SURF project at

California Institute of Technology

Modeling the Performance of Networks of Gravitational-Wave Detectors in Gravitational-Wave Burst Searches

Mentor
Dr. Patrick Sutton

Student
Maria Principe

September 2005

LIGO-T050224-00-R

At present several large-scale interferometric gravitational wave detectors including GEO, LIGO, TAMA and Virgo are operating or are being commissioned. A cooperative analysis by these observatories could be valuable for making confident detections of gravitational-waves and for extracting maximal information from them. This analytical pipeline is particularly useful for gravitational-wave bursts (GWBs) search, for which theoretical knowledge of the source and the resulting gravitational waveform is limited. Systems generating gravitational-wave bursts are core-collapse supernovae [8, 9,10, 11], black hole merger [12, 13], and gamma-ray bursters [14].

Such cooperative analysis presents several advantages. Coincident observations from interferometers placed in different sites cause a decreased background from random detector noise fluctuations, that is a decreased false alarm rate, an increase in the total observation time during which some minimum number of interferometer are operating and the possibility of locating a source on the sky and extracting polarization information. Independent observations using different detector hardware and software also decrease the possibility of error or bias.

Joint search presents also some disadvantages. Most notably, in a straightforward coincidence analyses the sensitivity of a network is limited by the least sensitive detector. In addition, differences in alignment of the detectors (it means that different detector are sensitive to different combinations of the two polarization components of the gravitational-wave) complicates attempts to compare signal amplitude or waveform as measured by different interferometer. Also the differences in hardware, software and algorithms make the collaborative analysis technically challenging [5].

Generally GW search codes have a single threshold which is varied to tune the analysis; lower thresholds allow weaker signals to be seen, but also allow higher false alarm rates from fluctuations in the noise background. Typically multi-detector GWB searches are tuned according the Nyman-Pearson criterion, which states that decision rule should be constructed in order to have the highest network efficiency, while not allowing network false alarm rate to exceed a certain fixed value. For example, LIGO searches are typically tuned for a false alarm rate of 0.1 events or less over the observation time [5]. To date this tuning is typically done “by hand”, which can lead to sub-optimal tuning (i.e. poorer detection efficiency). In addition, there is an inherent ambiguity in which kind of signal should be used to estimate the efficiency. Without a model for the network it is difficult to quantify the effects of uncertainties or possible errors in the description of detectors. Truly optimal tuning should be done using a quantitative model for the detector network. This is particularly important for searches involving detector with different sensitivities and different orientation. Developing tools for such a quantitative model has been the goal of this project.

This project was oriented to model the performance of networks of gravitational-wave detectors in trigger-based search for gravitational-wave bursts (GWBs). This involved creating a software package that can estimate the efficiency and the false alarm probability of a general network of gravitational-wave detectors. The main target of such a network simulator is tuning of analyses in actual network GW searches, to maximize network sensitivity, or, in other words, to find the best power threshold set satisfying *Neymann-Pearson criterion*. Further, it could be used to estimate sensitivity to populations of signals other than those which were directly tested in the search. Finally, the software tool could be used to quantify the effect, over the entire network, of the uncertainties or possible errors in the properties of the individual detectors (such as in calibration, or due to finite-number statistics in measurements of sensitivities).
The simulation engine was constructed in Matlab, and made publicly available. In the following the used approach and methods are outlined.

Data considered to test the tool were collected during S2 LIGO-TAMA analysis for GWBs search [5]. As part of the LIGO-TAMA search analysis, to estimate the sensitivity of LIGO-TAMA heterogeneous network simulated gravitational-wave signals were added (or “injected”) to the data stream from each detector and re-analyzed. This gave four sets of candidate signals (“event triggers”): analyses run 14 (playground data) and 17 (total data) include injections, while run 15 (playground data) and run 18 (total data) don’t include injections. Injected signals belong to the population of linearly polarized Gaussian-modulated sinusoids:

$$h_+(t) \propto h_{rss} \sin\left[2\pi f_0 \left(t - t_0\right)\right] e^{-\frac{(t-t_0)}{\tau^2}}$$
$$h_x(t) = 0$$

Here t_0 is the peak time of the signal envelope and f_0 is the central frequency of the injection which is picked randomly from the values 700, 849, 1053, 1304, 1615, 2000 Hz. The quantity h_{rss} is the root-sum-square amplitude of the plus polarization and it is found to be a convenient measure of the signal strength:

$$\left[\int_{-\infty}^{\infty} \partial t \, h_+^2(t)\right]^{1/2} = h_{rss}$$

1. Network efficiency estimation

The first step was to develop the software to estimate *efficiency curve* for a general network of gravitational wave detectors, which is the probability for a GWB with specific amplitude to be detected by all of the interferometers. This task was achieved writing first a Matlab function, *DetectorEfficiency.m*, to estimate the single detector *optimally oriented* efficiency curve as function of the SNR threshold and the observable amplitude, that is essentially the product of true injected amplitude h_{rss} by the antenna response factor corresponding to the source and polarization angle of the injection. Basically *DetectorEfficiency.m* function searches for time coincidences between trigger events collected by the interferometer and injected simulated GWB signals. For the events in time coincidence with an injection, the function checks the SNR value to be higher than the specified threshold and if so, the injected signal is considered detected, otherwise the detector wasn’t able to see that signal. The higher this threshold, the lower the false alarm rate, but the higher the probability that a weak GW signal cannot be seen. The fitting of the curve was done using a sigmoid function. *DetectorEfficiency.m* function has been used to obtain efficiency curves for each detector belonging to the network both averaged over all injected signals and for each kind of injected signals. Data with injections were used to obtain curves relative to the detectors H1, H2 and L1, shown below.

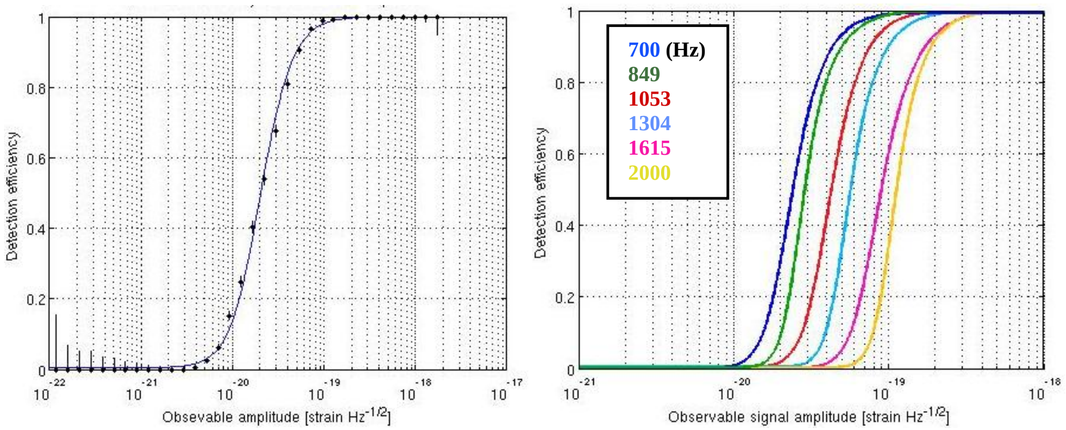

Figure 1. H1 efficiency for no power threshold averaged on all injected signals (left) and for each knd of them (right). Uncertainty in the latter picture is greater than former one because of the smaller number of simulations.

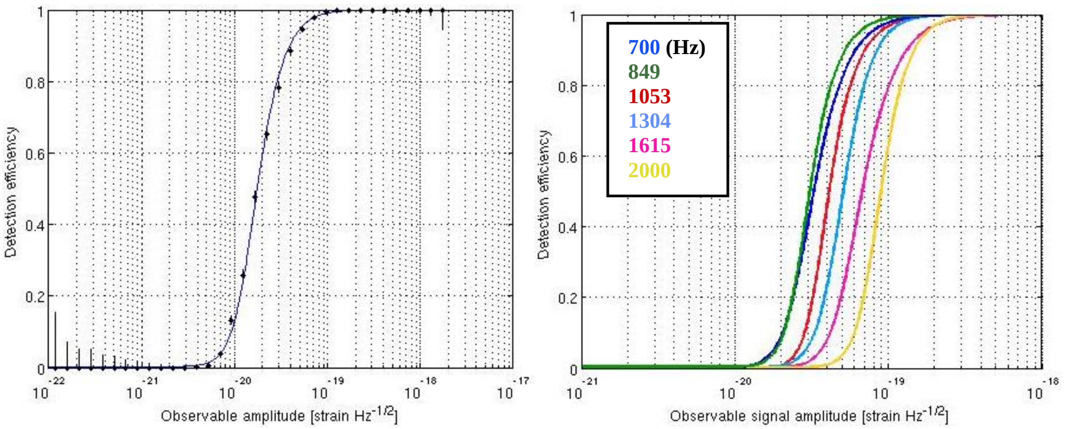

Figure 2. H2 efficiency for no power threshold averaged on all injected signals (left) and for each knd of them (right). Uncertainty in the latter picture is greater than former one because of the smaller number of simulations.

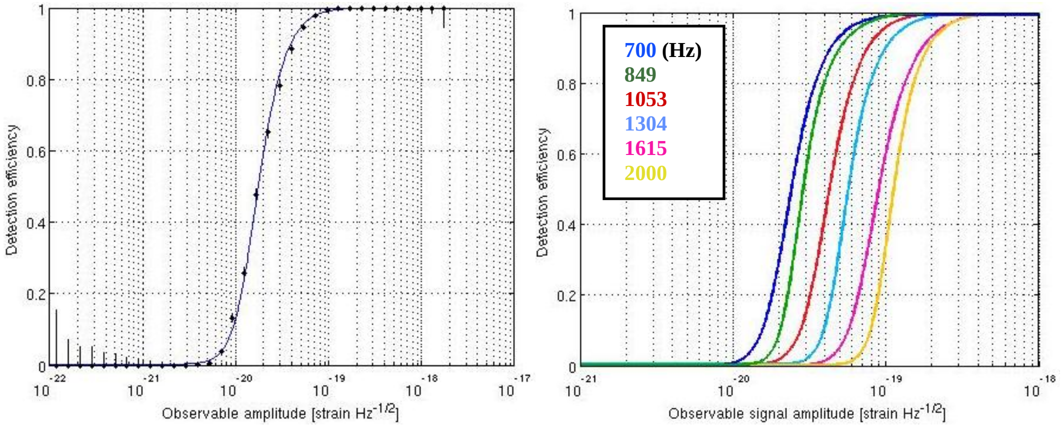

Figure 3. L1 efficiency for no power threshold averaged on all injected signals (left) and for each knd of them (right). Uncertainty in the latter figure is greater than former one because of the smaller number of simulations.

Detector efficiency curves above-shown, are obtained fixing SNR threshold to zero. SNR provided by *Event Trigger Generator* (ETG) algorithm, which can be different for each detector, is here meant as a measure of signal strength and sometimes, especially if noise is fluctuating, this measure can result to be zero.

Varying SNR threshold detection efficiency is expected to change, in particular increasing the threshold, detection probability is expected to decrease and this can be seen from following figures, representing H1 efficiency for SNR threshold equal to 0, 5 and 10.

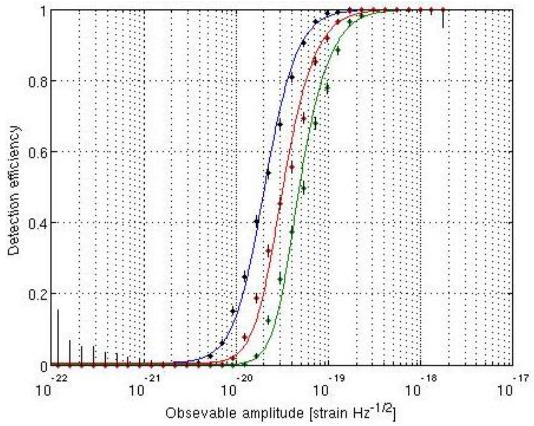

Figure 4. H1 efficiency for different values of SNR threshold: 0 (blu line), 5 (red line), 10 (green line).

These curves were used to estimate the network efficiency, which is the probability for a GW signal to be seen by all detectors. In theory this can be accomplished by computing the following integral, which basically computes the probability that all the interferometers see a GW signal, with specific amplitude, averaged over source angle and polarization angle.

$$E_{nw}(h_{rss}) = \int_0^{\pi} \partial\psi \int_0^{2\pi} \partial\varphi \int_0^{\pi} \partial\theta \sin\theta \prod_{i=1}^{N} E_i(|h_{obs}(\psi,\varphi,\theta)|)\, p(\varphi,\theta)\, p(\psi)$$
$$h_{obs}(\psi,\varphi,\theta) = F_+(\psi,\varphi,\theta)\cdot h_{rss} \qquad (1)$$

Here *p(ψ)* is probability distribution of polarization angle of GWB, while *p(φ,θ)* is the probability distribution of the source.

In order to compute the value of the previous integral a Matlab function *NetworkEfficiency.m* was developed. It works out the result through direct integration, that consists in approximate the integral with the following sum.

$$E_{nw}(h_{rss}) \approx \sum_{\psi}\sum_{\varphi}\sum_{\theta}\prod_{i=1}^{N} E_i(|h_{obs}(\psi,\varphi,\theta)|)\,p(\varphi,\theta)\,p(\psi)\sin(\theta)\Delta\theta\Delta\varphi\Delta\psi \qquad (2)$$

To implement the previous calculation choosing a grid of samples for the source angles and the polarization angle is required. Polarization angle, ψ, is uniformly sampled; concerning source angle, the function allows choosing among a uniform, a cosine-θ uniform, that is sampling φ and cosine θ uniformly, and a sinusoidal map sampling, which provides a uniform distribution of samples over the sphere. In the following, results with sinusoidal map sampling, choosing a number of θ samples equal to 22, are shown.

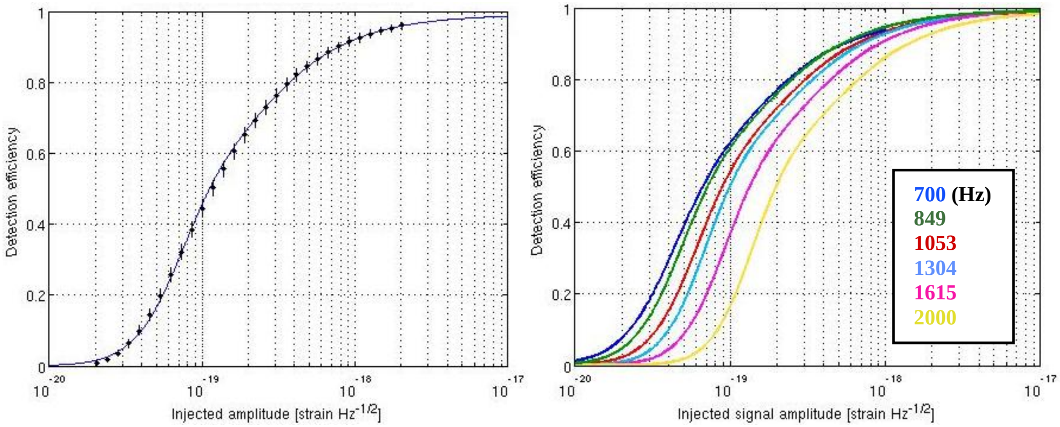

Figure 5. H1-H2-L1 network efficiency for no power threshold averaged over all injected signals (left) and for each injection (right). Uncertainty in the latter picture is greater then the former one because of the smaller number of simulations.

To have an idea of how SNR threshold set affects network efficiency, change this set from (0, 0, 0) into (0, 0, 5), i.e. make higher the threshold corresponding to L1, which is the least sensitive LIGO detector. Resulting curves for network efficiency are shown in following figures. You can see clearly from the center of sigmoid function, which increases roughly by an amount of 10%, that network efficiency makes worse.

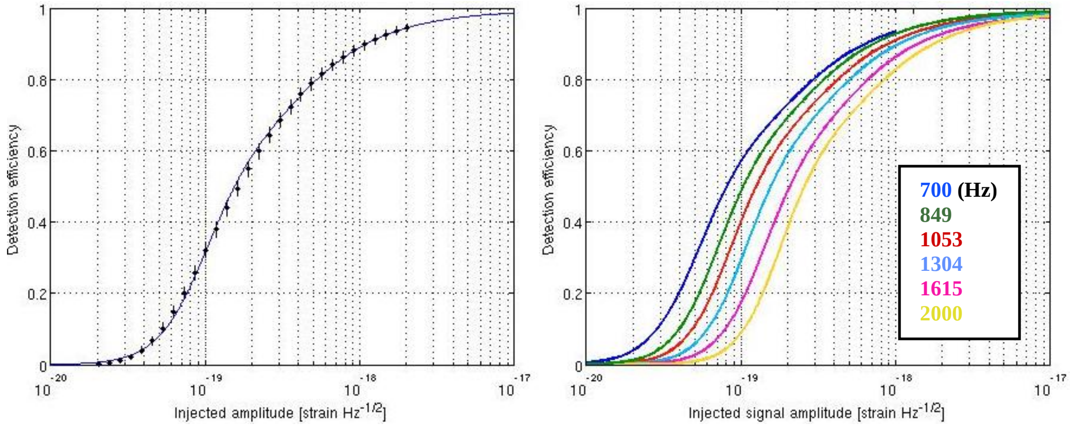

Figure 6. H1-H2-L1 network efficiency for SNR thresholds H1: 0, H2: 0, L1: 5.

2. Network False Alarm Rate

The next step of the project was to develop software to estimate network false alarm rate, which is the probability that a background event can occur in time, frequency and amplitude coincidence in all the detectors of the considered network. Amplitude comparison of observed signal amplitude is made hard by the differences in the alignment of the detectors. Not aligned detectors are sensitive to different combinations of the two polarization component, therefore amplitude cannot be compared in a trivial manner; simple comparison of observed amplitude can be done only for aligned detectors. The estimation of network false alarm rate was accomplished developing three function sets.

2.1 Time False Alarm Rate

The first set estimates the probability that a background event can occur in all the detectors in time coincidence. A function, *DetectorFalseRate.m*, was written to estimate for each detector time rate of background noise events occurring with SNR above specified threshold, with the assumption that the background is Poisson process. This function loads list of events not due to a GWB injection with SNR above fixed threshold and, based on the known total observation time, according to the formula shown below, estimates the time occurrence rate of background noise events in each detector.

$$R_t = \frac{N_e}{T_{obs}} \tag{3}$$

Obviously detector time false event rate depends on the fixed SNR threshold: higher it is, lower false alarm rate is. Following figures show false alarm rate as a function of SNR threshold value for LIGO detectors.

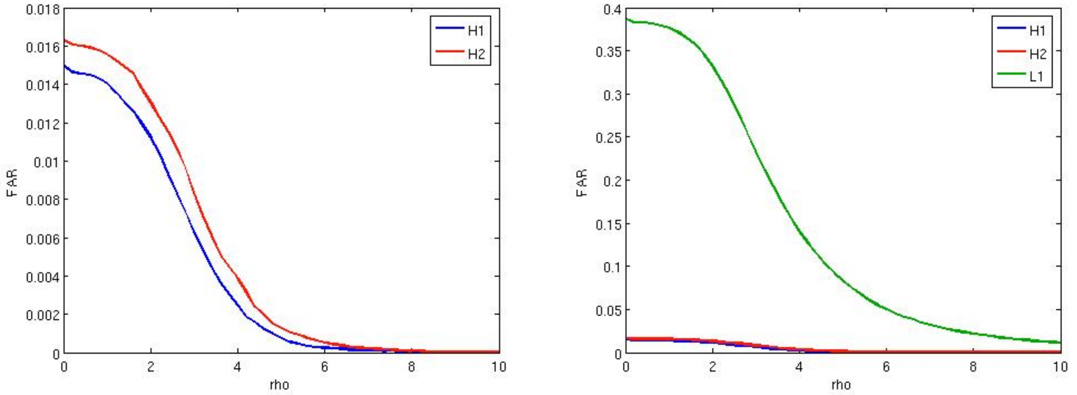

Figure 7. Time rate of background noise events occurring with SNR above specified threshold for LIGO detectors.

Successively, to estimate network time false alarm rate is fixed a coincidence window w_t, which accounts for travel light time between the detectors in question: in practice, for safety, w_t is chosen 10-20 ms longer than the light travel time. Events from two detectors are defined to be in coincidence if the difference in their peak times satisfies the following condition

$$\left|t_i - t_j\right| < w_t + \frac{1}{2}\left(\Delta t_i - \Delta t_j\right) \tag{4}$$

where t is the peak time and Δt is the duration of the signal. The duration-dependent term allows for the estimated peak time of coincident triggers to be further apart if the triggers are long compared to the coincidence window [5]; one may consider this as an allowance for the uncertainty in the determination of the peak time. A set of event triggers *i, j,.... , k* is defined to be in coincidence if each pair *(i, j), (i, k), (j, k), etc.*, is in coincidence. *NetworkTimeFalseRate.m* function estimates the probability that noise events generated by the detectors can be in time coincidence. Assuming the hypotheses that background noise in each detector is modeled as Poisson process, that $R_i\ w_t << 1$ and that coincidence window is the same for each pair of detector, the estimator for network time false alarm rate is

$$R_{t\_nw} = 2 w_t^{N-1} \prod_{i=1}^{N} R_i \tag{5}$$

To estimate false rates were used data not including injected simulated signals, to avoid injections affecting estimations. For no SNR threshold, in table 1 are shown values obtained for the LIGO network.

ρ = 0	H1	H2	L1	H1-H2-L1
R_t	0.0176	0.0153	0.3699	$1.5905\ 10^{-7}$

Table 1. Detector and netowrk time false rate for LIGO.

2.2 Frequency False Alarm Rate

The second function set was built with the intent to estimate the probability that background events can occur in different detectors in frequency coincidence. In order to estimate the probability density function of background events over central frequency and frequency bandwidth for each detector, *BackgroundFrequencyDistribution.m* function was developed; it considers background noise events detected with SNR above fixed threshold and, binning central frequency and frequency bandwidth, builds a 3-dimensal histogram counting the number of events for each frequency-bandwidth bin. Histograms resulting from analysis of run 15 data for LIGO detectors are shown in following figures.

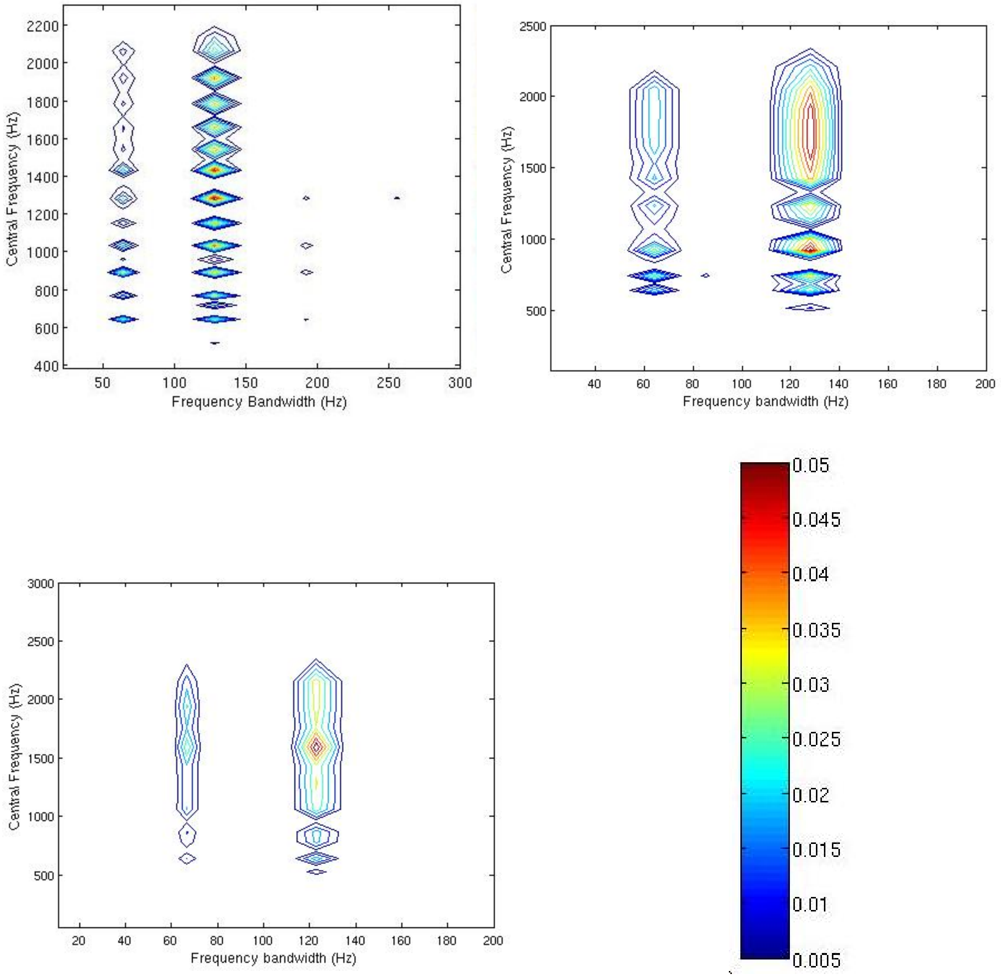

Figure 8. H1 (top left), H2 (top right) and L1 (bottom left) background noise ditribution measurement over central frequency and bandwidth from run 15 data, with no SNR threshold (The discrete structure happens because frequency and bandwidth of the triggers used are restricted to discrete values mulitples of 64 Hz).

The desired estimation is achieved by solving the following integral.

$$
\begin{gathered}
\int \partial f_1 \int \Delta f_1 \cdots \cdots \int \partial f_N \int \Delta f_N \, \Im(f_1, \Delta f_1, \ldots, f_N, \Delta f_N) \, p_1(f_1, \Delta f_1) \cdots p_N(f_N, \Delta f_N) \\
\Im(f_1, \Delta f_1, \ldots, f_N, \Delta f_N) = \begin{cases} 1 & \text{if } f_1, f_2, \ \ldots \ , f_N \quad \text{are in coincidence} \\ 0 & \text{otherwise} \end{cases}
\end{gathered}
\tag{6}
$$

The criterion to establish frequency coincidence between two events is very similar to that used for the coincidence in time: two events are defined to be in frequency coincidence if the following condition is true

$$\left|f_i - f_j\right| < w_f + a\left(\Delta f_i - \Delta f_j\right) \tag{7}$$

where *f* is the estimated central frequency value and *Δf* is the estimated frequency bandwidth. Multiple events are defined to be in coincidence if each pair of events is coincident. The integral (6) is solved through Monte Carlo method. For

- LIGO network
- $w_f = 0$; a = 0.5
- no SNR threshold
- 10^6 trials

The resulting estimation is $R_{f\_net} = 0.008159$.

2.3 Amplitude False Alarm Rate

Last step in computing network false rate is estimating network amplitude false alarm rate, which is the probability that background events occur with approximately the same amplitude in detectors. As mentioned earlier, this kind of test is applied only to detectors with the same orientation. Two events are defined to be in amplitude coincidence if

$$|H_i - H_j| < w_h \tag{8}$$

with H defined as log(h), where h is the signal amplitude measured by the detector.

To achieve desired estimation, we can proceed in the same way as done for the frequency false rate. *BackgroundAmplitudeDistribution.m* function estimates the probability density function of background events over amplitude, considering detected triggers not due to injected signals and whose corresponding SNR value is higher than specified threshold and building a histogram detailing the percentage of events occurred with specified amplitude. Histograms resulting from such a measure are shown in the following.

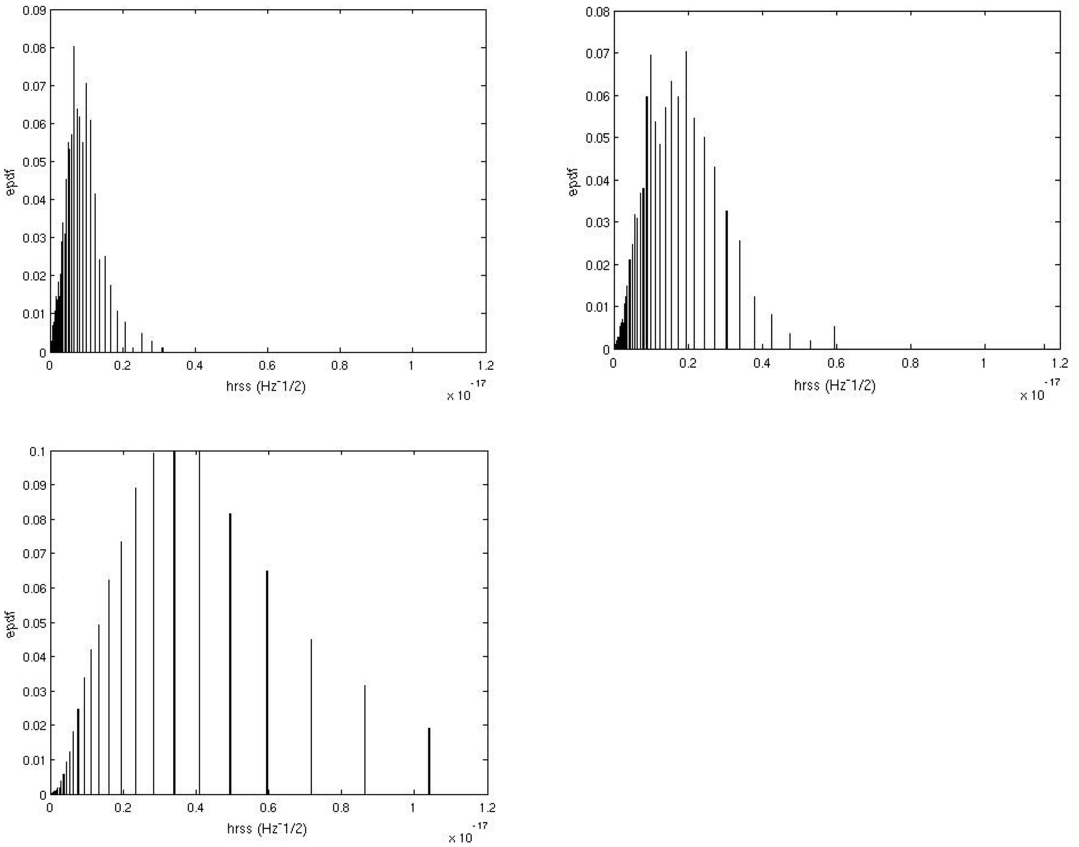

Figure 9. Measured background noise distribution over amplitude for LIGO detectors and with no SNR threshold.

The next step is done by *NetworkAmplitudeFalseRate.m* function, which performs, thorough Monte Carlo integration method, the following integral.

$$\int \partial h_1 \cdots \cdots \int \partial h_N \, \Im(h_1, h_N) p_1(h_1) \cdots p_N(h_N)$$

$$\Im(h_1, \dots, h_N) = \begin{cases} 1 & \text{if } h_1, \dots , f_N \quad \text{are in coincidence} \\ 0 & \text{otherwise} \end{cases} \qquad (9)$$

For

- LIGO network
- $w_h = 0.3$

- no SNR threshold
- 10^6 trials

The resulting estimation is $\hat{R}_{h\_net} = 0.4161$.

Once network time, frequency and amplitude false rate have been estimated, to obtain an estimation for the overall false rate is given by the product of this three quantities:

$$\hat{R}_{nw} = \hat{R}_{t\_nw} \cdot \hat{R}_{f\_nw} \cdot \hat{R}_{h\_nw}$$

This product rule is valid only in the hypothesis that time, frequency and amplitude coincidences are independent on each other. Applying this rule and considering previously computed quantities, network false events rate is $\hat{R}_{net} = 5.4\ 10^{-10}$. It's clear that the quantity $\hat{R}_{net}$ is dependent on SNR thresholds chosen for each detector. Suppose to fix H2 threshold value and to vary those corresponding to H1 and L1, variations in network false rate are shown in the following figure.

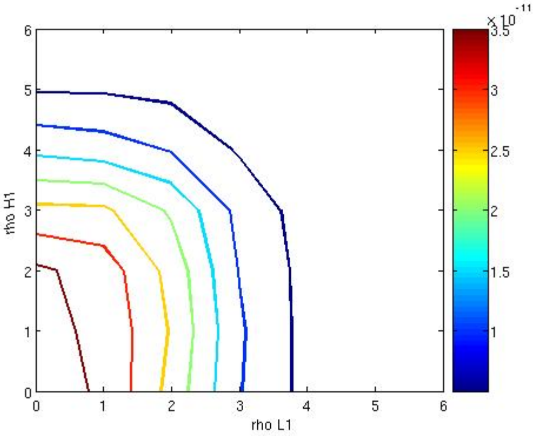

Figure 10. LIGO network false rate as a function of H1 and L1 SNR thresholds.

3. Find threshold SNR set satisfying Neymann-Pearson criterion

Once functions to estimate network efficiency and network false alarm rate were built, a master function was written to find the SNR threshold set satisfying Neymann-Pearson criterion. Principal steps are outlined. Fix a target network false alarm rate, choose a grid of SNR threshold set and check for sets allowing network false rate to be below fixed value. For selected sets compute network efficiency and choose the best one. The criterion to choose the best efficiency is to search for the curve having the lowest amplitude value, corresponding to a chosen efficiency value. The SNR threshold set corresponding to the best efficiency curve is the wanted set.

4. Future plans

An interesting feature to include is the possibility to choose different coincidence windows for each pair of detectors. Also the possibility to choose different SNR threshold set range for each detector can be included.

Further testing and optimization of the software tool are planned.

Other future tasks are applying developed tool to past search, for validation of results, and current search, to help it.

References

[1] Barish, Barry C. and Weiss, Ranier R, *LIGO and the Detection of Gravitational Waves*.

[2] Sigg, Daniel, *Gravitational Waves*. LIGO-P980007-00-D. Oct 1998.

[3] Thorne, Kip S., *Gravitational Waves*. arXiv:gr-qc/9506086 v1. Jun 1995.

[4] B. Abbott et al., *Upper limits on gravitational wave bursts in LIGO's second science run*. LIGO-P040040-07-R. arXiv:gr-qc/0505029. May 2005.

[5] B. Abbott et al., *Upper limits from the LIGO and TAMA detectors on the rate of gravitational-wave bursts*. LIGO-P040050-05-Z. arXiv:gr-qc/0507081. Jul 2005.

[6] By International Gravitational Event Collaboration (P. Astone et al.). *Methods and Results of the IGEC search for burst gravitational waves in the years 1997-2000.* Feb 2003. Published in Phys. Rev. D68:022001, 2003.

[7] By LIGO Collaboration (B. Abbot et al.), *First Upper Limits on Gravitational-Wave Bursts.* May 2004. Published in Phy. Rev. D69:102001, 2004.

[8] T. Zwerger and E. Muller, Astron. Astrophys. 388 917 (2002).

[9] H. Dimmelmeier, J. Font, and E. Muller, Astron. Astrophys. 320 209 (1997).

[10] H. Dimmelmeier, J. Font, and E. Muller, Astron. Astrophys. 320 209 (1997).

[11] C.Ott, A. Burrows, E. Livine, and R. Walder, Astrophys. J. 600 834 (2004).

[12] E. E. Flanagan and S. A. Hughes, Phys. Rev. D 57 4535 (1998).

[13] E. E. Flanagan and S. A. Hughes, Phys. Rev. D 57 4566 (1998).

[14] P. Meszaros, Ann. Rev. Astron. Astrophys. 40 3884 (1989).