

# **Capacity of Multiple-Input Multiple-Output Wireless Communication Systems Operating in the HF Band**

by

Nigel Leonard Brine

B E (COMPUTER SYSTEMS ENGINEERING)

Thesis submitted for the degree of

**Doctor of Philosophy**

in

School of Electrical and Electronic Engineering,

Faculty of Engineering,

Computer and Mathematical Sciences

The University of Adelaide, Australia

2010

## Chapter 6

# Data Collection and Analysis

The typical approach to calculating MIMO capacity involves using transmit and receive antenna arrays to record the channel matrix, which is then substituted into the general MIMO capacity equation. Such an approach is taken in [9]. For the HF scenario, calculating MIMO capacity in this way requires considerable cost and effort since transmit and receive systems featuring a large number of antenna array elements need to be implemented, and operated a large distance apart. Furthermore, because the multipath structure of the channel is not considered using this approach, the appropriate number of transmit and receive antenna elements to use is not known. The number of antenna elements used places an upper limit on the rank, and hence capacity, of recorded channel matrices.

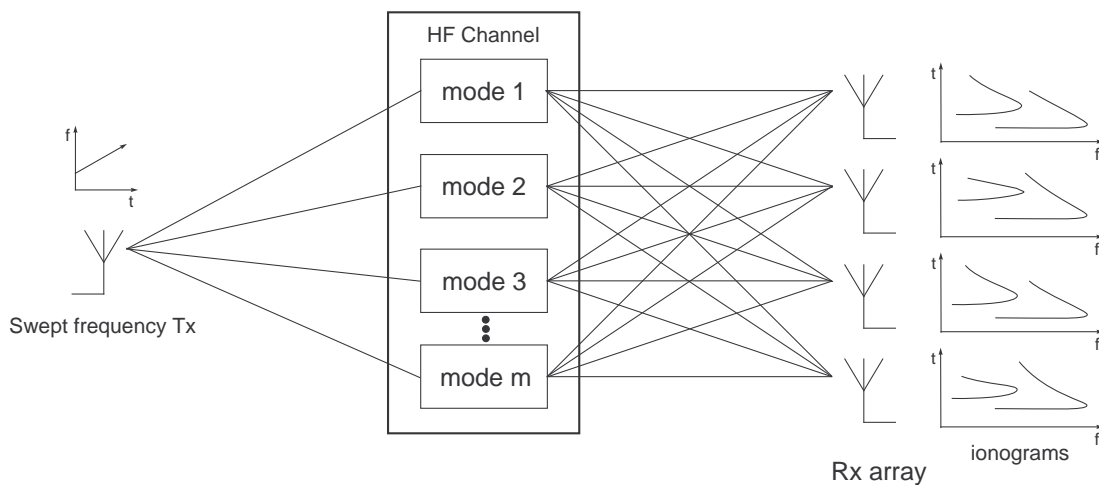
In the previous chapter an alternate approach to calculating HF MIMO capacity was devised which involves using the Gesbert MIMO channel matrix model equation (5.2) to generate channel matrices from antenna correlation and mode correlation matrices. The generated channel matrices are substituted into the general MIMO capacity equation with no CSI (2.9) to produce capacity results.

To determine the number of modes and calculate mode correlation, which reveals the underlying eigenmodes of the channel, and hence the MIMO capacity limit, a single transmit antenna, and a single receive antenna are required. An MCR (multi-channel receiver) with four antenna elements was implemented so that antenna correlation calculations could also be performed.

In the following subsections the experimental setup used to collect HF radio data is described, along with the issues involved with processing and analysing of the data.

### 6.1 Experimental Setup for Data Collection

A diagram of the experimental setup for data collection is shown in Figure 6.1. The receive antenna array used is shown in Figure 6.2, with angles between the lines intersecting pairs of antennas and True North shown. The direction of the transmitter was roughly north of the receive array. Photos showing the implementation of the MCR system are shown in Figures 6.3–6.5. See Appendix A for more information about the MCR system.



**Figure 6.1.** Experimental setup

The transmit signal used was a pre-existing oblique FMCW ionosonde transmit signal originating from Darwin, with a sweep rate of 125 kHz/s. The receiver was located in Adelaide and employed a local FMCW oscillator GPS locked to the transmitter. The receiver comprises two separate receiver sections, each with its own local oscillator. The receiver can therefore be tuned both to the band containing the ionosonde signals as well as an adjacent band containing just noise and interference, making it useful for the application of beamforming algorithms, and allowing good estimates of SNR to be made. The receiver may also be tuned to two sets of ionosonde signals simultaneously. If the two ionosonde transmit signals originate from the same spatial location, then recorded data can be processed to obtain channel time correlation calculations. Such calculations are important in determining the appropriate interval between channel matrix estimates at the receive end of a MIMO system.

Data was collected using this setup during February of 2005.

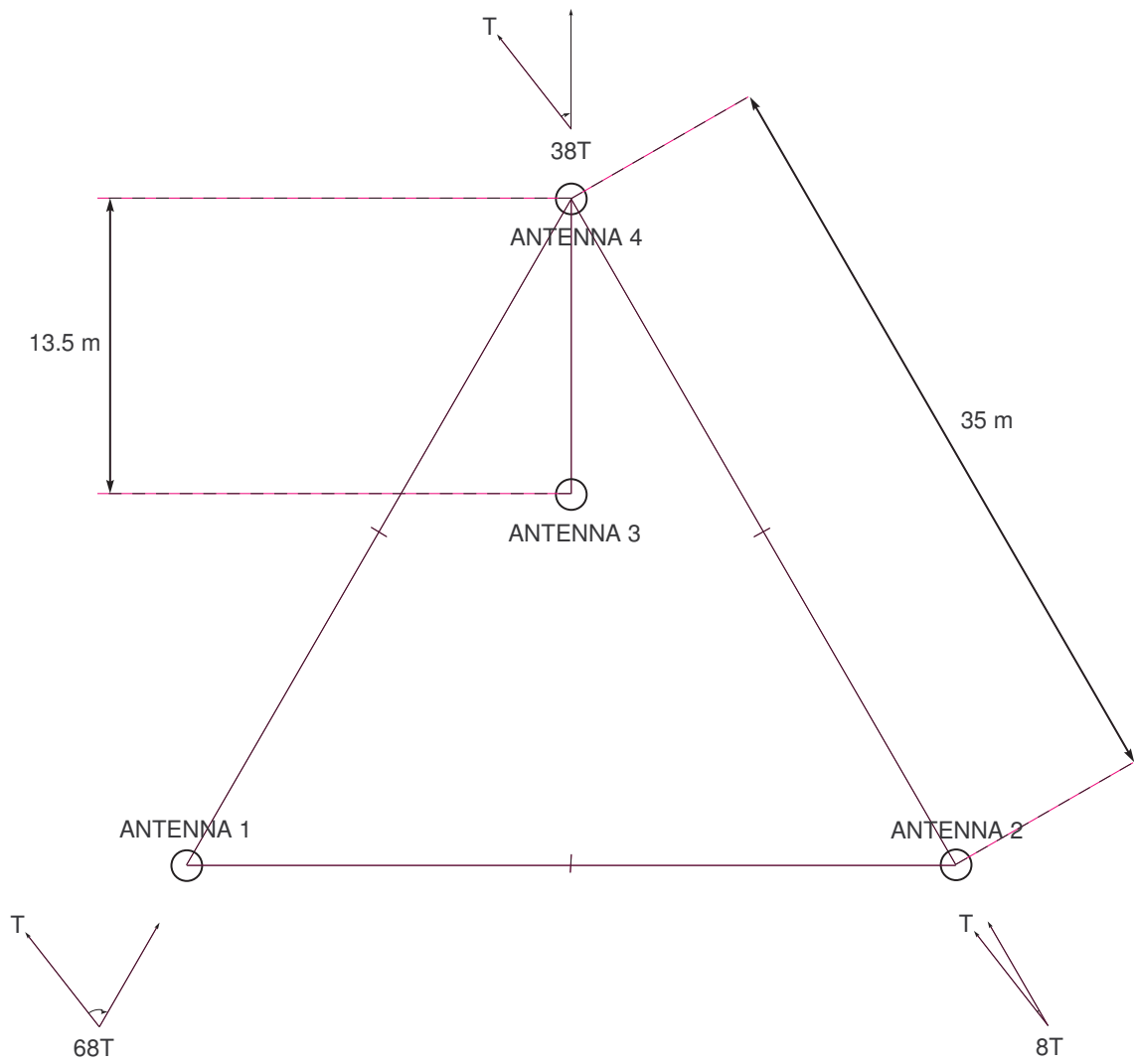
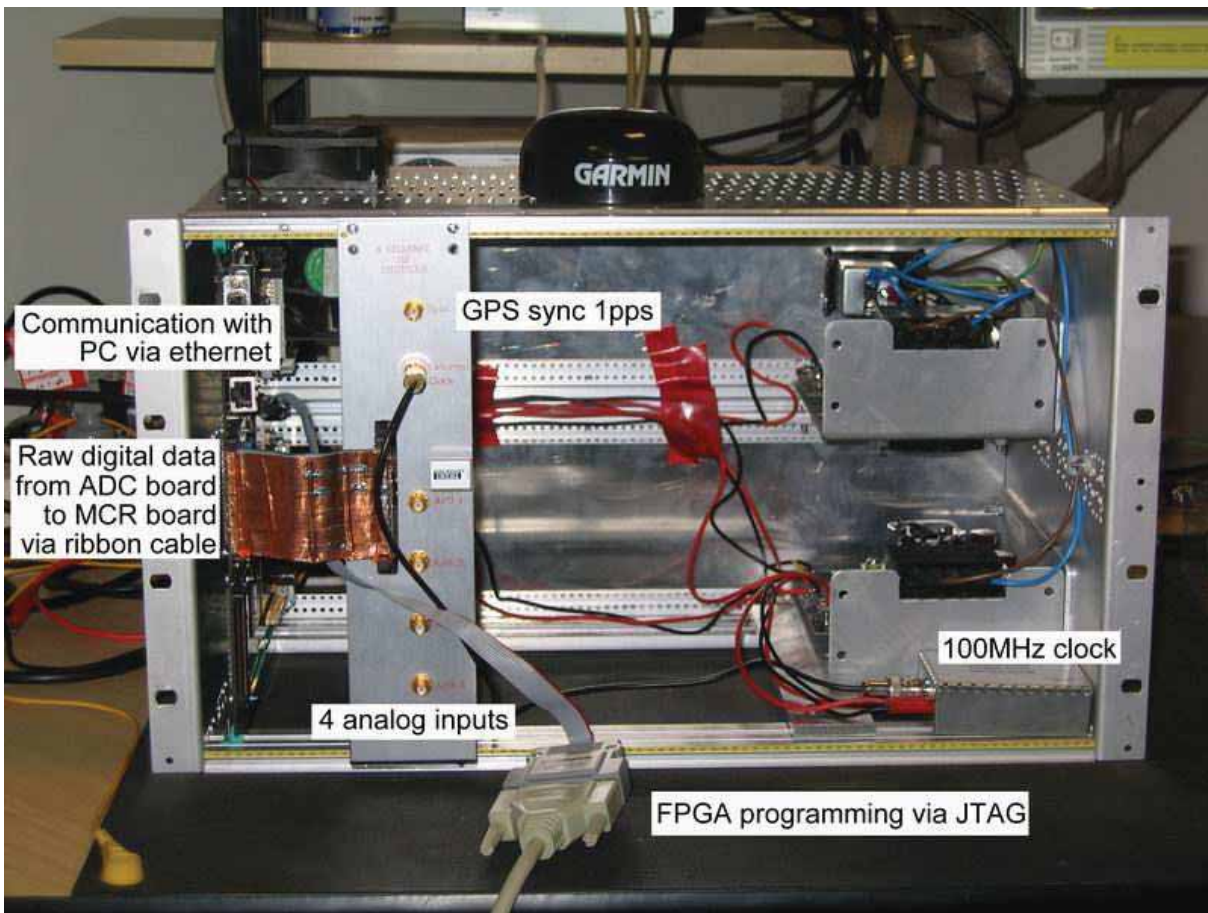


Figure 6.2. Receive antenna array

## 6.1 Experimental Setup for Data Collection

---



**Figure 6.3.** A photo of the complete MCR unit

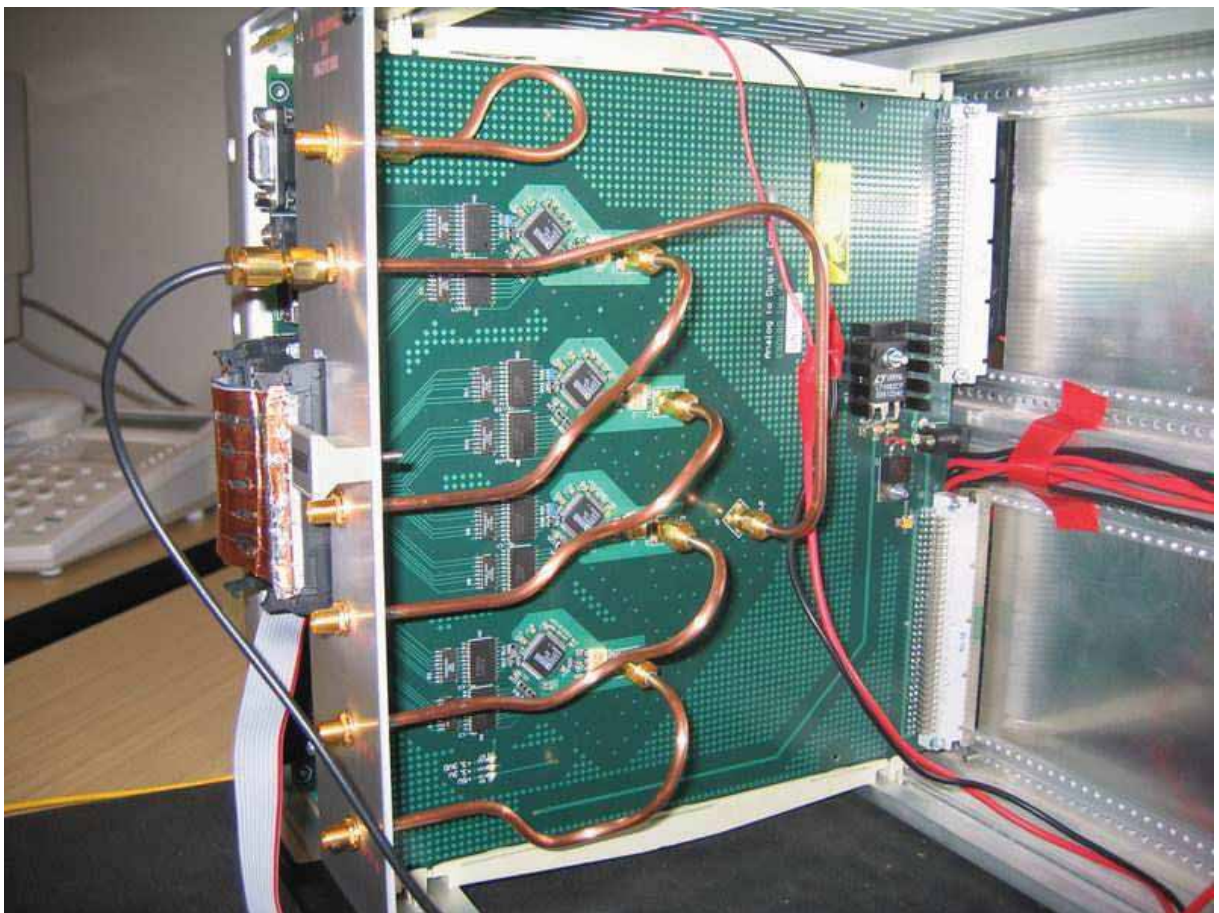
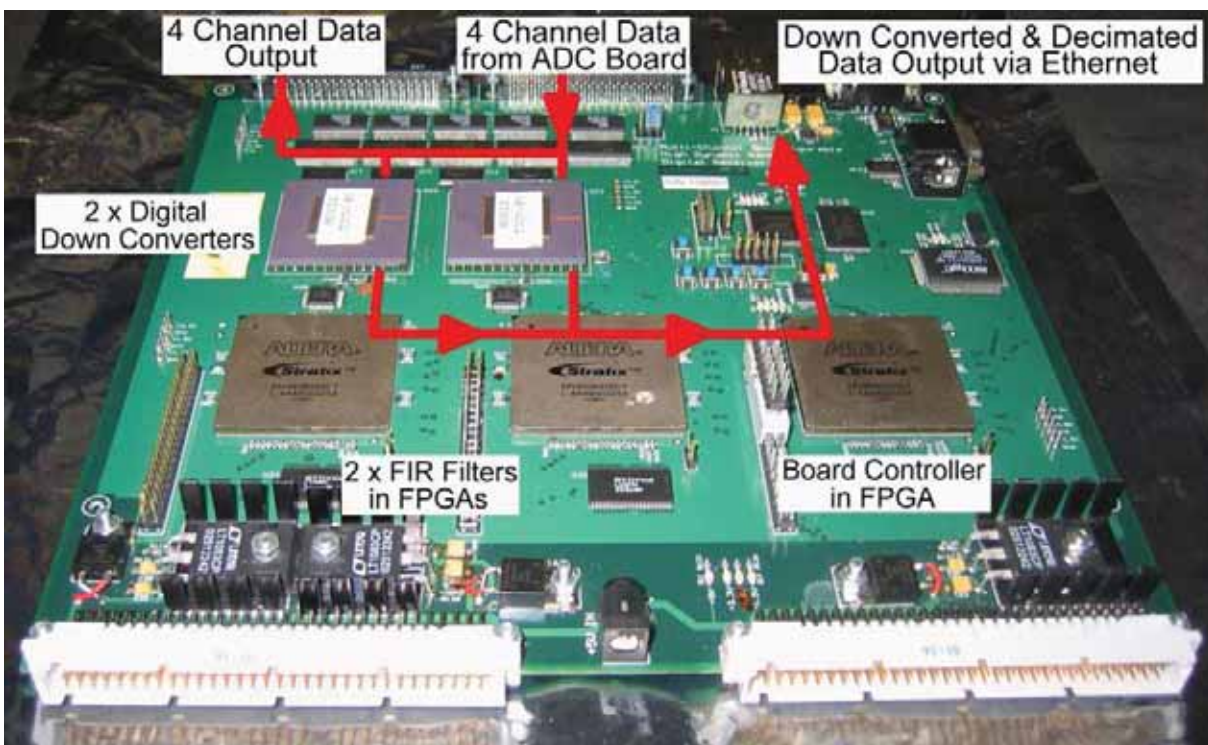


Figure 6.4. A photo of the ADC board



## 6.1 Experimental Setup for Data Collection

---



**Figure 6.5.** A photo of the MCR board

## 6.2 Oblique FMCW ionosonde theory

In this section, important oblique FMCW ionosonde equations are given, which are used to generate signals for use in the calculation of antenna and mode correlation from recorded HF radio data.

For the oblique FMCW ionosonde scenario, the transmitted signal is synchronized to the receive local oscillator, and the transmitted signal sweep rate is equal to the receive local oscillator sweep rate. Assuming a single propagating mode exists at the currently tuned frequency, by the time the transmitted signal arrives at the receiver, the receive local oscillator frequency  $f_{LO}$  has advanced by an amount given by the product of the sweep rate  $k$  and the propagation delay  $t$ , compared with the received signal frequency  $f_{Rx}$

$$f_{LO} = f_{Rx} + kt.$$

Mixing the receive local oscillator and the received signal produces a signal of frequency given by the product of the sweep rate  $k$  and the propagation delay  $t$

$$\begin{aligned} \cos(f_{LO}t) \cos(f_{Rx}t) &= \frac{1}{2} \{ \cos[(f_{LO} + f_{Rx})t] + \cos[(f_{LO} - f_{Rx})t] \} \\ &= \frac{1}{2} \{ \cos[(2f_{Rx} + kt)t] + \cos[(kt)t] \} \\ &\Rightarrow \frac{1}{2} \cos[(kt)t] \quad (\text{after filtering}) \end{aligned}$$

where signal amplitudes have been neglected. In practice, there are often several propagating modes present, and in such a case mixing the receive local oscillator with the received signal will produce a signal which is a sum of different frequencies, each of value  $kt_i$  where  $t_i$  is the propagation delay of propagating mode  $i$ . By grouping the time series data of a complete HF band sweep into small data blocks, and performing spectral estimation on the data blocks, the  $kt_i$  values for each propagating mode can be determined at regular frequency intervals across the HF band. If the transmit signal is not exactly synchronized to the receive local oscillator then a constant error will be present in the  $kt_i$  values across the band of frequencies swept through. Any difference between the transmit signal sweep rate and the receive local oscillator sweep rate will introduce an error in the  $kt_i$  values that increases with sweep frequency. Because the spectral data represents the group delays of propagating modes present, this data will be referred to as *group delay FFT data*. An ionogram can be generated from the group delay



## 6.2 Oblique FMCW ionosonde theory

---

FFT data. Such a plot shows the different modes of propagation present at each frequency, the corresponding group delays and attenuation levels.

Determining the propagating modes present is achieved by applying a peak detection algorithm to the spectral data. The propagation delay and path length for each mode can then be calculated. The following equations are used for these calculations.

An FFT frequency  $f_{FFT}$  is a function of the FFT index  $i_{FFT}$ , the total number of FFT points  $N_{FFT}$ , and the sampling rate  $f_s$

$$f_{FFT} = \frac{i_{FFT} f_s}{N_{FFT}}. \quad (6.1)$$

For complex time series data the Nyquist sampling requirement is that  $f_s > B$ , where  $B$  is the bandwidth of the data. Propagation delay  $t$  is given by

$$t = \frac{f_{FFT}}{k}. \quad (6.2)$$

Path length  $d$  can be calculated from the propagation delay using the equation

$$d = tv$$

where  $v = f\lambda$  is the speed of the propagating wave with frequency  $f$  and wavelength  $\lambda$ . Because the ionosphere is a refracting medium,  $v$  and  $\lambda$  of the signal will change over the course of its propagation, whilst  $f$  will remain constant. We can also express the path length in terms of number of wavelengths  $d_\lambda$  as

$$d_\lambda = \frac{tv}{\lambda} = tf. \quad (6.3)$$

Note also that  $f$  is related to ionogram block index  $i_b$  according to the equation

$$f = f_o + (b/2)(1/f_s)k + (b - \Delta)(1/f_s)k(i_b - 1) \quad (6.4)$$

where  $f_o$  is the initial frequency,  $b$  is the block size,  $\Delta$  is the block overlap size, and  $f$  is taken to be exact at the middle of the data block. The signal  $s_{i_b,i}$  for propagating mode  $i$  at an ionogram block index  $i_b$  is generated using amplitude  $A_{i_b,i}$ , frequency  $f_{i_b,i}$ , and phase shift  $\exp(j2\pi d_{\lambda_{i_b,i}})$

$$s_{i_b,i} = A_{i_b,i} \exp(j2\pi f_{i_b,i} t) \exp(j2\pi d_{\lambda_{i_b,i}}). \quad (6.5)$$

### 6.3 Path Length Resolution

The oblique FMCW ionosonde equations (6.1)–(6.5) are used to generate signals for use in the calculation of antenna and mode correlation from recorded HF radio data. The path length resolution of the signals generated using these equations impacts on correlation calculations. If the path length resolution is equal to one wavelength, each pair of generated signals has a constant phase relationship, resulting in a correlation of one. To obtain accurate correlation values, the path length resolution should be much less than a wavelength. The following equations show the relationship between FFT resolution and path length resolution.

FFT resolution,  $\Delta_{FFT}$ , is given by the equation

$$\Delta_{FFT} = \frac{f_s}{N_{FFT}}. \quad (6.6)$$

Group delay resolution  $\Delta_t$  and path length resolution  $\Delta_{d_\lambda}$  can be obtained from  $\Delta_{FFT}$  using the following equations

$$\Delta_t = \frac{\Delta_{FFT}}{k} = \frac{f_s}{N_{FFT} k} \quad (6.7)$$

$$\Delta_{d_\lambda} = \frac{\Delta_t v}{\lambda} = \Delta_t f = \frac{f_s f}{N_{FFT} k}. \quad (6.8)$$

If we impose a requirement  $\Delta_{d_\lambda} \ll \lambda$ , then the requirement for  $N_{FFT}$  is

$$N_{FFT} \gg \frac{f_s f_{max}}{k} \quad (6.9)$$

where  $f_{max}$  is the maximum frequency swept through. For our case,  $f_s = 868$  samples per second (S/s),  $f_{max} = 25$  MHz, and  $k = 125$  kHz/s, corresponding to a requirement  $N_{FFT} \gg 2^{18}$ . Antenna and mode correlation calculations were performed with  $N_{FFT} = 2^{20}$ , corresponding to a  $\Delta_{d_\lambda} = 0.17\lambda$ . Note that while increasing  $N_{FFT}$  gives an improved estimate of the location of each detected frequency, spectral resolution does not improve, and is instead dependent on the time series data block size used for each FFT.

### 6.4 Post-processing of HF Radio Data

Ionograms were generated from recorded HF radio data by performing spectral estimation on small blocks of time series data, as discussed in Section 6.2. A constant error in sweep rate

## 6.4 Post-processing of HF Radio Data

---

between the transmit signal and the receive local oscillator was found to be present, evidenced by the propagating mode with lowest delay (the bottom line on an ionogram) showing a propagation delay increase with frequency. This mode generally has a constant propagation delay across the band of frequencies it covers. The difference between transmit signal and receive local oscillator sweep rates introduces propagation delay errors which increase with frequency. The receive data was multiplied by a sweep error compensation signal to account for the error. In addition it was found that the received signal frequencies representing propagation delays occupied only a small portion of the 5 kHz receiver bandwidth. For lower processing requirements and better FFT frequency resolution, decimation filtering around the frequencies of interest was performed.

An example ionogram without any post-processing is shown in Figure 6.6. In the middle of the diagram we see a dark patch, which corresponds to the cut-off region of the receiver FIR (finite impulse response) filters. The time series data being analysed is complex, formed by adding the in-phase signal and the quadrature signal multiplied by  $j$ . For this reason received signal frequencies appear only in the region 0 to  $B$ , where  $B$  is the bandwidth of the receiver FIRs. Because the receiver FIRs are real, noise appears in the region  $f_s - B$  to  $f_s$ . For the case of complex data we note that  $f_s$  can be reduced by a factor of 2 without any aliasing problems.

A number of ionograms were surveyed in order to determine an appropriate sweep error compensation signal, with  $\exp(-j2\pi 0.725t^2)$  selected. The example ionogram after sweep error compensation is shown in Figure 6.7.

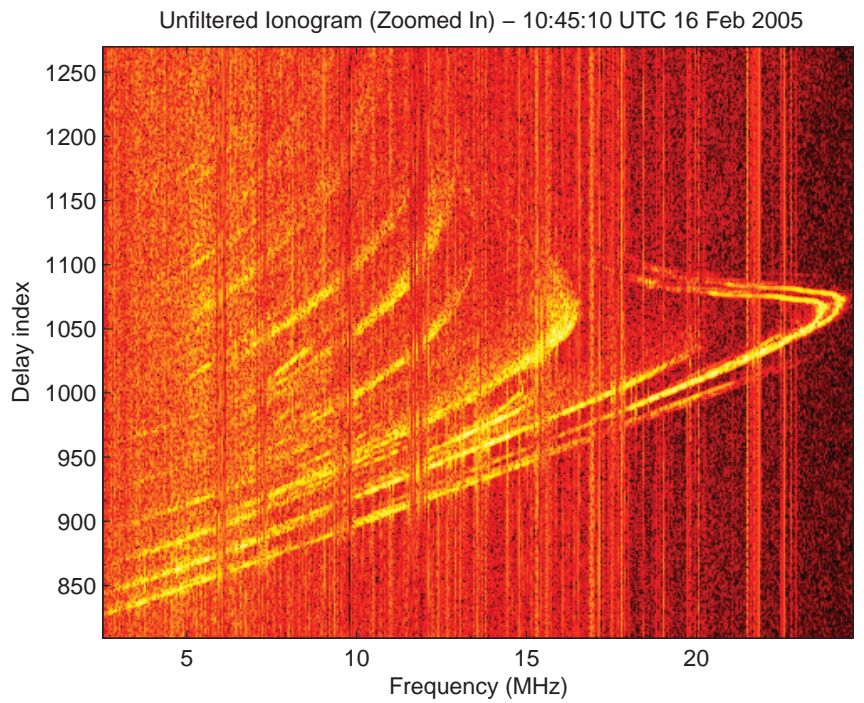
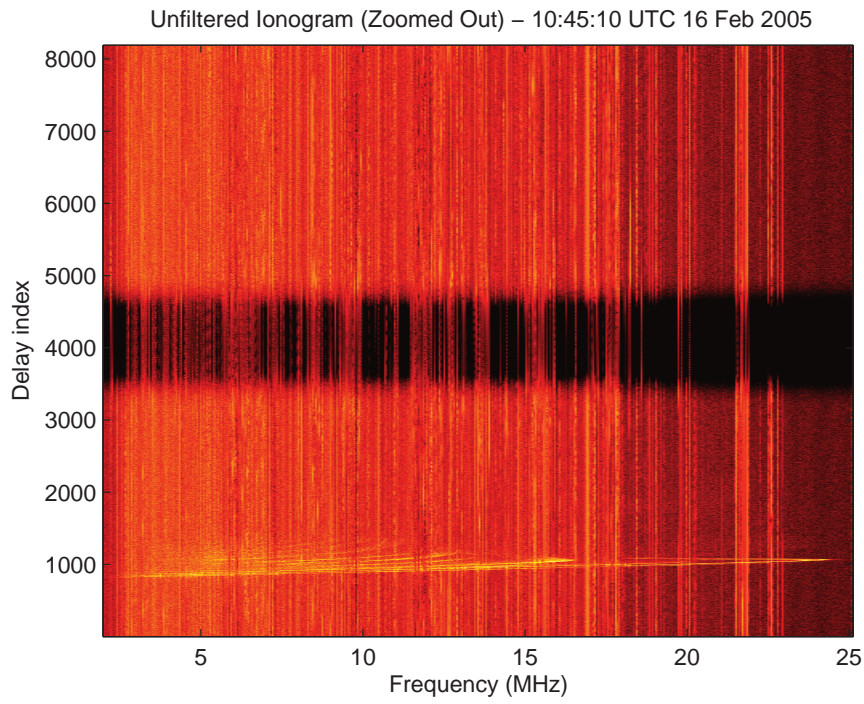
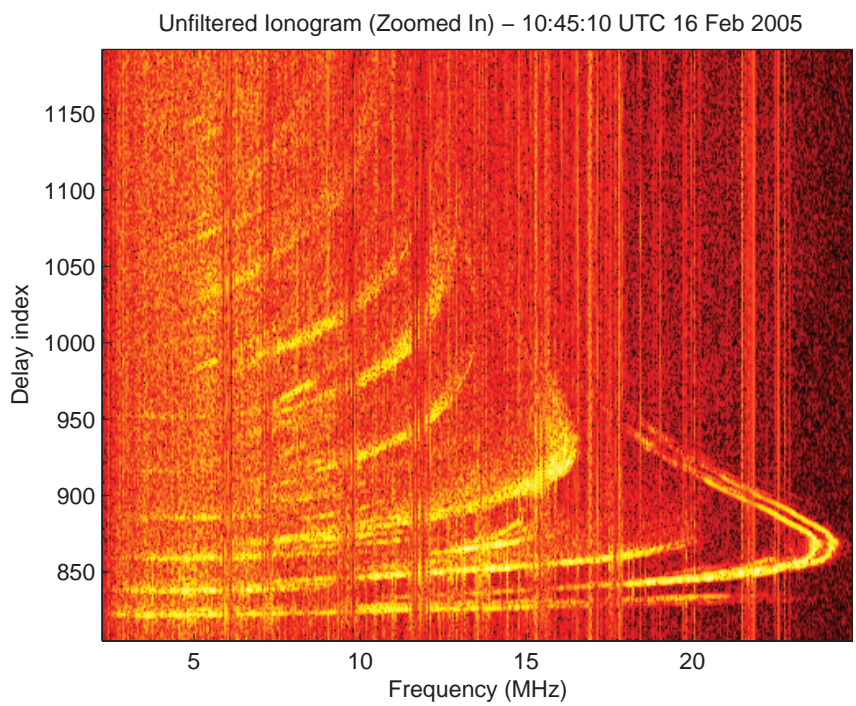


Figure 6.6. An example ionogram without any post processing



**Figure 6.7.** The example ionogram following sweep error compensation

Decimation filtering around the frequencies of interest was performed by multiplying the receiver data by a constant frequency signal to shift the frequencies of interest close to zero frequency, or *baseband*, and then applying a low pass FIR filter. A baseband conversion signal of  $\exp(-j2\pi 990t)$  was used. FIR coefficients were generated using `fdatool` in Matlab, with the filter parameters given in Table 6.1. A conservative FIR bandwidth of 700 Hz was set to accommodate modes with unusually long propagation delays. The data was then resampled at approximately  $f_s = 868$  S/s which corresponds to a post processing decimation factor of 12.

<b>Sampling rate</b>	10417 S/s
<b>Passband ripple</b>	0.01 dB
<b>Passband frequency</b>	700 Hz
<b>Stopband frequency</b>	800 Hz
<b>Stopband attenuation</b>	160 dB

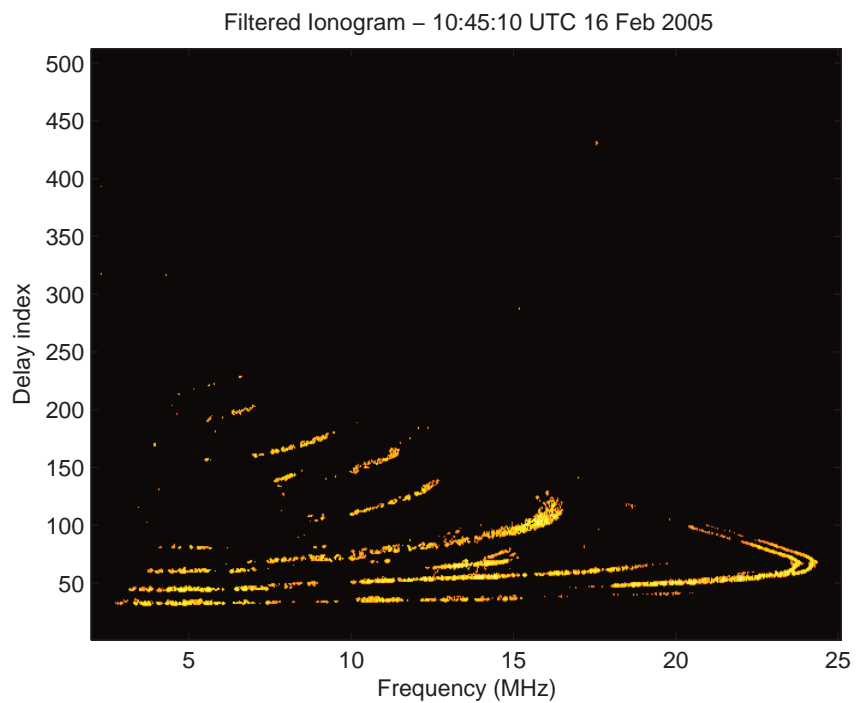
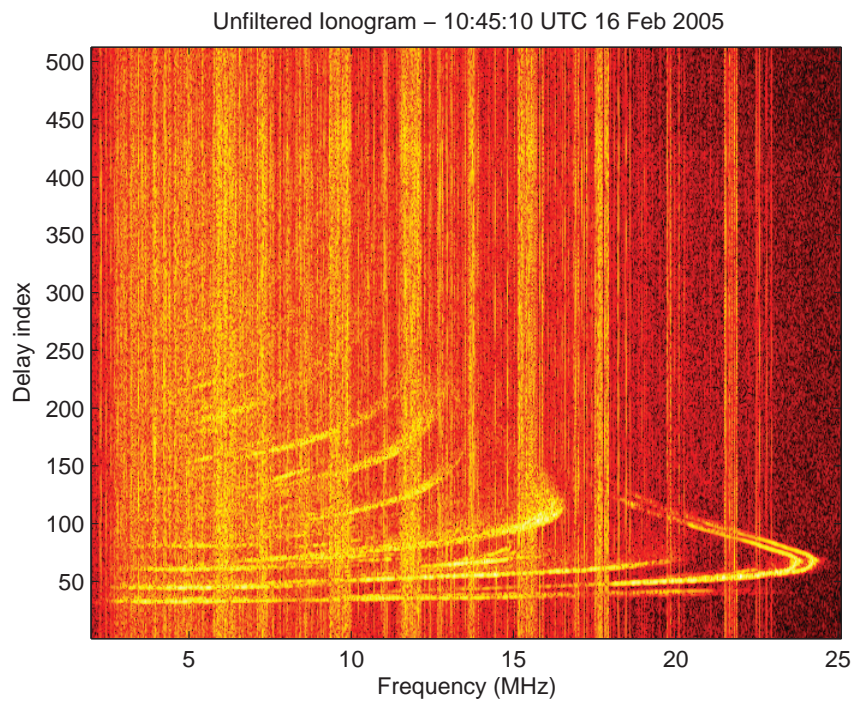
**Table 6.1.** Table of parameters used to generate the coefficients for the post-processing decimation FIR filter

Additional filtering can be provided using a basic image processing technique which involves the removal of points without a given number of neighbouring points within a defined search radius. Figure 6.8 shows the final post processed ionogram, both with and without the additional image processing filtering.



## 6.4 Post-processing of HF Radio Data

---

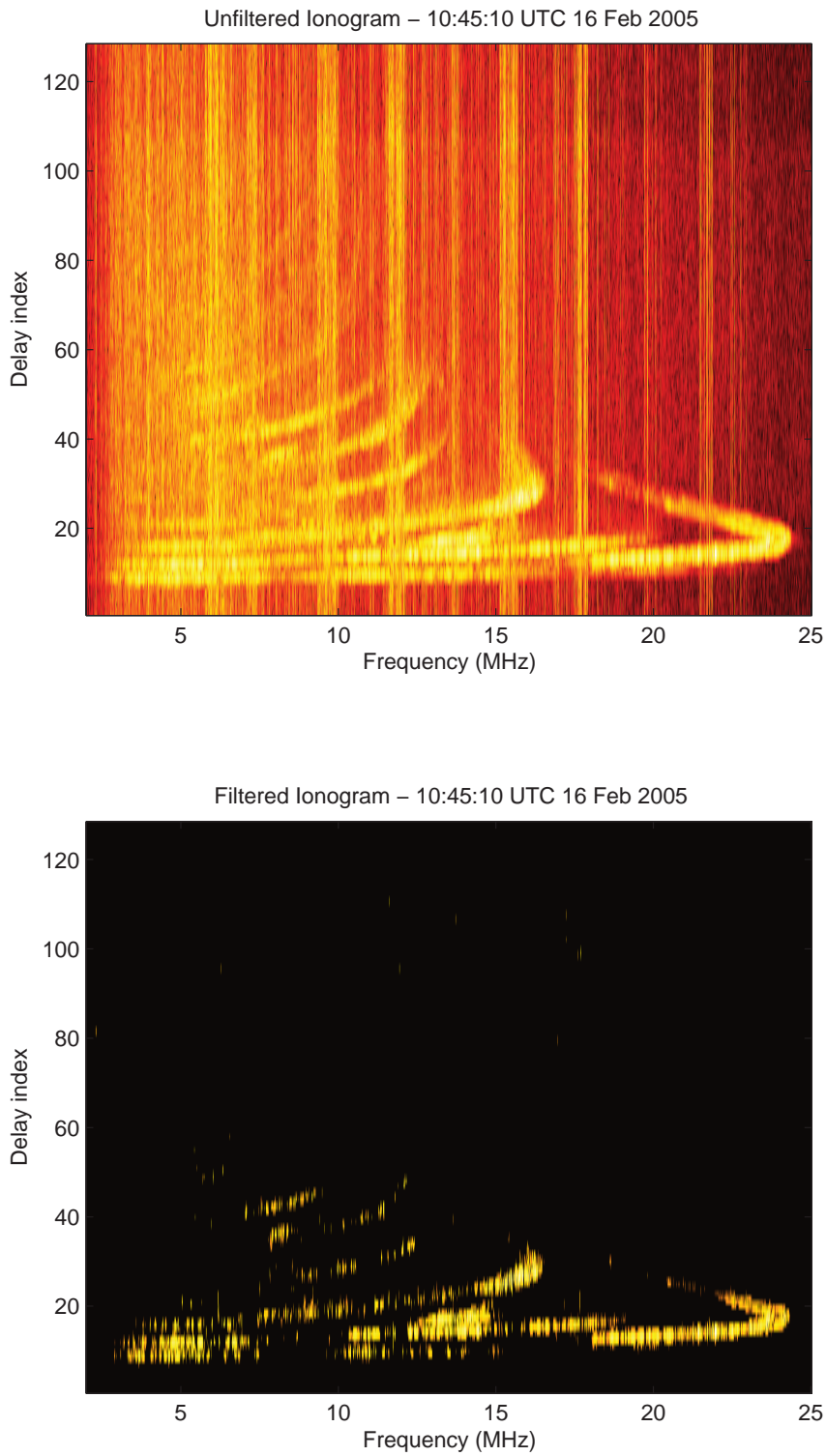


**Figure 6.8.** Final post processed ionogram

A further consideration is the selection of time series data block size to use in generating the group delay FFTs. A larger block size improves group delay resolution at the cost of ionogram frequency resolution, while a smaller block size improves ionogram frequency resolution at the cost of group delay resolution. This effect can be seen in Figures 6.9–6.14. A block size of 512 was selected which offers a reasonable compromise between group delay resolution and ionogram frequency resolution.

## 6.4 Post-processing of HF Radio Data

---



**Figure 6.9.** The example ionogram for a block size of 128

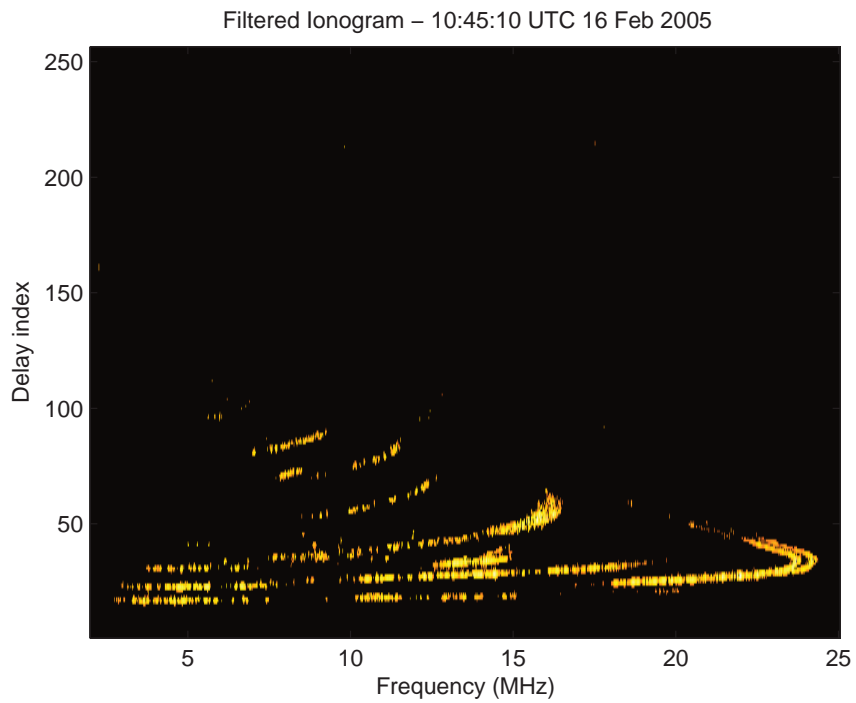
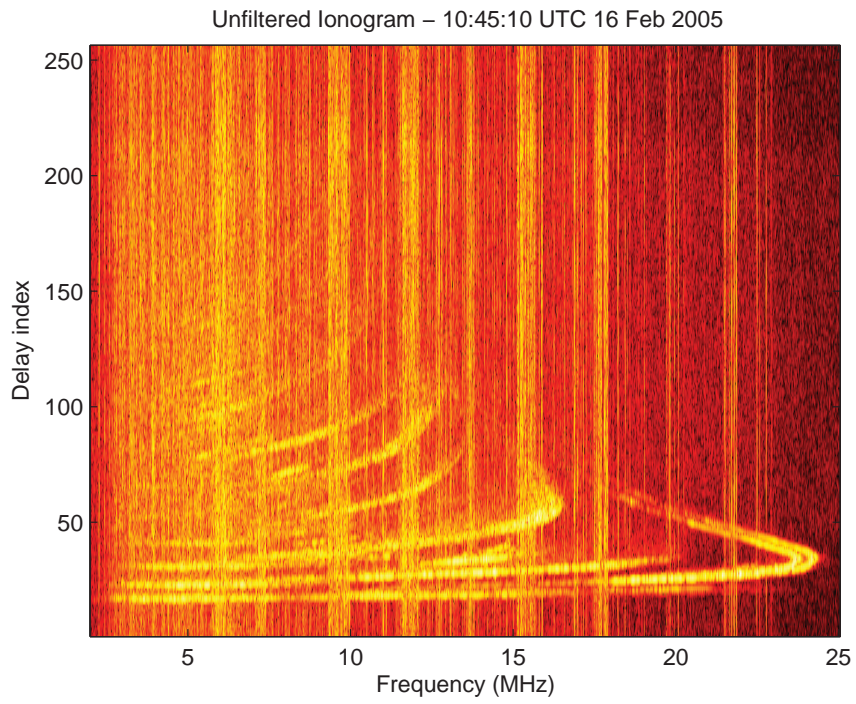
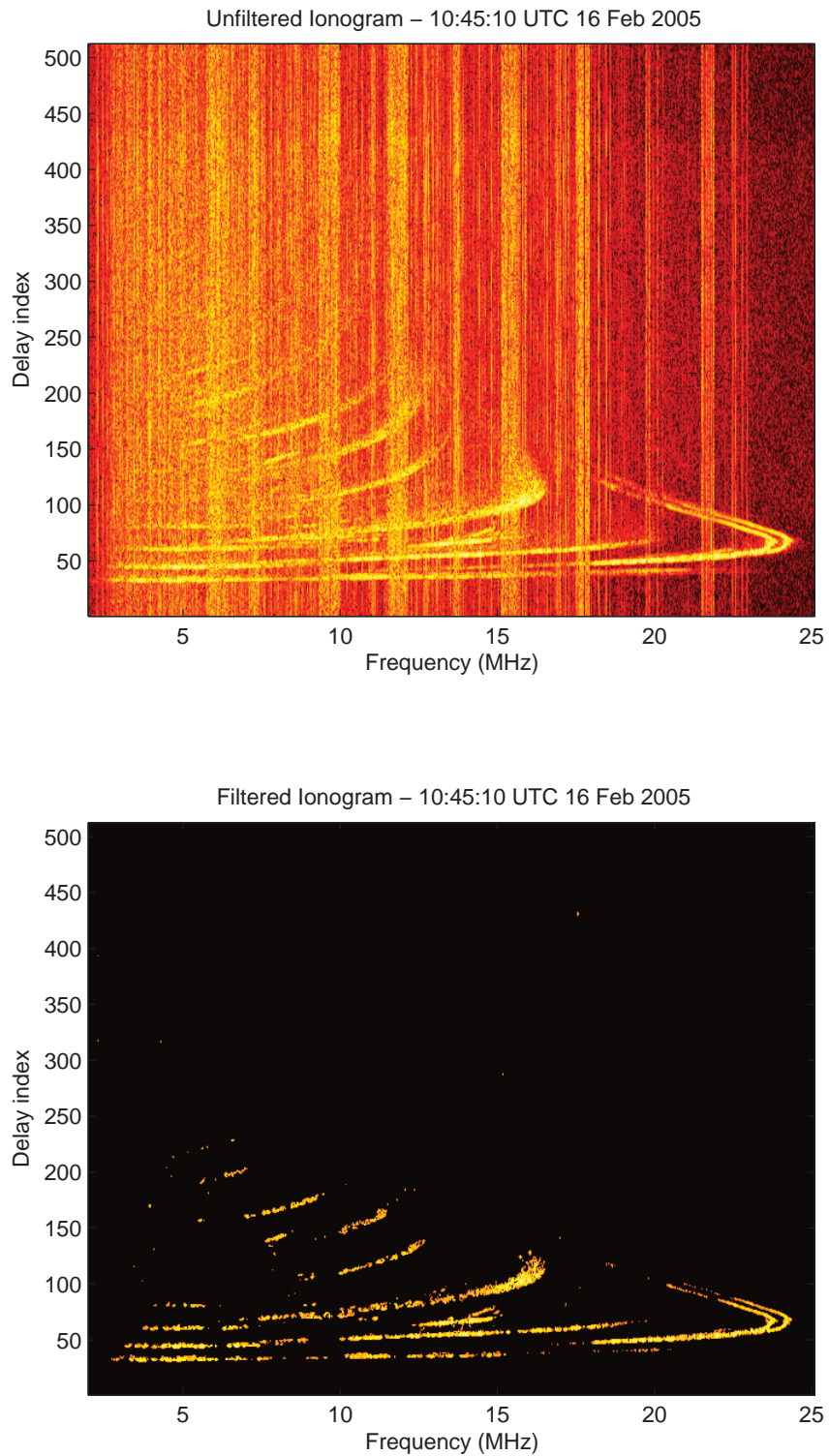


Figure 6.10. The example ionogram for a block size of 256

## 6.4 Post-processing of HF Radio Data

---



**Figure 6.11.** The example ionogram for a block size of 512



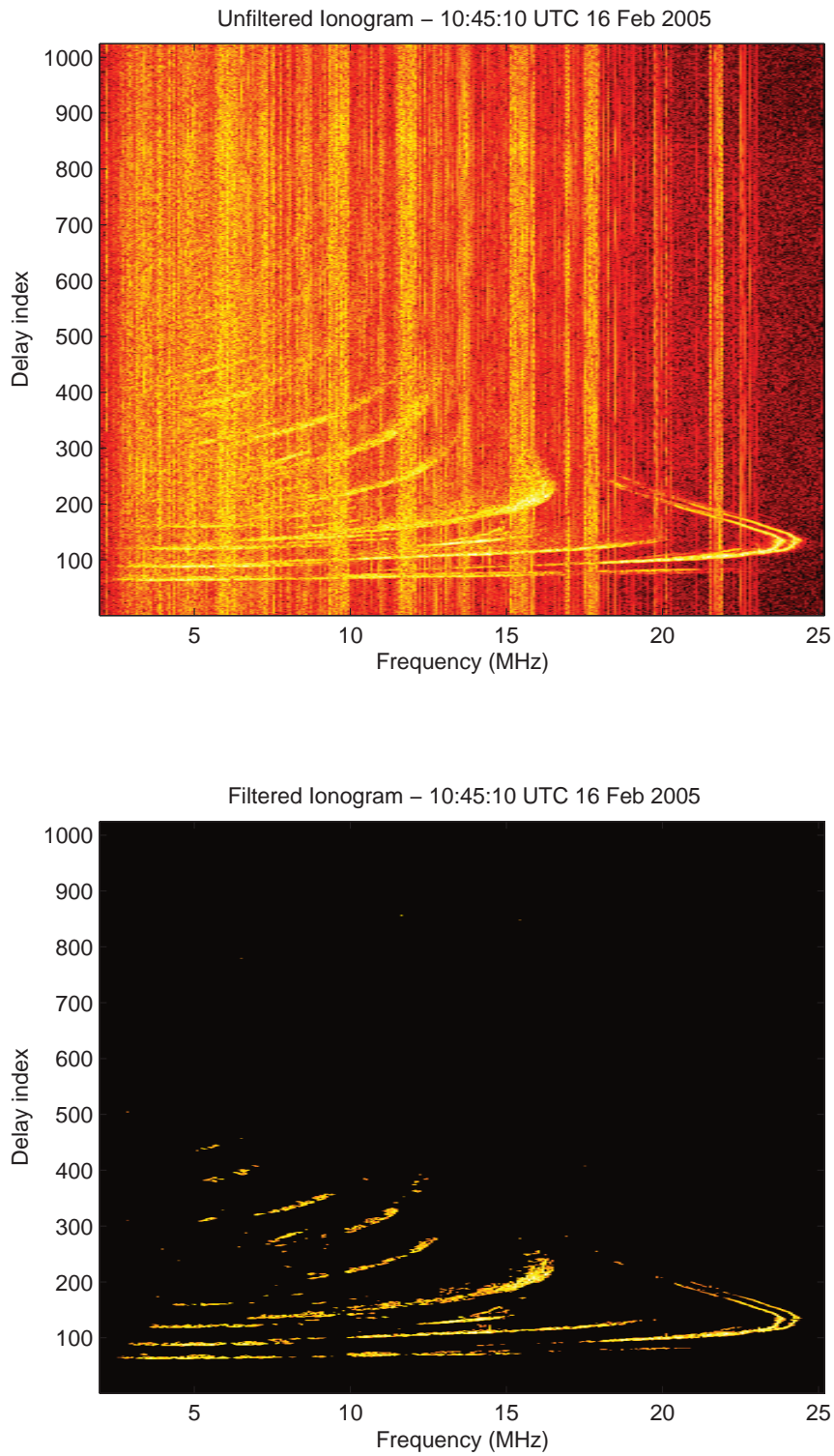
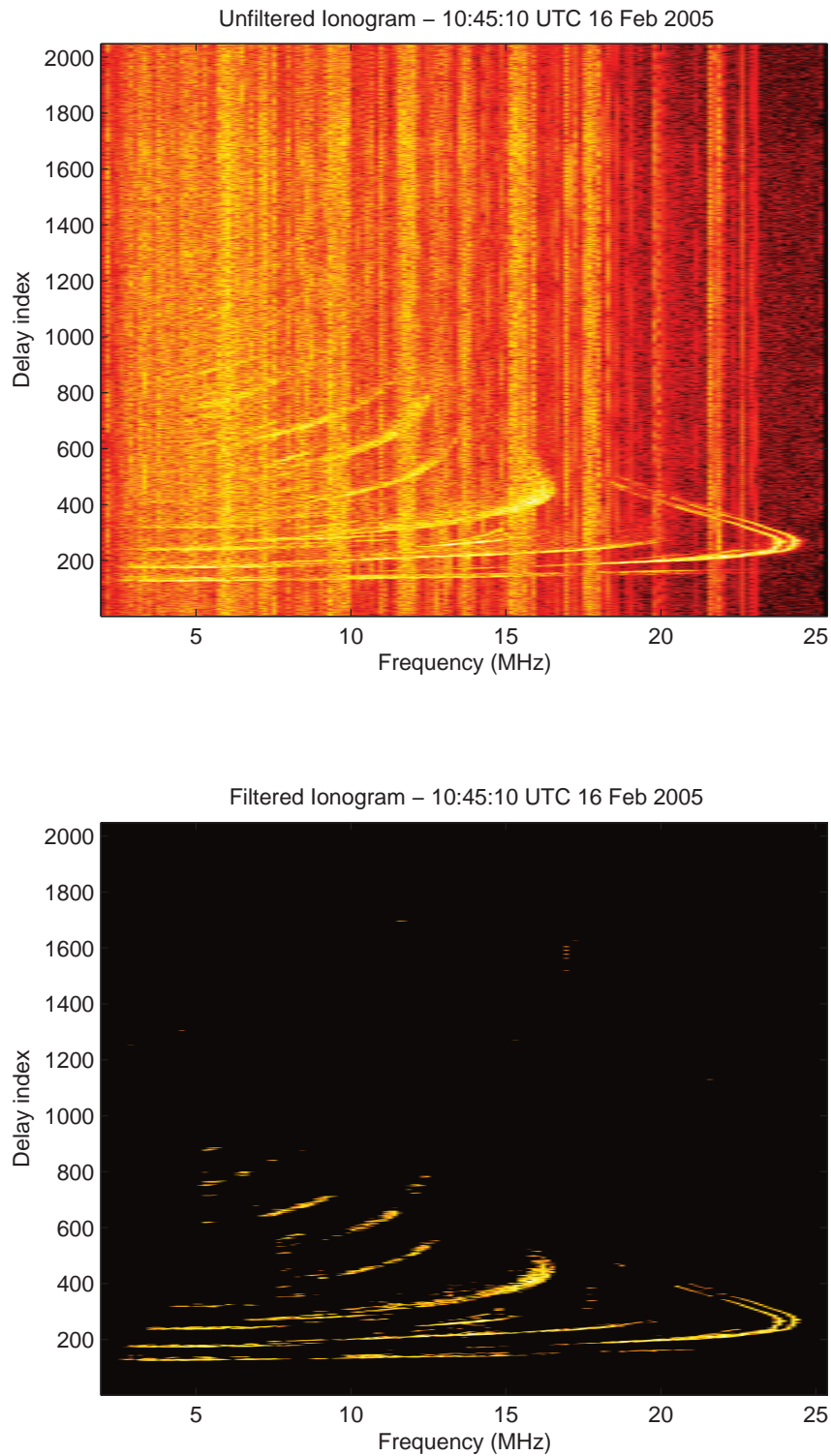


Figure 6.12. The example ionogram for a block size of 1024



## 6.4 Post-processing of HF Radio Data

---



**Figure 6.13.** The example ionogram for a block size of 2048

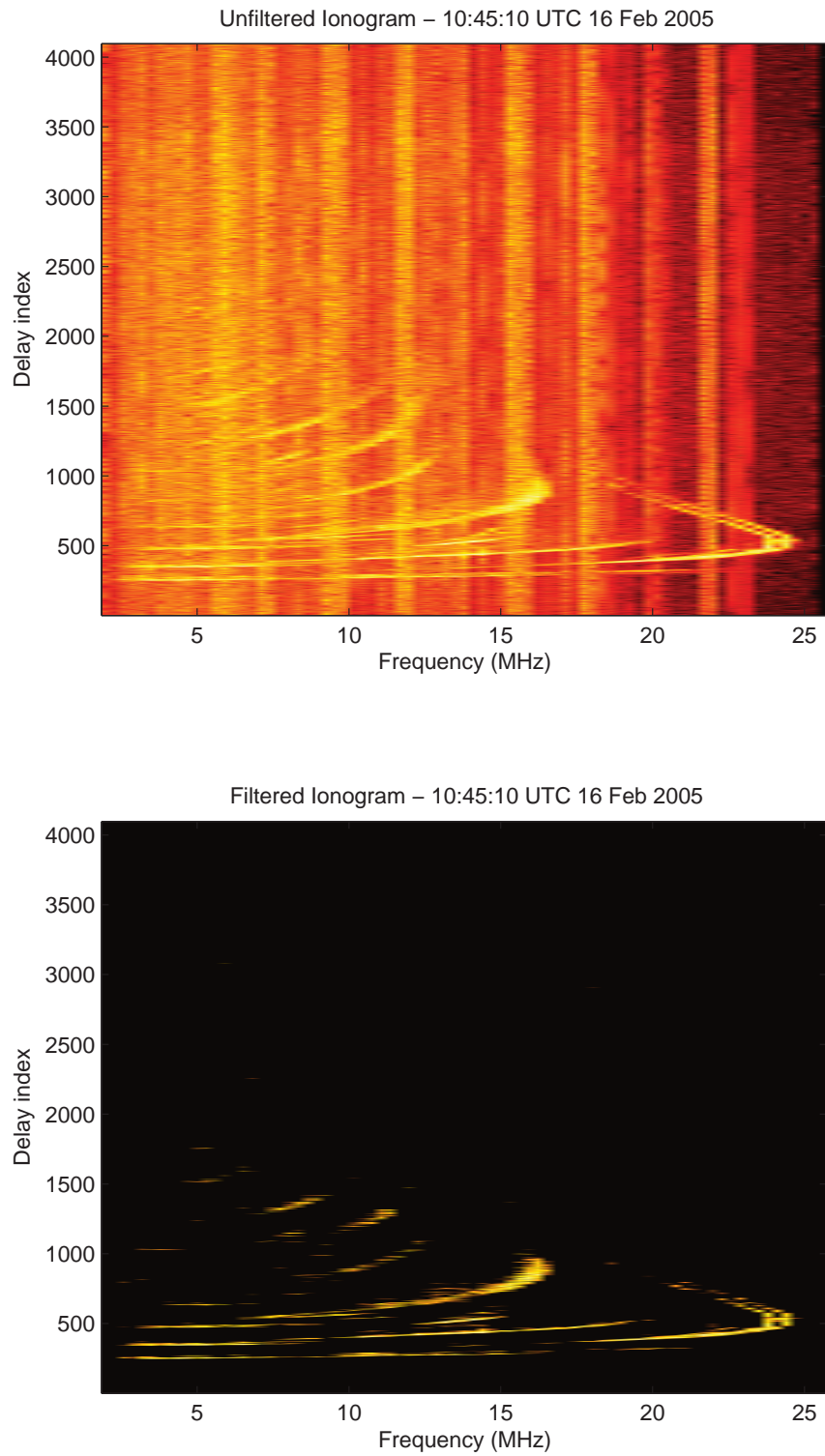


Figure 6.14. The example ionogram for a block size of 4096

## 6.5 Peak Detection

---

A peak detection algorithm was used to determine the propagating modes present in each block of group delay FFT data. Details of the algorithm are provided in Figures 6.15–6.17.

1. Set 'gap' parameter to a small fraction of NFFT to detect closely spaced peaks.

```
NFFT = 2^20;  
gap = 625;
```

2. Set 'threshold' parameter to the maximum of:
  - maximum of the noise/interference block psd + 1 dB
  - mean of the data block psd + 5 dB

```
threshold = max((max(int_block_psd)+1),(mean(data_block_psd)+5));
```

3. Set 'peaks' parameter to the number of data block psd points above 'threshold' divided by 'gap'.

```
peaks = length(find(data_block_psd>=threshold))/gap;
```

4. Apply 'threshold' to data block psd. All values below 'threshold' are set to the minimum value.

```
minimum = min(data_block_psd);  
data_block_psd2 = data_block_psd;  
data_block_psd2(find(data_block_psd2 < threshold)) = minimum;
```

data\_block\_psd2 plot:

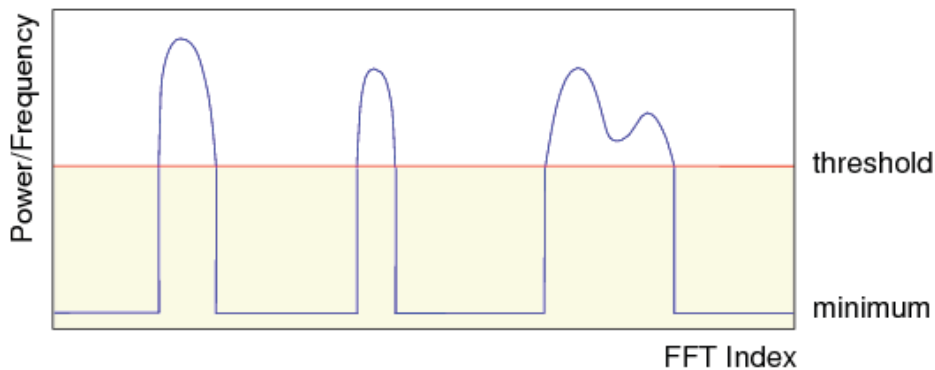


Figure 6.15. Peak detection algorithm steps 1-4

5. Iterate peaks times, each time calculating the maximum group delay FFT value and corresponding index, and zeroing the region around the peak, from FFT index  $\text{max\_i} - \text{gap}$  to  $\text{max\_i} + \text{gap}$ , such that the next largest peak can be detected on the next iteration.

```
for i = 1:peaks
    [max_val, max_i] = max(data_block_psd2);
    data_block_psd2(max_i-gap:max_i+gap) = minimum;
    if max_val = minimum
        peaks = i - 1;
        break; % all peaks detected
    else
        max_index(i) = max_i;
    end
end
```

max\_index values plot:

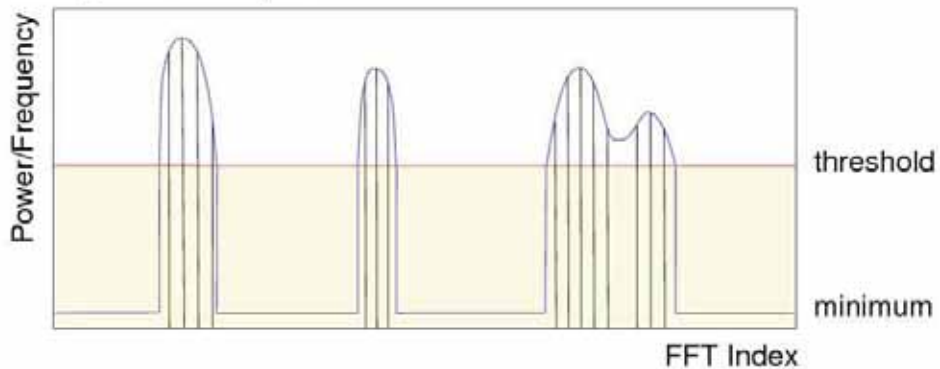


Figure 6.16. Peak detection algorithm step 5

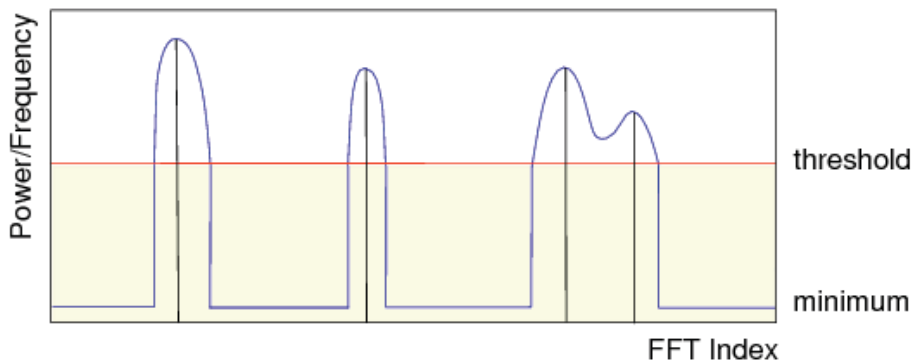
## 6.5 Peak Detection

---

6. Ensure detected peak is a local maximum

```
peaksrem = 0;
for i = 1:peaks
    if ( data_block_psd(max_index(i) - 1) > data_block_psd(max_index(i)) |
        data_block_psd(max_index(i) + 1) > data_block_psd(max_index(i)) )
        max_index(i) = length(data_block_psd) + 1;
        peaksrem = peaksrem + 1;
    end
end
max_index = sort(max_index)
peaks = peaks - peaksrem;
max_index = max_index(1:peaks);
```

max\_index values plot following local maximum check:



**Figure 6.17.** Peak detection algorithm step 6

---

## 6.6 Spectral Estimation and Windowing

---

Analysis of the recorded HF radio data involved dividing the data into small blocks and performing spectral estimation on the blocks. In this section issues concerning spectral estimation and windowing are considered, and different spectral estimation and windowing techniques are compared. The following useful resources were consulted [79], [80], [81], [82].

### 6.6.1 Windowing

The DFT (discrete Fourier transform) considers a sampled data sequence to be periodic. Periodic extension of such a sequence is achieved by replicating the sequence in both negative and positive time. If the finite duration interval over which the sequence is sampled is not an integer multiple of the period of a signal which is present, discontinuities are introduced at the boundaries of the replicated sequence. These discontinuities give rise to the phenomenon of spectral leakage. Windowing of data blocks is a technique used to reduce spectral leakage. Figure 6.18 provides a comparison between an unwindowed and a windowed signal. From this plot we see that the amplitude of the windowed signal is reduced at the boundaries of the set of samples.

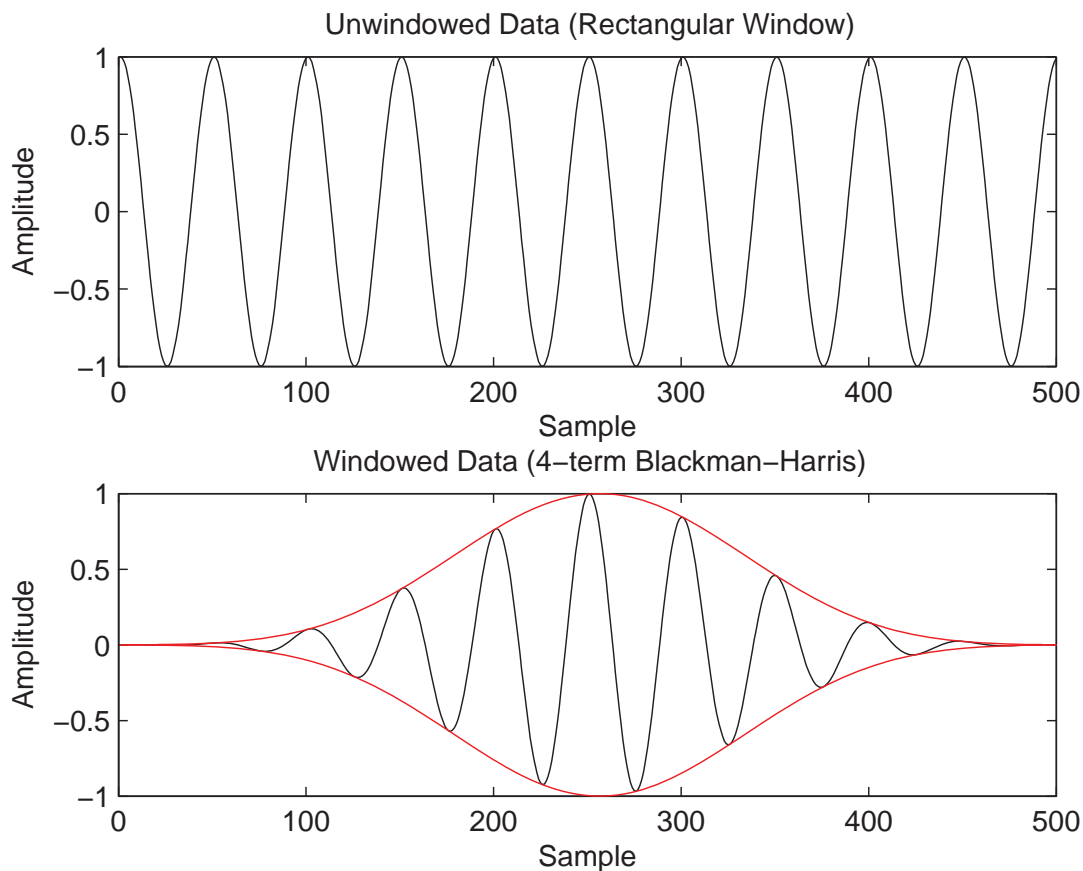
There are many windows to choose from, each with varying main lobe widths and sidelobe amplitudes. There is a trade off between main lobe width and sidelobe amplitude. A window with a wide main lobe will generally have low sidelobes, while a window with a narrow main lobe will generally have higher sidelobes. A wide main lobe makes it harder to detect two closely spaced frequencies, while high sidelobes are detected as peaks if they are higher than the peak detection threshold. Commonly used windows include Hanning, 4-term Blackman-Harris, and Kaiser-Bessel. The Hanning window has a fairly narrow main lobe, and moderate sidelobes, while the 4-term Blackman-Harris window has a wider main lobe than the Hanning window, but considerably lower sidelobes. The Kaiser-Bessel window is an example of a parameterized window, which uses the parameter  $\beta$  to trade off between main lobe width and sidelobe level. The optimal window choice for a set of data is the window with the narrowest main lobe that has sidelobes below the noise level.

To test out different windows, a block of test data comprising two closely spaced frequencies was generated, and sampled at  $f_s = 868$  S/s. The spacing of the frequencies was adjusted to investigate the spectral resolution of each window. To determine the sidelobe attenuation required, data recorded for a single sweep of the HF band was analysed, and the maximum



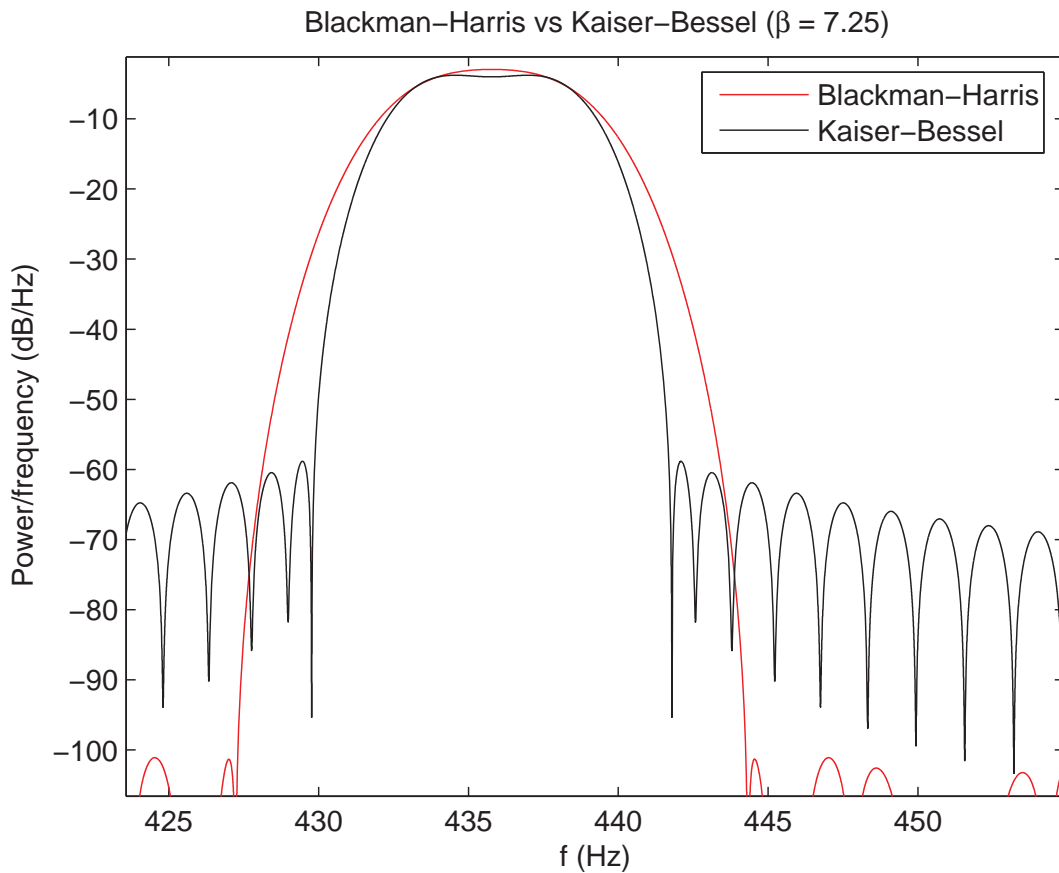
## 6.6 Spectral Estimation and Windowing

---



**Figure 6.18.** A comparison between an unwindowed and a windowed signal

difference between peaks and the peak threshold level calculated. A value of 43 dB was calculated, and a conservative target sidelobe attenuation of 53 dB was set. A rectangular window was found to give a sidelobe attenuation of 13 dB, while the Hanning window gave 32 dB and the 4-term Blackman-Harris 98 dB. The Kaiser-Bessel window was tuned to match the desired sidelobe attenuation of 53 dB, with a corresponding parameter  $\beta = 7.25$ . Figure 6.19 shows a comparison between the 4-term Blackman-Harris and Kaiser-Bessel ( $\beta = 7.25$ ) windows for two frequencies 3.5 Hz apart. The two frequencies can be distinguished using the Kaiser-Bessel window, but not using the 4-term Blackman-Harris window. The ability to tune the Kaiser-Bessel window to suit the recorded HF radio data make it the best choice from the set of windows considered.



**Figure 6.19.** The 4-term Blackman Harris versus Kaiser-Bessel ( $\beta = 7.25$ ) window for two frequencies 3.5 Hz apart. The two frequencies can (just) be distinguished when the Kaiser-Bessel window is used.

## 6.6.2 Spectral Estimation

There are many spectral estimation techniques offered in Matlab in addition to the standard FFT or periodogram technique, and these are listed in Table 6.2.

To test out the different spectral estimation techniques available, a block of synthetic test data was generated which comprised 12 unit amplitude complex signals with increasing frequency separations, and with frequencies constrained to between 100 and 650 Hz. This data was sampled at  $f_s = 868$  S/s. Unit amplitude complex noise was also added, to give an SNR of 0 dB for each of the 12 frequency components. This corresponded to a dynamic range of approximately 30 dB, similar to that of the recorded HF radio data.

## 6.6 Spectral Estimation and Windowing

---

Matlab function	Name of spectral estimation technique
<code>periodogram</code>	Periodogram method
<code>pwelch</code>	Welch's method
<code>pmtm</code>	Thomson multitaper method
<code>pburg</code>	Burg's method
<code>pcov</code>	Covariance method
<code>pmcov</code>	Modified covariance method
<code>pyulear</code>	Yule-Walker autoregressive method
<code>peig</code>	Eigenvector method
<code>pmusic</code>	MUSIC method

**Table 6.2.** Table of spectral estimation techniques available in Matlab

The PSD (power spectral density) generated using the modified periodogram function<sup>8</sup> with a Kaiser-Bessel window with  $\beta = 7.25$  is shown in Figure 6.20. The plot shows that the periodogram has a spectral resolution of around 4-8 Hz and gives a good estimate of the frequency locations and power content of each frequency.

One problem with the periodogram is the variance of the resulting PSD estimate. The variance of the estimate can be reduced using `pwelch`, which partitions the block of data to be analysed into segments, performs a modified periodogram on each segment, and then averages the PSD estimates to give an overall PSD estimate. The variance is inversely proportional to the number of data segments used, but at the same time spectral resolution degrades as more data segments are used, so there is a tradeoff between PSD variance and spectral resolution. Figure 6.21 shows the PSD generated using the `pwelch` function with a Kaiser-Bessel window with  $\beta = 7.25$ , and a partition value of 4. The plot clearly shows the reduced spectral resolution compared to the periodogram plot given in Figure 6.20.

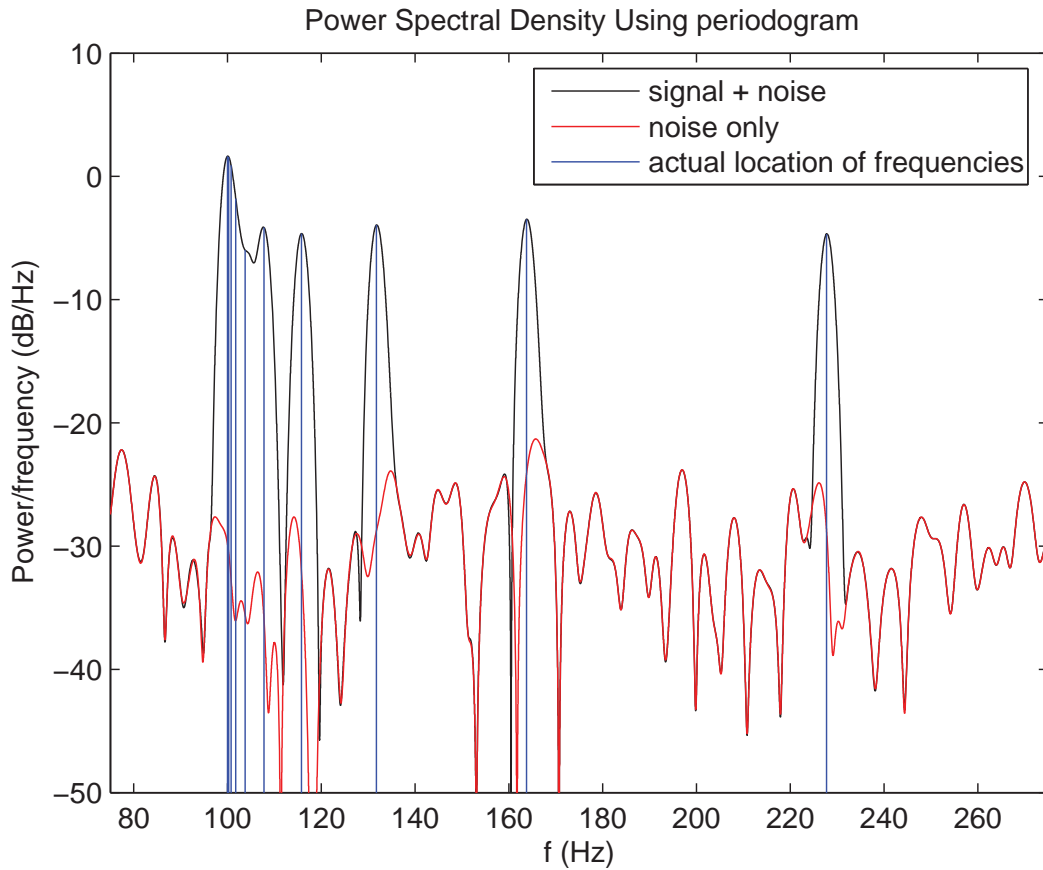
The other spectral estimation functions listed in Table 6.2 were also used on the test block of data. Functions such as `pmcov`, `peig`, and `pmusic` gave improved spectral resolution over the periodogram function, but also gave poor estimates of the power content of each frequency. The PSD plot obtained using the `pmcov` function is shown in Figure 6.22.

The periodogram function and a Kaiser-Bessel window with  $\beta = 7.25$  was selected for the processing of the recorded HF radio data, since this combination offers good spectral resolution and a good estimate of the power content of each frequency. The ability to tune the main lobe

---

<sup>8</sup>A modified periodogram differs from a conventional periodogram in that a (non-rectangular) window is first applied to the input signal.

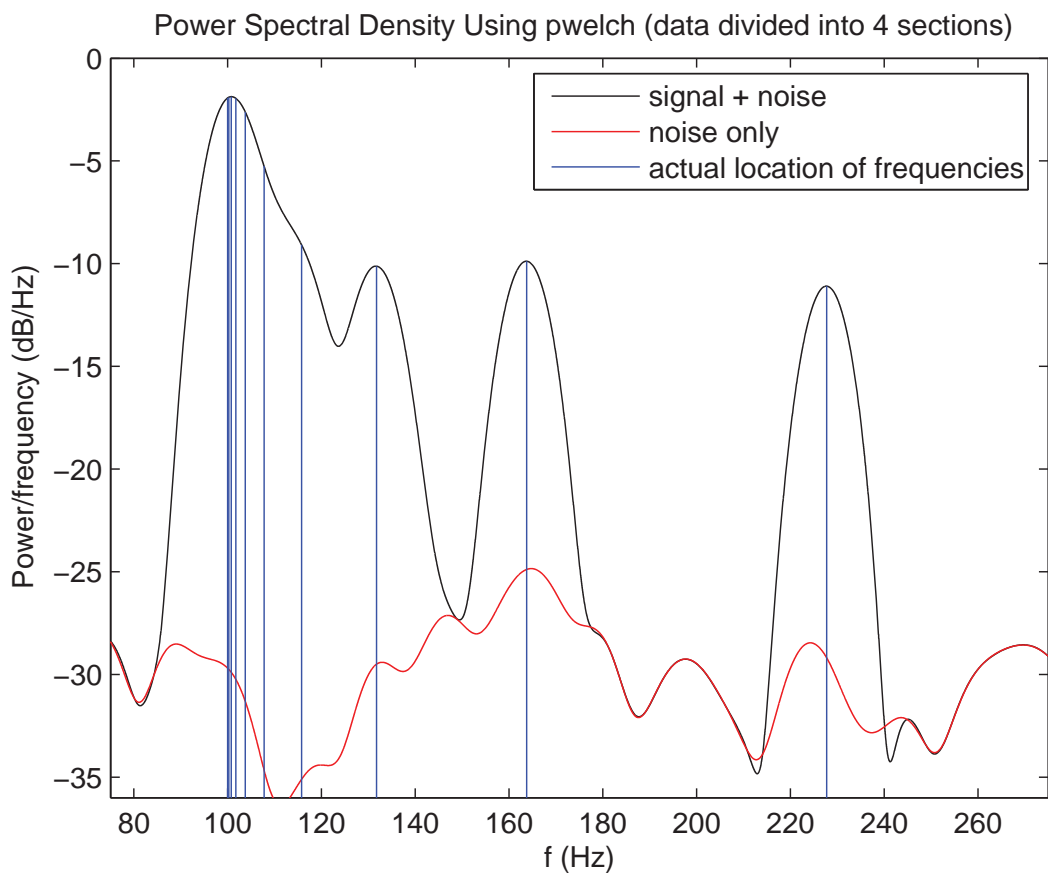
width and the side lobe attenuation of the Kaiser-Bessel window gave it a significant advantage over other windows such as the Hanning and Blackman-Harris.



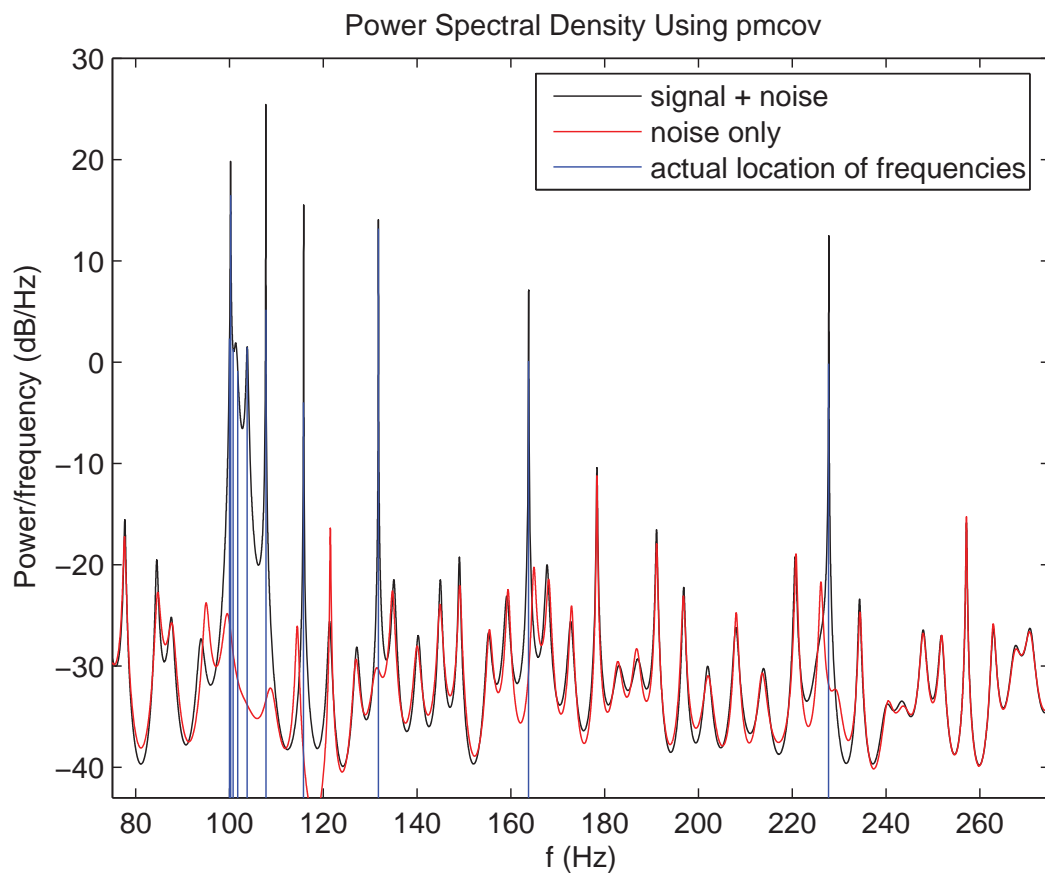
**Figure 6.20.** A zoomed in plot of the PSD of the synthetic test data generated using periodogram

## 6.6 Spectral Estimation and Windowing

---



**Figure 6.21.** A zoomed in plot of the PSD of the synthetic test data generated using `pwelch`



**Figure 6.22.** A zoomed in plot of the PSD of the synthetic test data generated using `pmcov`



### 6.7 SNR Measurement

---

Measurement of SNR, which appears in the general MIMO capacity equation (2.8), is another issue requiring consideration.

To investigate the measurement of SNR from spectral data, a block of synthetic test data was generated from a single unit amplitude complex signal and unit amplitude complex noise, to give an SNR of 0 dB. This data was sampled at  $f_s = 868$  S/s. The PSD of both the signal plus noise data, and the noise only data was generated using the periodogram function and a Kaiser-Bessel window with  $\beta = 7.25$ . To measure the power  $P$  contained in a data block from the PSD, the PSD needs to be integrated over the band of frequencies the signal covers. The integral can be calculated simply by using the rectangular approximation to give

$$P = \Delta_{FFT} \sum_{i=1}^{NFFT} P_{xx_i} = \frac{f_s}{NFFT} \sum_{i=1}^{NFFT} P_{xx_i}$$

where  $P_{xx_i}$  are the elements of the PSD vector  $P_{xx}$ .

Denoting the power of the signal plus noise  $P_{S+N}$ , the power of the signal  $P_S$ , and noise power  $P_N$ , the SNR in dB is given by

$$\text{SNR} = 10 \log_{10} \frac{P_S}{P_N} = 10 \log_{10} \frac{P_{S+N} - P_N}{P_N}.$$

It is important to note that the window used affects power measurements, so the window values need to be scaled by the coherent gain of the window, and the noise power measurement needs to be scaled by the equivalent noise bandwidth of the window. However, because the periodogram function already performs the appropriate scalings, no further steps are required.

Measuring the SNR of the individual signals present in blocks of the recorded HF data is complicated by the fact that peaks may be close together and interfere with one another, and not all the peaks present will be detected. In addition the noise measurement is taken in a band adjacent to the signal band, and so the noise only data will not be the same as the noise contained in the signal plus noise data.

### 6.7.1 Average Mode SNR Calculation - Method 1

The average mode SNR can be calculated using

$$\text{Average mode SNR} = 10 \log_{10} \left( \left( \frac{P_{S+N} - P_N}{P_N} \right) \left( \frac{1}{m} \right) \right)$$

where the bandwidth the SNR is being measured in is equal to the bandwidth of the post-processed data. The bandwidth the SNR is measured in affects the measurement, with a larger bandwidth yielding a larger noise power. For the flat fading assumption to hold, the bandwidth of a MIMO communication system should be set to be lower than the coherence bandwidth, which is inversely proportional to the multipath delay spread. The bandwidth of the post-processed data, 868 Hz, was found to be higher than the typical coherence bandwidth. In addition, negative values of average mode SNR were measured in 868 Hz.

### 6.7.2 Average Mode SNR Calculation - Method 2

Alternatively, the SNR of a mode can be measured in a bandwidth equal to the FFT bin size (1.69 Hz in our case), by subtracting the value of the signal plus noise PSD at the index of a detected mode, from the value of the noise only PSD at the same index. The average mode SNR can be found by averaging the SNRs for each of the detected modes. Capacity results presented in this thesis were calculated for SNRs measured using this approach. By measuring SNR in a small bandwidth, a large SNR value is obtained. In Figure 2.2 of Section 2.2.4, increasing SNR was shown to steepen the slope of the capacity plot. The bandwidth used to measure SNR should always be kept in mind when interpreting capacity results.

## 6.8 Summary

---

In this chapter the experimental setup used to collect HF radio data was described, along with the issues involved with processing and analysing of the recorded HF radio data. Equations provided in Section 6.2 (6.1)–(6.5) are used to generate signals for use in the calculation of antenna and mode correlation from recorded HF radio data. The antenna and mode correlation calculations are discussed in detail in the next chapter.