AN ABSTRACT OF THE THESIS OF

Gayani N. K. Gamage for the degree of Master of Science in Electrical and Computer Engineering presented on August 1, 1986.

Title: A Survey of Optimal Frequency Domain Deconvolution Methods

Redacted for Privacy

Abstract approved: _____
Roy C. Rathja

The resolution of multiple overlapped reflections is achieved by designing inverse filters using frequency domain deconvolution. A careful study reveals a number of problems involving frequency domain design of spike inverse filters. These include the presence of noise and aliasing in both the filter response and its output resulting from basic wavelets with zeros on or near the unit circle. Although Gaussian filters of much shorter length can be designed in some instances, they result in poor resolution. Higher resolution spike filters are designed using two frequency domain deconvolution methods that reduce the noise and aliasing problems while maintaining a reasonable accuracy. A comparison of the two methods for three different types of basic wavelets is also presented.

A Survey of Optimal Frequency Domain Deconvolution Methods

by

Gayani N. K. Gamage

A THESIS
Submitted to
Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Completed August 1, 1986
Commencement June 1987

APPROVED:

Redacted for Privacy

Associate Professor of Electrical and Computer Engineering
in charge of major

Redacted for Privacy

Head of Department of Electrical and Computer Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented August 1, 1986

ACKNOWLEDGMENT

I would like to acknowledge Professor Roy C. Rathja for his guidance and critical comments. I am grateful for his enthusiasm and support which made this research a most enjoyable and educational experience.

I would also like to express my appreciation to Professors J.H. Herzog, W. Kolodziej and R. Layton for their suggestions and comments as well as for serving on my dissertation committee.

Finally I wish to thank my husband, Nimal and my parents for their tremendous encouragement and support.

TABLE OF CONTENTS

		page
1.0	INTRODUCTION	1
	1.1 The Deconvolution Model	3
	1.2 Scope of the Thesis	8
2.0	PROBLEMS IN FREQUENCY DOMAIN DECONVOLUTION	9
	2.1 Deconvolution Noise	10
	2.2 Aliasing in the Time Domain	10
	2.3 Approximation of Nonminimum Phase Sequences	13
	2.3.1 Introduction of Output Delay	19
	2.4 Proximity of Poles of H(z) to the Unit Circle	22
	2.4.1 Use of Gaussian Reflector Series	25
3.0	FREQUENCY DOMAIN DECONVOLUTION METHODS	30
	3.1 Method 1 - Optimal Compensation Method	31
	3.2 Application of Method 1	35
	3.2.1 Examples	36
	3.2.2 Effect of the Parameter	36
	3.2.3 Optimization Criterion	37
	3.2.4 Design of Inverse Filter	44
	3.3 Method 2 - Least Squares Regression Method	52
	3.4 Comparison of the Two Methods	57
	3.5 Reduction of Aliasing Using the Two Methods	61
	3.6 Adaptive Filtering	63
4.0	RESOLUTION OF MULTIPLE OVERLAPPED SIGNALS	71
	4.1 The Low Pass Wavelet	72
	4.2 The Narrow Band Wavelet	74
	4.3 The Wide Band Wavelet	74
	4.4 The Chirp Wavelet	77
5.0	SUMMARY/CONCLUSION	82
	BIBLIOGRAPHY	88
	APPENDIX	89

LIST OF FIGURES

1.1 Representation of the Inverse Filter as a Linear Discrete System 4

2.1 A Minimum Phase Sequence 16
(a) Pole Zero Configuration in the Z Plane
(b) Inverse Filter Unit Sample Response Obtained by FFT.

2.2 A Maximum Phase Sequence 17
(a) Pole Zero Configuration in the Z Plane
(b) Left Handed Sequence in the Time Domain
(c) Inverse Filter Sequence Obtained by FFT.

2.3 A Mixed Phase Sequence 18
(a) Pole Zero Configuration in the Z Plane
(b) Two Sided Sequence in the Time Domain
(c) Inverse Filter Sequence Obtained by FFT.

2.4 Introduction of Output Delay 21
(a) Inverse Filter Response with 32 Sample Delay.
(b) Output for Zero Delay.
(c) Output for a 32 Sample Delay.

2.5 A Sin Pulse 23
(a) Basic Wavelet in Time Domain.
(b) Magnitude Response of Basic Wavelet.

2.6 Inverse Filter for Sin Wavelet 24
(a) Unit Sample Response.
(b) Output Obtained by Linear Convolution.
(c) Output Obtained by Circular Convolution.

2.7 Gaussian Pulse - Mean = 64, Std. Dev. = 2.5 27
(a) Gaussian Pulse in Time Domain.
(b) Magnitude Response of Gaussian Pulse.

2.8 Triangular Wavelet 28
(a) Basic Wavelet.
(b) Magnitude Response of Basic Wavelet.

2.9 Inverse Filter for Triangular Wavelet 29
(a) Unit Sample Response for Spike Output.
(b) Unit Sample Response for Gaussian Output.

3.1 Unit Sample Response of Inverse Filter[h(n)] 38
(a) $\beta = 1.0$ (0 dB)
(b) $\beta = 0.01$ (-20 dB)
(c) $\beta = 0.0001$ (-40 dB)

3.2 Graphs of Accuracy and Noise Vs. β for h(n) 40
(a) Using the Low Pass Basic Wavelet.
(b) Using the Narrow Band Basic Wavelet.
(c) Using the Wide Band Basic Wavelet.

3.3 Graphs of Accuracy and Noise Vs. β for y(n) 42
(a) Using Low Pass Basic Wavelet.
(b) Using Narrow Band Basic Wavelet.
(c) Using Wide Band Basic Wavelet.

3.4 Example 1 45
(a) The Low Pass Basic Wavelet.
(b) Magnitude Response of Low Pass Wavelet.

3.5 Inverse Filter for Example 1 46
(a) Magnitude Response.
(b) Unit Sample Response.
(c) Output for a Single Echo.

3.6 Example 2 47
(a) The Narrow Band Basic Wavelet.
(b) Magnitude Response of Narrow Band Wavelet.

3.7 Inverse Filter for Example 2 48
(a) Magnitude Response.
(b) Unit Sample Response.
(c) Output for a Single Echo.

3.8 Example 3 49
(a) The Wide Band Basic Wavelet.
(b) Magnitude Response of Wide Band Wavelet.

3.9 Inverse Filter for Example 3 50
(a) Magnitude Response.
(b) Unit Sample Response.
(c) Output for a Single Echo.

3.10 Accuracy Vs. $\beta(\Gamma)$ for Methods 1 and 2. 58
(a) Using Low Pass Basic Wavelet.
(b) Using Narrow Band Basic Wavelet.
(c) Using Wide Band Basic Wavelet.

3.11 Noise Vs. $\beta(\Gamma)$ for Methods 1 and 2 59
(a) Using Low Pass Basic Wavelet.
(b) Using Narrow Band Basic Wavelet.
(c) Using Wide Band Basic Wavelet.

3.12 R Vs. $\beta(\Gamma)$ for Methods 1 and 2 60
(a) Using Low Pass Basic Wavelet.
(b) Using Narrow Band Basic Wavelet.
(c) Using Wide Band Basic Wavelet.

3.13 Output of Inverse Filter for Example 2 (β=0.01) 62
(a) By Linear Convolution with Narrow Band Wavelet.
(b) By Circular Convolution with Narrow Band Wavelet.

3.14 Magnitude Response of Adaptive Filter - Method 1 65
(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

3.15 Unit Sample Resp. of Adaptive Filter - Method 1 66
(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

3.16 Magnitude Response of Adaptive Filter - Method 2 68
(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

3.17 Unit Sample Resp. of Adaptive Filter - Method 2 69
(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

4.1 Resolution of Low Pass Wavelets 73
(a) Composite Signal.
(b) Filtered Signal Using Spike Filter.
(c) Filtered Signal Using Gaussian Filter.

4.2 Resolution of Narrow Band Wavelets 75
(a) Composite Signal.
(b) Filtered Signal Using Spike Filter.

4.3 Resolution of Wide Band Wavelets 76
(a) Composite Signal.
(b) Filtered Signal Using Spike Filter.

4.4 Chirp Wavelet 79
(a) Basic Wavelet.
(b) Magnitude Response of Basic Wavelet.

4.5 Resolution of Chirp Wavelets 80
(a) Composite Signal (as in Table 3).
(b) Filtered Signal Using Spike Filter.
(c) Filtered Signal Using Gaussian Filter.

4.6 Resolution of Chirp Wavelets 81
(a) Composite Signal (as in Table 1).
(b) Filtered Signal Using Spike Filter.
(c) Filtered Signal Using Gaussian Filter.

LIST OF TABLES

		page
1.	Amplitudes and Arrival Times of Echoes for figs. 4.1(a), 4.2(a) and 4.5(a).	71
2.	Amplitudes and Arrival Times of Echoes for fig. 4.3(a).	72
3.	Amplitudes and Arrival Times of Echoes for fig. 4.4(a).	77

A SURVEY OF OPTIMAL FREQUENCY DOMAIN DECONVOLUTION METHODS

1.0 INTRODUCTION

There are many applications in signal processing where we wish to separate component signals of a convolution. Most often a signal is contaminated by the introduction of echoes, called reverberation, and it is necessary to identify them prior to their subsequent removal. Such applications where the resolution of multiple overlapping signals is required include radar, sonar, speech, optical and seismic signal processing. In speech processing, for example, the speech waveform is the convolution of the vocal tract impulse response with the excitation. In seismic applications, the deconvolution of reflected signals enables us to interpret the earths subsurface structure. In this case the reflected waveform can be considered as the convolution of the seismic wavefront with an impulse train, each impulse representing a reflection from a layer of the earth.

The decomposition of the composite waveform (convolution result) $y(n)$, into its components $x(n)$ and $h(n)$ is called deconvolution or inverse filtering.

Representing the convolution operator by a '*',

$$y(n) = x(n) * h(n) \tag{1.1}$$

$$h(n) = x(n)^{-1} * y(n) \tag{1.2}$$

where $x(n)^{-1}$ denotes the convolution inverse given by,

$$x(n)^{-1} * x(n) = \delta(n) \tag{1.3}$$

Taking Fourier Transforms on both sides of the discrete time convolution given in equation (1.1) yields,

$$Y(k) = X(k) \cdot H(k) \tag{1.4}$$

$$H(k) = Y(k)/X(k) \tag{1.5}$$

where the capital letters with the index 'k' in equation (1.5) denote the Discrete Fourier Transform(DFT) of the time domain sequences. It is of importance to mention at this point that the multiplication of DFT's in the frequency domain corresponds to a circular convolution in the time domain. Nevertheless, a circular convolution

could be equated to a linear convolution if the lengths of the sequences satisfies the following relationship.

$$N > N1+N2-1 \qquad (1.6)$$

where N1 and N2 are the lengths of the component signals (x(n) and h(n) in this case) and N is the length of the convolution result (y(n)). Therefore, a linear convolution as given by equation (1.1) transforms into the frequency domain as in equation (1.4) only if the condition given by (1.6) is satisfied. If not, the resulting DFT calculations would produce an aliased version of the true signal.

1.1 THE DECONVOLUTION MODEL [3]

The reflected wave from a layered media could be considered as a composite signal consisting of identical wavelets, differing only in magnitude and the time of arrival. Thus, the composite signal can be thought of as the convolution of a reflector wavelet with a train of pulses of different amplitudes and spacing. In order to identify the echoes in the composite signal, it should be processed to yield a single pulse at the time of arrival of each echo. If there are a number of layers in the media, multiple echoes would occur and a number of pulses equal to the number of reflections or echoes should be generated.

The generation of these pulses can be accomplished by designing an inverse filter as follows.

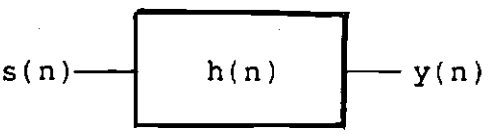

fig. 1.1 Representation of the Inverse Filter as a Linear Discrete System

The output of the filter, y(n) is given by,

$$y(n) = s(n) * h(n) \qquad (1.7)$$

where, s(n) = composite signal

h(n) = inverse filter

The composite signal is the convolution of a reflector wavelet w(n) with a reflector series r(n) as described in the previous paragraph. Hence,

$$s(n) = w(n) * r(n) \qquad (1.8)$$

Substituting for s(n) in equation (1.7),

$$y(n) = w(n) * r(n) * h(n) \qquad (1.9)$$

The desired output is the reflector series which signifies the occurrence of the echoes. Thus,

$$r(n) = w(n) * r(n) * h(n) \qquad (1.10)$$

Equation (1.10) reduces to,

$$w(n) * h(n) = \delta(n) \qquad (1.11)$$

Writing equation (1.8) for the case of a single echo yields,

$$w_b(n) = w(n) * r_s(n) \qquad (1.12)$$

where $w_b(n)$ is called the basic wavelet and $r_s(n)$ represents a single pulse instead of a series.

Writing equations (1.11) and (1.12) in terms of the Z Transform,

$$H(z) = 1/W(z) \qquad (1.13)$$

$$W(z) = W_b(z)/R_s(z) \qquad (1.14)$$

Combining equations (1.13) and (1.14) yields,

$$H(z) = R_s(z)/W_b(z) \qquad (1.15)$$

It is seen from equation (1.15) that the inverse filter can be designed by the deconvolution of the basic wavelet with the reflector series for a single echo. In this investigation it is assumed that the basic wavelet is known. The reflector series can be chosen to be any narrow width pulse train that allows reasonable resolution of the overlapping echoes. A special case of equation (1.15) would be selecting the reflector series to be a series of spikes or impulses. An inverse filter that produces such a spike output is called a spike filter. For such a case,

$$r_s(n) = \delta(n) \text{ and } R_s(z) = 1 \qquad (1.16)$$

As a result,

$$W(z) = W_b(z) \qquad (1.17)$$

and the spike filter is given by,

$$H(z) = 1/W_b(z) \qquad (1.18)$$

Since the DFT corresponds to samples of the Z Transform evaluated around the unit circle, the relationships in equations (1.15) and (1.18) can be written in terms of the DFT's as follows,

$$H(k) = R_s(k)/W_b(k) \qquad (1.19)$$

$$H(k) = 1/W_b(k) \qquad (1.20)$$

The deconvolution performed to obtain the inverse filter can be carried out using time domain as well as frequency domain techniques. Time domain techniques involve the inversion of large matrices, thus making the process tedious and time consuming. On the other hand, frequency domain analysis of the problem involves simple division of the Fourier Transforms of two waveforms. The computation of the DFT's for the discrete time signals can be accomplished with great efficiency by use of Fast Fourier Transform(FFT) routines. The apparent simplicity of the frequency domain deconvolution problem and its reduced computation time makes it extremely useful especially when dealing with long sequences.

1.2 SCOPE OF THIS THESIS

Further investigation of frequency domain deconvolution methods is carried out in the following chapters of this thesis.

Chapter 2 deals with a number of problems encountered in this apparently simple alternative to time domain deconvolution methods. It also provides ways to overcome entirely or reduce the adverse effects of the procedure in designing the filter which is given by equation (1.20).

Chapter 3 describes two frequency domain deconvolution methods that could be used to overcome the deconvolution noise. An illustration of the methods for various types of basic wavelets is shown. A performance measure for the filter is derived and a comparison of the two methods is carried out.

Chapter 4 illustrates the results of the resolution of overlapping signals for a number of different composite waveforms using the filters designed in Chapter 3.

Chapter 5 draws some conclusions concerning frequency domain deconvolution based on the results of the previous chapters.

2.0 PROBLEMS IN FREQUENCY DOMAIN DECONVOLUTION

The procedure presented in the previous chapter for the implementation of frequency domain deconvolution appears simple and straightforward. This chapter deals with situations which threaten the apparent simplicity of its implementation. A brief survey of the number of difficulties encountered in the process include,

1. Presence of deconvolution noise in the solution for the filter unit sample response.
2. Aliasing of the solution in the time domain due to FFT implementation of an Infinite Impulse Response(IIR) filter.
3. Approximation of minimum phase, maximum phase and mixed phase waveforms by a finite length sequence.
4. Very large filter lengths due to poles of H(z) being on or near the unit circle.
5. Reduced detection capability due to aliasing and deconvolution noise.

All of the above problems are associated with the location of poles of H(z) on the z plane or in other words location of zeros of $W_b(z)$ on the z plane. A detailed investigation into these problems and some suggested techniques for overcoming them are discussed in this chapter.

2.1 DECONVOLUTION NOISE

The division of Fourier Transforms of the input and output as given by equation (1.5) could cause the problem to be ill-conditioned when X(k)=0 or Y(k)/X(k) becomes indeterminate. As a result it gives rise to noisy solutions in the filter unit sample response h(n). Two methods are described in chapter 3 which could be used to minimize this deconvolution noise. The effectiveness of these methods on a variety of basic wavelets in a noisy environment is also demonstrated.

2.2 ALIASING IN THE TIME DOMAIN

When dealing with reflections from multilayered media as given in chapter 1, it is most convenient to choose the reflector wavelet to be a unit sample.

Then,

$$H(z) = 1/W_b(z) \qquad (2.1)$$

Since $w_b(n)$ is chosen to be a causal finite length sequence, its z transform will be a polynomial in z^{-1}, with multiple poles at the origin and zeros located elsewhere on the z plane. From equation (2.1), it can be seen that these zeros of $W_b(z)$ become the poles of H(z). This implies that the filter unit sample response is of infinite length, thus producing an IIR filter. If time domain sequences have real coefficients, the poles of its z transform will occur in conjugate pairs and in turn the characteristics of the real sequence will be determined by the location of these poles on the z plane. When a solution to h(n) is sought through a FFT method, a finite length approximation to the IIR filter has to be performed.

First, consider a finite length sequence x'(n) of length N and denote its N point DFT by X'(k). Then,

$$X'(k) = \sum_{n=0}^{N-1} x'(n) \cdot e^{\frac{-j2\pi kn}{N}} \qquad (2.2)$$

The z transform X(z) of an infinite length sequence x(n) is related to X'(k) by the following equation,[1]

$$X'(k) = X(z)\Big|_{z=e^{\frac{-j2\pi k}{N}}} \quad , \; k=0,1,\ldots,N-1 \tag{2.2}$$

Since,
$$X(z) = \sum_{n=-\infty}^{\infty} x(n)\cdot z^{-n} \tag{2.3}$$

from equation (2.2),

$$X'(k) = \sum_{n=-\infty}^{\infty} x(n)\cdot e^{\frac{-j2\pi nk}{N}} \tag{2.4}$$

Writing the above summation in blocks of length N,

$$X'(k) = \sum_{l=0}^{N-1} x(l)\cdot e^{\frac{-j2\pi kl}{N}} + \sum_{l=N}^{2N-1} x(l)\cdot e^{\frac{-j2\pi kl}{N}} + \ldots$$

$$= \sum_{n=0}^{N-1}\sum_{m=-\infty}^{\infty} x(n + mN)\cdot e^{\frac{-j2\pi kn}{N}} \tag{2.5}$$

Comparing (2.2) with (2.5),

$$x'(n) = \sum_{m=-\infty}^{\infty} x(n+mN) \tag{2.6}$$

Therefore, it is seen that the sequence obtained by FFT methods is an aliased version of the true IIR filter. If the significant coefficients of the filter can be included in the length N, the filter can be approximated by a finite sequence of length N with negligible error. In practice the filter length could be increased to include the significant components. This is true in the case of causal minimum phase sequences. In the case of nonminimum phase sequences some adjustment has to be carried out before approximation is performed. The next section deals with the approximation problem and explains how it is performed for different types of sequences, namely minimum phase, maximum phase and mixed phase.

2.3 APPROXIMATION OF NONMINIMUM PHASE SEQUENCES

A simple first order transfer function is chosen to generate the minimum and maximum phase sequences. Consider,

$$H(z) = 1/(1-az^{-1}) \qquad (2.7)$$

It can be seen from the above equation that the pole of H(z) is located at z = a. h(n) will be either a minimum phase or maximum phase sequence depending on whether a < 1 or a > 1.

From equation (2.7),

$$W_b(z) = 1-az^{-1} \qquad (2.8)$$

and it follows that,

$$w_b(n) = \delta(n) - a\delta(n-1) \qquad (2.9)$$

Also by writing in terms of the DFT,

$$H(k) = 1/W_b(k) \qquad (2.10)$$

h(n) can be determined by first evaluating the FFT of $w_b(n)$ (i.e. $W_b(k)$) and then applying equation (2.10) to produce H(k). The Inverse DFT's(IDFT) of H(k) for $a < 1$ and $a > 1$ produce a minimum phase and a maximum phase sequence respectively.

In general if all the poles of H(z) are located inside the unit circle stability requires the sequence h(n) to be right handed. Such a sequence is called a minimum phase sequence. A special case of this occurs if the z transform converges at infinity making it a causal sequence.[1] The above example for this particular case is shown in fig. 2.1(b).

If the poles happen to be outside the unit circle, h(n) will be a left handed sequence and is termed a maximum phase sequence (fig. 2.2(b)). Accordingly, a mixed phase sequence would have its poles both inside and outside the unit circle and will produce a two sided sequence in the time domain (fig. 2.3(b)).[1]

The example selected to illustrate the mixed phase sequence is,

$$H(z) = 1/(1-az^{-1})(1-bz^{-1}) \qquad (2.11)$$

The corresponding basic wavelet will be

$$w_b(n) = \delta(n) - (a+b)\cdot\delta(n-1) + a\cdot b\cdot\delta(n-2) \qquad (2.12)$$

where a = 1.5 and b = 0.6.

Since we are dealing with frequency domain analysis, it is of interest to observe the solution to h(n) using the FFT when $w_b(n)$ is known. The results for the three cases are shown in figs. 2.1(b),2.2(c) and 2.3(c).

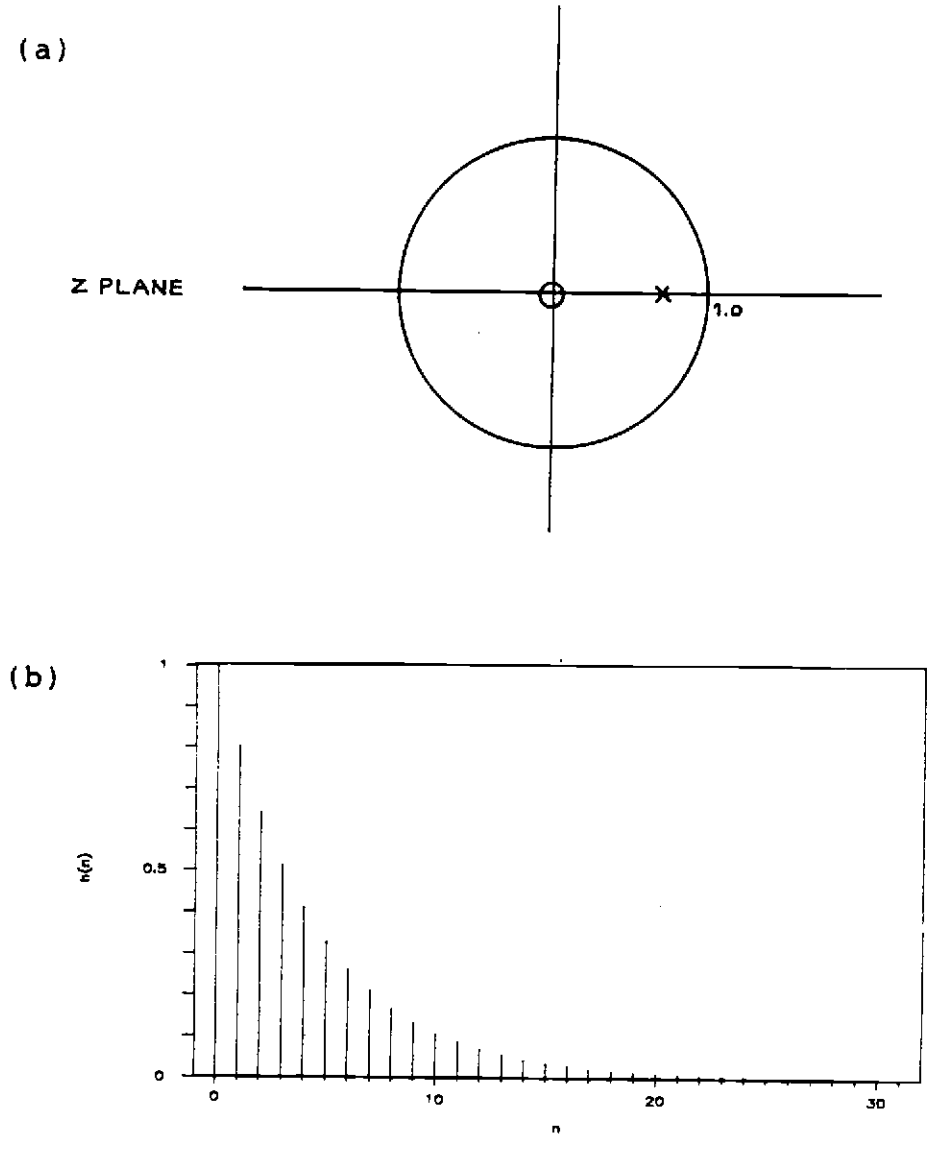

fig. 2.1 A Minimum Phase Sequence

(a) Pole Zero Configuration in the Z Plane.
(b) Inverse Filter Unit Sample Response Obtained by FFT.

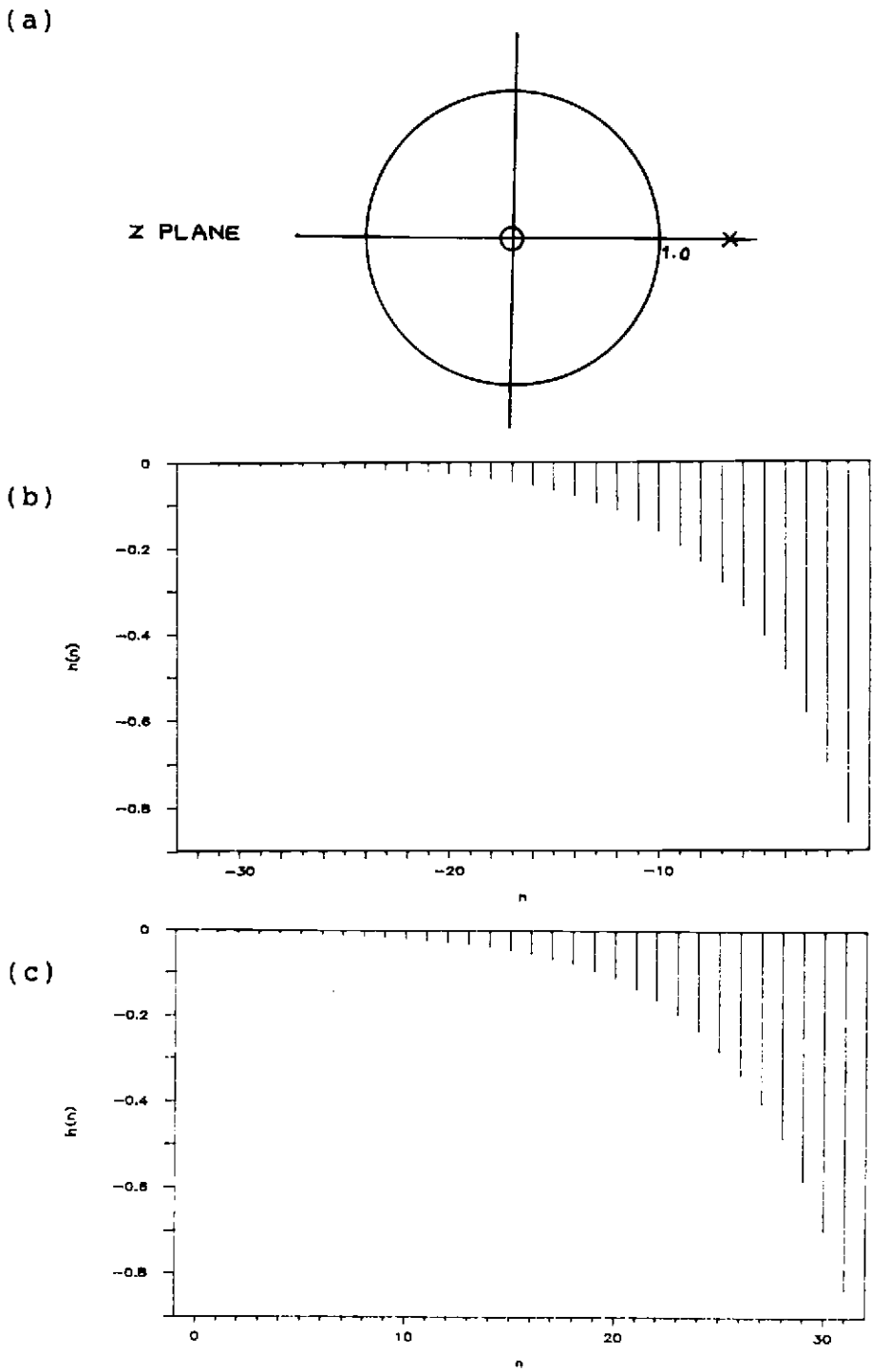

fig. 2.2 A Maximum Phase Sequence.

(a) Pole Zero Configuration in the Z Plane.
(b) Left Handed Sequence in the Time Domain.
(c) Inverse Filter Sequence Obtained by FFT.

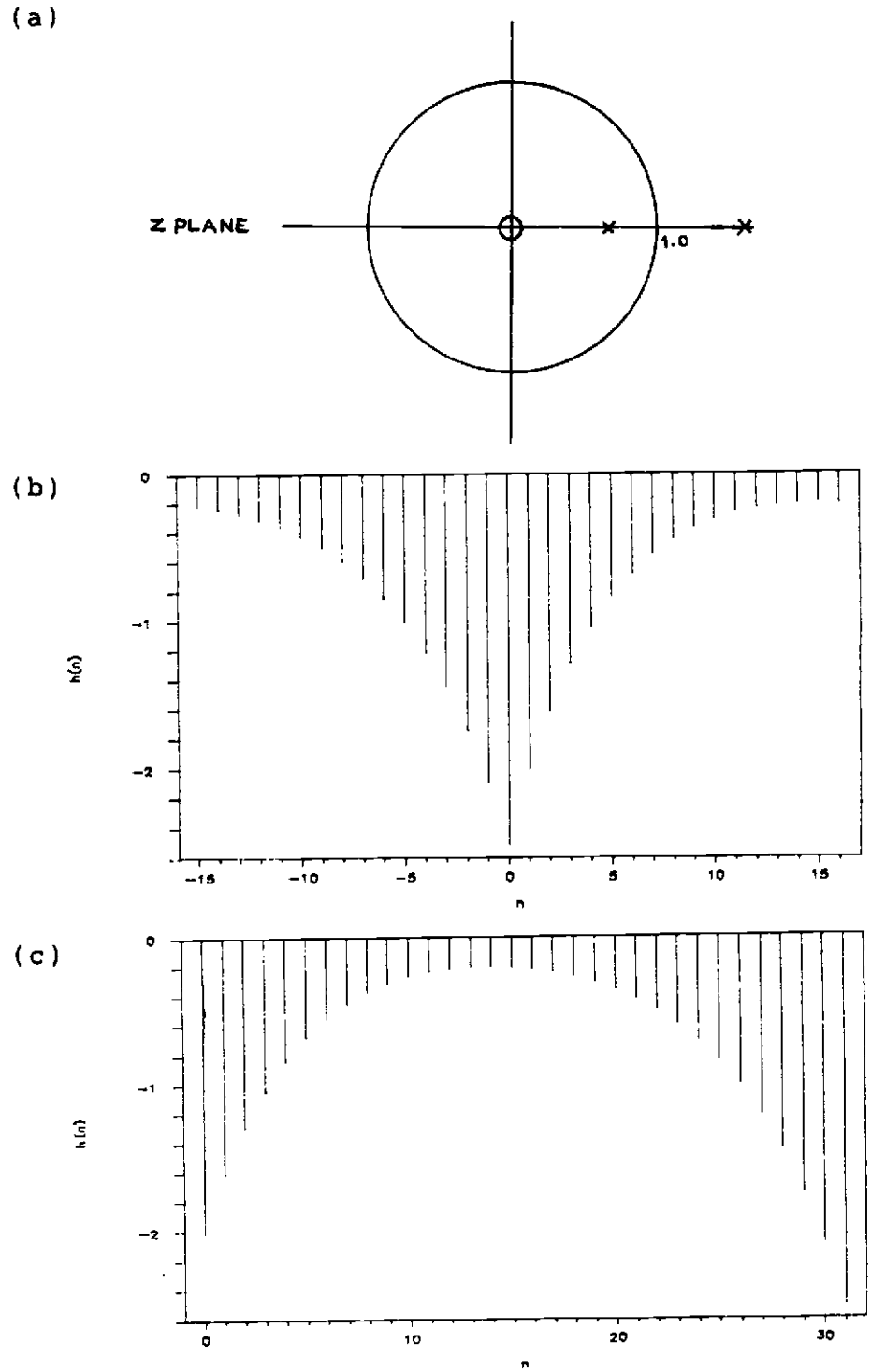

fig. 2.3 A Mixed Phase Sequence.

(a) Pole Zero Configuration in the Z Plane.
(b) Two Sided Sequence in the Time Domain.
(c) Inverse Filter Sequence Obtained by FFT.

It is seen that for minimum phase sequences the inverse filter coefficients are significant at the left end of the sequence and decay towards the right end. For a causal sequence the coefficients start at the origin and the IIR filter can be easily approximated by a finite length N that includes only the significant coefficients. For the maximum phase inverse sequence the significant coefficients are concentrated at the right end while the left end coefficients contribute only a very minimal energy. Such a sequence obtained using FFT methods is shown in fig. 2.2(c). For a mixed phase sequence the significant coefficients can be at both ends. Therefore, approximating non minimum phase inverse sequences by a finite length poses a problem. The desired output can be obtained from this type of aliased filter only through circular convolution, which is not a satisfactory alternative in real time applications.

2.3.1 INTRODUCTION OF DELAY AT THE OUTPUT

By delaying the output, a circular shift can be induced in the filter coefficients such that the significant magnitudes start at the left end of the sequence. As mentioned earlier this type of filter can be easily approximated and implemented in a linear convolution in later applications. In the case of non minimum phase

sequences, a circular shift would produce a shifted version of the true time domain signal. This delayed inverse filter response will produce an accurate output with the same delay.

An example of introducing an output delay is shown in fig. 2.4. The mixed phase filter response for the basic wavelet given by equation (2.12) is computed via a 64 point FFT and an output delay of 32 samples. Fig. 2.4(a) shows the resulting filter sequence which is essentially a circular rotation of the same sequence obtained with zero delay. In order to confirm the proper functioning of the filter, its output is calculated by convolving the basic wavelet with the designed filter response. Figs. 2.4(b) and (c) show the outputs for zero and 32 sample delay filters respectively. Output for a delay of 32 samples proves to be satisfactory while the output for zero delay appears to be aliased.

In general, causal minimum phase sequences will require no output delay. A maximum phase sequence produces best results when the delay is equal to the length N of the sequence. For a mixed phase sequence, optimum delay is an intermediate value between 0 and N.

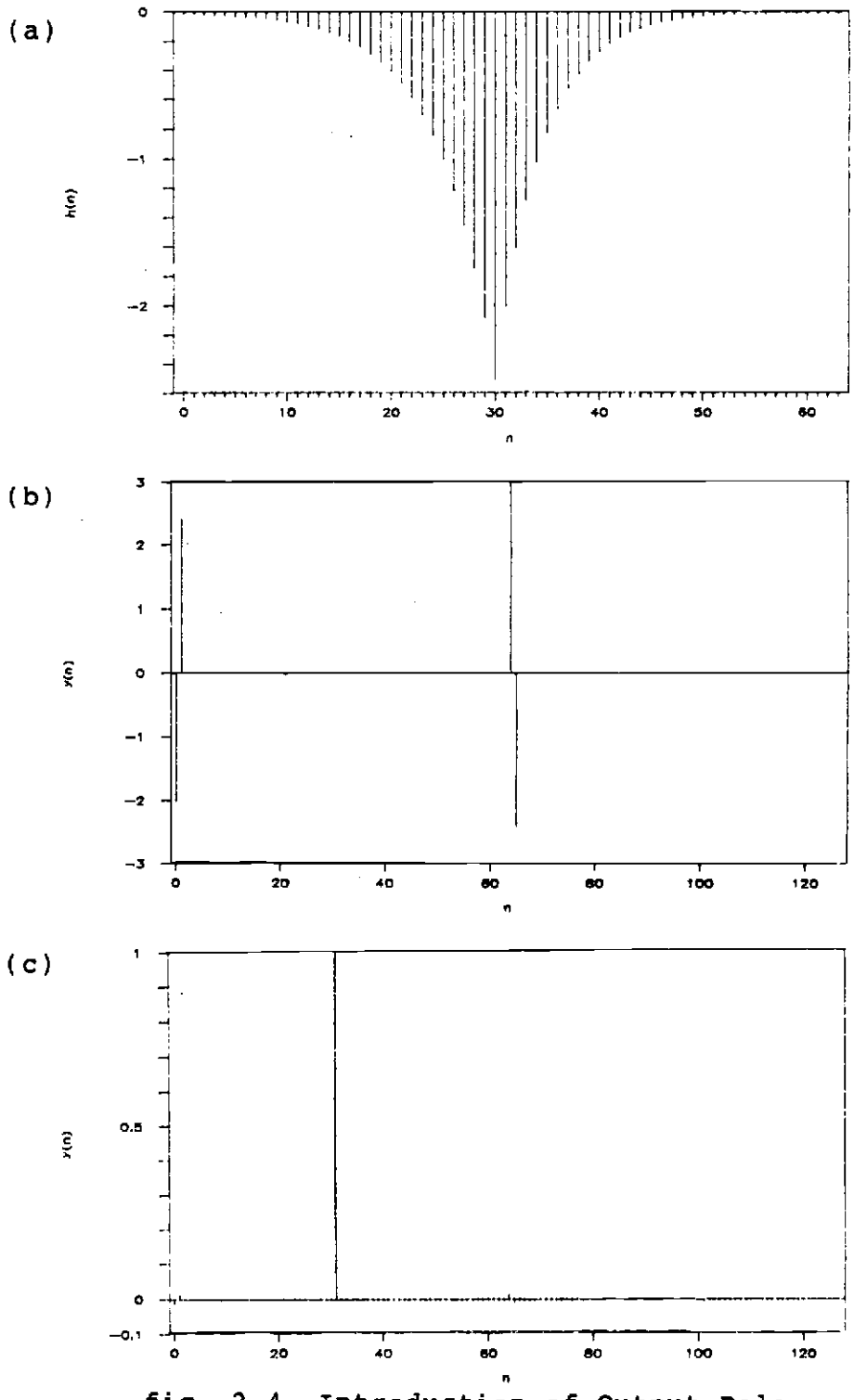

fig. 2.4 Introduction of Output Delay

(a) Inverse Filter Response with 32 Sample Delay
(b) Output for Zero Delay.
(c) Output for a 32 Sample Delay.

2.4 PROXIMITY OF POLES OF H(z) TO THE UNIT CIRCLE

Another problem encountered with the design of this filter is the proximity of poles to the unit circle. As mentioned in section 2.1, around frequencies on the unit circle that are close to a pole, the value of H(k) could be very large. This causes noisy solutions in the unit sample response. Another major draw back is the generation of very long filter sequences. The closer the poles are to the unit circle the longer the sequence. In order to prevent aliasing much longer FFT lengths must be used. An example of such a situation is shown in fig. 2.5, where $w_b(n)$ is chosen to be a sinusoid with the following specifications,

$$w_b(n) = \sin^2(\pi a n), \qquad (2.13)$$

$$n = 0,1,\ldots,45$$
$$a = 1/45$$

The zeros of $W_b(z)$ are placed around the unit circle in equally spaced intervals of $2\pi/N$. Consequently the poles of H(z) are placed in the same order. The aliased version of the filter obtained using the FFT method can be seen in fig. 2.6(a). An unsuccesful attempt to produce an impulse at the output using $w_b(n)$ as the input to this filter is shown in fig. 2.6(b). Although a linear convolution fails to produce the correct result, it can be obtained using a

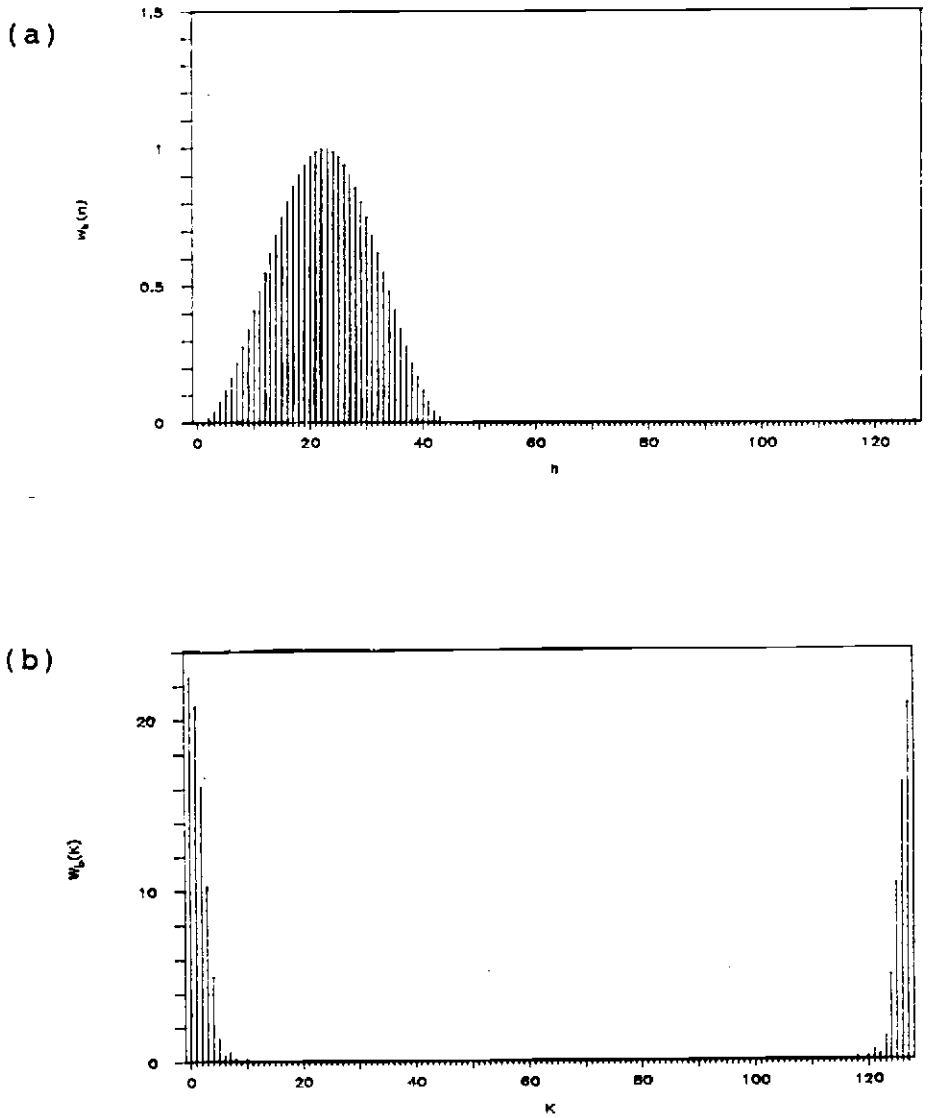

fig. 2.5 A Sin Pulse

(a) Basic Wavelet in Time Domain
(b) Magnitude Response of Basic Wavelet.

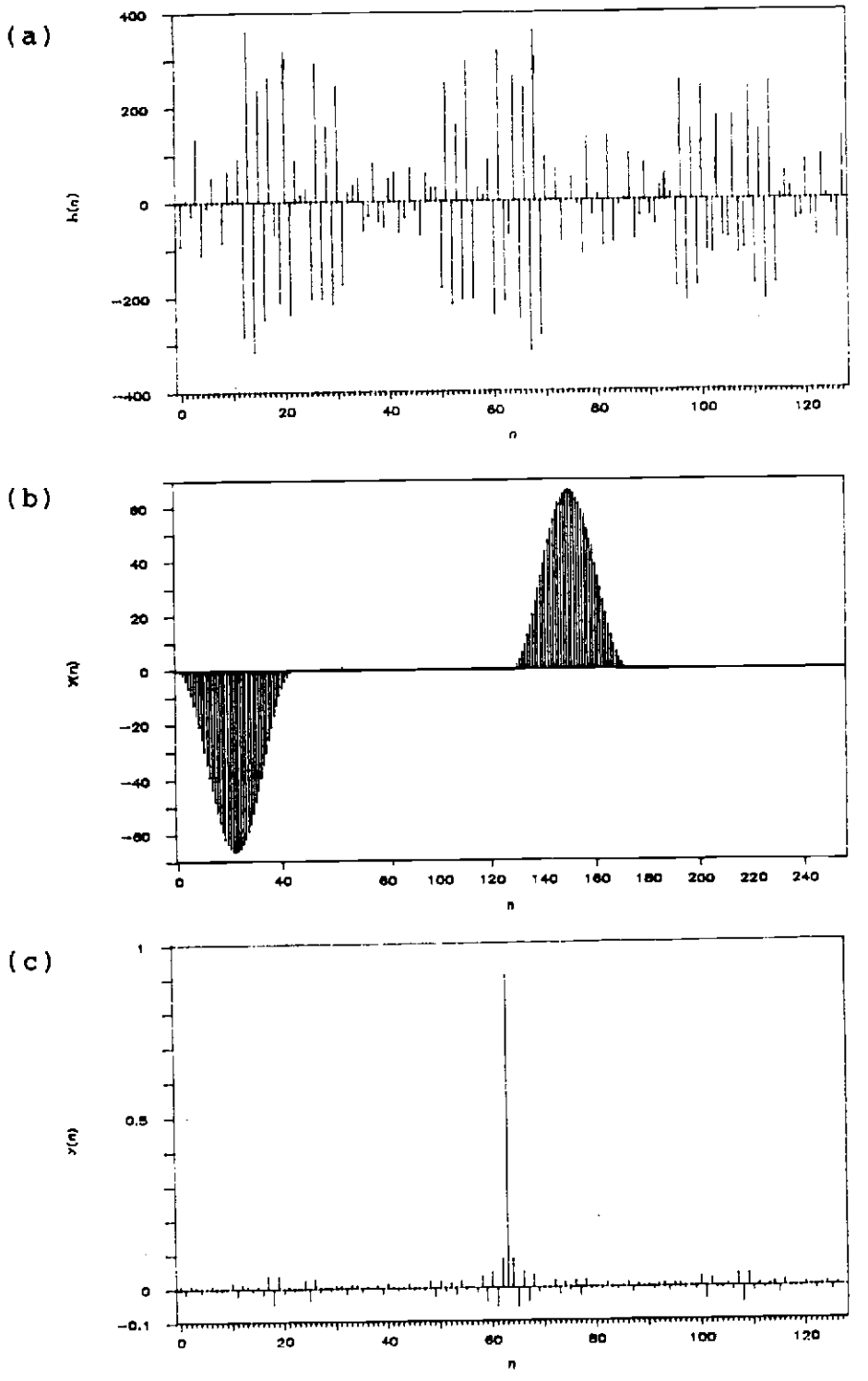

fig. 2.6 Inverse Filter for Sin Wavelet.

(a) Unit Sample Response.
(b) Output Obtained by Linear Convolution.
(c) Output Obtained by Circular Convolution.

circular convolution of the input with the aliased filter response (refer fig. 2.6(c)). Circular convolution is not very suitable in practical situations. Therefore, a different approach which makes it possible to use linear convolution should be thought of. There are a few alternative approaches and the selection of one depends entirely on the nature of the basic wavelet.

2.4.1 USE OF THE GAUSSIAN REFLECTOR SERIES

This problem of large filter lengths mainly occurs with reflector series of impulses. If $R_s(n)$ can be selected such that it cancels the effects of the poles of $W_b(z)$ to a certain extent, then it is possible to obtain shorter filter lengths. One such example is the use of Gaussian reflector series with $w_b(n)$ for which the poles of $H(z)$ are constrained only to one side of the unit circle. The unique relationship between its time and frequency domain representations is clearly seen by observing the Fourier Transform pair for a generalised Gaussian distribution.

$$\frac{1}{\sigma_t\sqrt{2\pi}}\exp\left[\frac{-(t-\mu)}{2\sigma_t^2}\right]^2 \quad \longleftrightarrow \quad \exp\left[\frac{-w^2\ \sigma_w^2}{2}\right]\exp[-j\mu w] \tag{2.14}$$

This implies that the FFT of a Gaussian produces another Gaussian in the frequency domain. The width of the

Gaussian pulse is characterized by its standard deviation σ which relates to the amount of deviation of the distribution from its mean value. The standard deviation in the frequency domain is $\sigma_w = 1/\sigma_t$, from equation (2.14), resulting in an inverse relationship. Thus, a Gaussian with a larger σ_t, i.e. a broader pulse in time domain will result in a narrower pulse in the frequency domain and vice versa. In order to maintain reasonable resolution, the time domain pulse should be made as narrow as possible. The limitation to this is the aliasing of the broader pulse in the frequency domain. It has been observed in previous work [2] that values of σ_t greater than 2 are sufficient in overcoming the problem of aliasing. A Gaussian pulse of $\sigma_t = 2.5$ and its FFT are shown in fig. 2.7. It can be seen from the FFT that the zeros of the Gaussian pulse occur on the left hand side of the unit circle. Hence it can be used for basic wavelets with similar zero configurations to cancel the effects of undesirable pole locations in the filter response. One such example of a basic wavelet, a triangular waveform and its FFT are shown in fig. 2.8. It can be seen that this also has zeros on the left hand plane. The corresponding spike filter designed for this case proves to be grossly aliased because of its nearly infinite length (fig. 2.9(a)). Fig. 2.9(b) shows that this problem could be overcome with a Gaussian filter of much shorter length.

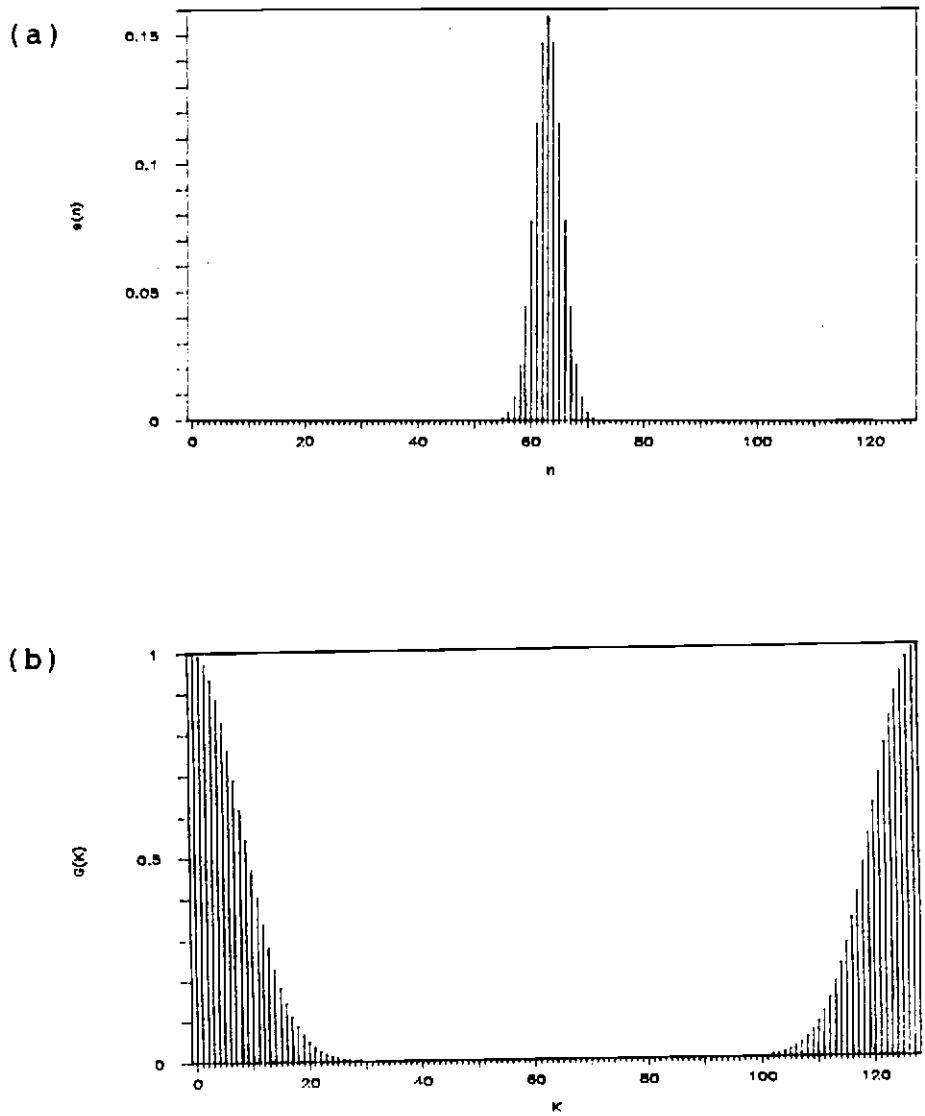

fig. 2.7 Gaussian Pulse - Mean = 64, Std. Dev. = 2.5

(a) Gaussian Pulse in Time Domain.
(b) Magnitude Response of Gaussian Pulse.

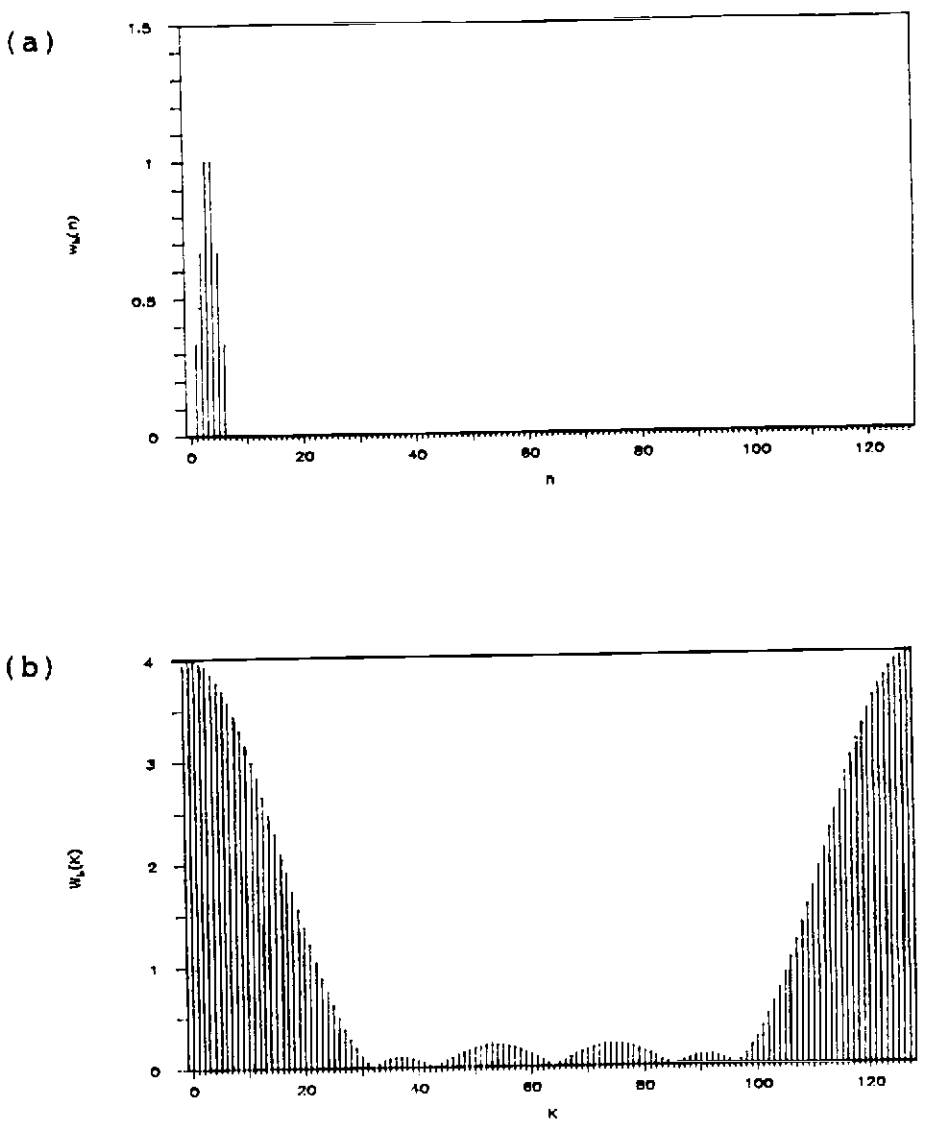

fig. 2.8 Triangular Wavelet.

(a) Basic Wavelet.
(b) Magnitude Response of Basic Wavelet.

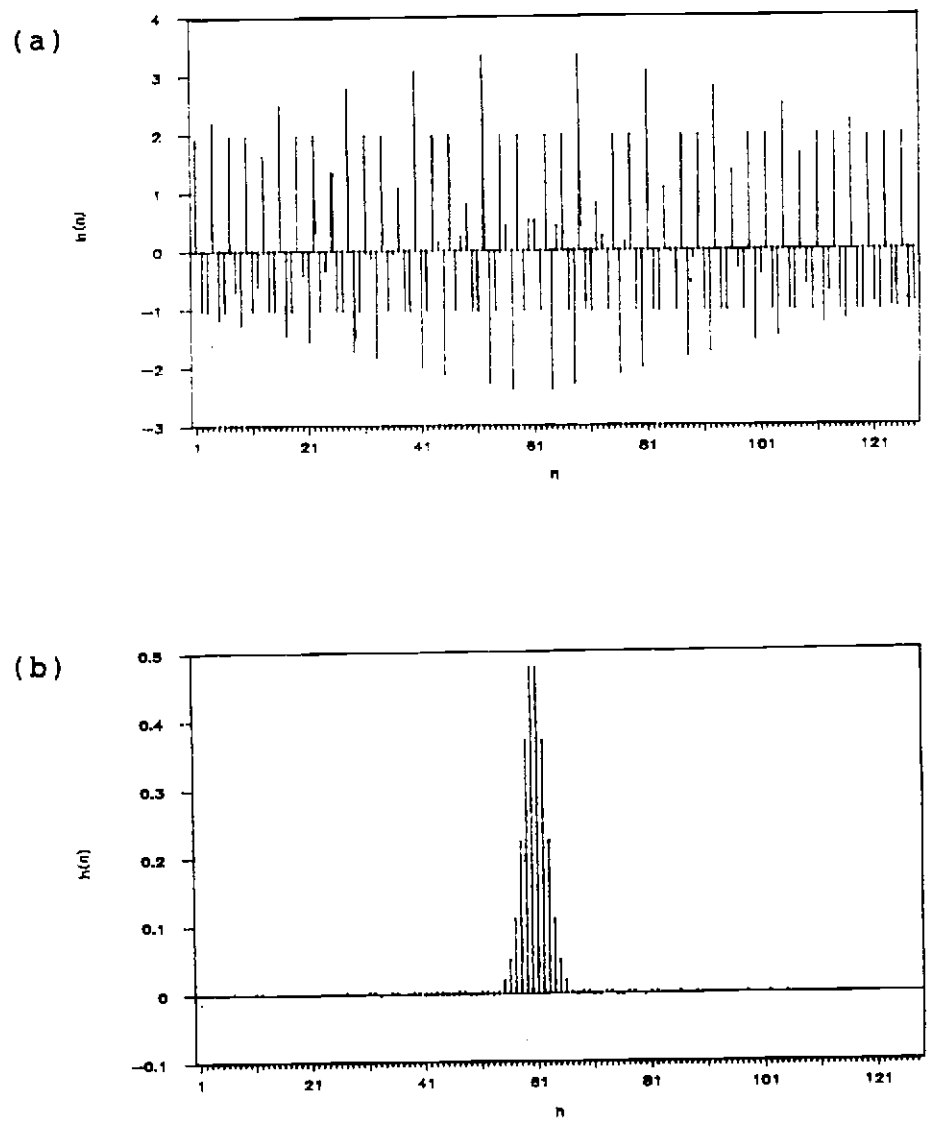

fig. 2.9 Inverse Filter for Triangular Wavelet

(a) Unit Sample Response for Spike Output.
(b) Unit Sample Response for Gaussian Output.

3.0 FREQUENCY DOMAIN DECONVOLUTION METHODS

The two frequency domain deconvolution methods described in this chapter are used to obtain a solution for the inverse filter specified by the deconvolution model introduced in chapter one. They are

1. The optimal compensation technique.[4]

and 2. The least squares regression method.[5]

Although these filters are designed to overcome the deconvolution noise and are particularly effective in this respect, they seem to be useful in reducing some of the other adverse effects mentioned in chapter 2 as well. The first method is designed by imposing conditions entirely in the frequency domain, while the second method is developed through specifications in the time domain which is later translated into its corresponding frequency domain version. In both cases the resulting equation for the frequency domain solution of the inverse filter frequency response depends on the selection of a parameter. The value of the parameter decides the tradeoff between the accuracy of the solution and its noise content. Graphs are obtained which allow a designer to select the region of suitable values of the parameter that produce an optimum filter.

It is also shown that the application of these two procedures is equivalent to using an adaptive filter on the true inverse filter and that the nature of the adaptive filter is entirely dependent on the nature of the frequency response of the basic wavelet.[6]

The derivation of the frequency domain expressions for the two methods are presented in this chapter along with illustrative applications. Also included are the effects of the adaptive filtering process on the solution for the filter unit sample response.

3.1 METHOD 1 - OPTIMAL COMPENSATION TECHNIQUE

In this method an optimal compensator is designed in the frequency domain. The compensator developed by Riad and Stafford [4] operates on the Fourier Transform of the convolution result $Y(e^{jw})$, to produce one of the inputs to the convolution. In this case the unknown input is taken as the filter frequency response. Hence, the compensator $C(e^{jw})$ is applied to $Y(e^{jw})$ to yield an estimate for $H(e^{jw})$.

$$H_e(e^{jw}) = Y(e^{jw}) \cdot C(e^{jw}) \qquad (3.1)$$

There are two major design criteria to be imposed on the design of the compensator.

1. Minimization of the squared error in the estimate defined by,

$$E_e = |H(e^{jw}) - H_e(e^{jw})|^2 \qquad (3.2)$$

2. Minimization of noise which is attributed to $C(e^{jw}) = 1 / X(e^{jw})$ being an unbounded quantity. Since $H(e^{jw})$ is a bounded function, the criterion is that of keeping E_c a finite quantity, where

$$E_c = |H(e^{jw}) \cdot C(e^{jw})|^2 \qquad (3.3)$$

These two criteria can be combined in one equation using an optimization parameter.

$$E = E_e + \beta \cdot E_c, \qquad \beta > 0 \qquad (3.4)$$

The problem is to minimize E_e while maintaining E_c finite. The value of β signifies the amount of weight given to each criterion. If β is small, minimization of error, E_e plays an important role whereas, if β is very large it turns out to be an extreme of keeping E_c finite.

An expression can be developed for the compensator using the above criteria as follows. Substituting equations (3.2) and (3.3) in (3.4) yields,

$$E = |H(e^{jw}) - H_e(e^{jw})|^2 + \beta|H(e^{jw})\cdot C(e^{jw})|^2 \qquad (3.5)$$

Writing $Y(e^{jw}) = H(e^{jw})\cdot X(e^{jw})$ in equation (3.1) gives

$$H_e(e^{jw}) = H(e^{jw})\cdot X(e^{jw})\cdot C(e^{jw}) \qquad (3.6)$$

Substituting for $H_e(e^{jw})$ in equation (3.5) and simplifying,

$$E = |H(e^{jw})|^2\cdot P(e^{jw}) \qquad (3.7)$$

where,
$$P(e^{jw}) = |X(e^{jw})\cdot C(e^{jw}) - 1|^2 + \beta|C(e^{jw})|^2 \qquad (3.8)$$

In order to minimize the error for any general form of $H(e^{jw})$, $P(e^{jw})$ must be minimized. Denoting real and imaginary parts of $X(e^{jw})$ and $C(e^{jw})$ by subscripts R and I,

$P(e^{jw})$ can be written as,

$$P(e^{jw}) = [X_R(e^{jw})\cdot C_R(e^{jw}) - X_I(e^{jw})\cdot C_I(e^{jw}) - 1]^2 + [X_R(e^{jw})\cdot C_I(e^{jw}) + X_I(e^{jw})\cdot C_R(e^{jw})]^2 + \beta[C_R^{\,2}(e^{jw}) + C_I^{\,2}(e^{jw})] \qquad (3.9)$$

Taking partial derivatives and equating them to zero,

$$\frac{Q(e^{jw})}{C_R(e^{jw})} = C_R(e^{jw})[\,|X(e^{jw})|^2 + \beta] - X_R(e^{jw}) = 0 \qquad (3.10)$$

$$\frac{Q(e^{jw})}{C_I(e^{jw})} = C_I(e^{jw})[\,|X(e^{jw})|^2 + \beta] + X_I(e^{jw}) = 0 \qquad (3.11)$$

Manipulating equations (3.10) and (3.11), it can be shown that,

$$C(e^{jw}) = X^*(e^{jw})/[\,|X(e^{jw})|^2 + \beta\,] \qquad (3.12)$$

This is the expression for the compensating function with the '*' denoting the complex conjugate.

Substituting for $C(e^{jw})$ in equation (3.1),

$$H_e(e^{jw}) = Y(e^{jw}) \cdot X^*(e^{jw}) / [\,|X(e^{jw})|^2 + \beta\,] \qquad (3.13)$$

where β is the optimization parameter whose value has to be decided for an optimum solution for h(n).

3.2 APPLICATION OF METHOD 1

In this section an application of the optimum compensation technique on the design of inverse filters is illustrated. All computations in the frequency domain are carried out using FFT's. Therefore, equation (3.13) takes the form,

$$H_e(k) = Y(k) \cdot X^*(k) / [\,|X(k)|^2 + \beta\,] \qquad (3.14)$$

for k=1,...,N, where N > the length of y(n). For the convolution model described in chapter one, X(k) should be replaced by the DFT of the basic wavelet $W_b(k)$ and Y(k) should be replaced by the DFT of the reflector series for a single echo $R_s(k)$.

3.2.1 EXAMPLES

Three basic wavelets with different frequency response characteristics have been selected. Since a spike filter is to be designed, the desired output will be a unit sample of appropriate delay. Hence, $R_s(k) = 1.0$.

1. A low pass signal - $w_b(n) = n \cdot e^{-An}$, $n = 0,1,\ldots,45$; $A = 0.2$

2. A narrow band signal - $w_b(n) = \sin(2\pi n/A) \cdot \cos(2\pi n/B)$; $n = 0,1,\ldots,25$; $A = 50$, $B = 5$

3. A wide band signal - $w_b(n) = A \cdot n$, $n = 0,1,\ldots,3$; $= A \cdot (7-n)$, $n = 4,\ldots 7$; $A = 1/3$

A small amount of uniform random noise is added to all the waveforms so that they represent noise contaminated signals.

3.2.2 EFFECT OF THE PARAMETER

An application of the above technique first involves the investigation of a selection criterion for the parameter β. Although the introduction of β minimizes the effects of small spectral values in $|X(k)|$, it has an

adverse effect on the accuracy of the solution. Selection criterion for the value of β depends on the amount of tradeoff made between the accuracy and the noise in the solution. As β is decreased, noise in the solution is increased since the compensating effect diminishes. As it is increased, the solution gets oversmoothed, producing a dampening effect on the solution. These effects of β are clearly seen from figs. 3.1(a),(b) and (c) which display the unit sample response computed using example 1 for $\beta = 1.0$, 0.01 and 0.001 respectively. Fig. 3.1(c) with the smallest value of β has the most noise while fig. 3.1(a) with the highest value of β has the least noise. It can also be seen from fig. 3.1(c) to fig. 3.1(a) that the significant values of the signal around $n = 60$ acquire smaller amplitudes as β is increased.

3.2.3 OPTIMIZATION CRITERION

The accuracy of the result and the noise produced in the result are two good indicators that could be used in deciding a reasonable value for β which in turn produces a satisfactory inverse filter. Since an apriori knowledge of the solution may not be available on many occasions, the true value of the result is unknown. Hence, noise and accuracy have to be measured using the deconvolution result itself.

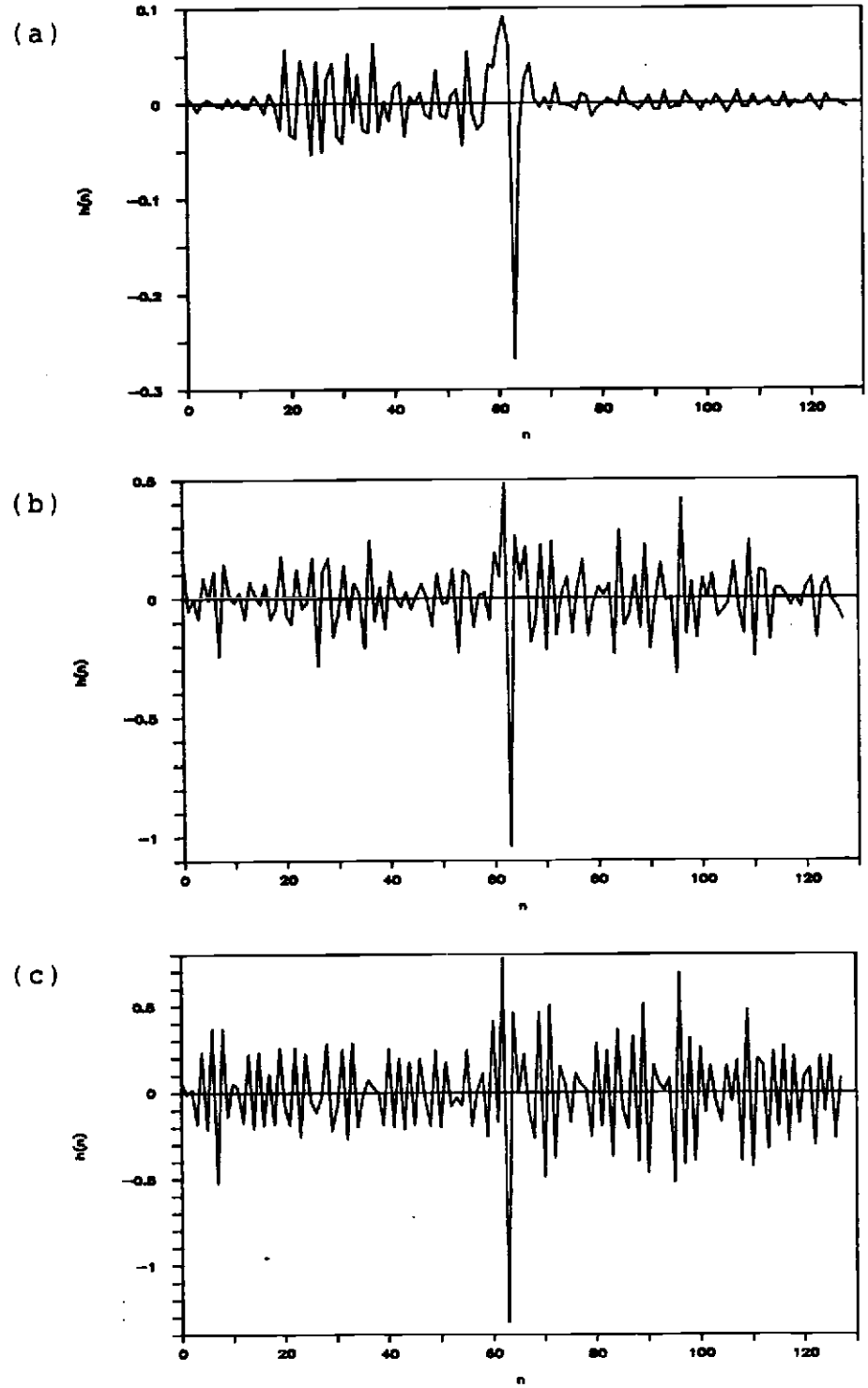

fig. 3.1 Unit Sample Response of Inverse Filter [h(n)]

(a) β = 1.0 (0 dB)
(b) β = 0.01 (-20 dB)
(c) β = 0.0001 (-40dB)

The second difference of h(n) given by,

$$N = \sqrt{\left[\sum_{i=2}^{M-2} [h(i-1)-2\cdot h(i)+h(i+1)]^2\right]} \quad (3.15)$$

can be used as a measure of the smoothness. For a noisy solution N would be high and as the solution gets smoother N would tend to zero. N is a good indicator of noise in this instance since it only depicts the jaggedness of the solution and is not affected by the errors produced by bias and accuracy.

The area under the waveform of the unit sample response of the filter is taken as the measure for accuracy. Hence it is given by,

$$A = \sum_{i=1}^{M} h(i) \quad (3.16)$$

As β is increased the solution gets damped and the area under the curve decreases.

Plots of noise(N) and accuracy(A) versus β for the 3 examples are shown in figs. 3.2(a),(b) and (c). It is clearly indicated in all cases that both noise and accuracy decrease as the value of β is increased. The net area of the graph of h(n) is negative in the case of the narrow band signal. Nevertheless, it can be observed that its absolute value decreases with increase in β although at a

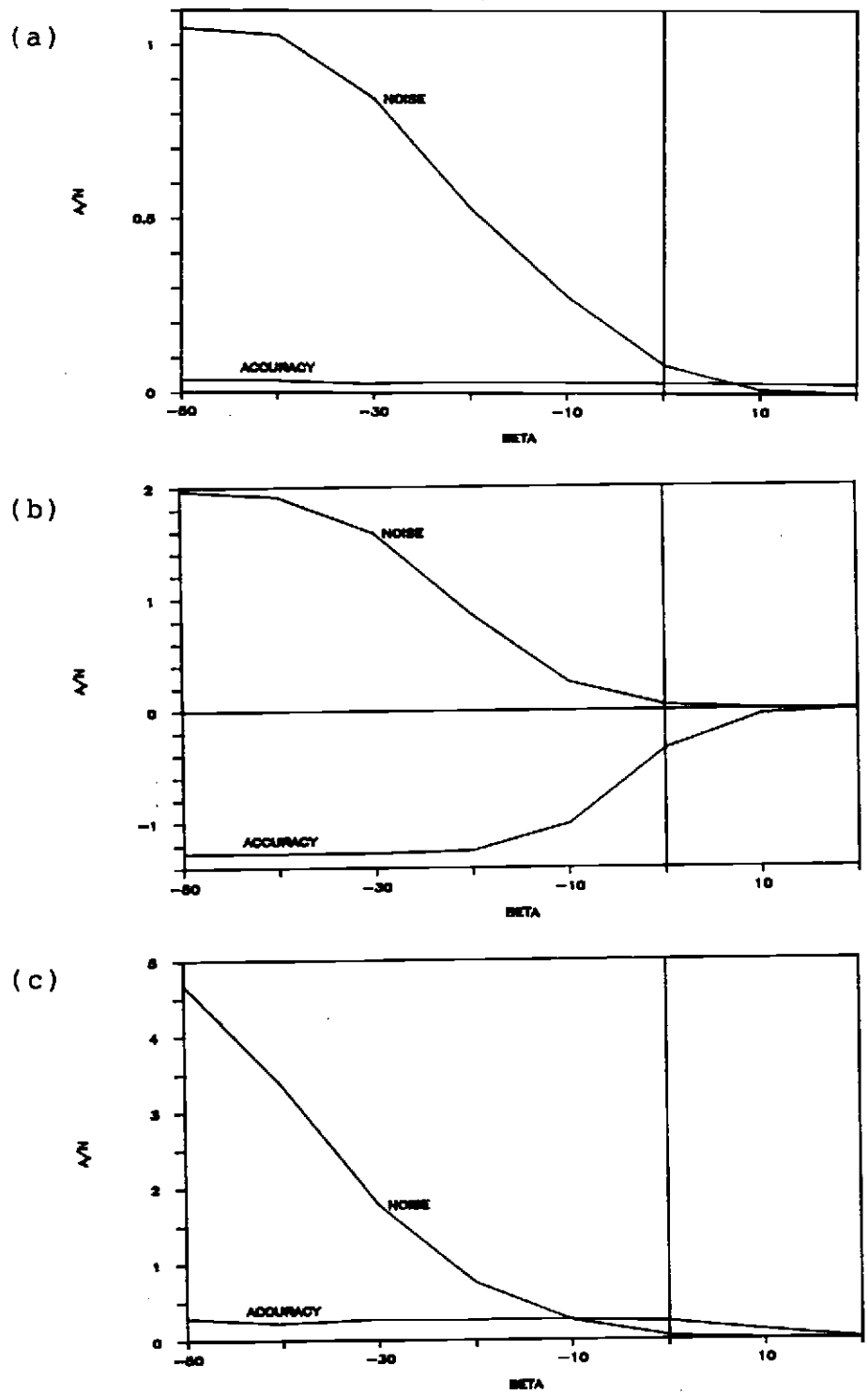

fig. 3.2 Graphs of Accuracy and Noise Vs. β for h(n)

(a) Using the Low Pass Basic Wavelet.
(b) Using the Narrow Band Basic Wavelet.
(c) Using the Wide Band Basic Wavelet.

slower rate than that of noise. These effects can also be observed in the output y(n), for a single pulse of the basic wavelet at the input which is computed using the above estimate for h(n). It should also be noted that, unlike h(n), the true value of the output y(n), is known. Consequently, its deviation from this actual waveform could be calculated in the form of the mean square error(mse).

In the resolution of multiple overlapped signals, it is important that the output from an inverse filter be free of noise and maintain a good S/N ratio for unambiguous detection of echoes. Thus, it is more relevant to use the output waveform rather than the unit sample response of the designed filter to measure its performance. Plots similar to that of h(n) are shown for y(n) in figs. 3.3(a),(b) and (c). The effects of noise and accuracy with variation in β show more promising results. Here again, the relative rate of decrease is different for the two indicators. This can be explained because the regions where $|X(k)|$ is very small contribute more to deconvolution noise than to the information content in the signal. The smaller values of β have a greater influence in these regions than in other regions where $|X(k)|$ is significant. Therefore for small values of β, the accuracy(A) will have a slower rate of

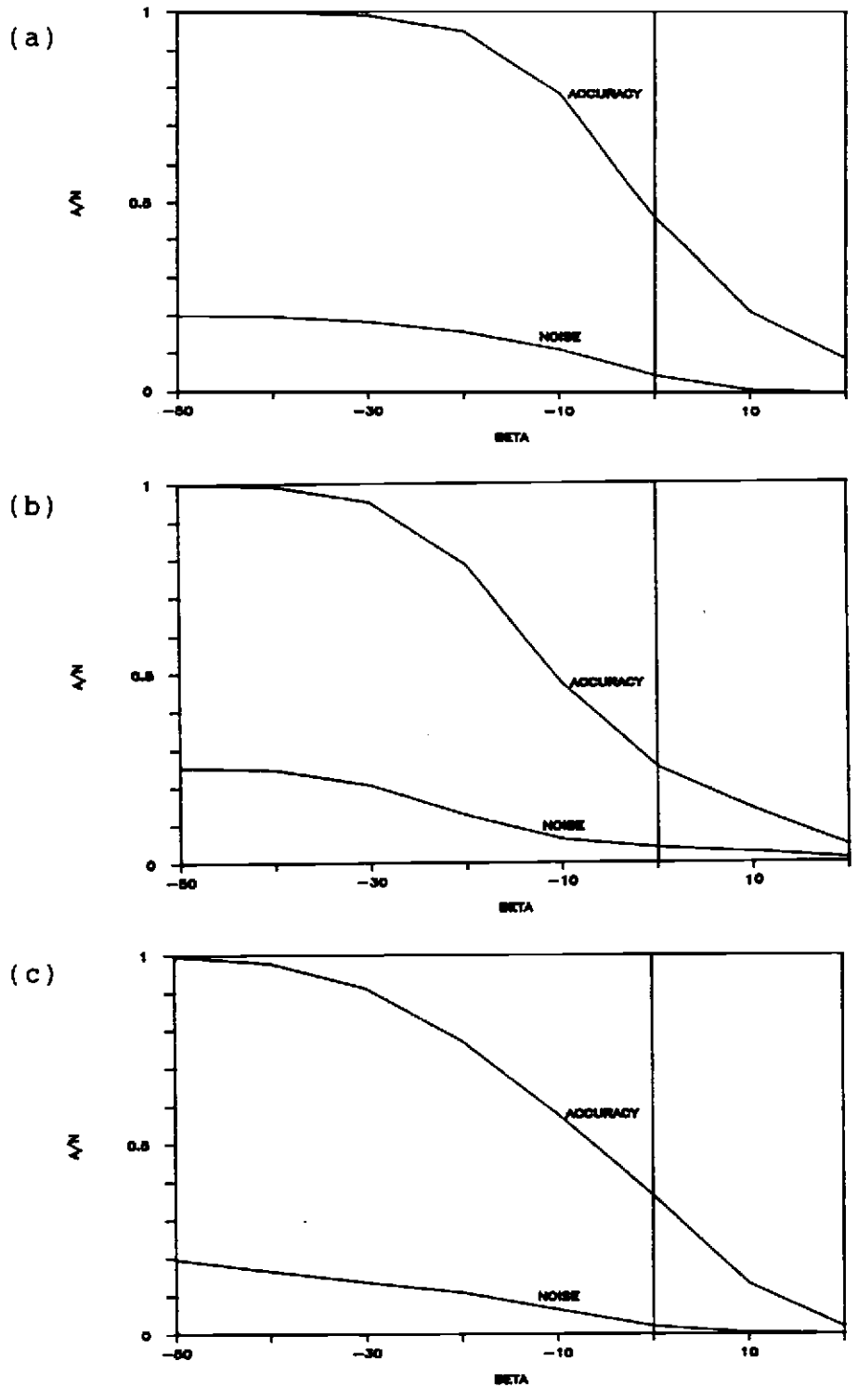

fig. 3.3 Graphs of Accuracy and Noise Vs. β for y(n)

(a) Using the Low pass Basic Wavelet.
(b) Using the Narrow Band Basic Wavelet.
(c) Using the Wide Band Basic Wavelet.

decrease compared to that of noise(N). On the other hand larger values of β can contribute to an over correction dampening the actual value and resulting in error in the information content.

A reasonably good operating region can be found in figs. 3.3(a) through (c) where the noise is much reduced while the accuracy is still maintained close to that of the true value. Analysing fig. 3.3(a), the region of operation for this particular example would be,

$$-10 \text{ dB} < \beta < 0 \text{ dB} \qquad (3.17)$$

When the filter is to be used for the detection of multiple echoes, an important factor would be the ability to distinguish between noise peaks and signal peaks. For example, if there are considerably high noise peaks, these could be mistaken for echos of smaller amplitudes. An appropriate measure for detecting this effect would be a S/N ratio defined as follows,

$$R = \frac{\text{signal peak}}{\text{highest noise peak}} \qquad (3.18)$$

It should be noted that a high R indicates only that the signal amplitude is high compared to that of noise; but it does not necessarily indicate a good accuracy for the signal. Therefore, all three measures should be consulted before reaching a decision on a satisfactory value for β. A comparison of R for the two methods described in this chapter are given in section 3.4.

3.2.4 DESIGN OF THE INVERSE FILTER

Now that an optimum region has been established, the application of this method for different types of basic wavelets is observed by implementing examples 1, 2 and 3 as mentioned earlier. The lowpass signal and its magnitude response are shown in fig. 3.4. The frequency response of the filter is determined by equation (3.13) with a value of $\beta = 1.0$ (0 dB) which falls within the region of operation. The magnitude response, unit sample response and the output of the designed filter are given in figs. 3.5(a),(b) and (c) respectively.

Fig. 3.6 shows the narrow band wavelet given in example 2 and its magnitude response. The optimum compensation filter designed for this basic wavelet and its unit sample response are displayed in figs. 3.7(a) and (b). The output of the designed filter for a single echo is shown in fig. 3.7(c).

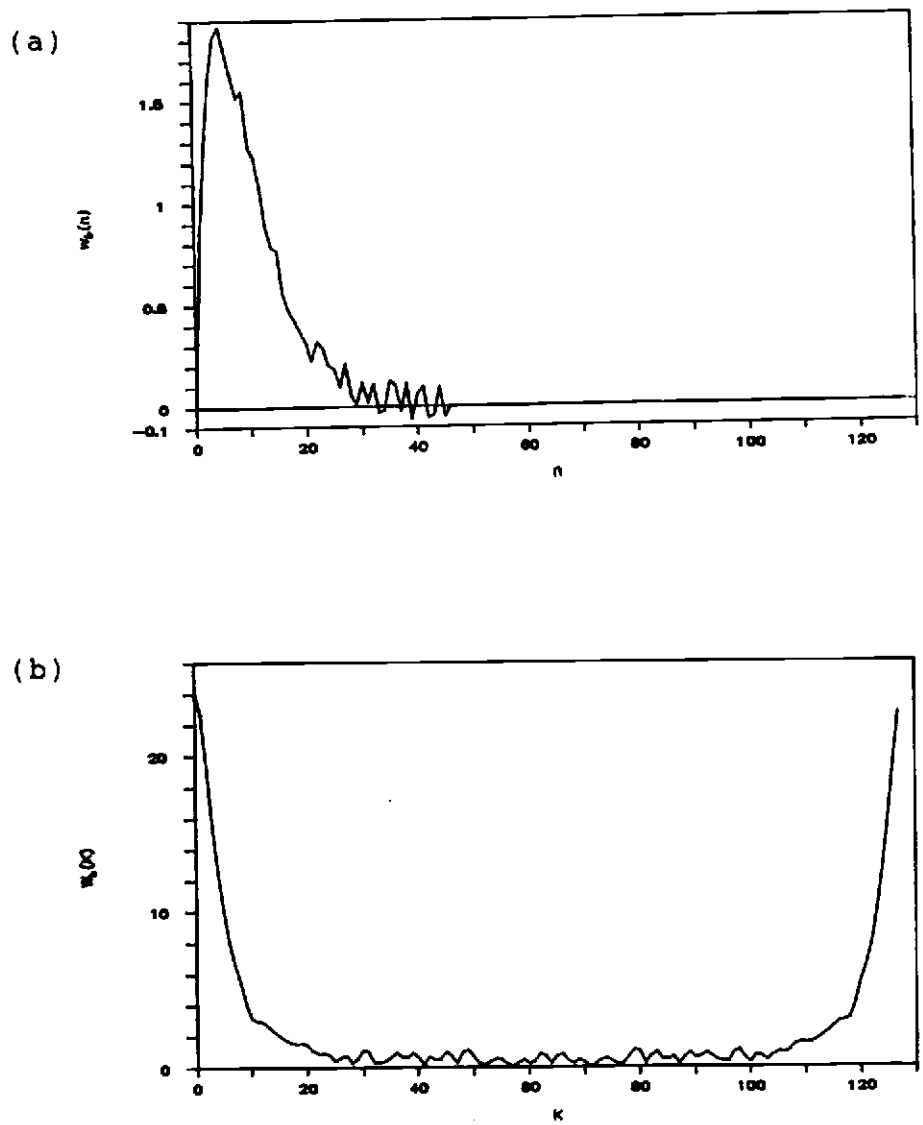

fig. 3.4 Example 1

(a) The Low Pass Basic Wavelet.
(b) Magnitude Response of Low Pass Wavelet.

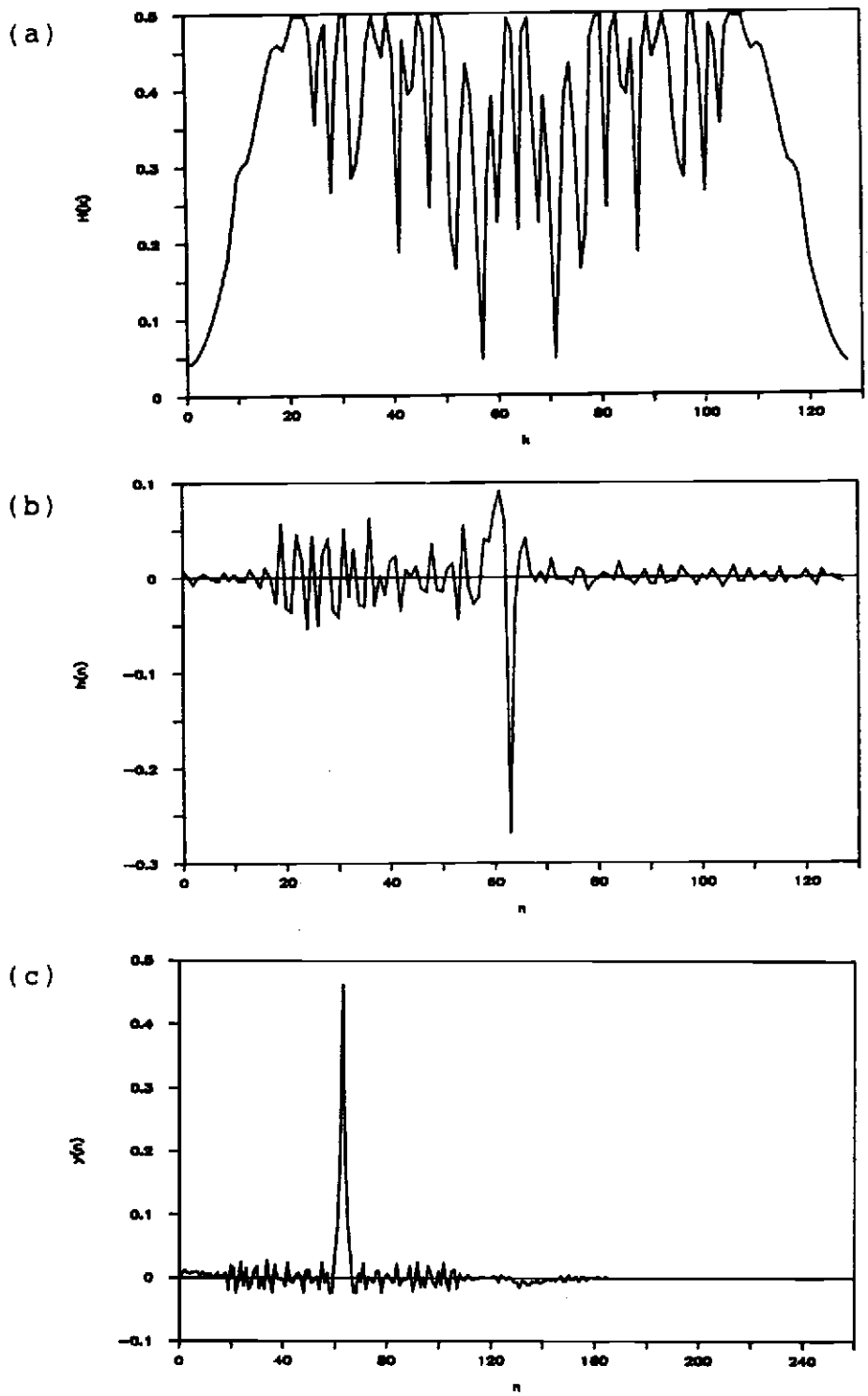

fig. 3.5 Inverse Filter for Example 1

(a) Magnitude Response.
(b) Unit Sample Response.
(c) Output for a single echo.

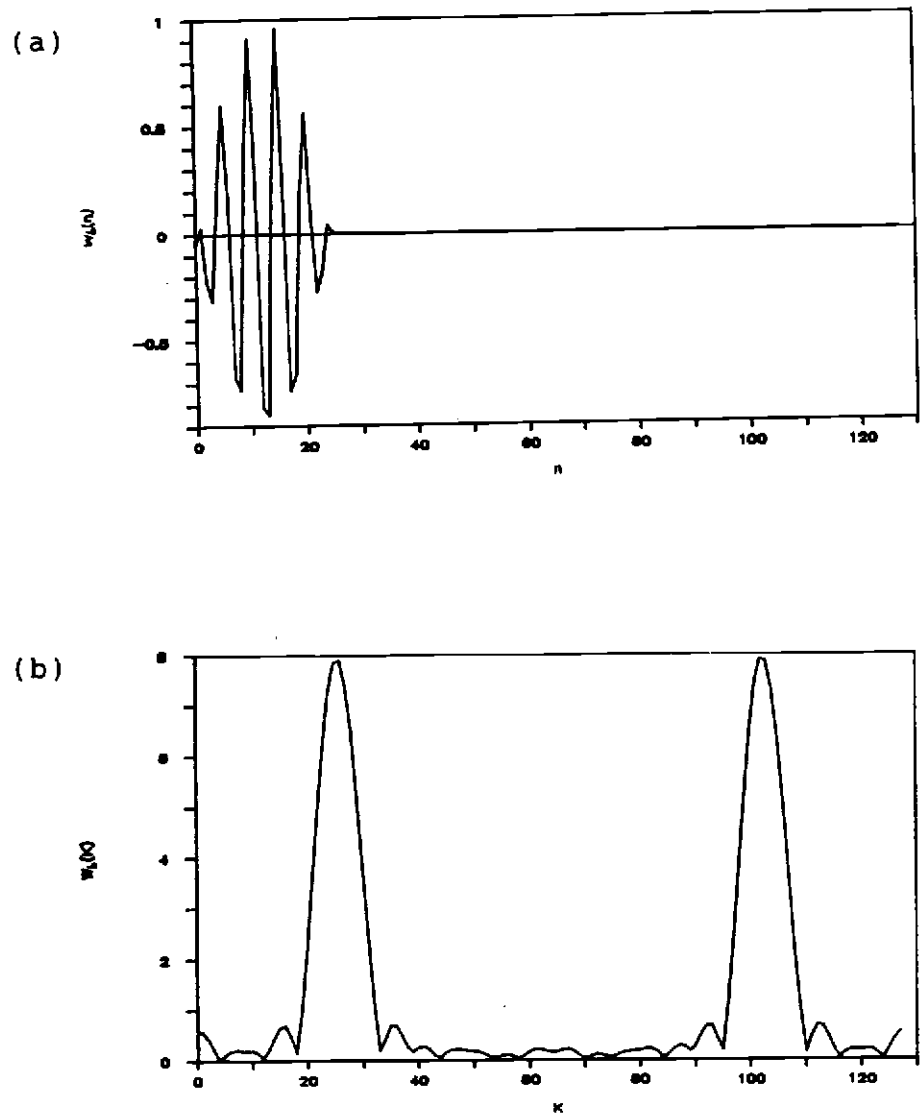

fig. 3.6 Example 2

(a) The Narrow Band Basic Wavelet.
(b) Magnitude Response of Narrow Band Wavelet.

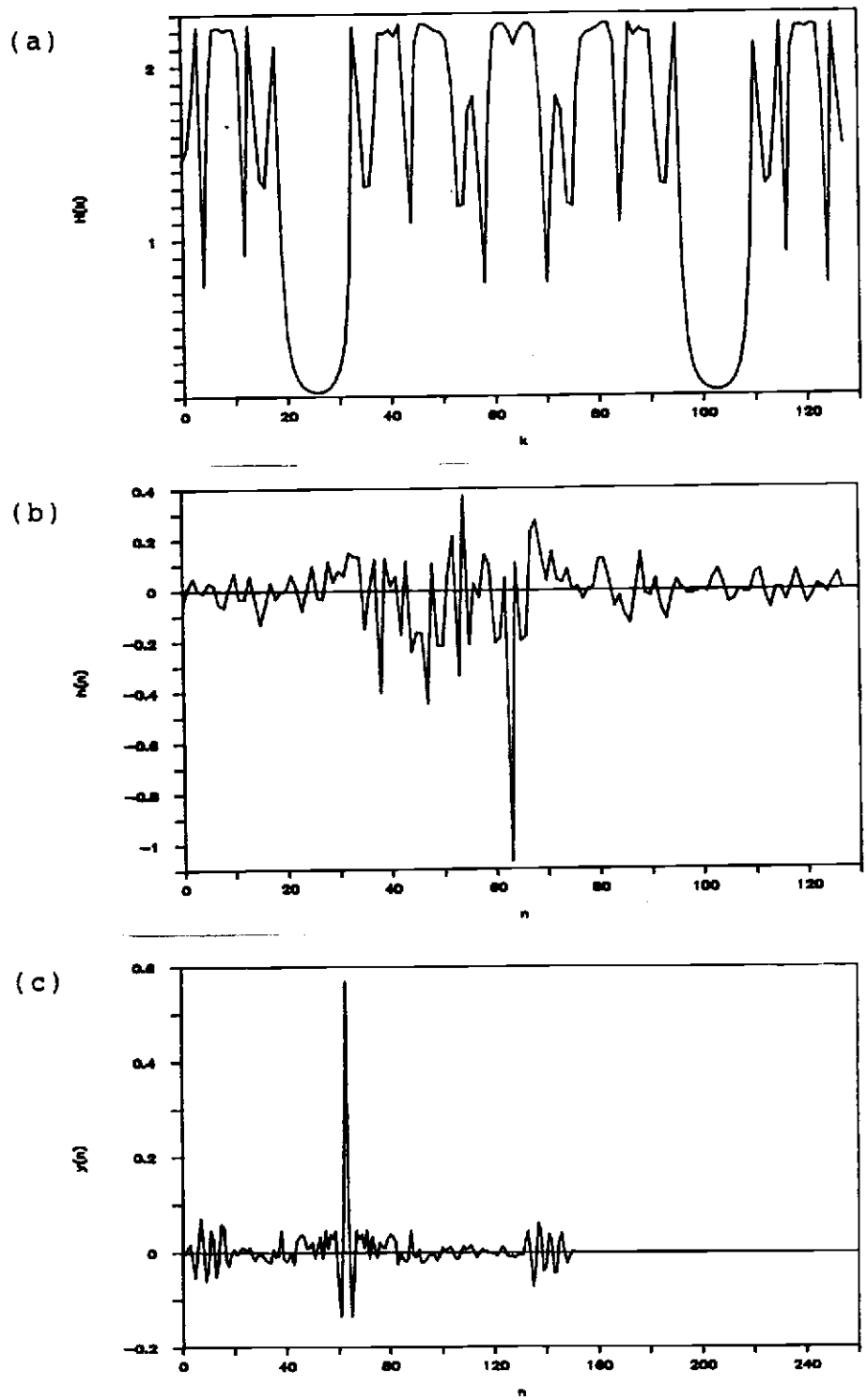

fig. 3.7 Inverse Filter for Example 2.

(a) Magnitude Response.
(b) Unit Sample Response.
(c) Output for a single echo.

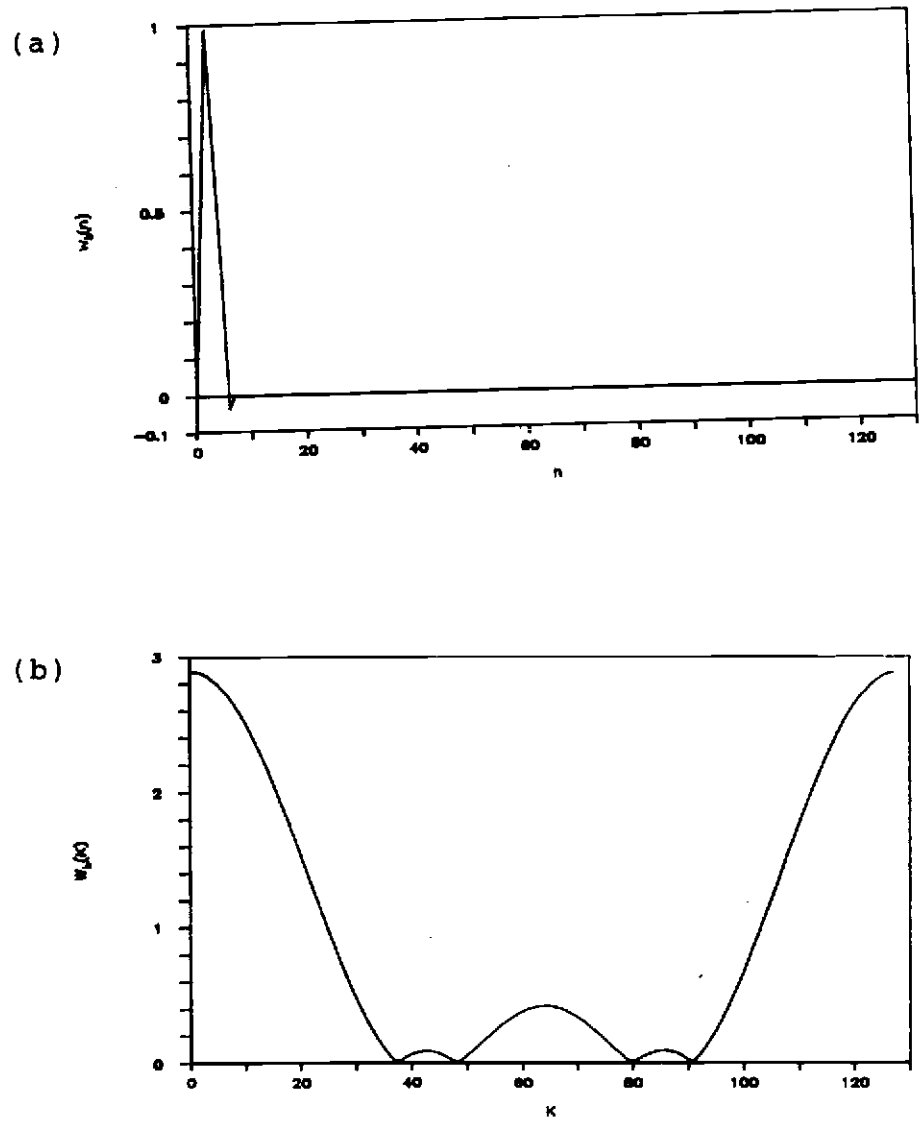

fig. 3.8 Example 3

(a) The Wide Band Basic Wavelet.
(b) Magnitude Response of Wide Band Wavelet.

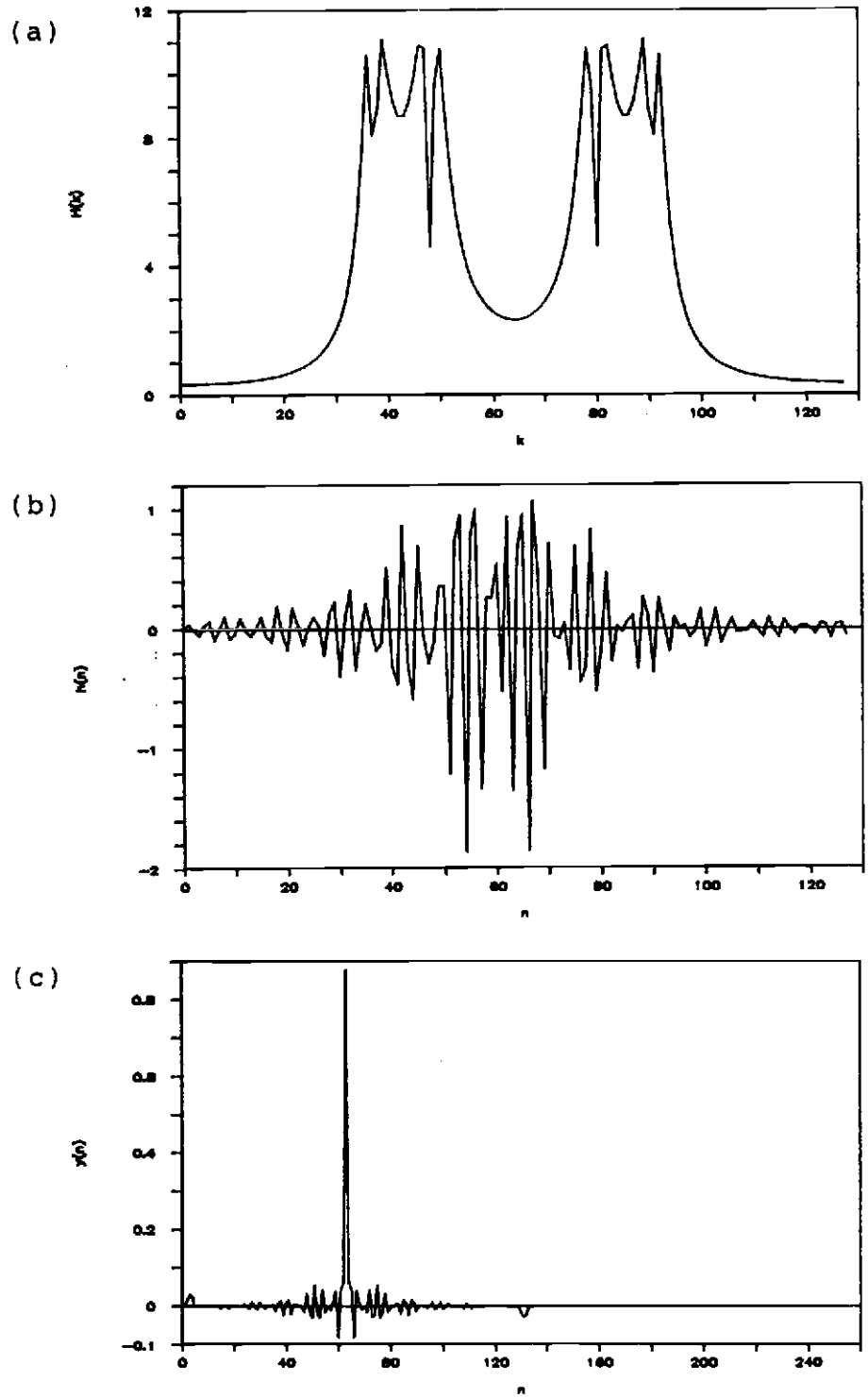

fig. 3.9 Inverse Filter for Example 3.

(a) Magnitude Response.
(b) Unit Sample Response.
(c) Output for a single echo.

Similar plots are obtained for example 3. The basic wavelet and its magnitude response are shown in fig. 3.8. The filter responses and the output are shown in figs. 3.9(a),(b) and (c).

As seen by fig. 3.4(b), the low pass signal has a number of singularities in its magnitude response. As a result fig. 3.5(a) shows considerable noise in the bandwidth corresponding to the region of zero crossings in fig. 3.4(b). The output produced by this filter has lost its accuracy by about 50% as a result of establishing a compromise between accuracy and noise content.

The narrow band signal also displays a number of singularities in its magnitude response, resulting in a noisy filter response. Here again, noise at the output has been maintained at a reasonable amount at the expense of accuracy. The output also displays more ringing around the main pulse.

The wide band signal is the same as the low pass except that it has a wider bandwidth covering most of the spectrum. Consequently, the filter magnitude response does not indicate as much noise as in the previous examples. This output has the best accuracy, almost 90% of the true value, while maintaining the noise reasonably low.

3.3 METHOD 2 - LEAST SQUARES REGRESSION METHOD

In the previously decribed method, the frequency domain formula was derived with the use of design criteria defined on the frequency domain, and no constraints were placed on the behaviour of the solution in the time domain. In contrast, the frequency domain equation developed by Hunt [5] starts its derivation in the time domain thus enabling the use of constraints for the time domain version of the solution.

Although ideally,

$$y(n) = x(n) * h(n) \tag{3.19}$$

there is always a basic uncertainty in the result caused by the errors in the process. As such,

$$y(n) = x(n) * h(n) + e(n) \tag{3.20}$$

or,

$$y(n) = \Sigma\, h(k) \cdot x(n-k) + e(n) \tag{3.21}$$

If the length of sequences x(n) and h(n) are p and q respectively, the length of y(n) will be N = p+q-1.

In matrix form this equation can be written as,

$$[Y] = [X]\cdot[H] + [E] \qquad (3.22)$$

where [Y],[H] and [E] are column matrices of Nx1,qx1 and Nx1 holding the corresponding sequences. [X] is the Nxq matrix given by,

$$[X]_{kj} = x(k-j), \quad 0 < k-j < p-1 \qquad (3.23)$$
$$= 0 \quad , \quad \text{otherwise}$$
$$\text{for } k = 0,\ldots,N-1 \text{ and } j = 0,\ldots,q-1$$

As seen in chapter (1), one major problem of deconvolution is that the problem can be ill conditioned, leading to noisy solutions. Hence, it is appropriate to minimize the second difference of the solution in order to maintain smoothness. The sum of the squares of the second difference for the sequence h(n) is given by,

$$S = \sum_{i=2}^{N-2} [h(i-1)-2\ h(i)+h(i+1)]^2 \qquad (3.24)$$

In matrix form,

$$S = [CH]'[CH] \qquad (3.25)$$

where C =
$$\begin{matrix} 1 & -2 & 1 & 0 & 0 & . & . & . \\ 0 & 1 & -2 & 1 & 0 & 0 & . & . & . \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 & . & . & . \\ 0 & . & . & . \\ . & . \end{matrix}$$

S can also be written as a convolution expression by,

$$S = [c*h]'[c*h] \qquad (3.26)$$

where the sequence c(n) = 1,-2,1,0,0...0

Equation (3.22) shows that there are more equations(N) than unknowns(q) in the solution of this problem. Therefore a least squares problem in constrained regression can be formulated as follows,

$$\text{Minimize} \quad [[C][H]]'[[C][H]], \qquad (3.27)$$

$$\text{Subject to } [[X][H] - [Y]]'[[X][H] - [Y]] = [E]'[E] \qquad (3.28)$$

The problem is to minimize the second difference while constraining the error to a prespecified quantity E.

The solution to this problem using Lagrange multipliers is,

$$[H] = [[X]'[X] + \Gamma[C]'[C]][X]'[Y] \quad (3.29)$$

where $\Gamma = 1/\beta$, β being the Lagrange multiplier. This could be easily computed in the frequency domain via FFT's. Rewriting (3.29) as,

$$[[X]'[X] + \Gamma[C]'[C]][H] = [X]'[Y] \quad (3.30)$$

enables the conversion to be performed easily. Taking Fourier Transforms on both sides of equation (3.30) produces,

$$[X^*(k)\cdot X(k) + \Gamma\cdot C^*(k)\cdot C(k)]H_e(k) = X^*(k)\cdot Y(k) \quad (3.31)$$

which reduces to,

$$H_e(k) = X^*(k)\cdot Y(k)/[|X(k)|^2 + \Gamma|C(k)|^2] \quad (3.32)$$

$H_e(k)$ is computed for a particular Γ and then the residual error is computed using the estimate obtained for $h_e(n)$ as given by,

$$r = [y(n) - h_e(n)*x(n)]'[y(n) - h_e(n)*x(n)] \tag{3.33}$$

If $r > e$ (prespecified), Γ is decreased; and if $r < e$, Γ is increased and the iteration continues for a new value of Γ.

3.4 COMPARISON OF THE TWO METHODS

The same examples used on the optimum compensation technique is used here for comparison. Similar analysis is carried out as before and the results show that basically the parameter Γ has the same effect as β on deconvolution noise and accuracy. Figs. 3.10(a),(b) and (c) display the comparison of accuracy while figs. 3.11(a),(b) and (c) display the comparison of noise obtained by the two methods for examples 1,2 and 3. It is observed that the noise is less and decreases more rapidly in the regression method than in the optimal compensation method as expected by its derivation. The accuracy is also less in the regression method for smaller values of the parameters. However, for higher values of the parameters the accuracy in the regression method improves over the other method. In all three cases the region of operation was found to be in the area which includes the lesser values of the parameter from the crossover point. Therefore, the regression method seems to provide better noise immunity at the expense of lowering its accuracy. Figs. 3.12(a),(b) and (c) compares R for the two methods, again using the same examples.

Analysing the methods for different types of basic wavelets show that the more suitable method for low pass signals is the least squares regression method. This may be attributed to the fact that the compensating function

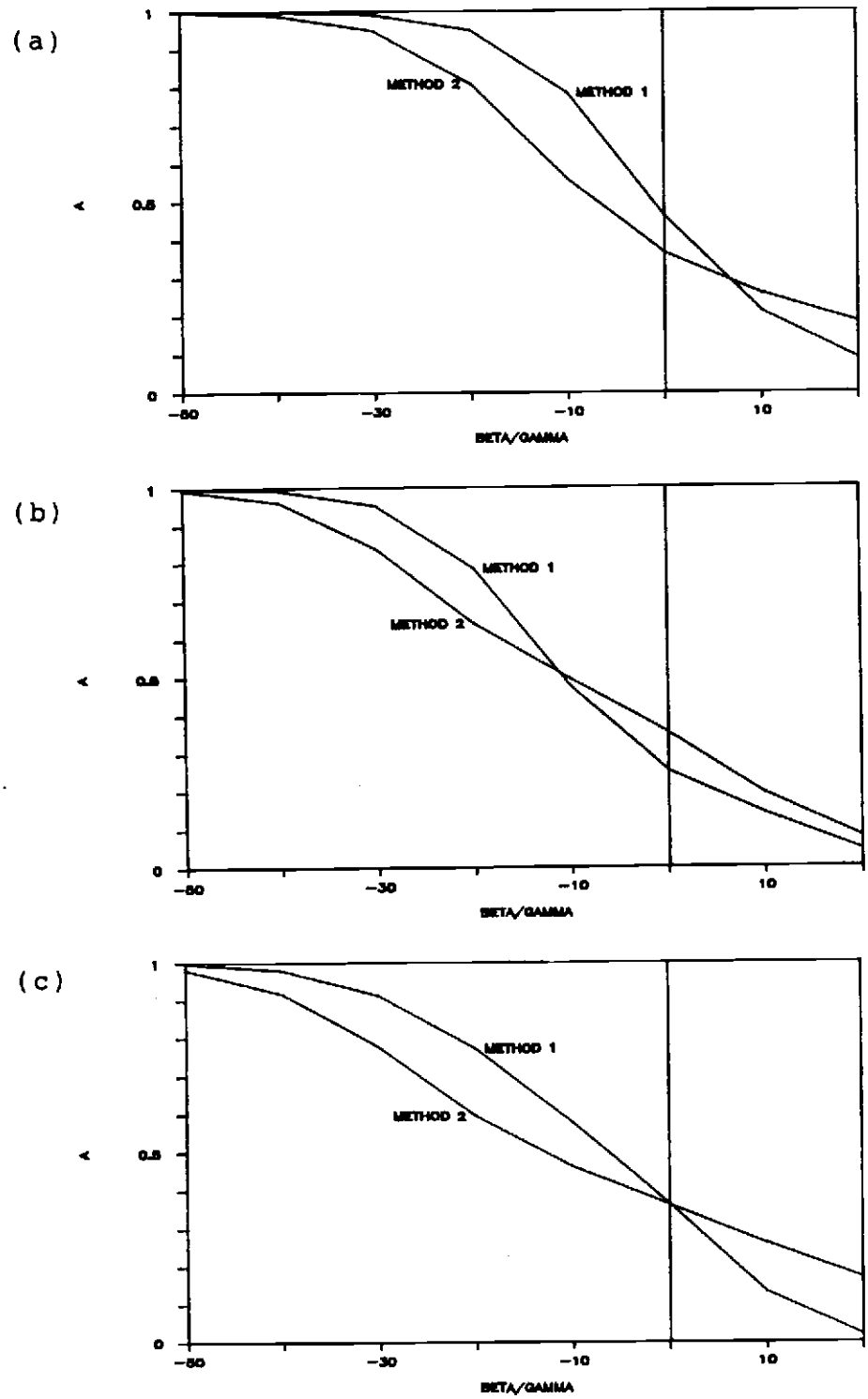

fig. 3.10 Accuracy Vs. $\beta(\Gamma)$ for Methods 1 and 2.

(a) Using the Low Pass Wavelet.
(b) Using the Narrow Band Wavelet.
(c) Using the Wide Band Wavelet.

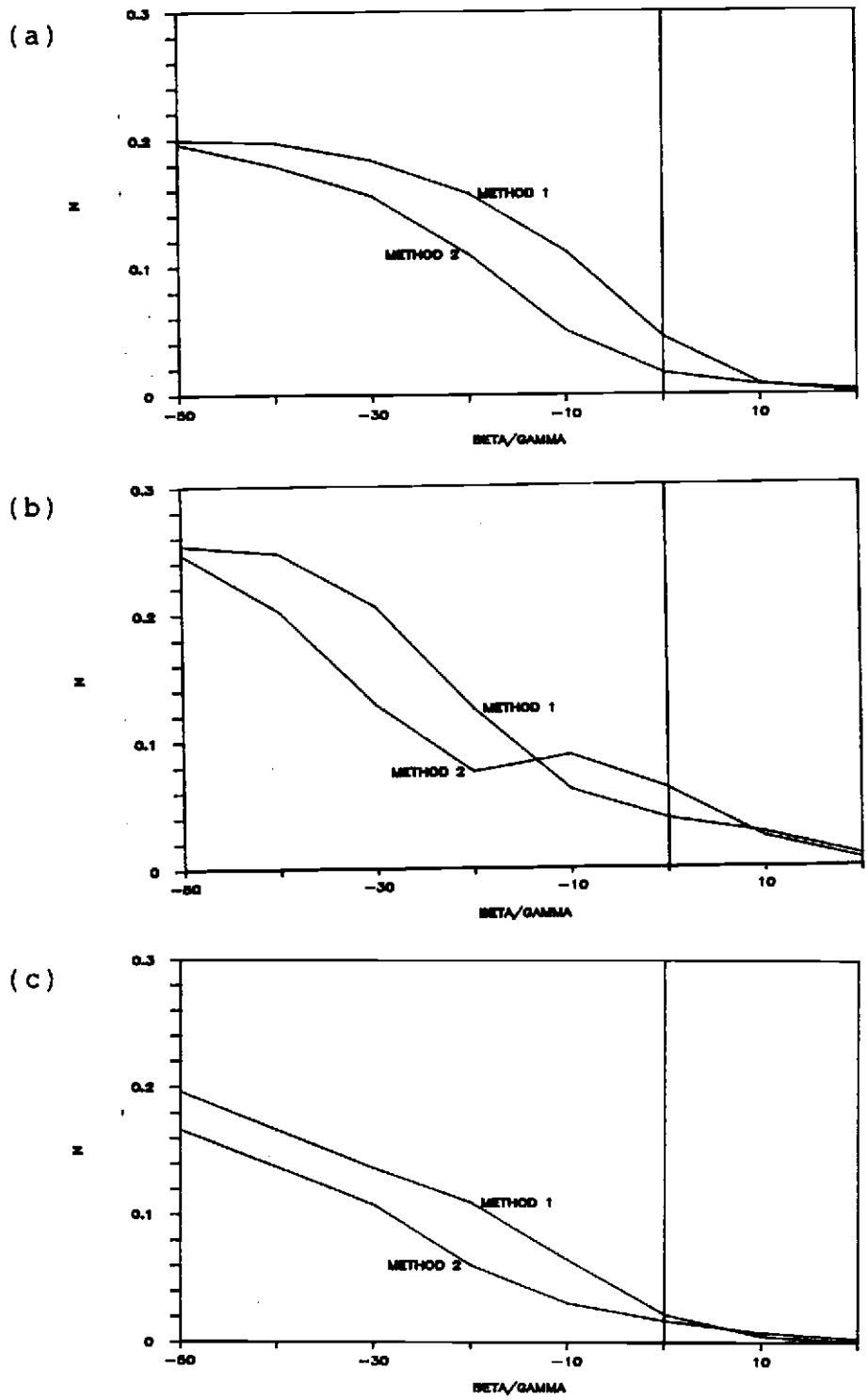

fig. 3.11 Noise Vs $\beta(\Gamma)$ for methods 1 and 2.

(a) Using the Low Pass Wavelet.
(b) Using the Narrow Band Wavelet.
(c) Using the Wide Band Wavelet.

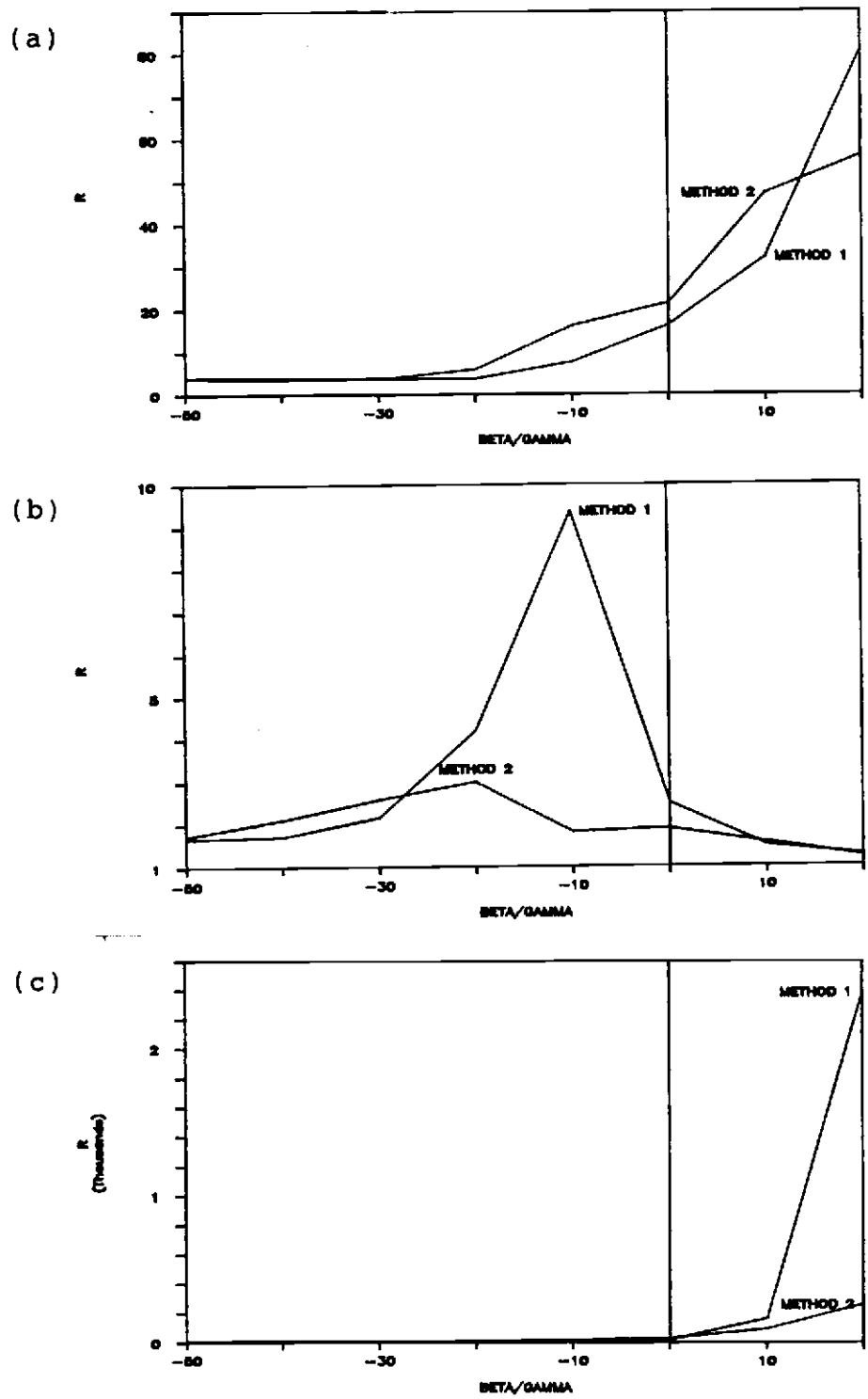

fig. 3.12 R Vs. β(Γ) for methods 1 and 2.

(a) Using the Low Pass Wavelet.
(b) Using the Narrow Band Wavelet.
(c) Using the Wide Band Wavelet.

$\Gamma \cdot |C(k)|^2$ contributes more for smaller values of $|W_b(k)|$ and less for larger values of $|W_b(k)|$. For the narrow band signal, both methods provide equal noise with the regression method providing better accuracy. An interesting observation in fig. 3.12(b) reveals that the maximum noise peak in the output derived from the regression method is higher than that of the compensation method. A further investigation showed that the noise peak was due to aliasing. Therefore, the compensation method is better at reducing aliasing in narrow band signals. The wide band wavelet benefits by both methods to almost the same extent with the regression method producing slightly better noise immunity over the other method.

3.5 REDUCTION OF ALIASING USING THE TWO METHODS

Another observation in this study was that the output displayed fairly significant values at the two extremes of the graph for low values of the parameters (see fig. 3.13(a). This happens when the value of β or Γ is not large enough to pull the poles of H(z) away from the unit circle. Hence, the filter lengths tend to become very long causing an aliased version to be computed by this procedure. This is verified by implementing a circular convolution to arrive at the output. The noise like effects at the two extremes disappear as shown in

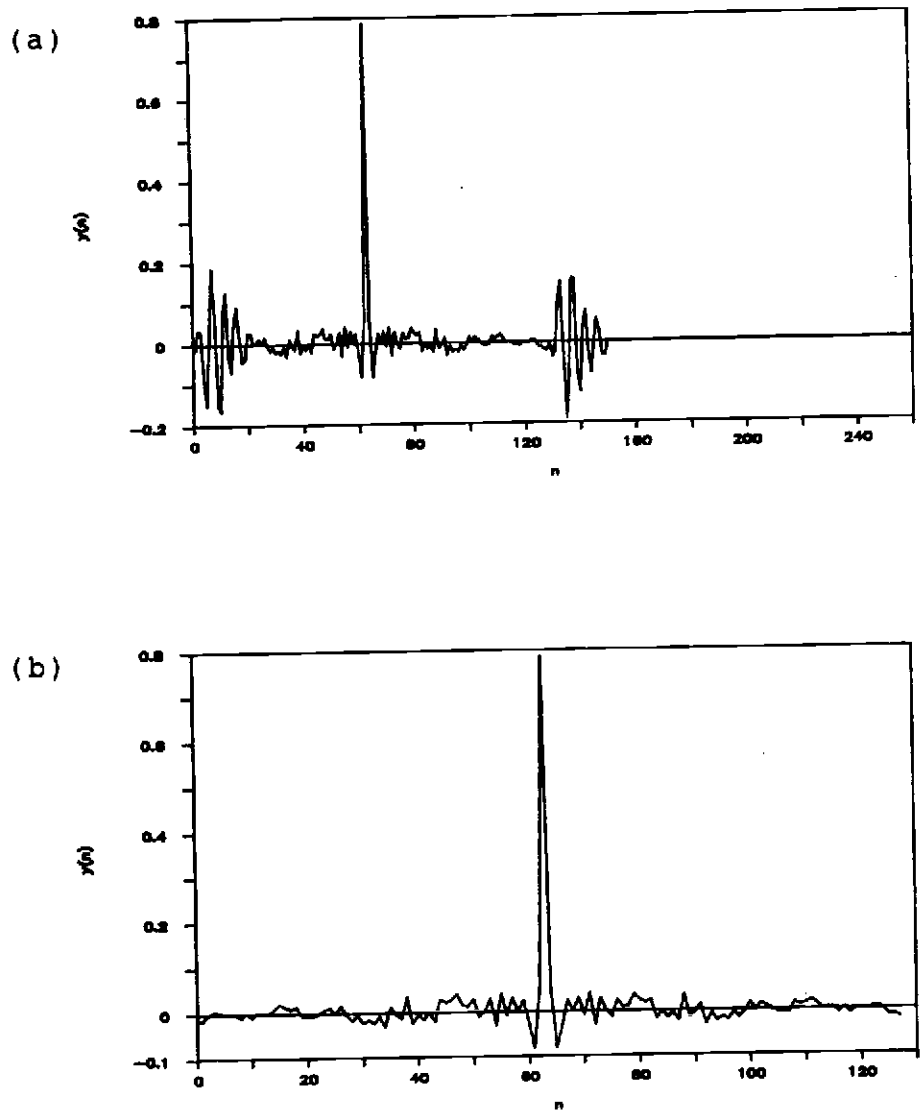

fig. 3.13 Output of Inverse Filter for Example 2 (β=0.01)

(a) By Linear convolution with Narrow Band Wavelet.
(b) By Circular Convolution with Narrow Band Wavelet.

fig. 3.13(b). It is shown here for the narrow band signal since it is the type that is most susceptible to this phenomenon.

The ratio of the output pulse height to that of the peak at the extremes is most significant for this type of basic wavelet. Thus, it is seen that introduction of β(or Γ) is helpful not only in decreasing the noise content but also in preventing aliasing.

3.6 ADAPTIVE FILTERING

The deconvolution methods can be viewed as adaptive filtering processes on the true filter response. This is explained by rewriting equations (3.13) and (3.32) in sections 3.1 and 3.3 in the form,

$$H_e(k) = [Y(k)/X(k)]\cdot F(k) \qquad (3.34)$$

F(k) is the adaptive filter used for noise reduction in the deconvolution filters. In the case of the optimum compensation technique it will be given by,

$$F_c(k) = 1/[1 + \beta/|X(k)|^2] \qquad (3.35)$$

In the least squares regression method it takes the form,

$$F_r(k) = 1/[1 + \Gamma|C(k)|^2/|X(k)|^2] \quad (3.36)$$

Ideally, from equation (3.34), F(k) should be equal to unity at all frequencies. Thus, the inverse response f(n) should be a unit sample at the origin.

It can be seen from figs. 3.14(a) and (b) that for the low pass and narrow band signals the magnitude response of the adaptive filter used by method 1 takes the same form as that of the signals. The low pass signal uses a low pass filter while the narrow band signal uses a narrow band filter. The wide band signal produces an adaptive filter that has a magnitude response given by fig. 3.14(c). It is equal to 1 except in a small region corresponding to singularities in the magnitude response of the signal (compare with fig. 3.8(b)). The resulting unit sample response of the adaptive filter is the closest to the ideal case when compared with the results obtained by the other two signals (figs. 3.15(a),(b) and (c)).

Similar plots of F(k) and f(n) generated by method 2 using equation (3.36) are shown in figs. 3.16 and 3.17. Comparing with the results of method 1, it is seen that both techniques employ a low pass type filter in the

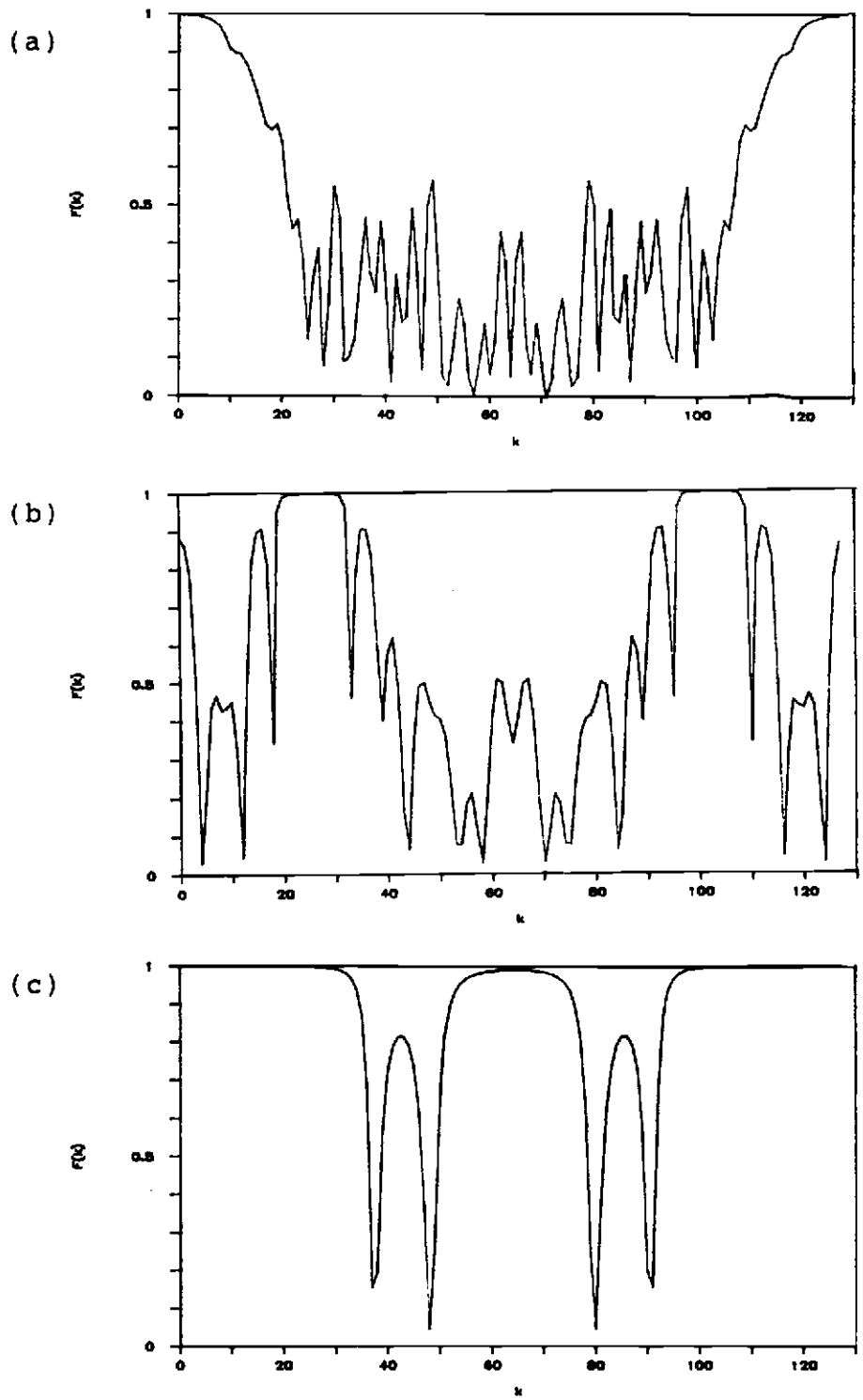

fig. 3.14 Magnitude Response of Adaptive Filter - Method 1

(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

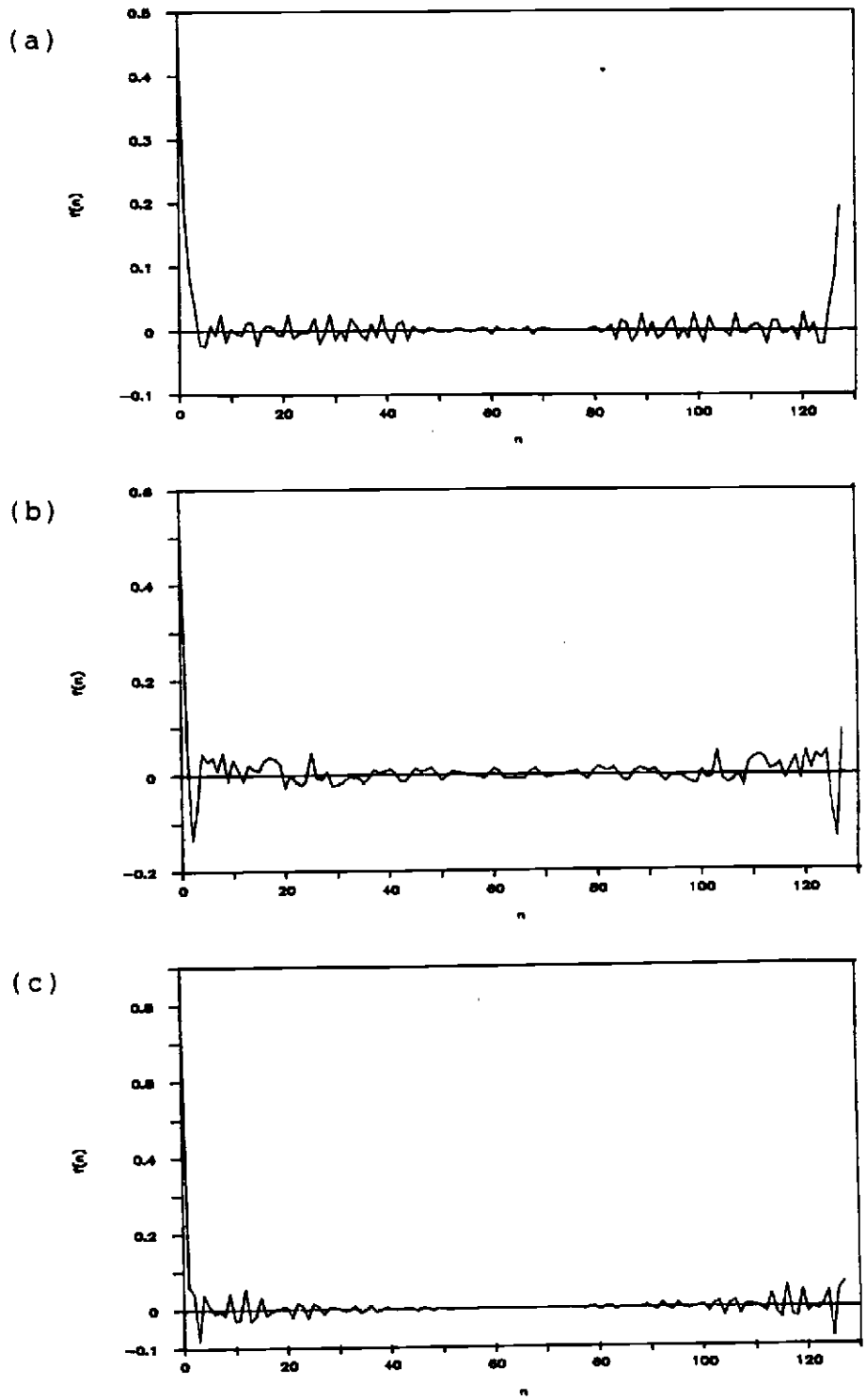

fig. 3.15 Unit Sample Resp. of Adaptive Filter - Method 1

(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

case of the low pass signal (compare with fig. 3.4(b)). It is also observed that the cutoff frequency of the two filters are the same as that of the magnitude response of the filter. In the case of the narrow band signal, the narrow band filter used in method 1 is replaced by a low pass filter in method 2 as shown in fig. 3.16(b). The low pass filter extends its cutoff frequency to the highest frequency of the narrow band signal magnitude response. The difference could be explained by rewriting a general equation for the adaptive filter as follows,

$$F(k) = 1/[1 + P(k)/|X(k)|^2] \qquad (3.37)$$

In the first method $P(k) = \beta$, which is a constant for all frequencies. For small values of $|X(k)|$ the denominator gets larger and $F(k)$ acquires a stop band. Since the filter has a stop band wherever $|X(k)|$ is small, it implies that the magnitude response of the adaptive filter and the signal are essentially of the same shape.

On the other hand, in the second method $P(k) = \Gamma|C(k)|^2$. The magnitude of $|C(k)|$ is largest at large values of k. Therefore, for higher values of k the stop bands in the signal would result in stop bands in the filter; but, stop bands at lower values of k would remain as pass bands since $|C(k)|$ also has a stop band in this region.

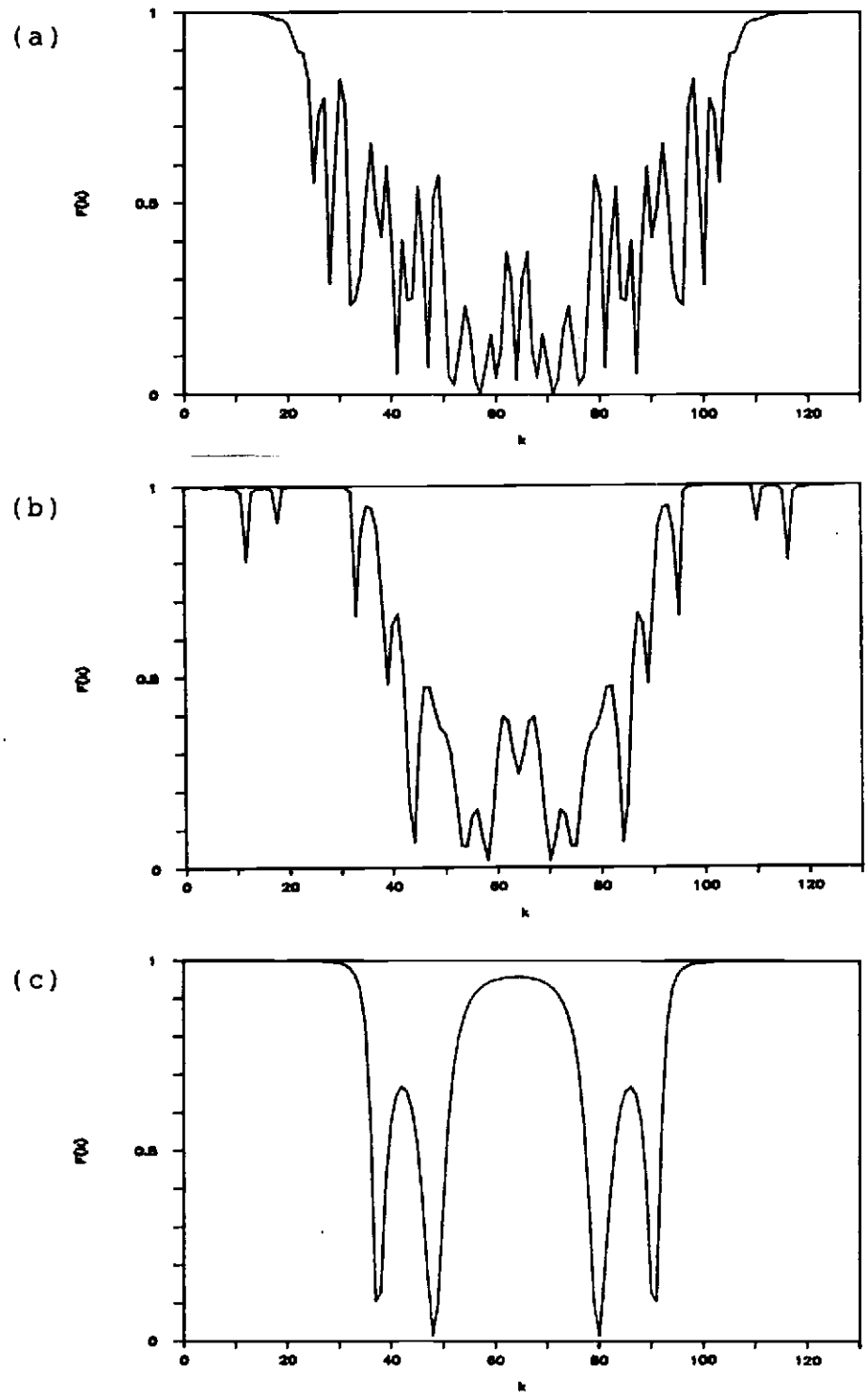

fig. 3.16 Magnitude Response of Adpative Filter - Method 2

(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

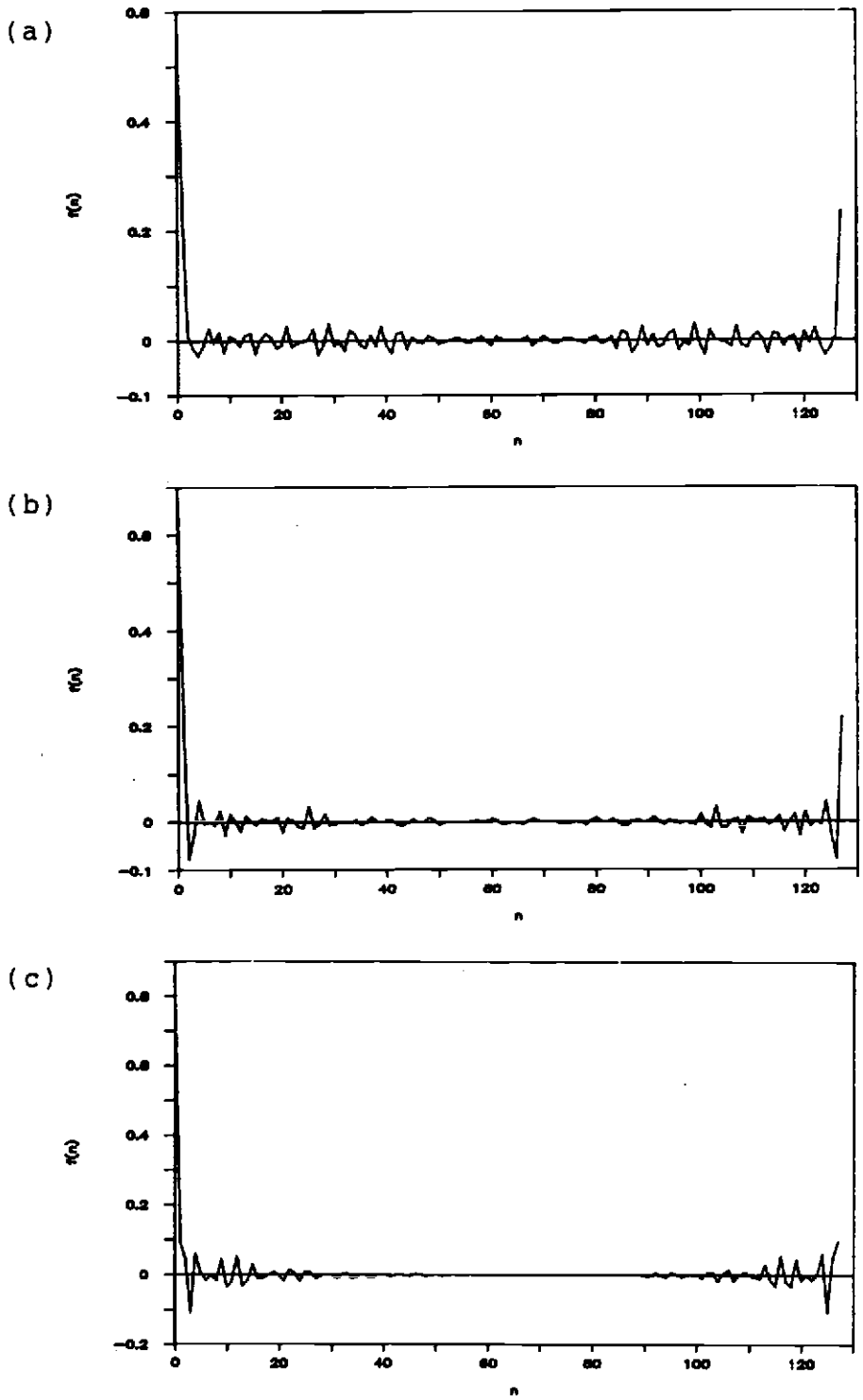

fig. 3.17 Unit Sample Resp. of Adaptive Filter - Method 2

(a) For the Low Pass Wavelet.
(b) For the Narrow Band Wavelet.
(c) For the Wide Band Wavelet.

In the case of the wide band signal both methods utilize the same type of filter, the only difference being the slight decrease in the magnitude of the middle lobe in the response from method 2. These differences between the two methods could be attributed to the fact that one uses a constant parameter while the other uses a parameter that increases with increasing values of frequency.

4.0 RESOLUTION OF MULTIPLE OVERLAPPED SIGNALS

This chapter illustrates the resolution of multiple overlapped signals using the filters designed in the previous chapter. In appropriate situations the use of Gaussian filters have been demonstrated in addition to the corresponding spike filters. The composite signal is analytically generated and represents a collection of 5 reflections with different amplitudes and arrival times. The amplitudes and arrival times used for low pass and narrow band signals are shown in Table 1. If the echoes of the composite signal for the wide band wavelet are to be indistinguishable before filtering, their arrival times should be more closely spaced than in the other two examples. Table 2 shows the amplitudes and arrival times for the composite used in this case.

Table 1 - Amplitudes and arrival times of echoes for figs. 4.1(a), 4.2(a) and 4.5(a)

arrival time	10	15	30	35	45
amplitude	0.9	0.5	-0.7	0.9	0.3

Table 2 - Amplitudes and arrival times of echoes for fig. 4.3(a)

arrival time	4	8	11	15	18
amplitude	-0.3	0.5	0.5	0.8	-0.4

4.1 THE LOW PASS WAVELET

The composite for the low pass wavelet is shown in fig. 4.1(a). It was shown in section 3.4 that method 2 provided better results over method 1 in the case of the low pass wavelet. Hence, the unit sample response of the filter designed in chapter 3 using method 2 is linearly convolved with the composite to arrive at the output pulse series. The pulse series shown in fig. 4.1(b) indicates the presence of a reflection with a spike, its position signifying the time of arrival and its amplitude signifying the strength. Although all 5 reflections are not clearly distinguishable in the composite signal, they can be easily identified in fig. 4.1(b) after filtering. It is observed that the peak noise level is kept down to around 10% of the maximum pulse.

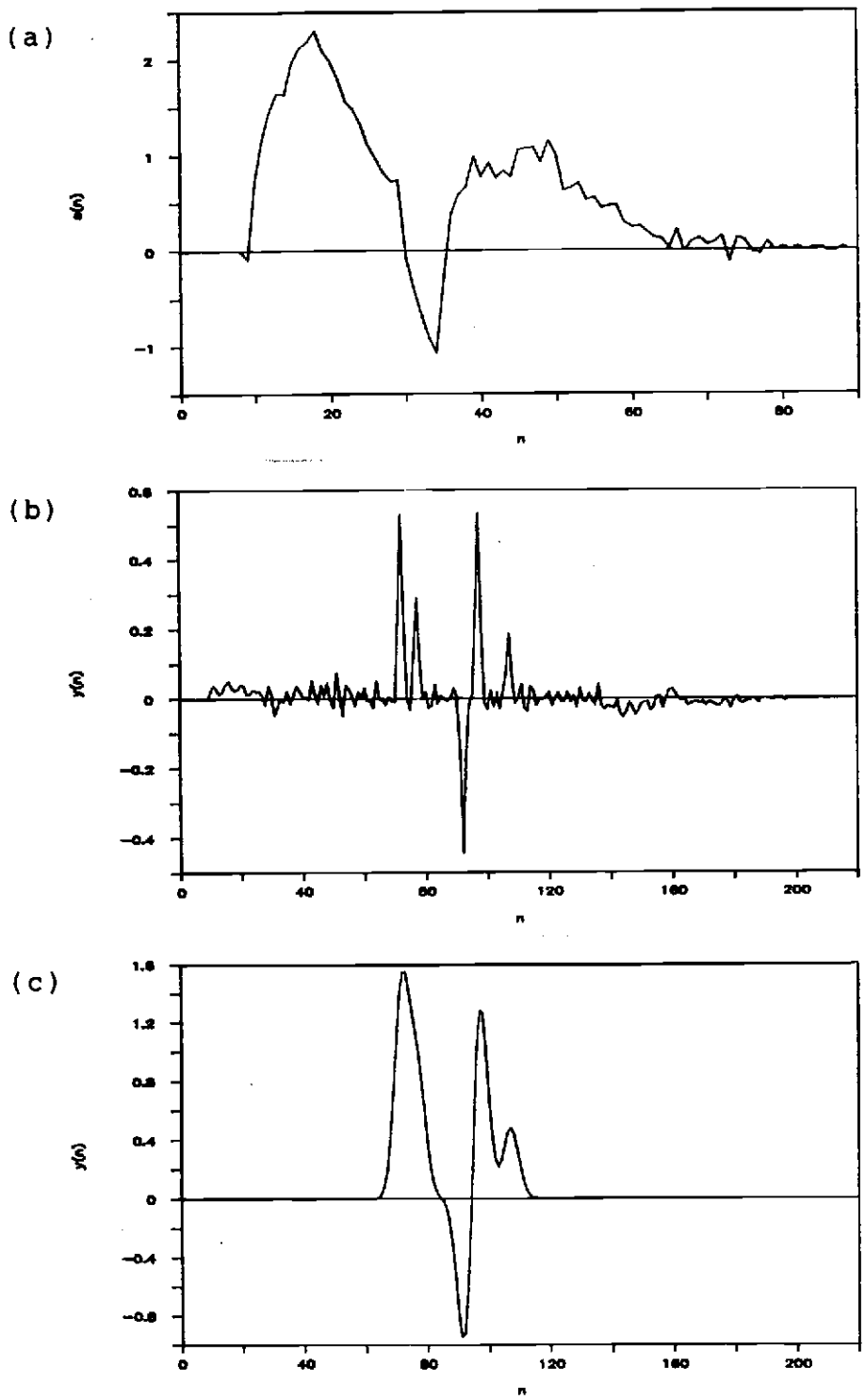

fig. 4.1 Resolution of Low Pass Wavelets

(a) Composite Signal
(b) Filtered Signal Using Spike Filter
(c) Filtered Signal Using Gaussian Filter

Reflections which have strengths below this are difficult to differentiate from noise. The noise and aliasing can be minimized by using an appropriate Gaussian filter. A Gaussian with $\sigma = 2.5$ could be used with the low pass signal since their bandwidths in the frequency domain are almost the same. Fig. 4.1(c) shows the corresponding Gaussian output. Although the noise has been taken care of in this case, the resolution is very poor compared to the output of the spike filter in fig. 4.1(b).

4.2 THE NARROW BAND WAVELET

The narrow band composite signal is shown in fig. 4.2(a). The resulting pulse series in fig. 4.2(b) is obtained via filtering using the unit sample response derived from method 1 (fig.(3.18)). This also displays fairly accurate signal strengths and times of arrival although the noise is higher than in the case of the low pass wavelet.

4.3 THE WIDE BAND WAVELET

Fig. 4.3(a) shows the composite for the wide band signal while fig. 4.3(b) displays the resolved pulse train. The time domain version of a wide band signal is characterized by a narrow signal width. Hence, filtering

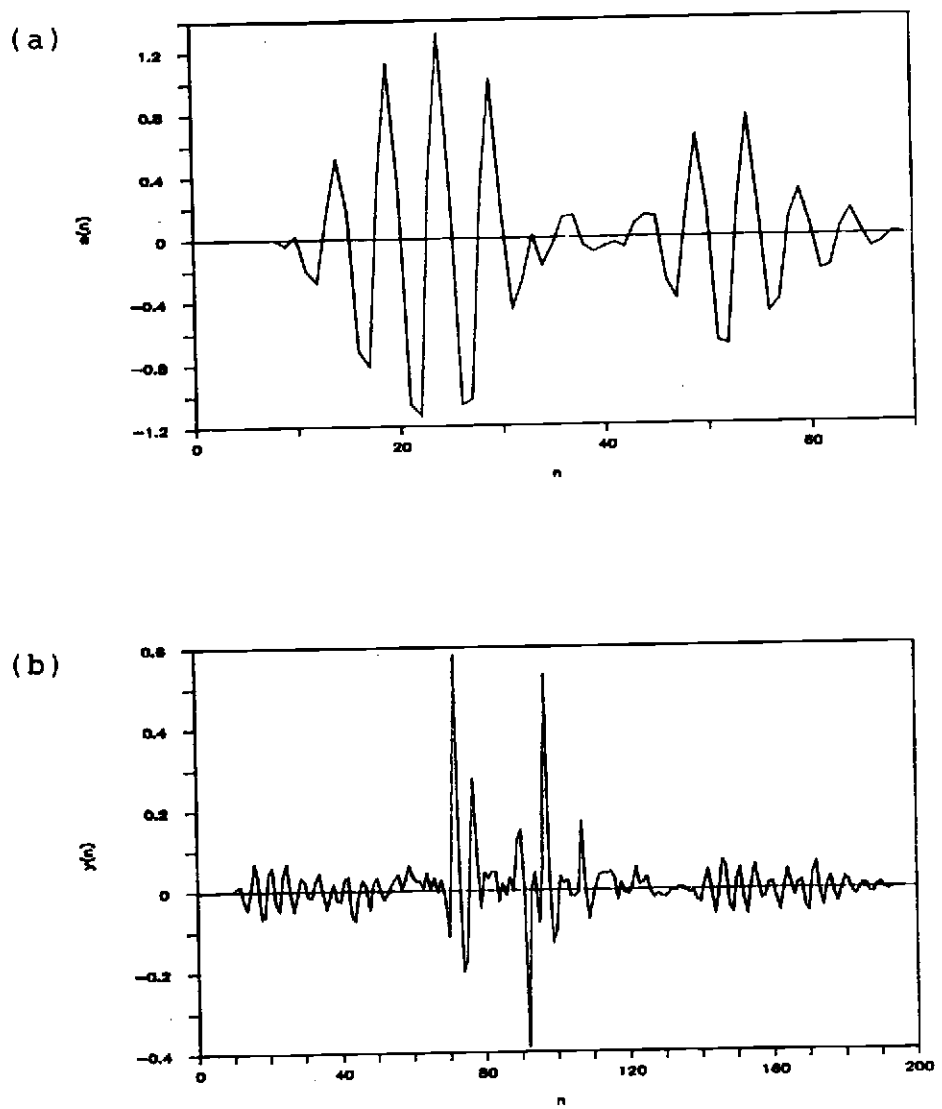

fig. 4.2 Resolution of Narrow Band Wavelets

(a) Composite Signal
(b) Filtered Signal Using Spike Filter

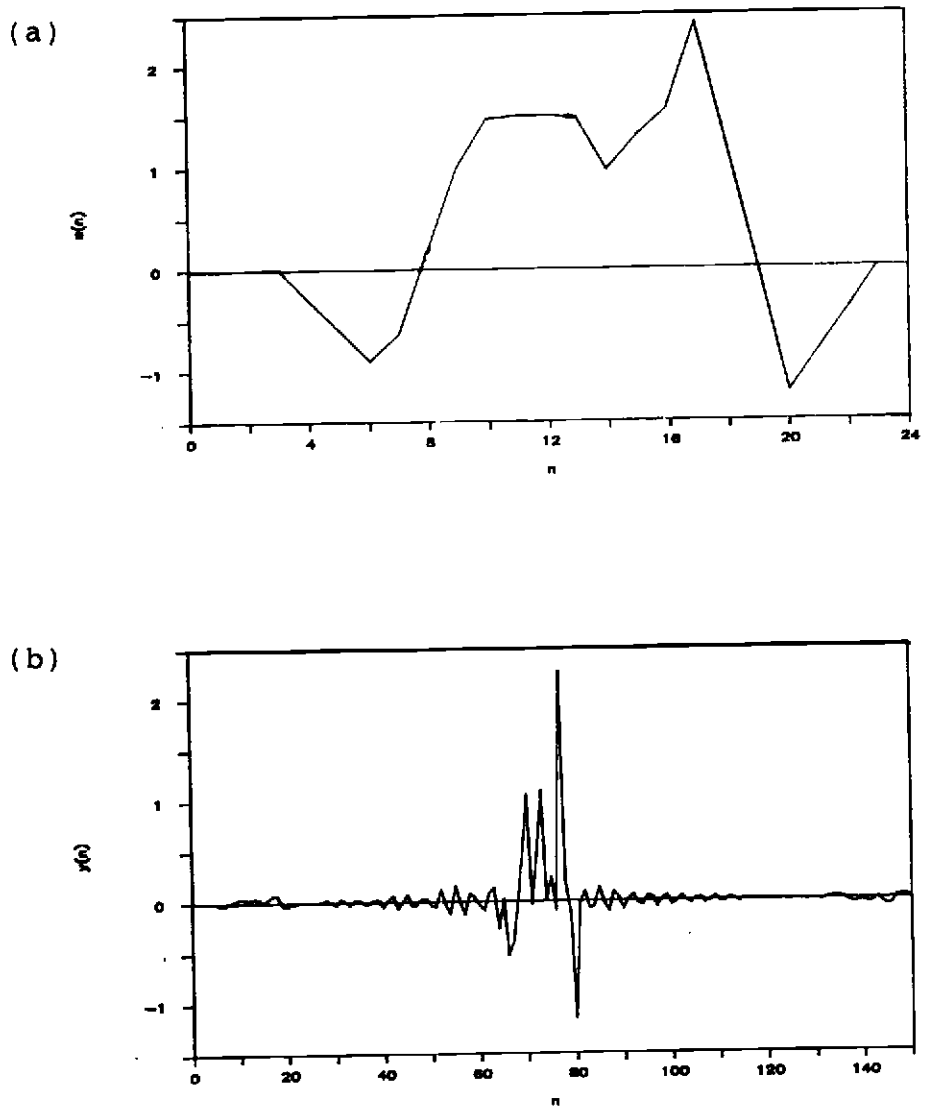

fig. 4.3 Resolution of Wide Band Wavelets

(a) Composite Signal
(b) Filtered Signal Using Spike Filter

becomes necessary in such cases only when the reflections are clustered very close together. Otherwise it is easily detectable without any further processing. In fact using a Gaussian filter with $\sigma = 2.5$ for this particular example does not serve any purpose since the Gaussian pulse is wider than the basic wavelet. The Gaussian that is narrower than the basic wavelet produces a wider bandwidth in the frequency domain and cannot be made use of due to aliasing. Therefore, with this type of basic wavelet a spike filter has to be used for resolution of closely spaced reflections.

4.4 THE CHIRP WAVELET

The chirp signal shown in fig. 4.4 is used as an example to illustrate the effectiveness of using one of the deconvolution methods described in chapter 3 in reducing aliasing. Table 3 gives the corresponding information on the composite for this case.

Table 3 - Amplitudes and arrival times for fig. 4.5(a)

arrival time	10	25	38	50	65
amplitude	0.9	0.5	-0.7	0.9	0.3

In previous work [2], circular convolution has been used since the filter happened to be grossly aliased. The spike filter designed by method 1 is used on a composite signal of 5 echoes. Fig. 4.5(b) shows the resulting pulse train obtained by linear convolution.

The magnitude response of the chirp signal shown in fig. 4.4(b) has a bandwidth that is wider than that of the Gaussian shown in fig. 2.11. Hence, using a Gaussian filter may provide better results with less noise than in fig. 4.5(b). Fig. 4.5(c) shows the resulting Gaussian pulse train which gives an excellent representation of the echoes.

Figs. 4.6(a) and (b) show the output of the respective filters when the composite signal consists of echoes with closer spacing than in the previous example. The amplitudes and arrival times used to build the composite for this case are same as those given in Table 1. The spike output still provides good resolution, although noisy, and the echoes are distinguishable from one another while the Gaussian output has very poor resolution with only 4 distinguishable echoes.

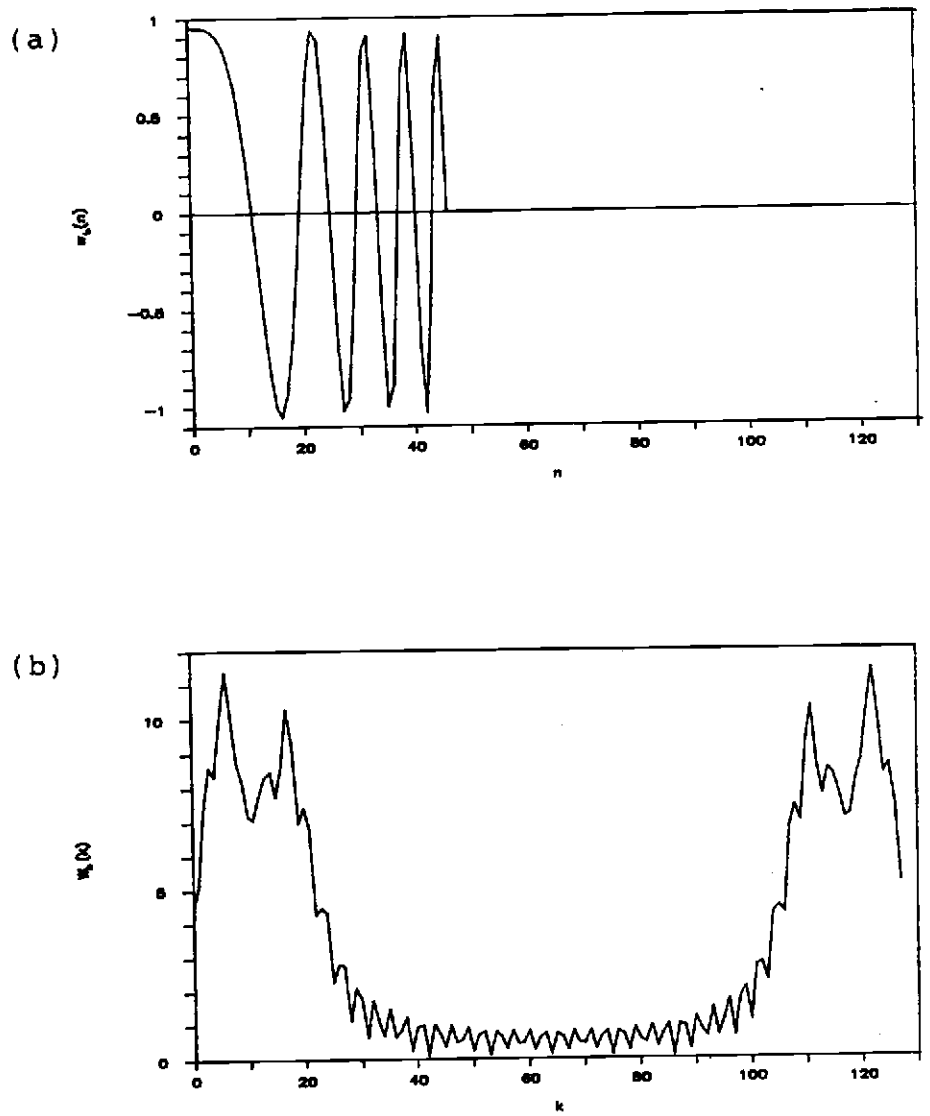

fig. 4.4 Chirp Wavelet

(a) Basic Wavelet
(b) Magnitude Response of Basic Wavelet

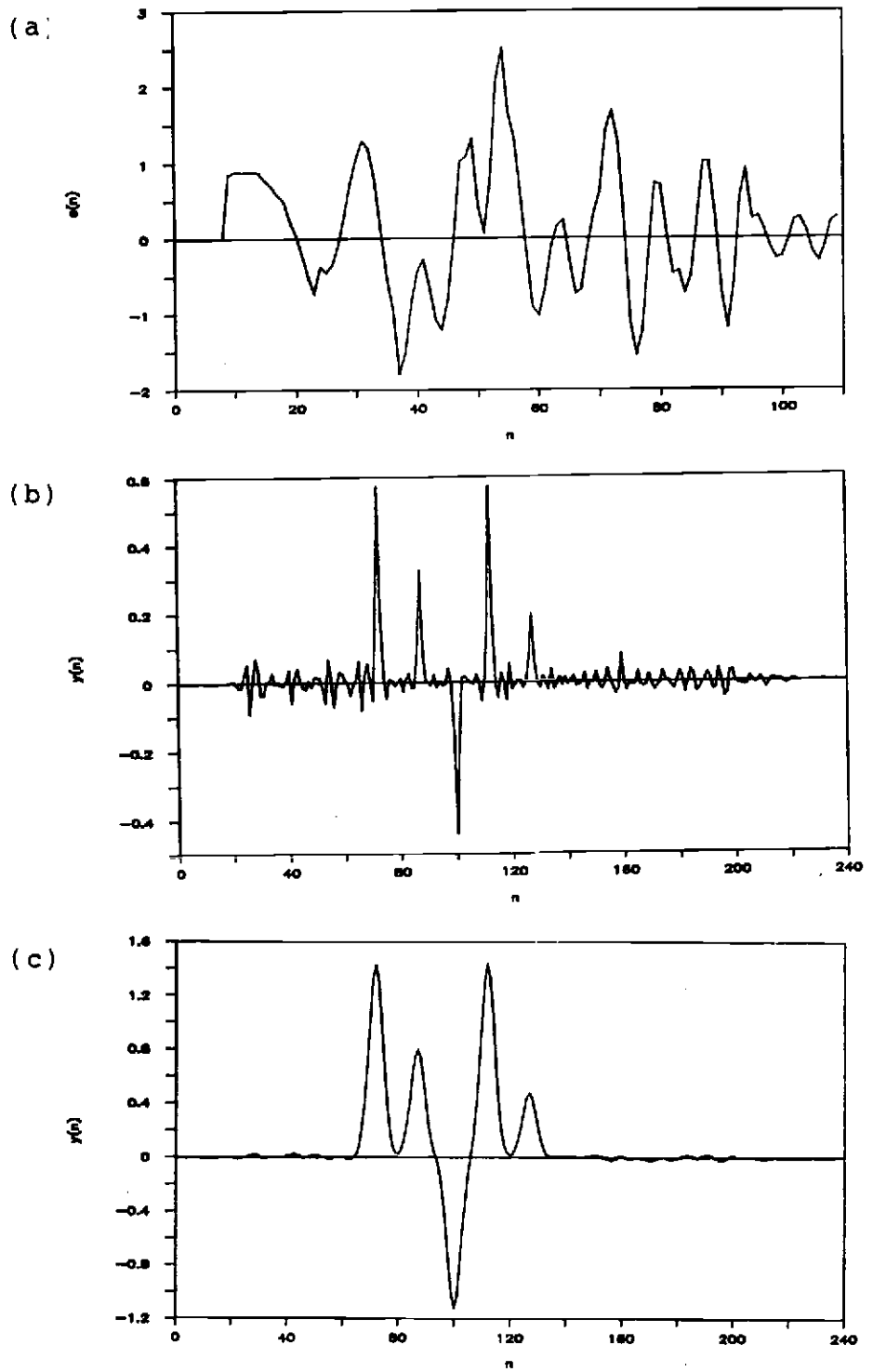

fig. 4.5 Resolution of Chirp Wavelets

(a) Composite Signal (as in Table 3)
(b) Filtered Signal Using Spike Filter
(c) Filtered Signal Using Gaussian Filter

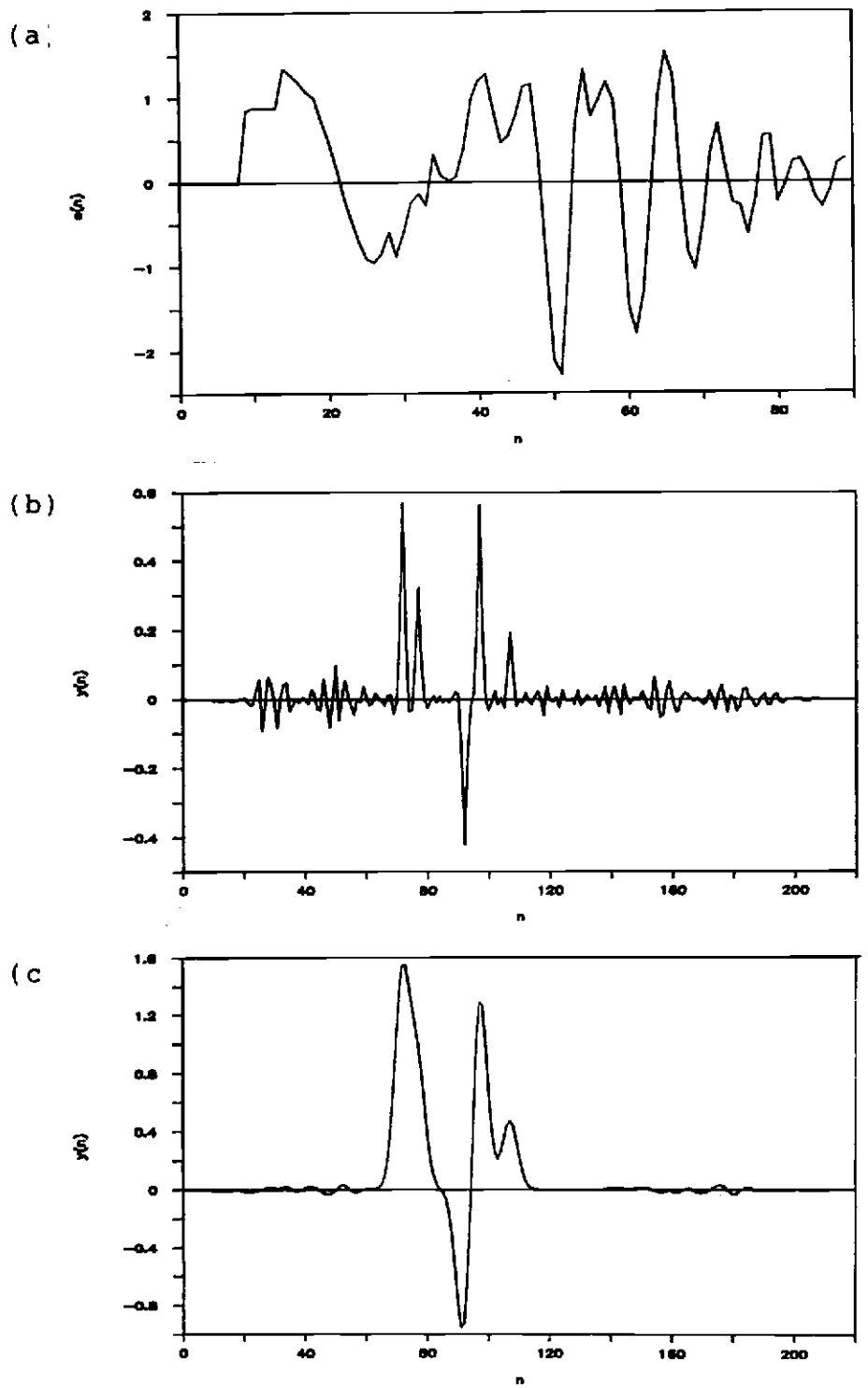

fig. 4.6 Resolution of Chirp Wavelets

(a) Composite Signal (as in Table 1)
(b) Filtered Signal Using Spike Filter
(c) Filtered Signal Using Gaussian Filter

5.0 SUMMARY/CONCLUSION

This thesis deals with the practicability of frequency domain deconvolution in the design of inverse filters. It was apparent that frequency domain deconvolution involved only a simple division of DFT's. The complexity and solution time required in time domain techniques is much reduced with the availability of efficient FFT routines.

An application of frequency domain deconvolution on the resolution of multiple overlapped signals revealed a number of problems involving, 1. noisy solutions

2. aliased filters.

All of the associated problems are addressed in chapter 2, and it is observed that they are greatly influenced by the location of zeros of the basic wavelet. The true inverse filter is shown to be an IIR filter. Using FFT's for the calculation forces a truncation of the filter to a finite length. Consequently, a problem arises if the significant filter coefficients are not included in this finite length. Sometimes an increase in the FFT length may solve the problem. Otherwise the solution will be an aliased version of the true filter.

An important fact to investigate before proceeding with the deconvolution is to find out what type of sequence is represented by the filter unit sample response. If it is a minimum phase sequence, no delay is necessary at the output. If it is a maximum phase sequence, a delay equal to the filter length would be required at the output. If it is a mixed phase sequence, the output delay will be something in between.

The location of zeros on the unit circle can be identified by inspection of the FFT of the basic wavelet. The two main problems encountered are noise due to singularities in the denominator of the frequency domain equation and aliasing due to long filter lengths. These adverse effects can be controlled to some extent by the use of an appropriate reflector series. The problem with the zeros of the basic wavelet could be minimized if the zeros of the reflector series are also in the same vicinity.

The Gaussian pulse of $\sigma = 2.5$ is one such example where zero cancellation could take place. As seen in chapter 2, the Gaussian has all its zeros on the left hand side of the z plane. Most basic wavelets fall in the category of low pass signals which also have their zeros on the left hand plane. For best results the band width of the Gaussian pulse should be less than or equal to that of the basic wavelet. This ensures that no poles of the filter will be left on or near the unit circle without a zero in its

vicinity. Unfortunately, although the noise and aliasing problems are greatly reduced in this instance, it results in poor resolution of closely spaced echoes. For values of σ less than 2.5, the bandwidth of the Gaussian gets larger and aliasing becomes significant although there is an improvement in the resolution. Therefore, a compromise has to be made between output pulse width and the output noise.

There are still other signals that do not benefit by the use of a Gaussian output. Narrow band signals fall in this category. These signals are characterized by zeros all around the unit circle. A spike output is the obvious choice for this type although it tends to have a considerable noise content and infinitely long filter lengths.

One of the main contributions of this thesis was to realize the importance of the problems mentioned above and to address them using the optimal compensation method and the least squares regression method described in chapter 3. When implementing these methods, the noise in the solution is reduced at the expense of accuracy. An optimization parameter selects a filter which negotiates a reasonable tradeoff between noise and accuracy. Plots of noise and accuracy against the parameter provide adequate information for a user to decide on a satisfactory region of operation. An estimate of the ratio of signal to peak noise given by R provide additional information about the relative

amplitudes of the true signal and maximum noise peak. In regions of satisfactory operation, it was found that in general, the optimum compensation method provided better accuracy while the least squares regression method provided better noise immunity.

Analysing the effects of the two methods on different types of basic wavelets revealed that for low pass wavelets, the least squares regression method was more suitable due to the nature of its compensating function in the denominator of the frequency domain equation. In the case of narrow band wavelets, although the noise in the signals are approximately the same for both methods, the signal to maximum noise ratio given by R shows that the compensation method is better at reducing aliasing which is a predominant problem in narrow band wavelets. Basic wavelets such as the wide band signal in example 3 are extremely narrow in the time domain. In order to distinguish closely spaced echoes of such a signal, one of the two methods described in chapter 3 has to be employed. A Gaussian filter proves useless in this case since the output pulse width would be too large to provide the necessary resolution. In the design of spike filters for this type of basic wavelet, both filters provide equal results with the exception that the regression method gives slightly better noise immunity over the other method.

Both methods employ an adaptive filter on the true frequency response of the inverse filter. The adaptive filter utilized in method 1 is the same type as the basic wavelet but the one used in method 2 is essentially a low pass filter with a bandwidth extending to that of the higher cutoff frequency of the basic wavelet.

Another important contribution of the thesis was the implementation of the two frequency domain methods to reduce aliasing, which is a common occurrence in the design of spike filters. In the past circular convolution was used to obtain the required output from such aliased filters. It was found in this work that aliasing could be reduced by utilizing the two methods to pull the zeros of the basic wavelet slightly away from the unit circle. Hence, the non aliased filter designed this way can be used in a linear convolution to produce satisfactory results. The chirp wavelet given in chapter 4 is a good example of using the methods to reduce the problem of aliasing.

Chapter 4 illustrates the resolution of signals using the filters designed in chapter 3. It was seen that spike filters provide better resolution with more noise while Gaussian filters provide poor resolution with less noise.

It was seen that higher resolution spike filters with reduced noise and aliasing could be designed using the two deconvolution methods. The next step would be to test their performance in the face of practical real time signals. The selection of a optimum value for the parameter, possibly using the mean square error is another area that requires future investigation. Finally, research involving the design of a frequency domain function that is based on the nature of the basic wavelet would result in an improvement on the methods presented in this thesis.

BIBLIOGRAPHY

1 Alan V. Oppenheim and Ronald W. Schafer, Digital Signal Processing, Prentice - Hall, Inc. 1975.

2 H. L. Tan, "Frequency Domain Deconvolution," MS Thesis submitted to the Department of Electrical and Computer Engineering at Oregon State University on Jan. 17, 1984.

3 S. Treitel and E. A. Robinson, "The Design of High Resolution Digital Filters," IEEE Transactions on Geoscience Electronics, vol. GE-4, no. 1, pp. 25 - 38, June 1966.

4 Sedki M. Riad and Robert B. Stafford, "Impulse Response Evaluation Using Frequency Domain Optimal Compensation Deconvolution," Proc. 23rd Midwest Symp. on Circuits and Systems (toledo,OH), pp. 521 - 525, Aug. 1980.

5 B. R. Hunt, "Deconvolution of Linear Systems by Constrained Regression and its Relationship to the Wiener Theory," IEEE Transactions on Automatic Control, pp. 703 - 705, Oct. 1972.

6 Bidyut Parruck and Sedki M. Riad, "Study and Performance Evaluation of Two Iterative Frequency Domain Deconvolution Techniques," IEEE transactions on Instrumentation and Measurement," vol. 33, No. 4 December 1984.

APPENDIX

PROGRAM FOR FINDING THE INVERSE FILTER USING

1. THE OPTIMUM COMPENSATION METHOD

2. THE LEAST SQUARES REGRESSION METHOD

```
      PROGRAM INVFILT
      COMPLEX H(128),X(128),Y(128)
      REAL REALX(128),REALH(128),REALY(128)
*************************************************
*                  INITIALIZE                   *
*************************************************
      OPEN(1,FILE='X.DAT',STATUS='NEW')
      OPEN(2,FILE='MAGX.DAT',STATUS='NEW')
      OPEN(3,FILE='MAGH.DAT',STATUS='NEW')
      OPEN(4,FILE='H.DAT',STATUS='NEW')
      OPEN(5,FILE='Y.DAT',STATUS='NEW')
      PI=2.0*ASIN(1.0)
      N=128
      NPWR=7
   10 FORMAT( E10.3,5X,F10.4)
*************************************************
*          COMPUTE THE BASIC WAVELET            *
*************************************************
      WRITE(*,*)'WHAT IS THE BASIC WAVELET? -(1-LOW PASS, 2-NARROW BAND,
     *3-WIDE BAND, 4-CHIRP)'
      READ(*,*)NTYPE
      IF (NTYPE .EQ. 1) CALL LP(REALX,NX)
      IF (NTYPE .EQ. 2) CALL NB(REALX,NX)
      IF (NTYPE .EQ. 3) CALL WB(REALX,NX)
      IF (NTYPE .EQ. 4) CALL CHIRP(REALX,NX)
      WRITE(*,*)'SPECIFY AMOUNT OF NOISE TO BE ADDED'
      READ(*,*)S
      ISEED=3
      DO 100 I=1,NX
      CALL UNIFRV(ISEED,U)
      REALX(I)=REALX(I)+(U-0.5)/S
      WRITE(1,10)REALX(I),FLOAT(I-1)
      X(I)=CMPLX(REALX(I),0.0)
  100 CONTINUE
      DO 200 I=NX+1,128
      REALX(I)=0.0
      WRITE(1,10)REALX(I),FLOAT(I-1)
      X(I)=CMPLX(0.0,0.0)
  200 CONTINUE
```

```
*************************************************
*            COMPUTE THE REFLECTOR SERIES       *
*************************************************
      WRITE(*,*)'SPECIFY TYPE OF FILTER TO BE DESIGNED
     *(1-SPIKE, 2-GAUSSIAN)'
      READ(*,*)IFILT
      WRITE(*,*)'SPECIFY THE AMOUNT OF OUTPUT DELAY'
      READ(*,*)M
      IF (IFILT .EQ. 1) THEN
      DO 300 I=1,N
      Y(I)=CMPLX(0.0,0.0)
  300 CONTINUE
      Y(M)=CMPLX(1.0,0.0)
      ELSE
      WRITE(*,*)'WHAT IS SIGMA?'
      READ(*,*)SIGMA
      ROOT=SQRT(2.0*PI)
      DO 400 I=1,N
      R=10.0*(EXP(-0.5*FLOAT(I-M)**2/SIGMA**2))/(ROOT*SIGMA)
      Y(I)=CMPLX(R,0.0)
  400 CONTINUE
      ENDIF
*************************************************
*            COMPUTE THE FFT's                  *
*************************************************
      CALL FFT(X,NPWR)
      WRITE(2,10)(CABS(X(I)),FLOAT(I-1),I=1,N)
      CALL FFT(Y,NPWR)
*************************************************
*      FIND THE INVERSE FILTER BY ONE OF THE    *
*            DECONVOLUTION METHODS              *
*************************************************
      WRITE(*,*)'SPECIFY THE METHOD TO BE USED'
      READ(*,*)MTHD
      IF (MTHD .EQ. 1) CALL FILT1(X,Y,H,N)
      IF (MTHD .EQ. 2) CALL FILT2(X,Y,H,N,NPWR)
      WRITE(3,10)(CABS(H(I)),FLOAT(I-1),I=1,N)
      CALL IFFT(H,NPWR)
      CALL IFFT(X,NPWR)
      DO 500 I=1,N
      REALX(I)=(X(I)+CONJG(X(I)))/2.0
      REALH(I)=(H(I)+CONJG(H(I)))/2.0
  500 CONTINUE
      WRITE(4,10)(REALH(I),FLOAT(I-1),I=1,N)
```

```
*************************************************
*    COMPUTE THE OUTPUT OF THE FILTER FOR A     *
*                  SINGLE ECHO                  *
*************************************************
      WRITE(*,*)'TO OBTAIN OUTPUT BY LINEAR CONVOLUTION TYPE 1'
      WRITE(*,*)'TO OBTAIN OUTPUT BY CIRCULAR CONVOLUTION TYPE 2'
      READ(*,*)IOUT
      N2=2*N-1
      IF (IOUT .EQ. 1) CALL FOLD(N,REALX,N,REALH,N2,REALY)
      IF (IOUT .EQ. 2) CALL CIRCON(REALX,REALH,REALY,N)
      WRITE(5,10)(REALY(I),FLOAT(I-1),I=1,N2)
      END


*************************************************
* NOTE: THE ARRAYS REALX,REALH,REALY CAN BE     *
*       PUT IN A COMMON STATEMENT TO BE         *
*       SHARED WITH SOME OF THE FOLLOWING       *
*       SUBROUTINES.                            *
*************************************************


      SUBROUTINE LP(REALX,NX)
* THIS SUBROUTINE COMPUTES THE SEQUENCE FOR THE LOW PASS WAVELET.
      REAL REALX(128)
      NX=46
      DO 100 I=1,NX
      X(I)=(FLOAT(I-1)*EXP(-0.2*FLOAT(I-1))
  100 CONTINUE
      RETURN


      SUBROUTINE NB(REALX,NX)
* THIS SUBROUTINE COMPUTES THE SEQUENCE FOR THE NARROW BAND WAVELET.
      REAL REALX(128)
      NX=26
      DO 200 I=1,NX
      X(I)=SIN(2.0*PI*FLOAT(I-1)/50.0)*COS(2.0*PI*FLOAT(I-1)/5.0)
  200 CONTINUE
      RETURN


      SUBROUTINE WB(REALX,NX)
* THIS SUBROUTINE COMPUTES THE SEQUENCE FOR THE WIDE BAND
* WAVELET.
      REAL REALX(128)
      NX=7
      NX1=NX+1
      DO 100 I=1,NX1/2
      REALX(I)=FLOAT(I-1)/3.0
      REALX(NX1-I)=REALX(I)
  100 CONTINUE
      RETURN
```

```
      SUBROUTINE CHIRP(REALX,NX)
* THIS SUBROUTINE COMPUTES THE SEQUENCE FOR THE CHIRP WAVELET.
      REAL REALX(128)
      NX=46
      DO 100 I=1,NX
      REALX(I)=COS(2.0*PI*0.002*(FLOAT(I-1))**2)
  100 CONTINUE
      RETURN


      SUBROUTINE FILT1(X,H,Y,N)
* THIS SUBROUTINE IMPLEMENTS THE OPTIMAL COMPENSATION METHOD.
* X = FOURIER TRANSFORM OF INPUT, Y = FOURIER TRANSFORM OF OUTPUT,
* H = FOURIER TRANSFORM OF INVERSE FILTER.
      COMPLEX X(128),H(128),Y(128)
      WRITE(*,*)'WHAT IS BETA?'
      READ(*,*)BETA
      DO 100 I=1,N
      H(I)=Y(I)*CONJG(X(I))/((CABS(X(I)))**2+BETA)
  100 CONTINUE
      RETURN


      SUBROUTINE FILT2(X,H,Y,N,NPWR)
* THIS SUBROUTINE IMPLEMENTS THE LEAST SQUARES REGRESSION METHOD.
* X = FOURIER TRANSFORM OF INPUT, Y = FOURIER TRANSFORM OF OUTPUT,
* H = FOURIER TRANSFORM OF INVERSE FILTER.
      COMPLEX X(128),H(128),Y(128)
      C(1)=CMPLX(1.0,0.0)
      C(2)=CMPLX(-2.0,0.0)
      C(3)=CMPLX(1.0,0.0)
      DO 100 I=4,N
      C(I)=CMPLX(0.0,0.0)
  100 CONTINUE
      CALL FFT(C,NPWR)
      WRITE(*,*)'WHAT IS GAMMA?'
      READ(*,*)GAMMA
      DO 200 I=1,N
      H(I)=Y(I)*CONJG(X(I))/((CABS(X(I)))**2+GAMMA*(CABS(C(I)))**2)
  200 CONTINUE
      RETURN
```

```
      SUBROUTINE FOLD(N1,REALX,N2,REALH,N3,REALY)
* THIS SUBROUTINE COMPUTES THE LINEAR CONVOLUTION OF TWO SEQUENCES
* INPUTS - REALX,REALH        OUTPUT - REALY
      DIMENSION REALX(128),REALH(128),REALY(256)
      N3=N1+N2-1
      DO 100 I=1,N3
      REALY(I)=0.0
  100 CONTINUE
      DO 200 I=1,N1
      DO 200 J=1,N2
      K=I+J-1
      REALY(K)=REALY(K)+REALX(I)*REALH(J)
  200 CONTINUE
      RETURN


      SUBROUTINE CIRCON(REALX,REALH,REALY,N)
* THIS SUBROUTINE COMPUTES THE CIRCULAR CONVOLUTION OF TWO SEQUENCES
* INPUTS - REALX,REALH        OUTPUT - REALY
      DIMENSION REALX(N),REALH(N),REALY(N)
      DO 100 I=1,INT(N/2)
      T=REALX(I)
      REALX(I)=REALX(N-I+1)
      REALX(N-I+1)=T
  100 CONTINUE
      DO 200 IND=1,N
      T=REALX(N)
      DO 300 I=N,2,-1
      REALX(I)=REALX(I-1)
  300 CONTINUE
      REALX(1)=T
      SUM=0.0
      DO 400 I=1,N
      SUM=SUM+REALX(I)*REALH(I)
  400 CONTINUE
      REALY(IND)=SUM
  200 CONTINUE
      RETURN
```

```
      SUBROUTINE FFT(Z,M)
*************************************************************
* THIS SUBROUTINE FINDS THE FFT OF A SEQUENCE OF LENGTH     *
* N AND PLACES THE COMPLEX FFT IN THE SAME SPACE THAT       *
* CONTAINED THE ORIGINAL SEQUENCE. NOTE : N = 2**M          *
*************************************************************
      COMPLEX Z(128),U,W,T
      N=2**M
      NV2=N/2
      NM1=N-1
      J=1
      DO 7 I=1,NM1
      IF (I .GE. J) GO TO 5
      T=Z(J)
      Z(J)=Z(I)
      Z(I)=T
    5 K=NV2
    6 IF (K .GE. J) GO TO 7
      J=J-K
      K=K/2
      GO TO 6
    7 J=J+K
      PI=2.0*ASIN(1.0)
      DO 20 L=1,M
      LE=2**L
      LE1=LE/2
      U=(1.0,0.0)
      W=CMPLX(COS(PI/FLOAT(LE1)),SIN(PI/FLOAT(LE1)))
      DO 20 J=1,LE1
      DO 10 I=J,N,LE
      IP=I+LE1
      T=Z(IP)*U
      Z(IP)=Z(I)-T
   10 Z(I)=Z(I)+T
   20 U=U*W
      RETURN


      SUBROUTINE IFFT(Z,M)
*************************************************************
* THIS SUBROUTINE FINDS THE INVERSE FFT OF A SEQUENCE       *
* OF LENGTH N.                                              *
*************************************************************
      COMPLEX Z(128),U,W,T
      N=2**M
      DO 4 I=1,N
      Z(I)=CONJG(Z(I))
    4 CONTINUE
      CALL FFT(Z,M)
      DO 9 I=1,N
      Z(I)=(CONJG(Z(I)))/FLOAT(N)
    9 CONTINUE
      RETURN
```