

## Background

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 analysis presents several advantages. Coincident observations from interferometers placed in different sites cause a decreased background from random detector noise fluctuations, 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 is to achieve the lowest false dismissal probability (best detection efficiency) at a given false alarm rate. For example, LIGO searches are typically tuned for a false alarm rate of 0.1 events or less over the observation time. 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 is the goal of this project.

## **Goals**

This project is oriented to model the performance of networks of gravitational-wave detectors in trigger-based search for gravitational-wave bursts (GWBs). This would involve creating a software package that can estimate the efficiency and the false alarm probability of a general network of gravitational-wave detectors. The network simulator could be used for tuning of analyses in actual searches, to maximize their sensitivity. 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 estimate the effect, over the entire network, of the uncertainties in the properties of the individual detectors (such as in calibration, or due to finite-number statistics in measurements of sensitivities).

The simulation engine will be constructed in Matlab, and made publicly available. In the following the approach and the methods expected to be used are outlined.

## **Procedure**

In a general network of  $N$  detectors, each of them is characterized by its position and orientation on the Earth, and by an efficiency function (detection probability) for a given type of GWB signal. The efficiency is a function of the signal amplitude and the false

alarm rate. Typically, for fixed signal amplitude, the detection efficiency improves as the false alarm rate increases. A tuning for the network therefore consists of choosing values for the  $N$  false alarm rates for each detector. My optimization problem is to choose the set of false alarm rates which gives the best network sensitivity while giving a network false rate at or below some maximum allowed value.

It is possible to estimate the network false alarm rate (for false alarms coincident in time) from the single detector false alarm rates and the light travel time between the detector sites, determined by the known detector location on the Earth. I will also attempt to estimate the effect on the false alarm rate of consistency tests in frequency and amplitude which are also used in real searches. This estimate will use the measured distributions of the frequency and amplitude of events from each detector. The network sensitivity for simultaneous detection by all the detectors can be estimated using the known orientation of the interferometers, which determines the precise amplitude of the GWB signal received by each, and the efficiency of each as a function of received amplitude. This can be done by direct integration over possible sky positions and polarizations of simulated GW signals, or by Monte Carlo analysis.

I will construct a network simulation engine that models an arbitrary network (number, position and type) of the GW detectors. Given the efficiency  $E_i(R_i, h)$  for each detector as a function of false alarm rate  $R_i$ , and the signal amplitude  $h$ , the simulation engine will determine the optimal tuning  $R_i$  to satisfy the Nyman-Pearson criterion with a given network false alarm rate. This optimization may be done by searching over a grid of false alarm values, by using random tuning choices according to Monte Carlo method, or by some other technique. The simulation tool will also be able to simulate an ensemble of networks in which each detector has associated uncertainties or probability distributions for its efficiency; this allows to incorporate, e.g., calibration uncertainties in our tuning, or time-varying efficiencies, and to estimate their effect on the final efficiency of the network. The network simulation engine can be used for tuning of analyses, for validation of analyses (match between actual network performance and predicted performance for the given tuning), for evaluating effects of systematic errors, and for estimating the network sensitivity to different population of GWBs.

As part of its development and testing, this engine will be applied to past and current LIGO GWB searches.

## **Current Status**

In the first three weeks of my project I read several papers, listed in the bibliography, about the methods and results of the trigger-based GWB search in LIGO.

As first task, I brought some changes to Matlab functions, written by Dr. P. Sutton, for loading from file triggers lists, injection signals list and time intervals analyzed, in order to load them in an easier manner. Subsequently I wrote a function to compute detector efficiency function from sets of event triggers, produced by search codes, and from injected and analyzed signals. The computed efficiency is a function of received signal amplitude and of signal-to-noise ratio threshold, which determines the false alarm rate. At the moment I'm testing this function.

Targets for the next month are completing testing of the efficiency function, writing a function to compute network efficiency from individual detector efficiencies and for specified sky distribution of GWBs. Afterwards I will test engine by applying it to some of the first four LIGO bursts searches, and possibly others using actual tuning choices from the search. Further goals will be writing an "optimizer" function able to find tuning  $R_i$  that maximizes network efficiency for a given false alarm rate and comparing optimal tuning to actual tuning. Finally, the predicted optimal efficiency will be compared to the measured one.