on a higher order of magnitude then this shape may be irrelevant, but due to the extremely short pulses this becomes a glaring error.

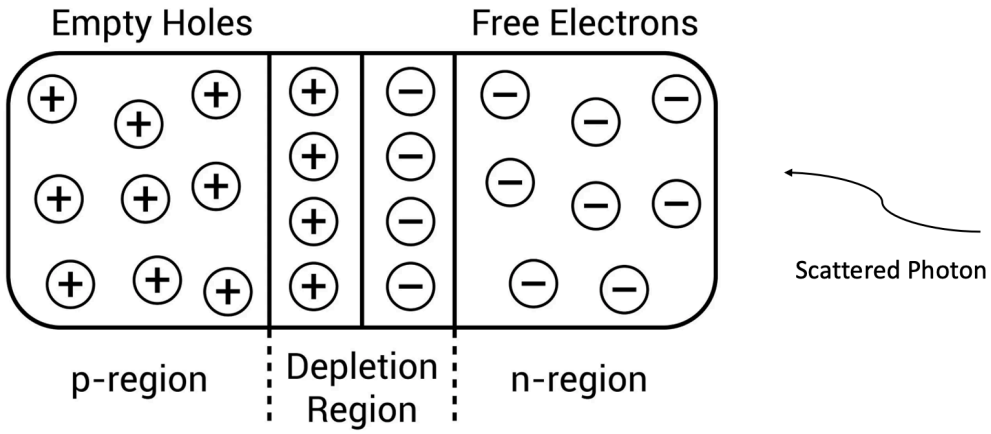


Figure 16: P-N Junction of the SPAD detector, where the n-region contains free electrons and the p-region contains holes where the electrons can jump to when absorbing incident photons.

Continuing the rest of the analysis for the 5% lipid solution with no ink added, we next trim the raw data for the laser response function and the observed lipid response function. After trimming, we apply the simple exponential decay fitting to the laser data and compare (Figure 17).

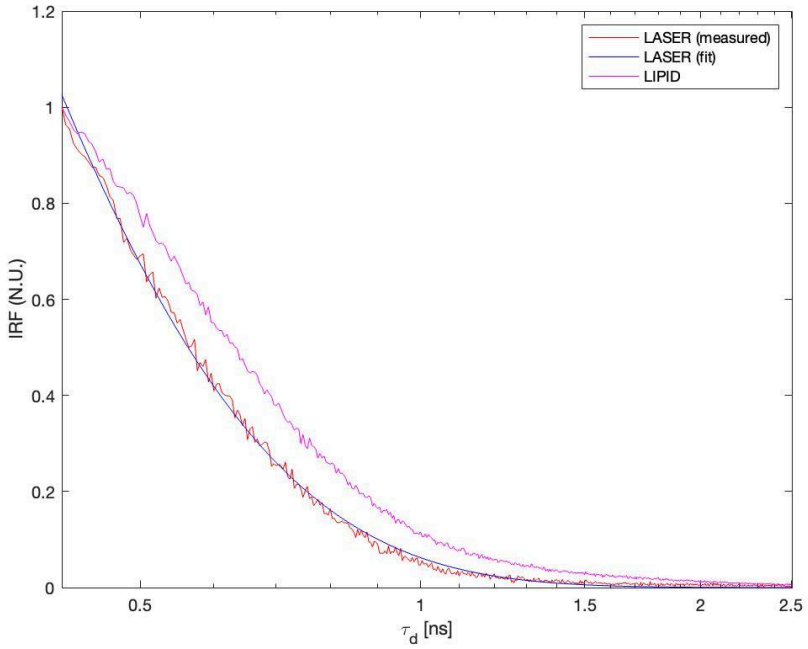


Figure 17: Plot of the fitted laser function (blue) with the measured laser response function (red) and the measured convoluted lipid response function (violet).

After achieving a sufficient fit for the laser response function, we may now complete a fitting for the lipid's true response function and compare (Figure 18).

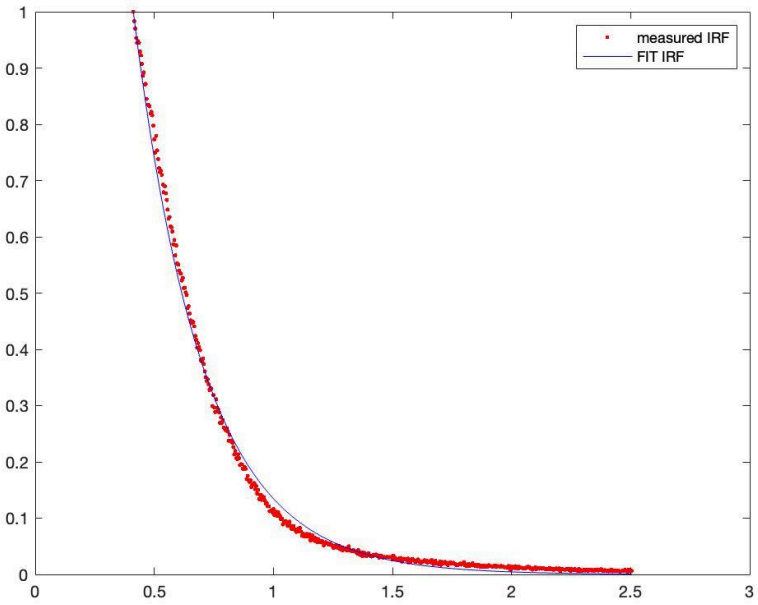


Figure 18: Fitted function for the lipid's true deconvolved signal (blue) compared to the actual measured convoluted lipid signal (red) [7].

Notice the fitted response function for the deconvolution of the lipid signal is less accurate than the results for the 10% lipid sample, but if we continue to fit for the reflectance using this function we may arrive at some more accurate results. Comparing this deconvolved function for the lipid's true signal with the original observed lipid signal and the laser's response function, we see a relationship more accurate to our expectation (Figure 19).

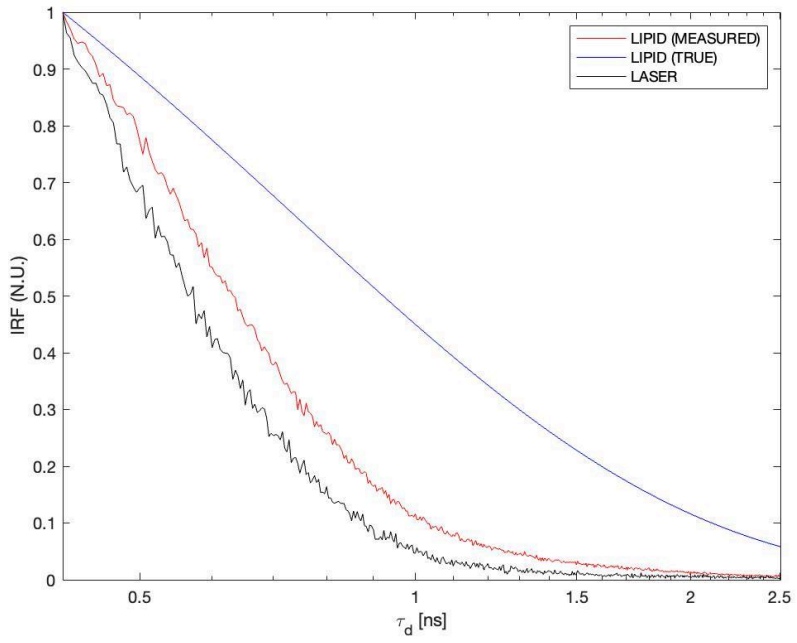
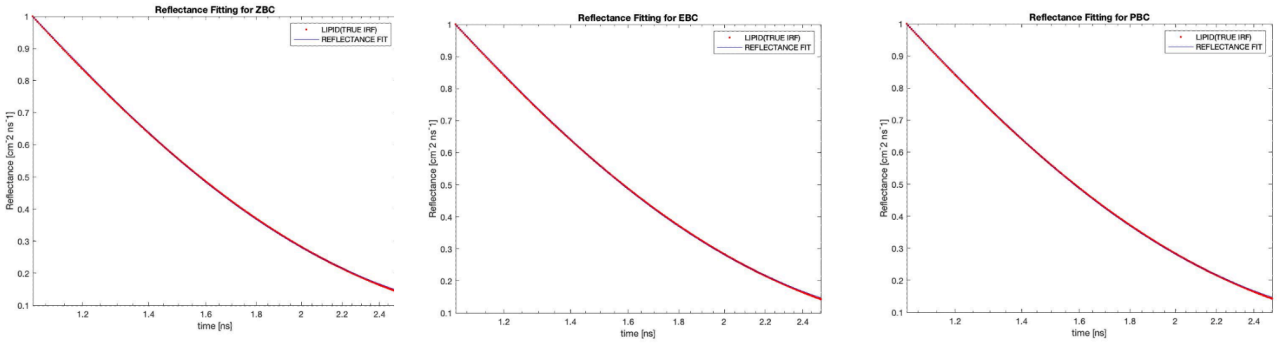


Figure 19: Comparison of the true deconvolved lipid response function (blue) with the laser's response function (black) and the lipid's observed response function.

Thus, in this trial the deconvolved lipid response function is observed to be a broader signal than either of the laser response function and the convolved observed lipid response function. This is as we would expect since the convolution between the lipid's true signal with the laser signal should reside somewhere between the two curves. Using this deconvolved function as our lipid's true response function, we may now fit for the reflectance of our sample and find results for the reduced scattering coefficient (Figure 20).



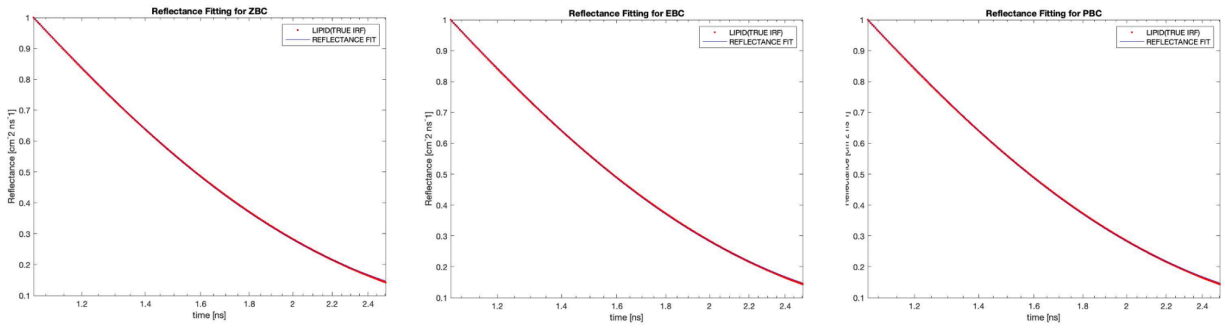
$\mu_s'(ZBC) = 54.81$
$\mu_a(ZBC) = 0.024$
(a)

$\mu_s'(EBC) = 54.75$
$\mu_a(ZBC) = 0.024$
(b)

$\mu_s'(PBC) = 54.63$
$\mu_a(ZBC) = 0.024$
(c)

Figure 20: Reflectance fitting of the 5% lipid solution at 1 cm path length for boundary conditions (a) ZBC, (b) EBC, and (c) PBC with no ink added. With both the reduced scattering coefficient and absorption coefficient results listed below the respective boundary condition fitting method. Plot includes the deconvolved true lipid response function (red) compared to the fitted reflectance equations with varying μ_s' and μ_a (blue).

We observe a much closer fit than previously for the reflectance, and all reduced scattering coefficient approximations are positive. Though this seems like a good result, we must still verify these results with reflectance fittings from the 5% lipid with different increments of ink added in order to test if the absorption coefficient changes and how (Figures 21, 22, and 23).

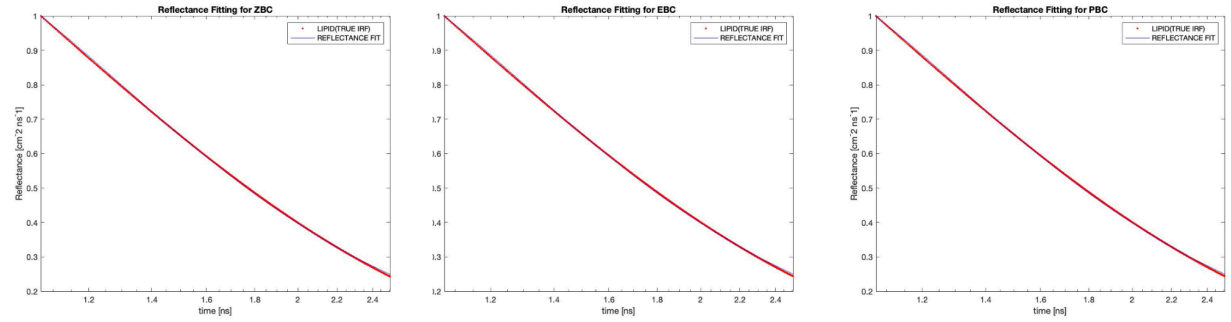


$\mu_s'(ZBC) = 54.82$
$\mu_a(ZBC) = 0.024$
(a)

$\mu_s'(EBC) = 54.76$
$\mu_a(ZBC) = 0.024$
(b)

$\mu_s'(PBC) = 54.63$
$\mu_a(ZBC) = 0.024$
(c)

Figure 21: Reflectance fitting of the 5% lipid solution at 1 cm path length for boundary conditions (a) ZBC, (b) EBC, and (c) PBC with $100\mu L$ ink added. Plot includes the deconvolved true lipid response function (red) compared to the fitted reflectance equations with varying μ_s' and μ_a (blue).



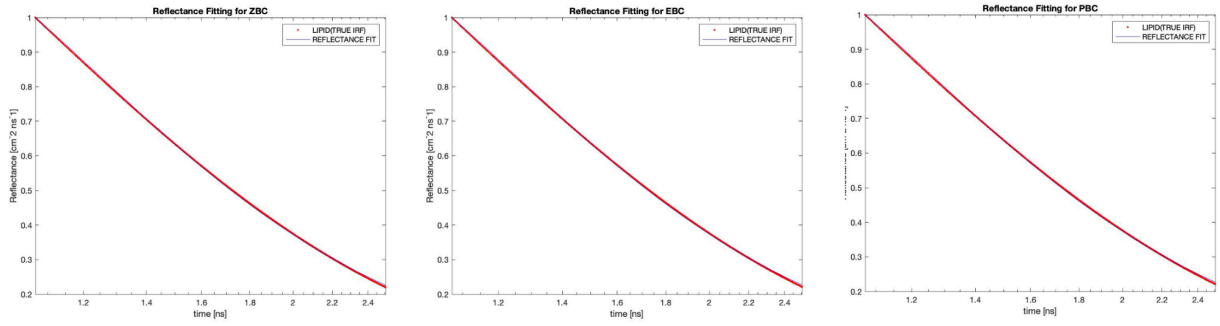
$\mu_s'(ZBC) = 55.70$
$\mu_a(ZBC) = 0.008$
(a)

$\mu_s'(EBC) = 55.64$
$\mu_a(ZBC) = 0.008$
(b)

$\mu_s'(PBC) = 55.51$
$\mu_a(ZBC) = 0.008$
(c)

Figure 22: Reflectance fitting of the 5% lipid solution at 1 cm path length for boundary conditions (a) ZBC, (b) EBC, and (c) PBC with $200\mu L$ ink added. Plot includes the deconvolved

true lipid response function (red) compared to the fitted reflectance equations with varying μ_s' and μ_a (blue).



$\mu_s'(ZBC) = 55.53$
$\mu_a(ZBC) = 0.011$
(a)

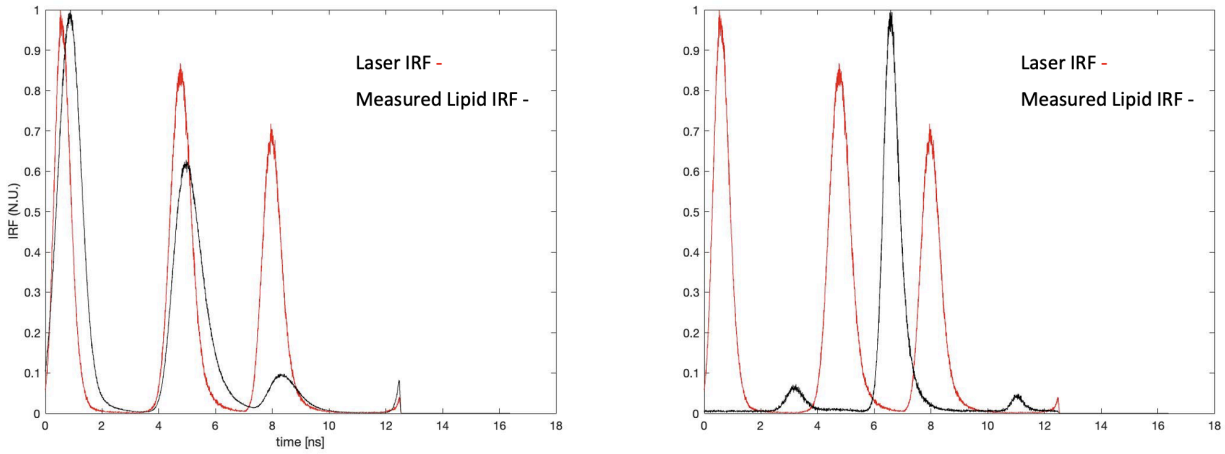
$\mu_s'(EBC) = 55.47$
$\mu_a(ZBC) = 0.011$
(b)

$\mu_s'(PBC) = 55.35$
$\mu_a(ZBC) = 0.011$
(c)

Figure 23: Reflectance fitting of the 5% lipid solution at 1 cm path length for boundary conditions (a) ZBC, (b) EBC, and (c) PBC with $300\mu L$ ink added. Plot includes the deconvolved true lipid response function (red) compared to the fitted reflectance equations with varying μ_s' and μ_a (blue) [8].

Among the different iterations of adding ink into our 5% lipid sample, we notice that the reduced scattering coefficient is relatively consistent across all of the intervals exactly as we proposed earlier. On the other hand, the absorption coefficient though changing between the different concentrations of ink seems to be randomly fluctuating rather than increasing as we expected. This change in absorption coefficient does not make physical sense with our expectations, as additional ink should undoubtedly increase the absorption coefficient. Investigating specifically

the case with 300 μL ink added to the solution and comparing this to the case where zero ink is added to the solution, we find that the raw data is inconsistent (Figure 24).



(a) (b)

Figure 24: Comparison of the raw data acquired for (a) 5% intralipid solution with zero ink added and (b) 5% intralipid solution with $300\mu L$ ink added, where the red line is the measured laser response function and the black line is the observed convoluted lipid response function.

Notice that the observed lipid signal in both cases differ from one another, and this should not occur. The addition of ink should not affect the shape of the signal so drastically, where one exhibits the "sawtooth" shape from before, but the other exhibits a middle peak being the maximum. Most importantly the shape of the laser's response function and the lipid's observed response function are no longer similar in the case where $300\mu L$ ink is added to solution, this inherently will produce error in both the reflectance fitting and the lipid's true response function fitting.

Conclusions:

Overall, the results from our experiment were inconclusive regarding characterization of the optical properties from tissue. The experimental trial of our 10% intralipid sample produced unrealistic results for the reduced scattering coefficient. In order to narrow down the origins of error of this experiment we designed a separate set of trials using a 5% intralipid sample with different additions of ink to the sample. In measuring this sample and fitting for both the reduced scattering coefficient and the absorption coefficient, we find that the reduced scattering coefficient remained consistent across the ink variations but the absorption coefficient fluctuated randomly. These results lead us to conclude that the origin of error resides in the actual collection of the raw data for the response functions rather than the data analysis of these functions. Specifically, the collection of the laser's response function due to the saturation of the detector as well as the array of ND filters leaves a glaring margin for error. Since our experimental measurements on the samples are acquired by measuring the scattering events, perhaps utilizing a scattering control sample like water and taking a measurement would provide a more accurate laser response function for deconvolving the lipid's measured signal than simply sending the beam through a filter array and measuring the transmitted signal. Aside from this, the SPAD detector's afterpulses can also cause error. However, these afterpulses are made so large due to the intensity of the beam as well as its extremely short pulses. Utilizing a less intense beam may more accurately measure the optical properties of the sample. Although our results are inconclusive for extracting the tissue optical properties from a photon distribution time-of-flight using an extremely short pulsed laser, this technique has been solved for continuous wave lasers for quite some time. With some improvements to the data collection and equipment we will be

much closer to solving this problem with a pulsed wave laser, which is more accessible and affordable for practical purposes. Future experimentation should take into account the results from these trials and the origination of the error within them. With careful consideration of these, we should next plan to measure the laser response function with a thin scattering layer between the beam emitter and detector fiber as well as test which beam intensity minimizes the afterpulses in a SPAD detector.

References:

[1] C. Fu, H. Zheng, G. Wang, Y. Zhou, H. Chen, Y. He, J. Liu, J. Sun, Z. Xu. "Three-Dimensional Imaging via Time-Correlated Single-Photon Counting". *Applied Sciences,* vol. 10, issue: 6, page: 1930, 2020. Link: https://doi.org/10.3390/app10061930. Date accessed: 04-10-2024.

[2] J. Lakowicz. "Principles of Fluorescence Spectroscopy". *Springer*, 4th edition, Chapter 4: Time-Domain Lifetime Measurements, pages 97-155, 2006. Link: https://doi.org/10.1007/978-0-387-46312-4_4. Date accessed: 04-16-2024.

[3] A. Kienle, M.S. Patterson. "Improved solutions of the steady-state and the time-resolved diffusion equations for reflectance from a semi-infinite turbid medium". *Journal of the Optical Society of America A*, vol. 14, no. 1, 1997. Link: https://opg.optica.org/josaa/viewmedia.cfm?uri=josaa-14-1-246&seq=0&html=true. Date accessed: 04-16-2024.

Appendix:

[4] Matlab Script for Analysis of 10% Lipid Data:

```
clear all;close all;clc
load irf
laserirf=double(dataArray_counts);
laserirf=laserirf/max(laserirf);
xx1=find(laserirf==1);
t=timeVec_ns;

load Lipid10_3cm
lipidirf=double(dataArray_counts);
lipidirf=lipidirf/max(lipidirf);
xx2=find(abs(lipidirf-1)<1E-6);

clc
figure(1);clf
plot(t,laserirf,'r')
hold on
plot(t,lipidirf,'k')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
legend('laser', 'observed lipid signal')
grid off

figure(2);clf                   %cut data for fitting
aa1=laserirf(xx1:end);          %laser irf
aa2=lipidirf(xx2:end);          %lipid irf
time_new=(1:length(aa2))*4*1E-3; %in nanosecond
x1=find(time_new>5,1,'first');   %futher triming data up to 5 ns
cut=find(abs(aa2-0.85)<1E-2,1,'first');
aa1=aa1(cut:x1);
aa2=aa2(cut:x1);
time_new=time_new(cut:x1);
plot(time_new,aa1,'r')
hold on
plot(time_new,aa2,'k')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
grid off
set(gca,'XScale','log')
legend('laser','lipid 1cm')

f = fit(time_new',aa1,'exp1');   %fitting for the laser resposne function

v = coeffvalues(f);
a = v(1);
b = v(2);
yy=a*(exp(b*time_new));

figure(3)
plot(time_new,aa1,'r')
 hold on
plot(time_new,yy,'b')
plot(time_new,aa2,'m')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
%formatplot
grid off
set(gca,'XScale','log')
legend('LASER (measured)','LASER (fit)','LIPID')

beta =  0.001;         %fitting for the lipid's true response function
k=0.001;
start  = [beta k];
result = fminsearch('IRFfittingexp1', start,[],time_new,aa2,a,b);
beta = result(1)       %true lipid response function coefficients
k=result(2)

lipidirf_fit = k*a*exp(time_new'.*(beta+b));
%%

%close all;clc
trueIRF = k*exp(-beta.*time_new');
lipid_trueirf=trueIRF/max(trueIRF);
figure(5)
plot(time_new,aa2,'r')
 hold on
plot(time_new,lipid_trueirf,'b')
plot(time_new,aa1,'k')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
grid off
set(gca,'XScale','log')
legend('LIPID (MEASURED)','LIPID (TRUE)','LASER')

%%
close all;clc    % parameters for reflectance fitting equations
trueIRF = k*exp(-beta.*time_new');
mu_s = 150;      %cm^-1
mu_a = .00001;   % cm^-1
n = 1.33;        % index of refraction
R_eff = 0.431;   % photons reflected @ boundary
c = (3E1)/n;     % effective speed of light in cm/ns
p = 1;           % radial distance in cm
time1 = time_new;

%ZBC for cutting fitted eq
z_0 = (1/(mu_a + mu_s));
z_b = 0;
D = (1/(3*(mu_a + mu_s)));
r_1 = sqrt(((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));
fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));
diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1')) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));
ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));
ref_fit_norm = ref_fit/max(ref_fit);
trimZBC = find(abs(ref_fit_norm-dcut)<1E-2,1,'last');
timeZBC = time1(trimZBC:end);
trueIRF_ZBC=trueIRF(trimZBC:end);

%EBC  for cutting fitted eq
z_0 = (1/(mu_a + mu_s));
z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));
D = (1/(3*(mu_a + mu_s)));
r_1 = sqrt(((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));
fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));
diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1')) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));
ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));
ref_fit_norm = ref_fit/max(ref_fit);
trimEBC = find(abs(ref_fit_norm-dcut)<1E-2,1,'last');
timeEBC = time1(trimEBC:end);
trueIRF_EBC=trueIRF(trimEBC:end);

%PBC  for cutting fitted eq
z = 0;
```

```
  z_0 = (1/(mu_a + mu_s));
  z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));
  D = (1/(3*(mu_a + mu_s)));
  fun = @(l) exp(-l/z_b).*exp(-(((z+z_0+l).^2)+(p^2)))./(4*D*c*time1'));
  L = integral(fun,0,Inf,'ArrayValued',true);
  fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z-z_0)^2)+(p^2))./(4*D*c*time1'))+exp(-(((z_0+z)^2)+(p^2))./(4*D*c*time1'))-(2/z_b)*L);
  ref_fit = (0.17*(fluence_fit));
  ref_fit_norm = ref_fit/max(ref_fit);
  trimPBC = find(abs(ref_fit_norm-dcut)<1E-2,1,'last');
  timePBC = time1(trimPBC:end);
  trueIRF_PBC=trueIRF(trimPBC:end);

  start = [mu_s mu_a];

   % time1 = time_new;                        % only activate for uncut fit

   time1 = timeZBC;                          % activate for cut fit
   trueIRF = trueIRF_ZBC/max(trueIRF_ZBC);   % activate for cut fit
  res_ZBC = fminsearch('reflectanceFittingZBCconst', start, [], time1, mu_a, c, p, R_eff, trueIRF);
  mu_sp_ZBC = res_ZBC(1)

   time1 = timeEBC;                          % activate for cut fit
   trueIRF = trueIRF_EBC/max(trueIRF_EBC);   % activate for cut fit
  res_EBC = fminsearch('reflectanceFittingEBCconst', start, [], time1, mu_a, c, p, R_eff, trueIRF);
  mu_sp_EBC = res_EBC(1)

   time1 = timePBC;                          % activate for cut fit
   trueIRF = trueIRF_PBC/max(trueIRF_PBC);   % activate for cut fit
  res_PBC = fminsearch('reflectanceFittingPBCconst', start, [], time1, mu_a, c, p, R_eff, trueIRF);
  mu_sp_PBC = res_PBC(1)
```

[5] Deconvolution Function:

```
function err = IRFfittingexp1(start,time_new,aa2,a,b,beta,k);
global cnt

beta = start(1);
k=start(2);


cnt=cnt+1;


lipidirf_fit = k*a*exp(time_new'.*(beta+b));

err=sum((aa2-lipidirf_fit).^2);% square error

figure(4);clf
plot(time_new,aa2,'r.')
hold on
plot(time_new,lipidirf_fit,'b')
drawnow
legend('measured IRF','FIT IRF')
```

[6] Reflectance Fitting Functions for Constant Absorption Coefficient (ZBC, EBC, and PBC respectively):

```
function err = reflectanceFittingZBCconst(start, time1, mu_a, c, p, R_eff, trueIRF);
global cnt

cnt = cnt + 1;

mu_s = start(1);

z_0 = (1/(mu_a + mu_s));
z_b = 0;
D = (1/(3*(mu_a + mu_s)));
r_1 = sqrt(((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));

fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));
diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1')) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));
ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));

ref_fit_norm = ref_fit/max(ref_fit);
trueIRF_norm = trueIRF/max(trueIRF);

err=sum((trueIRF_norm-ref_fit_norm).^2); % square error

figure(6);clf
plot(time1,trueIRF_norm,'r.')
hold on
plot(time1,ref_fit_norm,'b')
set(gca,'XScale','log')
drawnow
title('Reflectance Fitting for ZBC')
legend('LIPID(TRUE IRF)','REFLECTANCE FIT')
xlabel('time [ns]')
ylabel('Reflectance [cm^-2 ns^-1]')
```

```
function err = reflectanceFittingEBCconst(start, time1, mu_a, c, p, R_eff, trueIRF);
global cnt

cnt = cnt + 1;

mu_s = start(1);

z_0 = (1/(mu_a + mu_s));
z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));
D = (1/(3*(mu_a + mu_s)));
r_1 = sqrt(((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));

fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));
diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1')) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));
ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));

ref_fit_norm = ref_fit/max(ref_fit);
trueIRF_norm = trueIRF/max(trueIRF);

err=sum((trueIRF_norm-ref_fit_norm).^2); % square error

figure(7);clf
plot(time1,trueIRF_norm,'r.')
hold on
plot(time1,ref_fit_norm,'b')
set(gca,'XScale','log')
drawnow
title('Reflectance Fitting for EBC')
legend('LIPID(TRUE IRF)','REFLECTANCE FIT')
xlabel('time [ns]')
ylabel('Reflectance [cm^-2 ns^-1]')
```

```
function err = reflectanceFittingPBCconst(start, time1, mu_a, c, p, R_eff, trueIRF);
global cnt

cnt = cnt + 1;

mu_s = start(1);

z = 0;
z_0 = (1/(mu_a + mu_s));
z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));
D = (1/(3*(mu_a + mu_s)));

fun = @(l) exp(-l/z_b).*exp(-(((z+z_0+l).^2)+(p^2))./(4*D*c*time1'));
L = integral(fun,0,Inf,'ArrayValued',true);

fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z-z_0)^2)+(p^2))./(4*D*c*time1'))+exp(-(((z_0+z)^2)+(p^2))./(4*D*c*time1'))-(2/z_b)*L);
ref_fit = (0.17*(fluence_fit));

ref_fit_norm = ref_fit/max(ref_fit);
trueIRF_norm = trueIRF/max(trueIRF);

err=sum((trueIRF_norm-ref_fit_norm).^2); % square error

figure(8);clf
plot(time1,trueIRF_norm,'r.')
hold on
plot(time1,ref_fit_norm,'b')
set(gca,'XScale','log')
drawnow
title('Reflectance Fitting for PBC')
legend('LIPID(TRUE IRF)','REFLECTANCE FIT')
xlabel('time [ns]')
ylabel('Reflectance [cm^-2 ns^-1]')
```

[7] Analysis of 5% Lipid Ink Experiment:

```
clear all;close all;clc
load 'LaserIRF50%.mat'
laserirf=double(dataArray_counts(:,1));
laserirf=laserirf./max(laserirf);
xx1=find(laserirf==max(laserirf));
t = tofTimeVec_ns;

load 50perc_0ul_c1.mat
lipidirf=double(dataArray_counts(:,41));
lipidirf=lipidirf./max(lipidirf);
xx2=find(lipidirf==max(lipidirf));

clc
figure(1); clf
plot(t,laserirf,'r')
hold on
plot(t,lipidirf,'k')
grid off

figure(2); clf                  %cut data for fitting
dcut = 0.55;
aa1=laserirf(xx1:end);          %laser irf
aa2=lipidirf(xx2:end);          %lipid irf
time_new=[1:length(aa2)]*4*1E-3; %in nanosecond
x1=find(time_new>2.5,1,'first'); %futher triming data up to 5 ns
cut=find(abs(aa2-dcut)<1E-2,1,'first');
aa1=aa1(cut:x1);
aa2=aa2(cut:x1);
time_new=time_new(cut:x1);
plot(time_new,aa1,'r')
hold on
plot(time_new,aa2,'k')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
grid off
set(gca,'XScale','log')
legend('laser','lipid 1cm')

aa1 = aa1/max(aa1);
aa2 = aa2/max(aa2);

f = fit(time_new',aa1,'exp1');  %fitting the laser's response function
v = coeffvalues(f);
a = v(1);
b = v(2);
yy=a*(exp(b*time_new));

figure(3); clf
plot(time_new,aa1,'r')
 hold on
plot(time_new,yy,'b')
plot(time_new,aa2,'m')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
grid off
set(gca,'XScale','log')
legend('LASER (measured)','LASER (fit)','LIPID')

aa2 = aa2/max(aa2);

beta = 0.001;       %fitting for lipid's true response function
k= 0.001;
start  = [beta k];
result = fminsearch('IRFfittingexp1', start,[],time_new,aa2,a,b);
beta = result(1)    %result of true lipid response coefficients
k=result(2)

lipidirf_fit = k*a*exp(time_new'.*(beta*b));

%%
close all;clc
trueIRF = k*exp(-beta.*time_new');
lipid_trueirf=trueIRF/max(trueIRF);
figure(5)
plot(time_new,aa2,'r')
 hold on
plot(time_new,lipid_trueirf,'b')
plot(time_new,aa1,'k')
xlabel('\tau_d [ns]')
ylabel('IRF (N.U.)')
grid off
set(gca,'XScale','log')
legend('LIPID (MEASURED)','LIPID (TRUE)','LASER')

%%
close all;clc     % parameters for reflectance fitting equations
trueIRF = k*exp(-beta.*time_new');
mu_s = 150;       %cm^-1
mu_a = .01;       % cm^-1
n = 1.33;         % index of refraction
R_eff = 0.431;    % photons reflected @ boundary
c = (3E1)/n;      % effective speed of light in cm/ns
p = 1;            % radial distance in cm
time1 = time_new;

%ZBC for cutting fitted eq
z_0 = (1/(mu_a + mu_s));
z_b = 0;
D = (1/(3*(mu_a + mu_s)));
r_1 = sqrt((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));
fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));
diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1'))) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));
ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));
ref_fit_norm = ref_fit/max(ref_fit);
trimZBC = find(abs(ref_fit_norm-dcut)<1E-2,1,'last');
timeZBC = time1(trimZBC:end);
trueIRF_ZBC=trueIRF(trimZBC:end);

%EBC  for cutting fitted eq
z_0 = (1/(mu_a + mu_s));
z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));
D = (1/(3*(mu_a + mu_s)));
r_1 = sqrt((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));
fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));
diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1'))) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));
ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));
ref_fit_norm = ref_fit/max(ref_fit);
trimEBC = find(abs(ref_fit_norm-dcut)<1E-2,1,'last');
timeEBC = time1(trimEBC:end);
trueIRF_EBC=trueIRF(trimEBC:end);

%PBC  for cutting fitted eq
z = 0;
z_0 = (1/(mu_a + mu_s));
z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));
D = (1/(3*(mu_a + mu_s)));
fun = @(l) exp(-l/z_b).*exp(-(((z+z_0+l).^2)+(p^2))./(4*D*c*time1'));
L = integral(fun,0,Inf,'ArrayValued',true);
fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z-z_0)^2)+(p^2))./(4*D*c*time1'))+exp(-(((z_0+z)^2)+(p^2))./(4*D*c*time1'))-(2/z_b)*L);
ref_fit = (0.17*(fluence_fit));
ref_fit_norm = ref_fit/max(ref_fit);
trimPBC = find(abs(ref_fit_norm-dcut)<1E-2,1,'last');
timePBC = time1(trimPBC:end);
trueIRF_PBC=trueIRF(trimPBC:end);

start = [mu_s mu_a];

 % time1 = time_new;                          % only activate for uncut fit

 time1 = timeZBC;                            % activate for cut fit
 trueIRF = trueIRF_ZBC/max(trueIRF_ZBC);     % activate for cut fit
res_ZBC = fminsearch('reflectanceFittingZBC', start, [], time1, c, p, R_eff, trueIRF);
mu_sp_ZBC = res_ZBC(1)
mu_a_ZBC = res_ZBC(2)

 time1 = timeEBC;                            % activate for cut fit
 trueIRF = trueIRF_EBC/max(trueIRF_EBC);     % activate for cut fit
res_EBC = fminsearch('reflectanceFittingEBC', start, [], time1, c, p, R_eff, trueIRF);
mu_sp_EBC = res_EBC(1)
mu_a_EBC = res_EBC(2)

 time1 = timePBC;                            % activate for cut fit
 trueIRF = trueIRF_PBC/max(trueIRF_PBC);     % activate for cut fit
res_PBC = fminsearch('reflectanceFittingPBC', start, [], time1, c, p, R_eff, trueIRF);
mu_sp_PBC = res_PBC(1)
mu_a_PBC = res_PBC(2)
```

[8] Reflectance Fitting Functions for Varying Absorption Coefficient (ZBC, EBC, and PBC respectively):

```
function err = reflectanceFittingZBC(start, time1, c, p, R_eff, trueIRF);
global cnt

cnt = cnt + 1;

mu_s = start(1);
mu_a = start(2);

z_0 = (1/(mu_a + mu_s));

z_b = 0;

D = (1/(3*(mu_a + mu_s)));

r_1 = sqrt(((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));

fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));

diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1')) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));

ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));

ref_fit_norm = ref_fit/max(ref_fit);

trueIRF_norm = trueIRF/max(trueIRF);

err=sum((trueIRF_norm-ref_fit_norm).^2); % square error

figure(6);clf
plot(time1,trueIRF_norm,'r.')
hold on
plot(time1,ref_fit_norm,'b')
set(gca,'XScale','log')
drawnow
title('Reflectance Fitting for ZBC')
legend('LIPID(TRUE IRF)','REFLECTANCE FIT')
xlabel('time [ns]')
ylabel('Reflectance [cm^-2 ns^-1]')
```

```
function err = reflectanceFittingEBC(start, time1, c, p, R_eff, trueIRF);
global cnt

cnt = cnt + 1;

mu_s = start(1);
mu_a = start(2);

z_0 = (1/(mu_a + mu_s));

z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));

D = (1/(3*(mu_a + mu_s)));

r_1 = sqrt(((z_0)^2)+(p^2));
r_2 = sqrt(((z_0 + 2*(z_b))^2)+(p^2));

fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z_0)^2)+(p^2))./(4*D*c*time1'))-exp(-(((z_0+(2*z_b))^2)+(p^2))./(4*D*c*time1')));

diffuse_ref = ((1/2)*((4*pi*D*c)^(-3/2))*(time1'.^(-5/2)).*exp(-mu_a*c*time1')).*((z_0)*exp(-((r_1)^2)./(4*D*c*time1')) + (z_0 + 2*(z_b))*exp(-((r_2)^2)./(4*D*c*time1')));

ref_fit = (0.118*(fluence_fit))+(0.306*(diffuse_ref));

ref_fit_norm = ref_fit/max(ref_fit);

trueIRF_norm = trueIRF/max(trueIRF);

err=sum((trueIRF_norm-ref_fit_norm).^2); % square error

figure(7);clf
plot(time1,trueIRF_norm,'r.')
hold on
plot(time1,ref_fit_norm,'b')
set(gca,'XScale','log')
%set(gca,'YScale','log')
drawnow
title('Reflectance Fitting for EBC')
legend('LIPID(TRUE IRF)','REFLECTANCE FIT')
xlabel('time [ns]')
ylabel('Reflectance [cm^-2 ns^-1]')
```

```
function err = reflectanceFittingPBC(start, time1, c, p, R_eff, trueIRF);
global cnt

cnt = cnt + 1;

mu_s = start(1);
mu_a = start(2);

z = 0;

z_0 = (1/(mu_a + mu_s));

z_b = 2*(1/(3*(mu_a + mu_s)))*((1 + R_eff)/(1 - R_eff));

D = (1/(3*(mu_a + mu_s)));

fun = @(l) exp(-l/z_b).*exp(-(((z+z_0+l).^2)+(p^2))./(4*D*c*time1'));
L = integral(fun,0,Inf,'ArrayValued',true);

fluence_fit = (c*((4*pi*c*D*time1').^(-3/2))).*(exp(-mu_a*c*time1')).*(exp(-(((z-z_0)^2)+(p^2))./(4*D*c*time1'))+exp(-(((z_0+z)^2)+(p^2))./(4*D*c*time1'))-(2/z_b)*L);

ref_fit = (0.17*(fluence_fit));

ref_fit_norm = ref_fit/max(ref_fit);

%cutting fitting eq
% r1 = find(abs(ref_fit_norm-.965)<1E-2,1,'last');
% time2 = time1(r1:end);
% trueIRF = k*exp(-beta.*time2');
% ref_fit_norm = ref_fit_norm(r1:end);

trueIRF_norm = trueIRF/max(trueIRF);
% ref_fit_norm = ref_fit_norm/max(ref_fit_norm);

err=sum((trueIRF_norm-ref_fit_norm).^2); % square error

figure(8);clf
plot(time1,trueIRF_norm,'r.')
hold on
plot(time1,ref_fit_norm,'b')
set(gca,'XScale','log')
%set(gca,'YScale','log')
drawnow
title('Reflectance Fitting for PBC')
legend('LIPID(TRUE IRF)','REFLECTANCE FIT')
xlabel('time [ns]')
ylabel('Reflectance [cm^-2 ns^-1]')
```