

Improvements to Data Characterization with Hierarchical Algorithm for Curves and Ridges (HACR)

Christopher Kelly

June 1 – July 30, 2011

Contents

1.) Introduction

1. Gravitational Waves
2. Detectors
3. GEO600
4. Detector Characterization
5. HACR

2.) Initial problems

1. Number of triggers
2. Frequency ranges
3. Not enough channels

3.) Improvements

1. Filters
2. Tuning thresholds
3. Adding new channels

4.) Results

1. Comparison
2. Looking towards the future of HACR channels

1.) Introduction

1.1) Gravitational Waves

In the early 20th century, Albert Einstein published his paper, “On the Electrodynamics of Moving Bodies.” While it was primarily on his theory of relativity, it was also the start to the search for gravitational waves. He theorized that energy could be transported in the form of gravitational waves, which would travel and ripple through space-time. These gravitational waves are the result of large masses with time-varying quadrupole moments of mass distribution. These waves travel at the speed of light, and do not interact much with matter, which is a great opportunity for physicist and astronomers, yet also defines the main difficulty in gravitational wave detection. The waves contain almost unaltered information about the events that created them regardless of what material they've passed through, allowing researchers to learn more about our universe. The difficulty arises in the detection of these waves, because they do not leave much record of passing through matter.

1.2) Detectors

The best hope for gravitational wave detection has been vested in interferometers, which can detect very small changes in the lengths of their L-shaped arms that would be caused by a gravitational waves passing through. There are several operational interferometric detectors in the world. There are three LIGO detectors in the US; one of which has 4 km arms and is in Livingston, LA, and the other two (2 and 4 km) are in Hanford, WA. It is useful to have two detectors in one location in order to run consistency tests between the two of them since they should show the same responses to a gravitational wave. The Virgo Detector resides in Pisa, Italy with 3 km arms and is operated by a French-Italian collaboration. Another detector exists not too far away in Hanover, Germany; the GEO600 detector is operated by a British-German collaboration. Having more detectors around the world allows us to increase our sensitivity by comparing results and eliminating local noises.

1.3) GEO600

The GEO600 detector is a 600 meter Michelson interferometer with dual-recycled folded arms. Its main mirrors are suspended from triple pendulums with feedback control systems that work to keep the interferometer on a dark fringe, which means reflected light coming from the optical paths in the arms destructively interferes and nothing should be seen by the photo-detector. When light power is measured by it, that's when we become interested, since this indicates a difference in the lengths of the arms.

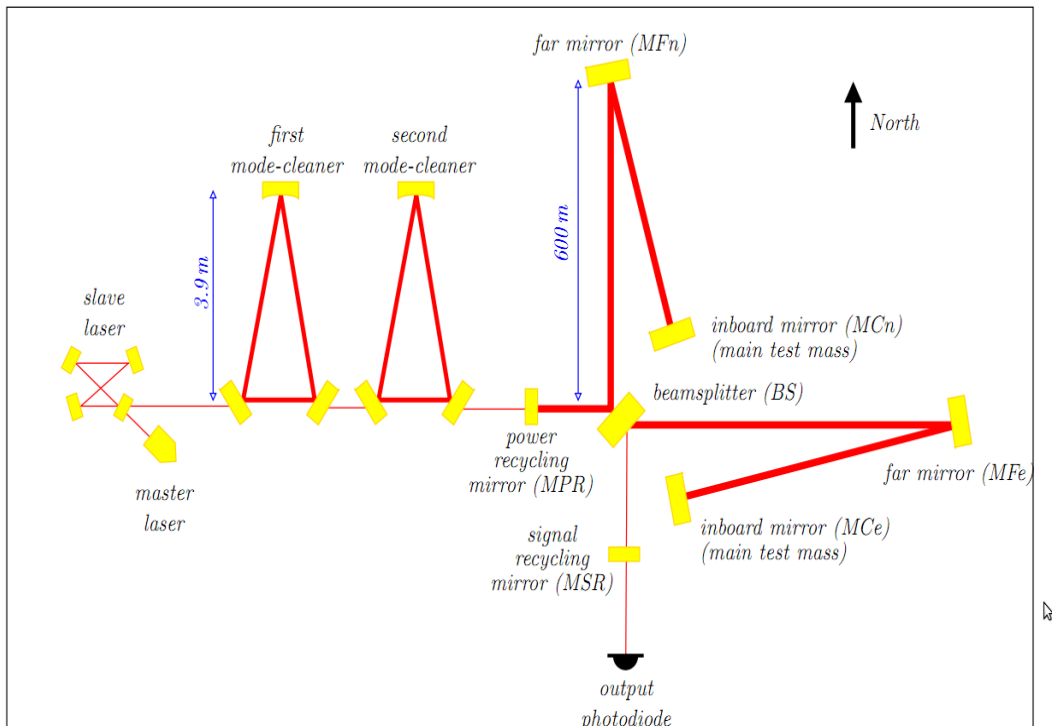


Figure 1: Layout of the GEO600 interferometer [1]

The above diagram shows the optical layout of GEO600. The master twelve watt laser sends a beam through two mode cleaners for spatial filtering. After going through the power-recycling cavity, the beam meets the splitter at about ten kW. When the two returning beams do not interfere destructively, the remaining light signal is amplified by the signal recycling mirror, before being passed on to the output mode cleaner(not pictured) and finally hits the photodiode.

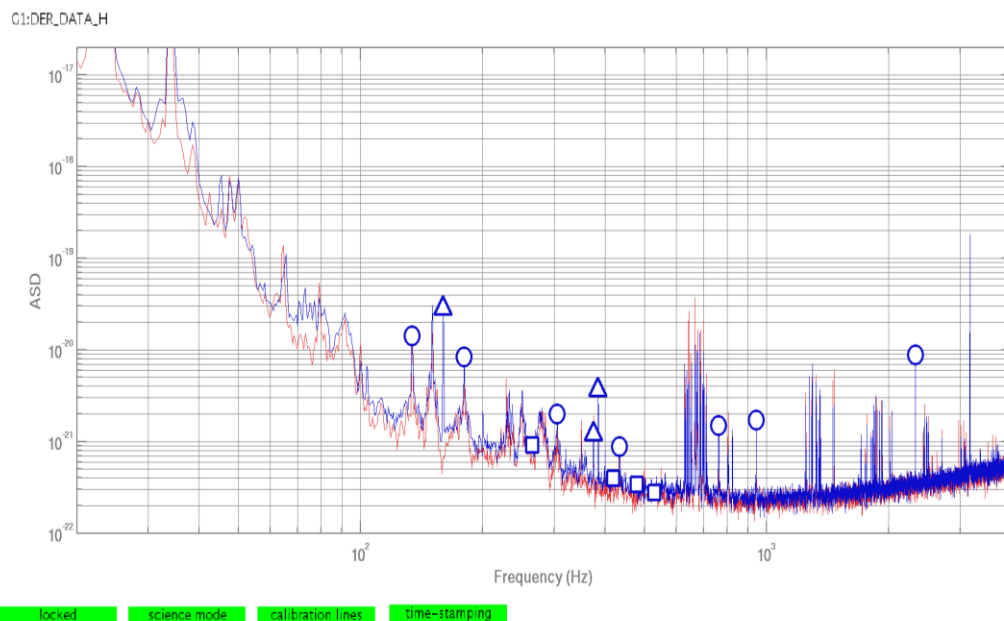


Figure 2: Recent sensitivity curve of GEO600 during S6E

1.4) Detector Characterization

While light power on the photo-detector is certainly one of the most important measurements we need to record, there are many other channels of information that hold significant data. Seismometers tell us about geological activity that could be interfering with our data. Microphones tell us about acoustical noise being introduced to the system. All of this noise needs to be understood in order to isolate real information about gravitational waves that may pass through. Detector characterization is the effort behind this understanding. At GEO600, the detector characterization group works to understand glitches that occur within the system and how glitches in one data channel might be related to glitches in another. In order to gain information about glitches that occur, we use multiple tools including spectrograms, time-series plots, and HACR plots.

1.5) Hierarchical Algorithm for Curves and Ridges (HACR)

HACR (pronounced: hacker) uses discrete Fourier transforms to model data in frequency versus time maps. It displays this data in the form of triggers that represent clusters of events. Events are any signals that occur within a channel. The mathematics of this algorithm are described quite thoroughly in Ajith Parameswaran's thesis [2], so I will not go into technical descriptions. For my work, it was important to understand the input parameters that HACR uses and how modifications for these influence the acquired data. Frequency ranges and thresholds are two of these parameters. A frequency range defines the range of frequencies HACR looks over for events. There are two signal-to-noise ratio (SNR) thresholds (lower and upper) that control what constitutes as a trigger. Anything below the first trigger is not significant and not considered for a trigger. A trigger is defined by a cluster of events that are all above the lower threshold and at least one of which is above the higher threshold. A graphical way to visualize it is a landscape covered in mountains and hills. Anything above the second threshold is a mountain and we want to know about any mountains in the data. We assume there will be also be surrounding hills (between the two thresholds), and number of these hills in addition to the mountain(s) defines the size of the trigger. These two thresholds can be modified in order to get a certain number of triggers, which leads directly into my work.

2.) Initial Problems

2.1) Number of triggers

It is vital we know where triggers are occurring, yet too much information crowds frequency-time plots and makes conclusions difficult to make. When I arrived, there were about 35 channels being analyzed by HACR. The HACR plots are online in the summary pages and cover eight hour stretches of data. For eight hours, we want a reasonably large number of triggers to show what trends occur and what happens when glitches occur. We aim to have about one thousand triggers, which, as is visible in Figure 3, is a good amount to see specific trends. Originally, the 35 channels on HACR all had the same thresholds, but did not get the same number of triggers. As it turns out, some channels are really quiet while others get a lot of activity, resulting in widely varying trigger counts from the 30 triggers in Figure 4 to the 9500 triggers in Figure 5. To get a more consistent trigger count, the thresholds must actually be tuned for every individual channel.

2.2) Frequency ranges

Many of the measurement devices have been upgraded to include higher frequencies in their detections over the past couple years. Unfortunately, HACR had not been updated to include these new frequency bands. Many of us were interested in events that may be happening in channels above the resolution range in HACR plots. Most HACR configurations for channels only went reached upwards to 2000 Hz (Figure 6), even though some channels were upgraded all the way to the 8000 Hz range! Also, some of the channel's HACR configurations did not start at zero, but there may be interesting events in lower frequencies as well. Therefore, any improvements to HACR would have to include frequency range expansions for many channels.

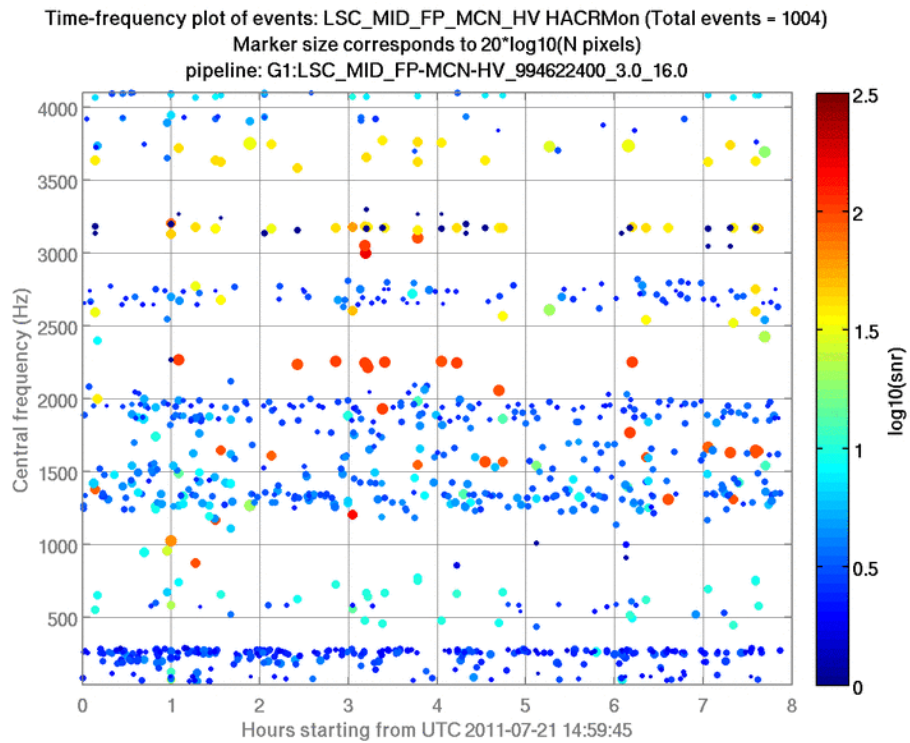


Figure 3: One thousand triggers can show clear trends without data overpopulation. Frequency bands with a lot of activity can be clearly spotted and single loud events can also be seen.

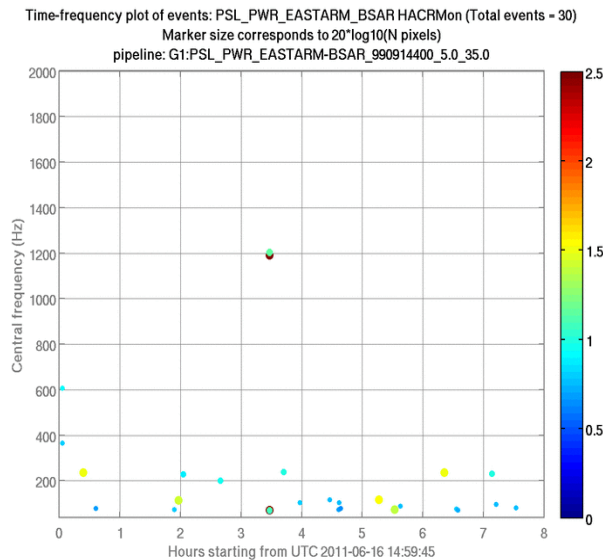


Figure 4: Too few triggers mean that HACR plots aren't informative and tell us little about glitches and noise in frequency bands.

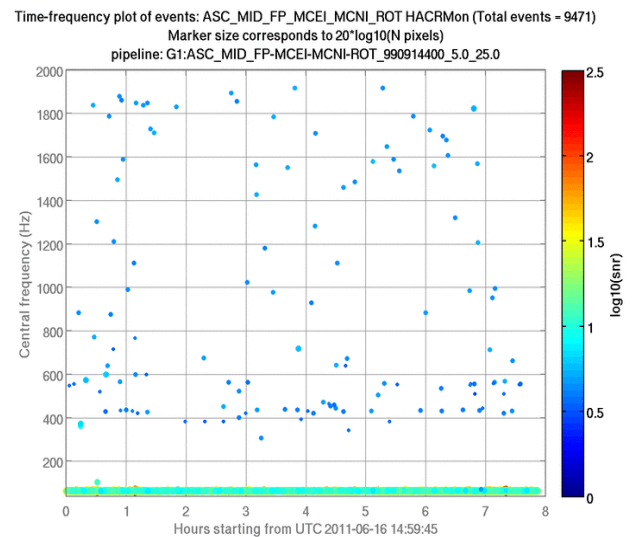


Figure 5: This may not seem like many triggers, but strong bar at the bottom actually contains so many triggers, the total comes to about 9500. This is also not informative since we only see one strong frequency band.

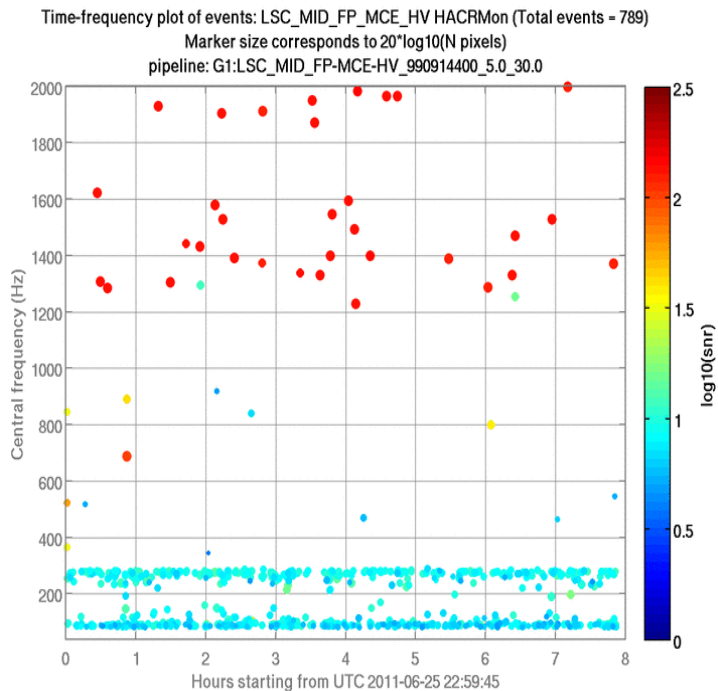


Figure 6: HACR plot only going to 2000 Hz
Firstly, there aren't enough triggers to tell much, but one is also missing quite a bit of activity

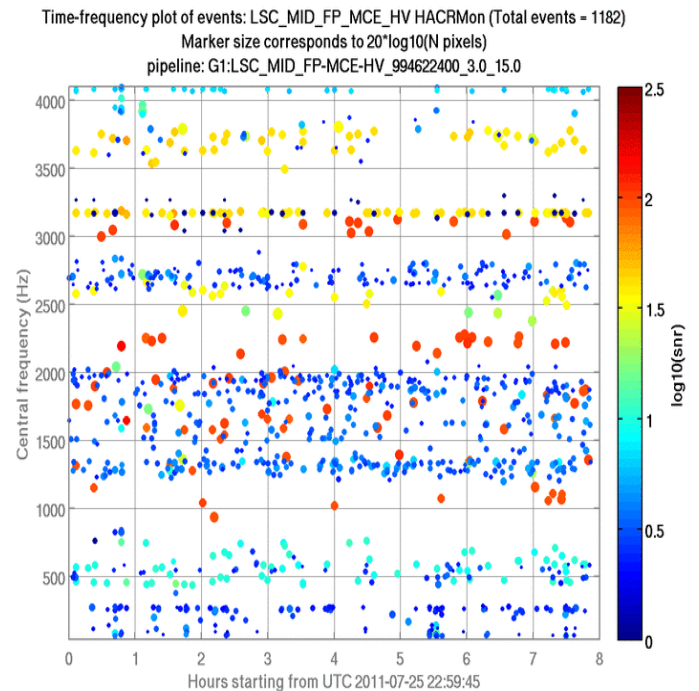


Figure 7: HACR plot going up to 4096 Hz
In the updated plot, it's incredible how much high frequency activity was missed beforehand!

2.3) Not enough channels

Originally, HACR was watching over about 35 channels, focusing mainly on seismic isolation and length sensing control channels, but there were many others that we were interested in seeing. These channels included even more length sensing controls channels, several suspension channels, and quite a few power related channels. These channels required not just tuning, but also needed relevant frequency ranges, and new filters (3.1) to be created. I worked to bring an additional twenty channels to HACR and also left documentation on adding any additional channels.

3.) Improvements

In this section, I'll describe the exact steps in improving older channels and adding new ones. The code mentioned will be included in appendices.

3.1) Filters

Many of the channels have particular frequency bands that receive too much noise, such as the $h(t)$ data, which has a large amount of noise in the low-frequency bands. In order to lessen this affect, filters are created to suppress specific narrow frequency bands. With these filters in place, we can actually find significant triggers outside of the loud noise regions. The filters are made using a MATLAB script (Appendix A) developed by Martin Hewitson to work with the `ltpda` toolbox used for data acquisition. For all of the channels originally on HACR, filters already existed, but more up-to-date filters were desired with the expanded frequency ranges, so these had to be recreated. Fortunately, since these channels have already been run through HACR in the past, the results could be compared to make sure the filters worked correctly. For new channels, filters had to be created for the first time.

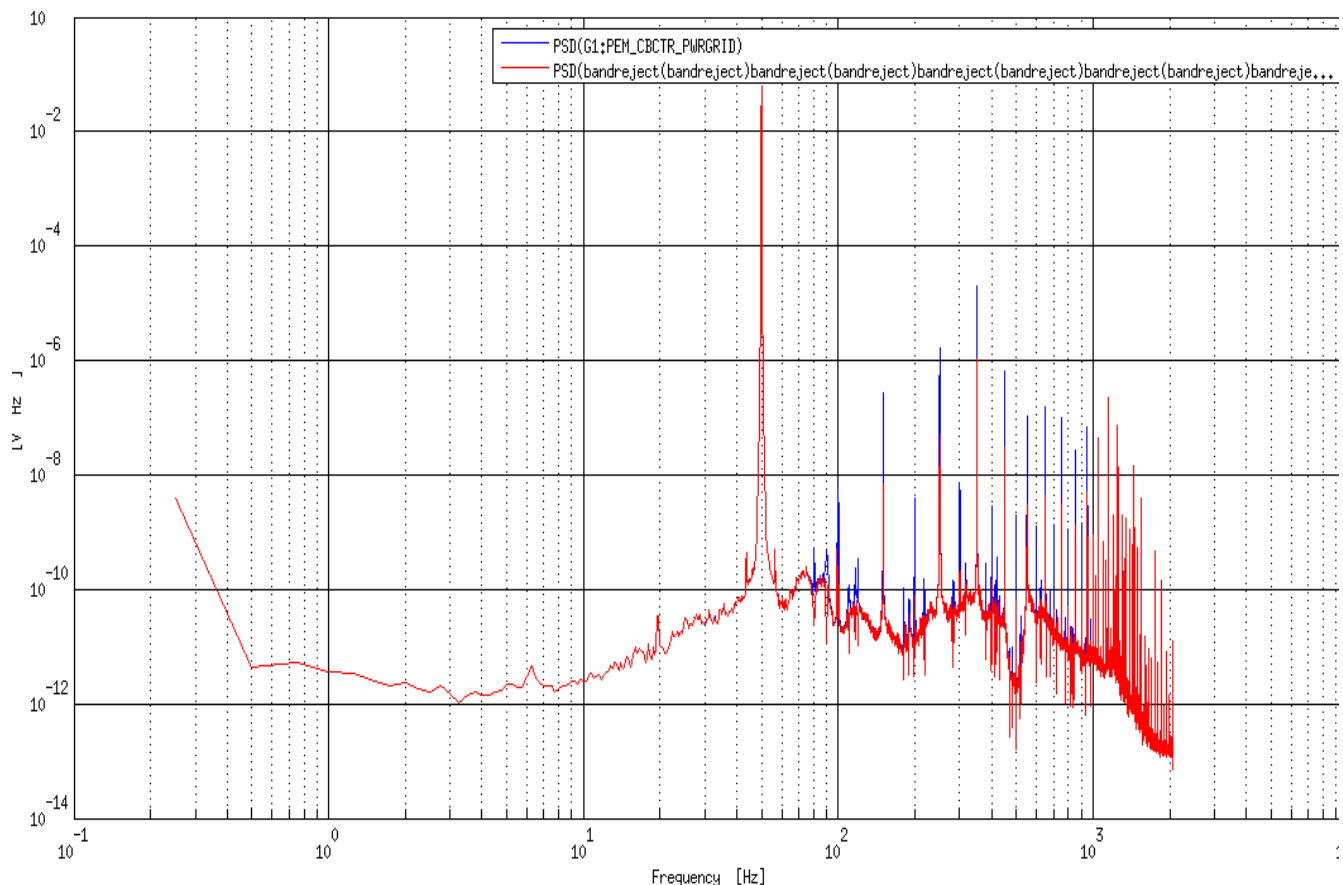


Figure 8: The blue lines represent the peaks before filtering and the red is afterwards

3.2) Retuning old channels

To actually run HACR, there is a server with all the filters, the HACR code, and the .job files with the configurations for the channels. While we are trying to tune the channels to get about one thousand triggers for eight hours, it would take far too long to test every configuration on eight hour stretches of data, so instead I looked for at least 250 triggers in two hour stretches. I therefore had to make a new configuration file for every channel after making the filters. In the file, I adjusted the frequency range to whatever was requested for that channel. I also had to find a recent two hour stretch with normal activity on that channel, so that the tuning would be more accurate. Then, I would normally just run HACR on the job file with the original thresholds for the sake of comparison with older plots. If the chosen times were okay, the plot would normally look very similar in the old frequency ranges they both cover. After this, I would have to change the threshold numbers in order to get less or more triggers. Going back to the mountain analogy in section 1.5 on HACR, the thresholds have different effects on the final plots. If the upper threshold is increased, less events are defined as mountains and the probability of a cluster having a mountain in it decreases, resulting in less triggers and vice-versa. The lower threshold doesn't have a strong influence on the number of triggers, but rather on the size of clusters. In order to change the number of triggers, I would therefore change the upper threshold.

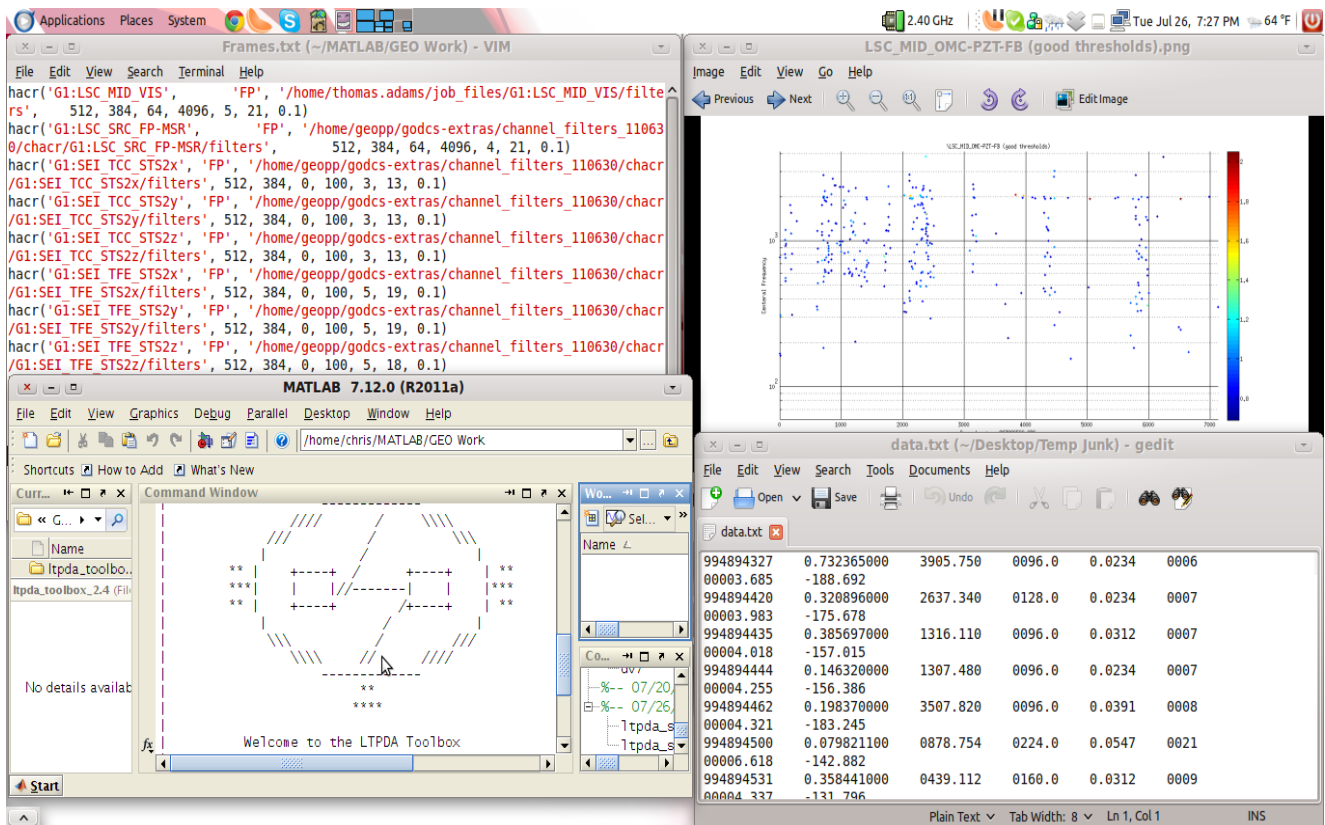


Figure 9: Threshold tuning cycle

Filters were first created using MATLAB (low-left) makeLineFilters.m script. Jobs were then set up (high-left) on the Fluffier server and ran using HACR which gave back the data (low-right) and the plots (high-right) were then created.

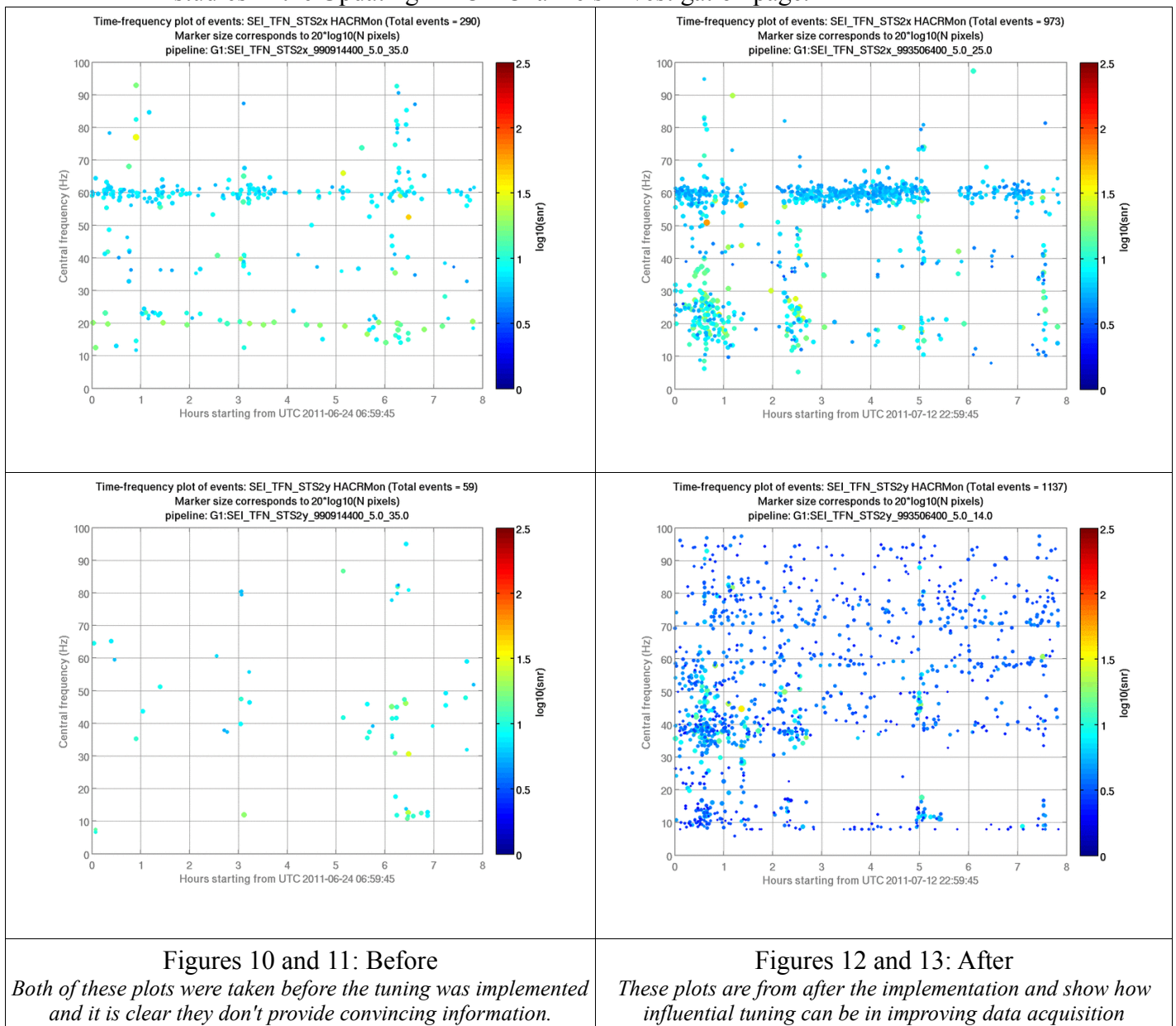
3.3) New Channels

After completing and double-checking the HACR tuning for the older channels, I requested that my configuration file be used for the real HACR job that is continuously run. The new configuration was put in place and the results were quite effective, so we immediately began to add more channels. The process for adding new channels was very similar. The main difference was there were no graphs to compare my results to, so I had to be careful and look over all the HACR plots to make sure the triggers were plentiful and informative.

4.) Results

4.1) Comparison

A full list of updated channels is on the GEO DC website [3] under glitch studies in the Updating HACR Channels investigation page.



4.2) Looking towards the future of HACR channels

Even after completing the first retuning on the HACR channels and adding so many more, I can see there is still much to be done in the future. While the trigger rates were certainly improved, they were still tuned on limited times and a more thorough approach would be to develop a script that finds good thresholds by analyzing and averaging many HACR jobs on different times. It was generally believed that developing a script like that would take much more time than I had available and would be complex due to the dual threshold system of event acceptance. A less time consuming approach would be to just re-tune channels that are getting too many or too few triggers by choosing another time that seems normal and average between previous results. I did this for some of the channels I had tuned and added that were slightly off. I found that some of the times I had chosen for channels happened to be really quiet or even in the middle of large glitches, so retuning was necessary. Either way, any improvement to HACR is a benefit to the entire GEO community. With more useful channels in HACR, glitches can be better understood and categorized by the detector characterization group. As this tool is developed even more, it could be used to increase the sensitivity of the GEO600 interferometer by removing more noise from our data, bringing us closer to the actual detection of gravitational waves.

Acknowledgement

I have so many people to thank for making this entire summer possible! Firstly, I thank my MIT LIGO supervisor, Gregg Harry, for recommending me to this research program and being my first mentor in gravitational waves. I'd also like to thank the University of Florida and the National Science Foundation for respectively organizing and funding this IREU. Specifically, I thank Prof. Bernard Whiting, Prof. Guido Mueller, Antonis Mytidis, and Kristin Nichola for giving me this opportunity and having it so well prepared. Next, I wouldn't have been able to complete anything had it not been for the entire GEO600 group. Thank you for letting me practice my broken German occasionally and for the excellent lunches! Thank you to Jacob Slutsky and Thomas Adams for guiding me through so much of my project and showing me a good time with the help of the person I must thank the most. Jonathan Leong, thank you for being a great friend and for teaching me so much about gravitational waves, GEO600, and the wonderful city of Hanover!

References

- [1] Hewitson, Martin. "On aspects of characterising and calibrating the interferometric gravitational wave detector, GEO 600", Ph.D. thesis, University of Glasgow, 2004
- [2] Parameswaran, Ajith. "On aspects of gravitational-wave detection: Detector characterisation, data analysis and source modelling for ground-based detectors", Ph.D. thesis, Max-Planck-Institut fur Gravitationsphysik (Albert-Einstein-Institut) und Institut fur Gravitationsphysik, Leibniz Universit " at Hannover, 2007
- [3] <https://wiki.ligo.org/GEODC/WebHome>

Appendix A: makeLineFilters.m script (written by Martin Hewitson)

```
%% Channel settings

server    = '130.75.117.73';
% server   = 'localhost';
port      = 9000;
% cal      = 0;
% rds      = 0;
channel    = '[insert channel name]';
start      =
geo.utils.timetools.UTC2GPS('2011-07-04
11:00:00');
% cal      = 1;
% rds      = 9;
nsecs     = 200;
% line settings
f1         = '[insert beginning frequency]';
% search from here...
f2         = '[insert ending frequency]'; %
to here.
thresh = 2;    % threshold for peak
detect
Nlines = 60;   % make filters for
largest N lines
% filter settings
fbw = 8;  % width of notches (Hz)


---


%% Get data
% get prime data
mid = ao(plist('built-
in','geoclient','type','frames','hostnam
e', server, 'channels', channel,
'nsecs', nsecs, 'starttimes', start));

%mid.data.y = cast(mid.data.y,'double');
% % need to highpass sometimes (ex:
%der_data with uneven spectrums)
% hp = miir(plist('type', 'highpass',
'order', 5, 'gain', 1, 'fs',mid.fs,
% 'fc', 120));
% % prime filter
% mid = filter(mid, hp);

% % chop transient
% mid = split(mid, plist('times', [10
inf]));


---


%% spectral analysis
nfft = 4*mid.fs;
psd_pl = plist('nfft', nfft, 'win',
'hanning');
pxx = psd(mid, psd_pl);


---


%% look for spectral lines
lines = linedetect(pxx, plist('N', Nlines,
'fsearch', [f1 f2], 'thresh', thresh,
'bw', 16, 'hc', 0.8));
iplot(pxx, lines, plist('markers', {'',
'o'}, 'linestyles', {'-', 'none'}))


---


%% Make filters for the largest lines
Nfilt = length(lines.x);
clear brf;
for jj=1:Nfilt
    ff = lines.x(jj);
    fname = sprintf('filt_%d_%.1f', mid.fs,
ff);
    brf(jj) = miir(plist('type',
'bandreject', 'order', 1, 'gain', 1,
'ripple', 0.1, 'fs', mid.fs, 'fc', [ff-fbw/2
ff+fbw/2]));
    exportMfilt(brf(jj),
plist('filename', fname))
end


---


%% Filter data
mid_f = filter(mid, brf, plist('bank',
'serial'));
mid_f_xx = mid_f.psd(psd_pl);
iplot(pxx, mid_f_xx)


---


%% check with hacr resolution
mid_hacr = psd(mid, plist('nfft', 512,
'scale', 'asd'));
mid_f_hacr = psd(mid_f, plist('nfft', 512,
'scale', 'asd'));
iplot(mid_hacr, mid_f_hacr)
```

Appendix B: List of channels and tuning

List of jobs

hacr('channel', 'DS', 'filters', nfft, nlap, fmin, fmax, lth, uth, outlier)

- hacr('G1:ASC_MID_FP-MCEI-MCNI-ROT', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/ASC_MID_FP-MCEI-MCNI-ROT/filters', 512, 384, 64, 4096, 9, 55, 0.1)
- hacr('G1:ASC_MID_FP-MCEI-MCNI-TILT', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/ASC_MID_FP-MCEI-MCNI-TILT/filters', 512, 384, 64, 4096, 7, 38, 0.1)
- hacr('G1:LSC_MIC_FP-PC-MU3', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MIC_FP-PC-MU3/filters', 512, 384, 64, 4096, 6, 28, 0.1)
- hacr('G1:LSC_MIC_FP-MMC2B', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MIC_FP-MMC2B/filters', 512, 384, 64, 4096, 5, 27, 0.1)
- hacr('G1:LSC_MIC_EP', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MIC_EP/filters', 512, 384, 64, 4096, 5, 28, 0.1)
- hacr('G1:LSC_MID_OPN', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_OPN/filters', 512, 384, 64, 4096, 5, 19, 0.1)
- hacr('G1:LSC_MID_OAN', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_OAN/filters', 512, 384, 64, 4096, 3, 13, 0.1)
- hacr('G1:LSC_MID_FP-MCEI-MCNI', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_FP-MCEI-MCNI/filters', 512, 384, 64, 4096, 5, 26, 0.1)
- hacr('G1:LSC_MID_FP-MCE-MCN', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_FP-MCE-MCN/filters', 512, 384, 64, 4096, 3, 17, 0.1)
- hacr('G1:LSC_MID_FP-MCE-HV', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_FP-MCE-HV/filters', 512, 384, 64, 4096, 3, 15, 0.1)
- hacr('G1:LSC_MID_FP-MCN-HV', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_FP-MCN-HV/filters', 512, 384, 64, 4096, 3, 16, 0.1)
- hacr('G1:LSC_MID_VIS', 'FP', '/home/thomas.adams/job_files/LSC_MID_VIS/filters', 512, 384, 64, 4096, 5, 24, 0.1)
- hacr('G1:LSC_SRC_FP-MSR', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_SRC_FP-MSR/filters', 512, 384, 64, 4096, 4, 21, 0.1)
- hacr('G1:SEI_TCC_STS2x', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TCC_STS2x/filters', 512, 384, 0, 100, 3, 13, 0.1)
- hacr('G1:SEI_TCC_STS2y', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TCC_STS2y/filters', 512, 384, 0, 100, 3, 13, 0.1)
- hacr('G1:SEI_TCC_STS2z', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TCC_STS2z/filters', 512, 384, 0, 100, 3, 13, 0.1)
- hacr('G1:SEI_TFE_STS2x', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TFE_STS2x/filters', 512, 384, 0, 100, 5, 19, 0.1)
- hacr('G1:SEI_TFE_STS2y', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TFE_STS2y/filters', 512, 384, 0, 100, 5, 19, 0.1)
- hacr('G1:SEI_TFE_STS2z', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TFE_STS2z/filters', 512, 384, 0, 100, 5, 18, 0.1)
- hacr('G1:SEI_TFN_STS2x', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TFN_STS2x/filters', 512, 384, 0, 100, 5, 25, 0.1)
- hacr('G1:SEI_TFN_STS2y', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TFN_STS2y/filters', 512, 384, 0, 100, 5, 14, 0.1)
- hacr('G1:SEI_TFN_STS2z', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SEI_TFN_STS2z/filters', 512, 384, 0, 100, 5, 20, 0.1)
- hacr('G1:PEM_CBCLN_ACOU-M', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PEM_CBCLN_ACOU-M/filters', 512, 384, 64, 4096, 4, 22, 0.1)
- hacr('G1:PEM_NBCLN_ACOU-M', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PEM_NBCLN_ACOU-M/filters', 512, 384, 64, 4096, 5, 21, 0.1)
- hacr('G1:PEM_EBCLN_ACOU-M', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PEM_EBCLN_ACOU-M/filters', 512, 384, 64, 4096, 4, 23, 0.1)
- hacr('G1:PEM_TCIB_MAG-X', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PEM_TCIB_MAG-X/filters', 512, 384, 64, 1000, 5, 32, 0.1)

- hacr('G1:PEM_TFE_MAG-X', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PEM_TFE_MAG-X/filters', 512, 384, 64, 1000, 3, 18, 0.1)
- hacr('G1:PEM_TFE_MAG-Z', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/chacr/PEM_TFE_MAG-Z/filters', 512, 384, 64, 1000, 5, 24, 0.1)
- hacr('G1:PSL_SL_PWR-AMPL-OUTLP', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PSL_SL_PWR-AMPL-OUTLP/filters', 512, 384, 64, 4096, 5, 24, 0.1)
- hacr('G1:PEM_CBCTR_PWRGRID', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PEM_CBCTR_PWRGRID/filters', 512, 384, 64, 1000, 100, 300, 0.1)
- hacr('G1:PSL_SL_PWR-AMPL2-INLP', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PSL_SL_PWR-AMPL2-INLP/filters', 512, 384, 64, 4096, 4, 23, 0.1)
- hacr('G1:LSC_MIC_VIS', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MIC_VIS/filters', 512, 384, 64, 4096, 4, 22, 0.1)
- hacr('G1:PSL_PWR_EASTARM-BSAR', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/PSL_PWR_EASTARM-BSAR/filters', 512, 384, 64, 4096, 4, 20, 0.1)
- hacr('G1:SUS-BD01_FLAG1_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_BD01_FLAG1_AC_DQ/filters', 512, 384, 0, 512, 5, 26, 0.1)
- hacr('G1:SUS-BD01_FLAG3_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_BD01_FLAG3_AC_DQ/filters', 512, 384, 0, 512, 5, 25, 0.1)
- hacr('G1:SUS-BS_FLAG1_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_BS_FLAG1_AC_DQ/filters', 512, 384, 0, 512, 5, 23, 0.1)
- hacr('G1:SUS-BS_FLAG3_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_BS_FLAG3_AC_DQ/filters', 512, 384, 0, 512, 5, 25, 0.1)
- hacr('G1:SUS-MCE_FLAG1_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MCE_FLAG1_AC_DQ/filters', 512, 384, 0, 512, 6, 35, 0.1)
- hacr('G1:SUS-MCE_FLAG3_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MCE_FLAG3_AC_DQ/filters', 512, 384, 0, 512, 6, 36, 0.1)
- hacr('G1:SUS-MCN_FLAG1_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MCN_FLAG1_AC_DQ/filters', 512, 384, 0, 512, 5, 20, 0.1)
- hacr('G1:SUS-MCN_FLAG3_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MCN_FLAG3_AC_DQ/filters', 512, 384, 0, 512, 5, 20, 0.1)
- hacr('G1:SUS-MPR_FLAG1_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MPR_FLAG1_AC_DQ/filters', 512, 384, 0, 512, 6, 54, 0.1)
- hacr('G1:SUS-MPR_FLAG3_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MPR_FLAG3_AC_DQ/filters', 512, 384, 0, 512, 7, 45, 0.1)
- hacr('G1:SUS-MSR_FLAG1_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MSR_FLAG1_AC_DQ/filters', 512, 384, 0, 512, 5, 30, 0.1)
- hacr('G1:SUS-MSR_FLAG3_AC_DQ', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MSR_FLAG3_AC_DQ/filters', 512, 384, 0, 512, 6, 30, 0.1)
- hacr('G1:SUS_MFE_flag3-AC', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MFE_flag3-AC/filters', 512, 384, 0, 512, 5, 16, 0.1)
- hacr('G1:SUS_MFN_flag-AC', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/SUS_MFN_flag-AC/filters', 512, 384, 0, 512, 5, 16, 0.1)
- hacr('G1:LSC_OMC_EP', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_OMC_EP/filters', 512, 384, 64, 4096, 4, 22, 0.1)
- hacr('G1:LSC_OMC_FP', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_OMC_FP/filters', 512, 384, 64, 4096, 5, 23, 0.1)
- hacr('G1:LSC_MID_OMC-PZT-FB', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/LSC_MID_OMC-PZT-FB/filters', 512, 384, 64, 4096, 6, 35, 0.1)
- hacr('G1:ASC_MCN_SPOT-PWR', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/ASC_MCN_SPOT-PWR/filters', 512, 384, 0, 100, 5, 16, 0.1)
- hacr('G1:ASC_MCN_SPOT-X', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/ASC_MCN_SPOT-X/filters', 512, 384, 0, 100, 5, 15, 0.1)
- hacr('G1:ASC_MCN_SPOT-Y', 'FP', '/home/geopp/godcs-extras/channel_filters_110630/chacr/ASC_MCN_SPOT-Y/filters', 512, 384, 0, 100, 5, 15, 0.1)