THE UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

SPECTRAL ANALYSIS OF SEISMIC DATA USING WAVELET TRANSFORM

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

degree of

MASTER OF SCIENCE

By

LI XIA

Norman, Oklahoma

1999

ACKNOWLEGEMENTS

Special thanks to Dr. John P. Castagna for the enormous support through the whole thesis study and for so many instructive advices on my thesis.

I thank Dr. Allen Witten for the careful correction of my thesis, as well as for the important items he pointed out to me, which I had not previously noticed. I am also grateful for Dr. R. Young for his careful review and important suggestions on the improvements of this thesis.

I am also thankful to a lot of researching staffs in the department for the enlightening discussion with me on the wavelet transform.

Table of Contents

List of Figures

Abstract

Spectral analysis provides the frequency information of the data. Fourier tansform projects the data from time domain into frequency domain, while the time information is completely lost after the procedure. Time-frequency analysis is a procedure to localiza the frequency content in time. Wavelet transform is a method to do the time-frequency analysis of the data. Many methods of wavelet transform have been developed such as the continuous wavelet transform, the multiresolution analysis, the wavelet packet and the matching pursuit decomposition. Each method has its own advantages of providing a time-frequency representation of the data. In the thesis, different methods of wavelet transform are introduced and performed on different data. The advantages and the disadvantages of different methods are analyzed so that different methods of wavelet transform could be applied in different situations. In practice, different methods should be used together to provide an good time-frequency representation for different data.

Introduction

Spectral analysis is an important procedure that is applied in many stages of seismic data analysis. The Fourier transform, one of the most fundamental methods in designing filters and in spectral analysis, is used to transform a given time series of data into the sum of a series of sinusoidal components. Traditionally, the 1-D Fourier transform is used to transform the data from the time domain to the frequency domain. However, the result is a global average representation of the frequency content of the time series and loses chronological information completely. It proves to be inadequate in giving a sufficient representation of a data set that has many local features. The ability to localize the frequency content in time is improved by performing the Fourier transform within a sliding short window. However, improvement of time localization can only be obtained by shortening the window length, which in turn decreases the frequency localization. Thus it is very limited in achieving good time and frequency resolution simultaneously according to the Heisenberg-Gabor Uncertainty Principle.

A new concept for spectral analysis, called time-frequency analysis using the wavelet transform, has been developed to further improve the time localization of frequency content. The method decomposes a signal into a family of elementary signals and the signal may be reconstructed by the superposition of these elementary signals. These elementary signals are the fundamental unit in the time-frequency plane and are thus

call kernel atoms. The choice of the kernel atoms must satisfy the condition that they are compactly supported both in time and frequency.

The concept was first introduced by Gabor in 1946, in which he developed the idea of the decomposition of signals into minimal grains of information (or "time-frequency atoms"). Since then, many methods have been developed about wavelet transform. Morlet (1982) constructed Morlet wavelet bases to be used as the kernel atoms instead of Gabor atoms. Grossmann et al. (1984) introduced continuous wavelet transform instead of looking only at discreet set of wavelets. Mallat (1989) introduced pyramid decomposition to perform multi-resolution analysis on the orthogonal wavelet bases. Mallat (1993) introduced another redundant algorithm, the Matching Pursuit Decomposition, and applied the method with the choice of modulated Gabor atom bases, which is the sinusoidal component with varied Gausian modulation. Feauveau (1990) introduced bi-orthogonal bases instead of orthogonal bases, which has become very useful in image processing. Other bases were introduced, which were no longer wavelet bases (in the sense of the wavelets with constant shape), e.g. the wavelet packets (a full library of orthonormal bases constructed from a quadrature mirror filter and equipped with an algorithm for best basis selection).

In recent years, the wavelet transform has found many applications in seismic imaging because of its improved time-frequency localization. Xin-Gong Li, et al. (1996) applied multi-resolution decomposition with the choice of Haar wavelet family to suppress the noise in the inversion to reconstruct tomographic models. Grubb and Walden (1997)

discussed the possibility of using different attributes from the discrete wavelet transform to characterize seismic time series by using multi-resolution decomposition and compactly supported Daubechies wavelets. Deighan and Watts (1997) applied a wavelet packet method with the choice of Battle-Lemarie wavelet bases, and got better result in suppressing the ground-roll by removing certain wavelets. Chakraborty and Okaya (1995) applied the Matching Pursuit Decomposition and Gabor wavelets to seismic data and got a high-resolution time-frequency image of the data. In my thesis, I dwell on the application of Matching Pursuit Decomposition and use the Ricker wavelet, the same order as the second derivative of the Gausian function, as the wavelet bases. The result is a much improved and clearer time-frequency representation of the seismic data than Fourier methods.

The thesis is divided into six sections. Chapter one gives a quantitative introduction to the Fourier transform and Short Time Fourier Transform (STFT). Chapter two provides some theoretical background for the continuous wavelet transform and the discrete wavelet transform. Chapter three and four provide theoretical explanation for the multi-resolution analysis and the wavelet packet. Chapter five introduces the Matching Pursuit Decomposition with the application of the Ricker wavelets. Chapter six reports initial attempts to do frequency-wavenumber-time decomposition using a combination of wavelet and Fourier transforms. This is followed by conclusions and discussions.

Chapter 1 Fourier Analysis

The Fourier Transform

The Fourier transform is defined as a way to map a signal $g = \{g(t);\ t \in \Gamma\}$, which depends on a parameter t in some domain Γ, into another signal $G = \{G(f);\ f \in \varsigma\}$, which depends on a new parameter or independent variable f, which is called *frequency*. As analogous to t, f is an independent variable that takes values in another independent domain, denoted ς.

There are two basic types of Fourier transform, as represented mathematically as follows:

$$G(f) = \Sigma_{n \in \Gamma} g(n)\, e^{-i2\pi fn} \quad \Gamma \text{ is discrete} \qquad (1.1)$$

Or

$$G(f) = \int_{-\infty}^{+\infty} g(t)\, e^{-i2\pi ft}\, dt \quad \Gamma \text{ is continuous} \qquad (1.2)$$

The reconstruction of the signal is:

$$G(t) = \int_{-\infty}^{+\infty} G(f)\, e^{i2\pi ft}\, df \quad \Gamma \text{ is continuous} \qquad (1.3)$$

A fundamental difference between the discrete and continuous transform is the fact that DTFT (Discrete Time Fourier Transform) shows periodicity with a period of frequencies from zero to the Nyquist frequency, while CTFT (Continuous Time Fourier Transform) is not periodic. Thus we have to restrict our consideration of the spectrum within one period when performing DTFT. In addition, both the DTFT and CTFT behave differently depending on whether or not the index set Γ is finite or infinite. The Fourier transform may not be sufficient if the limits defining the infinite sums or integrals do not exist.

The Fourier transform is a way to project a signal from one independent domain into a family of monochromatic waves. It is important because it sometimes provides the basis for a better comprehension of underlying phenomena and is an essential complement to the temporal description. For many years, the Fourier transform was the dominant method used to perform frequency analysis. It has the great advantage of ease of use, computational efficiency and complementing the information by providing the representation in another domain. However, from the mathematical definition, the transform results in the loss of all the chronological information after the transform. Thus it is only adequate in the case that the frequency content of the signal is invariant with time, which is called a stationary frequency spectrum. In addition, it projects signals into infinite sinusoidal series, which represent only global features in the sense that sinusoidal components have infinite duration in the time domain and are projected as points (monochromatic) in the frequency domain. It is incapable of giving a sufficient

representation of a signal that has many transient features. The time localization of the frequency content is improved by the short time Fourier transform, which is done by applying Fourier transform to the signal within a moving short window.

The Short Time Fourier Transform

The short time Fourier Transform (STFT) is performed by windowing the signal prior to successive Fourier Transform within the limited window length. It is a way to project signal onto a family of sinusoidal components modulated by a window with certain length and time shift. By performing short time Fourier Transform, we obtain a mixed representation, joint in time and frequency, which can be defined as follows.

Given a function $h \in L^2(\Re)$, we construct the corresponding wavelet family $\{h_{(b,\nu)} ; b \in \Re, \nu \in \Re\}$ obtained by shifting and modulating h, where b is the shift, and ν is the frequency.

$$h_{(b,\nu)}(t) = e^{-i2\pi\nu x} h(t-b) \qquad (1.4)$$

The function h is used to concentrate the analysis near the specific point *b* in the time domain. It is called the window function of the analysis. Once the family of the window functions is fixed, the STFT is defined as follows.

$$F_x(t,\nu) = \langle x, h_{t\nu} \rangle = \int_{-\infty}^{+\infty} x(s)\, h^*(s-t)\, e^{-i2\pi\nu s}\, ds \qquad (1.5)$$

(* denotes complex conjugate, $X(\xi)$ is the Fourier transform of $x(s)$.)

Thus by performing the STFT, we project the analyzed signal onto a family of atoms, which are derived from a mother element (the window function h(t)) by time and frequency shifts. As a result, we obtain a mixed representation in time and frequency.

The inverse STFT is defined as follows.

$$x(s) = \frac{1}{2\pi \|h\|^2} \int_{-\infty}^{+\infty} F_x(t,\nu)\, h_{(t,\nu)}(s)\, dt\, d\nu \qquad (1.6)$$

The representation (equation (1.4)) shows the typical restrictions of the Fourier transform: All the atoms in the wavelet family have the same shape. Therefore the high-frequency components oscillate too much, while the low-frequency components oscillate too little within the modulation window. It is not flexible enough to achieve both good time and frequency resolution simultaneously. The time-resolution could only be improved by shortening the window length, which in return reduces the frequency resolution. As a result, this method is not well adapted to describe the structures that have wide range of frequency contents.

The STFT possesses the invariance property, which means the STFT of a shifted copy of the signal x equals the corresponding time-shifted copy of the STFT of the signal.

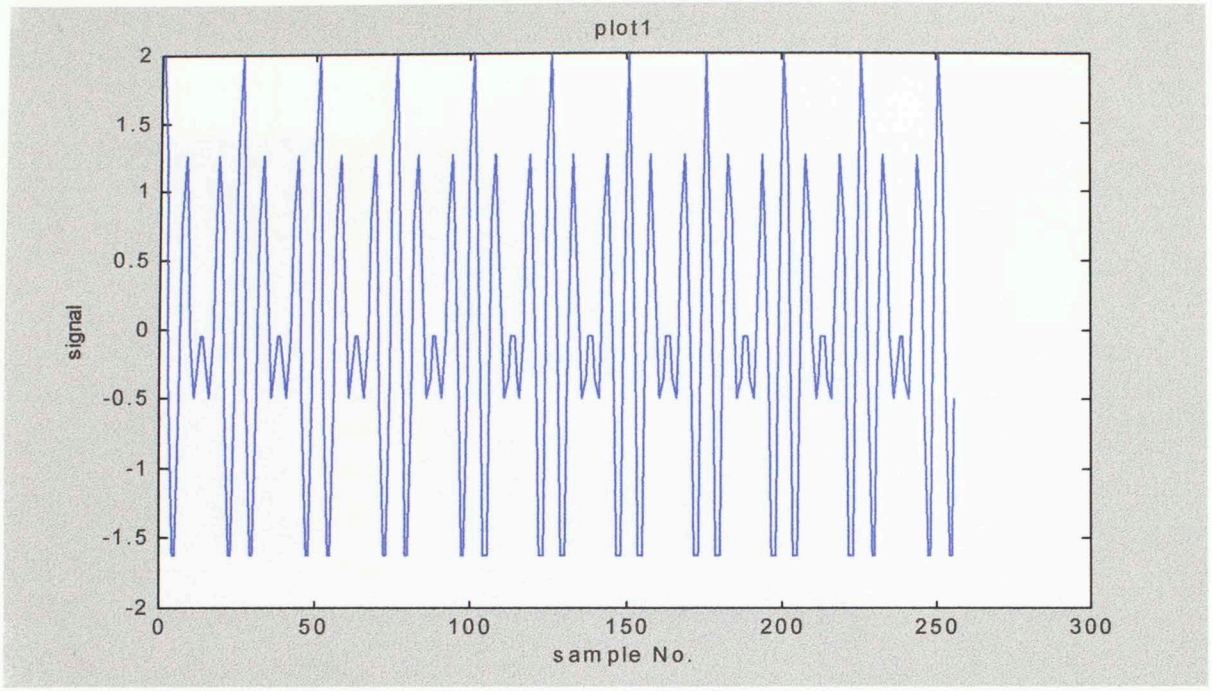

Figure 1. A trace of two sinusoidal components

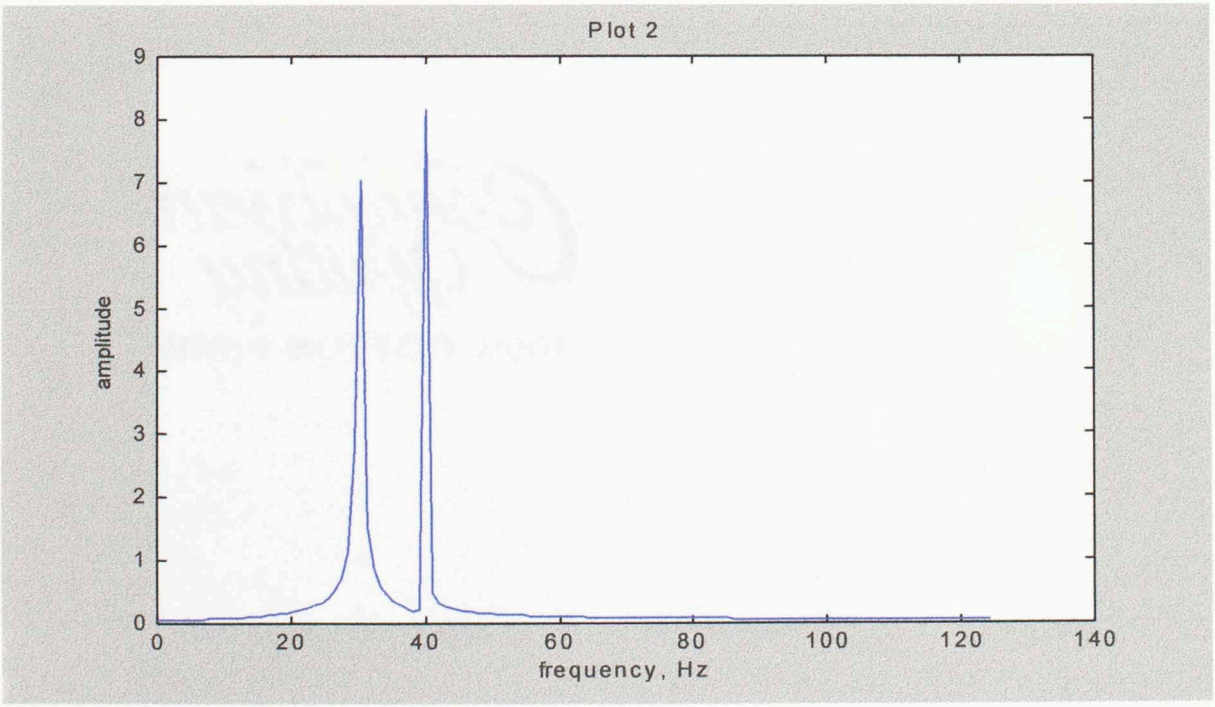

Figure 2. The Fourier transform of Plot 1.

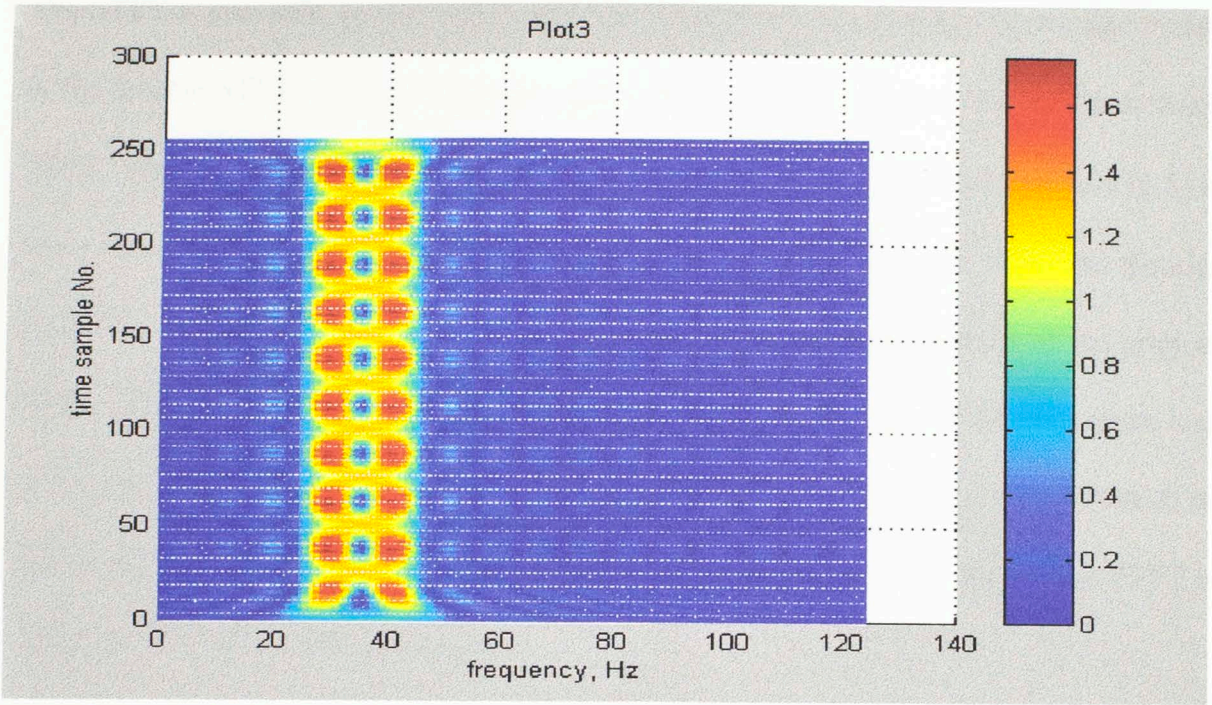

Figure 3. The STFT of Plot 1 with a boxcar window, with the window length of 32 samples.

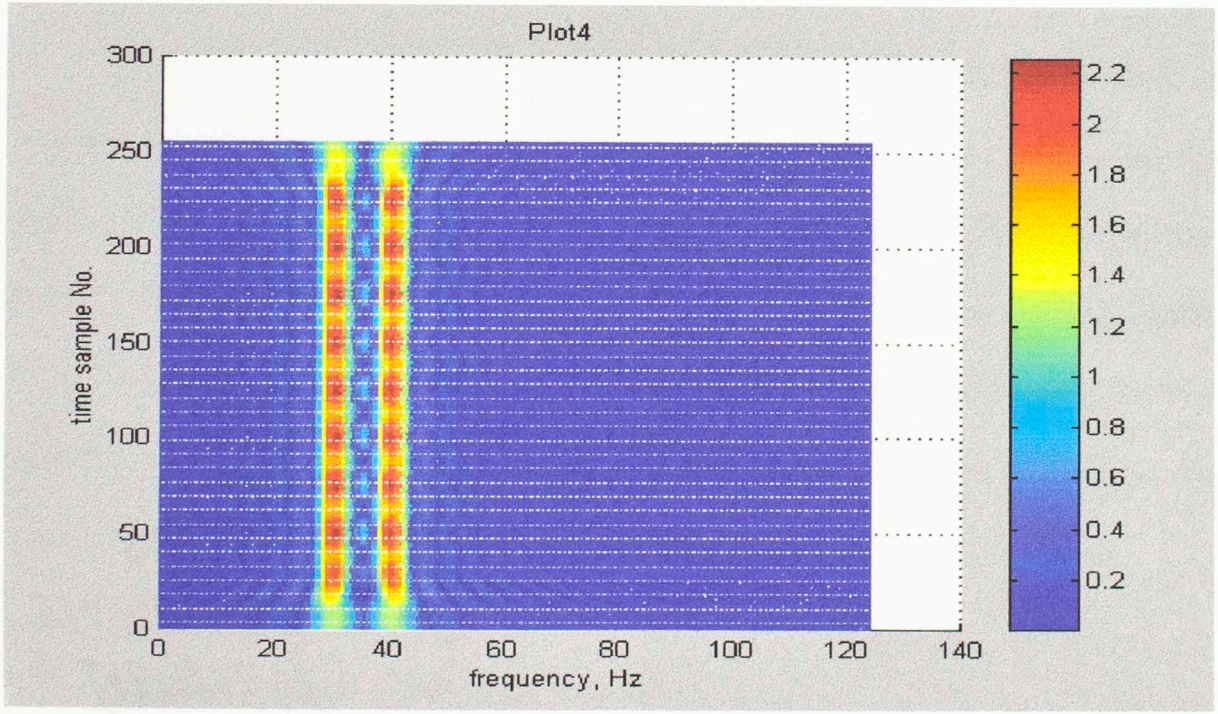

Figure 4. The STFT of Plot 1 with a boxcar window, with the window length of 64 samples.

The STFT is generally an extremely redundant algorithm. As a result, it is possible to use in the inverse STFT a window function that is different from the one used in the forward STFT. What is more, the resulting time-frequency spectrum $F_x(t,\nu)$ from equation (1.5) is the convolution of the Fourier transform of the window function with the Fourier transform of the signal. A deconvolution is needed to obtain the spectrum of the original signal. The translation step is difficult to determine too. Large translation may cause loss of the frequency information, while small translation may cause excessive representation of the same information. Also the wavelet family $\{h_{(b,\nu)} ; b \in \Re, \nu \in \Re\}$ used in the STFT is not generated on an orthogonal basis.

Data Analysis

A synthetic trace (Figure 1) is generated by adding two cosine components of frequencies 30 Hz and 40 Hz. The Fourier transform of the trace (Figure 2) shows that the signal is mainly composed of two harmonic components of frequencies 30 Hz and 40 Hz. In the STFT, boxcar-shape windows with different lengths are used. The window lengths are 32 samples (Figure 3) and 64 samples (Figure 4) respectively. In both cases, the two harmonic components are basically well resolved, while the frequency contents are resolved better in Figure 4 because of the use of a larger window length.

Another synthetic trace (Figure 6) is generated by adding three harmonic components with frequencies of 30 Hz, 40 Hz and 50 Hz. Two spikes are present at the 89*th* sample and 159*th* sample respectively. The Fourier transform of the signal (Figure 7)

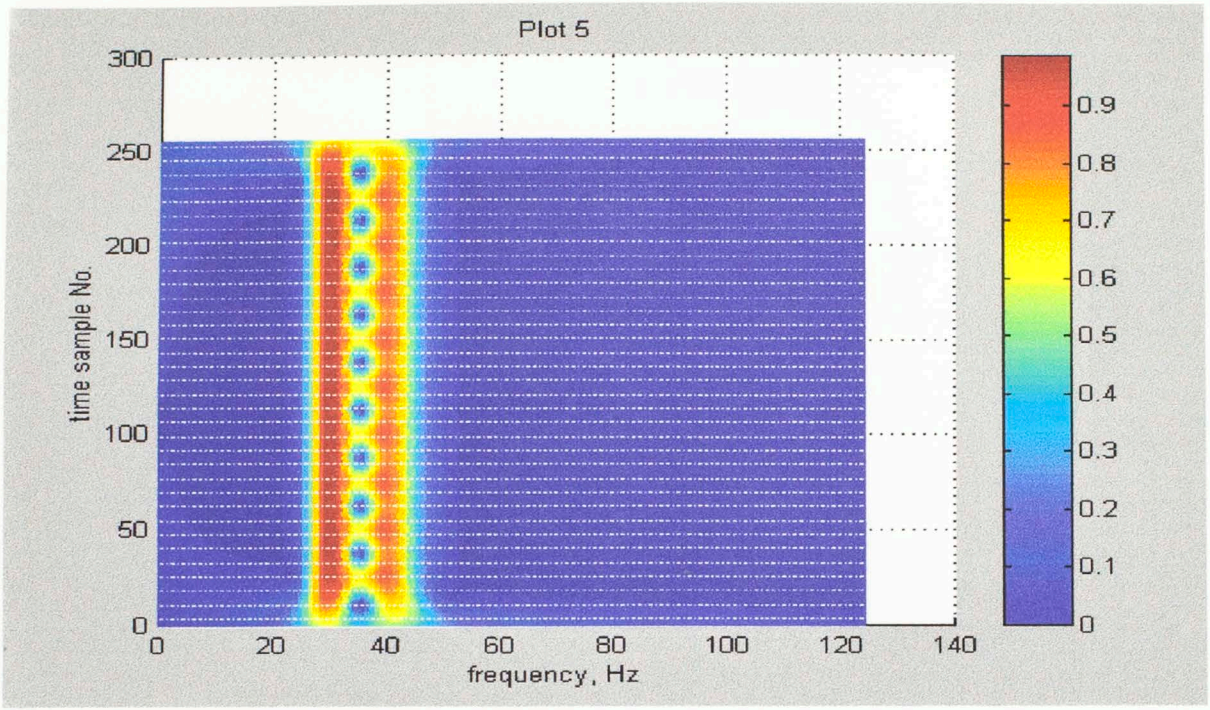

Figure 5. The CWT of Plot 1.

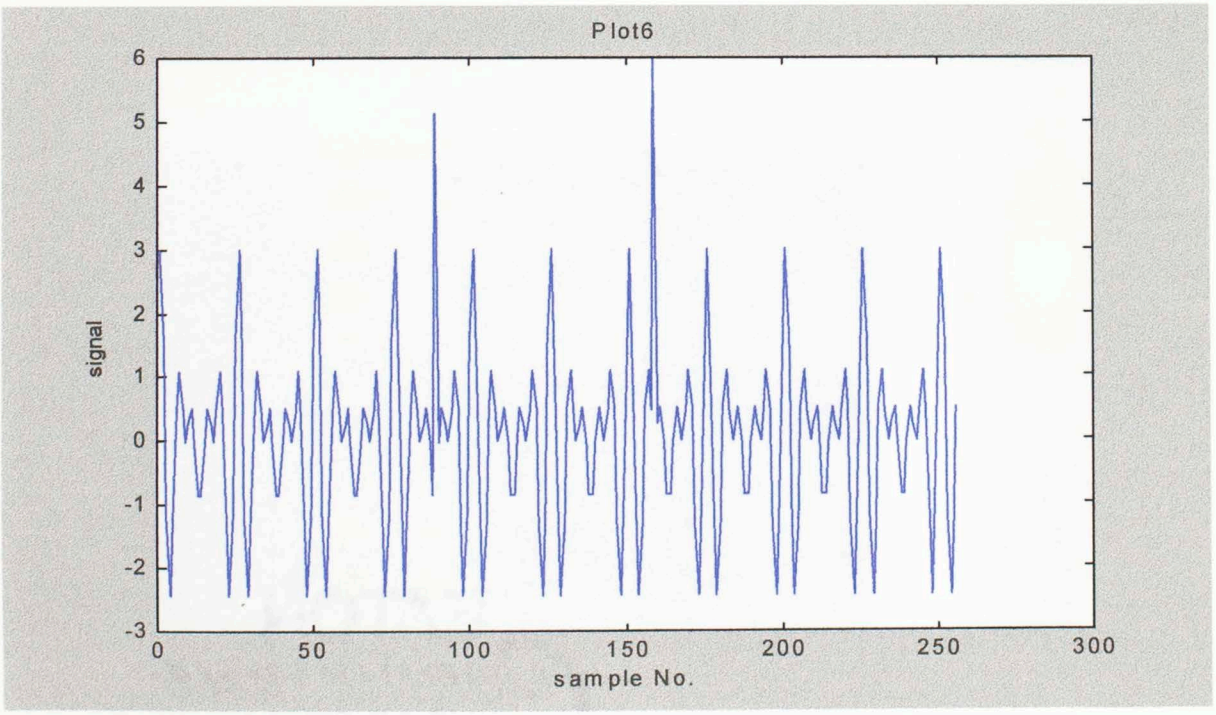

Figure 6. The signal with three cosine components and two impulses.

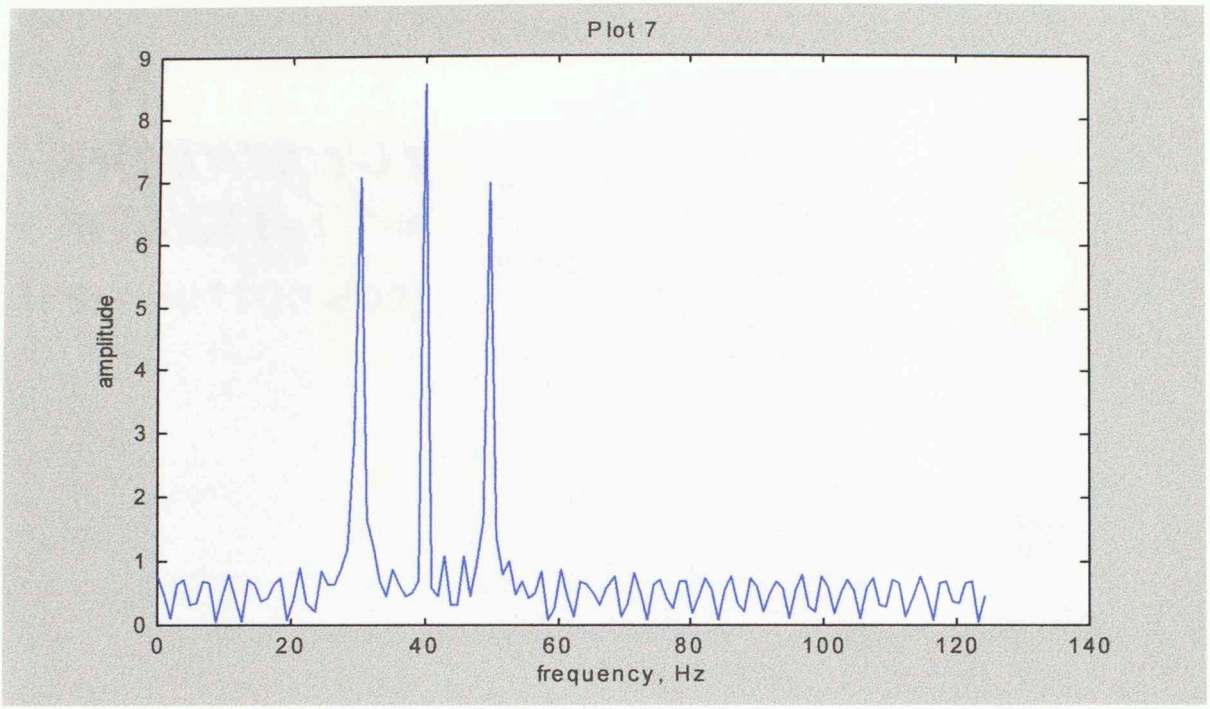

Figure 7. The Fourier transform of Plot 6.

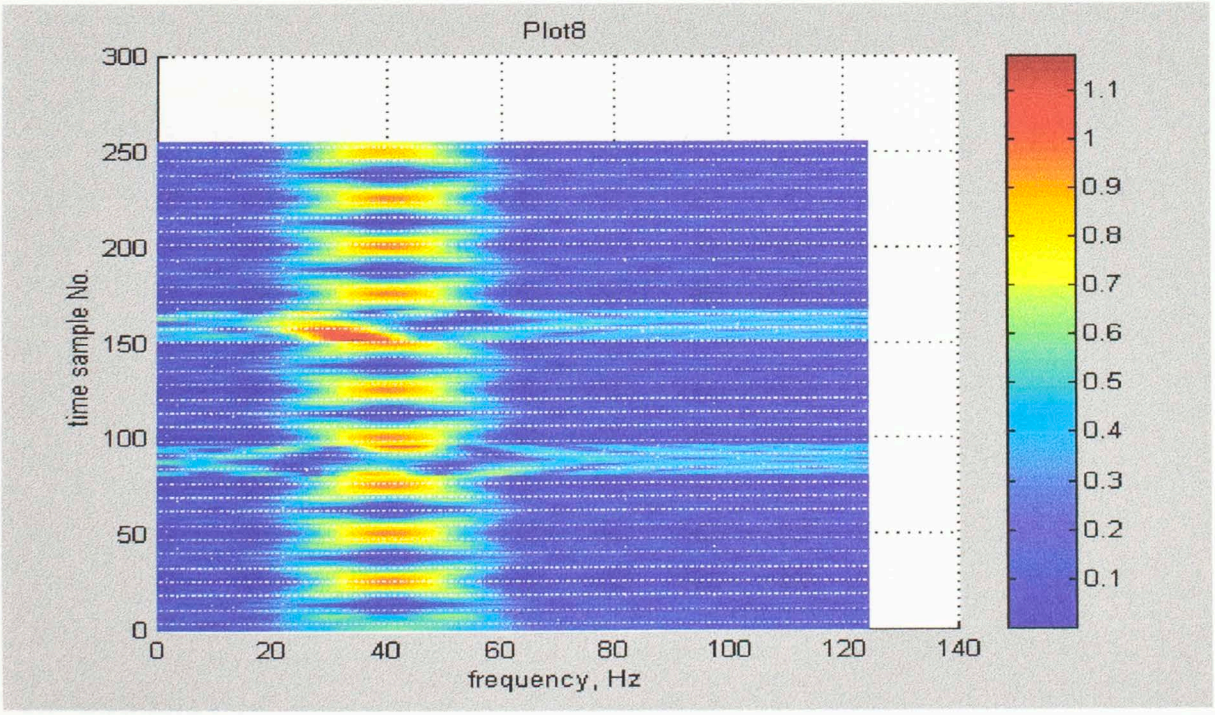

Figure 8. The STFT of Plot 6 with a boxcar window, with the window length of 16 samples.

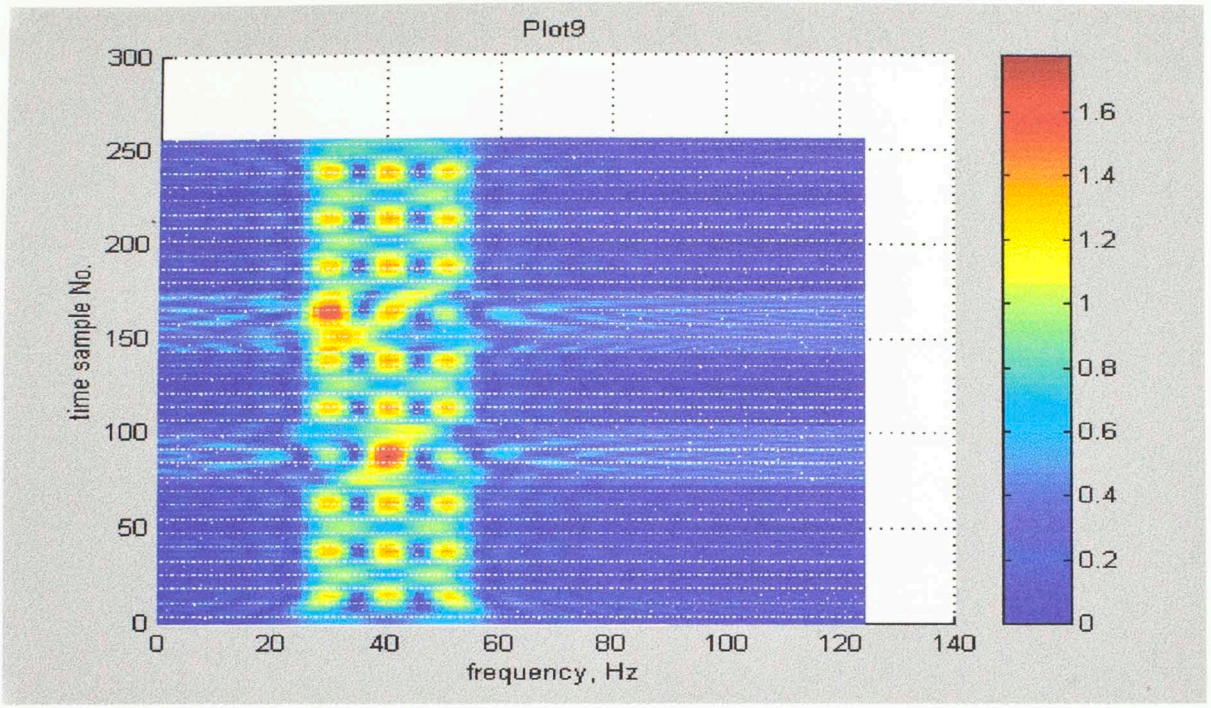

Figure 9. The STFT of Plot 6 with a boxcar window, with the window length of 32 samples.

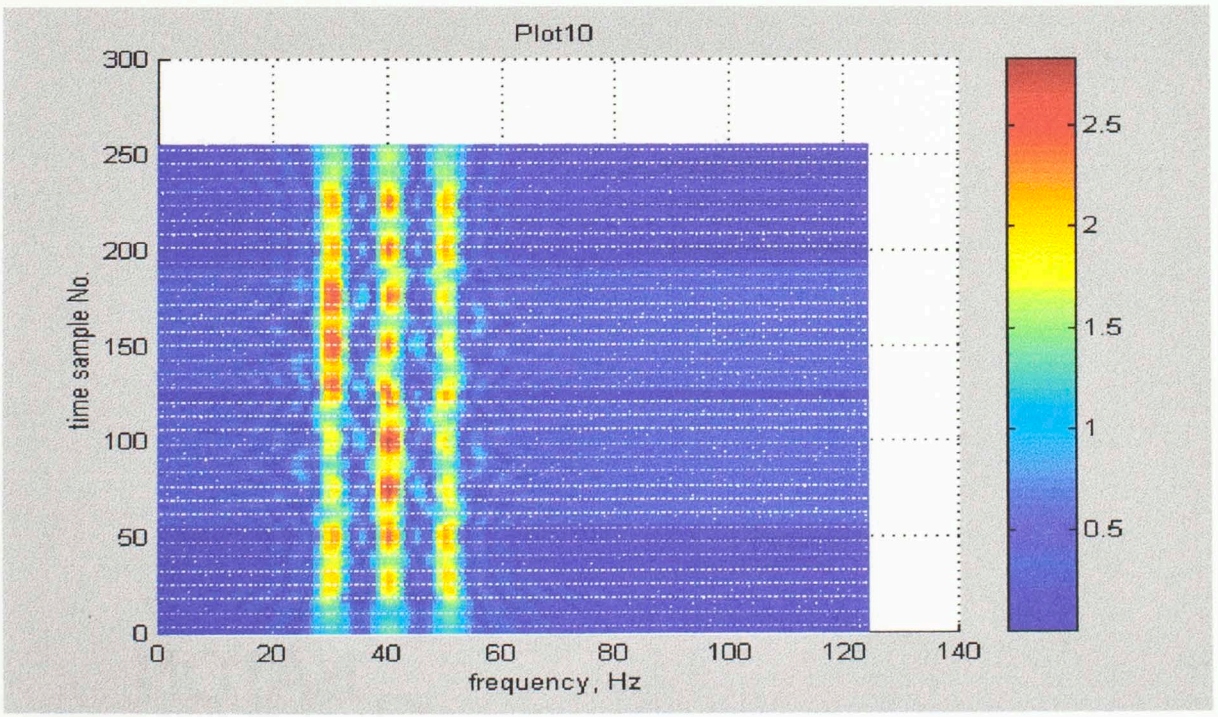

Figure 10. The STFT of Plot 6 with a boxcar window, with the window length of 64 samples.

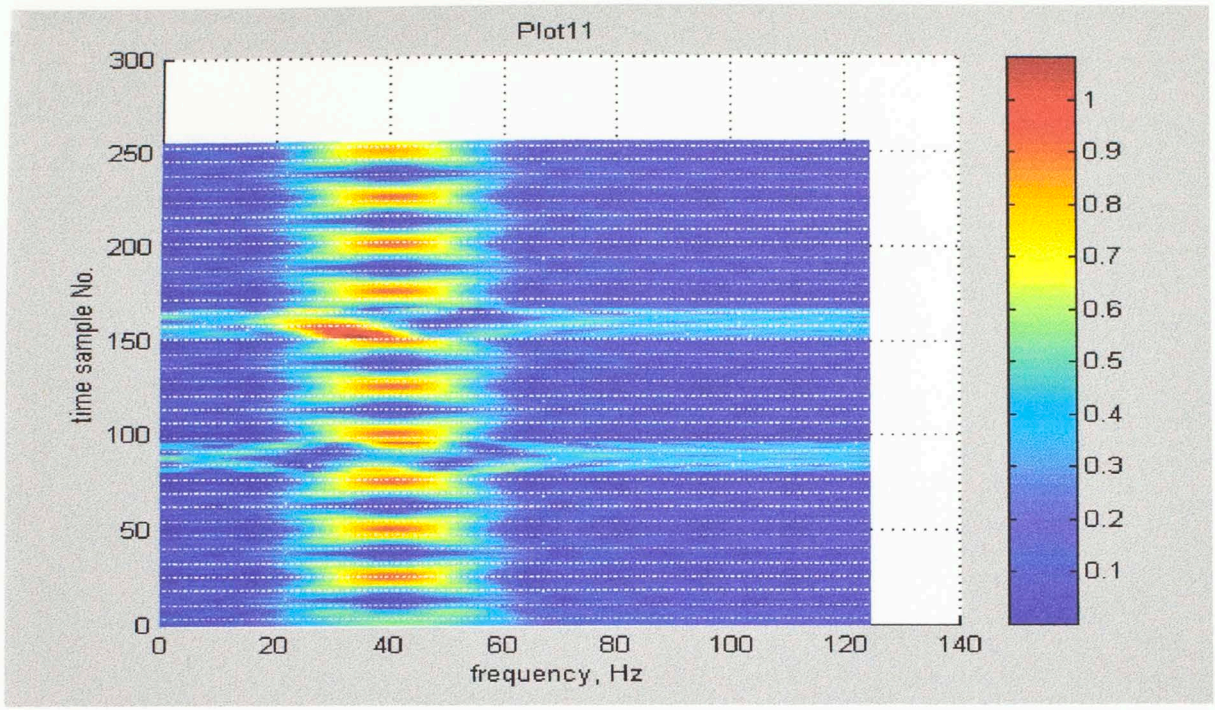

Figure 11. The STFT of Plot 6 with a Gaussian window, with the window length of 16 samples.

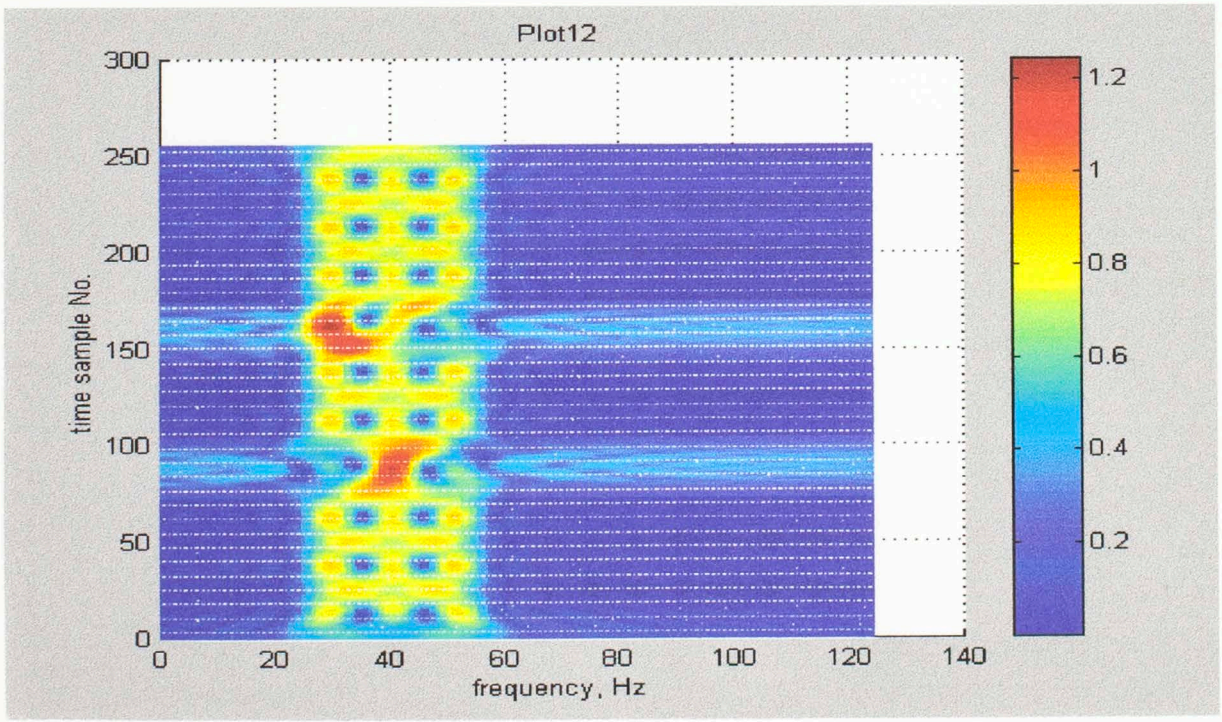

Figure 12. The STFT of Plot 6 with a Gaussian window, with the window length of 32 samples.

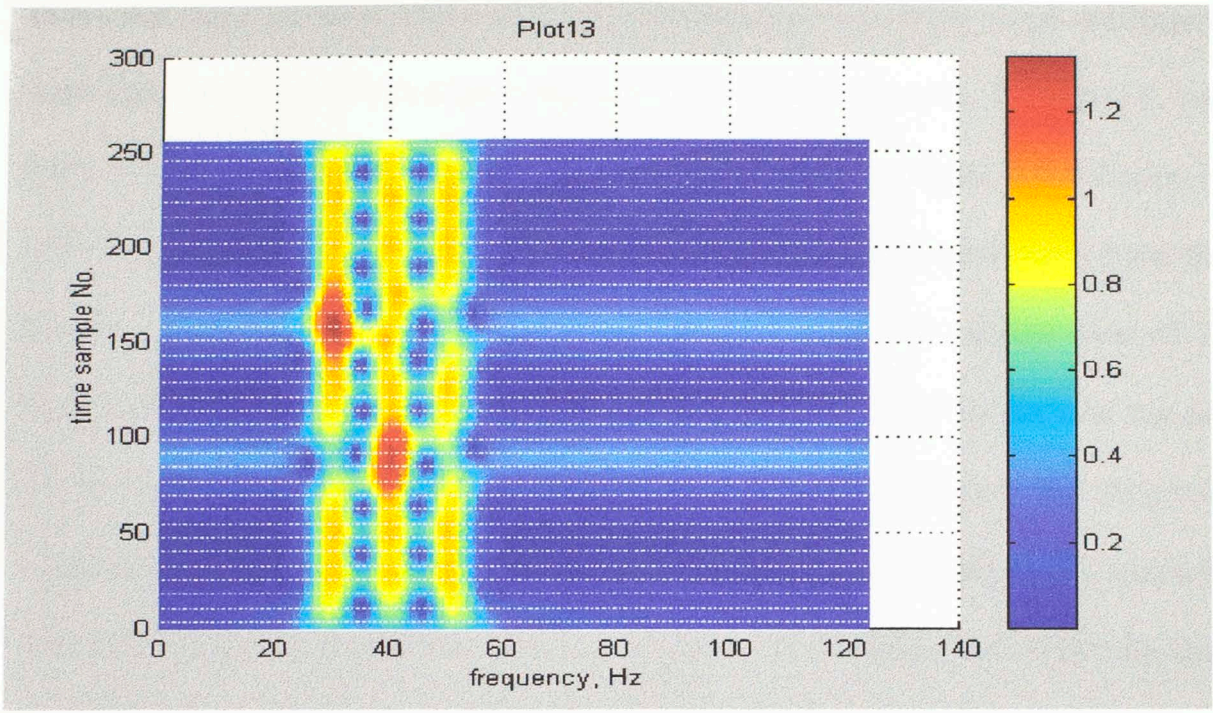

Figure 13. The STFT of Plot 6 with a Gaussian window, with the window length of 64 samples.

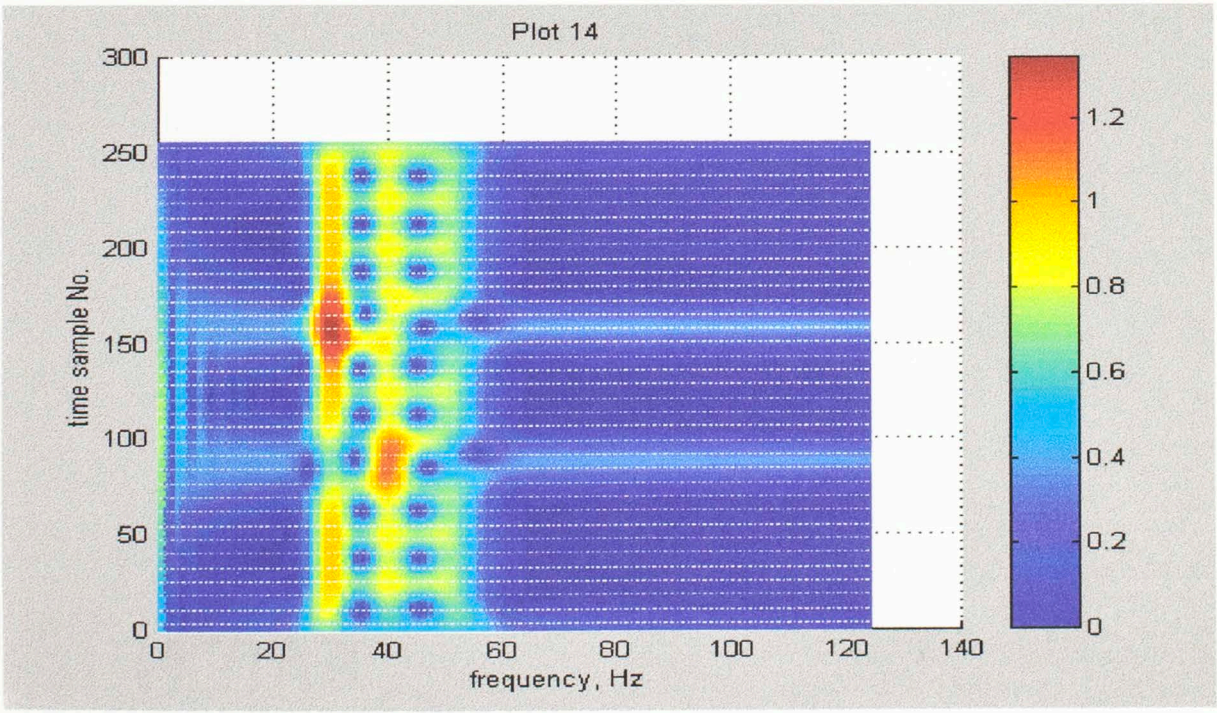

Figure 14. The CWT of Plot 6.

shows that there are mainly three harmonic components present in the signal and there is some signal pattern with its energy evenly distributed at all frequencies. In the STFT, two types of window shape are used, the boxcar window (Figure 8, Figure 9 and Figure 10 with the window sizes of 16 samples, 32 samples and 64 samples respectively.) and the Gaussian window (Figure 11, Figure 12 and Figure 13 with the window sizes of 16 samples, 32 samples and 64 samples respectively.). From these six figures, one can see that the use of a Gaussian window shows better time-frequency resolution than the use of a boxcar window. In both cases, there is the problem that the frequency resolution is decreased and the time resolution is increased with a small window size, while the time resolution is decreased and the frequency resolution is increased with a larger window size.

Chapter 2 Continuous Wavelet Transform

An important feature for the Fourier transform (equation (1.1)) is the fact that the integration of the analyzed function g(t) is performed over the whole time series, so that every point contributes to the calculation of G(f); it is thus difficult to recover local frequency information for f from G(f). In order to capture local Fourier information (called the instantaneous frequency), we may use an analyzing function Ψ that we require to be localized both in frequency (around some mean frequency ξ_Ψ) and in time (around some mean time X_Ψ). Such a function is generally called a wavelet, which means a wave of finite duration. Wavelet analysis decomposes a time series into a family of wavelets that have compact support both in time and frequency domains. A general family of time-frequency atoms can be generated by scaling, translating and modulating a single window function. The window function performs as a band-pass filter. Thus each wavelet is concentrated both in time and frequency. Therefore, wavelet analysis provides information concerning the frequency localization in the time domain. By performing the wavelet transform, we represent functions or signals simultaneously in terms of a time variable and a frequency (or scale) variable.

Because of the Heisenberg uncertainty principle ($\Delta x \Delta f \leq 1/2$, where Δx and Δf are the time and frequency resolution respectively and where the minimum value is achieved by the Gaussian wavelets), it is impossible to achieve infinitely precise time and frequency resolution simultaneously. As a result, we always have to find a wavelet basis that is well

adapted to the time- frequency distribution of the signal. A compromise between time resolution and frequency resolution has to be found to perform the wavelet transform. Therefore, the time-frequency representation is generally not unique for a given signal, which means that we could have multiple choices of wavelet bases to perform the wavelet transform.

Many aspects need to be considered in the wavelet transform method. Among these one aspect is the choice of the wavelet family which is optimized for the analysis of a given signal. The computation efficiency may be improved greatly with a wavelet choice that is better correlated with the signal. Another aspect is the interpretation of the time-frequency transform. From the introduction of the Short Time Fourier Transform by Gabor in the 1940s, many methods have been developed to perform wavelet analysis, among which are the continuous wavelet transform (Grossmann, Morlet et al, 1984), multi-resolution analysis on an orthogonal wavelet bases (Mallat et al, 1989), Matching Pursuit Decomposition (Mallat, 1993), and wavelet packet analysis (Coifman and Wickerhauser, 1992).

Continuous Wavelet Transform

The Continuous Wavelet Transform (CWT) is a similar operation to the Short Time Fourier Transform (STFT). In the STFT, a time-frequency resolution is achieved by sliding a window function to localize the Fourier transform in the time domain, in which the time parameter gives the time localization of the center of the window and a frequency

parameter gives the Fourier transform of the windowed signal. The STFT is based upon time and frequency translations. To the contrary, the CWT is based upon time translations and frequency scaling, which means the window shape varies as the frequency content changes.

Let $\Psi \in L^2(\Re)$ be a fixed function. The corresponding family of wavelets is $\{\Psi_{(b,a)} ; b \in \Re, a \in \Re\}$ generated by shifting and scaling Ψ defined as follows.

$$\Psi_{(b,a)}(x) = \frac{1}{\sqrt{|a|}} \Psi(\frac{x-b}{a}) \qquad (2.1.1)$$

in which b is the time translation, and a is the scale related to the central frequency. Ψ is the analyzing wavelet. It is also called the mother wavelet of the analysis. The wavelet $\Psi_{(b,a)}$ is normalized and can be viewed as a copy of the original wavelet Ψ re-scaled by the scale and centered around the translation. By projecting an arbitrary signal f (x) onto the family of functions defined as equation (2.1), the associated continuous wavelet transform is defined as follows.

$$T_f(b,a) = < f, \Psi_{(b,a)} > = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{+\infty} f(x) \Psi^*(\frac{x-b}{a}) dx \qquad (2.1.2)$$

(* denotes the complex conjugates.)

Let $\Psi \in L^2(\Re)$, C_Ψ is defined by

$$C_\Psi = 2\pi \int_0^{+\infty} \frac{|\Psi(\xi)|^2}{\xi} d\xi \qquad (2.1.3)$$

($\Psi(\xi)$ is the Fourier Transform of the wavelet $\Psi(t)$)

The inverse continuous wavelet transform is

$$f(x) = \frac{1}{C_\Psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} T_f(b, a) \frac{\Psi_{(b,a)}(x)}{a^2} da\, db \qquad (2.1.4)$$

Equation (2.1.4) could be made clear by taking inner product of both sides of the equation with any $g \in L^2(\Re)$.

The Continuous Wavelet Transform (CWT) is the projection of the signal onto a family of wavelets with varying scales and time translations. At each shift b, $\Psi_{(b,a)}$ is generated by scaling Ψ with the value a that is related to the central frequency. The value of *a* (scale) is large for wavelets with low central frequency, which results in a long window size in time (low time resolution) and a fine frequency resolution. The value of *a* is small for wavelets with high central frequency, which results in a short window size in time (high time resolution) and a low frequency resolution. The value $T_{f\ (b,a)}$ is the expansion coefficient of the signal f (x) on the wavelet $\Psi_{(b,a)}$. It contains the frequency information (related to the scale *a*) of the signal f (x) around the translation *b*. Because the

wavelets are dilated by the scale a, which is related to the frequency, the CWT provides a more flexible method to achieve a good compromise between the time and frequency resolution than the STFT, in which the window shape is constant.

Like the STFT, the CWT possesses the invariance property too, which means the CWT of a shifted copy of the signal equals the corresponding time-shifted copy of the CWT of the signal. However, the CWT is a redundant algorithm. In the inverse CWT, it is possible to use a wavelet family that is different from the one used in the forward CWT. In many circumstances, the redundancy of the continuous transform could be reduced by the carefully designed discretization.

One wavelet used in the thesis is the modulated Gabor wavelet. Examples of Gabor wavelets with central frequency (CF) of 15 Hz and 40 Hz and their frequency spectrums are given below.

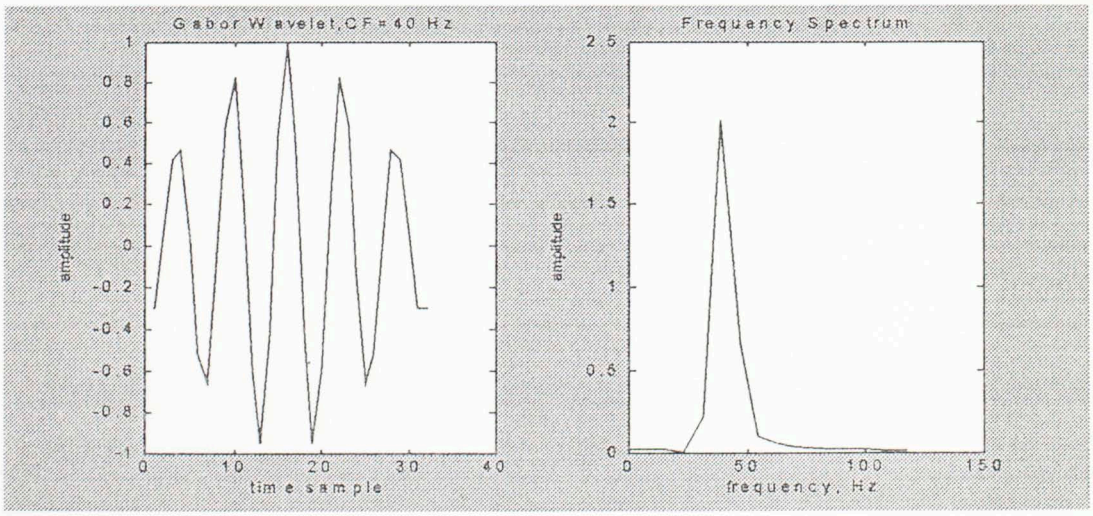

Figure 1.1, An example of the modulated Gabor wavelet with the central frequency of 40 Hz. The frequency spectrum of the wavelet is given on the right.

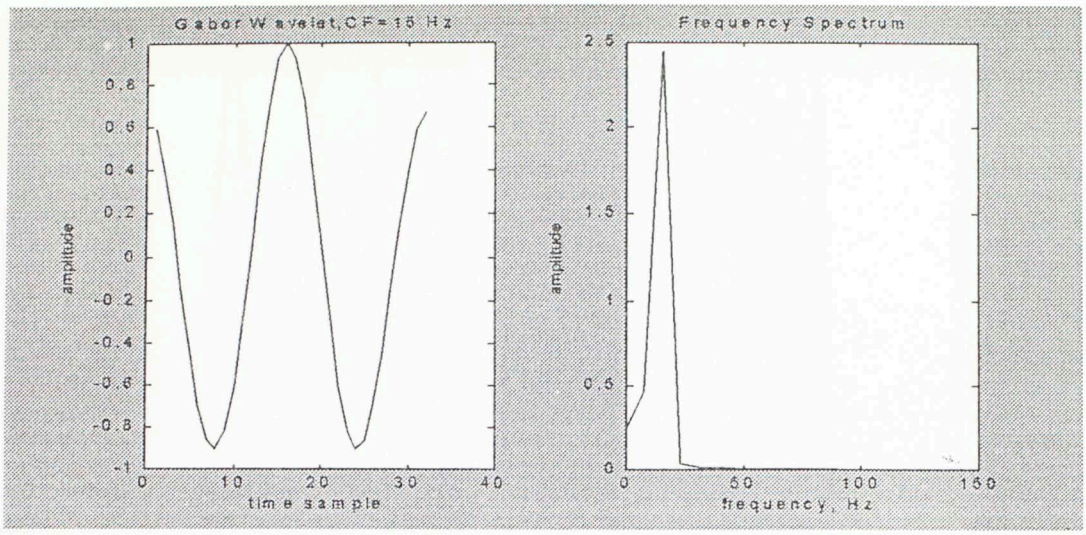

Figure 1.2, An example of the modulated Gabor wavelet with the central frequency of 15 Hz. The frequency spectrum of the wavelet is given on the right.

Data Analysis

A trace (Figure 1) is generated by adding two cosine components with the frequency of 30 Hz and 40 Hz. The CWT of the signal (Figure 5) shows that frequency resolution is not as good as the frequency resolution of the STFT with a window size of 64 samples (Figure 4), however better than the STFT with a window size of 32 samples (Figure 3).

A second trace (Figure 6) is generated by adding three harmonic components and two spikes. From the CWT of the signal (Figure 14), the time resolution increases as the frequency increases. Two impulses are shown on the CWT with increasing time resolution when the frequency increases.

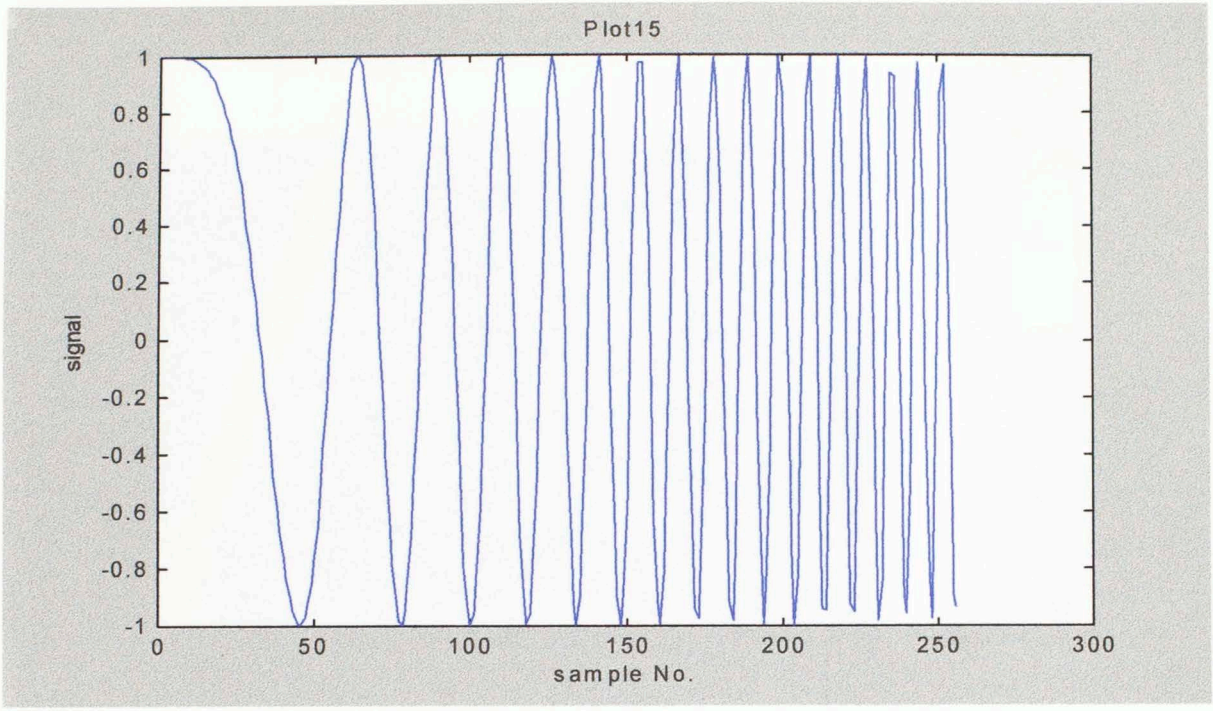

Figure 15. A linear chirp.

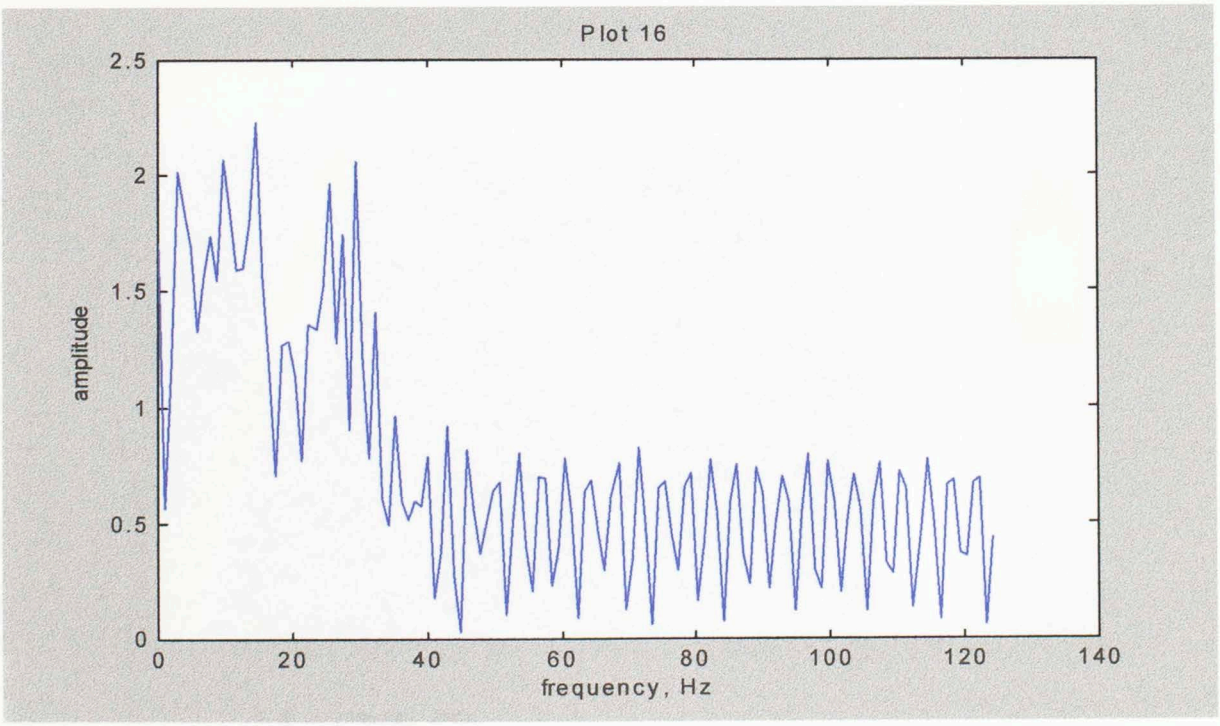

Figure 16. The Fourier transform of Plot 15.

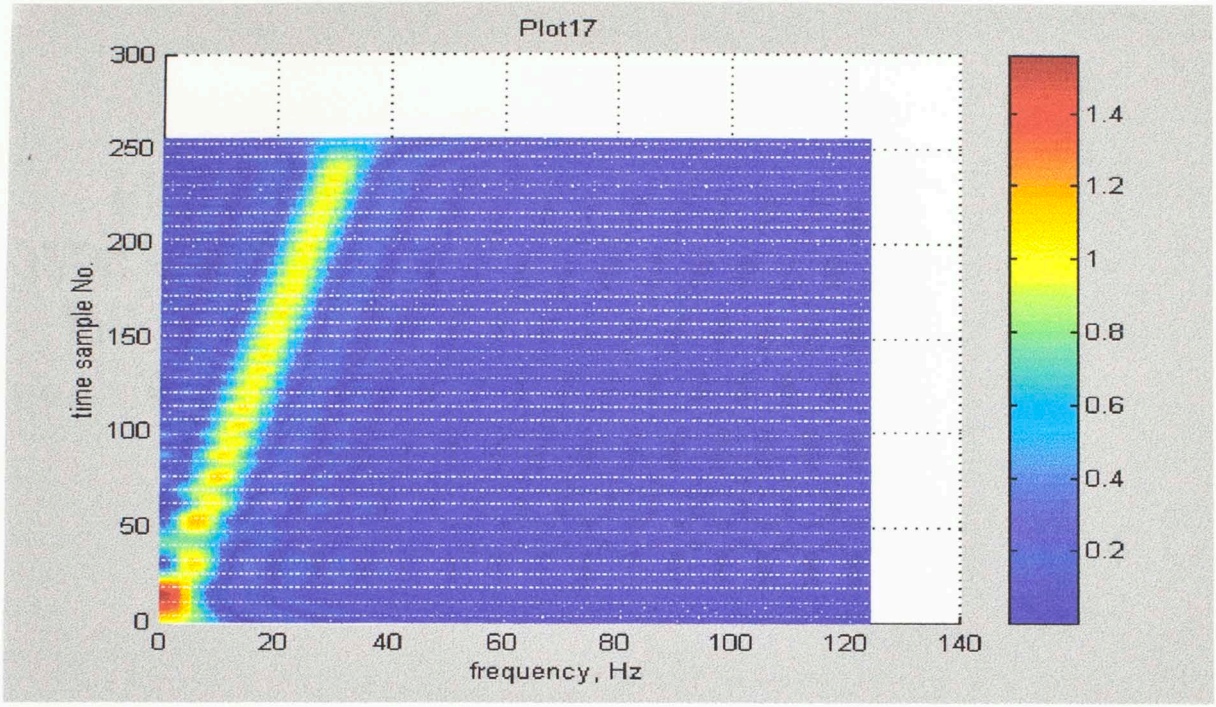

Figure 17. The STFT of Plot 15 with a boxcar window, with the window length of 32 samples.

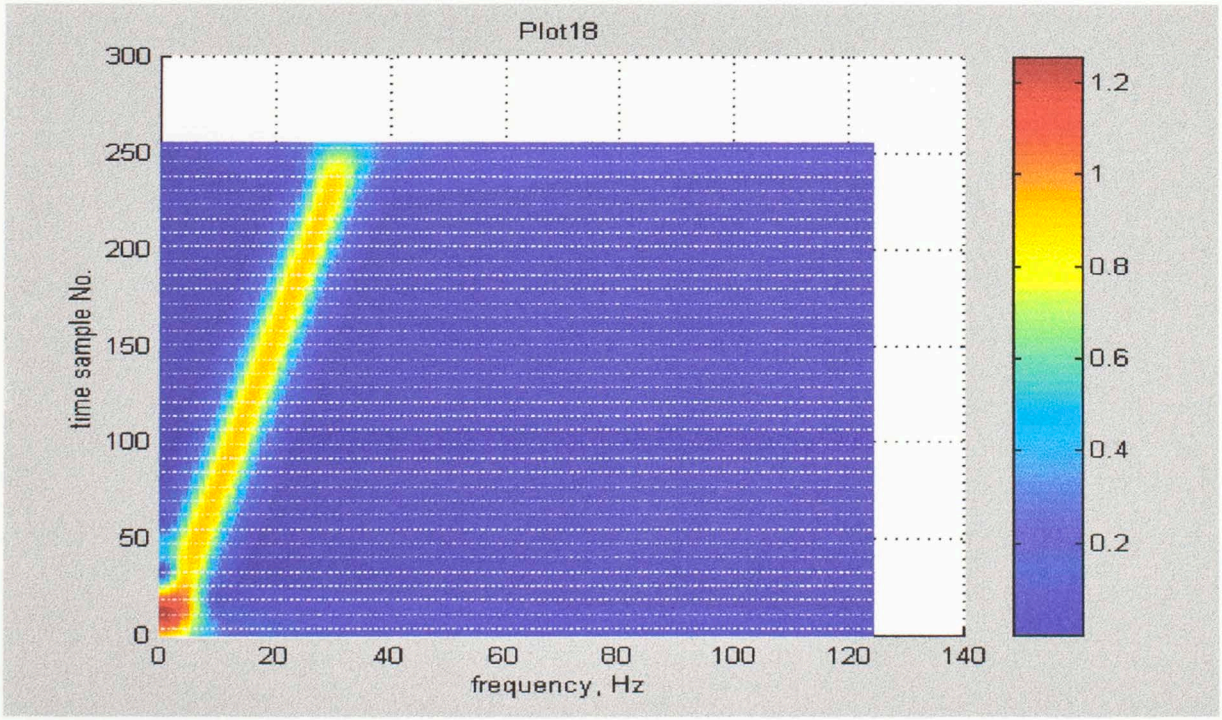

Figure 18. The STFT of Plot 15 with a Gaussian window, with the window length of 32 samples.

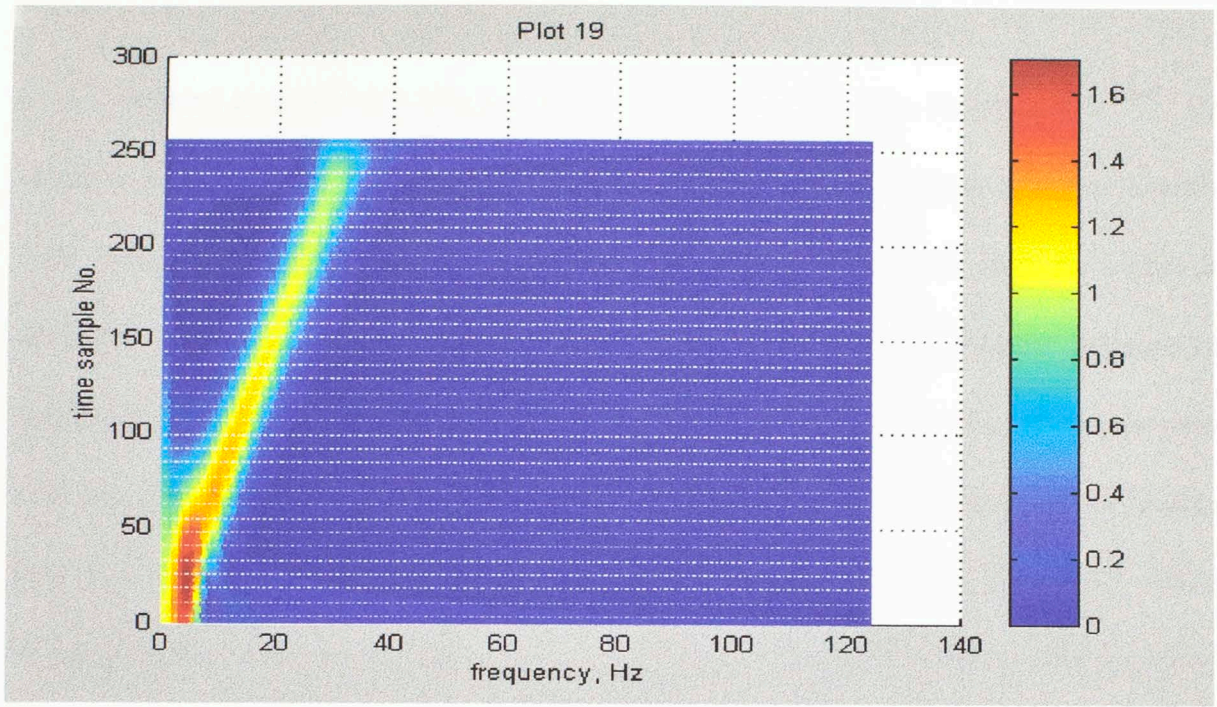

Figure 19. The CWT of Plot 15.

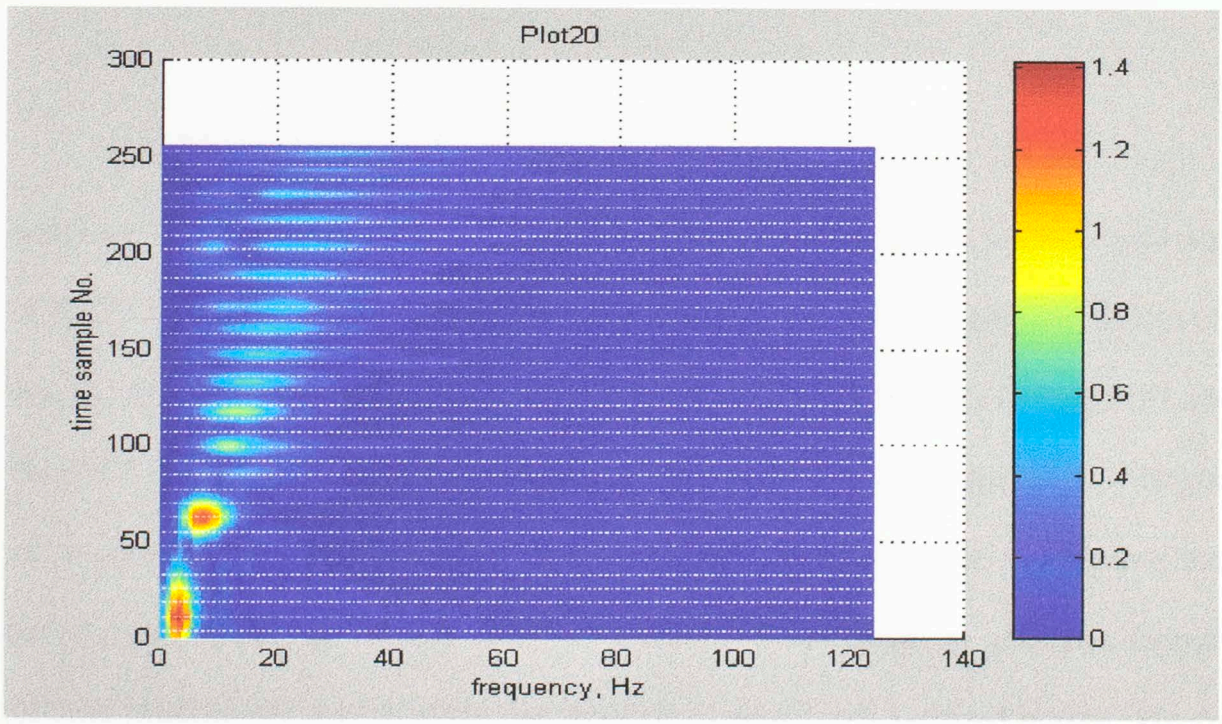

Figure 20. The MPD of Plot 15.

A third trace (Figure 15) is a linear chirp, whose analytic function is $e^{(2\pi at^2)i}$ (in which a is a constant). Therefore, the frequency content of the signal increases linearly with time. The Fourier transform of the signal (Figure 16) shows that the signal is decomposed into a wide range of frequencies. The property of the linear increase of frequency content with time is not reflected. In the STFT, two types of windows are used with the same length, the boxcar window (Figure 17) and the Gaussian window (Figure 18). The linear change of the frequency content with time shown in the STFT with a boxcar window is not as smooth as in the STFT with a Gaussian window. The sidelobes are clearly evident on the STFT with a boxcar window. The CWT of the signal (Figure 19) provides a clear description of the linear change of the frequency content with time, where the time resolution increases with the increase of the frequency.

Another trace (Figure 21) is generated by adding a linear chirp and two wavelets which are localized at the 84*th* sample and the 158*th* sample with the central frequencies of 40 Hz and 80 Hz respectively. The Fourier transform of the signal (Figure 22) shows that the signal has strong energy distributed around 40 Hz. Figure 23 and Figure 24 are the STFT of the signal generated with the use of a boxcar window of the window length of 32 samples and 64 samples respectively. From the comparison of the two figures, the use of a long boxcar (a window size of 64 samples, in this situation) generates serious sidelobe problem and neither of the two wavelets is clearly well localized in time. Figure 25 and Figure 26 are the STFT of the signal generated with use of a Gaussian window of the window length of 32 samples and 64 samples respectively. There is not much

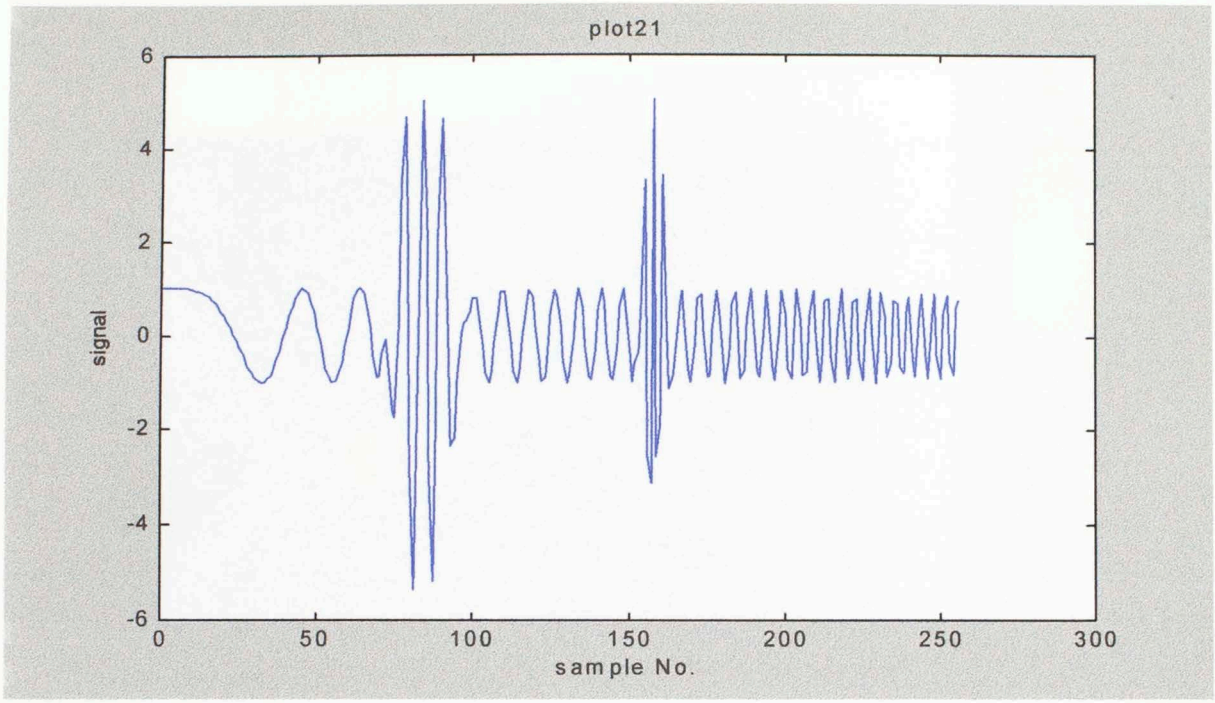

Figure 21. The signal of a linear chirp and two wavelets.

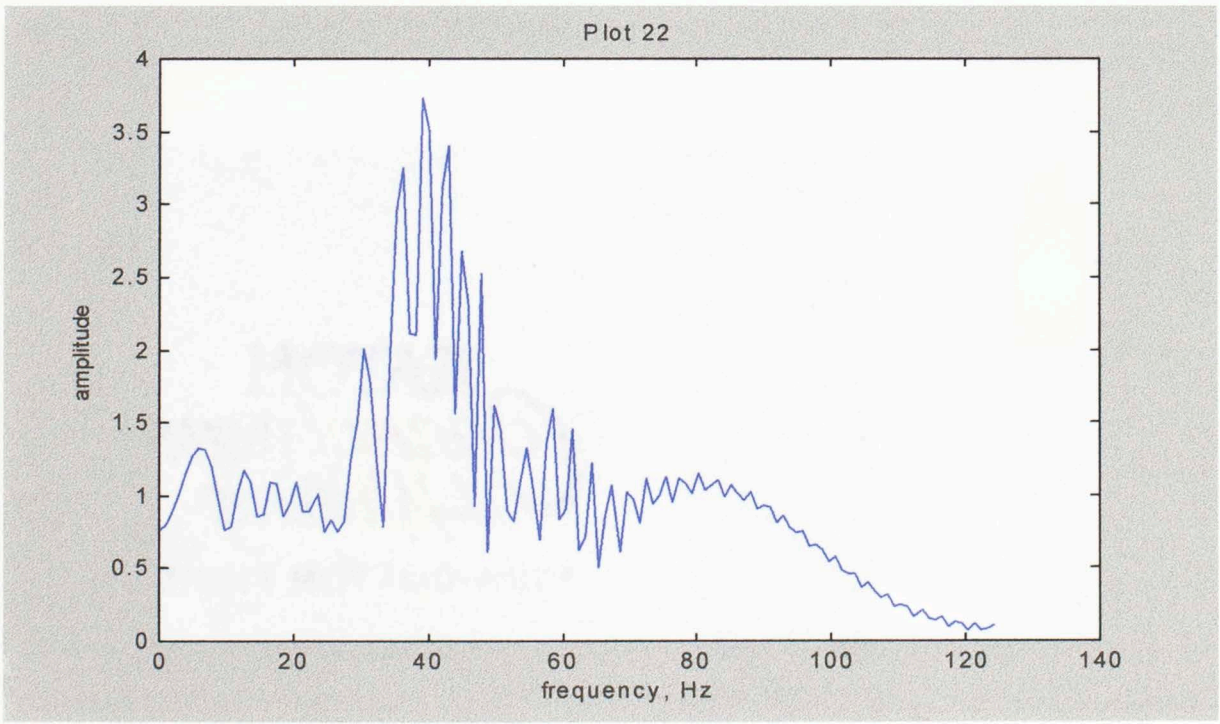

Figure 22. The Fourier transform of Plot 21.

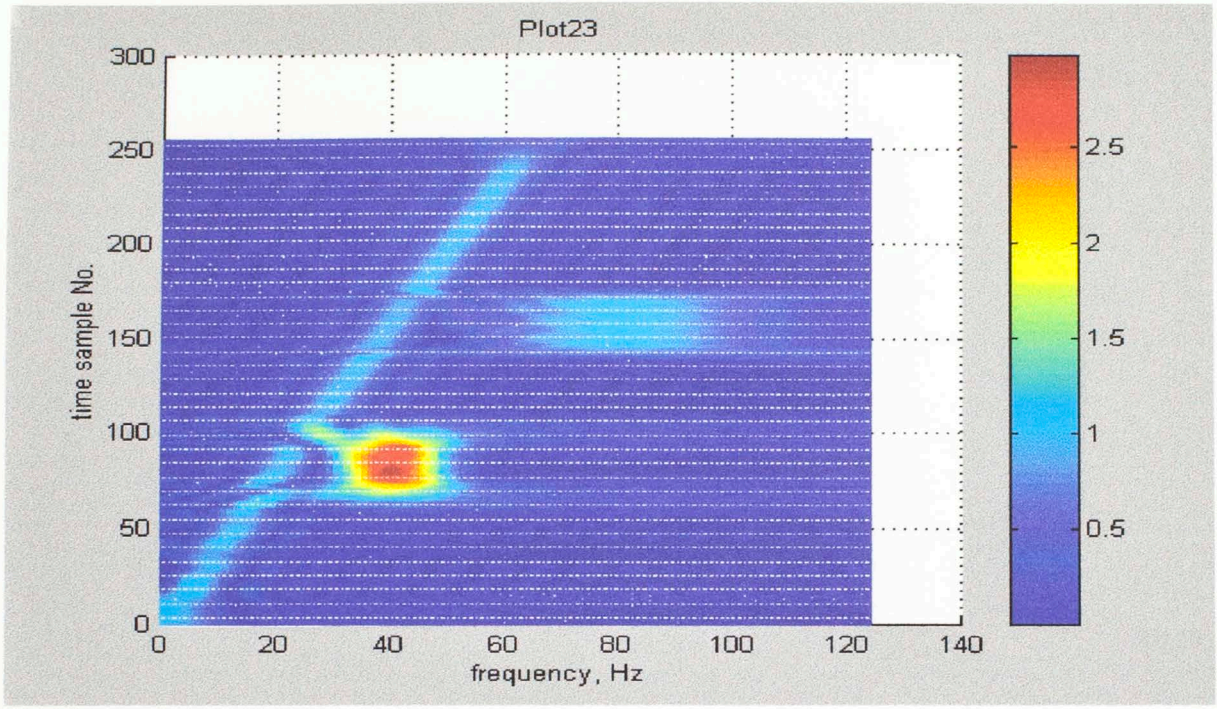

Figure 23. The STFT of Plot 21 with a boxcar window, with the window length of 32 samples.

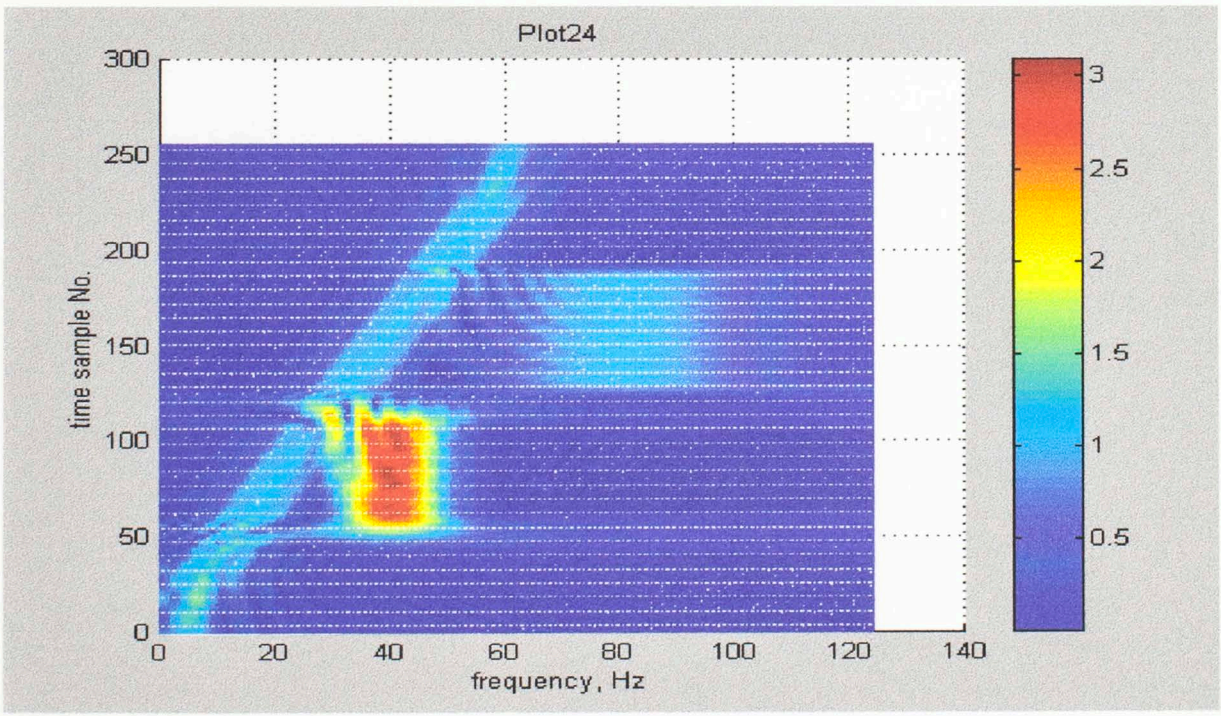

Figure 24. The STFT of Plot 21 with a boxcar window, with the window length of 64 samples.

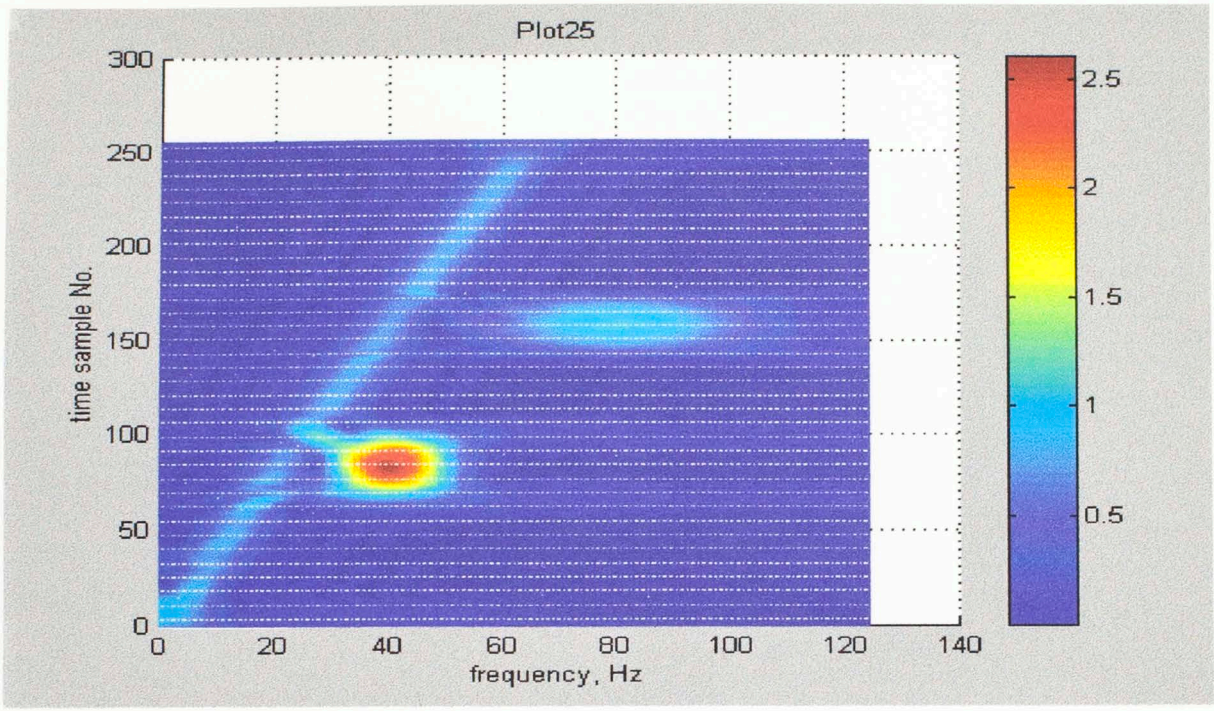

Figure 25. The STFT of Plot 21 with a Gaussian window, with the window length of 32 samples.

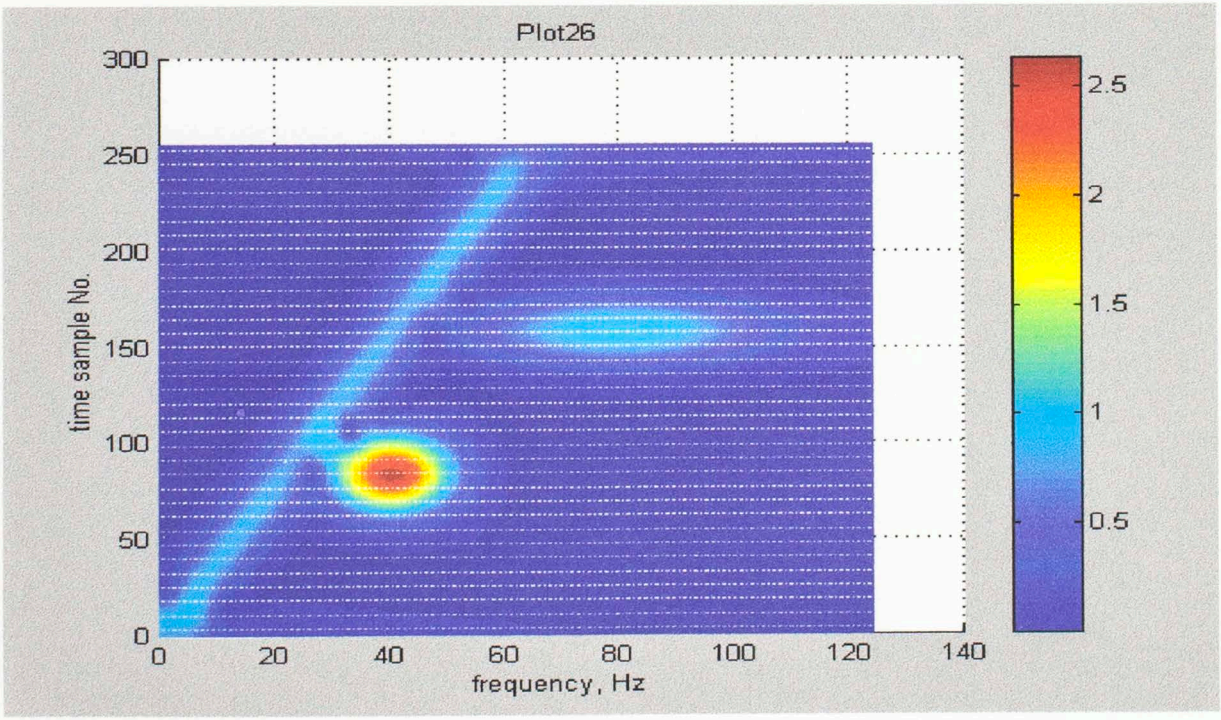

Figure 26. The STFT of Plot 21 with a Gaussian window, with the window length of 64 samples.

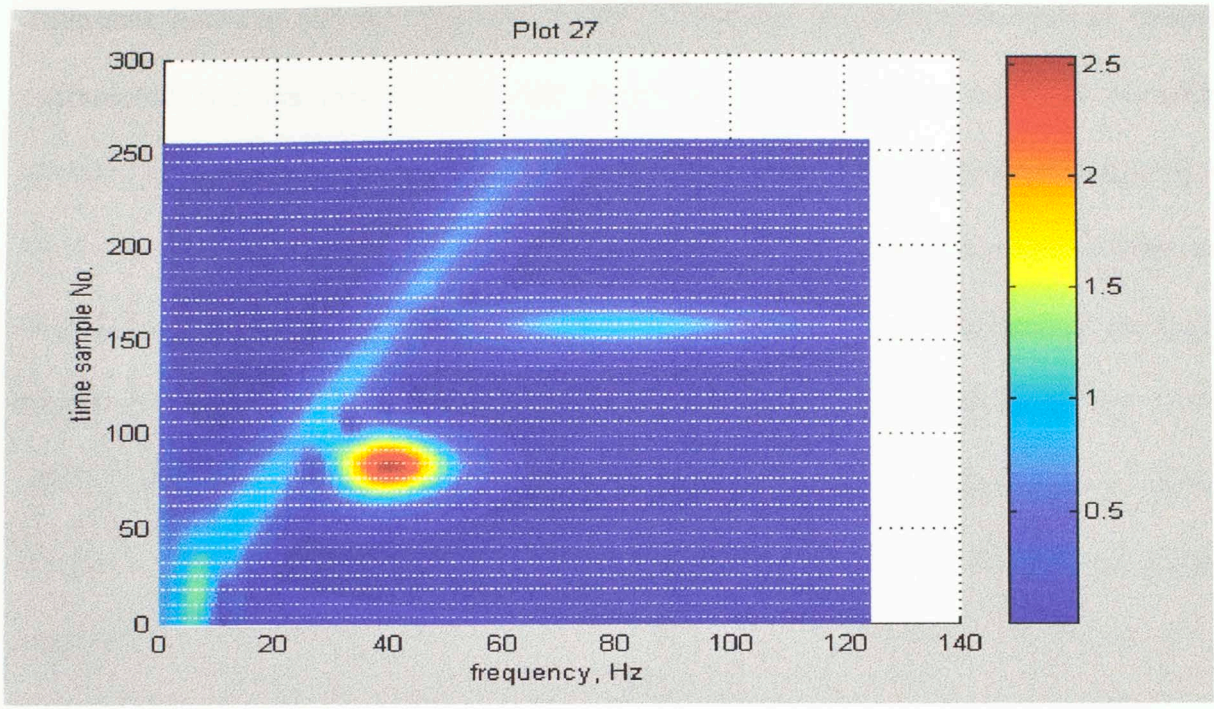

Figure 27. The CWT of Plot 21.

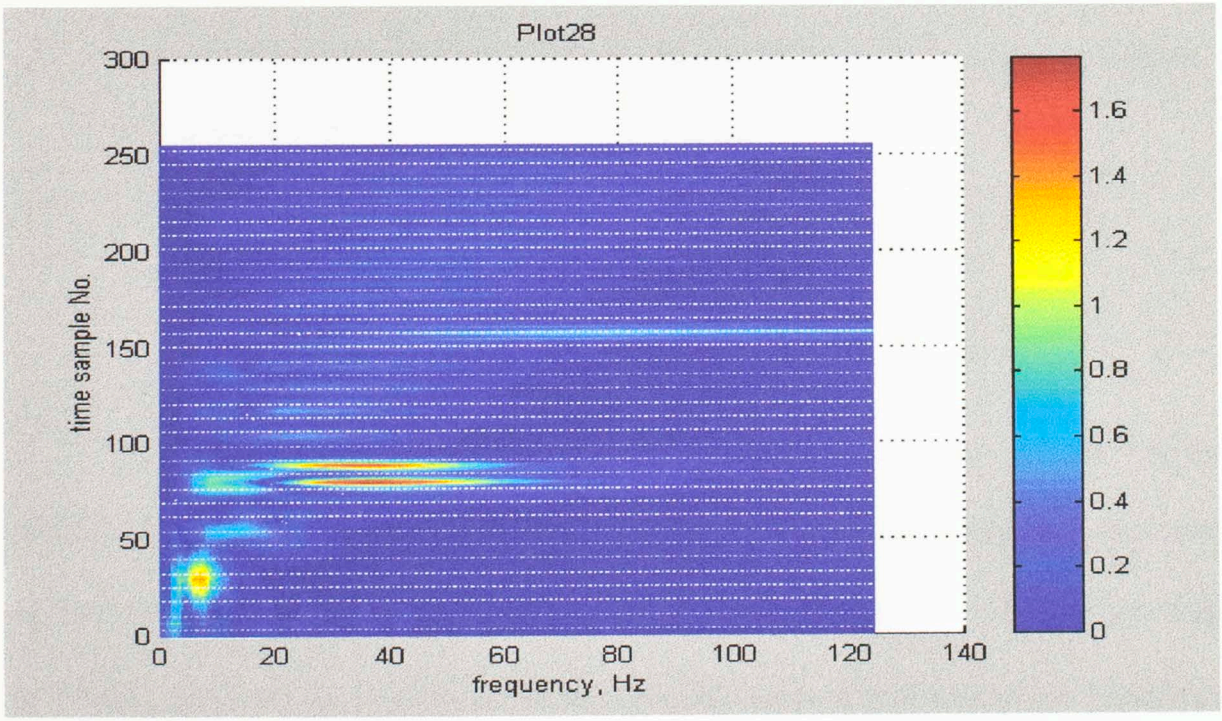

Figure 28. The MPD of Plot 21.

difference between the two figures. A linear frequency increase with time is cleared represented and the two wavelets are shown with the frequency contents correctly centered at 40 Hz and 80 Hz. However, from the representation of the wavelet centered at 40 Hz, the time localization of Figure 26 is a little worse than Figure 25 because of the use of a bigger window. Figure 27 gives the CWT of the signal. Compared with all the time-frequency representations given by the STFT, the representation given by the CWT provides the best time and frequency representation. Notice the representation of the wavelet centered at 80 Hz, it is well-localized in time because of its high frequency content.

Discrete Wavelet Transform

In the Continuous Wavelet Transform, the wavelet family is generated as follows.

$$\Psi_{(b,a)}(x) = \frac{1}{\sqrt{|a|}}\Psi(\frac{x-b}{a}) \quad (2.2.1)$$

The Discrete Wavelet Transform (DWT) is the discretization of the wavelet transform. In the Discrete Wavelet Transform (DWT), the values of a and b are restricted to discrete values on the time-frequency plane. The discrete wavelet bases are generated as $\{\Psi_{(m,n)}, m \in Z, n \in Z. \}$. There are several possible ways of discretizing the transforms (especially in the wavelet case), and which method to choose depends on the application.

Generally the scale is defined as $a = a_0^{\ m}$, and the translation is defined as $b = nb_0a$, where $a_0 \in \Re$, $b_0 \in Z$.

$$\Psi_{(m,n)} = \frac{1}{\sqrt{|a_0|^m}} \Psi\left(\frac{x - nb_0|a_0|^m}{|a_0|^m}\right) \qquad (2.2.2)$$

The DWT is therefore defined as follows.

$$T_{f\,(m,n)} = \langle f, \Psi_{(m,n)} \rangle = \int_{-\infty}^{+\infty} f(x)\, \Psi^*_{(m,n)}(x)\, dx \qquad (2.2.3)$$

(* denotes the complex conjugate.)

One concern about the DWT is how to build a complete discretized wavelet basis that could represent any signal f in the Hilbert space sufficiently, and to reconstruct the f in a numerically stable way from the $T_{f(m,n)}$. Any function to be used as the kernel wavelet needs to meet the following admissibility conditions (Shensa, 1992; Goupillaud et al., 1985):

1. $\Psi(t)$ should be absolutely integrable and square integrable (i.e., its energy is finite):

$$\int |\Psi(t)|\, dt < \infty \qquad (2.2.4)$$

and

$$\int |\Psi(t)|^2 dt < \infty \qquad (2.2.5)$$

2. $\psi(\omega)$ is band limited and has zero mean, where $\psi(\omega)$ is the Fourier Transform of $\psi(t)$;

$$\int \left| \frac{\Psi(\omega)}{\omega} \right| d\omega < \infty \qquad (2.2.6)$$

After a discretized admissible wavelet basis has been built, the problem left is to select the algorithm to perform the DWT. In the following part, three algorithms are introduced, the multi-resolution analysis, the wavelet packet analysis and the matching pursuit decomposition. The first two algorithms are mainly about how to generate an orthonormal wavelet basis and provide a globally optimized time-frequency representation on an orthonormal wavelet basis. The last one is mainly concerned with providing a locally optimized time-frequency representation on an unorthonormal wavelet basis.

Chapter 3 Multi-Resolution Analysis

The concept of Multi-Resolution Analysis (MRA) was introduced and perfected by Meyer and Mallat (1989). It gives a formal and detailed description of the intuitive idea that every signal can be constructed by a successive refinement, by adding details to approximation, and by iterating this process. It is performed on an orthogonal wavelet basis. The method involves the use of the Quadrature Mirror Filter (QMF), which is composed of two filters, a lowpass filter and a highpass filter. In a forward analysis, the signal is filtered by the two filters. The output of the lowpass filter is the approximation and the output of the highpass filter is the detail at the corresponding resolution level. At each iteration, the approximation at that resolution level is generated by the lowpass filter and the detail is generated by the highpass filter. The approximation is down-sampled by a factor of two and is sent to further iteration until a desired accuracy is achieved. In the inverse synthesis at a certain resolution level, the approximation and the detail at the next lower resolution level are up-sampled (inserting a zero between two successive samples), then the two filters are performed respectively and the approximation at that resolution level is achieved by adding the outputs from the two filters.

Precisely, a multi-resolution analysis of $L(\Re)$ is defined as follows.

(i) There exists a sequence of nested subspaces

$$\ldots \supset V_1 \supset V_0 \supset V_{-1} \supset \ldots; \quad (3.1)$$

(ii) $$\bigcap_{m\in Z} V_m = \{0\}; \qquad (3.2)$$

(iii) $$\bigcup_{m\in Z} V_m \text{ is dense in L;} \qquad (3.3)$$

(iv) $$x(t) \in V_m \Leftrightarrow x(2t) \in V_{m+1}; \qquad (3.4)$$

(v) There exists a function Φ so that $\{\Phi(t-n) ; n \in Z\}$ is a basis of V_0. (3.5)

A time-resolution of 2^m could be associated with each V_m, and the approximation of a signal x(t) at this resolution level is obtained by the projection onto the corresponding subspace. Due to the condition (iv), a basis of V_m can be derived from the basis of V_0 as follows:

$$\Phi_{nm}(x) = 2^{(m/2)} \Phi(2^m x-n) \qquad (3.6)$$

The approximation of signal f(x) at the resolution level of 2^m is

$$a_x[n, m] = \int_{-\infty}^{+\infty} f(x) \Phi_{nm}(x)\, dx \qquad (3.7)$$

At each approximation (at the resolution level 2^m), some information of the signal f(x) is lost, the lost information is called the detail at the resolution 2^m. The detail at the resolution of 2^m could be achieved by the orthogonal projection of the original signal on the orthogonal complement of V_m in V_{m+1}. It has been proved that a space W_m exists at

each resolution level 2^m that is the orthogonal complement of V_m in V_{m+1}. The relation is

$$V_{m+1} = V_m \oplus W_m \qquad (3.8)$$

W_m is the subspace that satisfies the five conditions described above and is the orthogonal complement of V_m. Therefore there exists a function $\Psi(t)$ such that $\{\Psi(t-n); n \in Z\}$ is a basis of W_0 and $\{\Psi_{nm}(t) = 2^{(-m/2)} \Psi(2^{-m} x-n); n, m \in Z\}$ constitutes an orthogonal bases of W_m. Hence the detail of signal f(x) at the resolution level of 2^m is

$$d_x [n, m] = \int_{-\infty}^{+\infty} f(x) \Psi_{nm}(x) dx \qquad (3.9)$$

Since $\Phi \in V_0 \subset V_{-1}$, and the $\Phi_{-1,n}$ are an orthonormal basis in V_{-1}, we have

$$\Phi(t) = \sqrt{2} \sum_n h[n] \Phi(2t - n) \qquad (3.10)$$

with

$$h[n] = \sqrt{2} \int_{-\infty}^{+\infty} \Phi(t) \Phi^*(2t - n) dt, \qquad \sum_{n=-\infty}^{n=+\infty} h^2[n] = 1. \qquad (3.11)$$

Similarly, we have

$$\Psi(t) = \sqrt{2}\sum_{n} g[n]\Psi(2t-n) \qquad (3.12)$$

with

$$g[n] = \sqrt{2}\int_{-\infty}^{+\infty}\Psi(t)\Phi^{*}(2t-n)dt\,, \qquad \sum_{n=-\infty}^{n=+\infty} g^{2}[n] = 1. \qquad (3.13)$$

Thus, a pair of quadrature mirror filters are defined as h[n] and g[n] among which h[n] is a lowpass filter and g[n] is a highpass filter. By using the two filters, a fast algorithm to obtain the approximation and the detail of the singal from the next higher approximation could be derived. The method is called the pyramidal algorithm and is defined as follows.

According to the equation (3.7)

$$a_x[n,m] = \int_{-\infty}^{+\infty} \mathrm{f(x)}\ \Phi_{nm}(\mathrm{x})\ \mathrm{dx}$$

$$= \int_{-\infty}^{+\infty} f(x)2^{m/2}\Phi(2^{m}x-n)dx$$

$$= \int_{-\infty}^{+\infty} f(x)2^{m/2}[\sqrt{2}\sum_{k=-\infty}^{+\infty} h[k]\Phi(2(2^{m}x-n)-k)]dx$$

$$= \sum_{k=-\infty}^{+\infty} h[k]\int_{-\infty}^{+\infty} f(x)2^{(m+1)/2}\Phi(2^{m+1}x-(k+2n))dx$$

$$= \sum_{k=-\infty}^{+\infty} h[k]a_x[k+2n,m+1]$$

Thus, we have proved the formula

$$a_x[n,m] = \sum_{k=-\infty}^{+\infty} h[k-2n]a_x[k,m+1] \tag{3.14}$$

Similarly, we have

$$d_x[n,m] = \sum_{k=-\infty}^{+\infty} g[k-2n]a_x[k,m+1] \tag{3.15}$$

Hence the coefficients of the approximation and the detail at a fixed resolution level can be derived by filtering the known approximation coefficients at the next higher level, followed by down-sampling. Operating step by step, a fast and recursive algorithm, which involves only the use of two filters, is achieved. Such an algorithm is called a pyramidal algorithm. The initialization of the algorithm is performed by a projection of the analyzed signal onto V_0.

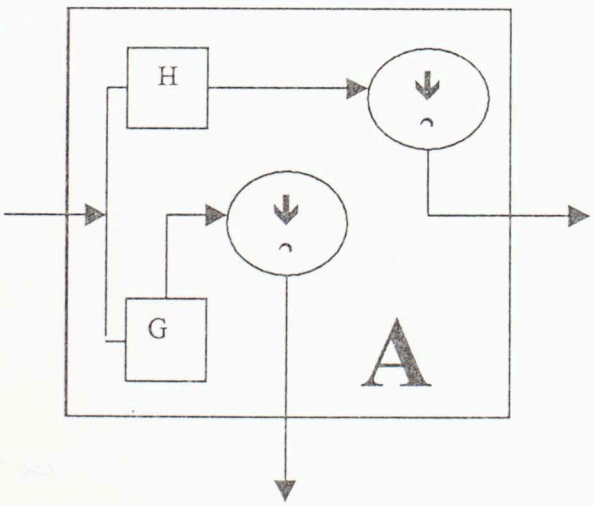

Figure 3.1. Illustration for the forward analysis of the pyramidal algorithm. The unit A maps an approximation to a coarser approximation and a detail by the action of a lowpass filter H and a highpass filter G (with impulse responses h[n] and g[n] respectively), both followed by a downsampling by a factor of 2.

The *analysis* algorithm could be inverted, which leads to a dual algorithm for the *synthesis*. It is an algorithm to derive the approximation at a fixed resolution level from the approximation and the detail at the next lower level. It is defined as follows.

$$a_x[n,m+1] = \sum_{k=-\infty}^{+\infty} h[n-2k]a_x[k,m] + \sum_{k=-\infty}^{+\infty} g[n-2k]d_x[k,m] \qquad (3.19)$$

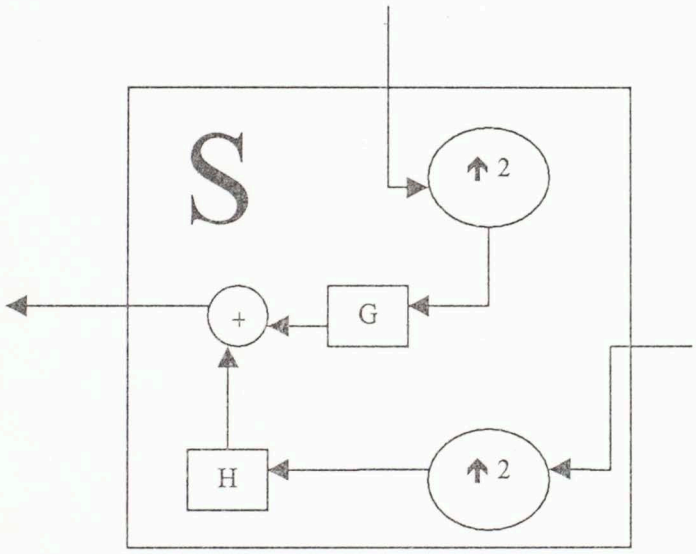

Figure 3.2, illustration of the inverse analysis. Each unit S operates on an approximation and a detail by producing a finer approximation. It first performs an interpolation (inserting a zero between two successive samples), then a filtering by h[n] and g[n] respectively, and finally adds the obtained output signals.

The MRA is a set of scale-invariant operations that enables us to interpret features of signal at different resolution levels. At different resolutions, the details of the representation generally characterize different structure features of the signal. It is not redundant because of the use of the orthogonal wavelet bases and it also provides a

natural framework for the understanding of wavelet bases and for the construction of the orthogonal bases. Numerous MRAs have been proposed corresponding to different orthogonal wavelet bases. Among them are the Littlewood-Paley-type MRAs by Y. Meyer, the spline MRAs by G. Battle and P. G. Lemarie, and the compactly supported wavelet MRAs by I. Daubechies. The detailed deduction of how to generate an orthogonal wavelet basis based on the properties given in conditions (3.1) to (3.5) is given in Daubechies book (Reference 8).

Chapter 4 Wavelet Packet

Multi-Resolution Analysis provides a scale-invariant interpretation of the signal, which means it divides the frequency axis by the same intervals. Therefore, it has some consequence for the localization properties in the time-frequency domain. However, complex time-frequency structures are always encountered in practice. It is thus desirable to be able to modify the time-frequency localization property of the wavelet basis to adapt to the features of specific signals and problems. Orthonormal wavelet packet bases use quadrature mirror filters to split the parent wavelet space into two subspaces at next level and then each subspace is further split into two new subspaces at higher level. The wavelet spaces at different levels are associated with different frequency resolution. The splitting is repeated until the desired resolution level is met. As a result, many wavelet spaces are obtained. By the "best basis algorithm", we could choose from all the wavelet spaces at different levels a basis that matches the time-frequency features of the signal best. The chosen wavelet spaces divide the frequency axis in separate intervals of various sizes and translate in time uniformly. Thus it provides a representation that is well adapted to the situation in which the signal has highly variant frequency contents.

Wavelet Packet algorithm was introduced by Coifman, Meyer, and Wickerhauser by generalizing the link between multi-resolution approximations and wavelets. In multi-resolution analysis, a space V_j is decomposed in a lower resolution space V_{j+1} plus a detail space W_{j+1}. This is done by dividing the orthogonal basis $\{\Phi_j (t - 2^j n), n \in Z, j \in Z\}$ into two new orthogonal bases

$$\{\Phi_{j+1}(t - 2^{j+1} n), n \in Z, j \in Z\} \text{ of } V_{j+1} \quad (4.1)$$

and

$$\{\Psi_{j+1}(t - 2^{j+1} n), n \in Z, j \in Z\} \text{ of } W_{j+1} \quad (4.2)$$

The Φ_{j+1} and Ψ_{j+1} are calculated by the QMFs h[n] and g[n], in which

$$\Phi_{j+1} = \sum_{n=-\infty}^{n=+\infty} h[n]\Phi_j(t - 2^j n) \quad (4.3)$$

and

$$\Psi_{j+1} = \sum_{n=-\infty}^{n=+\infty} g[n]\Phi_j(t - 2^j n) \quad (4.4)$$

By generalizing the result above to any space U_j , let h[n] and g[n] be a pair of QMFs. Let $\{\Psi_j(t - 2^j n); j,n \in Z\}$ be an orthonormal basis of the space U_j. Define

$$\Psi^0_{j+1} = \sum_{n=-\infty}^{n=+\infty} h[n]\Psi_j(t - 2^j n) \quad (4.5)$$

and

$$\Psi_{j+1}^{1} = \sum_{n=-\infty}^{n=+\infty} g[n]\Psi_{j}(t - 2^{j}n) \qquad (4.6)$$

According to the equation (3.8), the family

$$\{ \Psi_{j+1}^{0}(t - 2^{j} n),\ \Psi_{j+1}^{1}(t - 2^{j} n),\ n \in Z\} \qquad (4.7)$$

is equivalent to the orthonormal basis of U_j, which is $\{\Psi_j(t - 2^j n); j,n \in Z\}$.

Hence instead of dividing only the approximation space V_{j-1} to construct detail spaces W_j and the approximation space V_j, we could set $U_j = W_j$ and divide these detail spaces to derive new bases. The recursive splitting of vector spaces is represented in a binary tree. Any node W_j^p of a binary tree is labeled by its depth j and the number p of nodes that are on its left at the depth j, as illustrated in Figure (4.1).

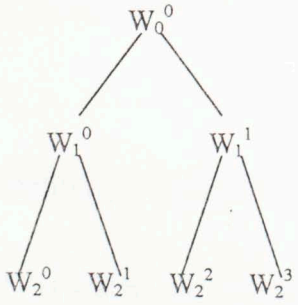

Figure 4.1, Wavelet basis splitting of the Binary tree.

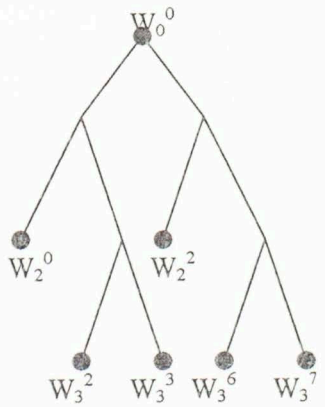

Figure 4.2, Example of admissible wavelet packet binary tree

Each node (j, p) of the binary tree corresponds to a space W_j^p, which admits an orthonormal basis $\{\{\Psi_j^p(t - 2^j n); j,p \in Z\}$. At the root of the tree, we set $W_0^0 = V_0$. This approximation space admits an orthogonal basis of scaling functions $\{\phi(t-n), n \in Z\}$, so $\Phi_0^0 = \Phi$. Now suppose W_j^p and its orthonormal basis $W_j^p = \{\Psi_j^p(t - 2^j n); j, p, n \in Z\}$ have been constructed at the node (j,p). The two orthogonal bases at the children nodes are defined by the splitting relations (4.5 and 4.6):

$$\Psi_{j+1}^{2p} = \sum_{n=-\infty}^{+\infty} h[n]\Psi_j^p(t - 2^j n) \qquad (4.8)$$

and

$$\Psi_{j+1}^{2p+1} = \sum_{n=-\infty}^{+\infty} g[n]\Psi_j^p(t - 2^j n) \qquad (4.9)$$

because $\{\Psi_j^p(t - 2^j n), n \in Z\}$ is orthonormal.

From relation (3.8), the two orthonormal bases $W_{j+1}^{2p} = \{\Psi_{j+1}^{2p}(t - 2^j n), n \in Z\}$ and $W_{j+1}^{2p+1} = \{\Psi_{j+1}^{2p+1}(t - 2^j n), n \in Z\}$ have the following relationship with W_j^p.

$$W_{j+1}^{2p} \oplus W_{j+1}^{2p+1} = W_j^p \qquad (4.10)$$

A binary tree of wavelet packet spaces could be obtained by performing the splitting defined above recursively where each node is divided in two orthogonal subspaces (as shown by the Figure (4.1)). The wavelet at the depth j of the binary tree specifies the scale 2^j of the time support. Figure 63 is an example of the eight wavelet spaces at the depth 3 generated from the Haar wavelet.

The best basis algorithm (Coifman, Wickerhauser, 1992) selects an "optimal" orthonormal basis within the wave packet binary tree. It is a procedure to choose a wavelet basis (an example is given in Figure (4.2)) that globally adapts to the signal. In their algorithm, the wavelet basis selects a basis that provides the minimum Shannon entropy defined as follows:

$$\sum_{n=1}^{N} |< f, \Psi_{\gamma_n} >|^2 \log_2 (|< f, \Psi_{\gamma_n} >|^2) \qquad (4.11)$$

The choice of this "optimal" orthogonal basis is thus obtained through a global minimization over the entire wavelet bases provided by the binary tree. Figure 64, Figure 65 and Figure 66 provide the time-frequency representation of three different signals using the best tree algorithm. The Haar wavelet is used in all the wavelet packet processing. On each figure, the left subplot is the wavelet basis chosen by the best tree algorithm introduced above. The subplot on the upper right is the signal input. The subplot on the bottom right is the time-frequency representation given by the wavelet packet, in which

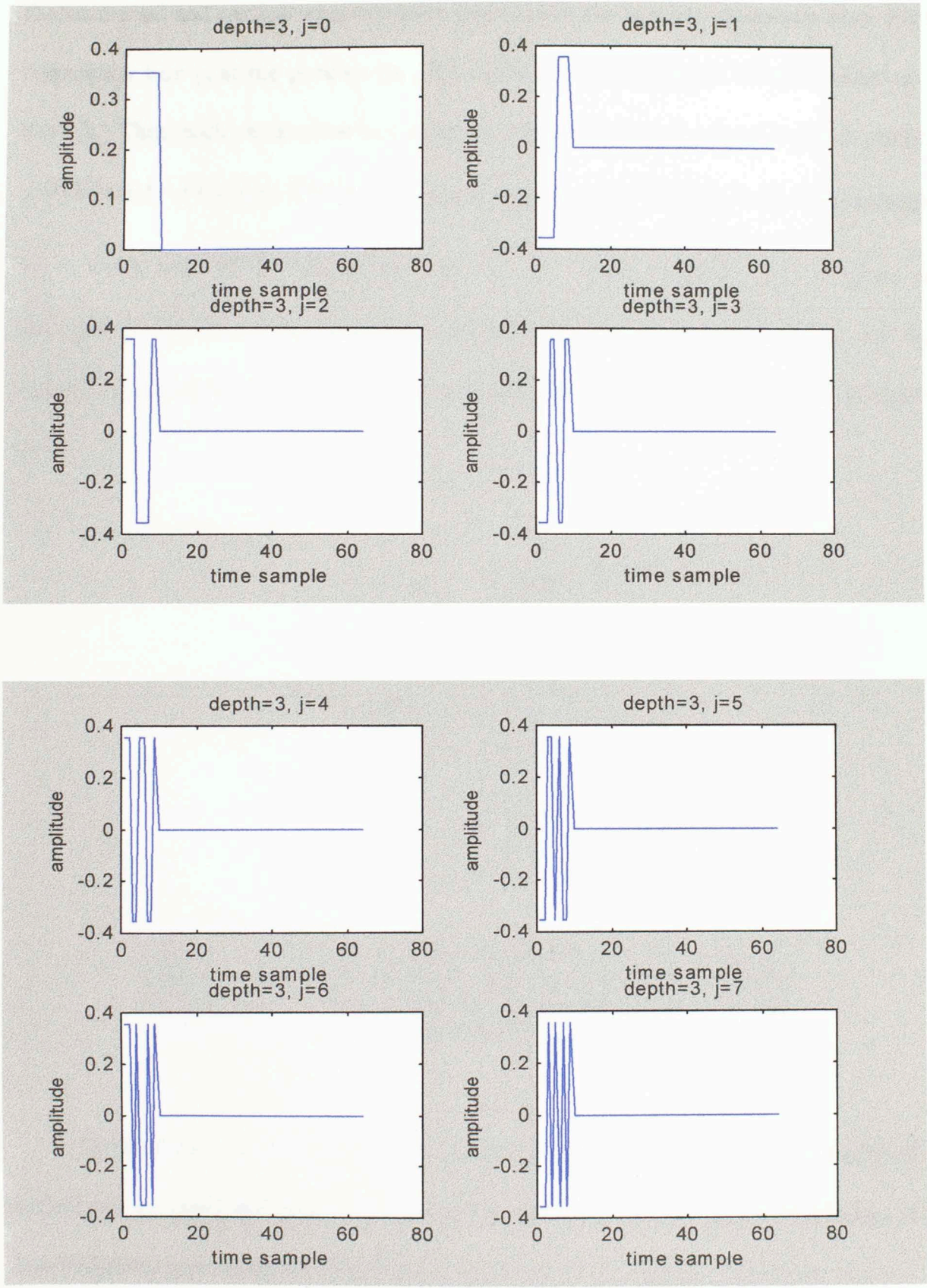

Figure 63. The Harr wavelet basis from the wavelet packet binary tree at the depth of 3.

the horizontal and vertical axes represent respectively the time and frequency axes. Each rectangular box γ_n at the position $[p_n , k_n]$ represents the coefficient of the wavelet at a level j_n . Thus each rectangular box could be represented by $[j_n, p_n, k_n]$ and it centers approximately at the time $2^{j_n}(p_n + \frac{1}{2})$ and at the frequency $2^{-j_n}(k_n + \frac{1}{2})$. The rectangle has a width scale of 2^{j_n} along time axis and 2^{-j_n} along frequency axis. It gives an approximate idea of the localization in time and frequency of each atom $[j_n, p_n, k_n]$. An example of a best basis and how it divides the frequency and time axes is given in Figure (4.3).

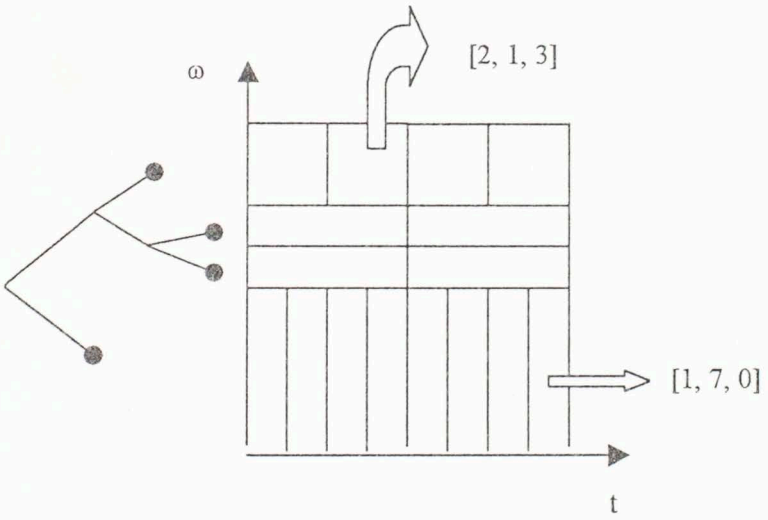

Figure 4.3, An example of a best tree and how the tree divides the frequency and time axes. The wavelet packet tree on the left divides the frequency axis in several intervals. The Heisenberg boxes of the corresponding wavelet packet basis are on the right.

In my thesis, several examples of the processing using the "best tree algorithm" method are provided with the use of the matlab wavelet toolbox. Figure 64 provides the time-frequency representation of a linear chirp (Figure 15). Figure 65 provides the time-frequency representation of a linear chirp with two wavelets (Figure 21). Figure 66

provides the time-frequency representation of a synthetic seismic trace (Figure 29). The time-frequency distribution achieved by the method is similar to the CWT. However, some signal features are not as clear as the CWT. The frequency localization is not as satisfactory as the CWT because the localization property of the Haar wavelets is not as good as the Gabor atoms. In the thesis, the effect of the method will not be dwelled on a lot. The examples are mainly used to compare with the results from several other wavelet methods.

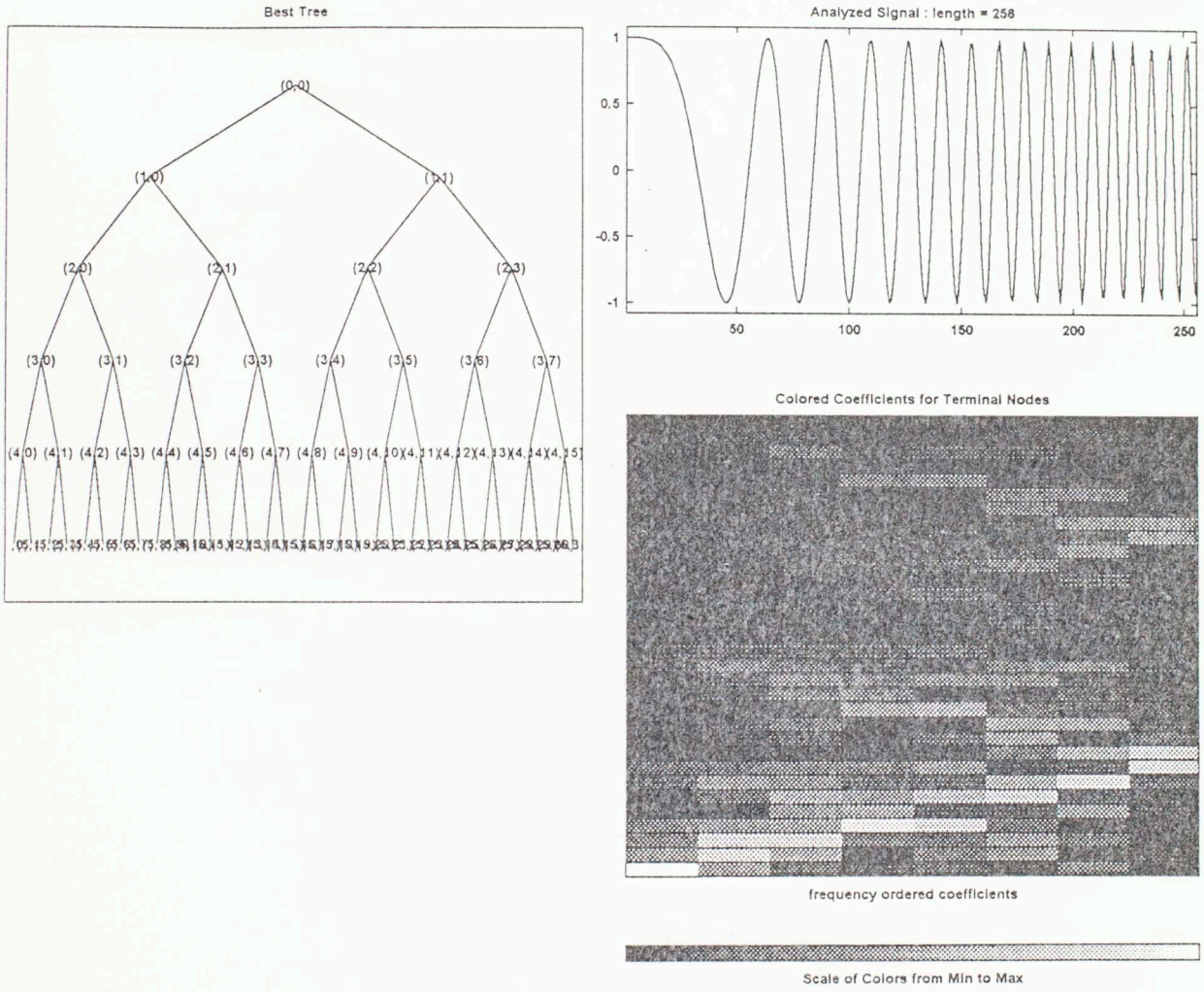

Figure 64. The time-frequency represention of a linear chirp provided by the best tree algorithm. The subplot on the upper left is the best tree selected from the binary tree. The subplot on the upper right is input data. The subplot on the bottom right is the time-frequency representation given by the best tree algorithm. The horizontal and vertical axes are the time and frequency axes respectively.

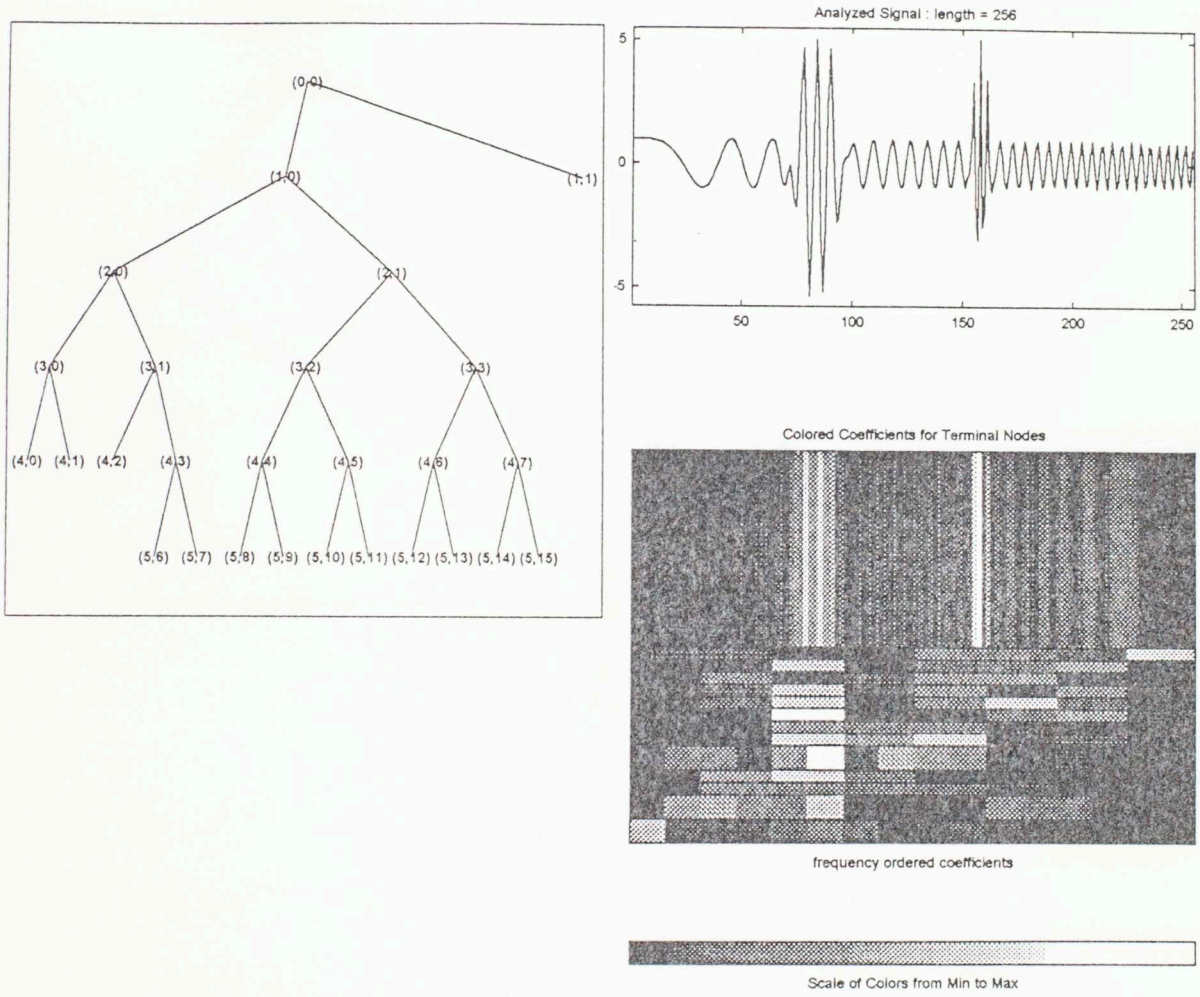

Figure 65. The time-frequency represention of a linear chirp plus two wavelets provided by the best tree algorithm. The subplot on the upper left is the best tree selected from the binary tree. The subplot on the upper right is input data. The subplot on the bottom right is the time-frequency representation given by the best tree algorithm. The horizontal and vertical axes are the time and frequency axes respectively.

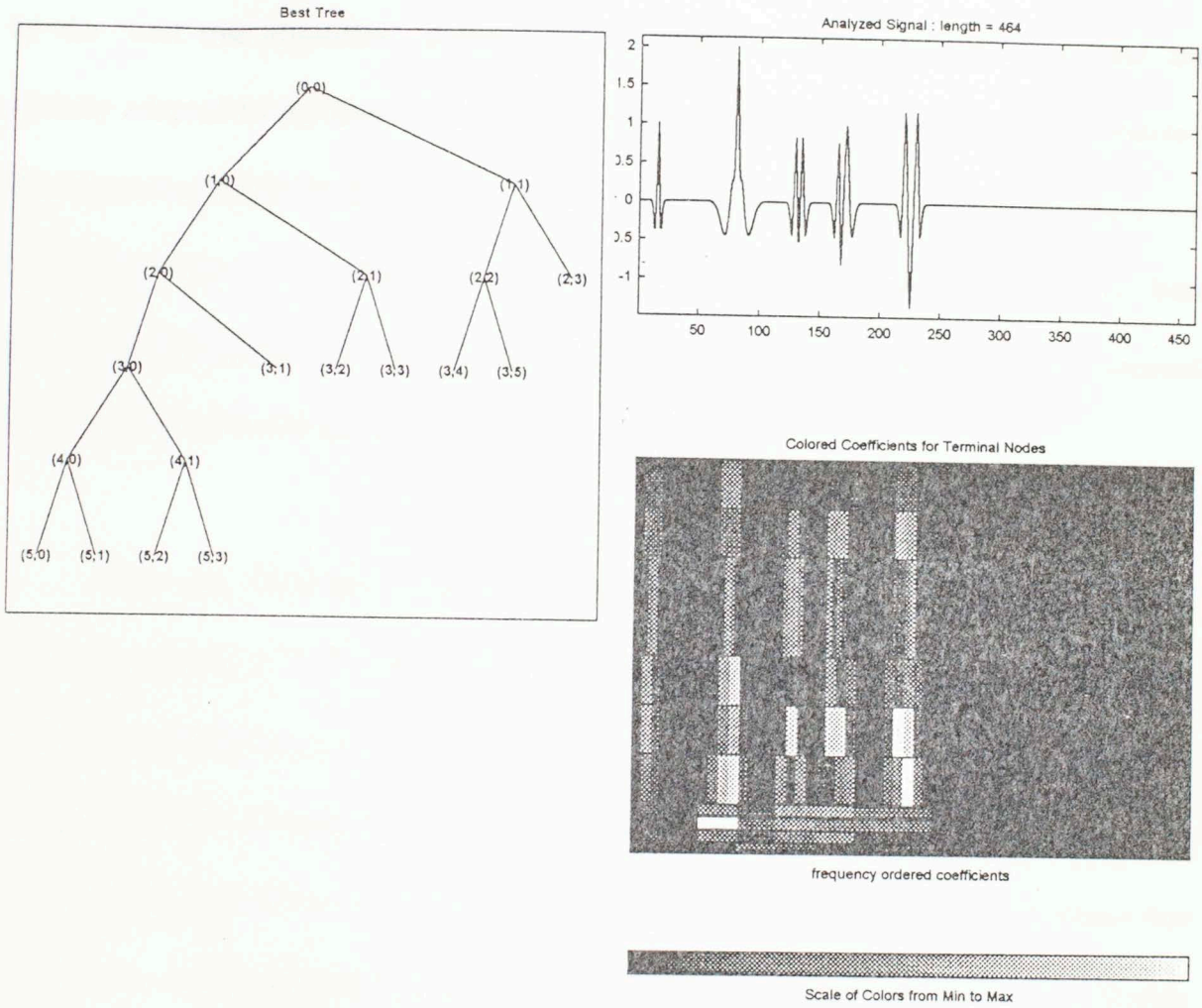

Figure 66. The time-frequency represention of a synthetic seismic trace provided by the best tree algorithm. The subplot on the upper left is the best tree selected from the binary tree. The subplot on the upper right is input data. The subplot on the bottom right is the time-frequency representation given by the best tree algorithm. The horizontal and vertical axes are the time and frequency axes respectively.

Chapter 5 Matching Pursuit Decomposition

A wavelet packet tree provides large number of families of orthogonal bases that include different types of local time-frequency atoms. A best wavelet packet basis obtained by the "best tree algorithm" decomposes the signal in time-frequency atoms that are globally adapted to the time-frequency structures of the signal. It is a globally optimized time-frequency representation of the signal. However, it can not work well where the signal has great non-stationary time-frequency structures. For example, there is no best wavelet packet basis if the signal includes different types of high-energy structures, located at different times but in the same frequency intervals.

Thus, the Matching Pursuit Decomposition (MPD) was introduced by Mallat (1993) to provide a flexible time-frequency representation of the signal that has highly variant local time-frequency structures. It is an algorithm that decomposes any signal into a linear expansion of waveforms that belong to a redundant dictionary of functions. At each iteration, a waveform is selected from a large dictionary of time-frequency atoms that matches the local structures at approximate part best. Therefore, it provides flexible decompositions that are particularly adapted to signal components whose localization in time and frequency varies widely. If a structure does not correlate well to any particular wavelet basis element, it is sub-decomposed into several elements and its information is diluted into these elements.

Ricker Wavelet Family

In time-frequency analysis, it is often desired to represent a signal with as few wavelet vectors as possible. Therefore, a wavelet basis that is well adapted to the signal is desirable. Ricker wavelets are one of the most frequently used and the most fundamental wavelets in seismology. In this thesis, a family of wavelets is defined by dilating, translating and modulating a single wavelet function g (t) $\in L^2(\Re)$.

The Ricker wavelet is defined as follows.

$$g(t,f) = (1-2\pi^2 f^2 t^2) \exp(-\pi^2 f^2 t^2) \qquad (5.1)$$

For any scale s > 0, frequency modulation ξ and translation τ, we denote $\gamma = (s, \tau, \xi)$. The index γ is an element of the set $\Gamma = \Re^+ \times \Re^2$. A family of wavelet functions is generated by scaling, translating, and modulating a single wavelet function as

$$g_\gamma(t,f) = \frac{1}{\sqrt{s}} g\left(\frac{t-\tau}{s}, \xi\right) \qquad (5.2)$$

A Ricker wavelet family is a family of non-orthogonal bases composed of wavelets that are well localized both in time and frequency (Figure 5.1). It is the second derivative of the Gaussian function and also satisfies the admissible conditions as shown in equations from (3.3) to (3.5). By using the Ricker wavelets, the seismic signal is decomposed into

fewer wavelets. As a result, a clear representation is achieved. The examples of the time series and the frequency spectrum of a 40-Hz Ricker wavelet are given as follows:

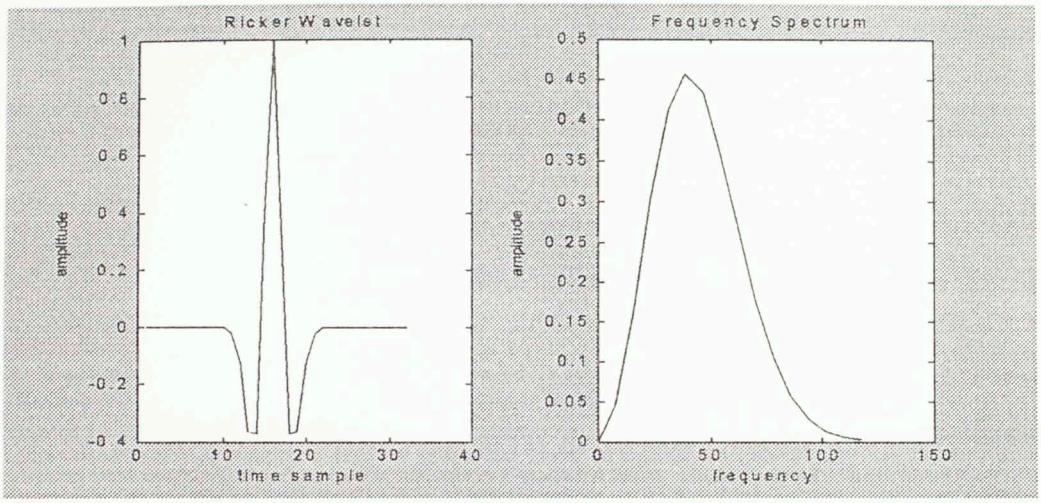

Figure 5.1, An example of Ricker wavelet with the central frequency of 40 Hz and its frequency spectrum.

Matching Pursuit Decomposition

After a dictionary is defined as a family $D = (g_\gamma)_{\gamma \in \Gamma}$ of vectors in a Hilbert space $H = L^2(\Re)$, the Matching Pursuit Decomposition is defined as the follows.

Let $f \in H$. We want to compute a linear expansion of f over a set of vectors selected from D that match its inner structures best. This is done by successive approximations of f with orthogonal projections on elements of D. Let $g_{\gamma 0} \in D$, the vector f can be decomposed into

$$f = \langle f, g_{\gamma 0} \rangle g_{\gamma 0} + Rf \qquad (5.3)$$

where Rf is the residual vector after approximating f in the direction of $g_{\gamma 0}$. If an orthogonal wavelet basis is used, $g_{\gamma 0}$ is orthogonal to Rf, hence the energy conservation is maintained.

$$\| f \|^2 = | \langle f, g_{\gamma 0} \rangle |^2 + \| Rf \|^2 \quad (5.4)$$

To minimize $\| Rf \|$, we must choose $g_{\gamma 0} \in D$ such that $| \langle f, g_{\gamma 0} \rangle |$ is maximum. In some cases, it is only possible to find a vector $g_{\gamma 0}$ that is almost the best in the sense that

$$| \langle f, g_{\gamma 0} \rangle | \geq \alpha \sup_{\gamma \in \Gamma} | \langle f, g_{\gamma} \rangle | \quad (5.5)$$

where α is an optimality factor that satisfies $0 < \alpha \leq 1$.

A MPD is an iterative algorithm that sub-decomposes the residue Rf by projecting it on a vector of D that matches Rf best, as was done for f. This procedure is repeated each time on the resulting residue that is obtained. The choice of a vector $g_{\gamma 0}$ is defined by a choice function C, which associates to any subset Λ of Γ an index that belongs to Λ, which is defined as

$$\Lambda_0 = \{\beta \in \Gamma : | \langle f, g_{\beta} \rangle | \geq \alpha \sup_{\gamma \in \Gamma} | \langle f, g_{\gamma} \rangle |\} \quad (5.6)$$

The choice of a vector $g_{\gamma 0}$ that satisfies relation (5.5) is equivalent to the choice of the index γ_0 within Λ_0, formally defined by $\gamma_0 = C(\Lambda_0)$. At each iteration, there exists at least one choice of wavelet. In practice, there may exist many ways to define the choice function, and it depends upon the specific implementation.

Let $R^0 f = f$. Suppose that we have computed the *nth* order residue $R^n f$, for $n \geq 0$. We choose, with the choice function C, an element $g_{\gamma n} \in D$ that closely matches the residue $R^n f$

$$| \langle R^n f, g_{\gamma n} \rangle | \geq \alpha \sup_{\beta \in \Gamma} | \langle R^n f, g_\beta \rangle | \quad (5.7)$$

The residue $R^n f$ is sub-decomposed into

$$R^n f = \langle R^n f, g_{\gamma n} \rangle g_{\gamma n} + R^{n+1} f \quad (5.8)$$

which defines the residue at the order $n+1$. With the use of an orthogonal wavelet base, $R^{n+1} f$ is orthogonal to $g_{\gamma n}$ and the energy conservation is maintained.

$$\| R^n f \|^2 = | \langle R^n f, g_{\gamma n} \rangle |^2 + \| R^{n+1} f \|^2 \quad (5.9)$$

Suppose the decomposition is carried up to the order m, then f is decomposed into the concatenated sum

$$f = \sum_{n=1}^{m-1} < R^n f, g_{\gamma_n} > g_{\gamma_n} + R^m f \qquad (5.10)$$

The original vector f is decomposed into a sum of dictionary elements that are chosen to match its residues best. When m increases to infinite, the residue $R^m f$ approaches infinity small. The lower the correlation between the signal residues and the dictionary elements, the slower the decay of the residues. Because Ricker wavelet basis is a non-orthogonal wavelet basis, energy conservation is not maintained in the MPD processing in this thesis.

The MPD is a redundant representation of a signal as an additive superposition of elementary functions with a sharp localization in the time-frequency plane. It works especially well where the signal has highly non-stationary local features.

Data Analysis

A linear chirp (Figure 15) is used to test the effect of the MPD on the signal with continuous frequency distribution. From the comparison of the MPD (Figure 20) with the CWT (Figure 19), both methods provide approximately the same time-frequency distribution of the signal, although the MPD is not as good as the CWT in providing the time-frequency representation of a signal with continuous frequency distribution because of sharp localization feature in the time- frequency plane of the MPD.

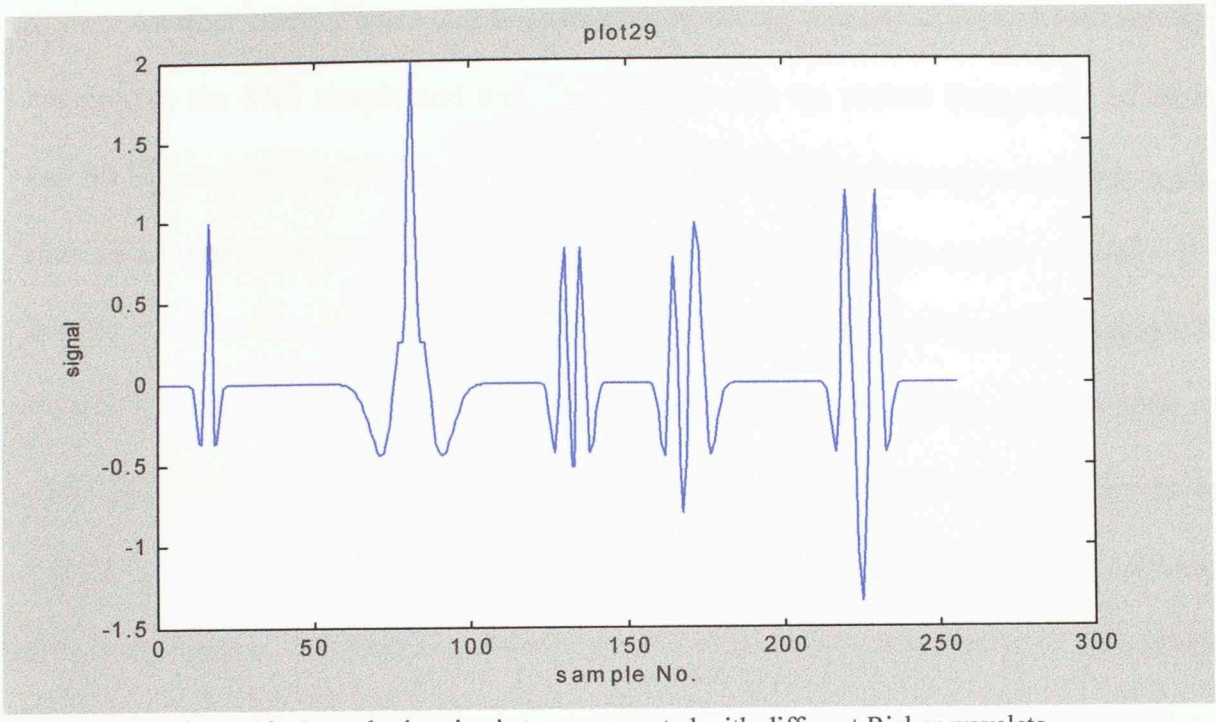

Figure 29. A synthetic seismic trace generated with different Ricker wavelets.

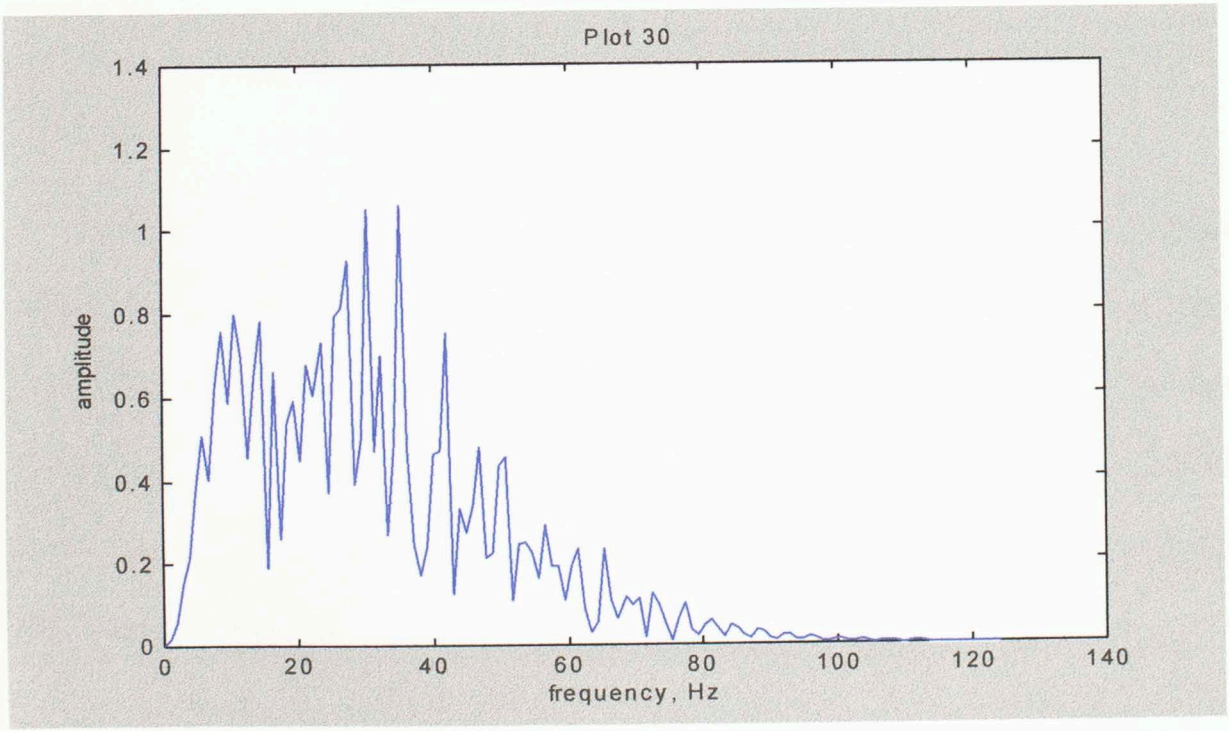

Figure 30. The Fourier transform of Plot 29.

Another trace (Figure 21) is generated by adding a linear chirp and two wavelets centered at the 84*th* sample and the 158*th* sample with the central frequencies of 40 Hz and 80 Hz respectively. The trace consists of both continuous frequency-changing signals such as a linear chirp as well as some highly localized signals such as two wavelets. It is used to compare the mutual time-frequency localization property of the CWT (Figure 27) and the MPD (Figure 28). From the comparison, both methods provide approximately the same time-frequency distribution information of the signal. However, as shown on the second wavelet centered at the 158*th* sample, the frequency resolution of MPD decreases greatly because of the wide frequency support for the Ricker wavelets with high central frequency. Therefore, the CWT provides a better time-frequency representation for a signal with the energy mainly with a continuous frequency distribution pattern.

The Comparison among different STFTs, CWT and MPD

The last trace (Figure 29) is a synthetic seismic trace generated by the convolution of Ricker wavelets with a reflectivity series. The reflectivity series has (A) a positive spike at n=16(n is the time sample), (B) a positive spike at n=81, (C) a set of 2 (positive-positive)spikes at n=130 and n=135, (D) a set of 2(positive-positive)spikes at n=165 and n=172 and (E) a set of 3(positive-negative-positive)spikes at n=220, n=225 and n=230. The Fourier transform of the signal is given in Figure 30.

Reflection A is created using a 40 Hz Ricker wavelet. Reflection B is created using a 40 Hz Ricker wavelet and a 10 Hz Ricker wavelet. Event C is generated with a 30 Hz

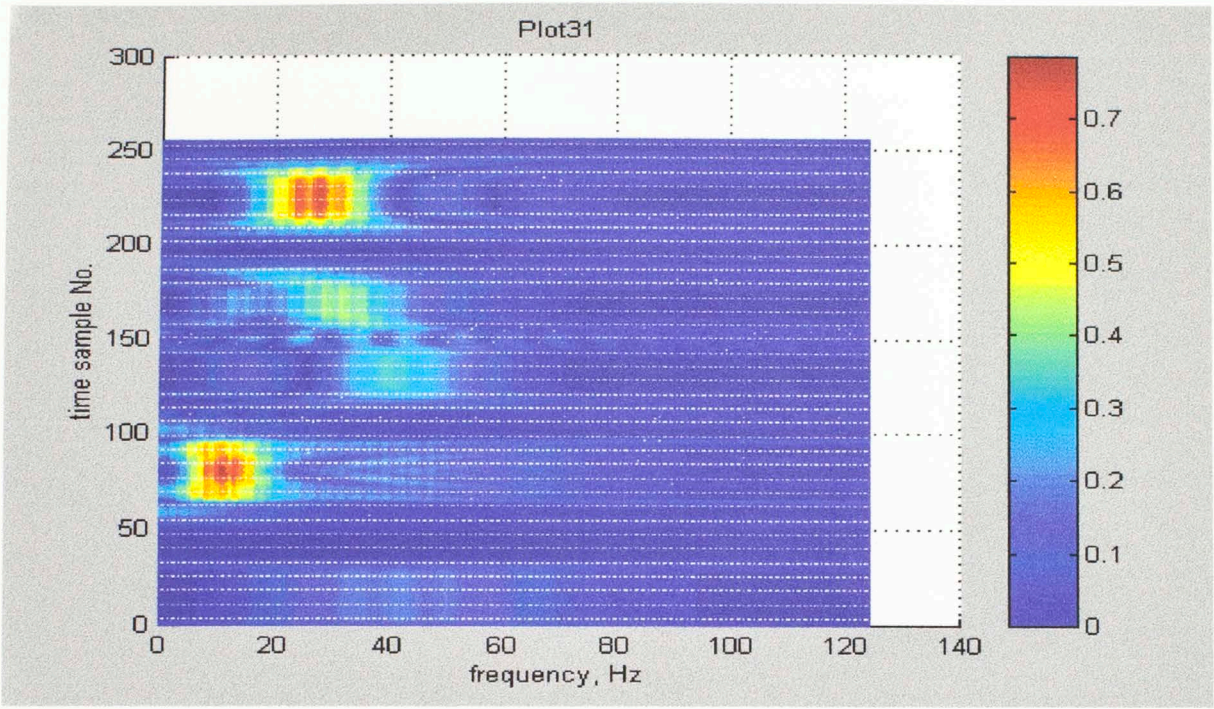

Figure 31. The STFT of Plot 29 with a boxcar window, with the window length of 64 samples.

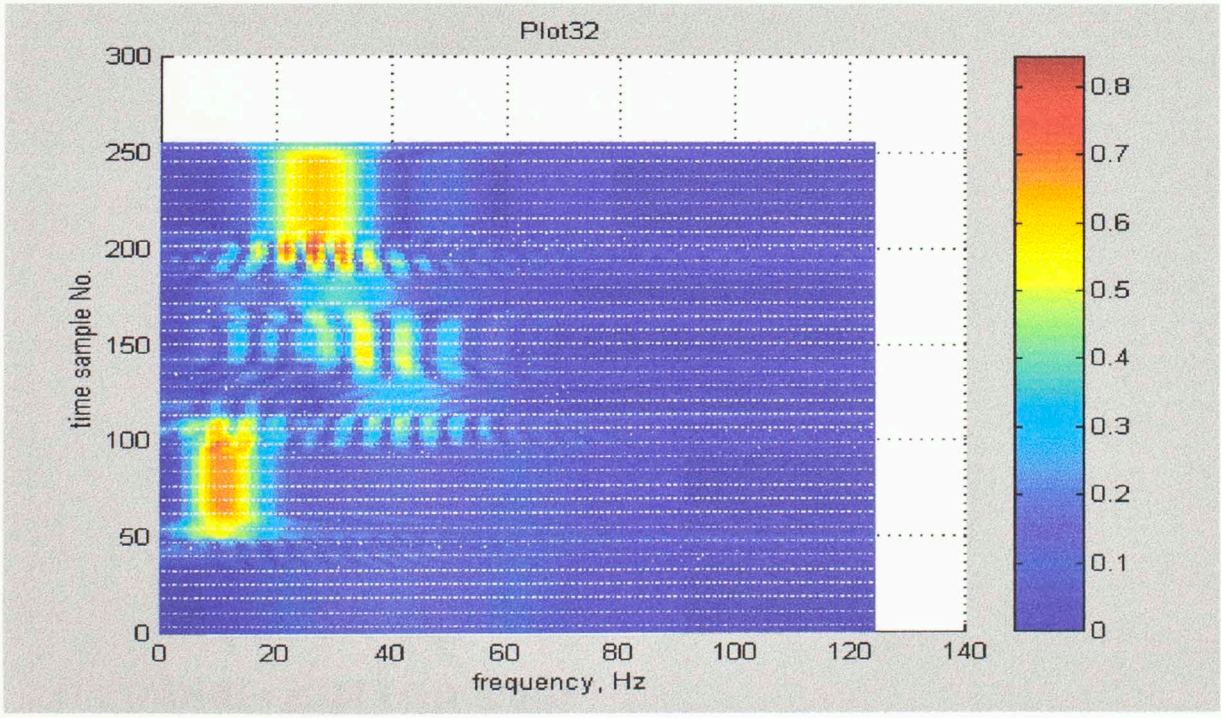

Figure 32. The STFT of Plot 29 with a Gaussian window, with the window length of 32 samples.

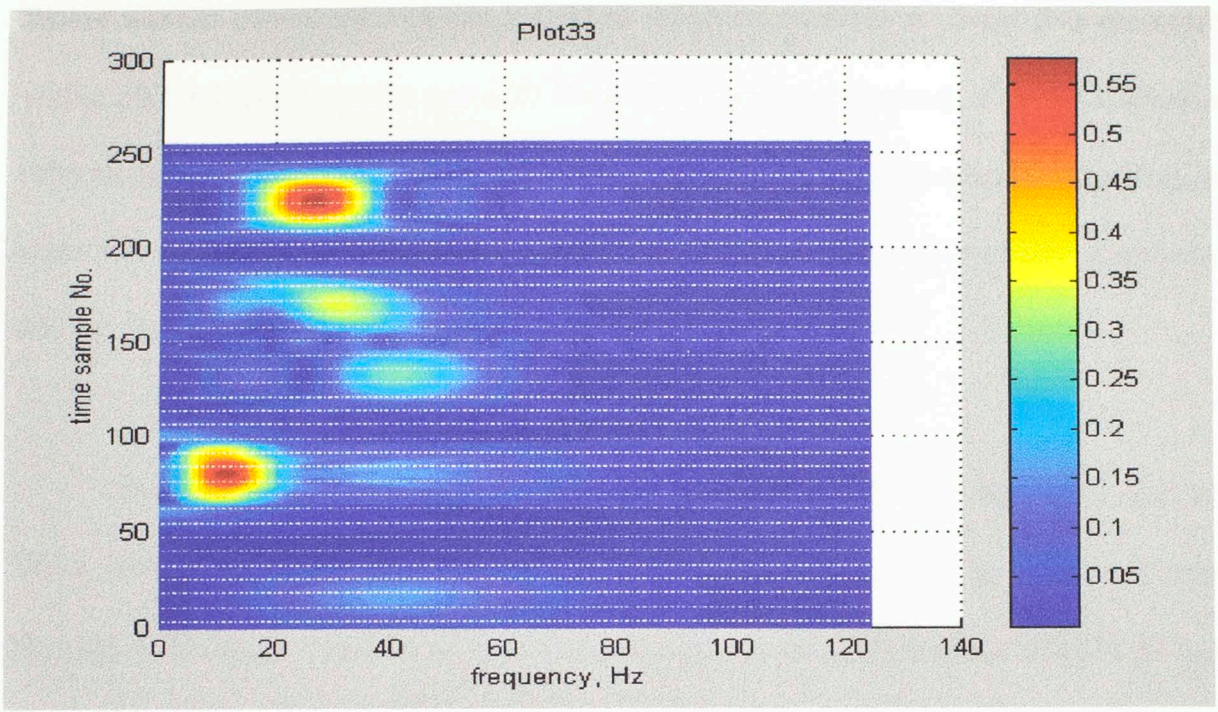

Figure 33. The STFT of Plot 29 with a Gaussian window, with the window length of 32 samples.

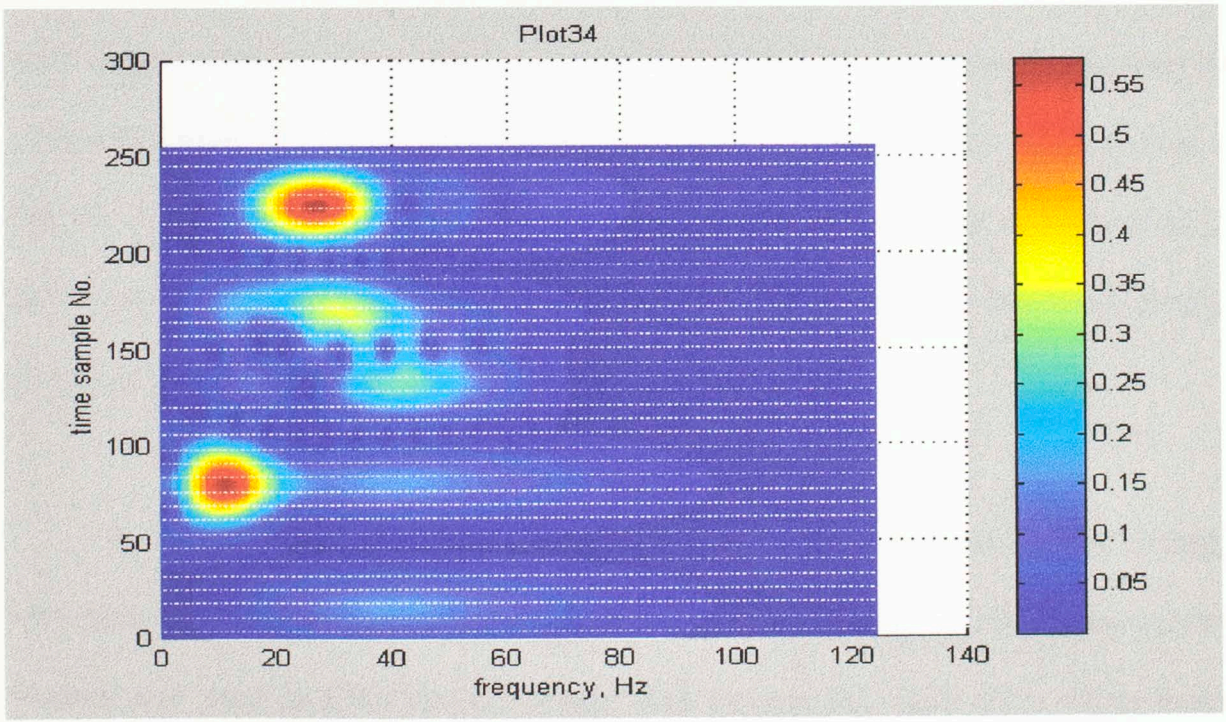

Figure 34. The STFT of Plot 29 with a Gaussian window, with the window length of 64 samples.

Ricker wavelet convolved with two individual reflection coefficients. Event D is generated with a 30 Hz Ricker wavelet and a 20 Hz Ricker wavelet convolved with two individual reflections respectively. Event E is generated with a 30 Hz Ricker wavelet. The Fourier transform of the signal (Figure 30) shows the energy of the signal is mainly distributed among the middle-frequency range.

Here, the trace is processed with three methods, the STFT, the CWT and the MPD. In the STFT, two types of windows are used, boxcar (Figure 31 and Figure 32 with the size of 32 samples and 64 samples respectively) and Gaussian window (Figure 33 and Figure 34 with the size of 32 samples and 64 samples respectively). With the use of a big-size boxcar (Figure 32), the five events are smeared in time. With the use of a small-size boxcar (Figure 31), the five events are separated in time, however, the time localization is poor as the events are represented by squares. The time localization is greatly improved by using a Gaussian window in the STFT. With the use of a long Gaussian window (Figure 34), the five events are not completely separated with some interference between event C and event D. However, the property of time localization has been improved greatly compared to the STFT with the use of a boxcar window.

By CWT (Figure 35), the five events are well separated and well located in time, with slight interference between them, which may be because for the middle-frequency, the window size used in CWT is even longer than 64 samples, since there exists more interference than Figure 34. Figure 36 is another view of the Figure 35. It shows that the events are not well separated because the envelope of the window is large.

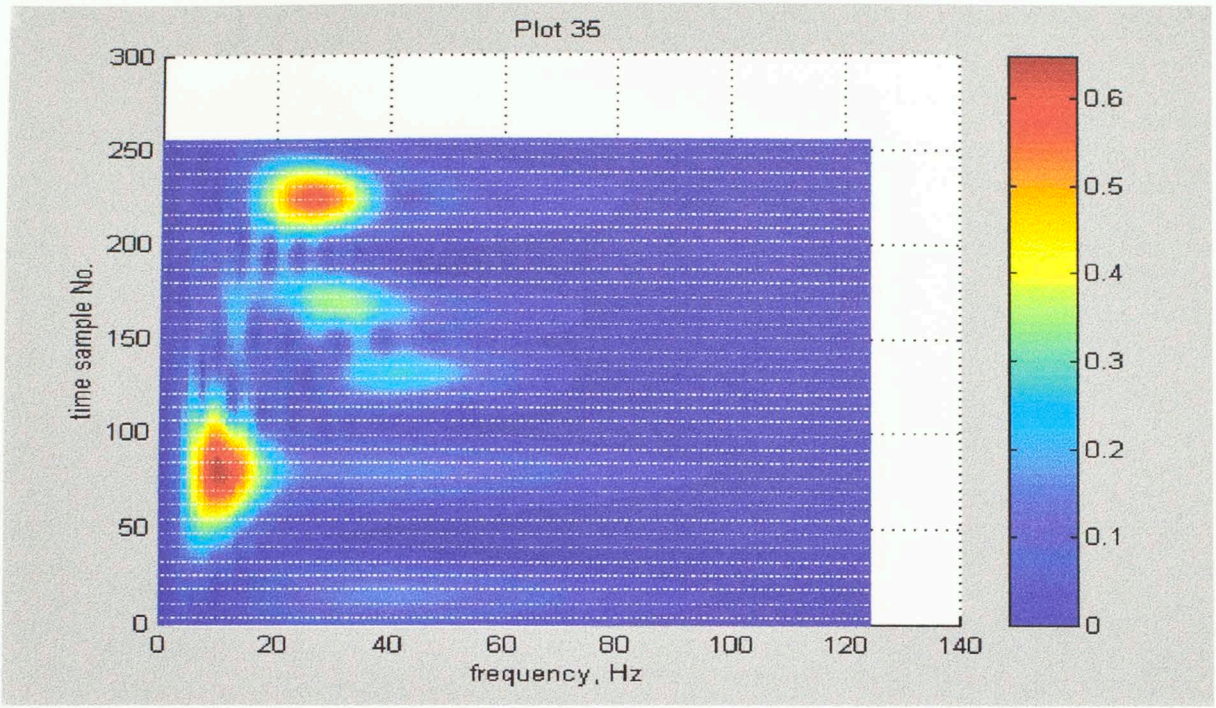

Figure 35. The CWT of Plot 29.

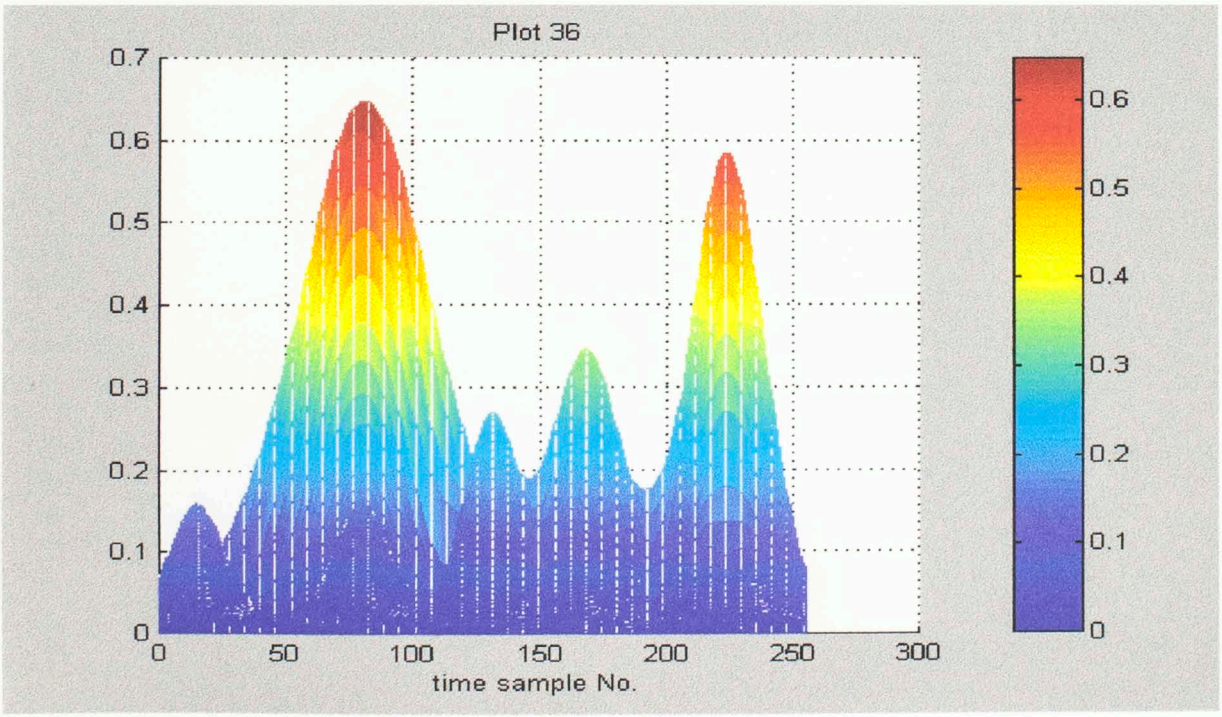

Figure 36. Another view of Plot 35.

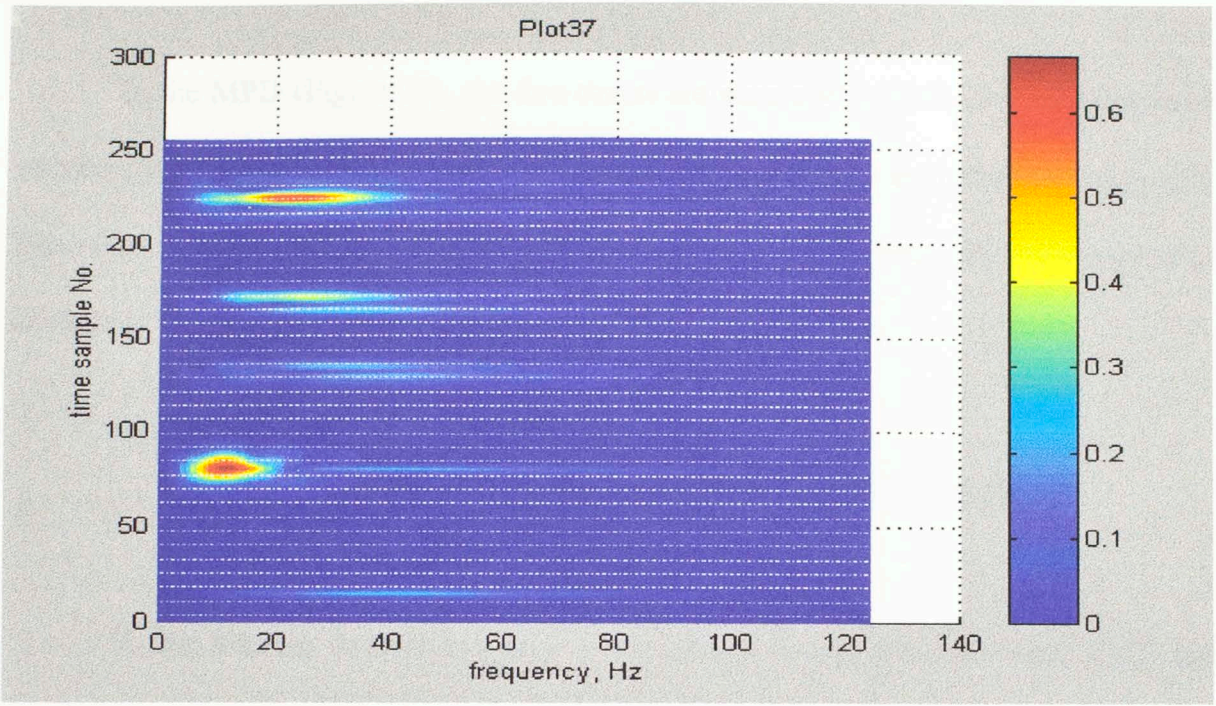

Figure 37. The MPD of Plot 29.

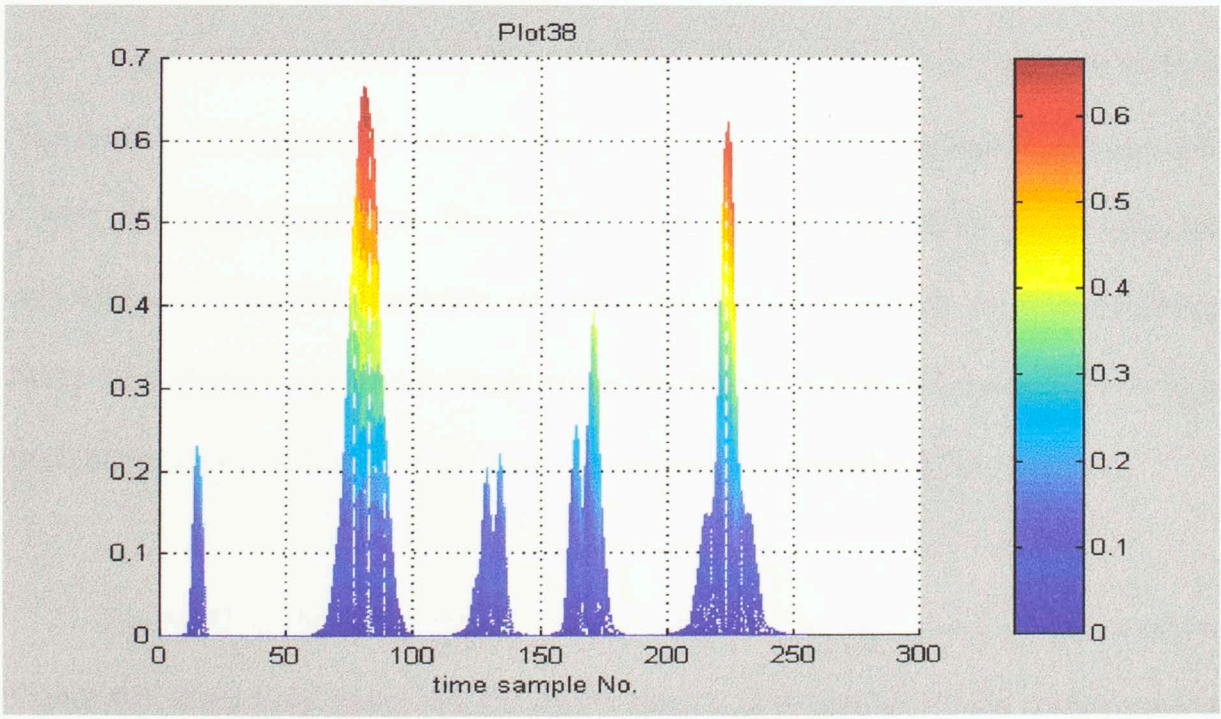

Figure 38. Another view of Plot 37.

In the MPD (Figure 37), the five events are separated in both time and frequency excellently. The information of the time-frequency distribution is represented correctly. Figure 38 is another view of the Figure 37, which shows a pronounced time-localization advantage of this method by comparing Figure 38 with Figure 29.

The Application of the MPD in Real Seismic Data

In this section, three traces from a real seismic survey are processed. CWT and MPD are used on the data and the results are compared. From the seismogram, it can be observed that there are three strong ground rolls that come across all the traces.

Figure 39 is the 1*st* trace from 48 traces of the survey. The Fourier transform (Figure 40) of the trace shows that the data is composed of high-amplitude components in a wide range of frequencies. The CWT (Figure 41) and the MPD (Figure 42) are compared. Both methods provide relatively the same information of time-frequency distribution of the trace. Three strong noises are shown on both representations, with a much better time localization of the frequency distribution provided by the MPD.

Figure 43 is the 24^{th} trace from 48 traces of the survey. The Fourier transform (Figure 44) of the trace shows that the data is composed of high-amplitude components in a wide range of frequencies. The CWT (Figure 45) and the MPD (Figure 46) are compared. Three strong noises are spread out compared to trace 1 on both

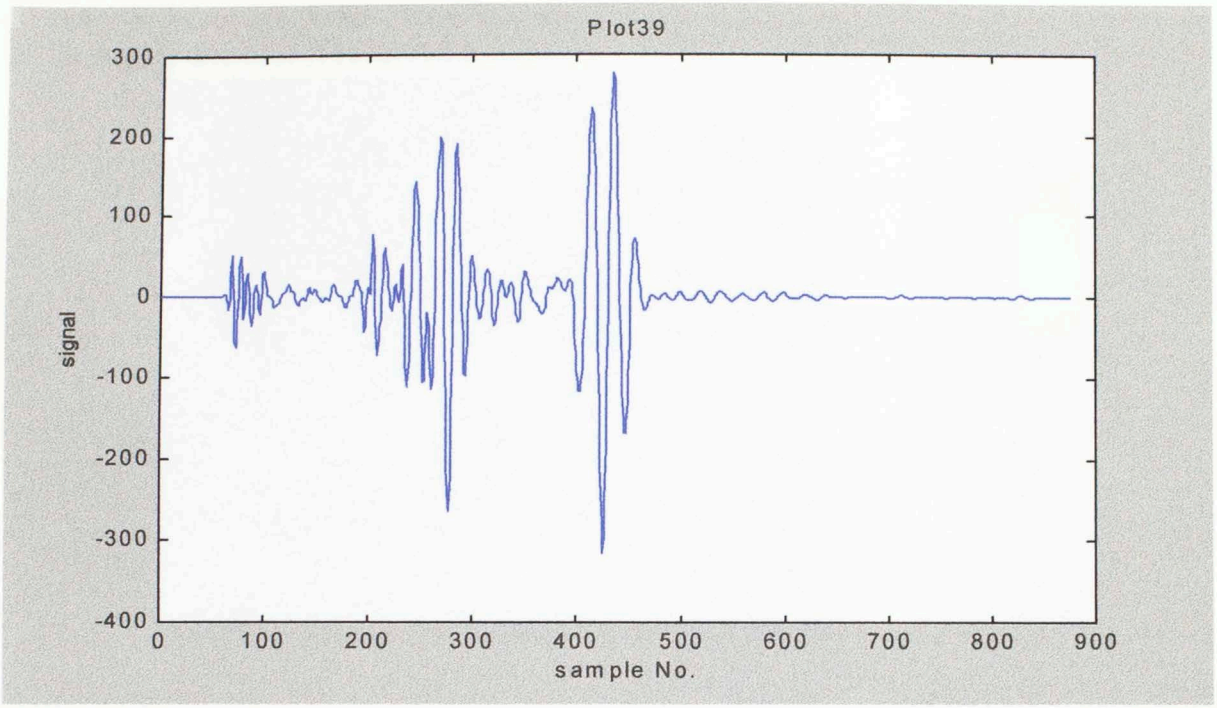

Figure 39. The 1st trace from a real seismic survey.

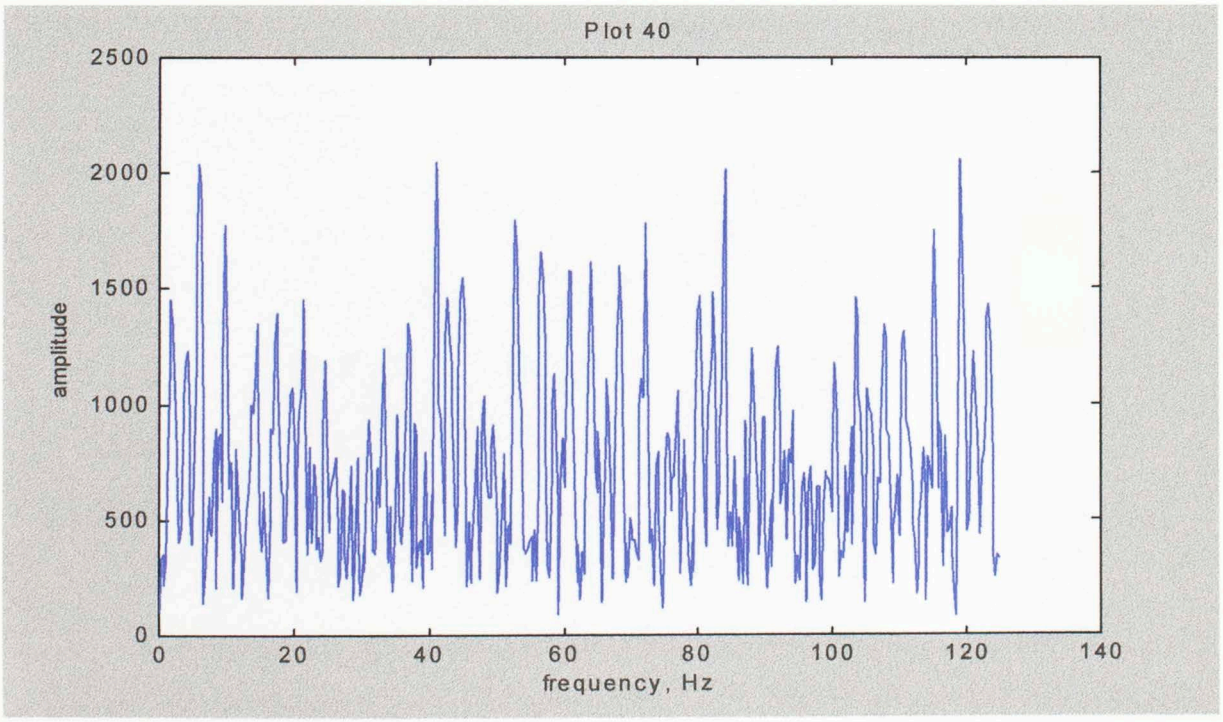

Figure 40. The Fourier transform of Plot 39.

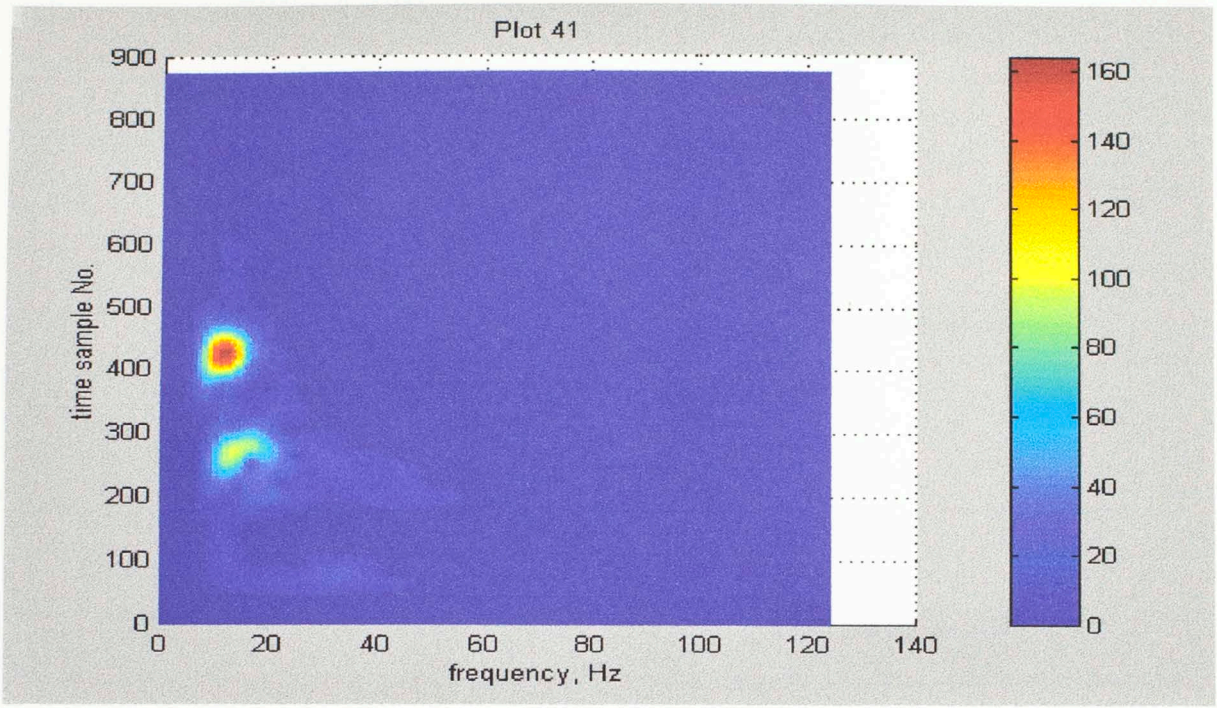

Figure 41. The CWT of Plot 39.

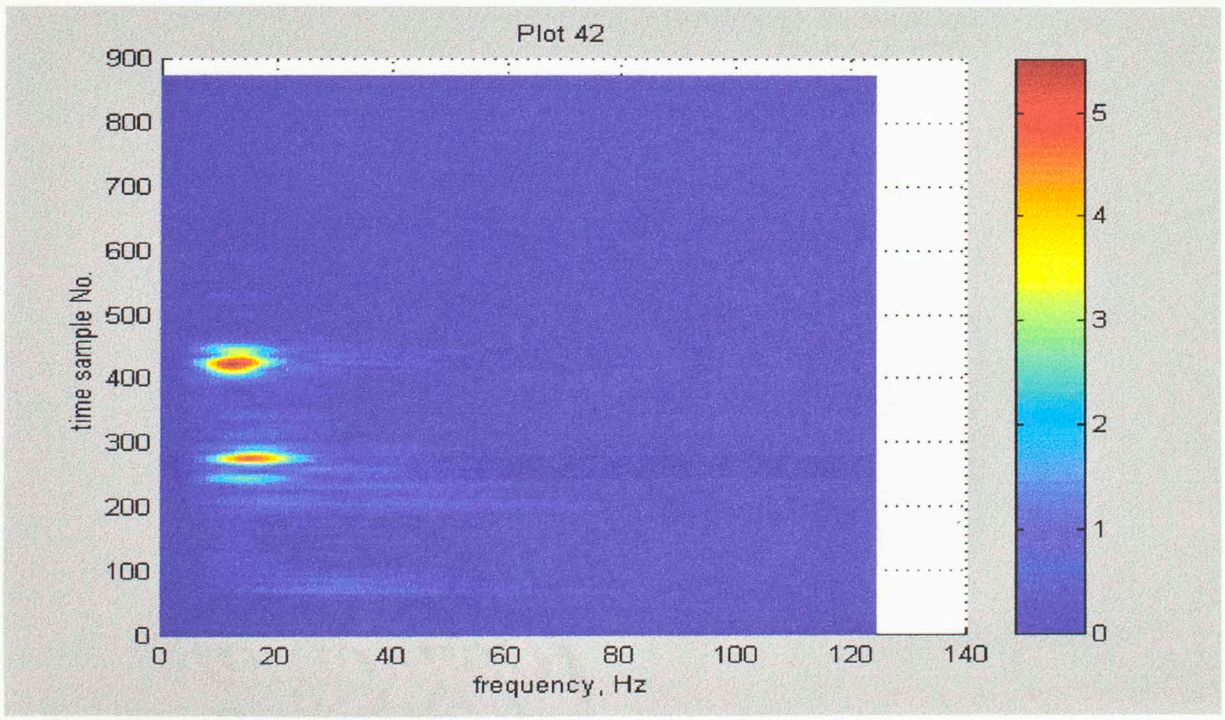

Figure 42. The MPD of Plot 39.

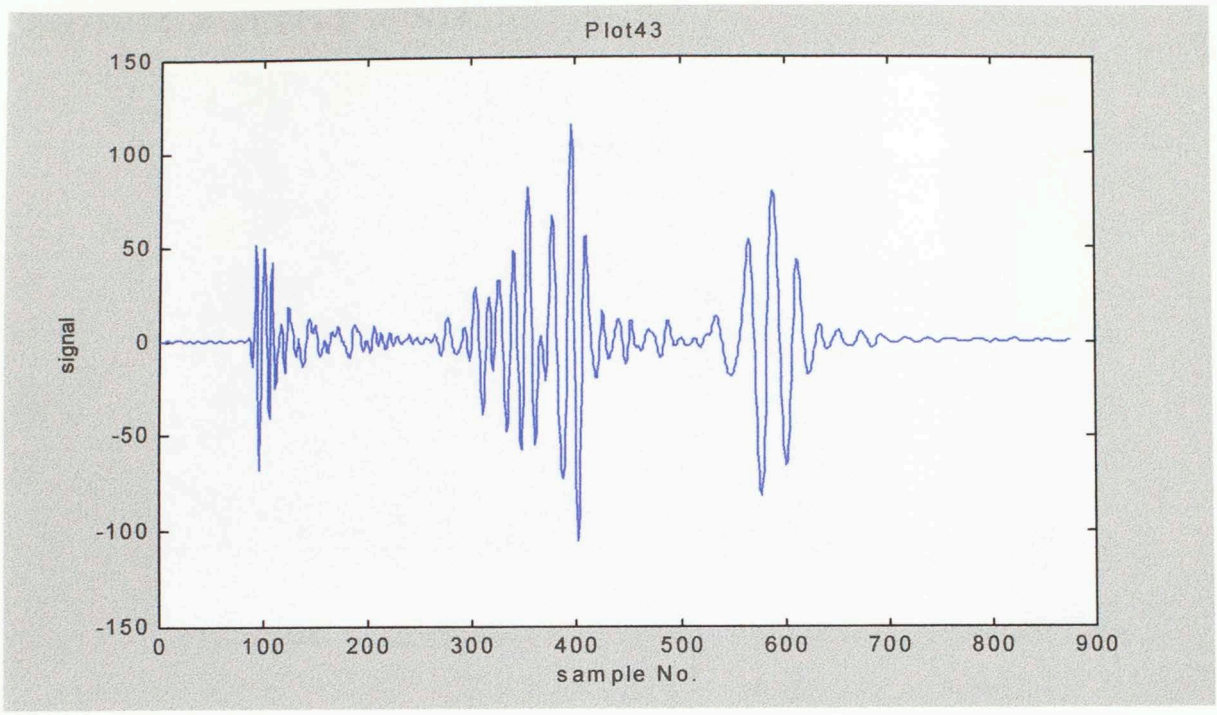

Figure 43. The 24th trace from the seismic survey.

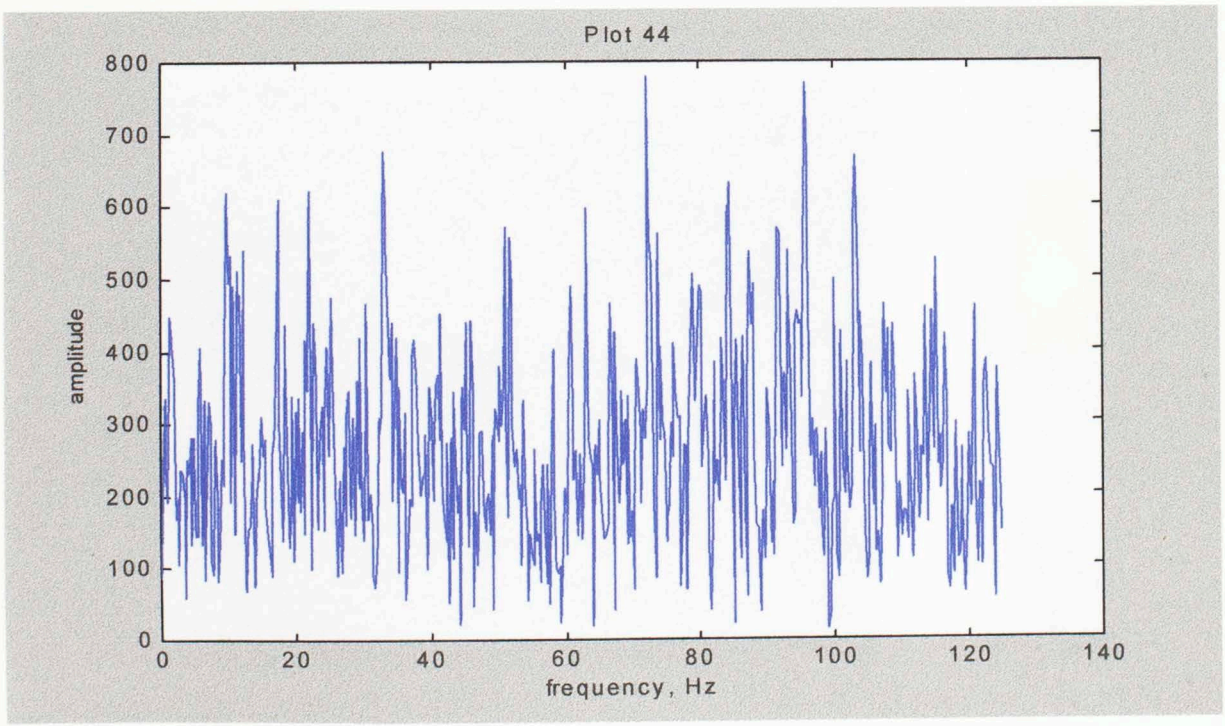

Figure 44. The Fourier transform of Plot 43.

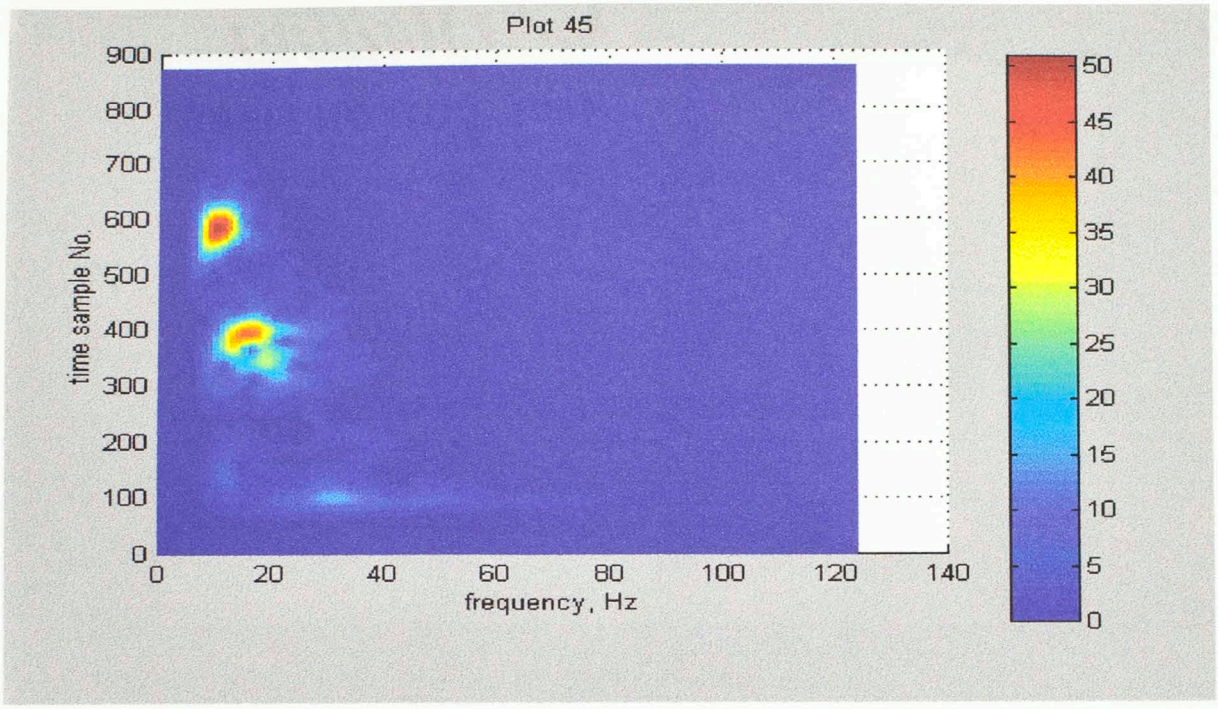

Figure 45. The CWT of Plot 43.

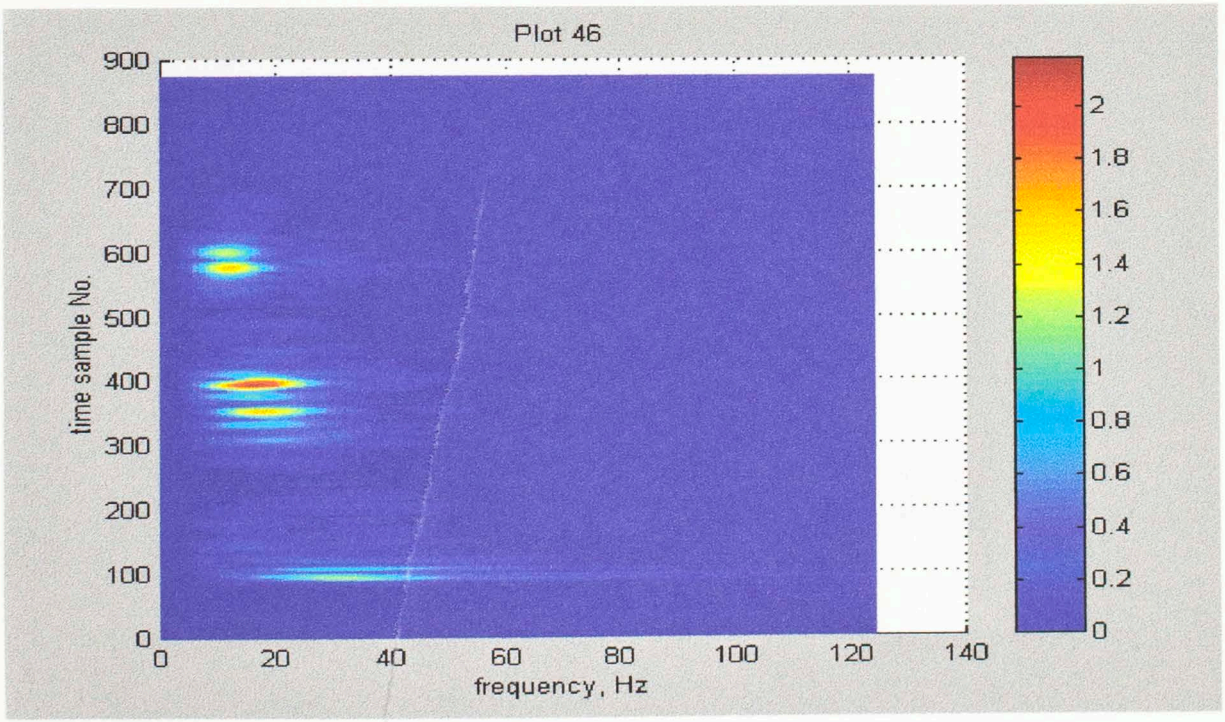

Figure 46. The MPD of Plot 43.

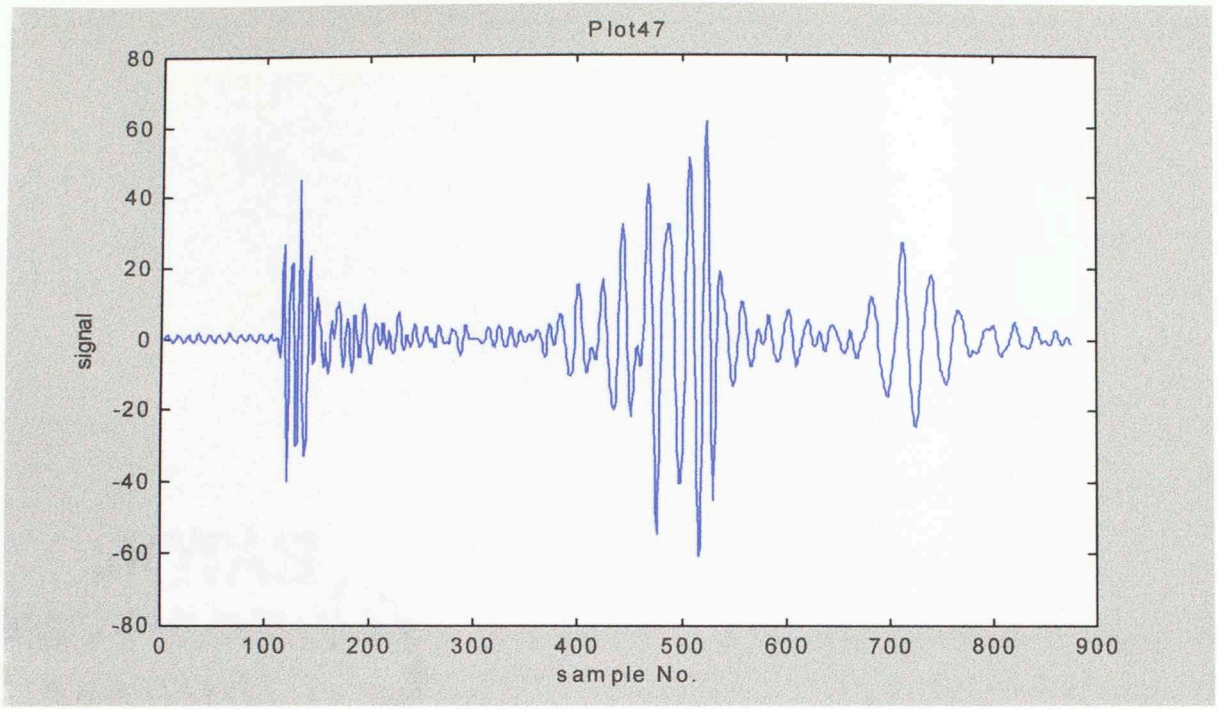

Figure 47. The 48th trace from the seismic survey.

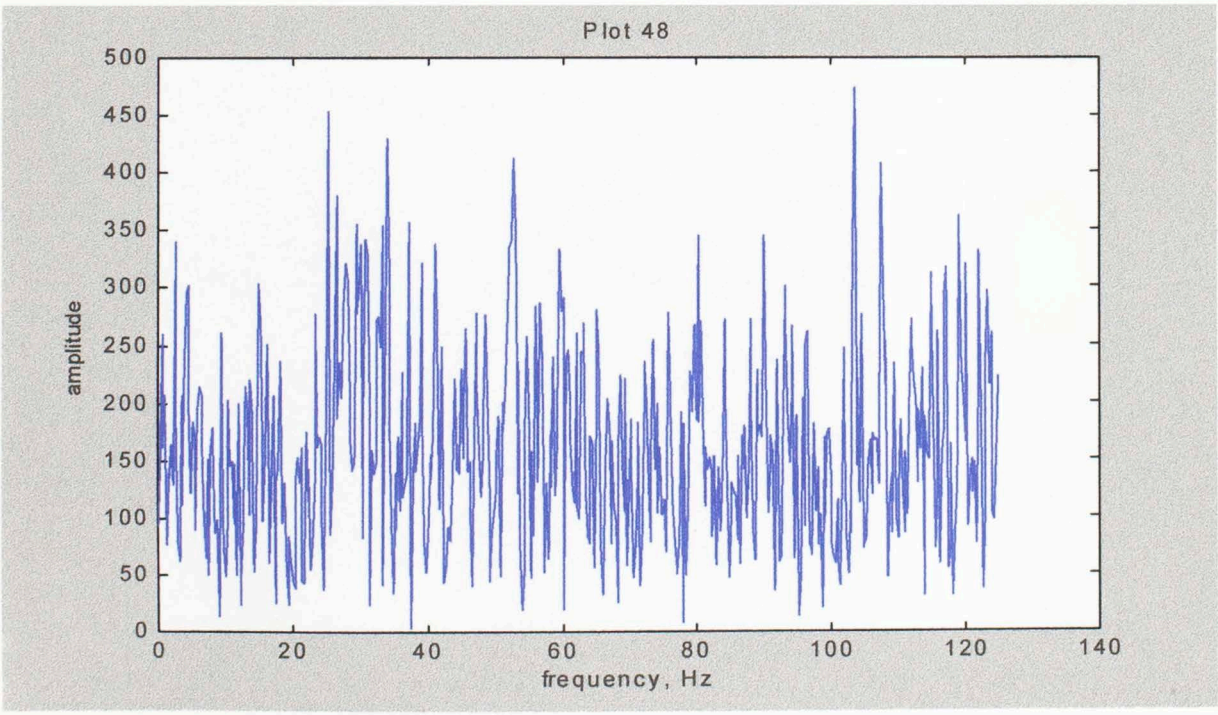

Figure 48. The Fourier transform of Plot 47.

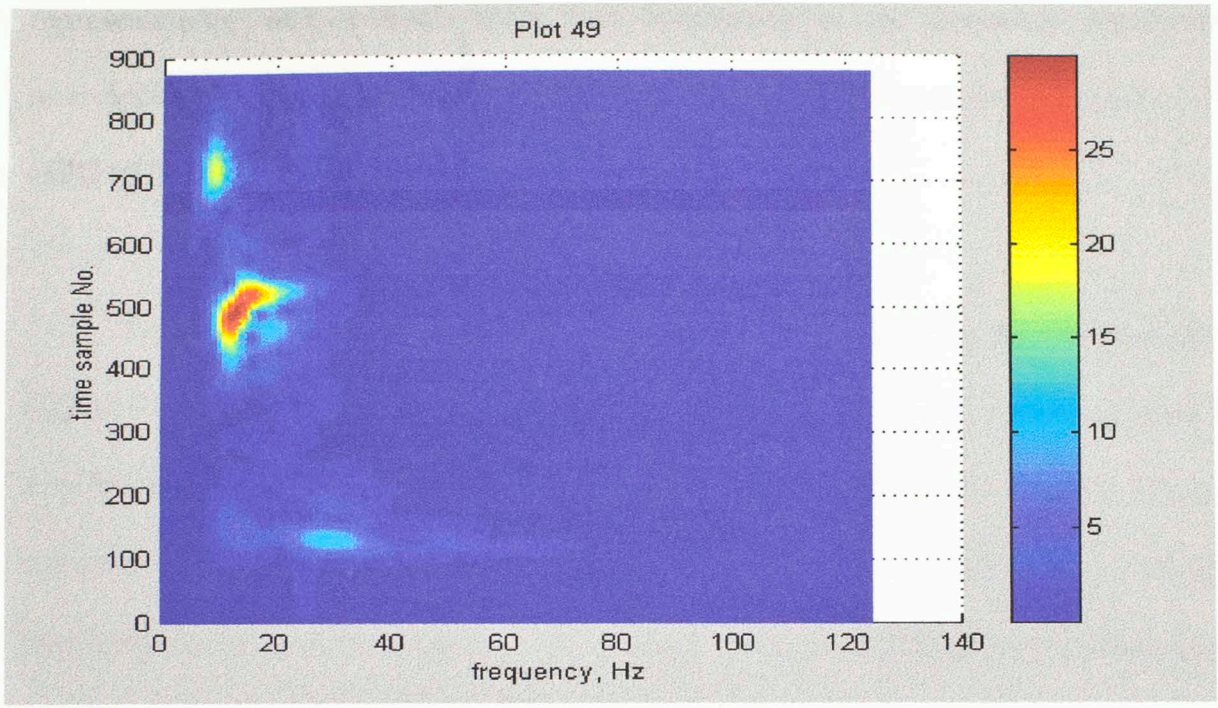

Figure 49. The CWT of Plot 47.

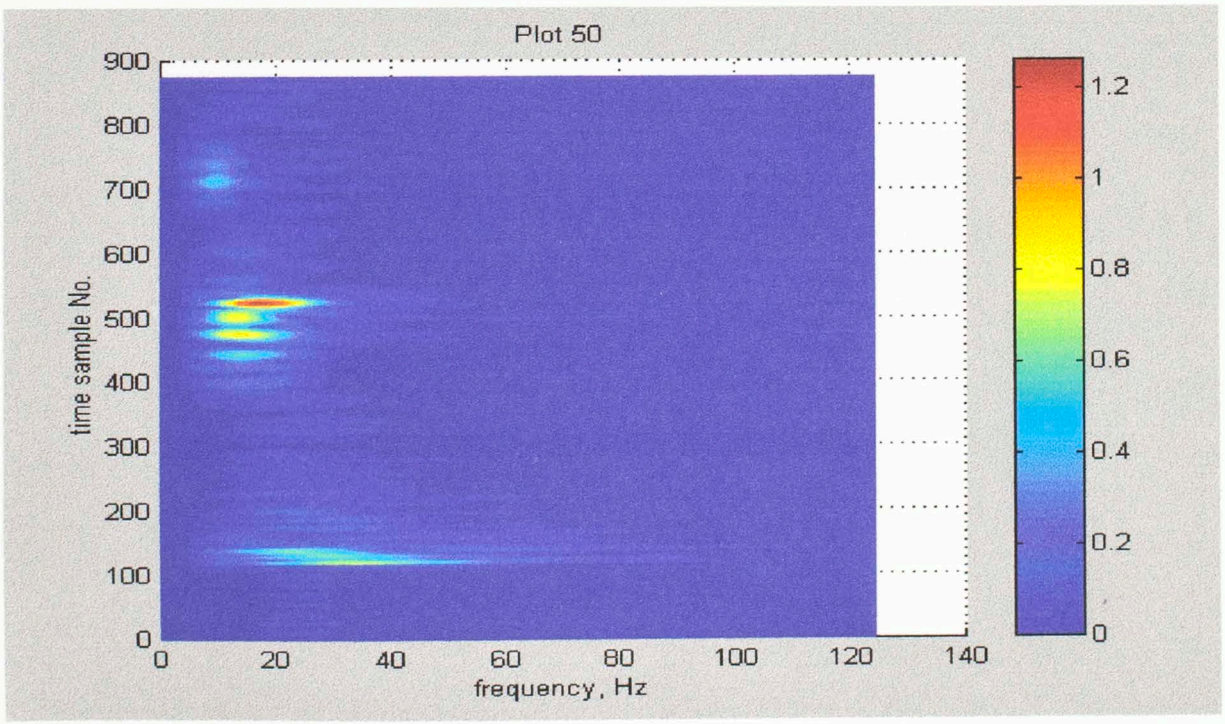

Figure 50. The MPD of Plot 47.

representations, with a much better time localization of the frequency distribution provided by the MPD. There is a linear moveout of the three groundrolls compared to the MPD of the 1st trace (Figure 42).

Figure 47 is the 48th trace from 48 traces of the survey. The Fourier transform (Figure 48) of the trace shows that the data is composed of high-amplitude components in a wide range of frequencies. The CWT (Figure 49) and the MPD (Figure 50) are compared. Three strong noises are further spread out compared to the 1*st* trace and the 24*th* trace on both representations, with a trend of decreasing frequency content with time. Compared with the MPDs of the 1st trace and the 24th trace, there is further moveouts for the three groundrolls.

Chapter 6 Multi-Channel Analysis Using The Wavelet Transform

Multi-channel processing operations are performed on several traces simultaneously. They are generally used to separate events that have different dips or moveouts. The 2-D Fourier transform is a basic implementation of multi-channel processes. In the 2-D FFT, all the points within the T-X area contribute to each single point in the F-K area. Therefore, it is an average F-K representation of all the information in the T-X plane. It is a good representation of some coherent signal with certain moveouts. However, it is not able to provide a good F-K representation of highly non-stationary data that has different *uncorrelated* signal patterns.

Because different moveout is a distinguished feature of signals and because traces are generally composed of different non-stationary frequency signal patterns, an effort is made to try to separate different events with different moveouts in a non-stationary trace by performing a wavenumber (K) transform after wavelet transform. Because MPD has the advantage of excellent time localization and middle-frequency localization property, it is used to perform the wavelet transform. Here I show initial attempts to analyze some characteristics of this multi-channel processing (K-MPD transform) procedure.

The procedure of the K-MPD transform is described as follows. After a set of traces is obtained, the MPD is performed on each trace. The result of the MPD is a time-frequency two-dimensional distribution. For each trace, an instantaneous frequency spectrum is obtained for each time sample. Then a traditional Fourier transform is

performed over the frequency spectrums of all the traces at each time sample. The final result is a three-dimensional time 3 frequency 3 wavenumber distribution. In contrary to the traditional F-K transform where the result is a F-K distribution, there is a F-K distribution for each time sample after the K-MPD transform. Several examples of the K-MPD transform are given below.

64 traces (Figure 51) are generated by the convolution of a 40 Hz Ricker wavelet with a linear-moveout reflection series. From the traditional F-K transform (Figure 52), the velocity of the event is approximately 4.95 (km/s). It is very close to the velocity of 5.0 (km/s) used in generating the traces. After K-transform is performed after the MPD, the total amount of data is a 256 3 128 3 32 three-dimensional array, with 256 time samples, 128 frequency samples and 32 wavenumber samples. At each trace, one wavelet is extracted by the MPD. Two F-K plots are shown here from 256 F-K plots generated. Figure 53 is the F-K Figure at the 24*th* time sample, and Figure 54 is the F-K Figure at the 64*th* time sample. They are two slanted ovals with nearly the same F-K distribution because the K-transform is performed on the same wavelets extracted by the MPD. The slope of the ovals is a little different from that of the traditional F-K Figure, because the MPD instead of the Fourier transform is used before the K-transform. After the MPD, the extracted wavelets have limited time duration, thus only those K-transforms performed near the extracted wavelets could collect the information of the wavelets. Therefore, the ovals are broader than the line in the traditional F-K transform because less information is taken into the K-transform .

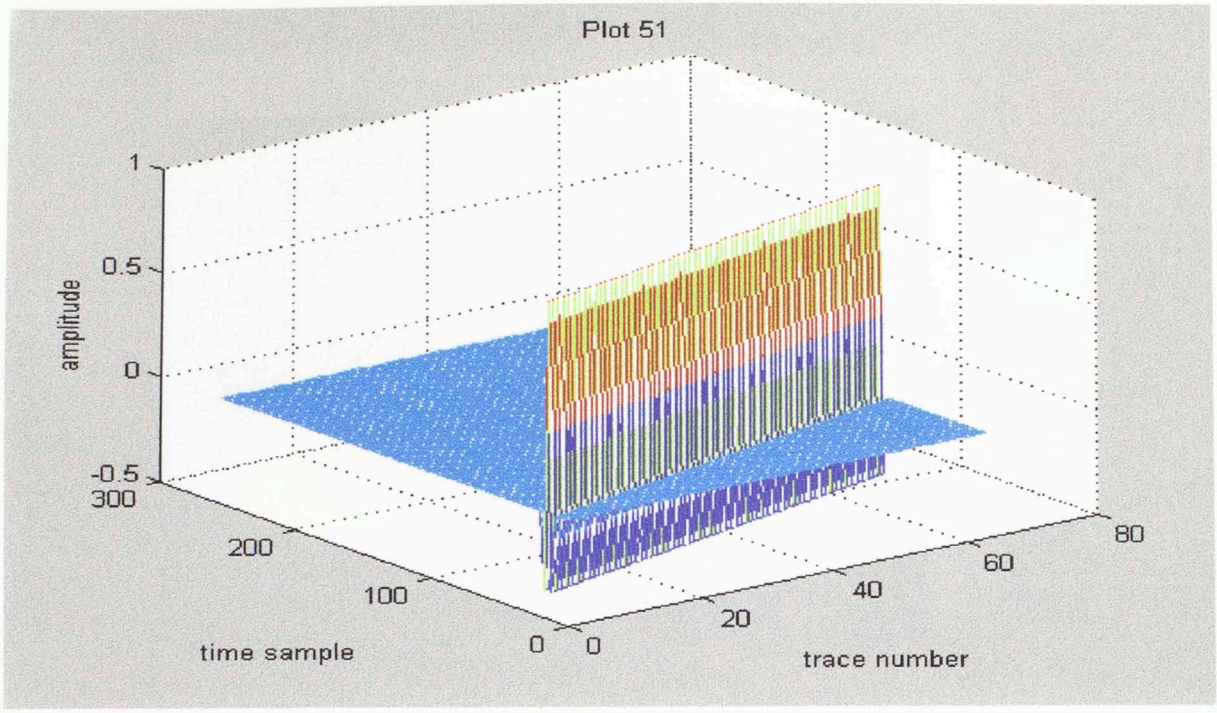

Figure 51. 64 traces with linear moveout. The source is a 40 Hz Ricker wavelet.

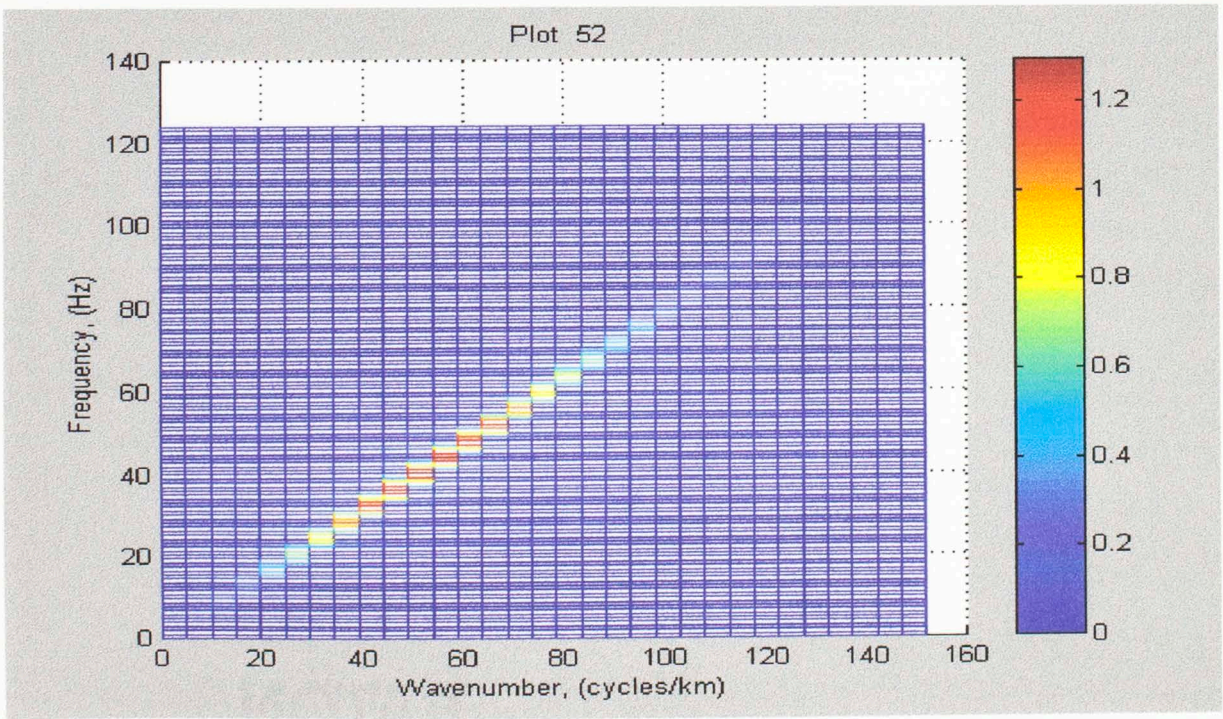

Figure 52. The traditional 2-D Fourier transform of Plot 51.

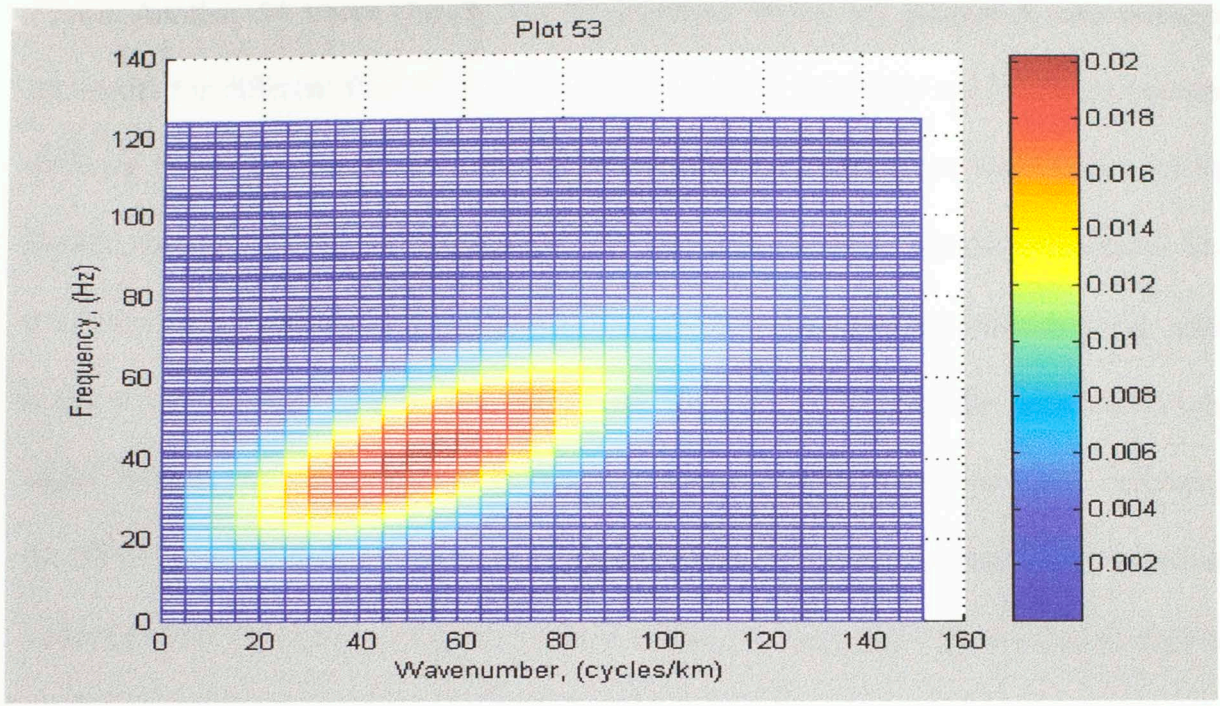

Figure 53. A slice from the K-MPD transform of the traces in Plot 51. It is the F-K plot at the 24th time sample.

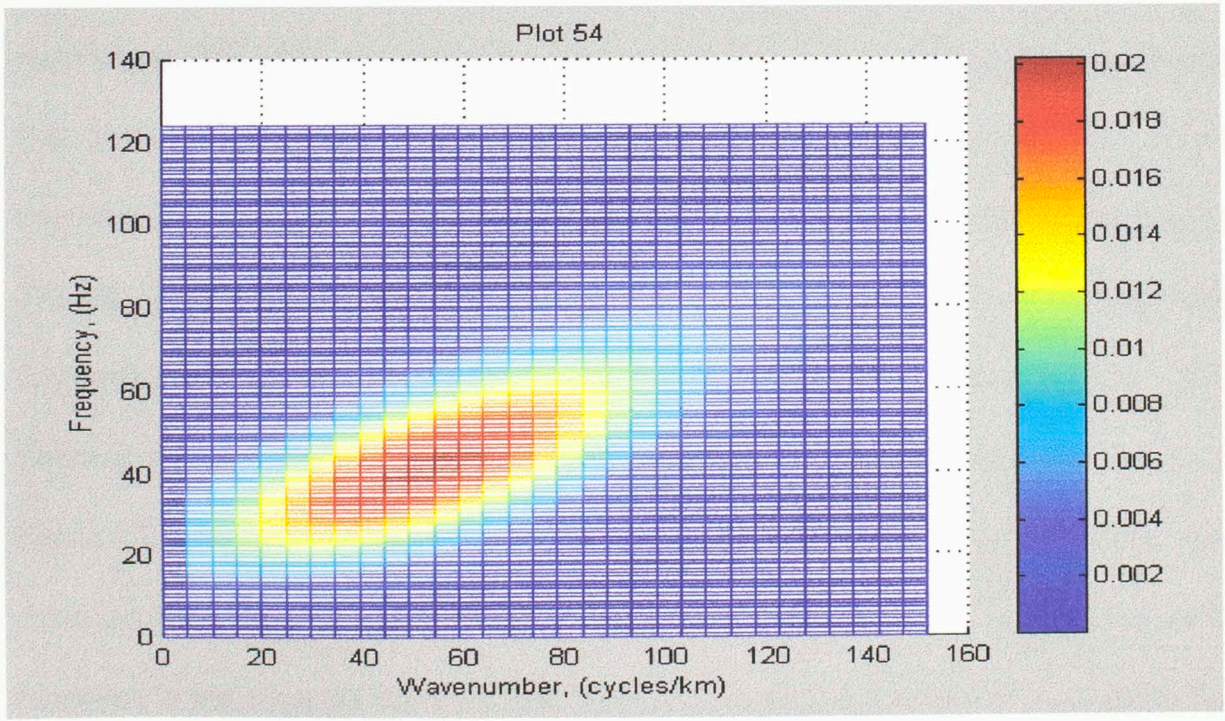

Figure 54. A slice from the K-MPD transform of the traces in Plot 51. It is the F-K plot at the 64th time sample.

Another 64 traces (Figure 55) of dispersive waves are generated with different moveouts for different frequency components. Different moveouts are produced because different frequency components have different velocities. The source wavelet is a 25 Hz Ricker wavelet. It propagates horizontally and spreads out as it travels horizontally. The traditional F-K transform (Figure 56) shows some resemblance to the velocity curve used in the generation of the traces. The MPD extracts a series of wavelets for each trace as the signal spreads out. Each wavelet extracted is uncorrelated to other wavelets. Following the MPD, because each wavelet has limited time duration, fewer points will be collected from the same wavelet to contribute to the K transform compared with the traditional F-K transform where every point contributes to the result. Hence, the ovals on the F-K plot generated from the K-MPD transform are much broader than the sharp line on the traditional F-K plot. Because wavelets with high central frequency have longer frequency support, the F-K plots generated from high central-frequency wavelets are much sharper than those with low central frequency in the K-MPD transform. Two F-K plots at two time samples are included in my thesis. They are the F-K plots at the 18*th* sample (Figure 57) and 138*th* sample (Figure 58) respectively. Those slanted ovals with coherent F-K distribution are thought of more or less the F-k distribution for the same wavelet. Their slopes bear some resemblance to the slope in the traditional F-K transform (Figure 56). In Figure 57, there exists a slanted coherent oval in the low-frequency range. There also exist some incoherent shapes, which are thought as because the information collected for the K transform is not from the same wavelet. They are generally not useful in discovering the moveout pattern of an event. In Figure 58, there exists a slanted coherent oval in the mid-frequency range. They are sharper than the ovals within the low- frequency range because

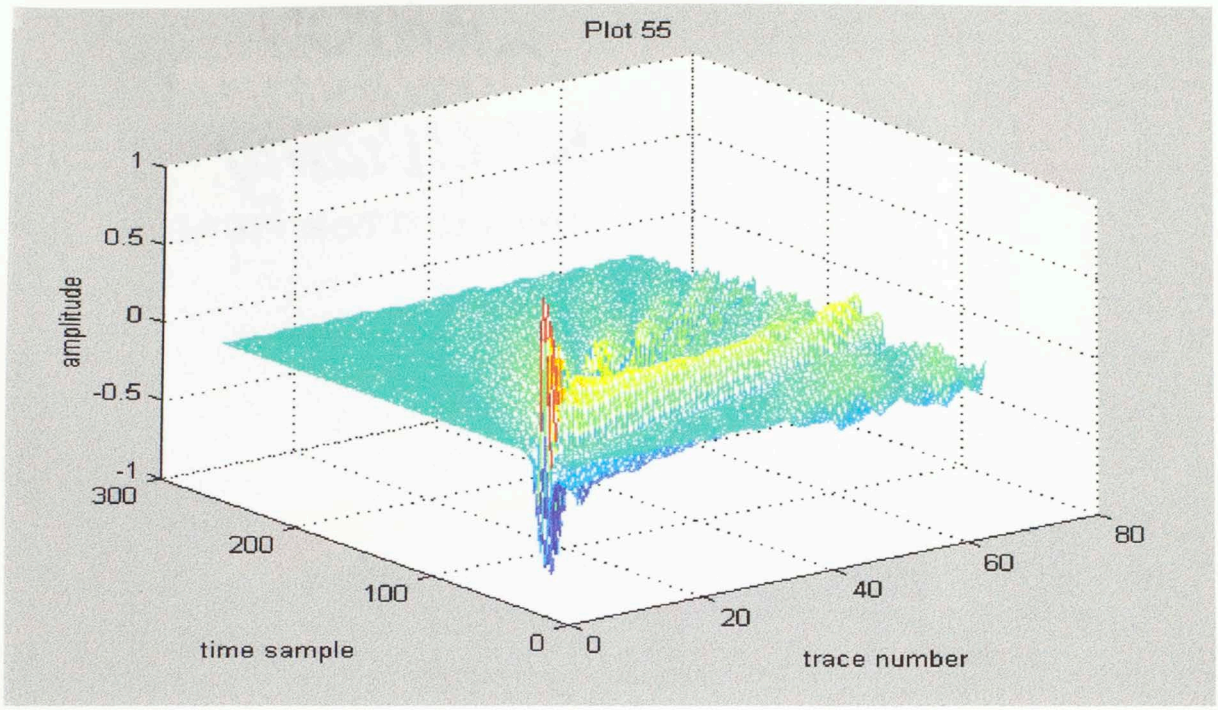

Figure 55. 64 traces of dispersive waves. The source is a 25 Hz Ricker wavelet.

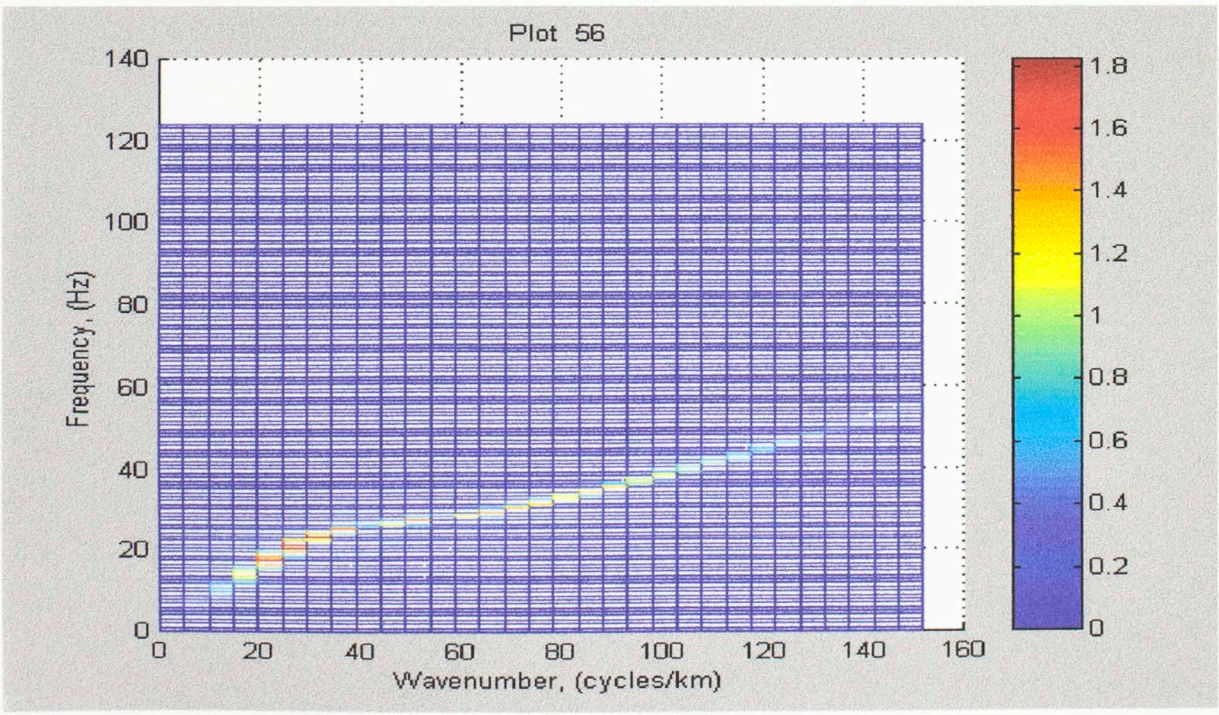

Figure 56. The F-K transform of the traces in Plot 55.

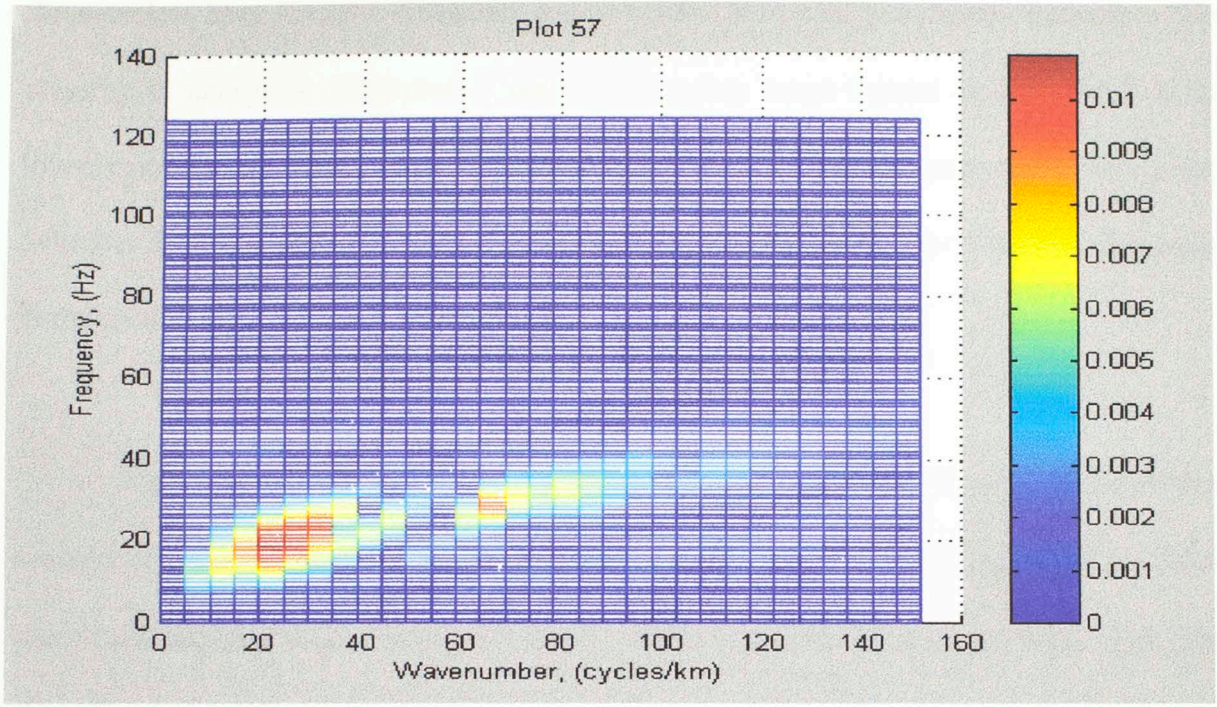

Figure 57. A slice from the K-MPD transform the the traces in Plot 55. It is the F-K plot at the 18th time sample.

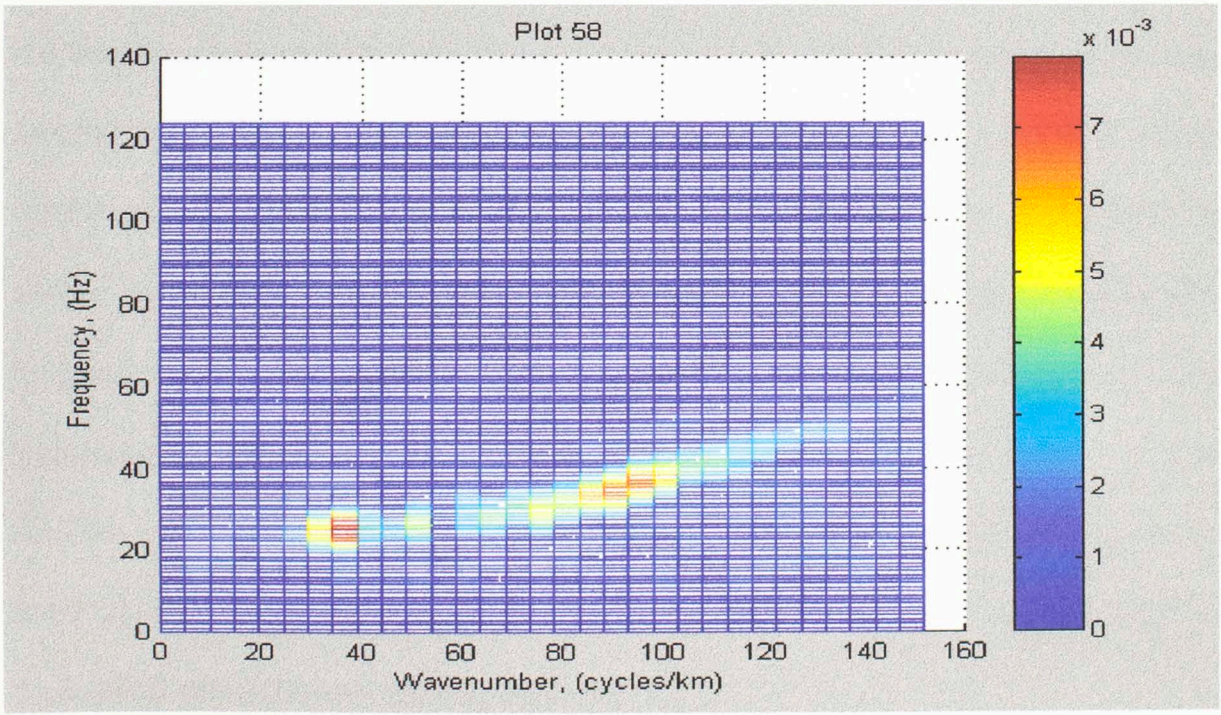

Figure 58. A slice from the K-MPD transform of the traces in Plot 55. It is the F-K plot at the 138th time sample.

the wavelets have longerer frequency support than those with low central frequency. The majority of energy is distributed in the mid-frequency range instead of distributed in the low-frequency range in Figure 57 because the higher frequency components have lower velocity. Thus the energy of high-frequency components comes later than low-frequency components.

Figure 59 is another 64 traces generated by the multiplication of a 40 Hz Ricker wavelet and a linear chirp. Among them, the Ricker wavelet has linear moveout and linear chirp has zero moveout. Because the instantaneous frequency content increases with time for a linear chirp, Fourier transform provides only an average frequency representation of the frequency change in the signal. Therefore, the traditional F-K transform (Figure 60) provides only average F-K information of the traces. It is unable to reflect the change of the frequency content with time. Two F-K plots from the K-MPD transform at two time samples are included, which are from the 18*th* time sample and 78*th* time sample respectively. They clearly show the trend of frequency increase with time. At the 18*th* time sample, two slanted ovals with similar F-K distribution are close and lie within the mid-frequency and mid-wavenumber range. Two slanted ovals separate more and more into high-frequency and low-frequency ranges with time. As shown on Figure 62 (taken from the 78*th* time sample), two ovals have separated far apart into the high-frequency and the low-frequency ranges respectively. The slopes of all the ovals in the F-K plots taken from the K-MPD transform are relatively the same.

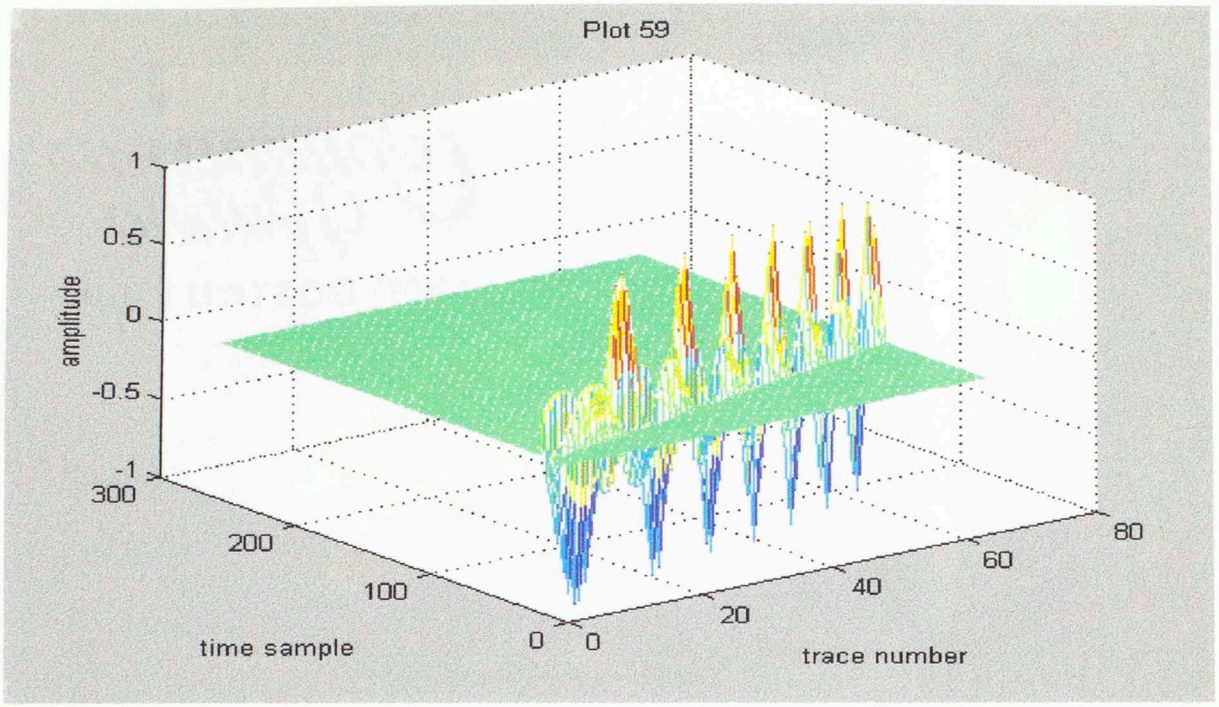

Figure 59. 64 non-stationary traces with linear moveout. The source is a 40 Hz Ricker wavelet modulated by a linear chirp.

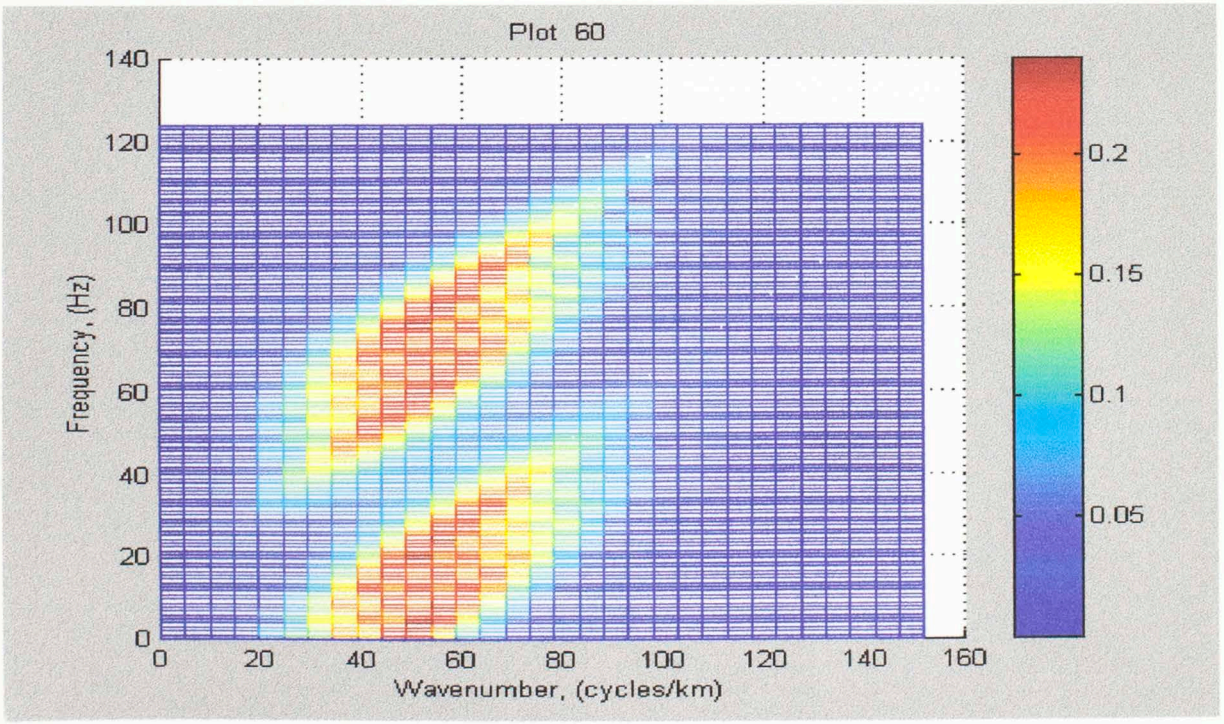

Figure 60. The traditional 2-D Fourier transform of the traces in Plot 59.

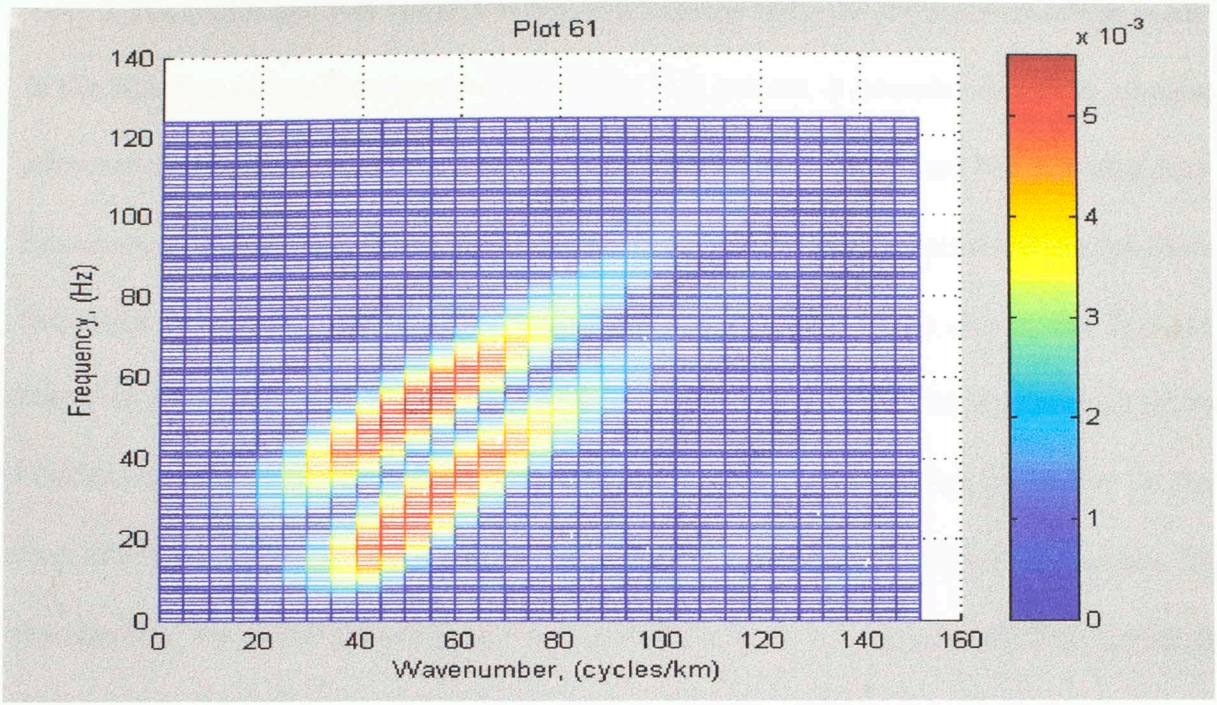

Figure 61. A slice from the K-MPD transform of the traces in Plot 59. It is the F-K plot at the 18th time sample.

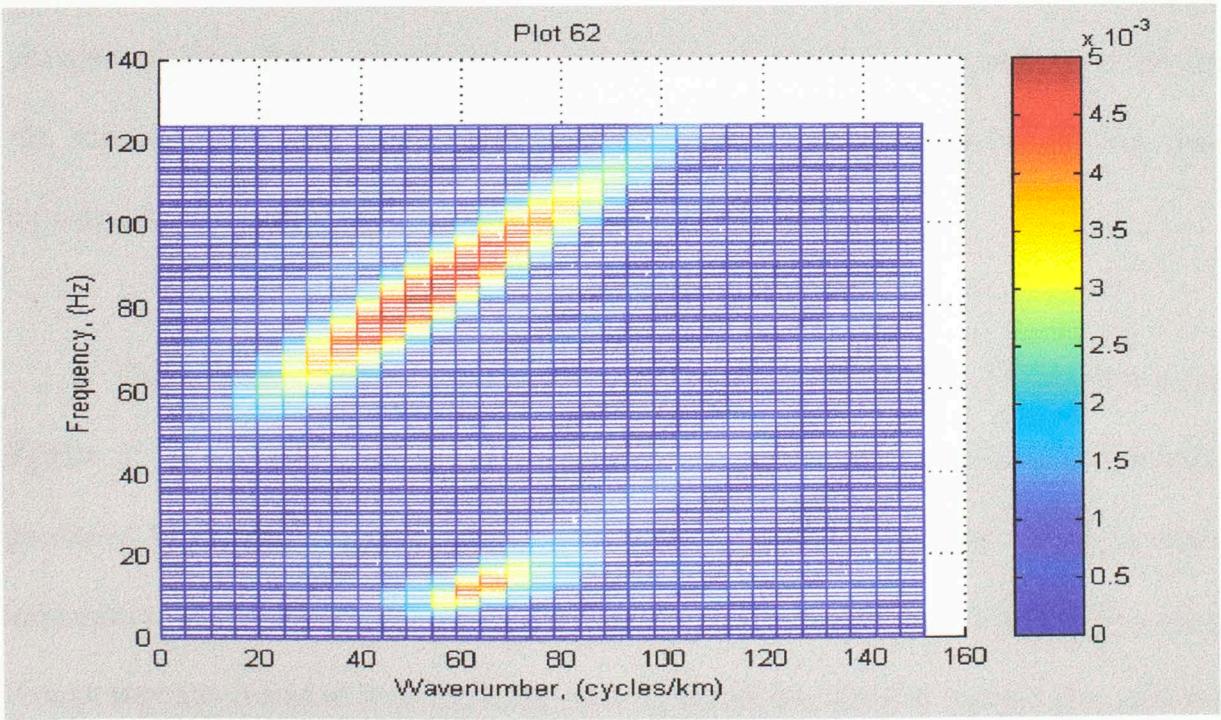

Figure 62. A slice from the K-MPD transform of the traces in Plot 59. It is the F-K plot at the 78th time sample.

The traditional F-K transform provides average moveout information of the events. In the situation where the signal has stationary F-K pattern, it provides excellent moveout information of the events. However, in the situation where the signal has non-stationary frequency or wavenumber pattern, it can only provide an average representation for all the frequency or wavenumber change. From the three examples given above, the first two examples are events with stationary signal patterns, which means the distributions of the Fourier equivalents of the time and space do not change with the time and space. In that case, traditional F-K transform provides an excellent F-K representation of the events. On the contrary, the K-MPD transform does not work as well as the traditional F-K transform because less information could be included into the K transform due to the limited time duration of the wavelets. However, in the situation where the event has non-stationary signal as shown in example three, the traditional F-K is unable to reflect the frequency change with time. It is a smeared image and the F-K distribution is not well localized. On the contrary, the K-MPD transform provides better F-K representation at each time because of the excellent time localization property of the MPD.

The K-MPD works better where the events are composed of non-stationary signals. In the case where more than one wavelet are extracted for an event, the F-K plots generated from the K-MPD transform at each time only represent part of the moveout information of the event because different wavelets are uncorrelated to each other even though they are extracted from the same event by the MPD. The F-K images generated by the K-MPD are not as sharp as those generated by the traditional F-K transform in the case that the events are composed of stationary signal. The signal patterns in the

seismogram are variant. Hence, careful design of the multi-channel processing including different processing procedures need to be used in discovering the accurate moveout features of seismic events.

Chapter 7 Discussion and Conclusion

Spectral analysis is an important procedure in seismic data processing. It uncovers many features contained in the signal other than what can be represented in time alone. Many methods have been used to perform spectral analysis.

Traditional Fourier analysis provides a projection of the signal from the time domain to the frequency domain. Every point from the time representation of the signal contributes to the frequency representation of the signal in the Fourier transform, and vice versa. It supplements the information about the data. However, the time information is completely lost after the Fourier transforms. Hence, for non-stationary data where the frequency content changes with time, the frequency representation after the Fourier transforms is an average representation of the frequency content change through all the time. It can't reflect the instantaneous change of the frequency content with time. For those highly non-stationary data that have many local features, the Fourier transform may even provide misleading frequency features of the data.

The Short Time Fourier Transform provides the time localization of the frequency content by performing the Fourier transform on the data cut from the signal with a window and assuming that the result belongs to the point at the center of the window. The shape, size and shift of the window are the important aspects in the design of the STFT. From the examples given above, two types of window shape are tested, the boxcar windows and the Gaussian windows. The application of the Gaussian window generally

provides a better time resolution because of the excellent time and frequency localization property of the Gaussian window. The sidelobe problem is much more pronounced with the application of the boxcar window. Therefore, Gaussian windows provide a better choice of the window in many situations. However, despite of all the improvements, the STFT could not provide a time-frequency representation that is well adapted to the time-frequency distribution of the signal because the window size is fixed in the STFT. The choice of the window size depends on the signal. When the signal contains many low frequency contents (e.g., less than 15 Hz), a relatively long window size may provide a better representation, e.g., 64 samples. While for the signal contains many middle-to-high frequency contents (e.g., 30 Hz to 80 Hz), a small window size may be preferred. In most situations, shifting the window on a fine grid provides a sufficient representation.

The Continuous Wavelet Transform (CWT) provides a better time-frequency representation of the signal than the STFT in that the window size changes with the frequency change. From the plots discussed in the above, the window size used in CWT is sensitive to the changes of high-frequency content. Therefore, the CWT has good time localization property for frequency contents within high frequency range. The CWT provides a representation that is well adapted to signals with continuous time-frequency distribution such as chirps. However, it is still an average representation of the frequency content within a window and can not provide a good time-frequency representation to signals that have many local features because the time localization of the CWT is not sensitive to the changes of frequency content in the range of low to middle frequencies. Hence, it can not provide a flexible time-frequency representation for signals that have many local features within the low to middle frequencies.

The Wavelet Packet is an algorithm to calculate time-frequency representation based on a best wavelet tree selected from a wavelet binary tree. It is an algorithm derived from Multi-Resolution Analysis. A wavelet binary tree is a family of orthonormal bases composed of vectors that are well localized both in time and frequency. It provides a scale-variant approximation of the signal. However, large shift of the window is taken in order to generate an orthonormal basis. Hence, it may not provide a sufficient representation on all the scale levels.

The MPD is a redundant method that chooses a wavelet among a family of wavelets to match the residue of the signal best at each iteration. It is well adapted to the local features. In a seismogram, the data can, to the first order, be represented by a convolution model, which is the convolution between the wavelet and discrete reflection coefficients. A seismogram is generally highly non-stationary data with mainly middle-frequency content. Using MPD provides sharp time-frequency localization for patterns with non-stationary middle-frequency contents. However, for signals with high-frequency non-stationary contents, MPD using a Ricker wavelet can not provide a good frequency resolution because a Ricker wavelet with high central frequency has a broad frequency spectrum.

In the STFT, the wavelets with better time and frequency localization property prove to be a better choice of the window. Therefore, the application of Gaussian windows shows great advantages over the boxcar windows. However, CWT proves to be an improvement to the STFT both in theory and application. If the signal has continuous

time frequency distribution, which means the frequency at a single time point is somewhat related to the nearby points, CWT may provide a good time-frequency representation. If the signal has many local features within the low to middle frequency range, the MPD using the Ricker wavelets proves to be a good method in giving a time-frequency representation.

In the real seismic data, many types of noise coexist with discrete reflections, such as coherent noise and incoherent noise. Most of the signals in seismic data are non-stationary and have many different time-frequency features. Many methods of the wavelet transform have their own advantages of providing time-frequency distribution to the signal. Different methods of F-T analysis need to be used to understand the frequency distribution of the signal, depending on the situation.

References

1. Mallat S. and Zhang Z. , 1993, Matching Pursuit with time frequency dictionaries: IEEE Trans. , Signal proc., 41, 3397-3415.
2. Avijit Chakraborty and David Okaya, 1995, Frequency-time Decomposition of Seismic Data using wavelet-based methods : Geophysics, 1906-1915.
3. Barnes, A., 1991 , Instantaneous Frequency and Amplitude at the Envelope peak of a constant-phase wavelet: Geophysics, 56, 1058-1060.
4. Bosman c. and Reiter, E., 1993, Seismic Data Compression using wavelet transforms: 63rd Ann. Internet. Mtg., Soc. Expl. Geophysics Expanded Abstracts.
5. Stephene Mallat, 1997, a Wavelet Tour of Signal processing.
6. Mladen Victor Wickerhauser, 1993, Adapted Wavelet Analysis from Theory to Software.
7. Jean-Pierre Kahane, 1995, Fourier Series And Wavelets.
8. Ingrid Daubechies, 1992, Ten Lectures on Wavelets.
9. Rene Carmona, Wen-Liang Hwang and Bruno Torresani, 1998, Practical Time-Frequency Analysis.
10. Patrick Flandrin, 1999, Time-Frequency/Time-Scale Analysis.
11. S. Mallat, 1989, A Theory for Multi-resolution Signal Decomposition: The Wavelet Representation: IEEE Trans. Pattern Analysis and Maxhine Intelligence, 11, 674-693.
12. R. Coifman and V. Wickerhauser, Entropy-based Algorithms for Best Basis Selection: IEEE Trans. Informat. Theory, vol. 38, Mar. 1992.

13. Li-Xiao-Ping, Wavelet Power Spectrum Analysis of Heterogeneity from Sonic Velocity Logs: Geophysical Prospecting, 46; 5, 455-475,1998.

14. Deighan-Androw.J, Watts-Doyle-R, Ground-roll Suppression using the Wavelet Transform: Geophysics 62; 6, 1896-1903, 1997.

15. Grubb-H-J, Walden-A-T, Characterizing Seismic Time Series Using the Discrete Wavelet Transform: Geophysical Prospecting. 45; 2, 183-205, 1997.

16. Li-Xin-Gong, Ulrych-Tadeusz-J, Tomography Via Wavelet Transform Constraints: SEG Annual Meeting Expanded Technical Program Abstracts with Biographies. 65; 1070-1073, 1995.

17. Ozdogan Yilmaz, 1987, Seismic Data Processing.

APPENDIX

A sample program for the extraction of the Ricker wavelets from the signal is given in this appendix. It is composed of two part, the main program and a subroutine to generate a phase shifted Ricker wavelet.

The following program is the main program.

```
c       The following code is used to extract  the wavelets
c       from the data. The input file is 'data'. It is a
c       trace with 256 samples. The sampling rate is 4ms.
c       The output file is 'temp_data'. It contains the shift,
c       the central frequency, the wavelet coefficient, the
c       correlation coefficient, and the energy decreasing
c       coefficient of the wavelet extrated at each iteration.


      INTEGER LT,LF
      REAL W2(256,128)
      REAL SR
      REAL RICK(512),SAM,CORL(512),COF,FUN(512),D(512),Y,Z
      INTEGER M,MAX,J,K,LY,I,LW,TSTART,TBEGIN,TSHIFT,TEMP
      INTEGER SHIFT,S,COUNT,Q,FLAG,NFFT,S1,RECL
      REAL X,CF,F

C       The length of the trace is 256 samples. The sample rate
c       is 4ms.

      LT=256
      TSTART=1-LW/2
      SR=0.004
      TSHIFT=0
      COF=0.
      PSHIFT=0.
      RECL=0

c       Input file is 'data'.

      OPEN(2,FILE='data')
      DO 7 I=1,LT
      READ(2,*)FUN(I)
    7 CONTINUE
      CLOSE(2)

      A=0.
      Y=1.
      MAX=1
      FLAG=1

c       The output file is 'temp_data'.
      OPEN(3,FILE='TEMP_DATA')

c       'flag' is the indication of the decrease of the energy.

      DO WHILE(FLAG==1)
      MAX=1
```

```
      COUNT=0
      A=0.
      A1=0.

      DO 10 I=1,100
      IF(I.LE.5)   THEN
        LW=128
      ELSE IF(I.LT.12)  THEN
        LW=64
      ELSE
        LW=32
      END IF
      F=REAL(I)
      SAM=0.
      COF=0.

      CALL RICKROT(RICK,LW,F,SR,TSTART,PSHIFT,COF)

      TBEGIN=1-LW/2
      LY=LT+LW-2
      DO 15 J=1,512
      CORL(J)=0.
15    CONTINUE

      MAX=1
      DO 25 J=1,LY
      TSHIFT=TBEGIN+J-1
      E1=0.
      DO 30 K=1,LW
      TSTART=INT(1-LW/2)
      M=TSTART+K-1+TSHIFT
      IF((M.GE.1).AND.(M.LE.LT))  THEN
        SAM=FUN(M)
      ELSE
        SAM=0.
      END IF
      E1=E1+SAM**2
      CORL(J)=CORL(J)+SAM*RICK(K)
30    CONTINUE
      CORL(J)=CORL(J)/SQRT(COF)
      IF(E1.GT.(1E-3))  THEN
        CORL(J)=CORL(J)/SQRT(E1)
      ELSE
        CORL(J)=0.
      END IF
      IF((ABS(CORL(J))-ABS(CORL(MAX))).GT.(1E-4))  THEN
            MAX=J
            SHIFT=TSHIFT
      END IF
25    CONTINUE
      IF((ABS(A)-ABS(CORL(MAX))).LT.(1E-4)) THEN
         A=CORL(MAX)
         S=SHIFT
         PHASE=PSHIFT
         CF=F
         NFFT=LW
      END IF
      E1=0.
      E2=0.
```

```
 10  CONTINUE


c  RICKROT is a subroutine to give a given phase Ricker wavelet.

      CALL RICKROT(RICK,NFFT,CF,SR,TSTART,PHASE,COF)
      AMP=0.
      TSTART=1-NFFT/2
      DO 70 I=1,NFFT
      M=TSTART+S+I-1
      IF((M.GE.1).AND.(M.LE.LT))  THEN
          AMP=AMP+FUN(M)*RICK(I)
      END IF
 70   CONTINUE


      AMP=AMP/COF
      E1=0.
      DO 60 I=1,LT
      E1=E1+FUN(I)**2
 60   CONTINUE
      DO 80 I=1,NFFT
      M=TSTART+S+I-1
      IF((M.GE.1).AND.(M.LE.LT))  THEN
          FUN(M)=FUN(M)-AMP*RICK(I)
      END IF
 80   CONTINUE
      E2=0.
      DO 90 I=1,LT
      E2=E2+FUN(I)**2
 90   CONTINUE

      X=Y
      Y=Y*E2/E1
      WRITE(3,*)S,CF,AMP,A,Y
      WRITE(*,*)S,CF,AMP
      RECL=RECL+1
      DO 200 I=1,LT
      IF(ABS(FUN(I)).LE.0.1)  THEN
      COUNT=COUNT+1
      END IF
 200  CONTINUE
      IF(((X-Y).LT.(1E-3)).AND.(COUNT.GT.LT-4))  THEN

         FLAG=0
      ELSE
         FLAG=1
      END IF
      END DO

      CLOSE(3)

      END
```

The following program is the subroutine to generate a ricker wavelet with a given phase.

```
c subroutine rickrot provides a ricker wavelet with phase rotation
c wvlt = output wavelet (output)
c nfft = length of output wavelet (must be a power of 2) (input)
c cf = center frequency for ricker (input)
c sr = time sampling rate for wavelet (input)
c tstart = start time for wavelet (output)
c pshift = phase rotation (input)
      subroutine rickrot(wvlt,nfft,cf,sr,tstart,pshift,cof)
      dimension wvlt(512),amp(512),phase(512),temp(512)
      complex work(512),sig(512)
      integer nfft,f,k,w,tstart
      real pi,sr,pshift,dis,vel
      cof=0.
C      write(*,*)nfft,cf,sr
      pi=3.14159265
      do 10 i=1,512
      wvlt(i)=0.
      temp(i)=0.
      amp(i)=0.
      phase(i)=0.
      work(i)=(0.,0.)
      sig(i)=(0.,0.)
10    continue
      npts=nfft-1
      call ricker(cf,sr,wvlt,npts,tstart)
      signi=1.
      do 11 i=1,nfft
      work(i)=cmplx(wvlt(i),0.)
11    continue
      call fork(nfft,work,signi)
      do 2 i=1,nfft
      f=int((i-1)/(sr*nfft))
      arg=real(work(i))**2 + aimag(work(i))**2
      amp(i)=sqrt(arg)
      phase(i)=0.
2     continue
c add phase shift to phase spectrum
      do 4 i=1,nfft
      shift=pshift
      if(i.gt.nfft/2) shift=-pshift
      phase(i)=phase(i)+shift*3.14159265/180.
4     continue

c convert phase spectrum to complex array sig
      call specomr(sig,amp,phase,nfft)

c convert to time domain
      signi=-1.
      call fork(nfft,sig,signi)
      m=nfft/2
      do 5 i=1,nfft/2-1
      t=tstart+(i-1)*sr
      temp(i)=real(sig(m+i+1))
      temp(m+i)=real(sig(i+1))
5     continue
```

```
      temp(m)=real(sig(1))
      do 2010 i=1,nfft
      t=tstart+sr*(i-1)
      wvlt(i)=temp(i)
      cof=cof+wvlt(i)**2
2010  continue
      RETURN
      end

c  A subroutine to give a zero-phase Ricker wavelet.
      subroutine ricker(cf,sr,wvlt,npts,tstart)
      real wvlt(npts)
      integer tstart
      tstart=-(npts+1)/2+1
      pi=3.14159265
      do 1 i=1,npts
      t=(tstart+i-1)*sr
      wvlt(i)=(1.-2.*(pi*cf*t)**2)*exp(-(pi*cf*t)**2)
1     continue
      return
      end

c  A subroutine to calculate the Fourier spectrum of a given data.
      SUBROUTINE FORK(lx,cx,signi)
      complex cx(lx),carg,cexp,cw,ctemp
      j=1
      sc=sqrt(1./lx)
      do 30 i=1,lx
      if(i.gt.j) go to 10
      ctemp=cx(j)*sc
      cx(j)=cx(i)*sc
      cx(i)=ctemp
10    m=lx/2
20    if(j.le.m) go to 30
      j=j-m
      m=m/2
      if(m.ge.1) go to 20
30    j=j+m
      l=1
40    istep=2*l
      do 50 m=1,l
      carg=(0.,1.)*(3.14159265*signi*(m-1))/l
      cw=cexp(carg)
      do 50 i=m,lx,istep
      ctemp=cw*cx(I+L)
      cx(i+l)=cx(i)-ctemp
50    cx(i)=cx(i)+ctemp
      l=istep
      if(l.lt.lx) go to 40
      return
      end

c specomr: convert amplitude and phase to complex array
      subroutine specomr(sig,amp,phase,nf)
      complex sig(nf)
      real amp(nf),phase(nf)
      do 1 i=1,nf
      x=amp(i)*cos(phase(i))
      y=amp(i)*sin(phase(i))
      sig(i)=cmplx(x,y)
```

```
    1     continue
          return
          end

c specomf: convert complex array to amplitude and phase
          subroutine specomf(sig,amp,phase,nf)
          complex sig(nf)
          real amp(nf),phase(nf)
          do 1 i=1,nf
          x=real(sig(i))
          y=aimag(sig(i))
          amp(i)=sqrt(x**2+y**2)
          if(x.eq.0.) phase(i)=45.*3.14159265/180.
          if(x.eq.0.) go to 1
          phase(i)=atan(y/x)
    1     continue
          return
          end
c costap: generate a tapered cosine wavelet
          subroutine costap(cf,sr,nsamp,wvlt)
          dimension wvlt(256)
c calculate period,start-time, and sample rate
          period=1./cf
          stime=-1.5*period
          sr=1.5*period/nsamp
c zero out original array
          do 10 i=1,nsamp
10        wvlt(i)=0.
c calculate cosine function
          taper=1.
          do 1 i=1,nsamp
          time=stime+(i-1)*sr
          if(time.lt.stime) go to 1
          if(time.gt.-stime) go to 1
          taper=(1.-time/stime)
          wvlt(i)=cos(2.*3.14159265*cf*time)*taper
c         write(7,*)time,wvlt(i)/taper,wvlt(i)
1         continue
          return
          end
```