Seismic Isolation and Suspension System Instrument Characterization for Advanced LIGO Commissioning

Chase Kernan, Franklin W. Olin College of Engineering
Mentors: Alan Weinstein, Jameson Rollins

***Abstract*—Reaching the intended sensitivity and astrophysical reach of the Laser Interferometer Gravitational-Wave Observatory (LIGO) will require significant effort in characterizing environmental and detector subsystem noise. Identifying the creation and propagation of non-Gaussian 'glitches' will aid on-site commissioners in fixing instrumentation problems and achieving nominal equipment performance. We focus on the installed seismic isolation (SEI) and suspension (SUS) subsystems and leverage existing event trigger generators (ETGs) to develop statistical analyses of glitch behavior. We create novel visualization tools for investigative work and detail the nature of glitches originating from the inner stages of the SEI and SUS subsystems.**

I. Background

HAVING completed its sixth science run, LIGO is being rebuilt—and rechristened as Advanced LIGO—in order to significantly increase its sensitivity and extend its astrophysical reach [5]. Noise introduced by the environment and detector subsystems will require investigation and characterization. Non-fundamental noise sources may be localized in time or frequency and must be culled from the output gravitational wave (GW) strain signal to avoid reductions in sensitivity [2]. The search for continuous GWs and any stochastic GW background will be detrimentally impacted by spectral line noise, while transient GW events may be masked by short-duration 'glitches' [4].

Instrument characterization will help identify noise artifacts whose origin can be physically located and corrected, or barring that, be flagged for data quality and veto generation [3]. Artifacts are often noticed as non-Gaussian fluctuations in data coming from auxiliary channels that record the status of individual components as well as the surrounding physical environment [1]. Correlating non-Gaussianity statistics across multiple channels can help debug noise sources and quickly bring instrumentation performance to nominal levels.

Much infrastructure for identifying glitches already exists and is being brought online as Advanced LIGO opens more auxiliary channels. This project utilizes the existing *Omicron* event trigger generator (ETG), which detects bursts of non-Gaussianity in the time and frequency domains by match-filtering against sine-Gaussians. However, there is significant work to be done in summarizing the *Omicron* outputs for efficient use by commissioners.

In particular, commissioners have installed several seismic isolation (SEI) and suspension (SUS) subsystems, which require meaningful performance analysis. Each subsystem

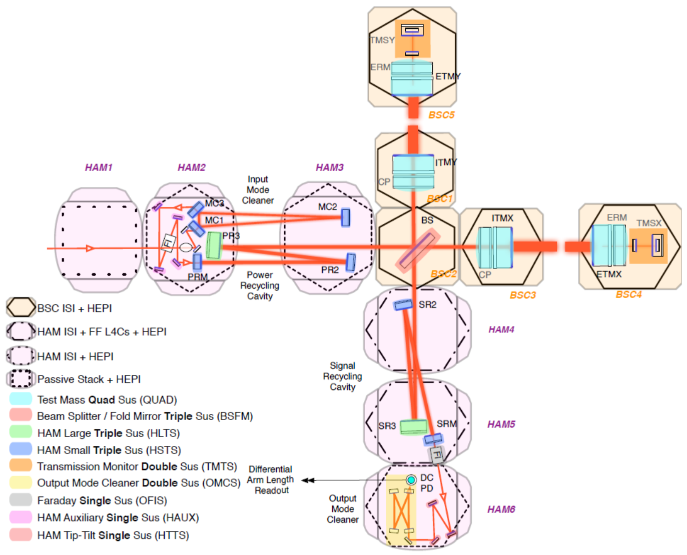

TMSY
ERM
ETMY
BSC5
ITMY
CP
BSC1
HAM1
HAM2
HAM3
Input Mode Cleaner
MC2
MC1
MC3
PR3
PRM
MC2
PR2
Power Recycling Cavity
BS
BSC2
ITMX
CP
BSC3
ERM
TMSX
ETMX
BSC4
SR2
HAM4
Signal Recycling Cavity
SR3
SRM
HAM5
DC PD
HAM6
Differential Arm Length Readout
Output Mode Cleaner
BSC ISI + HEPI
HAM ISI + FF L4Cs + HEPI
HAM ISI + HEPI
Passive Stack + HEPI
Test Mass Quad Sus (QUAD)
Beam Splitter / Fold Mirror Triple Sus (BSFM)
HAM Large Triple Sus (HLTS)
HAM Small Triple Sus (HSTS)
Transmission Monitor Double Sus (TMTS)
Output Mode Cleaner Double Sus (OMCS)
Faraday Single Sus (OFIS)
HAM Auxiliary Single Sus (HAUX)
HAM Tip-Tilt Single Sus (HTTS)

Fig. 1. Seismic isolation (SEI) system configuration with enclosed optics at Livingston (L1) and Hanford (H1) observatories.

consists of a series of stages—some active, some passive—that work to reduce the influence of ground motion on the enclosed optical equipment. Ideally, noise is only introduced at the ground stages, but mechanical malfunctions and improper installations may inject noise at any stage. Hence, we expect the statistical profile of non-Gaussianity to vary among the several installed systems, despite identical designs.

We can diagnose discrepancies using the many auxiliary channels that record movement in every degree of freedom at each stage in the SEI/SUS subsystems. As we begin to monitor more SEI subsystems, we can compare 'glitch statistics' across same-design subsystems to identify faulty equipment. At the time of this report, we collect data from the ETMY and HAM3 isolators at Hanford, and the HAM2-5, ITMX, ITMY, BSX, and BSY isolators at Livingston (see Figure 1).

II. Objectives

We aim to characterize the behavior of Advanced LIGO's SEI/SUS subsystems by means of 'glitch' correlations and other statistical studies of non-Gaussianity. From this, we hope to gather sufficient insight to direct on-site commissioners towards malfunctioning equipment and other sources of noise. The more we localize poor performance in time, frequency, and space (i.e. affected channels), the more likely commissioners can physically identify the offending instrumentation.

III. Results

We approached this investigative task roughly as follows. Drawing on commissioning status and information from Jessica McIver, we select a small subset of working SEI/SUS auxiliary channels to analyze. These 89 channels are divided into nine mechanically-connected groups corresponding to the SEI/SUS subsystems shown in Figure 1. We then feed channel data through the *Omicron* ETG, obtaining a time series of localized glitches. From this, we can look for statistical outliers—such as glitches with very high signal to noise ratios—and time correlations between connected channels.

A. Event Trigger Tiles

Omicron emits trigger 'tiles', which are bounded by time and frequency ranges (see the left subfigure of Figure 2). They also possess properties describing the corresponding glitch, such as the signal-to-noise ratio (SNR), amplitude, q-value, and peak frequency. Due to the nature of match-filtering, however, a single glitch typically produces many trigger tiles that overlap in time and/or frequency. To obtain accurate statistics at the event level, we must cluster all tiles corresponding to a single event into a single tile. Roughly speaking, all tiles that overlap (or nearly overlap) in time-frequency space are merged in an iterative algorithm described in Section V. Properties like peak frequency are merged by taking the SNR^2-weighted-average of the peak frequencies of the overlapping tiles. This results in an approximately 100:1 reduction in the number of trigger tiles (see Figure 2, right).

B. Coincident Glitches

Some background level of glitches can be expected under nominal conditions, with ground movement introducing bursts of noise that mechanically propagate through the SEI and SUS subsystems. However, faulty hardware may also induce glitch propagation not at the ground level, but rather at some intermediate SEI/SUS stage. Mechanical propagation will manifest itself as several glitches that are coincident in time across multiple channels. Using an algorithm similar to trigger clustering, we identify and group all events that begin within $\Delta t = 0.1$s of each other (see Section V). A set of coincident glitches is akin to a chain, where the head corresponds to the first-occurring event. Analysis of chain patterns and statistics should point to faulty SEI/SUS stages and modes of propagation.

This analysis is complicated by the low-frequency nature of seismic isolators—we are fundamentally limited by the period of our match-filtered sine-Gaussians when determining the time range of a glitch. In the seismic band (1-10 Hz), this period-dependent imprecision is much greater than the speed of mechanical propagation. Thus, we cannot determine the order of a seismic glitch chain. We are forced to assume that glitches will proceed in the mechanical ordering of isolation stages.

C. Analysis and Visualization

We devoted much of this project to developing tools for analyzing triggers and coincident glitches; we produced many interactive visualizations designed to emphasize patterns and ease the process of manual investigation. These interfaces are web-based and available on the `ldas-jobs.ligo.caltech.edu` server, with accompanying RESTful APIs to data sources.

We displayed raw and clustered *Omicron* outputs using a scrolling set of tiles color-coded by SNR (Figure 2). An unpursued offshoot of this project lead to the development of real-time scrolling capabilities (latencies limited to the ability of the ETG, roughly 4 s). By navigating through several glitch events, a user can readily identify the frequency profile of a class of glitches in a particular channel.

The interface central to our project depicts coincident chains of glitches in a particular SEI/SUS subsystem (Figure 3). With chains, we are primarily concerned with the following attributes:

- *Affected channels*: shown as color-coded bars linked by lines corresponding to individual chains
- *Peak SNR*: shown by the thickness of the links
- *Attenuation*: whether or not the SNR is decreasing as the glitch propagates, shown as green lines for decreasing SNR, red for increasing
- *Peak frequency*: selectable on the percentage plot on the right
- *Time*: shown as a vertical offset on each color-coded bar
- *Propagation mode*: shown as the left-to-right ordering of bars (keeping in mind the aforementioned timing assumption)

A user can filter the displayed links by either drawing a region in SNR-frequency space on the right-hand percentage plot or by hovering the mouse over a bar or individual link. When hovering over a channel bar, only glitches that propagate through that channel at that position in the chain are shown. For example, hovering over the third bar down on the left shows only glitches that originate at the third stage. Hovering over an individual link shows additional statistics, and, when clicked, opens a visualization of the raw glitch data. We designed the raw glitch visualization to emphasize the timing and amplitude profile of cascading glitches (Figure 4). A vertical ruler that follows the mouse shows the calibrated amplitude in consistent units for a particular time. For each channel in the subsystem, there are two amplitude plots: one that has been bandpassed for the frequencies identified by *Omicron* and another that is left unprocessed.

Finally, we developed a visual query system to find coincident glitches based on selected attribute ranges (Figure 5). For each glitch chain, we plot the starting time, duration, maximum SNR, SNR attenuation ratio, peak frequency, frequency bandwidth, chain length, starting channel, duration, and mean delay, followed by a table detailing the frequency and SNR components of each channel in the chain. Such plots both show the distribution of attribute values and allow the user to visually select a value range to filter by—the table and other plots are dynamically updated to only show the filtered chains.

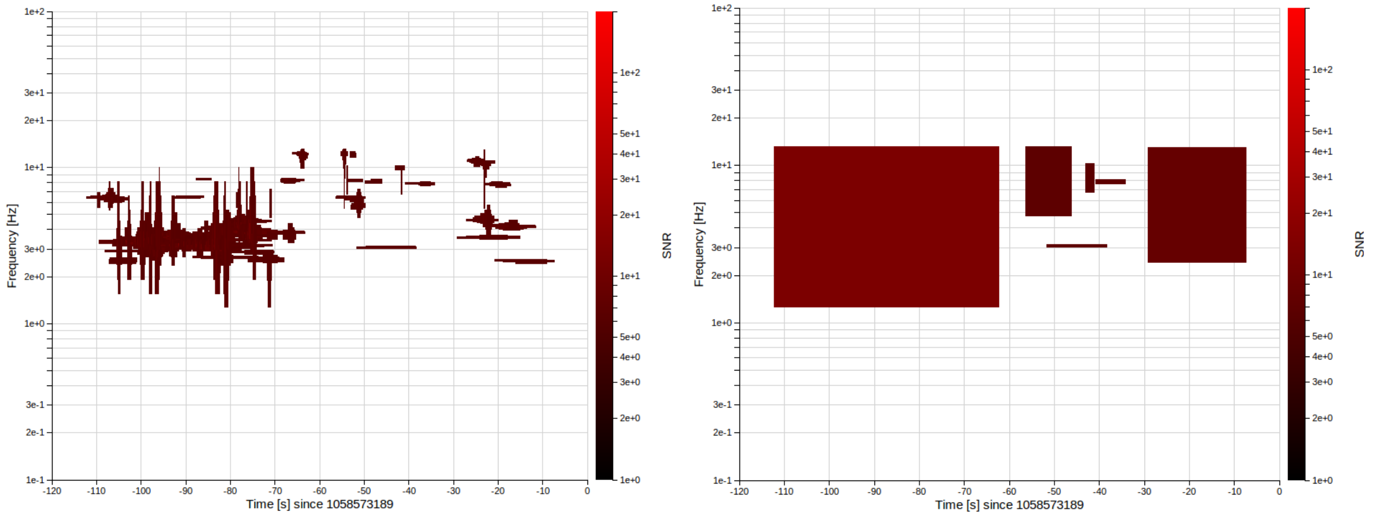


Fig. 2. The raw output of the *Omicron* ETG, shown on the left, includes many triggers that overlap in frequency and/or time. We assume that triggers that overlap in time or frequency (or are very close, i.e. $\Delta t < 0.05$ s or $\Delta f < 0.5$ Hz) correspond to the same event. Hence, we cluster overlapping triggers to produce the conglomerate triggers shown on the right.

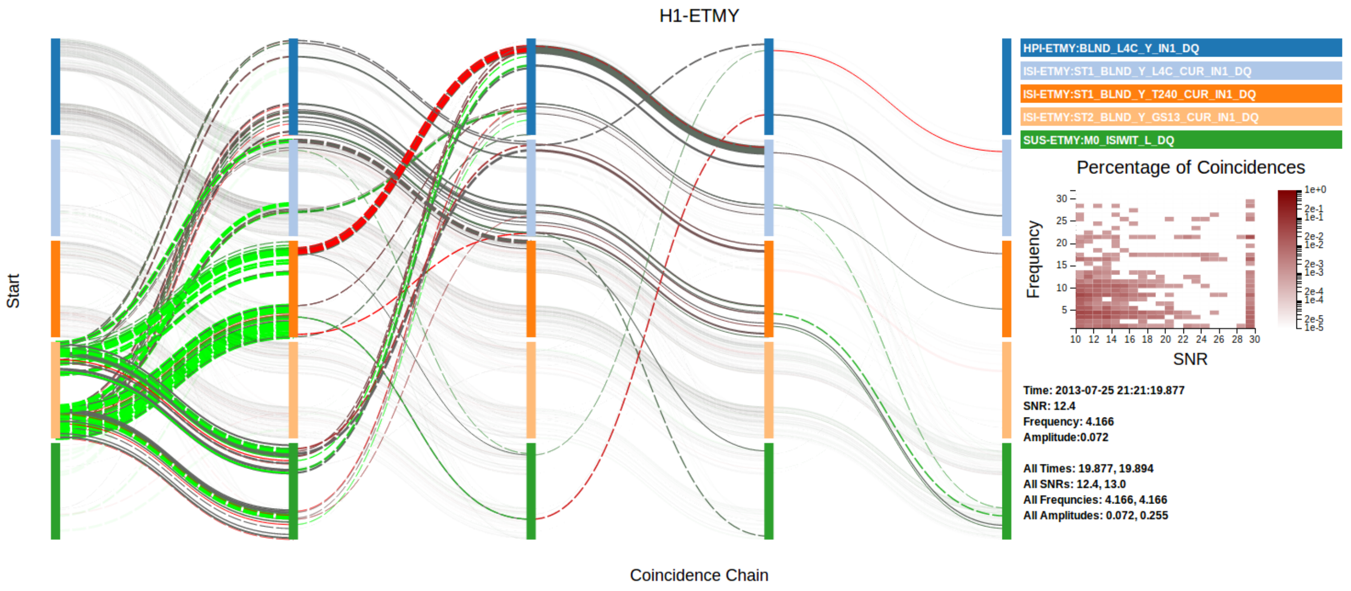

H1-ETMY
Start
Coincidence Chain
HPI-ETMY:BLND_L4C_Y_IN1_DQ
ISI-ETMY:ST1_BLND_Y_L4C_CUR_IN1_DQ
ISI-ETMY:ST1_BLND_Y_T240_CUR_IN1_DQ
ISI-ETMY:ST2_BLND_Y_GS13_CUR_IN1_DQ
SUS-ETMY:M0_ISIWIT_L_DQ
Percentage of Coincidences
Frequency
SNR
Time: 2013-07-25 21:21:19.877
SNR: 12.4
Frequency: 4.166
Amplitude:0.072
All Times: 19.877, 19.894
All SNRs: 12.4, 13.0
All Frequncies: 4.166, 4.166
All Amplitudes: 0.072, 0.255

Fig. 3. Glitch coincidences in the ETMY SEI/SUS subsystem at the Hanford observatory. Each connected line represents a glitch coincident on the channels it connects. Each column represents a time difference of less than 0.1s. The leftmost column represents the origin of each coincidence. When a user places their mouse over a channel bar or a line, the coincidences are highlighted. In this case, the user has selected all coincidences that begin in the fourth stage of the ETMY subsystem. Users are also capable of filtering by trigger signal-to-noise ratios (SNR) and peak frequencies by dragging over the percentage plot on the right.

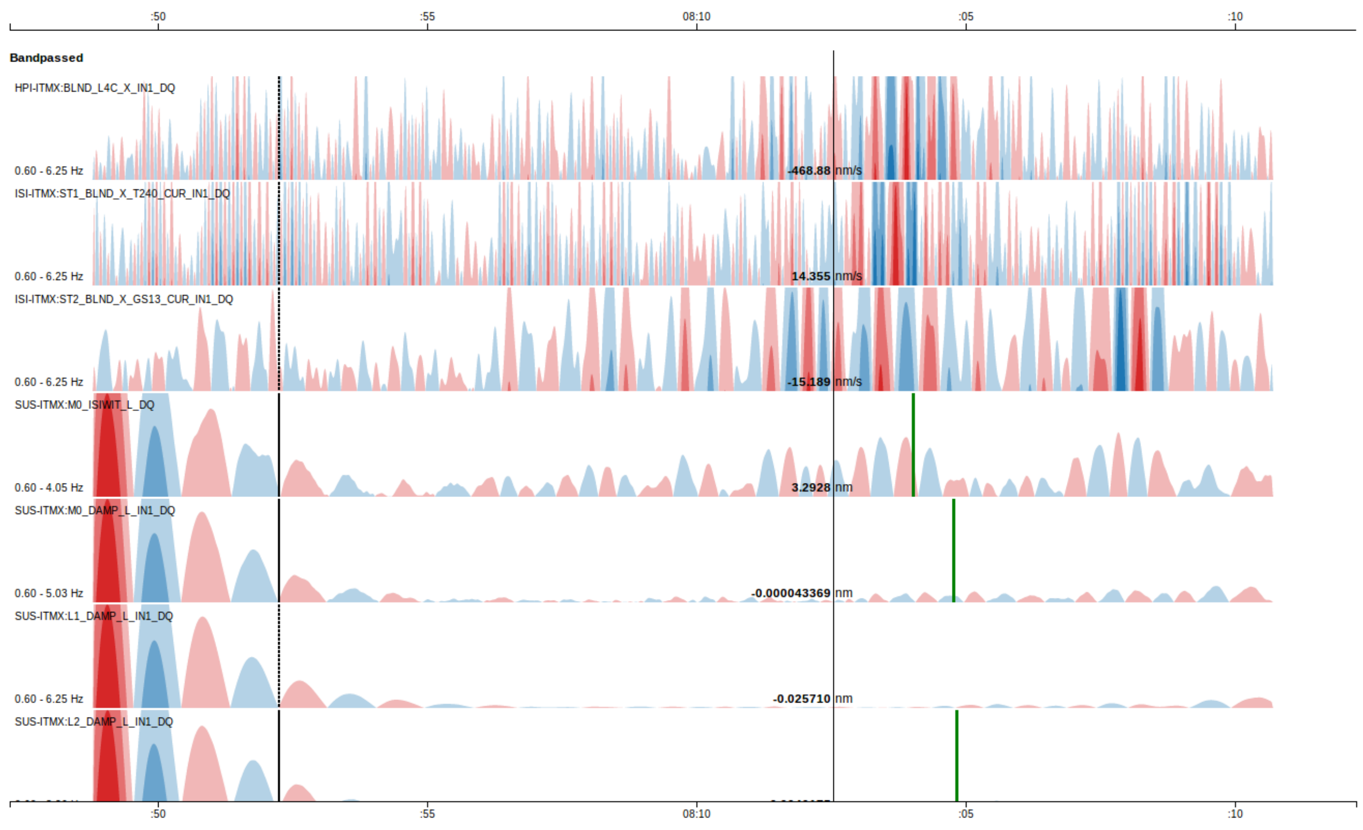


Fig. 4. A visualization of a single coincident glitch as it propagates through the ITMX subsystem at the Hanford observatory. Each row corresponds to a single channel of raw data that has been bandpassed to include only the glitching frequencies identified by *Omicron*. In order to have high vertical resolution, the amplitude data is shown as a horizon plot, where color is indicative of sign (i.e. blue for positive values, red, negative) and magnitude (darker colors correspond to multiples of the vertical scale). The vertical black line follows the user's mouse and shows the bandpassed-amplitude at that moment in time. Just to the right of the black line is a strong visual indicator of a glitch event coincident across four channels. Not shown are several non-bandpassed plots available to the user below.

For example, a user can easily query all glitches that occur in the morning, have large SNR, do not attenuate SNR properly, and originate in an intermediate isolation stage.

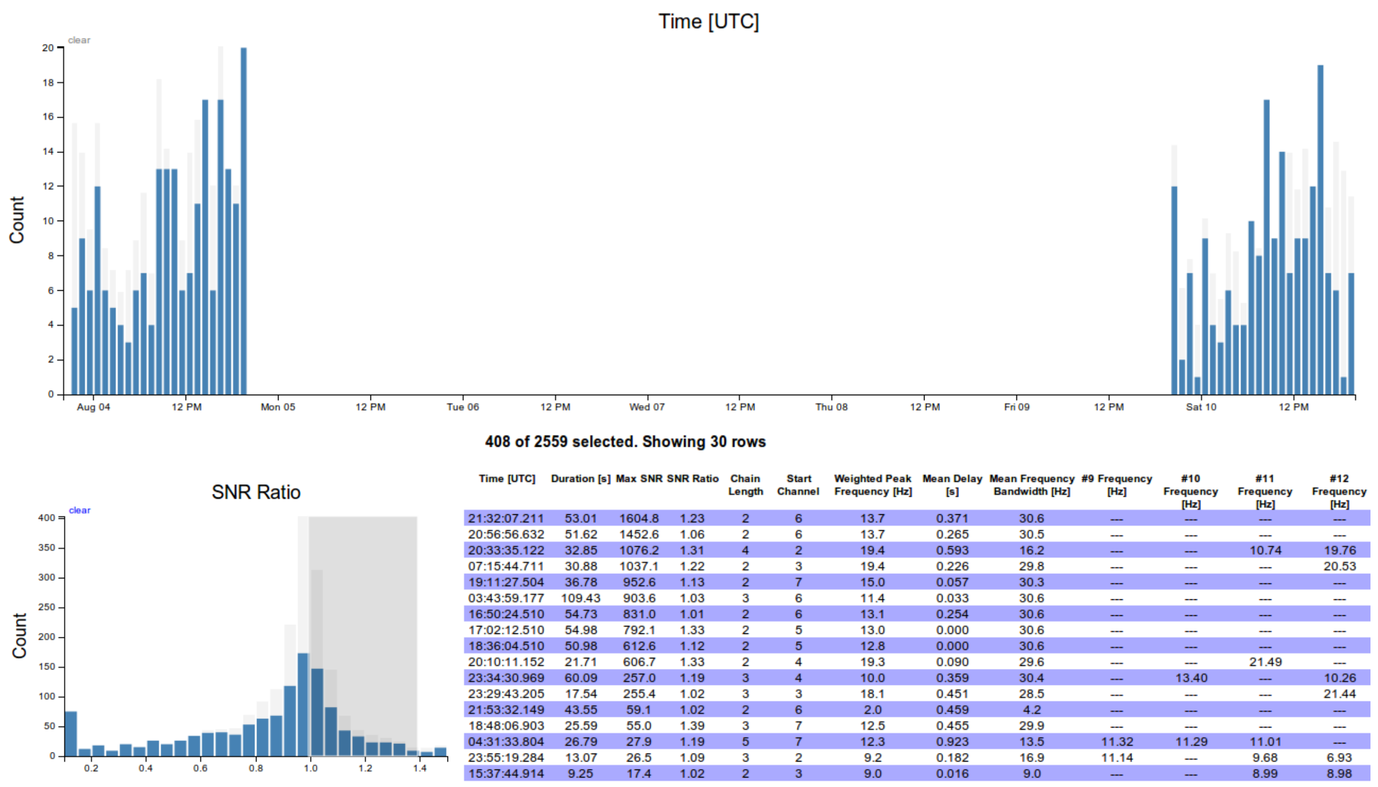
IV. Future Work

Armed with a large toolkit and dataset of glitches and coincidences, we can begin to investigate possible causes and equipment malfunctions. We will coordinate with Jessica McIver and on-site commissioners in order to find good state times at each of the observatories and start to identify the physical manifestations of prominent glitch classes.

V. Methods

Delivering interactive visualizations to users required creating an extensive computational framework. This consists primarily of a Python webserver coupled with an HDF5 database and a series of Javascript visualizations made with the D3.JS library. HDF5 databases allow us to store large quantities of triggers and glitches with high compression ratios and very fast querying operations, while the D3.JS library enables multidimensional visualization, interaction, and easy SVG manipulation. The global control flow can be described as follows:

1) Analysis
   a) Run *Omicron* for good state times on each channel in the SEI/SUS subsystems. This outputs a set of XML files containing trigger data.
   b) Parse the *Omicron* data and cluster triggers into events. These events are stored in an HDF5 database.
   c) Identify glitch coincidences, which are also stored in an HDF5 database.
   d) For each coincidence, download the raw channel data from LDS and store it alongside a bandpassed version in an HDF5 database.
2) Server
   a) Start a CGI server in Python using the Bottle.py library.
   b) Based on incoming HTTP requests, query the appropriate HDF5 database and return the requested data in the JSON format.
3) Client



408 of 2559 selected. Showing 30 rows

Time [UTC]	Duration [s]	Max SNR	SNR Ratio	Chain Length	Start Channel	Weighted Peak Frequency [Hz]	Mean Delay [s]	Mean Frequency Bandwidth [Hz]	#9 Frequency [Hz]	#10 Frequency [Hz]	#11 Frequency [Hz]	#12 Frequency [Hz]
21:32:07.211	53.01	1604.8	1.23	2	6	13.7	0.371	30.6	---	---	---	---
20:56:56.632	51.62	1452.6	1.06	2	6	13.7	0.265	30.5	---	---	---	---
20:33:35.122	32.85	1076.2	1.31	4	2	19.4	0.593	16.2	---	---	10.74	19.76
07:15:44.711	30.88	1037.1	1.22	2	3	19.4	0.226	29.8	---	---	---	20.53
19:11:27.504	36.78	952.6	1.13	2	7	15.0	0.057	30.3	---	---	---	---
03:43:59.177	109.43	903.6	1.03	3	6	11.4	0.033	30.6	---	---	---	---
16:50:24.510	54.73	831.0	1.01	2	6	13.1	0.254	30.6	---	---	---	---
17:02:12.510	54.98	792.1	1.33	2	5	13.0	0.000	30.6	---	---	---	---
18:36:04.510	50.98	612.6	1.12	2	5	12.8	0.000	30.6	---	---	---	---
20:10:11.152	21.71	606.7	1.33	2	4	19.3	0.090	29.6	---	---	21.49	---
23:34:30.969	60.09	257.0	1.19	3	4	10.0	0.359	30.4	---	13.40	---	10.26
23:29:43.205	17.54	255.4	1.02	3	3	18.1	0.451	28.5	---	---	---	21.44
21:53:32.149	43.55	59.1	1.02	2	6	2.0	0.459	4.2	---	---	---	---
18:48:06.903	25.59	55.0	1.39	3	7	12.5	0.455	29.9	---	---	---	---
04:31:33.804	26.79	27.9	1.19	5	7	12.3	0.923	13.5	11.32	11.29	11.01	---
23:55:19.284	13.07	26.5	1.09	3	2	9.2	0.182	16.9	11.14	---	9.68	6.93
15:37:44.914	9.25	17.4	1.02	2	3	9.0	0.016	9.0	---	---	8.99	8.98

Fig. 5. A visual query system for finding coincident glitches with certain properties. For example, a user may select a range of SNR ratios (in this case, ratios greater than unity have been selected) and the glitches listed in the table to the bottom right will be filtered accordingly. The other query charts will be dynamically updated to reflect the distribution of selected glitches. The full interface contains many more query plots: peak frequency, duration, peak SNR, etc. The large segment of time with no glitches shown above is a result of bad state times.

a) Request data from the server using HTTP requests and parse the returned JSON strings.
b) Use the D3.JS library to create interactive visualizations.

The entire codebase for this project can be found as a Git repository at https://github.com/chase-kernan/lsc-seis-gcm.

There are two important algorithms developed in the course of this project, which are outlined below.

A. Trigger Clustering

```
$clusters \leftarrow []$
$heap \leftarrow []$
**for all** $trigger$ in $triggers$ ordered by $start\_time$ **do**
    {Remove any clusters in the heap that occur before this and all following triggers}
    **for all** $cluster$ in $heap$ **do**
        **if** $end\_time(cluster) < start\_time(trigger)$ **then**
            add $cluster$ to $clusters$
            remove $cluster$ from $heap$
        **end if**
    **end for**

    {Try to find a cluster that this trigger overlaps with}
    $found\_match \leftarrow$ **false**
    $new\_cluster \leftarrow trigger$
    **for all** $cluster$ in $heap$ **do**
        **if** $overlaps(new\_cluster, cluster)$ **then**
            $found\_match \leftarrow$ **true**
            remove $cluster$ from $heap$
            $new\_cluster \leftarrow$ merge $cluster$ with $new\_cluster$
        **end if**
    **end for**

    **if not** $found\_match$ **then**
        $new\_cluster \leftarrow initialize(new\_cluster)$
    **end if**

    add $new\_cluster$ to $heap$
**end for**
concatenate $heap$ to $clusters$
```

B. Finding glitch coincidences

```
$triggers$ {is a map from channels to trigger arrays}
$rows \leftarrow \{\}$
$ends \leftarrow \{\}$
$times \leftarrow \{\}$
**for all** $c$ in $channels$ **do**
    $rows[c] \leftarrow 0$
    $ends[c] \leftarrow length(triggers[c])$
    $times[c] \leftarrow start\_time(triggers[c][0])$
```

```
end for
while $times$ is not empty do
  sort $times$ by value and then by channel order
  $start\_channel, start\_time \leftarrow first(times)$
  $linked \leftarrow [start\_channel]$

  $window\_end = start\_time + 0.1$ {seconds}
  for all $match\_channel, match\_time$ in $rest(times)$ do
    if $match\_time <= window\_end$ then
      add $match\_channel$ to $linked$
      $window\_end \leftarrow match\_time + window$
    else
      break
    end if

    if $length(linked) > 1$ then
      $linked\_triggers \leftarrow []$
      for all $c$ in $linked$ do
        add $triggers[c][rows[c]]$ to $linked\_triggers$
      end for
      save $linked\_triggers$ as a glitch coincidence
    end if

    for all $c$ in $linked$ do
      $row \leftarrow rows[c] + 1$
      $rows[c] \leftarrow row$
      if $row < ends[c]$ then
        $times[c] \leftarrow start\_time(triggers[c][row])$
      else
        remove $c$ from $times$
      end if
    end for
  end for
end while
```

VI. Acknowledgements

I would like to thank Professor Alan Weinstein, Jessica McIver, Jameson Rollins, and Scott Dossa for their help in advising me during this project. I would also like to thank the California Institute of Technology SURF program for granting me the opportunity to pursue this project. I would also like to thank the NSF for the funds to carry out this project.

References

[1] N. Christensen. LIGO S6 detector characterization studies. *Class. Quantum Grav.*, 27(194010), 2010.

[2] B. P. Abbot et al. LIGO: the laser interferometer gravitational-wave observatory. *Rep. Prog. Phys.*, 72(076901), 2009.

[3] D. M. Macleod et al. Reducing the effect of seismic noise in LIGO searches by targeted veto generation. *Class. Quantum Grav.*, 29(055006), 2012.

[4] Virgo Collaboration, LIGO Scientific Collaboration. Characterization of the LIGO detectors during their sixth science run. *LIGO Document*, (P1000142), 2013.

[5] K. Riles. Gravitational waves: Sources, detectors and searches. *Progress in Particle and Nuclear Physics*, 68:1–54, 2013.