LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

**Technical Note** **LIGO-T1500194–vX** 2015/08/04

# Adaptive Feedforward Seismic Noise Cancellation at the 40m Interferometer

Jessica Pena
Mentors: Eric Quintero, Koji Arai, Rana Adhikari

**California Institute of Technology**
**LIGO Project, MS 18-34**
**Pasadena, CA 91125**
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

**Massachusetts Institute of Technology**
**LIGO Project, Room NW22-295**
**Cambridge, MA 02139**
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

**LIGO Hanford Observatory**
**Route 10, Mile Marker 2**
**Richland, WA 99352**
Phone (509) 372-8106
Fax (509) 372-8137
E-mail: info@ligo.caltech.edu

**LIGO Livingston Observatory**
**19100 LIGO Lane**
**Livingston, LA 70754**
Phone (225) 686-3100
Fax (225) 686-7189
E-mail: info@ligo.caltech.edu

http://www.ligo.caltech.edu/

# 1 Background

## 1.1 Noise in LIGO Interferometers

LIGO Interferometers send a laser beam through a beam splitter to be locked in Fabry-Perot cavities and recombined so that the effects in the cavity can be observed [1] However, many other factors aside from gravitational waves affect the resulting signal from the lasers. These outside factors are noise that must be filtered out either through the construction of a mathematical filter or physically building a filter to dampen the noise. Many kinds of noise affect LIGO Interferometers, but Seismic noise and instrumental self-noise are most problematic because they affect frequencies LIGO is trying to detect and are more difficult to precisely control mechanically [1]. Mechanical filters, including stabilizing the concave mirrors in each cavity through magnetic fields and using oscillators to dampen seismic noise, are already in place but are always being improved.

## 1.2 Feedforward vs Feedback Filtering

Mathematical filters are just as important as physical filters. Feedforward techniques predict what noise will be in the signal and adjust the detector so that this particular noise is not part of the output. Feedback adjusts the detector after noise has propagated through the system [2]. The differences between feedforward and feedback are depicted below (figure 1). In a feedback loop, the disturbances and noise sources must pass through the system in order to be detected, whereas with feedforward these noise sources are predicted and preemptively filtered out. Thus, there is no time lag in feedforward filters that is present in feedback filters [2]. Wiener filters in particular are desirable to use for feedforward techniques because they determine the value of an unknown signal given known signals. Using feedback, one can determine the noise of the system and then construct a Wiener filter based off of that [3]. A feedforward Wiener filter can then be created. These filters will have an extensive impact, minimizing seismic noise through multiple degrees of freedom [4]. Therefore, we will only need to construct one Wiener filter to filter out noise from several sources.

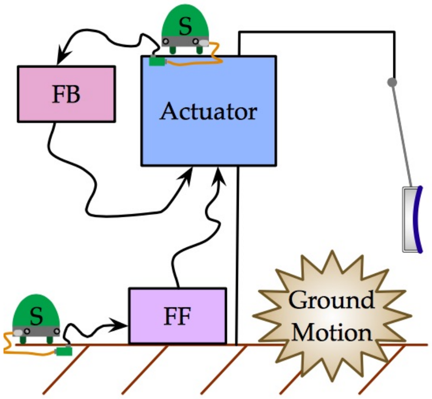


Figure 1: The differences between a feedback loop and a feedforward loop [3]

### 1.3 IIR Wiener Filters

In this case, IIR Wiener filters are ideal because they require fewer parameters than FIR filters, which is important because Wiener filters have to create a very precise model of the system, so the fewer parameters required means that there is less room for error. This also drastically reduces computational time of the filters by using IIR instead of FIR coefficients [4]. This is especially important for a feedforward system, which must predict how to filter the signal prior to receiving it. Also, IIR filters are more likely to achieve the lowest mean-square error because they are computed by integral, while a FIR filter is a summation [5]. Therefore, FIR filters can come close to achieving the results of IIR filters, but will never be quite as good. However, IIR filters are difficult to calculate, and they also can introduce noise into controls of the system because of how each control interacts with others [4]. Therefore, it will be difficult to implement an IIR filter that also optimizes subtraction and introduces minimal noise into other parts of the system.

## 2 Goals During the First Seven Weeks

### 2.1 June Goals

During the first several weeks of the summer, I compared various algorithms and techniques of filtering noise from our sensors to determine self-noise. Self-noise is important to know because it cannot be mechanically be filtered out and gives us a minimum goal. We use both seismometers and piezoelectric accelerometers to detect seismic noise, so it is important to know the self-noise of both. Piezoelectric accelerometers are solid state devices, consisting of a crystal and a seismic mass. When a force is applied to the crystal by seismic motion, a voltage is created across it which we then measure.[6]. A seismometer consists of a weight on a spring, with the attached frame connected to the earth's surface. Seismometers act as a transducer between the input acceleration and the position of the mass [7]. They also behave similarly to accelerometers at small frequencies. In either case, we can measure the input and output of individual sensors, but have no way to physically isolate the noise from a single sensor. As a result, we must use multiple sensors, which all measure the same signal, and then use the differences in these results to determine the noise in an individual sensor. I applied the Three-Cornered Hat Technique and Wiener filtering to accelerometers which we had performed a huddle test on. Wiener filtering was applied to seismometers.

### 2.2 July Goals

After analyzing data using the Three-Cornered Hat Technique and FIR (Finite Impulse Response) Wiener filters, I explored other options for noise reduction. While the first two methods did filter out a significant amount of noise, the filtered noise curve was still significantly higher than the predicted self-noise of the accelerometers. Pre-filtering the data and then applying a Wiener filter is one way to further reduce data. Pre-filtering is important as, if excess noise can be removed before running the data through a filter, the filter will be more accurate and a greater overall reduction of noise will occur [8]. This month, I focused on researching various forms of pre-filtering and ultimately used an elliptic bandpass filter.

I also explored the IIR (Infinite Impulse Response) Wiener filter as another offline alternative to the FIR Wiener filter. The IIR filter will also be easier to implement in an online subtraction.

# 3 Three-Cornered Hat Method

I used the Three-Cornered Hat method to determine the noise in accelerometers subject to the huddle test. To perform a huddle test, the six accelerometers were grouped together so that each measures the same quantity. They were also clamped to the table to reduce differences in seismic noise between them. Given the results of the huddle test, I was able to calculate the Power Spectral Density (PSD) line for each accelerometer and determine its self-noise using the Three-Cornered Hat technique. Following the steps outlined in Section 3, I made the plots in two sets of three, determining the noise of accelerometers 1-3 separately from accelerometers 4-6. Figure 2 shows the noise levels for the first accelerometer using the Three-Cornered Hat method. It can be seen here that at lower frequencies, especially between 1 and 10 Hz, the Amplitude Spectral Density (ASD) has been significantly lowered, while at higher frequencies the Three-Cornered Hat method did not filter as much out. Comparing this to the expected noise, which is the green line, it can be seen that what has been filtered out by the Three-Cornered Hat method is still significantly higher than is desired between 10 and 1000 Hz.

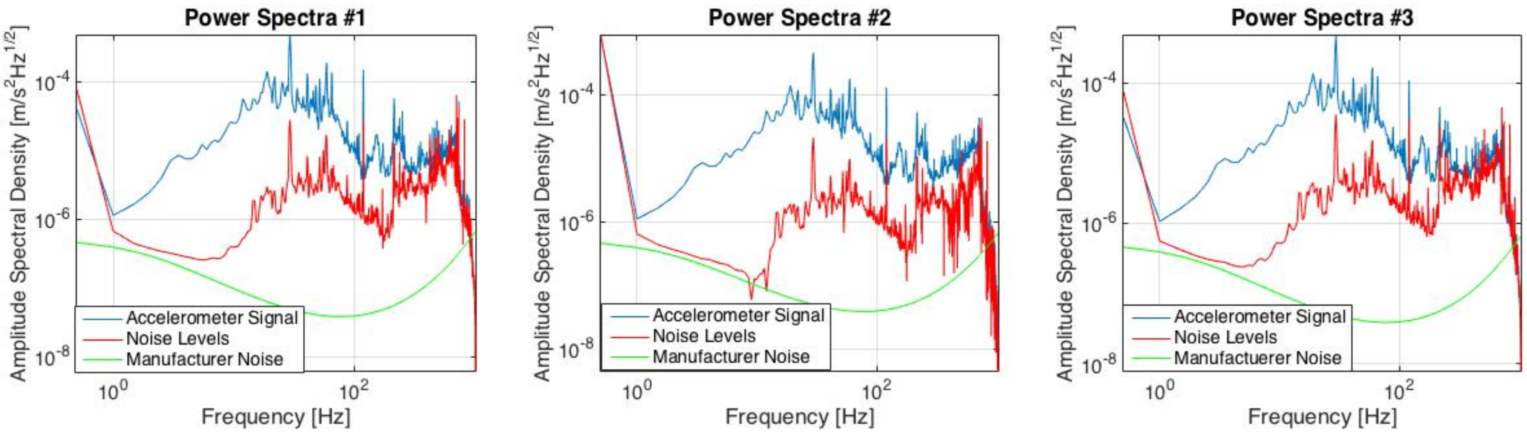


Figure 2: PSD plots for the noise in the first three accelerometers in the huddle test, made by the Three-Cornered Hat method

# 4 Constructing a Wiener Filter

Wiener filters are used to calculate noise by comparing multiple input signals to an output signal. This is useful in determining the self-noise of accelerometers and seismometers, and has the potential to filter noise better than the Three-Cornered Hat method can. The coefficients of a Wiener filter come from solving this RMS minimization equation:

$$\xi =< d^2 > -2\vec{\omega}^T\vec{p} + \vec{\omega}^T R\vec{\omega} \tag{1}$$

where $\xi$ is what we will minimize to calculate the Wiener coefficients, d is the signal we wish to filter, $\vec{\omega}$ is the Wiener filter we are solving for, $\vec{p}$ is the cross-correlation vector between witness and target signals, and R is the matrix [9]. Given an array of vector sources and a vector output, the Wiener filter determines coefficients that can be applied to the input data to remove noise from it. The Block-Levinson method was used to calculate a SISO (Single Input Single Output) filter, while a brute force method for matrix inversion was used when calculating MISO (Multiple Input Single Output) Wiener filters.

# 5 MISO Filtering of Accelerometers

I first created a MISO Wiener filter and applied it to the accelerometers subject to the huddle test. This way, I can later compare the Three-Cornered Hat method of determining self-noise to the ability of the Wiener filter to do the same. To create a Wiener filter for these accelerometers, I used a function that implemented the matrix algebra described in the section above. Each accelerometer was a separate input. There was no specific output. Instead, I used 2 accelerometers as input and a third as output to determine the self-noise of that particular accelerometer. The results of this (figure 3) were successful. At low frequencies, the Wiener filtered noise was still above the expected ASD, shown by the green line below, but it filtered out much more than the Three-Cornered Hat method did at higher frequencies. Comparing the Wiener filter of the accelerometers to the Three-Cornered Hat method of determining self-noise of accelerometers, as is pictured below (figure 4), I found that the Wiener filter worked much better, especially taking high frequencies into account. In low frequency ranges both methods produce similar results, but the significantly lower noise curve at high frequencies produced by the Wiener filter is promising. It motivates further exploration into the Wiener filter, including adaptive as opposed to static filters. A combination of Wiener filtering and the Three-Cornered Hat method also holds potential to filter noise down to the manufacturer's predicted level.

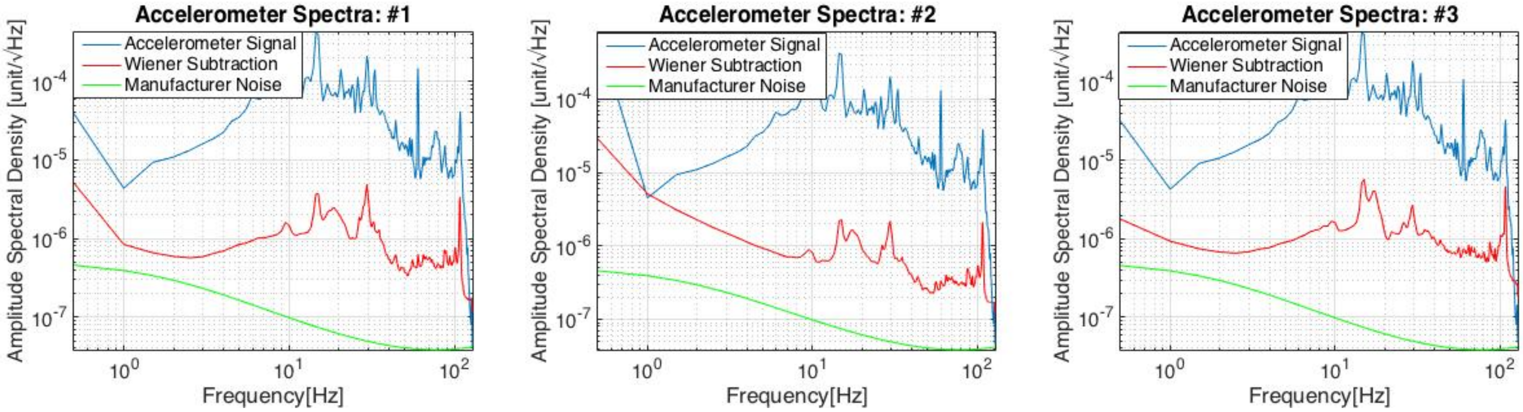
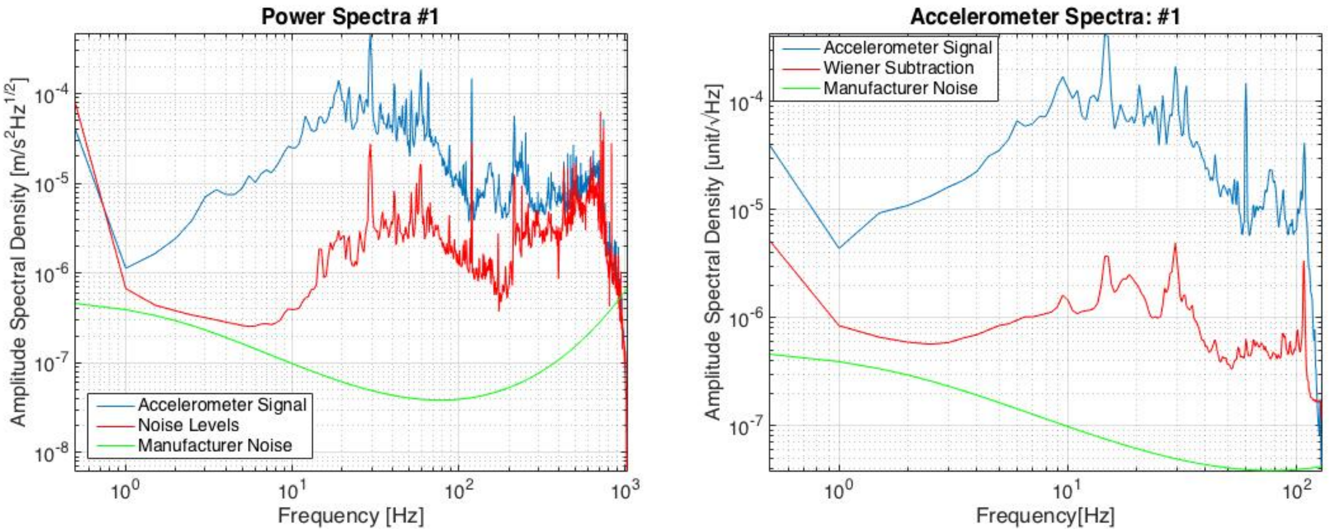


Figure 3: MISO Wiener filtering of the first three accelerometers in the huddle test



(a) Three Cornered Hat method applied to the first accelerometer in the huddle test

(b) MISO Wiener filtering of the first accelerometer in the huddle test

Figure 4: Comparison between the Three-Cornered hat method (a) and MISO filtering (b)

## 6 MISO Filtering of Seismometers

I then used Matlab to apply MISO filtering to a seismometer having components in the x, y, and z directions. The seismometer was measuring noise around the mode cleaner, so all three components had an output corresponding to the mode cleaner. There are also two accelerometers, each with three directional components, that are measuring noise at different ends of the mode cleaner. I used the accelerometers and seismometers to find one set of Wiener coefficients. Subtracting all of this from the output and then finding the PSD resulted in the plot below (figure 5). This filter successfully reduced noise in the 3-10 Hz range.

## 7 Pre-Filtering Methods

Pre-filtering data will improve the subtraction of the Wiener filter by emphasizing certain frequencies more than others. Typical filters include Butterworth, Chebyshev I, Chebyshev II, and Elliptic filters. These in turn can be bandpass, lowpass, or high-pass filters. Lowpass filters remove high frequencies while high-pass filters filter out lower frequencies. Bandpass filters allow a certain range of frequencies through to be emphasized, which is why that is the ideal filter for the mode cleaner length, which has noise both above and below the frequencies of interest.Once the filter type is chosen, there are specific parameters of the filter that must be specified. Passband ripple and stop-band attenuation are two of these parameters that determine the goodness of the filter. As can be seen in figure 6, the passband ripple is the amount of variation allowed in the passband, or the region containing allowed frequencies. The stop-band attenuation allows for fluctuation in the region of blocked frequencies [12]. A lower passband ripple and higher stop-band attenuation has proved to work best in the data I have filtered so far. Each filter results in a transfer function of varying complexity.

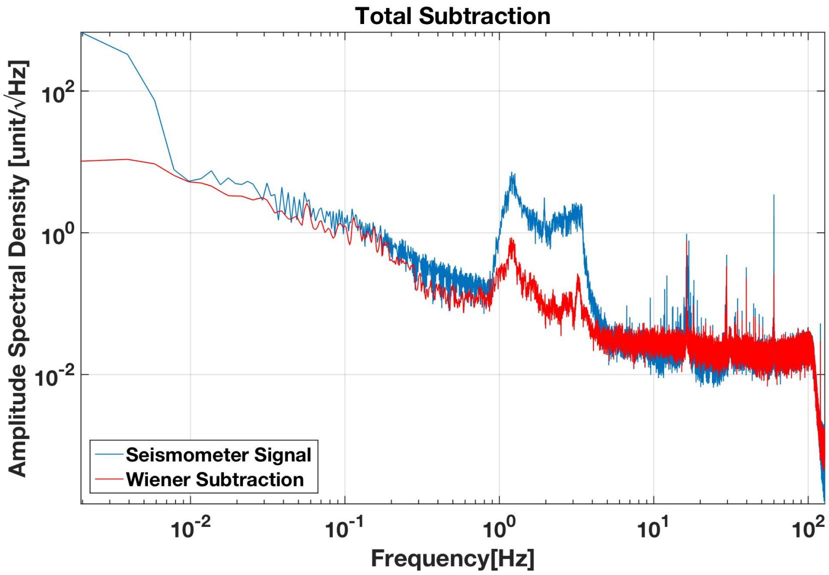


Figure 5: Filtering noise in seismometer and 2 accelerometers out of mode cleaner

Butterworth filters are the simplest filters, with a transfer function H(s) [10] of

$$H(s)H(-s) = 1 + (-s^2)^n \tag{2}$$

This will result in a maximally flat passband, and is also simple enough that passband ripple and stop-band attenuation are not taken into account. This transfer function, while simple, also means that the transition between the passband and stop-band is very slow, so there will be no sharp drop as is depicted in figure 6, but there will instead be a slope and excess noise from the stop-band will be allowed through the filter [10]. Both Chebyshev filters are more precise than the Butterworth filter. Chebyshev I filters allow for passband ripple only, while Chebyshev II filters only allow ripple in the stop-band [11]. This makes each of them more precise and complex than Butterworth filters. Elliptic filters are the most complex filter, allowing both passband ripple and stop-band attenuation. This transfer function is much more complex [10]

$$H(jw)^2 = 1 + [\epsilon R_n(\frac{w}{w_B}, L)]^2 \tag{3}$$

By accounting for passband ripple and stop-band attenuation, elliptic filters are most precise and will calculate a filter with a very low order that has the same result as a high ordered Butterworth filter [10]. This makes elliptic filters computationally ideal, as Matlab will be able to calculate a low-ordered elliptic filter much faster than a high ordered Butterworth filter. With decreased run-time, implementing an online pre-filter will become much more plausible.

# 8 Pre-Filtering of Mode Cleaner Length Data

To reduce noise in the data for the mode cleaner length, I implemented a third order elliptic bandpass filter in Matlab. I focused on frequencies between 1 and 20 Hz. These frequencies

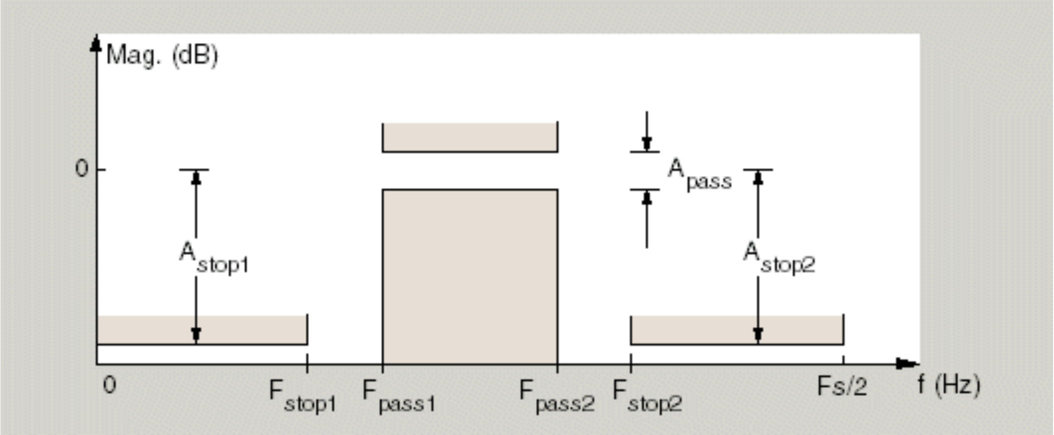


Figure 6: Image of stop-band attenuation and passband ripple [12]

encompass the range that we want to focus on, so noise reduction is most important in this range. So far, the results from this filter are a minimal improvement from Wiener filtering alone. Figure 6 illustrates this, and that noise is injected in ranges outside of 1-20 Hz. However, since there are no excessive noise injections outside this range, meaning that the excess noise never rises far above the original signal, the filter is still good to have. To calculate this filter, I used a passband ripple of 1 dB and a stop band attenuation of 20 dB. As I did not derive a method to choose these numbers and just thought about ideal ranges for passband ripple and stop band attenuation, the derivation of a method would be a good thing to look into. I will also explore other methods of pre-filtering, to see if an elliptic bandpass filter itself is not the most ideal filter.

# 9 IIR Wiener Filtering Using Vectfit

Once a working pre-filter had been calculated for the mode cleaner length, I ran the pre-filtered data through vectfit to create an IIR (Infinite Impulse Response) Wiener filter. Similar inputs and outputs to the ones described in section 4 are used in IIR Wiener filters. However, the transfer function must first be calculated for each input before data can be interpreted by vectfit. It is also important to pre-weight the data before it is used by vectfit. Pre-weighting emphasizes one region of frequencies, allowing one region to be minimized the most by the subtraction. Vectfit take the transfer function and finds the zeros, poles, and gain (zpk) of the function [13]. As vectfit returns the zpk coefficients in the s-domain, I then transform these to the z-domain. From there, a second-order section model is created, which can then be applied to the data and filter it appropriately.

# 10 Results of IIR Wiener Filtering

Using an IIR Wiener filter has made little difference from using a FIR Wiener filter. It is probable that the pre-weighting factor used needs to be optimized. In Figure 8, the pre-

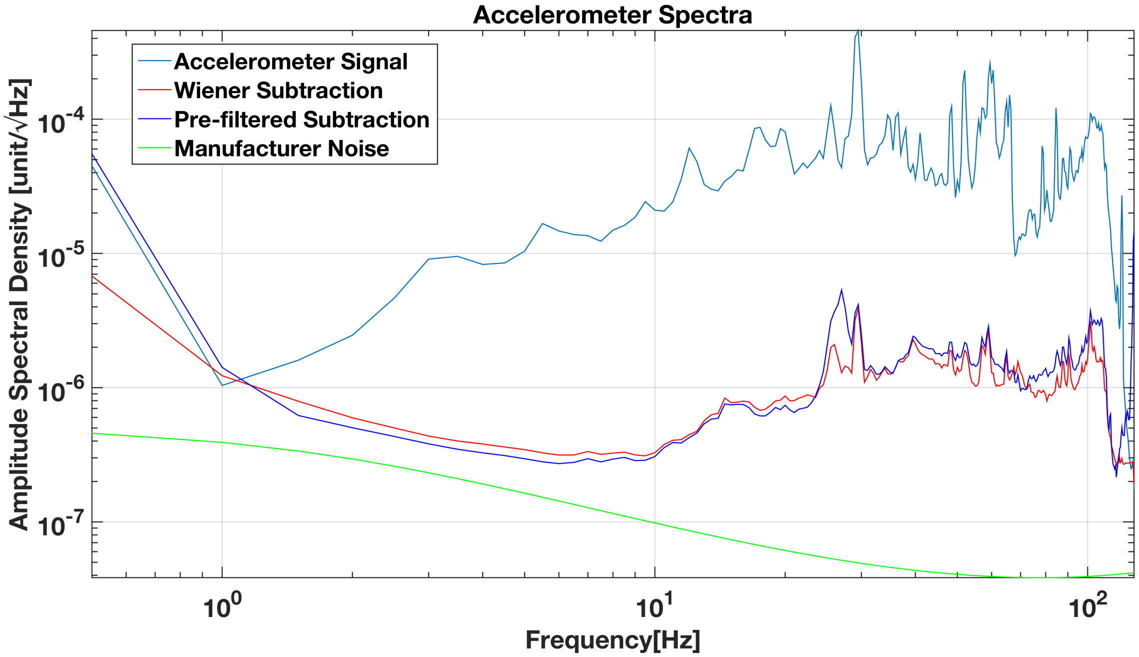


Figure 7: The use of a pre-filter (dark blue) showed little improvement from a Wiener filter without pre-filtering

weighting for the IIR Wiener filter was $1/\sqrt{tf}$, where tf is the transfer function. This is an improvement from no pre-weighting, but designing a pre-weight to optimize the area between 1 and 20 Hz is the next step in this process. Online IIR subtraction will be implemented in the coming week, with the hopes that it will lower the noise. As the current settings have not increased noise levels and did as well as FIR Wiener subtraction, it is the hope that IIR online subtraction will work just as well.

# 11 Long Term Goals

Generally, by the end of summer I want to construct a static online IIR filter and then create an adaptive online IIR filter based off of that.

Week 8: Construct a static online IIR filter, finish up and turn in Progress Report 2. Continue work on pre-filtering and finding new techniques to do so more efficiently. Work to pre-filter the Arm length data, not just the mode cleaner data. Complete the ALS Delay line box by choosing the proper front panel.

Week 9: Using the static IIR filter, I will begin working on the adaptive online IIR Wiener filter. Continue work with pre-filtering

Week 10: Finish up the IIR adaptive online filter and work on my presentation and final paper.

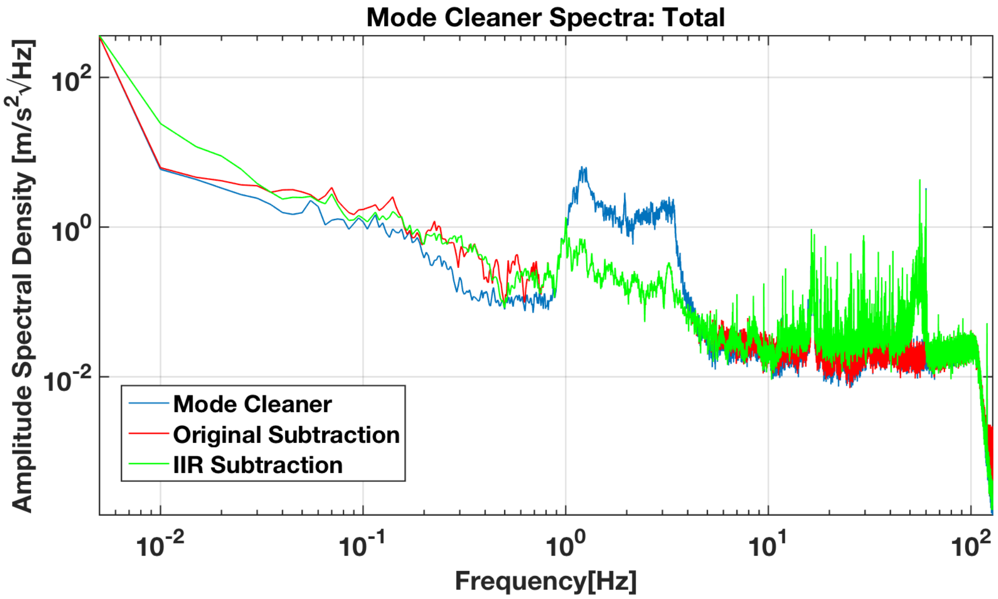


Figure 8: Comparison of FIR vs. IIR Wiener Filtering of accelerometer self-noise out of the mode cleaner length

# 12 Appendix

## 12.1 Arm Length Stabilization Delay Line Time Delay

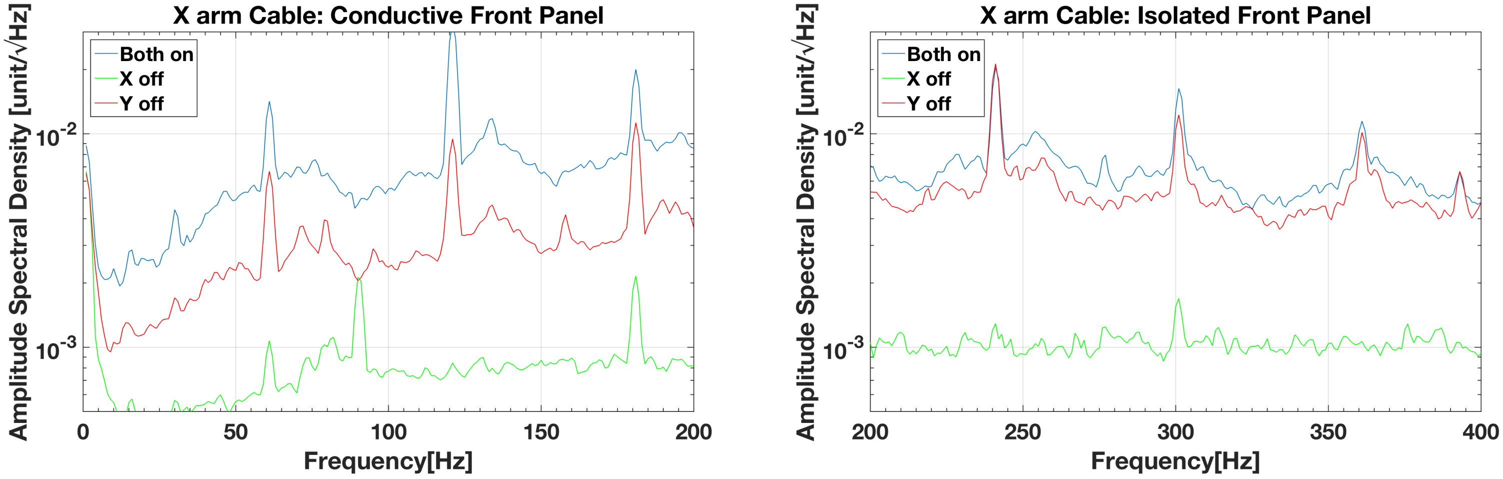
During my first three weeks working on filtering, I also worked on another project trying to physically isolate the 50m long ALS (Arm Length Stabilization) Delay line cables. My goal was to keep the cables seismically and electrically isolated. To do this, I wanted to put them inside the same conductive box so that they feel the same seismic vibrations and are inside a Faraday cage to keep them electrically isolated. When the cables arrived, I had to determine the time delay in the cables. To do this, I created Bode plots for each cable. I then used this data to write a code in Matlab that found the time delay in each cable. Both had a time delay of 127 ns.

## 12.2 ALS Delay Line Box Testing

The front panels for the delay line box arrived, so I was able to build the ALS Delay Line box. I divided it with a conductive grid of metal to prevent crosstalk between the two cables. I also covered both cables with foam to prevent them from moving around too much. Once the box was constructed, I tested both the front panels with conductive and isolated SMAs to determine if crosstalk was occurring between the cables. I did this by driving a frequency through one of the cables, and driving another frequency less than 800 Hz different from the first through the second cable. Looking at the PSDs of these should reveal if crosstalk occurs, as if it does, spikes will be visible at the frequency of the difference between the driving frequencies.

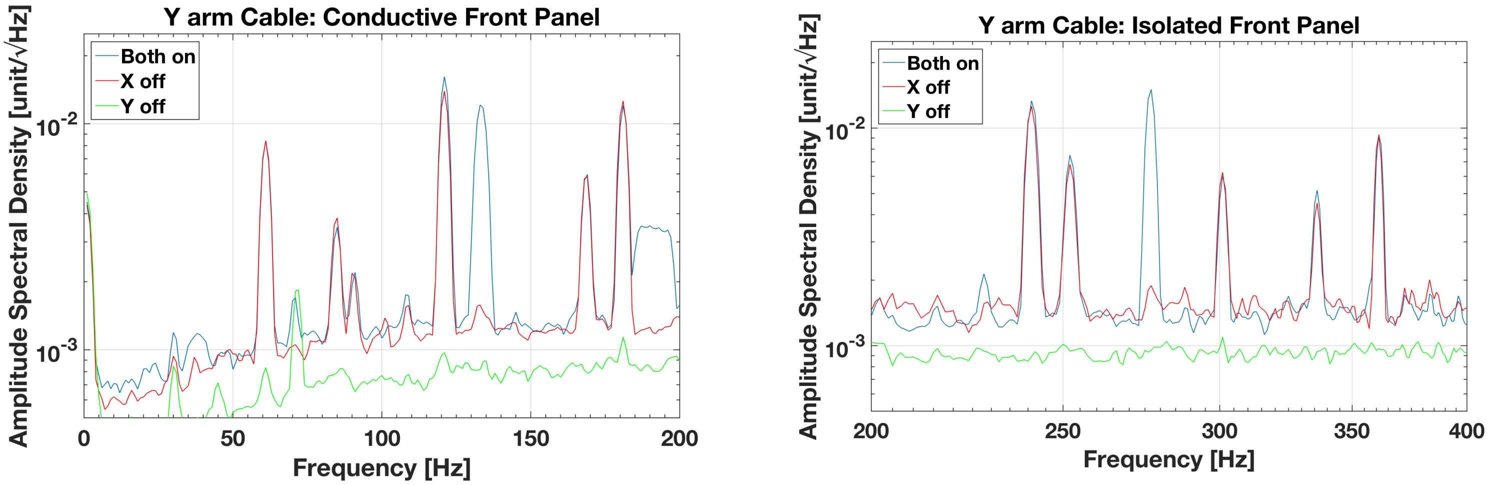
### 12.3 ALS Delay Line Analysis

For the conductive SMA panel, I drove 22.329 MHz through the X Arm and 22.3291 MHz through the Y Arm, giving a difference of 100 Hz. The X Arm PSD is shown in figure 9a and the Y Arm PSD is shown in figure 10a. Three sets of data are shown in each plot. The first set was taken when frequencies were being driven through both cables, while the next two were when one of the cables had a frequency driven through it and the other was off. I centered each plot around 100 Hz, because if crosstalk was occurring then a spike would appear when a frequency was driven through both cables but not when one cable was off. So, in the Y Arm data, a spike can be seen at around 120 Hz when both cables were on but not when there was no frequency driven through the X cable. This indicates that some crosstalk may be going on when the conductive SMAs are used.



(a) X arm cable data using conductive front panel (b) X arm cable data using isolated front panel

Figure 9: Comparison between the conductive panel (a) and the isolated panel (b)



(a) Y arm cable data using conductive front panel (b) Y arm cable data using isolated front panel

Figure 10: Comparison between the conductive panel (a) and the isolated panel (b)

However, data from the isolated SMA panel shows similar results to the conductive panel.

Driving 22.294 MHz through the X Arm and 22.2943 MHz through the Y Arm, there was a difference of 300 Hz between the cables in the isolated front panel. Similarly to the data for the conductive front panel, it can be seen in figures 9b and 10b that there is no noticeable spike at 300 Hz for the X Arm data, while there is one around 300 Hz for the Y Arm data. This spike indicates potential crosstalk between the cables. While the spike is slightly higher than 300 Hz in the Y Arm PSDs, it is highly likely that the frequency generator is  30 Hz off, as the frequency was on the order of 23 MHz. It is interesting that the crosstalk only shows up in the Y Arms, and that it makes no difference whether conductive or isolated SMAs are used. It is possible that this spike is not due to crosstalk but has other causes, which I will be investigating in the coming weeks.

# References

[1] G. Cella and A Giazotto *Interferometric Gravity Wave Detectors*. Review of Scientific Instruments 82, 101 101 (2011).

[2] John Bechhoefer, *Feedback for Physicists: A Tutorial Essay on Control*. Review of Modern Physics, Volume 77 (July 2005).

[3] J. Driggers, M. Evans, K. Pepper, and R. Adhikari, *Active Noise Cancellation in a Suspended Interferometer*. arxiv:1112.2224. (December 2011).

[4] R. DeRosa, J. Driggers, D. Atkinson, H. Miao, V. Frolov, M. Landry, J. Giaime, and R. Adhikari, *Global Feed-forward Vibration Isolation in a km scale Interferometer*. Quantum Grav. 29 215008 (2012).

[5] Natanael Fontes *An Analysis of the IIR and FIR Wiener Filters with Applications to Underwater Acoustics*. Naval Postgraduate School (1997).

[6] *Introduction to Piezoelectric Accelerometers* PCB Piezotronics (2015).

[7] M.J. Usher, C. Guralp, and R.F. Burch. *The Design of Miniature Wideband Seismometers* Geophysical Journal of the Royal Astronomical Society (1978).

[8] S. Orfanidids *Introduction to Signal Processing* Pearson Education Inc., 2010.

[9] J. Driggers *Noise Cancellation for Gravitational Wave Detectors* California Institute of Technology (2015).

[10] R. Daniels *Approximation Methods for Electronic Filter Design* Bell Telephone Laboratories, 1974.

[11] S. Smith *The Scientist and Engineer's Guide to Digital Signal Processing* California Technical Publishing (1997).

[12] http://www.mathworks.com/help/dsp/ref/fdesign.bandpass.html

[13] https://www.sintef.no/projectweb/vectfit/passivity/