DETECTION OF GRAVITATIONAL WAVES FROM TYPE II SUPERNOVA USING PRINCIPAL COMPONENT ANALYSIS

DAVID CURRY

Acknowledgements: Patrick Sutton, Cardiff University

1. Abstract

In this paper we consider rotating Type II supernova as candidates for gravitational wave emission, which includes an analysis of the waveform catalogs and possible methods of burst detection through the X-Pipeline computing structure. Type II supernovae are expected to occur approximately once of every two years in the range of 3-5 Mega parsecs, and while these events are rare, they are highly energetic and provide the conditions necessary for the creation of a gravitational wave within current detector limits. The drawback is the unknown nature of the microphysics governing the seconds before and after the core-collapse, which is when the key features of the gravitational wave burst are created. As a result there are numerous catalogs of possible waveforms describing type II supernovae, leading to difficulties in creating certain data analysis tools necessary for the detection of these bursts, such as templates, matched filters, and bayesian tests. In this paper we seek to generalize the various supernova waveforms into a small number of basis vectors to aid in computation and statistical analysis using the methods of principal component analysis. These principal components are then used within the x-pipeline burst search through the implementation of bayesian analysis, which uses the reduced basis vectors as a prior knowledge of the bursts.

2. Overview

- Principal Component Analysis.
  - Theory
  - Matlab Code
  - Results
- X-Pipeline.
  - Overview of a burst search
  - Introducing Bayesian statistics

3. Principal Component Analysis

3.1. **Theory.** As mentioned in the introduction the motivation for applying the methods of principal component analysis is to reduce the number of vectors needed to describe a given supernova catalog, as well as generalizing the data set. The reasoning here is twofold: the exact composition of the burst is unknown and so we would like to describe the different waveforms with a few "representative" waveforms, and we seek to reduce computation time both within matlab and x-pipeline. PCA(principal component analysis) will help us do just this.

The first step is to find the covariance matrix of a given catalog. All catalogs and PCA matrices are in the form n x m, where n is the number of data points and m is the number of waveforms in the catalog. For $_n$ dimensions, the first order covariance matrix(m =1) is

$$\mathbf{V}_{n,m} = cov(x_1, ..x_m) = E[(x_1 - u_1)(x_m - u_m)]$$

where E is the expectation value of the n,mth array element. Intuitively we understand the diagonal elements to be the variance between each vector and itself, self-correlation, and the off-diagonal elements are the cross-correlation terms. We would now like to understand this covariance matrix in terms of the "spread" of variance, rather than in the relation between data points and traditional x-y-z axis(ie., for gravity waves we would like to look at the data in terms of its' variance and not strain amplitude over time). To do this we find a new basis vector set by which to define the data by eigenvalue and vector decomposition of our covariance matrix.

For our covariance matrix $\mathbf{V}$, eigenvalue decomposition is easily described as

$$\mathbf{VP} = \mathbf{PD} \rightarrow \mathbf{V} = \mathbf{PDP}^{-}1$$

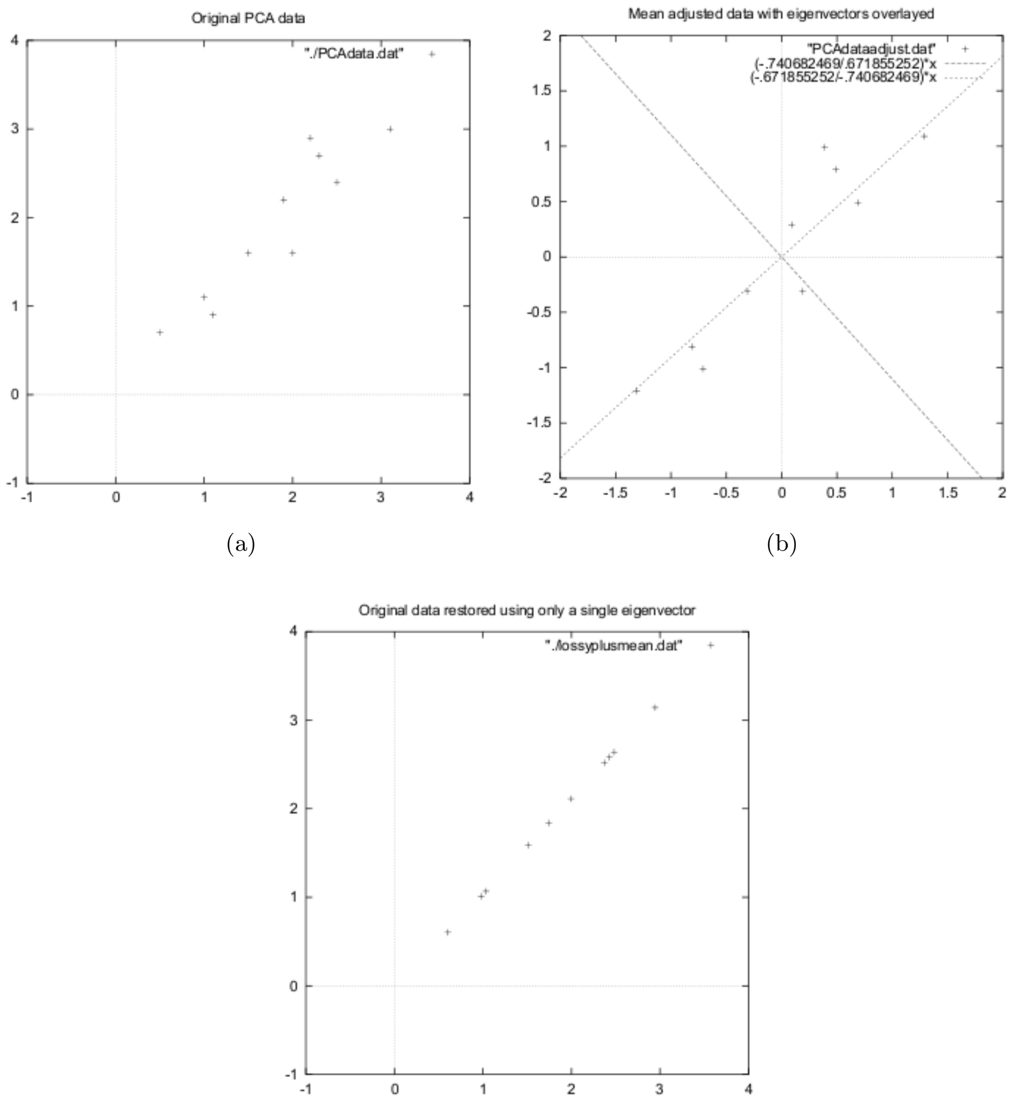
where, $\mathbf{P}$ is an array of eigenvectors,

$$\begin{pmatrix} x_1 1 & ... & x_1 m \\ ... & x_2 2 & f \\ x_n 1 & ... & x_n m \end{pmatrix}$$

and $\mathbf{D}$ is a scalar array of eigenvalues which are ranked from largest to smallest

$$\begin{pmatrix} \lambda_1 & ... & 0 \\ ... & \lambda_2 & ... \\ 0 & ... & \lambda_m \end{pmatrix}$$

To aid the understanding of the previous eigenvector decomposition it is helpful to consider a toy model; in this case we look at a small data set of GPA versus hours spent studying. In figure 1.a the data distribution is shown. Superimposed onto the data, in figure 1.b, are the first and only two eigenvectors(since the data is two dimensional. In the case of a 55 waveform catalog we could have up to 55 eigenvectors). The largest basis vector cuts right through the middle of the data set, while the second basis vector is orthogonal to the first, as it should be. The key is to notice that the first basis vector describes the largest component of variance in the data set, which is the spread from bottom left to top right. All PCA does is look at the data in respect to these new basis vectors, and of course we can work our way backwards to reconstruct the original data set, but how many PCA components are used is up to us.



(a)

(b)

(c)

Figure 1

Figure 2.c shows a reconstruction of the data using only the first basis vector and it is noted that only the variance with respect to the first component is retained. Applying this logic to the supernova catalogs we can ask the question, "How many basis vectors are needed to adequately describe a catalog composed of varying wave types?". As long as the variance within a group of waveforms is isolated to a minimal number of PCA components, it should be able to generalize them in this manner. In the next section we move through the task of carrying out the PCA analysis in Matlab, with a brief summary of the code necessary to carry this out.

3.2. **Matlab Code.** Much of this section is devoted to specifying the location and usage of the the PCA Matlab codes, including catalog importation and manipulation. Before PCA decomposition of a catalog can occur, all the desired waveforms must be massaged into an appropriate and similair form. The matlab codes necessary to carry this out can be found at https://alexandria.astro.cf.ac.uk/dokuwiki/doku.php?id=collaborations:lvc:supernova:start. The process goes as follows:

SCRIPT-SVD is the mains script that performs the PCA analysis. The parameters that have to be choose are:

- number of basis vectors
- sample frequency. This is the frequency you want all your waveforms to be resampled to.
- time scale. This is desired length that all waveforms will be set at.
- desired catalog(s)

An outline of script-svd:

loadsnwaveforms.m: Imports desired catalog(s) and creates cell structure where each cell element is a waveform array consisting of two columns, strain and time.

resamplewaveform.m: "Conditions" one or more input waveforms to render them more suitable for further analysis. Resamplewaveform linearly detrends, resamples to uniform time samples at a user-specified sample rate, and finally zeropads the input waveforms to the desired length.

talign.m: Uses cross-correlation to determine the time shifts that maximize the overlaps between a set of waveforms. It outputs two matrices, one of which holds the maximum match between each pair of waveforms, and the other holds the time shift that maximizes the match. This code is used to line up the varying waveforms to where their peaks are at the most common time.

svdtp.m: Computes principal components of the given waveforms.

decomposewaveform.m: computes the overlap between a collection of waveforms and a set of basis vectors. It then constructs approximants to the waveforms using the basis vectors, and computes the overlap between the original and approximants as a function of the number of basis vectors used.

This last function contains the most relevant processes regarding quality of decomposition and waveform reconstructing, as well as saving all pertinent data. The key matrices to come out of this function are,

U = [Fs x B], where B is the number of desired basis vectors and Fs is sampled frequency.

hshift = [sampled frequency x m], where m is number of imported waveforms. This is just the imported catalog(s), but time shifted and conditioned.

coefficients = [B x m]. This matrix is the match between each waveform and each basis vector, where element (i,j) is the overlap between basis vector i and waveform j. The value comes from computing the inner product between **U** and **hshift**. For effectiveness the inner product is computed in the frequency, rather than time, domain using a Fourier transform. Once the integral is computed over all relevant frequencies an inverse transform is preformed to get back to the time domain.

approximate = [sampled frequency x m]. This is the reconstructed waveform catalog using the formula,

$$\mathbf{approximate} = \mathbf{U} * \mathbf{coefficients}$$

$$approximate_{Fs,m} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{fs,1} & a_{fs,2} & \cdots & a_{fs,m} \end{pmatrix} = \begin{pmatrix} U_{1,1} & U_{1,2} & \cdots & U_{1,B} \\ U_{2,1} & U_{2,2} & \cdots & U_{2,B} \\ \vdots & \vdots & \ddots & \vdots \\ U_{fs,1} & U_{fs,2} & \cdots & U_{fs,B} \end{pmatrix} * \begin{pmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,m} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{B,1} & c_{B,2} & \cdots & c_{B,m} \end{pmatrix}$$

Note that in calculating the approximate matrix, the value B(number of basis vectors) drops out, so we are free to chose how many basis vectors we'd like in recreating our waveforms.

overlap = [B x m]. Element (i,j) is the overlap between waveform j and the approximate to waveform j using i basis vectors. Again, a sliding dot product is used to compare the reconstructed to original.

With the given code in place we can now start to answer the questions, can a waveform catalog be adequately described by a low number of PCA components, and can this be done across several catalogs of varying physical properties? First, we look at the distribution of variance, or energy content, amongst the basis vectors for individual catalogs. In figure two the variance distribution is shown for several catalogs composed of similar physical core-collapse microphysics. A list of the waveform catalogs used, as well as their filename used in the x-pipeline repository, are below.

a) BEA07: Models from Burrows et al. 2007, ApJ 665, 416.
7 waveforms; gravitational-wave emission from convective overturn and the standing-accretion-shock instability (SASI).

b) DOA09: arXiv:0910.2703v1 [astro-ph.HE]
106 waveforms; The accretion-induced collapse (AIC) of a white dwarf (WD) may lead to the formation of a protoneutron star and a collapse-driven supernova explosion.

c) DOM08: Dimmelmeier, H., Ott, C.D., Marek, A., and Janka, H.-T, Phys. Rev. D, submitted, (2008).
136 waveforms; This archive is a catalogue of the maximum density evolution of 136 supernova core collapse models.

d) DOM07: Dimmelmeier, H., Ott, C.D., Marek, A., Janka, H.-T., and Müller, E. Phys. Rev. Lett., 98, 251101, (2007).
54 waveforms.

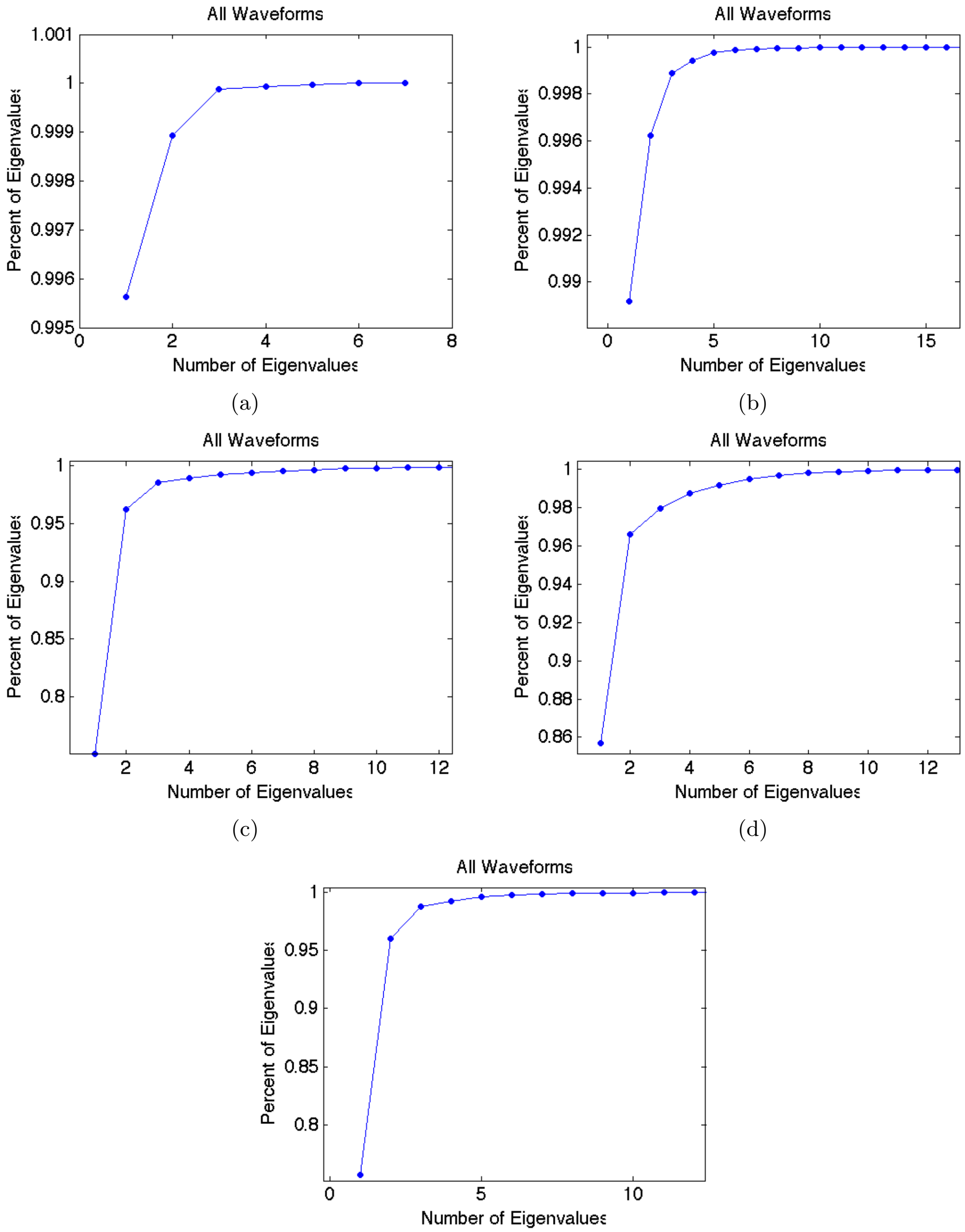
e) DOM02: Dimmelmeier, H., Font, J.A., and Muller, E. Astron. Astrophys., 388, 917-935 (2002); astro-ph/0204288.
26 waveforms.

not shown

BOM09: Jeremiah W. Murphy, Christian D. Ott, Adam Burrows. Submitted to the Astrophysical Journal, arXiv:0907.4762.
16 waveforms; We characterize the matter GW signatures of prompt convection, steady-state convection, the standing accretion shock instability (SASI) , and asymmetric explosions.

OEA09: Models from Ott et al. 2006, PRL 96, 201102.
16 waveforms; The results, which are based on axisymmetric Newtonian supernova simulations, indicate that the dominant emission process of gravitational waves in core-collapse supernovae may be the oscillations of the protoneutron star core.

OTT09: Christian D. Ott, CQG 26, 063001 (2009).
66 waveforms.



(a)

(b)

(c)

(d)

(e)

Figure 2

For the five catalogs shown the energy distribution is always within ninety-nine percent amongst the first 10 basis vectors. These results indicate common physical features between the varying gravitational waves and make it likely that a generalized, reconstructed set of waves can be computed and, in effect, make the need for using the entire catalog unecessary. There is also a common point around 10 basis vectors, at which any additional information from higher order basis vectors is minimal compared to the cost of adding more complexity to our basis set. However, another important statistic is the match levels: on a normalized scale we compare how well a reconstructed waveform matches the original as a function of basis vectors used in the recreation. This statistic will also aid us in determining if we can use PCA components in place of entire catalogs in future calculations.

In figure three the min and max match values describe the best and worst recreated waveform; ten, fifty, and ninety percent indicate the percentage of waveforms at that value. Of interest is the fact that the match levels don't always coincide with the energy disrtibutions(figure two). This indicates that a more robust statistic in the search for PCA efficiency are the match levels. However, what exactly is an acceptable match limit will have to come from how well the PCA components do within a bayesian search during X-pipeline post-processing.

Up to this point we have only performed the PCA analysis and subsequent statistical checks on single catalogs. The next step will be to consider what happens when the same analysis is performed on several catalogs appended together, where each catalog describes a different physical mechanism which dominates the core-collapse process. In all, seven catalogs comprised of 455 waveforms where used and the results are shown in figure 4. Even though a few of the catalogs from figure three have poor match statistics, this did not cause the entire 455 waveform composition to poorly, in fact it outperformed the lone catalogs.

Finally, we take a step back to look at the larger result of how well each waveform matches up to its' reconstructed version as a function of basis vectors used.

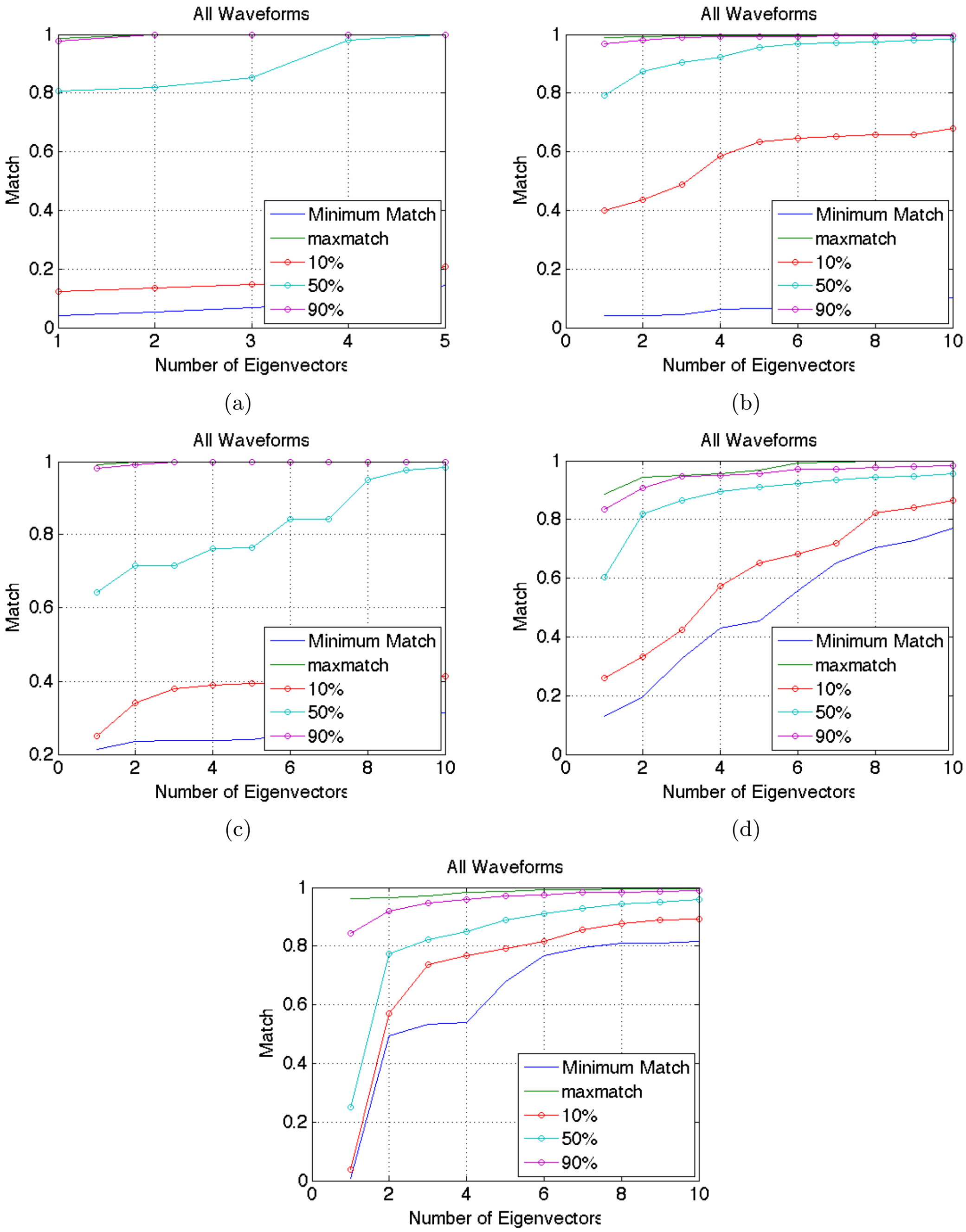
4. X-Pipeline

4.1. **Overview of a Burst Search.** X-Pipeline is an autonoanous computing tool designed to perform a gravitational wave burst analysis of LIGO/VIRGO data. All of the code is Matlab based and is ran as a coherent search, ie. data from several detectors is first combined and then analysed for candidate signals, as opposed to identifying candidates before conjoining the detector data.

Several search specific input parameters are required: desired detecors, sky position of candidate source, which coherent energies to compute, and sampling rate.

Instructions for completing both a standard supernova and modified PCA supernova search can be found at https://geco.phys.columbia.edu/xpipeline/wiki/Documentation/Searches/.

Th output of x-pipeline, for our purposes, is a detector efficiency statistic that tells us how well our gravitational wave detectors operate at a given sensitivity. To get to this statistic we must first search a data for an initial "loudest event" value, which can be



(a)

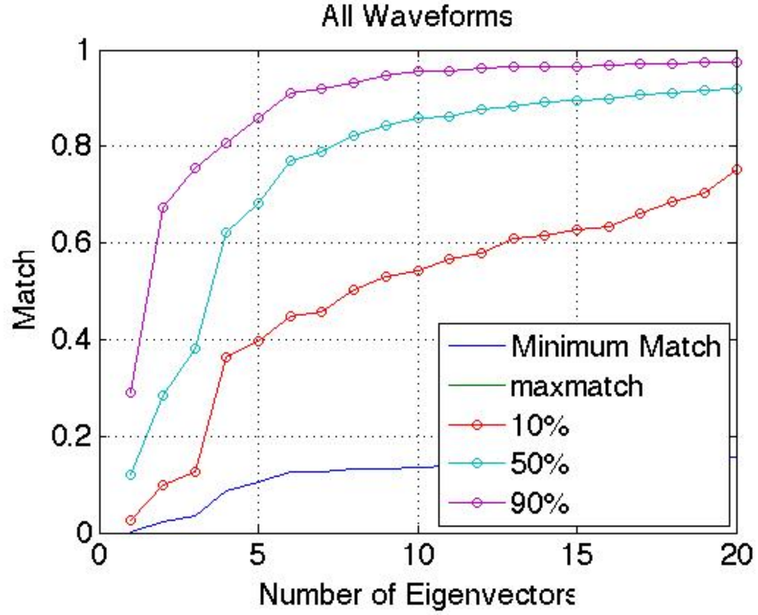
(b)

(c)

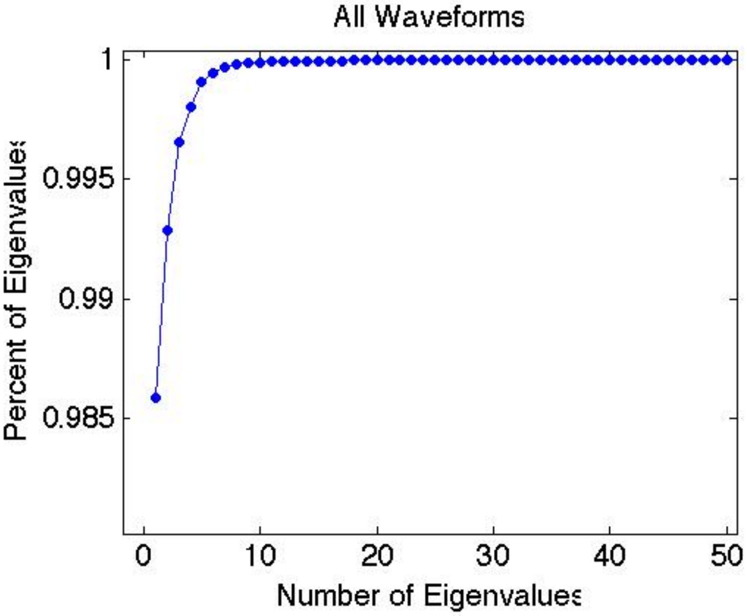
(d)

(e)

Figure 3


All Waveforms
Match
Number of Eigenvectors
0
0.2
0.4
0.6
0.8
1
0
5
10
15
20
Minimum Match
maxmatch
10%
50%
90%

(a)


All Waveforms
Percent of Eigenvalues
Number of Eigenvalues
0.985
0.99
0.995
1
0
10
20
30
40
50

(b)

Figure 4

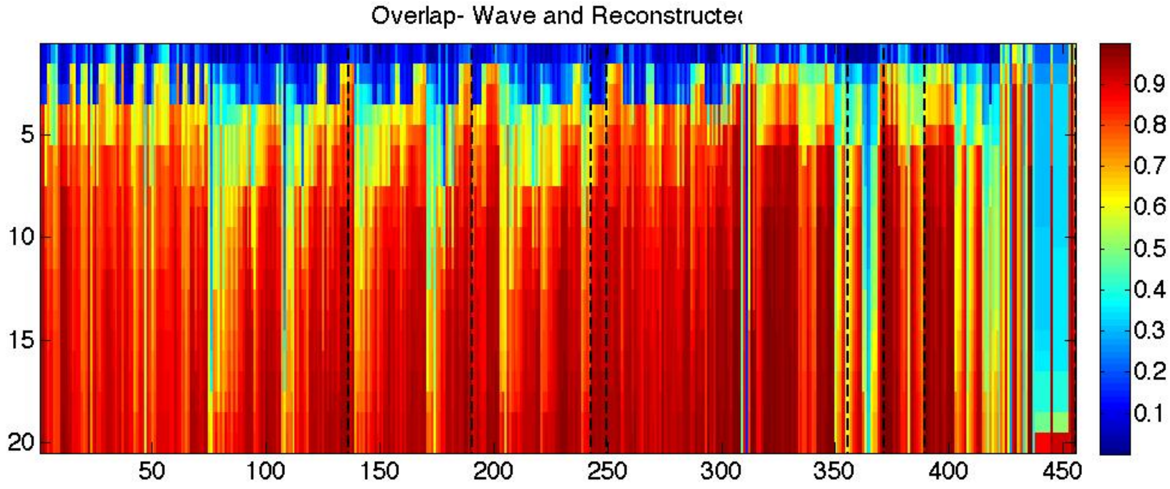
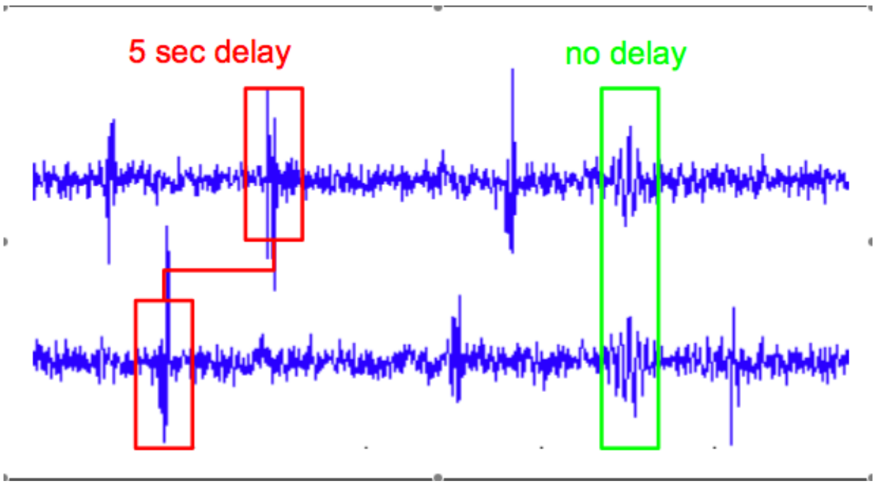


Figure 5. Here we are shown an the big picture. Black dotted lines indicate separation between catalogs

either noise or candidate, but either way, sets up a bar by which injections can be measured up to. These injections are what we believe our gravitational wave bursts to look like. They are inserted into the collected data at various amplitudes and x-pipeline then searches for them above the loudest event. All candidates are identified by the method of coherent energy computation, where by coherent I mean the data from all detectors are combined first and analysed second. In figure six it is shown how two detector streams are cross-correlated and all signals which line up, when we know they should not, are relegated to the null stream energy category.

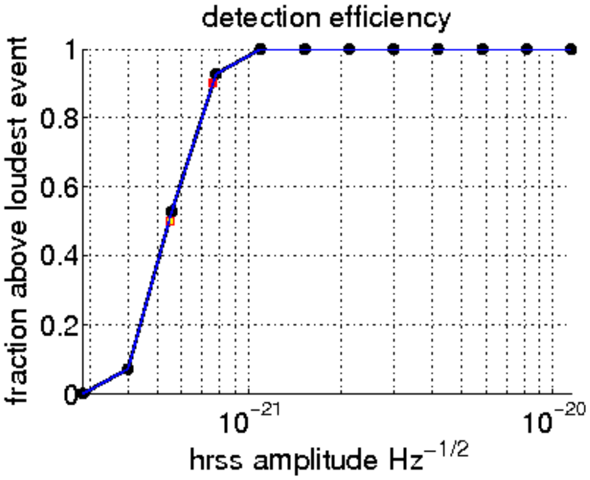
Figure seven displays the final output of detector efficiency for the standard, non-bayesian case. The ultimate goal of this project is to then run the injections over again, but this time search for their presence using our PCA components within a bayesian search framework.

slide.png



(a)

Figure 6



(a)

Figure 7

References