

LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY  
- LIGO -  
CALIFORNIA INSTITUTE OF TECHNOLOGY  
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

|                               |                          |                  |
|-------------------------------|--------------------------|------------------|
| <b>Document Type</b>          | <b>LIGO-T08xxxx-00-Z</b> | 20 February 2008 |
| <b>Coherent Event Display</b> |                          |                  |
| R. A. Mercer and S. Klimenko  |                          |                  |

*Distribution of this draft:*  
LIGO Scientific Collaboration  
**DRAFT**

**California Institute of Technology**  
**LIGO Project - MS 51-33**  
**Pasadena CA 91125**  
Phone (626) 395-2129  
Fax (626) 304-9834  
E-mail: info@ligo.caltech.edu

**Massachusetts Institute of Technology**  
**LIGO Project - MS NW22-295**  
**Cambridge, MA 01239**  
Phone (617) 253-4824  
Fax (617) 253-7014  
E-mail: info@ligo.mit.edu

WWW: <http://www.ligo.caltech.edu/>

# Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                                   | <b>3</b>  |
| <b>2</b> | <b>Usage</b>  | <b>4</b>  |
| 2.1      | Creating the Grid Proxy . . . . .                     | 4         |
| 2.2      | Starting the Coherent Event Display . . . . .         | 5         |
| <b>3</b> | <b>Implementation of the Coherent Event Display</b>   | <b>6</b>  |
| 3.1      | Setting the Environment . . . . .                     | 7         |
| 3.2      | Starting the Coherent Event Display . . . . .         | 7         |
| 3.2.1    | Parsing the Event Configuration File . . . . .        | 7         |
| 3.2.2    | Parsing the Site Configuration File . . . . .         | 7         |
| 3.2.3    | Locating Appropriate Frame Data . . . . .             | 8         |
| 3.3      | Starting the ROOT Session . . . . .                   | 8         |
| 3.3.1    | Differences from Coherent WaveBurst . . . . .         | 9         |
| 3.3.2    | Identifying the Event . . . . .                       | 9         |
| 3.3.3    | Plot Production . . . . .                             | 9         |
| 3.3.4    | LIGO.LW XML Creation . . . . .                        | 10        |
| 3.4      | Website Creation . . . . .                            | 10        |
| <b>4</b> | <b>Description of Coherent Event Display Contents</b> | <b>11</b> |
| 4.1      | Job Parameters . . . . .                              | 11        |
| 4.2      | Event Parameters . . . . .                            | 11        |
| 4.3      | Network Data Matrix . . . . .                         | 13        |
| 4.4      | Time-Frequency Maps . . . . .                         | 13        |
| 4.5      | Likelihood Time-Frequency Maps . . . . .              | 14        |
| 4.6      | Reconstructed Detector Responses . . . . .            | 16        |
| 4.7      | Skymaps . . . . .                                     | 17        |
| <b>5</b> | <b>Running under Condor</b>                           | <b>24</b> |
| 5.1      | Grid Proxy . . . . .                                  | 24        |
| 5.2      | Creating the List of Events . . . . .                 | 25        |
| 5.3      | Generating the Submit File . . . . .                  | 25        |
| <b>6</b> | <b>Installing the Coherent Event Display</b>          | <b>26</b> |
| 6.1      | Installation . . . . .                                | 26        |
| 6.2      | Configuration . . . . .                               | 26        |
| 6.2.1    | Setting Locations . . . . .                           | 26        |
| 6.2.2    | Site Configuration File . . . . .                     | 26        |
|          | <b>References</b>                                     | <b>27</b> |

# 1 Introduction

As a worldwide network of gravitational-wave detectors is now operating with an unprecedented sensitivity it is becoming increasingly important to be able to easily visualise gravitational-wave event candidates from various search pipelines using these detector networks. The Coherent Event Display (CED) has been developed with the goal of providing a simple and easy to use tool for performing follow up analyses of burst gravitational-wave event candidates. The CED produces a web page detailing reconstructed parameters, time-frequency maps, reconstructed detector responses, likelihood time-frequency maps, and reconstructed parameter skymaps. The CED supports events from all 2, 3, 4, and 5 detector network combinations of GEO, LIGO, and Virgo.

This document covers the current stable release of the CED, version 0.4.8, released on 20th February 2008. In Section 2 we provide detailed usage instructions for running single instances of the CED on the Caltech Cluster, in Section 3 the implementation and structure of the CED is discussed, Section 4 discusses the different sections that make up the CED, and Section 5 discusses how multiple instances of the CED can be run simultaneously using the Condor workload management system. Finally, in Section 6 we discuss the requirements and installation procedure for the CED.

## 2 Usage

The Coherent Event Display (CED) has been designed to be as easy, and straightforward, to use as possible. It is simply a matter of creating a simple configuration file describing the event under investigation, referred to as the Event Configuration File, and then running a single script to generate the CED web page. This section discusses how to create the Event Configuration File and run the CED. For information regarding running multiple instances of the CED under the Condor workload management system see Section 5.

The Event Configuration File contains all the parameters needed to uniquely identify the event and therefore extract all relevant parameters required for the CED. The parameters that need to be specified are the GPS start and stop times of the event and the applied time shift for each detector in the network. The format of this configuration file follows that of a standard Python Configuration File <sup>1</sup> consisting of sections, led by a “[section]” header and followed by “name = value” entries. Lines beginning with “#” or “;” are treated as comments and are ignored.

The Event Configuration File contains up to two sections, [event\_params] which specifies the parameters of the requested event and an optional section [job] which specifies the science segment within which the event occurs; the contents of these sections are now discussed.

**[event\_params]** Here the parameters used to uniquely identify the event are specified. The event is specified by using the GPS start, stop times and the lag times of the event in each detector of the network using the following parameters: g1\_start, g1\_stop, g1\_lag for GEO, h1\_start, h1\_stop, h1\_lag for H1, h2\_start, h2\_stop, h2\_lag for H2, l1\_start, l1\_stop, l1\_lag for L1, and v1\_start, v1\_stop and v1\_lag for Virgo. The detectors are added to the network in alphabetical order unless the order is specified using the order parameter.

**[job]** Here the GPS start and stop times of the science segment within which the event specified in [event\_params] section exists. science\_start specifies the start time and science\_stop the stop time.

An example Event Configuration file for the 3 detector LIGO only network, i.e. L1, H1, and H2, can be seen below:

```
; s5_event.ini - coherent event display
;               - configuration file example, s5 event

[event_params]
order = L1,H1,H2
l1_start = 817216890.806200
l1_stop = 817216891.475200
l1_lag = 203.125000
h1_start = 817216687.681200
h1_stop = 817216688.350200
h1_lag = 0.000000
h2_start = 817216687.681200
h2_stop = 817216688.350200
h2_lag = 0.000000

[job]
science_start = 817216613
science_stop = 817216975
```

### 2.1 Creating the Grid Proxy

Once the Event Configuration File has been created the next step is to ensure that the data encompassing this event can be located. The CED determines the location of the required data by making queries to the LSCsegFindServer, for

<sup>1</sup><http://docs.python.org/lib/module-ConfigParser.html>

ensuring the requested event occurs within a valid science segment, and the LDRdataFindServer, for determining the location of the frame files containing the required data. Before queries to these servers can be made, it needs to be checked that there is a valid Grid Proxy available for authentication. The validity of the grid proxy can be checked by using the `grid-proxy-info` command. If there is a valid proxy available the the output from the above command will be of the form:

```
$ grid-proxy-info
subject   : /DC=org/DC=doegrids/OU=People/CN=User/CN=12345678
issuer    : /DC=org/DC=doegrids/OU=People/CN=User
identity  : /DC=org/DC=doegrids/OU=People/CN=User
type      : Proxy draft (pre-RFC) compliant impersonation proxy
strength  : 512 bits
path      : /tmp/x509up_u123
timeleft  : 11:59:58
```

However if there is not a valid proxy available the output will be:

```
$ grid-proxy-info
ERROR: Couldn't find a valid proxy.
Use -debug for further information.
```

If this is the case then a grid proxy will need to be created, this is done using the `grid-proxy-init` command, the output of which is of the form:

```
$ grid-proxy-init
Your identity: /DC=org/DC=doegrids/OU=People/CN=User
Enter GRID pass phrase for this identity:
Creating proxy ..... Done
Your proxy is valid until: Tue May 15 23:42:10 2007
```

## 2.2 Starting the Coherent Event Display

Once it has been determined that a valid Grid Proxy is available the CED can be started. The CED is currently installed on the Caltech cluster and is compiled as a 64 bit binary, therefore it will only successfully run on the `ldas-grid` and `ldas-pcdev1` head nodes, as well as the nodes. The latest stable release can be started by using the following command:

```
$ ~ram/ced/stable/event_display.sh config.ini website_dir
```

where `config.ini` is the Event Configuration File and `website_dir` specifies the location to which the Coherent Event Display web page will be saved to. An example session running the CED is shown below:

```
$ ~ram/ced/stable/event_display.sh event.ini ~/public_html/ced
Querying LSCsegFindServer for valid Science Segment...
Querying LDRdataFindServer for frame lists...
Creating required input files...
Running ROOT session...
Generating XML...
Generating Coherent Event Display...
```

Upon successful completion, the CED creates a web page detailing the specified event and saves it to the requested location: in the above example to `~/public_html/ced`.

### 3 Implementation of the Coherent Event Display

The Coherent Event Display (CED) is built on top of the Coherent WaveBurst (cWB) algorithms [1, 2] as a series of ROOT, Python and Bash Scripts. A flowchart showing the structure of the CED can be seen in Figure 1, rectangles represent components that are implemented using the ROOT framework and the boxes with rounded corners represent components that are implemented using Python. A description of each of these components follows.

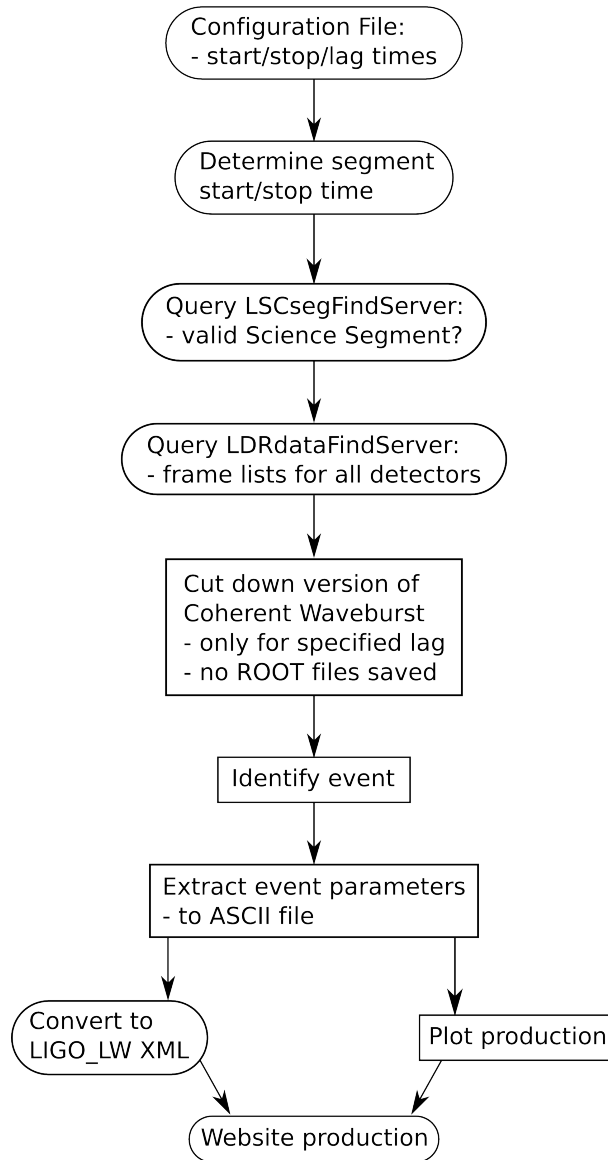


Figure 1: A flowchart showing the structure of the Coherent Event Display, rectangles represent components that are implemented using the ROOT framework and the boxes with rounded corners represent components that are implemented using Python.

## 3.1 Setting the Environment

The CED is started by using the environment setup script, `event_display.sh`. This script ensures that the system environment is correctly setup prior to starting the main CED script, `ced.py`. It ensures that the ROOT framework, the LIGO/Virgo Frame Library, and the Wavelet Analysis Tool (WAT) are correctly setup within the system environment and then executes the main CED script with the appropriate options.

## 3.2 Starting the Coherent Event Display

Now that the environment is correctly setup, the main CED script can be started. The first task that this script must perform is to obtain the relevant parameters describing the requested event, specified within the event configuration file, and any site specific configuration parameters, specified within the site configuration file.

### 3.2.1 Parsing the Event Configuration File

The event configuration file follows the format of a standard Python configuration file that can be parsed using the `ConfigParser` module. It contains up to two sections, `[event_params]` that specifies the parameters of the requested event and an optional section `[jobs]` that specifies the science segment in which the event occurs. A further discussion of the Event Configuration File can be found in Section 2.

### 3.2.2 Parsing the Site Configuration File

As well as the parameters describing there are other parameters that need to be specified, these are dependant upon the computer system on which the CED is executed. It specifies the required parameters for querying the `LSCsegFindServer` and `LDRdataFindServer` in order to locate the appropriate frame data, see Section 3.2.3 for further information.

The Site Configuration File also follows the format of a standard Python configuration file and contains several sections. `[segfind]`, specifies the required parameters for querying a `[segments]` specifies a series of segment lists that can be queried instead of querying the `LSCsegFindServer`, `[datafind]` specifies the parameters for querying a `LDRdataFindServer`, `[framecache]` specifies parameters for parsing a QScan framecache for frame data information, and `[general]` which specifies several different general site parameters. Finally the `[s4_frame]` and `[s5_frame]` sections specifies the frame type to use for S4 and S5 events respectively.

**[segfind]** Here two parameters need to be specified, `server` the address of the `LSCsegFindServer` to query, and `skip_segfind` a boolean that determines whether or not to skip the query to the `LSCsegFindServer`.

**[segments]** Here segment lists can be specified which are used in preference to querying the `LSCsegFindServer`, currently only S5 events for L1, H1, and H2 support this using the `s5-l1h1h2` parameter.

**[datafind]** Here two parameters need to be specified, `server` the address of the `LDRdataFindServer` to query, and `match` a regular expression used for matching against found frames.

**[framecache]** Here two parameters need to be specified, `cache` the location of the QScan frame cache to query, and `match` a regular expression used for matching against found frames.

**[general]** Here any general site parameters are specified, these include `resample_buffer` which specifies the size of the resample buffer to add to the data segment and `filter_dir` which specifies the location of the required filters.

**[s4-frame]** Here the frame type, to use when the event in question is from the S4 Science Run, is specified for each detector, i.e. `g1`, `h1`, `h2`, `l1`, and `v1`.

**[s5-frame]** Here the frame type, to use when the event in question is from the S5 Science Run, is specified for each detector, i.e. `g1`, `h1`, `h2`, `l1`, and `v1`.

An example site configuration file can be seen below:

```

; site.ini - coherent event display
;           - site configuration example

[segfind]
server = ldas-cit.ligo.caltech.edu
skip_segfind = False

[segments]
s5_h1h2l1 = /archive/home/ram/ced/segments/S5_H1H2L1_segments.txt

[datafind]
server = ldas-cit.ligo.caltech.edu
match = localhost/data

[framecache]
cache = /archive/home/qonline/q/framecaches/S5.txt
match = data

[s4_frame]
g1 = G1_RDS_C02_L3
h1 = H1_RDS_C04_LX
h2 = H2_RDS_C04_LX
l1 = L1_RDS_C04_LX
v1 = V1_RDS_C04_LX

[s5_frame]
g1 = G1_RDS_C02_L3
h1 = H1_RDS_C02_LX
h2 = H2_RDS_C02_LX
l1 = L1_RDS_C02_LX
v1 = V1_RDS_C02_LX

[general]
resample_buffer = 64
filter_dir = /usr1/igor/waveburst/data64

```

### 3.2.3 Locating Appropriate Frame Data

In order to start the main CED ROOT script, the location of the appropriate frame data needs to be known. This is determined via a three step process, the first of which is to check that the specified times exists within a valid Science segment by making a query to a `LSCsegFindServer`. The second step is to determine where the required data is located, this is done by making a query to `LDRdataFindServer` for each detector within the network. Finally, from this list of frames the required input files are constructed ready for starting the ROOT portion of the CED.

## 3.3 Starting the ROOT Session

Now that the appropriate input files have been created execution of the ROOT script can start. The ROOT script is used for determining the full set of parameters for the event and for the creation of various plots used in the CED.



### 3.3.1 Differences from Coherent WaveBurst

There are several ways in which the CED differs from cWB, the main of which is related to the way that time shifts are handled. For cWB, if the requested time shift would cause the new time to exist outside of the segment under analysis the extra time is looped round to the start of the segment. A diagram illustrating this can be seen below in Figure 2.

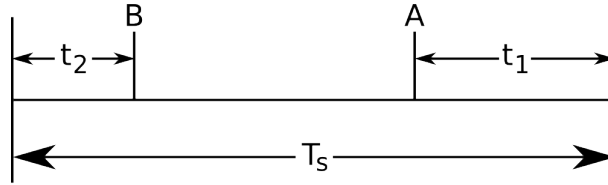


Figure 2: A diagram illustrating how time shifts are applied within Coherent WaveBurst. Consider an event at point  $A$  within the segment of length  $T_s$ , if a time shift of  $t$  is applied and is greater than the different in time between the end of the segment and point  $A$ , i.e.  $t > t_1$ , then the remaining time  $t_2$  will loop round to the beginning of the segment. Therefore after a time shift of  $t = t_1 + t_2$ , the event at  $A$  will have been shifted to a time  $B$ .

This method of applying time shifts is therefore dependant upon the length of the analysis segment that is used. For the case of the CED, the length of the analysis segment is not known *a priori* and as such handling time shifts in this way could result in events not being identified correctly. This is the reason why the time lag, at which the event occurred, is one of the required parameters.

These time lags are then used to apply static time shifts to the data so that a standard “zero-lag” analysis can be performed, therefore bypassing the time shift algorithm within cWB.

The other main difference from cWB is that the full parameter set for all detected event are not saved in the CED, only those of interest.

### 3.3.2 Identifying the Event

Once the main analysis loop of the ROOT script has finished all the events that have been found are stored within a `netevent` object. The ROOT script then “walks” through this object checking if the central time of each detected event, for each detector in the network, is within the range of GPS times specified within the Event Configuration File. When a match has been found the `lag_index`<sup>2</sup> and `cluster_index` are recorded and a counter recording the number of matching events is incremented.

Currently the CED only supports the generation of a web page when a single event is found within the specified GPS time ranges, if multiple events are found then the CED will exit with an error. In a future version it is planned that if multiple events are found then multiple web pages will be created.

Once the specified event has been identified all appropriate parameters, for the specified event only, are exported and saved in a ASCII file that is universally accessible and therefore simple to read by processes other than ROOT; such as the web page generation script, which is written in Python.

### 3.3.3 Plot Production

Now that the specified event has been identified and the associated parameters saved, the script now turns its attention to the generation of various plots. In order for these plots to be produced it is necessary to ensure that the data set has been set to the appropriate resolution for the specified event. This is done by determining the wavelet layer of the event from the `rate` parameter and transforming accordingly. Plots of the reconstructed waveforms, likelihood skymaps and likelihood time frequency maps are then generated.

<sup>2</sup>Due to the way that time shifts are handled, by applying a static shift, the `lag_index` is always zero, but it still needs to be specified due to the structure of the `netevent` object.

### **3.3.4 LIGO\_LW XML Creation**

Once execution of the ROOT script has concluded all the required parameters and plots associated with the requested event are available, execution of the main CED script `ced.py` now continues.

The first task that `ced.py` performs after resuming execution is to construct a LIGO\_LW XML file containing the parameters of the requested event. LIGO\_LW XML is used as it is a self describing format and a widely used standard within the LSC, therefore storing triggers in this format allows easy comparison with triggers from other searches.

### **3.4 Website Creation**

Now that the LIGO\_LW XML file has been generated `ced.py` has all the information it requires to generate the CED web page, it has the parameters from the LIGO\_LW XML file and the figures created during the execution of the ROOT script. The web page is then constructed and execution of the CED concludes.

## 4 Description of Coherent Event Display Contents

The Coherent Event Display (CED) is split up into sections containing information regarding different aspects of the requested event. In this Section the contents of each of these sections is described.

### 4.1 Job Parameters

The first section contains parameters specific to the CED job, this includes the GPS and UTC start time of the data segment that was used for analysis. An example Job Parameters table can be seen below in Figure 1.

|                             |                                 |
|-----------------------------|---------------------------------|
| <b>GPS Segment Time (s)</b> | <b>817216613</b>                |
| <b>UTC Segment Time</b>     | <b>Nov 28 2005 12:36:40 UTC</b> |

Table 1: An example Job Parameters table showing the GPS and UTC start times of the analysis segment used for the Coherent Event Display job.

### 4.2 Event Parameters

The Event Parameters table contains all parameters associated with the requested event. An example Event Parameters table can be seen below in Table 2. This example is from the CED running on a 100 Hz Sine Gaussian injection with a  $h_{rss} \sim 1^{-21}$ , as can be seen from the table the CED reconstructs these parameters fairly accurately.

- The first parameter in the Event Parameter Table, “Time Lag”, details what (if any) time shift has been applied to the data for the given detector; further details on how time shifts are performed is discussed in Section 3.3.1.
- The next series of parameters are related to the time at which the event occurs: “GPS Start Time” records the GPS start time of the event in each detector, “GPS End Time” records the GPS stop time of the event, “GPS Central Time” records the central time, as weighted by likelihood, of the event, and “Central Time” records the central time of the event with respect to the start of the data segment analysed. This central time recorded here because the plots that follow are plotted with respect to the start of the analysis data segment, therefore including this time here makes these plots easier to interpret. The final network parameter that is displayed is “Event Duration”, which records the duration of the detected event, this is simply the event start time subtracted from the Event stop time; it is included here for completeness.
- The next parameter that is displayed in the event parameters table is  $h_{rss}$  which represents the reconstructed gravitational wave strain from the event in question.
- The next two parameters displayed are “SNR” and “Rank SNR”, these parameters record the SNR and the Rank SNR of the detected event respectively.
- The next series of parameters that are displayed are different forms of the statistical significance of the detected event. In total there are three different significance parameters, “Rank Significance”, “Gaussian Significance”, and “Geometric Significance”.
- The next parameter to be displayed is “Likelihood”, this is the main coherent statistic that the cWB Algorithms, and therefore the CED, uses to detect events. Likelihood is defined in Equation 1:

$$\mathcal{L} = \sum_{ij} \sum_{k=1}^K \frac{1}{\sigma_k^2} \left[ w_k^2 [i, j] - (w_k [i, j] - \xi_k [i, j])^2 \right] \quad (1)$$

| IFO                               | L1            | H1            | H2            |
|-----------------------------------|---------------|---------------|---------------|
| Time Lag (s)                      | 203.125       | 0.000         | 0.000         |
| GPS Start Time (s)                | 817216890.906 | 817216687.781 | 817216687.781 |
| GPS Stop Time (s)                 | 817216891.375 | 817216688.250 | 817216688.250 |
| GPS Central Time (s)              | 817216891.273 | 817216688.157 | 817216688.157 |
| Central Time (s)                  | 278.273       | 75.157        | 75.157        |
| Event Duration (s)                | 0.468750      |               |               |
| hrss                              | 9.69e-21      | 5.23e-22      | 5.23e-22      |
| Null                              | 7.29e+01      | 6.89e+01      | 9.66e+01      |
| Noise                             | 5.01e-23      | 3.65e-23      | 7.57e-23      |
| SNR                               | 3.06e+04      | 2.49e+02      | 9.25e+01      |
| Rank SNR                          | 7.06e+02      | 1.13e+02      | 5.10e+01      |
| Rank Significance                 | 3.01e+02      | 1.73e+01      | 3.27e+00      |
| Gaussian Significance             | 1.43e+04      | 4.04e+01      | 2.90e+00      |
| Geometric Significance            | 3.25e+00      |               |               |
| Likelihood                        | 3.07e+04      |               |               |
| Phi (degrees)                     | 336.50        |               |               |
| Theta (degrees)                   | 90.50         |               |               |
| Size                              | 73            |               |               |
| flow (Hz)                         | 64.00         |               |               |
| fhigh (Hz)                        | 320.00        |               |               |
| Central Frequency (Hz)            | 105.70        |               |               |
| Bandwidth (Hz)                    | 256.00        |               |               |
| Resolution (Hz)                   | 128           |               |               |
| F+                                | 3.73e-01      | 8.20e-02      | 8.20e-02      |
| Fx                                | 1.31e-01      | -2.15e-01     | -2.15e-01     |
| Normalisation Factor              | 1.00e+00      |               |               |
| Pearson's Correlation Coefficient | 4.30e-01      |               |               |
| Network Correlation Coefficient   | 6.01e-01      |               |               |
| Correlated Energy                 | 3.59e+02      |               |               |
| Effective Correlated Energy       | 1.86e+02      |               |               |
| Energy Disbalance                 | 1.32e+01      | 2.74e+01      | -4.07e+01     |

Table 2: An example Event Parameters Table from the Coherent Event Display run on a Sine Gaussian injection. The injection was performed at a frequency of 100 Hz with a  $h_{rss} \sim 1^{-21}$ , as can be seen from the table the Coherent Event Display reconstructs these parameters fairly accurately.

where  $w_k [i, j]$  is the sampled detector data for the  $k$ th detector, at time index  $i$  and frequency index  $j$ ; and  $\xi_k [i, j]$  is the detector response for the  $k$ th detector, at time index  $i$  and frequency index  $j$ , defined by Equation 2:

$$\xi_k [i, j] = F_+ [i, j] h_+ [i, j] + F_\times [i, j] h_\times [i, j] \quad (2)$$

where  $F_+ [i, j]$  and  $F_\times [i, j]$  are the antenna patterns for the + and  $\times$  polarisations respectively.

- The next parameter that is reported is ‘‘Correlation’’, in order to understand what this parameter represents, the overall detected energy needs to be considered. The detected energy is given below in Equation 3:

$$2\mathcal{L} = \sum_{m,n} C_{mn} \langle w_m w_n \rangle \quad (3)$$

where  $m$  and  $n$  index the multiple detectors that make up the network,  $C_{mn}$  are the correlation coefficients, and  $w_{m,n}$  is the detector output for the given detector. This detected energy can be split up into two separate quantities, incoherent and coherent.

$$\sum_{m,n} C_{mn} \langle w_m w_n \rangle = E_{m=n} + E_{m \neq n} \quad (4)$$

where the diagonal terms correspond to the incoherent energy, and off diagonal terms the coherent energy. The network correlation is then defined as:

$$C_{\text{net}} = \frac{E_{m \neq n}}{\text{Null} + E_{m \neq n}} \quad (5)$$

The network correlation is used for the post-production selection of triggers, as a gravitational wave event should have a high value of network correlation whereas a accidental coincidence will have a lower value.

- The next displayed parameters are “Phi” and “Theta”, these record the reconstructed sky position for the detected event, as weighted by the likelihood. The coordinate system is the Earth fixed frame with  $\phi = 0^\circ$  corresponding to the Greenwich Meridian and  $\theta = 90^\circ$  corresponding to the equator.
- The next network parameter is “Size”, this records the number of Time-Frequency pixels in the appropriate Time-Frequency decomposition layer that make up the detected event.
- The next series of parameters are related to the frequency of the detected event: “flow” records the low frequency limit of the detected event, “fhigh” records the high frequency, ”Central Frequency” records the central frequency as weight by likelihood, “Bandwidth” records the total bandwidth of the detected event, and “Resolution” records the wavelet decomposition frequency of the detected event.

### 4.3 Network Data Matrix

The next section of the CED display contains the Network Data Matrix, this represents the total detected energy of the event in question. The diagonal components of the matrix make up the incoherent energy and the off-diagonal components make up the coherent energy. An example Network Data Matrix can be seen below in Table 3, this was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

|          |          |          |
|----------|----------|----------|
| 3.04e+04 | 3.27e+02 | 3.21e+01 |
| 3.27e+02 | 2.97e+00 | 4.43e-01 |
| 3.21e+01 | 4.43e-01 | 1.18e-01 |

Table 3: An example Network Data Matrix from the Coherent Event Display run on a 100 Hz Sine Gaussian injection, with  $h_{rss} \sim 1^{-21}$ .

### 4.4 Time-Frequency Maps

The next section of the CED displays Time-Frequency Maps, these show the wavelet coefficients normalised by the noise RMS as a function of time and frequency for the event in question. Below are two different examples of Time-Frequency Maps for the 4 km LIGO Livingston interferometer, created after running the CED on the same 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ . The first plot, Figure 3, is plotted using the default “Pixel” plotting style.

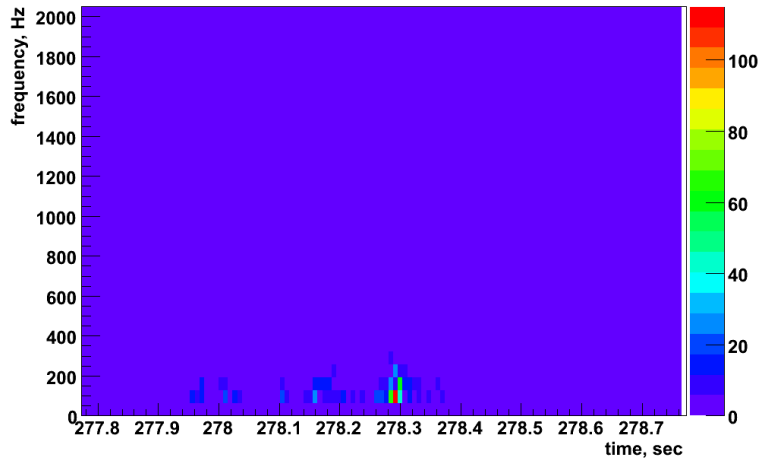


Figure 3: An example Time-Frequency Map from the Coherent Event Display run on a 100 Hz Sine Gaussian injection, with  $h_{rss} \sim 1^{-21}$ , on the 4 km LIGO Livingston Interferometer using the “Pixel” plotting style.

The second example, Figure 4, is plotted using the “Shaded” plotting style, this plotting style interpolates between adjacent data points to give a smoother plot.

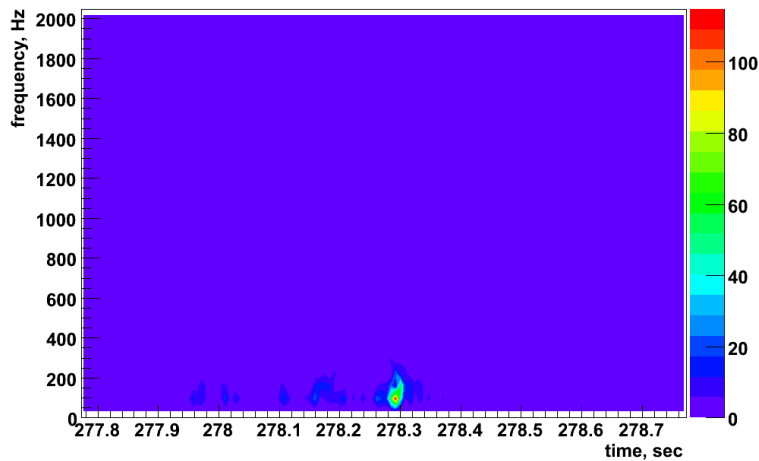


Figure 4: An example Time-Frequency Map from the Coherent Event Display run on a 100 Hz Sine Gaussian injection, with  $h_{rss} \sim 1^{-21}$ , on the 4 km LIGO Livingston Interferometer using the “Shaded” plotting style.

## 4.5 Likelihood Time-Frequency Maps

The next section of the CED contains the Likelihood Time-Frequency Maps, these plots show the Likelihood, defined in Equation 1, as a function of time and frequency for the entire detector network for the event in question. Below are three different examples of Likelihood Time-Frequency Maps, created after running the CED on the same 100 Hz

Sine Gaussian injection with  $h_{r_{SS}} \sim 1^{-21}$ . The first plot, Figure 5, is plotted using the default “Pixel” plotting style:

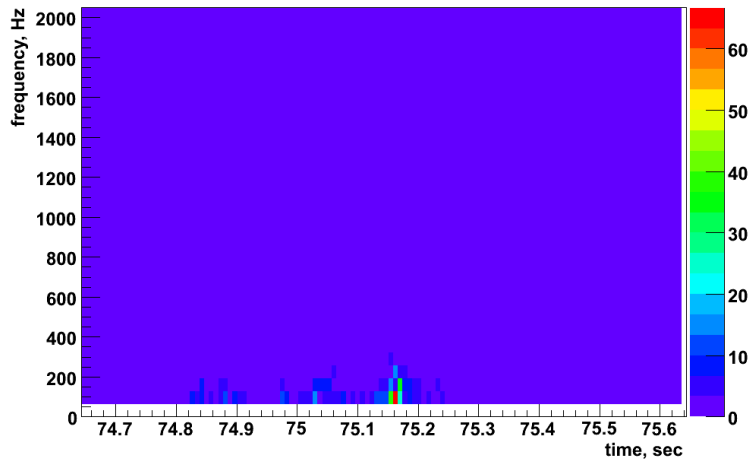


Figure 5: An example Likelihood Time-Frequency Map from the Coherent Event Display run on a 100 Hz Sine Gaussian injection, with  $h_{r_{SS}} \sim 1^{-21}$ , using the “Pixel” plotting style.

The second example, Figure 6, is plotted using the “Shaded” plotting style, this plotting style interpolates between adjacent data points to give a smoother plot:

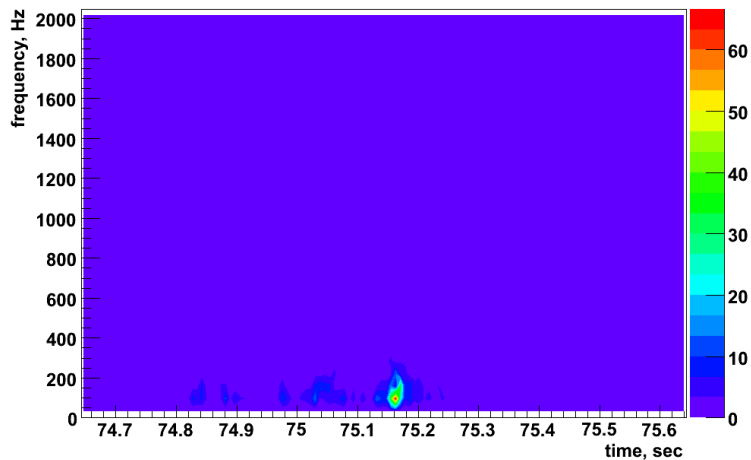


Figure 6: An example Likelihood Time-Frequency Map from the Coherent Event Display run on a 100 Hz Sine Gaussian injection, with  $h_{r_{SS}} \sim 1^{-21}$ , using the “Shaded” plotting style.

The final example, Figure 7, is plotted using the “Cluster” plotting style, which reconstructs only the event cluster requested.

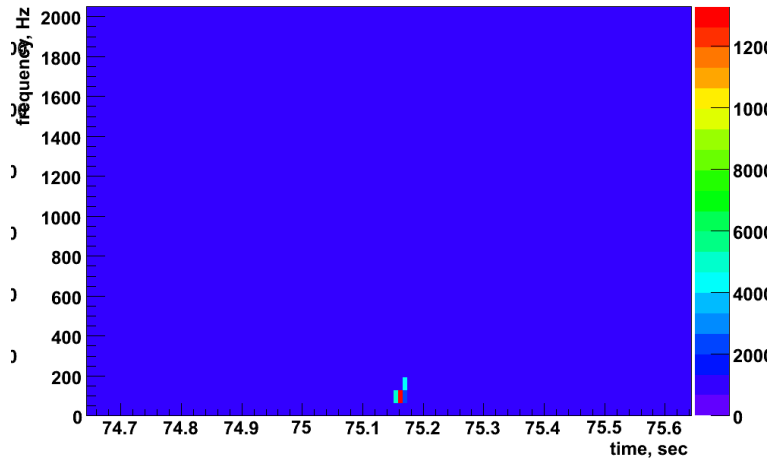


Figure 7: An example Likelihood Time-Frequency Map from the Coherent Event Display run on a 100 Hz Sine Gaussian injection, with  $h_{rss} \sim 1^{-21}$ , using the “Cluster” plotting style.

#### 4.6 Reconstructed Detector Responses

The next section of the CED display plots showing the Reconstructed Detector Responses. There are three types of plot, the first of which shows the reconstructed strain for the detected event. An example reconstructed strain plot can be seen below in Figure 8, this plot was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

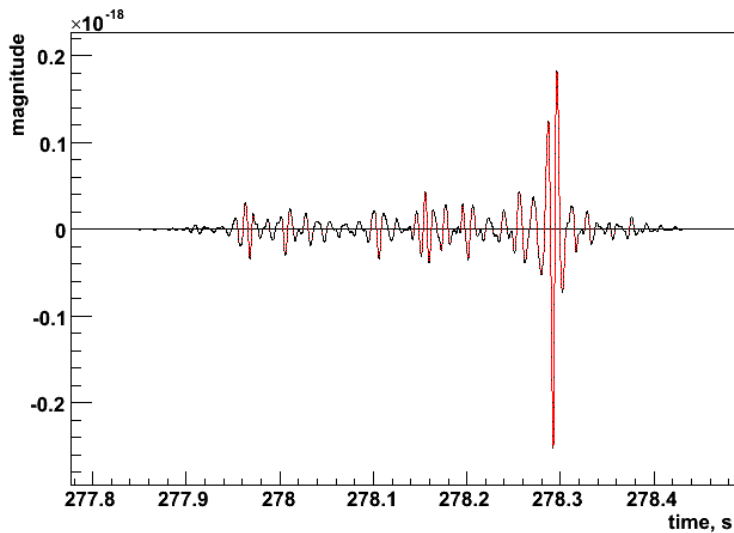


Figure 8: An example reconstructed strain signal plot from the Coherent Event Display run on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$  for the 4 km LIGO Livingston interferometer.

The next two plots in the Reconstructed Detector Response section contain two different reconstructed waveforms,



the reconstructed strain and the band limited detector output. The first of these two plots, Figure 9, shows the detector response and the strain reconstructed using the reconstructed signal, this plot was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

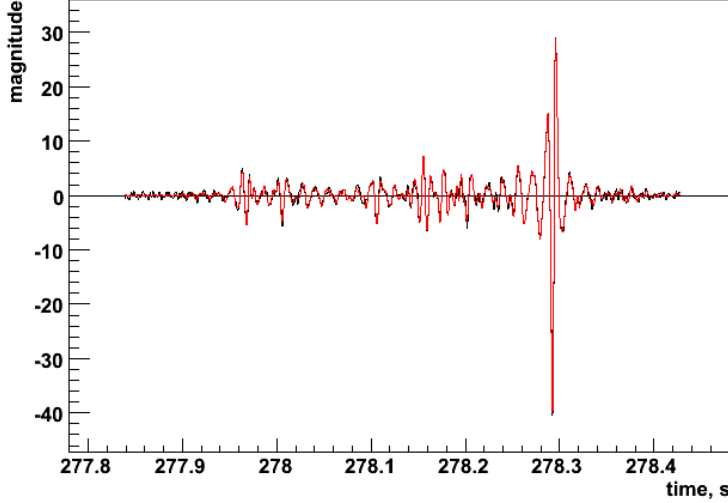


Figure 9: An example reconstructed detector response plot from the Coherent Event Display run on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$  for the 4 km LIGO Livingston interferometer. This plot shows the detector response and the strain reconstructed using the reconstructed signal.

The final reconstructed detector response plot, Figure 10, again shows the detector response and the strain but this time the signal and noise signals are used for the reconstruction, this plot was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

## 4.7 Skymaps

The CED displays several different skymaps, the first two of which shows the sensitivity of the detector network to the plus and cross polarisations as a function of sky position. In order to understand what these skymaps represent, it is necessary to introduce the total detected SNR which is given by Equation 6:

$$\text{SNR}_{\text{total}} = 2g (\langle h_1^2 \rangle + \epsilon \langle h_2^2 \rangle) \quad (6)$$

where  $h_1$  and  $h_2$  are the two different gravitational wave polarisations. The parameter  $\epsilon$  is the Network Alignment which measures the alignment of the network to the two different polarisations of gravitational waves. Low values of the alignment factor indicate that the detector network is only sensitive to one of the gravitational wave polarisations. The Network Alignment is defined below, in Equation 7:

$$\epsilon = \frac{g_r - |g_c|}{g_r + |g_c|} \quad (7)$$

where  $g_r$  and  $g_c$  can be considered to be the overall network antenna patterns, given by:

$$g_r = \sum_{k=1}^K \frac{F_{+k}^2 + F_{\times k}^2}{4\sigma_k^2} \quad (8)$$

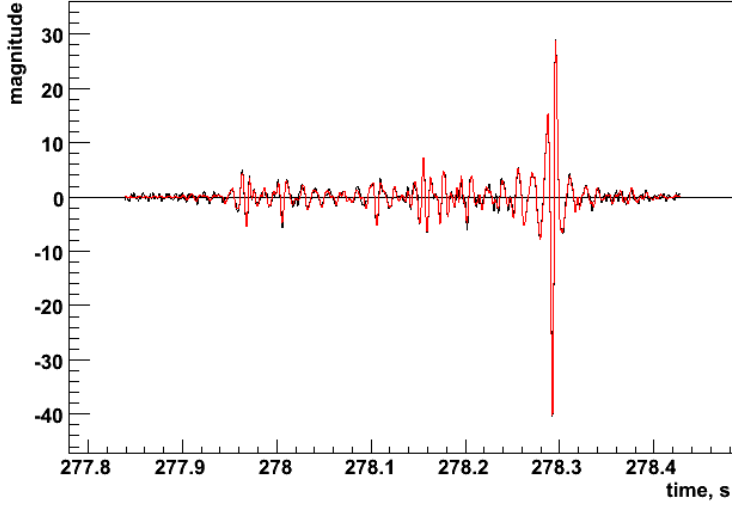


Figure 10: An example reconstructed detector response plot from the Coherent Event Display run on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$  for the 4 km LIGO Livingston interferometer. This plot shows the detector response and the strain reconstructed using the reconstructed signal and noise.

$$g_c = \sum_{k=1}^K \frac{[F_{+k} + F_{\times k}]^2}{4\sigma_k^2} \quad (9)$$

The factor  $g_r + |g_c|$  measures the sensitivity to the plus polarisation and the factor  $g_r - |g_c|$  measures the sensitivity to the cross polarisation. Examples of these skymaps can be seen below in Figures 11 & 12, these were created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

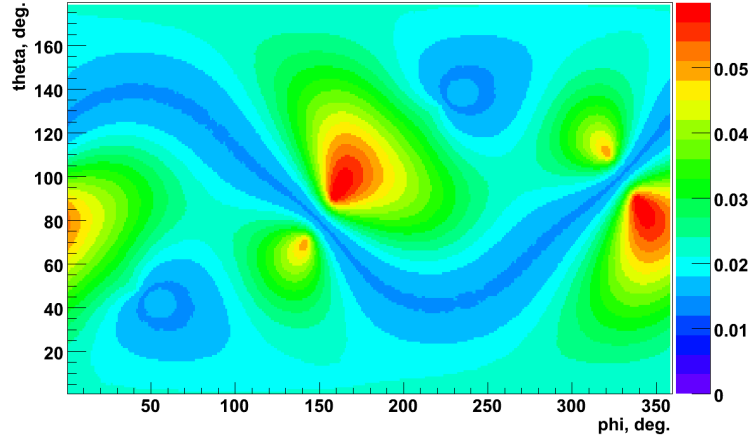


Figure 11: An example Network Alignment Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

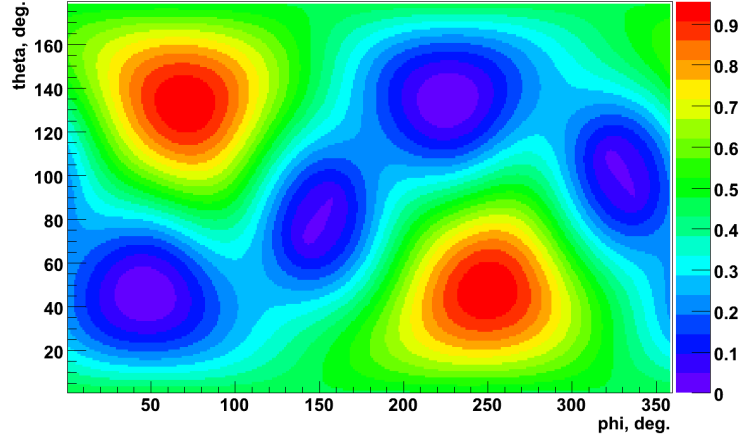


Figure 12: An example Network Sensitivity Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{r_{SS}} \sim 1^{-21}$ .

The next skymap shows the penalty factor as a function of sky position; in order to understand what this skymap represents it is necessary to introduce one of the constraints used in the analysis, the signal-noise orthogonality requirement, given below in Equation 10:

$$\Lambda_k = \sum_{ij} (w_k [i, j] s_k [i, j] - s_k^2 [i, j]) = 0 \quad (10)$$

where  $w_k [i, j]$  is the sampled detector data for the  $k$ th detector, at time  $i$ , and frequency  $j$ ; and  $s_k [i, j]$  is the whitened detector response for the  $k$ th detector, at time  $i$ , and frequency  $j$ . The reason for this constraint is to prevent the reconstruction of unphysical detector responses when the energy of the response is greater than the total energy in the detector data, i.e.:

$$E_k > S_k \quad (11)$$

where  $E_k$  is the total energy in the detector data, given by Equation 12; and  $S_k$  is the energy of the reconstructed detector response given by Equation 13.

$$E_k = \sum_{ij} w_k^2 [i, j] \quad (12)$$

$$S_k = \sum_{ij} s_k^2 [i, j] \quad (13)$$

This constraint, Equation 10, is applied during the likelihood variation procedure of the cWB pipeline in the form of the penalty factor, by penalising sky positions where this constraint is not satisfied, i.e.:

$$P_k = \begin{cases} \sqrt{\frac{E_k}{S_k}} & : E_k < S_k \\ 1 & : E_k > S_k \end{cases} \quad (14)$$

where  $P_k$  is the penalty factor for the  $k$ th detector, therefore the overall penalty factor  $P_f$  is given by the maximum value of  $P_k$ :

$$P_f = \max_k P_k \quad (15)$$

It is this quantity  $P_f$  that is used to generate the Penalty Factor skymap, an example of which can be seen below in Figure 13, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

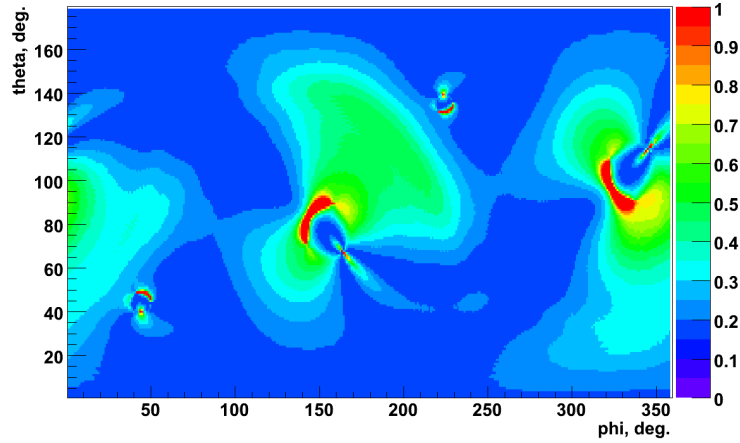


Figure 13: An example Penalty Factor Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

The next skymap shows the maximum value of the network Correlation, Equation 5, as a function of sky position. An example Network Correlation skymap can be seen below in Figure 14, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

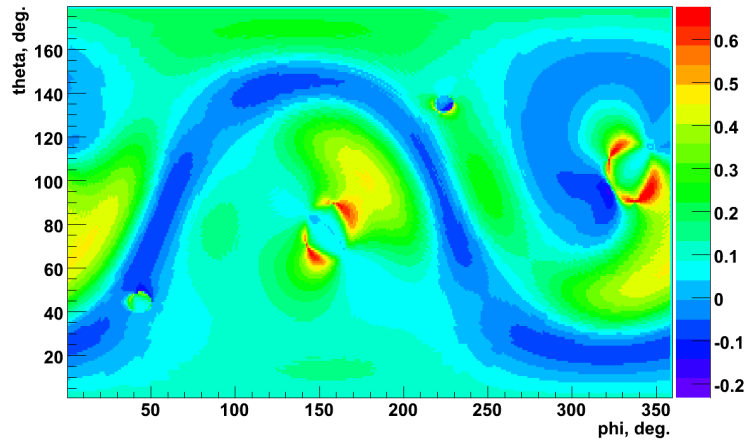


Figure 14: An example Network Correlation Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

The next skymap shows the maximum value of the Likelihood, Equation 1, as a function of sky position. An example Likelihood skymap can be seen below in Figure 15, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

One problem with the Likelihood skymap is that it does not take in account the detector response, therefore regions with high Likelihood could correspond to sky locations to which the detector network has a limited sensitivity. As it is

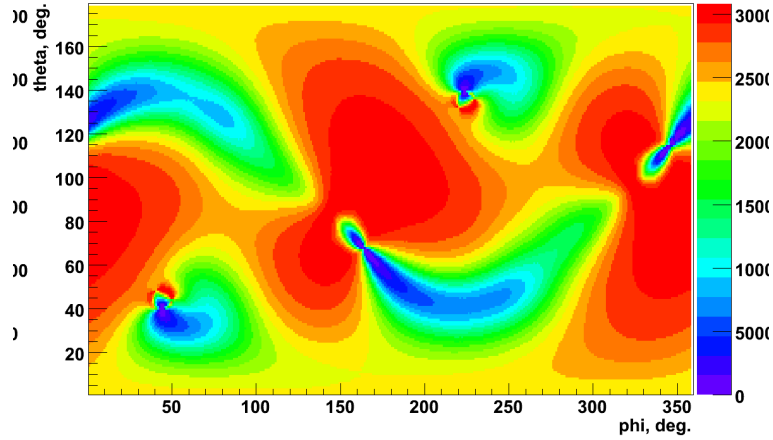


Figure 15: An example Likelihood Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

unlikely that a real gravitational wave signal will be detected from these regions of limited sensitivity the Likelihood does not provide an accurate statistic on which to claim detection. For this purpose the Sky Statistic is introduced, this can be considered to be the Likelihood weighted with the detector response and is given by:

$$\text{Sky Stat} = \mathcal{L} P_f C_{\text{net}} \quad (16)$$

where  $\mathcal{L}$  is the Likelihood,  $P_f$  the penalty factor (defined in Equation 15), and  $C_{\text{net}}$  the network correlation coefficient (defined in Equation 5). An example Sky Statistic skymap can be seen below in Figure 16, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

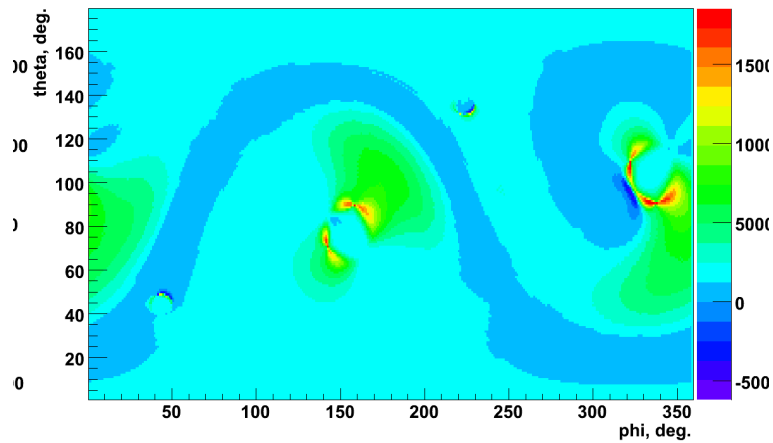


Figure 16: An example Sky Statistic Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

As shown in Equations 3 and 4 the total energy can be considered to be made of two components,  $E_{m=n}$  the

incoherent energy, and  $E_{m \neq n}$  the coherent energy. The next skymap shows the coherent energy  $E_{m \neq n}$ , sometimes referred to as the correlated energy, as a function of sky position. An example Correlated Energy skymap can be seen below in Figure 17, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

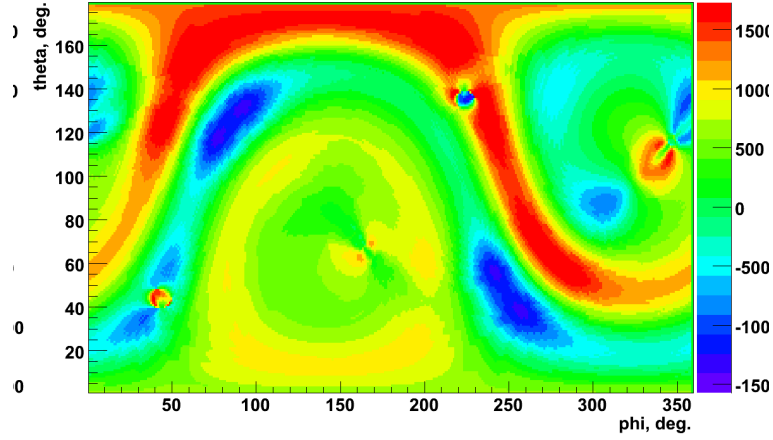


Figure 17: An example Correlated Energy Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

Another way that the Likelihood, Equation 1, can be written is in term of the normalised and reconstructed energies,  $E$  and  $N$  respectively:

$$\mathcal{L} = E - N \quad (17)$$

where  $E$  the total normalised energy in the detector output streams is given by Equation 18, and  $N$  the total reconstructed energy of the noise is given by Equation 19:

$$E = \sum_{ij} |\mathbf{w}[i, j]|^2 \quad (18)$$

$$N = \sum_{ij} |\mathbf{w}[i, j] - \mathbf{s}[i, j]|^2 \quad (19)$$

where the vector  $\mathbf{w}[i, j]$  is defined by:

$$\mathbf{w}[i, j] = \left\{ \frac{w_1[i, j]}{\sigma_1[i, j]}, \dots, \frac{w_K[i, j]}{\sigma_K[i, j]} \right\} \quad (20)$$

and the vector  $\mathbf{s}[i, j]$  is defined by:

$$\mathbf{s}[i, j] = \left\{ \frac{\xi_1[i, j]}{\sigma_1[i, j]}, \dots, \frac{\xi_K[i, j]}{\sigma_K[i, j]} \right\} \quad (21)$$

The quantity  $N$  is called the null stream and is plotted, as a function of sky position, as the Null Energy skymap; an example of which can be seen below in Figure 18, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

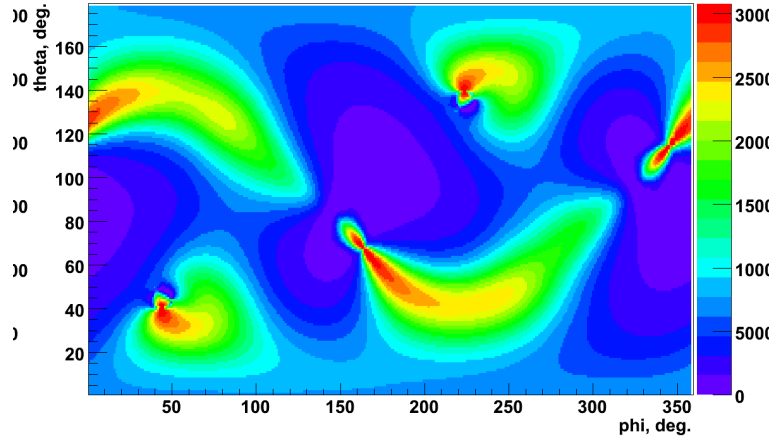


Figure 18: An example Null Energy Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

As discussed earlier, one of the constraints that is applied during the analysis is the signal-noise orthogonality requirement given by Equation 10. This constraint is applied to the likelihood by means of the penalty factor, Equation 15, not by rigidly enforcing the constraint. The Energy Disbalance skymap shows the sum, over detectors, of  $\Lambda_k$  as a function of sky position. An example Energy Disbalance skymap can be seen below in Figure 19, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

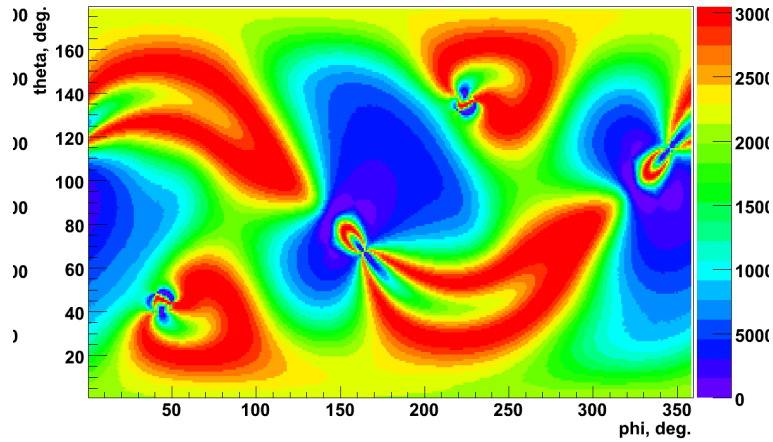


Figure 19: An example Energy Disbalance Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

The final skymap shows the normalisation factor, this is an internal check to ensure that the likelihood calculation is being performed correctly. If any region of this is not equal to unity then that indicates that there is a problem with the chosen parameters. An example Normalisation Factor skymap can be seen below in Figure 20, this skymap was created by running the CED on a 100 Hz Sine Gaussian injection with  $h_{rss} \sim 1^{-21}$ .

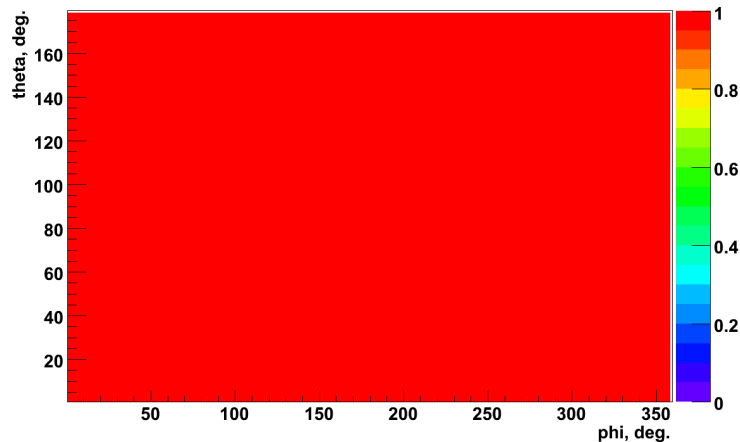


Figure 20: An example Normalisation Factor Skymap created after running the Coherent Event Display on a 100 Hz Sine Gaussian injection with  $h_{r_{SS}} \sim 1^{-21}$ .

## 5 Running under Condor

Multiple instances of the Coherent Event Display (CED) can be run under the Condor Workload Management System so speed under creation of event pages. A script, `ced_condor.py`, has been written which creates a submit file for submitting a list of CED jobs to a Condor cluster.

### 5.1 Grid Proxy

As the CED uses the `LSCsegFindServer` and the `LDRdataFindServer`, which uses Grid Certificate Authentication, a valid Grid Proxy must be available in order to the CED to correct run. When running on a node under Condor the location of the appropriate Grid Proxy must be passed on the node to allow the CED to query the `LSCsegFindServer` and `LDRdataFindServer`.

Therefore a Grid Proxy needs to be created on the submit machines and its location noted. The Proxy can be created using the following:

```
$ grid-proxy-init
```

and entering the appropriate password when requested. The location of the Grid Proxy can be determined using the following:

```
$ grid-proxy-info
subject   : /DC=org/DC=doegrids/OU=People/CN=User/CN=12345678
issuer    : /DC=org/DC=doegrids/OU=People/CN=User
identity  : /DC=org/DC=doegrids/OU=People/CN=User
type      : Proxy draft (pre-RFC) compliant impersonation proxy
strength  : 512 bits
path      : /tmp/x509up_u123
timeleft  : 11:59:58
```

This shows that the Grid Proxy is located at `/tmp/x509up_u123`, which is not visible to the nodes. Therefore this Grid Proxy needs to be copied to a location that is visible from the nodes - a sub-directory within your home directory for example, such as `~$HOME/MyGrid`.



```
$ mkdir ~/MyGrid
$ cp /tmp/x509up_u123 ~/MyGrid
$
```

## 5.2 Creating the List of Events

The script for generating a submit file needs a list of event configuration files for the jobs that are to be run under condor. This list need to contain the full path to the event configuration files, for example:

```
/archive/home/ram/s5run47/1.ini
/archive/home/ram/s5run47/2.ini
/archive/home/ram/s5run47/3.ini
/archive/home/ram/s5run47/4.ini
/archive/home/ram/s5run47/5.ini
...
```

## 5.3 Generating the Submit File

The next step is to generate the condor submit file, this is done using the `ced_condor.py` script as follows:

```
$ ~/ram/ced/stable/scripts/ced_condor.py --events=s5run47.lst \
--output-path=/archive/home/ram/public_html/ced/s5run47 \
--log-path=/usr1/ram \
--proxy=/archive/home/ram/MyGrid/x509up_u123 > s5run47.sub
```

updating the above path appropriately. This submit file can then be submitted to Condor as usual.

## 6 Installing the Coherent Event Display

As the Coherent Event Display (CED) uses the Coherent WaveBurst (cWB) algorithms, as such the dependencies of the CED are very similar to those of cWB. So essentially any system that can run cWB can also run the CED. The CED is built up as a series of ROOT, Python, and Bash scripts, therefore to be able to run the CED the system needs a working installation of the ROOT framework<sup>3</sup> and a Python interpreter<sup>4</sup>. The CED also requires an installation of LIGOTools<sup>5</sup>, in order to convert the times of the events from GPS to UTC.

### 6.1 Installation

The CED is simple to install, it is just a simple matter of downloading the latest release from the CED web page, at [http://tier2.ihepa.ufl.edu/~ram/private/event\\_display](http://tier2.ihepa.ufl.edu/~ram/private/event_display), and untarring the contents to the required location. Once this has been done, the CED needs to be configured; the following section discusses this.

### 6.2 Configuration

Now that the CED has been installed it needs to be correctly configured before it can be used. There are several files that need to be edited so that the CED knows where it is installed and what data is available.

#### 6.2.1 Setting Locations

The first location that needs to be specified for the CED to function correctly is the location in which it is installed. This is specified in the files `event_display.sh` and `setup.sh` using the `ED_LOCATION` environment variable. The second location that needs to be specified is the location that the ROOT framework is installed in. This is specified in the files `event_display.sh` and `setup.sh` using the `ROOT_LOCATION` environment variable.

#### 6.2.2 Site Configuration File

The main configuration process that needs to be performed before the CED can be used is to create the Site Configuration File. This file, named `site.ini`, specifies site specific parameters such as the addresses of the appropriate `LSCsegFindServer` and `LDRdataFindServer` servers to query so that the CED can locate data. The Site Configuration File, as discussed in Section 3.2.2, contains six sections containing different parameters associated with different aspects of the CED.

The current CED release, version 0.4.8, includes two example Site Configuration Files that are used for running the CED on the Caltech Cluster, `site.CIT.ini` and on the Livingston Cluster, `site.LLO.ini`.

---

<sup>3</sup><http://root.cern.ch>

<sup>4</sup><http://www.python.org>

<sup>5</sup><http://www.ldas-sw.ligo.caltech.edu/ligotools>

## References

- [1] S. Klimenko, S. Mohanty, M. Rakhmanov, and G. Mitselmakher. Constrains likelihood analysis for a network of gravitational wave detectors. *Physical Review D*, 72:122002, 2005.
- [2] S. Klimenko, I. Yakushin, R. A. Mercer, and G. Mitselmakher. Coherent WaveBurst - WAT version 4.7.0, 2008. LIGO-T080011-00-Z.