Characterization of the Omicron Trigger Generator and Transient Analysis of aLIGO Data

Hunter Gabbard*

Virgo Detector Characterization Group, Laboratoire de l'accelerateur lineaire
Centre scientifique d'Orsay, Batiment 200, Orsay, France
Department of Physics and Astronomy, University of Mississippi
University, Mississippi 38677-1848 USA
Department of Physics, University of Florida
Gainesville, FL 32611-8440 USA
hagabbar@go.olemiss.edu

August 12, 2014

Abstract

Omicron is a burst-type trigger generator. We performed coincidence tests between Omicron generated triggers of ER5 LIGO data and various types of injection waveforms (Sine-Gaussian, White-Noise-Burst, and String Cusp) using a Coincidence Finder program that we developed. Through these tests we determined the efficiency at which the Omicron trigger generator is able to detect specific transient events with varrying sets of parameters. We tested and debugged a new version of Omicron and utlized Omicron to perform a close analysis of lock time data at the gravitational wave detector in Livingston, Louisiana. From this analysis we were able to classify noise events and determine several of their sources.

I. Introduction

Einsteins Theory of General Relativity predicts the existence of gravitational waves that are small perturbations in the fabric of space-time given in the Minkowski Metric $g_{\mu\nu}$[1]

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}, \tag{1}$$

Where $\eta_{\mu\nu}$ is the Minkowski metric for flat space-time and $h_{\mu\nu}$ is a small purturbation in space-time. Possible sources of gravitational waves could result from supernovae explosions, asymmetries of spinning neutron stars, binary neutron star systems, binary black hole systems, and several other sources. Some energy is released from these systems in the form of gravitational waves which propogate outward from the system at the speed of light. These gravitational waves (GW) produce a strain on free-falling test masses that is miniscule in amplitude. For example, If we consider a pair of 10 $M_\odot$ black holes revolving around each other at 200 Mpc, we see that the strain amplitude that they produce is equivalent to $h \sim 1 \times 10^{-24}$, where h is the amplitude of the GW[1]. This is an extremely small change in distance as we are trying to measure such changes on the order of 1/1000th the diameter of a proton. To measure these minute changes in length we utlize a larger and enhanced version of the Michelson-Morley Interferometer.

The Laser Interferometer Gravitational-Wave Observatory Scientific Collaboration consists of two large interferometric gravitational

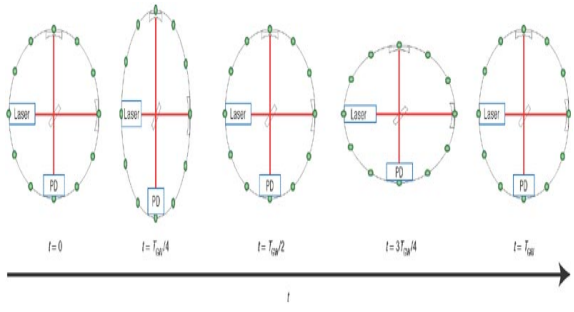
*Advisor: Dr. Florent Robinet

wave detectors, each with an arm length of ∼4km. One is stationed in Hanford, Washington and the other in Livingston, Louisiana. These two interferometers are a part of a larger worldwide detector network with sites in Hannover (GEO 600), Pisa (Virgo), and Tokyo (KAGRA).

A beam of coherent light ($\sim 1064nm$) is produced and sent through an input mode cleaner (IM), which is an in-vaccuum suspended triangular optical cavity. The IM 'cleans' the laser beam by minimizing directional and geometric fluctuations, as well as providing frequency stabilization[2]. The coherent light then travels through a beam spliter and both beams enter a Fabry-Perot cavity. This cavity amplifies the power of the laser as well as the distance traveled. These two beams are then reflected back and forth. The two beams then meet again at the beam splitter where, if a gravitational wave does not pass through the system, they combine destructively to produce a dark fringe.

When a gravitational wave passes through a system the wave plus-polarizes and cross-polarizes a set of particles in a plane.

Figure 1: *Example of plus polarization*



When this happens there is a relative difference in arm length between the x and y arms, as the end test masses act as these free particles. Thus the two coherent light beams traveling through the interferometer have different path lengths due to stretching and contracting of spacetime along the x and y arms. We measure the path difference in the laser on the photo detector when the two seperate beams interfere at the beam splitter.

We measure the strain amplitude time-series $h(t)$ that is induced on the interferometer from gravitational waves in gravitational wave (GW) channels

$$h(t) = F_+ h_+(t) + F_\times h_\times(t), \tag{2}$$

Where F_+ and $F_\times$ are the attenna patern functions, which depend on the sky position of the GW source, while h_+, $h_\times$ represent the plus and cross polarization of a gravitational wave respectively[1]. Sometimes, external forces cause unwanted disturbances within GW channels that can mimic a gravitational wave signal, we call these *transients* or *glitches*. At the LIGO-Virgo Collaboration (LVC) there is a group solely dedicated to the detection and classification of these transients, the detector characterization group (detchar). In the upcoming sections I will discuss the intent and purpose of the detchar group, as well as my research in characterization of the Omicron trigger generator and LIGO subsystems.

II. Detector Characterization

The purpose of the detchar group is to identify non-astrophysical instrumental and/or environmental disturbances in the interferometer gravitational-wave observatories. The detchar group sometimes performs a process called "noise hunting" where we isolate each noise source and determine a coupling to other channels if there is any.

Both LIGO and Virgo utilize hundreds of different sensors, called auxiliary channels, that measure these external disturbances. These channels include, but are not limited to, accelerometers, microphones, seismomitors, and voltage monitors. We use auxiliary channels to determine if a candidate event in a GW channel is really a gravitational wave, or just a glitch coupled with an external disturbance seen in the auxiliary channels.

The detchar group identifies events with similar properties (e.g. same frequency band). We see if an event is correlated with some environmental or physical disturbance and check the event times with external scheduled events.

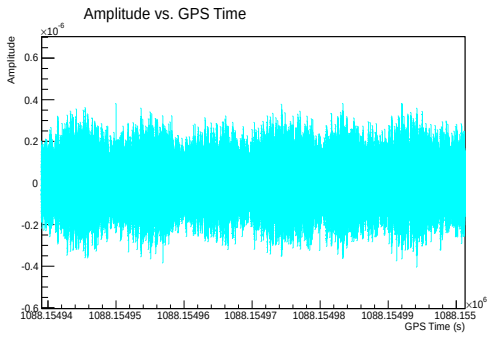
Then investigations are made to figure out if an event in one channel is coincident with an event in other auxiliary channels by using various statistical algorithms such as the use-percentage-veto algorithm (UPV). If we can identify the source of the noise we attempt to reduce/mitigate the noise source. If we are not able to understand the source of the noise we must veto the corrupted data by using data quality flags[3]. There are several pipelines that are used to identify "glitches" (Kleine Welle, PCAT, dmt_Omega, Omicron). Some pipelines are considered better than others in specific areas. For example, Kleine Welle performs well at high frequency bands, but poor in low frequency bands. So, it is essential to understand which pipeline is the most useful to detchar investigations. In the upcoming section I will explain my work in understanding the strengths and weaknesses of the Omicron trigger generator, as well as the role it will play in the future for the LIGO/Virgo Collaboration.

III. Characterization of the Omicron Trigger Generator

I. How Does Omicron Work?

Omicron is a burst-type trigger generator that reads raw data from given LIGO subsystem channels specified by the user at a given sampling frequency or working frequency (must be a power of 2).

Figure 2: *This figure is taken from the raw data time series in the DARM channel, this is the main interferometer output signal for GW detection.*



The pipeline loads data by chunks and then breaks these chunks into segments to be analyzed. These segments have a overlap defined by the user (usually 2s) so as to avoid edge effects. The power spectrum density for each chunk of data is then computed. The data vector for each segment is fourier-transformed and normalized using the PSD. Data is projected onto a paramter space which is tiled in three demensions, time, frequency, and Q planes. Q represents the quality factor (how gaussian is the waveform). The amount of Q planes are defined by a mis-match threshold that is set by the user. The mis-match threshold is the loss in energy between tiles, which can be no greater than 20 percent; this also determines the resolution of the tile. Frequency is then logarithmicaly distributed in rows which are then linearly distributed over time. The amplitude signal-to-noise ratio (SNR) is then computed for each tile. Omicron produces *triggers* which can be defined as a tile with an energy above a given threshold[4].

II. Coincidence Finder Programs

Seeing as one of the main purposes of the detchar group is to classify *glitch* families, it is advantageous to determine how well Omicron identifies and characterizes discrete types of simulated burst wave forms. To do this I developed several codes to find coincidence between a set of discrete injections (sine-gauss, white noise burst, string cusp). Injections are produced by the LIGO toolset lalapps_binj and triggers are produced by Omicron. We utilized an eight hour long stretch of science data from the Hanford Interferometer.

The Coicindence Finder program is written in a C++ format and utilizes ROOT/GWOLLUM libraries. Coincidence Finder reads injection files produced by lalapps_binj. Lalapps_binj is a program that creates a set of randomly distrubted injection paramters for a specific type of injection. Coincidence Finder stores the relavant parameters from the injection file in vectors. Coincidence Finder performs checks to verify that injection

parameters (amplitude (hrss) range, frequency, time) are within Omicron search range. Triggers are read and clustered in a .root format and trigger variables are stored in TTree containers. Time coincidence testing is then performed between Omicron triggers and injections based upon a pre-defined coincidence window. Coincidence Finder will define a coincident event if a trigger and an injection *overlap* in time. If there are several matches within the window, we take the first injection in time, thus preventing any bias in time.

To quantify the performance of Omicron several figures of merit are computed in Coincidence finder.

$$\epsilon_{hrss} = \frac{N_{det}}{N_{tot}}, \tag{3}$$

Where ϵ_{hrss} represents the detection efficiency as a function of amplitdue, N_{det} are the injection events found to be coincident by Coincident Finder, and N_{tot} are the total amount of injection events.

$$\epsilon_{freq} = \frac{N_{detfreq}}{N_{totfreq}}, \tag{4}$$

Where ϵ_{freq} represents the detection efficiency as a function of injection frequency, $N_{detfreq}$ is detected injection event at a given frequency, and $N_{totfreq}$ is the total amount of injection events with given frequency values.

$$\Delta_{peak} = (t_{trig} - t_{inj}), \tag{5}$$

Where Δ_{peak} is the peak time or change in time between Omicron trigger and injection time, t_{trig} is the trigger time, and t_{inj} is the injection time.

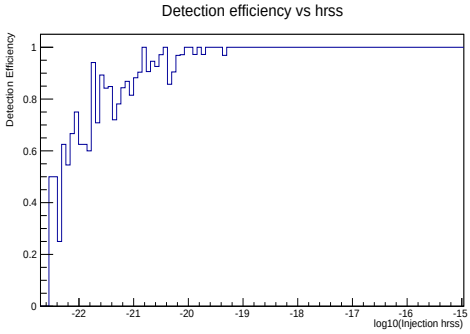
$$\Lambda_{asy} = \frac{((log10(f_{trig}) - log10(f_{inj}))}{(log10(f_{trig}) + log10(f_{inj}))}, \tag{6}$$

Where Λ_{asy} is the symmetry of the frequency reconstruction, f_{trig} is the trigger frequency, and f_{inj} is the injection frequency. It is important to explain, in regards to the upcoming investigations, a phenomenon called *random coincidence*. This phenomenon results from a statistical artifact in the Coincidence Finder program, whereby random triggers found by Omicron at low SNR are found to be in coincidence with injections (within overlap window) made by lalapps_binj.

III. Sine-Gaussian Injections

We first test how robust Omicron is in retrieving all-sky sine-Gaussian injections. Sine-Gauss injections are dependent upon a Q value which determines the width of the waveform, a frequency value, and an hrss value (amplitude).

Figure 3: *Efficiency curve for sine-Gaussian injections with a Q of 75 as a function of the log10(hrss).*



We see in figure 3 that Omicron is able to detect sine-Gaussian injections starting at an hrss of 10^{-22}, quite a promising result. However, as we can see in figures 4 and 5, as the Q value decreases it appears the detection effciency of Omicron decreases as well.

Figure 4: *Efficiency curve for sine-Gaussian injections with a Q of 30 as a function of the log10(hrss).*

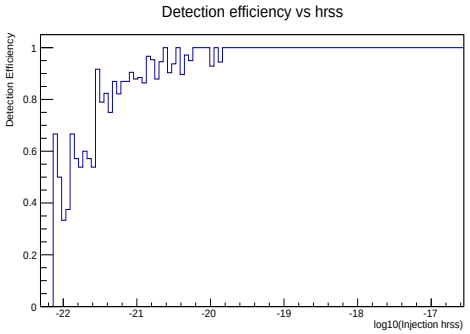
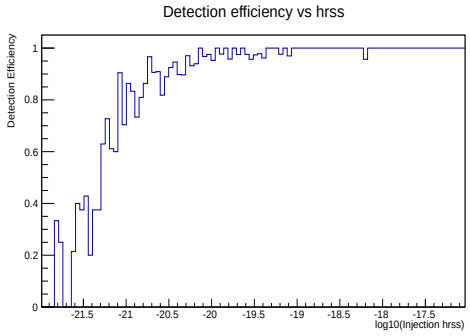
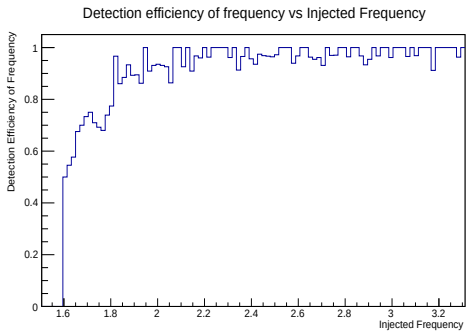


Figure 5: *Efficiency curve for sine-Gaussian injections with a Q of 5 as a function of the log10(hrss).*



Over the course of our investigations we found that this discrepency in the Q factor was correlated to an issue with the detection efficiency as a function of the injected frequency. Because Omicron does not determine a trigger based on its inherant frequency, there should be no relationship between the detection efficiency of Omicron and the injection frequency. However, as we will see in figure 6 below, this is not case.

Figure 6: *Sine-Gaussian effciency curve as a fucntion of injection frequency for a Q of 5.*



We concluded that this trend is due to a conversion done in the injection program we were utilizing. The injection program "lalapps_binj" takes the input hrss range the user gives in the command prompt, and converts this to a new hrss. This is done to make injections lie along the 50 percent efficiency curve better[5]

$$hrss_{true} = \sqrt{2} \times \pi \times hrss_{inj} \times (f/Q). \quad (7)$$

Where the $hrss_{inj}$ represents the converted hrss, $hrss_{true}$ the inputed hrss, f the injection frequency, and Q the quality factor, which is related to the width of the waveform by the relation below

$$\lambda_w = \frac{2\pi}{Q^2}. \quad (8)$$

Where λ_w is the width of the waveform, and Q is the quality factor. So we see that the total overal amplitude of the injections decreases with a small f and a large Q, and vice versa. The quality factor relationship is illustrated well in figure 7 below where we see that the overal amplitude shifts as Q changes.

Figure 7: *Number of injections as a function of amplitude. Q of 75 is in black, Q of 30 in blue, and Q of 5 in red.*

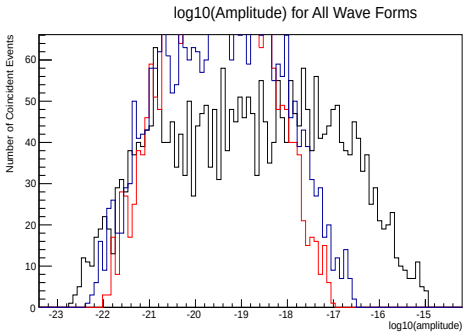
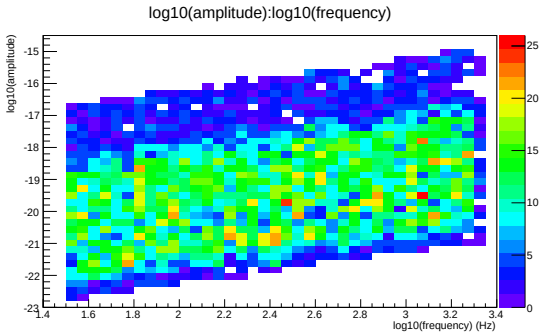


Figure 8: *Plot of Q=75, Q=30, and Q=5 coincident triggers. This illustrates the linear relationship between amplitude and frequency.*



In figure 8 we see that there is also a linear relationship with amplitude as a function of

frequency in the injection parameters. We corrected both of the issues seen in figures 7 and 8 by first adding a step in Coincidence Finder to convert back to true hrss. Then, to unbias our injection sample we weighted injections so that those with a high amplitude in low frequency bins were favored more than injections with a high amplitude in high frequency bins.

$$W_{inj} = hrss_{inj}, \tag{9}$$

Where W_{inj} represents the weight given to a particular set of injections, and $hrss_{inj}$ represents the injected hrss.

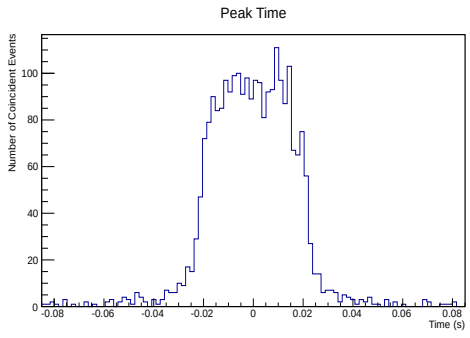
$$\epsilon = \frac{\sum_{i=0}^{N_{det}} hrss_{inj}}{\sum_{i=0}^{N_{inj}} hrss_{inj}}, \tag{10}$$

Where ϵ represents the efficiency, $\sum_{i=0}^{N_{det}} hrss_{inj}$ is the sum of the detected injections at a given amplitude, and $\sum_{i=0}^{N_{inj}} hrss_{inj}$ the sum of total injections at a given amplitude.

Thus, we can confidently say that Omicron detection effciency is not related to the Q value; or frequency value, the trend we saw is instead an artifiact of the injection program lalapps binj and Coincidence Finder.

As mentioned previously in Eq(5), we calculated the peak time for each injection type.

Figure 9: *Plot showing peak time for Sine-Gaussian Q of 30.*

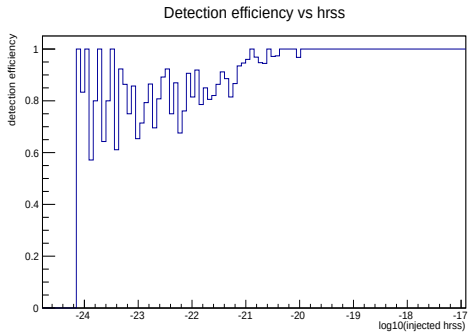


The peak time is calculated by subtracting the trigger time from the injection time to find out how close in time Omicron retrieves injections. As we can see in Figure 9 the distribution is centered around 0. However, according to Coincidence Finder, Omicron detects injections between $+20ms$ and $-20ms$. Ultimately we determined this is not an issue associated with Omicron, but instead one related to Coincidence Finder. When an injection is made it is randomly given a sky location. The injection time is then referenced by the time at the center of the earth. We calculated that the time it takes for a gravitational wave to propagate from the center of the earth to the surface was $\sim 20ms$. Thus, depending upon the sky location the time at which Omicron detects the injection at the surface of the earth will vary between $+-20ms$. In the future we should add arguments to Coincidence Finder to compensate for this discrepency.

IV. Bound Limited White Noise Burst Injections

White noise burst injections are dependent upon an equivalent isotropic radiated energy (solar mass per parsec squared), duration of injection (seconds), and the bandwidth of the injection (Herz). We first measured the detection efficiency as a function of hrss.

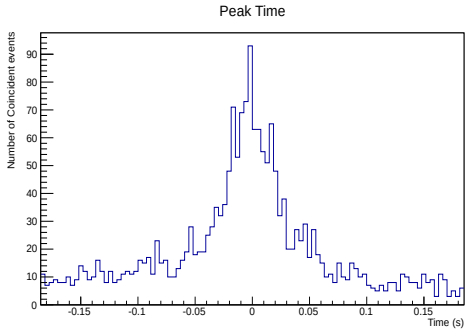
Figure 10: *Efficiency vs. hrss curve.*



The poor efficiency of the white noise burst (wnb) injections can be attributed to the units that are associated with them (equivalent isotropic radiated energy). Because we did

not know what value the injection generator program would produce for a given isotropic radiated energy, we had to guess and check with our amplitude ranges. At low value injection amplitudes ($\sim 10^{-24}$) there is a 100 percent detection efficiency. This is due to a statistical artifact where there are only one or two injections made at that injection amplitude and Coincidence Finder happens to find random coincidence.

Figure 11: *Peak Time of White Noise Burst injection.*



In figure 11 we see that again there is an excellent reconstruction of the Omicron trigger time around the injection time. However, we notice that there is an increase in the overall background noise ~ 10. This anomaly is related to the duration of white noise burst injections. Since white noise bursts can have a duration of up to 2 seconds we initially decided we had to increase the size of the injection window. However, because the window for possible coincidence was much larger, there was a large increase in random coincidence found by Coincidence Finder. So, we changed the requirements for coincidence to determine that there was a match if an injection event and trigger event *overlap* in time.

Figure 12: *Reconstruction of frequency symmetry found by Coincidence Finder*

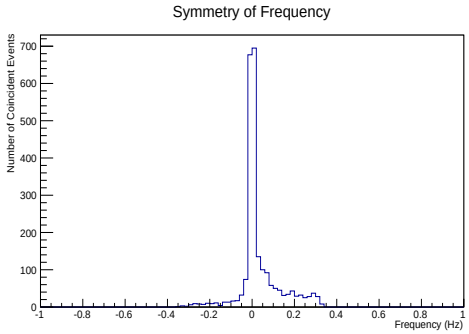
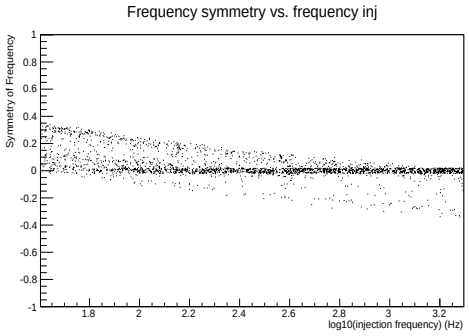


Figure 13: *Symmetry of Frequency as a function of injection frequency.*

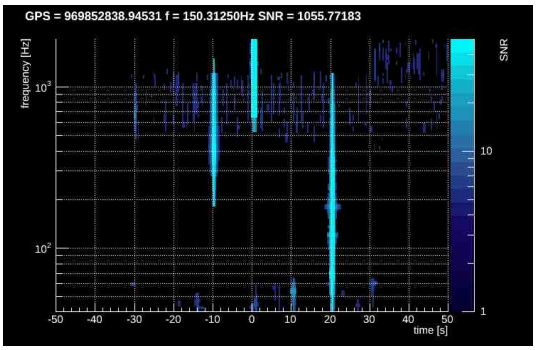


While investigating the frequency reconstruction of the Omicron trigger generator we came across a recurring problem in all injection types. As we can see in figure 12, not all injections are detected at their "true" injection frequency. Instead, we see that there are a large amount of injections detected at higher frequency than they actually are while there are also a handful detected at a lower frequency than the "true" injection frequency. This trend should not exist since Omicron does not determine the significance of a trigger based upon its central frequency value.

In figure 13 we note that as the central injection frequency increases the asymmetry in coincident triggers decreases. At a certain injection frequency, the asymmetry flips and becomes negative. We believe that this is not the fault of Omicron, so we took a closer look at the lock data that we were using. It turns out that

in the H1:LDAS-STRAIN channel during this time chunk there is an excess of noise around $1KHz$. Due to this set of transient events, injections with a central frequency value lower than that of this transient family would be found to be randomly coincident with said transient series at a higher value and *vice versa*. This is clearly illustrated in figure 14 where we see the faint structures of this "glitch" family.

Figure 14: *Frequency-time decomposition of 100 second time window plotting frequency as a function of time with a verticle scale in SNR.*



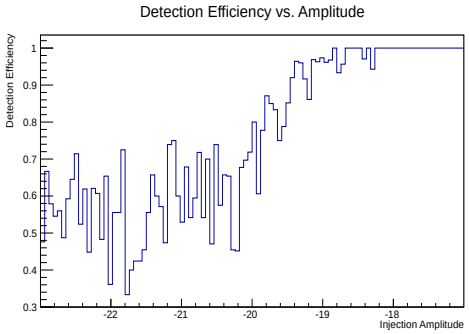
V. String Cusp

The third and final injection types we investigated were string cusps. These injections are dependent upon a cut off frequency (Hz) and an amplitude. The string cusp wave equation is given below,

$$h(f) = A * f^{-5/3} * \theta(f - f_{inj}), \qquad (11)$$

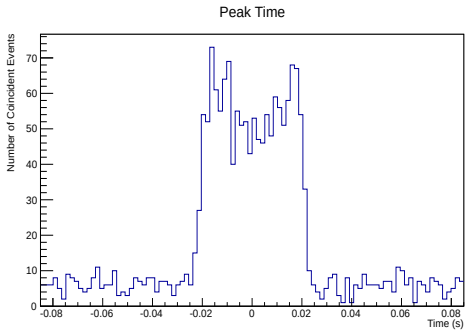
Where A is the amplitude of the waveform, f is the frequency, θ represents whether the waveform is positive or negative, and f_{inj} is the cutoff frequency.

Figure 15: *Detection efficiency as a function of injection amplitude.*



In the detection efficiency graph of figure 15 we can see that Omicron starts to detect injections at an amplitude of $\sim 10^{-20}$. As you will notice, the efficiency curve does not appear to continue falling towards zero, but instead flatens out at $\sim 10^{-21}$. This non-linear behavior represents the zero percent line of the efficiency curve and is essentially random coincidence matched by Coincidence Finder.

Figure 16: *Peak Time of string cusp injections*



In figure 16 above, we see that the peak time is centered around zero. We can assume that the elevated background activity around the peak point is due to stochastic noise coincidence.

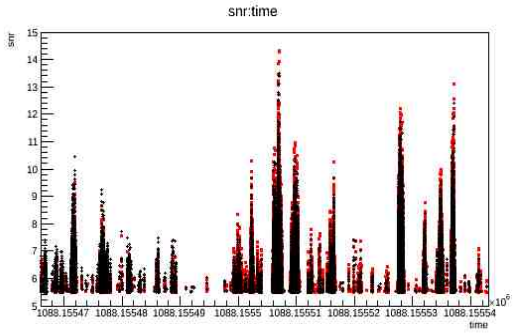
VI. Optimized Omicron

There is an updated version of Omicron that will be utilized in both the Virgo and LIGO detectors on a daily/hourly basis coming out

in August 2014. Before the release of the updated Omicron we performed tests utlizing Coincidence Finder and GWOLLUM functions. The new version of Omicron was optmized in such a way that no changes were made to the algorithms used to calculate the significance of triggers. For example, one of the changes made was that some steps in Omicron that were previously done during the trigger generation process, were moved to the beginning of the program so as to decrease computation time. Thus, we should see no change in the triggers produced by the optmized version of Omicron.

Initially we found through Coincidence Finder that new Omicron detects injections at a lower efficiency than its predecessor. As we can see in figure 16 below, the overall signal to noise ratio of transients decreases, thus producing triggers with lower resolution at a given time and frequency band.

Figure 17: *The figure below shows triggers produced by the old version of Omicron in black superimposed on triggers produced by the new version of Omicron in red.*



In order to determine the root cause of the decrease in overall SNR we go step-by-step through the Omicron process. In figure 17 below we plot the data time series of the time chunk after high pass and filtering with the new version of Omicron in blue and the old version in black doted lines.

Figure 18: *Data time series after high pass and filtering.*

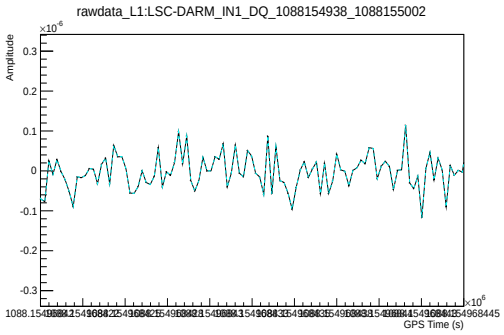
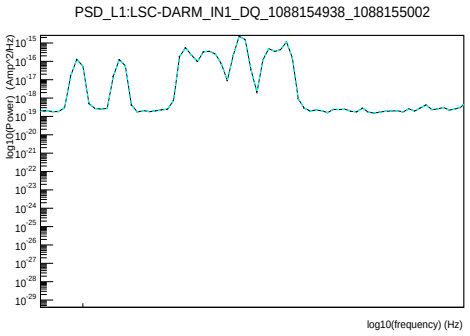


Figure 17 is a zoomed in view of the amplitude of the data as a function of time. As we can see the new verision of Omicron in blue is the exact same as the old in the dotted black line. Next, we can take a quick look at the power spectrum density.

Figure 19: *Zoomed in view of power as a function of frequency computed over the full data time series.*



Again, we clearly see the "bug" is not related to this process. After further investigations we determined that the cause of this "bug" was related to the conditioning of the data. During conditioning the data time series, after it has been high passed and filtered, is normalized by the power spectrum density computed over the full data chunk. During this process there was an errant square root function.

IV. Transient Analysis of Commissioning LIGO Data

Recently, the interferomter in Livingston, Louisiana has achieved several 2hr plus locks, where the commisioning team held the two arms of the interferomter in the correct positions for the laser light to resonate in the cavity properly[6]. We performed a detailed analysis of these time stretches to understand and mitigate "glitch" families within the differential arm lock channel.

I. Investigation Procedure

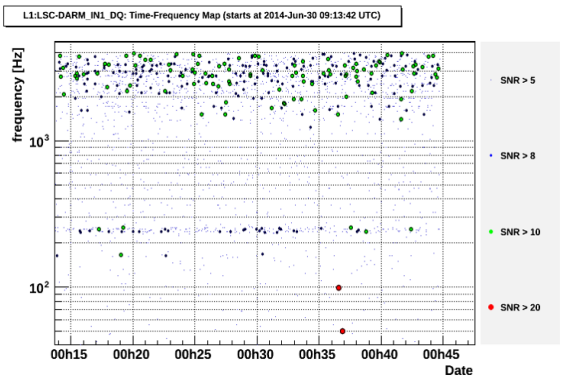
First, we run the Omicron trigger generator over the full stretch of *locked* data in a GW channel. We then utilize the Omicron function *GetOmicronPlots* which produces several different histograms (frequency distribution, Time-frequency map, SNR distribution, etc) that we can use to have a good overall understanding of the lock stretch. In particular, we look at the *glitchgram*, which is a frequency vs. time distribution of triggers with a verticle scale in SNR. We then identify discrete sets of "glitch" families by a set of triggers in a particular frequency band.

A few trigger times are identified within each of these glitch families, we then use these trigger times to make Omiscans. An Omiscan is a Omicron function that produces a set of frequency-time decomposition plots for a set of channels given by the user, and an SNR threshold also defined by the user. Channels that appear to have similar transient events to the main GW channel are compiled in a list. UPV is run over this channel list to identify any glitch-to-glitch couplings. These results are then discussed with individuals on the Interferomter site to verify the source of glitches and transients within the main GW channel.

II. June 30th, 2014 Lock

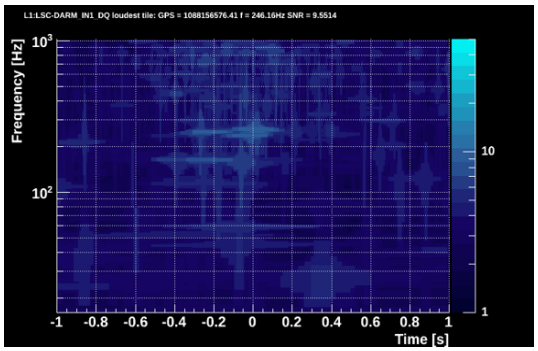
During this time period we looked specifically at the DC readout time in the differential arm lock channel (DARM) $\sim$ 45 minutes. This channel measures the difference between the x and y arm lengths (Lx-Ly). This is the main interferomter output for GW detection. After running Omicron we initially see in the glitchgram, which plots frequency as a function of time with a verticle scale in SNR, that there are two distinct "glitch" famlilies at ($\sim 2-3kHz$) and ($\sim 250Hz$).

Figure 20: *Glitchgram plotting frequency as a function of time.*



The series of "glithces" at $\sim 2-3kHz$ has a duration of $\sim 1s$ and a large bandwidth of $\sim 3-4kHz$ with an SNR of $\sim 9-13$. We initially believed this may be correlated with the "glitch" family at $250Hz$, as they both often occur at the same time.

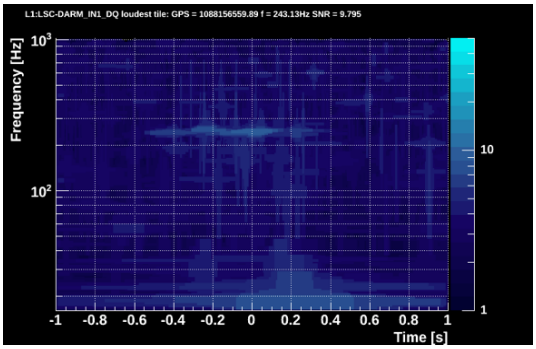
Figure 21: *Example of 2-3kHz "glitch" family using Omiscan. Frequency is plotted vs. time with a verticle SNR scale.*



The $250Hz$ "glitch" family has a duration of $\sim 1s$ and a short bandwidth with an SNR of $\sim 9-13$. This type of event is usually preceded by three loud glitches at lower frequency

bands $< 60Hz$, suggesting that this transient event may be correlated with the suspension channels.

Figure 22: *Example of 250Hz "glitch" family using Omiscan. Frequency is plotted vs. time with a verticle SNR scale.*



To determine if there is any possible coherence with the differential arm lock channel among other channels, we run an algorithm called use-percentage-veto (UPV). UPV is an algorithm to study glitch-to-glitch couplings between channels. When a coupling is identified it is possible to produce data quality vetoes. The UPV algorithm considers Omicron triggers from two input channels: an auxiliary channel and a main channel. These triggers are time clustered. A time coincidence is performed between these two cluster samples. Two clusters are coincident if they overlap in time. Then several figures of merit are computed.

We consider that the coupling between two channels is real if the use-percentage is greater than a given threshold (usually 0.5). A large use percentage value indicates that a trigger in the auxiliary channel is coupled (coincident) with a high probability to a trigger in the main channel[7].

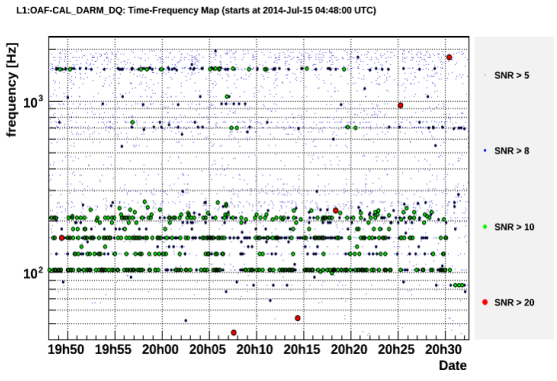
After running the algorithm we find that there are several candidates for coincidence. However, many have a use percentage that is far too high to be considered "safe" channels. This is due to the assumption that channels with an unusually high use percentage ($50 - 100$ percent) generally are the "same" as the main channel, thus they are considered "unsafe" channels. We found that the suspension channels in the end test masses of both the X and Y arms showed strong correlation with the DARM channel, and are considered somewhat "safe" depending upon the actuation of each channel (veto efficiency = 40 percent to 3 percent).

After presenting my results and discussions with the detchar team in Livingston, we determined the sources of these two "glitch" families. The series of events at $\sim 250Hz$ was the result of laser intensity noise in the pre-stablized laser (PSL) optical table and periscope. We found that there were some alignment issues. Thus, we understand why the PSL channels did not perform well in UPV due to the fact that there is no glitch-to-glitch coupling between the two. The "glitch" family at $\sim 2 - 3khz$ is possibly related to noise modulated by angular fluctuations in the differential ETM control module.

III. July 15th, 2014 Lock

During this time segment we want to see if transients present in the previous lock time were either mitigated, or reduced. We can have a good initial idea of this by comparing the glitch grams of the two lock times.

Figure 23: *Glitchgram of July 15th lock time plotting frequency as a function of time.*



The "glitch" family at $\sim 2 - 3kHz$ is completely mitigated while the family at $250Hz$ has been reduced significantly. Altough, we do see that between 100 Hz and 200 Hz there are many more lines of transients that have been

produced. We also notice a strong line around $1.5kHz$. If we look closer at the discrete lines we notice that the transient events at $103Hz$ and $128Hz$ are essentially same. This is also true for the pair at $158Hz$ and $207Hz$.

Figure 24: *Omiscan of 103hz glitch family.*

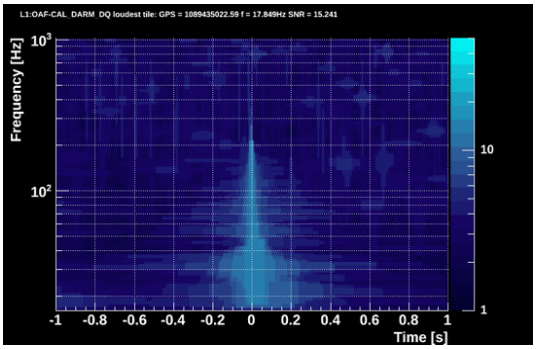


Figure 25: *Omiscan of 128hz glitch family.*

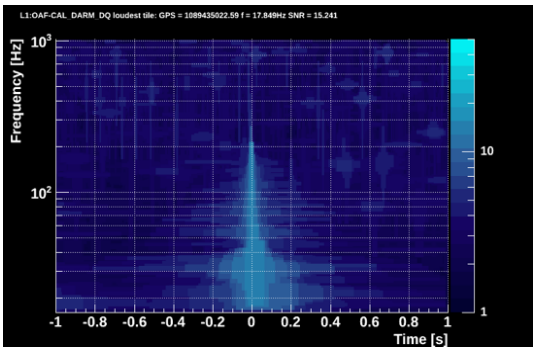


Figure 26: *Omiscan of 158hz glitch family.*

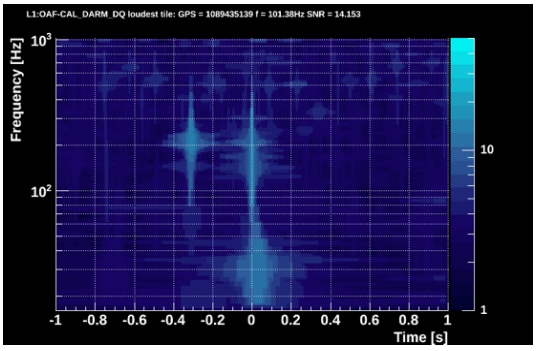
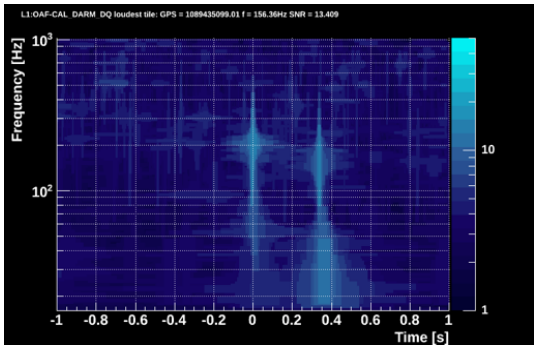


Figure 27: *Omiscan of 207hz glitch family.*



We believe that these two pairs are essentially a part of the same "glitch" family. After running UPV over the targeted channels we determined that these broadband low frequency "glitches" in the DARM channels were the result of motion in the PSL tables, as noted by coherence found in the PSL accelerometer channel by STAMP-PEM runs. These might also be coupled with suspension channels as given by the effciciency (vetoe efficiency = 10 percent). UPV did not select acceleromter channels well because the coupling is probably non-linear. The source of the family at high frequencies may be from a mix of mechanical and/or digital artifacts.

V. Final Remarks

This project has been fairly successful in determining the robust nature of Omicron when dealing with different injection "types". In regards to sine-gaussian injections we determined that, unlike Kleine Welle, Omicron is able to detect transients with an equal efficiency at both low and high frequency. In addition, we found that injections with an amplitude as low as a strain amplitude of 10^{-22}-10^{-21} were still able to be detected. Despite the reference time of the injection, Omicron is able to reconstruct the peak time and frequency with a low error. In White noise burst signals we initially found an asymmetry in the frequency reconstruction, and determined the cause not to be the fault of Omciron, but instead associated with background noise in the

channel.

Through these studies we can conclude that Omicron classifies "glitch" families with a great precision and has been useful in the characterization of aLIGO sub-systems on post-ER5 data. The Coincidence Finder programs will be used in future versions of Omicron to make sanity checks. The results from this project are important as they validate the usefulness of Omicron as both a trigger generator and burst type search pipeline that will be used in future observational runs at LIGO and Virgo.

VI. Acknowledgments

I would like to thank the National Science Foundation for funding this IREU program. Thank you to the University of Florida, Dr. Bernard Whitting, Dr. Guido Muller, and Dr. Andonis Mitidis for allowing me this wonderful opportunity to experience a new and exciting research environment. A special thanks to Dr. Florent Robinet for his constant help and support during this project (I sometime wonder how he ever got any work done). Thank you to all the members of the LAL group at Orsay, who were extremely warm and welcoming. Lastly, thank you to my advisor at my home institution Dr. Marco Cavaglia, who partly inspired me to pursue research in gravitational physics.

References

[1] Saulson, Peter R. Interferometric Gravitational Wave Detectors. Singapore: World Scientific Publising, 1994. Print.

[2] "LIGO Channel List." CIS. Caltech, n.d. Web.<https3A2F2Fcis.ligo.org2Fchannel2F>.

[3] LSC Members. "The Characterization of Virgo Data and Its Impact on Gravitational-wave Searches." ArXiv (2012): 1-50. ArXiv. Web. June-July 2014. <http://arxiv.org/>.

[4] Robinet, Florent. "GWOLLUM and Friends." GWOLLUM. LAL/Virgo, n.d. Web. <http3A2F2Fvirgo.in2p3.fr2FGWOLLUM2Findex.html3FFriend

[5] Brown, Duncan. "LALApps âĂŤ LSC Algorithm Library Applications." University of WisconsinâĂŚMilwaukee (2005): 1-182. LSC Algorithm Library Applications. Web. June-July 2014. <https://www.lsc-group.phys.uwm.edu/daswg/projects/lalapps/lalapps-5.0.pdf>.

[6] "LIGO Locks Its First Detector!" LIGO Locks Its First Detector! N.p., n.d. Web. 29 July 2014. <http://www.phys.lsu.edu/mog/mog17/node11.html>.

[7] Isogai, Tomoki, and The Ligo Scientific Collaboration A Collaboration. "Used Percentage Veto for LIGO and Virgo Binary Inspiral Searches." Journal of Physics: Conference Series 243 (2010): 012005. Web.