A prototype system for gravitational wave data analysis

by

W.J. Watkins

A thesis submitted to the
University of Wales
for the degree of
Doctor of Philosophy

September, 1991
This thesis is dedicated to my Mother and the memory of my Father.
DECLARATION

I declare that this work has not already been accepted in any substance for any degree, and is not being currently submitted in candidature for any degree.

W. J. Watkins
Candidate

Except where otherwise stated, this work is wholly the result of the candidate’s own investigation. Suitable credit is given to joint work with colleagues, and to work of others throughout the thesis.

W. J. Watkins
Candidate

Supervisor
ACKNOWLEDGEMENTS

I would like to thank my supervisor, Professor B.F. Schutz, for his constant encouragement, guidance and patience.

I should also like to thank the Gravitational Physics group at the University of Glasgow, headed by Professor J. Hough, who supplied the data analysed in this thesis.

Abstract

This thesis is concerned with a prototype gravitational wave data analysis system. Work will shortly be underway on the construction of large scale broad-band laser interferometers that should produce data containing gravitational waves. Subsequently, systems that can analyse this data, extracting any waves within the noise, need to be developed now. The prototype system described in this thesis is the first fully automated gravitational wave data analysis system to be developed. It is designed in parallel, to run on several processors simultaneously, as future systems are certain to employ parallelism to some degree. This thesis is also concerned with the analysis of actual data by this system, produced by the prototype interferometer owned and operated by the University of Glasgow.

The software is described in detail, beginning with a description of the parallelism within the system. The procedure employed to split the multiplexed data into its constituent streams is discussed followed by a description of the methods employed to optimise the detector noise for the extraction of possible signals from short duration gravitational wave bursts. This is followed by a description of both the statistical analysis and the search for events carried out on the calibrated and optimally weighted interferometer stream. This search looks for wideband burst events in the time series and also coalescing binary signals found by cross-correlating the noise with a bank of templates.

An account of the results of the analysis is given, concentrating on the statistics of detector noise, identifying its main characteristics both when the detector is operating properly and also when the housekeeping data suggests it is not. A description is given of a parameter, easily calculated from the noise, that is shown to be a good diagnostic indicator of the state of the detector. The distribution of events found by the analysis is discussed, and it is shown how the number of events vary with the state of the detector.
Contents

1 INTRODUCTION.
   1.0.1 Gravitation .................................................. 9
   1.0.2 Gravitational Waves ......................................... 9
   1.0.3 Detectors ...................................................... 13
   1.0.4 Data analysis ................................................ 16
   1.0.5 Contents of the thesis ...................................... 17

2 SOURCES OF GRAVITATIONAL WAVES ................................. 19
   2.1 Introduction ..................................................... 19
   2.2 BURSTS .......................................................... 20
      2.2.1 Gravitational collapse. .................................. 20
      2.2.2 Coalescing Binaries ...................................... 21
   2.3 CONTINUOUS WAVE SIGNALS .................................... 25
      2.3.1 Periodic and other Sources ............................... 25
      2.3.2 Stochastic Background .................................... 26
      2.3.3 Low Frequency Sources .................................. 26

3 MATHEMATICAL TECHNIQUES .......................................... 27
   3.1 Introduction ..................................................... 27
   3.2 Fourier analysis ............................................... 27
      3.2.1 The Fourier transform ................................... 27
      3.2.2 Correlation ................................................ 28
      3.2.3 The Discrete Fourier Transform (DFT) .................. 28
      3.2.4 The Fast Fourier Transform (FFT) ....................... 30
   3.3 Extraction of a signal from a noisy background ............... 32
      3.3.1 Noise ...................................................... 32
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.2</td>
<td>Matched Filtering</td>
<td>34</td>
</tr>
<tr>
<td>4</td>
<td>The GLASGOW PROTOTYPE DETECTOR</td>
<td>38</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>38</td>
</tr>
<tr>
<td>4.1.1</td>
<td>The 10m Prototype.</td>
<td>39</td>
</tr>
<tr>
<td>4.1.2</td>
<td>Data taken by the Glasgow prototype</td>
<td>41</td>
</tr>
<tr>
<td>4.1.3</td>
<td>The Garching prototype detector.</td>
<td>43</td>
</tr>
<tr>
<td>4.1.4</td>
<td>The 100 hour run.</td>
<td>45</td>
</tr>
<tr>
<td>5</td>
<td>DESCRIPTION OF SOFTWARE</td>
<td>46</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>46</td>
</tr>
<tr>
<td>5.1.1</td>
<td>The Transputer</td>
<td>47</td>
</tr>
<tr>
<td>5.1.2</td>
<td>Overview</td>
<td>50</td>
</tr>
<tr>
<td>5.2</td>
<td>Communication within the network.</td>
<td>53</td>
</tr>
<tr>
<td>5.3</td>
<td>ROOT.F77</td>
<td>57</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Program details.</td>
<td>58</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Software</td>
<td>58</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Main body of program.</td>
<td>62</td>
</tr>
<tr>
<td>5.4</td>
<td>SECD1.F77</td>
<td>64</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Introduction</td>
<td>64</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Main body of task</td>
<td>65</td>
</tr>
<tr>
<td>5.4.3</td>
<td>Calibration</td>
<td>66</td>
</tr>
<tr>
<td>5.4.4</td>
<td>Weighting</td>
<td>67</td>
</tr>
<tr>
<td>5.4.5</td>
<td>SECD2</td>
<td>69</td>
</tr>
<tr>
<td>5.4.6</td>
<td>CORR1 and CORR2</td>
<td>72</td>
</tr>
<tr>
<td>6</td>
<td>RESULTS.</td>
<td>79</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>79</td>
</tr>
<tr>
<td>6.2</td>
<td>Details of results obtained from tapes B6 and C4</td>
<td>79</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Introduction</td>
<td>79</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Internal data</td>
<td>80</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Gaussian considerations</td>
<td>81</td>
</tr>
<tr>
<td>6.2.4</td>
<td>External housekeeping</td>
<td>83</td>
</tr>
<tr>
<td>6.2.5</td>
<td>Events</td>
<td>84</td>
</tr>
<tr>
<td>6.2.6</td>
<td>Sensitivity of the detector</td>
<td>85</td>
</tr>
</tbody>
</table>
CONTENTS

7 SUMMARY AND CONCLUSIONS

BIBLIOGRAPHY

APPENDIX 1: TESTING THE CROSS-CORRELATION ROUTINES
   A1: Introduction
   A2: Case 1: signal directly proportional to filter
   A3: Case 2: general chirp in noise

APPENDIX 2: THE STRUCTURE OF A RESULTS BLOCK

APPENDIX 3: PROGRAM LISTING
Chapter 1

INTRODUCTION.

1.0.1 Gravitation

It is necessary to know only that a body has mass to also know that it exerts a gravitational force on every other body possessed of mass in the universe. This was first realised by Isaac Newton who developed the concept of gravitational attraction from the experiments of Galileo and Kepler’s 3 kinematic laws of planetary motion. He arrived at the following expression for the force of attraction due to gravity, \( F_{12} \), between two masses, \( M_1 \) and \( M_2 \) separated by a distance \( d \),

\[
F_{12} = \frac{GM_1 M_2}{d^2},
\]

(1.1)

where \( G \) is the Gravitational constant which has the value \( 6.673 \times 10^{-11} \text{Nm}^2/\text{kg}^2 \). This equation appeared in Newton’s *Principia* (1687), which effectively defined man’s understanding of gravitation for well over two centuries. It remains to this day accurate enough for almost all practical purposes on Earth as well as in the solar system.

Thinking changed at the start of this century when Albert Einstein formulated the theories of Special and General Relativity (1905,1915). Einstein utilised the mathematics of curved spaces constructed by Riemann (1854), to produce a mathematical model of space and time. He represented gravity in tensor form not as a force or a field but as a spacetime geometry, and derived the field equations that described spacetime in the region around gravitating bodies be they moving or stationary.

1.0.2 Gravitational Waves

History

Gravitational wave research started in 1916, when Einstein began to investigate whether solutions to his field equations predicted the propagation of energy in the form of waves in spacetime, analogous to
CHAPTER 1. INTRODUCTION.

solutions of Maxwell's equations predicting electromagnetic waves. Einstein found a wave equation that suggested that they did. In the following years, Einstein himself (1918), Weyl (1922) and Eddington (1924) further developed this work, until the linearised theory for gravitational waves from non self-gravitating systems was thought to be fully understood.

The first major work on emission from non-linear, self-gravitating systems was produced by Landau and Lifshitz (1941). Following this physicists such as Bondi (1957,1960), Penrose (1963) and Isaacson (1968) continued the work on the subject steadily until the early 70's since when, motivated by the considerable activity surrounding detector research, an ever increasing number of theorists have striven to develop models for sources of gravitational radiation and the associated waveforms that detectors may discover. Consequently, many potential sources have been identified and analysis techniques developed to extract their characteristic wave forms from detector noise.

The nature of gravitational waves

Gravitational waves are a consequence of any relativistic theory of gravity. They are ripples in the existing curvature of spacetime that propagate with the speed of light. Within the most widely accepted of these theories,i.e the General Theory of Relativity, they are described by $h^{\alpha \beta}$, the weak field deviation of the spacetime metric from flat spacetime. $h^{\alpha \beta}$ is given by the solutions of Einstein's field equations in a vacuum. These are all contained within,

$$\left(-\frac{\partial^2}{\partial t^2} + \nabla^2\right) \bar{h}^{\alpha \beta} = 0. \quad (1.2)$$

The solutions to the above are well documented in many texts such as Schutz (1985) and Misner, Thorne and Wheeler (1973) so they are not included here.

Gravitational waves are produced only by non-symmetric motions. They are emitted in all directions, to varying degrees dependent upon the motions of the system producing them. Gravitational waves influence matter in directions perpendicular to their direction of propagation, altering the separation of particles according to the waves two distinct linear polarisation states, represented by $h_+\text{ and } h_\times$. These, in contrast to the polarisation states for electromagnetic waves which are separated by 90°, are rotated by only 45° relative to each other. The differing physical effects of these states on a ring of free particles is shown in Figure 1.1. The top row shows the deformation of the ring for various phases for the wave cycle of the $h_+$ polarisation of a gravitational wave. The bottom row is the equivalent representation for the $h_\times$ polarisation. Notice that the axes of deformation are at 45° to each other.
CHAPTER 1. INTRODUCTION.

Figure 1.1: Elastic ring deformed by both polarisation states separately. The strain in space is \( \delta l/l = 1/2h \). Diagram taken from Kawamura et al (1989).

Sources of gravitational waves

Any two particles moving non-symmetrically in some way relative to each other should emit gravitational waves. For these waves to be of sufficient magnitude to be even potentially detectable on Earth however, necessitates very rapid non-symmetrical motions in very massive bodies.

These sources could produce waves with a number of fundamental characteristics. The most important in terms of this thesis is the short duration burst signal, whether it is spread over a wide frequency range, e.g., stellar core collapse, or whether it is in some way frequency specific, confined to some small frequency range, e.g., coalescing binary system where the components are neutron stars and/or black holes. Observation of one such system, the binary pulsar PSR 1913+16 discovered by Hulse and Taylor in 1974, has already provided indirect evidence of the existence of gravitational waves. Its observed rate of energy loss due to orbital decay is correctly predicted in terms of energy loss due to emission of gravitational waves, and in fact it will coalesce in a time less than the Hubble time (Weisberg and Taylor (1984), Taylor (1987)).

The waves could also be continuous, and produce steady signals over periods of time long enough for the signal to vary significantly because of the rotation of the Earth. This effect is known as frequency and amplitude modulation. The best known prospective candidates for the production of
continuous signals are spinning neutron stars or pulsars. A third detectable source could be the
\textit{stochastic} background of gravitational radiation whose various contributors could include cosmic
strings and events in the early universe. This thesis will not be concerned with the detection of such
wave. For continuous waves, see the thesis of Livas(1987).

\textbf{Motivation behind gravitational wave research.}

The development and construction of a gravitational wave detector is an extremely large undertaking,
requiring much money and manpower. In order to justify such a commitment, a strong scientific
case had to be put, listing the benefits of the detection of gravitational waves. These benefits are
many, both astrophysical and technical.

The most immediate consequence would be the potential detection of the waves themselves. Gravitational
waves are the last major prediction of Einstein’s theory yet to be independently confirmed
by experimentation. Their detection could further strengthen General Relativity in relation to other
gravitational wave predicting theories of gravity, if the information received from the wave, in terms
of the amplitude, direction and polarisation are related in a way consistent with General Relativity.
Of course if the arrival of gravitational waves could be associated with an optical event, such as the
brightening of a supernova, then the delay between the two observations could be used to confirm,
or otherwise, that gravitational waves also travel at the speed of light.

The real benefits to astrophysics, however would come later with the vast new window that
gravitational waves would open on the universe. As welcome as radio and x-ray astronomy were as
additions to optical astronomy, their information content is limited in that they are all specifically
produced by individual ions, atoms and molecules in various states of excitement, frequently in
sparse, low velocity regions, whose emissions can be easily blocked by matter between Earth and
the source. Gravitational wave astronomy will be radically different, their sources specifically being
dense, massive and necessarily subject to high velocities. Gravitational waves pass through matter
essentially unaffected; hence they could enable investigation of regions never seen directly before.

Specifically, detection of supernovae by gravitational wave detectors could lead to a greater
understanding of core collapse, providing statistics showing the frequency and distribution of local
core collapses, the ratio of the production of black hole to neutron stars and also the frequency of
core collapses not producing visible supernovae. Additionally we may be able to imply the extent
of asymmetry in the collapse as well as the collapse and rebound timescales, the mass and angular
velocity of the neutron star (if produced), and possibly even be able to put an upper limit on the
masses of neutron stars. One of the major consequences of the detection of coalescing binaries,
CHAPTER 1. INTRODUCTION.

Figure 1.2: Masses oscillating due to a gravitational wave. Diagram from Davies (1980)

apart again from the obvious distribution statistics and component mass information, is that their
distances can be quite easily calculated. Schutz (1986b) has shown that this could lead to an
accurate determination of the Hubble constant which in turn would determine the age of the Universe
as well establishing a distance scale to galaxies external to our Local Group. Continuous sources
and stochastic background radiation would also supply astrophysical information, concerning mass
distribution and the early universe, which could influence theories on galaxy formation. Of course
it's entirely possible that there may be gravitational waves detected from unexpected or indeed
completely unknown sources.

There are of course implications outside of astrophysics to gravitational wave research. Nuclear
physics could benefit from studies of neutrons in neutron stars, under conditions not obtainable on
Earth. Also there are many potential technological benefits from detector development in such areas
as laser technology, vacuum systems and seismic isolation systems.

1.0.3 Detectors

Weber (1960) showed that a mass quadrupole harmonic oscillator would be excited by a gravita-
tional wave passing through it. The simplest form of this is just two masses joined by a spring, as
shown in Figure 1.2. The impinging wave would alter the separation of the masses against the action
of the spring, setting up oscillations within the system of the form of those produced by a forced,
damped harmonic oscillator. This provides the physical basis for the two main gravitational wave
detector designs. In bar detectors the masses are represented by a single solid bar of material such
as aluminium, which also supplies the spring, in the form of the material's elasticity. In laser inter-
**CHAPTER 1. INTRODUCTION.**

**ferometers** the masses are supported as pendula with the opposing force supplied by the component of \( g \), the acceleration due to gravity along the direction of the pendulum arc.

**Bar detectors**

The first serious effort made towards developing a detector for gravitational radiation was by Professor J. Weber at the University of Maryland in the late 50's. Weber's bar detectors consisted of a cylindrical bar of aluminium a few tons in weight and a few meters in length, freely suspended from some fixed frame. The component of the gravity wave passing through the bar, perpendicular to its length, would cause the ends of the bar to displace relative to each other, the amplitude of which depends upon the amplitude of the wave and the inter-molecular forces within the bar. These displacements are converted into electrical signals by sensors, amplified and displayed on a monitor. Burst events would pass through the bar in less then 1ms, but using certain suitable materials for the bar such as niobium or aluminium, a very large damping time is possible, maintaining vibrations in the bar long after the wave has passed, improving the chances of detection. The wave passing through the bar would excite several different modes of vibration in the bar. If the bar is tuned so that one of these modes is the same as the bar's fundamental mode of vibration then large excitement could be obtained. Most bars in operation today are tuned to a fundamental frequency of about 1kHz and have a very narrow bandwidth. This optimises them for the detection of core collapses and also makes data analysis much less troublesome.

There are three types of bar in operation today,

1. **Room temperature bars** such as the one at Maryland, these have a strain sensitivity between \( 10^{-16} \) and \( 10^{-17} \). These detectors are limited by thermal noise i.e. the material of the bar.

2. **The thermal noise problem** is reduced in cryogenic bars where the bar is cooled to only a few degrees K, achieving thus far, sensitivity of about \( 10^{-18} \). Examples of this type of detector may be found in Rome, Perth, Stanford and Louisiana State University.

3. **The third type of detector** is the so called Torsion Pendulum, which consists of two large masses connected by a torsion fiber. These detectors are as sensitive as other bars but are tuned to a continuous source, e.g. the pulsars, rather than looking for burst events. They gain sensitivity for continuous sources as,

\[
\text{sensitivity } \propto \text{ [observing time]}^{1/2}.
\]

A detector of this type has been constructed in Tokyo specifically to look for radiation from the Crab pulsar (Owa et al(1986)).
CHAPTER 1. INTRODUCTION.

With a best sensitivity of around $10^{-18}$, improvements of at least 3 orders of magnitude are necessary, for the detectors to become useful. Current bars however, could potentially detect supernova in our galaxy and they remain our best hope of a detection in the next few years at least. The way towards increased sensitivity for bars is via ultra cryogenic bars now being developed at temperatures below 100mK or less.

Ultimately bar sensitivity must go below the Standard Quantum Limit to reach $10^{-21}$. This limit is reached when the energy of vibration of the bar is exactly 1 phonon or $\hbar \omega$. Caves(1980) showed that squeezing could allow this limit to be exceeded. If M is the mass of bar, $\omega$ the angular frequency of the bars mode, $\hbar$ Planck's constant and $X_1$ and $X_2$ are a pair of quantum-mechanically conjugate observables, then the uncertainty principle confines the accuracy to which $X_1$ and $X_2$ are measured, by

$$\Delta X_1 \Delta X_2 \geq \frac{\hbar}{(2M\omega)} \quad (1.3)$$

where, $\hbar = \frac{\hbar}{(2\pi)}$. \quad (1.4)

It is then possible in principle, by squeezing, to reduce the uncertainty $\Delta X_1$ in $X_1$ at the expense of $X_2$ and then use $X_1$ as a measure of the gravitational wave amplitude. This could eventually allow strain sensitivities below $10^{-20}$, but thus far no practical implementation of squeezing for bars has been demonstrated because the technical problems seem formidable. For an up to date discussion see Michelson,P.F.et al (1991). Most recent attention has been focused on interferometric detectors.

Laser interferometric detectors

Although Weber, in collaboration with Forward (see Forward(1971)) originally suggested the use of laser interferometers to detect gravity waves in the early 60's, it was not until the early 70's that research and development began in earnest.

The basic design is similar to that of the Michelson interferometer, with 3 freely suspended masses, in the form of 2 orthogonal arms of equal length with a beam splitter attached to the mass at the intersection and mirrors attached to the masses at the ends. The beam splitter transmits equal intensities of light along both arms, the reflected light from both is then recombined at the splitter and transmitted to the photo diode.

When a gravity wave is incident upon the detector the relative length of the arms change, which produces a shift in the interference fringe from which the difference in lengths, $\delta l$, of the arms can be inferred.

The change $\delta l$ is dependent on the length of the arms, and a linear combination, $h$, of the amplitudes of the polarisation states, $h_+$ and $h_-$,

$$\delta l = l_0 h \quad (1.5)$$
where $l_0$ is the stationary arm length.

To get an idea of the magnitude of the problem faced in detecting $\delta l$ consider a detector with arm length 10m (as in the case of the Glasgow prototype), and a wave of amplitude $10^{-21}$ then, $\delta l = 10^{-20}$m, which is 5 orders of magnitude smaller than the classical radius of an electron.

To make measurements of this sensitivity possible, modern interferometers will have long arm lengths (3-4km) which is effectively increased by keeping the light in the arms for many reflections. Light power is built up further by power recycling (Drever 1983), while narrow banding may be achieved by signal recycling (Meers 1988), demonstrated in the lab by Strain and Meers (1991).

There are several working prototype interferometers in operation at the moment, in Glasgow, Munich, Caltech and Japan, with all but the Japanese prototype giving sensitivity of order $10^{-18}$ comparable with the most sensitive cryogenic bars. Many long armed detectors are currently planned by these groups and others around the world, particularly the VIRGO collaboration between Italy and France. They should eventually achieve sensitivities of $10^{-22}$ or even better.

"Space Based Detectors"

Currently prospective gravitational wave detection in space amounts only to tracking interplanetary spacecraft. The future however, could see space based Laser Gravitational Wave Observatories, consisting of triangular arrangements of 3 spacecraft, separated by $10^7$km. This huge arm length should enable the detection of waves with comparable wavelengths with subsequently far lower frequencies than any detectable on Earth, ie in the range $10^{-5} - 10^{-2}$Hz. This is made possible by the lack of low frequency ground vibration, that Earth based detectors are unable to escape. This could enable the detection of very many white dwarf binaries as well as long period massive binary systems long before coalescence.

1.0.4 Data analysis

One of the problems associated with an operating broadband gravitational wave detector, such as a laser interferometers, is the quantity of data produced. To be effective in the required frequency range requires a sampling rate of at least 10000Hz for 2 Byte data, which amounts to of order $10^{12}$ Bytes per year when in continual operation, without even consideration of any addition diagnostic or environmental data that may also be taken. This quantity of data, regardless of the method of storage, would amass very rapidly if its analysis could not keep up with the data taking rate. Hence the development of an automatic analysis system that continually analyses the data, online, immediately reporting certain types of events such as coalescing binaries and supernovae explosions.
CHAPTER 1. INTRODUCTION.

Immediately reporting supernovae would be particularly important as they may well be apparent to gravitational wave detectors before any other observatories, giving optical and other types of astronomers the chance to observe one actually brightening, as opposed to studying it only after the event as has been the case up to now. Any system should also be conducting pulsar searches and be able to cross-correlate results with those obtained by other detectors as well as cross-correlating actual data from several detectors looking for the stochastic background as well as new unpredicted sources, although this beyond the scope of this thesis.

1.0.5 Contents of the thesis

The work described in thesis is the first attempt to build an automated data analysis system, which was tested on actual data taken by a prototype gravitational wave detector. This data was taken by the Glasgow prototype detector during a 100 hour data taking run, coincident with a similar run on another prototype in Germany, in March 1989. The data from the Glasgow run was recorded on video cassettes by an Exabyte device. The analysis software was designed in parallel, splitting the analysis up into 5 distinct parts each of which was carried out simultaneously, on 5 distinct processors. The decision to make the system parallel was taken mainly because the final system, having to deal with large amounts of data very rapidly, would probably also have to be parallel in some way, hence future parallel designs could benefit from experience gained now.

The scope of the analysis was obviously limited by the processors used, the transputers, both in terms of their processing speeds and their memories. A system could have been developed that analysed the data as completely as wanted, searching for burst events- reconstructing many hundreds of filters for each new stretch of data- as well as passing large amounts of the data continuously back and forth to the hard disk of the PC for the pulsar searches and even auto-correlating large stretches of data, searching for stochastic background and new sources. Such a system would be ungainly, complex and very very slow but perhaps justifiable if there were any real chance of the data containing any signals. The sensitivity of the detector is such, however that this is almost certainly not the case. For this reason therefore, more than any other, the routines were designed primarily, to be efficient in terms of time, searching only for time series bursts and coalescing binaries, crosscorrelating the data with only as many filters as can be used without slowing the system, the main output being a statistical picture of both the noise and the environmental housekeeping data as well as providing false alarm statistics.

Briefly then the process read the data into the transputer network for analysis. The data was composed of the noise potentially containing the gravitational wave signal and also streams of house-
keeping data. The noise was firstly optimally filtered, before any event searches were carried out. The time series was searched for wide frequency bursts as well as being cross-correlated with a suite of filters based on the predicted characteristic shape of coalescing binaries. An event search was also carried out on this data. The results of the analysis were written to another video cassette.

Summary of Chapters

Chapter 2 describes in more detail the sources mentioned above, concentrating on core collapse and coalescing binaries providing mathematics necessary for the analysis.

Chapter 3 is in two separate parts. The first gives an outline of all relevant mathematics used, concentrating on the Fast Fourier Transform (FFT), which is the main mathematical tool used in the analysis, giving it both in its analytical and discrete forms, and showing how it can be used to find the crosscorrelation between two data streams. Noise is briefly discussed statistically and the mathematical technique employed to extract signals from noise, Matched Filtering, is described in some detail. The second part of this chapter briefly describes the computational hardware used and also discusses parallel processing.

Chapter 4 describes the Glasgow prototype detector in detail, describing it as a Fabry-Perot laser interferometer. It is compared and contrasted with the German detector which is of a different design, being a delay line interferometer. The structure of the data taken by the Glasgow detector is described in detail.

Chapter 5 describes the software in detail. It begins by describing the overall parallel structure of the program before going on to describe the separable procedures within the program in more detail. Included in this are many figures that show the changes in the noise as it passes through the system.

Chapter 6 gives the results of the analysis. This includes an assessment of the noise, including how it correlates with the various housekeeping channels. It also includes an an assessment of the numbers and distributions of the events seen, investigating if they could be due to something that may appear in the housekeeping channels.

Chapter 7 gives a summary of the work described in the previous chapters, commenting on the results gained and lessons learned in the development and implementation of the system.

Beyond this there is a complete bibliography, followed by 3 Appendices, one of which is a complete listing of the software described in Chapter 5.
Chapter 2

SOURCES OF GRAVITATIONAL WAVES

2.1 Introduction

Much of the incentive behind the drive to design and construct a gravitational wave detector comes from an understanding of its potential sources and the vast new avenues of research that their discoveries could promote. Compared with those regions of the electromagnetic spectrum that have been explored for years now, gravitational radiation is of low frequency and would provide a new observational perspective, allowing the universe to be considered in terms of mass distribution rather than as just a collection of regions emitting various forms of electromagnetic radiation.

Gravitational wave signals looked for on Earth are generally considered to fall into three main classes; burst, continuous and stochastic. Any signal whose duration is of such length that the rotation of the Earth does not have to be taken in account, when extracting it from a data stream, is called a burst. Any longer signal then is considered to be continuous. Numerous weak signals would overlap producing varying levels of stochastic background radiation at all frequencies.

A good upper bound for the amplitude of the gravitational waves $h$, from these sources at the Earth is given by

$$|h| \leq \frac{\Phi_N \Phi_{int}}{c^4},$$

(Schutz 1984) where $\Phi_N$ and $\Phi_{int}$ are the Newtonian potentials at the observer's distance ($r$) and inside the source respectively, with

$$\Phi_N = \frac{GM}{r}.$$
2.2 BURSTS

2.2.1 Gravitational collapse.

Detection of the gravitational radiation produced by the collapse of the cores of massive stars has, over the years, been the main motivation behind gravitational wave detector research and development. It remains to this day the most likely form of event that bar detectors are likely to find. These collapses are possibly associated with type II supernovae and produce neutron stars or black holes in the mass range 1-10\(M_\odot\), radiating waves in the frequency range 1-10 kHz.

The amplitude of the gravitational radiation received on Earth from one of these events, would depend on the energy released (and therefore the mass of the star) during the collapse, the distance of the source from Earth and the extent to which the core collapse was non-spherical. The axisymmetric calculations of Muller(1982) and Piran and Stark(1986) show that the greater the mass of the body produced, the greater the percentage of that mass that is radiated away as gravitational waves. For example for a 1 \(M_\odot\) neutron star 0.00001\% of its mass is realised in radiation whereas for a black hole of mass \(10\ M_\odot\) the figure is \(10^{-2}\%\).

This nonaxisymmetric consideration hinges largely on the rotation speed of the collapsing core. If the core rotates only moderately rapidly then during the collapse it will flatten out axisymmetrically. With a relatively larger rotation speed however, initially insignificant nonaxisymmetric instabilities may start growing rapidly, altering the shape of the core, making the rotation less and less axisymmetric increasing the amount of gravitational radiation produced.

The structure of the waves emitted in either of the above scenario is still not known analytically, (although numerical models for core collapse are currently under development, Finn(1989) and Muller(1991)) hence detection by matched filtering is not possible. For this reason it is necessary for the waves to be emitted at amplitudes great enough to cause them to stand out above the noise within a detector, to enable them to be detected.

Using figures computed by Piran and Stark(1986) based on axisymmetric collapse an estimate of the upper limit for the wave amplitude is

\[ h \approx 4 \times 10^{-22}, \]

for supernovae leading to the production of black holes in the Virgo cluster of galaxies, where \(\rho_\odot\) \(5^\circ\) supernovae are thought to occur each year. Hence with a sensitivity of \(10^{-22}\), which detectors should be achieving within the next ten years or so, the hopefully large number of appropriate events in Virgo that approach this upper limit should be detected. An estimate for the amplitude of waves produced by nonaxisymmetric collapse can be obtained from Schutz(1990) using equation 2.1 taking
$H_{\text{int}} \approx 0.3$ and $M \approx 10M_\odot$ for a black hole in the Virgo cluster we get
\[ h \approx 1 \times 10^{-20}. \]

However, if a gravitational collapse took place in our galaxy then $h$ could be of the order $10^{-18}$ which is a sensitivity achieved now by cryogenic bars and some prototype LIGO's. Estimates of how frequently such events may occur in the Milky Way vary from about 3-30 years or so, but bars are fairly easy to run over long periods of time, so a supernova in the next few years, in our galaxy, may well provide the first direct evidence for the existence of gravitational waves.

2.2.2 Coalescing Binaries

Binary systems emit gravitational waves along their rotational axis. These waves carry energy away from the system, causing both the separation of the bodies to decrease and the frequency and amplitude of the gravitational waves emitted to increase until coalescence. The process by which a particle experiences a reaction force when it emits radiation and hence loses energy is called radiation reaction. When both components of the system are compact, i.e., neutron stars and/or black holes, then frequencies and amplitudes of an order that should be detected by planned Earth-based detectors, may be achieved before coalescence.

The mathematical model describing the coalescence and the nature of the gravitational radiation emitted is believed to be well known. Clarke and Eardley (1977) established a reasonably straightforward Newtonian approximation to the time evolution of the waves that Krolak (1989) found to be close to the post-Newtonian order. The model is supported by observations of the most famous example of a binary system that will, eventually, coalesce, i.e., PSR 1913+16, which lies in our own galaxy, with both components at about 1.4 solar masses. One component of the system is a pulsar and its companion a neutron star. Currently, the system's orbital period is about 8 hours, and falling very slowly. Observations of it are sufficiently accurate to quantity the extent of this orbital decay and have shown that the figure achieved agrees to a high degree of accuracy with that predicted by the model calculations due to the loss of energy by the emission of gravitational waves (see Taylor and Weisberg (1989) and references there in). This provides indirect evidence for the existence of gravitational waves. In $1.49 \times 10^8$ years the components of PSR 1913+16 will coalesce, with it emitted waves attaining in the last few hundredths of a second before coalescence, frequencies which would enable it to be seen by currently planned ground detectors on Earth, with
\[ h \approx 10^{-19} \]
CHIRP

FFT OF ABOVE (real part only)

FIGURE 2.1
CHAPTER 2. SOURCES OF GRAVITATIONAL WAVES

at frequencies of around 300-1000 Hz. This type of characteristic waveform, with increasing amplitude and frequency, is known as a chirp. A chirp with a frequency range between 300 and 1000 Hz is shown in Figure 2.1, together with its Fourier transform.

Having a good model of the signal produced near coalescence greatly increases the chances of detecting such an event. The nature of the chirp, between those frequency ranges detectable on Earth, is dependent on both the masses of the components of the system, (which can be combined into one number referred to as the mass parameter and defined below) and also the initial phase of the wave as it achieves some appropriate frequency, chosen to be 100Hz in this analysis. In detecting an incoming chirp these parameters must be established along with its time of arrival. This is achieved by Matched Filtering which extracts the signal from the detector noise by cross correlating the data with a template of filters each based on a different expected gravitational wave signal ie varying in mass parameter and initial phase. This technique will be discussed further in Chapter 3. The expressions used to calculate the filters are given below.

Mathematical description of coalescing binaries.

It can be shown by equating the rate of energy loss in the system with the gravitational luminosity that the time scale, \( \tau \), over which the orbit decays is given by

\[
\tau = \frac{f}{\dot{f}} = 7.8 M^{-2/3} \mu^{-1} \left( \frac{f}{100\text{Hz}} \right)^{-8/3} \rho \epsilon c
\]  

(2.3)

where \( f \) is linear frequency, \( M \) is the total mass of the system (= \( M_1 + M_2 \)) and \( \mu \) is the reduced mass (= \( M_1 M_2 / M \)), both in units of solar masses. This can be written in the form,

\[
\tau = \frac{f}{\dot{f}} = \alpha f^{-8/3},
\]  

(2.4)

where \( \alpha \) is,

\[
\alpha = \frac{7.8 \times 100^{8/3}}{\rho} \rho \epsilon c^2.
\]

\( \rho \) is the mass parameter given by,

\[
\rho = \mu M^{2/3} / M_0^{5/3}.
\]

It should be noted that as the rate of decay increases so rapidly, the true time to coalescence is only \( \frac{3}{8} \tau \).

By integrating equation 2.4 over \( f \) and \( t \) from some intial point when \( f \) and \( t \) are known and given by \( f_0 \) and \( t_0 \) to some later point, an expression giving the evolutionary relationship between \( f \) and \( t \) is found,
\[ f = \left( \frac{-8}{3\alpha} (t - t_0) + \frac{v_0}{3} \right)^{-3/8} H_z \]  

and also therefore \( \omega \), the angular frequency, since

\[ \omega = \pi f. \]

The frequency of the waves is twice the orbital frequency of the system because the initial configuration occurs twice in every orbit.

The amplitudes of both the polarisation states are given by

\[ h \approx \frac{4G^5/3\pi 2/3}{c^4} \left[ \frac{\mu}{r} \right] [f M]^{2/3} \cos(2\pi f t f(t) \gamma t + \xi) \]  

where \( \xi \) is some phase constant contained with the received waves.

By substituting equation 2.5 into the above we obtain an expression for the chirp in terms of time that has unit amplitude and a frequency of 100Hz when \( t = 0 \),

\[ \text{Chirp}(t) = \beta^{-1/4} \cos \left( \frac{320\pi(1 - \beta)^{3/8}}{0.34\rho} + \phi \right) \]  

where \( \phi \) is the phase of the chirp at 100Hz and

\[ \beta = 1 - 0.34\rho t. \]

This is used as the model for the formation of the filter template used for matched filtering.

Using equation 2.5 an expression for the time \( t \) in terms of the frequency \( f \) is found,

\[ t_f = \frac{1 - (f/100Hz)^{2/3}}{0.34\rho} \]  

From the above, an expression for the time taken for the frequency of the chirp to go from one frequency, \( f_0 \), to a higher frequency, \( f_1 \), is given by,

\[ t_{(f_0 - f_1)} = \frac{(f_0/100Hz)^{2/3} - (f_1/100Hz)^{2/3}}{0.34\rho} \]  

The maximum value of the gravitational wave amplitude \( h_{\text{max}} \) from a binary system at a distance \( r \) is achieved along the axis of the system and is given by,

\[ h_{\text{max}} = 2.6 \times 10^{-23} \rho \left( \frac{f}{100\text{Hz}} \right)^{2/3} \left( \frac{100\text{Mpc}}{r} \right). \]  

This can be shortened to

\[ h_{\text{max}} = \gamma \left( \frac{f}{100\text{Hz}} \right)^{2/3} \left( \frac{100\text{Mpc}}{r} \right). \]
where $r$ is the distance in Mpc and

$$
\gamma = 1.207 \times 10^{-22} \rho.
$$

It can be seen comparing equation 2.4 and equation 2.11 that the product $h_{\text{max}} r$, for a particular frequency is dependent solely on the distance to the binary. Hence, compensating for whatever non alignment there is between the Earth and the axis of the system, it is possible to reliably work out the distances to any coalescing binaries detected (see Schutz (1986b) and Bourzeix et al (1990)) and hence establish the distances to the system, be it a galaxy or a group of galaxies, containing the binary. Comparing these figures with those obtained by studying the redshift of the system should allow the most accurate assessment of the Hubble constant $H_0$ yet made.

An obvious consideration in the design of equipment and data analysis techniques specialised towards finding coalescing binaries is that there should be enough potential events of sufficient amplitude to make the search worthwhile. Current estimates on the event rate, based on the pulsar birth rate and the observed percentage of these that turn out to be binary pulsars suggest a figure of 3 events per year out to 200Mpc rising to 24 at 400Mpc and 81 at 600Mpc. Krolak (1989) arrived at an equation for the best signal-to-noise ratio achievable for a binary system at a distance $r$ in terms of the lower cut off frequency, $f_c$, of the detector, its sensitivity $h$ and the systems mass parameter $\rho$,

$$
\left( \frac{S}{N} \right) = 22 \rho^{1/2} \left[ \frac{h}{10^{-22}} \right]^{-1} \left[ \frac{f_c}{100\text{Hz}} \right]^{-7/6} \left[ \frac{r}{100\text{Mpc}} \right]^{-1}.
$$

(2.12)

Applying the above equation to the Glasgow prototype, taking $h \approx 10^{-19}$, $f_c = 300$Hz and the $S/N$ to be 3.5, we obtain $r = 0.2$Mpc for a system of two $1.4M_\odot$ neutron stars. This represents a volume of space roughly equal to only 1/8 of the volume of the local group of galaxies and comparing it with the figures quoted above make it unlikely in the extreme that any event may have picked up by the Glasgow detector during its hundred hour run of March 1989. A prospective future detector with $h \approx 10^{-22}$ and $f_c \approx 100$Hz, would be observing out to 650Mpc when there should be several events picked up per month. Obviously if either one or both the components of the system is a black hole then they would be detectable out to far greater distances.

**Black Hole Ringing**

There are several other possible applications of matched filtering in gravitational wave detection. One of these may be the result of a non axisymmetrical gravitational collapse leading to the production a deformed black hole, which would then undergo damped vibrations emitting gravitational radiation. The parameter dependent shape of the waveform, produced by the ringing down, is believed to be
known, (see Chandrasekhar and Detweiler (1975)), hence filter templates could again be built up that could pick the signal out of the data stream.

2.3 CONTINUOUS WAVE SIGNALS

Gravitational wave signals lasting more than 30 minutes are classified as continuous. When trying to detect signals of this duration, frequency broadening due to the doppler shifts that occur because of the motions of the Earth must be taken into account. In the short term only sources of known frequency and direction may be detected, because the sensitivities over long periods of time that detectors would need to achieve so as to search for unknown sources are not attainable. Knowing the direction and frequency of a source would enable a detector to be optimised, concentrating its efforts in looking specifically for that source. The chances of detection could then be further increased by cross-correlating data from several detectors.

2.3.1 Periodic and other Sources

Pulsars

It is thought that pulsars would be the main source of continuous wave signals. They are thought to be supernovae remnants and emit gravitational radiation by virtue of their non-axisymmetric magnetic fields. The amplitude of this radiation by itself is sufficiently slight as to be beyond the scope of any detector likely to appear in the near future. If, however, the neutron star has any significant deviation from symmetry around its rotation axis then the amplitude of the radiation produced may be sufficiently large as to be detected. This asymmetry may be caused by the deformation of the surface of the neutron star, due possibly to non uniform core collapse and internal magnetic pressures. The extent to which the overall shape of the star may be distorted in this way is the subject of much debate but if the ellipticity (or fractional distortion) of the star is $\delta$ then an estimate of the amplitude $h$ of the radiation emitted for a star of mass 1.4 $M_\odot$ and radius 10 km is,

$$h \approx 6 \times 10^{-22} \delta \left( \frac{f}{100 \text{Hz}} \right)^2 \left( \frac{10 \text{Kpc}}{r} \right),$$

(2.13)

where $f$ is the frequency of the wave (which is twice the frequency of rotation of the neutron star) and $r$ is its distance in units of 10 Kpc. Obviously the energy radiated away from the star cannot exceed its observed loss of kinetic energy in any given time period, hence knowing its mass allows the calculation of an upper limit for $\delta$. Knowing the pulsar distance and frequency also enables an estimate of the upper limit of $h$ for the pulsar, e.g. for the Crab pulsar, which emits waves of frequency...
60 Hz at a distance of 2 Kpc, δ and h are bounded by

\[ \delta < 10^{-3} \]

and

\[ h < 1.1 \times 10^{-24}. \]

**Wagoner Radiation**

If a rapidly spinning neutron star has a binary companion from which it accretes mass, its angular momentum grows and may eventually reach a point of instability after which the system will, in striving for stability, radiate away angular momentum in the form of gravitational waves. This would continue until a balance point is reached when the amount of energy radiated away matches that gained through accretion. When this point is reached the system becomes and remains, a fixed frequency emitter. Gravitational radiation from this source is known as Wagoner Radiation and is another potentially detectable periodic source (see Wagoner, 1984).

### 2.3.2 Stochastic Background

The stochastic background of gravitational radiation could consist of many waves of unpredictable amplitudes and frequencies. Possible causes would be all potential emitters of gravitational radiation, predicted and unpredicted, including sources already discussed in this chapter whose radiative amplitude, because of the source distance or whatever, is too slight to detect on Earth. Possible other sources could include cosmic strings and events in the early universe.

This background could only be detected by cross-correlating the output of several detectors. The detectors must be closer than roughly half the wavelength of the expected radiation if they are to have optimum sensitivity. For a detailed description Michelson (1987).

### 2.3.3 Low Frequency Sources

As previously mentioned, these sources would be the province of the space based detector. Possible sources could include long period binaries in our own galaxy and the formation and coalescence of massive (> 10^6M_☉) black holes in other galaxies.
Chapter 3

MATHEMATICAL TECHNIQUES

3.1 Introduction

This Chapter discusses the principles behind the essential mathematics used in this thesis. It is concerned mainly with Fourier analysis, and how associated techniques are used to extract signals from streams of noise. It begins by defining the Fourier transform of a function, and then shows how this transform is related to the cross-correlation of two functions. Both the Discrete Fourier transform and the Fast Fourier transform are then defined, and the equivalent cross-correlation relationship is shown. This is followed by a discussion about the statistics of noise and how matched filtering methods can be used to extract signals from this noise.

3.2 Fourier analysis

3.2.1 The Fourier transform

One of the principal tools of signal analysis is the Fourier transform which decomposes a signal down into a combination of sinusoidal waveforms. The Fourier transform of a continuous function $g(t)$ is given by,

$$G(f) = \int_{-\infty}^{+\infty} g(t) e^{-2\pi i ft} dt, \quad -\infty < f < \infty,$$

where $G(f)$ is a function of the frequency $f$.

This transform enables a process described in the time domain to be described in the frequency domain and vice versa, because the inverse transform also applies,

$$g(t) = \int_{-\infty}^{+\infty} G(f) e^{2\pi i ft} df, \quad -\infty < t < \infty.$$
CHAPTER 3. MATHEMATICAL TECHNIQUES

Power

The power spectrum is defined as $|G(f)|^2$, which leads to

$$\text{total power} = \int_{-\infty}^{\infty} |G(f)|^2 df = \int_{-\infty}^{\infty} |g(t)|^2 dt.$$  \hspace{1cm} (3.3)

This famous result is known as Parseval's Theorem.

3.2.2 Correlation

The \textit{cross-correlation} of two functions, $g(t)$ and $h(t)$, is a function $c(t)$ of $t$, and given by,

$$\text{CrossCorr}(g, h) = c(t) = \int_{-\infty}^{\infty} g(t_1 - t)h(t_1) dt_1 = \int_{-\infty}^{\infty} g(t_2)h(t_2 + t) dt_2.$$ \hspace{1cm} (3.4)

If $G(f)$ and $H(f)$ are the Fourier transforms of $g(t)$ and $h(t)$ respectively, and if the Fourier transform of the correlation is $F[\text{CrossCorr}(g, h)]$, then

$$F[\text{CrossCorr}(g, h)] = G(f)H(f)^*,$$ \hspace{1cm} (3.5)

where $^*$ represents the complex conjugate. From this obviously $c(t)$ can be found by taking the inverse Fourier transform,

$$c(t) = \text{CrossCorr}(g, h) = \int_{-\infty}^{\infty} G(f)H(f)^* e^{2\pi ift} df.$$ \hspace{1cm} (3.6)

If a function $g(t)$ is cross-correlated with itself, the resulting function $a(t)$ is called the \textit{auto-correlation} function and is given by,

$$\text{AutoCorr}(g) = a(t) = \int_{-\infty}^{\infty} g(t_1 - t)g(t_1) dt_1 = \int_{-\infty}^{\infty} |G(f)|^2 e^{2\pi ift} df.$$ \hspace{1cm} (3.7)

3.2.3 The Discrete Fourier Transform (DFT)

The data streams we deal with are not read continuously but are sampled discretely at evenly spaced intervals of time. If $\Delta$ is the time interval between consecutive samples then a function $g(t)$ would be read and stored in memory as a series $g_n$ given by,

$$g_n = g(n\Delta) \cdot n = 0, 1, 2, 3...N - 1.$$ \hspace{1cm} (3.8)

The sampling rate (or frequency), \textit{i.e} the number of samples each second, is just $1/\Delta$.

Because of this discrete sampling it is not possible to use the Fourier transform in its continuous form.
Discrete Fourier Transform

If we define our signal as a series of \( N \) numbers, \( g_k \) \((k = 0,1,\ldots,N-1)\), where \( N \) is a power of 2, and the DFT of the signal as a series \( \tilde{g}_j \) \((-N/2,\ldots,N/2-1)\), then these series are related by the transform pair,

\[
\tilde{g}_j = \frac{1}{N} \sum_{k=0}^{N-1} g_k e^{-2\pi i jk/N} \quad j = -N/2,\ldots,N/2-1
\]  

(3.9)

and

\[
g_k = \sum_{j=-N/2}^{N/2-1} \tilde{g}_j e^{2\pi i jk/N} \quad k = 0,\ldots,N-1.
\]  

(3.10)

The time duration of the signal is \((N-1)\Delta\) and the frequencies are resolved only at discrete values given by,

\[
f_n = \frac{n}{N\Delta}, \quad n = -N/2,\ldots,N/2.
\]  

(3.11)

The above equation gives \((N+1)\) values of \(f_n\), but only \(N\) distinct values as \(f_n\) for \(n = -N/2\) and \(n = N/2\) are identical.

The discrete Fourier transform of a data stream can only correctly pick out frequencies of up to half the sampling frequency. This critical frequency \(f_c\), is known as the Nyquist frequency, and is given by,

\[
f_c = \frac{1}{2\Delta}.
\]  

(3.12)

If a sampled function contained no frequencies greater than the Nyquist frequency, then it would be completely determined by sampling over an infinite time duration. However if \(f_c\) is exceeded then problems can arise due to aliasing, which allows relatively large frequencies to masquerade as frequencies less than \(f_c\). This can be seen by considering a sine wave of frequency slightly greater than half the sampling frequency. If the wave is sampled at its positive peak, the next sampled point is just after the wave trough. However an external observer would interpret the point as being just prior to the trough, getting the wrong wavelength and hence the wrong frequency. This can be seen mathematically for a sine wave of frequency \(f > f_c\) sampled with frequency \(f_s\),

\[
\sin 2\pi ft = \sin 2\pi [(f - f_s)t + f_s t]
\]  

(3.13)

\[
= \sin 2\pi (f - f_s) t \cos 2\pi f_s t + \sin 2\pi f_s t \cos 2\pi (f - f_s) t
\]  

(3.14)

\[
= \sin 2\pi (f - f_s) t_n \quad \text{at} \quad t_n = n/f_s, \quad n = 0,1,2,\ldots.
\]  

(3.15)

Then by a simple inductive argument,

\[
\sin 2\pi f t_n = \sin 2\pi (f - f_s) t_n = \sin 2\pi (f - n f_s) t_n, \quad n = 1,2,3,\ldots
\]  

(3.16)

and if \(|f - n f_s| < f_c\), it would be picked up as a lower frequency, by the discrete Fourier transform.
CHAPTER 3. MATHEMATICAL TECHNIQUES

30

Power

Parseval's Theorem has a discrete form also,

\[ \sum_{k=0}^{N-1} |f_k|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |\tilde{f}_n|^2. \]  

(3.17)

Discrete Correlation

When carrying out actual analysis we may want to cross-correlate a finite data set \( h_j \) of length \( N \) with an infinitely long set \( g \). This is not possible, so it becomes necessary to break \( g \) up into many data sets \( g_k \) again of length \( N \), and then carry out the correlations set by set.

Therefore if we have two data sets \( g_k \) and \( h_j \) each of \( N \) elements, with \( k, j = 0, 1, ..., N - 1 \), then their crosscorrelation, \( c_j \), is given by the Circular Correlation formula,

\[ c_j = \frac{1}{N} \sum_{k=0}^{N-1} g_k h_{k+j}, \quad j = 0, ..., N - 1. \]  

(3.18)

It is necessary to make \( h_j \) periodic, ie

\[ h_{j+N} = h_j \quad \text{for all } j, \]  

(3.19)

unfortunately however problems will then arise from this periodicity. As \( k \) increases the beginning of the time series would be multiplied by the end of the filter which is clearly erroneous. This wraparound can be partially overcome by ensuring that \( h_j \) is largely made up of zeros, in which case \( g_k \) is much longer than the non zero part of \( h_j \) as they are the same total length. In this case, if there are \( N_k \) non-zero points in \( h_j \), then the last \( N_k \) points of the correlation would have to be ignored, provided that the non zero part of the filter is at its start.

Analogous to crosscorrelation using the continuous Fourier transform, the DFT of the circular correlation \( c_j \), given by \( \tilde{c}_k \), is related to the DFT's of \( g_j \) and \( h_j \), given by \( \tilde{g}_k \) and \( \tilde{h}_k \) respectively by the discrete version of equation given in 3.5, ie

\[ \tilde{c}_k = \tilde{g}_k(\tilde{h}_k)^* \quad k = -N/2, ..., N/2 - 1, \]  

(3.20)

where * again represents the complex conjugate.

3.2.4 The Fast Fourier Transform (FFT)

Calculating the DFT of a series of \( N \) numbers normally would require \( O(N^2) \) multiplications, which, when many DFT's are required of long series, can be very time consuming, even when the fastest of today's computers are used. Hence the widespread use of the fast Fourier transform.
CHAPTER 3. MATHEMATICAL TECHNIQUES

The FFT is a computer algorithm that, given a series of \( N \) numbers, produces the DFT of the series with only \( O(N \log_2 N) \) multiplications, which compares very favorably with \( O(N^2) \) when \( N \) is large. The proof is discussed in detail in Bracewell (1986) and many other publications, including Press et al (1986) from which the FFT algorithms used in this work were taken. It follows generally from the fact that when \( N \) is an even number, the DFT of the series can be decomposed into a linear combination of the DFT’s of two series of length \( N/2 \), the first series being the even numbered points of the original stream, and the second the odd,

\[
\tilde{a}_j = \frac{1}{N} \sum_{k=0}^{N-1} a_k e^{-2\pi i j k/N} \quad \text{(3.21)}
\]

\[
= \frac{1}{N} \sum_{k=0}^{N/2-1} a_{2k} e^{-2\pi i j (2k)/N} + \frac{1}{N} \sum_{k=0}^{N/2-1} a_{2k+1} e^{-2\pi i j (2k+1)/N} \quad \text{(3.22)}
\]

\[
= \frac{1}{2} \left\{ \frac{2}{N} \sum_{k=0}^{N/2-1} a_{2k} e^{-2\pi i j k/(N/2)} + e^{-2\pi i j/N} \sum_{k=0}^{N/2-1} a_{2k+1} e^{-2\pi i j k/(N/2)} \right\} \quad \text{(3.23)}
\]

In equation 3.23, the summations within the brackets are simply the DFT’s of the even and the odd numbered elements of \( \tilde{a}_k \) respectively, for \( j < N/2 \). When \( j \geq N/2 \), the summations are equal to those for \( j - N/2 \). This can be shown, considering only the even numbered summation, as follows,

\[
\frac{2}{N} \sum_{k=0}^{N/2-1} a_{2k} e^{-2\pi i j k/(N/2)} = \frac{2}{N} \sum_{k=0}^{N/2-1} a_{2k} e^{-2\pi i (j-(N/2)+(N/2)) k/(N/2)} \quad \text{(3.24)}
\]

\[
= \frac{2}{N} \sum_{k=0}^{N/2-1} a_{2k} e^{-2\pi i (j-(N/2)) k/(N/2)} e^{-2\pi i (N/2) k/(N/2)} \quad \text{(3.25)}
\]

\[
= \frac{2}{N} \sum_{k=0}^{N/2-1} a_{2k} e^{-2\pi i (j-(N/2)) k/(N/2)} \quad \text{(3.26)}
\]

Analogously, the same can be applied to the other summation over the odd numbered elements of \( \tilde{a}_k \).

Thus, from the two transforms, over \( (N/2) \) elements, \( \tilde{a}_k \) can be completely constructed with only the \( N \) extra complex multiplications due to the factor \( e^{2\pi i j/N} \) in front of the second summation. If \( N \) is a power of 2, these steps can be repeated recursively until the DFT is just a linear combination of transforms of length 1. This occurs after \( \log_2 N \) steps. Each step requires \( N \) complex multiplications to produce the transforms needed one step higher, hence the total number of operations is of order \( N \log_2 N \). The Inverse Fast Fourier Transform (IFFT) is performed using a very similar method.

When performing circular correlations computationally it is most efficient, in terms of time, to use equation 3.20, utilizing FFT algorithms to find the DFT's.
3.3 Extraction of a signal from a noisy background

As described in Chapter 2, the three main classes of signal are continuous, burst and stochastic. This analysis is primarily concerned with bursts, in particular milli-second broadband bursts, ie events whose energy is spread throughout a wide frequency range, typically produced by supernovae and also ‘chirp’ waveforms produced by coalescing binaries.

Either of these types of bursts could stand out above the noise, and could hence be detected by simply exceeding some preset threshold in $S/N$, but often this would not be the case, they would be hidden within the noise and therefore be invisible to the simple $S/N$ criteria. As the chirp waveform is relatively long and, hopefully at least, well known, the probability of its detection in particular, would benefit from Matched Filtering which is used to pull the signal out, above the noise and make it detectable by the same $S/N$ criteria as above.

3.3.1 Noise

In developing a model for the matched filtering it was necessary to assume that the noise in a detector would be a random variable in both frequency and time, with zero expectation value $\langle n(t) \rangle$ (or mean), ie,

$$\langle n(t) \rangle = \langle \tilde{n}(f) \rangle = 0.$$  \hfill (3.27)

From this the variance, $\sigma^2(f)$, of the noise is given by,

$$\sigma^2(f) = \langle [\tilde{n}(f) - \langle \tilde{n}(f) \rangle]^2 \rangle = \langle \tilde{n}^2(f) \rangle.$$  \hfill (3.28)

It was also necessary to assume that the general statistical properties of the noise don’t change with time, in which case the noise is said to be stationary and its spectral density, $S(f)$, is defined by the equation,

$$\langle \tilde{n}(f) \tilde{n}^*(f') \rangle = S(f) \delta(f - f'),$$  \hfill (3.29)

where $\delta(f - f')$ is a delta function.

This implies that noise at any frequency $f$ is not correlated with the noise at any other frequency $f'$ with $f \neq f'$, and that the function, $S(f)$, is equal to the variance of the noise at that frequency $f$, ie,

$$S(f) = \sigma^2(f)$$  \hfill (3.30)

If the prevailing statistic within the noise, ie the variance or the mean, changed during the course of an analysis run, then the results of the filtering would be called into question, hence the noise
statistics must be monitored continually during the run. It is assumed here that the noise is white, ie the spectral density $S(f)$, is not a function of $f$.

For noise that is both white and Gaussian, the probability density function $P(x)$ is given by,

$$P(x) = \frac{1}{(2\pi)^{1/2}\sigma} e^{-(x-\mu)^2/2\sigma^2}. \quad (3.31)$$

This means that the probability of any one sample lying in an interval $dx$ at $x$ is given by $P(x)dx$. This number is numerically equal to the number of samples in the interval $dx$, divided by the total number of samples.

This also has the property, now assuming zero mean, that

$$P(-\infty < x < \infty) = \frac{1}{(2\pi)^{1/2}\sigma} \int_{-\infty}^{\infty} e^{-x^2/2\sigma^2} \, dx = 1, \quad (3.32)$$

as the sample has to lie somewhere.

The probability then, that a sample would lie at a distance, greater than some threshold $T$ from the mean (still assumed to zero), is given by,

$$P(|x| > T) = \frac{1}{(2\pi)^{1/2}\sigma} \int_{-\infty}^{-T} e^{-x^2/2\sigma^2} \, dx + \frac{1}{(2\pi)^{1/2}\sigma} \int_{T}^{\infty} e^{-x^2/2\sigma^2} \, dx \quad (3.33)$$

$$= \frac{1}{\sigma} \left( \frac{2}{\pi} \right)^{1/2} \int_{T}^{\infty} e^{-x^2/2\sigma^2} \, dx \quad (3.34)$$

$$\approx \left( \frac{2}{\pi} \right)^{1/2} \left( \frac{\sigma}{T} \right) - \left( \frac{\sigma}{T} \right)^3 + \ldots \right) e^{-T^2/2\sigma^2}. \quad (3.35)$$

The last line is an asymptotic approximation to the previous equation.

In order to limit the expected number of false alarms to one in $N$ data points we must solve,

$$P(|x| > T) = \frac{1}{N}, \quad (3.36)$$

for $T$. If the Glasgow sampling rate of 20kHz were applied for a whole year, and we wanted only one false alarm in that time, then we would need $T \approx 7.1\sigma$.

When gravitational wave detectors are established around the world, clearly a suspected event in one could only be credited as an actual event if witnessed in other detectors as well. This coincidence criterion allows the threshold $T$ to be reduced as a false alarm would be a threshold crosser simultaneously in two or more detectors.

For two detectors with noise levels $\sigma_1$ and $\sigma_2$ the equation to solve is now,

$$P(|x| > T_1)P(|x| > T_2) = \frac{1}{N}. \quad (3.37)$$

If we consider two identical detectors at the same site running at 20kHz for a year, the threshold for both is $T \approx 4.85\sigma$. For three it reduces further to $T \approx 3.8\sigma$. 
There are several kinds of noise produced within interferometers, and their sources will be discussed in Chapter 4. In the Glasgow prototype, seismic noise is dominant at low frequencies, $f_s \leq 300$ Hz, which effectively provides a low frequency cutoff at $f_s$ which would have to be taken into account when forming filters.

### 3.3.2 Matched Filtering

Matched filtering is a mathematical technique that is employed to extract signals of a known form from noise. It is performed by cross-correlating a stream of noise with a filter resembling the signal to be found. Essentially the filter moves along the noise point by point, comparing each stretch of data with the filter in turn. The value returned then would be proportional to the extent of the matching of the filter, with the length of data, following the first point where the filter is applied. For a more complete discussion of matched filtering and its applications within the search for gravitational waves, see Schutz(1990).

Quantitatively if a signal $h(t)$ is present, buried in noise $n(t)$, then the output $o(t)$ of the detector is given by,

$$o(t) = n(t) + h(t).$$  (3.42)
CHAPTER 3. MATHEMATICAL TECHNIQUES

Given the detector output, to establish whether or not there is a signal with certain parameters present, the output should be cross-correlated with the corresponding filter, i.e., a filter designed on the basis of the same parameters. If the filter is \( q(t) \) then the correlation \( c(t) \) is given by,

\[
c(t) = \int_{-\infty}^{\infty} o(t')q(t + t')dt',
\]

(3.43)

which from equation 3.6, can also be written in terms of \( \hat{o}(f) \) and \( \hat{q}(f) \) the Fourier transforms of \( o(t) \) and \( q(t) \) respectively as,

\[
c(t) = \int_{-\infty}^{\infty} \hat{o}(f)\hat{q}(f)^*e^{2\pi ift}df.
\]

(3.44)

Provided the noise is Gaussian, the expectation value of \( c(t) \) is given by,

\[
\langle c(t) \rangle = \text{CrossCorr}(h, q)
\]

(3.45)

\[
= \int_{-\infty}^{\infty} \hat{h}(f)\hat{q}(f)^*e^{2\pi if\tau}df.
\]

(3.46)

The correlation is maximised when \( t = \tau \), the time shift between the filter and the signal in the noise, which gives,

\[
\langle c_{\text{max}} \rangle = \int_{-\infty}^{\infty} \hat{h}(f)\hat{q}(f)^*df.
\]

(3.47)

The standard deviation of the correlation noise is given by the square root of the variance of the correlation, where the variance is given by,

\[
\langle [c(t) - \langle c(t) \rangle]^2 \rangle = \langle [\text{CrossCorr}(n, q)]^2 \rangle
\]

(3.48)

\[
= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (\hat{n}(f)\hat{n}(f'))|\hat{q}(f)^*\hat{q}(f')^*e^{2\pi if't}e^{2\pi if't}|df'df
\]

(3.49)

\[
= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(f)\delta(f - f')|\hat{q}(f)^*\hat{q}(f')^*|df'df
\]

(3.50)

\[
= \int_{-\infty}^{\infty} S(f)|\hat{q}(f)^*|^2df
\]

(3.51)

\[
= \int_{-\infty}^{\infty} S(f)|\hat{q}(f)|^2df.
\]

(3.52)

The signal to noise ratio then is given by,

\[
\frac{\langle c(t) \rangle}{\langle [c(t) - \langle c(t) \rangle]^2 \rangle^{1/2}} = \frac{\int_{-\infty}^{\infty} \hat{h}(f)\hat{q}(f)^*e^{2\pi ift}df}{\int_{-\infty}^{\infty} S(f)|\hat{q}(f)|^2df}^{1/2}.
\]

(3.53)

It can be shown that the filter \( q(t) \), that maximises the chance of finding a signal \( h(t) \), is as expressed in the Fourier plane,

\[
\hat{q}(f) = \frac{\hat{h}(f)}{S(f)}
\]

(3.54)

which leads to the largest attainable \( S/N \) which from equation 3.35 is,

\[
\left( \frac{S}{N} \right)^2 = \frac{\langle c_{\text{max}} \rangle}{\langle [c(t) - \langle c(t) \rangle]^2 \rangle}
\]

(3.55)
\[
\begin{align*}
&= \frac{(\int_{-\infty}^{\infty} \hat{h}(f)\bar{\hat{h}}(f)\,df)^2}{\int_{-\infty}^{\infty} S(f)\bar{\hat{h}}(f)\,df} \\
&= \frac{(\int_{-\infty}^{\infty} \hat{h}(f)\bar{\hat{h}}(f)\,df)^2}{\int_{-\infty}^{\infty} S(f)S(f)\,df} \\
&= \frac{(\int_{-\infty}^{\infty} \hat{h}(f)\bar{\hat{h}}(f)\,df)^2}{\int_{-\infty}^{\infty} \hat{h}(f)\bar{\hat{h}}(f)\,df} \\
&= \int_{-\infty}^{\infty} \frac{\hat{h}(f)}{S(f)}\,df \\
&= 2 \int_{0}^{\infty} \frac{\hat{h}(f)}{S(f)}\,df.
\end{align*}
\] (3.60)

Thus by the use of matched filtering, chirp signals can be made to stand above the noise within the correlation and hence can be found by the application of the \( S/N \) threshold criteria mentioned at the start of this section.

It should be noted that the search for broadband bursts, although effectively simpler - just applying the threshold criteria to a stretch of data - is strongly analogous to matched filtering, with the filter being a \textit{delta} function in time. This enables us to demonstrate why narrow band sources may be easier to detect than broadband sources of the same energy.

Assuming that \( S(f) \approx \sigma^2(f_c) \) where \( f_c \) is some central frequency then equation 3.60 becomes,
\[
\left( \frac{S}{N} \right)^2 = \frac{2}{\sigma^2(f_c)} \int_{0}^{\infty} |\hat{h}(f)|^2 \,df
\] (3.61)
\[
= \frac{2}{\sigma^2(f_c)} \int_{0}^{\infty} |\hat{h}(t)|^2 \,dt.
\] (3.62)

If the broadband burst is centered at frequency \( f_b \) then its duration would approximately be \( 1/f_b \) and \( |h(t)| \approx h \) then from equation 3.62 we would get,
\[
\frac{S}{N} \approx \frac{h}{\sigma(f_b)f_b^{1/2}}.
\] (3.63)

Narrow band signals centered at frequency \( f_n \) would last for a time \( n/f_n \) where \( n \) is the number of cycles so equation 3.62 would become,
\[
\frac{S}{N} \approx \frac{hn^{1/2}}{\sigma(f_n)f_n^{1/2}}.
\] (3.64)

It is apparent from the above that \( S/N \) is a factor of \( n^{1/2} \) times higher in the narrow band case where clearly it is beneficial to include as many cycles in the filters as is computationally expedient. The narrow band case, when comparing coalescing binaries with supernovae, can also benefit from having a lower central frequency.

If a signal \( h(t) \) is picked up over a stretch of data by its corresponding filter \( s(t) \) then,
\[
\hat{h}(f) = A \hat{s}(f),
\] (3.65)
where $A$ is just some linear multiple. When $\bar{s}(f)$ is normalised to unity then $A$ represents the actual amplitude of the wave. So putting equation 3.65 in equation 3.60 we get,

$$
\left( \frac{S}{N} \right)^2 = 2 \int_0^\infty \frac{|\tilde{h}(f)|^2}{S(f)} \, df \quad (3.66)
$$

$$
= 2A^2 \int_0^\infty \frac{|\bar{s}(f)|^2}{S(f)} \, df. \quad (3.67)
$$

This gives an expression for $A$,

$$
A = \left( \frac{S}{N} \right) \left( 2 \int_0^\infty \frac{|\bar{s}(f)|^2}{S(f)} \, df \right)^{-1/2}. \quad (3.68)
$$

Again assuming that $S_h(f) \approx \sigma^2(f_c)$, equation 3.68 becomes,

$$
A = \sigma \left( \frac{S}{N} \right) \left( 2 \int_0^\infty |\bar{s}(f)|^2 \, df \right)^{-1/2}. \quad (3.69)
$$

In applying equation 3.69 to real, sampled data, its discrete version is used, i.e,

$$
A = \sigma \left( \frac{S}{N} \right) \left( 2 \delta f \sum_{i=1}^N |\bar{s}_i|^2 \right)^{-1/2}, \quad (3.70)
$$

where $\delta f$ is the frequency spacing between samples and $N$ the total number of samples.
Chapter 4

The GLASGOW PROTOTYPE DETECTOR

4.1 Introduction

Work on the development of a gravitational wave detector, by a group in the Department of Physics and Astronomy at Glasgow University, began in the very early seventies, with the initial emphasis on resonant bar detectors. The group, headed at that time by Professor R.W.P Drever, concluded in about 1976 that the detection of changes in the relative displacements of nearby masses by optical interferometry, an idea first suggested by Weber in the 60’s, represented a more promising means of detecting gravitational waves. For roughly the next three years they worked on a small scale delay line interferometer, switching in 1979 to the development of a 10 m Fabry-Perot prototype interferometer. It is this detector that the group, now headed by Professor J Hough, are currently working on. In the Max-Planck-Institut für Quantenoptik, Garching, Germany a group have been developing their own delay line interferometer, with 30 m long arms. In March 1989 the two groups carried out a 100 hour coincident data taking run. The run took place shortly after the reported optical observation of a 2 kHz pulsar in SN 1987a. The success of the run showed that long period coincidence runs are possible, maintaining the integrity of the data taken a high percentage of the time. The recorded data taken at both sites has been sent to Cardiff for full analysis. The purpose of this thesis is the analysis of the data obtained in Glasgow. In the rest of this chapter, I discuss the prototype in Glasgow in more detail, including the structure of the data produced during the run. I go on briefly to mention the Garching prototype, comparing it with the prototype in Glasgow and end by briefly discussing the practical implications of the 100 hour run.
4.1.1 The 10m Prototype.

The Glasgow prototype detector, shown in figure 4.1, consists of two perpendicular Fabry-Perot cavities, each comprising two mirrors optically contacted on to two test masses. These masses are suspended as pendula, by wires, this giving them a large degree of mechanical isolation. A beamsplitter is also suspended as a pendulum at the intersection of the lines through the two cavities. There are servo systems attached to each of the suspended masses within the system which allow their orientations to be controlled and would also serve to damp any unwanted low frequency oscillation of the masses. The light from an argon laser is polarised and then directed at the beamsplitter which transmits it equally into both cavities. The cavities and the beamsplitter are housed in a vacuum system maintained at a pressure of about $5 \times 10^{-4}$mbar in order to try to minimise the effect on the masses of the pressure due to any residual gas particles. The beams are bounced up and down the length of the cavities over 1000 times along the same lines without significant loss of energy because the mirrors are designed in such a way as to minimise the amount of light energy absorbed at each reflection. This dramatically increases the length of the light path and multiplies any relative displacements between the end masses by the number of bounces making any incoming gravitation radiation easier to detect. The light leaks from the cavities where it is then recombined to form an interference pattern which can be analysed by the use of a photodetector.

The laser beam is frequency locked to the primary cavity, —ie the cavity perpendicular to the direction of the laser— to keep the cavity resonating by ensuring that the length of the cavity is always a whole number of wavelengths of the light. In practical terms, as any change in the relative displacements of the masses within the cavity, due to impinging gravitation radiation or otherwise, is detected the information is fed back to the laser which alters its frequency accordingly. The signal transmitted by the feedback circuit is recorded as one of the relevant data streams and is called the primary error point signal.

The secondary cavity is locked to the laser in such a way as to maintain resonance within the cavity by altering its length when the frequency of the light changes. A signal measuring the response of the photodiode is continually fed back to the magnetic servo drive at the end of the cavity and as the beams go out of resonance the drive accordingly alters the length of the cavity. The signal transmitted to this feedback circuit is called the secondary feedback signal. Another data stream associated with this signal contains the secondary error point signal, that is searched for gravitational waves in this thesis.

The whole process, by which the laser is locked to the primary cavity and the length of the secondary is locked to the laser, serves to double the effect that any incoming gravitational wave
Figure A.3: Diagram showing the main optical components of the Glasgow prototype detector.

Figure 4.1: Taken from Hough et al (1989)
may have on the system. This is due to the influence that the wave would have upon the lengths of the cavities. As the primary cavity expands the secondary contracts and vice versa, thus as the laser wavelength changed to accommodate the change in the primary cavity the signal transmitted to the drive at the end of the secondary would alter its length to follow the laser, in effect to cause the secondary to follow the contraction or expansion of the primary in opposition to the influence of the gravitational wave. So if at some point the change in the length of the primary cavity due to an incoming wave was \( \approx x \), then the change in the length of the secondary should be \( \approx -x \) but the drive would compel it to be \( \approx x \) hence changing the length of the cavity by \( \approx 2x \) ie double the effect of the gravitational wave.

4.1.2 Data taken by the Glasgow prototype

During the 100 hour coincident data taking run, mentioned above, sensors tracking various aspects of the detector and its surroundings produced many streams of data. These channels were interleaved producing cycles of 10 bytes, each cycle being sampled at 10 kHz. The analogue channels comprise the secondary error point data, which is the stream upon which the search for gravitational wave events was actually carried out, and also substantially more housekeeping data, which is the output of sensors that monitored the actual running of the detector and its peripherals and also the detector's environment. There is also one digital byte of housekeeping.

Secondary error point data.

This signal is the combination of two 12 bit signals each sampled at 10 kHz symmetrically in time to give an overall sampling rate of 20 kHz. This symmetry ensures that the time delay between the signals is the same in either direction. It is essentially just a stream of noise in which a gravitational wave may be lurking. As mentioned above this data is the feedback signal that adjusts the length of the secondary cavity and as such is likely to pick up noise from in and around the detector. It was hoped that the noise would be both white and Gaussian, however with so many potential noise sources it is inevitable that there would be a number of periods of bad noise, ie stretches where the noise statistics could not be confidently predicted.
CHAPTER 4. THE GLASGOW PROTOTYPE DETECTOR

Non white noise was produced both because the feedback system from which the secondary error point data was taken has a non-uniform frequency and also because filters were applied to suppress seismic noise at low frequencies and to cope with aliasing at high frequencies. To compensate for these factors, calibration signals were applied periodically throughout the data taking run. This was achieved by applying a variation on the square wave function for 1.6384 sec every 210 sec. The data taken during this application should be deemed to be bad. These signals appear in frequency space as the peaks of a comb at odd multiples of 234.375 Hz with the first comb at 234.375 Hz representing a displacement of $1.19 \times 10^{-13}$ m and all the others a displacement of $1.19 \times 10^{-16}$ m. Additional calibration combs were also applied after periods when the laser was out of lock allowing the data to be re-calibrated when lock had been re-established and the system was thought to be working properly. In that the combs represented real displacements of known amplitude, their application provided the means of attaching a real amplitude to any threshold crossing event picked up in the analysis. This is described in Chapter 5.

Housekeeping data

The housekeeping data consists of a number of streams taken from various sensors in and around the detector, the most notable of which, ie those considered in this thesis, I briefly describe below.

- The Microphone signal is the combined signal from 3 microphones, one at each end mass and the third at the center. It is an 8 bit signal sampled at 10 KHz.

- The Secondary feedback signal is obtained from the same cavity feedback coils that produce the secondary error point signal. Because of the feedback systems bandwidth it contains information below about 1kHz only. Normally this stream would be searched for low frequency gravitational waves, but not in this analysis as the calibration combs bring out the low frequency regions in the error point signal.

- The Primary cavity error point signal is input to the cavity feedback coil in the primary cavity. It is also a 12 bit signal sampled at 10kHz.

- The Multiplexed signal is another 12 bit signal sampled at 10 kHz, this signal comprises 6 slow signals multiplexed together. This means only one of the slow signals is sampled during each cycle and only after 6 cycles is the same signal sampled again, giving each of these signals an effective sampling rate of 1667 Hz. Of these 6 streams only 2 are analysed in this work.

  1. The Seismometer signal is taken from 4 seismometers in the system.

  2. The Visibility signals are slow signals monitoring the extent of the resonance in both the cavities. They contain no gravitational wave information. They may show the laser going
out of lock before it becomes very apparent in any of the other housekeeping streams.
The secondary visibility signal only was examined in this analysis.

- Within the digital byte, each bit is given a different interpretation, either being set OFF or ON. For example one of the bits is set ON only when a calibration signal is applied. Another is set ON only at the start of every minute enabling the time to be updated continually whilst reading the data.

**Data format**

The data was written by an *Exabyte* machine to tape in files of blocks, each block containing 32768 bytes of information. This means that each block contains 3276 cycles, each of 10 bytes with the last 8 bytes not containing data. The last 4 bytes of each block were used as block count. This means that each block contains 6552 points of secondary error point data, 3276 points of all those streams sampled at 10000Hz and 3276/6 = 546 points of each of the multiplexed channels. Preceding the first data block on any of the tapes is a *filemark* and also 5 blocks of *header* information. A single filemark acts as a tag identifying the tape as one containing data at some point after the filemark. The header blocks contain, in the first block, an information text file and in the second, time and data information that can be used as a reference throughout the tape. The last 3 header blocks are empty. Should the tape contain more than one file of data then any file after the first will begin with a single filemark and then 5 header blocks. Two consecutive filemarks indicate the end of the tape. Further information about the data channels as well as the structuring of the cycles and the header blocks is given in Chapter 5.

### 4.1.3 The Garching prototype detector.

The Max-Planck-Institut für Astrophysik began work on laser interferometers in 1974, when a prototype delay line interferometer with arms of length 3m was constructed. The knowledge gained in the development of this prototype was put to good use when the 30m detector was commissioned in 1983, and built thereafter. The detector is shown diagrammatically in figure 4.2.

The major structural differences between the Glasgow and Garching detectors are mainly due to the differences between delay line and Fabry-Perot interferometers in general. In a delay line detector the light is bounced up and down the cavity along spatially *separate* paths. This necessitates bigger mirrors and also restricts the number of bounces within the cavity.

Against these disadvantages, delay line interferometers are far less complex to operate in that they do not need such precise control of the laser frequency and the arm length to maintain resonance
Figure A.1: Schematic diagram of the Garching 30 metre prototype. The light from an argon ion laser is sent to the interferometer via a monomode glass fibre (MF). The Pockels cells P1, P2 serve for modulating the optical path as well as for maintaining the proper point of operation. The frequency of the light is stabilised in two stages: firstly with respect to a reference cavity, and then to the full path length of the interferometer, monitored by photodiode D2.

Figure 4.2: Taken from Hough et al (1989)
in the cavity, they just need to maintain control over the relative phases of the light coming from the cavities.

The Garching data was sampled at 10 kHz, only half the frequency of the Glasgow data and there was also far less housekeeping data taken. For these reasons there was far less data taken in Garching in total and it was written to standard nine track tape rather than to video tape via the Exabyte machine.

4.1.4 The 100 hour run.

As mentioned in the introduction the 100 hour run demonstrated that both detectors could be run over sustained periods, taking information reliably a large portion of the time. The percentage of the total time when the arms are in lock, is called the duty cycle. During the runs, the duty cycle was a high 80% for the Glasgow detector and an extremely high 99% for the Garching detector. Both detectors were observed to measure strains, i.e. displacement noise level divided by arm length, of \( \approx 10^{-19}/\sqrt{\text{Hz}} \) during the run, at the best sensitivity between 1500Hz and 2000Hz.
Chapter 5

DESCRIPTION OF SOFTWARE

5.1 Introduction

The software was written for the purpose of analysing the data produced by the Glasgow prototype gravitational wave detector in March 1989. The data was written on 28 Sony video cassettes. It was read by an Ezabyte-CTS tape drive, (hence forth referred to as Drive0) and the results of the analysis were written to a similar drive (Drive1). The data was read directly from the tapes in Drive0 onto a board of 5 transputers housed in the Compaq 386/20 PC, an IBM compatible operating under MSDOS. The software was written in a version of Parallel FORTRAN utilising the 3L Parallel FORTRAN Compiler, it was divided into five distinct parts, one running on each of the transputers. These routines ran independently and communicated by passing data back and forth to each other when necessary.

The analysis then was carried out in parallel on the 5 transputers. The overall structure was a 3 tier pipeline. This meant that 3 distinct groups of data would be at some stage of their analysis within the network at any one time. The Root transputer reads the raw data from the tapes in Drive0 and extracts various data streams from this raw data. The housekeeping data are held within the Root where the relevant statistics are calculated. The secondary error point data, on which the search for gravitational waves is actually carried out, is then passed to Slave1, the first of the slave transputers. Here the FFT of the secondary error point data is found and the data are calibrated, which serves to remove any frequency bias, and also go through a weighting procedure that optimises the data for the extraction of possible signals within. Simultaneously the root is reading and unpacking a second data group. From Slave1 the data are sent on to the three remaining transputers, Slave2, Slave3 and Slave4, where specific analysis is carried out. Slave2 conducts a search for wideband burst events such as supernovae, while Slave3 and Slave4 both employ matched filtering techniques to attempt to
locate the characteristic chirp waveform that is thought to be emitted by coalescing binaries in the last few moments, before coalescence. Again these analysis routines were carried out simultaneously with those in the Root and Slave1 dealing with 2 other distinct data groups.

In this Chapter I describe all aspects of the software construction, firstly giving a brief description of the transputer and the ideas behind parallel processing, followed by an overview of the software design and then a more detailed description of all the routines.

5.1.1 The Transputer

Parallel processing

Over the last 40 years or so, the vast majority of the work done in the development of computing systems has been towards sequential processing ie the development of faster and faster processing chips that would carry out whole programming tasks by themselves.

In recent years however a lot of the attention has shifted towards the design and development of parallel processing systems, ie networks that split long processes up into smaller tasks and then farm them out to a number of smaller/slower/cheaper processors that are connected in parallel in such a way that data transfer between different processors is possible.

The best known parallel processor is the human brain in which neurons and synapses take the place of processing chips and wire links. Consideration of this gives some idea of the relative advantages and disadvantages of parallel processing. The brain is excellent at pattern recognition as is obvious from every day life, immediately recognizing objects and faces from visual data supplied by the eyes. It has however, proved very difficult to design and implement a mechanical system, of one or several processors, that in receiving the same kind of input from an optical scanner has anything like the pattern recognizing ability of the brain. Against this the brain is very poor at simple arithmetic calculations even when compared only to a single processing chip.

Single processing chips are not going to be able to be made faster forever. At some point, in the fairly near future, quantum effects will become insurmountable and the fastest possible chip will have been made. So in the long term, improved computational performance will only be achievable through the development of parallel systems. This is part of the reason why this prototype data analysis system was designed in parallel. There is no prohibitive reason why the analysis could not have been carried out by a fast single processor, however a parallel design was chosen, partly because the analysis lends itself to parallelism and partly because, in the long term, the final system is almost certain to be some kind of parallel system and its design could benefit from experience gained now.
CHAPTER 5. DESCRIPTION OF SOFTWARE

Transputers

Transputers are manufactured by Inmos, they are basically single processing chips with a small amount of fast on-chip memory as well as on-chip input-output processors. They are connected together on a circuit board, each being individually associated with blocks of, rather more, external memory held elsewhere on the board. We used five T800 transputers, each having 4 kB of fast on chip memory, four with 1 MB of external memory and the fifth, the Root transputer, with 4 MB of external memory. The board used was the TBX05 Transputer Module Motherboard manufactured by Systems West.

Each transputer has four links that can be used to connect it with other transputers or other devices. Each link is a two way connection enabling information to be passed to and from the transputer at a rate of 20 Mbits/s. Transputers in a network can only be connected via these links, numbered 0, 1, 2 and 3, in certain preset ways. They form a pipeline with link 2 of any transputer being connected with link 1 of the following transputer and links 3 and 0 being connected with links 3 and 0 with either the last-but-one or next-but-one transputer. The 0 link in the Root transputer is always connected to the host processor. In our network the pipeline was circular which means the final transputer in the network is connected to the Root. The links and connections, both used and unused, are shown in Figure 5.1. The Root transputer is the only hardware link between the host processor and the network and therefore the only way off getting data in to the network is via the Root transputer.

The transputers were found to be able to operate at .455 Mflops and .5 Mflops for floating point multiplications and additions respectively and at 1.11 Mflops for both integer multiplication and addition. These figures, although somewhat lower than advertised by the manufacturers, mean that the transputers are still marginally faster than the 386/387 chip housed in the Compaq. Since FFT's play a large part in this work, Figure 5.2 shows the time taken for the $3N \log_2(N)$ operations involved in finding FFT's of various data streams of length $N$.

The Exabyte-CTS tape drive

The transputer read data directly from, and write results directly to two Exabyte drives. The drive's full title is the Exabyte EXB-8200 cartridge tape subsystem. It is a "high performance, high capacity" device that utilizes "helical scan technology" to write data to an 8 mm tape with a very high recording density and hence a very large data storage capacity. As mentioned earlier, the tapes used are the easily available V8 video cassettes and, depending on the means of writing, up to 2.2 Gbytes of information can be written to each cassette. The drives are connected to the Compaq by a Datarace
FIG 5.1

ROOT

SLAVE 1

SLAVE 2

SLAVE 3

SLAVE 4

LINKS USED

LINKS NOT USED
**CHAPTER 5. DESCRIPTION OF SOFTWARE**

<table>
<thead>
<tr>
<th>No of points</th>
<th>Time of FFT (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>0.007</td>
</tr>
<tr>
<td>256</td>
<td>0.015</td>
</tr>
<tr>
<td>512</td>
<td>0.030</td>
</tr>
<tr>
<td>1024</td>
<td>0.064</td>
</tr>
<tr>
<td>2048</td>
<td>0.1375</td>
</tr>
<tr>
<td>4096</td>
<td>0.300</td>
</tr>
<tr>
<td>8192</td>
<td>0.600</td>
</tr>
<tr>
<td>16384</td>
<td>1.333</td>
</tr>
<tr>
<td>32768</td>
<td>2.733</td>
</tr>
<tr>
<td>65536</td>
<td>5.800</td>
</tr>
</tbody>
</table>

Figure 5.2: Transputer benchmarking using FFT's.

_SCSI_ (Small Computer Systems Interface) card housed in the back of the PC.

### 5.1.2 Overview

The Glasgow prototype detector was not thought to be sensitive enough for there to have been any realistic chance of finding any real events. For this reason, hardware constraints affecting the overall time of the analysis were allowed to become the main priority in the software construction rather than, for example, the number of filters used in the cross-correlation.

In parallelising the analysis procedure it was first necessary to consider it sequentially, to assess how best to divide it into parallel tasks. The adopted criterion had to take into account the available memory within each of the transputers and also, more importantly the time taken for each of the procedures in each of the transputers. Obviously for the sake of time efficiency, the duration of each of the tasks, in each of the transputers should be roughly the same and also the programs should be written in such a way as to minimise the amount of time that any of the transputers stands idle whilst waiting to either send or receive data from any other transputer. This was achieved by basically writing the program in serial and then testing each part separately for the time taken and other considerations. Figure 5.3 shows the coarse sequential break down of the program and also how it may be divided for parallelisation.

Bench marking routines established that the most time consuming single operation was the reading and unpacking of the data from the tape. This could only be done in the Root as only this transputer can read directly from Drive0 and also access the subroutine library that is used to
BUILD UP A TEMPLATE OF FILTERS (FOR MATCHED FILTERING) IN ARRAY.

| READ RAW DATA FROM TAPE IN DRIVE O. |
| EXTRACT THE VARIOUS DATA STREAMS. |
| COMPILE HOUSEKEEPING STATISTICS. |

----------------------------------------
FIND FFT OF SECONDARY ERROR POINT DATA.

| CALIBRATE THE DATA. |
| DIVIDE THROUGH BY TRANSFER FUNCTION. |

----------------------------------------
FIND THE CROSS-CORRELATION OF THE DATA WITH EACH FILTER IN TURN.

| CONDUCT AN EVENT SEARCH. |
| BUILD UP A RESULTS ARRAY. |

----------------------------------------
FIND THE IFFT OF THE DATA.

| COMPILE DATA STATISTICS. |
| CONDUCT AN EVENT SEARCH. |
| BUILD UP A RESULTS ARRAY. |

----------------------------------------
COMBINE RESULTS.

| OUTPUT RESULTS TO TAPE. |
| READ NEXT RAW DATA SET. |

FIGURE 5.3
unpack the raw data. The volume of housekeeping data concerned was so great that it could only be held in the memory of the Root which has four times the memory of any of the other transputers, hence its statistical analysis was also confined to the Root. The duration of these processes dictate the minimum overall analysis time for the data and hence, for time efficiency, must be the only processes running on the Root.

The analysis required one FFT and also one IFFT for every group of secondary error point data. These groups each contained 32768 elements therefore, as shown in Figure 5.2, each FFT and IFFT took 2.733 seconds, making them relatively time consuming; hence it made sense to place them in different transputers. The total time taken for the FFT, together with the times for the calibration and weighting procedures that must follow the FFT, was found to be only a little less than the total time for the processes conducted in the Root, hence it was logical to place these processes exclusively in the first Slave transputer. This would receive the data from the Root and carry out its own functions as the Root read and unpacked more data.

When Slave1 completes its task the true analysis procedures begin. The wideband event search was basically just a threshold crossing time series search and hence required an IFFT of each whole 32768 point data stream, before the search could be carried out. This IFFT followed by the event search and then the accumulation of time series data statistics took just enough time to warrant sole occupation of Slave2.

The last two transputers were then left to conduct the search for chirps in the data. This search involved cross-correlating the secondary error point data with filters that were built up into a template at the initiation of the program and held in memory. The filters were stored in frequency space each being only 4096 points in length (representing only 0-1250 Hz), their construction is described later in this Chapter. The cross-correlation for each of these filters was calculated by multiplying the filter with the first 4096 points of the FFT of the secondary error point data supplied by Slave1, and then taking the IFFT of the resulting 4096 point array. A time series event search was then carried out. The total time for these processes determined how many cross-correlations could be done in total for each data stream. Obviously as the minimum total time taken for the analysis was dependant on the time taken for the procedure in the Root transputer, the duration of the tasks in both Slave3 and Slave4 were not allowed to exceed this time. Hence only 20 filters were used in total.

The analysis conducted in Slave2, Slave3 and Slave4, is simultaneous with another stretch of data being processed in Slave1 and a third in the Root. Following these event search procedures the results are passed back to the Root which writes them to the results tape in Drive1 before reading
a fourth data group from the tape in Drive0.

A diagrammatic overview of the parallelisation and pipelining is given in Figure 5.4.

5.2 Comunication within the network.

All communications between transputers within the program are controlled by a small number of Parallel FORTRAN commands. These commands refer to addresses, each address being associated with one transputer. These connections and addresses are defined within the Configuration (.CFG) file.

The Configuration file

This file describes both the hardware connections between the transputers and also the software connections that are used in the main body of the program.

In Figure 5.5 taken from the beginning of the .CFG file, the processors are declared and given names. The host is the 386 chip within the Compaq that houses the board. The root is the Root transputer and similarly slave1 is Slave1 etc. The wire command defines the physical hardwire connection between the named transputers. The numbers in [ ] are the link numbers, from 0-3, and are defined as shown in Figure 5.1.

The next stage, shown in Figure 5.6, defines all the software tasks that run separately in the system. For each task the total number of connections to other tasks must also be defined. Each connection enabling data to be received is summed under ins, those through which data is sent, summed under outs.

The place command associates each of the tasks with its intended processor. The aserver task runs on the host processor. Upon execution it loads and then executes the other tasks on the appropriate transputers as defined by the place commands. Throughout the program run it acts as the software interface between the transputer board and the host system, performing any MSDOS functions that may be necessary, such as accessing a file stored on the hard disc of the PC.

The connect commands define all the data transfer connections between each of the tasks. A separate connect command has to be written for each data transfer direction, i.e the last connect defines the connection that allows data to be transferred from task filt5 to task filt4. The numbers in [ ] are again the link numbers used. For each task the number of links used must not exceed the number given in the task command. All the software connections must lie on one of the hardware connections defined at the start of the file. The hardware connections are bi-directional i.e they support data transport either way, hence every hardware connection supports two software
**HARDWARE**

processor host
processor root
wire ? host[0] root[0]
processor slave1
processor slave2
processor slave3
processor slave4

**FIGURE 5.5**

connections.

Communication within Parallel FORTRAN

A full list of the commands within parallel FORTRAN that can expedite communications between separate tasks is given in the manual Parallel FORTRAN written by 3L Ltd that describes their compiler along with the communication commands specific to their version of Parallel FORTRAN. The commands are contained within several groups of files called *packages*, with each package of files serving a different general purpose. The files within any package are declared for use within any task program by the use of the INCLUDE command at the start of each of the programs. The general form of the command is INCLUDE 'package name.INC' The only package used in this work is the CHAN package and so the command line at the start of each task is,

```
INCLUDE 'CHAN.INC'
```

Also towards the start of each of the tasks, before any data is transferred within the network, the *channel addresses* must be defined in accordance with the **connect** commands within the .CFG
****SOFTWARE****

task afserver ins=1 outs=1

task root ins=5 outs=5

task secd1 ins=2 outs=2

task secd2 ins=3 outs=3

task filt4 ins=2 outs=2

task filt5 ins=2 outs=2

place afserver host

place root root

place secd1 slave1

place secd2 slave2

place filt4 slave3

place filt5 slave4

connect ? root[1] afserver[0]

connect ? afserver[0] root[1]

connect ? root[2] secd1[0]

connect ? secd1[0] root[2]

connect ? secd1[1] secd2[0]

connect ? secd2[0] secd1[1]


connect ? secd2[2] filt4[0]

connect ? filt4[0] secd2[2]


connect ? filt4[1] filt5[0]

connect ? filt5[0] filt4[1]

FIGURE 5.6
file. These are integer quantities and have the following form,

\[ \text{INCHAN} = \text{F77\_CHAN\_IN\_PORT} (2) \]
\[ \text{OUTCHAN} = \text{F77\_CHAN\_OUT\_PORT} (2) \]

The number in the brackets is that referred to in the connect commands listed in the .CFG file given above. INCHAN refers to the software channel in task Root that receives data from task Secd1 and OUTCHAN, (initially defined as an integer), refers to the channel in task Root that sends data to task Secd1.

To send or receive one four byte WORD, (ie any single precision real or integer), the commands are of form,

\[ \text{CALL F77\_CHAN\_IN\_WORD (WORD, INCHAN)} \]
\[ \text{CALL F77\_CHAN\_OUT\_WORD (WORD, OUTCHAN)} \]

The top line asks for a four byte WORD from the address defined by INCHAN, and the bottom line tries to send WORD to the address defined by OUTCHAN.

The format is similar for longer stretches of data called MESSAGES. Here the length in bytes of the message has also to be given. So the commands to give or receive the message DATA of length LENGTH bytes are,

\[ \text{CALL F77\_CHAN\_IN\_MESSAGE (LENGTH, DATA, INCHAN)} \]
\[ \text{CALL F77\_CHAN\_OUT\_MESSAGE (LENGTH, DATA, OUTCHAN)} \]

These are the only data transfer commands used.

5.3 ROOT.F77

Introduction

This program operates within the Root. Its main purpose to act as a liason between the transputers that actually conduct the gravitational wave search and the outside world which, in this instance,
was the tape drives, the hard disc of the PC and the monitor screen. It reads data from the Glasgow tapes in Drive0, it then unpacks the data i.e splits the raw data up into its various constituent streams and sends the Secondary error point data on into the rest of transputer network for analysis. It compiles and stores statistics calculated from the various Housekeeping streams which are then merged with the received results from the other transputers, which it then outputs to a results tape in Drive1. It's designed to incorporate these features within a 3 tier structure which means that all the housekeeping data associated with 3 separate groups has to be contained within the memory of the Root at any one time. For this reason the Root has 4MBytes of memory, four times the memory of any of the others transputers. Also the results from any one group were returned to this task after the following but one group had been read from the tape so the passage of each of the groups through Root had to be monitored, with the commands to receive data from the other tasks positioned within Root so as to optimise the transfer of data. Below I describe the main features of the routine, additional details can be found in the program listing. This task along with the others is listed in full in Appendix 3. The lines in this Appendix are numbered and I refer to them occasionally in certain parts of the rest of this Chapter.

5.3.1 Program details.

Communication with the tape drive and extraction of data streams

5.3.2 Software

J.R.Shuttleworth (Dept of Physics, UWCC) wrote a file server using Parallel FORTRAN source codes, supplied on request by 3L and partially based on software written by N.L.Mackenzie (Dept of Physics, Glasgow University) that enabled data to be read from and written to the two CTS Drives by commands contained in software running on the Root transputer. These routines are held in the precompiled subroutine library EXLBNGTG.BIN. When reading from Drive0, data is transferred in blocks of 32 Kbytes, stored momentarily in the memory of the Compaq and then passed via the hardwire link into the memory of the Root transputer. Here the raw data is unpacked into its various constituents data streams by calling a number of subroutines written in C, again by Shuttleworth to some extent based on routines written by N.L.Mackenzie.

Once unpacked the data is passed around the transputer network and analysed by routines written in the 3L version of Parallel FORTRAN. Parallel FORTRAN is basically FORTRAN 77 with additional commands to expedite the transfer of data between transputers. The software was written for each transputer separately and contained information about the links to other transputers within its structure. Each .F77 source code is individually compiled to a .BIN object file and then
CHAPTER 5. DESCRIPTION OF SOFTWARE

linked to a .B4 executable file. The configuration (.CFG) file describes the various hardware and software connections and under the CONFIG command, produces an application (.APP) file.

The application file is an executable file run in the host processor. When executed within the host, it installs and executes the .B4 files in the workspaces of the appropriate transputers. All 5 transputers run their .B4 files simultaneously, in parallel, with the only periods of inactivity being due to any one transputer having to wait to give or receive data from any other.

At the execution of this program commands are sent that initialize the tape drives, carry out diagnostic procedures and make ready Drive0 to read the tape. These commands are sent via routines contained in EXLBNOTG.BIN, called within the program.

The first routine called is SETUP0K,

CALL SETUP0K (GOOD)

The logical variable GOOD returned is set to .TRUE. if the SCSI bus is operating properly and .FALSE. if not. TAPE0K returns .GOOD. set to .TRUE. only after successfully confirming that there is a tape in the drive, loading the tape, moving over the first filemark and setting the data transfer size to 32 kbytes. Its written in the form,

CALL TAPE0K (0,GOOD)

where 0 (or 1) is the drive number. The last routine needed before the analysis can begin is HEADER which attempts to read the header blocks, i.e. the first five blocks after the initial filemark.

CALL HEADER (0,INDATA,GOOD,EOTAPE,SVHEAD,'NULL',ITIME)

The only returned arguments relevant to the task, are the logical variable GOOD, set to .TRUE. if the routine is successful and the 6 element array ITIME into which the date and time at the start of the experiment are loaded. Other routines sometimes called are listed below. See program listing for more details.

CALL SPACE(0,NTMIN,NUMOUT,RES,FILMK) ! Attempts to space over NTMIN ! blks.
CALL WTMARK (1,GOOD) ! Attempts to write filemark on 1.
CALL RWOUND (1,GOOD) ! Attempts to rewind tape in 1.
CALL EJECT (0,GOOD) ! Attempts to eject tape in 0.

As explained in Chapter 4, the various data streams produced during the detector run were interleaved into one data stream and written to exabyte tape. The routine RDTAPE used within this
task to read this stream from the tape, straight into the memory of the Root is called in the following way,

    CALL RDTAPE(0, INDATA, NBLK, GOOD, FILMRK).

Here 0 is the drive number, INDATA the array to which the data are read, and NBLK the number of the block on the tape used. GOOD and FILMRK are logical variables that are returned with GOOD set to .TRUE. and FILMRK set to .FALSE. if the routine is successful. INDATA is a stream of 1 byte digits which can only be read within Parallel FORTRAN as a character array and therefore INDATA is an array of 32768 1 byte character elements.

Almost all the remaining routines used all take INDATA as an argument and return one of the signals interleaved within INDATA.

    CALL GTSERR(INDATA, SECERR)   ! secondary error point data 6552 pts
    CALL GTMICR(INDATA, MICRO)    ! microphone signal 3276 pts
    CALL GTPERR(INDATA, PRIM)     ! primary error point data 3276 pts
    CALL GTSEIS(INDATA, SEIS)     ! seismic signal 512 pts
    CALL GTSVIS(INDATA, SECV)     ! secondary visibility signal 512 pts
    CALL GTSFED(INDATA, SFED)     ! secondary feedback signal 3276 pts
    CALL GTDIGT(INDATA, DIGITS)   ! digital data 3276 pts

The digital data is again a 1 byte character stream but the rest is all 4 byte reals.

There are two further logical functions used that when passed an element from the character array DIGITS, interpret it as string of digital bytes and extract certain bits from whose setting, on or off, the presence of a minute mark or a calibration comb can be inferred and, if necessary, looked for.

    IF (CALIB(DIGITS(I))) THEN
      WRITE(6,*) 'CALIBRATION COMB PRESENT IN DATA.'
    ENDIF

    IF (MINMRK(DIGITS(I))) THEN
      WRITE(6,*) 'MINUTE MARK PRESENT AT THIS POINT.'
    ENDIF

Finally when trying to write data to the tape in Drive1 the routine WTAPIE is used.
CALL WTTAPE(1,CHRES,GOOD)

CHRES is the 32768 element character array written to the tape in Drive1. GOOD is again set to .TRUE. if the routine is successful.

Timing within the analysis

The date and time at the start of the experiment are given in the header blocks in the array ITIME. This is not however the time at the start of the first actual data block so before the commencement of the actual analysis this time was found by moving into the real data and looking for the first minute mark. This was achieved by moving through the blocks, extracting the digital array and then giving the function MINMRK, each element of the digital array as an argument, one by one until MINMRK returned .TRUE..

The real data starts about 10 seconds after the header, so by using the time at the start of the experiment it was possible to judge roughly where the first minute mark occurs. Should this point be some way into the data then the search for the first minute mark is started after the start of the actual data but definitely before the first minute mark. This procedure is carried out between lines 238 and 296.

If the first minute mark is found in block NBLK (which is 0 at the start) and then at point JNBLK within NBLK, the time, $T_{minmrk}$, taken since the start of the actual data is,

$$T_{minmrk} = NBLK \times 0.3276 + JNBLK \times 0.0001.$$  

Each block contains 3276 digital points sampled at 10 kHz. One block therefore represents 0.3276 seconds and each element is 0.0001 second. The minute marks are set at the start of each minute when ITIME(6) and possibly ITIME(5) and ITIME(4) as well are updated to the start of the actual data with ITIME(6) given by,

$$ITIME(6) = 60 - T_{minmrk} = 60 - NBLK \times 0.3276 - JNBLK \times 0.0001.$$  

When the change of minute occurs after the header but before the start of the data proper, then the minute mark representing the next minute is found and the start then found accordingly.

The array ITIME is then updated for every subsequent minute mark found. There are 183.15 blocks in a minute’s data, so the search for minute marks in the data is not begun until 183 blocks after the start of the block that contained the previous mark. This minimised the time taken looking for the minute marks with an average of only 5 blocks in every 183 being searched over the whole.

Finding a minute mark where it should be allows confirmation of the timing within the analysis and allows reliably accurate times to be attached to each of the events. For each minute mark its
position is recorded in F1BLK, which is a real number that gives the position of the mark to the nearest 1000th of a block, ie to the nearest 0.0003276th of a second. If an event has a position EBLK, the time in seconds since the last minute mark is TEBLK which is given by,

\[ T_{TEBLK} = (EBLK - F1BLK) \times 0.3276. \]  (5.1)

This allows the time of the event in hours minutes and seconds to be found.

5.3.3 Main body of program.

Each analysis group consists of 32768 or \(2^{15}\) secondary error data points. This represents 5 blocks of 6552 points each plus 8 additional points set to zero to achieve the power of 2. The program is dominated by the contents of a loop, that reads data from the tape, builds up an analysis group and sends it on to the other tasks for analysis. In each group only the first 26208 points (or 4 blocks) are analysed hence in each loop only 4 new blocks are read and the final block from the previous group is placed at the start of the next. This means also that 5 blocks must be read initially so one is read before the start of the first loop. This is necessary to prevent the wraparound effect in the discrete correlations as described in Chapter 3.

With each group there is more than twice as much associated housekeeping data that must be stored and statistically analysed. When the results are returned to Root they are combined with the appropriate housekeeping data and written to the results tape in Drive1. In the period between a group being read from the tape in Drive0 and its results being written to tape in Drive1, two more groups are read from the tape in Drive0. This housekeeping data is also stored hence each group of housekeeping data must be tagged in someway to associate it with a particular results group. This is achieved by use of the three tier structure mentioned earlier.

Initially 1 block of data is read from the Glasgow results tape into the array INDATA. This array is then passed to the routines listed above that produce the Secondary error point signal as well as the housekeeping streams. The housekeeping is stored and the Secondary error point data is passed on to the next task in Slave1.

At this stage the major loop begins. In each loop 4 blocks are read from the tape. For each one the Secondary error point signal as well as the housekeeping streams are extracted. The Secondary error point data is sent to the next task and the housekeeping stored. The separate housekeeping arrays are stored by use of the equivalence statement in single large arrays. The equivalence statement when applied to 2 arrays compels them to occupy the same memory areas. This is utilised in this task to build the small separate housekeeping arrays produced for each block into single larger arrays. This is shown for the microphone signal in Figure 5.7.
EQUIVALENCE (MICROP(1),MICRO(1)),(MICROP(3277),MICRO1(1)),
+ (MICROP(6553),MICRO2(1)),(MICROP(9829),MICRO3(1)),
+ (MICROP(13105),MICRO4(1)),(MICROP(16381),MICRO5(1)),
+ (MICROP(19657),MICRO6(1)),(MICROP(22933),MICRO7(1)),
+ (MICROP(26209),MICRO8(1)),(MICROP(29485),MICRO9(1)),
+ (MICROP(32761),MICRO10(1)),(MICROP(36037),MICRO11(1))

FIGURE 5.7

MICROP is an array of 39312 elements and MICRO, MICRO1 etc are all of length 3276 elements. Initially, before the start of the loop, the routine GTMICR is supplied with INDATA from the first block and MICRO is returned. MICRO, although in itself only 3276 elements in length, immediately occupies the first 3276 elements of MICROP as well. Within the first loop the next 4 blocks are read and MICRO1, MICRO2, MICRO3 and MICRO4 are found. As can be seen above MICRO1 also occupies 3276 elements of MICROP from 3277 to 6552. Similarly for MICRO2, MICRO3 and MICRO4, the first 5 blocks worth of MICROP are now filled, up to element 16380. Analogously the same is true for the other housekeeping streams although the multiplexed arrays are 6 times shorter.

This structuring forms the basis of the 3 tier structure mentioned earlier because in the second loop MICRO5, MICRO6, MICRO7 and MICRO8 are found and these occupy elements 16381 to 29484 of MICROP. In the third loop MICRO9, MICRO10, MICRO11 and MICRO12 are found but only the first 3 are included in the equivalence statement and these occupy the remaining elements of MICROP. Again this is also true for the other housekeeping streams and therefore all the housekeeping data associated with 3 distinct groups of data is stored in several large arrays. Each of the tiers has an associated flag number, 1 for the first loop, 2 for the second and 3 for the third. The results data from the analysis of the secondary error point data taken in the first loop ie when the flag is 1, is returned at the end of the third loop when the flag is set at 3. This will always be the case thus when data is returned and the flag is set at 3 it is associated with the first third of the large housekeeping data arrays. Similarly when data is returned and the flag is set to 1 or 2 then it is associated with the second and third parts of the housekeeping arrays respectively.

At the end of the third tier when the results have been combined with the housekeeping data and output to tape the flag is then reset to 1 and the 13th small housekeeping array (MICRO12 in the case of the microphone signal) is written in a loop into the first blocks worth of the larger arrays (the first 3276 points of MICROP) becoming the first block of the next group. As the next 4 blocks
are read the housekeeping data extracted overwrites the larger arrays. More results are returned, combined with the appropriate housekeeping data and written to tape. The flag is reset from 1 to 2 and the next loop begins etc.

The blocks of Secondary error point data sent on to Slave1 have one extra point appended to them. This point is array element SECER1(6553) (or SECER2 or 3 or 4). Usually this extra point contains no useful information in which case it is set to 0, but it can tell the other tasks to look for calibration combs or to return statistics. It is also used to halt the tasks in the other transputers.

After the extraction of each block the digital arrays are searched for indications of the presence of calibration combs in the secondary error point data. If found SECER1 is given the value 1.

The analysis routines running the slaves produce histogram statistics of the data that the root writes to disc occasionally to prevent their complete loss should the program crash for some reason. The signal to return these statistics to the root is SECER1 equal to 2.

When the program has completed its allotted number of loops SECER1 is set to -1 which triggers the termination procedures in each of the slaves.

The housekeeping statistics are calculated after the last block from each group is read from the tape, and before the results from the previous but one group are retuned to the Root. The means and standard deviations for each of the relevant housekeeping streams are found except for the secondary visibility signal which acts as a long period diagnostic for the system for which reason only its maximum value for each group is found. When the event search results are returned to the Root, the deviation from the group mean for each of the housekeeping streams at those points are found and divided by the calculated standard deviation for that stream. The ratio's obtained can be examined to establish if any of the streams were possibly the cause of the event. The event information is combined with this information and output to the results tape in drive1. The structure of each results block is given in more detail in Appendix 2.

5.4 SECD1.F77

5.4.1 Introduction

This task is located in the first slave transputer, Slave1. It receives data in blocks from the root which it builds into groups of five blocks. The position in the 3 tier structure in task Root from which the data is sent is irrelevant in this task, as it treats all groups in the same way. Within this task several procedures are carried out that compensate for any ‘bad’ noise due to either nonwhite noise produced by the components of the system or frequency bias in the actual detector. These adjustments are
made in frequency space. Hence firstly the Discrete Fourier Transform, of 32768 points (5 blocks plus 8 points set to 0), is found through the use of two Fast Fourier Transform algorithms, REALFT and FOUR1, taken from Numerical Recipes by Press et al (1986).

The data was then calibrated according to combs that were periodically applied to the system and appear in the data. The calibration data is recalculated whenever a new comb is found. This calibration serves to remove any frequency bias within the data. The calibration routines are CALCOM and UPDCOM. The presence of a comb in the stream is noted in the first element of a element array SDT, which is essentially used just to pass useful numbers through the system.

The second element of SDT is the calculated mean of the current group of raw data before it is altered in any way.

After the calibrated data is returned to the main program it is immediately sent to the routine WEIGHT in which the extent of the influence of the bad noise in the data is reduced and also statistical data used to calculate event amplitudes is found and written to the elements 3 and 4 of the array SDT.

Finally the calibrated and weighted FFT data is sent on to the remaining 3 transputers for the actual event search procedures after which the routine asks for the next four blocks of time series data.

5.4.2 Main body of task

Initially it received 5 blocks from the task Root which it concatenated into a 32768 element array using equivalence statements (lines 8-10). This array was then the basic analysis group upon which this task performed several procedures after which it passed the altered group onto the remaining tasks running in the other transputers.

The fifth block initially received, was copied into an array which in its unaltered state became the first block of the second group which was then completed by the next 4 blocks received from the root (lines 165-166 and 210-211). The process is then repeated through all the data, the last block of the previous group becoming the first block of the next, regardless of the current position in the tier. No statistical analysis is actually carried out on the final block of any group hence by putting it at the start of the next group it is not analysed twice.

The 6553rd element of each of the blocks is examined for a relevant message. If it was found to be set to either -1 or 5 the instruction were to either terminate all the tasks or return histogram data respectively. In either case the same information is conveyed onwards to the rest of the tasks by setting point 32761 of the group to the same values at the end of the task. If the 6553rd element
TIME SERIES DATA CONTAINING COMB.

![Time series graph showing amplitude vs time (sec).](image1)

LOG(POWER SPECTRUM OF ABOVE)

![Power spectrum graph showing amplitude vs frequency (Hz).](image2)

FIGURE 5.8
is set to 1 then a new calibration comb has been found to start somewhere in that block. The very next block is then sent to the subroutine CALCOM where, as described below, the relevant arrays are updated. This is the first block that is sure to have a comb applied over all its total length as the comb is applied for 1.6384 seconds which is equivalent to 5 blocks.

After the whole group is formed, its FFT is found prior to it being passed to the subroutines described below. Following this the next four blocks are received from Root.

### 5.4.3 Calibration

There are two subroutines within secsl that are concerned with the calibration of the data. The first, CALCOM (lines 220-252), is only called every time the presence of a comb is picked up in the time series. The routine is sent a block of data over the whole of which a comb was known to have been applied. Only the first 4096 points of the block are used to provide the FFT from which the comb is extracted. The teeth of the comb are at fixed frequencies $f_n$ given by,

$$f_n = 234.375 + 468.75(n - 1), \quad n = 1, 2, ..., 21.$$

These frequencies fall exactly on points given in the discreetly sampled FFT. Hence the comb’s amplitudes can be exactly found. Apart from the first tooth of the comb, which is 1000 times larger, all the other teeth of the comb produce a displacement in the detector of $1.19 \times 10^{-16}$ m. Figure 5.8 shows a section of time series data with a comb applied in the central region and also the log of the power spectrum of the time series showing the combs.

For the comb an inverse comb is then calculated whose teeth have amplitudes such that when any of its teeth are multiplied by the equivalent tooth from the original comb, the original combs true displacement is produced. So if COMB(n) is the height of the n'th tooth, the equivalent tooth from the inverse comb, ICOMB(n) is given by

$$ICOMB(n) = \frac{1.19 \times 10^{-16}}{COMB(n)} \quad n = 2, 21 \quad (5.2)$$

$$ICOMB(n) = \frac{1.19 \times 10^{-19}}{COMB(n)} \quad n = 1. \quad (5.3)$$

The comb and its inverse comb are shown in Figure 5.9. The inverse comb is stored in the real array ICOMB. The points on the inverse comb between the calculated peaks are found by linear interpolation. To enable these points to be found the slopes of the lines between the combs are found and stored in the 20 element array SLOPE. The slope of the line between inverse combs ICOMB(n) and ICOMB(n + 1) is written to SLOPE(n) which is given by,

$$SLOPE(n) = (ICOMB(n + 1) - ICOMB(n))/768, \quad n = 1, 20. \quad (5.4)$$
FIGURE 5.9
In the above equation, 768 is used because it is the number of complex points between the combs when 16384 complex points are considered in total. These arrays are then passed back to the main program.

The second subroutine directly associated with the combs is \textsc{updcmb} (line 257-290). The whole 32768 data points of the Fourier transform of each group are sent to this subroutine along with the arrays \textsc{icomb} and \textsc{slope} that are used to calibrate the data. The calibration is achieved by multiplying each point in the data by the calculated height of the curve through the inverse comb. This height is calculated using \textsc{icomb} and \textsc{slope}. If \textsc{icomb}(n) is the height of inverse comb \(n\) and \textsc{slope}(n) the slope of the line between comb \(n\) and \(n+1\) then at any complex point \(j\) between the combs the heights \(\text{ih}_\text{real}(j)\) and \(\text{ih}_\text{imag}(j)\) are given by,

\[
\begin{align*}
\text{ih}_\text{real}(j) &= \text{icomb}(n) + (2j) \times \text{slope}(n), \quad \text{and} \\
\text{ih}_\text{imag}(j) &= \text{icomb}(n) + (2j - 1) \times \text{slope}(n), \quad j = 1, 2, \ldots, 384.
\end{align*}
\]

For the region before the first tooth, \textsc{ih} is set to \textsc{icomb}(1) and for the region after the last tooth, \textsc{ih} is set to \textsc{icomb}(21).

The data is calibrated by multiplying each complex point by the calculated height of the inverse comb at that point. This is done for all regions of data between two combs separately. Hence at some point \(j\) of complex heights \(\text{h}_\text{real}(j)\) and \(\text{h}_\text{imag}(j)\) between two combs where \(j = 0\) at the previous comb (or, if it is the start of the data at the first data point) the calibrated heights, \(\text{c}_\text{real}(j)\) and \(\text{c}_\text{imag}(j)\) are given by,

\[
\begin{align*}
\text{c}_\text{real}(j) &= \text{ih}_\text{real}(j) \times \text{h}_\text{real}(j), \quad \text{and} \\
\text{c}_\text{imag}(j) &= \text{ih}_\text{imag}(j) \times \text{h}_\text{imag}(j), \quad j = 1, 2, \ldots, 384.
\end{align*}
\]

Figure 5.10 shows the log of the power spectrum of a stretch of real data before and after calibration. It should be noted that with the frequency bias removed, the second plot gives a true representation of the displacement noise in the detector.

### 5.4.4 Weighting

It is shown in Chapter 3 that the optimum filter used for the extraction of a signal by cross-correlation techniques, \(\bar{q}(f)\), is given by,

\[
\bar{q}(f) = \frac{\tilde{h}(f)}{\tilde{S}(f)}.
\]

\(\tilde{S}(f)\) has to be calculated separately for each length of data. Rather than divide each filter in turn by \(\tilde{S}(f)\) before cross-correlation, the data itself is divided through by \(\tilde{S}(f)\). This optimises the data
for simple time series burst event searches as well as for Chirp searches because, just searching a data stream for points crossing some threshold is equivalent to cross-correlating the stream with a delta function acting as a filter. This procedure is carried out in the subroutine WEIGHT (lines 295-340).

\( S(f) \) is assumed to vary both slowly and smoothly, which allows the data to be split up into regions, with one value of \( S(f) \) calculated for each region. The 16384 complex data points are divided into 128 groups of 128 points. \( S(f) \) is then calculated for each of these groups in turn, approximated as the variance \( \sigma^2(f) \), of the data in each group. For any group whose complex elements are \( C_{ij} \) to \( C_{i128} \) where \( i = 1...128 \) and \( \bar{C} \) its mean, its variance \( \text{Var}(i) \) is given by,

\[
\text{Var}(i) = \frac{1}{128} \sum_{j=1}^{128} |C_{ij}|^2 - \bar{C}^2 \\
= \frac{1}{128} \sum_{j=1}^{128} \text{Re}[C_{ij}]^2 + \text{Im}[C_{ij}]^2 - \bar{C}^2. \tag{5.9}
\]

These 128 values of \( S(f) \) are then written to an array.

In a loop, every point is then replaced by its own value divided by the variance of the group in which it lies. Hence the value of \( C_{ij} \), with \( i \) and \( j \) defined as above, becomes,

\[
C_{ij} = \frac{C_{ij}}{\text{Var}(i)}.
\]

The effect of this is to optimise the filters, by reducing the influence of high noise at certain frequencies in the time series, after which the weighted data is then returned to the main routine.

The effect of this procedure can be seen in Figure 5.11 which shows calibrated data before and after this weighting procedure. Frequencies at which the noise is high (and hence undesirable) before weighting, show noise levels below the full spectrum mean afterwards. This graphically demonstrates the most serious potential drawback of this procedure, as the high noise levels tend to be at low frequencies, less than 1000Hz, which is also the region that any chirp present would occupy. Therefore in reducing the influence of the high noise, the influence of the chirp in the noise would also be diminished. Another potential problem could occur at any frequency, if some narrowband event stood out above the noise in frequency space to such an extent that it enhanced one or several values of \( S(f) \). The weighting procedure could then cause the strong signal to effectively cancel itself out. In Chapter 3 it is shown that the amplitude, \( A \), of any wave that may be found in the data is given by,

\[
A = \sigma(f) \left( \frac{S}{N} \right) \left( 2 \int_0^\infty |\bar{s}(f)|^2 \, df \right)^{-1/2}. \tag{5.11}
\]

Here \( \sigma \) is the standard deviation over the frequency range of the data over which the search was made.

There are two frequency ranges involved, 0-10000Hz \( ie \) the whole stream for the
LOG(POWER SPECTRUM AFTER CALIBRATION.)

LOG(POWER SPECTRUM AFTER DIVISION BY Sh(F))

FIGURE 5.11
broad band search and also 300-1000Hz for the chirp search as these are the frequencies over which the filters are defined. These values are calculated in this routine, before the weighting, and are returned to the main program where they take up the last 2 elements (3 and 4) of the array SDT.

5.4.5 SECD2

Introduction

This routine runs in the third transputer. It receives 2 data messages from the second transputer in the analysis of a group of data. The first of these is the 32768 points of the FFT of the optimised time series and is held in SECERR. The second is the four element array SDT containing data calculated prior to the \( \sum_{j=1}^{N} h_{i} \sigma \) operation. The 32761st point in the time series is used to pass certain instructions into the rest of the system. If point 32761 is set to -1 then the program is to be terminated and all results are returned to the root and the task halted. If point 32761 is set to 5555 then the current statistics histogram (explained below) is sent to the root.

The first 4096 points (representing 1250 Hz) are then transmitted to the next task in the next transputer along with an additional 4097 point which is normally set to 0 but when the termination or statistics return commands are set then point 4097 is set to -1 or 5555 respectively.

SECERR is then returned to its time series by finding its inverse fast Fourier transform using the same routines (REALFT and FOU1) as used in the previous task. The data are then sent to the statistics routine explained below and from there on to the event search routine also explained below.

Statistics

The STATS (lines 83-127) routine is passed the group of weighted and calibrated time series data and returns its mean and standard deviation \( \sigma \). It is also passed a 500 element array containing histogram data which is updated using data from the group and then returned to the main program along with a single number from which the extent to which the noise is Gaussian, can be assessed.

For each group the first 4 blocks or 26208 (= \( N \)) elements only are analysed. If an element is written as \( x_i \), then the mean of the noise \( \bar{x} \) is found along with the sum of the squares of each of the elements which gives the standard deviation by the following equation,

\[
\sigma = \left( \frac{1}{N} \sum_{i=1}^{N} x_i^2 - \bar{x}^2 \right)^{1/2}.
\] (5.12)

The histogram array is a 500 element array that contains a record of the number of elements at various displacements from the mean accumulated over all the groups on that tape preceding the current group.
The distribution of data is assumed to symmetrical about its mean and all the displacements are taken as amplitude (positive) displacements starting at the mean and going up to 10 (which represents a displacement of 10\(\sigma\)) in bins of width 1/50 (\(\sigma/50\)). Any points beyond 10 are counted in bin 500. The bin number \(i\) to which a particular element \(x_j\) should be added is given by,

\[
i = \text{int}(|x_j - \bar{x}| \times 50/\sigma),
\]

(5.13)

where int means "the integer part of". When this is established 1 is added to the number already counted in bin \(i\). As a safety precaution the array is requested occasionally by the main program in the root from where it is written to the hard disk of the PC. The shape of the binned data as it should appear is shown in Figure 5.12 which is a theoretically generated histogram.

**The Gaussian Parameter**

The numbers in the bins for each group separately, are used to provide a quick diagnostic test of the Gaussian extent of the noise in that group. If the 26208 points of noise considered in each group are Gaussian then the number of elements that *should* be in each of the bins can be found. The first 5 bins of displacements 0 to 1/10 (0 to \(\sigma/10\)) were considered. The number of points \(N_{0,1/10}\) that should occupy these bins was found as follows,

\[
N_{0,1/10} = A \int_{0}^{1/10} e^{-x^2/2} \, dx \quad \text{where} \quad A = 26208\sqrt{2/\pi}
\]

(5.14)

\[
= A \int_{0}^{1/10} \left(1 - \frac{x^2}{2} + \frac{x^4}{8} + \ldots\right) \, dx
\]

(5.15)

\[
= A \left[x - \frac{x^3}{6} + \frac{x^5}{40} + \ldots\right]_{0}^{1/10}
\]

(5.16)

\[
= 2088.
\]

(5.17)

By comparing this number with the number that were found to occupy the first five bins a measure of the extent to which the noise is Gaussian is possible. This can be quantified by defining the *Gaussian Parameter*, \(N_g\), by

\[
N_g = \frac{N_{0,1/10}}{2088}.
\]

(5.18)

The nearness of \(N_g\) to 1 can then be used as a guide to the *goodness* of the noise. It will be shown in the next Chapter where the results of the analysis are considered that this parameter is a good diagnostic. \(N_g\) is returned by the subroutine to the main task along with the updated histogram array as well as the mean and standard deviation of the group.
CHAPTER 5. DESCRIPTION OF SOFTWARE

Event search

The search for threshold crossing events takes place in the subroutine called SEARCH (lines 134-214). It is given a stretch of time series data, held in the array SEGMENT along with the standard deviation and the mean of the data calculated by the previous statistics subroutine. It is also sent the four element array SDT that contains data used to attach a real value to the amplitude of an event. It returns the 200 hundred element array EVENT containing the results of the event search of the current group.

A loop is set up that goes through each of the 26208 points one by one, the first operation in each group establishing whether or not that particular point’s absolute displacement from the data mean exceeds some preset multiple of the standard deviation. If it does then it is treated as an event.

The search is based around the possibility that events could be multiple, ie come in groups of two or more. No multiple events were seen in preliminary testing of the Glasgow data, however they were found to be quite common in the Garching data as well as the chirp search data from both data sets, hence it seemed reasonable to incorporate in the possibility of multiple events in the analysis. For this reason after a first event is found nothing is written to the array EVENT until the next point is also checked. If this point is also an event then its size is compared with the previous point and then again nothing is written to the results array until the next point is checked. When some point following a group of events is found not to be above threshold, then the results from the previous events are written to the results array. Associated with each set of events are 10 points in the results array. For this reason, with 200 points in total and the first 7 holding a mini header, data corresponding to only the first 19 events sets are written for each data group. This is reasonable on the basis of threshold chosen as it was found only non Gaussian noise actually produced this number of events. The multiple of the standard deviation that gives the event threshold was taken to be 4 as this was seen to produce sufficient events generally without exceeding the capacity of the results array. For perfectly Gaussian data one would expect only 2-3 events on the basis of this threshold for every 26208 points. But experimentation with the Glasgow data showed that on average more events were seen even in stretches of data made up from groups whose Gaussian assessment parameters were consistently very close to 1. For groups such as these, there were on average 5-7 events per group although there were also a significant number of groups that produced 15 or more events. For stretches of bad, non Gaussian data, 19 events were usually found long before the end of the group.

The first 7 points of EVENT are taken up with data pertaining to the whole 26208 point group. The first 3 of these points are written in this subroutine. They are, in order, the total number of events found, the groups calculated standard deviation and the groups calculated mean. The 4 other
points are written when the data is returned to the main body of the task. They consist of 3 of the elements contained in the array SDT and also the Gaussian parameter $N_\gamma$ returned by the statistics routine.

The first of the 10 element results strings starts at element 8 in EVENT. The first element of each string is its event group number, from 1 to 19, depending on the number of events previously found in the group. The next three elements give the position of the event within the group. These are numbers from 1 to 26208 and represent, in order, the position of the first threshold crossing event in the set, the position of the largest of the threshold crossing events and the position of the last of the events in the set. It is enough to use these positions only as the indicator of the position of the events in time as the time is calculated periodically throughout the analysis and the time from one of these points to an event can be calculated as the exact sampling frequency of the data is known. Should a point be an event independent of any other points then obviously all three of these position indicators will be the same.

The next 2 points written are concerned with the amplitude of the largest component of the multiple event. The first point is the ratio of its size to the standard deviation, ie the signal to noise, hence this is a number at least equal to four. The second point gives the calculated value for the actual amplitude of the gravitational wave that may have produced such an event. This was calculated using equation 5.11, where $\sigma(f)$ is given in SDT (4). As the filter is effectively a delta function, the integral in equation 5.11 was approximated as follows,

$$\int_0^\infty |\tilde{s}(f)|^2 \, df \approx \frac{10000}{16384} \sum_{i=1}^{16384} 1 = 1000, \quad (5.19)$$

where $10000/16384$ is the frequency spacing between each sampled point.

The last four of the ten elements allocated to each event set are left blank (set to 0). These are used in the root transputer where relevant housekeeping data is added. The event array is shown diagrammatically in Appendix 2.

The array EVENT is then returned to the root task along with the current histogram if requested and the task is ready to receive the next data group.

### 5.4.6 CORR1 and CORR2

**Introduction**

These routines ran simultaneously in both the fourth and fifth transputers respectively. They cross-correlated data received from task secdd2 with a suite of filters built up on the basis of the accepted mathematical model for the evolution of the gravitational radiation emitted from binary systems of
varying mass parameter and phase just prior to coalescence. This characteristic shape will subsequently be referred to as a chirp. Unlike the 2 previous tasks these were not written as a sequence of subroutines but as continuous programs. This was because this task was not obviously divisible into separate sections that would facilitate the operation of the processes in a time efficient manner. This is essentially due to the nature of the problem. The same extended process involving FFT’s, statistics and event searches had to be repeated in cross-correlating the data with each of the filters in turn. If these lesser tasks were each allocated to a subroutine then the analysis of each group of data would involve many data transfers between routines which would certainly slow the program up.

Basically, each group was cross-correlated with each pair of filters (both based on the same mass parameter) in turn. The results from all of these correlations were statistically analysed. This was followed by an event search, the results from which were added to an overall 200 element results array associated with that group. This array contained the results from all the cross-correlations with the filters in that transputer. These arrays from both transputers were then returned to the Root task. The various procedures employed, in the correct order, are explained below.

Commands to terminate the tasks or return histogram statistics are sent in point 4097 of the data. This point plays no part in the correlation. The array SDT was also received from secd2.

**Filters**

The suite of filters was built up at the start of each analysis run in both tasks. This takes place only while the Exabytes are undergoing their initiation procedure. Hence the program is not delayed by the filter formation. All the filters on each of the tasks are built into part of the same array called FILTERS. The filters are initially built up in the time domain as the chirp signals for particular mass parameters and phases. Each filter is 32768 points long with a relatively small calculated non zero part at the start and all other points equal to zero. The FFT of the filter is then found, of which the first 4096 points are written to the array FILTERS in the appropriate place. This represents the first 1250 Hz which includes all the significant power in the chirp transform. Ten filters are used in total in each task which means that FILTERS is \( 10 \times 4096 = 40960 \) elements long.

The filters are built using equations given in function statements defined at the start of the program (lines 13-16). These statements are based on equation 2.7 to equation 2.9 given in Chapter 2. The central statement, defining the chirp is

\[
\text{CHIRP}(\text{TIM}, \text{PAR}, \text{PHASE}) = A(\text{TIM}, \text{PAR}) \cdot \cos(B(\text{TIM}, \text{PAR}) + \text{PHASE} \cdot 1.570796)
\]

where TIM is the time in seconds starting at zero when the frequency of the chirp reached 100 Hz.
CHAPTER 5. DESCRIPTION OF SOFTWARE

PHASE is the phase of the wave at 100 Hz and PAR the mass parameter defined by,

\[ \text{PAR} = (\text{reduced mass})(\text{total mass})^{2/3} = \frac{M_1 M_2}{(M_1 + M_2)^{1/3}}, \]  \hspace{1cm} (5.20)

where \( M_1 \) and \( M_2 \) are the masses of the components of the binary system in units of solar mass. \( A(TIM, \text{PAR}) \) is defined by the statement,

\[ A(TIM, \text{PAR}) = (1 - 0.34 \times TIM \times \text{PAR})^{**(-.25)} \]

and \( B(T/M, \text{PAR}) \) by,

\[ B(TIM, \text{PAR}) = (1005.31 \times (1 - A(TIM, \text{PAR})^{**(-5/2)}) / (0.34 \times \text{PAR})) \]

Thus for different masses and phases the chirp signal at different times after 100 Hz can be calculated. The frequency range considered here is from 300 Hz to 1000 Hz. It was hence necessary to know the time \( t_{300} \) for each mass parameter at which the signal reached 300 Hz and also the time taken for the frequency of the wave to increase to 1000 Hz. From equation 2.8, the time \( t_f \) taken for the chirp to reach a frequency \( f \) with \( f \geq 100 \text{Hz} \) is given by,

\[ t_f = \frac{[1 - (f/100)^{-8/3}]}{0.34 \times \text{PAR}}. \]  \hspace{1cm} (5.21)

Hence the time taken to reach 300 Hz is \( t_{300} \) given by,

\[ t_{300} = \frac{[1 - 3^{-8/3}]}{0.34 \times \text{PAR}} \]  \hspace{1cm} (5.22)

and the time taken, \( TD(\text{PAR}) \) for the frequency to go from 300 Hz to 1000 Hz is the fourth function statement and is given from equation 2.9,

\[ TD(\text{PAR}) = (3.**(-8/3)-10.**(-8/3))/(.34*\text{PAR}) \]

The sampling rate of the data cross-correlated with the filters was 20 kHz hence this must also be the sampling rate at which each of the filters is formed. So the minimum number of points, \( NPTS \), that must be calculated in a loop increasing with time by 0.0005 sec each loop and starting at \( t_{300} \) is given by,

\[ NPTS = \text{INT}(TD(\text{PAR}) \times 20000.) + 1 \]

The 10 filters in each task, have 5 different mass parameters and 2 phases, with the mass parameters representing the chirp obtained from coalescing binaries with equal components. Sathyaprakash, et al (1991) show that ideally mass parameters should increase by about 2% for each filter pair, starting at about 1.39 going up to about 120. With so few filters, covering this whole range of potential parameters is impossible. It was hence decided to cluster the parameters in one task around the
value expected for most neutron stars, i.e., about 1.4 increasing by about 5% each step, and in the other to cover a wider range from 2 to 6 with variable percentage increases. The component masses ranged from $1.395 \, M_\odot$ to $1.412 \, M_\odot$ with a step size of 0.0583 in task corr2 and $1.74 \, M_\odot$ to $3.017 \, M_\odot$ with a step size of 1.1487 in task corr1. The 2 phases used just needed to differ by $\pi/2$ hence 0 and $\pi/2$ were chosen. The filters are built up in two loops over mass parameter and phase shown in lines 43-90.

As in the previous task, to calculate the true amplitude of any event, the integral in equation 5.11 has to be calculated. For a filter, $\tilde{g}(f)$, sampled at points $\tilde{g}_i$, the integral can be approximated as follows,

$$
\int_0^\infty |\tilde{g}(f)|^2 \, df \approx \frac{10000}{16384} \sum_{i=492}^{1639} \tilde{g}_i^2,
$$

(5.23)

where 10000/16384 is the frequency gap between each sample. This was calculated for each filter in turn, at the start of the tasks, with the results stored in the array FILNORM.

**Correlation**

When the 4096 points of the FFT of the data group (FDATA) were received by corr1 it was immediately sent on to corr2. Simultaneously then, in both these tasks the data is cross-correlated with all the filters stored in the template array FILTERS. The data was correlated with each 4096 point filter within a DO loop that moves through FILTERS by the mass parameters one by one calculating in each case the cross-correlation for both phases at that parameter. To move through both filters of any one mass parameter, the position indicator within FILTERS needs to be incremented by 8192 in each loop.

The cross-correlations were calculated by application of Equation 5.24 originally derived in Chapter 3.

$$
cross - correlation = \text{IFFT} \left[ \text{FDATA} \times \text{FILTER}^* \right],
$$

(5.24)

where IFFT is the inverse FFT, FILTER is one of the 4096 element segments of the template and * implies the complex conjugate. Both FDATA and FILTER are complex and hence when multiplied, produce a complex result. So if element $(a + ib)$ from FDATA is multiplied by element $(c + id)^*$ from FILTER the result is $(ac + bd) + i(bc - ad)$. The IFFT of this stream produced the 4096 point real data stream, CORR, which was the time series cross-correlation representing the same time interval (1.6384 sec) as the original 32768 point group, with 1/8 of the sampling frequency.

In Chapter 3 the wraparound phenomenon that effects discrete cross-correlation is described. In this analysis its effect was mitigated by limiting the number of non-zero points at the start of any of
the filters, but still if a filter had \( NPTS \) non-zero elements, then the last \( NPTS \) of \( \text{CORR} \) were meaningless and hence were ignored. The smallest mass used was 1.395 \( M_\odot \) which provided the largest non-zero part of any of the filters with a non-zero length of 2162 points. This is less than the length of one data block and is the reason why the event search covers only the first 4 blocks in any group. This is also why, in order not to lose any data, the original version of the fifth block in any group becomes the first block in the next group.

Statistics and event search

As explained above only the equivalent of the first four blocks or 26208 points were actually analysed in any group, hence as 4096 points of \( \text{CORR} \) represent the whole group, only the first 3276 (\( = 4096 \times 26208/32768 \)) points were actually studied in the event search. The event search itself was carried out in a similar way to the search in \( \text{secd2} \) producing a 200 element results array \( \text{EVENT} \) for each group in both of the tasks. Whereas however, in \( \text{secd2} \) there are 10 elements associated with each event, here there are 12. This means a maximum of only 16 events for each group from all the filters was recorded. Again this is not overly worrying as it was found that even in badly behaved data the number of events rarely exceeded 6 or 7.

Firstly the means and standard deviations, \( \sigma_0 \) and \( \sigma_{\pi/2} \), of the two cross-correlation streams \( \text{CORR}_0 \) and \( \text{CORR}_{\pi/2} \) associated with any one mass parameter are found. Histogram statistics are again compiled over the correlation data up to \( 10\sigma \) with the data being written to 100 bins. These are again returned, when requested, to the main task where they are written to the hard disc of the PC.

Any incoming wave would not have been likely to have had the same phase as either of the filters. Hence the true cross-correlation, \( \text{CORR} \), would have to have been a phase dependant, linear combination of the correlations with both filters. Hence, for correlation element \( i \), \( \text{CORR}(i) \) is given by,

\[
\text{CORR}(i) = \cos(\theta)\text{CORR}_0(i) + \sin(\theta)\text{CORR}_{\pi/2}(i)
\] (5.25)

where both \( \text{CORR}_0(i) \) and \( \text{CORR}_{\pi/2}(i) \) here represent the deviations of these points from their respective means. This is a function of \( \theta \) and is maximised when \( \theta \) is the actual phase of the correlation. Hence by differentiating Equation 5.25 and setting the resulting expression to 0, \( \theta \) can be found as follows,

\[
\frac{d\text{CORR}(i)}{d\theta} = -\sin(\theta)\text{CORR}_0(i) + \cos(\theta)\text{CORR}_{\pi/2}(i), \quad \theta = 0.
\] (5.26)

This leads to the following expression for phase,

\[
\theta = \arctan \left( \frac{\text{CORR}_{\pi/2}(i)}{\text{CORR}_0(i)} \right).
\] (5.27)
\[ \pi \text{ or } 2\pi \text{ are added to the value of } \theta \text{ where appropriate to make the angle positive and less than } 2\pi. \] This is decided on the basis of the signs of \( \text{CORR}_0(i) \) and \( \text{CORR}_{\pi/2}(i) \) that establish the quadrant in which \( \theta \) lies.

If \( \theta \) is the true phase then \( \text{CORR}_0(i) = \cos(\theta)\text{CORR}(i) \) and \( \text{CORR}_{\pi/2}(i) = \sin(\theta)\text{CORR}(i) \). Hence it was possible to find a phase independent value for the amplitude of the true cross-correlation given by,

\[ |\text{CORR}(i)| = ((\text{CORR}_0(i))^2 + (\text{CORR}_{\pi/2}(i))^2)^{1/2}. \] (5.28)

It is this amplitude stream over which the event search was carried out.

Any point of \( \text{CORR} \) whose amplitude deviation exceeds the allocated threshold \( \text{THR} \) is considered to be an event. The deviation from the mean is already incorporated within \( \text{CORR} \). The standard deviations \( \sigma_0 \) and \( \sigma_{\pi/2} \) are used to establish the standard deviation \( \sigma \) of the stream \( \text{CORR} \) where \( \sigma \) is given by,

\[ \sigma = (\sigma_0^2 + \sigma_{\pi/2}^2)^{1/2}. \] (5.29)

\( \text{THR} \) is then, just a fixed multiple of \( \sigma \). The multiple was chosen to be 3.5 by essentially the same criteria as the threshold in \text{secd2}, ie to ensure a reasonable number of events for good data.

Tests carried out showed that this software was able to correctly locate various artificial chirps buried in Gaussian noise giving an accurate estimate of the mass parameter but only accurately finding the phase when it was close to some integer multiple of \( \pi/2 \). This is in accordance with work done by Dhurandar and others (private communication). Details of the tests are given in Appendix 1.

Again any event found was treated as though it was the first of an unbroken set of events. The event in the set with the highest amplitude was taken to be the main event whose amplitude, mass parameter, and phase were written to \text{EVENT}. Also calculated was the true amplitude of each event, using equation 5.11. The standard deviation of the noise over the frequency range was taken from the array element \text{SDT}(3) \) and the value of the integral for that filter, from \text{FILNORM}. This too is written to \text{EVENT}.

As there are only an 8th as many points to search for events here as in \text{secd2} the accuracy to which events can be located is also only an 8th as great. To improve the accuracy the values of the correlations either side of the main event were used in conjunction with the main event to predict the position of the peak correlation for that filter.

Two arrays were set up, one \( Y \), containing the amplitudes of the 3 points and the other, \( X \), the three values \(-1.0\) and \(+1\), representing the positions of the points. These arrays were sent to the routine \text{POLCOE} taken from Numerical Recipes by Press et al (1986) which returns the coefficients of
a quadratic polynomial that gives the equation of the best curve through the points, plotting the correlations against the positions. The true positions of the points were not used as they could be extremely large which could lead to coefficients that exceed the transputer capacity.

If the coefficients returned are $A$, $B$ and $C$ then the curve can be written in terms of $x$ and $y$ by,

$$y = Ax^2 + Bx + C.$$  \hspace{1cm} (5.30)

The correlation is maximised at the point where the above Equation is maximised which can be found differentiating and setting the result to 0. This point is given by,

$$\frac{dy}{dx} = 2Ax + B = 0.$$  \hspace{1cm} (5.31)

This gives the position of peak correlation, $x$ given by,

$$x = \frac{-B}{2A}$$

where $x$ is guaranteed to be within the range $(-1,1)$ as $Y(2)$ was the maximum correlation found and as such, at least as large as both $Y(1)$ and $Y(3)$.

The actual position, $x_{\text{max}}$, of the peak correlation in the data, is given by,

$$x_{\text{max}} = x + x_{0,\text{max}},$$

where $x_{0,\text{max}}$ is the actual position of $Y(2)$ in the data.

This number $x_{\text{max}}$ produced was then made equivalent to the positions given by the event search in the time series data by the following equation,

$$P_{\text{max}} = \text{int}(8 \times x_{\text{max}} + 0.49),$$  \hspace{1cm} (5.32)

where $P_{\text{max}}$ is the event's best position in a stream 26206 elements long. So if, for example, $x_{0,\text{max}} = 2013$ and $x_{\text{max}} = 2012.84$ then $P_{\text{max}} = 16103$. By experimentation this method was, in general, found to improve the accuracy of the event's positions. This is described further in Appendix 1.

$P_{\text{max}}$ is also written to the results array along with the positions of the first and last events in the set of events that surround the peak event. The results arrays from both tasks are then returned to the root where the relevant housekeeping data is added to the 4 remaining unfilled elements associated with each event. All the results array for that group are then combined into a larger array which is at the appropriate time written to tape. The structure of this array is explained in greater detail in Appendix 2.
Chapter 6

RESULTS.

6.1 Introduction

The analysis run carried out using the software developed in the last Chapter began on the 19th November 1990. Initially 6 weeks seemed a reasonable estimate for its duration based on the knowledge that the analysis ran at roughly 8 times the data taking rate of the detector over its 100 hour run. This estimate turned out to be rather optimistic as the exabyte machines seemed to be incapable of maintaining sustained usage and continually jammed or crashed or whatever exabytes do when they fail to work. The analysis was finally completed on the 11th February 1991 after 13 weeks. This problem was then further compounded by the exabyte's inability to write to certain points of some of the tapes, and then to read data just written which led to the results of the analysis initially being spread over smaller parts of a number of different tapes.

In this Chapter I discuss the results of the analysis, describing the noise statistics of both the gravitational wave (secondary error point) data, and the various streams of housekeeping data, as well as any common features between the 2. I also discuss the threshold crossing events found from the analysis.

6.2 Details of results obtained from tapes B6 and C4

6.2.1 Introduction

Tapes B6 and C4 began recording data on Friday, 3rd March 1989 at 10:29 and Saturday, 5th March at 23:02 respectively. They both ended approximately 229 minutes later at 14:18 and 2:31. Upon investigation the results obtained from these tapes were found to be typical of the overall results obtained from the whole run. Hence here I use the results obtained from these tape to demonstrate the relationships
EVOLUTION OF STREAMS WRITTEN TO TAPE C4:— (FIGURE 6.1)
EVOLUTION OF STREAMS WRITTEN TO TAPE BE- (FIGURE 3)

CALIBRATION COMES DATA (65)

SEISMIC DATA (65)

PRIMARY ERROR POINT DATA (65)
between the housekeeping streams, the secondary error point data and also the Gaussian parameter which was described in the previous Chapter and which here I show to be a good diagnostic. For each group, the means of each of the housekeeping streams were found and written to the results tape. The group means of all these streams, for tapes B6 and C4, are shown in Figures 6.1-6.4.

6.2.2 Internal data

The internal data streams are those obtained from sensors reading directly from the detector. These are the primary error point signal, read from the primary cavity and both the secondary feedback and the secondary visibility signals read from the secondary cavity as well as the secondary error point signal upon which the event search is carried out. Figures 6.1-6.4 show that these streams maintain a relatively fixed value most of the time, with any significant change in one appearing in the others as well. By consultation with the diary kept by the Glasgow group during the 100 hour run, it is clear that by far the most significant of these changes occurred when the one or other, of the cavities went out of lock. When this occurred the noise levels in each of these streams jumped up to several times its in lock level.

When the noise was seen to go bad it would do so abruptly with the means of both the secondary error point and the secondary feedback data suddenly jumping to a far higher level. The primary data also shows this but frequently a second or so after it is seen in the secondary signals which is surprising as often the source of the bad noise is the primary cavity when it goes out of lock. In analysis terms however the visibility signals may turn out to be more significant as the value of the secondary visibility signal sometimes jumps in the analysis group, (of four blocks), before the group that shows the bad noise in the other secondary streams. This could represent recognition of as much as an additional 1.3 seconds of bad noise. In determining the end of the bad data stretch the visibility data is not so helpful as it returns to its in lock state several seconds before the end of the bad data according to the other secondary streams. The primary data shows correlation only at the start of the bad stretches returning to its good data mean long before the secondary data.

Scatter diagrams showing the extent of the correlations between high noise in both the secondary feedback and primary error point signals and the secondary error point signal is shown in Figure 6.5A and Figure 6.5B. This data represents the whole of tape C4.

Figure 6.5A shows the primary error point data plotted against the secondary error point data. The abrupt distinction between low and high noise in both streams is noticeable in that the middle region of the Figure contains very few points, with concentrations in the bottom left corner (low noise in both) and also in the top right which is high noise in both. The line of points along the bottom of
SCATTER DIAGRAMS PLOTTING INTERNAL HOUSEKEEPING STREAMS AGAINST RAW SECONDARY ERRORPOINT DATA: FIGURE 6.5
the Figure results from the tendency (mentioned above) for the primary cavity to regain lock before the secondary during a stretch of high noise.

Essentially the same correlation is seen with the secondary feedback signal, which is not surprising as the two signals are basically different frequency range samples of the same signal, with low and high noise regions common to both.

6.2.3 Gaussian considerations

The major consequence deducible from the above section is that it is necessary only to look at the secondary error point noise to determine when the detector is operating properly, with well behaved, low level noise produced in both cavities. When the detector is well behaved, the noise produced is declared to be good and when the detector is not well behaved, with the cavities out of lock, the noise is declared to be bad. Then when analysing these regions of both good and bad noise, it was found that good noise is essentially Gaussian and the bad noise not. Figure 6.6 shows graphically the distinctions between good and bad noise. The diagrams are built up from part of the histogram data taken during the run. The left hand plots are formed from a histogram built up over 500 groups where the noise was known to be good and similarly the right hand side where the noise is bad. The top plots show the histogram data plotted against displacement (which is the actual displacement divided by the standard deviation) from the mean.

The top left hand plot associated with the good data clearly shows the traditional bell like Gaussian shape. The traditional Gaussian shape is given by the equation $A e^{-x^2/2}$ where $A$ is a constant and $x$ is the displacement in terms of $\sigma$. Obviously then if the logarithm of a Gaussian is plotted against $x^2$ then the resulting curve would be a straight line of slope $-\frac{1}{2}$. This is seen to be roughly the case with the good data as shown in the bottom left hand plot but not with the bad data where the slope of the curve changes sharply at roughly $3\sigma$ splitting the diagram up into 2 regions both containing extended straight line portions. This lead to the idea that in bad noise two separate Gaussian statistics were present, one dominant in the region before $3\sigma$ and the other in the region after $3\sigma$.

 Obviously, as the diagram is a logplot, there are far fewer elements concerned with the region after $3\sigma$ than with the region before. If the two statistics are defined on the basis of two standard deviations $\sigma_1$ and $\sigma_2$ and if when combined, the number of elements controlled by statistic 1 are given by $n$ and the total number of elements is $N$ then the standard deviation, $\sigma_t$, of the whole is given by,

$$
\sigma_t = \sqrt{\frac{n}{N} \sigma_1^2 + \left(1 - \frac{n}{N}\right) \sigma_2^2}.
$$

(6.1)
TYPICAL AMPLITUDE DISTRIBUTIONS - Glasgow 10m prototype. (FIGURE 6.6)
Typical amplitude distributions of filtered time series: class B ion prototype (Figure 7).
CHAPTER 6. RESULTS.

From this it can be seen, not surprisingly, that if the elements obeying the statistics with the lower \( \sigma_1 \) far out number the others then \( \sigma_t \) is roughly \( \sigma_1 \), provided \( \sigma_2 \) is not very much greater (ie an order of magnitude or so) than \( \sigma_1 \). For example if \( \sigma_1 \) is half \( \sigma_2 \) and if \( \sigma_1 \) comprises 98% of the total then,

\[
\sigma_t = 1.03 \sigma_1.
\]

This is significant when examining the slope of the regions. The first region has a slope close to, but slightly less than, \(-\frac{1}{2}\) which suggests that it is primarily normal Gaussian noise polluted by a small amount of noise with a higher \( \sigma \) that becomes dominant in the higher displacement bins. In this instance the first region would inevitably have a slope less than \(-\frac{1}{2}\) as \( \sigma_t > \sigma_1 \) and hence each bin interval, \( \sigma_t / 50 \), would be slightly greater than the interval for purely Gaussian data, subsequently the first bins would inevitably collect more points than when just considering gaussian data, with each bin increased by more than the next, leading to a severer slope.

It should be noted that even the good data plots in Figure 6.6 seem to show some kind of contamination, as the slope of the logplot is not quite straight being slightly less than -1/2 before 3\( \sigma \) and slightly greater afterwards. This was seen to be true even for the very best data stretches seen. This contamination seems to be fundamentally different from that seen in the bad data, in at least extent and probably cause, as there is no apparent middle ground between good and bad data. Either the data is roughly Gaussian or strongly non Gaussian.

Figure 6.7 shows similar diagrams built up for filtered data, ie time-series data produced by the cross-correlation of the original time-series with one of the chirp filters. The left hand diagrams show a region in which the detector was well behaved, producing good noise, whereas the right hand diagrams cover a region where the original time series noise was known to be bad. Clearly there was a lot more high amplitude activity during the bad noise, but it shows no sign of the glitch at 3\( \sigma \) seen in Figure 6.6. This has some fairly significant implications, as the filtered data contains no frequencies above 1250Hz. This strongly suggests that the source of the contamination in the full time series operates at some higher frequency than this.

Gaussian parameter

In order to assess the noise, ie deem it to be either good or bad, the Gaussian parameter was developed. As described in the previous Chapter, the parameter is calculated for each group on the basis of the first 5 histogram bins calculated for that group. The parameter is given by the total number of elements found to occupy these bins divided by the total number that should if the noise is Gaussian. Hence the nearness of the parameter to 1 would allow a decision about the goodness of
DISTRIBUTION OF GAUSSIAN PARAMETER FOR BOTH GOOD AND BAD DATA FOR TAPE C4: FIGURE 6.8

HISTOGRAM OF GAUSSIAN PARAMETER FROM BAD DATA

HISTOGRAM OF GAUSSIAN PARAMETER FROM GOOD DATA
DISTRIBUTION OF GAUSSIAN PARAMETERS FOR BOTH TAPES BE AND C4: FIGURE 6.9

HISTOGRAM OF GAUSSIAN PARAMETERS FROM TAPE C4

HISTOGRAM OF GAUSSIAN PARAMETERS FROM TAPE BE
CHAPTER 6. RESULTS.

the noise.

Figure 6.8 shows a histogram displaying the spread of the Gaussian parameter over both a region where the noise is known to be low (and hence good) and also a region where the noise is high (and hence bad). Each number on the y axis represents the sum of the number of parameters found to have values given by the numbers on the x axis. It is seen that for good data the parameter is always relatively close to 1, whereas for bad data it varies from just below 1 to beyond 4. This is consistent with Figures 6.1-6.4, where any significant changes in the Gaussian parameter occur abruptly, straddling areas of high noise in the internal housekeeping channels. These general characteristics were seen over all relevant stretches of good and bad data, from the whole 100 hour run. The Gaussian parameter could therefore be used, on its own, as a quick diagnostic assessment of the noise within the detector, and hence an immediate veto on events found in regions of bad noise within the detector.

In Figure 6.8, it is obvious that the mean of the parameter is slightly greater than 1 for the good noise. This is consistent with the high $\sigma$ contamination speculated upon above. The first 5 bins would contain more than they should; hence the parameter would be consistently forced above 1. Where the cutoff in the parameter between acceptable and unacceptable noise should be set is never going to be certain. However from studying the evolution of the parameter over the whole run it seems resonable to treat groups with parameters between 0.9 and 1.15 as good Gaussian groups. These estimates may well, on further analysis need to be expanded or contracted or even graduated fixing various levels of goodness within the upper and lower bounds. Figure 6.9 shows the histogram variations of the Gaussian parameter over the whole of both tapes B6 and C4.

6.2.4 External housekeeping

These streams are those tracking the detector's external environment, i.e. the microphone and seismic streams. From Figure 6.1-6.4 and indeed the rest of the tapes as well, it is fairly clear that neither seismic or audio activity have any significant influence on the large scale structure of the noise within the detector. This can also seen in Figure 6.10 that shows scatter plots of group standard deviations of both streams against the group standard deviation of the raw secondary error point data, taken from the whole length of tape C4. The streams vary along lines parallel to each axis, largely independent of each other. There are a few points high in both streams in each plot that suggest, that occasionally, high noise in the detector may be due to some external factor but this is certainly not the norm.

Figure 6.11 shows 2 plots concerning the microphone data. The first is a histogram that plots the
SCATTER DIAGRAMS PLOTTING EXTERNAL HOUSEKEEPING STREAMS AGAINST RAW SECONDARY ERRORPOINT DATA: FIGURE 6.10
displacement of randomly selected microphone data points from its group mean along the abscissa against the total number of points found to be at certain displacements along the y axis. The plot is quite standard, with the envelope appearing roughly gaussian. The rapid oscillation from bin to bin results from the nature of the microphone signal which takes only a limited number of integer values, which artificially biases certain displacement bins. The second plot is a scatter diagram, plotting the displacement of raw secondary error point data from its mean divided by its group $\sigma$ against the simultaneous microphone data plotted divided by its group $\sigma$. This plot shows no obvious pattern of high amplitude events simultaneous in both streams.

Figure 6.12 shows similar plots based on the seismic data. Again the histogram appears roughly Gaussian and similarly the scatter diagram shows no obvious correlation between the two streams.

**Microphone analysis**

It was possible, using the Glasgow log of the 100 hour run, to attribute some of the peaks in the microphone signal to noises within the lab. This is demonstrated in Figure 6.13 which shows the microphone data taken from tape B6. The letters above some of the peaks indicate the source of the noise as follows. B:-computer beep within the lab, P:-phone in lab, D:-door in lab, DC:-distant crash, L:-lorry.

**6.2.5 Events**

Events were found in the vast majority of the groups, more normally in the time series data where the events tended to be single point as opposed to the cross-correlation results that often came in adjacent groups of 2 points or more, but were considered all to be associated with the same event. The number of threshold crossing events found in each group was found to vary quite drastically with the noise. Figure 6.14 and Figure 6.15 are both histogram plots of the number of groups (of four blocks) against the number of events found from analysing the data within each of the groups. The plots given in Figure 6.14 show the results from a stretch of good data. The y axis represents the number of groups giving a particular number of events. The abscissa represents the number of events. The top graph gives the results of the time series analysis where the maximum number of events is only 19. The lower graph combines the results from both cross-correlation results set for each group. Therefore the maximum number of events possible is 32. The final bin in both graphs (the 19th and the 32nd) obviously represent saturation and means only that the number of events found reached or exceeded the maximum number possible to write to tape.

It can be seen from this Figure that, for good data, the majority of the groups, each 1.3104 seconds
HISTOGRAM OF SEISMIC DATA

SCATTER PLOT

SEISMIC SIGNAL: FIGURE6.12
EVOLUTION OF MICROPHONE SIGNAL OVER TAPE BE: (FIGURE 6.13)
HISTOGRAM OF THE NUMBER OF EVENTS FOR BAD NOISE - FIGURE 6.15

NUMBER OF GROUPS

0 50 100 150

NUMBER OF GROUPS

0 100 200 300

NUMBER OF EVENTS

CROSSCORRELATION EVENTS (bad data)

TIME SERIES EVENTS (bad data)
CHAPTER 6. RESULTS.

long, contains roughly 5-6 threshold crossing events in the time series and 0-3 in the cross-correlation data. Figure 6.15 shows the equivalent graphs for a stretch of bad data. Here the average number of events in each group is much higher with 30 – 60% of the groups saturating their results array. It should be noted that even data that appears to be highly Gaussian produces more threshold crossings than true Gaussian data which on average would produce slightly less than 2.4 events per events group.

Figure 6.16 shows the relative heights of housekeeping streams at confirmed events in the secondary error point stream. The values displayed at each point are the amplitude of the point minus the mean of the stream and then divided by the streams standard deviation, \( \sigma \). It can be seen in both streams that the amplitudes are primarily distributed fairly evenly around 1. However it can also be seen, particularly in the microphone stream, that occasionally the amplitudes do deviate considerably more than \( \sigma \) from the mean. Both streams are also seen in Figure 6.17 in histogram form.

Working on the basis that the data that produced Figure 6.11 is typical microphone data, its examination revealed that of the 65536 samples in the data, only 3 exceeded 4\( \sigma \) which is less than 0.005% of the total. The microphone data in Figure 6.17, although largely similar in distribution to the data in Figure 6.11, showed 21 points in total from the 6000 where the deviation exceeds 4\( \sigma \). This represents 0.35%, from which it seems clear that occasionally events are caused by audio activity.

Similarly with the seismic data, Figure 6.12 shows no samples at all that exceed 4\( \sigma \), whereas the data plotted in Figure 6.17 shows 3. Of these, 2 points are adjacent in the stream and are preceded by an unusual region that casts some doubt on the integrity of the seismic data in this area and the third is only just above 4\( \sigma \). It seems unlikely then that seismic activity is systematically responsible for events.

6.2.6 Sensitivity of the detector

The dimensionless strain sensitivity \( h \), corresponding to a level of 1\( \sigma \), in the detector noise was calculated individually for each group within the analysis. For Gaussian data, \( h \) varied from about \( 0.8 \times 10^{-19} - 1.2 \times 10^{-19}/\sqrt{\text{Hz}} \).
RELATIVE HEIGHTS OF HOUSEKEEPING SAMPLED AT EVENTS (FIGURE 16)
HISTOGRAMS OF EXTERNAL HOUSEKEEPING SAMPLED AT EVENTS (FIGURE 17)

Displacement

Seismic Signal

Displacement

Microphone Signal

Number in bins

50

100

200

300
Chapter 7

SUMMARY AND CONCLUSIONS

In this thesis, I have described a system that analysed data collected by the 10 m prototype interferometer in the Department of Physics and Astronomy at the University of Glasgow. It searched this data, looking for several specific forms of gravitational wave, as well as compiling noise statistics. I discussed these potential sources in Chapter 2 and described the main mathematical techniques used in the analysis in Chapter 3. The data under examination here was briefly described in Chapter 4. In the rest of this Chapter I describe the main consequences of the software development described in Chapter 5 and the results of the analysis given in Chapter 6.

The gravitational wave data analysis system was developed to run automatically, without human interference, on a small parallel system. With its limited ambitions and negligible chance of actually finding any real event, it was intended only as a prototype from which those designing future systems may benefit. It was designed on a network of 5 transputers, to read the data directly from tape, analyse it and then write the results back to tape in as time efficient a manner as possible. This essentially meant ensuring that the duration of the tasks on the 4 Slave transputers, that did the actual analysis, did not exceed that of the task in the Root that had to both read the data from the tape and unpack it into its various constituent streams. Each group of data, of duration 1.3104 seconds, took about 30 seconds to pass through the whole network which, as the network was three tier, implied that the average time between completing one group and completing the next was about 10 seconds. This is roughly 7.5 times real time. This time was dictated by the Root task which spent some 80% of its time just separating the secondary error point data and the various housekeeping streams. This represented quite a significant overhead in the overall analysis. It was essentially a consequence of the fairly complex way the channels were multiplexed together as the de-multiplexing routines involved many complicated and time consuming bit operations. The Glasgow data is currently being read from the existing tapes and written to others, in a simpler
format more conducive to rapid recovery. It is also likely future data streams sampled in and around gravitational wave detectors will not be written in this way.

The transputers themselves performed admirably, the transfer of data between them producing no significant or indeed measurable overhead even when quite large amounts of data were being passed. Their main drawback was their lack of memory, which meant that the de-multiplexing routines mentioned above were all effectively confined to the Root transputer as only the Root had both access to the de-multiplexing library and sufficient memory to contain all the housekeeping data.

Before the noise was searched in any way for the presence of a signal, it underwent two procedures in frequency space. The first removed any frequency bias within the detector and the second optimised the noise for cross-correlation with the coalescing binary filters. To allow an assessment of the extent of the frequency bias in the detector, calibration combs were periodically applied to the timeseries every 210 seconds. These combs revealed themselves in frequency space as a row of teeth of varying amplitudes at various equally spaced frequencies. The teeth were all originally applied with the same amplitude (except for the first which was 1000 times stronger). Knowing these amplitudes, the observed amplitudes of the teeth in the data were used to produce an inverse comb that when multiplied through the data removed any bias. This method was not wholly satisfactory mainly because of the nature of the noise. The noise was found to be very high at low frequencies, up to about 500-600 Hz, which contained the first tooth, but then to drop quite abruptly to the region containing the second tooth. Subsequently in the inverse comb the amplitudes of the first two teeth differ quite considerably which, despite the linear interpolation between the two teeth, when applied to the noise, results in the connection between the two regions not appearing smooth. This is not as significant over the rest of the range as the noise changes only slowly. This situation could be improved by the addition of more closely spaced combs, particularly over the lower frequency regions.

The second procedure in which the data was optimised for cross-correlation, was a consequence of the matched filtering theorem. It required finding an average value for $S_h(f)$, calculated as a variance, over 128 narrow bandwidths each containing 128 complex points. Each point across the full bandwidth was then divided by the value of $S_h(f)$ found for the group in which it lies. This procedure is potentially destructive if there is some kind of structure in the noise on the same scale as the narrow bandwidth. This is distinctly possible as 128 points represents a bandwidth of nearly 80 Hz. This method could be improved by not just building up the spectra over just one group, but many, $$\text{This increases the number of points representing the same frequency range, allowing}$$
a significant decrease in the bandwidth over which the variance is calculated, while maintaining a reasonable number of points within each bandwidth. After this, of course, rather than dividing through the whole data stream by an abruptly changing series of histograms, a smooth curve could be fitted to the points.

After the calibration and weighting, noise statistics were compiled for the time series data. It was observed that the group root mean square value of the secondary error point data remained steady and low for long periods, occasionally abruptly leaping in value and becoming erratic. These periods always coincided with similar periods in those housekeeping channels reading directly from the detector and they were invariably found to be due to one of the laser cavities going out of lock.

The noise in the detector should ideally be Gaussian. I found, by studying histograms of amplitude statistics that in the steady low noise regions mentioned above, the noise was roughly but not quite Gaussian, with a few points showing some high amplitude deviation from the Gaussian the rest of the points seemed to describe. In the high noise regions however, the noise took on a different character, with many points making up a large non-Gaussian tail beginning at about $3\sigma$. The straightness of both regions, before and after $3\sigma$, in the logplot suggested the presence of two Gaussian distributions in the noise, one with a significantly higher standard deviation than the other. An investigation of the possible sources of this tail is currently underway.

A single number was calculated for each group from that group’s histogram data, by summing the number of points that lie between the mean and $\sigma/10$ and then dividing the result by the number of points that would occupy this region if the noise is Gaussian, the nearness of this number to one allowed an assessment of the extent of the Gaussian nature of the noise. This number, named the Gaussian parameter was shown to be a good diagnostic, correlating strongly with the secondary error point data as well as the relevant internal housekeeping data, giving numbers very close to, if slightly over, one for data that is roughly Gaussian and jumping to 3 or 4 for non-Gaussian stretches. The analysis tended to show that the noise in the detector was either in one of these states or the other. The states were given as good when the detector noise was low and gaussian and bad when the noise was high and non-Gaussian.

Following on from the above, another important consequence of the gaussian parameter, is its implications for the housekeeping data. Significant variations in the gross structure of both the principle internal housekeeping streams, ie the primary error point and the secondary feedback data, showed up simultaneously or even before in the Gaussian parameter. This effectively renders these housekeeping streams obsolete as indicators of behavior in the detector. The same also appears to be basically true of the external housekeeping, notably the seismic and microphone signals, for
completely the opposite reason. Fluctuations in these streams showed no definite correlation with either the internal signals or the Gaussian parameter. The possible exceptions to the above are the primary and secondary visibility signals that may possibly reveal bad noise even before the Gaussian parameter. This requires further investigation.

Each group was searched for time series wideband events that exceeded the group's mean by a preset multiple, in this case 4, of the group's standard deviation. The number of events found in each group tended to correlate strongly with the Gaussian parameter, with good noise producing an average of about 5-6 events per group and bad usually completely saturating the event array, i.e. producing more events than space was allocated to record. It should be noted that in the good data twice as many events were found as would have been the case for pure Gaussian noise, presumably due to whatever caused the slight deviation from Gaussian noise noted above. Each event was treated as being potentially part of a larger group of events. This turned ou: to be largely unnecessary as, in contrast to the German data, very few events were seen to be consecutive.

The first 1250 Hz were correlated with 2 templates of coalescing binary filters. Both of these templates were built up at the initiation of the analysis and stored in the transputer memory throughout. The resulting cross-correlation data was statistically analysed and found to be Gaussian over stretches of known good data, not showing the deviation from pure Gaussian found in the equivalent time series. This leads to the suggestion that the non-Gaussian component in the time series may be at higher frequencies than those present in the correlation, i.e. greater than 1000 Hz. As with the time series the number of events found varied with the Gaussian parameter. For good noise the most common number of events was in fact none at all, tapering off rapidly after about 3 or 4. For bad data, the event arrays were most frequently saturated. Again each event was treated as though it were part of a larger group of events and this time this was found, in general, to be the case, with far more multiple events of 2,3,4 or more points in a row than single threshold crossers. It also transpired that in both sets of events, a small proportion of the total number, were attributable to something that also showed up in one of the housekeeping streams, primarily the microphone stream.

Ultimately the routines analysed and reduced the data to \( \frac{1}{50} \) th of its original volume. The results occupied approximately half of one exabyte tape, which represents about 0.7 Gbytes. Within this, there is undoubtedly some information that is, in retrospect, superfluous as well as other information that is inefficiently recorded. Doubts have to be raised over the usefulness of a large portion of the housekeeping data, indeed in future detectors there may be little point in recording some of these streams, particularly the internal housekeeping. It seems clear that a larger network of transputers or some similar parallel assembly could have completely analysed the data and indeed some parallel
element is certain to be present in the final analysis system that reads data directly from the full sized detector. This analysis system will, hopefully in the not too distant future, enable the announcement of the first detection of gravitational waves.
Bibliography


BIBLIOGRAPHY


Appendix 1: Testing the cross-correlation routines

A.1 Introduction

In this appendix I briefly describe the diagnostic tests carried out on the routines running in two of the tasks, CORR1 and CORR2 that cross-correlate 1.6 seconds of secondary error point data with a template of filters built up and held within each of the tasks. In order to test the task routines I buried a variety of artificial chirps in Gaussian noise and then supplied the resulting data to a routine based around the relevant source code taken from the tasks.

A.2 Case 1:- signal directly proportional to filter

In this case, the filter $q_0$ and the artificial signal, given by $Hq_0$, are defined on the basis of the same mass parameters and initial phases, where $H$ is simply a linear factor representative of and proportional to the amplitude of the signal. This would certainly be very unlikely in any real observations but serves as a means of comparison when considering the various cases when the signal and filters are not defined on the basis of the same parameters.

Figure A.1 shows a chirp and filter respectively. Both were constructed on the basis of a mass parameter of 2 and zero phase. The non zero parts of both therefore are identical apart from some multiplicative factor starting at 300 Hz and climbing to 1000 Hz. When the filter and chirp are cross-correlated the result, which in this instance is effectively an auto-correlation also shown in Figure A.1, gives a distinct peak at the time where the chirp starts.

When the same chirp is buried in noise, cross-correlation becomes the only means of its detection. Figure A.2 shows the chirp starting at 0.65 seconds. The Gaussian noise was artificially generated by routines found in Numerical Recipes by Press et al(1986). When the chirp is added to noise whose standard deviation equals the chirp's maximum value and is sampled at the same rate, the chirp is still just about noticeable within the noise to the unaided eye but would probably not distinguish itself in anyway to a simple threshold crossing search routine. When the data, (signal + noise), was cross-correlated with the filter template a number of the filters gave peak threshold crossings at the right time with the highest for the filters with mass parameter 2 as expected which gave a peak S/N of 8.89. Figure A.2 also shows the cross-correlation data for each of the filters with a mass parameter of 2. Both give very obvious peaks at the correct time with the correlation with the filter of phase 0, as would be expected, being slightly higher than the correlation with the filter of phase $\pi/2$. The deepest that any chirp was buried in noise and still picked up by cross-correlation giving a S/N ratio of 3.5 (the ratio chosen in the analysis), was for a chirp based on a mass parameter of 1 with 0 phase whose maximum amplitude was only 9% of that of the noise.
CHIRP. (Mpar=2.0, Phase=0.0)

FILTER. (Mpar=2.0, Phase=0.0)

CORRELATION OF NOISE WITH FILTER.

CLOSE UP OF ABOVE.
CHIRP. (Mpar=2.0, Phase=0.0)

CHIRP IN NOISE. ($\sigma_{\text{noise}} = \text{chirp max}$)

CORRELATION OF NOISE WITH FILTERS OF Mpar=2.0, Phase=0.0

CORRELATION OF NOISE WITH FILTERS OF Mpar=2.0, Phase=1.57049

FIGURE A.2
A.3 Case 2:– General chirp in noise.

Any chirp picked up in a data group could not be expected to be an exact match with any of the filters built up and held in the template. They would differ in both mass parameter and initial phase, the most significant consequence of which being the resultant uncertainty in the position of the event in the group from which is calculated the event's time of arrival.

Mass parameter.

Figure A.3 shows a chirp based on the mass parameter 5.5, 0.5 seconds into the group, which was then buried in Gaussian noise with the standard deviation of the noise 4 times the maximum value of the chirp. When correlated with the filter template the correlation streams gave peak S/N of 3.82 and 3.74 for the filters based on the mass parameters 5 and 6 respectively. These correlation streams are also shown in figure A.3. It can be seen that the two peaks are displaced slightly either side of the correct time. This displacement in the time of arrival was first demonstrated by Schutz (1986a). It is a consequence of the lengths of the filters. The non-zero part of the filter based on the smaller mass parameter is longer than the actual chirp in the noise whilst covering the same frequency range hence the whole correlation is spread out as the higher frequency regions meet and correlate before the lower frequency areas whereas with identical chirps the peak correlation occurs instantaneously as the same frequency regions in either chirp meet at the same time. The peak is shifted below the true event position because the higher frequency, higher amplitude region contributes more to the correlation than the lower frequency and amplitude regions hence as the higher region correlations occur first, the peak is shifted.

The opposite case occurs with the higher mass parameter. The non-zero part of the filter is shorter than the chirp in the noise hence the peak correlation produced when the high frequency regions pass over each other occurs after the lower frequency correlation causing the peak correlation to be shifted beyond the true event position.

As the mass parameter of the signal was varied, the signal's general detectability was seen to worsen the further the mass parameter of the signal got from a mass parameter on which a filter was based. This is shown in figure A.4 where the peak S/N in the correlation streams for 5 signals is shown. The mass parameters vary from 3 to 4 as shown on the abscissa and their maximum values are 18% of the standard deviations of the noise. It can be seen that the peak S/N at both 3 and 4 exceed 3.5 and hence would be picked up by the analysis routines as at a mass parameter of 3.5 the peak S/N, (ie the larger of the peaks obtained from the correlations with filters 3 and 4), dips slightly below 3.5 and so would be missed. In general the peak correlation obtained for signals that
CHIRP. (Mpar=5.5, Phase=0.0)

CHIRP IN NOISE. ($a_{\text{noise}} = 4 \times \text{chirp max}$)

CORRELATION OF NOISE WITH FILTERS OF Mpar=5.0, Phase=0.0

CORRELATION OF NOISE WITH FILTERS OF Mpar=6.0, Phase=0.0

Figure A.3
FIGURE A.4
fall half way between two filters is \( 70 - 95\% \) of that obtained when the signal falls directly on one of the filters. The higher the mass parameter the smaller the difference.

In the tests carried out, the largest displacement of a maximum peak from the true position of the event when the signal and filter that gave the peak differed by a mass parameter of 0.5 was 225 points which represents 0.01125 seconds which is over twice the size of the light travel time, of 0.0045 seconds, between Glasgow and Garching that defines the coincidence window. This, of course, could lead to coincident events being missed if each event from each filter is treated separately but if events in successive filters that are reasonably close together are regarded as due to the same event then plotting a best fit curve through several of the associated peaks could enable true positions and maximum S/N's to be found. This is shown in figure A.5 which shows a quadratic curve fitted through the three peak correlations for one particular signal in noise. It is seen that the curve peaks just beyond point 10000 which is where the chirp was buried in the noise. It should be noted that in tests, a curve through 3 points occasionally did not improve the estimate of the events position however, in these cases, fitting a quartic or an even higher order polynomial (if possible) frequently did.

Figure A.6 shows two related plots essentially both showing how the peak S/N varies with mass parameter. The upper plot shows the change in peak S/N with mass parameter for signals buried equally deep in the same stretch of noise. Its seen that higher the mass parameter the higher the peak S/N. This follows from theory outlined in Chapter 3 that relates the S/N to the number of cycles in the signal and hence also its time duration and total energy. The relationship can be expressed as follows,

\[
S/N \propto \sqrt{\text{number of cycles}} \propto \sqrt{\text{time}} \propto \sqrt{\text{energy}}.
\]

This is shown in the lower plot.

Phase.

The analysis failed to correctly estimate the phase of any of the signals other than those that were defined on almost precisely the same phase and mass parameters as those that defined the filters. This is shown diagrammatically in Figure A.7 which shows the results of cross-correlating 9 chirps with the same mass parameter but varying phase with a pair of filters with the same mass parameter. The plot shows the phase of the chirps calculated by the analysis against their actual phases. This error in the calculated phase is minimal only when the signal is a linear multiple of one of the filters. In the future if it is possible at all to accurately estimate the phase of any coalescing binary event it is going to have to be by a far more sophisticated method than employed in this work.
FIGURE A.6
PHASE OF TEST CHIRP

FIGURE A.7
The four plots given in Figure A.8 show a chirp built up with a phase of π/4 which is then buried in noise whose σ is 2.5 times that of the chirp and also the correlation of this chirp in noise with the two filters with the same mass parameters. It is seen that the correlations are virtually identical in form with very similar amplitude peaks.
CHIRP. (Mpar=3.0, Phase=0.785245)

CHIRP IN NOISE. ($\sigma_{\text{noise}} = 2.5 \times \text{chirp max}$)

CORRELATION OF NOISE WITH FILTERS OF Mpar=3.0, Phase=0.0

CORRELATION OF NOISE WITH FILTERS OF Mpar=3.0, Phase=1.57049

FIGURE A.8
APPENDIX 2: THE STRUCTURE OF A RESULTS BLOCK

RESULTS BLOCK (8192 points)

Block padded out with 236 zeros

13 Groups each of 612 points

RESULTS GROUP

HEADER BLOCK

12 points
1. Tape number.
2. Year.
3. Month.
4. Day.
5. Hour.
6. Minute.
7. Second.
8. Microphone mean.
10. Seismic mean.
11. Secfed mean.
12. Secvis max.

3 Subgroups

SUBGROUPS 2 and 3 (200 points)

MINI-HEADER FOR 2 AND 3.

1 point
1. Number of events in subgroup.

EVENTS IN SUBGROUPS 2 AND 3.

12 points
1. Event number (1-16).
2. S/N peak.
3. h value.
4. Pos of main peak.
5. Mass parameter.
6. Phase.
7. Pos of first peak.
8. Pos of final peak.
11. Secfed/mean.

SUBGROUP 1 (200 points)

MINI-HEADER FOR 1.

7 points
1. Number of events in subgroup.
2. \( \sigma \) of timeseries.
3. \( \mu \) of timeseries.
5. Gaussian parameter.
6. \( \mu \) before calibration.
7. SDT4 for h.

EVENTS IN SUBGROUP 1.

10 points
1. Event number.
2. Data point of first threshold crosser.
3. Data point of peak threshold crosser.
4. Data point of final threshold crosser.
5. S/N peak.
6. h value.
7. Micro/mean.
8. Prierr/mean.
10. Seismic/mean.
Appendix 3

This Appendix contains a complete listing of the software described in this thesis. Each task is distinct and the lines are numbered. Also included is a copy of the listings of those routines taken from Numerical Recipes (Press et al., 1986).
ELSE
    CALL PKFEND
    GOTO 172
END IF

GO TO 177

EXIT

END
When receiving the coordinates of a polynomial from which
the result of the Bessel function of the integral orders is obtained.

The formula is given by

\[ J_n(x) = \frac{1}{2} \int_0^\infty \cos(x \sin(\theta)) \sin(n \theta) d\theta \]

Where

- \( J_n(x) \) is the Bessel function of order \( n \)
- \( x \) is the argument
- \( \theta \) is the variable of integration

The integral is evaluated by numerical methods, such as

1. **Trapezoidal Rule**: Approximates the integral using trapezoids.
2. **Simpson's Rule**: Uses parabolic arcs to approximate the curve.
3. **Gaussian Quadrature**: Selects specific points and weights to
   achieve higher accuracy.

These methods require careful selection of the number of
points and the range of integration to achieve desired accuracy.

For practical applications, software libraries like

- **NumPy** in Python
- **MATLAB**
- **SciPy**

provide functions to compute Bessel functions efficiently.

---

**Example declarations**

```c
#include <math.h>

float Jn(float x, int n) {
    float c = 0.0;
    float y = 0.0;
    int i;

    for (i = 0; i <= n; i++) {
        c += pow(x, 2 * i) / (factorial(2 * i) * factorial(i) * 2 * i + 1);
    }
    return c;
}
```

**Function definition**

```c
float Jn(float x, int n) {
    float c = 0.0;
    float y = 0.0;
    int i;

    for (i = 0; i <= n; i++) {
        c += pow(x, 2 * i) / (factorial(2 * i) * factorial(i) * 2 * i + 1);
    }
    return c;
}
```

This function computes the Bessel function of the first
kind, order \( n \), using the infinite series expansion:

\[ J_n(x) = \sum_{k=0}^{\infty} \frac{(-1)^k}{k! (n+k)!} \left( \frac{x}{2} \right)^{n+2k} \]

**Accuracy and range**

- **Range**: \(-\infty < x < \infty\)
- **Accuracy**: Depends on the implementation and
  can range from single to double precision.

**Usage**

- In engineering and physics, particularly
  for problems involving cylindrical coordinates.
- In digital signal processing for analyzing
  signals in the frequency domain.

---

**Summary**

The Bessel function of the first kind is a solution
to the Bessel differential equation and is widely
used in various scientific and engineering applications.

---

**Further reading**

- *Handbook of Mathematical Functions* by
  Abramowitz and Stegun
to the position of the first point in the group.

If the second point lies beyond the first point, the two points are assumed to be the start and end points, respectively.

If the second point lies on or before the first point, the two points are assumed to be the end and start, respectively.

If the second point lies after the first point, the two points are assumed to be the start and end, respectively.

CONTRAST

If (T1-R1) > 0 THEN

If (T1-R1) < 0 THEN

If (T1-R1) = 0 THEN

The position of the first point is the start of a new group, and the second point is the end of the previous group.

If the second point lies on or beyond the first point, the two points are assumed to be the end and start, respectively.

If the second point lies before the first point, the two points are assumed to be the start and end, respectively.

If the second point lies after the first point, the two points are assumed to be the start and end, respectively.

CONTRAST

If (T1-R1) > 0 THEN

If (T1-R1) < 0 THEN

If (T1-R1) = 0 THEN

The position of the first point is the start of a new group, and the second point is the end of the previous group.

If the second point lies on or beyond the first point, the two points are assumed to be the end and start, respectively.

If the second point lies before the first point, the two points are assumed to be the start and end, respectively.

If the second point lies after the first point, the two points are assumed to be the start and end, respectively.

CONTRAST

If (T1-R1) > 0 THEN

If (T1-R1) < 0 THEN

If (T1-R1) = 0 THEN

The position of the first point is the start of a new group, and the second point is the end of the previous group.

If the second point lies on or beyond the first point, the two points are assumed to be the end and start, respectively.

If the second point lies before the first point, the two points are assumed to be the start and end, respectively.

If the second point lies after the first point, the two points are assumed to be the start and end, respectively.

CONTRAST

If (T1-R1) > 0 THEN

If (T1-R1) < 0 THEN

If (T1-R1) = 0 THEN

The position of the first point is the start of a new group, and the second point is the end of the previous group.

If the second point lies on or beyond the first point, the two points are assumed to be the end and start, respectively.

If the second point lies before the first point, the two points are assumed to be the start and end, respectively.

If the second point lies after the first point, the two points are assumed to be the start and end, respectively.

CONTRAST

If (T1-R1) > 0 THEN

If (T1-R1) < 0 THEN

If (T1-R1) = 0 THEN

The position of the first point is the start of a new group, and the second point is the end of the previous group.

If the second point lies on or beyond the first point, the two points are assumed to be the end and start, respectively.

If the second point lies before the first point, the two points are assumed to be the start and end, respectively.

If the second point lies after the first point, the two points are assumed to be the start and end, respectively.