Developing Methods to Characterize Frequency-Varying Combs in LIGO Strain Data

Joseph Talucci
Dr. Ansel Neunzert

Abstract

Narrow spectral artifacts in strain spectra pose a unique challenge to searches for continuous waves. They appear as small vertical lines, obscuring potential detections. Combs are patterns of equally-spaced narrow lines representing the harmonics of a fundamental frequency. Characterizing combs and identifying their individual causes is challenging, and is made more difficult by the fact that some of them change in fundamental frequency over time. We call these roaming combs, as their narrow spectral artifacts appear to roam around spectra. The Fscan spectral monitor used for continuous-wave detector characterization can automatically identify combs present on a particular day, but lacks a method to analyze how combs have changed in frequency over time. Our objective for this project was to develop a method to analyze frequency-roaming combs. We produced a script which automatically detects when combs have changed in frequency since their last appearance and can also be used to examine known combs which are not detected by Fscans. We also present results on known, high-priority combs contaminating Hanford data.

Introduction

Gravitational waves (GWs) are disturbances in the curvature of spacetime propagating at the speed of light. GWs are emitted when an asymmetric, massive object moves in spacetime. They have two types of polarizations, often denoted as plus and cross, each causing spacetime to stretch and compress periodically in different orientations. As a gravitational wave passes, it alternately stretches and compresses space in one direction while simultaneously compressing and stretching it in the perpendicular direction. A particle in the path of a gravitational wave will oscillate back and forth along with the stretching and compressing.

In general relativity, GW emission is modeled by a time-varying mass quadrupole moment, analogous to how the distribution of mass changes in a system over time. A common example of this effect is in the case of a merging system of two orbiting, massive objects like black holes or neutron stars. Over time, they will spin inward radially towards one another as the system loses energy to the emission of GWs. Then they merge together and emit one last burst of gravitational waves, commonly called a chirp signal. This particular kind of event is called a compact binary coalescence, and the gravitational waves it emits are classified as non-continuous, or transient, as the event has a well-defined, observable end. Detecting non continuous GWs has become trivial for the current-generation of detectors, but continuous GWs remain elusive.[1]

Continuous GWs are thought to be emitted by much more stable, longer-lasting sources, such as high-frequency spinning neutron stars which are not perfectly symmetric. However, the changing mass quadrupole associated with one star spinning as opposed to a revolving binary is significantly less prominent. And in the case of pulsars, those objects lose energy to emitting both GWs and electromagnetic waves. In any case, continuous GWs represent a much lower energy emission from the system than transient GWs, so they are much more difficult to detect. Searches for GWs have been unsuccessful so far.[2]

Gaining the capability to detect continuous GWs (CWs) would be an enormous victory for stellar astrophysics. If CWs from a pulsar were detected, then astronomers would empirically prove that neutron stars exhibit asymmetry – Electromagnetic radiation tells astronomers nothing about the shape of the star. And if GW astronomers find nothing, then they may confidently say that neutron stars are completely symmetric. This capability would also help astronomers locate objects which are otherwise unobservable. For example, supernova remnants like Formalhaut b are thought to possibly contain unidentified, young neutron stars. By conducting successful CWs searches we would be able to say for sure. Further, many pulsars' emissions are simply pointed away from the Earth, so they are currently inaccessible for observation. Thus, capability for detecting CWs would also be an immense boon for multi messenger astronomy.[2]

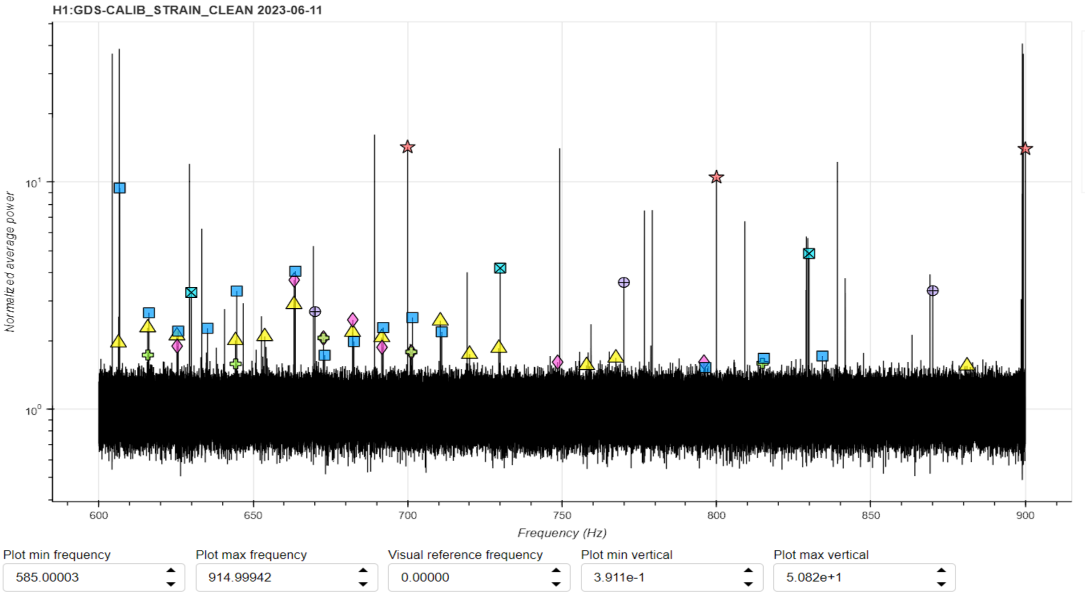
The Laser Interferometer Gravitational-wave Observatory (LIGO) sites are a group of interferometers around the world which detect gravitational waves. They are located in Hanford Washington, Livingston Louisiana, Italy, Japan, and soon Indias. In an interferometer, a beam of light from a powerful laser is split into two, allowed to travel a far distance forward then back in either direction, then recombined. If there is a difference in arm length – called strain – due to a gravitational wave (i.e. if the length of either arm changes) then there will be a detectable phase difference between the recombined beams. To this end, each interferometer features two 3 or 4-kilometer long Fabry-Perot arm cavities – consisting of two mirrors at either end which trap light between them – which serve to extend the distance traveled by the light for additional sensitivity. The entire system is vacuum-sealed, nitrogen-cooled, features multiple protections against seismic activity, and has a fence to defend against wind. The result is an instrument that can detect a change in the length of its arms to a small fraction of the size of a proton.

In general, LIGO scientists use frequency-domain data to identify the type of signal observed, so it is desirable to take the short fourier transforms (SFTs) of intervals of time-domain strain data to extract the individual frequencies that make up the overall waveform. It is necessary to split the time-domain into these sections, called TSFTs, due to the large computational cost required to perform a single SFT on the entire domain. The resulting frequencies all lie in small intervals called frequency bins whose length is the reciprocal of the TSFT length. As a result, the frequency-domain data is effectively resolution limited, with peaks only being able to be as narrow as a single frequency bin. This frequency-domain data is displayed as normalized power spectra with units of strain per square root hertz.

To identify transient events, LIGO scientists look for predictable signal templates in this data which increase in frequency and amplitude over time. If a CW were to be observed, it would remain near-single frequency for a very, very long period of time, and so it would appear as a narrow peak on the frequency spectra, gradually decreasing in frequency as it loses energy. However, GWs are not the only thing that appear on these frequency spectra. Various sources of noise such as electronics, thermal noise, acoustic effects within the instrumentation, and seismic activity can move the mirrors within the arm cavities, causing a false shift in the length of the cavity not caused by GWs. And not all light behaves the same way in the cavity due to scattering effects. To counteract these effects, LIGO scientists average their data over a certain duration of time, which greatly enhances the visibility of signals which are consistent over that period. Scientists working in Detector Characterization (DetChar) use a software suite called Fscan to perform this analysis. By default, Fscan produces SFTs with an 1800 second TSFT and returns two different plots: Frequency-domain normalized average power spectra.

However, some noise, called narrow spectral artifacts or lines, are highly consistent themselves. These sharp peaks, representing a persistent detection at a near-single frequency, pollute much of the frequency-domain spectra. This is problematic for transient searches, but especially for CW searches, as these narrow spectral artifacts are just as persistent as CWs are much higher in magnitude. Many CW search techniques rely on the persistence of a near-single frequency peak over a long period of time, so the presence of very strong, persistent noise peaks complicates that process greatly. Worse, some waves with a harmonic frequency will create narrow spectral artifacts at every multiple of that fundamental frequency. This spectral structure is called a comb.

Figure 1: Fscan daily summary on 2023-06-11 in 600-900Hz domain



Plot min frequency 585.00003
Plot max frequency 914.99942
Visual reference frequency 0.00000
Plot min vertical 3.911e-1
Plot max vertical 5.082e+1

Figure 1 shows a typical interactive summary plot from the Fscan pipeline, with an averaging duration of a full day and a TSFT length of 1800 seconds. Note the upper label of the spectrum H1:GDS-CALIB_STRAIN_CLEAN. This is the name of a channel which measures strain at Hanford. There are many channels which represent various sensors which measure electronic noise, magnetic noise, and others. Also notice the presence of narrow peaks in the spectrum. These represent the frequencies of waves which comprise the original waveform. All of these are actually lines. Some lines in Figure 2 are marked with a small shape; these represent the harmonic frequencies of a wave. All of the narrow spectral artifacts belonging to a harmonic frequency together are called a comb, and its lines are called teeth of that comb. We usually define a comb in terms of the frequency spacing between each of its teeth and its overall frequency offset.

Distinguishing between noise and actual signals is a highly computationally intensive and difficult task for the CW group which relies heavily on the DetChar group's success in correctly identifying and characterizing that noise. Typically analysts identify the source of narrow spectral artifacts by finding specific change points in their behavior. That is, when they appear and disappear. If a narrow spectral artifact of a frequency typically seen with seismic noise appears and disappears at the same time as a seismic event begins and ends, then we may say with confidence that the noise is seismic in origin. However, much of this noise is much harder to detect as it comes from instrumental noise in the detector, and so change points must be cross-referenced with changes made to the detector on particular dates and other arcane events. In the meantime, DetChar analysts attempt to keep a list of known lines at particular frequencies and keep data on when they appear and disappear for other groups to reference in their own analyses.

Between LIGO's third observing run (O3) and throughout the fourth beginning in May 2023 (O4) some combs were observed to shift slightly frequency over time. These combs, which we have called roaming combs, present a unique problem for searches as their shifting frequencies sometimes resemble the spin-down patterns of real sources. Their changing frequencies also make building and maintaining a list of lines difficult, as now that information may be outdated on a particular day of analysis. Further, when averaging over a long period of time, these roaming combs pollute an array of frequency bins rather than just a few. All of these problems make these combs highly problematic for CW searches.

But these combs changing in frequency over time actually provides a valuable opportunity to identify changepoints for these combs, and then possibly to identify what is causing them. To this end, we produced two pieces of software for displaying how roaming combs have changed in frequency over time. We then applied this software to every known comb in the H1:GDS-CALIB_STRAIN_CLEAN channel as they appeared in the daily Fscan summaries in O4. We confirmed that three combs – with spacings 9.474Hz, 9.4754Hz, and 9.480Hz – change in frequency over time and that they share directional changepoints, increasing from 05-24-2023 to 06-05-2023, then decreasing from 06-21-2023 to 06-23-2023. We hope that this software will be useful in identifying changepoints in and investigating the stability of other high-priority combs in the future.

Methods

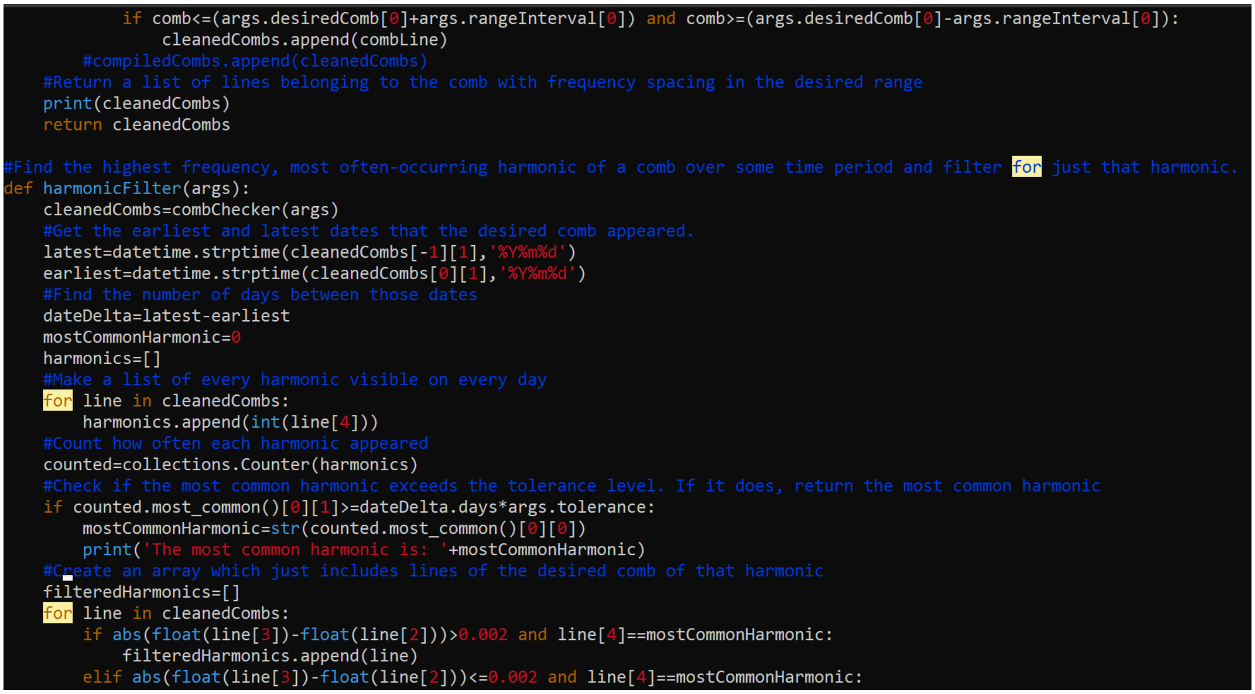
Fscan & Autolines-complete.txt

Fscan is a software suite used to produce frequency-domain short fourier transformed spectral data for a particular date epoch. The user specifies the averaging duration and TSFT length. For the daily spectra produced by Fscan automatically, it takes an averaging duration of 1 day and a TSFT length of 1800 seconds. Fscan then uses scripts to identify lines topographically by comparing each frequency bin's height against the height of neighboring bins. Finally, it attempts to assign fundamental frequencies to the newly-produced list of identified lines for the epoch by looking for groups of several lines that are spaced evenly throughout the spectrum. It stores all of the identified lines and combs in a file called Autolines-complete.txt before plotting all of its data. It also stores all the information about that particular run of Fscan in a dictionary object, including the segment type, the epoch start and end date, the averaging duration, and the path to the Autolines-complete.txt file. All of this information is very useful for the purposes of this project as it would allow the script I create to automatically search for the specified data it must run on.

combComparer.py

The first script I made is called combComparer.py. It takes a list of dates for a particular channel and averaging duration and reads the Autolines-complete.txt file for each day in the epoch. It makes a list of every comb identified by Fscan from every day. Then it takes user-supplied combs we wish to analyze and it filters the list so that just those combs remain. The heights of each line of the comb vary per day such that a particular line may not be identifiable above other noise while other lines of the comb are. So, it is necessary for the script to then filter for the line of each comb which appears the most over the epoch – That is, which line of the desired comb was high enough in magnitude to be identifiable as a line the most. If there is a tie, then it selects whichever most often-occurring line is higher in harmonic. Some of those lines occasionally are incorrectly assigned frequency offsets due to being incorrectly placed a frequency bin too far – usually because that line lied exactly on the border of two frequency bins – so in that case combComparer.py will subtract that offset. What we are left with then is a list of lines belonging to a particular comb and their frequencies on the date they occurred, which then can be plotted as a simple scatterplot of frequency over time or displayed as a table.

Figure 2: Code snippet from combComparer.py



```
            if comb<=(args.desiredComb[0]+args.rangeInterval[0]) and comb>=(args.desiredComb[0]-args.rangeInterval[0]):
                cleanedCombs.append(combLine)
        #compiledCombs.append(cleanedCombs)
    #Return a list of lines belonging to the comb with frequency spacing in the desired range
    print(cleanedCombs)
    return cleanedCombs

#Find the highest frequency, most often-occurring harmonic of a comb over some time period and filter for just that harmonic.
def harmonicFilter(args):
    cleanedCombs=combChecker(args)
    #Get the earliest and latest dates that the desired comb appeared.
    latest=datetime.strptime(cleanedCombs[-1][1],'%Y%m%d')
    earliest=datetime.strptime(cleanedCombs[0][1],'%Y%m%d')
    #Find the number of days between those dates
    dateDelta=latest-earliest
    mostCommonHarmonic=0
    harmonics=[]
    #Make a list of every harmonic visible on every day
    for line in cleanedCombs:
        harmonics.append(int(line[4]))
    #Count how often each harmonic appeared
    counted=collections.Counter(harmonics)
    #Check if the most common harmonic exceeds the tolerance level. If it does, return the most common harmonic
    if counted.most_common()[0][1]>=dateDelta.days*args.tolerance:
        mostCommonHarmonic=str(counted.most_common()[0][0])
        print('The most common harmonic is: '+mostCommonHarmonic)
    #Create an array which just includes lines of the desired comb of that harmonic
    filteredHarmonics=[]
    for line in cleanedCombs:
        if abs(float(line[3])-float(line[2]))>0.002 and line[4]==mostCommonHarmonic:
            filteredHarmonics.append(line)
        elif abs(float(line[3])-float(line[2]))<=0.002 and line[4]==mostCommonHarmonic:
```

It is necessary to do all of this instead of just looking at the comb's frequency spacing because the value of frequency given for the comb's spacing is a fitted value subject to rounding. When I first began my project, I began by just looking at the comb's spacing on each day in the epoch.

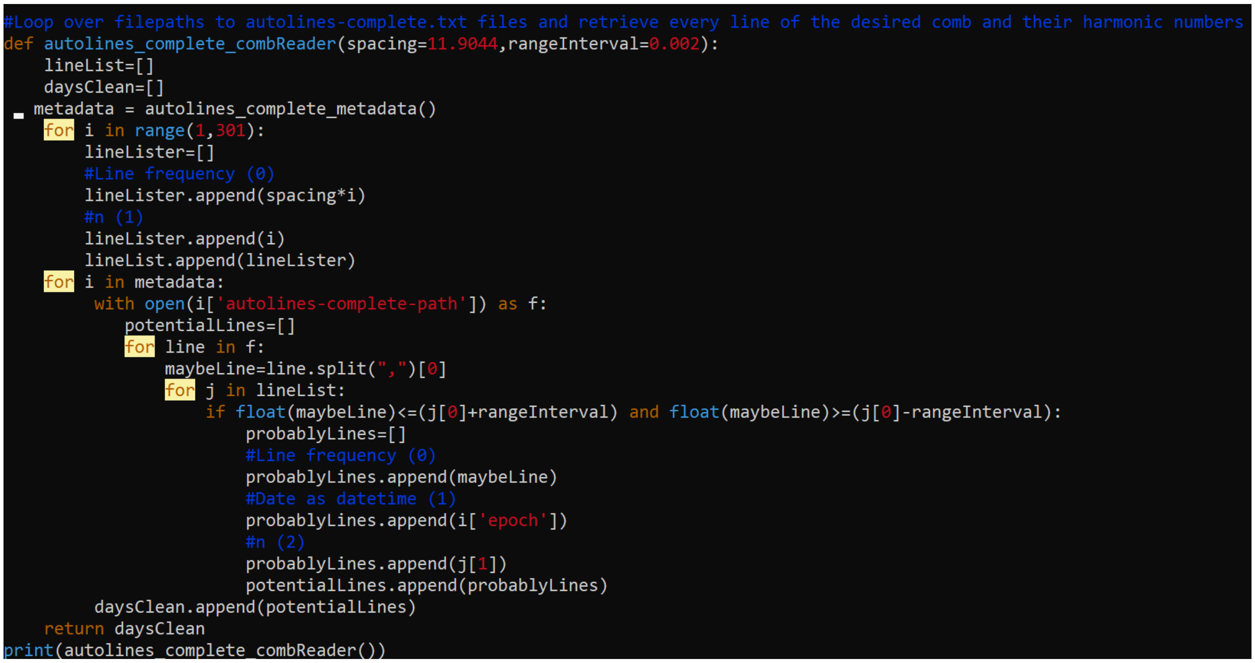
While it was more straightforward, the fitted value for spacing is rounded – It does not have the same precision as looking at a particular line's frequency in its frequency bin, even if that value too is resolution limited.

The script combComparer.py was built with the intention that it could be integrated into the daily Fscan pipeline. By looking at the combs identified on the day Fscan was run for, combComparer.py can look back and quickly display how each of those combs have changed over O4 up to that day. Or it can say how much each comb has changed since its last appearance, detecting change points as they occur. It is also functionally modular such that if a user just wanted all of the combs for a particular day but did not want to filter for any particular comb or plot it, the user can supply parameters to only run the script up to that function, which may be useful in future analysis which require other scripts.

manualComparer.py

Some combs are not detected by the Fscan pipeline even though they are known to exist. For these combs, it was necessary to create a second script, manualComparer.py, which takes a frequency spacing as input and attempts to find lines which are multiples of that spacing within some very small allowed percentile uncertainty (as we expect a frequency shift over time). It checks if every line from every day in the epoch is within a small uncertainty around the target spacing or a higher multiple of it, *n*, the harmonic number. If it finds a line of the desired comb spacing successfully, it appends it to a list along with its harmonic number.

Figure 3: Code snippet from manualComparer.py



```
#Loop over filepaths to autolines-complete.txt files and retrieve every line of the desired comb and their harmonic numbers
def autolines_complete_combReader(spacing=11.9044,rangeInterval=0.002):
    lineList=[]
    daysClean=[]
   metadata = autolines_complete_metadata()
    for i in range(1,301):
        lineLister=[]
        #Line frequency (0)
        lineLister.append(spacing*i)
        #n (1)
        lineLister.append(i)
        lineList.append(lineLister)
    for i in metadata:
         with open(i['autolines-complete-path']) as f:
            potentialLines=[]
            for line in f:
                maybeLine=line.split(",")[0]
                for j in lineList:
                    if float(maybeLine)<=(j[0]+rangeInterval) and float(maybeLine)>=(j[0]-rangeInterval):
                        probablyLines=[]
                        #Line frequency (0)
                        probablyLines.append(maybeLine)
                        #Date as datetime (1)
                        probablyLines.append(i['epoch'])
                        #n (2)
                        probablyLines.append(j[1])
                        potentialLines.append(probablyLines)
         daysClean.append(potentialLines)
    return daysClean
print(autolines_complete_combReader())
```

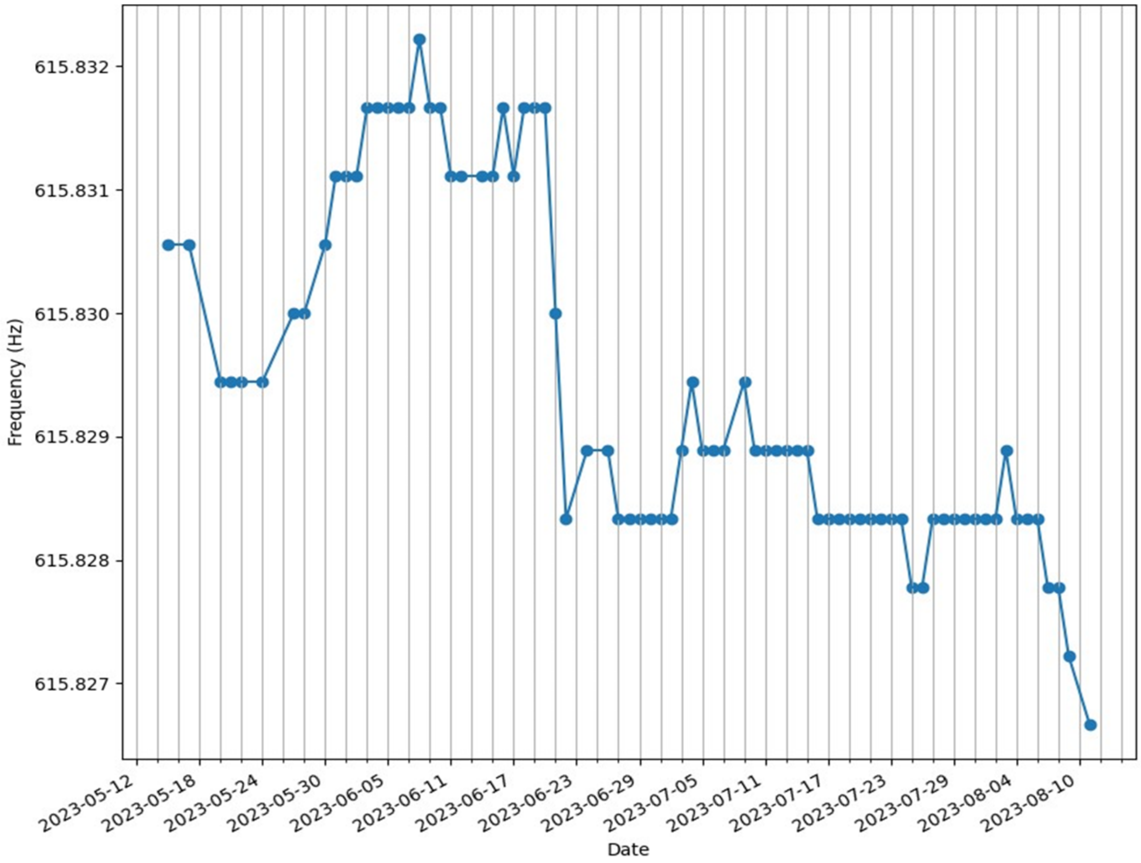
Results

The 9.474Hz, 9.4754Hz, 9.480Hz, 4.98423Hz, 29.96518Hz, and 1.6611Hz combs were investigated by running combComparer.py for a desired spacing supplied by the list of high-priority combs identified by the DetChar group. The 11.9044Hz comb was investigated using combComparer.py as it is often undetected by Fscan. All lines were retrieved from the daily Fscan data (TSFT 1800s) from the H1_GDS-CALIB_STRAIN_CLEAN channel. Graphs which showed no change were confirmed by printing the values plotted – Matplotlib automatically defaults to a large y-axis scale if the slope of the scatter plot is zero. A point on the scatter plot being missing for a particular day means that the tracked line of this particular comb was not present on that day.

Figure 4: Combs investigated

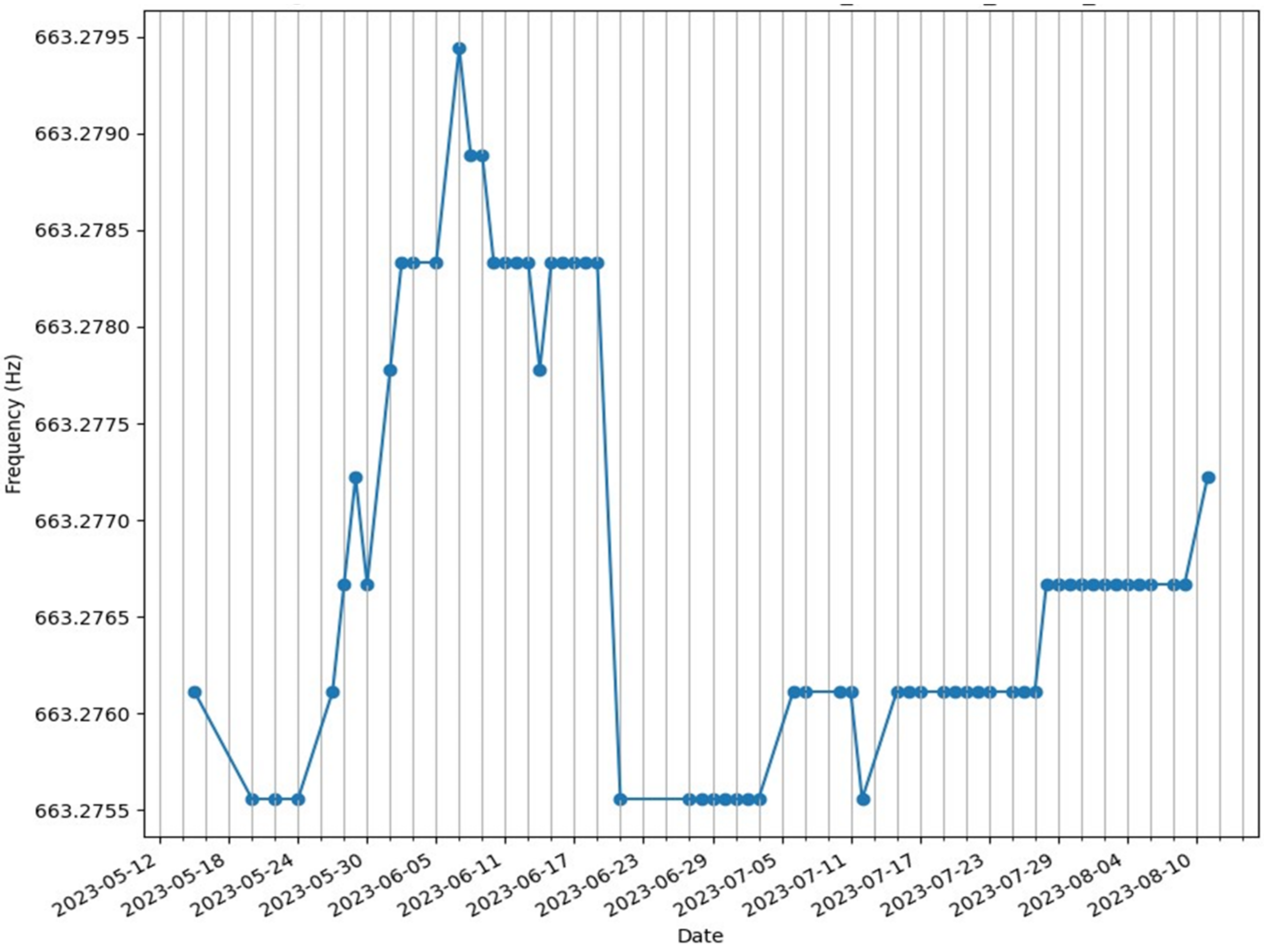
Comb spacing (Hz)	9.474	9.4754	9.480	4.98423	29.96518	1.6611	11.9044
Roaming in frequency?	Yes	Yes	Yes	No	No	No	No

Figure 5: 9.474Hz spacing, n=65, Significant change points



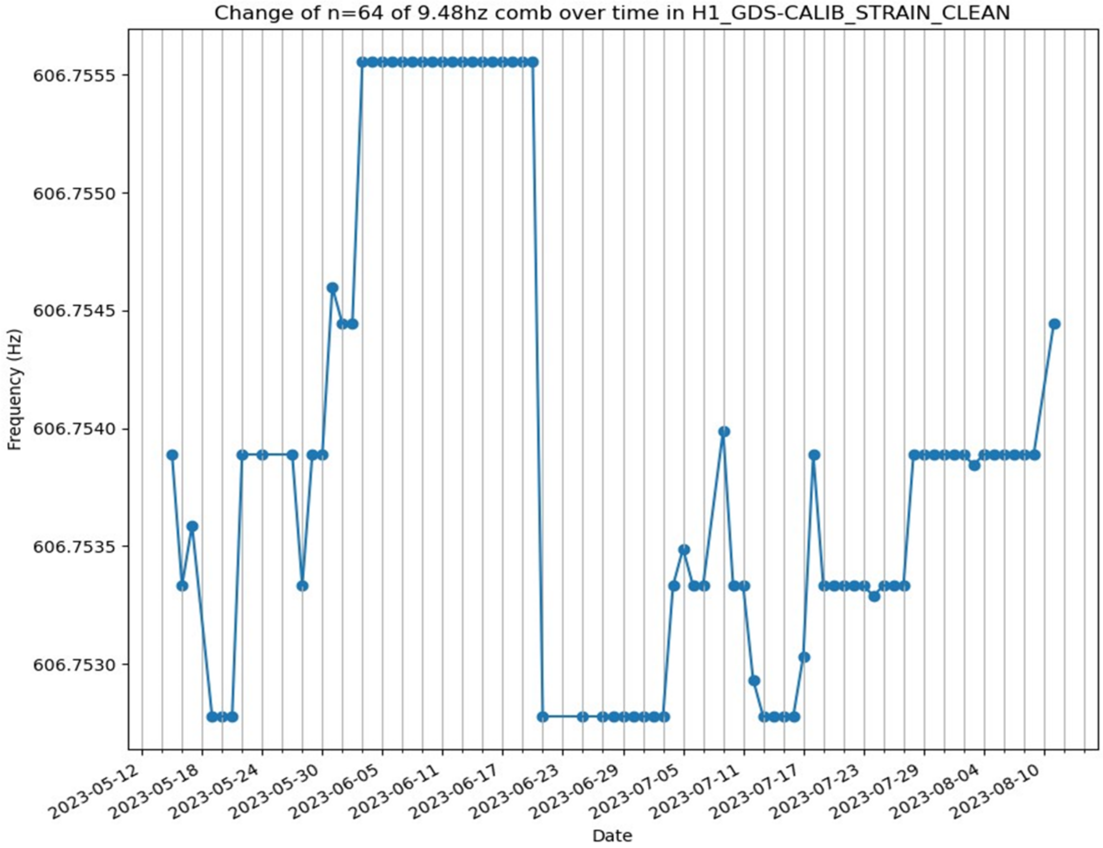
The 9.474Hz comb increases from 05-24 to 06-05 and sharply decreases from 06-22 to 06-24. We will see that these change points are shared by the 9.474Hz and 9.480Hz combs. It is worth noting that these changes are on the order of 10^-6, which is a large enough change for the comb's lines to roam between frequency bins.

Figure 6: 9.4754Hz spacing, n=70, Significant change points



The 9.4754Hz comb shares similar change points with the 9.474Hz and 9.480Hz combs of a sharp increase in frequency from 05-26 to 06-05 and a sharp decrease from 06-21 to 06-23.

Figure 7: 9.480Hz spacing, n=64, Significant change points



In the 9.480Hz comb we see a sharp increase from 05-30 to 06-05 and a sharp drop from 06-22 to 06-23.

Figure 8: Changepoint dates of the 9.474Hz, 9.475Hz, and 9.480Hz combs

Frequency (Hz), n	Increase dates	Decrease dates
9.474, n=65	05-24 to 06-05	06-22 to 06-24, 08-06 to 08-18
9.4754, n=70	05-26 to 06-05, 07-29 to 08-11	06-21 to 06-23
9.480, n=64	05-30 to 06-05, 07-29 to 08-11	06-22 to 06-23

Specific attention should again be drawn to the shared dates of increase and decrease, and then also to the similar harmonic. These three combs' most often-visible harmonics all appear in the 600-900 frequency band and share significant changepoints in their shifts upward and downward in frequency. It is clear for certain from the data that these three combs are related, and those dates likely correspond to instrumental changes to the detector. It appears, for example, that all three combs decreased in frequency on the same week that the laser power was reduced back to 60W.[3]

Figure 9: 4.98423Hz spacing, n=11, Constant Frequency

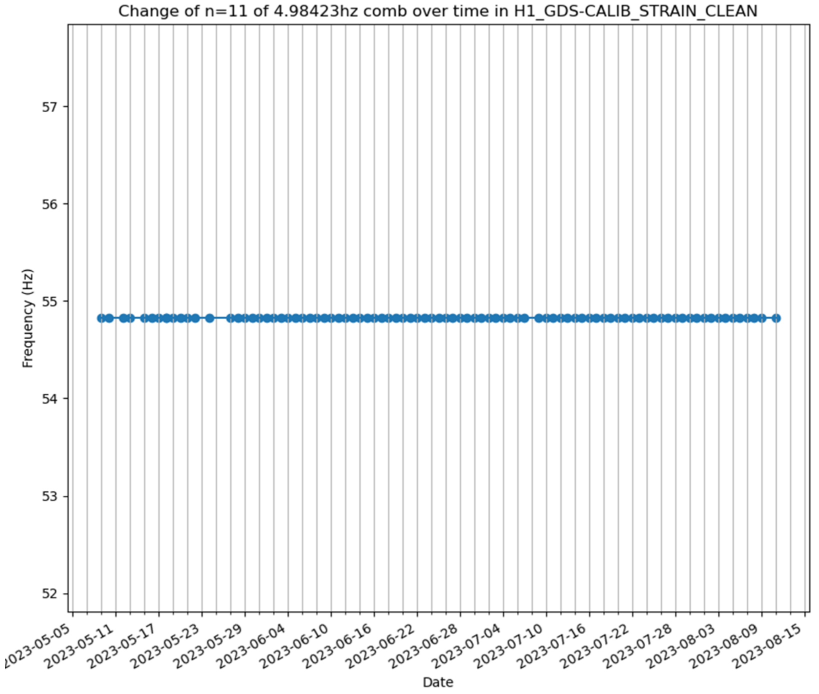


Figure 10: 29.96518Hz spacing, n=13, Constant Frequency

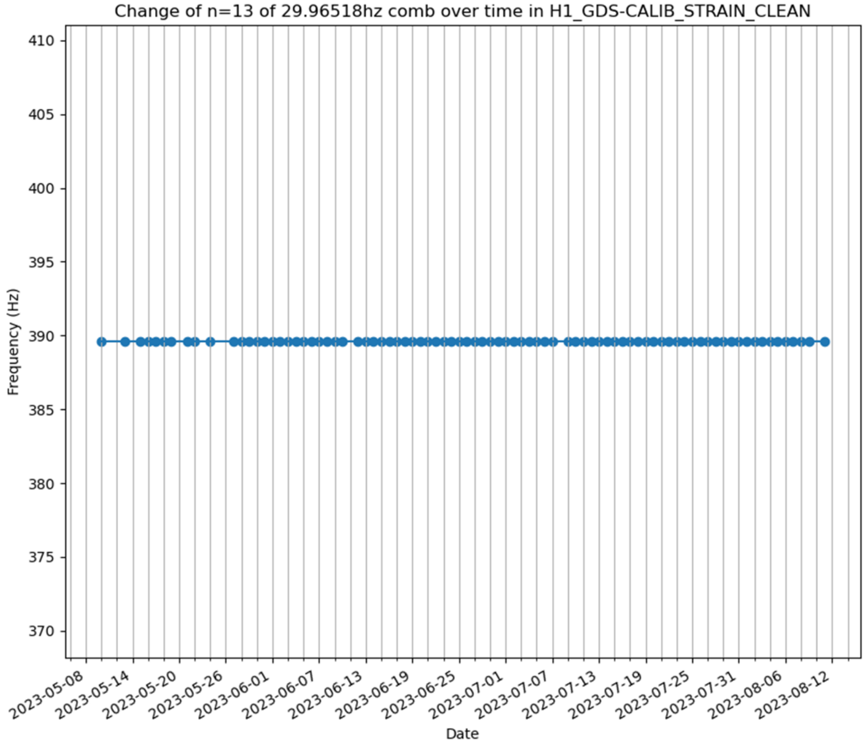
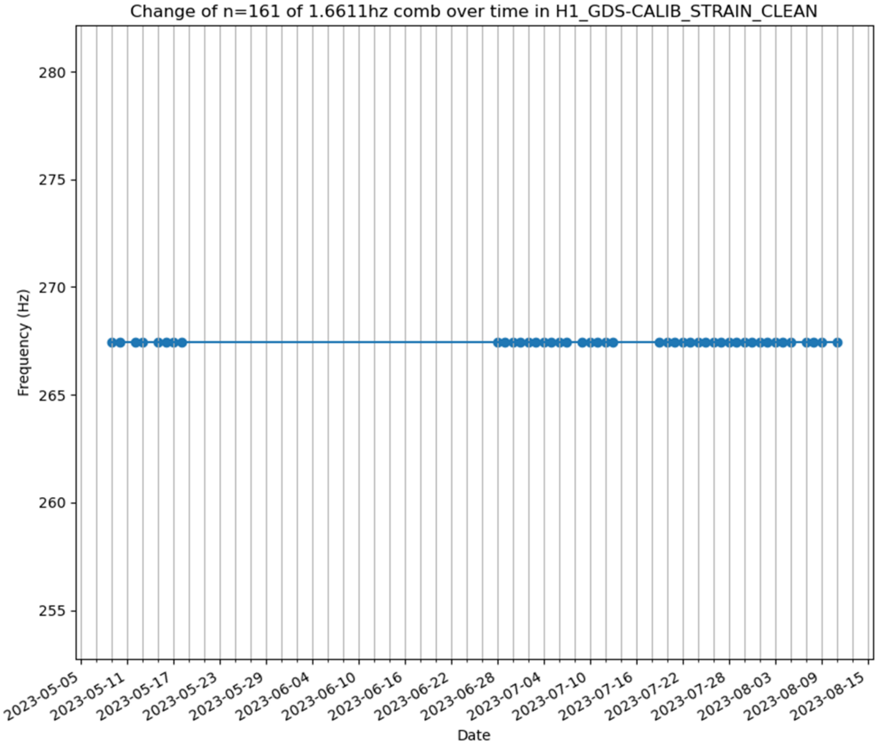
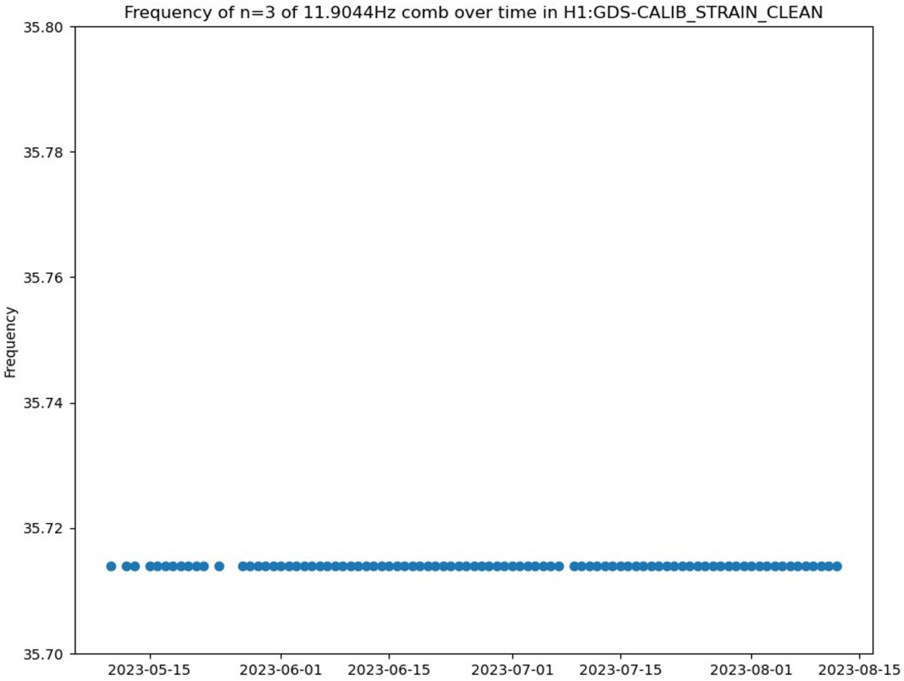


Figure 11: 1.6611Hz spacing, n=161, Constant Frequency



It is worth noting that this comb was found to be caused by the OM2 Beckhoff Cable.[4]

Figure 12: 11.9044Hz spacing, n=3, Constant Frequency



The 4.98423Hz, 29.96518Hz, 1.6611Hz, and 11.9044Hz combs do not roam outside of their frequency bins. This is a surprising finding as the 11.9044Hz comb was observed to change in frequency between O3 and O4, but its lines appear to never shift outside of its frequency bins in O4. But we know now that this comb is much more stable in frequency than previously thought. In order to investigate these combs' frequency shift at a higher spectral resolution, Fscan would need to be run at a higher TSFT than 1800s. Unfortunately, we are limited for now by the heavy computational cost of performing short fourier transforms over a longer TSFT, so this is not possible. Yet we still can observe frequency shifts at a TSFT of 1800s in the 9.474Hz, 9.4754Hz, and 9.480Hz combs, so we can somewhat confidently say that if the other combs are shifting, it is a negligible amount

Conclusion

This project introduced a method for looking at the frequency-varying behavior of combs for the purposes of detector characterization, and it provided useful tools for performing that work. Future researchers should continue to use these tools or their own to explore other novel ways to characterize artifacts. We hope that the identification of changepoints for the 9.474Hz, 9.4754Hz, and 9.480Hz combs will assist in their characterization and removal, and that combComparer.py and manualComparer.py will assist in similar analysis in the future, especially by aiding the search for continuous waves.

Acknowledgements

I would like to sincerely thank the staff at LIGO and the other wonderful SURFs for this amazing opportunity. I would like to thank my mentor Ansel Neunzert for their incredible patience and support with me, Brina for her very generous work onboarding myself and the other SURFs which made us feel so welcome, Mike Landry and Eadie Balint for their outstanding professionalism and assistance as we proceeded through the summer, Rachel McQueen and Caden Swain for being the most fun roommates and coworkers I could ask for and learn from, Taylor Starkman and Julianna Lewis for all their help, Louis Dartez for chaperoning us for our visit to Caltech and for his career guidance, Kim Burtnyk and Oli Patane for their help finding the LIGO Hanford cat Bug, and the NSF for funding our work.

References

1. Thorne, Kip S. (1980). Multipole Expansions of Gravitational Radiation. *APS*. https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.52.299
2. Riles, Keith. (2023). Searches for Continuous-Wave Gravitational Radiation. *Living Reviews in Relativity 26*. https://arxiv.org/abs/2206.06447
3. Sigg, Daniel. (2023). Interferometer Mode Matching and Reflectivity. *aLIGO LHO logbook.* https://alog.ligo-wa.caltech.edu/aLOG/index.php?callRep=70985
4. Neunzert, Ansel. (2023). 1.6611 Hz comb gone after OM2 Beckhoff Cable Disconnected. *aLIGO LHO logbook*. https://alog.ligo-wa.caltech.edu/aLOG/index.php?callRep=72269

We would like to thank the LIGO Laboratory and Caltech Student Faculty Programs office for the opportunity to participate in this SURF program and for their support throughout the summer. We thank the NSF REU program for their support.