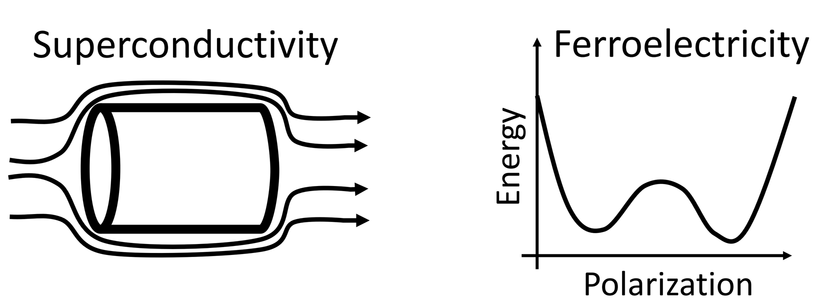
Optimizing Data Collection for Terahertz Spectroscopy

Ishir Garg[1, 3], Dana Berdichevsky[2, 3], Jacob Warshauer[3], Wanzheng Hu[3]

Leland High School, 6677 Camden Ave, San Jose, CA 95120[1]; Newton South High School, 140 Brandeis Rd, Newton, MA 02459[2]; Boston University, Boston, MA 02215[3]

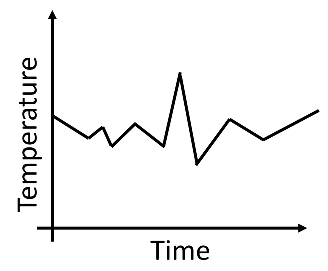
Introduction

Terahertz spectroscopy uses light pulses to analyze low energy quantum orders of crystals, such as superconductivity and ferroelectric properties.



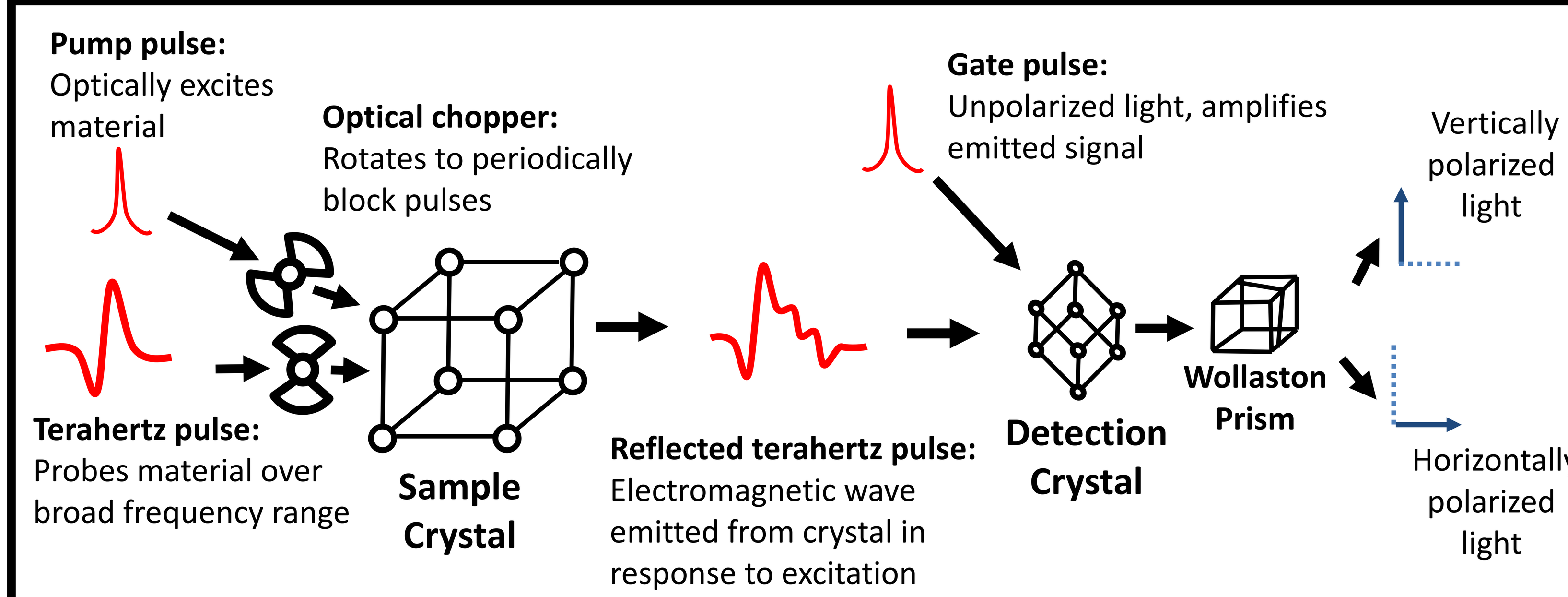
Problem: Data is noisy due to laser stability. Hence, it is optimal to perform as many scans of the terahertz signal as possible and average them.

Data can only be collected for limited periods of time, as the quality of the vacuum surrounding the crystal deteriorates. Water vapor leaks into the vacuum and freezes to form ice on the crystal at cryogenic temperatures.

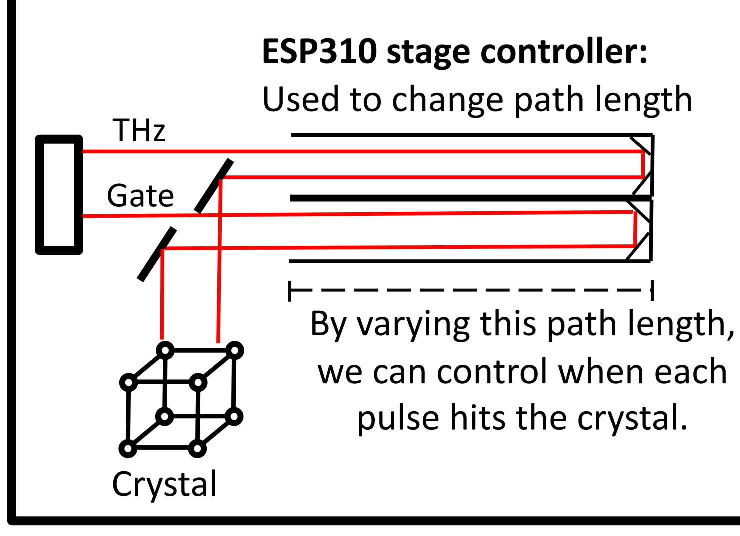


How do we collect more data during the vacuum period, so that the average of our signal is less noisy?

Experimental Setup



The material is excited out of an equilibrium state with the pump and probed with the terahertz pulse. We use **electro-optic sampling** to directly measure the electric field of the reflected pulse. The reflected pulse induces a birefringence in the **detection crystal**, biasing one of the polarizations in the gate pulse. A Wollaston prism splits the gate pulse into vertically and horizontally polarized light after sampling. A **lock-in amplifier** measures the difference between the amplitudes of these polarizations, allowing us to graph the reflected terahertz pulse. The **choppers** periodically block the terahertz and gate pulses, in order to isolate the reflected pulse from ambient light and analyze the effect of the pump.



Scanning across delay times for the lasers, allows us to measure the equilibrium and transient behavior of the crystal. Performing multiple scans, and taking their average, yields a less noisy result.

Goal: Reduce the time spent plotting and processing the measured signal by optimizing the interface running experiments, so that more data can be collected.

Conclusion

Time Complexity Improvements:

Using prefix sums reduces calculations with the signal and its FFT from $\boldsymbol{O(N^2)}$ **to** $\boldsymbol{O(N)}$.

Exponentially weighted vector averaging and RMS run in $O(N^2)$. In practice, these are infrequently calculated; the **amortized time complexity of calculations is** $\boldsymbol{O(N \log N)}$ **due to the FFT.**

Comparison of different methods:

Graphing the signal runs in $O(N)$, and the FFT runs in $O(N \log N)$, implying that our optimization of plotting should have a minimal effect. However, in practice, N is small enough that the time spent graphing often exceeds the time spent doing calculations.

Further decreasing the time spent graphing signals would significantly increase the data sampling rate.

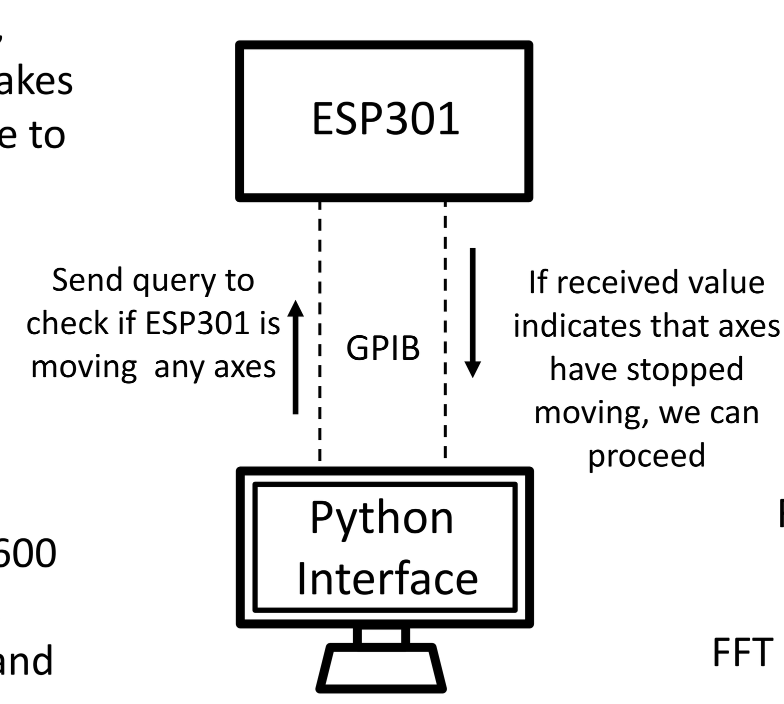
Further improvements:

Faster algorithms could be created to optimize the exponentially weighted vector average and root-mean-square.

Methods

Method 1:
Convert interface from LabVIEW to Python and optimize plotting

Python is significantly faster than LabVIEW, hence switching the interface to Python makes it faster. Additionally, we alter the interface to only plot measured signals when the user chooses to look at the graphs.



Method 2:
Maintain communication between the ESP301 and main interface

Previous programs have used a constant 2600 millisecond delay between measurements. This allows the stage controllers to move, and the lock-in amplifiers to update. Instead, constantly checking when the stage controller has finished moving allows the program to quickly move on to the next measurement.

Method 3:
Optimize calculations for the average signal and Fast Fourier Transform (FFT)*

$$\mathscr{F}\{f\}[k] = \sum_{n=0}^{N-1} f[n] e^{-\frac{i2\pi kn}{N}}$$

**Signal f is assumed to be an array of length N*

Given n scans $X_0, X_1, \dots, X_{n-1}$ over the same terahertz pulse, we calculate a few different quantities:

Signal Average: $$A[k] = \frac{1}{n}\sum_{j=0}^{n-1} X_j[k]$$

FFT Vector Average: $$A[k] = \frac{1}{n}\sum_{j=0}^{n-1} \mathscr{F}\{X_j\}[k]$$

FFT Root-mean-square: $$A[k] = \sqrt{\frac{1}{n}\sum_{j=0}^{n-1} |\mathscr{F}\{X_j\}[k]|^2}$$

FFT Peak-hold: $$A[k] = \max\{|\mathscr{F}\{X_j\}[k]| : j = 0 \dots n-1\}$$

Normally, every calculation for A below runs in $\boldsymbol{O(N^2)}$ **time**. Since scans are gradually added over the course of an experiment (more X_i are added), we use prefix arrays to compute in just $\boldsymbol{O(N)}$ **time**.

We wish to average over the last m scans for $m < n$.

For each of the four quantities, P denotes the 2-D prefix array, which is updated in $\boldsymbol{O(N)}$ **time** every time a scan X_i is added. A can be computed from P.

Signal Average: $$P[k] = P[k-1] + X_k, \quad P[0] = X_0$$ $$A[k] = \frac{1}{m}(P[n-1] - P[n-1-m])$$

FFT Vector Average: $$P[k] = P[k-1] + \mathscr{F}\{X_k\}, \quad P[0] = \mathscr{F}\{X_0\}$$ $$A[k] = \frac{1}{m}(P[n-1] - P[n-1-m])$$

FFT RMS: $$P[k] = P[k-1] + |\mathscr{F}\{X_k\}|^2, \quad P[0] = |\mathscr{F}\{X_0\}|^2$$ $$A[k] = \sqrt{\frac{1}{m}(P[n-1] - P[n-1-m])}$$

FFT Peak-hold*: $$P[k] = \max(P[k-1], \mathscr{F}\{X_k\}), \quad P[0] = \mathscr{F}\{X_0\}$$ $$A[k] = P[k]$$

**Only for m = n*

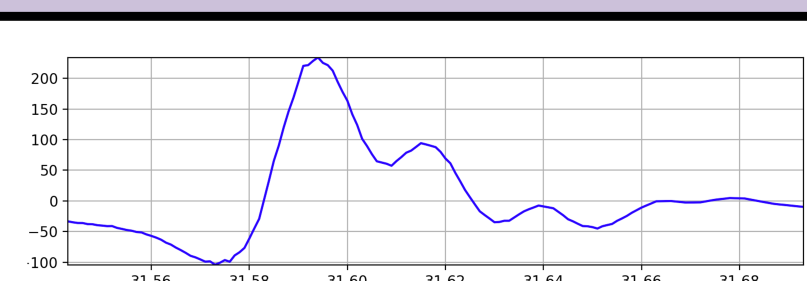
Results

Maintaining communication between the ESP301 and interface reduces run time by 76.2% from **2100 ms to 500 ms.** Previous interfaces used a 2600 ms delay to allow the stages to move and lock-in amplifiers to update. In practice, only 500 ms are needed, meaning that older interfaces would sit idle for an extended period.

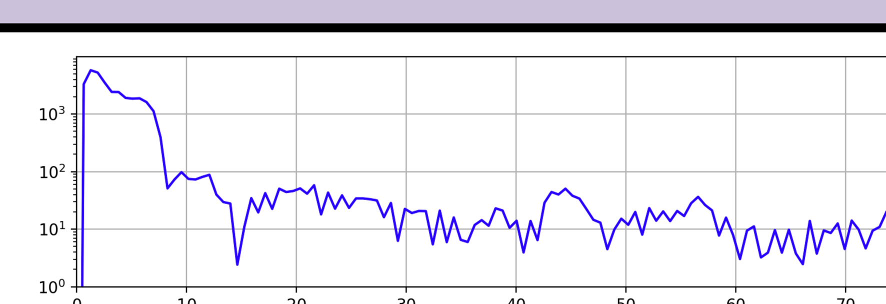
Altering the interface to plot signals only when the user is checking the graphs further reduces run time by 32%, from **500 ms to 340 ms.**

Implementing optimizations for time domain and frequency domain calculations further reduces run time by 35.3% from **340 ms to 220 ms.**

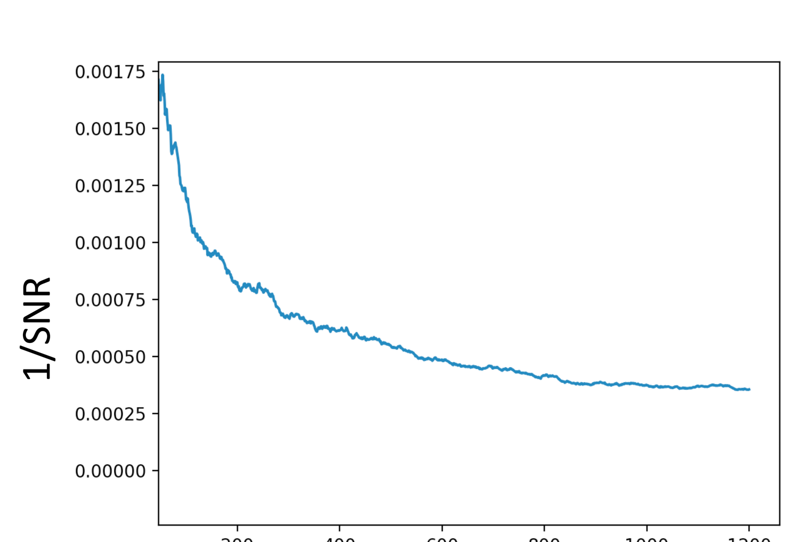
Original data sampling rate: 0.385 data/sec
New data sampling rate: 1.67 data/sec



Time-domain Reflected Signal



Frequency-domain Reflected Signal



Number of Completed Scans

This graph shows how the inverse of the signal to noise ratio (SNR) decreases with the number of completed scans, for a step count of 1000.

These changes increase the number of measurements taken over 5 days from **175,000 to 750,000.**

This increases the signal to noise ratio by 108% for 1000 steps.

Additional Notes

I worked on an additional project which involved measuring the flow rate of helium gas through a flowmeter in the helium recycling system, using an Arduino and Raspberry Pi. It has not been included in this research poster for conciseness.

References

1. Tokyo Instruments. About FFT Spectrum Analyzers. https://www.tokyoinst.co.jp/product_file/file/SR21_tech01_ja.pdf (accessed Aug. 4, 2022).
2. Physics Today 62, 8, 36 (2009); https://doi.org/10.1063/1.3206093
3. Brown University. Ultrafast Laser Spectroscopy. https://www.brown.edu/research/labs/mittleman/sites/brown.edu.research.labs.mittleman/files/uploads/lecture18.pdf (accessed Aug. 8, 2022)

Acknowledgements

I would like to thank Dr. Wanzheng Hu for giving me the opportunity to work in her lab, and Jacob Warshauer for his continual guidance and mentorship during the RISE program.