YiER Final Report: Physics Based Optimization Algorithms and Machine Learning for THz-TDS

Theo Smith: 32809344

April 2025

Abstract

Terahertz time domain spectroscopy is a technique that examines the properties of materials using THz radiation. Extracting these properties experimentally can be unstable and laborious. In particular, existing methods struggle to stably extract data with multiple reflections. This is largely due to the fact that the Newton-Raphson fitting method typically uses an analytical derivative of the transfer function. The current methods also struggle to account for multiple layers. This project has developed new extraction methods to allow for stable and accurate extraction for samples using the models defined by the Fresnel equations, internal reflections in the emitter and samples with multiple layers. Fitting is performed with a combination of Bayesian optimization and gradient descent. The use of PyTorch's Autgrad engine eliminates the need for an analytical derivative. Matrix transfer methods are used as the forward pass in this process. These new optimization algorithms were tested on both simulated and experimental data. The matrix transfer methods are also capable of creating large datasets of time domain scans for different material parameters and sample characteristics. These datasets have been used to train a CNN to identify the number of layers in a sample with an accuracy of 80 % on unseen data.

Contents

1 Theoretical Background

The Terahertz (THz) region of the electromagnetic spectrum typically covers frequencies from 0.3 to 10 THz [10] which is found between the microwave and infrared regions. This region has proven problematic to work in as designing suitable emitters and detectors is difficult [25]. This is because electronic components do not operate at wavelengths greater than several hundred gigahertz. Meanwhile optical devices such as semiconductor diodes (designed for visible and near IR) cannot simply be extended to longer wavelengths [13]. This is because at room temperature the thermal energy of the environment (kT) is comparable with the photon energies of THz radiation. This makes population inversion impossible (at this temperature) as all energy levels are equally filled. Cryogenic cooling of the sample would be necessary to use traditional laser techniques to produce THz.
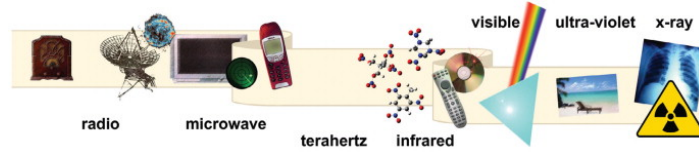

Figure 1: Schematic of the electromagnetic spectrum showing the location of terahertz from [13].

Several features of THz radiation make it ideal for applications in testing and analysis. For example they penetrate most non-metal and non-polar materials, meaning substances can be tested through packaging [13] [30] reducing the risk of exposure and making the testing process faster. Additionally, photon energies are low enough to be non-ionizing allowing them to be used in personnel scanning, common in airports for security screening [44] [17]. Industrial solutions for THz Imaging exist through providers such as Menlo Systems. These systems provide an out of the box solution for THz-TDS and imaging. TeraLyzer is a software solution provided that can provide material characterisation and analysis from TDS data. Existing software solutions are not optimized for extracting all characteristics of multi-layered samples. However they are capable of extracting thickness through an iterative minimization process. This leaves scope for more sophisticated algorithms. Similar solutions are provided by TeraView, who provide technology solutions for a variety of THz applications. These include multi-layered systems such as batteries and drug coatings.

1.1 THz Generation and Detection

As traditional laser methods cannot be used to produce THz radiation, an alternative method is used. This is achieved by exciting a photoconductive antenna (PCA) with a femto-second (fs) pulsed laser [41]. The characteristics of the pulses and the repetition rate of the laser, determine the THz profile achievable from the PCA. For this application a typical fs-pulsed laser produces 800 nm light with a 12 fs-pulse width and an 80 MHz repetition rate [41].
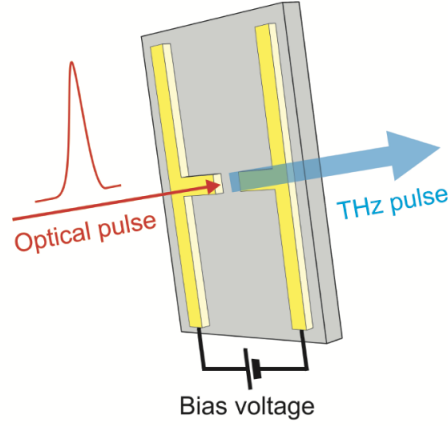
The pulsed laser is directed through a beam splitter and down two arms of the setup. One arm contains the emitter PCA, with a bias voltage applied across it. The light from the pulsed laser produces charge carriers in the PCA that are accelerated by the potential difference across it. This creates a Hertzian dipole emitting an electric field that is proportional to the rate of change of the photocurrent within the PCA [11]. This electric field is directed towards the detector PCA and lasts for the duration of the pulse and lifetime of the charge carriers [16]. The second arm of the pulsed laser has a second PCA (detector). This does not have a bias voltage across it, instead; when the pulsed laser light and the THz radiation are incident on the PCA the electric field from the THz radiation accelerates the charge carriers produced. This photocurrent is measured by an ammeter. PCA responses to the fs-pulsed laser are modelled in literature [16].
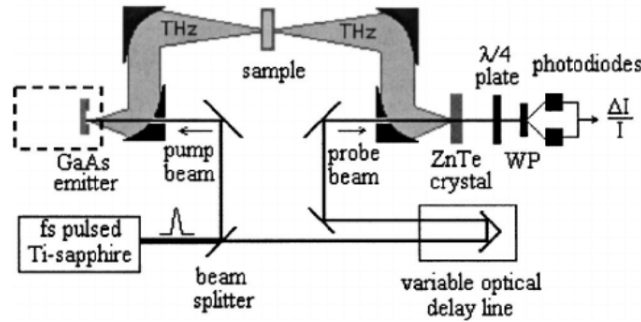

Figure 3: Schematic of THz-TDS setup, showing a fs-pulsed laser being directed down two arms synchronously hitting an emitter and detector with a delay line on one arm. The delay line allows the detector to scan through the time domain of the THz pulse. The diagram is taken from [41].

Given that the two arms are synchronous, this would measure a single point on the THz pulse.

To measure the entire time range of the THz pulse, a delay line is added to one arm and is used to scan the THz pulse in time. Figure 4 shows the time domain signal for the reference and sample pulses in a 3 mm thick sample of silicon glass. Given the signals are synchronous, all background radiation is effectively averaged out over during scan [25]. This allows for the detection of very small signals by limiting the effect of environmental noise, as this would also need to be synchronous at both ends of the setup.
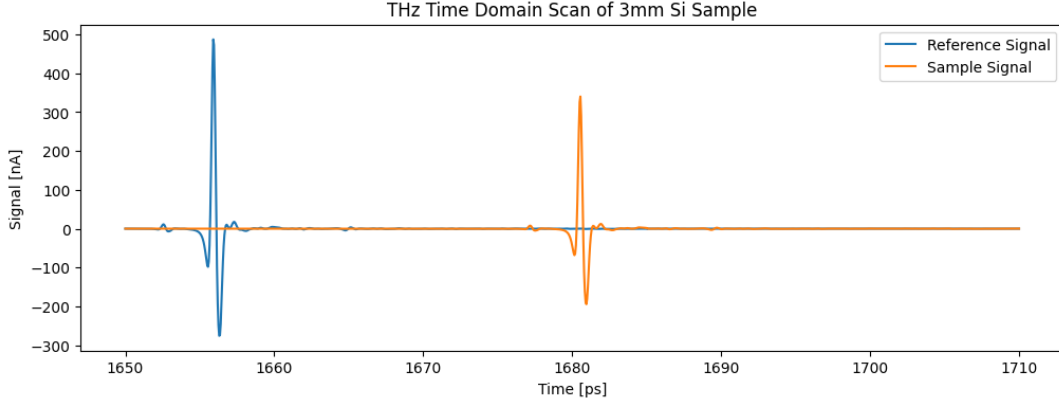

Figure 4: Time domain scan for a 3 mm thick sample of Si. The reference pulse (blue) propagates through air to profile the THz pulse, the sample pulse (orange) propagates through the sample.

It is possible to calculate an average real refractive index from this time domain scan by measuring the delay between the reference and sample pulse, using the equation below:

$$n_{average} = \frac{c \cdot \Delta t_{peak}}{L} + 1 \tag{1}$$

Where L is the thickness of the sample, c is the speed of light in a vacuum and Δt_{peak} is the time delay.

If k is negligible, the following analytical solution can be used.

$$n(\omega) = -[\frac{\phi(\omega)}{L} \cdot \frac{c}{\omega} + 1] \tag{2}$$

$\phi(\omega)$ is the phase of the transfer function, c is the speed of light, ω is the angular frequency, L is the length of the path the wave takes through the sample [25]. This allows for an analytical approximation of a frequency dependent refractive index. This can also be performed for $k(\omega)$ with the following expression.

$$k(\omega) = \ln[\frac{4n(\omega)}{|H(\omega)| \cdot (n(\omega) + 1)^2}] \cdot \frac{c}{\omega L} \tag{3}$$

1.2 Existing Methods for THz-TDS Material Parameter Extraction

THz-TDS measures the time domain of the electric field associated with the THz pulse as it propagates through a medium and through air [15]. When this is transformed to the frequency domain

usually using an FFT (fast Fourier transform), the resulting spectrum is complex, allowing for a direct extraction of complex refractive index. This allows us to measure refractive index and the absorption coefficient without relying on assumptions [22]. The extraction process involves taking two measurements of the electric field in time, once without the sample (the reference pulse) and once with the sample. These are then Fourier transformed to the frequency domain and the sample is divided by the reference producing a frequency dependent transfer function. Comparing this to a chosen theoretical model allows us to fit material parameters such as complex refractive index or conductivity [22]. The relationship between the input and output signals in the sample can be expressed using a transfer function in the frequency domain.

$$Y(\omega) = H(\omega) \cdot X(\omega). \tag{4}$$

Where $Y(\omega)$ is the output (experimentally this is the sample signal), $H(\omega)$ is the transfer function and $X(\omega)$ is the input signal (experimentally, the reference signal). The experimental transfer function can be found by dividing the sample by the reference. This assumes the system is linear and time invariant [11]. Figure 6 shows the frequency domain of a THz-TDS scan of a 3 mm sample of Si. These are then divided to get the transfer function.

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} \tag{5}$$

When working with a complex valued transfer function, the phase can be calculated as:

$$\phi(\omega) = \arctan(\frac{Im[H(\omega)]}{Re[H(\omega)]}) \tag{6}$$

This is bound between $-\pi$ and π, this causes the arctangent to jump between these two values creating a wrapped phase. If phase increases linearly with frequency, then the resulting wrapped phase will resemble a sawtooth function. This can be seen in Figure 5.
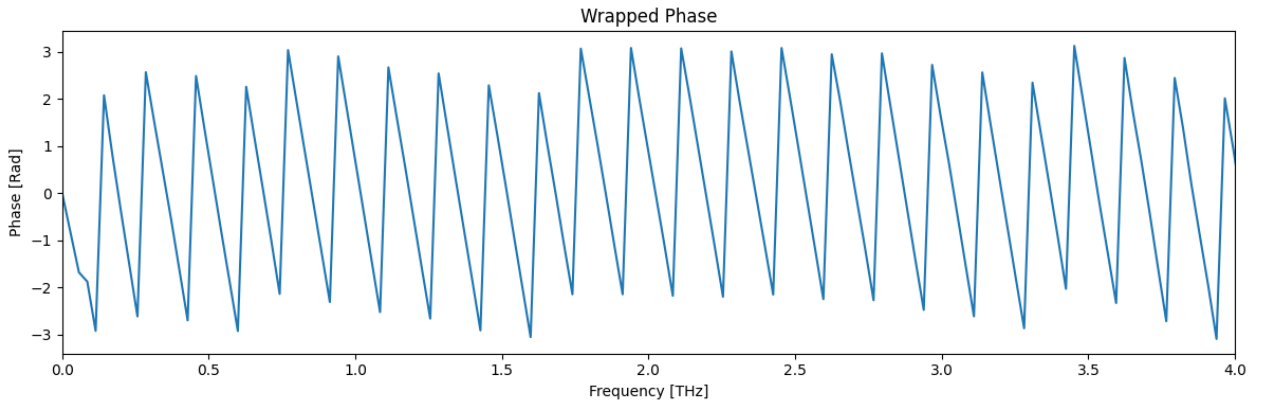

Figure 5: Wrapped phase from a reference pulse in a THz-TDS. The wrapping resembles a sawtooth function.

Phase unwrapping can be performed by detecting changes in phase of over π radians. At this point 2π is added to that phase point. This results in a smooth accumulation of phase. This can be seen in the phase of the FFT in Figure 6.

7
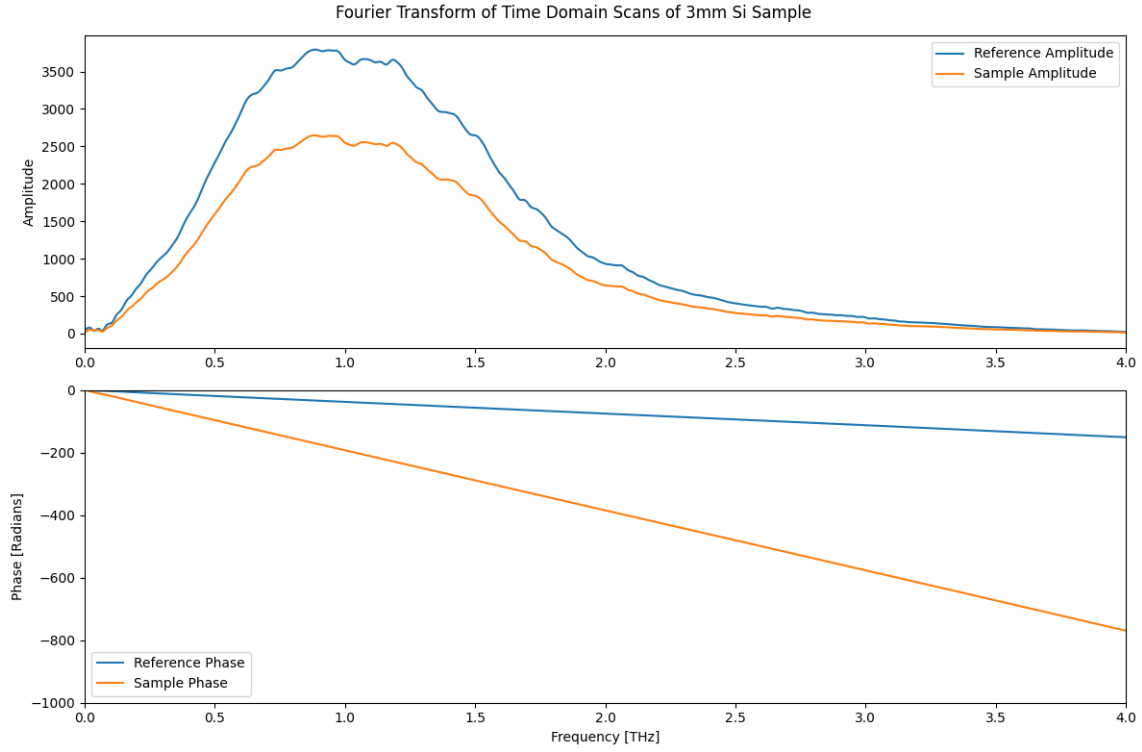
Figure 6: The Fourier transform of the reference and sample pulse for a 3 mm sample of silicon. TOP: Amplitude of the Fourier transform, Shows the distribution of frequencies from the emitter. BOTTOM: Unwrapped phase of the Fourier transform. The increasing phase delay in the sample relative to the reference reflects the additional optical path length due to the refractive index of silicon.

In order to extract the material parameters, a theoretical model that can calculate a transfer function from a set of material parameters is required. Here we will describe a simple model to extract material parameters from the time-domain scans. The Fresnel equations model the propagation of the THz pulse through the sample and air. Figure 7 shows a schematic of the path of a THz pulse with no reflections through a sample.

8

THz beam — Air: $P_1(n = 1, L_1)$ — THz Emitter

L

Transmitted Beam — Air: $P_2(n = 1, L_2)$ — THz Detector

Sample: $P_0(n = n, L)$

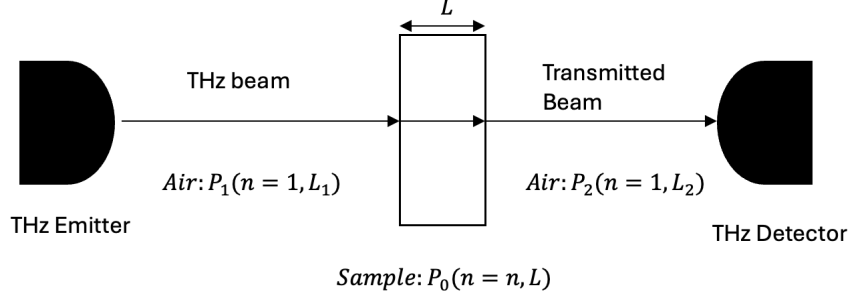The propagation coefficient (P_α) of the electric field through a sample of length L and complex refractive index $n_\alpha(\omega)$, at angular frequency ω, is given by the relationship below [11]:

$$P_\alpha = e^{-in_\alpha(\omega)\omega L/c} \tag{7}$$

Here, i is the imaginary number and c is the speed of light in meters per second. From the propagation coefficients it is possible to get the electric field using (8) and (9).

$$E_{sample} = \eta(\omega) \cdot P_1(\omega, L_1) \cdot T_1 \cdot P_0(\omega, L) \cdot T_2 \cdot P_2(\omega, L_2) \cdot E(\omega) \tag{8}$$

$$E_{reference} = \eta(\omega) \cdot P_1(\omega, L_1) \cdot P_3(\omega, L) \cdot P_2(\omega, L_2) \cdot E(\omega) \tag{9}$$

For the reference pulse, the term with the sample is replaced by air, with refractive index $n = 1$, giving the propagation coefficient P_3. T_1 and T_2 are the transmission coefficients at the border between air and the sample, and the sample and air, respectively. $\eta(\omega)$ is the admittance of the material. Substituting the values for air and our generic sample into the electric field equations, and subsequently into the definition of the transfer function, we obtain (10). Then, by using (7), we derive (11). This will be referred to as the Fresnel model when comparing it with other methods of modelling THz radiation.

$$H(\omega, n(\omega)) = \frac{E_{sample}}{E_{reference}} = \frac{T_1 \cdot P_0(\omega, L) \cdot T_2}{P_3(\omega, L)} \tag{10}$$

$$H(\omega, n(\omega)) = \frac{4n}{(n + 1)^2} e^{-i((n-1)\omega L/c)} \tag{11}$$

In order to find the complex refractive index, the following condition must be met.

$$H_{theoretical}(\omega, n(\omega)) = H_{exp}(\omega) \tag{12}$$

$H_{theoretical}$ is the theoretical transfer function (11) and H_{exp} is the experimentally obtained transfer function. The traditional method for solving this equality is to vary the complex value of $n(\omega)$ in $H_{theoretical}$ until the equality is solved. This is done for all frequencies, giving us a frequency-dependent complex refractive index. The Newton-Raphson method is the traditional (not

machine learning) method that is used to accurately fit the theoretical transfer function to what we have measured experimentally [11]. This is an iterative root finding method following the relation below:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \qquad (13)$$

In this expression, $f(x)$ is some arbitrary function, $f'(x)$ is its derivative with respect to x and x_i approaches the root of $f(x)$. To apply this to our system, we make the following alterations to the above equality and use this iterative method to find the root value of $n(\omega)$. This is repeated for all values of ω. By taking the log of the transfer function, the phase and amplitude are separated.

$$f(n(\omega)) = \ln(H_{theoretical}(\omega, n(\omega))) - \ln(H_{exp}(\omega)) \qquad (14)$$

Figure 8 shows an extraction from a 3 mm thick sample of Si. This was performed using an algorithm implemented in Python3. The results of the extraction can be compared to the literature in [12].
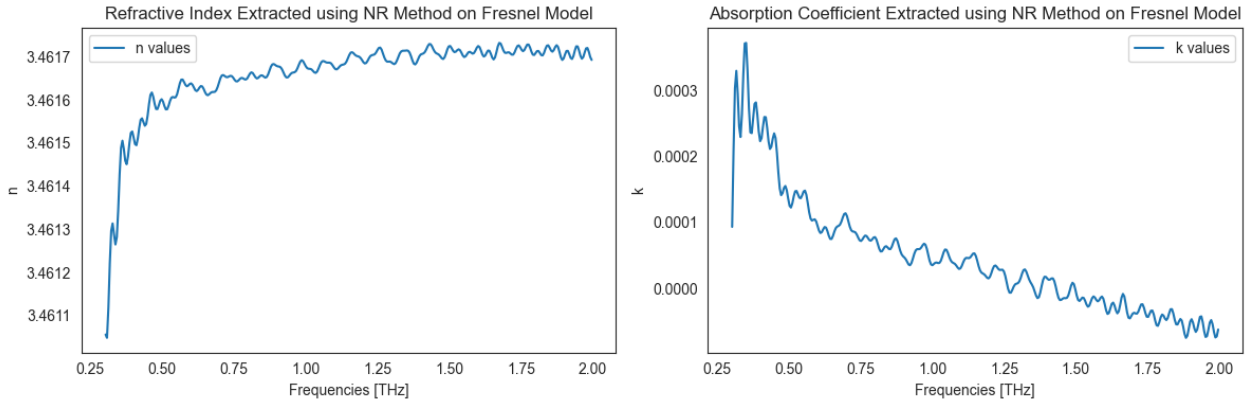
Figure 8: Extraction of index of refraction and absorption coefficient in the range of 0.3 to 3 THz for a 3mm silicon sample using the Newton-Raphson iterative method.

The model chosen to fit a frequency dependent refractive index depends on specific characteristics of the sample, with thickness being the primary focus. Thin samples risk containing echoes caused by the THz being reflected at the internal boundary of the sample as shown in figure 9. Samples with thickness on the order of $10\,mum$ risk containing echoes that overlap with the main pulse in the time domain, these are impossible to remove by windowing. In thick samples, these extra peaks are not detected if the delay line is not long enough to access these regions of the time domain.
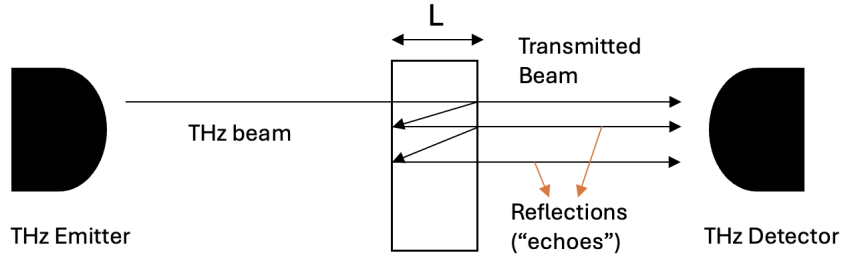
Figure 9: Diagram illustrating the internal reflections of a THz pulse within a sample of thickness L, leading to echoes in the time-domain signal.

These reflections cause an etalon in the sample pulse and subsequently, the transfer function[34]. This oscillation is not accounted for in the single transmission Fresnel model described above. One solution to this is to window the time domain signal such that the first signal is preserved while undesirable echoes are removed. This is only possible if the sample is thick enough and the echoes don't overlap with the main pulse. Figure 10 shows a frequency spectrum for a sample with echoes and the etalon effect can be seen on the sample's spectrum (orange). A Tukey window was then applied to the time-domain signal, centred on the sample signal's peak. Transforming this signal into the frequency domain results in a much smoother spectrum, as shown in Figure 11.

Figure 10: Fourier transform of the reference pulse and the signal transmitted through a 486 μm thick $LiNbO_3$ sample. The spectrum of the transmitted signal exhibits periodic features due to the etalon effect, caused by internal reflections within the sample.

While windowing can be an effective method of pre-processing our time signals, it is possible that echoes or other signals can overlap with the main pulse. In this situation windowing would be ineffective and a new approach must be found to fit this data. This new approach would need to fit the data in such a way that the internal reflections are considered. There are Fresnel based model that account for infinite reflections, however solving these for experimental data using the Newton-Raphson method is not stable and often converges to unphysical solutions. This is because there are multiple solutions.
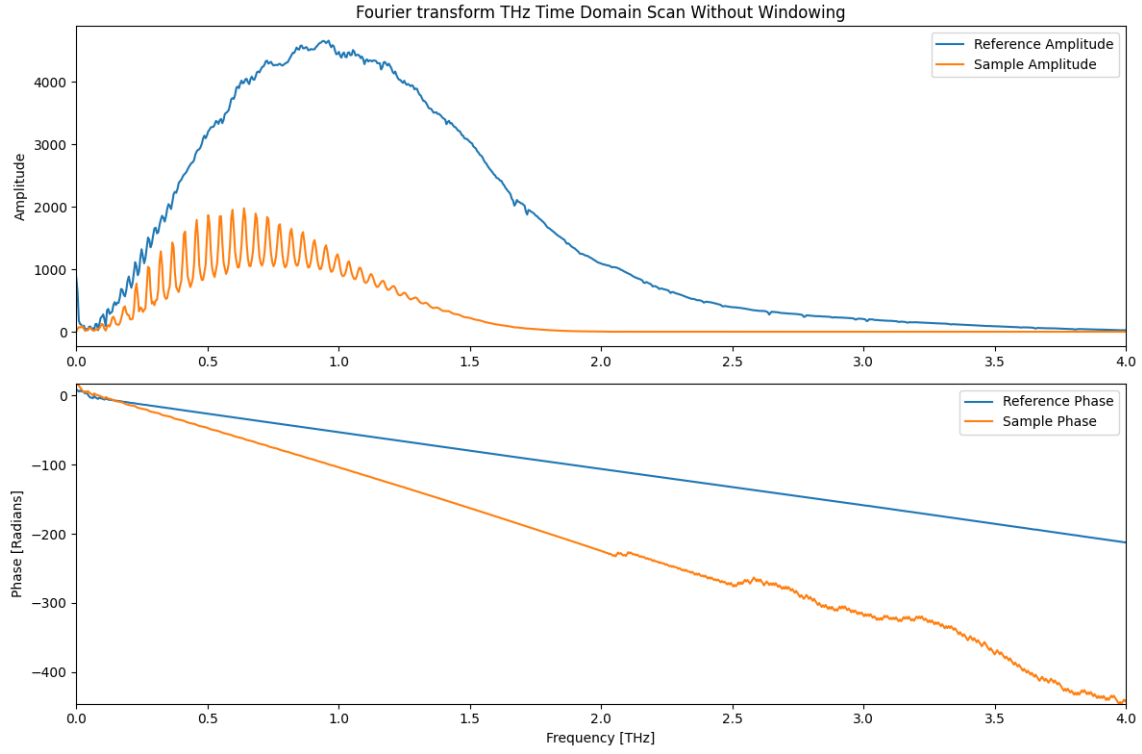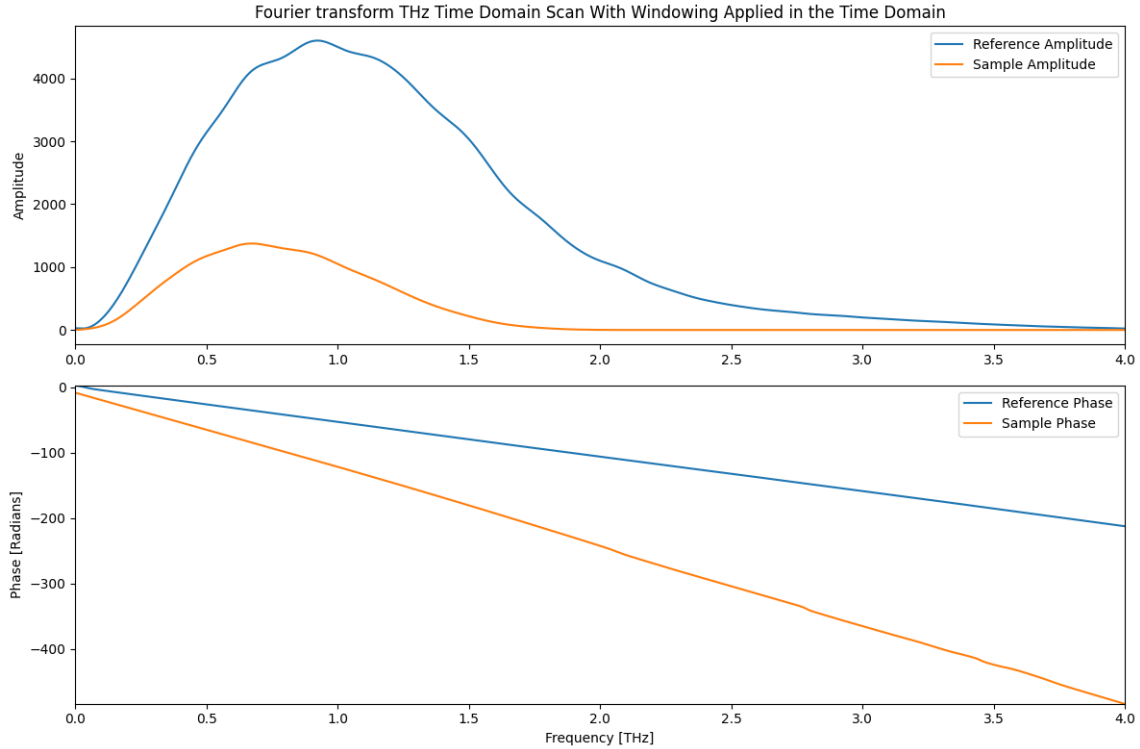
Figure 11: Fourier transform of the reference pulse and the signal transmitted through a 486 μm thick $LiNbO_3$ sample, with the time-domain signal windowed using a Tukey window centred on the pulse peak. The windowing suppresses the etalon effect, resulting in a smooth spectral distribution for the sample pulse.

1.3 Machine Learning Methods

Machine learning is a sub-field of artificial intelligence. The field of AI research has been accelerating recently, leading to many breakthroughs. The release of GPT-3 by OpenAI in 2020 brought the field international acclaim [31]. The 2024 Nobel Prize in Chemistry was jointly awarded to the developers of AlphaFold, a machine learning-based method for predicting protein structures developed at Google DeepMind [36]. DeepMind's work has also extended to physics with the development of FermiNet, an open-source machine learning model that uses neural networks as an ansatz wave function to solve for many-electron systems [42]. This model has recently been used to compute excited states [39] and has applications in computational chemistry, where it provides an 'ab initio' approach to simulations. A key distinction between a traditional algorithm and a machine learning model is that traditional algorithms rely on pre-defined rules, while machine learning relies on patterns in data. This can lead to better generalization and error handling [19].

There are many architectures in the field of machine learning; the most basic are Artificial Neural Networks (ANNs) consisting of layers of nodes (or neurons) connected by a series of weights, biases with a non-linear activation function. Each pair of inputs connected to an output is called a 'perceptron' and a basic ANN is called a multi-layered perceptron [40]. These weights and biases

are optimized during training to be able to replicate the transformation between the input data and the target data. This training process uses stochastic gradient descent to minimize a loss function between the predicted dataset and the target dataset, this is called supervised learning, as the dataset includes labelled target values[37]. This process can be improved with optimizers like Adam (adaptive moment estimation), these use different algorithms to calculate and apply the gradients of loss during the learning process [23].

1.4   Applications of Machine Learning in THz-TDS

The current method of numerical extraction for THz-TDS can be hard to implement and do not always result in stable convergence, especially for signals with echoes or overlapping pulses. It also is not able to easily model multi-layered samples. Machine learning can aspire to try and solve these issues. Fully connected neural networks (FCNNs) or convolutional neural networks (CNNs) can be taught to extract material parameters [24] or other material features. FCNNs can be taught to solve the Fresnel model (or an alternative). CNNs pass a filter over the data[37] allowing them to identify patterns based on the sequencing and structure of data and values. This structural aspect can make them better suited to the problems this field is attempting to solve. However this comes at a cost of computational speed and intuitive understanding of the underlying data transformations. It is also possible that the use of CNNs will allow for extraction or sample identification directly from the time domain. It has been shown that CNNs are capable of identifying non-trivial patterns and differences in structured data. This method can also be used to learn the characteristics of a specific setup and reduce errors introduced by the experimental equipment.

Researchers examined the diffraction patterns of THz waves at varying depths in a THz imaging setup and using a novel physics-informed neural network, they were able to extract the depth of the object to a high degree of accuracy [45]. Just to the human eye, the differences in the diffraction pattern were not noticeable but the model was able to make distinctions between datasets to extract depth.

Currently, the neural networks are being trained on simulated datasets that are created by reversing the transfer function equation (11). The networks were trained for a range of complex refractive indices and a single thickness. Changing this to a different physical model will produce a network that is capable of understanding more complex samples. There are several potential models and methods that are being looked at. CNNs are a good candidate for spectroscopic or time domain data as they take the location of data points in their sequences into account [37][28]. This is optimal for structured data [29] such as time series or diffraction patterns. These can be used for both regression tasks such as parameter extraction or classification tasks such as sample identification. Classification tasks using THz spectroscopic data have been performed in the food industry with a reasonable degree of success [21]. However, this was a supervised approach and may be difficult to replicate unless a large experimental dataset for THz-TDS can be sourced. The principle component analysis that forms part of the process may not perform well on experimental data if the model is trained on simulated data.

A key requirement of machine learning is a large dataset. THz data can be time consuming to measure experimentally and the range of samples cover a limited range of material parameters. Synthetic data is a clear alternative, however this comes with it's own set of biases. A better way to combat the lack of experimental data is to combine synthetic (simulated) data with experimen-

14

tal data [6]. This can be done by simply mixing the datasets together, but a better approach is transfer learning. This involves two training steps; pre-training and fine-tuning. Pre-training is the first step in transfer learning, where the model is trained on a large dataset. This step allows the network to learn general features and underlying patterns relevant to the problem domain, forming a strong foundation of 'knowledge'. This is typically synthetic data. The weights calculated by this model are then used as the initial weights for the fine tuning phase. Fine-tuning uses a smaller, typically experimental dataset, the aim being to teach the network the nuances of the data it will be given to make predictions with. Transfer learning is commonly used in natural language processing, fine-tuning pre-trained models such as GPT-3.5 to perform specific tasks such as scoring [27] and high quality summarization [33]. In our case, pre-training on a large synthetic dataset, covering our desired parameter space, the model can then be fine-tuned by researchers at different labs on their own setups. This will provide a consistent standard for extraction across different institutions.

1.5 Gradient Descent for Solving Inverse Problems

The traditional method of finding the complex refractive index, detailed above, relies on having a transfer function with an analytical derivative that can be calculated at each frequency point [11]. The Newton-Raphson iterative method is then used to find the best value of n and k that reproduces transfer function measured experimentally at that frequency. For samples with large reflections (for example thin samples) or multi-layered samples the theoretical transfer function becomes more complicated as well as it's derivative. This means that Newton's algorithm becomes more sensitive to initial conditions and can struggle to converge on the correct solution. One way to address this is to switch to a fitting method that doesn't use an analytical derivative of the function being fitted.

This work addresses an inverse problem [35] by optimizing a differentiable, physics-based forward model to extract material parameters that best reproduce an experimentally measured transfer function. The approach is similar to training a neural network, but instead of using a network of neurons for the forward pass, a physical model is used. This enables parameter updates to be informed directly by the underlying physics of the system. The current model is defined in Equation 11, with the objective of identifying the parameters $\tilde{n}(\omega)$ that correspond to the observed experimental transfer function. The inverse problem can be approached using a modified objective function and optimization strategy. The objective function numerically describes a goal that the algorithm is trying to achieve, in this case the goal is to minimize the loss and the objective function is the loss function. Objective functions typically link predicted data with the experimental target data. The Adam optimizer [23] is a machine learning optimization technique that can be employed to improve gradient descent. This method iteratively minimizes a loss function by computing its gradient with respect to each of the model's parameters and using that information to refine predictions.

To determine the frequency-dependent tensors n_i and k_i (i is an index representing each frequency in the range being optimized) that best recreate an experimental transfer function $H_{\exp}(\omega)$. Given an initial estimate, n and k are iteratively updated using the Adam optimizer in order to minimize a loss function.

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \left[(\mathrm{Re}(H_{\mathrm{pred},i}) - \mathrm{Re}(H_{\exp,i}))^2 + (\mathrm{Im}(H_{\mathrm{pred},i}) - \mathrm{Im}(H_{\exp,i}))^2 \right] \tag{15}$$

where $H_{\mathrm{pred},i}$ is the predicted transfer function for frequency index i and N is the total number

of frequency points. This is the mean squared error (MSE) for the real and imaginary components of the complex transfer function. The traditional method deals with the unwrapped phase and amplitude of the transfer function. However, the phase unwrapping is not a differentiable process, this means that unwrapping the phase will break the computational graph of gradients [1]. This will prevent the model from learning. The theoretical transfer function is computed as:

$$H_{\text{pred},i} = \frac{4\tilde{n}_i}{(\tilde{n}_i + 1)^2} e^{-\frac{i(\tilde{n}_i - 1)\omega_i d}{c}},$$
(16)

$n_{\text{complex},i} = n_i + ik_i$ is the complex refractive index. The 1:1 nature of this model in terms of frequency (ie. H at each frequency point only depends on n and k at that frequency point) means that the loss function can be expressed as:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_i(n_i, k_i)$$
(17)

This means that the derivatives are localized:

$$\frac{\partial \mathcal{L}}{\partial n_j} = \frac{1}{N} \frac{\partial \mathcal{L}_j}{\partial n_j}, \quad \frac{\partial \mathcal{L}}{\partial k_j} = \frac{1}{N} \frac{\partial \mathcal{L}_j}{\partial k_j}, \quad \forall j \in \{1, \dots, N\}.$$
(18)

Thus, each frequency component is updated independently. The Adam optimizer updates n_i and k_i as follows:

$$n_i^{(t+1)} = n_i^{(t)} - \alpha \frac{\hat{m}_{n_i}^{(t)}}{\sqrt{\hat{v}_{n_i}^{(t)}} + \epsilon},$$
(19)

$$k_i^{(t+1)} = k_i^{(t)} - \alpha \frac{\hat{m}_{k_i}^{(t)}}{\sqrt{\hat{v}_{k_i}^{(t)}} + \epsilon}$$
(20)

Where $\hat{m}_{n_i}^{(t)}$ and $\hat{v}_{n_i}^{(t)}$ are the first and second moment estimates of the gradients, α is the learning rate and t indicates the iteration step. These help stabilize and adapt learning rates while fitting. Specific definitions of the moment estimations and how they are calculated can be found in the paper that first proposed the Adam optimizer [23]. Since the gradients are frequency-independent, Adam applies per-frequency updates correctly without mixing information across different frequencies.

The new fitting method follows the basic logic outlined in the Figure 12. The process works by using the free parameters (n, k and thickness) to create a prediction dataset using a physics based model (forward pass). Loss is then computed between the experimental data and the predicted data. This loss is back-propagated using Torch's autograd engine and used to adjust the material parameters (19) and (20).
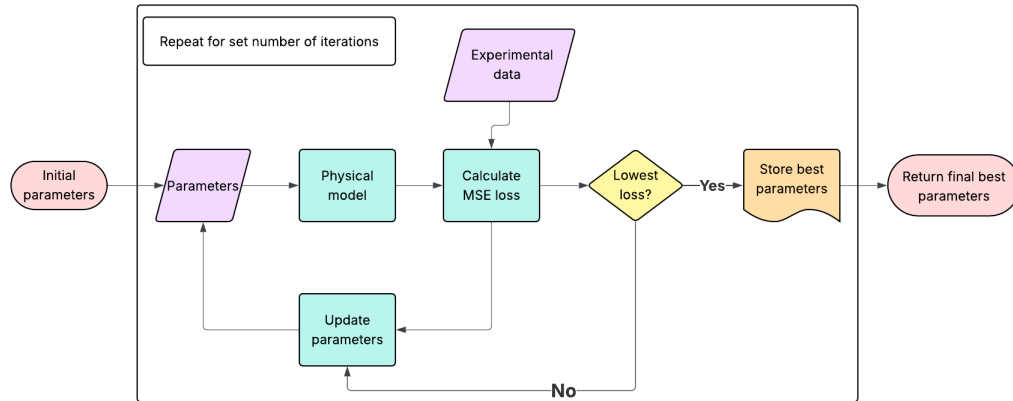
Figure 12: Flow chart for solving an inverse problem with gradient based fitting on experimental data.

This method is not constrained to only the Fresnel model. Any model that can be built using differentiable operators, possible with the PyTorch implementation of most mathematical operations, can be solved with this gradient descent method. In a future section a matrix transfer model is used that will allow for extraction on multi-layered samples directly from the time domain.

2 Gradient Descent on the Fresnel Model

In this section Adam optimized gradient descent will be used to fit the Fresnel model (11) to the experimental data in Figure 4. The results will then be compared with the Newton-Raphson method, both in terms of accuracy and speed. We perform the same processing steps for the time domain as in the traditional method of extraction and Figure 13 shows the amplitude and phase of the experimental transfer function for $0.3\,\mathrm{THz}$ to $2\,\mathrm{THz}$. We will use the loss function above (15) as the function for the optimiser to minimize. This loss function uses real and imaginary components of the transfer function and does not directly look at amplitude and phase of transfer function.
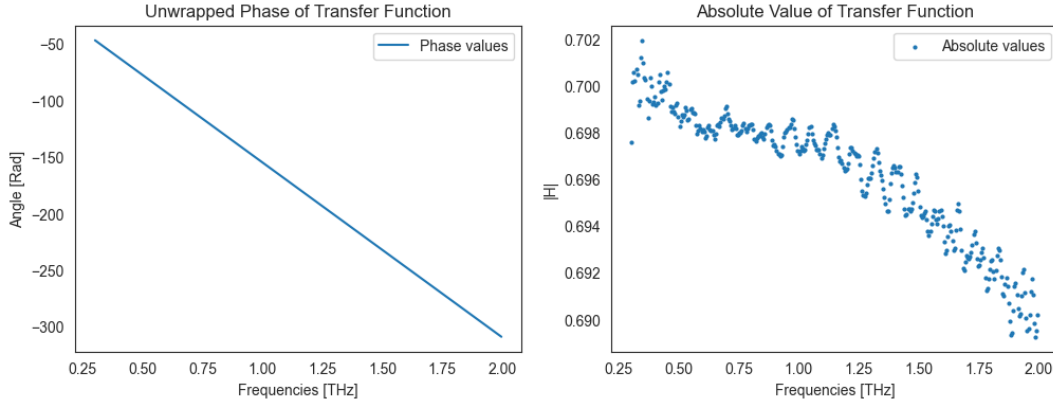

Figure 13: Amplitude and unwrapped phase of the transfer function for a $3\,\mathrm{mm}$ sample of Si between 0.3–$2\,\mathrm{THz}$.

A tensor for the frequency dependent values of n and k is initialized, these contain the initial values for the gradient descent. For this optimization we start at $n = 3.44$ and $k = 0.001$. This value for n is quite close to the known value, however it is on par with what can be determined by measuring the time delay between the two pulses, this value gave a rough value of $n = 3.4632$. There are now two tensors n_j and k_j each value of these corresponds to a frequency point (frequency values are stored in the tensor ω_j) we have in our experimental. These components can be combined into a tensor $\tilde{n}_j = n_j + ik_j$. The forward pass for this process is the Fresnel model.

$$H_{pred,j} = \frac{4\tilde{n}_j}{(\tilde{n}_j + 1)^2} e^{-i((\tilde{n}_j - 1)\omega_j L/c)} \tag{21}$$

This is implemented using PyTorch tensors and in the optimizer code, the parameters are defined such that Torch can automatically track the gradients of the loss function with respect to the parameters [1] [2]. MSE loss between the real and imaginary components of the transfer functions is used as a loss function. This returns a single value for the entire frequency range. Importantly, because the gradient of loss w.r.t. each component of the tensors n_j and k_j is what is actually used to make the updates to each component. The updates will be for each frequency component of the parameter tensor. This is outlined in more detail in the introduction to gradient descent above. The algorithm is run for 250 iterations. Below the loss and log of loss values are plotted, the plot of the log(loss) is to ensure the optimizer has reached a true minima.
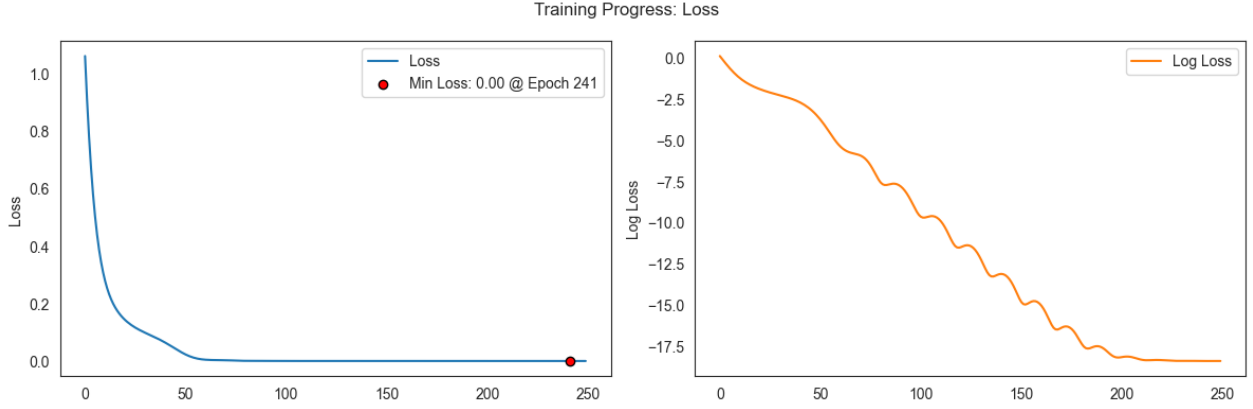
Figure 14: Loss and log(Loss) for Adam optimized gradient descent fitting of $n(\omega)$ and $k(\omega)$ for a 3 mm sample of Si. Log of loss allows us to clearly visualize the convergence behaviour over several orders of magnitude and to identify when the optimization reaches a plateau, indicating minimal further improvement.

This model reached a best MSE loss of 1.823×10^{-10}. The resulting material parameters can be seen in Figure 15.
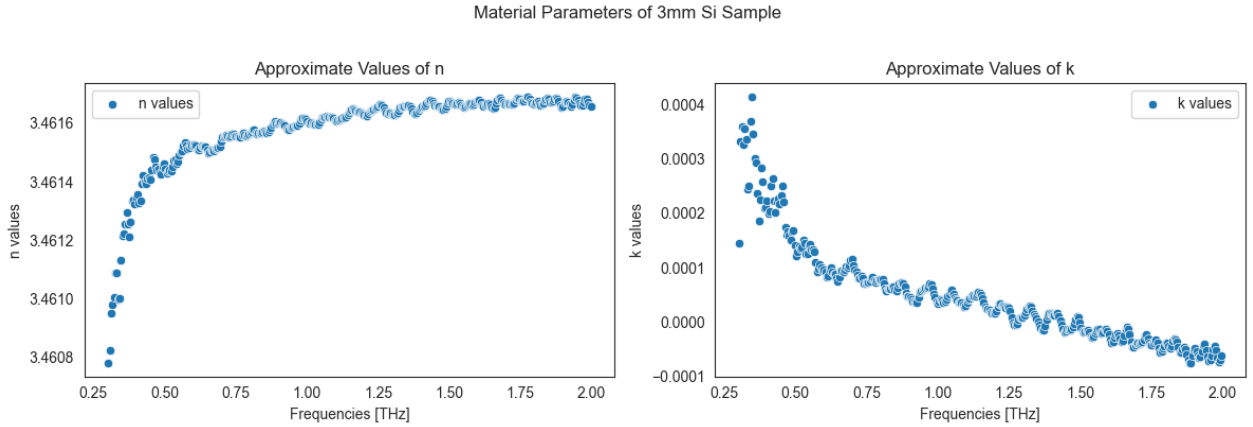
Figure 15: The frequency dependent material parameters $n(\omega)$ and $k(\omega)$ for a 3 mm sample of Si. These were found by solving the single transmission Fresnel model using the Adam optimized gradient descent on tensors n_j and k_j where these run for all values of frequency.

The optimized material parameters can be used to reconstruct the transfer function. This can be compared to the experimental data to cross reference the results. This can be seen in Figure 16.
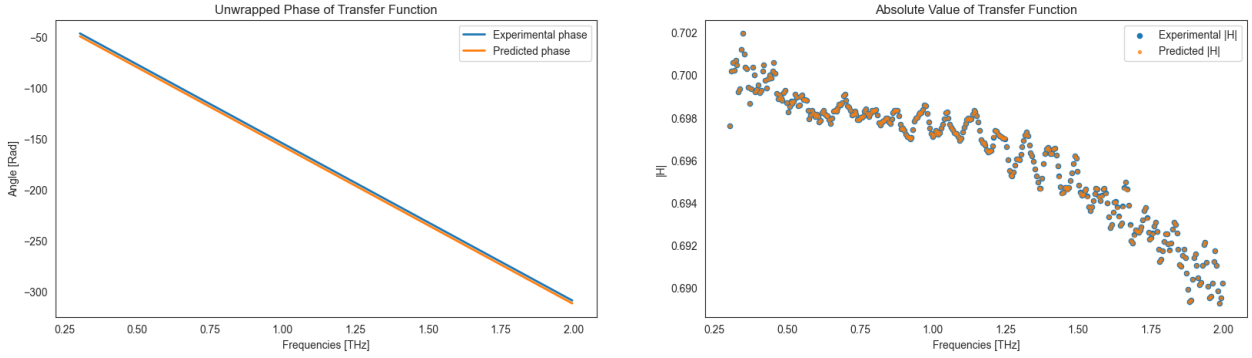
Figure 16: Amplitude and phase of experimental transfer function (blue) and a reconstruction (orange) using the results from the Adam optimized gradient descent based model. The reconstruction shows a high degree of accuracy in the solution.

This takes 0.0582 seconds, a large improvement over the 4.921 seconds taken by the traditional NR method. This shows that we have massively improved the performance of the fitting while reserving a high degree of accuracy. We can see this below where we have plotted the two sets of results and the residuals.

Figure 17: A comparison of the solutions from Newton-Raphson model (blue) and the gradient based model (orange) for a 3 mm sample of Si.

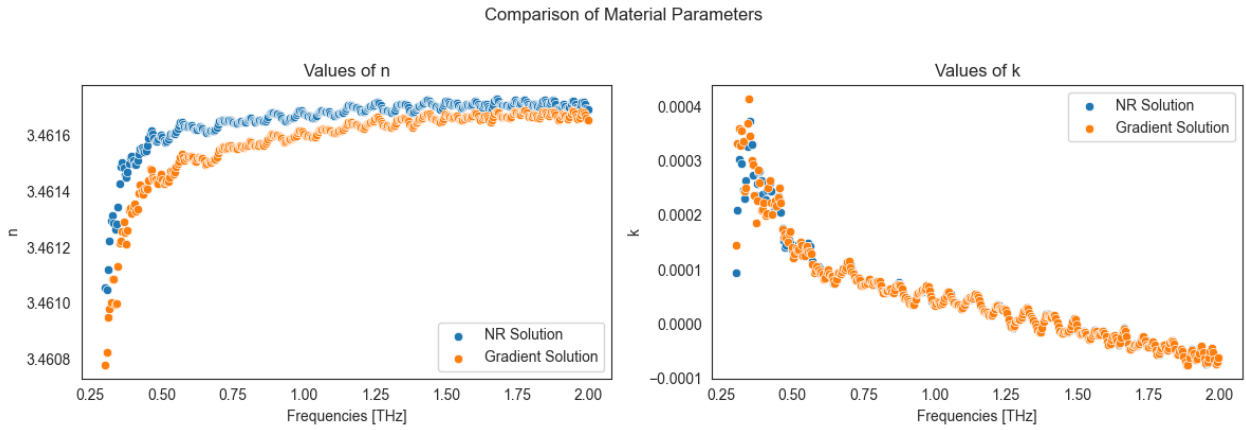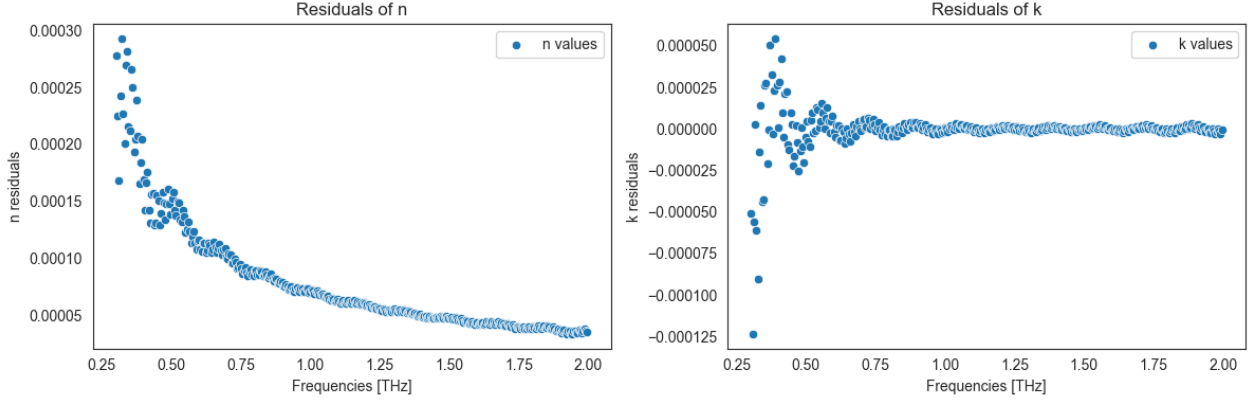Figure 18: The residuals between Newton-Raphson model and the gradient based model results for a 3 mm sample of Si. The residuals show a high degree of agreement between the models, this suggests that further improvement to the material parameters for this setup would need to be from a change in the underlying model used.

3   Time Domain Extractions

The methodology described in Figure 12 is not constrained to simple models like the Fresnel model we have shown already. Any process that comprises of purely differentiable operators can be used as the forward pass in our optimization system. Utilising an existing Python2 codebase designed to perform matrix transformations on time domain pulses for multi-layered samples. On inspection, the method was seen to be built upon entirely differentiable functions. Using this as the forward pass in the inverse problem framework described above results in a novel optimization method capable of extracting material parameters from the time domain. As we are extracting from the time domain without windowing our reflections and extra pulses, it will also be possible to extract the thickness of samples from the data, given some initial guess. This will be tested later in this section.

3.1   Introduction to Matrix Methods for THz

This approach is based on transfer matrix theory [47]. At each boundary the THz pulse is partially transmitted and reflected [48] while accumulating phase. These effects can be combined to get an overall transmission (the sample pulse in the time domain). By enforcing the continuity boundary conditions at the interfaces of the materials, the reflection and transmission coefficients are calculated for each layer. Let's consider a pulse moving from a medium n_0 to n_j where $n_j = n_j' + ik_j$ with the second medium having a thickness of d_j. The optical path length is $D_j = n_j \cdot d_j$. The reflection and transmission coefficient can then be calculated [47].

$$r = \frac{i}{2} \frac{\frac{n_0}{n_j} - \frac{n_j}{n_0}}{cos(n_j D_j) + \frac{i}{2}(\frac{n_0}{n_j} + \frac{n_j}{n_0})sin(n_j D_j)} \tag{22}$$

$$t = \frac{1}{cos(n_j D_j) + \frac{i}{2}(\frac{n_0}{n_j} + \frac{n_j}{n_0})sin(n_j D_j)} \tag{23}$$

21

This is performed for each layer and then recursively combined these for all layers in the sample with a reference pulse. This gives an overall T and R which are the total reflection and transmission coefficient for the stack. Phase is encoded in T meaning that attenuation and dispersion effects on the propagating pulse are accounted for.

Once T is calculated, it can be used on a provided reference pulse $x(t)$. This is converted to the frequency domain with an FFT (fast Fourier transform), $X(\omega)$. The frequency domain response is calculated using the transmission coefficient as the transfer function.

$$Y(\omega) = T(\omega) \cdot X(\omega) \tag{24}$$

An IFFT (inverse fast Fourier transform) is used to convert this back to the time domain $y(t)$. Figure 19 shows a completely simulated system with 3 layers n = 2,3,2 and d = 1, 0.1, 1 mm. The sample signal has had artificial noise with a maximum value of 0.002 added to it.
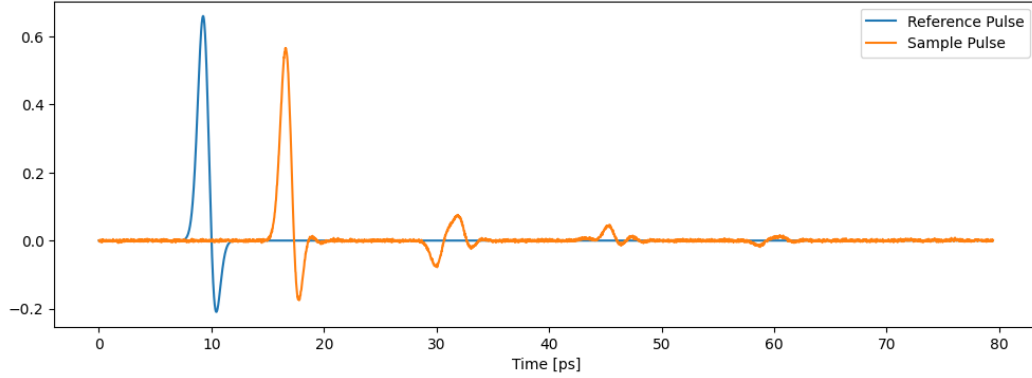

Figure 19: Time domain of simulated reference and sample pulse for a 3 layered material with material parameters: n = 2,3,2 and d = 1,0.1,1 mm respectively. These were simulated using a matrix transfer methods.

A key benefit of this is that imperfections in the reference pulse are considered when making the forward propagation. Consider the reference pulse below, the imperfections around the main pulse can be seen. Unlike the Fresnel model, this model is able to make sense of these extra features, as a result, windowing them out as part of any signal pre-processing is not required. However, in its current implementation, only a single value for refractive index can be retrieved. This value is effectively an average over the frequency range of the emitter. This has little impact for samples where the refractive index is relatively constant over the frequency domain of our emitter. The computation largely takes place in the frequency domain, this means that this process can become frequency dependent if given more development time. This will allow for spectroscopic analysis of multi-layered samples using this method.
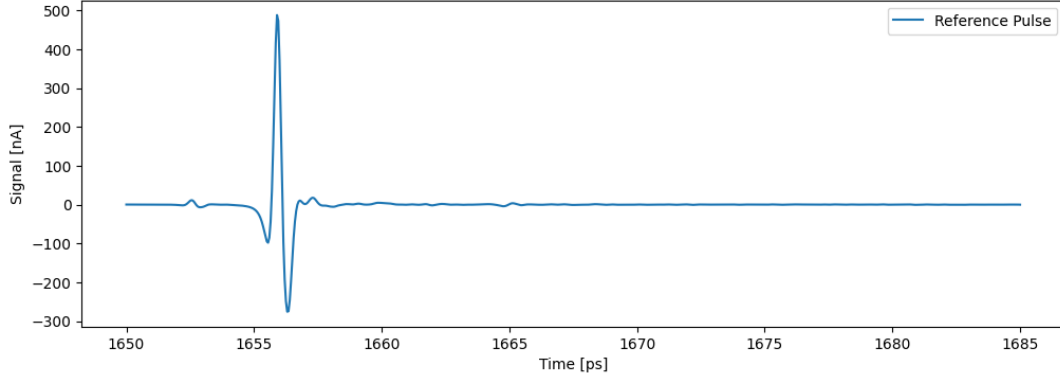
Figure 20: Experimental reference from Si experimental data. We can see some imperfections around the main pulse.
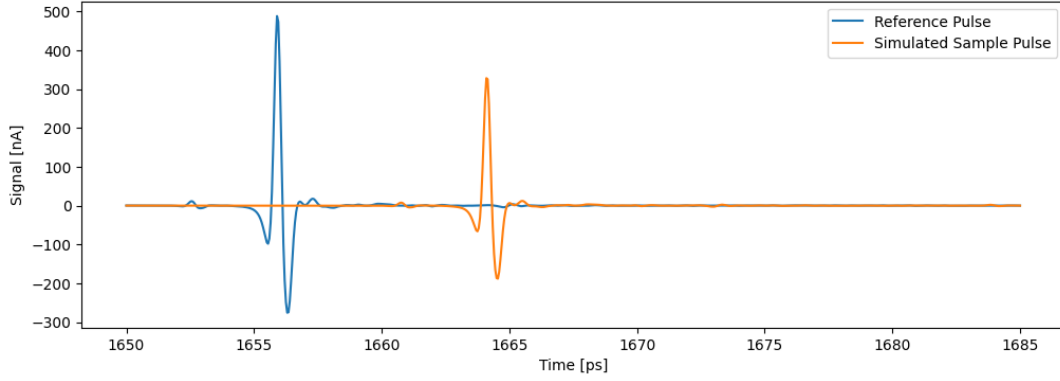
Figure 21: Simulated sample pulse with material parameters: $n = 3.46 - 0.001i$ and $d = 1\,mm$. We can see features from the reference being carried over to the simulated pulse.

3.2 Time Domain Extraction on Single layered samples

Combining the matrix transfer model with the gradient based optimization, the material parameters can be extracted for a sample based on a loss function comparing the predicted time domain with the one measured experimentally. The main benefit of this method is that we can stably extract material parameters for multi-layered samples and for emitters with imperfections (such as reflections) without needing to remove information by windowing or altering the model we are using in the forward pass.

3.2.1 Single Layer Synthetic Pulse

This method will first be tested using completely synthetic data, this is data where the reference and sample pulse have been simulated. The simulated reference pulse is created with the equation:

$$x(t) = A \cdot T \cdot \exp\left(-\left(\frac{T}{t_{width}}\right)^2 - \frac{T}{\tau}\right) \tag{25}$$

where $T = t - t_{offset}$ is to centre the pulse at some delay time, A is a scaling factor, τ is a decay rate and t_{width} is a factor that gives a width to the pulse. This is the detected pulse that as been through the equivalent length of air as our sample.

A time domain pulse corresponding to a $1\,\mathrm{mm}$ sample with material parameters of $n = 3.46$ and $k = -0.001$ is simulated using our forward pass.
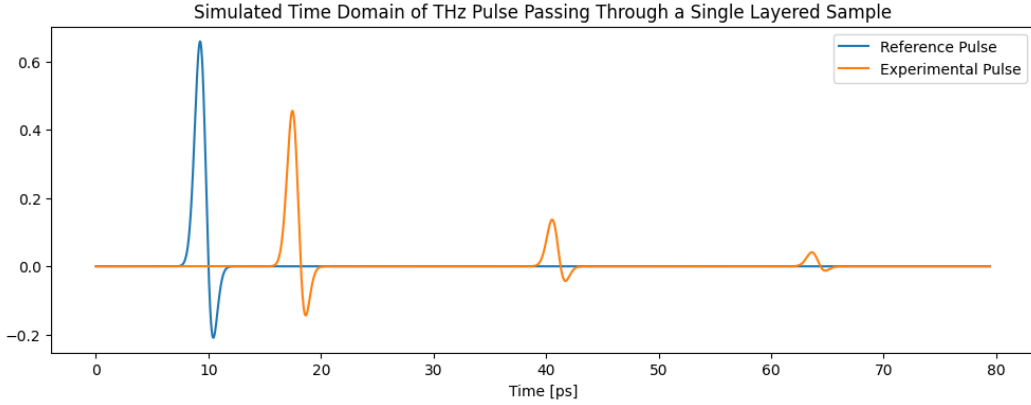
Figure 22: Simulated time domain of a simulated THz pulse (blue line) through single, $1\,\mathrm{mm}$, layer of a material with $\tilde{n} = 3.46 - 0.001i$ (orange).

The gradient fitting method can now be used to attempt to find values of n and k that reconstruct the sample pulse (orange). Starting at initial values $n = 3.4$ and $k = -0.0015$ the model is able to reduce loss to 7.98×10^{-7} this is the mean squared error (MSE) between the signals, as this is simulated data, the loss has arbitrary units. However for experimental data this will be a measurement of the amplitude of electric field associated with the THz pulse. The model found values of $n = 3.459996$ and $k = -0.000995329$. The residual for n is 3.7766×10^{-6} and for k the residual is 4.6714×10^{-6}. Looking at the reconstructed time domain pulse, using the forward pass of the model, it is possible to visually compare the pulses to see if the entire time domain is being reconstructed correctly.

Figure 23: Time domain reconstruction of using the optimized parameters from the Bayesian and gradient methods. The reconstruction (green dashed line) shows a high level of agreement with the simulated experimental data (orange line).
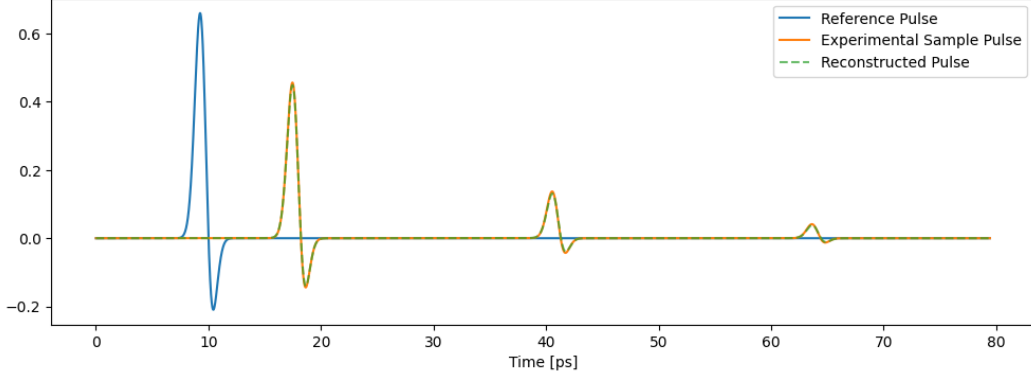
The accuracy of this method is heavily influenced by the value of thickness used during optimization. If a value of $1.01\,\text{mm}$ is used for thickness rather than the correct value of $1\,\text{mm}$, the model converges to $n = 3.4322$ and $k = -0.001220$. This means it is important to have a high tolerance measurement of the sample. There is also the assumption that the sample has a constant thickness across the area that the THz pulse is incident.

3.2.2   Single Layer Experimental Pulse

To test this method on experimental data, an extraction on the silicon pulses in Figure 4 was performed. The results in Figure 17 show that the refractive index does fluctuate greatly in the frequency domain of the emitter. This means the average value we will extract using this time domain method is expected to be accurate. Starting at an initial value of $n = 3.43$ and $k = 0$ and passing the thickness of $3\,\text{mm}$ to the model. The model converged to a value of $n = 3.46019$ and $k = 4.46781 \times 10^{-5}$. These values are consistent with the results in Figure 17. The reconstruction from these parameters is shown in Figure 24.
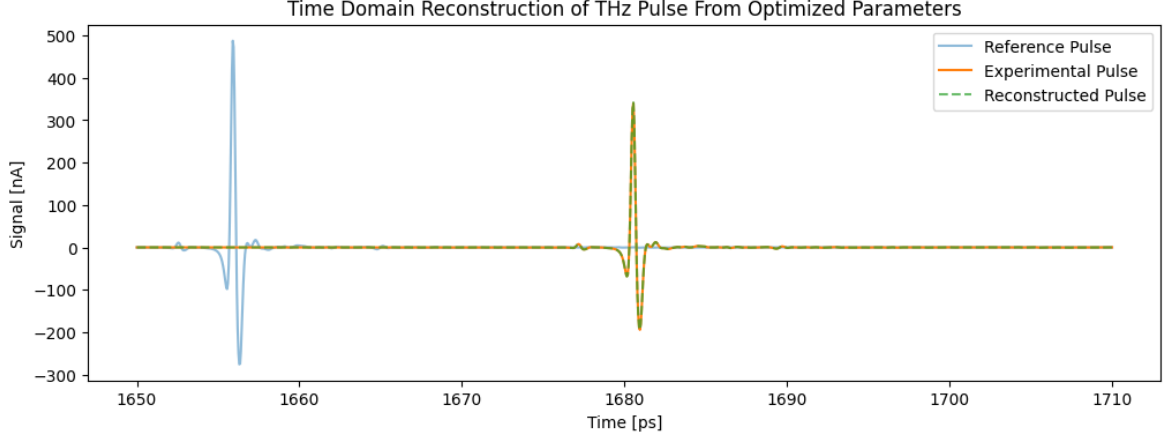
Figure 24: Time domain reconstruction of using the optimized parameters from the Bayesian and gradient methods. The reconstruction (green dashed line) shows a high level of agreement with the experimental data simulated using an experimental reference pulse (orange line).

3.3 Bayesian Optimization Prior to Gradient Descent

A key improvement of this model over previous approaches is the ability to extract material parameters from a composite sample composed of multiple layers. As the number of parameters increases, the time required for optimization also grows. This is largely due to the cost of computing gradients during back-propagation. Benchmarking revealed that approximately 60% of the total fitting time is spent calculating gradients. Therefore, to improve efficiency, a non-gradient-based method is used to generate good initial guesses for the parameters before switching to the gradient-based optimizer for fine-tuning.

Given the relatively low dimensionality of the problem (up to 9 parameters for a 3-layer system), Bayesian optimization with a Gaussian process (GP) surrogate model [18] is well-suited for this task. The goal is to minimize an objective function in this case, the root mean square error (RMSE) between the experimental and simulated time-domain pulses:

$$\theta_{\text{optimized}} = \arg\min_{\theta}(\mathcal{L}(\theta)) \tag{26}$$

$$\theta \equiv [n_i, k_i, d_i], \text{ where } i \text{ indexes the layers} \tag{27}$$

Here, θ is a vector of the material parameters: the real part of the refractive index n_i, the imaginary part k_i, and the layer thickness d_i for each layer i.

To enable efficient search, Bayesian optimization uses a Gaussian process as a probabilistic surrogate for the true loss function:

$$\mathcal{L}(\theta) \approx GP(\mu(\theta), k(\theta, \theta')) \tag{28}$$

Here, $\mu(\theta)$ is the GP's mean function, the expected value of the loss given the parameters θ. The kernel function $k(\theta, \theta')$ defines the covariance [14] between two sets of parameters θ and θ', encoding

how similar the model expects their losses to be.

The process begins by evaluating the loss at an initial set of randomly sampled points θ_j within the bounded parameter space. These evaluations update the GP model, refining its mean and uncertainty estimates. Based on the updated GP, an acquisition function typically Expected Improvement (EI) [4] is used to decide where to sample next. The acquisition function balances exploration (sampling uncertain regions) and exploitation (sampling where the predicted loss is low). The next parameter set θ is chosen to maximize the expected improvement, and the cycle repeats for a set number of iterations. The user can reconstruct the time domain associated with these parameters to determine if further fitting is required.

This method does not reach the same levels of precision and accuracy as the gradient based method. The experimental data for the 3mm Si was fit using only the Bayesian optimization process. This searched the parameter space $n \in [3.33, 3.53]$ and $k \in [-0.01, 0.01]$, resulting in $n = 3.46003$ and $k = 0.00012662$. These visually reconstruct the data well, as shown in Figure 25, but this method does not outperform the gradient based model in terms of accuracy or precision.
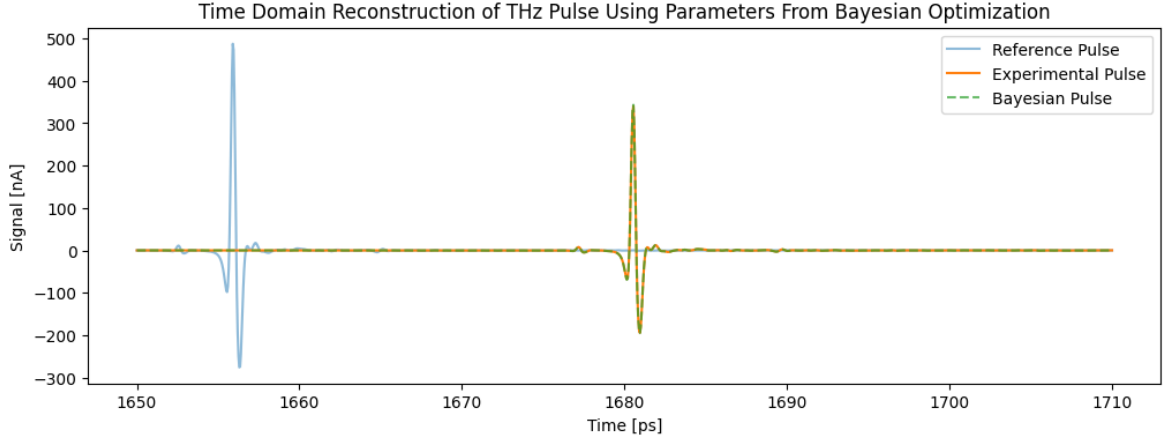
Figure 25: Reconstruction of the experimental sample pulse, and the optimized reconstruction after fitting material parameters using only the Bayesian optimization method. This is for a 3mm sample of Si.

3.4 Time Domain Extraction on Multi-layered Samples

In this section, the Bayesian optimization and gradient model will be used together to extract the material parameters of multi-layered samples. In multi-layered samples, it is also possible to extract the thickness of each layer. Typically this will require the entire time domain to be used. This is not always possible experimentally, as the length of the time domain is set by the length of the delay line in the setup.

3.4.1 Two Layered Synthetic Pulse

A material with two layers was simulated with composition: $n_1 = 3.46$, $k_1 = 0.01$, $d_1 = 0.5\,mm$, $n_2 = 2.5$, $k_2 = 0.015$ and $d_2 = 0.25\,mm$. Simulating this with the reference pulse used for Figure 22

gives the time domain below:

Figure 26: Simulated time domain of a two layered sample with material parameters: $n_1 = 3.46$, $k_1 = 0.01$, $d_1 = 0.5\,mm$, $n_2 = 2.5$, $k_2 = 0.015$ and $d_2 = 0.25\,mm$ using the matrix transfer methods.
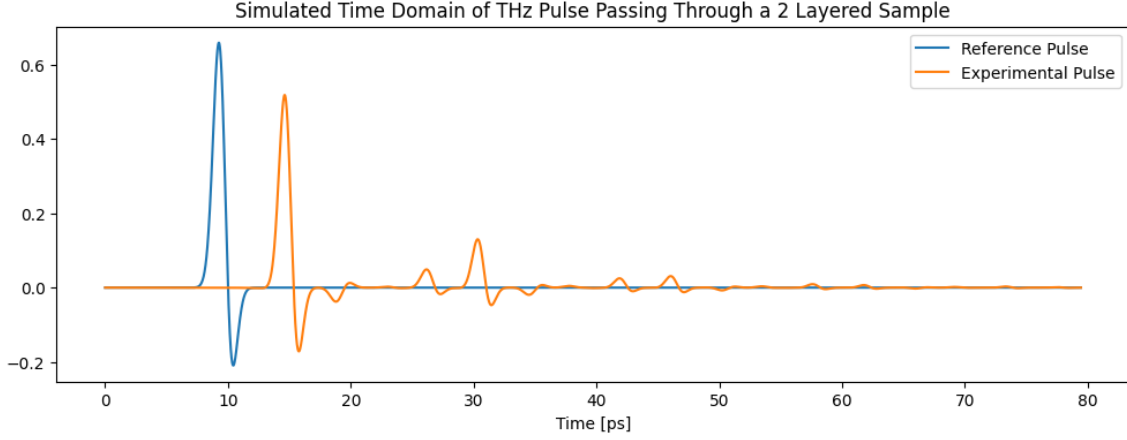
The Bayesian optimizer is then initialized to fit for all materials parameters (n_i, k_i and d_i) with initial values: $n_1 = 3.44$, $k_1 = 0.015$, $d_1 = 0.6\,mm$, $n_2 = 2.45$, $k_2 = 0.017$ and $d_2 = 0.3\,mm$. The results from this are then fed directly into the gradient based optimizer. After 100 iterations with the gradient model, a loss of 1.579×10^{-4} is reached. The optimized material parameters: $n_1 = 3.45578$, $k_1 = 0.013247$, $d_1 = 0.50061\,mm$, $n_2 = 2.50900$, $k_2 = 0.015392$ and $d_2 = 0.24896\,mm$. This shows excellent agreement with the simulation parameters, highlighted by the residuals between our results: n_1: 0.004218, k_1: -0.003247, d_1: $-6.162 \times 10^{-4}\,mm$, n_2: -0.009004, k_2: -0.0003917, d_2: $1.039 \times 10^{-3}\,mm$.
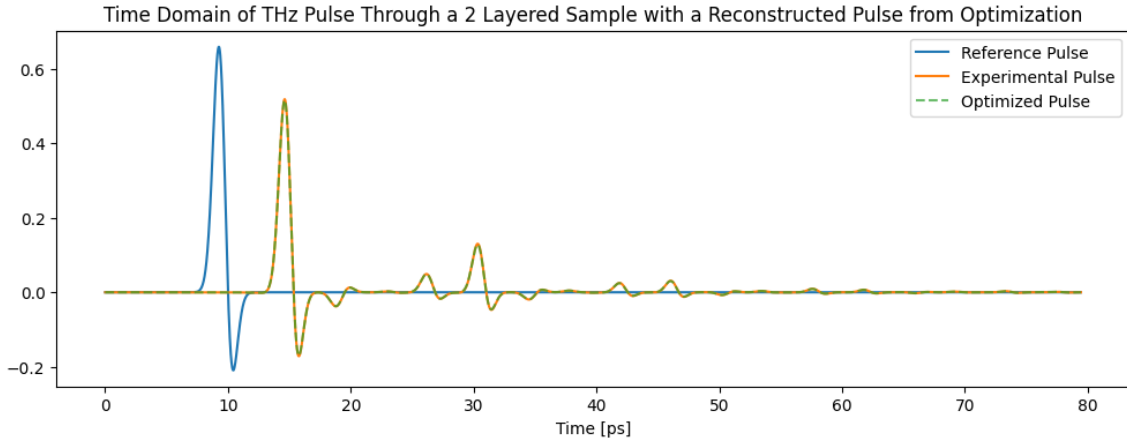

Figure 27: Time domain reconstruction of THz pulse passing through a 2 layered sample. The reconstruction was created using optimized parameters from a physics based gradient descent model.

When the models target only n_j and k_j, the best loss value improves to 4.906×10^{-5} and the residuals are n_1: 7.722×10^{-4}, k_1: 0.001202, n_2: 7.255×10^{-4}, k_2: 0.001228. The thickness provided to the model can have a large impact on the accuracy of the result, in cases where the thickness is not precisely known or there are uncertainties in the value of thickness for layers, the ability to optimize the thickness while keeping the value constrained within some confidence interval can help the optimizers converge efficiently.

3.4.2 Three layered Experimental pulse

A 3 layered material composed of a 1080 μm slab of quartz, a 490 μm slab of GaAs and a 1040 μm slab of quartz was made and tested using a THz spectrometer with a 50 ps delay line. The time domain is plotted in Figure 28. The thickness of the individual slabs were measured with a micrometer with an uncertainty of $\pm 10 \mu m$. Scans of each slab were taken as well as the composite material. Extractions on each component as will be used to validate the results of the extraction of the composite material. The large area GaAs PCA used as an emitter [26] to take this scan has echoes caused by internal reflections of the THz pulse. These are the secondary and tertiary peaks in the time reference pulse's time domain. While these are accounted for in the forward pass, it is possible that they can mask features, resulting in reduced accuracy in the final result. The forward pass does not account for any DC bias in the output signal, to ensure this does not give an unwanted contribution to the loss the average of sample pulse before the signal starts is taken and subtracted from both pulses.
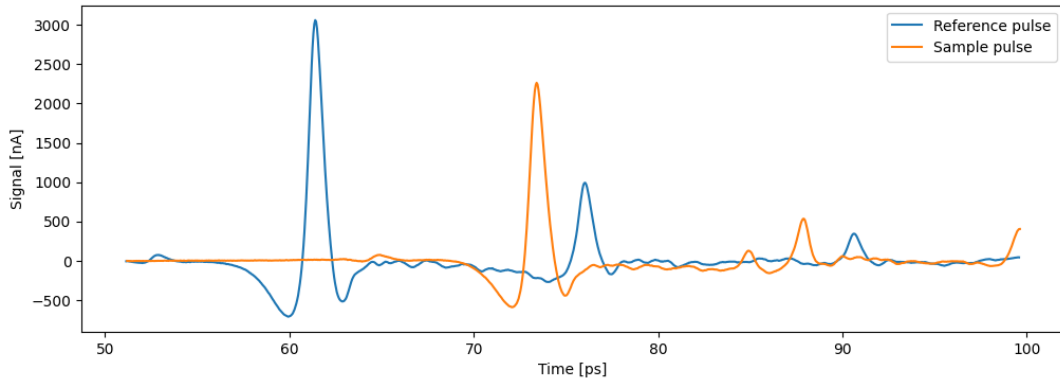
Figure 28: 50 ps THz-TDS reference pulse and Scan of a 3 layer Quartz-GaAs-Quartz sample. The GaAs PCA used as the THz emitter has internal reflections causing secondary and tertiary peaks in the reference pulse.

Finding an analytical approximation to serve as an initial value for each of the parameters is not a simple task. To find an appropriate initial guess, the results from Sushko et al. [43] for Quartz and Yahya et al. [46] for GaAs were used. These results were also used to validate the extractions. Figure 29 shows the variation of real refractive index for quartz and Figure 30 shows the real refractive index of GaAs.
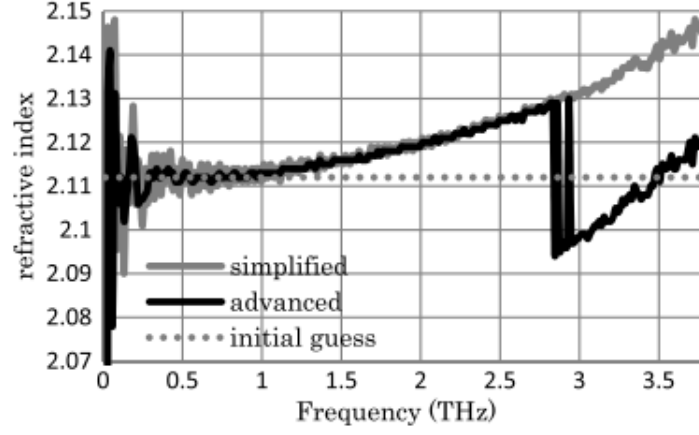
Figure 29: Frequency dependent real refractive index (n) for Quartz from Sushko et al. [43]
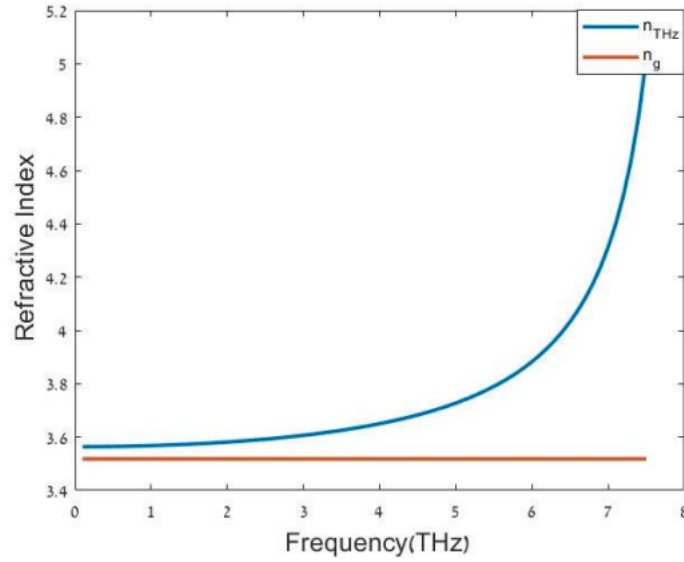
Figure 30: Frequency dependent real refractive index (n) for GaAs from Yahya et al. [46]

These results are for a frequency dependent refractive index, however this method finds an average value. To appropriately decide the correct value of n to use for each of these, the peak frequency of the emitter is used. This is achieved by looking at the Fourier Transform of the reference pulse. To remove the etalon caused by the echoes, the secondary and tertiary peaks are windowed out using a Tukey window centred on the primary pulse.
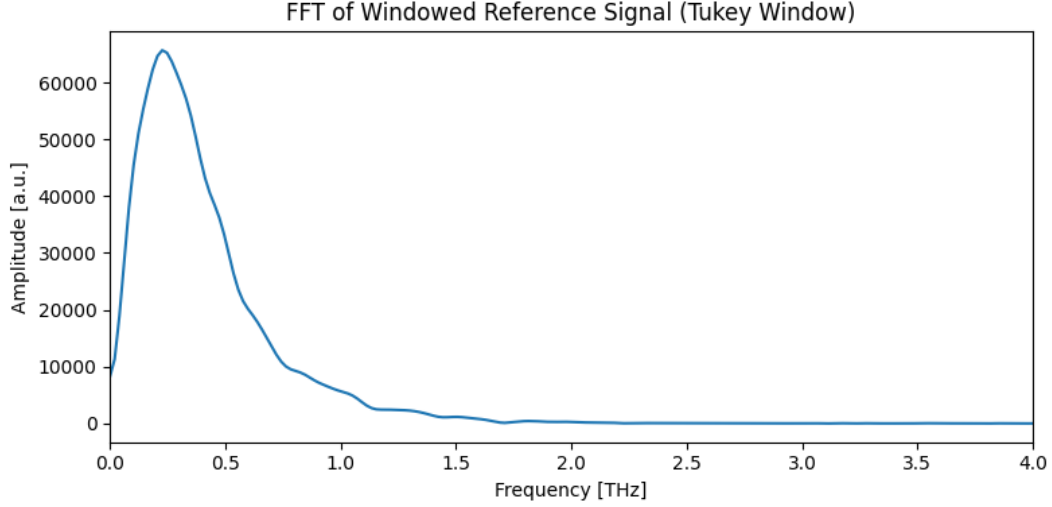
Figure 31: Fourier Transform of Experimental reference pulse. A Tukey window was applied to the pulse in the time domain to remove an etalon effect caused by secondary and tertiary peaks from internal emitter reflections.

The reference pulse has a peak frequency of $0.25\,\mathrm{THz}$, This corresponds to a refractive index of 2.11 for quartz and just below 3.6 for GaAs. As these are simply informing the initial values, a high degree of precision is not required here. To ensure the method is able to converge from further initial values, the following parameters were passed to the Bayesian optimizer: $n_{quartz} = 2.05$, $k_{quartz} = 0$, $n_{GaAs} = 3.4$ and $k_{GaAs} = 0$.

The Bayesian optimization searched the parameter space $n_j \pm 0.1$ and $k_j \pm 0.01$ with the thickness of each material passed in as a set parameter. The gradient optimization then ran for 200 iterations, resulting in the following reconstruction and parameters:

Layer	Thickness	n	k
Quartz	$1080\,\mu m$	2.09939	-0.03454
GaAs	$490\,\mu m$	3.50147	0.004689
Quartz	$1040\,\mu m$	2.13129	0.024459

These results are then used in the forward pass to reconstruct their associate time domain.
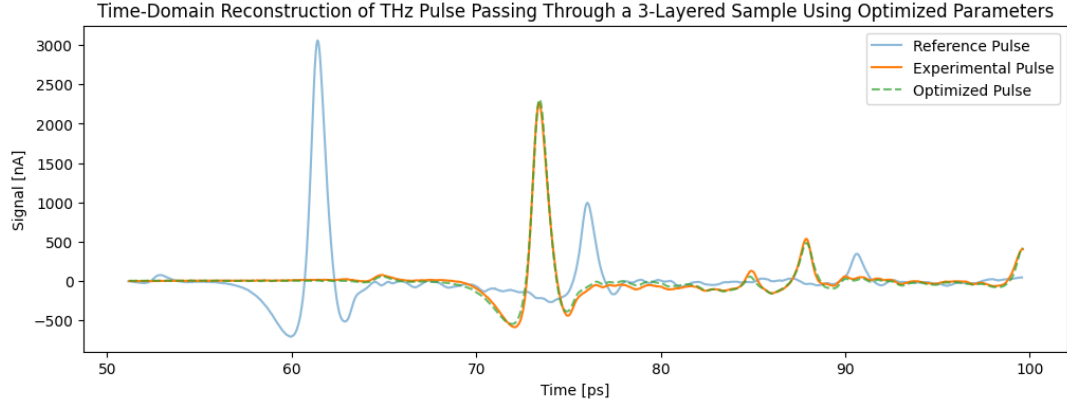
Figure 32: Time domain reconstruction using the optimized parameters from the gradient model (green dashed). This is laid over the experimental data (orange) to validate the optimized parameters from the gradient based fine tuning.

Individual scans of the GaAs and the $1040\,\mu m$ quartz were taken and used to extract the material parameters using the same model. These values can then be used to validate the results of the multi-layered extraction. It would be expected for these results to match as the materials do not change when stacked together. Figure 34 and Figure 33 show the reconstructed time domains of the single sample scans. The parameter optimized to produce these are in the table below.

Layer	Thickness	n	k
GaAs	$490\,\mu m$	3.58596	-0.011203
Quartz	$1040\,\mu m$	2.12676	-0.013734
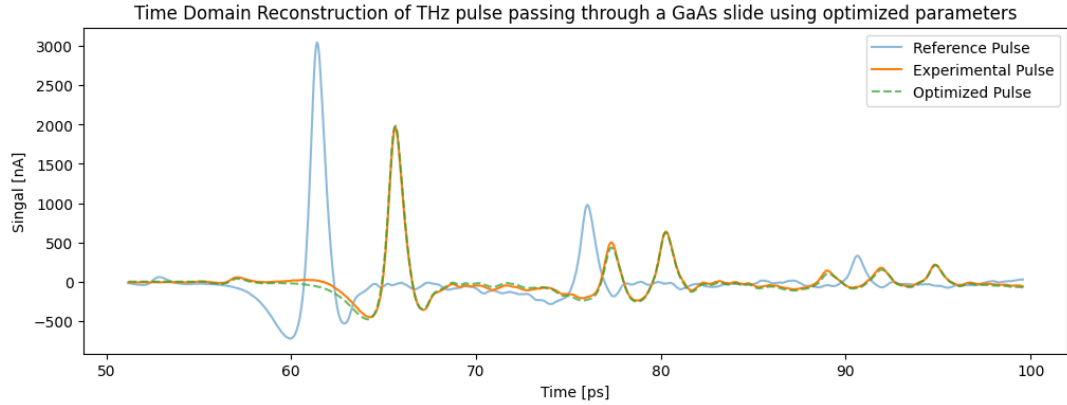
Figure 33: Reconstruction of the time domain for a single $490\,\mu m$ GaAs sample using optimized parameters from the Bayesian optimization and fine tuned with gradient descent.
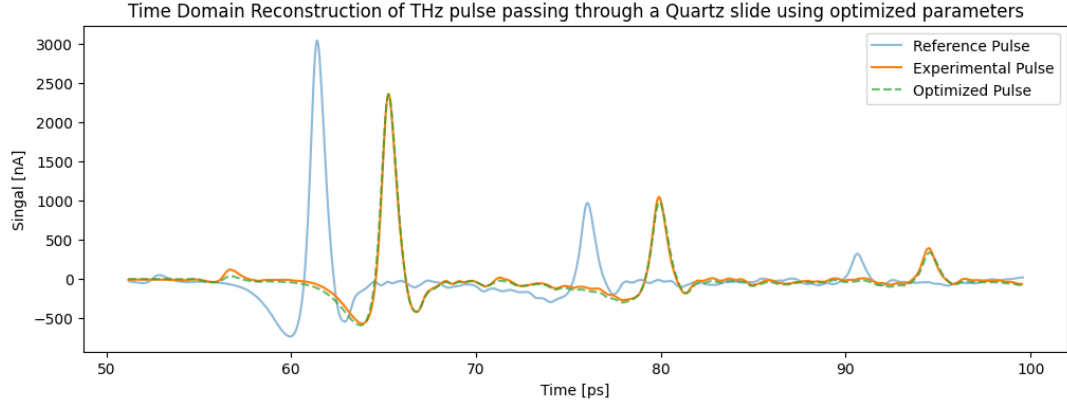
Figure 34: Reconstruction of the time domain for a single $1040\,\mu m$ quartz sample using optimized parameters from the Bayesian optimization and fine tuned with gradient descent.

It can be seen that the reconstruction in Figure 32 does not perfectly match the experimental data in all places (loss at optimal parameters: $30.22\,\mathrm{nA}$). However the gradient optimization had already plateaued after this number of iterations (Figure 35). The errors between the parameters found in the 3 layered scan and the single scans are most likely caused by the short delay line in this specific setup. There are features outside of the range of the time axis that can be used during the optimization process. The more features there are in the time domain, the more potential contributors to loss. This provides better clarity to the model and will allow for more accurate results.
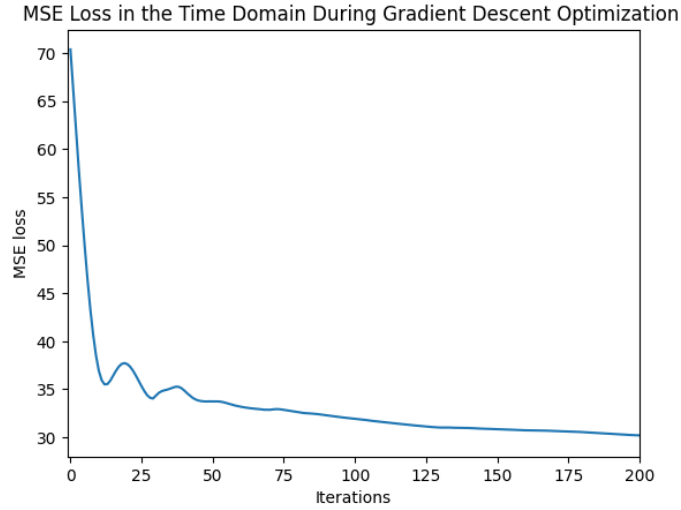
Figure 35: Loss from gradient based fine tuning of material parameters for a 3 layered composite sample.

3.5 Error Sources in Time Domain Extraction

In time domain extraction there are three main sources of error. Two of these relate to the forward pass while the final one relates to the experimental setup. Passing an incorrect value of thickness to the model introduces a systematic error that mostly impacts the final value of n (real refractive index). This makes sense as the largest source of loss will be errors in the primary pulses (as these have the largest magnitude). Passing an incorrect thickness will force the model to find a value of n that corresponds to the observed delay. This can be addressed by allowing the model to alter the value of thickness within provided boundaries. This increases the complexity of the optimization task and can lead to reduced precision and increased optimization times. As a result, it is recommended to use a micrometer or alternative method to measure samples before taking a scan.

The next source of errors come from the FFT portion of the forward pass model. As the samples get thicker and the refractive indexes increase, more wavelengths fit within the sample. This causes more rapid oscillations in the transfer function. If the sample rate in the frequency domain is not also increased, a beating effect can be observed in the predicted time domain. This is not a physical feature of the experimental data, this will lead to additional and unphysical loss contributions that will impact the optimization process. The solution to this is to increase the sampling rate before the FFT is computed. While this does work, it increases the computational time required during the forward pass of the model, increasing optimization time.

The final source of error comes from the length of the delay line. As samples get thicker, more layers are introduced or the values of n are increased, the time delay of the THz pulse also increases. This means not all features are always captured in the experimental data. This reduction in features introduces ambiguity into the model as the full time domain of the sample pulse is not able to be used, the loss contributions from these areas cannot be used in back-propagation, reducing the accuracy of the optimization process. There are no algorithmic solutions to this, increasing the length of the delay line is the only way to include more information in the experimental data.

4 Time Domain Extraction for Liquid Crystals

Being able to model multi-layered media means it is possible to extract the material parameters for samples that are held within some container like glass slides. A good example are liquid crystal cells. These are an active area of research within the THz community [7] [5] [8]. Liquid crystals (LCs) exhibit birefringence [5] this is where the refractive index differs along different spacial axes. This means that for thin samples, time domain scans will exhibit minor differences when held in different orientations. Solving for these thin samples with low refractive index variation will be a good test of how the loss in the time domain translates to accuracy in the optimized parameters.

4.1 Simulated Liquid Crystals

The Bayesian and gradient optimizers will first be tested using simulated data. The table below shows the material parameters for all of the components. Note: Each scan is 3 layers, quartz-LC-quartz. This is done twice, once with each orientation of the LC.

Layer	Thickness	n	k
Quartz	$1000\,\mu m$	1.963	-6.207×10^{-5}
LC (Horizontal)	$20\,\mu m$	1.5	-1×10^{-4}
LC (Vertical)	$20\,\mu m$	1.9	-1×10^{-4}
Quartz	$1000\,\mu m$	1.963	-6.207×10^{-5}
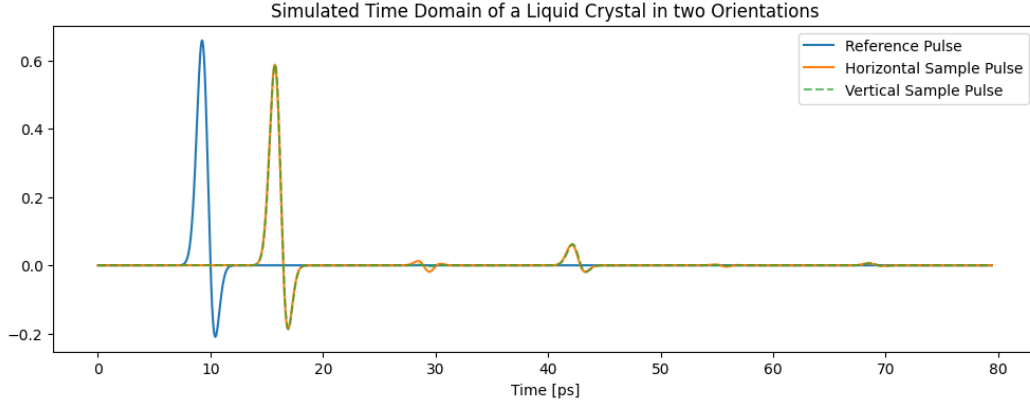
Figure 36: Simulated time domain of a liquid crystal in horizontal and vertical orientations. The birefringent nature of the sample causes differences in the time domain of the two pulses.

Looking at the simulated time domain, there are a few minor differences between the two datasets. In order to extract the LC parameters as efficiently as possible, we will pass the material parameters to the optimizers as set parameters. This means we will not be optimizing the parameters of the quartz slides. The Initial values are found using the time delay approximation (1), in this case the time contributions from the quartz slides are subtracted from Δt_{peak}. This gave approximate values of $n_{horizontal} = 1.53$ and $n_{vertical} = 1.83$. Each optimizer was run for 30 iterations after the Bayesian optimization process. For the horizontal orientation the results were: $n = 1.50016$ and $k = -0.000156$ the residuals are 0.00016 for n and 5.56×10^{-5} for k. For the vertical orientation, $n = 1.90016$ and $k = -0.000269$. The residuals are 0.00016 for n and 0.000169 for k. The large discrepancy in the k values is likely caused by how thick the sample is, it is hard to accurately measure attenuation through such a thin sample when the attenuation is so low.

Figure 37: Reconstruction of the time domain pulses for the horizontal (top) and vertical (bottom) orientated simulated liquid crystals with a thickness of 20 μm. The reconstruction used the optimized parameters after an initial Bayesian optimization and then fine-tuning using 100 iterations of a gradient descent optimizer.

These results show that the time domain extraction can be used to identify differences in real refractive index of weakly birefringent materials while they are in a quartz holder.

4.2   Experimental Liquid Crystals

To further test the utility of these optimization methods on weakly birefringent samples, experimental data on 1929 liquid crystal cells was taken. The cell is 50 μm thick and is held between two 1040 μm quartz slides. These slides were measured to have material parameters of $n = 1.99$ and $k = 1 \times 10^{-5}$. The scans were taken using the same setup as the 3 layered sample above. This means we have a 50 ps delay line with echoes in the reference from the emitter.
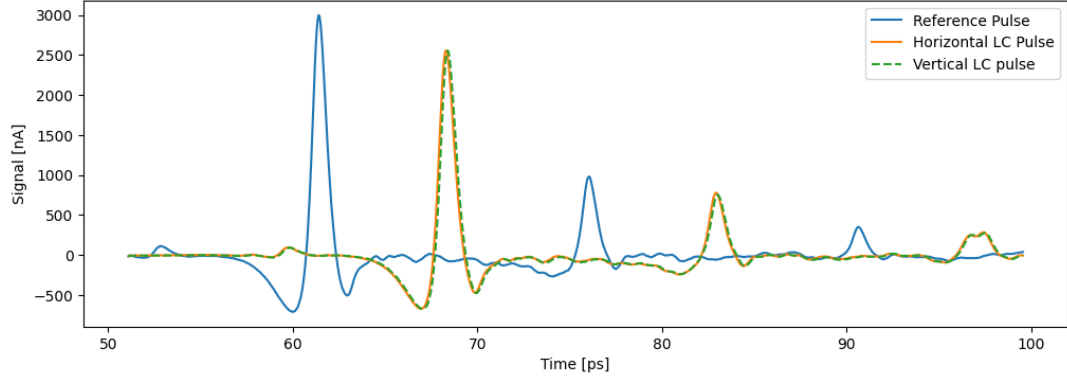
Figure 38: Time domain of a reference pulse and a $50\,\mu m$ 1929 liquid crystal cell. The cell is rotated 90° between scans to investigate the refractive index of the two axis of the samples. The samples are visibly similar meaning differenced in n will be hard to extract.

As with the 3 layered sample, there is a bias of -134 nA is removed from the experimental data prior to optimization. Results from Bennis et al. [9] are used to set the initial conditions and help to cross reference the results. These readings were taken for the visible light range (400-1600 nm). They found $\Delta n = 0.3375$ at 636 nm and 23°. Individual values of refractive index $n_e \approx 1.9$ and $n_o \approx 1.5$. These values are taken for 1600 nm while the THz wavelengths are on the order of $300\,\mu m$ so it is not expected to match, these are simply providing an initial condition for the optimizer.

After the Bayesian model runs to perform the first level of optimization, the gradient models are then used for a total of 300 iterations each. The results are held in the table below and The time domain reconstruction using these parameters can be seen in Figure 39.

Layer	Thickness	n	k
Quartz	$1040\,\mu m$	1.99	-1×10^{-5}
LC (Horizontal)	$50\,\mu m$	1.33636	0
LC (Vertical)	$50\,\mu m$	1.70682	0
Quartz	$1040\,\mu m$	1.99	-1×10^{-5}

37
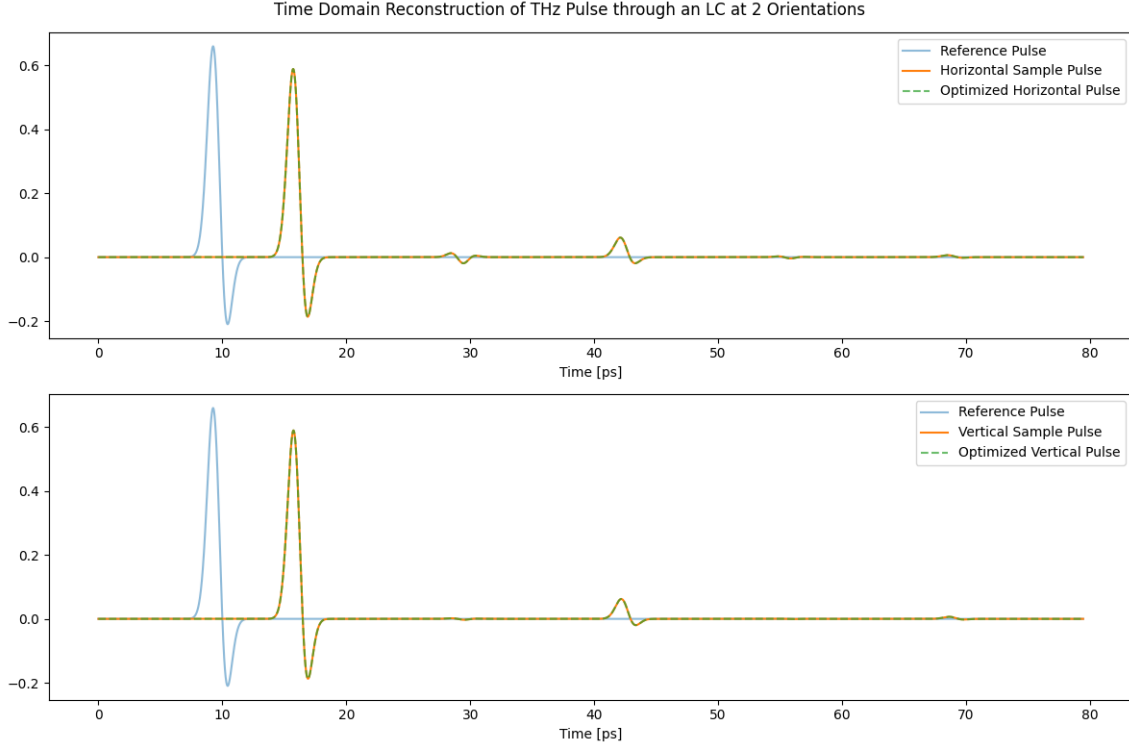
Figure 39: Reconstruction of the time domain pulses for the horizontal (top) and vertical (bottom) orientated 1929 liquid crystals. The reconstruction used the optimized parameters after an initial Bayesian optimization and then fine-tuning using 300 iterations of a gradient descent optimizer. Errors in the time domain may have been propagated by inaccurate values for the quartz or a lack of features due to a short delay line.
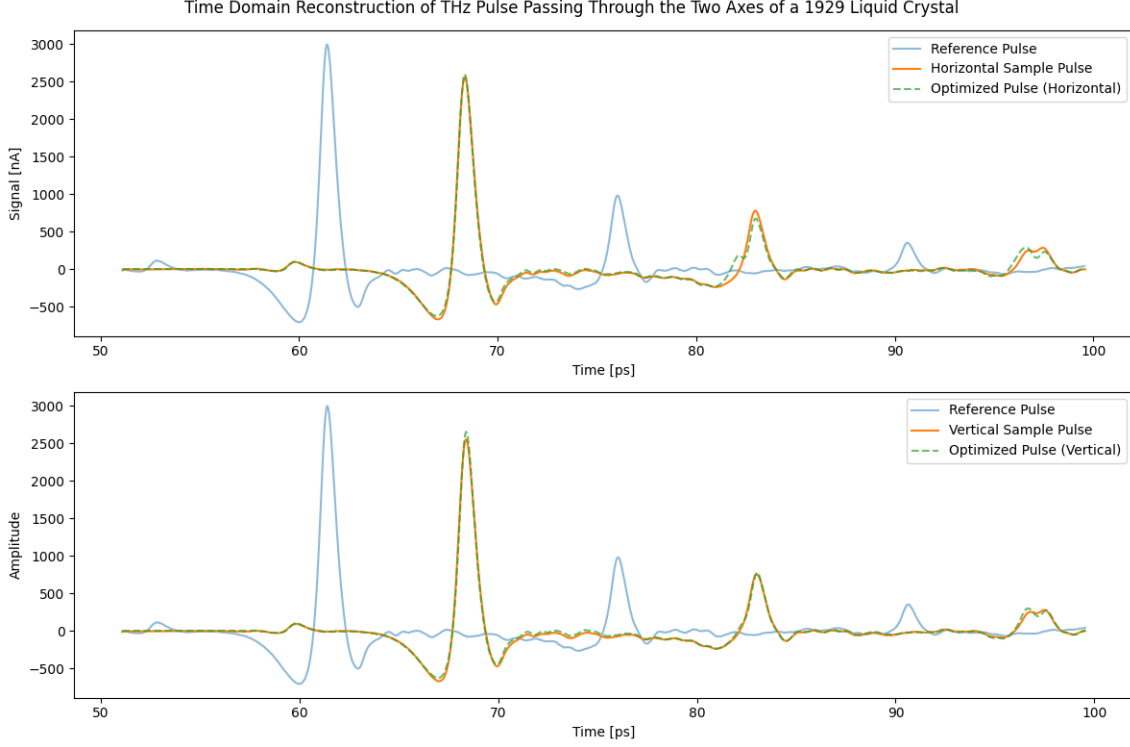
The extracted parameters are for the THz domain, this means they cannot be directly compared to those from Bennis et al. As those were found for the visible region. The mean birefringence (Δn) was found to be $\Delta n = 0.37046$ for the range of THz frequencies in this emitter (Figure 31).

The time domain reconstructed from the optimal parameters (Figure 39) does not match exactly with the experimental scan. Specific errors can be seen in the secondary pulse, where the reconstructed wave is starting to split into two peaks while the experimental pulse does not, this is significant as it may be a showing a feature that is being masked by the extra emitter pulses. Also the reconstructed tertiary pulse has a more pronounced central dip than the experimental does. These discrepancies are indicative of errors in the parameters being used for the forward pass. This is not necessarily the LC parameters. It is possible that the values of n, k and d for the quartz contain errors. This error could also be due to a lack of features caused by a short delay line (50 ps). The gradient optimizer reached a plateau for loss for both orientations meaning that from the information it was provided it was not able to reduce the loss any further. These samples are too thin to be able to accurately extract the extinction coefficient, errors in the extinction coefficient can propagate to the real refractive index, for this reason the extinction coefficient was assumed to be 0 and fixed during optimization. The code for the optimization is setup to allow for an optimization mask to be applied to the layers. This means that any combination of the material parameters (n,

38

k and d) for any of the layers can be fixed or allowed to be optimized.

5   Machine Learning for Layer Identification

Machine learning can be used to learn how features of the time domain or frequency domain relate to material characteristics of samples. This can provide a new method to extract material parameters for different types of samples. The matrix transfer methods used as the forward pass in the time domain gradient descent optimization process can be used to generate large datasets of time domain pulses for a variety of multi-layered materials. These datasets can then be used to train neural networks to complete a range of tasks. Here, a 1 dimensional CNN is trained on 60,000 synthetic time domain pulses, all with varying material parameters (n, k and d) as well as the number of layers in the samples varying. Simulated pulses within the parameter space described in the table were used in the training set.

Parameter	Lower Bound	Upper Bound
Layers	1	3
n	1.1	6
k	-0.1	0.01
Thickness (individual layer)	$50\,\mu m$	$500\,\mu m$

No additional information was passed to the passed to the CNN alongside the sample's time domain pulse.

The network's goal is to identify the number of layers in a sample purely from the time domain's pattern. This is accomplished with supervised training, using cross entropy loss as the loss function that needs to be minimized [3]. The pulses were all generated from a single reference pulse, to increase the generalizability it could be possible to construct a 2 dimensional CNN, passing both the reference and the sample pulse to the model. This would aim to use the relation between the reference and the sample pulse to identify layers or sample characteristics such as the material parameters. Alternatively, a set of reference pulses can be taken on an experimental setup and used to create the training set. This would provide a large performance increase when attempting to use the trained model on experimental data. The drawback being that different users would need to train models for different setups.

Given an unknown sample, the ability to identify the number of layers would comprise the first step of material characterization, the results from this could then be used alongside other methods to prepare initializations for the optimizers.

Before attempting to fit the data using a CNN, it is important to understand the underlying structure. Dimensionality reduction in the form of UMAP (Uniform Manifold Approximation and Projection) is a non-linear dimensionality reduction technique that is well suited to high dimensional data such a time series [38]. UMAP provides a 2 dimensional, interpretable visualisation of the global relationships within high dimensional datasets. Each time domain pulse was reduced from 1024 (number of sample points in the simulated scan) down into 2 dimensions and plotted with a colour map relating to the target classification.
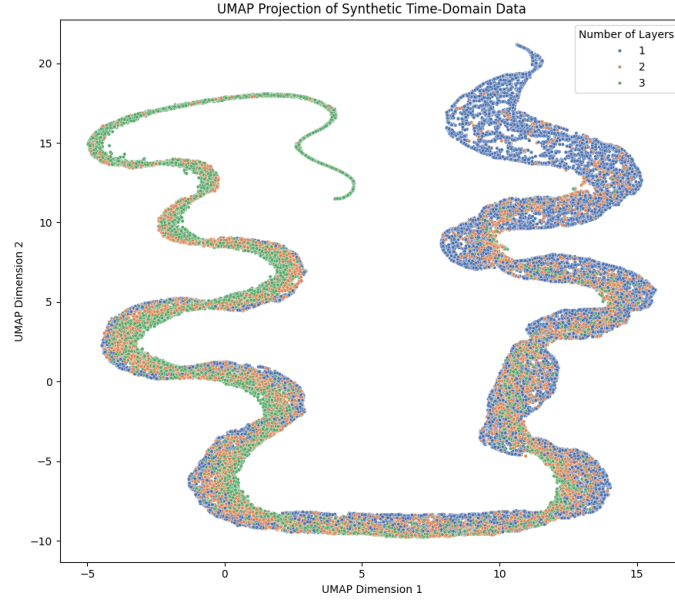
Figure 40: 2 dimensional projection of 60,000 time domain scans using UMAP. The colour map shows the target classification of each pulse. This method used unsupervised UMAP meaning the algorithm was not given information about the target or the material parameters used for each simulated scan.

The resulting plot shows a significant overlap between samples with one layers and two layers. Given the wide range of material parameters used this overlap makes sense. There are regions where the layers are clustered more tightly, these are located at the tails of the the manifold shape in Figure 40. This suggests that a CNN will be able to perform identification to some degree of success without needing additional information.

The CNN used takes single channel of 1024 values (time points in the scan), the data is passed through 3 convolutional layers with increasing channel depth ($32 \rightarrow 64 \rightarrow 128$ filters) with a kernel size of 5 and ReLu activations. Each of the first two layers are followed by a max pooling operation, this reduced the resolution in the time domain to gradually extract more abstract features. Global average pooling is then applied, this flattens the features in to a 1 dimensional vector that can be passed to the classification head. The classification head is a 2 layer fully connected network that reduce down to the 3 classes. A 30% dropout layer is added between these to prevent over-fitting and increasing generalisability of the classification head.

The model was trained for 30 epochs with a learning rate of 0.001 and a batch size of 256. The samples were shuffled each epoch. The accuracy of the model is calculated by taking the number of matching values (correctly predicted) and dividing this by the total number of samples. The model reached an accuracy of 0.8125 on the training dataset. A confusion matrix was plotted to highlight where the model is going wrong.

Figure 41: Confusion Matrix showing predicted and true classes of pulses classified using the trained CNN model. The dataset for this matrix is the same as the one used for training.

The trained model is then validated on 200 unseen images, generated using parameters randomly sampled from the defined parameter space. On the unseen validation set the model achieved an accuracy of 0.805. This shows it is able to maintain accuracy on data that it has not been directly trained on, a key requirement for any model that needs to be used on experimental data.
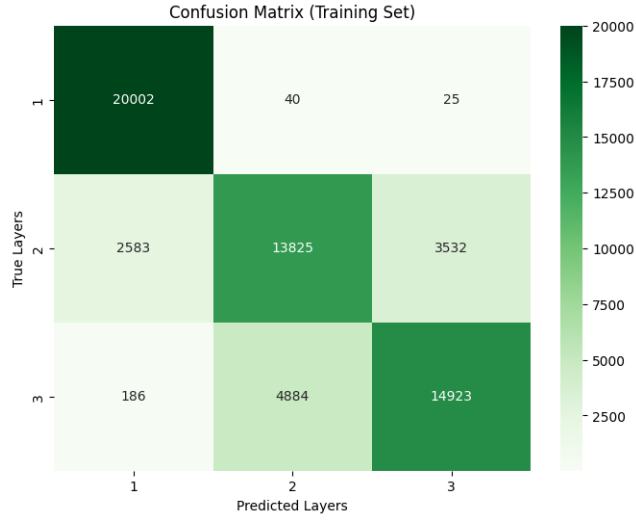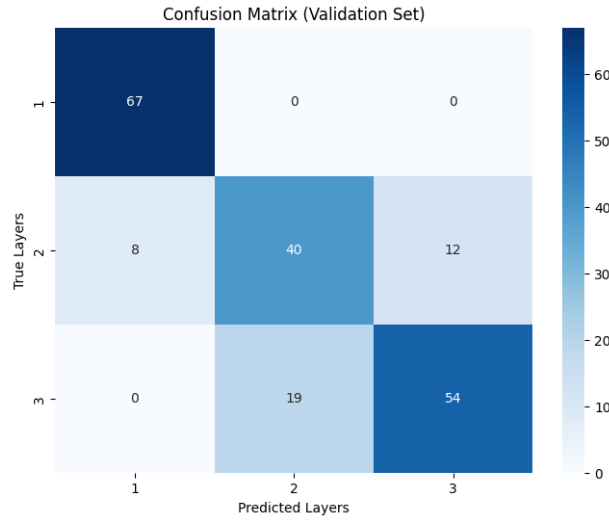

Figure 42: Confusion Matrix showing predicted and true classes of pulses classified using the trained CNN model. These simulated scans were unseen by the model and their material parameters were inside the training parameter space.

The models were trained using an NVIDIA RTX 3090 GPU. Training the models is not a par-

ticularly slow process (in the context of machine learning tasks) taking roughly an hour for 60,000 samples. The true bottle neck of this process is generating datasets. As mentioned in the section 3.5, a sufficient sample rate and upscaling is required when using the matrix transfer methods forward pass. For the parameter space used, a time domain resolution of 2^{12} was used, with upscaling of $\times 4$ and $\times 8$ before the conversion to the frequency domain. This is then downsampled to 2^{10} prior to being used in training. In it's present state, the method of generating datasets is CPU bound, these tasks can be parallelized across numerous CPU cores. Even with this optimization the dataset generation is still a very time intensive task. This model does not account for the reference pulse meaning effects such as laser drift, changing the emitter PCA or the delay line length will cause errors in the layer identification process. Different setups would require a training set generated with reference pulses specific to their experimental setups.

6   Conclusion and Future Work

This work has proposed several new methods to improve to the field of material classification using THz-TDS. These include algorithm based improvements to existing models as well as a new methods for THz waves during the extraction process. The core of this methodology is to pose the task as an inverse problem, where the goal is to identify the parameters of a physics based forward model for a given set of experimental data (THz-TDS scan). Solving this results in the identification of the material parameters for the sample. Traditional THz-TDS extraction relies on the Newton-Raphson method to numerically solve the basic Fresnel model for a single transmitted pulse through a single layered sample. This method solved for each frequency point sequentially. Scans where the sample is thin enough for internal reflections to be included in the time domain cannot be fit using the Fresnel model (11). Two possible solutions to this were to apply a window to the time domain to remove the extra pulses or to alter the Fresnel model to an alternative that can account for the reflections. Reflections can also be caused by the emitter. It is possible that reflections or other effects can overlap with the main pulse, in this case windowing is not effective. Newton's method can struggle to converge for more complicated versions of the Fresnel model, this is largely due to NR's usage of an analytical derivative.

   The Fresnel model was implemented by minimizing a loss function in frequency for the real and imaginary components of the transfer functions (predicted and experimental). The function is minimized using gradient descent with the Adam optimizer. This method is more robust for complicated transfer functions where an analytical derivative is not always possible. This method also allows for all frequency components to be calculated simultaneously as the loss function is global (calculated across all frequencies) while the optimizations take place at the individual frequency level. This implementation has comparable accuracy to the NR method while running much faster. To further test this implementation, tests should be run for two situations, time domain data with multiple reflections, this will confirm the improvements from the use of the gradient descent model. This algorithm should also be tested on a material with a refractive index that has a large change in the THz frequencies such as LiNbO [32]. The gradient descent algorithm on the Fresnel still requires an initial condition within 0.03 of the correct value, for single layered samples, this is easily achievable by looking at the pulse peak delay as in (1). For samples where the refractive index deviates more than 0.06 across the frequency range being optimized, the tensors used as initial conditions should also change with frequency. Efficient methods to do this with experimental data need to be looked into further.

While this implementation can deal with reflections, it does not combat the issue of noise in the reference pulse or multiple layers. A matrix transfer approach is proposed and used to create time domain simulations of THz pulses. This method takes a reference pulse in time and applies transformations at each boundary of a layered material. This method can be used as the forward pass in the gradient descent optimization algorithm. Using this model allows the algorithm account for all features in the time domain, this also allows for fitting material parameters for samples with multiple layers. Due to the computational expense of the of the forward and back passes on this model, fewer iterations are possible (for an extraction in a reasonable time). To combat this, a Bayesian optimization is performed before the gradient descent model is used. Bayesian optimization uses a probabilistic model (a Gaussian process) to approximate the loss landscape and identify the best regions of the parameter space. The acquisition function balances exploring the parameter space while targetting the best areas. This selects an optimal parameter set that can then be fine tuned using the gradient based algorithm. This method was tested, first using simulated data and then on experimental data. The model was also used to test the birefringence of a 1929 liquid crystal it identified a Δn comparable to that found by Bennis et al.[9] in the visible light region. Fitting the simulated data has shown that this method is capable of extracting material parameters to a high degree of accuracy (on the order of 10^{-4}). To further assess the accuracy of the models on experimental data, further scans using a longer delay line should be used. In this setup scans can be taken of all the materials to ensure errors are not caused by the surrounding materials.

The next step for the matrix transfer methods is to create a functional frequency dependent model. This is possible as the bulk of the computation takes place in the frequency domain. However, the model already takes a long time to optimize 2 or 3 parameters per layer from the time domain, this would increase dramatically with the introduction of varying parameters with frequency. One possible solution would be to return to calculating loss in the frequency domain. This should allow for loss to be parallelized with frequency. The loss landscape in frequency may be more complicated as it will be complex valued, but this can be tested and optimized for. Another option would be to attempt to approximate the frequency dependent refractive index using some n^{th} order polynomial and optimize the coefficients. This could provide the higher frequency resolution of the Fresnel model while only requiring a few more parameters to be optimised. For example, a 3rd order approximation of n and k would have 4 values for n and k each for each layer, given a good initial value this is still calculable within a reasonable time frame for 2 or 3 layers.

The matrix transfer methods used as the forward pass in the layered extraction method can be used to train neural networks. Convolutional neural networks pass a filter over structured data. Neural networks have been used to solve the Fresnel model [24] at the frequency level. The ability to simulate high fidelity time domain data opens the door to train models to perform a variety of tasks in the time domain. CNNs have been shown to accurately identify the number of layers in multi-layered samples. The models were trained on 60,000 time domain scans with material parameters sampled randomly from a defined parameter space. These achieved an 81 % accuracy with 80 % accuracy on unseen data. A study into the effects of the hyper-parameters of this model such as the number of layers and filter sizes could allow for greater generalisation and possibly higher accuracy for the same dataset. There are several routes to explore regarding CNNs in THz-TDS. Currently, the models are trained with a single reference pulse, training models on a variety of reference pulses. If this is ineffective, creating a 2-d CNN that takes both the reference pulse and the sample pulse could allow for a more general model that can identify layers. The main bottleneck of this work flow is the dataset generation, 60,000 scans at the required resolution took roughly 10 hours to generate despite running across all cores of the CPU. This could be remedied by trying to run the

data generation across the GPU, alternatively, high performance computing (HPC) solutions with high core counts are a viable option. These offer large core counts allowing for a greater level of parallelization. Once this is sorted, large datasets can be generated to train CNNs to extract material parameters directly from the time domain. As the complexity of the task increases, hybrid network structures can be used to pass additional information to the categorization head in the network, this will add a level of conditional inference. For example, passing total thickness to the classifier can help distinguish between pulses of different layers where the time domains exhibit similar features. Training 2-d CNNs to extract material parameters could be possible given the thickness of each layer as well as the number of layers. The conditional inference will remove the confusion between pulses that look similar but contain different material parameters.The 2-d architecture should teach the model to look for relationships between the reference and sample rather than simply trying to identify features in the sample pulse.

Another solution to the issue of training networks would be to employ the use of generative adversarial networks (GANs). GANs are a mixture of two neural networks, a generative one (G) and a discriminator (D). The generator creates synthetic data, with the goal of tricking the discriminator into classifying it as real data. The discriminator takes two sets of data, synthetic (from the generator) and real and learns to distinguish them. The two networks are trained as a zero-sum game where the generator tries to maximize the loss from the discriminator while the discriminator tries to minimize that loss. The goal is to converge to a generator that produces highly realistic synthetic data [20]. A conditional GAN can be trained to generate synthetic data with a condition related to the material parameters such as number of layers, thicknesses or the refractive indexes. This method is able to run on the GPU, this level of parallelization will provide a major performance improvement, allowing for the generation of much larger datasets. These larger datasets can improve the accuracy of the CNN models as well as increasing the complexity of the tasks the models can be trained to complete. Another benefit of GANs would be the ability to perform real-time augmentation during training. This is where the GAN produces new data during the each epoch of training, this allows for much greater generalization as the model can see an infinite and diverse training set. The GAN can also target areas of the parameter space that have higher than average error or introduce errors such as random noise.

This report shows novel optimization methods within THz-TDS. They have also opened the door for new machine learning models. This can allow for efficient and stable extraction of material parameters for complicated samples such as layered media or for setups with reflections in the reference pulse. This work extends the required foundations that will path the way to advanced machine learning methods to be used in THz-TDS and other material characterization techniques. We intend to publish the results from these new algorithms. These fitting methods can be incorporated into the existing TeraLyze library to extend the range of possible experimentation.

References

[1] `https://pytorch.org/blog/how-computational-graphs-are-executed-in-pytorch/`, 2025. Accessed:2025-01-15.

[2] `https://pytorch.org/blog/overview-of-pytorch-autograd-engine/`, 2025. Accessed: 2025-01-15.

[3] `https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html`, 2025. Accessed: 2025-03-10.

[4] Sebastian Ament, Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy. Unexpected improvements to expected improvement for bayesian optimization, 2025.

[5] Ben Beddoes, Malgosia Kaczmarek, V. Apostolopoulos, and Vassili Fedotov. All-optical switching of liquid crystals at terahertz frequencies enabled by metamaterials. *Optics Express*, 31:18336–18345, 05 2023.

[6] Ben Beddoes, Nicholas Klokkou, Jon Gorecki, Patrick R. Whelan, Peter Bøggild, Peter U. Jepsen, and Vasilis Apostolopoulos. Thz-tds: extracting complex conductivity of two-dimensional materials via neural networks trained on synthetic and experimental data. *Opt. Express*, 33(7):14872–14884, Apr 2025.

[7] Ben Beddoes, Nicholas Klokkou, Malgosia Kaczmarek, Vassili Fedotov, and V. Apostolopoulos. Demonstrating terahertz induced optical non-linearity in liquid crystal using z-scan method. page 31, 03 2025.

[8] Ben Beddoes, Nicholas Klokkou, Eleni Perivolari, Malgosia Kaczmarek, Vassili A. Fedotov, and Vasilis Apostolopoulos. All-optical reorientation of liquid crystals at terahertz frequencies enabled by metamaterials. In *2024 49th International Conference on Infrared, Millimeter, and Terahertz Waves (IRMMW-THz)*, pages 1–2, 2024.

[9] N. Bennis, T. Jankowski, O. Strzezysz, A. Pakuła, D. C. Zografopoulos, P. Perkowski, J. M. Sánchez-Pena, J. M. López-Higuera, and J. F. Algorri. A high birefringence liquid crystal for lenses with large aperture. *Scientific Reports*, 12(1):14603, 2022.

[10] Erik Bründermann, Heinz-Wilhelm Hübers, and Maurice F. Kimmitt. *Introduction*, pages 1–22. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[11] A.L. Chung. *Material parameter extraction in terahertz time domain spectroscopy*. PhD thesis, University of Southampton, September 2012.

[12] Jianming Dai, Jiangquan Zhang, Weili Zhang, and D. Grischkowsky. Terahertz time-domain spectroscopy characterization of the far-infrared absorption and index of refraction of high-resistivity, float-zone silicon. *J. Opt. Soc. Am. B*, 21(7):1379–1386, Jul 2004.

[13] A. Giles Davies, Andrew D. Burnett, Wenhui Fan, Edmund H. Linfield, and John E. Cunningham. Terahertz spectroscopy of explosives and drugs. *Materials Today*, 11(3):18–26, 2008.

[14] David Duvenaud. *Automatic model construction with Gaussian processes*. PhD thesis, Apollo - University of Cambridge Repository, 2014.

[15] L. Duvillaret, F. Garet, and J.-L. Coutaz. A reliable method for extraction of material parameters in terahertz time-domain spectroscopy. *IEEE Journal of Selected Topics in Quantum Electronics*, 2(3):739–746, 1996.

[16] Lionel Duvillaret, Frederic Garet, Jean-Francois Roux, and Jean-Louis Coutaz. Analytical modeling and optimization of terahertz time-domain spectroscopy experiments, using photoswitches as antennas. *Selected Topics in Quantum Electronics, IEEE Journal of*, 7:615 – 623, 08 2001.

[17] John F Federici, Brian Schulkin, Feng Huang, Dale Gary, Robert Barat, Filipe Oliveira, and David Zimdars. Thz imaging and sensing for security applications—explosives, weapons and drugs. *Semiconductor Science and Technology*, 20(7):S266, jun 2005.

[18] Peter I. Frazier. A tutorial on bayesian optimization, 2018.

[19] Wolfgang Ganglberger, Gerhard Gritsch, Manfred Hartmann, Franz Fürbass, Hannes Perko, Ana Skupch, and Tilmann Kluge. A comparison of rule-based and machine learning methods for classification of spikes in eeg. *Journal of Communications*, 01 2017.

[20] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[21] Luelue Huang, Miaoling Liu, Bin Li, Bimal Chitrakar, and Xu Duan. Terahertz spectroscopic identification of roast degree and variety of coffee beans. *Foods*, 13(3), 2024.

[22] Peter Uhd Jepsen. Phase retrieval in terahertz Time-Domain measurements: a "how to" tutorial. *Journal of Infrared, Millimeter, and Terahertz Waves*, 40(4):395–411, April 2019.

[23] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[24] N. Klokkou, J. Gorecki, and V. Apostolopoulos. Thz-tds parameter extraction via machine learning. In *2021 46th International Conference on Infrared, Millimeter and Terahertz Waves (IRMMW-THz)*, pages 1–2, 2021.

[25] Nicholas Klokkou. *Integrating Terahertz Time-Domain Spectroscopy with microfluidic platforms and machine learning for protein hydration studies*. PhD thesis, University of Southampton, January 2023.

[26] Nicholas Klokkou. Integrating terahertz time-domain spectroscopy with microfluidic platforms and machine learning for protein hydration studies. pages 23–24, January 2023.

[27] Ehsan Latif and Xiaoming Zhai. Fine-tuning chatgpt for automatic scoring, 2023.

[28] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[29] Jinchao Liu, Margarita Osadchy, Lorna Ashton, Michael Foster, Christopher J. Solomon, and Stuart J. Gibson. Deep convolutional neural networks for raman spectrum recognition: a unified solution. *Analyst*, 142:4067–4074, 2017.

[30] Ja-yu Lu, Li-jin Chen, Tzeng-fu Kao, Hsu-hao Chang, Hung-wen Chen, An-shyi Liu, Yi-chun Chen, Ruey-beei Wu, Wei-sheng Liu, Jeng-inn Chyi, and Chi-kuang Sun. Terahertz microchip for illicit drug detection. *IEEE Photonics Technology Letters*, 18(21):2254–2256, 2006.

[31] Analytics India Magazine. Openai gpt-3: Language model, 2025. Accessed: 2025-01-09.

[32] Zong-Liang Mao, Bi-Hui Hou, Li Wang, Yi-Min Sun, Guo-Qing Liu, and Wei Hao. The study of the terahertz spectral of linbo3 crystal. In *2006 Joint 31st International Conference on Infrared Millimeter Waves and 14th International Conference on Teraherz Electronics*, pages 465–465, 2006.

[33] Ankan Mullick, Sombit Bose, Rounak Saha, Ayan Kumar Bhowmick, Aditya Vempaty, Pawan Goyal, Niloy Ganguly, Prasenjit Dey, and Ravi Kokku. Leveraging the power of llms: A fine-tuning approach for high-quality aspect-based summarization, 2024.

[34] M. Naftaly and R.E. Miles. A method for removing etalon oscillations from thz time-domain spectra. *Optics Communications*, 280(2):291–295, 2007.

[35] Martin Norgren and Sailing He. A gradient-based optimization approach to the inverse problem for multi-layered structures. *International Journal of Applied Electromagnetics and Mechanics*, 10(4):315–335, 1999.

[36] The Nobel Prize Organization. The nobel prize in chemistry 2024 - press release, 2025. Accessed: 2025-01-09.

[37] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.

[38] Clément Pealat, Guillaume Bouleux, and Vincent Cheutet. Improved time-series clustering with umap dimension reduction method. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5658–5665, 2021.

[39] David Pfau, Simon Axelrod, Halvard Sutterud, Ingrid von Glehn, and James S. Spencer. Accurate computation of quantum excited states with neural networks. *Science*, 385(6711):eadn0137, 2024.

[40] Maria Schuld and Francesco Petruccione. *Machine Learning*, pages 23–78. Springer International Publishing, Cham, 2021.

[41] Yao-chun Shen, Prashanth Upadhya, Edmund Linfield, H. Beere, and AG Davies. Ultrabroad-band terahertz radiation from low-temperature-grown gaas photoconductive emitters. *Applied Physics Letters*, 83:3117–3119, 10 2003.

[42] James S. Spencer, David Pfau, Aleksandar Botev, and W. M. C. Foulkes. Better, faster fermionic neural networks, 2020.

[43] Oleksandr Sushko, Kastriot Shala, Rostyslav Dubrovka, and Robert Donnan. Revised metrology for enhanced accuracy in complex optical constant determination by thz-time-domain spectrometry. *J. Opt. Soc. Am. A*, 30(5):979–986, May 2013.

[44] Olive Emil Wetter. Imaging in airport security: Past, present, future, and the link to forensic and clinical radiology. *Journal of Forensic Radiology and Imaging*, 1(4):152–160, 2013.

[45] Mingjun Xiang, Hui Yuan, Kai Zhou, and Hartmut G. Roskos. Hybrid multi-head physics-informed neural network for depth estimation in terahertz imaging, 2024.

[46] Adnan Yahya, Nezah Balal, Avi Klein, Jacob Gerasimov, and Aharon Friedman. Improvement of the electro-optical process in gaas for terahertz single pulse detection by using a fiber-coupling system. *Applied Sciences*, 11:6859, 07 2021.

[47] Pochi Yeh. *Optical Waves in Layered Media*. Wiley, 1988.

[48] Ji-Yang Zhang, Jiao-Jiao Ren, Li-Juan Li, Dan-Dan Zhang, Jian Gu, Jun-Wen Xue, and Qi Chen. Terahertz spectroscopy and effective medium theory for thickness measurement of adhesive bonds. *NDT & E International*, 147:103216, 2024.