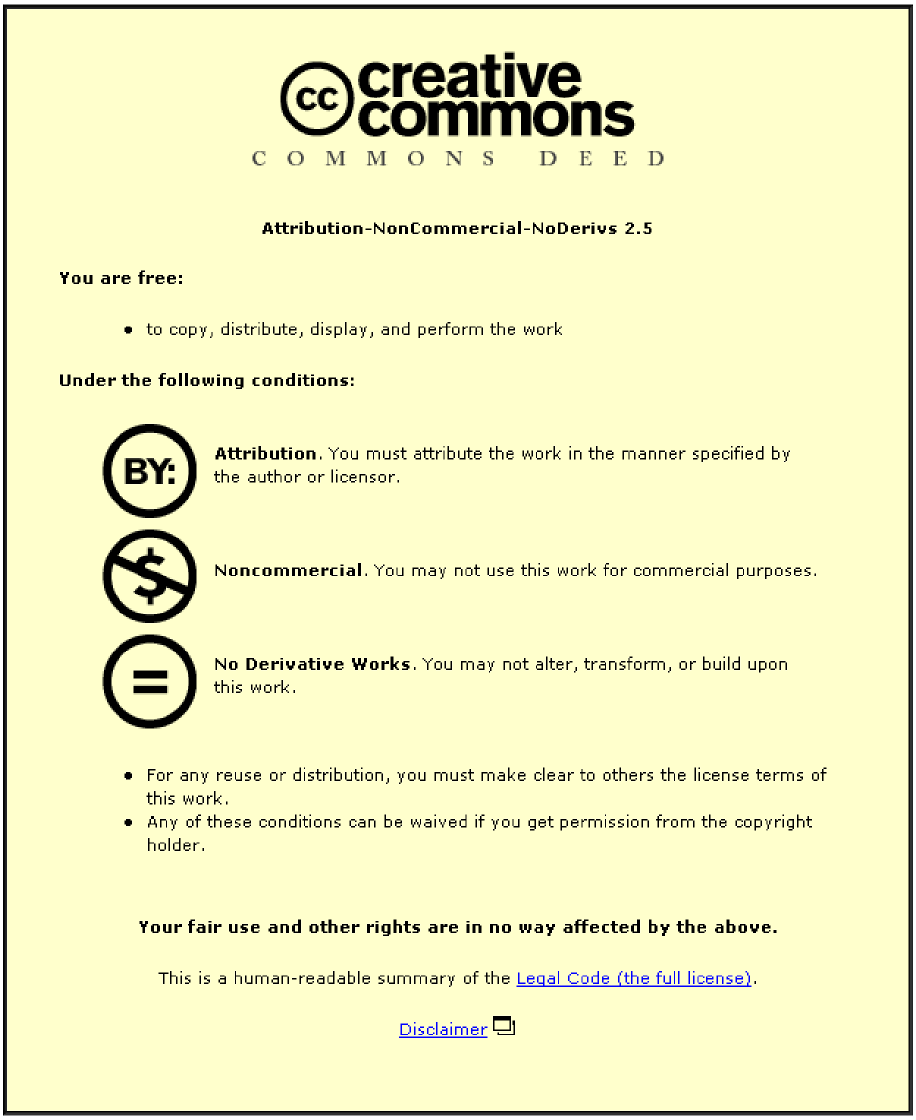


DESIGN OF ENERGY EFFICIENT PULSE TRAINS FOR RADAR

by

UMBERTO SOMAINI

A Doctoral thesis

Submitted in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy of the University of Technology, Loughborough.

July, 1976

Supervisor: Dr. J.E. Hudson,
Department of Electronic and Electrical Engineering.

To my Parents

ABSTRACT

The work presented in this thesis is concerned with the design of discrete coded waveforms for improving range resolution and clutter performance of radar systems. This approach to signal design offers many advantages in terms of waveform shaping and digital implementation of processors. Assuming a matched filter receiver, the bulk of the work is concentrated on studying the autocorrelation function properties of these waveforms, which are directly related to the range resolution.

The main objective is to synthesize pulse trains subject to a fixed amplitude constraint, whose autocorrelation sidelobes are as low as possible. Constant amplitude waveforms are attractive for a number of reasons; the principal one being the optimum utilization of transmitter power.

Pulse train signals can be synthesized directly by factorizing the spectrum of specified autocorrelation functions. The problems which arise if the autocorrelation function is only given in magnitude are considered and a design method is presented. For some applications, especially digital implementation, the design objective may be to approximate the response characteristics of a given analogue waveform. It is shown that virtually all the desired properties of analogue signals can be retained if the sampling interval is chosen properly. In addition various suggestions for reducing the range sidelobes of the autocorrelation function are discussed.

An attempt is made to solve the signal design problem using numerical optimization methods that incorporate the fixed amplitude constraint. In particular, a constrained optimization technique is developed for synthesizing binary sequences with good autocorrelation function properties. Moreover, the problem of designing pairs of

phase coded pulse trains with low autocorrelation sidelobes and small mutual crosscorrelation is considered.

In the case of impulse-equivalent sequences known as Huffman codes, a synthesis method based on uniform pulse trains is shown to yield sequences with good energy efficiency. Furthermore, a new approach to the signal design problem using Huffman codes and parameter variational techniques is presented.

Although the range sidelobes can be reduced quite effectively by numerical methods, for some applications they might still be too large. Thus optimum sidelobe reduction filters, which minimize the detection loss subject to a set of sidelobe constraints, are derived by mismatching the receiver filter.

Finally, in the case where significant target velocity is encountered, it becomes necessary to consider not only the range but also the velocity resolution properties of the transmitted waveform. This is done for the various types of pulse trains using the standard range-doppler ambiguity description of Woodward.

ACKNOWLEDGEMENTS

The author wishes to thank his supervisor, Dr. J.E. Hudson, for his encouragement during the period of research. In addition, he would like to thank Dr. M.H. Ackroyd for his genuine interest and guidance. Thanks are also given to Professor J.W.R. Griffiths for the use of the facilities in the Department of Electronic and Electrical Engineering.

The help the author has indirectly received through comments and criticisms of his colleagues and particularly Dr. J.P. Highmore, is gratefully acknowledged. Thanks are also given to Mrs. S.M. Peach for her excellent typing of the manuscript.

Finally, the author is grateful to the Department of Education of the Kanton Solothurn, Switzerland, for sponsoring this project.

CONTENTS

Page

LIST OF PRINCIPAL SYMBOLS AND ABBREVIATIONS

$a(n)$, $a(nT)$, $a(n)$	number sequence, time series, samples of $a(t)$
$A(z)$	z-transform of $a(n)$
$A(f)$	spectrum of $a(n)$
$A(k)$	spectrum samples
A	input matrix defined by Eq. (8.14)
ABF	ambiguity function
ACF	autocorrelation function
AM	amplitude modulation
a.m.ph.m.	amplitude and phase modulated pulse train
$C(k)$	samples of the ACF of the power spectrum
DFT	discrete Fourier transform
$d(n)$	desired filter output sequence
$E\{\cdot\}$	expected value of $\{\cdot\}$
E^n	n-dimensional vector space
E_s	sidelobe energy (ratio)
E	energy of time series
f_c	carrier frequency
$F(\underline{x})$	performance index, scalar function
$F_p(\underline{x})$	ℓ_p-measure, performance index
FT	Fourier transform
FFT	fast Fourier transform
$g(n)$	filter output sequence
$h(n)$	filter weighting sequence
$H(f)$	filter transfer function
IDFT	inverse discrete Fourier transform
IFT	inverse Fourier transform
$Im[\cdot]$	imaginary part of $[\cdot]$
$J(\underline{x})$	Jacobian matrix
$J_n(x)$	Bessel function of the first kind

FM	frequency modulation
LFM	linear FM
L_s	mismatch loss factor
$m(t)$	squared envelope of ACF
m_c	compression ratio
MF	matched filter
MMF	mismatched filter
NM	Nelder and Mead algorithm
$\max\limits_{k} \lvert r(k) \rvert$	peak sidelobe level
PAT	pattern search algorithm
POC	Powell-Coggin search algorithm
POG	Powell-Golden-Section search algorithm
POS	Powell's sums of squares algorithm
$P(\underline{x},\underline{s})$	penalty-function
PM	phase modulation
$Q(\underline{x})$	quantization penalty-function
Q	quadratic form
QP	quadratic phase code
$r(\tau)$	autocorrelation function
R	autocorrelation matrix
r.m.s.	root-mean-square
$r_{12}(\tau)$	crosscorrelation function
$\mathrm{Re}\,[\cdot]$	real part of $[\cdot]$
rect(t/T)	rectangular function
$s(t)$	complex signal envelope
$S(f)$	spectrum of $s(nT)$
$\lvert S(f) \rvert^2$	power spectrum of $s(nT)$
S_{max}	peak sidelobe level
SNR	signal-to-noise ratio
T(mil)	execution time in millunits (1 millunit ≃ 0.5 seconds)

TDMA	time-division multiple-access systems
T	sampling interval
T_s	signal duration
$\mathrm{tri}(t/T)$	triangular function
W	bandwidth
$w(nT)$	samples of white Gaussian noise
$\underline{x}$	vector of design variables
X	radius of Huffman codes
Y	DFT matrix
z	complex frequency variable
ZT	z-transform

Greek letters and symbols

α	gain factor
$\underline{\delta}$	correction vector
$\delta(t)$	Kronecker delta function
∂	partial derivative symbol
Δ	change in variable or vector, step length
∇	gradient
ε	arbitrary small positive number
$\varepsilon(t)$	phase correction function
$\varepsilon(n)$	error sequence
η_i	eigenvalues of ACF matrix R
θ	an angle
$\theta(f)$	phase spectrum
λ	weighting parameter
ν	doppler frequency
$\underline{\nu}$	parameter vector in penalty algorithm
ρ	energy efficiency
τ	time delay
$\tau_g(f)$	group delay function
$\underline{\phi}$	phase vector

$\lvert\chi(\tau,\nu)\rvert$	ambiguity function
$\psi(\tau,\nu)$	MF response to received signal
Superscripts	
k	denotes stages in the minimization of $F(\underline{x})$
T	transpose of a vector
$(\ )$	encloses stage number
$*$	complex conjugate
Overlays	
$\sim$	complex conjugate transpose of a vector
$\wedge$	vector at minimum, optimum solution
$'$	used to differentiate one constant from another; also used as shorthand notation for d/dt.
Other signs	
$\lVert\cdot\rVert$	magnitude of a vector, the norm
$\lvert\cdot\rvert$	absolute value of a scalar
$\{\ \}$	set, sequence
$\subset$	subset of
$\in$	element of
$\underline{x}$	denotes vector of variables
col$(\cdot)$	column vector
Π	product operator
$*$	convolution operation
$\leftrightarrow$	Fourier transform pair

CHAPTER 1

INTRODUCTION

1.1 Background

Radar is a technique for remote sensing using radio waves. Its basic purpose is to detect the presence of a target of interest and to provide information concerning the target's location, motion, size and other parameters. The problem of target detection is accomplished in a typical radar system (Fig. 1.1) by transmitting a radio signal and detecting the waveform reflected by the target in the presence of unavoidable system noise and reflections from undesired scatterers (clutter). If a return signal of adequate strength is received, it is further analysed to determine the target's range, velocity and so forth. This process is known as parameter estimation. The range of the target is determined by measuring the delay of the return signal. Similarly, the velocity of the target can be estimated, neglecting higher order effects, by measuring the shift in carrier frequency (doppler shift) of the received waveform. Furthermore, the transmitted signal can be carefully chosen and generated so as to optimise its capability for extracting the desired information about the target.

Target detection and parameter estimation are difficult practical problems, particularly for small targets at great distances. In principle, however, both problems are simple when only a single target is present. Target resolution, which may be defined as the capability of a radar system to recognise a particular target in the presence of others, is one of the most important but also most demanding tasks[1,2].

For high performance radar systems these tasks become increasingly complex. This explains the continuing effort being directed towards improving the resolution capabilities of modern radars. Some improvements are still being made in the components which affect radar performance;

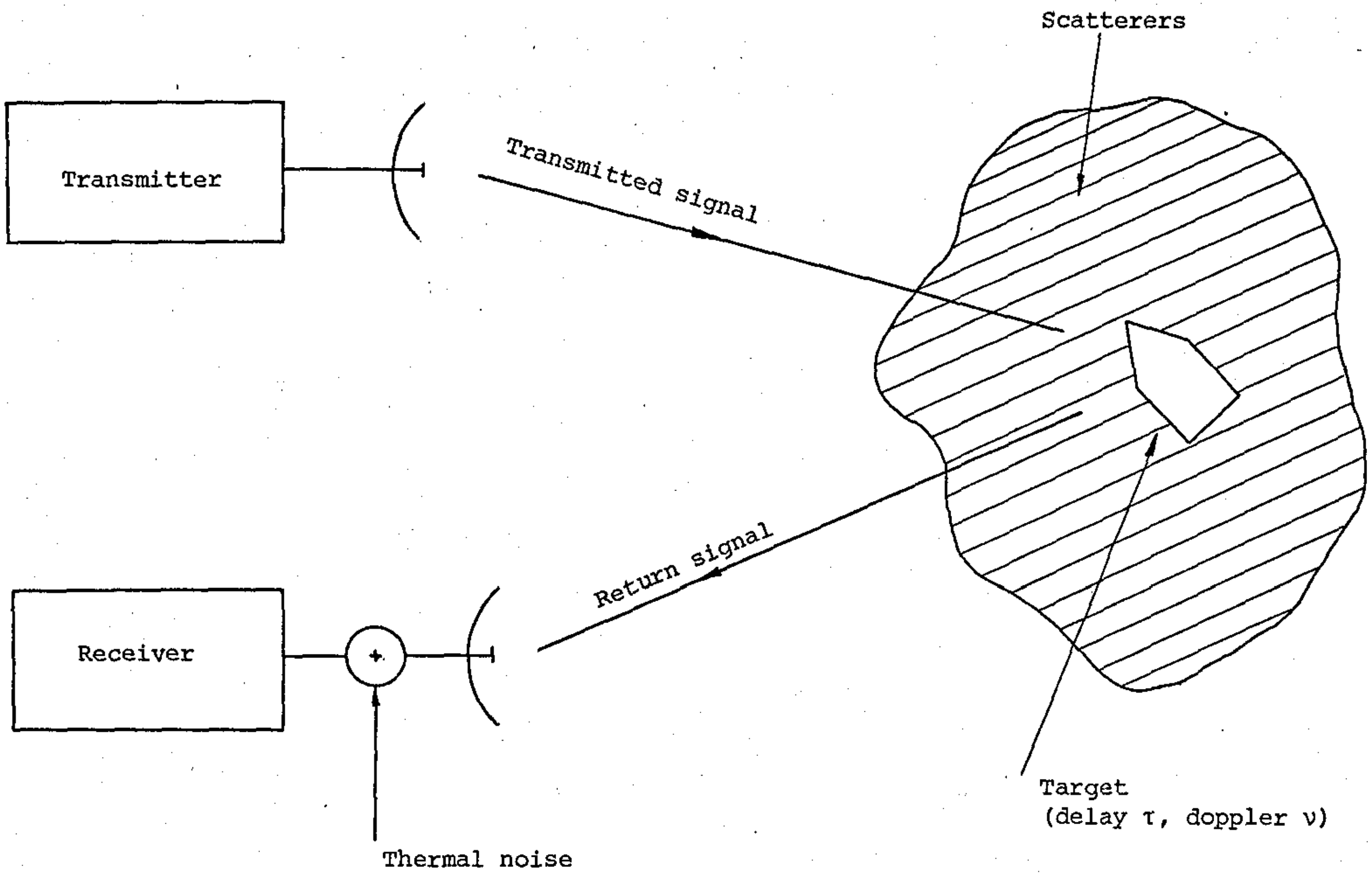


Fig. 1.1 A typical radar system.

for example, receivers with low noise figures, transmitters with higher output power, and antennas with more gain. Further improvement can be obtained by means of elaborate signal processing schemes. In recent years a considerable amount of work has been done in digital processing for radar. The technical problems imposed by modern radar systems are those of processing (real-time) a large number of data and the requirement of complex signal processing operations. These problems can only be solved reliably by the use of digital techniques, although in some cases modern optical processing schemes can offer an alternative.

The choice of a suitable transmit waveform is an important problem in radar design. This is so because the waveform controls resolution and clutter performance and also bears heavily on the system cost. As compactness, cheapness and computational speed of digital microcircuits continue to increase their use in signal processing applications becomes more practical. In particular, the advent of solid-state antenna arrays has its impact on radar system designers in two principal ways. First, peak-power limitations of solid-state array elements have necessitated the use of waveforms with long durations in order to achieve the required signal energy over a desired range. The required stability and reproducibility of such signals can only be satisfied reliably by digital signal generation and processing. Secondly, the ability to switch the beam of solid-state arrays at high speeds gives the radar a multi-function capability, thus requiring the flexibility to enable a variety of waveforms to be employed[3]. These requirements have made digital signal processing with its inherent adaptability an attractive alternative to analogue processing.

Theoretical studies which provide the basis for technical advances have not, so far, solved the general signal design problem. The knowledge of the properties of pulse trains, a class of signals particularly well suited to digital processing, is therefore of increasing practical importance.

An early suggestion for using discrete coded waveforms in radar appeared in a paper by Siebert[4] treating the general problems of radar. Siebert noted that certain binary coded waveforms offered substantial improvement in range and velocity resolution. However, it was shown that in order to obtain these improvements, it would be necessary to employ long periodic binary sequences known as pseudo-random sequences[5]. Later, Lerner[6] suggested that the periodic sequence could be modified to form an aperiodic signal and yet retain the nearly optimum resolution property of the waveform. The use of aperiodic signals allows the construction of passive matched filter receivers. At about this time the signal design problem was approached in a slightly different way. Assuming a matched filter receiver, the range resolution capability (in the absence of doppler shift) was found to be directly related to the autocorrelation function of the transmitted waveform. Therefore, the approach consisted of attempting to design aperiodic binary sequences having optimum autocorrelation properties[7-9]. These sequences were called 'optimum finite code groups' or Barker sequences.

Since these early evaluations a number of authors have made valuable contributions in the field of waveform design[10-15]. An interesting analytical method for generating binary codes was reported by Boehmer[10] using Number Theory. Another quite different approach to the problem is discussed in a paper by Vakman[11] and Varakin[12]. The authors suggest a synthesis procedure on the basis of spectral theory and the method of stationary phase.

Heimiller[16], Frank[17] and Zadoff[18] have shown that there are other suitable codes if the restriction of 0^o-180^o phase shifting is removed[19]. In the case of Frank codes[17], higher order poly-phase coded words can be generated by coding each sub-pulse into one of M phases. Huffman[20] considered the problem of designing amplitude and phase modulated pulse trains. He has shown that finite-length signals with nearly ideal autocorrelation functions can be generated. This property, however, is achieved at the expense of amplitude modulation which results in increased system complexity and lower energy utilization at the transmitter. Nevertheless, the additional expense of encoding and decoding of amplitude and/or phase modulated waveforms may be justified for radars that must cope with land clutter or operate in a dense-target environment. However, the use of amplitude modulated pulse trains is precluded in most high power applications due to the inevitable loss in energy efficiency.

In spite of considerable effort that has been devoted to the problem of designing waveforms with high range resolution there seems to be a lack of signal design techniques and theory. All present methods tend to contain an element of trial and error and, moreover, rely on the skill and ingenuity of the designer. In short, the study of the properties of pulse trains does not appear to have progressed much beyond an understanding of the types described above. The currently accepted belief that there is no ideal waveform is not surprising considering the various different tasks modern radar systems have to perform. On the other hand, the inability to find an ideal waveform is not an excuse for failure to search for locally optimum waveforms for specific radar applications and environments.

The effort in this thesis is directed towards the improvement of a factor which constitutes a fundamental limitation to radar performance; namely the transmitted waveform. Although the ways in which the transmitted signal affects the system performance are well understood[21], there seems to be no obvious solution to the problem of designing energy efficient pulse trains for high resolution radars. Therefore, the work presented in this thesis is concerned primarily with the study and development of design methods for improving the range resolution capability of pulse trains. The pulse sequences discussed later, besides representing an interesting mathematical area, are also of practical significance in related fields such as digital communication and navigation.

The main objective in this work is to consider optimum finite-length pulse trains subject to an amplitude constraint, that is the amplitude of each sub-pulse of the transmittted signal is constant. This reduces the waveform design problem to an optimum selection of the phase modulating function only. The constant amplitude constraint of the sub-pulses is attractive for a number of reasons; the pulses are convenient to gate, and when clipped are distorted less than amplitude modulated signals, reduction of system complexity and probably the most important of all, the transmitter power is most efficiently utilized.

1.2 Outline of Investigation

This thesis is concerned with a number of different aspects of the pulse train design problem for radar systems. As a basis for later work, the principles of waveform and processor design are outlined in Chapter 2. This is carried out assuming matched filtering and digital signal generation and processing.

In Chapter 3 the important problem of synthesizing pulse trains from autocorrelation functions, which are specified at discrete points in phase and magnitude or in magnitude only, is tackled.

The resolution properties of pulse trains approximating FM type signals are studied in Chapter 4. This approach to code generation is perhaps more one of analysis rather than that of design. The work includes various suggestions of reducing the range sidelobes of the autocorrelation function.

In Chapter 5 the signal design problem is approached via numerical optimization techniques. This is done by minimizing an appropriately defined performance measure reflecting the resolution capability of a signal. The optimization problem is formulated so as to incorporate a fixed amplitude constraint which arises from practical radar considerations. It was necessary, however, to consider first the performance of the different non-linear optimization algorithms when applied to functions of medium to high dimensionality. For this purpose four favoured algorithms are selected and briefly described.

Chapter 6 presents the relative merits of the four non-linear algorithms described in the preceeding chapter. The most efficient optimization method was then applied to generate phase modulated pulse trains of various lengths. This chapter also treats a new method of designing binary sequences using constrained optimization techniques. Moreover, the properties of pairs of phase coded sequences having low autocorrelation sidelobes and small mutual crosscorrelation are studied.

Much of the material included deals with purely phase coded sequences. For certain applications, however, the use of amplitude modulation can provide a useful means of improving range resolution and clutter rejection. Chapter 7 presents the results of the synthesis of energy efficient amplitude and phase modulated pulse trains. In connection with Huffman codes a new approach to the signal synthesis problem using parameter variational techniques is developed.

Chapter 8 stands apart somewhat from the other chapters in that it is concerned with sidelobe reduction techniques using mismatched filters. The central problem in mismatched filtering is to consider the trade-off possibilities between resolution and degradation in signal detectability.

Finally, Chapter 9 considers the combined range and velocity resolution properties of the synthesized pulse trains. This is done using the standard range-doppler ambiguity function of Woodward[21].

CHAPTER 2

SIGNAL PROCESSING CONCEPTS AND WAVEFORM DESIGN

2.1 Introduction

This chapter includes some of the general principles of waveform and processor design. The choice of the transmit waveform and of the receiver configuration involves in general two separable design problems[4]. The waveform must be chosen to optimize performance in some total environment, e.g. clutter, dense-target environment, etc. However, there is no universal waveform which would meet all the requirements for any arbitrary environmental conditions. The design of the radar processor (hardware) is somewhat separable, since there are generally a number of ways to implement a near optimum receiver for a given waveform. Cost, complexity and reliability are usually the bounds on processor design rather than physical realizability.

The complexity of a signal processing system depends on the complexity of each elementary operation the system has to perform and on their number. This in turn depends on the number of input and output channels. For modern high resolution radars the complexity of the signal processing system and the amount of data to be handled easily reaches critical limits. However, this thesis will be restricted to studying the coded waveforms as modulating functions. The various possibilities of modulating a carrier with these functions and the ways and means of implementing processors will not be considered here.

This chapter also introduces the waveform design problem with the now standard range-doppler ambiguity function description of Woodward[21]. The ambiguity function discussion will only contain a summary of the results pertinent to the descriptions of the specific waveforms. The bulk of the work will be concentrated on the study of the zero doppler cross-section of the ambiguity function, since the particular concern is the range resolution capability of a signal. Detailed descriptions of the general properties of various waveform classes and their ambiguity functions can be found in many excellent references[22-26].

The general description of the ambiguity functions is, despite their wide study, not without its limitations. While with modern computers it is not difficult to derive the ambiguity function for a particular waveform, it is generally not possible to derive a specific waveform, starting with a given ambiguity function. Moreover, a thorough and bounded description of the target and clutter environment is required for a unique selection of the appropriate waveform.

2.2 Representation of Pulse Trains

Pulse trains are waveforms particularly well suited to digital processing. In general, these waveforms can be regarded as a finite number of contiguous coherent carrier pulses (sub-pulses) each modulated in amplitude, phase and frequency. Analytically this class of signals can be expressed in the form

$$u(t) = \sum_{n=o}^{N} a(n)\, \mathrm{rect}(t/T-n)\, \cos\{2\pi(f_c + f(n))t + \phi(n)\} \qquad (2.1)$$

where T = pulse duration

f_c = carrier frequency

$a(n)$ = amplitude of nth pulse

$\phi(n)$ = phase of nth pulse

$f(n)$ = frequency deviation of nth pulse

and rect(t/T) denotes Woodward's rect function shown in Fig. 2.1 and defined as

$$\mathrm{rect}(t/T) = \begin{cases} 1 & , \ |t| \leqslant T/2 \\ 0 & , \ \text{elsewhere} \end{cases} \qquad (2.2)$$

It is noted that for a given carrier frequency, f_c, and sub-pulse duration, T, the signal is completely specified by the ordered

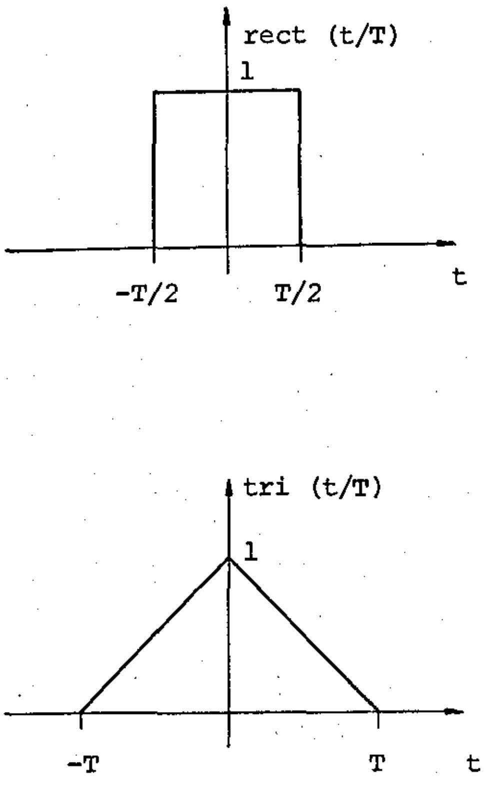


Fig. 2.1 Rect and tri functions.

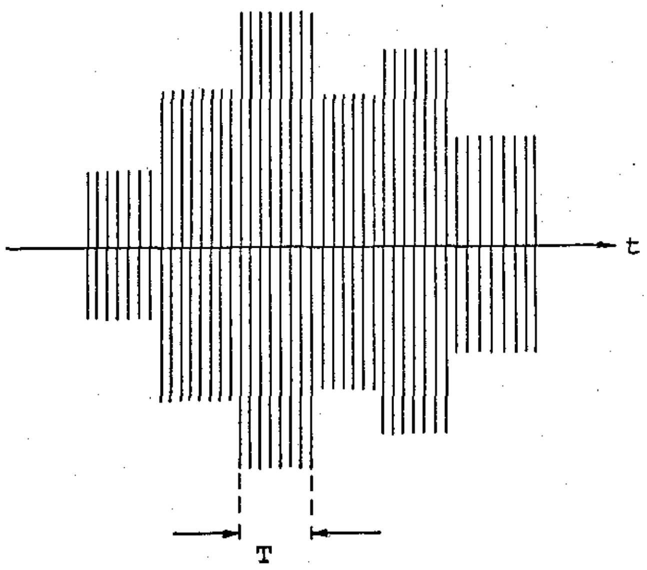


Fig. 2.2 Amplitude and phase modulated pulse train.

sequences $\{a(n)\}$, $\{\phi(n)\}$, $\{f(n)\}$.

Pulse trains can be conveniently divided into three basic groups, as follows :

Group I : $\{\phi(n)\} = \{f(n)\} = 0$, $\{a(n) = 1,0\}$

Group II : $\{f(n)\} = 0$

Group III : $\{a(n) = 1,0\}$,$\{\phi(n)\} = 0$

The sub-class of pulse trains considered here belong to Group II and are referred to as amplitude and phase modulated (a.m.ph.m.) pulse trains[27]. A typical waveform of this type is shown in Fig. 2.2.

Although this thesis is not concerned with the various methods of generating a.m.ph.m. pulse trains, a few words seem appropriate at this point.

In digital signal generation the generator presents a sampled and quantized signal at the output that can be used either for amplitude, phase or frequency modulation of a carrier signal. Therefore, a digital signal generator is in general a time-clocked unit capable of furnishing a sequence of digital data which are utilized to reconstruct, by filtering or modulation or both, the output signal. In most radar applications phase modulation, and particularly digital phase modulation, is the most attractive modulation method. In the case of phase modulation the signal can be reconstructed by either of the two methods shown in Fig. 2.3. The digital phase sample generator produces samples as binary numbers which are converted into cos $\phi(n)$ and sin $\phi(n)$ and thus into samples of $s_r(t)$ and $s_i(t)$ (see Eq. (2.5)). Alternatively the binary number can be used to digitally phase modulate a carrier signal in a digital phase modulator.

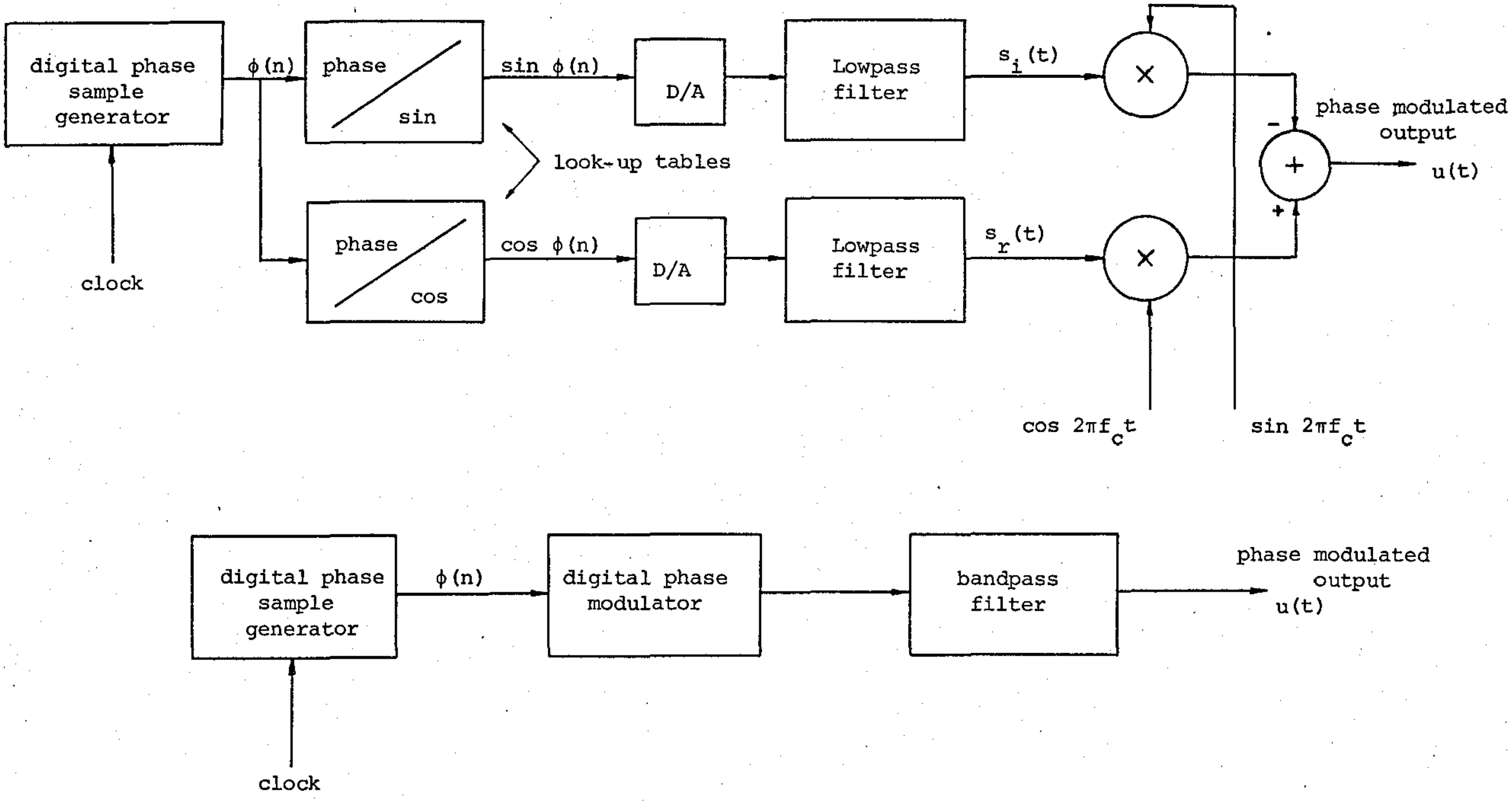


Fig. 2.3 Phase modulation of a carrier signal.

2.2.1 Complex Envelope and Bandpass Filtering

Since a.m.ph.m. pulse trains are of the 'bandpass' type their theory can be simplified by the use of the complex envelope representation[28-30]. The bandpass signal represented by Eq. (2.1) can be expressed as

$$u(t) = \mathrm{Re}\,\{s(t)\, \exp(j2\pi f_c t)\} \tag{2.3}$$

where s(t) is the complex envelope of u(t) and is given by*

$$s(t) = \sum_{n=o}^{N} a(n)\, \mathrm{rect}(t/T-n)\, \exp(j\phi(n)) \tag{2.4}$$

As seen from a comparison with Eq. (2.1), the real signal envelope is the absolute value of s(t). Thus the bandpass signal u(t) is completely described by a knowledge of its carrier frequency, f_c, and its low frequency complex envelope s(t). In terms of real and imaginary parts of s(t), the bandpass signal u(t) is given by

$$u(t) = s_r(t)\cos 2\pi f_c t - s_i(t)\sin 2\pi f_c t \tag{2.5}$$

where

$$s(t) = s_r(t) + j\, s_i(t)$$

The low pass signals $s_r(t)$ and $s_i(t)$ are called the *in-phase* and *quadrature components*, respectively, of the bandpass signal. From the foregoing it is evident that the complex envelope s(t) is independent of the carrier frequency, f_c. Therefore, it is sufficient to consider the complex envelope as the transmitted signal and to ignore the carrier term $\exp(j2\pi f_c t)$.

The great advantage of complex signal representation is that operations such as linear bandpass filtering (convolution) can be

*As presented here the waveform is not normalised, i.e. $\int_{-\infty}^{\infty} |s(t)|^2\, dt \neq 1$. In order to accomplish normalization it is necessary to multiply (2.4) by a factor $(T \sum_{o}^{N} a^2(n))^{-1/2}$. Without affecting the general discussion to follow and in most subsequent work this factor has been omitted for convenience.

expressed directly in terms of the complex envelope. In other words, bandpass filtering of a signal can be treated simply in terms of complex lowpass signals[27,28]. This filtering operation and how it can be implemented in terms of real lowpass filters is shown in Fig. 2.4. The case of particular concern is the situation when a pulse train is applied to a filter whose impulse response is itself a pulse train with the same carrier frequency, f_c, and the same individual sub-pulse duration, T.

Consider the discrete coded waveform, consisting of (N+1) pulses representing the impulse response of a linear filter as an ordered set of complex numbers

$$\{h(n)\} = (h(0),\ h(1),\ \ldots.,\ h(N)) \tag{2.6}$$

In signal processing terminology such a sequence is often referred to as a time series. Alternatively the sequence {h(n)} with values h(n) = h(nT) can also be visualized as being generated by sampling the complex envelope h(t) of the corresponding continuous waveform every T seconds. Although sequences do not always arise from sampling analogue waveforms, for convenience h(n) will often be referred to as the nth sample of the sequence. In addition h(n) will also be used as a shorthand notation for the sequence {h(n)}.

The magnitude of the complex number h(n) represents the amplitude of the nth pulse while the angle of h(n) specifies its phase. The complex envelope of the filter impulse response $\gamma(t)$ can easily be obtained by convolving the time series of Eq. (2.6) with Woodward's rect function.

$$\gamma(t) = \sum_{n=o}^{N} h(n)\delta(t-nT) * \text{rect}(t/T)$$

$$= \sum_{n=o}^{N} h(n)\int_{-\infty}^{\infty} \delta(\tau-nT)\ \text{rect}\left[(t-\tau)/T\right]\ d\tau$$

$$\gamma(t) = \sum_{n=o}^{N} h(n)\ \text{rect}(t/T-n) \tag{2.7}$$

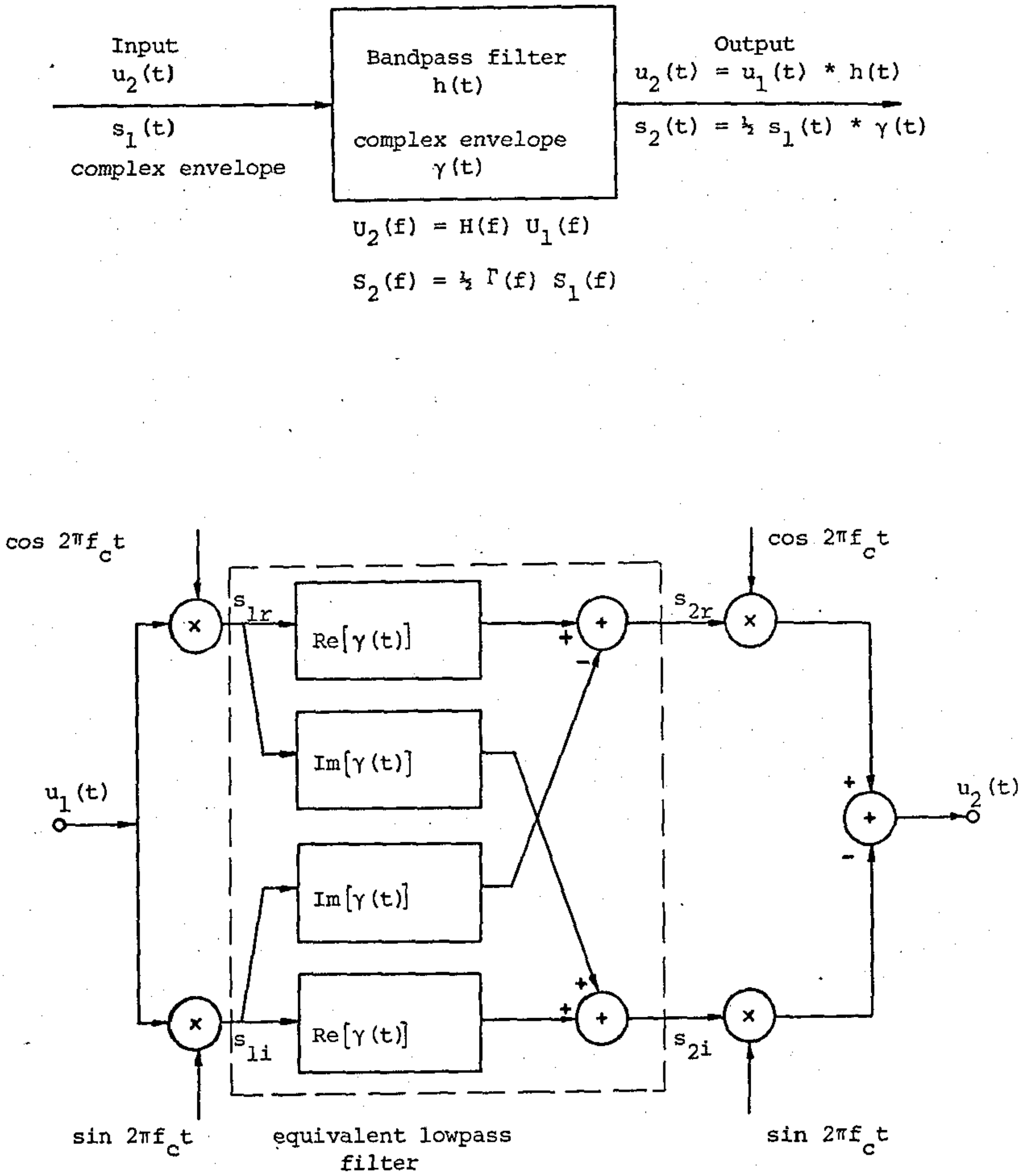


Fig. 2.4 Baseband filtering.

where * indicates the convolution operation.

The Fourier transform (FT) of Eq. (2.7) can now be found by using the familiar rules of transform theory.

$$s(t-T) \longleftrightarrow S(f)\ \exp\ (-j2\pi fT)$$

$$\text{rect}(t/T) \longleftrightarrow T\ \text{sinc}(fT)$$

where $$\text{sinc}\ (fT) = \frac{\sin\pi fT}{\pi fT}$$

The spectrum of the complex envelope $\gamma(t)$ is therefore

$$\Gamma(f) = T\text{sinc}(fT)\sum_{n=o}^{N} h(n)\ \exp\ (-j2\pi fnT)$$

or

$$\Gamma(f) = T\text{sinc}(fT)H(f) \tag{2.8}$$

The term Tsinc(fT)denotes the spectrum of the function rect(t/T). It is evident that the spectrum H(f) is independent of the form of a single sub-pulse. However, it is periodic with a period of 1/T and it represents the spectrum of the coded sequence h(n) as illustrated in Fig. 2.5. The relationship between the FT's of an analogue signal and its sampled version is thus given by

$$H(f) = \sum_{n=-\infty}^{\infty} H_a(f + n/T) \tag{2.9}$$

where

$$H_a(f) = \int_{-\infty}^{\infty} h(t)\ \exp(-j2\pi ft)\ dt$$

$$H(f) = \sum_{n=-\infty}^{\infty} h(nT)\ \exp(-j2\pi fnT)$$

The above relationship essentially formulates the time domain sampling theorem which states that a continuous function of time whose spectrum is limited to the band ($\pm W/2$) is completely defined by time domain samples taken at intervals of 1/W.

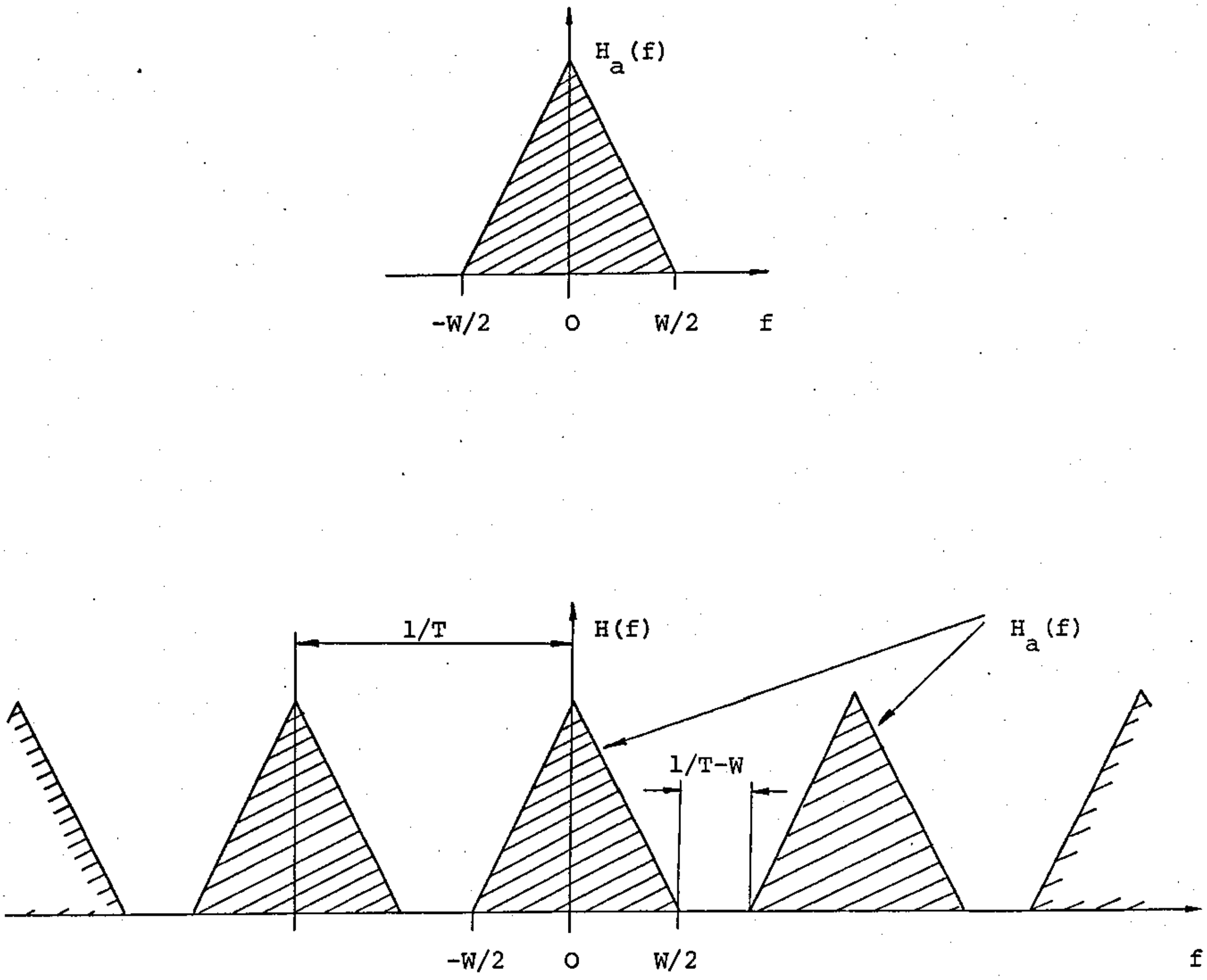


Fig. 2.5 Effect of time domain sampling on bandlimited signal.

Similarly the input signal, having (M+1) pulses is represented by a(n). The complex envelope e(t) of the output signal has a FT which is, according to standard transform rules of linear filtering, the product of the FT's of the complex envelopes of the input waveform and the filter impulse response scaled by a factor 1/2. Thus,

$$E(f) = 1/2\{\sum_{n=o}^{M} a(n)\ e^{-j2\pi fnT}\}\{\sum_{n=o}^{N} h(n)\ e^{-j2\pi fnT}\}\ T^2\ \mathrm{sinc}^2(fT)$$

$$= 1/2\{\sum_{n=o}^{M}\ \sum_{k=o}^{N} a(n)\ h(k)\ e^{-j2\pi f(n+k)T}\}\ T^2\ \mathrm{sinc}^2(fT)$$

$$= 1/2\{\sum_{m=o}^{M+N} c(m)\ e^{-j2\pi fmT}\}\ T^2\ \mathrm{sinc}^2(fT) \tag{2.10}$$

where

$$c(k) = \sum_{n=o}^{k} a(n)\ h(k-n)$$

$$k = 0,1,2,\ldots,(N+M)$$

The complex envelope of the output signal e(t) is given by the inverse FT (IFT) of Eq. (2.10). Using the relationship

$$T^2\mathrm{sinc}^2(fT) \longleftrightarrow T\ \mathrm{tri}(t/T)$$

leads to

$$e(t) = T/2 \sum_{m=o}^{M+N} c(m)\ \mathrm{tri}(t/T-m) \tag{2.11}$$

The function tri(t/T) is shown in Fig. 2.1 and is defined as

$$\mathrm{tri}(t/T) = \begin{cases} 1 - |t|/T & , \ |t| \leqslant T \\ 0 & , \text{ elsewhere} \end{cases} \tag{2.12}$$

At non-integer multiples of the sub-pulse duration, T, the complex envelope of the output waveform is given by linear interpolation between adjacent values. The output number sequence c(m) thus specifies, except for a constant scale factor, the output waveform at regular sampling instants T.

The main conclusion from the foregoing is that complex envelope representation of pulse trains simplifies the discrete linear filtering process which can be regarded merely as a multiplication of polynomials. In addition only the spectrum of the coded sequences need be considered to specify the waveform at integer multiples of T.

2.2.2 The z - Transform

It has been shown in the previous section that the FT can be used to describe the frequency properties of pulse trains. Another compact notation for the FT of such signals is the z - transform (ZT)[31]. The ZT is also a very convenient method of representing a signal by a set of poles and zeros in the complex z-plane. This is quite similar to Laplace transform techniques used for analogue systems which can be represented by poles and zeros in the complex s-plane.

The ZT of an arbitrary number sequence, a(n), is simply a polynomial in powers of z^{-1}, given by

$$A(z) = a(O) + a(1)\ z^{-1} + \ldots\ldots\ldots + a(N)\ z^{-N}$$

$$A(z) = \sum_{n=o}^{N} a(n)\ z^{-n} \tag{2.13}$$

where z is usually expressed in the polar form z = exp(sT). In general the frequency variable z has both real and imaginary parts. Thus if

$$s = \sigma + j2\pi f$$

$$z = e^{(\sigma+j2\pi f)T} = e^{\sigma T}\left[\cos(2\pi fT) + j\sin(2\pi fT)\right]$$

The variable z is often referred to as a 'shift' operator, since exp(-j2πfT) implies a time delay of T seconds while exp(j2πfT) represents a time advance of T seconds.

If the ZT, A(z), is evaluated on the unit circle in the z-plane ($|z| = 1$), the spectrum of the time series a(n) is obtained.

$$A(z)\Big|_{(|z|=1)} = A(f) = \sum_{n=o}^{N} a(n)\ \exp(-j2\pi fnT) \tag{2.14}$$

It is noted that the spectrum A(f) is a continuous function in frequency. In practice, however, the spectrum of discrete time series is usually evaluated using digital computers. This means that the spectrum A(f) can only be estimated at discrete points in f which is generally referred to as discrete Fourier transform (DFT). Although the spectrum can only be estimated at suitably chosen intervals in f, it can be shown that for time limited or periodic signals such a discrete representation of the underlying continuous function does not result in any loss of essential information[31]. This is sometimes referred to as the frequency sampling theorem.

Using the DFT, Eq. (2.14) can now be rewritten as a finite-length sequence

$$A(k) = \sum_{n=o}^{N} a(n)\ e^{-j2\pi kn/(N+1)} \tag{2.15}$$

$$k = 0,1,2,\ldots.,N$$

Thus the frequency spacing between successive harmonics is 1/(N+1)T and the frequency of the kth harmonic is therefore k/(N+1)T. It can be seen from Eq. (2.15) that the sequence A(k) is periodic with a period of (N+1); i.e. A(0) = A(N+1), A(1) = A(N+2), etc. Similarly the inverse transform (IDFT) of Eq. (2.15) can be written as

$$a(n) = 1/(N+1) \sum_{k=o}^{N} A(k)\ e^{j2\pi kn/(N+1)} \tag{2.16}$$

$$n = 0,1,2,\ldots.,N$$

where the multiplying factor 1/(N+1) has been included for convenience. Alternatively the above equations can be expressed in matrix form

$$\underline{a} = 1/(N+1)\ Y^*\ \underline{A} \tag{2.17}$$

where the (N+1)-element column vectors $\underline{a}$ and $\underline{A}$ are given by

$$\underline{a} = \text{col}\ (a(0), a(1), \ldots, a(N))$$

$$\underline{A} = \text{col}\ (A(0), A(1), \ldots, A(N))$$

The (N+1)x(N+1) matrix Y* is the conjugate of the DFT matrix Y given by

$$Y = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & y & y^2 & \ldots & y^N \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ 1 & y^N & y^{2N} & \ldots & y^{N^2} \end{bmatrix} \tag{2.18}$$

where $y = \exp(-j2\pi/(N+1))$.

Eq. (2.17) is the IDFT and its inverse (DFT) is therefore given by

$$\underline{A} = Y\ \underline{a} \tag{2.19}$$

It is noted that the DFT matrix Y has the property

$$Y\ Y^* = (N+1)\, I$$

where I is the identity matrix.

Although, in principle, the DFT can be evaluated using Eq.(2.15) or Eq. (2.19) in practice the fast Fourier transform (FFT) algorithm is used[32,33].

From the foregoing it is clear that a finite-duration sequence can be expressed exactly by samples of its ZT. Moreover, the periodic sequence obtained by sampling the ZT at (N+1) equally spaced points on the unit circle ($|z|=1$) in the complex z-plane is identical to the DFT. The sequence corresponding to these frequency samples is a periodically repeated version of the original sequence, such that if (N+1) samples of the ZT are used no 'overlapping' or 'aliasing' occurs. Thus, in general, a finite-duration sequence is represented as one period of a periodic sequence.

In addition it has been shown that the ZT of a pulse train is simply a power series of z^{-1} in which the coefficients of the various terms are equal to the corresponding samples. When a waveform is expressed in this form, it is possible to regenerate the number sequence merely by inspection; a number having the index n is simply the coefficient of z^{-n} in the ZT.

The linear filtering operation of Eq. (2.10) can now be rewritten using the shorthand notation of the ZT. If the sequence c(n) represents the convolution of the two sequences a(n) and h(n), then the ZT of c(n) is the product of the ZT's of a(n) and h(n), i.e. if

$$c(n) = \sum_{k=o}^{n} a(k)\ h(n-k)$$

$$n = 0,1,2,\ldots,(N+M)$$

then

$$C(z) = A(z)\ H(z) \tag{2.20}$$

This can easily be shown considering the following expressions,

$$C(z) = \sum_{h=o}^{N+M} \{ \sum_{k=o}^{n} a(k)\ h(n-k)\}\ z^{-n}$$

interchanging the order of summation yields

$$C(z) = \sum_{k=o}^{M} a(k) \sum_{n=k}^{N-k} h(n-k)\ z^{-n}$$

letting m = n-k leads to

$$C(z) = \sum_{k=o}^{M} a(k)\{ \sum_{m=o}^{N} h(m)\ z^{-m}\}\ z^{-k}$$

hence

$$C(z) = A(z)\ H(z)$$

However, if the DFT is used to evaluate Eq. (2.10), the sequences a(n) and h(n) have to be modified. Taking the straightforward DFT of finite-duration sequences and then inverse transforming the products of their spectra is equivalent to circularly convolving the periodic sequences created from the given sequences. To obtain the linear convolution both a(n) and h(n) must be (N+M+1)-point sequences. This is achieved by appending the appropriate number of zeros to both a(n) and h(n), i.e.

$$(a(0),a(1),\ldots,a(N),\overbrace{0,0,\ldots 0}^{M\text{ zeros}})$$

$$(h(0),h(1),\ldots,h(M),\overbrace{0,0,\ldots 0}^{N\text{ zeros}})$$

The (N+M+1)-point DFT of a(n) and h(n) is then taken, multiplied, and inverse transformed to obtain the correct sequence c(n).

The ZT, a familiar analytical technique in modern control and sampled data systems, thus provides an excellent tool for studies of digital systems and signals such as pulse trains. Therefore, throughout this work use is made of the ZT representation whenever possible.

2.3 Optimum Processing of Radar Signals

The transformations and interference effects to which a radar signal is subjected during its path from the transmitter to the receiver will now be analysed using Fig. 2.6. It is assumed that the signal, although generated digitally, is analogue filtered prior to transmission. The transmitted signal s(t) first passes through a time-invariant processor, which accounts for the unknown round trip amplitude attenuation α, time delay τ, doppler shift ν, and phase shift θ of the signal. Such a treatment of the transmitted signal assumes a point target (no range extent). This is a convenient assumption in analysing system performance.

In order to avoid continual repetition several general assumptions are made for subsequent discussions of radar signal processing techniques.

(i) Point targets are assumed.

(ii) Target acceleration is negligible, i.e.

$$a << \lambda/T_s^2$$

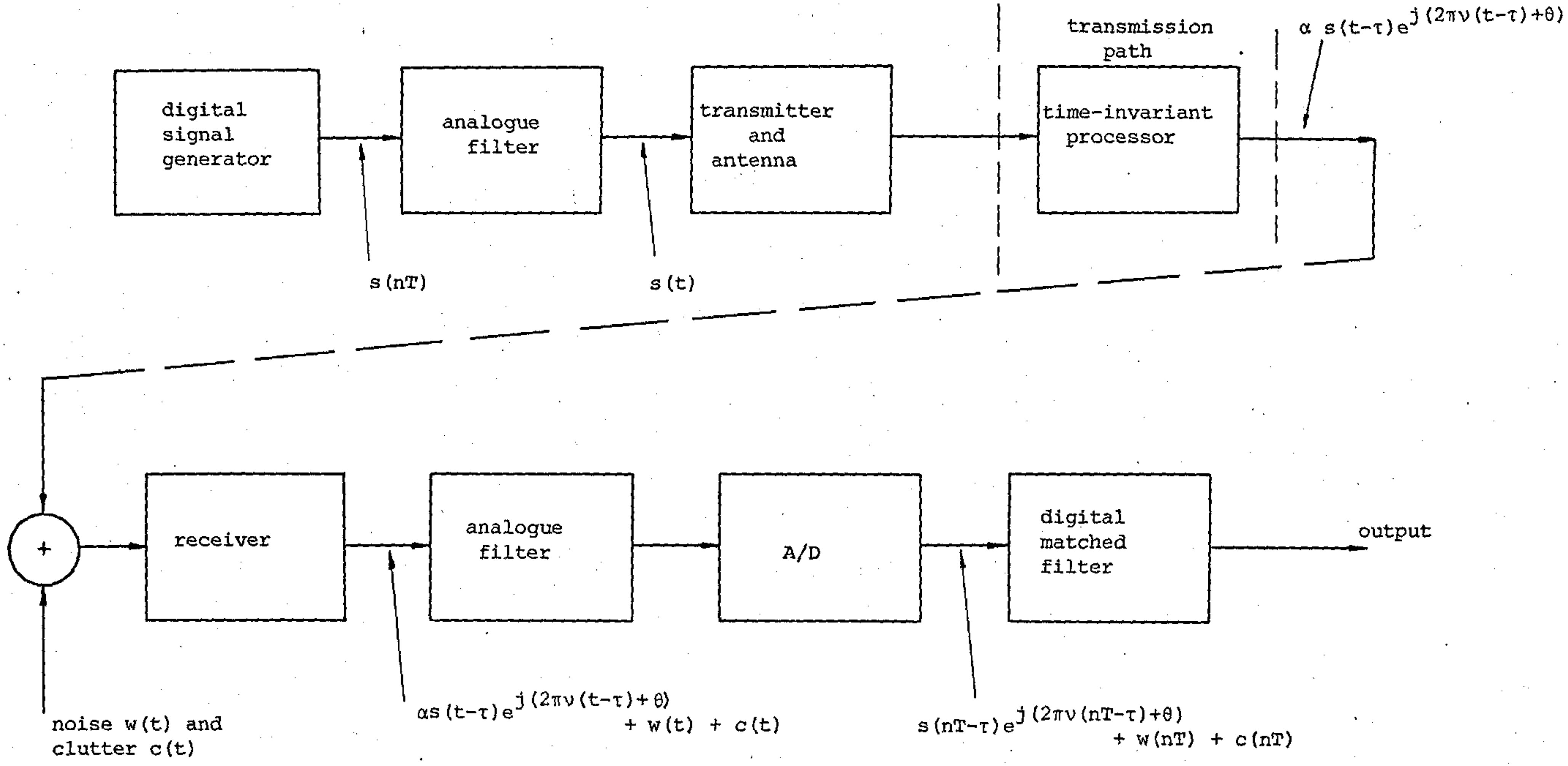

digital signal generator
analogue filter
transmitter and antenna
transmission path
time-invariant processor
$s(nT)$
$s(t)$
$\alpha\ s(t-\tau)e^{j(2\pi\nu(t-\tau)+\theta)}$
+
receiver
analogue filter
A/D
digital matched filter
output
noise $w(t)$ and clutter $c(t)$
$\alpha s(t-\tau)e^{j(2\pi\nu(t-\tau)+\theta)} + w(t) + c(t)$
$s(nT-\tau)e^{j(2\pi\nu(nT-\tau)+\theta)} + w(nT) + c(nT)$

Fig. 2.6 Block diagram of signal model.

where a is the target acceleration, λ is the carrier wavelength, and T_s is the signal duration.

(iii) Mismatch of the envelope of the received signal and the transmitted waveform due to high relative velocities is negligible, i.e.

$2v_r/c << 1/WT_s$

where v_r is the radial target velocity relative to the radar, c is the velocity of light, and W is the signal bandwidth.

(iv) All signals are narrow band,

$W << f_c$

where f_c is the carrier frequency.

Since radar returns are always immersed in noise and interference from all kinds of objects illuminated by the antenna beam, the receiver must be optimised in some manner. Additive interference introduced by a large number of independent target-like reflections such as composite returns from an extended scattering region containing terrain, rain, seawaves etc., is usually called clutter.

The return signal, immersed in clutter and noise, is now analogue filtered, digitized and processed. Three approaches have been used to derive optimum processors for radar signals,[4,21,34]

(i) Signal/noise ratio criterion (SNR)

(ii) Likelihood ratio criterion

(iii) Inverse probability criterion

Any of these criteria lead to the matched filter receiver, provided the signal is corrupted only by additive white Gaussian noise. Moreover, Woodward[21] has shown that this type of receiver also preserves all the information in the radar return. Even in situations where matched filter processing is not optimum, for example when interference from clutter is significant (coloured noise), matched filters usually provide a reasonable compromise between system performance and complexity[35]. On the other hand, matched filter processing may be used simply because information about the target environment to design more optimum processors is not available. The problem of target resolution and optimum detection for a specified clutter environment has been studied by a number of authors and will not be of prime concern here[36-39]. Therefore, unless otherwise specified a matched filter receiver is assumed.

2.3.1 Digital Matched Filter

The characteristics of the matched filter (MF) can be designated by either a frequency response function or a time response function, each being related to the other by a FT operation. In the frequency domain the MF transfer function H(f), is the complex conjugate function of the spectrum of the signal that is to be processed, except for an arbitrary scale factor and a linear phase shift.

$$H(f) = \alpha\, S^*(f) \exp(-j2\pi f T_d) \tag{2.21}$$

where $S^*(f)$ denotes the complex conjugate spectrum of the input signal $s(t)$. The scale factor α and the linear phase shift $\exp(-j2\pi f T_d)$ do not affect the signal-to-noise ratio (SNR) and may therefore be ignored. Thus,

$$H(f) = S^*(f) \tag{2.22}$$

In the time domain the corresponding relationship is obtained by taking the IFT of Eq. (2.22). This leads to the result that the impulse response of a MF is the mirror image of the complex conjugate of the transmitted signal $s(t)$, and the general relationship is given by

$$h(t) = \alpha\, s^*(T_d - t)$$

or simply

$$h(t) = s^*(-t) \tag{2.23}$$

However, in a digital radar receiver, the received waveform is sampled every T seconds. Thus the impulse response of the digital MF is given by the sequence

$$h(nT) = s^*(-nT)$$

or simply

$$h(n) = s^*(-n) \tag{2.24}$$

So far it has been assumed that the spectrum of the reflected signal is completely known. However, even for the simplest case of a single point target, the return signal contains two unknown parameters, doppler shift ν and time delay τ. In general the spectrum of the received signal is of the form

$$S(f) = S_o(f-\nu)\ \exp(-j2\pi f\tau - \theta) \tag{2.25}$$

where $S_o(f)$ denotes the spectrum of the transmitted signal. Therefore, for matched conditions, a receiver with the frequency characteristic, (neglecting a constant phase term and amplitude factor)

$$H(f) = S_o^*(f-\nu)\ \exp(j2\pi f\tau + \theta) \tag{2.26}$$

is required for optimum detection. It is clear that for stationary or slowly moving targets ($\nu \simeq o$) a receiver matched to the transmitted waveform is optimal*. However, since the doppler frequency depends on the range rate of the target and is not known beforehand, optimum reception of signals reflected from moving targets cannot be accomplished by only one matched filter. An optimum receiver in this case requires a bank of matched filters with incremented frequencies, $\Delta\nu$, of the doppler shift ν in the expected domain. This is illustrated in block diagram form in Fig. 2.7. The output of the matched filter bank is

*In the present context 'stationary' means that the duration of the signal is too short for the effects of target motion to be noticeable.

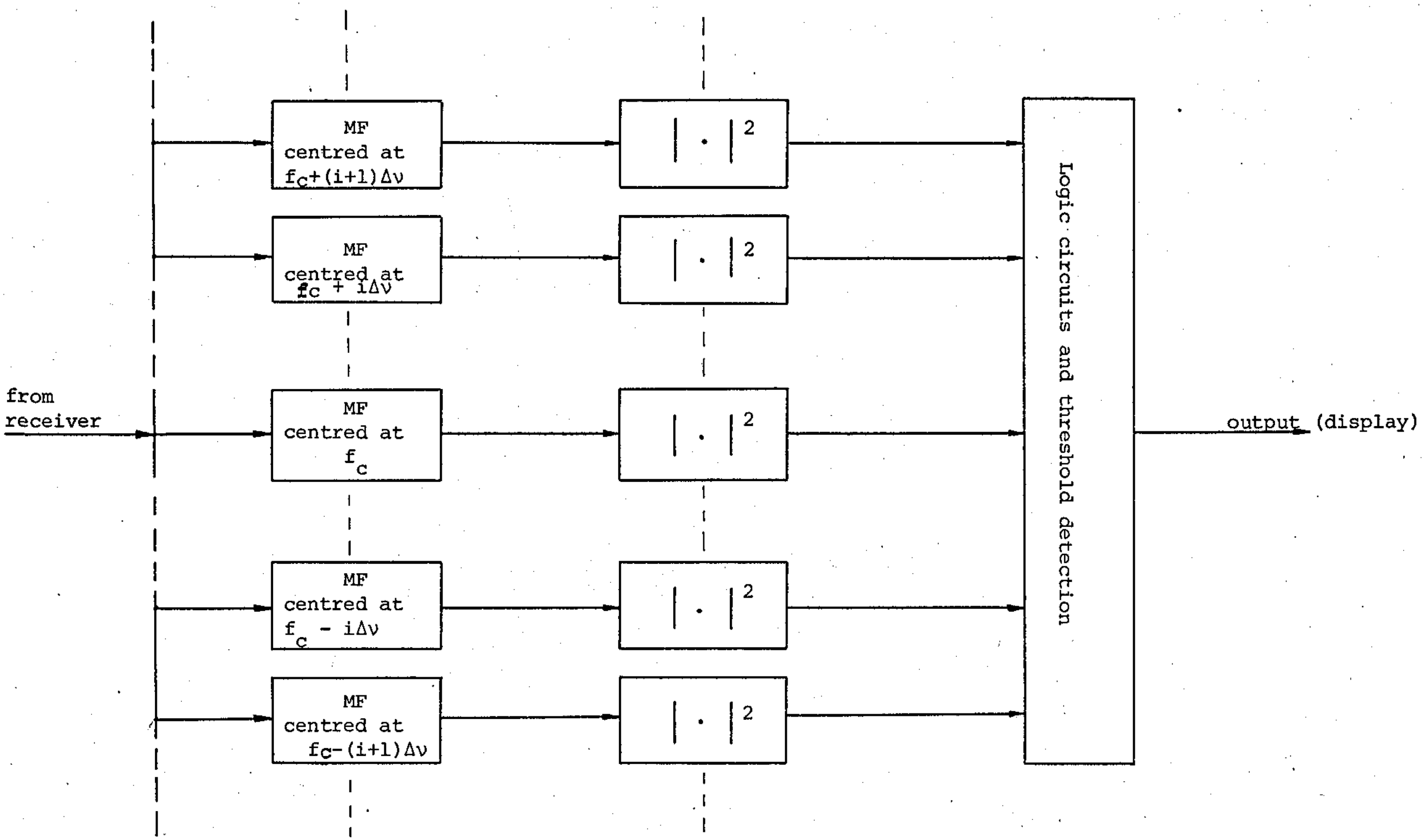


Fig. 2.7 Bank of parallel MF's.

usually applied to a square law or envelope detector and compared with a common threshold. If any of the outputs crosses the threshold a signal is deemed detected, and from the corresponding delay and doppler shift the target's range and velocity are estimated.

The response of the digital MF to an input signal reflected from a target, at range r and radial velocity v_r, is obtained by convolving the filter impulse response with the input signal.

$$\Psi(\tau,\nu) = \sum_{n=-\infty}^{\infty} \{s(nT-\tau)\, e^{-j2\pi\nu(nT-\tau)} + w(nT)\}\, s^*(-(mT-nT))$$

$$= e^{-j2\pi\nu(mT-\tau)} \sum_{n=-\infty}^{\infty} s((n+m)T-\tau)\, e^{-j2\pi\nu nT}\, s^*(nT)$$

$$+ \sum_{n=-\infty}^{\infty} w((n+m)T)\, s^*(nT) \qquad (2.27)$$

where

$$\tau = 2r/c \qquad \text{and} \qquad \nu = (2v_r/c)f_c$$

$$f_c = \text{carrier frequency}, c = \text{velocity of light}$$

The noise w(mT) is assumed to be a Gaussian random variable with zero mean and variance σ^2. Furthermore, w(mT) and w(kT), for any integer $k \neq m$, are uncorrelated and therefore statistically independent. Eq. (2.27) can be simplified by ignoring the non-essential phase factor, $\exp(-j2\pi\nu(mT-\tau))$, and by letting $mT-\tau=\tau'$. Thus,

$$\Psi(\tau',\nu) = \sum_{n=-\infty}^{\infty} s(nT+\tau')\, s^*(nT)\, e^{-j2\pi\nu nT} + \sum_{m=-\infty}^{\infty} w((n+m)T)s^*(nT) \qquad (2.28)$$

Although the input signal and the impulse response are discrete, it is noted that the function in Eq. (2.28) depends on the two continuous variables τ and ν.

Since the radar detector at the output of the MF usually removes the phase information, the function of interest is generally

$$|\Psi(\tau,\nu)|^2 = |\sum_{m=-\infty}^{\infty} s(nT+\tau)s^*(nT)\ e^{-j2\pi\nu nT}|^2 + |\sum_{n=-\infty}^{\infty} w((n+m)T)s^*(nT)|^2$$

$$+ \text{ cross-products} \tag{2.29}$$

The first term in the equation above is the signal term resulting only from the target reflection

$$|\chi(\tau,\nu)|^2 = |\sum_{n=-\infty}^{\infty} s(nT+\tau)s^*(nT)\ e^{-j2\pi\nu nT}|^2 \tag{2.30}$$

The function $|\chi(\tau,\nu)|^2$ is commonly referred to as the ambiguity function*. The investigation of the ambiguity function has been a field of extensive study since its introduction by Woodward[21]. The origin of the ambiguity function (τ=o, ν=o) may be thought of as the output of the matched filter tuned in time delay and frequency shift to the signal reflected from the target (point source) of interest. For zero relative doppler shift, $|\chi(\tau,o)|^2$ represents the squared magnitude of the autocorrelation function (ACF) of the transmitted signal. This is the filter response to reflections at a different range but at the same doppler as the target. Similarly, $|\chi(o,\nu)|^2$ is the response to reflections at the same range as the target but with other doppler shifts. Another property which reflects the fundamental constraint of radar signal design is the total volume under the ambiguity function. It is shown below that this volume is independent of the shape of the transmitted waveform.

$$V = \int_{-\infty}^{\infty} \int_{-1/2T}^{1/2T} |\chi(\tau,\nu)|^2\ d\tau\ d\nu \tag{2.31}$$

*In the literature the terms $\chi(\tau,\nu)$, $|\chi(\tau,\nu)|$ and $|\chi(\tau,\nu)|^2$ are often used synonymously as ambiguity function.

Substituting Eq. (2.30) into Eq. (2.31) and integrating first with respect to ν

$$V = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} s(nT+\tau) s^*(mT+\tau) s(mT)\, s^*(nT)$$

$$\times \frac{\sin \pi(n-m)}{T\pi(n-m)} \quad d\tau$$

Integrating with respect to τ and noting that the autocorrelation of the signal is given by

$$r((n-m)T) = \int_{-\infty}^{\infty} s^*((n-m)T+\tau) s(\tau)\; d\tau$$

leads to

$$V = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} s^*(nT) s(mT)\; r((n-m)T) \frac{\sin \pi(n-m)}{T\pi(n-m)}$$

since

$$\frac{1}{T} \frac{\sin \pi(n-m)}{\pi(n-m)} = \begin{cases} 0 & m \neq n \\ \frac{1}{T} & m = n \end{cases}$$

$$V = \frac{1}{T} r(o) \sum_{n=-\infty}^{\infty} |s(nT)|^2 = \frac{1}{T} r(o)\; E$$

$$V = \frac{1}{T}\; E^2 \tag{2.32}$$

where E denotes the signal energy.

Hence the volume under the ambiguity function depends only on the total signal energy. This implies that any reduction of ambiguity anywhere in the (τ,ν)-plane will cause it to appear elsewhere. Eq.(2.32) is particularly important in clutter and multiple target environments. The radar signal design problem can be considered, therefore, as a process of rearranging the undesired portions of the ambiguity function (range and doppler ambiguity away from the origin) into a region of little importance.

The second term in Eq. (2.29) is due to the white noise. The computation of the mean noise power is simplified by noting that the noise power is an uncorrelated zero mean, random Gaussian variable. Thus the expected value of the various cross-products in Eq. (2.29) are zero. Hence, the mean-square noise power at the output of the MF is given by

$$|\overline{N(\tau,\nu)}|^2 = E\{ \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} w(nT)\ s^*(nT+\tau)\ w^*(mT)\ s(mT+\tau) \times e^{-j2\pi\nu(n-m)T}\}$$

$$|\overline{N(\tau,\nu)}|^2 = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} E\{w(nT)\ w^*(mT) s^*(nT+\tau)\ s(mT+\tau) \times e^{-j2\pi\nu(n-m)T}\}$$

$$|\overline{N(\tau,\nu)}|^2 = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} r_w(nT,mT)\ s^*(nT+\tau)\ s(mT+\tau)\ e^{-j2\pi\nu(n-m)T}$$

where

$$r_w(nT,mT) = E\{w(nT)\ w^*(mT)\} = \sigma^2\delta((n-m)T)$$

hence

$$|\overline{N(\tau,\nu)}|^2 = \sigma^2 \sum_{n=-\infty}^{\infty} |s(nT)|^2 = \sigma^2\ E = N_o\ E \tag{2.33}$$

Thus for a given signal energy, the mean-square noise power at the output of the optimum receiver is a constant over the (τ,ν)-plane.

So far it has been implied that the two variables τ and ν are continuous. However, from practical considerations only signals of finite duration can be processed. It is therefore assumed that the received signal duration is $(N+1)T$ and that the delay τ and doppler shift ν are expressed as integer multiples of the sampling period T and fundamental frequency 1/(N+1)T, respectively. With these notations Eq. (2.30) can now be written as

$$|\chi(k,\ell)|^2 = |\sum_{n=o}^{N-|k|} s(n+k)\ s^*(n)\ e^{-j2\pi\ell n/(N+1)}|^2$$

$$k,\ell = 0, \underline{+}1, \underline{+}2, \ldots, \underline{+}N \tag{2.34}$$

In subsequent chapters attention will be focused on the case for slowly moving or stationary targets. In other words, the relative doppler spread of the targets is assumed to be negligible ($\nu \approx 0$). The output of the MF for zero doppler is the ACF of the transmitted signal and is given by

$$r(k) = \sum_{n=o}^{N-|k|} s(n+k)\ s^*(n) \tag{2.35}$$

$$k = 0, \underline{+}1, \underline{+}2, \ldots, \underline{+}N$$

An alternative way of representing the autocorrelation sequence r(k) above is to use the ZT technique described in Section 2.2.2. Thus, the ZT of Eq. (2.35) is given by

$$\begin{aligned} R(z) &= r(-N) + r(-N+1)\ z^{-1} + \ldots + r(0)\ z^{-N} + \ldots \\ &\qquad \ldots + r(N-1)\ z^{-2N+1} + r(N) z^{-2N} \\ &= (s^*(N) + s^*(N-1)\ z^{-1} + \ldots + s^*(0)\ z^{-N}) \\ &\qquad x\ (s(0) + s(1)\ z^{-1} + \ldots + s(N)\ z^{-N}) \\ &= z^{-N}\ (s^*(0) + s^*(1)\ z + \ldots + s^*(N)\ z^{N}) \\ &\qquad x\ (s(0) + s(1)\ z^{-1} + \ldots + s(N)\ z^{-N}) \end{aligned}$$

$$R(z) = z^{-N}\ S^*(1/z)\ S(z) \tag{2.36}$$

The coefficients of $R(z)$ are labelled with index values running from $-N$ to N. The kth coefficient of the complex envelope of the ACF at a time shift is given by Eq. (2.35)

From the foregoing it is clear that $R(z)$ is an even function which has the property of complex conjugate symmetry, that is

$$r(-n) = r^*(n) \tag{2.37}$$

The main response peak $r(o)$ given by

$$r(o) = \sum_{n=o}^{N} |s(n)|^2 = E \tag{2.38}$$

is always real and represents the energy contained in the sequence. Moreover, it can easily be shown that

$$r(o) \geqslant |r(n)| \quad n = 0, 1, 2, \ldots, N$$

The MF for discrete coded waveforms could be implemented using a tapped delay line as shown in Fig. 2.8. It is assumed that the input is at the RF carrier (or IF) and that each delay element is an integral number of wavelengths. In addition the sub-pulse matched filter must also be centered at that frequency[40]. Such a processor filters the input signal directly as a bandpass signal. However, for long sequences ($N+1 > 31$) the bandwidth of the delay line presents a practical problem in that its N cascaded stages must have an overall bandwidth $\geqslant 1/T$, the reciprocal of the sub-pulse duration. Therefore it is often preferable to process signals at baseband (zero IF or homodyne receiver) particularly when digital implementation is required.

In a typical digital processor the delay lines are replaced by digital memories, Fig. 2.10. In this configuration the RF or IF signals are heterodyned to zero carrier frequency with a single-sideband or quadrature mixer whose two video outputs represent the in-phase, I,

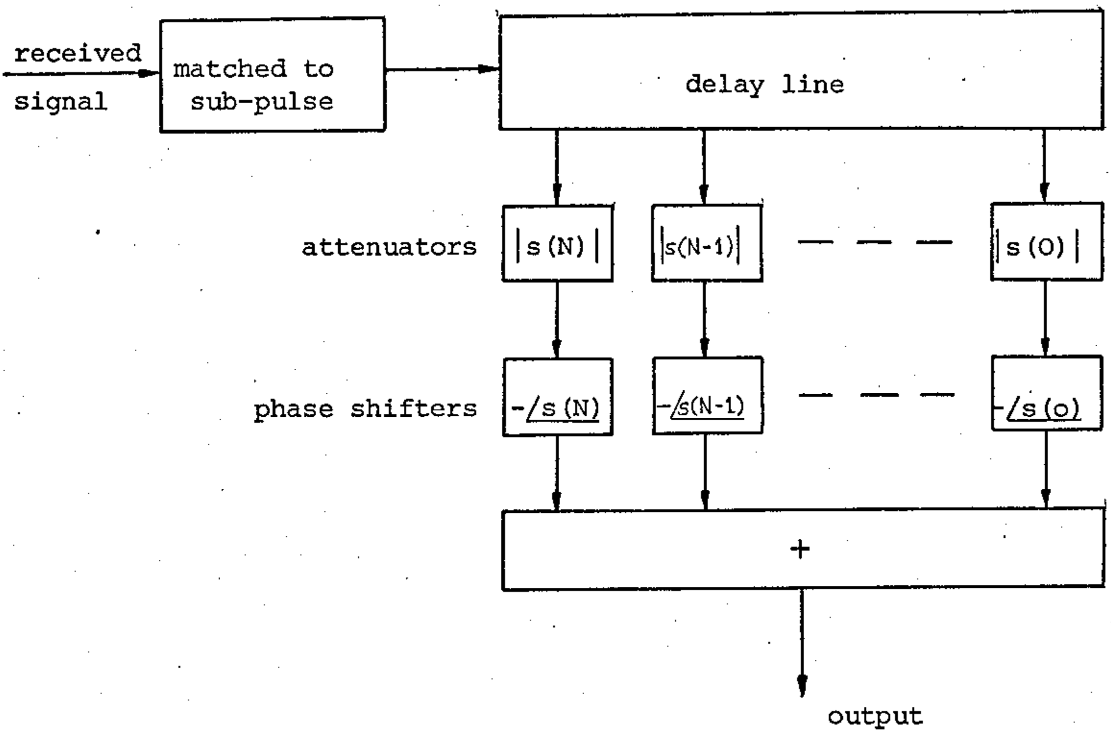


Fig. 2.8 Tapped delay line MF.

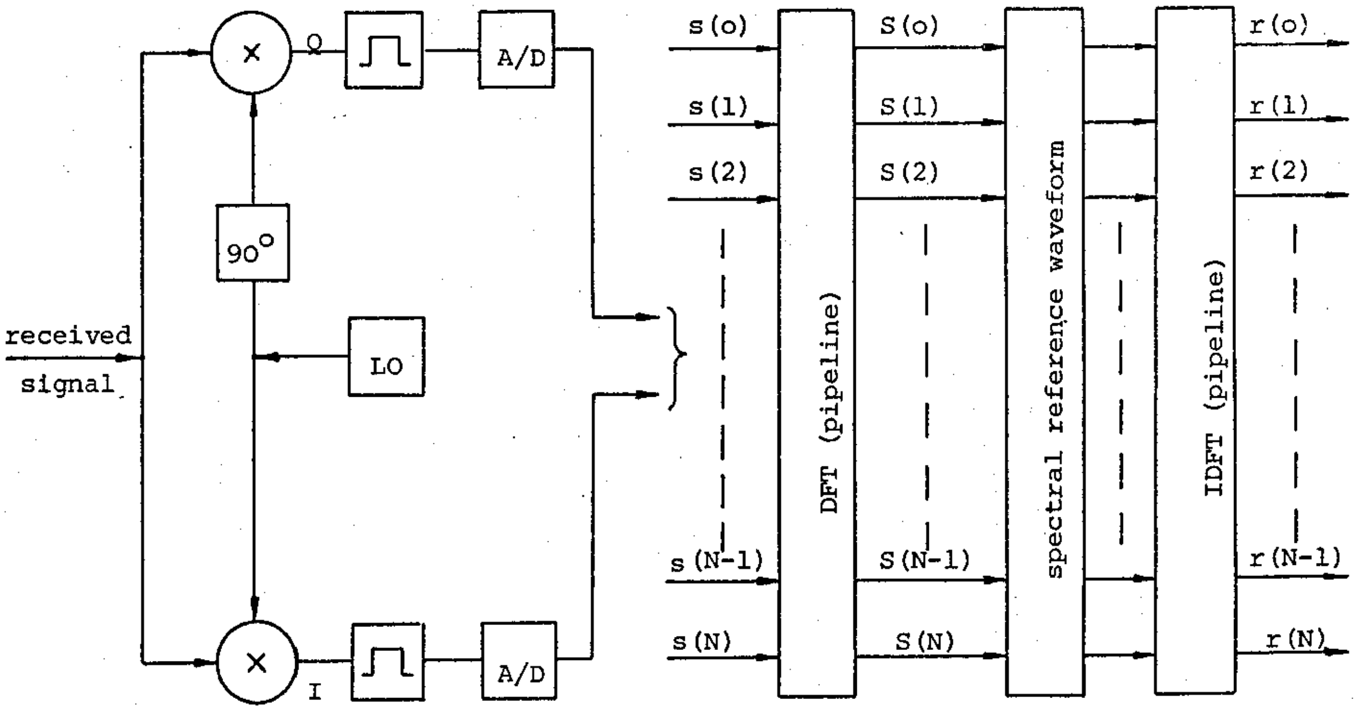


Fig. 2.9. DFT realization of MF.

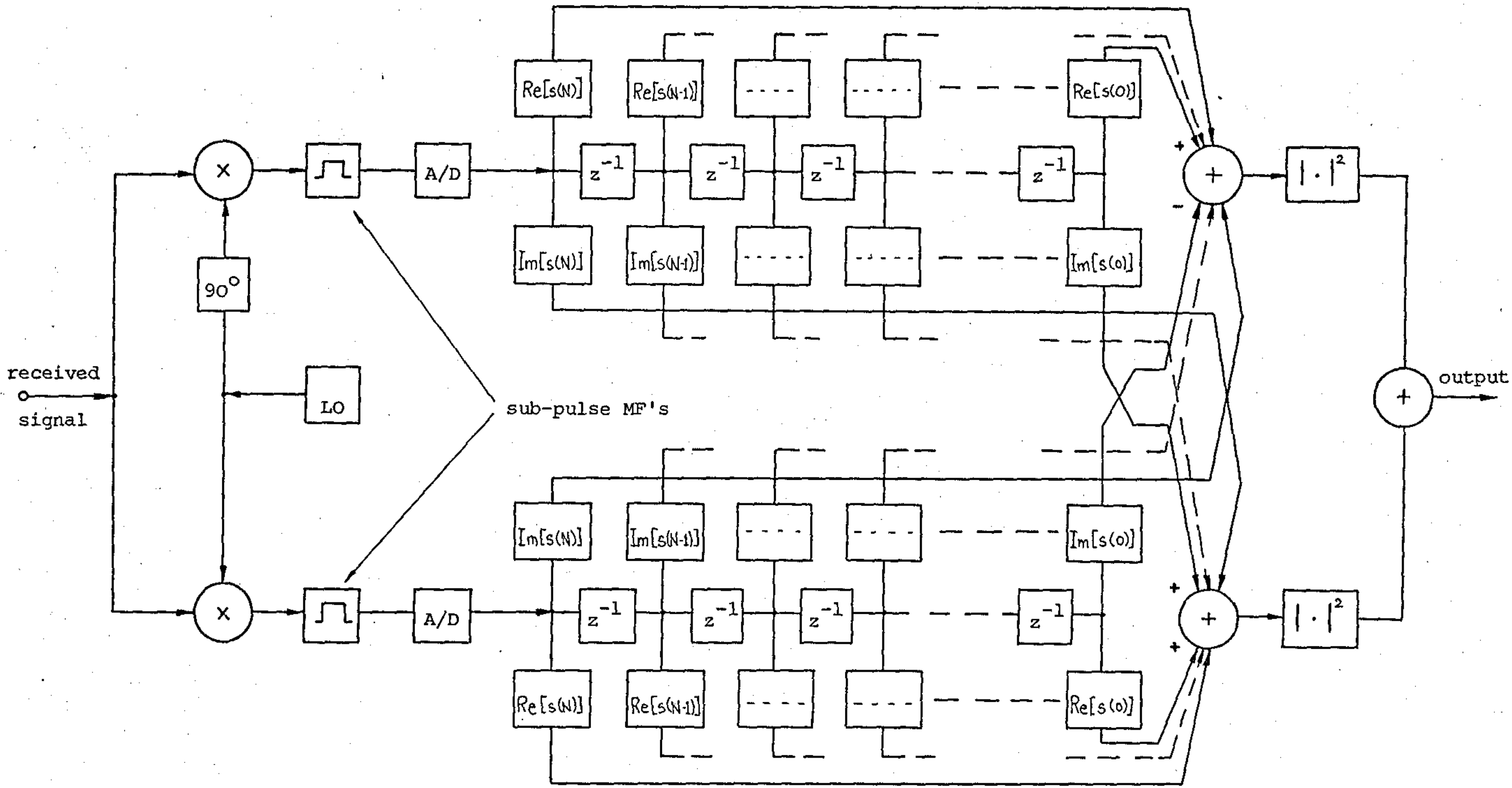


Fig. 2.10 Digital baseband MF.

and quadrature, Q, components of the signal (Sec. 2.2.1). It is even possible to add minor doppler shifts (if known) to the local oscillator (LO) to prevent degradation of the matched filter output. The baseband signals are then low pass filtered, sampled and converted to digital form at high speed. The digital matched filtering operation can be carried out using a digital tapped delay line (transversal filter) as shown in Fig. 2.10,or by the high speed pipeline FFT method[41-43] shown in Fig. 2.9. However, in most radar applications the FFT realization of the filter is more desirable. This is particularly the case for signals with a large time-bandwidth product, since FFT methods tend to be more efficient at performing convolution operations.

2.3.2 Pulse Compression Radar

It has been shown that, regardless of the signal shape, the MF produces one global maximum output value which is equal to the signal energy. Because of this concentration of the entire signal energy its detection against a white noise background is enhanced. In addition, to achieve high accuracy and resolution of radar measurements, it is required that the maximum in the MF output be as narrow as possible. Since the sharpness of the MF output signal (autocorrelation function) is inversely proportional to the r.m.s. signal bandwidth, compression of the received signal into a narrow spike can be accomplished provided the signal has a large bandwidth. Thus the essence of pulse compression radar systems is to provide this large bandwidth without degrading radar performance in other respects such as range resolution.

An obvious way to achieve a large bandwidth is simply to reduce the duration of the transmitted pulse. However, since target detectability and measurement precision depends on the signal energy, the transmitted power must be increased proportionally, to keep the energy constant. Unfortunately, the peak power limitations of transmitters sets a lower limit on the pulse duration. Therefore, the need for a large bandwidth must be met by modulating the pulse rather than reducing the pulse duration.

In principle any of the three basic types of modulation could be used to increase the signal bandwidth, namely; amplitude (AM), phase (PM) and frequency (FM) modulation. (Here, PM is considered as a general type of modulation where the phase of the signal is varied). AM modulation, however, is generally not desirable for use with radar waveforms due to its inherent disadvantages. First, it is an inefficient way of increasing the signal bandwidth, in that the function actually applied to the modulators must be wide-band. Secondly, as already pointed out, transmitter tubes operate most efficiently under constant amplitude conditions. Thirdly, it is expensive and difficult to achieve good amplitude linearity throughout the radar system over the entire dynamic range of interest. Therefore, AM modulation is of interest as a means to improve system performance, rather than as a primary method of achieving large signal bandwidth. The case of quantized FM will not be considered here as, due to complexity in frequency synthesizing it is less practical, except in a few cases, than PM.

As implied by the term pulse compression the objective of the receiver filter is to compress the received long pulse having a time duration of T_s seconds and a bandwidth of W hertz into a short pulse of duration 1/W, to allow recognition of closely spaced targets. The ratio of the duration of the long pulse to that of the short pulse is called the compression ratio. Thus the compression ratio is given by

$$m_c = T_s/(1/W) = T_s W \qquad (2.39)$$

which is equal to the time-bandwidth product of the waveform. In a digital system the signal duration T_s is equal to (N+1)T, where T is the sampling interval and (N+1) is the number of samples. Furthermore, if the sampling process is carried out at the Nyquist rate T = 1/W, then

$$m_c = (N+1)T\ W = N+1 \tag{2.40}$$

that is, the compression ratio or time-bandwidth product is equal to the total number of samples.

The main conclusion from the foregoing is that in all cases where the transmitted signal spectrum is substantially widened by modulation, recompression of signals can be accomplished during reception.

2.3.3 Range Resolution in a Matched Filter Radar

The estimation of target resolution performance is probably the most difficult problem to solve in modern high performance radar systems. In some cases the interfering objects may themselves be targets of interest, whereas in others they may be undesirable scatterers introducing a type of noise, known as clutter, into the system. As mentioned in Section 2.3.1 the optimum receiver for maximum resolution is not necessarily a MF. In practice, however, the typical target situation is too complex and not enough prior information is available to implement anything but a MF receiver or an approximation. For good target resolution, it has thus been necessary to retain a MF processor but to optimize the signal waveform so as to reduce the mutual interference (self-clutter) between targets.

The ways in which the transmitted waveform limits the radar performance in white noise are well known[21]. Three parameters are of prime importance. These are the bandwidth W, the time duration T_s, and the total signal energy E. Specifically:

(i) Range resolution for a MF receiver and stationary targets is determined by the spectrum envelope of the signal. For a given spectral shape, range resolution is proportional to 1/W. Therefore, good range resolution is achieved with a spectrum for which the total occupied frequency band is large.

(ii) Making use of the time-frequency duality, velocity resolution (radial) for targets having the same range is determined by the time structure or envelope of the signal. For a given envelope, velocity resolution is proportional to $1/T_s$. Hence low velocity ambiguity requires a waveform that occupies, with significant energy, a large total time interval.

(iii) Target detectability is determined by the ratio of received signal energy to received noise power (SNR). For given system parameters, signal detectability, and thus range, can only be improved by increasing the transmitted energy.

Therefore, it is desirable to transmit a waveform which has both a rectangular envelope as well as a rectangular spectrum.

In order to appreciate the resolution problem consider two stationary targets slightly separated in range. Neglecting an amplitude attenuation factor the combined received signal is of the form

$$s(nT) = s_o(nT-\tau_o) + s_o(nT-\tau_o-\tau) \qquad (2.41)$$

where

$$s_o(nT) = \text{transmitted waveform}$$

$$\tau_o = \text{position of first target}$$

$$\tau = \text{target separation}$$

To be able to distinguish the two target returns it is necessary to select a suitable waveform $s_o(nT)$. A measure of distinguishability, given by the sum of the square difference of the two signals, can be expressed as

$$\varepsilon^2 = \sum_{n=-\infty}^{\infty} \left| s_o(nT - \tau_o) - s(nT - \tau_o - \tau) \right|^2$$

$$\varepsilon^2 = \sum_{n=-\infty}^{\infty} \{ |s(nT-\tau_o)|^2 + |s(nT-\tau_o-\tau)|^2 - 2\mathrm{Re}\left[s(nT-\tau_o) \; s^*(nT-\tau_o-\tau) \right] \}$$

The first two terms represent the signal energy E and are therefore constants. The last term is recognized as the ACF, $r(\tau)$. For the two signal returns to be as different as possible it is required to maximize the equation above, that is

$$\max \varepsilon^2 = \{E - \mathrm{Re}\left[r(\tau) \right]\} \tag{2.42}$$

Hence, in a MF radar the optimum waveform to use would be one whose receiver response, or ACF, has an envelope consisting of a single spike at $\tau = 0$ of a width smaller than the spreading of the targets in range (delay). However, in practice such waveforms cannot usually be realized. Actual waveforms have ACF's whose envelopes show one or more of the properties indicated in Fig. 2.11. Basically there are three types of MF responses; a single lobe (a), a narrow main lobe accompanied by relatively large sidelobes, (b), and a single lobe surrounded by a noise-like, low-level response spread out in time (c). Each one of these response types presents its own resolution problems. Suppose the

MF output consists of a single lobe (a). If the separation of the targets is larger than the width of the lobe there will be no problem in resolving them. However, for closely spaced targets the responses overlap and the envelope of the combined output will depend on the phase relationship of the echoes, Fig. 2.12. Thus, in the region of overlap,target resolution is difficult to achieve. Therefore, to distinguish two or more targets their returns must be separated by at least the half power (3dB) response width. A different type of resolution difficulty is caused by the MF response shown in Fig. 2.11(b). Although the main lobe may be sufficiently small to meet the required target resolution, target returns separated at multiples of $\Delta\tau_d$ are completely masked. Problems of yet a different nature are introduced by the response of type (c). Again the main lobe may be narrow enough for the desired close target resolution. However, for targets with widely varying cross-sections it still may not be possible to distinguish them. The pedestal-like extension of the response due to a strong target may have an amplitude strong enough to obscure the main response peak of weaker targets. This effect is aggravated particularly in a multiple-target environment where the combined sidelobes from many returns may build up to a level that even relatively strong targets can no longer be recognized.

As shown by Woodward[21] the MF receiver utilizes the full information available from the return signal. The width of the main response lobe can be regarded as a measure of uncertainty about the exact target range, while the spread of the response introduces ambiguity of the target location. Both effects, although conceptually different, are lumped together in a figure of merit known as the time resolution constant[21].

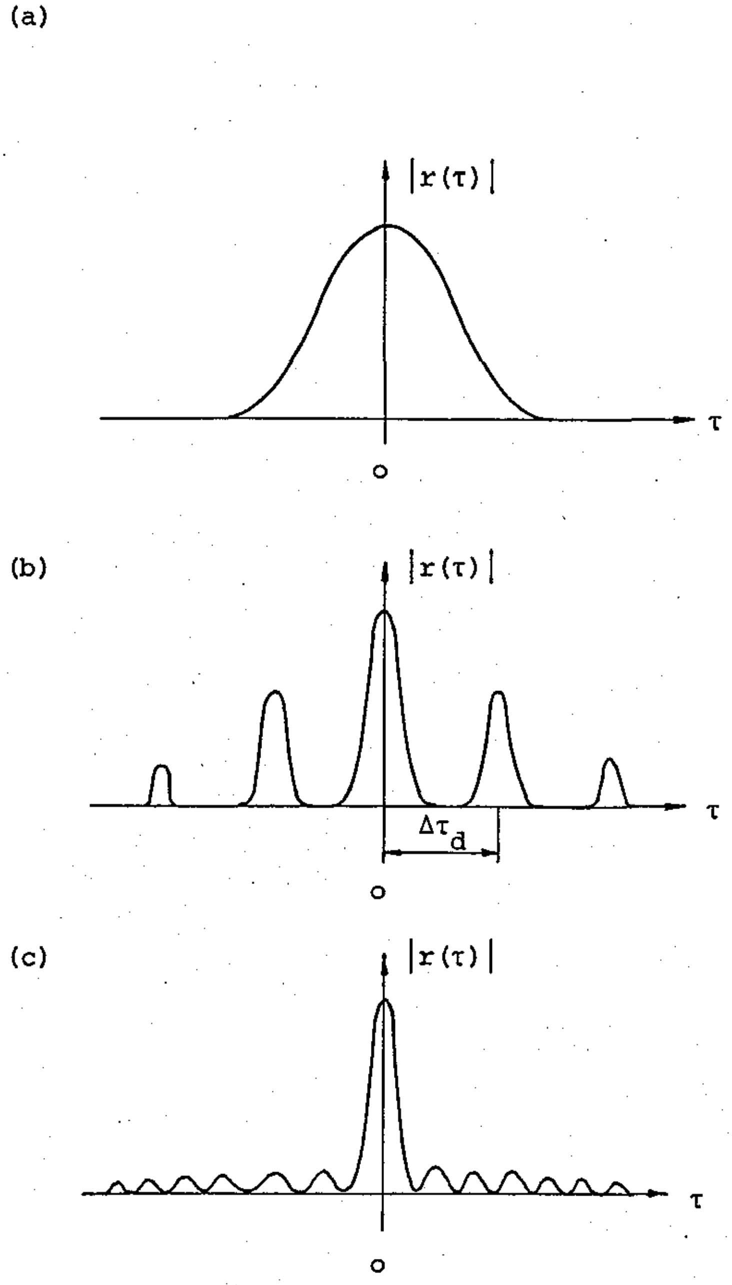


Fig. 2.11 Forms of range ambiguities.

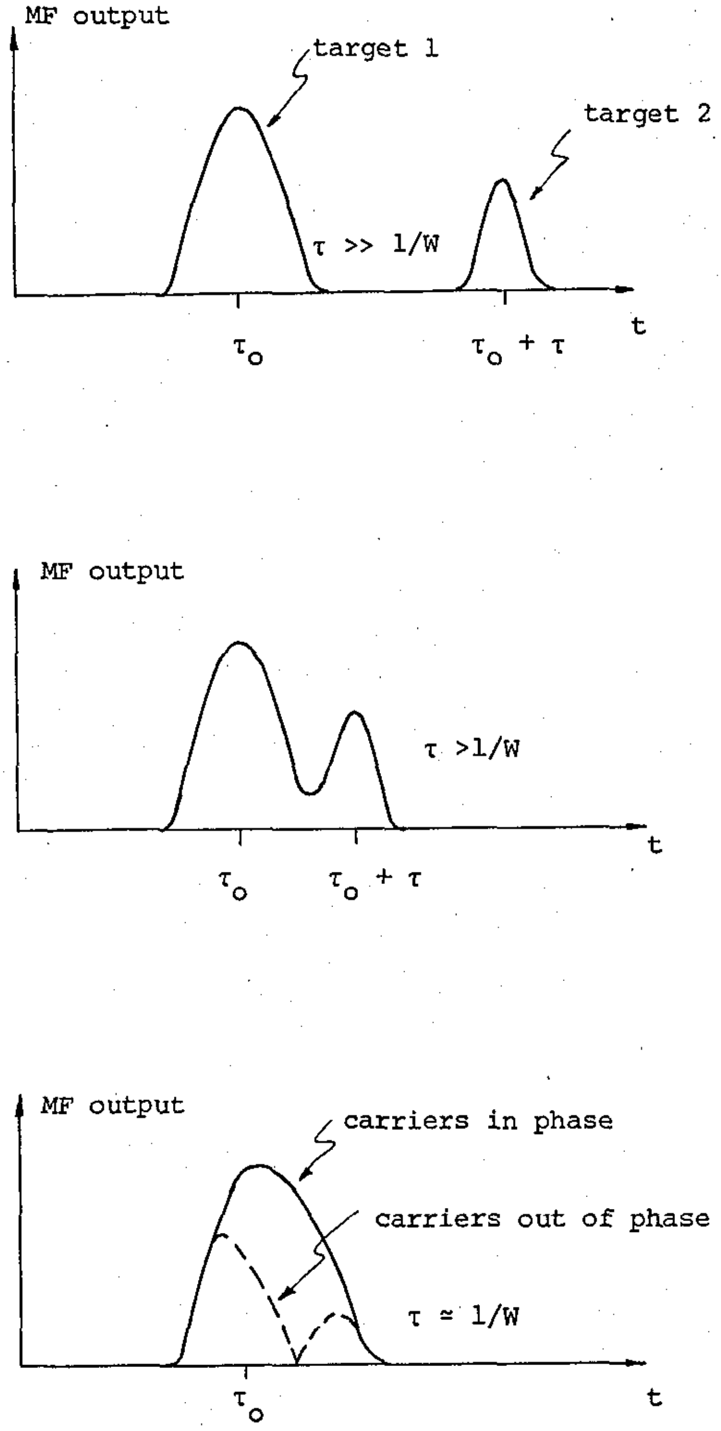


Fig. 2.12 Overlap of two target returns.

$$\Delta\tau = \frac{\int_{-\infty}^{\infty} |r(\tau)|^2 d\tau}{|r(o)|^2} \qquad (2.43)$$

The resolution problem has been discussed here in a purely qualitative manner. Analytical treatments of the resolution in parameters can be found in many excellent radar books[22,23,26,44]. Nevertheless, the preceeding discussion allows the general formulation of the requirements which have to be met for target resolution. First, the output signal to white noise ratio must be large enough for reliable detection. Secondly, the target of interest must be separated sufficiently from any other target of comparable or larger cross-section to prevent overlap of the main response lobes. Thirdly, the combined interference from other targets must not be so strong as to mask the target return of interest.

Interference from other targets due to response sidelobes acts like clutter caused by undesired scatterers. However, the term clutter implies that the interference causing reflectors are so dense that they cannot be resolved. The type of clutter due to sidelobes is often called self-clutter to distinguish it from the effect of undesired objects.

In summary, the resolution performance of a radar thus depends not only on the width of the main response lobe but also on the low-level response surrounding the main peak. In the published literature resolution is often referred to as the 3dB points of the main response lobe. In the present context, however, target resolution means the ability to recognize a target in the presence of others.

CHAPTER 3

SYNTHESIS OF PULSE TRAINS FROM SPECIFIED AUTOCORRELATION FUNCTIONS

3.1 Introduction

The problem of synthesizing signals which realise the desired autocorrelation function (ACF) can be divided into a number of interrelated problems, each of which has an independent practical significance. The separate problems could be formulated as follows[44]:

(i) Determine the class of functions which are realizable ACF's for arbitrary signals.

(ii) Determine the sub-class of ACF's for various signal structures such as discrete coded waveforms.

(iii) Synthesis of signals whose ACF is a close approximation to the desired ACF not belonging to the class of realizable functions.

(iv) Synthesis of signals (pulse trains) which satisfy a given set of requirements, e.g., range resolution, energy utilization, etc.

So far, a comprehensive analytical treatment of these problems and their solution has not been formulated. For example, a simple criterion for determining the realizability of an ACF has not yet been found.

To appreciate the nature of the problem arising here consider the ACF

$$r(\tau) = \sum_{n=-\infty}^{\infty} s(nT+\tau)\ s^*(nT) \tag{3.1}$$

or its equivalent expression

$$r(\tau) = \int_{-W/2}^{W/2} |S(f)|^2\ e^{j2\pi f\tau}\ df \tag{3.2}$$

where $|S(f)|^2$ is the power spectrum of the signal s(nT) and is assumed to be bandlimited. In other words S(f) is zero outside some range (-W/2,W/2). Strictly speaking the requirement for a finite bandwidth W is incompatible with a finite-duration signal. However, an approximation to the finite spectrum condition can be reached if a major portion of the signal energy is concentrated within a specified frequency band.

From Eq. (3.2) it can be seen that the ACF and the power spectrum form a Fourier transform pair. Hence it follows that for $r'(\tau)$ to be a realizable ACF its spectrum, R(f), must be real and non-negative. Even if the given ACF is realizable the synthesis problem cannot be solved uniquely. Since the phase information is lost in the power spectrum, it is not possible to determine S(f) itself which is necessary to find s(nT). Therefore, all signals whose spectra differ only in phase will have the same ACF. Thus the synthesis problem may be divided into the following two steps:

1. The power spectrum $|S(f)|^2$ is determined from the given ACF. (It is assumed that the ACF is realizable, i.e., $R(f) = FT\ \{r(\tau)\} \geq 0$, if not R(f) is replaced by $|R(f)|$).
2. From the determined power spectrum one signal having such a spectrum is derived by assigning an arbitrary phase function $\theta(f)$, i.e., $s(nT) = IFT\ \{|S(f)|\exp(j\theta(f))\}$.

For digital applications, however, only finite-length sequences can be processed. The next section will, therefore, be devoted to the problem of factorizing the power spectrum using ZT techniques.

3.2 Synthesis of Pulse Trains if the ACF is known in Magnitude and Phase

If the ACF is given in phase and magnitude at discrete points its ZT can be written as (Eq. (2.36))

$$R(z) = z^{-N}\ S(z)\ S^*(1/z)$$

As mentioned previously the ZT provides a convenient method to represent a signal in the form of its zero pattern which is obtained by factorizing its polynomial in z. In factorized form the equivalent representation of a polynomial S(z) of order N in powers of z^{-1} can be written as

$$S(z) = s(0) \prod_{i=1}^{N} (1 - z_i/z) \tag{3.3}$$

where z_i are the zeros of S(z), i.e. $S(z_i) = 0$, $i = 1,2,\ldots.,N$. Similarly, S*(1/z) can be represented as

$$S^*(1/z) = s^*(0) \prod_{i=1}^{N} (1 - z\, z_i^*) \tag{3.4}$$

$$= s^*(0)\ (-1)^N \left(\prod_{i=1}^{N} z_i^*\right) \prod_{i=1}^{N} (z-1/z_i^*)$$

$$S^*(1/z) = s^*(N) \prod_{i=1}^{N} (z-1/z_i^*)$$

Hence

$$R(z) = s(0)\ s^*(N) \prod_{i=1}^{N} (1-z_i/z)(z-1/z_i^*)$$

where the unessential delay factor z^{-N} has been neglected. Since R(z) essentially represents a power spectrum the above equation can be regarded as the factorized power spectrum[45].

The equivalent expressions above allow the study of pulse trains using their zero patterns in the complex z-plane. The conditions $s(0) \neq 0$, and $s(N) \neq 0$, are clearly equivalent to

$$S(0) \neq 0 \qquad \text{and} \qquad S^*(0) \neq 0$$

It is easy to verify that if $S^o(z)$ denotes the polynomial

$$S^o(z) = z^{-N}\, S^*(1/z)$$

then

$$(S^o(z))^o = S(z) \tag{3.5}$$

and

$$(S(z)\ P(z))^{o} = S^{o}(z)\ P^{o}(z) \tag{3.6}$$

In addition it follows from Eq. (3.3) and Eq. (3.4) that for $|z| = 1$

$$|S(z)| = |S^{*}(1/z)| \tag{3.7}$$

Moreover, if a polynomial S(z) of degree N has p_1 zeros inside the unit circle, $|z| = 1$ (counting multiples), p_2 on the unit circle and p_3 zeros outside, where $p_1+p_2+p_3 = N$, it is referred to as of the type (p_1,p_2,p_3). Since it has been assumed that $S(0) \neq 0$, it is clear that z_j is a zero of S(z) if $1/z_j^*$ is a zero of $S^*(1/z)$. The zeros z_j and $1/z_j^*$ have the same angle in the z-plane but reciprocal magnitudes as indicated in Fig. 3.1. It is clear that S(z) is of the type (p_1,p_2,p_3) if $S^*(1/z)$ is of type (p_3,p_2,p_1).

The class of polynomials, P, for which P(z) and $P^*(1/z)$ have the same set of zeros are known as self-inversive polynomials[46]. It is apparent that a polynomial P(z) is self-inversive if its zeros are symmetric with respect to inversion on the unit circle. From the foregoing it should be clear that:

(i) A self-inversive polynomial of degree N is of type $(p,N-2p,p)$ for $p \geqslant 0$.

(ii) Since the polynomial R(z) consists of the product of the two factors S(z) and $S^*(1/z)$ it is of type $(p_1+p_3,2p_2,p_1+p_3)$, where $2(p_1+p_2+p_3) = 2N$. Consequently, R(z) is self-inversive and its zeros must occur in reciprocal conjugate pairs.

Thus for a finite pulse train to be an ACF it has to satisfy condition (ii). The design technique for pulse trains from a given realizable ACF can now be summarized as follows:

1. Factorization of the ZT polynomial which represents the ACF.

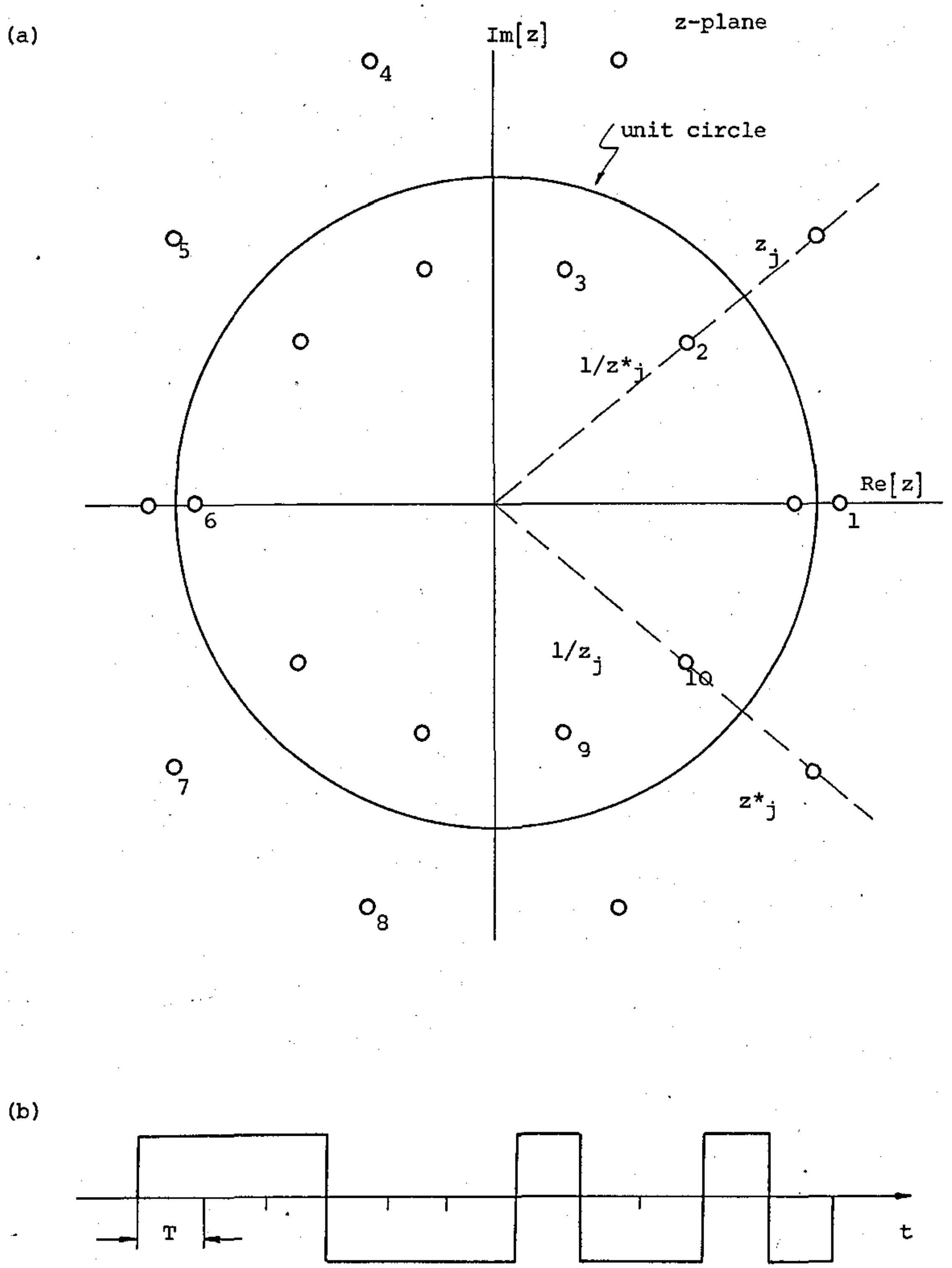


Fig. 3.1 (a) Factorized ACF of 11-element Barker code

(b) Resulting sequence when choosing the zero pattern 1,2,3,...,10.

2. Selection of a suitable zero pattern to obtain the signal after multiplication.

This synthesis method is not unique in that a whole family of pulse trains having the same ACF usually exists. For example, an ACF polynomial R(z) of order 2N has exactly 2N zeros. Half of these zeros belong to S(z) while the other half belongs to S*(1/z). Hence there are 2^N possible zero patterns and thus 2^N pulse trains having the same ACF (see Chapter 7).

The design procedure is probably best illustrated by an example. Consider the 11-element Barker code* whose ACF has the ZT representation

$$R(z) = -1-z^{-2}-z^{-4}-z^{-6}-z^{-8}+11z^{-10}-z^{-12}-z^{-14}-z^{-16}-z^{-18}-z^{-20}$$

This polynomial is now factorized on a digital computer using a standard root-finding algorithm. The resulting zeros are given in Fig. 3.1(a). All the zeros occur in reciprocal conjugate pairs. In addition as a consequence of the coefficients of R(z) being real, all complex zeros must occur in conjugate pairs. The selection of the zeros for S(z) and multiplying them out completes the design procedure. The resulting sequence choosing the zeros labelled as 1,2,3,....,10 is shown in Fig. 3.1(b). More detailed information on the selection and structure of zero patterns and how they affect the complex envelope of the resulting pulse train is presented in Chapter 7 when dealing with Huffman codes.

*The 11-element Barker code is given by (1,1,1,-1,-1,-1,1,-1,-1,1,-1)

3.3 Synthesis of Pulse Trains if only the Magnitude of the ACF is known

In the previous section the magnitude and phase of the ACF at discrete points was required to find a solution to the synthesis problem. However, from practical considerations only the magnitude of the function is usually known, since the phase does not affect the accuracy and resolution of the range measurements. In this section the design procedure is extended to the case where only the magnitude of a realizable ACF is given at discrete points. The basic underlying idea of the method presented here is due to Vakman[44] and is also implied by Voelcker[47] in a different context.

Before proceeding further it is necessary to recall the convolution theorem derived from basic Fourier transform theory

$$r(t) * r^*(-t) \longleftrightarrow |R(f)|^2 \tag{3.8}$$

$$|r(t)|^2 \longleftrightarrow R(f) * R^*(-f) \tag{3.9}$$

The above relationships show the duality between the ACF of the spectrum R(f) and the ACF of the time signal r(t). As r(t) is assumed to be bandlimited it can be represented by its Nyquist samples. However, due to the convolution process in the frequency domain the squared envelope, $|r(t)|^2$, will have twice the bandwidth of r(t)[29]. In other words, if $|r(t)|^2$ is sampled at the Nyquist rate, r(t) is sampled at twice that rate. Hence it is assumed that the squared envelope of the ACF is known at integer multiples of T/2 = T'.

Since $|r(t)|^2$ is of finite duration it is completely defined by frequency domain samples taken at intervals 1/T'. Such a signal has a finite Fourier representation of N terms

$$m(t) = |r(t)|^2 = \sum_{k=o}^{N-1} C(k) \exp(j2\pi kt/T_s) \tag{3.10}$$

where T_s is the duration of the signal and $N = WT_s$.

The samples of the envelope are thus given by

$$m(n/W) = \sum_{k=o}^{N-1} C(k) \exp(j2\pi kn/N) \qquad (3.11)$$

$$n = 0,1,2,\ldots.,(N-1)$$

By substituting the symbol z for $\exp(j2\pi t/T_s)$ the finite Fourier series can be rewritten as a polynomial in powers of z

$$m(z) = C(0) + C(1)\ z + \ldots. + C(N-1)z^{N-1}$$

$$m(z) = \sum_{k=o}^{N-1} C(k)\ z^k \qquad (3.12)$$

This transformation can be regarded as the dual of the ZT discussed in Chapter 2.

Consider now the polynomial of order L representing the ACF, r(t)

$$r(z) = R(0) + R(1)\ z + \ldots. + R(L)\ z^L \qquad (3.13)$$

Clearly, for r(z) to be realizable all R(n) must be real and non-negative ($R(n) \geqslant 0$), since the coefficients of the polynomial are the power spectral samples. The squared modulus of r(t), $|r(t)|^2$ can thus be represented as a polynomial multiplication

$$m(z) = (R(0) + R(1)\ z + \ldots. + R(L)\ z^L)\ x$$

$$(R^*(L) + R^*(L-1)\ z + \ldots. + R^*(0)\ z^L)$$

$$m(z) = z^L(R(0) + R(1)z + \ldots. +R(L)z^L)(R^*(0) + R^*(1)z^{-1}+\ldots.+R^*(L)z^{-L})$$

$$m(z) = z^L\ r(z)\ r^*(1/z) \qquad (3.14)$$

where the coefficients of m(z) are given by

$$C(k) = \sum_{n=o}^{L-|k|} R(n)\ R(n+k)$$

$$k = 0,1,\ldots.,\pm L$$

Since r(z) is a polynomial with real coefficients

$$m(z) = z^{L} r(z) r(1/z)$$

Thus the operation of convolution in the frequency domain reduces to a multiplication of two polynomials. This clearly reflects the duality of time and frequency as pointed out earlier.

Since m(z) has the form of an ACF it is possible to proceed in a similar manner to that described in Section 3.1 in order to find the power spectral components R(n) given m(z). The properties of m(z) are revealed by studying the zeros in the complex z-plane. The coefficients of m(z) specify the ACF of the spectrum of r(z). Its 2L zeros must therefore occur in reciprocal conjugate pairs. In addition, since all coefficients are real (and in particular non-negative) they all occur in complex conjugate pairs. Thus if z_j is a zero of m(z), then z_j^*, $1/z_j^*$ and $1/z_j$ also must be zeros of m(z). This relationship is illustrated in Fig. 3.1(a). Consequently, if m(z) is to represent a realizable power spectrum ACF the following conditions must be satisfied:

(i) m(z) is finite and its zeros occur in complex conjugate reciprocal pairs.

(ii) The coefficients, R(n), of r(z) must be real and non-negative.

If these conditions are met then at least one and in general a whole set of ACF's having the same magnitude can be found.

The steps in the design procedure can be summarized and probably best illustrated by using the 7-element Barker code as an example*.

*The 7-element Barker code is (1,1,1,-1,-1,1,-1)

1. Given the samples of $|r(t)|$, where $r(t)$ is assumed to be bandlimited and sampled at twice the Nyquist rate, Fig. 3.2(a), one computes the DFT of the sequence $|r(kT')|^2$, i.e.

   $$\text{DFT } \{(|r(O)|^2,|r(T')|^2,....|r(13T')|^2,|r(13T')|^2,...,|r(T')|^2)\}$$

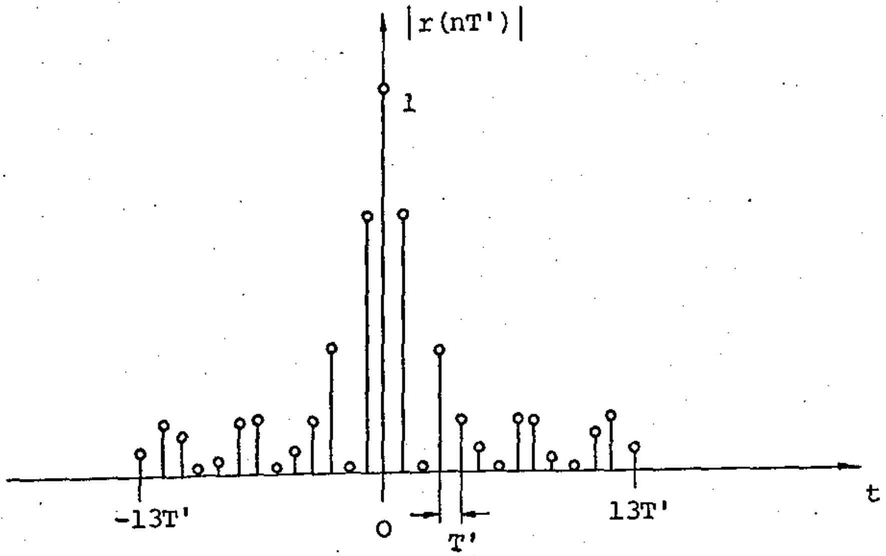
   This gives, except for a scale factor, the (2L+1) Fourier coefficients of the periodically repeated square envelope of the ACF, Fig. 3.2(b).

2. Factorize the polynomial whose coefficients are these Fourier components. The 2L roots should occur in reciprocal complex conjugate pairs, Fig. 3.2(c).

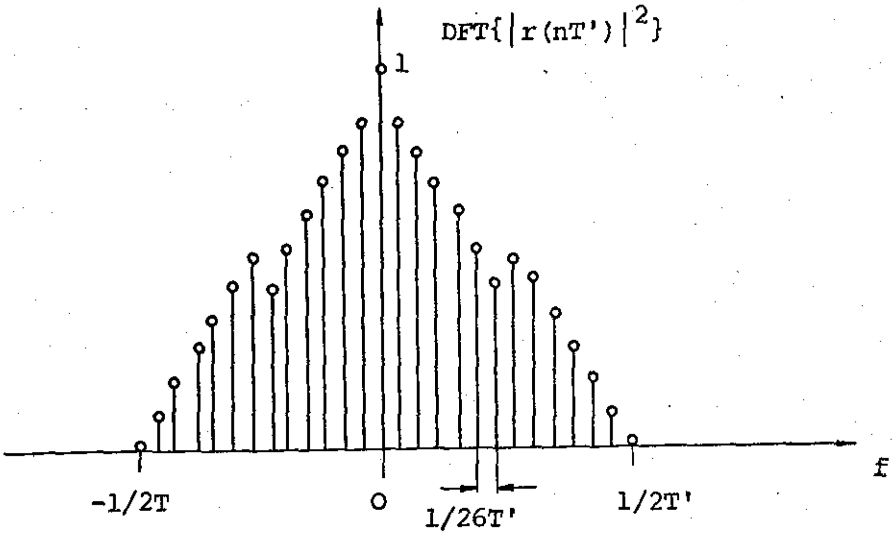
3. Select L roots from each reciprocal conjugate pair and its conjugate. Now, multiply out to obtain a set of (L+1) Fourier coefficients which are, neglecting a scale factor, the DFT of the samples of $r(t)$. Verify that $r(t)$ is indeed an ACF. This is done simply by making sure that the coefficients obtained are all real and non-negative. If the test fails, select a new zero pattern and repeat the precedure from step 3, until a realizable ACF is obtained, Fig. 3.2(d). From the L roots only L/2 can be chosen independently, since the zeros must be selected in complex conjugate pairs. Hence, there are in general $2^{L/2}$ possible zero patterns. However, not all zero combinations will result in realizable ACF's. The zeros chosen in this case are labelled 1,2,....,13, Fig. 3.2(c).

4. The final stage of the synthesis procedure is to take the IDFT of the Fourier coefficients to obtain the

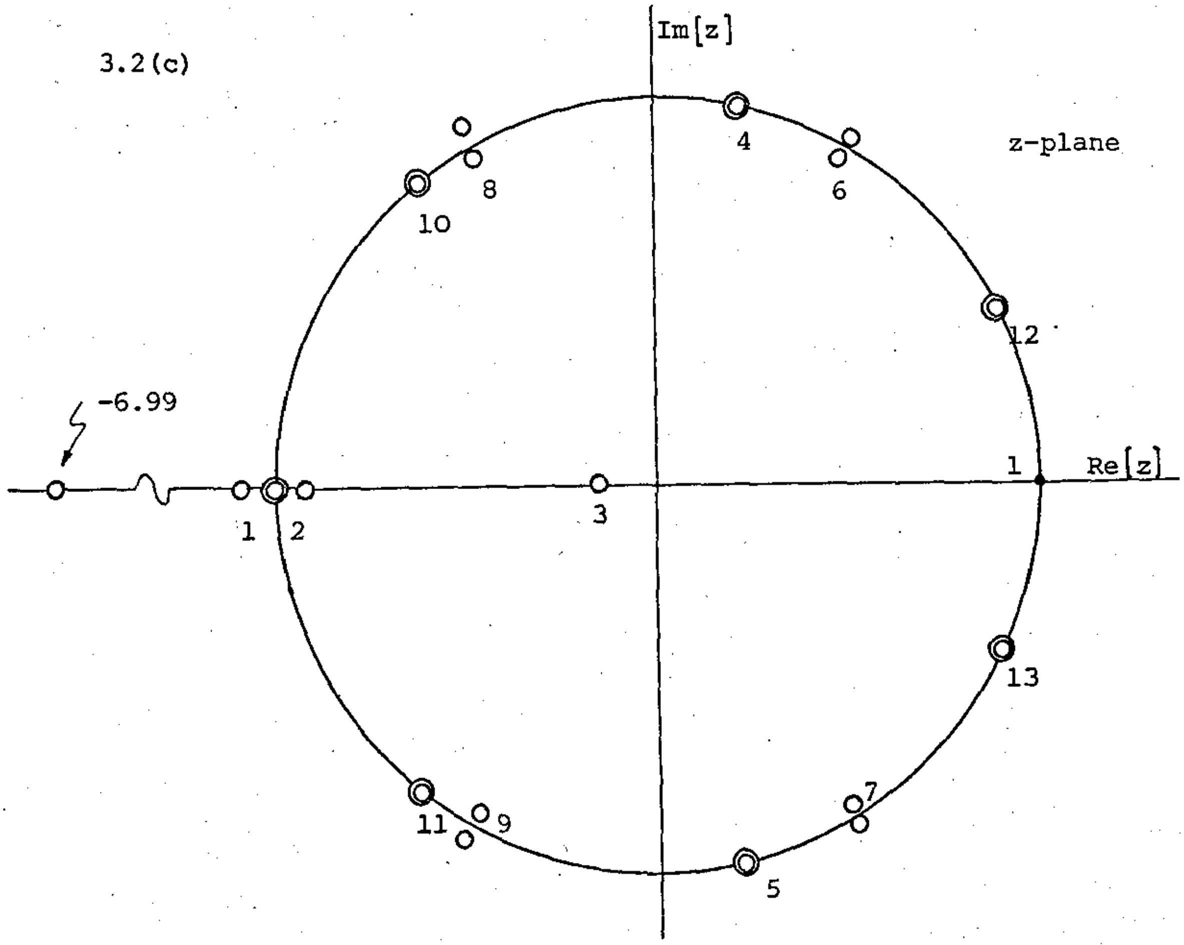
3.2(a)


|r(nT')|
1
-13T'
0
T'
13T'
t

3.2(b)


DFT{|r(nT')|²}
1
-1/2T
0
1/26T'
1/2T'
f

3.2(c)


Im[z]
z-plane
4
6
8
10
12
-6.99
1
2
3
1
Re[z]
13
7
11
9
5

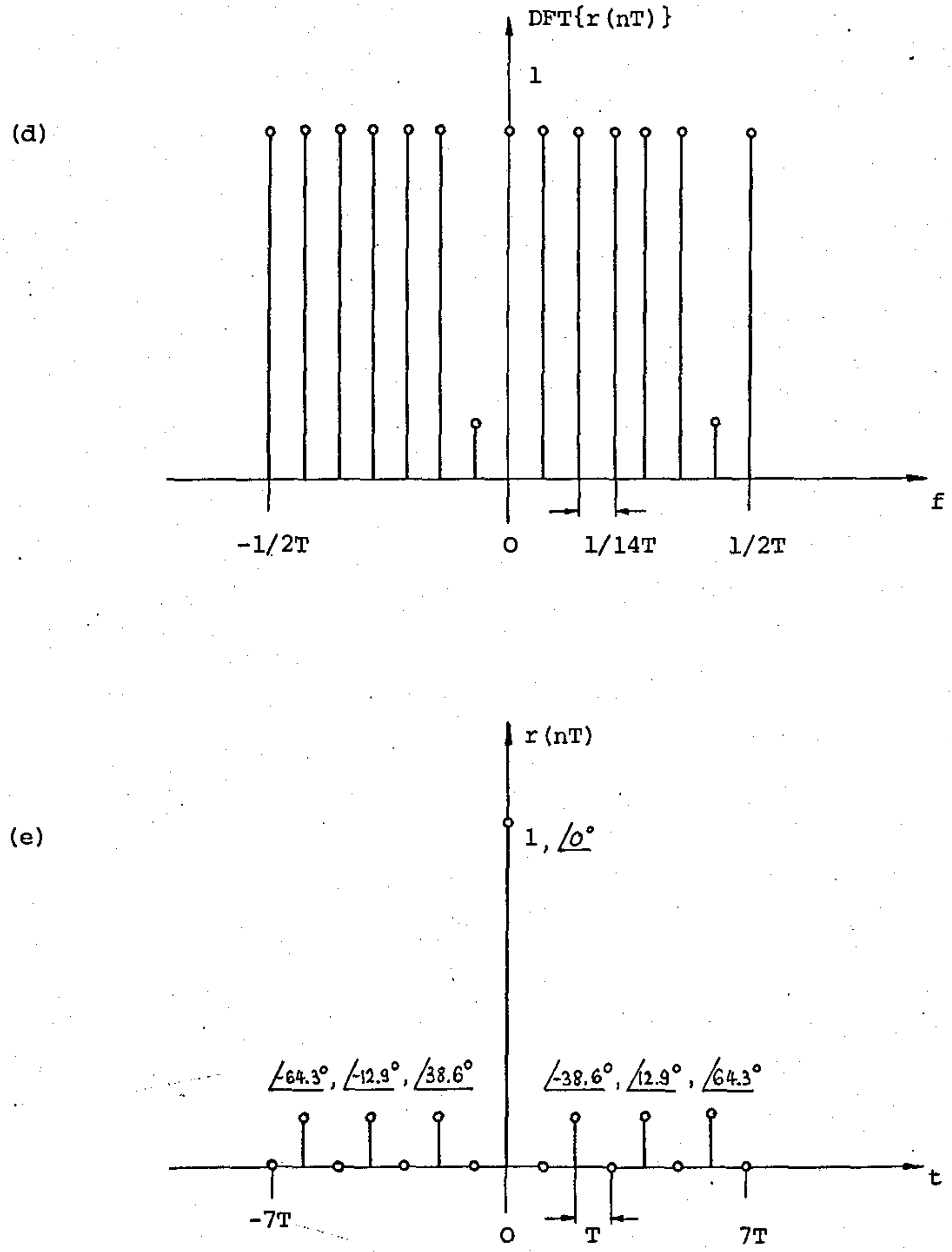


Fig. 3.2 (a) Samples of the ACF magnitude

(b) Spectrum samples of squared envelope of (a)

(c) Zero pattern of (b)

(d) Spectrum samples of the ACF

(e) ACF in magnitude and phase.

sampled values of r(t), Fig. 3.2(e). Once the ACF has been found in magnitude and phase, it is then straightforward to synthesize a pulse train which realizes this ACF by following the procedure outlined in Section 3.2.

3.4 Summary

The design procedure outlined in this chapter may appear to solve the synthesis problem satisfactorily. However, in practice this solution has limited applications for the following reasons. In the first place, an exact solution can only be found in those cases where the given ACF is realizable, that is, there must exist a signal which, if correlated with its conjugate time reverse, produces that ACF. In many practical cases only the gross features of the ACF are known. Usually when a signal is to be synthesized very little is known about the sought after ACF. In general only the following requirements should be met. First, to improve the accuracy and resolution of the range measurements, the mainlobe of the ACF must be as narrow as possible. Secondly, to increase the multiple-target resolution capability, the function must have low sidelobes. In most cases more detailed information about the ACF is not available. Hence, the given ACF's usually turn out to be non-realizable, i.e., their spectra do not satisfy the condition (3.2). It becomes necessary, therefore, to replace the spectrum R(f) by its magnitude $|R(f)|$ which is a non-negative function and thus to seek an approximation to the desired ACF. For example, the ideal ACF from the viewpoint of range resolution consists of a single narrow spike which is not realizable with a finite pulse train.

Additional difficulties of computational nature are encountered in the task of factorising the polynomial to obtain its zeros. The present NAGF* library root-finding algorithm of the Loughborough University computer centre run on an ICL 1904A digital computer can handle polynomials of orders up to about one hundred. Another difficulty is to compute the coefficients of a polynomial from its zeros. Standard methods usually give excessive error accumulation with polynomials of orders greater than about 40. A method which is to some extent free of that difficulty has been suggested by Ackroyd[48].

Although the difficulty in factorizing a polynomial can be avoided in the case of Huffman sequences[20] (Chapter 7), the problem of selecting a zero pattern in order to obtain a pulse train with specified properties still remains. The relationship between the zeros and the coefficients of a polynomial are given by a set of non-linear equations (Eq. (7.11), Chapter 7). For a given set of zeros it is difficult, if not impossible in practice, to solve this set of equations for any arbitrary constraints on the coefficients.

The shortcomings of this design method seriously limit its application from the practical point of view. Subsequently, an essentially different approach will be considered in the next chapter.

*Nottingham Algorithm Group

CHAPTER 4

DISCRETE PHASE APPROXIMATION TO FM SIGNALS

4.1 Introduction

With radar signal processing becoming more and more digital it will become increasingly attractive to consider phase modulated pulse train approximations of continuous FM signals, such as linear and non-linear FM. Therefore, in this chapter attention will be concentrated on the application of the design procedures used for continuous FM waveforms to pulse trains.

The problem of designing a long-duration, rectangular FM signal having an arbitrary ACF has been solved by Key et al[49]. Later Fowle[50] extended the theory for the non-rectangular case. However, as pointed out earlier, the rectangular envelope of the transmitted signal offers a number of advantages.

The application of the principle of stationary phase (Appendix A) to the class of FM signals with large time-bandwidth product has shown that a non-linear FM modulating function can control the form of the spectrum amplitude and thus the time response at the matched filter output[49,50]. Especially for applications where the transmitted signal envelope must be essentially rectangular the FM modulating function can be used to control the range sidelobes. This permits the receiver amplitude response to be matched directly to the transmitted waveform, thus eliminating receiver mismatch loss associated with range sidelobe reduction techniques (Chapter 8).

The effect of spectrum shaping is illustrated in Fig. 4.1. The two spectra curves representing the fourth power of the spectrum are assumed to have equal area. The resulting total ambiguity A is, therefore, in both cases the same, since

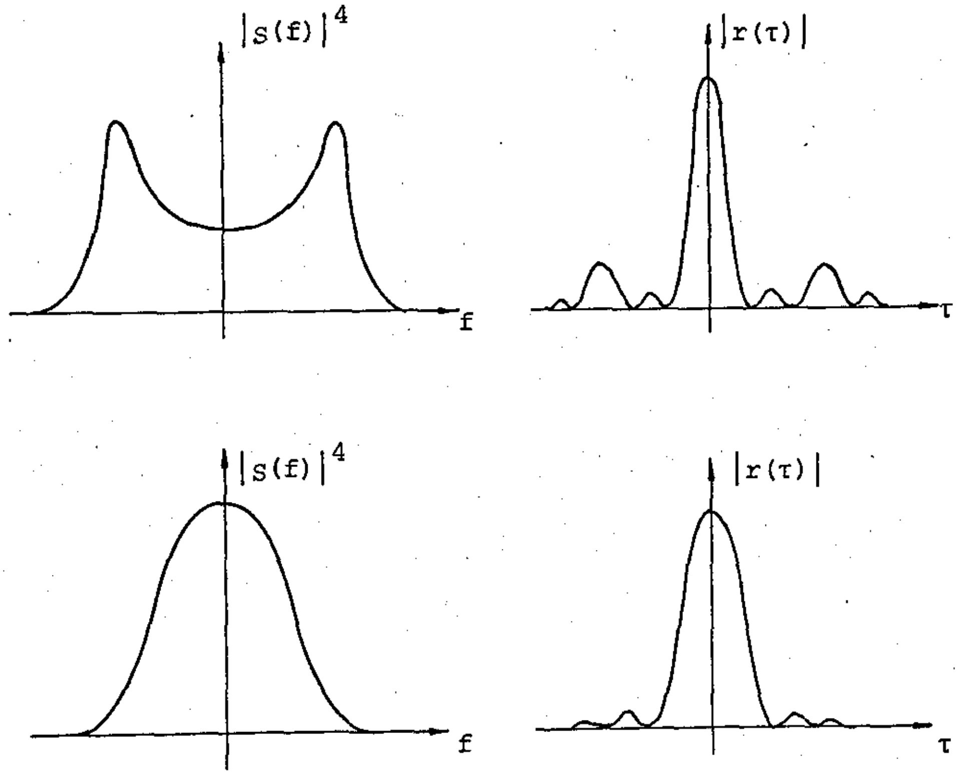


Fig. 4.1 Effect of spectral shaping on the ACF.

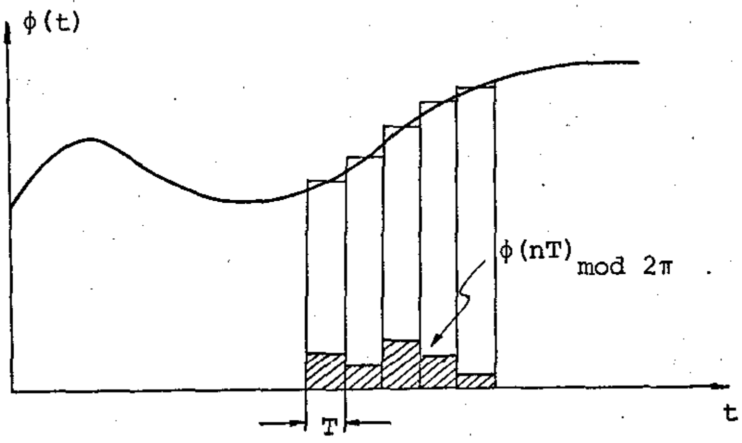


Fig. 4.2 Approximation to a continuous phase function.

$$A = \int_{-\infty}^{\infty} |r(\tau)|^2 \, d\tau \tag{4.1}$$

can be expressed using Parseval's theorem as

$$A = \int_{-W/2}^{W/2} |S(f)|^4 \, df \tag{4.2}$$

The spectrum with the energy shifted away from the origin will give rise to range ambiguities of the disconnected type. On the other hand, a spectrum whose energy is concentrated about the origin results in sidelobes close to the main response peak. This is confirmed by the Fourier transform theory where sharp edges of the spectrum produce large time sidelobes.

4.2 The Method of Stationary Phase applied to FM Signals

In general the problem of designing a FM signal of a given envelope shape to have a specified ACF requires the solution of

$$S_m(f)\, e^{j\,\theta(f)} \longleftrightarrow s_m(t)\, e^{j\,\phi(t)} \tag{4.3}$$

where $S_m(f) = |S(f)|$ and $s_m(t) = |s(t)|$.

With the two moduli functions $S_m(f)$ and $s_m(t)$ specified, it is necessary to find expressions for $\phi(t)$ and $\theta(f)$ in terms of the given quantities to satisfy the Fourier transform relationship. However, it is not generally possible to find a solution (or a reasonable close approximation) if the two moduli functions are specified. Unfortunately, a concise set of necessary and sufficient conditions which have to be met by the moduli of a Fourier pair, has so far not been formulated. Nevertheless, Fowle[50] has shown that when the time-bandwidth product is sufficiently large (>>1) approximate Fourier pairs with the independently specified moduli can be constructed.

The unusual type of integral equation implied in Eq. (4.3) may be approximated using the method of stationary phase (Appendix A). This technique provides a parametric expression for the spectrum of a FM signal[50].

$$S(f) \simeq (2\pi)^{\frac{1}{2}} \frac{s_m(\xi)}{(|\phi''(\xi)|)^{\frac{1}{2}}} \quad \exp\{j(-2\pi f\xi + \phi(\xi) \underline{+}\pi/4)\}$$

$$S_m(f) \simeq (2\pi)^{\frac{1}{2}} \quad s_m(\xi)/(|\phi''(\xi)|)^{\frac{1}{2}} \tag{4.4}$$

where

$$\phi''(t) = \frac{d^2\ \phi(t)}{dt^2}$$

The stationary point is given where

$$\frac{d}{dt}\{\ \phi(t)\ -\ 2\pi ft\} = 0$$

$$\phi'(\xi)\ -\ 2\pi f\ \ = 0 \tag{4.5}$$

The value of t satisfying Eq. (4.5) is denoted by ξ. At this point it is perhaps of interest to note that the stationary phase method yields only those frequency components of the spectrum which coincide with the instantaneous frequency of the signal for its duration. This dependence of the spectrum on the phase function and its derivatives is indicated in Eq. (4.4). As shown by Fowle[50] this relationship can be rewritten in the form

$$\int_{-\infty}^{t} s_m^2(\xi)\ d\xi \ = \int_{-\infty}^{f} S_m^2(\eta)\ d\eta \tag{4.6}$$

The above equation establishes the relationship between the instantaneous frequency, f, and the envelope function $s_m(\xi)$, with time t.

Carrying out the integration in Eq. (4.6) leads to

$$P(t) = Q(f) \tag{4.7}$$

where P(t), Q(f) are indefinite integrals of $s_m^2(\xi)$ and $S_m^2(\eta)$ respectively (assuming $P(-\infty) = Q(-\infty) = 0$). The solution of Eq. (4.7) for f is symbolized as

$$f = Q^{-1}\{P(t)\} \tag{4.8}$$

Substituting Eq. (4.8) into Eq. (4.5) yields

$$\phi'(t) = 2\pi\, Q^{-1}\{P(t)\}$$

Thus

$$\phi(t) = 2\pi \int Q^{-1}\{P(t)\}\, dt + C_1 \tag{4.9}$$

where C_1 is the constant of integration.

Eq. (4.9) makes it possible to synthesize FM signals having any given power spectrum.

A similar relationship can be shown to exist for θ(f), which leads directly to the result that the group delay function θ'(f) and the instantaneous frequency ϕ'(t) are approximate inverse functions[50].

The design procedure for large time-bandwidth product FM signals can thus be summarized as follows:

1. Given a power spectrum $|S(f)|^2$ the integration in Eq. (4.6) is performed, assuming a rectangular envelope ($s_m(t) = 1$), to find Q(f) = t

2. The function Q(f) = t is inverted to obtain

   $$\phi'(t) = 2\pi\, Q^{-1}(t)$$

   This is done quite easily on a digital computer.

3. Integration of ϕ'(t) leads to the sought after phase modulation function.

The relationships above were obtained using the stationary phase principle and are thus approximate. For a rigorous formulation, the problem requires the solution of the integral equation

$$S(f) = \int_{-T_S/2}^{T_S/2} s_m(t) \exp\{j(\phi(t) - 2\pi ft)\}dt \tag{4.10}$$

for the unknown phase function $\phi(t)$ or the instantaneous frequency $\phi'(t)$ with the spectrum magnitude $|S(f)|$ and the envelope $s_m(t)$ given. Moreover, the approximation of the stationary phase method is suited only for signals having a steep, monotone FM law, because it does not consider the 'edge effects' associated with the signal fronts. However, within its limitations this method has proved very useful in designing analogue FM waveforms.

Discrete phase coded waveforms on the other hand cannot be found directly with the help of the method of stationary phase, Eq. (4.4), since discontinuities of the phase modulating function leads to several points of stationary phase. However, it is anticipated that approximating the continuous phase function $\phi(t)$ with piecewise constant segments of equal duration will result in phase modulated pulse trains having virtually all the desired properties of FM signals. The process of approximation is illustrated in Fig. 4.2. Each sub-pulse of the pulse train is at the same carrier frequency. That would not be the case if the phase function $\phi(t)$ is approximated by straight line segments. Since phase shifts of a multiple of 2π are not effective, the stepped phase function of Fig. 4.2 can be replaced with its modulo-2π version as indicated. This method of approximation can also be visualized as being a sampling process of the complex envelope of a purely phase modulated analogue signal. Consequently, the study of such pulse trains and their ability to improve target resolution is the main objective in the following sections of this chapter.

4.3 Discrete Phase Approximation to Linear FM Signals

The linear FM (LFM) or 'chirp' waveform is probably the principal type of signal transmitted by a radar or sonar system. The LFM waveform is derived from Eq. (4.4) by assuming a rectangular power spectrum. The main advantage in using such waveforms lies in their ease of generation and insensitivity to small doppler shifts. The description and properties of analogue chirp techniques are well documented in the literature and will not be repeated here[22,25,51].

In general the complex envelope of a LFM signal can be expressed as

$$s(t) = \exp(j\ \mu t^2/2) \tag{4.11}$$

where

$$\mu = 2\pi W/T_s$$

$$W = f_2 - f_1 = \text{frequency change during sweep}$$

$$T_s = \text{time duration of sweep}$$

As implied by the term LFM, the instantaneous frequency is swept linearly from f_1 at $t = 0$, to a maximum value of f_2 at $t = T_s$. The complex envelope can be written in terms of the pulse compression ratio (time-bandwidth product), m_c

$$s(t) = \exp(j\pi(Wt)^2/m_c)$$

If the waveform is sampled at uniform time intervals of T seconds

$$s(nT) = \exp(j\pi(WnT)^2/m_c)$$

and with

$$m_c = NTW$$

$$s(nT) = \exp(j\pi WTn^2/N) \tag{4.12}$$

$$n = 0,1,2,\ldots,(N-1)$$

The total phase change over the signal duration is

$$\Delta\phi = \pi WT(N-1)^2/N$$

Furthermore, if T is set equal to the Nyquist rate, then

$$T = 1/W$$

and

$$s(nT) = \exp(j\pi n^2/N) \tag{4.13}$$

Since each segment is coded into one of N possible phases, these sequences are sometimes referred to as polyphase codes. In particular the sequences whose phase follows a quadratic progression will be called quadratic phase (QP) codes. An interesting property of QP codes is their periodic ACF which is zero for all non-zero time lags,[52] provided the sequence is coded as in Eq. (4.13) if N is even and is modified to

$$s(nT) = \exp(j\pi n(n+1)/N) \tag{4.14}$$

for N odd.

Higher order polyphase codes with zero circular ACF ($k \neq 0$) have been described by Frank, Zadoff and Heimiller[16-18]. The length N of such codes, however, is restricted to perfect squares. While the QP sequences and Frank codes have ideal cyclic autocorrelation, they do not have, of course, perfect aperiodic autocorrelation.

Another property of practical importance is the simple generation of QP pulse trains if the sequence length is chosen properly. This can be demonstrated by expanding the expression n^2/N as

$$n^2/N = q + q_1(\tfrac{1}{2}) + \alpha_n/N$$

where

$$q = 0,1,2,\ldots\ldots.$$

$$q_1 = 0 \text{ or } 1$$

$$\alpha_n = \text{remainder}$$

since

$$\exp(jq\pi) = \begin{cases} -1 \quad , & q \text{ odd} \\ 1 \quad , & q \text{ even} \end{cases}$$

and

$$\exp(j\pi/2) = j$$

the QP code can also be written as

$$s(nT) = (\underline{+}) \ (\text{or } \frac{1}{j}) \ \exp(j\pi\alpha_n/N) \qquad (4.15)$$

The number of different samples to be generated is thus a function of the number of distinct remainders of n^2/N. Roy and Lowenschuss[53] have shown that for a proper choice of N the number of different samples can be kept very small indeed. For example for N = 16 only three different values must be generated, $\exp(j\pi/16)$, $\exp(j\pi/4)$ and 1. Incidentally, this property has also been exploited in the Blustein algorithm which computes the DFT using a chirp filter[54].

4.3.1 Properties of the Compressed Pulse Train

The exact expression for the ACF can be obtained by substituting Eq. (4.12) into Eq. (3.1)

$$r(kT) = \sum_{n=o}^{N-1-k} e^{j\pi WT/N\,(n^2-(n+k)^2)}$$

$$= e^{j\pi WTk^2/N} \sum_{n=o}^{N-1-k} e^{-j2\pi WTnk/N}$$

$$k = 0,1,2,....,(N-1)$$

The summation in the last expression is of the form of a geometric progression and can be written in closed form. By rewriting the sum term as

$$\sum_{n=o}^{N-1-k} r^n$$

where

$$r = e^{-j2\pi WTk/N}$$

it is recognized that the series containing a total of (N-k) terms has a sum of

$$S = \frac{r^{(N-k)}-1}{r-1}$$

Therefore

$$r(kT) = e^{-j\pi WTk^2/N} \left\{ \frac{e^{-j2\pi WTk(N-k)} - 1}{e^{-j2\pi WTk/N} - 1} \right\}$$

$$= e^{-j\pi WTk^2/N} \; \frac{e^{-j\pi WTk(N-k)/N}}{e^{-j\pi WTk/N}} \left\{ \frac{e^{-j\pi WTk(N-k)/N} - e^{j\pi WTk(N-k)/N}}{e^{-j\pi WTk/N} - e^{j\pi WTk/N}} \right\}$$

$$r(kT) = e^{-j\pi WTk(N-1)/N} \; \frac{\sin\left[\pi WTk(N-k)/N\right]}{\sin\,(\pi WTk/N)} \tag{4.16}$$

Since only the magnitude of the ACF is of interest

$$|r(kT)| = \left| \frac{\sin\left[\pi WTk(1-k/N)\right]}{\sin(\pi WTk/N)} \right|$$

If T is equal to the Nyquist sampling rate, i.e. WT = 1, Eq. (4.16) becomes

$$|r(kT)| = \left| \frac{\sin\left[\pi k(1-k/N)\right]}{\sin(\pi k/N)} \right| \tag{4.17}$$

Because r(kT) exhibits complex conjugate symmetry with respect to k=O, it is sufficient to consider only positive time lags. The nature of the function (Eq. (4.17)), in the vicinity of k=O has the form of a sinc function with a peak value of N. Because of the periodicity of the expression this characteristic will be repeated at intervals I/N. For even length sequences the function is symmetrical with respect to N/2. The effect of the term (1-k/N) can be explained, considering that

$$|\sin[\,\pi k(1-k/N)\,]| = |\sin(\pi k^2/N)|$$

Thus it modulates the frequency of the ripples in a 'chirp-like' fashion (Fig. 4.3).

Whenever a sequence is used as a modulating function it is always of interest to consider its spectrum. Since the power spectrum contains the relevant information, it is sufficient to consider the amplitude spectrum only. For convenience the spectrum is assumed to be zero outside some range (O,W) (analytical signal). This is no restriction, since any bandlimited signal can be brought into this form by a suitable frequency translation.

The magnitude of the ACF, amplitude spectrum and group delay function are displayed in Fig. 4.3 for a QP sequence of length N = 128 when sampled at the Nyquist rate, WT=1. All graphs were plotted using straight line interpolation between adjacent samples. The spectrum was computed by evaluating Eq. (2.15) using the FFT algorithm[32,33]. The group delay function defined as

$$\tau_g(f) = \frac{1}{2\pi}\frac{d}{df}\ \theta(f) \tag{4.18}$$

was calculated with the help of the expression (Appendix B)

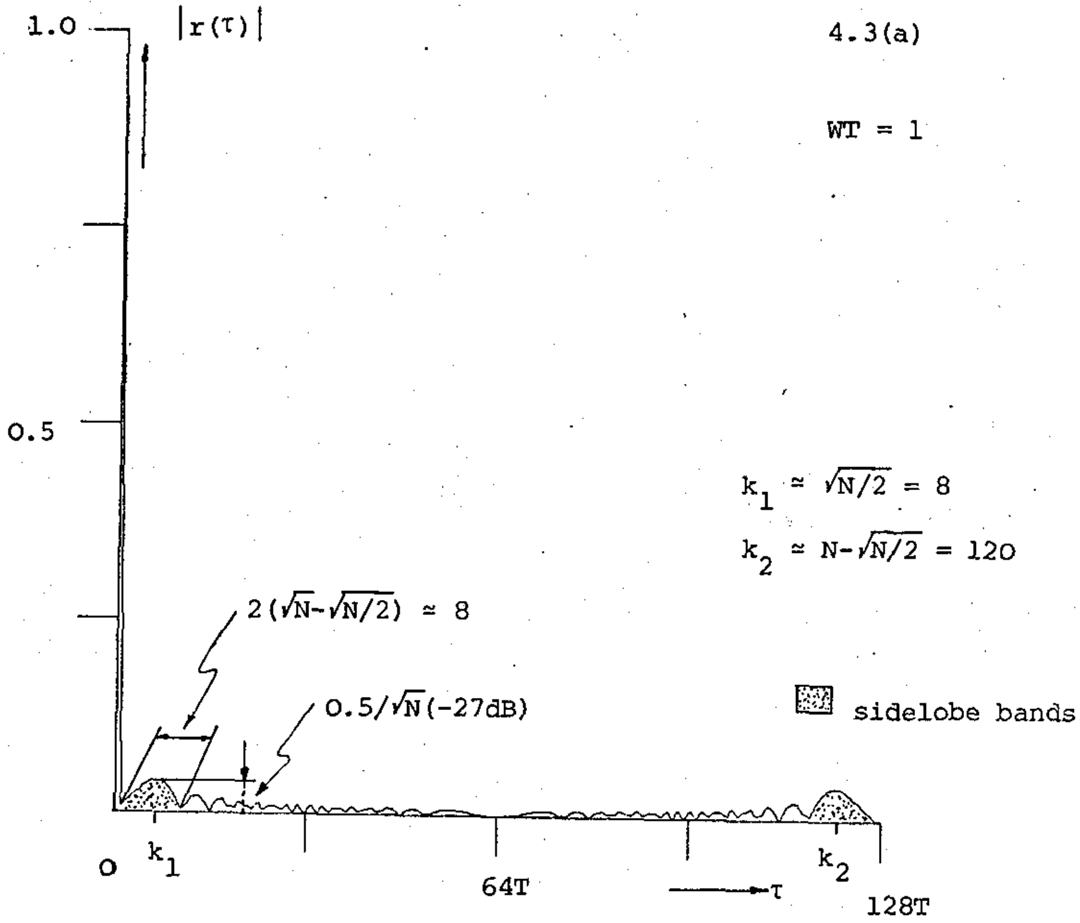
$$\tau_g(f) = \mathrm{Re}\ \{T(\sum_{n=o}^{N-1} n\ s(n)z^{-n}/\sum_{n=o}^{N-1} s(n)z^{-n})\} \tag{4.19}$$

The above equation can be evaluated using DFT techniques

$$\tau_g(n/NT) = \mathrm{Re}\ \{T \quad \frac{\mathrm{DFT\ (1st\ moment\ of\ s(n))}}{\mathrm{DFT(s(n))}}\}$$

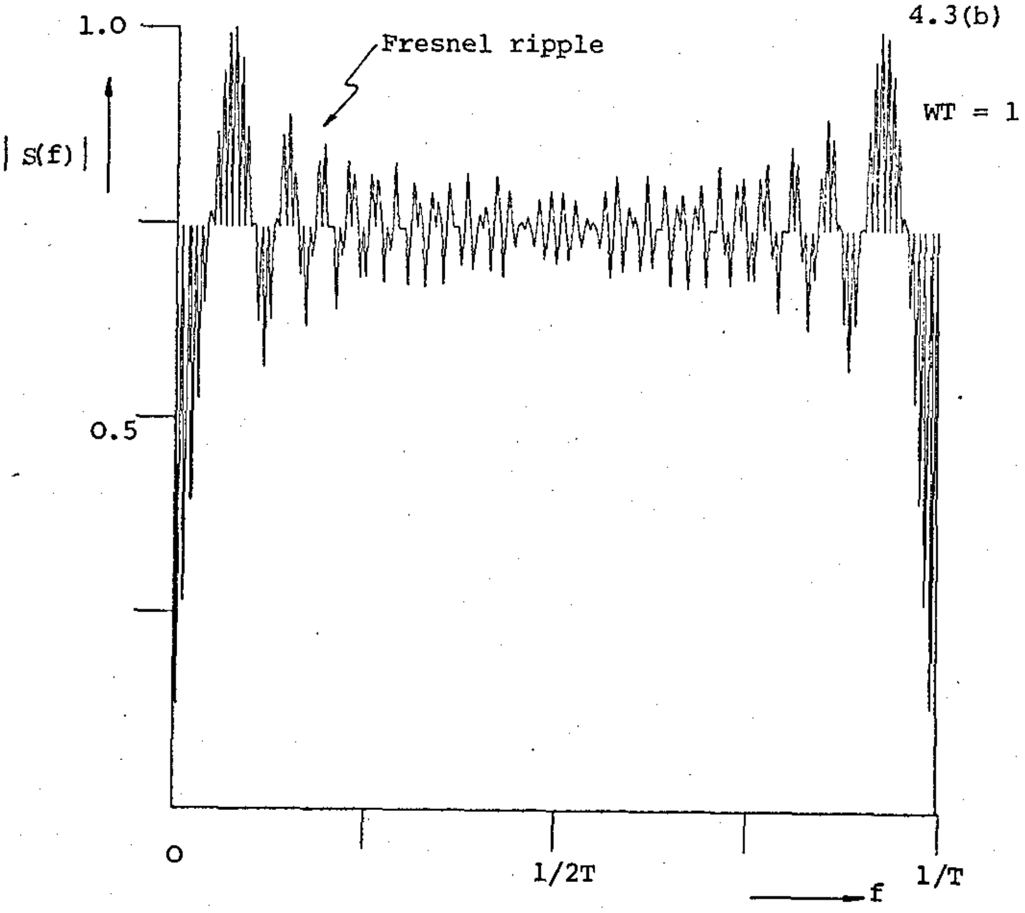
In the computations, T was assigned the value of unity so as to normalize the scales of the resulting graphs. In addition $\tau_g(f)$ is plotted only over that frequency range where the spectrum is significant.

The figures show quite clearly the LFM type properties. The sinc effect can be observed near the mainlobe and at time shifts where k is approximately NT. The maximum sidelobes, however, are smaller than the



4.3(a)

1.0
|r(τ)|
WT = 1
0.5
$k_1 \simeq \sqrt{N/2} = 8$
$k_2 \simeq N-\sqrt{N/2} = 120$
$2(\sqrt{N}-\sqrt{N/2}) \simeq 8$
$0.5/\sqrt{N}$(-27dB)
▨ sidelobe bands
0
k_1
64T
→ τ
k_2
128T



4.3(b)

1.0
Fresnel ripple
WT = 1
|S(f)|
0.5
0
1/2T
→ f
1/T

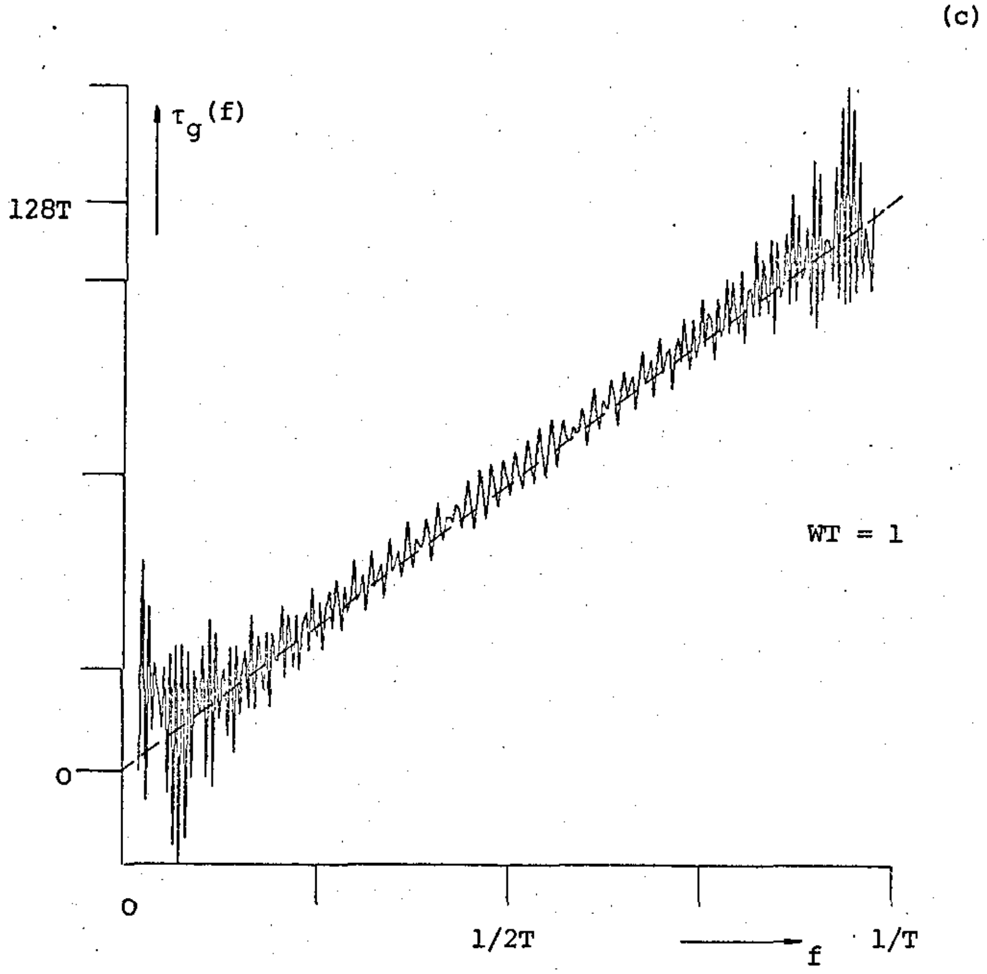


Fig. 4.3 (a) ACF, (b) amplitude spectrum and (c) group delay of a 128-element QP code for WT = 1.

-13dB achieved with analogue LFM waveforms, Fig. 4.4. The peak sidelobe can be calculated using the sinc approximation for small time lags. Thus for k << NT

$$|r(kT)| \simeq \left| \frac{k \sin \pi k^2/N}{\pi k^2/N} \right|$$

The first maximum occurs where

$$k \approx \sqrt{N/2}$$

hence

$$\max_{k \neq o} |r(kT)| \approx \frac{\sqrt{2N}}{\pi} = 0.45 \sqrt{N}$$

Using the exact expression a more precise value can be obtained numerically, which turned out to be 0.48 $\sqrt{N}$. The two main sidelobe bands occur at time shifts where $k = \sqrt{N/2}$ and $k = (N-\sqrt{N/2})$ respectively. This property of increased sidelobes towards the end of the matched filter response is quite different from the analogue waveform where the sidelobes decrease uniformly with increasing time shift. However, the overall properties of the analogue LFM signal is preserved. This is reflected in the familiar Fresnel ripples of the spectrum and the linear group delay (Appendix A).

The characteristics of the QP pulse trains when sampled at a lower or faster rate than the Nyquist rate (under or over-sampling) are depicted in Fig. 4.5 to 4.7. These graphs reveal some interesting properties. First, if sampled at the Nyquist rate (WT=1), the ACF consists of a sharp narrow spike with low residue sidelobes. Secondly, if WT < 1, over-sampling occurs and sidelobes near the main peak appear. This is not surprising, since increased sampling rate implies a closer approximation to the analogue FM signal whose maximum sidelobes are immediately adjacent to the mainlobe, as shown in Fig. 4.4. In other

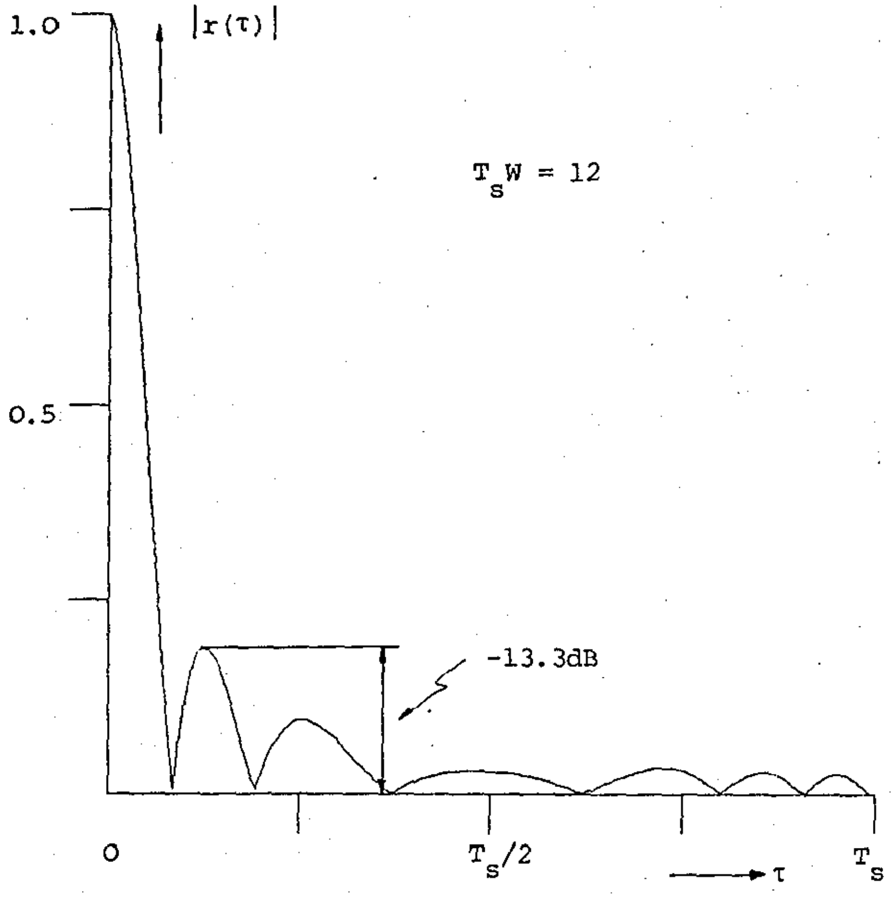


Fig. 4.4 ACF of a LFM signal.

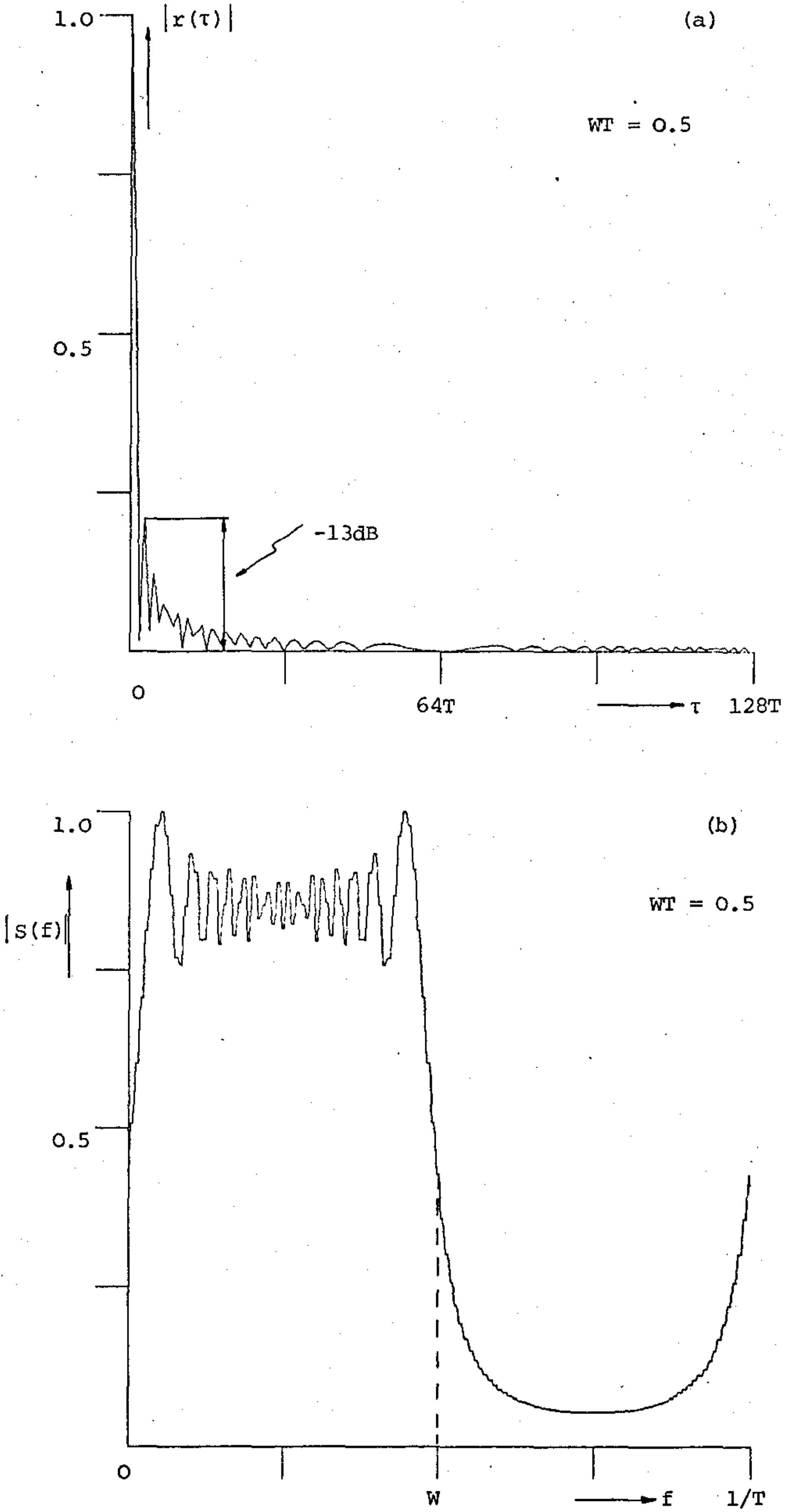


Fig. 4.5 (a) ACF, (b) amplitude spectrum of 128-element QP code for WT = 0.5 (over-sampling).

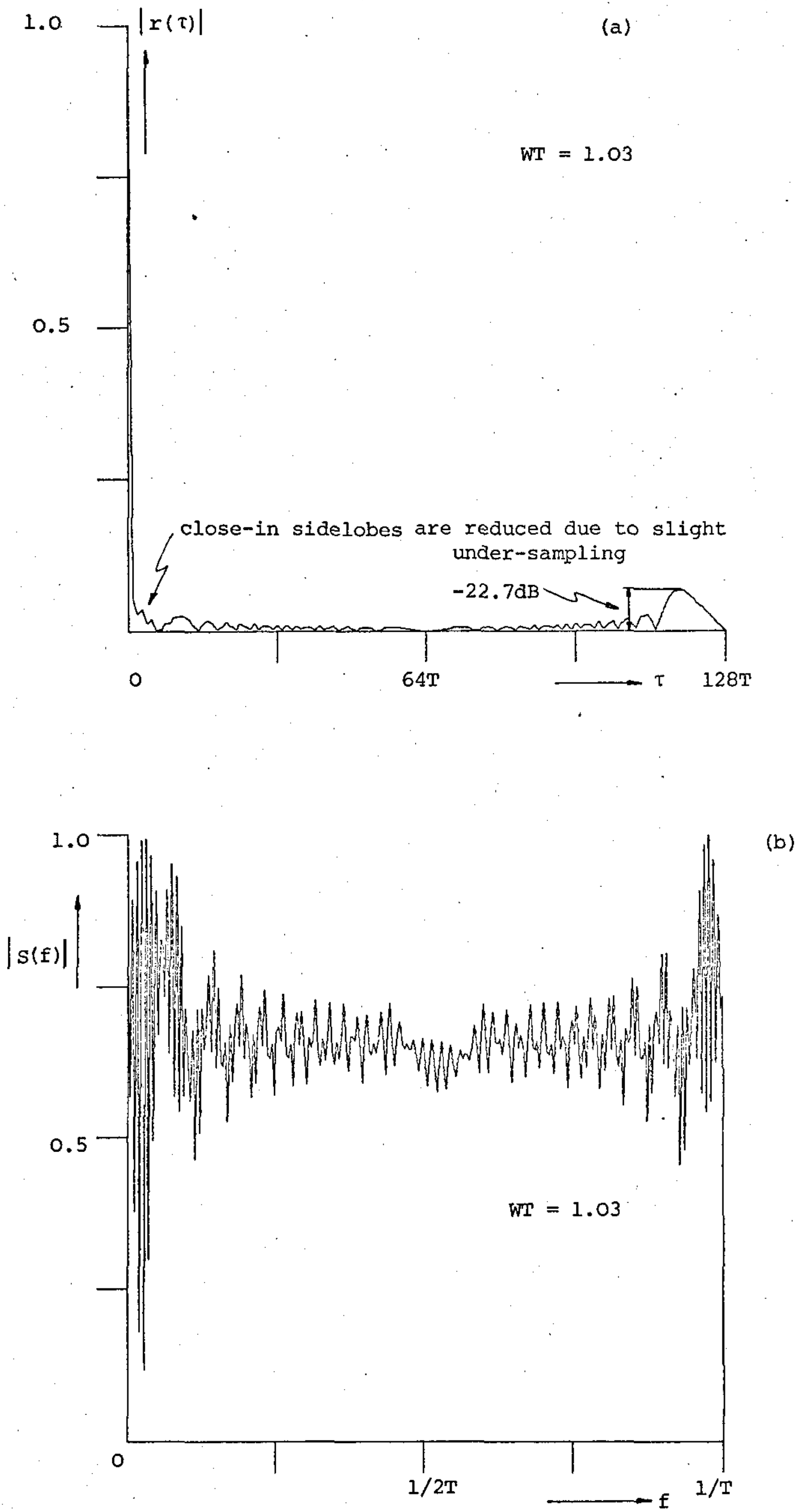


Fig. 4.6 Effects of slight under-sampling on the (a) ACF, and (b) the amplitude spectrum.

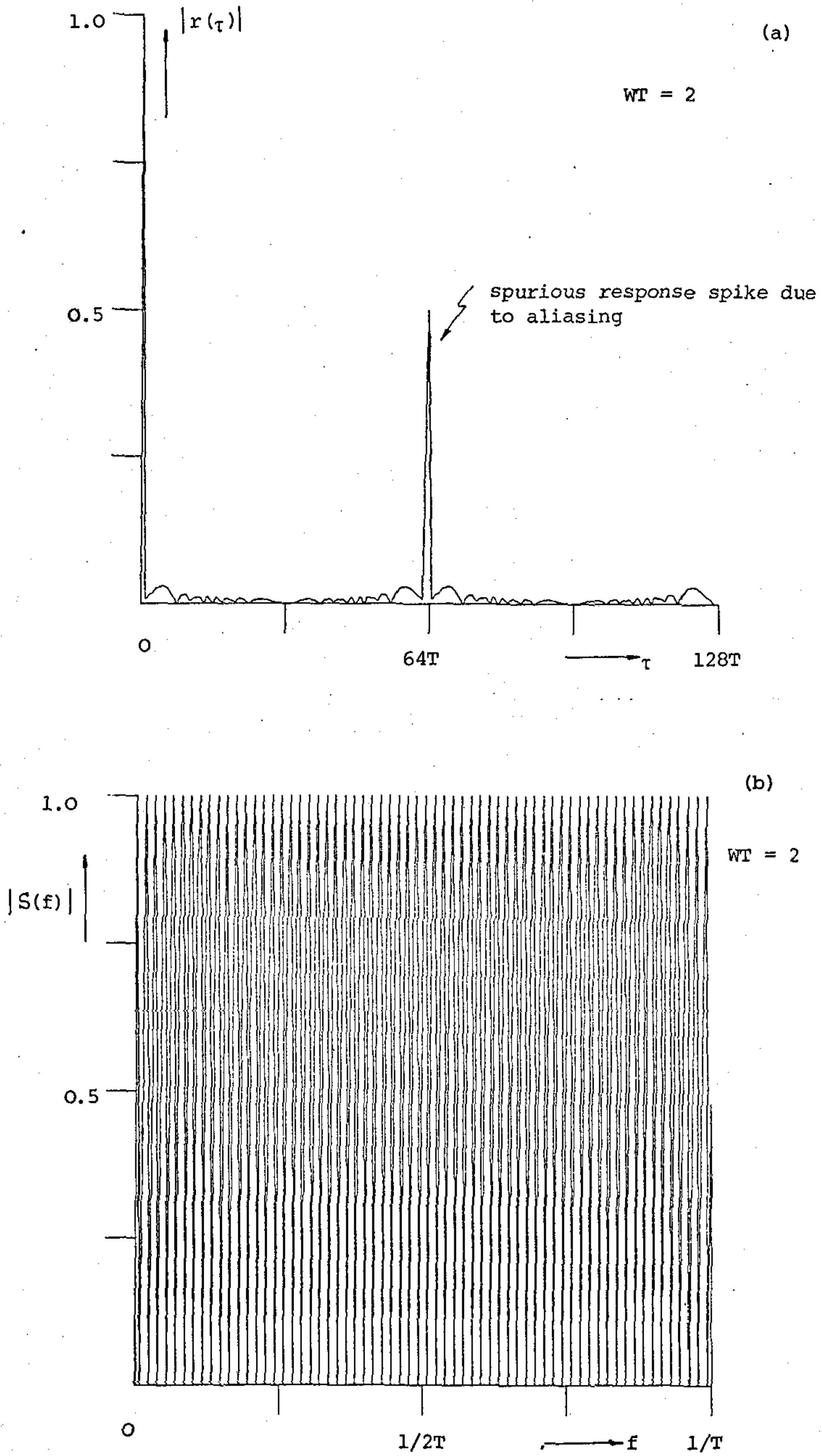


Fig. 4.7 Aliasing effects on the ACF (a), and amplitude spectrum (b), due to under-sampling.

words, the sampling points do not miss these large sidelobes as is the case for WT=1. Thirdly, for under-sampling WT > 1, the aliased versions of the ACF will produce significant range ambiguity at time shifts k=N/WT from the compressed pulse, i.e. for WT=2, k=N/2 as illustrated in Fig. 4.7. The cause of the spurious response peaks can probably be best explained by recalling the relationship between the spectra of an analogue waveform and its sampled version, Eq. (2.9)

$$H(f) = \sum_{n=-\infty}^{\infty} H_a(f+n/T)$$

Hence the ambiguity function of a pulse train is periodic in the frequency direction with period 1/T.

$$\chi(\tau,\nu) = \sum_{n=-\infty}^{\infty} \chi_a(\tau,\nu+n/T) \tag{4.20}$$

This is illustrated in Fig. 4.8 for a QP signal with its typical diagonal ridge structure of the ambiguity function (see Chapter 9).

Another property of interest is the sidelobe energy ratio defined as

$$E_s = \sum_{k=1}^{N-1} |r(k)|^2/|r(o)|^2 \tag{4.21}$$

In a dense uncorrelated target environment E_s serves as a measure for the self-clutter interference. This quantity is related to Woodward's range resolution constant, (Eq. (2.43) as

$$\Delta\tau = 1 + 2E_s$$

In Fig. 4.9 the ratio E_s and $\max_k |r(k)|$ are plotted as functions of the sequence length, N. It is noted that the sidelobe energy is falling off steadily with increasing N.

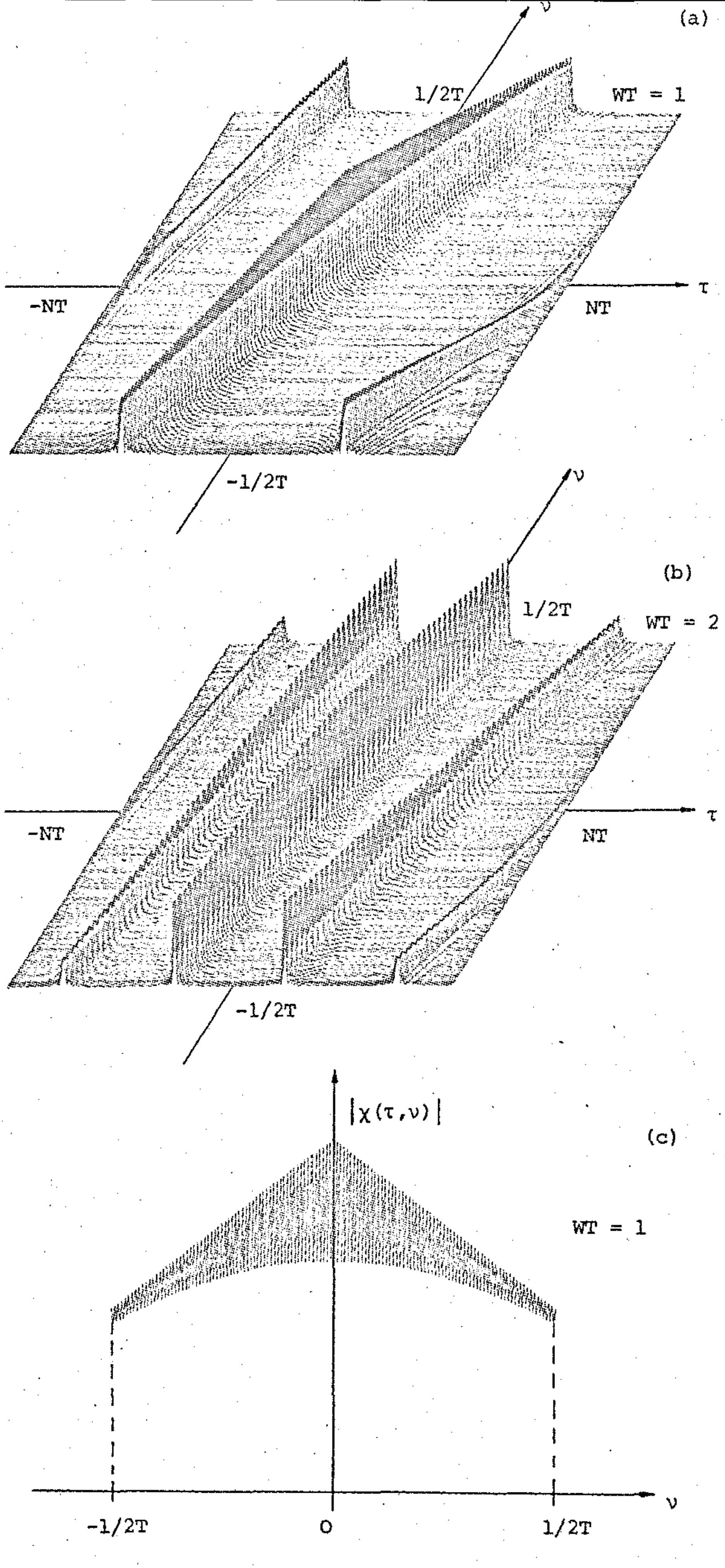


Fig. 4.8 Effect of sampling on the ambiguity function, (a) and (b); (c) peak response as a function of doppler shift ν.

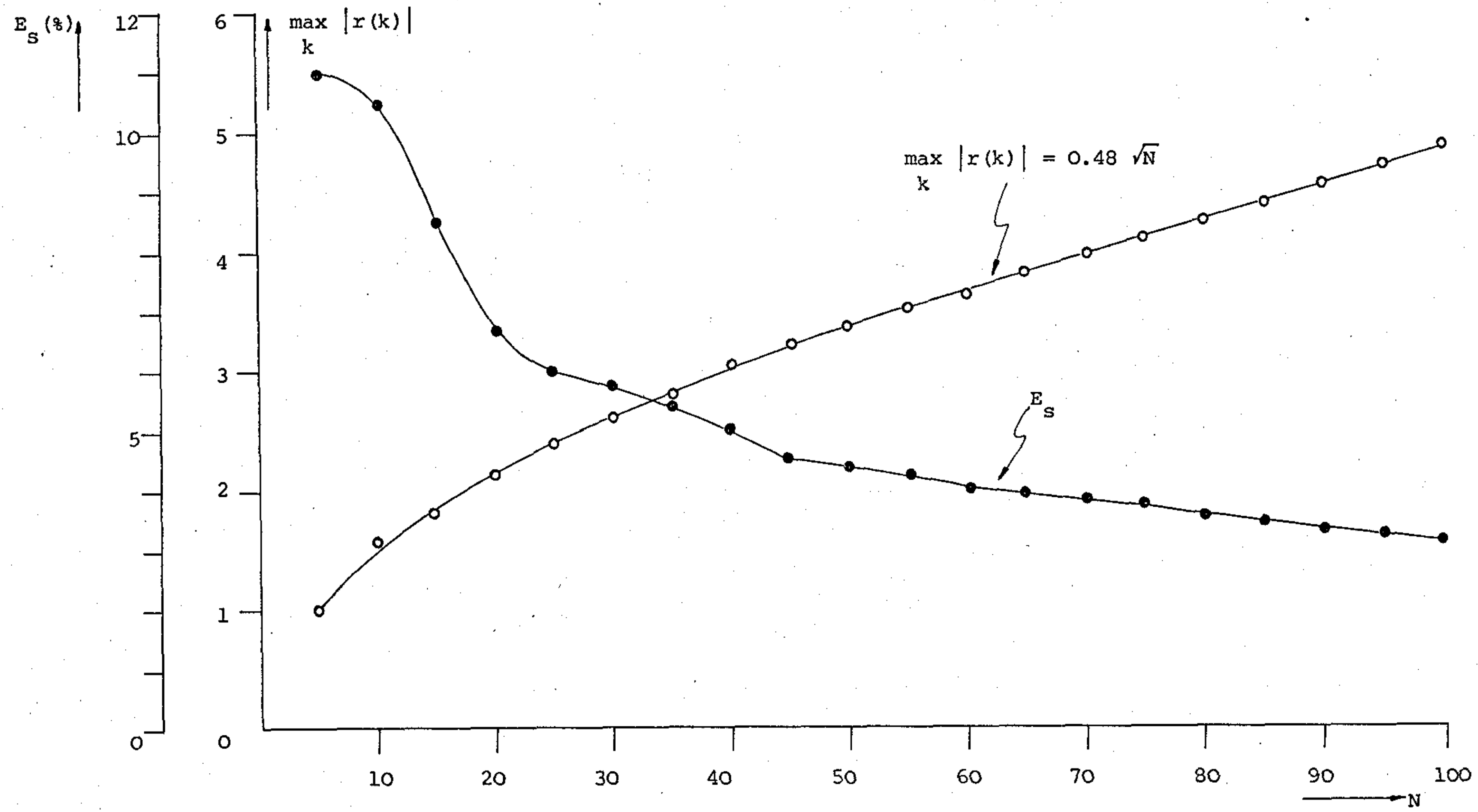


Fig. 4.9 Maximum sidelobe and sidelobe energy ratio for QP codes.

So far it has been shown that spurious response peaks can be avoided provided the waveform is sampled at the Nyquist rate. The resulting sequences have low sidelobes and low sidelobe energy and thus are suitable in a multiple-target environment. However, there may be specific cases where WT is made slightly greater than one, to attain somewhat better range resolution. By comparison of Fig. 4.6 with Fig. 4.3 it can be seen that a small increase in WT tends to reduce the sidelobes close to the mainlobe at the expense of an increase towards the end of the response. The remainder of this section, however, will be limited to the discussion of QP sequences where WT=1.

From the graphs it is avident that the sidelobe structure consists mainly of two 'humps' referred to as sidelobe bands (Fig. 4.3). The duration of the bands is approximately $2(\sqrt{N} - \sqrt{N/2})$. These sidelobe bands are produced by the pronounced Fresnel ripple of the amplitude spectrum shown in Fig. 4.3. The ripple spectrum is a function of the compression ratio WT_s and can be thought of as a combination of a slowly varying function multiplied with a high oscillatory component over the bandwidth 1/T[22,51]. The sidelobe bands can thus be regarded as being similar to those of amplitude modulation but with time and frequency interchanged. Since the two major bands occur at $k \simeq \sqrt{N/2}$ and $k \simeq N - \sqrt{N/2}$, it can be assumed that the ripple spectrum consists largely of two sinusoidal components at frequencies $\sqrt{2/N}$ and $1/(N - \sqrt{N/2})$ and their cross-modulation products.

The important properties of QP pulse trains may now be summarized as follows:

(i) If sampled at the Nyquist rate QP codes have virtually all the properties of LFM signals. Their ACF consists of a single spike with low level sidelobes.

(ii) The codes have zero periodic ACF's for $k \neq 0$.

(iii) For $WT < 1$, large close-in sidelobes appear, whereas for $WT > 1$ spurious response peaks occur further away from the mainlobe.

(iv) The major sidelobes occur in two bands approximately centred at time shifts $k = \sqrt{N/2}$ and $k = (N - \sqrt{N/2})$ respectively. The width of the bands is about $2(\sqrt{N} - \sqrt{N/2})$. Moreover, for sequences of even length N, the sidelobe structure is symmetrical with respect to $k = N/2$.

(v) The maximum sidelobes increase approximately as $0.5 \sqrt{N}$ and the energy ratio $E_s < 5\%$ for $N > 40$.

(vi) Relative simple generation of such pulse trains with a suitable choice of N.

Hence QP sequences have good range resolution properties which make them suitable for a dense-target environment. For example, the peak range sidelobe for N = 128 is -27.5 dB down on the main response and the r.m.s. sidelobes are about -36.6 dB down.

4.3.2 An Iterative Method for Reducing the Ripple Spectrum

In the preceeding section it was observed that the sidelobe bands are caused by the ripples in the Fresnel spectrum which was obtained using the stationary phase method. This method, however, does not consider the 'edge' effects associated with the waveform fronts. Sharp leading and trailing edges, as well as any sharp discontinuities in general, give rise to spectrum ripples of appreciable magnitude. A reduction of the ripple spectrum, therefore, would subsequently result in a reduction of the time sidelobes of the ACF.

A method considering the 'edge' effects was proposed by Vakman[44] in connection with narrow beam antenna design. To illustrate the application of this method to the signal design problem consider the signal obtained by the stationary phase technique as an initial approximation to the desired signal whose power spectrum P(f) and envelope $|s_o(nT)|$ are specified. Thus

$$s_o(nT) = a_o(nT)\ e^{j\phi_o(nT)}$$

and

$$S_o(f) = |S_o(f)|\ e^{j\theta_o(f)} = \mathrm{FT}\ \{s_o(nT)\}$$

The Fourier transform is evaluated numerically using the FFT algorithm. If the conditions for application of the stationary phase are satisfied, then $|S_o(f)|^2$ will be a reasonable approximation to the given power spectrum P(f). It is noted that the signal envelope $a_o(nT)$ coincides with the specified amplitude function. In attempting to synthesize a signal having a given power spectrum P(f), the spectrum

$$S_1(f) = \{P(f)\}^{\frac{1}{2}}\ e^{j\theta_o(f)}$$

retaining the initial phase function $\theta_o(f)$, is taken as the next approximation. Hence, the first corrected signal is given by

$$s_1(nT) = a_1(nT)\ e^{j\phi_1(nT)} = \mathrm{IFT}\{S_1(f)\}$$

Here the given power spectrum is realized exactly. However, the envelope $a_1(nT)$ will differ from the specified envelope $a_o(nT)$. The next step, therefore, is to take $a_o(nT)$ and assigning the phase $\phi_1(nT)$ to it, i.e.

$$s_2(nT) = a_o(nT)\ e^{j\phi_1(nT)}$$

The spectrum corresponding to this signal

$$S_2(f) = |S_2(f)| \; e^{j\theta_1(f)}$$

differs in magnitude from $\{P(f)\}^{\frac{1}{2}}$. However, in many cases these differences are smaller than for the initial approximation. This procedure can be repeated iteratively and in general

$$s_i(nT) = a_o(nT) \; e^{j\phi_{i-1}(nT)}$$

Unfortunately, this method of successive approximations does not converge to an optimal solution. This is so because a spectrum with a finite bandwidth cannot be realized by a signal of finite duration. Nevertheless, this technique has been successfully applied in some antenna design problems. However, for pulse train waveforms such an approximation procedure might not yield any improvement, because of their discrete nature. Moreover, it can be shown using the stationary phase principle that the envelope $|s(nT)|$ decreases as $1/nT$ near the edges. But the envelope of the pulse trains considered here is constant, and furthermore, only a finite number of samples are taken. Naturally, such a truncation of the time series will always introduce spectrum ripples and thus range sidelobes.

The method of successive approximation to a rectangular spectrum was programmed on a digital computer. The results are shown below in Table 4.1. As expected the method did not converge. In general the peak sidelobe level oscillated between a maximum and minimum value and in most cases no significant improvement was observed. However, the sidelobe energy ratio usually decreased to about two thirds of its original value.

Code length	Initial values		Final values	
N	$\max_k \lvert r(k) \rvert$	E_s (%)	$\max_k \lvert r(k) \rvert$	E_s (%)
16	1.85	6.31	1.44	3.48
20	2.18	5.76	2.18	5.75
25	2.46	4.75	1.63	2.41
30	2.62	5.57	2.46	4.94
35	2.82	4.42	1.43	2.32
40	3.08	3.94	2.78	3.38
45	3.26	3.70	2.69	2.68
50	3.40	3.74	2.13	1.99
55	3.51	3.62	3.24	3.24
60	3.73	3.50	2.57	2.54
64	3.88	3.29	2.11	1.27
70	4.05	3.16	3.57	2.58
75	4.17	2.99	3.27	2.06
80	4.26	2.87	2.72	1.62
85	4.42	2.78	3.31	1.77
90	4.57	2.74	4.24	2.23
95	4.71	2.73	3.54	1.64
100	4.83	2.61	3.05	1.51

Table 4.1 Sidelobe reduction by spectrum approximation.

Although some improvement in sidelobe and energy performance is observed this technique fails to reduce the Fresnel ripples substantially. Therefore, if sequences with very low sidelobes are required such a method does not solve the problem.

4.3.3 Tapering the Pulse Train

Another method to reduce the Fresnel ripples is to taper the edges of the pulse train. Such a tapering function is shown in Fig. 4.10, using a cosine taper over 1/10 of each end of the signal length. The shaping function can be expressed analytically as

$$u_T(t) = \begin{cases} \cos^2(5\pi t/T_s) & ; \quad -T_s/2 \leq t \leq -4T_s/10 \\ 1 & ; \quad -4T_s/10 \leq t \leq 4T_s/10 \\ \cos^2(5\pi t/T_s) & ; \quad 4T_s/10 \leq t \leq T_s/2 \end{cases} \qquad (4.22)$$

In general the frequency spectrum of this type of tapering function is given by[55]

$$U_T(f,p) = \frac{\sin([(2p-1)/2p]\pi f T_s)}{[(2p-1)/2p]\pi f} \left\{ \frac{(2\pi p/T_s)^2 \cos(T_s/4p)}{(2\pi p/T_s)^2 - (2\pi f)^2} \right\} \qquad (4.23)$$

When p=5 the spectrum of Eq. (4.22) is obtained.

The effects of such a tapering on the spectrum magnitude of the QP code and its ACF are shown in Fig. 4.11 for a code of length N = 128.

The figures clearly show the removal of the high frequency ripple component and thus the more distant sidelobes. However, the smoothing of the spectrum is obtained at the expense of an increase of the sidelobes near the main peak. This may be of little concern in applications where the targets are separated by more than the sidelobe duration. An additional penalty is the reduction in energy efficiency

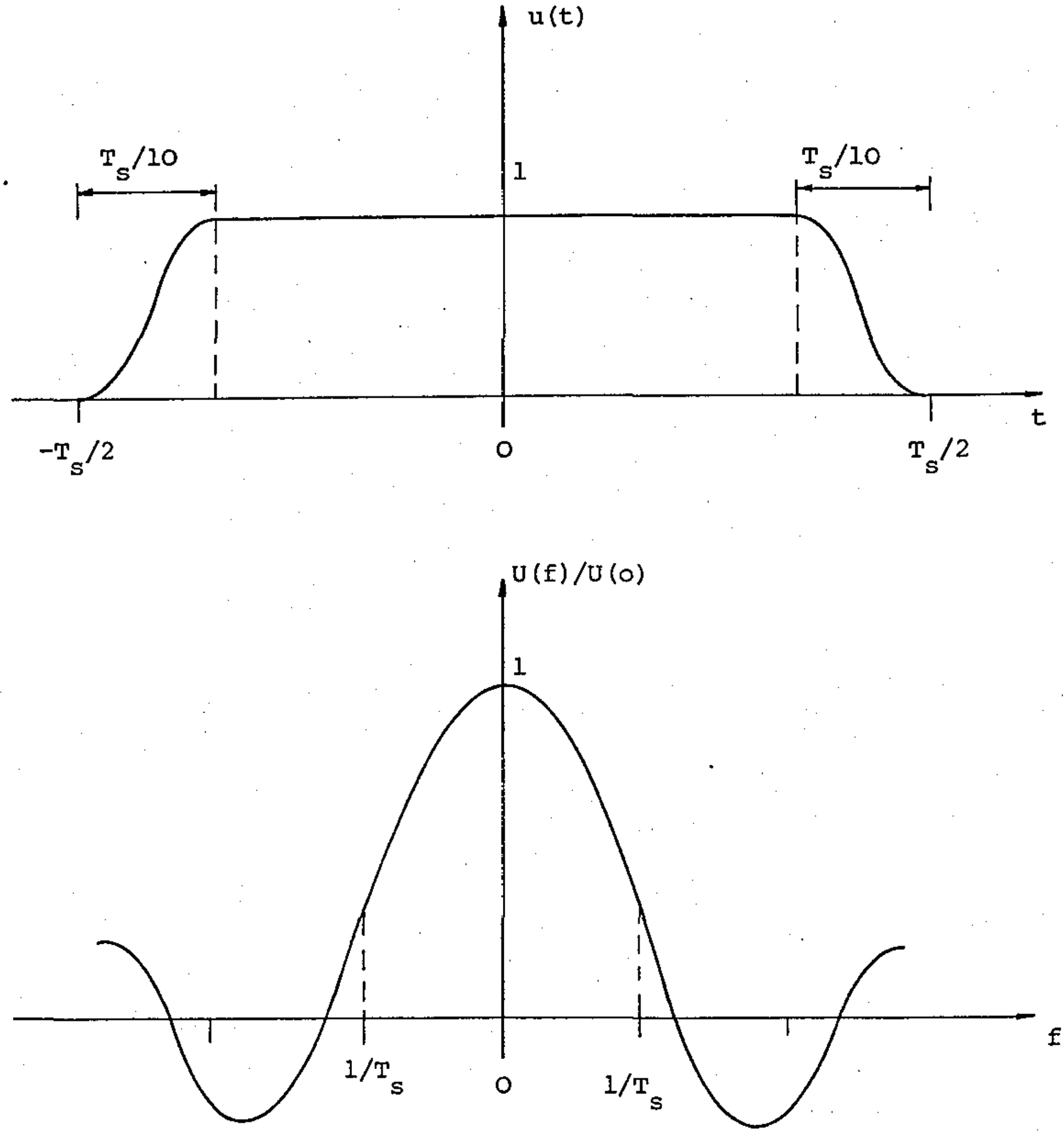


Fig. 4.10 Shaping function and its spectrum.

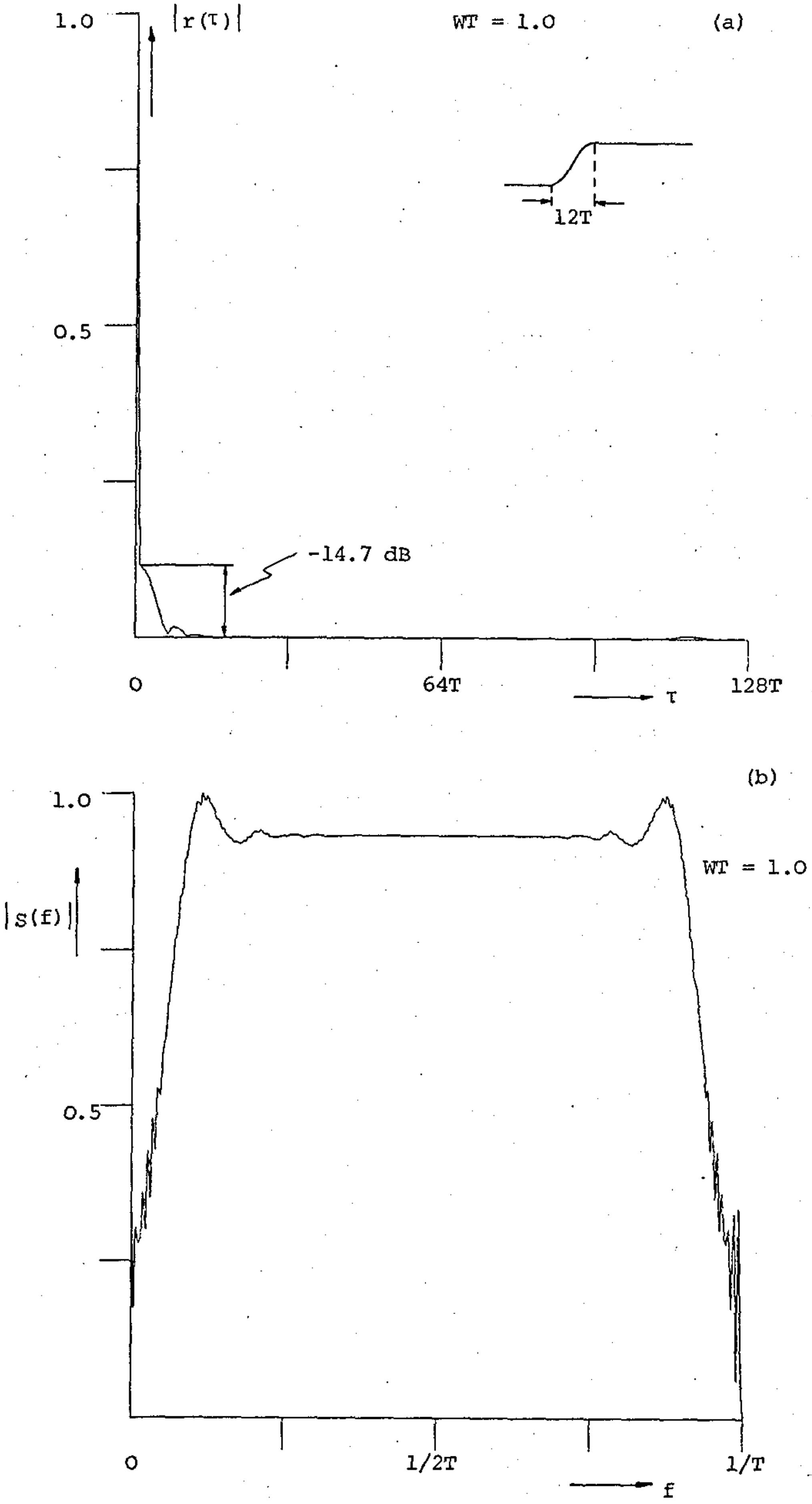


Fig. 4.11 Effect of tapering the edges of the pulse train on the ACF (a), and amplitude spectrum (b).

which is about -0.7 dB for p=5 and N=64 and N=128 respectively. In practice the tapering function could be approximated by simply increasing the rise time of the transmitted waveform. Moreover, for a low power radar the loss in transmitted energy can be kept relatively small so it does not seriously affect the detection capability.

Unfortunately, this method of improving the range resolution is not applicable to most high power radar systems. There the tapering function cannot be controlled without a serious loss in output power. Another approach to reduce the range sidelobes which does not suffer from this drawback is to modify only the phase of the individual subpulses.

4.3.4 Sidelobe Reduction by Phase Correction

To achieve very low sidelobes (below -30 dB) the stationary phase method is not adequate, because it does not account for the quasi-oscillatory nature of the actual spectrum. Additional ripples are introduced by a quantized approximation of the LFM law. If a small loss in transmitted power is acceptable, tapering of the edges of the pulse train has been shown to be a useful means for reducing the more distant sidelobes. However, in many applications this approach is not desirable. It should be possible to achieve similar effects by modifying the phase function only.

Introducing a phase correction term $\varepsilon(t)$ the complex envelope of the transmitted waveform can be written as

$$s(t) = s_o(t)\ \exp(j\ \varepsilon(t)) \qquad (4.24)$$

where $s_o(t)$ is the original signal.

The phase term $\varepsilon(t)$ could be any arbitrary function, provided it has the effect of reducing the sidelobes. To study how the ACF is affected by such a phase perturbation, assume that $\varepsilon(t)$ is a sinusoidal function given by

$$\varepsilon(t) = \varepsilon \sin(2\pi ft + \phi) \qquad (4.25)$$

where

ε = peak phase correction

f = frequency of phase correction

ϕ = arbitrary phase

Expanding $\exp(j\,\varepsilon(t))$ in Bessel functions of the first kind (Appendix C) leads to

$$e^{j\varepsilon(t)} = \sum_{k=-\infty}^{\infty} J_k(\varepsilon)\; e^{jk(2\pi ft+\phi)}$$

For Bessel function of integral order the following relationships exist

$$J_k(-\varepsilon) = (-1)^k\, J_k(\varepsilon) \qquad (4.26)$$

$$J_{-k}(\varepsilon) = (-1)^k\, J_k(\varepsilon) \qquad (4.27)$$

Thus

$$s(t) = s_o(t) \sum_{k=-\infty}^{\infty} J_k(\varepsilon)\; e^{jk(2\pi ft+\phi)} \qquad (4.28)$$

The ACF of the sampled signal can now be written as

$$r(\tau) = \sum_{n=-\infty}^{\infty} s(nT)\; s^*(nT+\tau) = \sum_{n=-\infty}^{\infty} s_o(nT)\; s_o^*(nT+\tau)\, e^{j\left[\varepsilon(nT)-\varepsilon(nT+\tau)\right]}$$

$$r(\tau) = \sum_{n=-\infty}^{\infty}\; \sum_{k=-\infty}^{\infty}\; \sum_{\ell=-\infty}^{\infty} s_o(nT)\,s_o^*(nT+\tau)\,J_k(\varepsilon)\,J_\ell(\varepsilon)\,e^{j\{2\pi f\left[(k-\ell)nT-\ell\tau\right] + (k-\ell)\phi\}} \qquad (4.29$$

Now let ε be small ($\varepsilon \lesssim 0.4$ radians) so that the following approximations can be made

$$J_o(\varepsilon) \simeq 1$$

$$J_1(\varepsilon) \simeq \varepsilon/2$$

and $\quad J_k(\varepsilon) \simeq 0 \qquad$ for $k \geqslant 2$

Hence

$$r(\tau) = \sum_{n=-\infty}^{\infty} s_o(nT)\; s_o^*(nT+\tau)\; \{J_o(\varepsilon) + J_1(\varepsilon)\; e^{j(2\pi fnT+\phi)}$$

$$+ J_{-1}(\varepsilon)\; e^{-j(2\pi fnT+\phi)}\} \times \{J_o(\varepsilon) + J_1(\varepsilon)\; e^{-j(2\pi f(nT+\tau)\; +\; \phi)}$$

$$+ J_{-1}(\varepsilon)\; e^{j(2\pi f(nT+\tau)\; +\; \phi)}\}$$

Neglecting higher order terms of the form $J_1^2(\varepsilon)$ and using property (4.27) the above expressions can be simplified to

$$r(\tau) = \sum_{n=-\infty}^{\infty} s_o(nT) s_o^*(nT+\tau)\; \{J_o^2 + J_o\; J_1\; e^{-j(2\pi f(nT+\tau)\; +\; \phi)}$$

$$- J_o\; J_1\; e^{j(2\pi f(nT+\tau)\; +\; \phi)} + J_o J_1\; e^{j(2\pi fnT\; +\; \phi)} - J_o J_1\; e^{-j(2\pi fnT\; +\; \phi)}\}$$

It is recognized that

$$\chi_o(\tau,f) = \sum_{n=-\infty}^{\infty} s_o(nT)\; s_o^*(nT+\tau)\; e^{j(2\pi f(nT+\tau))}$$

is the ambiguity function of the signal, hence

$$r(\tau) = J_o^2\;\; r_o(\tau) + J_o\; J_1\; e^{-j\phi}\; \chi_o(\tau,\, -f) - J_o\; J_1\; e^{j\phi}\; \chi_o(\tau,f)$$

$$+ J_o\; J_1\; e^{-j(2\pi f\tau-\phi)}\; \chi_o(\tau,f) - J_o\; J_1\; e^{j(2\pi f\tau-\phi)}\; \chi_o(\tau,\, -f)$$

Substituting the approximate values for $J_o(\varepsilon)$ and $J_1(\varepsilon)$ leads to

$$r(\tau) = r_o(\tau) + \frac{\varepsilon}{2}\;\; \{\chi_o(\tau,-f)\; e^{-j\phi} - \chi_o(\tau,f)\; e^{j\phi}\}$$

$$+ \frac{\varepsilon}{2}\;\; \{\chi_o(\tau,f)\; e^{-j(2\pi f\tau-\phi)} - \chi_o(\tau,-f) e^{j(2\pi f\tau-\phi)}\}$$

$$r(\tau) = r_o(\tau) + \varepsilon\; \sin(\pi f\tau)\{\chi_o(\tau,-f)\; e^{j(\pi f\tau-\phi-\frac{\pi}{2})}$$

$$- \chi_o(\tau,f)\; e^{-j(\pi f\tau-\phi-\frac{\pi}{2})}\} \qquad (4.30)$$

The effect of a sinusoidal phase correction can be visualized as adding phase shifted cross-sections of the ambiguity surface to the ACF.

The property of cancellation and reinforcement of the range sidelobes is inherent in Eq. (4.30). Depending upon the value of ε, the peak value of the phase perturbating function and the phase ϕ, it is possible under certain conditions to obtain various patterns of sidelobe cancellation.

For QP sequences the range-doppler ambiguity has a diagonal ridge structure, Fig. 4.8(a). The equation of the ridge in the delay-doppler plane for integral time lags of T is given by (see also Chapter 9)

$$\nu = k/NT$$

The magnitude of the ambiguity function along the ridge reduces to (Fig. 4.8(c))

$$|\chi(k,\nu)| = (1 - |k|/N)\ N \tag{4.31}$$

Therefore, for any given cross-section $\chi(k,\nu)$ the major contribution when added to the ACF occurs at a time shift where $k = \nu NT$. Thus the ith sidelobe is given by

$$r(i) = r_o(i) - \varepsilon\chi_o(i,i/NT)e^{-j(\pi i^2/N-\phi-\frac{\pi}{2})}\ \sin(\pi i^2/N) \tag{4.32}$$

With a proper choice of ε and ϕ, the ith time sidelobe could be completely cancelled, i.e. $r(i) = 0$.

The analysis of the elementary case of one single sinusoidal correction function can be extended to more complex functions. The analytical approach is shown below for two or more sinusoids. This is not a rigorous restriction, since other functions could always be expanded into a Fourier series.

For two sinusoidal functions Eq. (4.24) becomes

$$s(t) = s_o(t)\ e^{j(\varepsilon_1 \sin(2\pi f_1 t + \phi_1) + \varepsilon_2 \sin(2\pi f_2 t + \phi_2))} \tag{4.33}$$

hence

$$s(nT) = s_o(nT) \left\{ \sum_{k=-\infty}^{\infty} \sum_{\ell=-\infty}^{\infty} J_k(\varepsilon_1) J_\ell(\varepsilon_2) e^{j(2\pi\left[(kf_1+\ell f_2)nT\right]+k\phi_1+\ell\phi_2)} \right\}$$

The following approximations can be made, provided that $\varepsilon_1 = \varepsilon_2 \lesssim 0.4$ radians

$$J_o(\varepsilon_1) = J_o(\varepsilon_2) \simeq 1$$

$$J_1(\varepsilon_1) \simeq \varepsilon_1/2 \text{ and } J_1(\varepsilon_2) \simeq \varepsilon_2/2$$

$$J_k(\varepsilon_1) = J_k(\varepsilon_2) \simeq 0 \qquad k \geqslant 2$$

After some tedious but straightforward manipulations, again neglecting higher order terms of the form J_1^2, $J_o^2 J_1^2$, $J_o J_1^2$, etc.,the ACF is given by

$$\begin{aligned} r(\tau) = r_o(\tau) &+ \varepsilon_1 \sin(\pi f_1 \tau)\left\{\chi_o(\tau,-f_1) e^{j(\pi f_1 \tau - \phi_1 - \pi/2)}\right. \\ &\left. -\chi_o(\tau,f_1)\ e^{-j(\pi f_1 \tau - \phi_1 - \pi/2)}\right\} \\ &+ \varepsilon_2 \sin(\pi f_2 \tau)\left\{\chi_o(\tau,-f_2)\ e^{j(\pi f_2 \tau - \phi_2 - \pi/2)}\right. \\ &\left. -\chi_o(\tau,f_2)\ e^{-j(\pi f_2 \tau - \phi_2 - \pi/2}\right\} \end{aligned} \qquad (4.34$$

or in condensed form

$$r(\tau) = r_o(\tau) + \chi_1(\tau,f_1,\phi_1) + \chi_2(\tau,f_2,\phi_2) \tag{4.35}$$

where

$$\begin{aligned} \chi_n(\tau,f_n,\phi_n) = \varepsilon_n \sin(\pi f_n \tau)&\left\{\chi_o(\tau,-f_n)\ e^{j(\pi f_n \tau - \phi_n - \pi/2)}\right. \\ &\left. -\chi_o(\tau,f_n)\ e^{-j(\pi f_n \tau - \phi_n - \pi/2)}\right\} \end{aligned} \tag{4.36}$$

For m sinusoidal components Eq. (4.35) can thus be written as

$$r(\tau) = r_o(\tau) + \sum_{n=1}^{m} \chi_n(\tau, f_n, \phi_n) \tag{4.37}$$

Neglecting higher order terms and satisfying the conditions $\varepsilon_n \leq 0.4$ radians, each individual sinusoidal term can be treated independently. Each component produces a pair of cross-sections of the ambiguity response that are added to the ACF. This is indicated in Fig. 4.12 for two terms with amplitudes $\varepsilon_1 = \varepsilon_2 = 0.2$ and frequencies $f_1 = 32/NT$ and $f_2 = 96/NT$. The contribution to the ACF can clearly be seen on the graph at time shifts $k = 32T$ and $k = 96T$. Even if the condition $\varepsilon_n \leq 0.4$ radians is not met, the single terms can be treated more or less individually in this case, because of the ridge-like ambiguity surface.

The property of reinforcement and cancellation has been used to reduce the sidelobes close to the main response peak. This is illustrated in Fig. 4.13 when choosing a phase correction function of the form

$$\varepsilon(nT) = \sum_{i=1}^{9} \varepsilon_i \cos(2\pi f_i nT)$$

It is noted that the cancellation of the close-in sidelobes has been achieved without otherwise affecting the overall ACF. However, it is not possible to reduce the sidelobe bands near the end of the MF response ($k \simeq NT$), using this technique. The major reason for this is that the ambiguity function along the ridge decreases rapidly, Eq.(4.31) for increasing time shifts.

The analysis of the effects of phase correction on the ACF is in many ways similar to the paired echo phenomenon[22]. However, in paired echo analysis the effects due to phase and amplitude mismatch (distortion) of the MF are examined, while here the transmitted waveform and the filter are assumed to be perfectly matched.

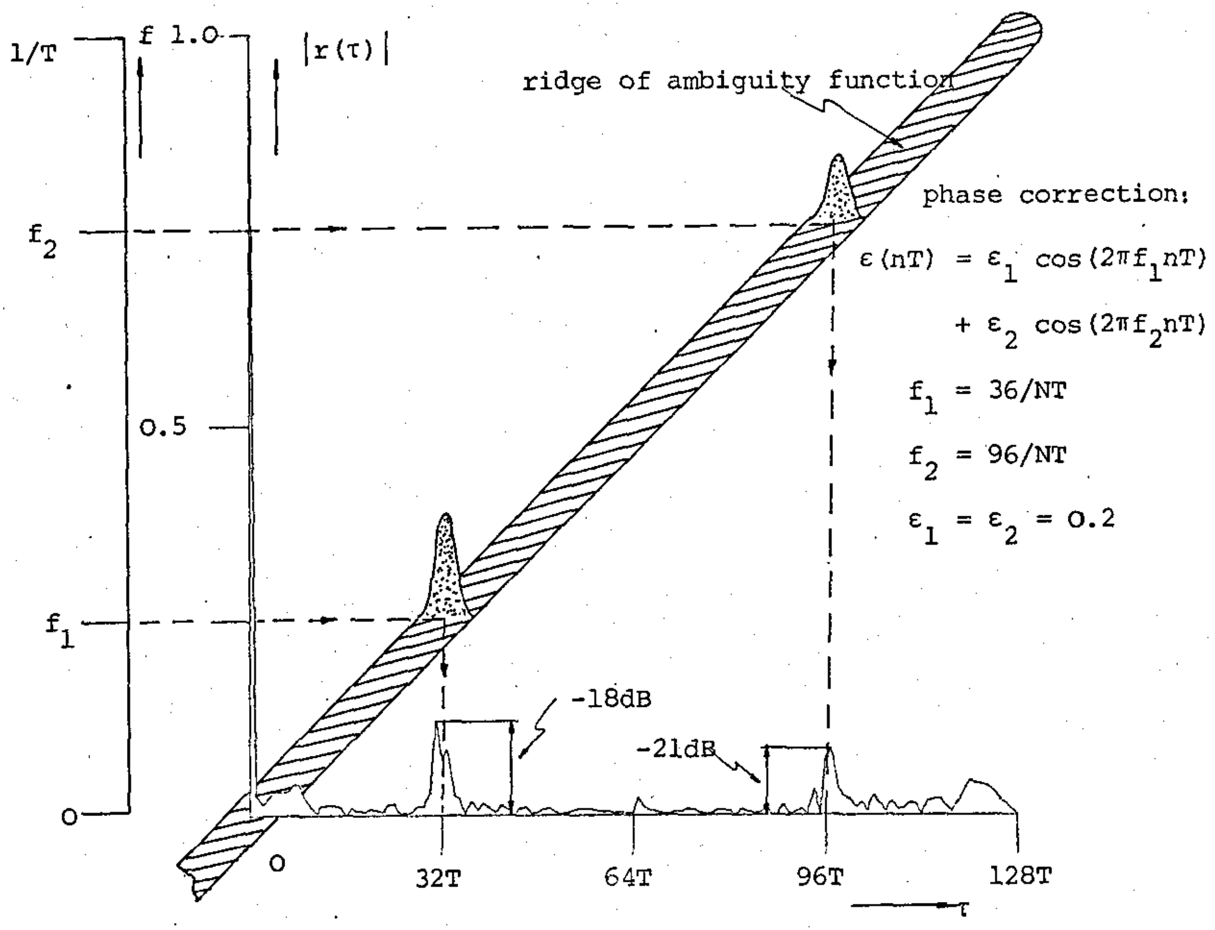


Fig. 4.12 Effects of the phase correction function on the ACF of a QP code.

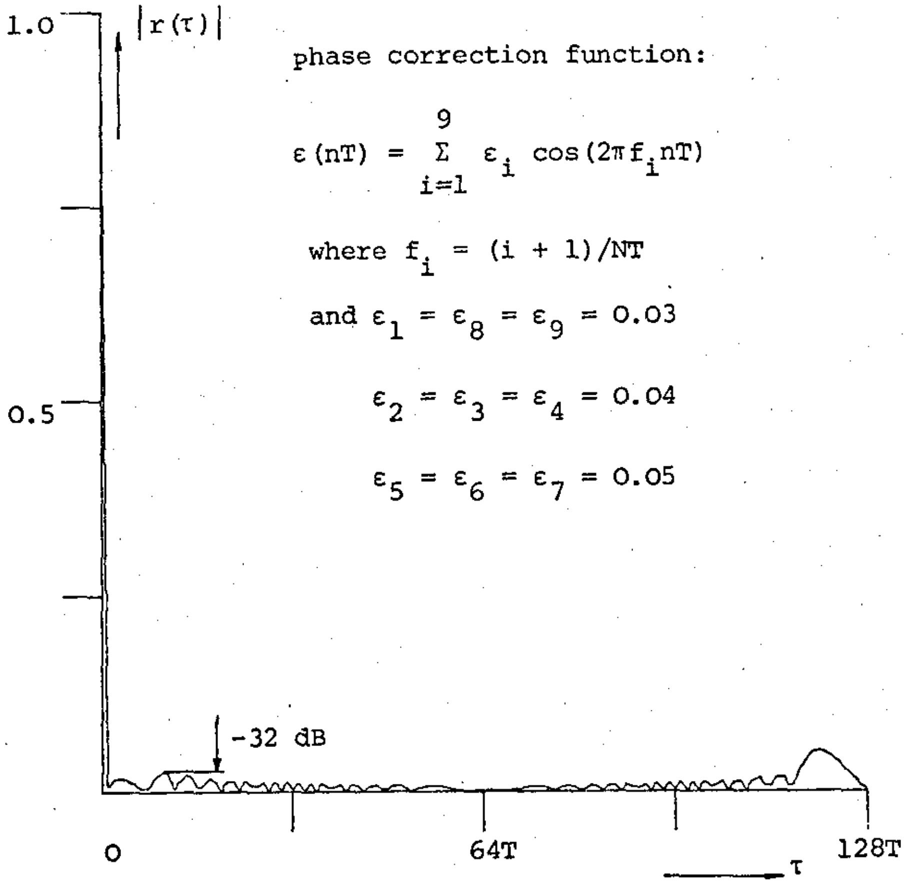


Fig. 4.13 Reduction of sidelobes using phase correction.

Two methods of controlling the range ambiguities of a QP waveform have been described. One method, by tapering the edges of the transmitted signal, removes the high frequency ripple component of the Fresnel spectrum. The close-in sidelobes, however, could not be reduced. In addition a penalty in the signal energy efficiency has to be paid. The second method, modifying the phase function, does not suffer from a loss in detection performance and thus is applicable to high power radars. The best results using this method are obtained for LFM type waveforms because of the ridge-like structure of the range-doppler ambiguity function. However, there is no reason why this method could not be applied successfully to other waveforms. It has been shown that in the case of QP codes only the close-in sidelobes can be controlled efficiently. Thus the two methods differ in that one basically controls the more distant sidelobes and the other the ambiguities close to the mainlobe.

Unfortunately, LFM type signals are not universally desirable waveforms. Therefore, in the next section another technique to control the range ambiguities is discussed.

4.4 Discrete Phase Approximation to Non-linear FM Signals

As pointed out by Fowle[50], the stationary phase analysis gives the phase modulation function for any specified power spectrum and signal envelope of sufficient time-bandwidth product. The advantage of a non-rectangular spectrum is that it allows control of the sidelobes, without loss in SNR, which might otherwise be achieved by mismatching (see Chapter 8). However, an undesirable side-effect associated with spectral shaping is the widening of the mainlobe. The reason for the broadening of the main response is determined intuitively from the Fourier transform reciprocity relationship. When the spectral shape is made narrower, the time domain waveform becomes broader and vice versa. Hence, in general, the resultant MF output waveform will be broader for non-linear FM than for the LFM case. Since the property of these waveforms is basically retained when approximated by discrete phase steps, this is also true for the resulting pulse trains. Nevertheless, a system designer may feel that it is worthwhile to accept a higher sidelobe level near the compressed pulse to give a lower level further away from the main response peak. On the other hand, a non-linear FM type waveform may simply be used because a LFM signal is not appropriate.

The weighting of the spectrum can be done in many different ways. It is difficult to define an optimum spectral shaping function if the trade-off possibilities of the particular application are not known. A class of signals having a rectangular envelope and a generalized power spectrum is defined as

$$|U_H(f,n,c)|^2 = A_n \{c + (1-c)\ H^n(f/W)\} \qquad (4.38)$$

where

$H^n(f/W)$ = any function of interest, $0 \leq f \leq W$

c = pedestal function

A_n = constant

A special form of the above power spectrum is given by

$$|U_{\cos}(f,n,c)|^2 = A_n \{c + (1-c)\ \cos^n(f/W)\} \qquad (4.39)$$

The frequency shaped compressed pulse then becomes

$$r(\tau) = \int_0^W |U_{\cos}(f,n,c)|^2\ e^{j2\pi f\tau}\ df \qquad (4.40)$$

Using Eq. (4.6) and assuming a constant signal envelope of duration NT, the group delay functions $\tau_g(f,n,c)$ are obtained by direct integration for various values of n and c. For zero pedestal (c=0), these functions are given by

$$n = 1 \quad \tau_g(f,1,0) = NT/2\{ 1 + \sin(\pi(f-W/2)/W) \}$$

$$n = 2 \quad \tau_g(f,2,0) = NT\{ 1/2 + (f-W/2)/W + 1/2\pi \sin[2\pi(f-W/2)/W]$$

$$n = 3 \quad \tau_g(f,3,0) = NT/2 \{1 + \tfrac{1}{2}[\sin(\pi(f-W/2)/W)(\cos^2(\pi(f-W/2)/W)+2)]\}$$

It is noted that for c = 1 the LFM type signals are obtained. These group delay functions are shown in Fig. 4.14. The constant A_n is obtained by evaluating $\tau_g(f,n,c)$ at one of its end points. Group delays for c ≠ 0 are found in analogous manner. Inverting these functions and performing a second integration, which is easily carried out on a digital computer, the discrete phase modulation functions are obtained.

The ACF's, amplitude spectra and group delays of discrete coded waveforms whose spectra approximate Eq. (4.39) are shown in Fig. 4.15 to 4.17 for WT = 1 and c = 0. The influence of the sampling period T on the shape of the MF response is illustrated in Fig. 4.18.

In general two effects can be observed. First, spectrum tapering results in a widening of the mainlobe. This effect depends on the tapering function, but usually extends about 2 to 3 sampling intervals. The far away sidelobes (k > N/4) are smaller in magnitude than for the QP sequences. Secondly, as in the LFM case, if WT < 1 a further reduction in sidelobe level is observed at the expense of an increase of the close-in ambiguities. For WT > 1, the aliased versions of the ACF will produce spurious response peaks. The range ambiguities, however, are not so pronounced as in the LFM case, because the ridge-like structure is partly destroyed by the non-quadratic phase law (Chapter 9).

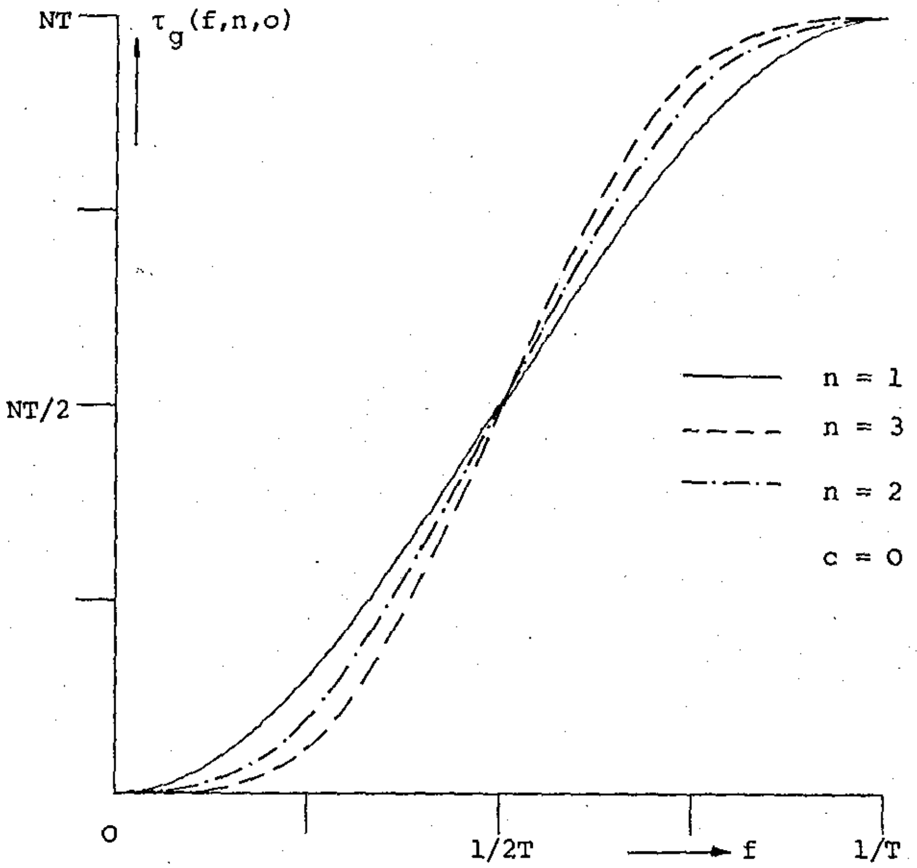
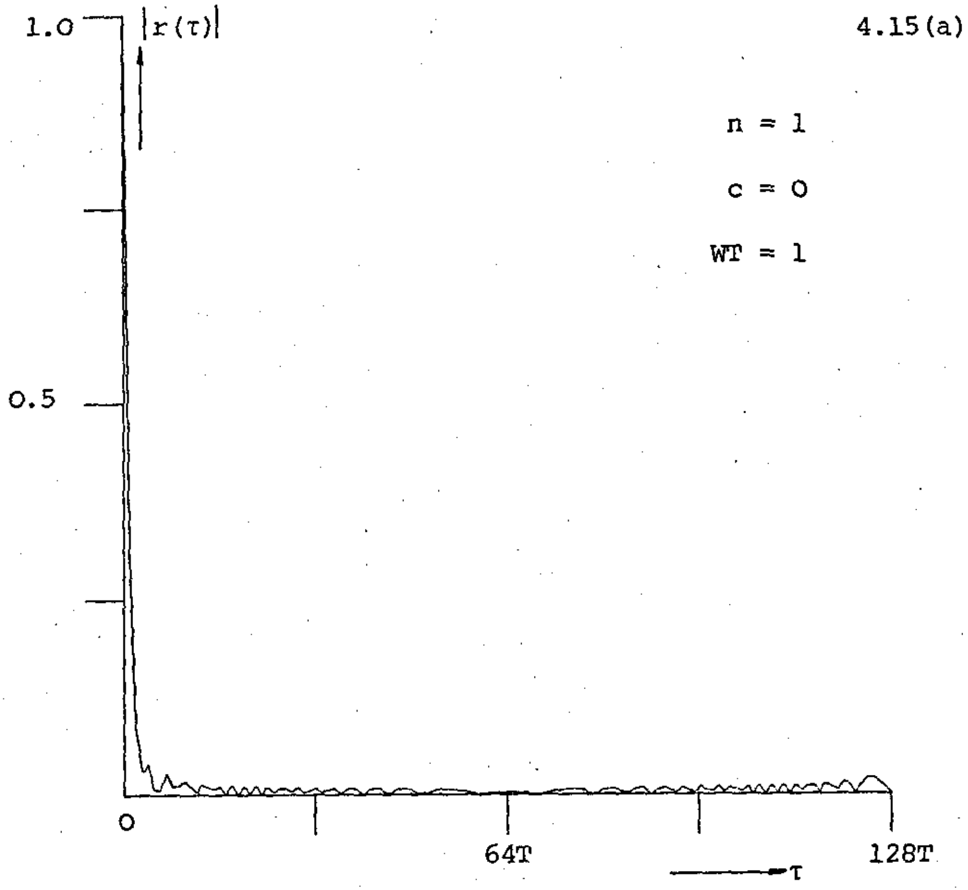


Fig. 4.14 Group delay function for various n.



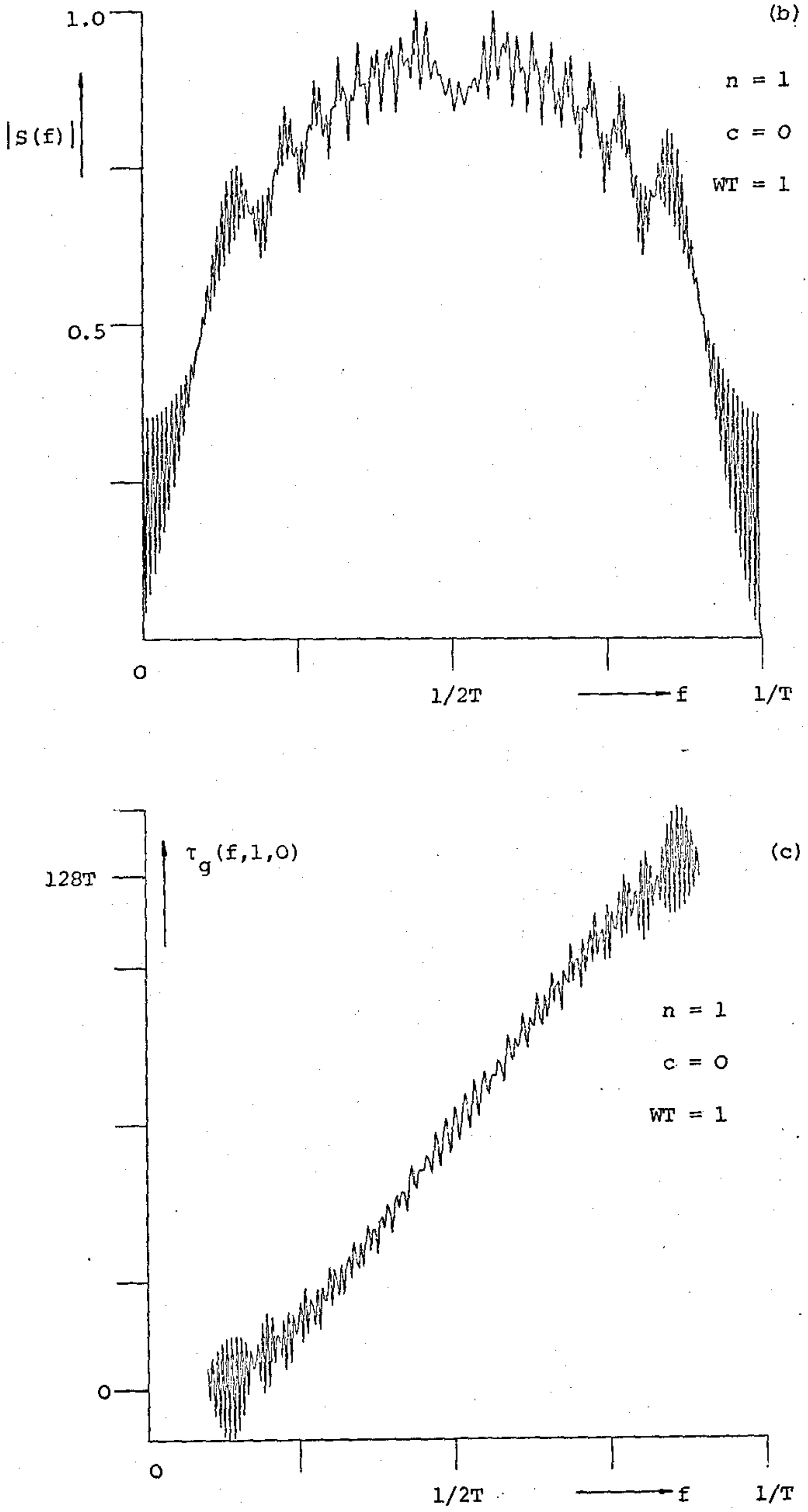
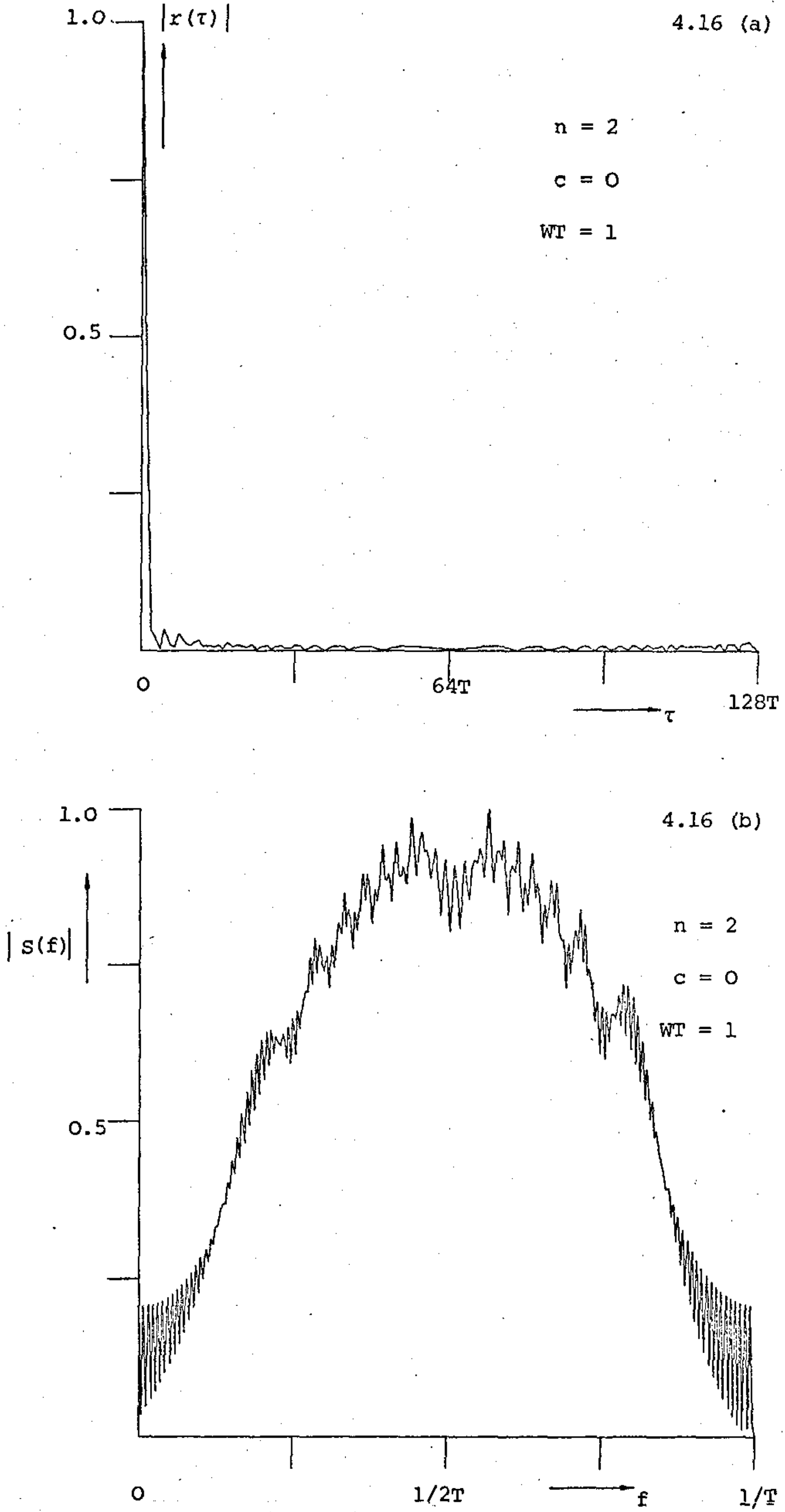


Fig. 4.15 (a) ACF, (b) amplitude spectrum and (c) group delay for non-linear FM type pulse train (n=1).


1.0
|r(τ)|
4.16 (a)
n = 2
c = 0
WT = 1
0.5
0
64T
τ
128T
1.0
4.16 (b)
|S(f)|
n = 2
c = 0
WT = 1
0.5
0
1/2T
f
1/T

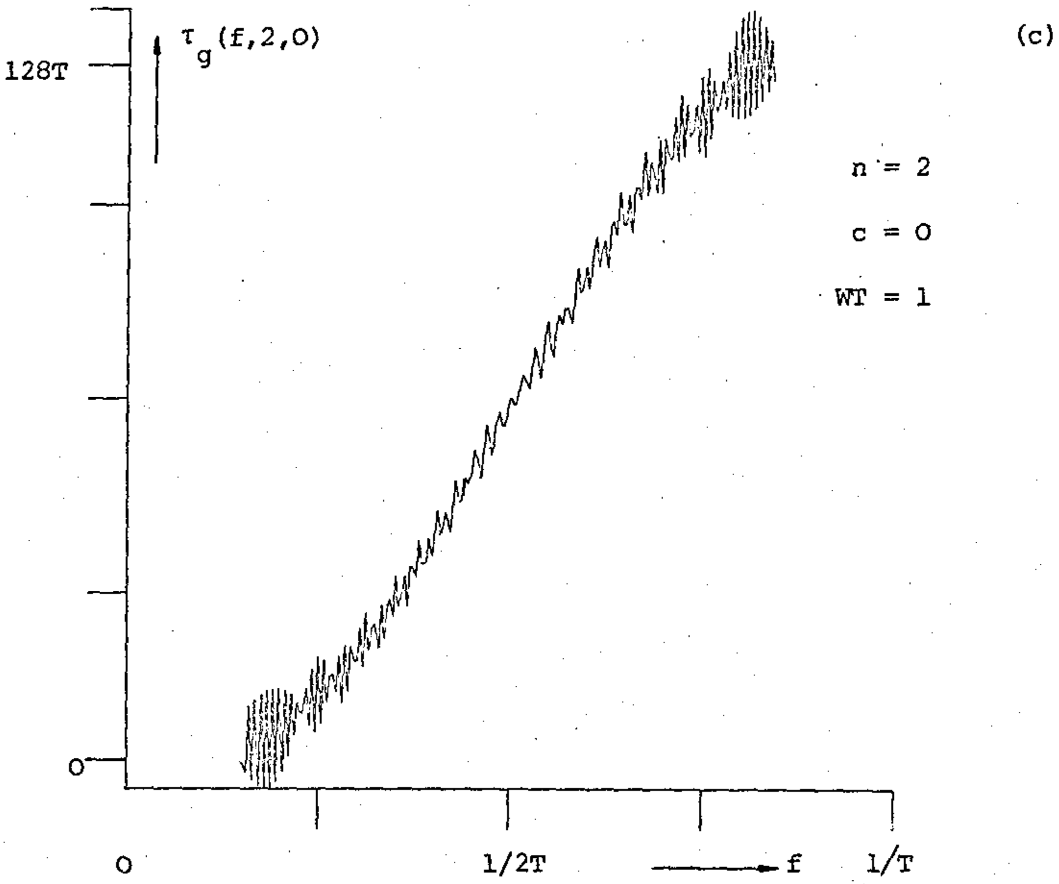
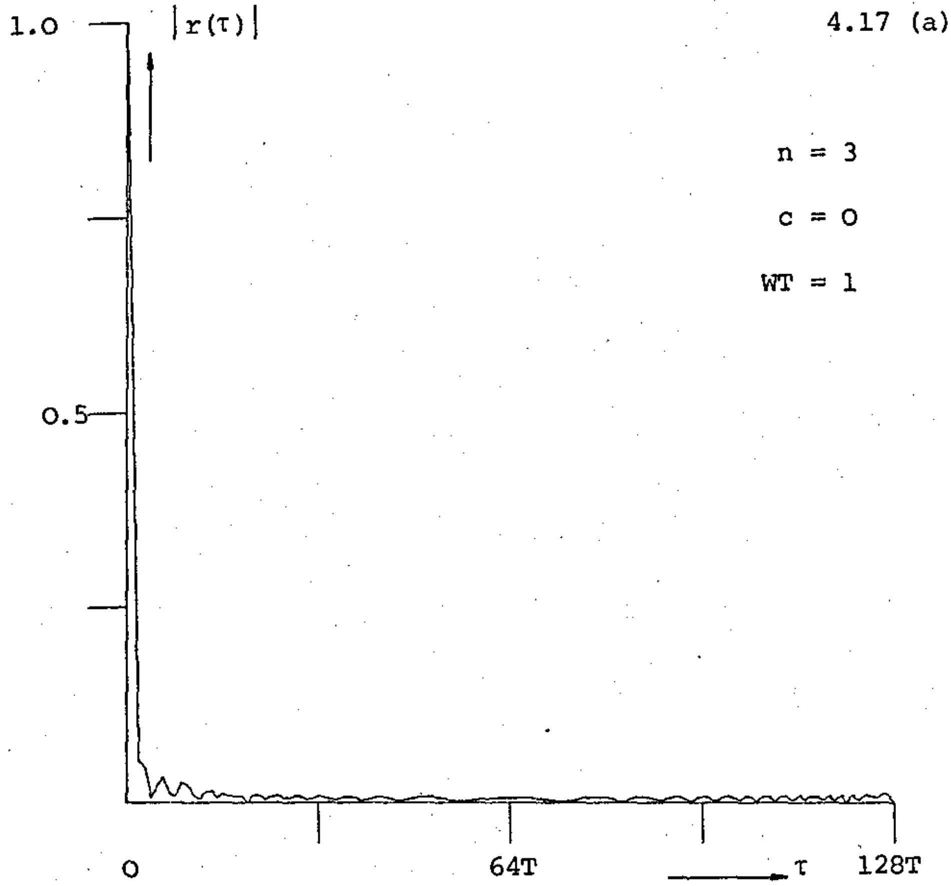
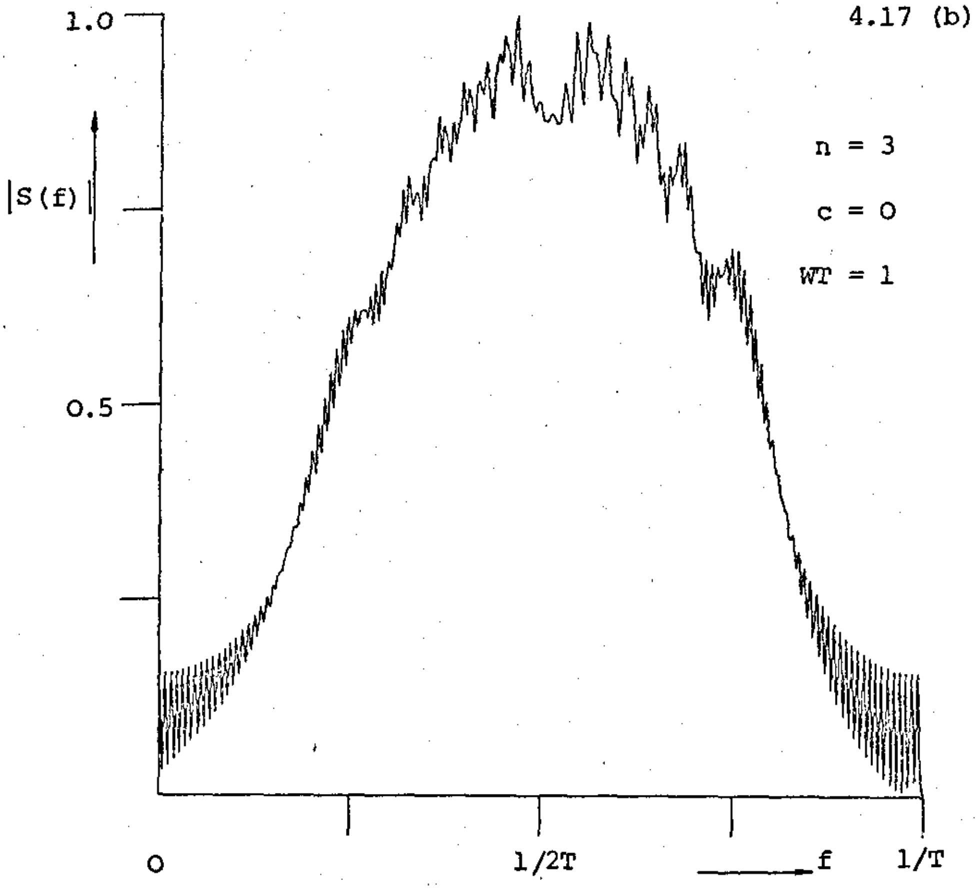


Fig. 4.16 (a) ACF, (b) amplitude spectrum and (c) group delay for non-linear FM type pulse train (n = 2).



4.17 (a)



4.17 (b)

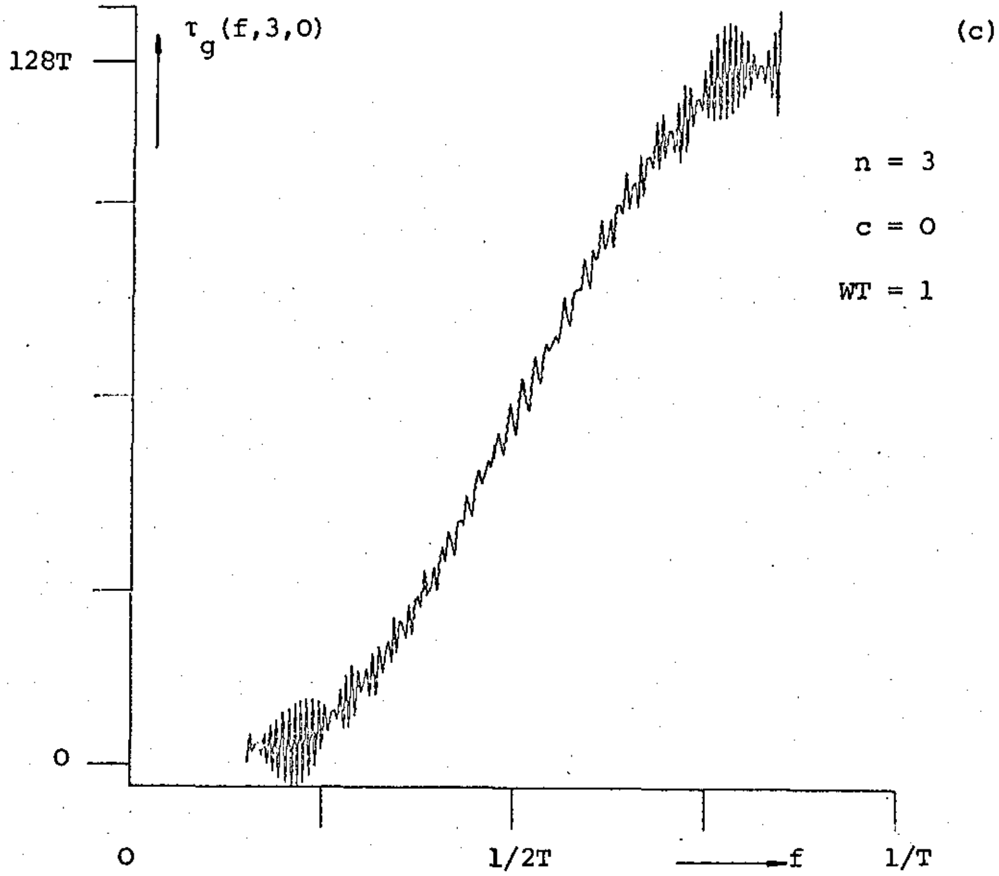


Fig. 4.17 (a) ACF, (b) amplitude spectrum and (c) group delay for non-linear FM type pulse train (n=3).

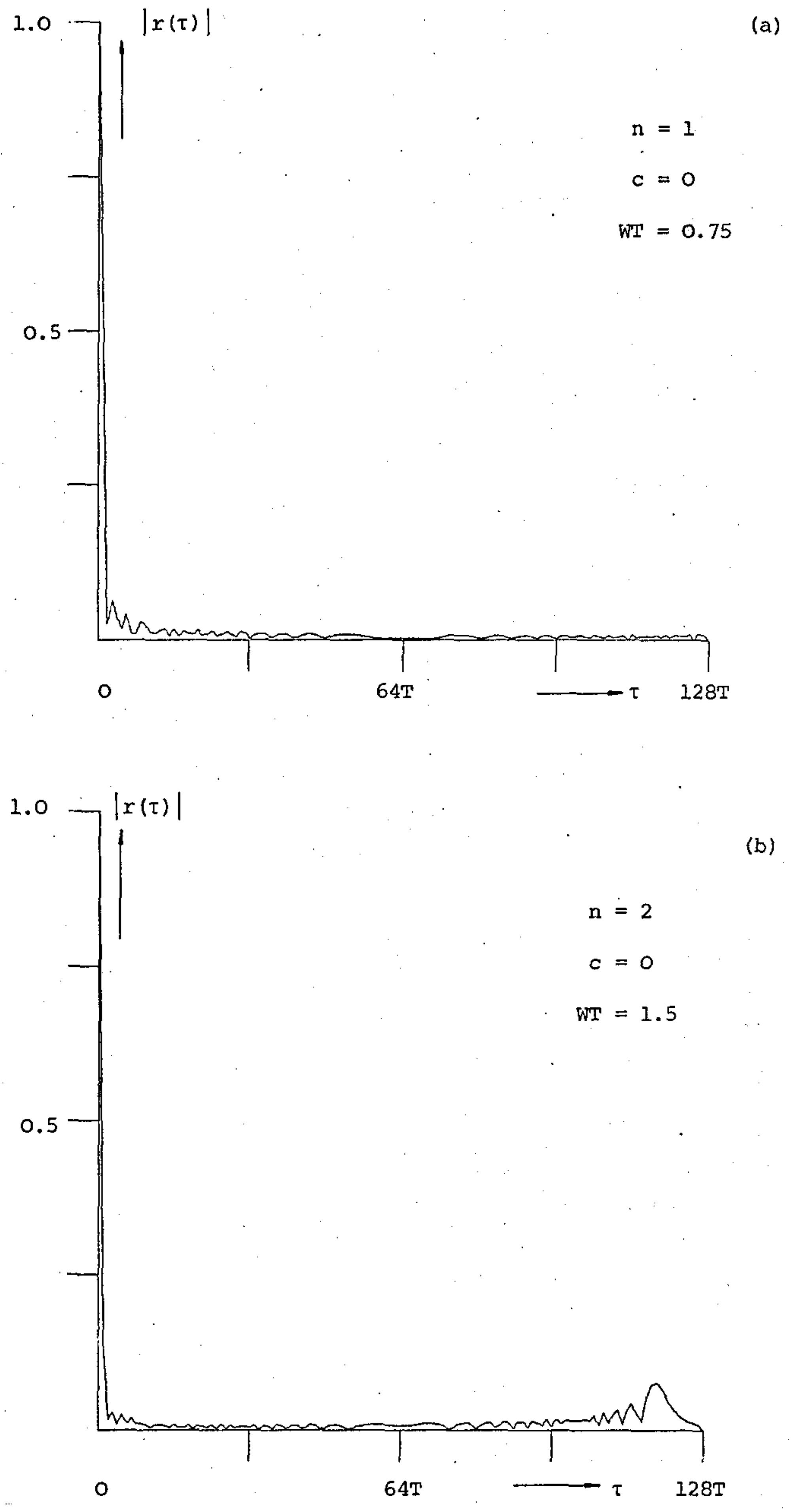


Fig. 4.18 Effects of sampling rate on the ACF of nonlinear FM type pulse trains.

To reduce the mainlobe widening effect one can use Eq. (4.39) to obtain a group delay function for $c \neq 0$. The distinguishing feature between this weighting function and the cosine function is that the amplitude of the spectrum at its edges is not zero. In the analogue case, for example, a cosine-squared spectrum for a value of $c = 0.08$ approximates the familiar Taylor weighted spectrum[22].

The ACF's and spectra for $n = 2$ and a pedestal of $c = 0.08$ and $c = 0.5$ are shown in Fig. 4.19 and Fig. 4.20.

For small values of $c (c \simeq 0.1)$ there is little difference from a purely cosinusoidal taper with no pedestal. On the other hand, for $0.8 < c < 1.0$ one approaches the LFM case. In between these values there is a whole range of non-linear FM type pulse trains each of which contains to some extent the LFM component.

Another interesting property of such pulse trains is their zero pattern which is obtained by factorizing the corresponding ZT polynomial. Similar to the QP signals, Fig. 4.21, the zeros tend to lie on a spiral with approximately half the zeros inside and the other half outside the unit circle. This is illustrated in Fig. 4.22 for a pulse train with $n = 2$, $c = 0$ and $N = 64$. It is conjectured that these types of zero patterns are common to all pulse trains whose spectra are weighted smoothly in a cosinusoidal fashion because of their relatively large LFM content.

4.5 Summary

The stationary phase principle is a useful method for designing continuous signals having a steep monotonic FM law. This requirement precludes the synthesis of pulse trains directly using this method. In addition such an analysis is only approximate in that it does not include the edge effects associated with the signal. This causes Fresnel

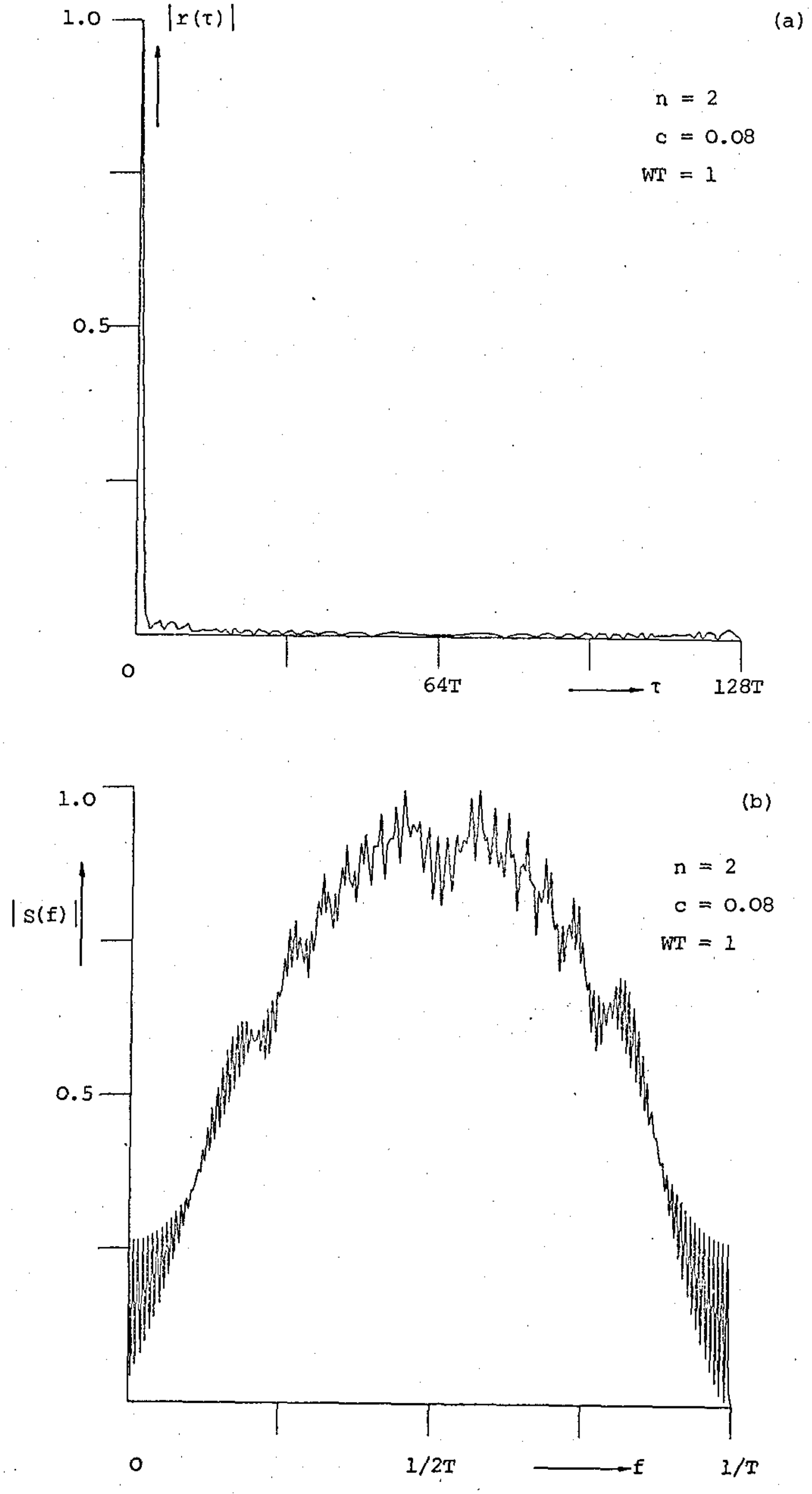


Fig. 4.19 Effects of the pedestal function c on the ACF (a), and amplitude spectrum (b).

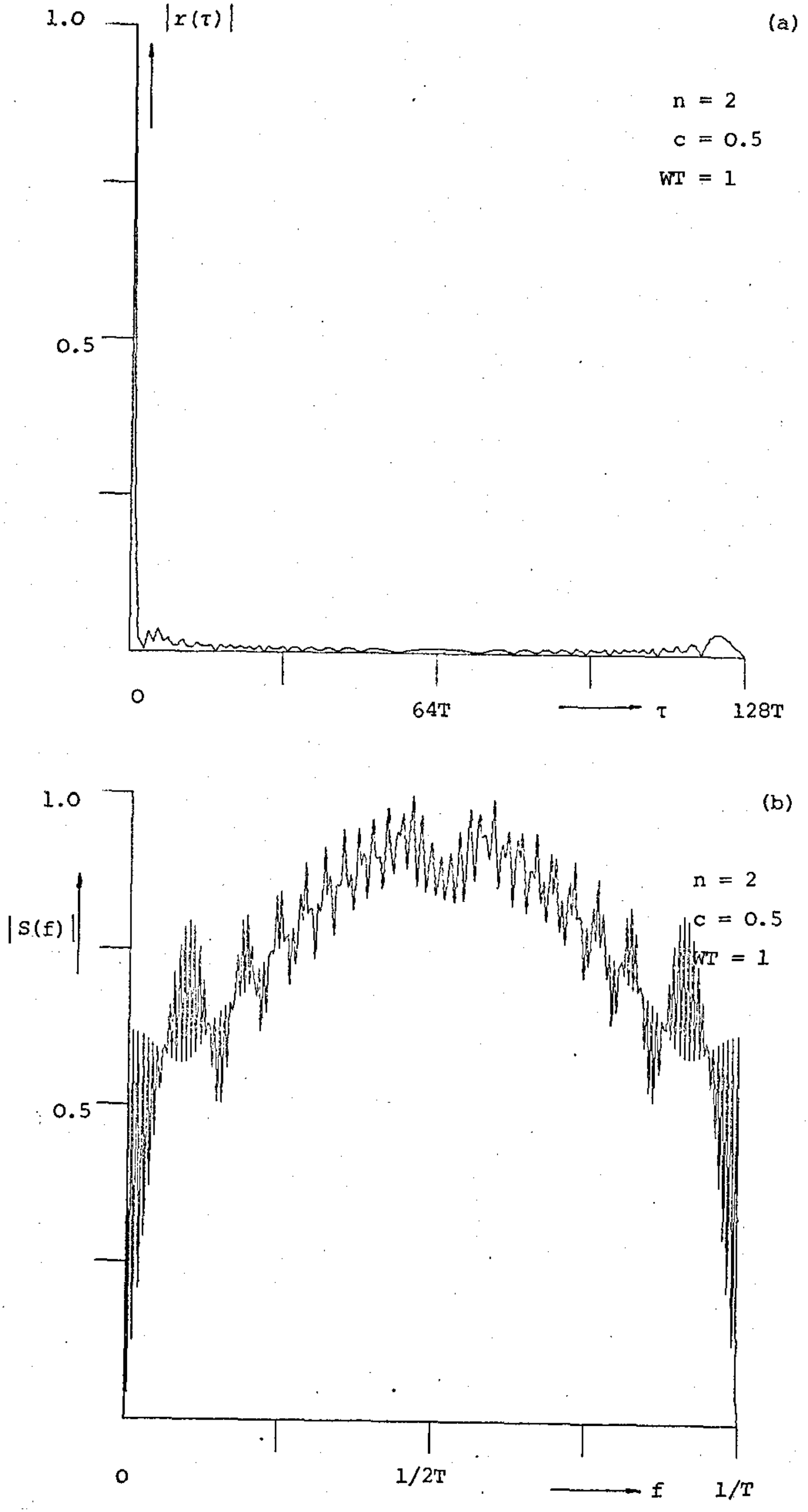


Fig. 4.20 Effects of pedestal function c on the ACF (a), and the amplitude spectrum (b).

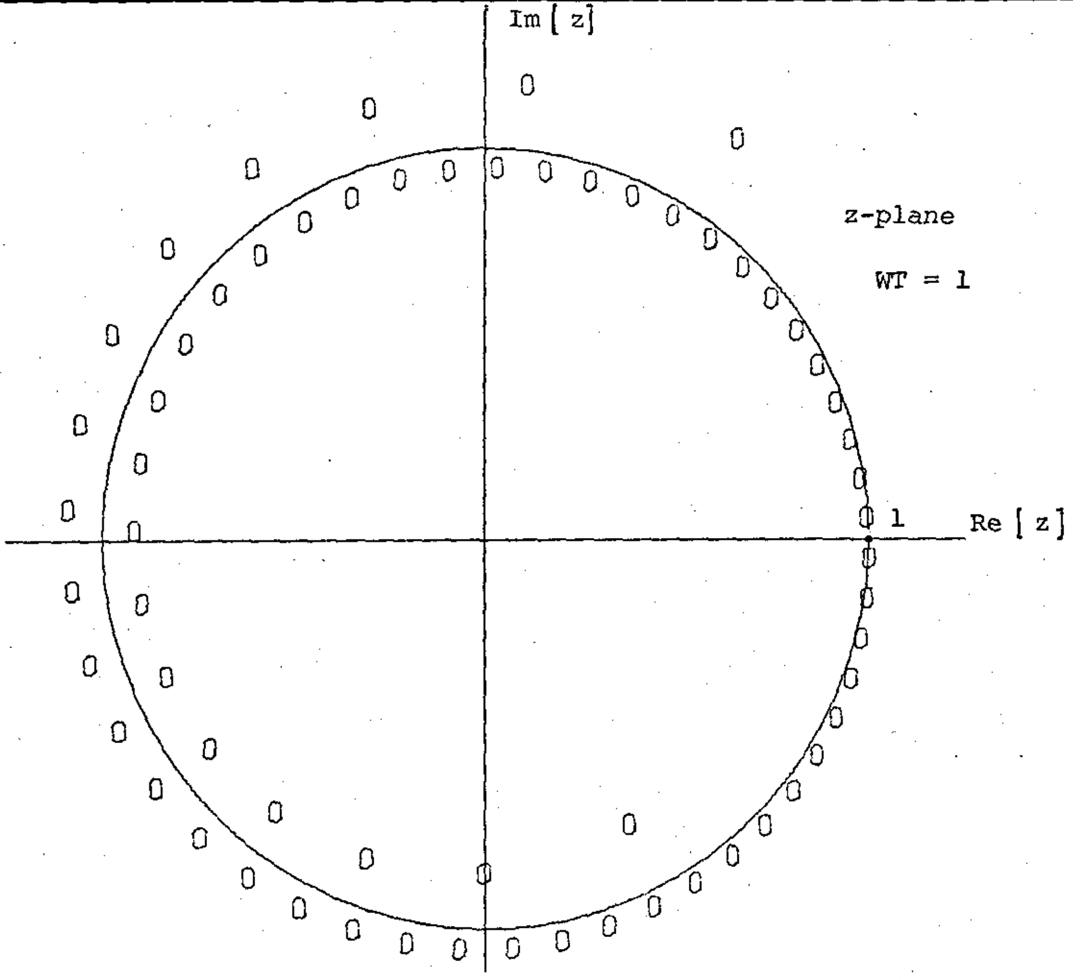


Fig. 4.21 Zero pattern of the 64-element QP code.

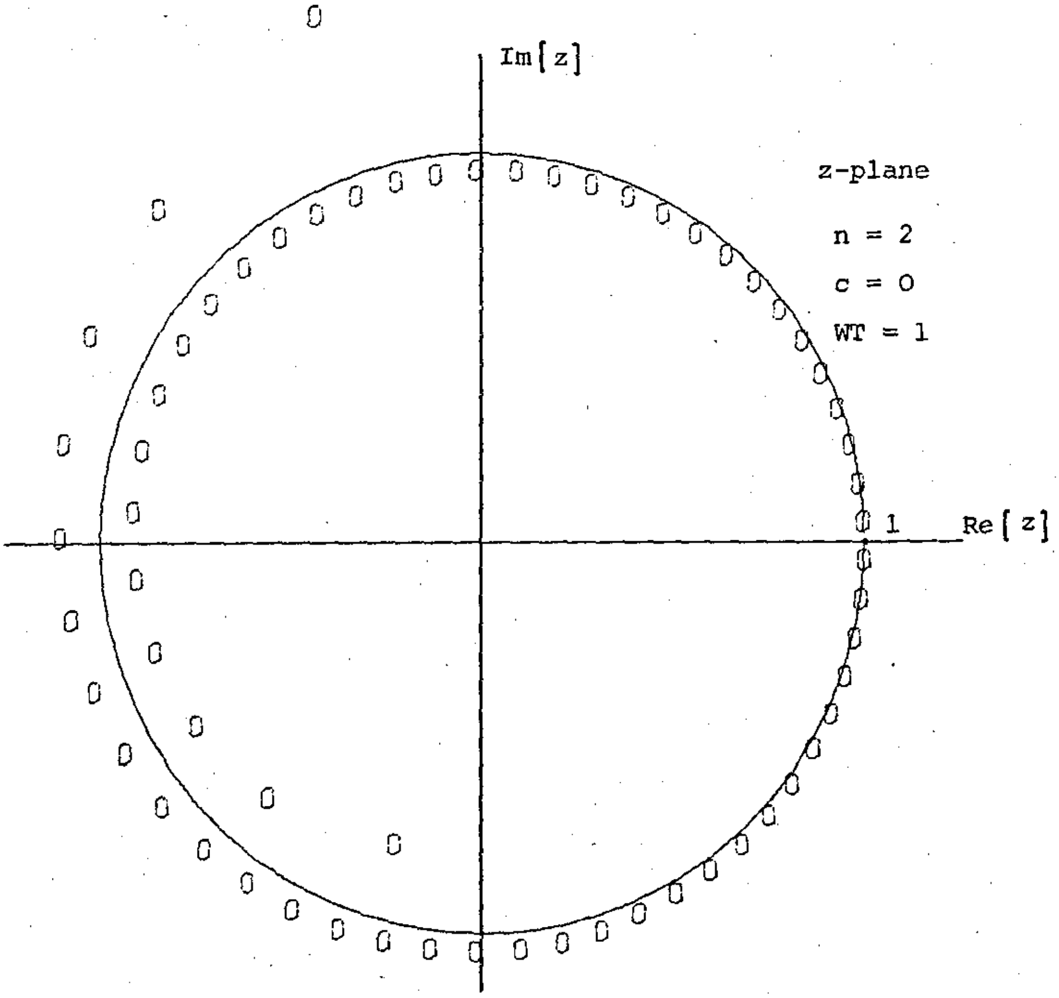


Fig. 4.22 Zero pattern of 64-element non-linear FM type pulse train.

ripples which in turn result in range ambiguities. However, it has been shown that in principle all the desired properties of the FM waveforms can be retained by a discrete phase approximation, provided sampling occurs at the Nyquist rate, i.e. $T = 1/W$. For values other than the Nyquist rate range ambiguities will appear either far or close to the mainlobe depending whether $T \gtrless 1/W$.

Apart from introducing periodicity in the frequency domain a quantization process of this nature produces an additional high oscillatory ripple component in the spectrum, thus causing sidelobe-bands at time shifts where $k \approx NT$ to appear. Since uniform pulse trains have at least one sidelobe of unit magnitude at time shifts $\pm(N-1)T$, there will always be a ripple component of the form $2\cos(2\pi f(N-1)T)$ superimposed on the power spectrum. That ripple component can be reduced by tapering the edges of the pulse train. In practice this could be done by simply increasing the rise time of the transmitted signal, thus minimizing system complexity. Since tapering of the time waveform does not result in a reduction of the effective signal bandwidth, no main peak widening of the ACF is observed. However, the remaining range ambiguity is now concentrated adjacent to the compressed pulse, which may or may not lead to resolution problems. Moreover, for high power radars such an approach to reduce range sidelobes is not readily available because of its inevitable loss in SNR. Therefore, another method that perturbed only the phase of the signal has been investigated. With this method it is possible to effectively reduce the sidelobe-band close to the mainlobe, whilst otherwise the overall ACF is little affected.

For the discrete non-linear FM case it has been shown that the range ambiguities can be controlled to some extent. The method of weighting the power spectrum suffers, however, from the mainlobe widening effect because of the resulting reduction in signal bandwidth. This could seriously affect radar resolution performance if closely spaced targets with widely varying cross-sections are expected. In some cases the pulse widening effect can be reduced by sampling at a slightly lower rate than the Nyquist rate, i.e. $WT = 1.2$ to 1.5. Nevertheless, for many applications this method offers an alternative to the system designer, particularly in cases when LFM type signals are not desirable. Although the phase shifts for the discrete non-linear FM case will have arbitrary values in general, one can subdivide the phase interval of 2π into M increments and select the one closest to the actual value of the phase step. Naturally such a quantization would give rise to an additional increase of the overall sidelobes (see Chapter 6, Section 6.3.1).

At this point it is perhaps of interest to note that discrete phase approximation could also be applied to discrete frequency waveforms. For example,Frank polyphase codes[17] can be regarded as such an approximation to discrete frequency codes. In addition the pulse trains discussed here yield a more flexible means of waveform generation as compared to analogue techniques. For example,it is possible to 'scramble' the ordering of the sub-pulses to eliminate the LFM ambiguity in the delay-doppler plane.

The variations possible with phase coded pulse trains are almost infinite in that the phase, frequency, and time of the transmission of each segment can be varied. Thus these waveforms are compatible with the multi-function requirement of modern radar systems.

The design procedure outlined in this chapter seems to be satisfactory from the practical viewpoint. Unfortunately, methods based on the stationary phase principle are limited to a class of signals whose phase functions vary smoothly with time. Signals with noise-like properties such as binary waveforms cannot be synthesized using this method. In addition if very low range sidelobes are required the stationary phase method is not adequate and the sequences obtained using this technique may be regarded merely as a first order approximation.

In subsequent chapters an essentially different method of synthesizing discrete coded signals with desired autocorrelation properties is described. The method is based on numerical optimization techniques. Such an approach has a number of advantages. First, no restriction on the class of admissible phase functions is necessary. This allows the synthesis of waveforms with noise-like properties. Secondly, no information of the signal's phase structure is usually required. Thirdly, these methods are flexible in a sense that it is possible to control particular sidelobes or sidelobe regions. In general, numerical methods operate in an iterative manner to find the optimum phase functions and are not without their disadvantages. For example, the sequences are found in a completely automatic fashion and in many cases little or no insight is gained from the analytical viewpoint.

However, before proceeding to the formulation of the synthesis problem using optimization techniques some of the basic principles used when dealing with numerical algorithms are pointed out.

CHAPTER 5

THE OPTIMIZATION PROBLEM

5.1 Introduction

The methods available for optimization may be conveniently divided into two distinctly different categories which are classified as analytical and numerical, respectively.

Analytical methods usually employ the mathematical theory of calculus and variational methods. These techniques are well suited to relatively simple functions with a small number of variables, but are not able to handle highly non-linear functions of large dimensionality.

Numerical methods employ, in general, a branch in the field of numerical mathematics known as programming methods. Recent developments in this field are reflecting the rapid growth in computing capacities offered by modern digital computers. The major areas of mathematical programming embrace linear programming, dynamic programming and various types of non-linear programming.

In analytical methods the optimum solution is found exactly through the solution of a set of equations expressing the conditions for optimality. In the numerical methods, on the other hand, near optimal design is sought in an iterative manner. An initial 'guess' is used as a starting point for a systematic search for increasingly better designs. The search is terminated when certain criteria are satisfied which ensure that the current solution is sufficiently close to the true optimum.

While analytical methods require complete mathematical formulation of the function to be m inimized, numerical methods can minimize functions whose structure is unknown but is being explored step by step.

During the last decade there has been a rapid development in programming methods as well as in the application of such methods to design problems. The bulk of the research has been concentrated in the area of linear programming. The contributions have been of such a magnitude that most linear problems can be solved[56] (Fig. 5.1). Non-linear programming, on the other hand, deals with the optimization of non-linear functions subject to linear and/or non-linear constraints. No general method exists to solve non-linear problems in the sense that the simplex algorithm exists to solve problems in which all the functions are linear. Many strategies have been suggested, but more algorithms have been proposed than have been successfully applied. Thus the range of applicability of existing non-linear programming algorithms is limited. This explains the continuing effort to improve known methods or to invent new ones. Therefore, to some extent, non-linear programming still remains an experimental field of research.

In the area of signal design and related topics it is anticipated that the improvements in numerical algorithms, together with the high speed operations offered by modern computers, will attract increasing attention in years to come.

5.2 Fundamentals of Optimization

The discrete nature of pulse trains complicates their synthesis considerably. No ordinary methods can be used in an attempt to design such sequences unless a discrete phase approximation to analogue signals is made. As shown in the previous chapter the method based on the stationary phase principle results in relatively large sidelobes (> - 30 dB) and, moreover, limits the class of admissible waveforms. Instead of synthesizing pulse trains from a given power spectrum one can try to find the actual signal itself which, if not ideal, has at least a satisfactory approximation to the desired ACF. In general, approximation is essential, since the specified ACF is rarely realizable for a given set of constraints on the signal waveform. After an initial solution has been obtained it is

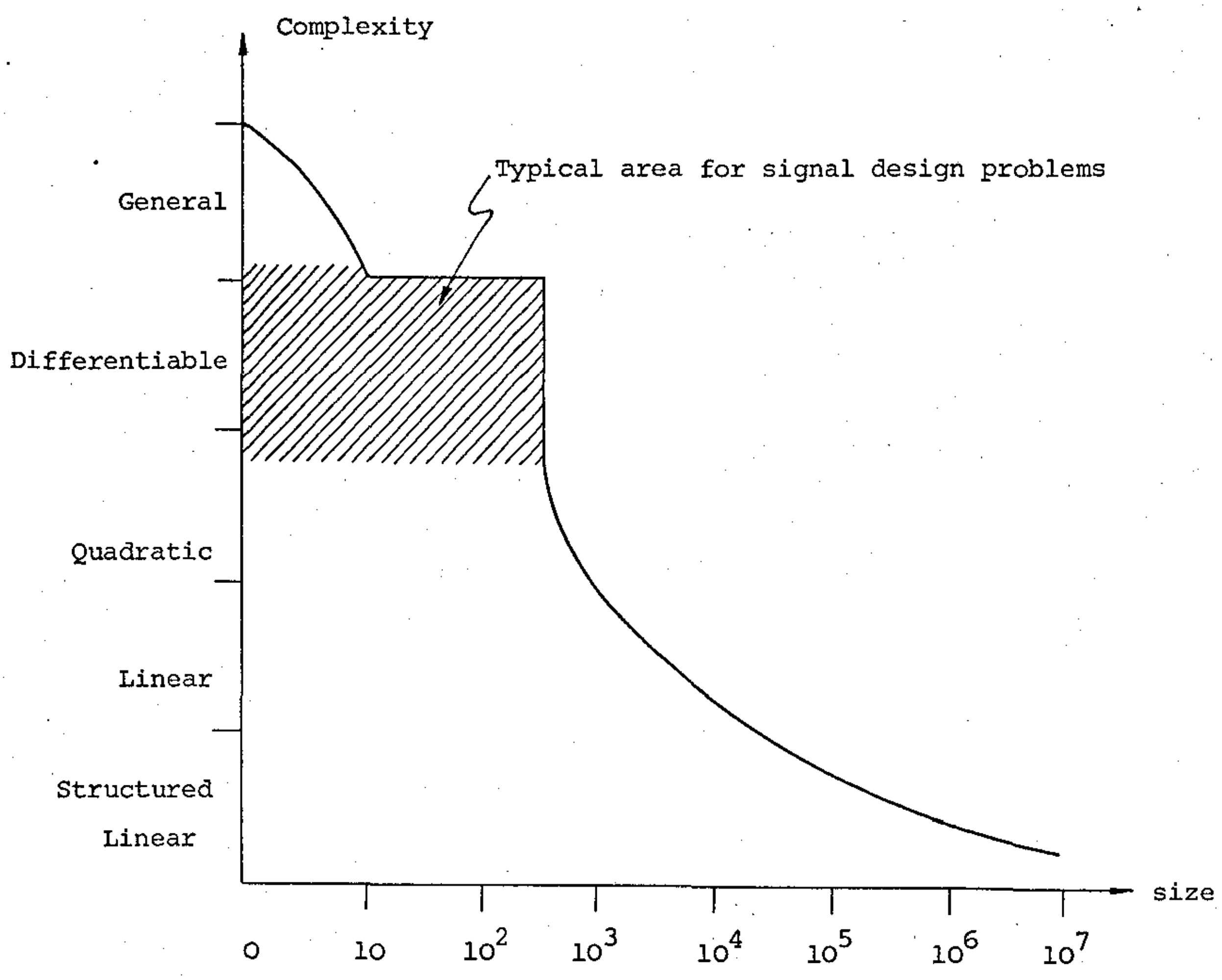


Fig. 5.1 Size of problems that can be handled according to Wolfe[56]. (Size = mean number of equations plus variables).

compared with the required MF response. The result of the comparison is the approximation error and the objective is to reduce that error by modifying the initial solution. It should be noted that such a successive approximation procedure has all the essential properties for automatic control; namely the comparison of actual and desired performance and a feedback path to reduce the approximation error.

Before presenting the optimization techniques used on the class of signal design problems considered here it is necessary to state the design problem in a suitable form.

5.2.1 Formulation of the Synthesis Problem

The most important steps in the design procedure may be outlined, with reference to Fig. 5.2, as follows.

First, at the outset the designer should have a clear understanding of the functions to be performed by the signal. This step includes an evaluation of the specific radar tasks such as multiple-target resolution capability, transmitter power limitations, etc.

Next, the principal waveform type has to be selected. Here the designer has to decide whether to use signals of Group I, II, or III (Chapter 2). The signal under consideration may be characterized by means of a set of design variables in amplitude, $|a_n|$, phase, ϕ_n, and frequency f_n. It is often convenient to replace ordered sequences by vectors to permit the notation of linear vector spaces. Thus

$$\underline{x} = (|a_o|, |a_1|, \ldots, |a_{N-1}|, \phi_o, \phi_1, \ldots \phi_{N-1}, f_o, f_1, \ldots f_{N-1})^T \qquad (5.1)$$

where $\underline{x}$ is the design vector and $(\cdot)^T$ denotes transposition.

The next step is the mathematical formulation of the system. This is expressed as a set of equations

$$h_j(\underline{x}) = 0 \qquad j = 1,2,\ldots,\ell \qquad (5.2)$$

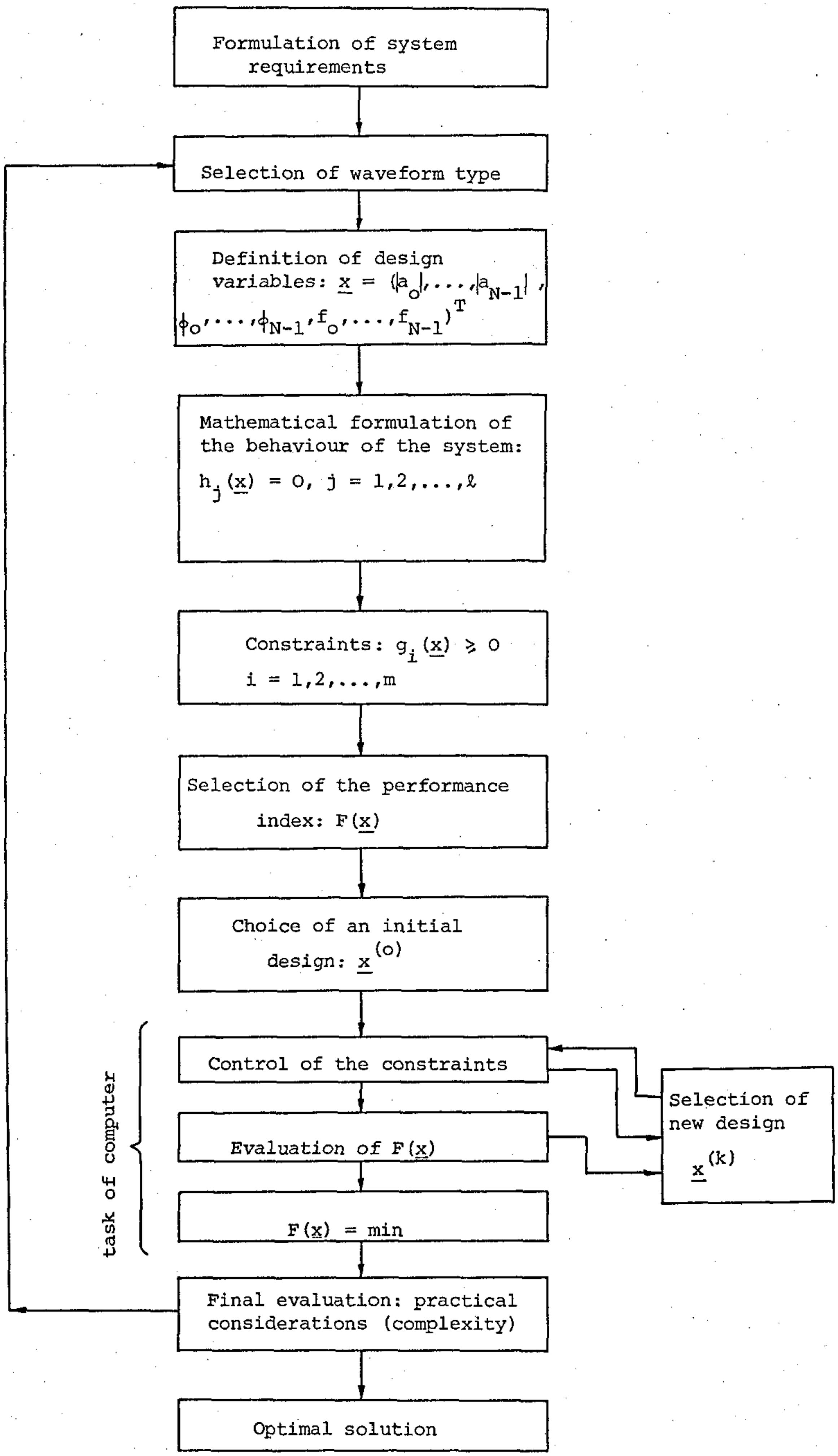

Formulation of system requirements
Selection of waveform type
Definition of design variables: $\underline{x} = (|a_o|,\dots,|a_{N-1}|,\phi_o,\dots,\phi_{N-1},f_o,\dots,f_{N-1})^T$
Mathematical formulation of the behaviour of the system:
$h_j(\underline{x}) = 0,\ j = 1,2,\dots,\ell$
Constraints: $g_i(\underline{x}) \geqslant 0$
$i = 1,2,\dots,m$
Selection of the performance index: $F(\underline{x})$
Choice of an initial design: $\underline{x}^{(o)}$
task of computer
Control of the constraints
Evaluation of $F(\underline{x})$
$F(\underline{x}) = \min$
Selection of new design
$\underline{x}^{(k)}$
Final evaluation: practical considerations (complexity)
Optimal solution

Fig. 5.2 Flow diagram of the design procedure.

In most practical cases there are restrictions on the permissible values of the design variables which may be written in the form

$$g_i(\underline{x}) \geqslant 0 \qquad i = 1,2,\ldots,m \qquad (5.3)$$

These constraints exclude undesirable solutions for a particular application.

All solutions which simultaneously satisfy Eq. (5.2) and Eq. (5.3) correspond to acceptable or feasible designs. The size of the feasible region or solution space will, of course, depend on the constraints and the system behaviour. It is even possible that the solution space is empty for a particular set of constraints. In this case it becomes necessary to relax the conditions above.

At this point a criterion or objective function must be adopted to determine when the 'best' solution is arrived at for a given degree of complexity. There are several alternative ways to define a measure of best performance. For example one may choose to minimize the sum of the squares of the errors, the absolute error, the sum of the absolute errors, and so forth. The choice of any of these criteria is usually dependent on many factors, the principal ones being the tasks the system has to perform. In fact if the optimization technique is adequate, the performance index will completely determine the final system. In the present case the criterion may be written as a scalar function, $F(\underline{x})$, of the design variables $\underline{x}$. In general $F(\underline{x})$ will be a non-linear function in $\underline{x}$.

At this stage a method to search for the optimum solution remains to be chosen. Several techniques exist for minimizing non-linear functions some of which will be described subsequently. Having found the optimum solution the designer must, of course, himself undertake the final judgement of the solution obtained.

The signals for the class of design problems considered here are the a.m.ph.m. pulse trains given by

$$s(t) = \sum_{n=0}^{N-1} a_n \operatorname{rect}(t/T - n) \tag{5.4}$$

The function of interest is the sampled square magnitude of the MF response

$$|r(k)|^2 = \Big| \sum_{n=0}^{N-1-k} a_n \, a^*_{n+k} \Big|^2 \tag{5.5}$$

$$k = 0,1,\ldots,(N-1)$$

The aim is to find the design vector $\underline{x} = (|\underline{a}|, \underline{\phi})^T$ which minimizes the objective function or performance index*

$$F(\underline{x}) = \sum_{k=0}^{N-1} f\{|r(k)|^2 - |\hat{r}(k)|^2\} \tag{5.6}$$

where $\hat{r}(k)$ is the desired ACF and $f(\cdot)$ is a suitable function of interest. For most practical applications the signal will be subjected to the constraints

$$|a_n| = 1 \qquad n = 0,1,2,\ldots,(N-1) \tag{5.7}$$

Eq. (5.6) and the above conditions represent a constrained non-linear optimization problem. It should be noted that minimization of the real valued function $F(\underline{x})$ in general involves 2N real variables; namely, the $|a_n|$ and the ϕ_n, subject to certain constraints. It is possible, however, to transform the above constrained problem into an unconstrained minimization problem. In this case the design vector $\underline{x}$ can be written as

$$\underline{x} = (|a_0|, |a_1|, \ldots\ldots, |a_{N-1}|, \phi_0, \phi_1, \ldots, \phi_{N-1})^T \tag{5.8}$$

* Also known as error criterion or cost function.

The objective is to minimize

$$\min_{\underline{x}} \; F(\underline{x}) = F(|a_o|,|a_1|,\ldots,|a_{N-1}|,\phi_o,\phi_1,\ldots,\phi_{N-1}) \qquad \underline{x} \in E^{2N} \tag{5.9}$$

Subject to

$$|a_n| = 1 \qquad n = 0,1,\ldots,(N-1)$$

hence

$$F(\underline{x}) = F(1,1,\ldots,1,\phi_o,\phi_1,\ldots,\phi_{N-1}) = F(\phi_o,\phi_1,\ldots,\phi_{N-1}) \tag{5.10}$$

In other words the minimization of $F(\underline{x})$ in the feasible domain reduces to the unconstrained problem

$$\min_{\underline{\phi}} F(\underline{\phi}) \qquad \underline{\phi} \in E^N \tag{5.11}$$

where E^N denotes the N-dimensional space of real column vectors.

The solution is accomplished by finding $|r(k)|^2$ in terms of the phase vector $\underline{\phi}$.

$$|r(k)|^2 = |r(k,\underline{\phi})|^2 \tag{5.12}$$

The error criterion $F(\underline{\phi})$ can now be derived for certain forms of $f(\cdot)$, such as $f = (|r|^2 - |\hat{r}|^2)^2$, or $f = ||r|^2 - |\hat{r}|^2|$. Here the case of the mean-square optimization is considered. Thus,

$$f = \{|r|^2 - |\hat{r}|^2\}^2 \tag{5.13}$$

The objective function $F(\underline{\phi})$ then becomes

$$F(\underline{\phi}) = \sum_{k=o}^{N-1} \{|r(k,\underline{\phi})|^2 - |\hat{r}(k)|^2\}^2 \tag{5.14}$$

The first terms of a multi-dimensional Taylor series expansion of $F(\underline{\phi})$ are given by

$$F(\underline{\phi} + \Delta\underline{\phi}) = F(\underline{\phi}) + \nabla F^T \Delta\underline{\phi} + 1/2\Delta\underline{\phi}^T H \Delta\underline{\phi} + \ldots \tag{5.15}$$

where

$$\Delta\underline{\phi} = \text{col } (\Delta\phi_o, \ldots\ldots, \Delta\phi_{N-1}) \tag{5.16}$$

and

$$\nabla F = \text{col}\ (\partial F/\partial\phi_o, \ldots., \partial F/\partial\phi_{N-1}) \tag{5.17}$$

is the error gradient vector. The symmetric NxN matrix H

$$\nabla(\nabla F)^T = H = \begin{bmatrix} \partial^2 F/\partial\phi_o^2 \ldots\ldots \partial^2 F/\partial\phi_o\partial\phi_{N-1} \\ \begin{matrix} \cdot & \qquad\qquad & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \end{matrix} \\ \partial^2 F/\partial\phi_{N-1}\partial\phi_o \ldots.. \partial^2 F/\partial\phi_{N-1}^2 \end{bmatrix} \tag{5.18}$$

contains the second partial derivatives of $F(\underline{\phi})$ and is called the Hessian matrix. *For* $F(\underline{\phi})$ *to be a minimum at* $\hat{\underline{\phi}}$, *the following* conditions have to be satisfied

$$\nabla F(\hat{\underline{\phi}}) = 0 \tag{5.19}$$

and the Hessian matrix H must be positive definite[57].

Thus,

$$\partial/\partial\phi_i\{F(\underline{\phi})\} = \partial/\partial\phi_i \sum_{k=o}^{N-1} \{|r(k,\underline{\phi})|^2 - |\hat{r}(k)|^2\}^2 = 0 \tag{5.20}$$

$$i = 0,1,\ldots,(N-1)$$

Expanding the above equation, by squaring, leads to

$$\partial/\partial\phi_i\{\sum_{k=o}^{N-1} |\hat{r}(k)|^4 - 2\sum_{k=o}^{N-1} |\hat{r}(k)|^2 |r(k,\underline{\phi})|^2 + \sum_{k=o}^{N-1} (|r(k,\underline{\phi})|^2)^2\} = 0$$

which is equal to

$$\sum_{k=o}^{N-1} |\hat{r}(k)|^2 \partial/\partial\phi_i |r(k,\underline{\phi})|^2 - \sum_{k=o}^{N-1} |r(k,\underline{\phi})|^2 \partial/\partial\phi_i |r(k,\underline{\phi})|^2 = 0 \tag{5.21}$$

$$i = 0,1,\ldots.,(N-1)$$

To solve Eq. (5.21) the two terms $|r(k,\underline{\phi})|^2$ and $(\partial/\partial\phi_i)|r(k,\underline{\phi})|^2$ need to be expressed explicitly.

The expression $|r(k,\underline{\phi})|^2$ is simply given by

$$|r(k,\underline{\phi})|^2 = \{\sum_{n=o}^{N-1-k} \cos(\phi_n - \phi_{n+k})\}^2 + \{\sum_{n=o}^{N-1-k} \sin(\phi_n-\phi_{n+k})\}^2 \tag{5.22}$$

The partial derivatives are readily obtained

$$(\partial/\partial\phi_i)|r(k,\underline{\phi})|^2 = 2\{\sum_{n=o}^{N-1-k} \sin(\phi_n - \phi_{n+k})\left[\cos(\phi_i - \phi_{i+k}) - \cos(\phi_{i-k}-\phi_i)\right]$$

$$- \sum_{n=o}^{N-1-k} \cos(\phi_n - \phi_{n+k})\left[\sin(\phi_i - \phi_{i+k}) - \sin(\phi_{i-k} - \phi_i)\right]\}$$

$$= 2|r(k,\underline{\phi})|^2 \,\mathrm{Re}\{\frac{\partial/\partial\phi_i \; r(k,\underline{\phi})}{r(k,\underline{\phi})}\} \tag{5.23}$$

It can be seen that $|r(k,\underline{\phi})|^2$ is independent of $\underline{\phi}$ for $k = 0$ and $k = N-1$. Moreover, one phase variable can be chosen arbitrarily, for example $\phi_o = 0$, since only the relative difference $(\phi_n - \phi_{n+k})$ is of importance.

These expressions can now be substituted into Eq. (5.21) to produce a set of non-linear variational equations.

$$\sum_{k=o}^{N-1} |\hat{r}(k)|^2 \, |r(k,\underline{\phi})|^2 \,\mathrm{Re}\{\frac{\partial/\partial\phi_i \; r(k,\underline{\phi})}{r(k,\underline{\phi})}\}$$

$$- \sum_{k=o}^{N-1} |r(k,\underline{\phi})|^4 \,\mathrm{Re}\{\frac{\partial/\partial\phi_i \; r(k,\underline{\phi})}{r(k,\underline{\phi})}\} = 0 \tag{5.24}$$

$$i = 1,2,\ldots,N-1$$

For practical cases the desired ACF, $\hat{r}(k)$, is usually set equal to zero for $k \neq 0$. The condition for optimality thus reduces to

$$\sum_{k=1}^{N-1} |r(k,\underline{\phi})|^2 \,\mathrm{Re}\{\frac{\partial/\partial\phi_i \; r(k,\underline{\phi})}{r(k,\underline{\phi})}\} = 0 \tag{5.25}$$

$$i = 1,2,\ldots,N-1$$

The above development describes the general synthesis problem for phase modulated pulse trains.

In principle, the optimal solution could be found by solving the set of non-linear equations (5.25) for the unknown variables $\underline{\phi}$ employing numerical techniques. For any reasonable complex problem, however, this straightforward approach is, unfortunately, unsound for numerical analysis. In addition the solution of the complicated functional relationship (Eq. 5.25) is equivalent in complexity to the optimization problem itself (Eq. 5.11). Therefore, it may be argued that an attempt to minimize the criterion $F(\underline{\phi})$ directly using non-linear optimization methods might be a more practical approach. However, for direct minimization a slight modification of the performance index is suggested. This modification should reflect the multiple-target resolution requirements.

For pulse compression sequences two characteristics are of particular concern (Chapter 2). One is the total sidelobe energy given by

$$E_s = \sum_{k=1}^{N-1} |r(k)|^2 \tag{5.26}$$

In a dense-target environment the self-clutter power at the MF output is proportional to this quantity. (Since the autocorrelation is an even function, only one half is considered, i.e. the actual sidelobe energy would be twice the value given here). Another property of interest is the peak sidelove level, $\max_k |r(k)|$, which represents a source of mutual interference that can obscure weaker targets. Therefore, it is required to minimize a suitable measure which incorporates these characteristics. Naturally, any such measure is to some extent arbitrary, but in general will be of the form $F(|r(k)|)$. Again specifying the desired ACF as being zero ($k \neq 0$), the objective function may be redefined as

$$F_p(\underline{\phi}) = \sum_{k=1}^{N-1} w(k)\ |r(k,\underline{\phi})|^p \tag{5.27}$$

where $p \geq 1$ and $w(k) \geq 0$. The weighting sequence $w(k)$ allows control of the sidelobe structure of the resulting ACF. A good compromise is to set $w(k) = 1$ for all k, if no, or little, prior information about the radar environment is known.

The minimization of $F_p(\underline{\phi})$ is often referred to as the least pth approximation[58-60]. The error gradient of this type of objective function is given by

$$\nabla F_p(\underline{\phi}) = \sum_{k=1}^{N-1} \mathrm{Re}\{p|r(k,\underline{\phi})|^{p-2}\, r^*(k,\underline{\phi})\nabla r(k,\underline{\phi})\} \qquad 2 \leq p \leq \infty;\ \text{even integer} \tag{5.28}$$

Provided the derivatives $\nabla r(k,\underline{\phi})$ are available, a suitable gradient method could be used to minimize $F_p(\underline{\phi})$.

Incidentally certain optimization problems can only satisfactorily be described using several error criteria. A suitable overall performance index could then consist of a linear combination of functions of the form (5.27), i.e.

$$F = \lambda_1 F_p^{(1)} + \lambda_2 F_p^{(2)} + \lambda_3 F_p^{(3)} + \ldots\ldots$$

where the λ_i-weighting factors would be given values according to the importance of $F_p^{(1)}$, $F_p^{(2)}$, ... etc. (See Chapter 6, Section 6.5).

5.2.2 Properties of the Performance Index $F_p(\underline{x})$

For a rigorous formulation, the synthesis problem requires the solution of

$$r(k,\underline{x}) = \begin{cases} E, & k = 0 \\ 0, & k = \pm 1, \pm 2, \ldots, \pm(N-2) \\ 1, & k = \pm(N-1) \end{cases} \tag{5.29}$$

subject to

$$|a_n| = 1 \qquad n = 0,1,\ldots,(N-1)$$

where E is the energy of the sequence given by

$$E = \sum_{n=o}^{N-1} |a_n|^2 = \|\underline{a}\|^2 \tag{5.30}$$

The above conditions form a set of (N-1) non-linear equations in the unknowns $(\phi_1, \phi_2, \ldots\ldots, \phi_{N-1})$. As was pointed out previously one of the unknowns ϕ_n can be chosen arbitrarily, i.e. $\phi_o = \phi_{N-1}$. It is not possible to solve this system of equations by ordinary methods. Therefore, a formulation which is equivalent to the solution of Eq. (5.29) has been adopted in the form of the constrained minimization problem

$$\min_{\underline{x}} F_p(\underline{x}) = \sum_{k=1}^{N-1} |r(k,\underline{x})|^p \tag{5.31}$$

subject to

$$r(0,\underline{x}) = E$$

$$|a_n| = 1 \qquad \text{for all } n$$

which is equivalent to the unconstrained minimization problem, $\min_{\underline{\phi}} F_p(\underline{\phi})$. Clearly, any solution satisfying Eq. (5.29) will minimize $F_p(\underline{x})$. If there are no constraints on the admissible values for $|a_n|$, it can be shown (Chapter 7) that there exist 2^{N-1} possible solutions. Hence, for the unconstrained case the objective function $F_p(\underline{x})$ has at least 2^{N-1} relative minima with a value of unity. In addition if $\underline{x}_o = (|\underline{a}|,\underline{\phi})^T$ is a vector that minimizes $F_p(\underline{x})$, then $\underline{x}_1$ given by

$$\underline{x}_1 = (|\underline{a}|,\ \underline{\phi} + \Delta\underline{\phi})^T$$

where

$$\Delta\phi_n = \alpha + n\beta \qquad\qquad n = 0,1,2,\ldots,(N-1)$$

and α,β are arbitrary constants, also minimizes $F_p(\underline{x})$. This means that the performance index is invariant to an arbitrary constant (α) or linear (β) phase shift, since

$$r(k,\underline{x}_1) = e^{-j\beta k} \quad r(k,\underline{x}_o) \tag{5.32}$$

Hence

$$|r(k,\underline{x}_1)| = |r(k,\underline{x}_o)| \tag{5.33}$$

In this sense, therefore, $F_p(\underline{x})$ has an infinite number of minima.

The introduction of the constraints $|\underline{a}| = 1$, reduces the solution space drastically. The size of the domain of convergence depends, of course, upon the system of equations. In general this size is inversely related to the degree and number of equations. While for two simultaneous second order equations almost any initial estimate will lead to one of the roots, it may be very difficult to obtain an initial estimate from which the iteration converges for a larger number of equations. In fact, no solution for (N-1) > 3 that satisfies Eq. (5.29) exactly is known so far. This suggests that the domain of convergence for the constrained set of equations is either very small indeed or even zero (Chapter 7). Another difficulty in finding a solution is the addition of a multitude of local minima due to the constraints. Therefore, any attempt to find the global minimum would require an extensive search procedure.

At this point the value of p remains to be chosen. It should be noted that for p = 2 minimization of

$$F_p(\underline{\phi}) = \sum_{k=1}^{N-1} |r(k,\underline{\phi})|^p \tag{5.34}$$

results in minimizing the sidelobe energy E_s. However, for p = 2 and sequence lengths N > 20 the measure reacts weakly to large peak values, while for large p it will not respond to the energy criterion, since for well-behaved functions

$$\lim_{p\to\infty} \{ \sum_{k=1}^{N-1} |r(k,\underline{\phi})|^p\}^{1/p} = \max_k |r(k,\underline{\phi})| \tag{5.35}$$

The measures $(F_p)^{1/p}$ are called Chebyshev or uniform norms because of the consequences of Eq. (5.35). Minimization with respect to $(F_p)^{1/p}$ for large p is often referred to as minimax approximation[59].

In other words, the least pth approximation tends to the minimax approximation as $p \to \infty$. Hence as suggested by Eq. (5.35) a Chebyshev solution could be approached in principle, by successively minimizing F_p each time incrementing the index p, i.e. $p = 2,4,6,\ldots$ etc. In most cases acceptable minimax approximations can be obtained with relatively moderate values of p ($p \simeq 10$).

It is not clear which characterizes the 'goodness' of a sequence more fully, the maximum peak value or the energy criterion. Therefore, for short sequences ($N < 20$) $p = 2$ should be adequate, while for longer sequences a larger value for p, for example, $p = 4$ has the desirable effect of reducing large single sidelobe peaks.

At this point a word about the significance of the weighting function w(k) (Eq. (5.27)) and the index p is appropriate. Generally speaking their purpose is to emphasize large errors. However, the use of a weighting function to achieve this would require prior knowledge where these large errors occur. Consequently, a weighting function may be used as a means to emphasize certain sidelobe regions, but it is a poor approach for enhancing single large errors.

For $p = 2$ there exists an equivalent formulation of the criterion $F_2(\underline{\phi})$ in the frequency domain. Using Parseval's theorem Eq. (5.34) can be written as

$$\sum_{k=1}^{N-1} \left|r(k,\underline{\phi})\right|^2 = T/2 \int_{-1/2T}^{1/2T} \left(\left|S(f)\right|^2 - N\right)^2 df \tag{5.36}$$

where the power spectrum $|S(f)|^2$ is given by

$$\left|S(f)\right|^2 = N + 2 \sum_{k=1}^{N-1} \left|r(k,\underline{\phi})\right| \cos(2\pi fkT + \alpha_k) \tag{5.37}$$

and

$$\alpha_k = \tan^{-1} \left\{ \frac{\mathrm{Im}\left[r(k,\underline{\phi})\right]}{\mathrm{Re}\left[r(k,\underline{\phi})\right]} \right\}$$

The power spectrum consists of a constant term N and a ripple spectrum. Thus, minimal sidelobe energy is equivalent to requiring minimum energy in the ripple components of the spectrum.

It might also be of interest to note that the performance index defined by Eq. (5.34) is known as the class of ℓ_p-measures. For $p \geq 1$ these measures define a norm which has found applications in the theory of estimation and smoothing[61].

Minimization of $F_p(\underline{\phi})$ requires computational methods. An important step in any optimization procedure is the choice of a suitable algorithm. Unfortunately, for problems of high dimensionality very little information on the relative performance of the algorithms for unconstrained optimization is available. Because of the lack of published data, consideration needs to be given initially to the efficiency and limitations of minimization algorithms. The remaining sections of this chapter are, therefore, concerned with the selection and brief documenation of four favoured algorithms which are later used in a comparative study.

5.3 Unconstrained Non-linear Programming Methods

Non-linear optimization methods can be classified as:

(i) Methods that use derivatives, known as gradient type methods.

(ii) Methods that do not use derivatives, commonly known as direct search methods.

However, such a classification is not clear cut, because some techniques evaluate the derivatives using difference schemes or minimize in the direction of a gradient by search methods.

The number of variables in the signal design problem considered here is equal to the sequence length N. In most radar applications sequence lengths of N ≃ 30 up to as large as a few hundred may be desirable. Hence, only algorithms that can handle functions of high dimensionality can be employed. This requirement precludes most of the derivative type methods. Consequently, four currently favoured search algorithms (no derivatives required) are considered and briefly described in the following sections of this chapter.

5.3.1 Unconstrained Minimization without using Derivatives

In derivative-free methods the directions of minimization are generally determined solely from successive evaluations of the objective function. As a rule, in solving unconstrained non-linear programming problems, gradient and second derivative methods converge faster than direct search methods. However, the difficulty encountered with these methods is that in problems with a modestly large number of variables it is laborious (large amount of computer time) or often impossible to provide analytical functions for the derivatives. Although, in principle, evaluation of the derivatives by difference schemes can be substituted for the analytical derivatives, the numerical error introduced can impair the use of such substitutions. In any case direct search methods do not require regularity and continuity of the objective function and the existence of derivatives. Because of these difficulties direct search methods have been derived that, although slower to execute for simple problems, in practice may prove more satisfactory from the users viewpoint than derivative methods.

Many of the search type algorithms require a one-dimensional search along a line in N-space as part of their overall strategy. The methods of locating a minimum of a function of one variable will, therefore be the discussion of the next section.

5.3.2 Efficient Unidimensional Searches

The problem to be considered here is that of finding a local minimum of the function $F(\underline{x})$ of n variables $\underline{x} = (x_1, x_2, \ldots, x_n)^T$.

Nearly all search methods are based on the iteration

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \mu^{(k)} \underline{s}^{(k)} \tag{5.38}$$

where $\underline{s}^{(k)}$ is the direction of search. The parameter $\mu^{(k)}$ is chosen so that it minimises $F(\underline{x}^{(k)} + \mu^{(k)} \underline{s}^{(k)})$, i.e. $\partial F/\partial\mu = 0$. The location of the minimum along the line $\mu^{(k)} \underline{s}^{(k)}$ (linear search) is equivalent to minimizing a function of only one variable. Excellent descriptions of the linear search problem are given by Box et al[62], Powell[63] and Wilde[64]. To apply these one-dimensional search techniques one needs to know an initial uncertainty interval, or 'bracket', $\Delta\underline{x}^{(o)}$, which contains the minimum of $F(\underline{x})$. In addition $F(\underline{x})$ must be unimodal in that interval. There are a number of different methods of reducing the initial bracket, $\Delta\underline{x}^{(o)}$, to a final interval, $\Delta\underline{x}^{(M)}$, [62,63]. A few remarks concerning two techniques which are to be used in Powell's search algorithm[68], seem appropriate.

The first search is based on dividing an interval into two segments, F_1 and F_2; a technique known in ancient times as 'golden section'. The ratio of the whole interval to the larger segment is equal to the ratio of the larger segment to the smaller, i.e.

$$\begin{aligned} F_1 + F_2 &= 1 \\ F_2/1 &= F_1/F_2 \end{aligned} \tag{5.39}$$

hence

$$F_1 = F_2^2$$

$$F_1 = (3 - \sqrt{5})/2 \simeq 0.38$$

$$F_2 = (\sqrt{5} - 1)/2 \simeq 0.68$$

The two segments F_1, F_2 are also known as Fibonacci fractions[57].

The initial bracket on the minimum of $F(x)$ is obtained by a series of increasingly larger steps in the independent variable x. The next bracket, for the kth cycle is then computed each time dividing the interval containing the minimum according to Eq. (5.39) until the minimum is located within a specified accuracy.

Other methods of unidimensional minimization locate a point near the minimum by extrapolation and interpolation. The attraction of these techniques is that polynomial interpolation converges rapidly (at least quadratically) in the vicinity of the minimum, while the golden section search is only linearly convergent. In the Coggin-Powell search[57] for example, a quadratic approximation is carried out using the first three points obtained in the direction of search. The minimum of the quadratic function is then determined. This is continued at suitably chosen points until the minimum of $F(x)$ is located.

These types of linear search algorithms are well suited to implementation on a digital computer.

5.4 Direct Search Methods

Conceptually the simplest method is that of alternating directions in which n searches are made along coordinate directions until the minimum is reached. This process, however, turns out to be highly oscillatory and usually fails to converge if there are interactions between the variables. That is, if terms involving products of the variables occur in the objective function. Therefore, this method cannot be recommended unless the interactions are insignificant.

Out of a number of available search techniques, four methods have been chosen which seem suitable for the signal design problem considered here. The first three are pure search methods that do not require derivatives. The fourth technique described is of the hybrid type and although the derivatives are not required analytically, they are assumed to exist and are evaluated using difference schemes.

The chosen algorithms are well documented in the contemporary literature[56,57]. To avoid unnecessary repetition only two of the search methods, which were found to be most efficient (Chapter 6), will be described in some detail.

5.4.1 Pattern Search Method

The pattern search (PAT) method is a logically simple strategy of search that is completely described by Hooke and Jeeves[65]. The algorithm consists of two major phases; an 'exploratory search' around the base point and a 'pattern search' in the desired direction for minimization. The flow diagram of the algorithm is shown in Fig. 5.3 and may be described briefly as follows:

1. A starting point $\underline{x}^{(o)}$ (initial guess) and an initial step size $\Delta\underline{x}^{(o)}$ must be provided.

2. Exploratory search. Various moves from the base point in the direction of the coordinate axes are conducted. To be specific each variable $x_i^{(o)}$ is changed in rotation, one at a time, i.e., $x_i^{(1)} = x_i^{(o)} + \Delta x_i^{(o)}$. If $F(\underline{x})$ is reduced, the new reference value is adopted. If the increment fails to improve $F(\underline{x})$, $x_i^{(o)}$ is changed to $x_i^{(1)} = x_i^{(o)} - \Delta x_i^{(o)}$ and the value of the objective function is compared as before. If $F(\underline{x})$ is not reduced by either increment $\pm\Delta x_i^{(o)}$, $x_i^{(o)}$ is left unchanged and a new variable $x_{i+1}^{(o)}$ is perturbed.

3. Success? If the best value found for $F(\underline{x})$ during the exploratory search is better than its value at the last base point, a new base point is established. Otherwise

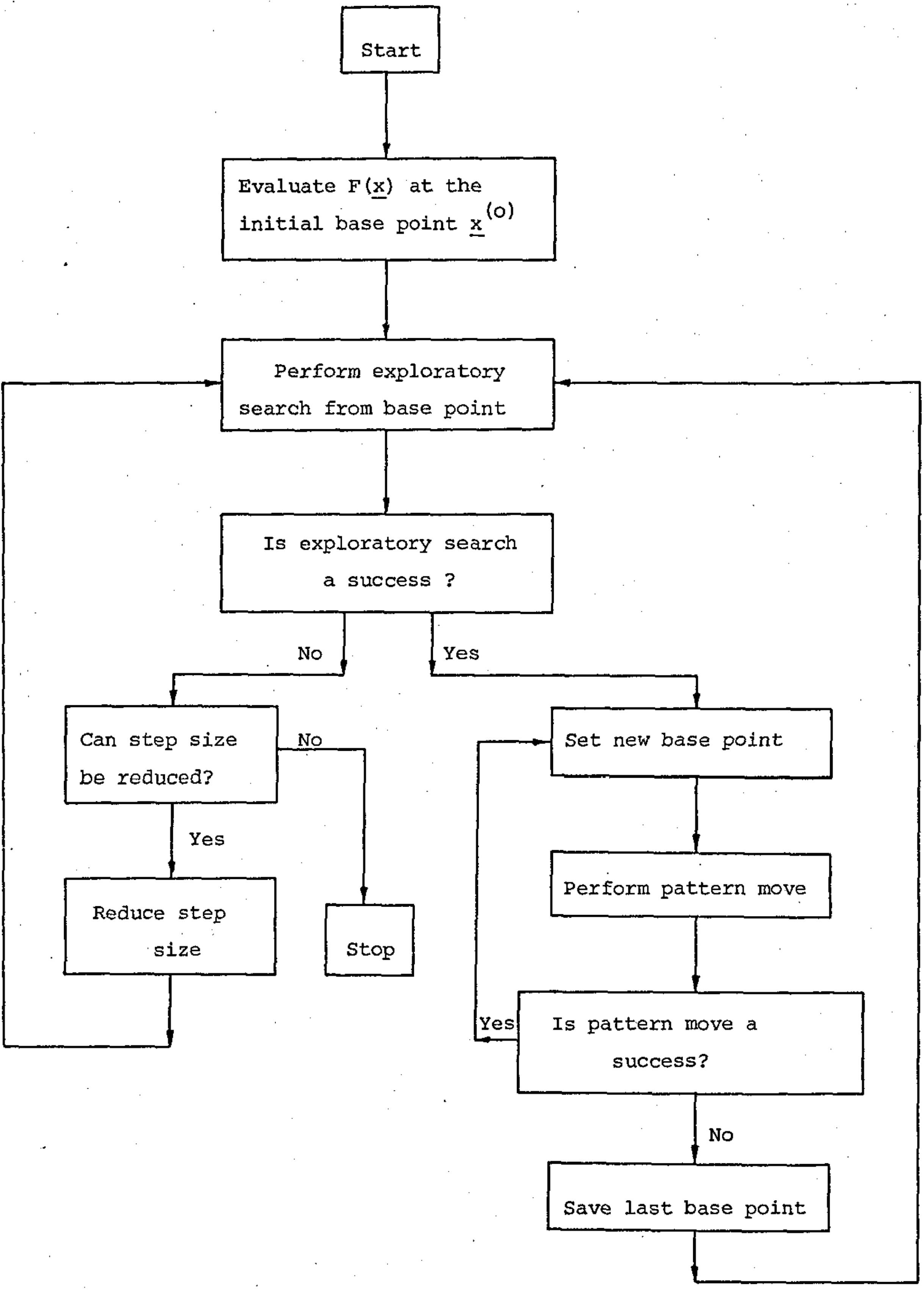

Start
Evaluate $F(\underline{x})$ at the initial base point $\underline{x}^{(o)}$
Perform exploratory search from base point
Is exploratory search a success ?
No
Yes
Can step size be reduced?
No
Yes
Reduce step size
Stop
Set new base point
Perform pattern move
Yes
Is pattern move a success?
No
Save last base point

Fig. 5.3 Flow diagram of pattern search (PAT) algorithm.

the last base point remains.

4. The new base point is stored and a pattern move is conducted in the direction of the difference between the old and the new base point values. This is done in a series of accelerating steps as long as $F(\underline{x})$ is decreased by each pattern move.

5. The pattern move fails to improve $F(\underline{x})$. The last base point is restored and the independent variables are set to the values corresponding to the last base point. The functional value becomes the initial reference for testing the individual moves of the following exploratory search (step 2).

6. Pattern and exploratory search fail to decrease the objective function. If the step sizes for all independent variables are at their minima, the search is completed.

7. The step size is reduced and another exploratory search is performed by restarting at step 2.

It can be seen that the final termination of the search is made when the step size is sufficiently small to ensure that the optimum has been closely approximated. However, the step size must be kept above the practical limits imposed by round off errors.

5.4.2 Flexible Polyhedron Search

Another method often used and well suited for implementation is given by Nelder and Mead[66]. The method is based on the simplex algorithm which is well documented in the literature[57,67].

This method operates by evaluating the objective function $F(\underline{x})$ at points in E^n (E^n is the n-dimensional Euclidean space) located at the vertices of a simplex. Moreover, the techniques described by Nelder and Mead permits the simplex to alter in shape and thus is often referred to as flexible polyhedron rather than simplex method.

The Nelder and Mead (NM) algorithm minimizes a function of n variables using (n+1) vertices of a flexible polyhedron in E^n. Each vertex can be defined as a vector $\underline{x}_i$, $i = 1,2,\dots,n+1$. The vertex which yields the highest value of $F(\underline{x})$ is projected through the centre of gravity (centroid) of the remaining vertices. Improved values of the objective function are obtained by successively replacing the point with the largest value of $F(\underline{x})$ by better points until the minimum of $F(\underline{x})$ is reached.

Nelder and Mead suggested terminating the search when the following condition is satisfied

$$\varepsilon \geqslant \left[\frac{1}{n+1} \sum_{i=1}^{n+1} \{ F(\underline{x}_i^{(k)} - F(\underline{x}_{n+2}^{(k)}))\}^2 \right]^{\frac{1}{2}} \qquad (5.40)$$

where ε is a small arbitrary number and $F(\underline{x}_i^{(k)})$, $F(\underline{x}_{n+2}^{(k)})$ are the values of the objective function at the ith vertex and the centroid respectively on the kth stage of the search.

5.4.3 Powell's Search Method

The algorithm proposed by Powell[68] is one of the currently favoured techniques. The method locates a minimum of a function $F(\underline{x})$ of several variables by successive unidimensional searches from an initial point $\underline{x}_o^{(k)}$. In general Powell's method on the kth stage employs n linear independent search directions. The search is initiated at $\underline{x}_o^{(k)}$ and the transition to a point $\underline{x}_m^{(k)}$ is given by

$$\underline{x}_m^{(k)} = \underline{x}_o^{(k)} + \sum_{i=1}^{m-1} \lambda_i^{(k)} \, \underline{s}_i^{(k)} \qquad (5.41)$$

The initial search directions $\underline{s}_1^{(0)}$, $\underline{s}_2^{(0)}$, ..., $\underline{s}_n^{(0)}$ are taken to be parallel to the coordinate axes of E^n. The parameters $\lambda_i^{(k)}$ are determined sequentially by unidimensional searches such that for the ith parameter $F(\underline{x}_o^{(k)}+\lambda_i^{(k)}\underline{s}_i^{(k)})$ is a minimum. After minimizing $F(\underline{x})$ in each of the n directions a test is usually made to ascertain linear independence of the search directions. The search is terminated if the change in the objective function or in each independent variable is less than the required accuracy ε.

It should be noted that this search technique is also known as the method of conjugate directions.

5.5 Sums of Squares Method

The problem considered here is the solution of a system of non-linear equations

$$f_i(x_1,x_2,\ldots,x_n) = 0, i = 1,2,\ldots,n \tag{5.42}$$

by minimizing a function of the form

$$F(\underline{x}) = 1/2 \sum_{i=1}^{n} \{f_i(\underline{x})\}^2 \tag{5.43}$$

or simply

$$F(\underline{x}) = 1/2\ \underline{f}^T\underline{f} = 1/2\ \|\underline{f}\|^2$$

where $\underline{f}^T$ is the row vector $(f_1(\underline{x}), f_2(\underline{x}),\ldots,f_n(\underline{x}))$.

In the classical Newton iteration the gradient of $F(\underline{x})$ is required

$$\underline{g}(\underline{x}) = \nabla F(\underline{x}) = J(\underline{x})^T\ \underline{f}(\underline{x}) \tag{5.44}$$

where $J(\underline{x})$ is the matrix of the first derivatives of $\underline{f}$ (Jacobian) with

$$J_{kj} = \partial f_k(\underline{x})/\partial x_j$$

If the Jacobian is non-singular at the solution and if $F(\underline{x})$ is twice differentiable, it is easy to show that the Hessian is given by

$$H(\underline{x}) = \nabla^2 F(\underline{x}) = \sum_{i=1}^{n} f_i(\underline{x}) \, \nabla^2 \, f_i(\underline{x}) + J(\underline{x})^T J(\underline{x}) \tag{5.45}$$

where

$$\{\nabla^2 F\}_{ij} = \partial^2 F/\partial x_i \partial x_j$$

It is interesting to note that the computation of the Jacobian $J(\underline{x})$ not only furnishes the gradient ∇F, but also part of $\nabla^2 F$.

At the point where $F(\underline{x})$ is a minimum, denoted by $(\underline{x} + \underline{\delta})$, the conditions

$$\nabla F(\underline{x} + \underline{\delta}) = 0 \tag{5.46}$$

are satisfied. A common strategy is to approximate Eq. (5.46) by the first two terms of the Taylor series in $\underline{\delta}$ about $\underline{x}$

$$\nabla F(\underline{x}) + \nabla^2 F(\underline{x}) \, \underline{\delta} \simeq 0 \tag{5.47}$$

Furthermore, the least squares method works on the assumption that the term

$$\sum_{i=1}^{n} f_i(\underline{x}) \, \nabla^2 \, f_i(\underline{x})$$

can be ignored, thus

$$J(\underline{x})^T J(\underline{x}) \, \underline{\delta} = -\nabla F(\underline{x})$$

and hence

$$\underline{\delta} = -(J^T J)^{-1} J^T \underline{f}(\underline{x}) \tag{5.48}$$

The matrix $J^+ = (J^T J)^{-1} J^T$ is known as the generalized inverse of J and is an extension of the concept of an inverse for matrices which are singular or rectangular[69]. If rank (J) = n is satisfied in Eq. (5.48), the algorithm reduces to the Newton method

$$\underline{\delta} = -J^{-1} \underline{f}(\underline{x}) \tag{5.49}$$

In general the least squares methods are based on the iteration

$$\underline{x}^{(k+1)} = \underline{x}^{(k)} + \mu^{(k)} \underline{\delta}^{(k)}$$

where $\underline{\delta}^{(k)}$ solves a linear system of equations of the form

$$A^{(k)} \underline{\delta}^{(k)} = -\nabla F(\underline{x}^{(k)}) \qquad (5.50)$$

and $A^{(k)}$ is a (nxn) square matrix characteristic of the method. The parameter $\mu^{(k)}$ is usually chosen in such a way as to ensure that $F(\underline{x}^{(k+1)}) < F(\underline{x}^{(k)})$.

Unfortunately the solution of Eq. (5.48) is not as straightforward as it might appear. In many practical cases the Jacobian J is ill-conditioned or even singular (rank (J) < n). An attempt to cover these cases has been made by Levenberg[71] and Marquard[72] by introducing a positive parameter λ into the initial least squares formulation of Eq. (5.48).

5.5.1 Powell's Least Squares Method

Powell's sums of squares technique[68,70] is based on a compromise between the Newton-Raphson algorithm and the method of steepest descent. This algorithm, unlike earlier methods, does not search along a line in the space of the variables $\underline{x}$. Instead of a linear search, the functions $\underline{f}(\underline{x})$ are calculated for only one value of $\underline{x}$ on each iteration.

A major drawback of the Newton method is that the value of the objective function is guaranteed to be improved on each cycle only if the Hessian matrix $H(\underline{x})$ of the objective function is positive definite[57]. (A matrix is positive definite if and only if all the eigenvalues are positive). For strictly convex functions $H(\underline{x})$ is positive definite, but for general functions the Newton method may lead to search directions diverging from the minimum of $F(\underline{x})$. Therefore, to obtain the correction vector $\underline{\delta}$ (Eq. (5.48)), Marquard and Levenberg[71,72] suggested the solution of

$$\sum_{j=1}^{n} \{\lambda I_{ij} + \sum_{k=1}^{n} J_{ki} J_{kj}\} \delta_j = - \sum_{k=1}^{n} J_{ki} f_k(\underline{x}) \qquad (5.51)$$

$$i = 1,2,\ldots,n$$

where I is the unit matrix and λ is a parameter. It is noted that the characteristic matrix A (Eq. (5.50)) in this case is given by

$$A = \lambda I + J^T(\underline{x}) J(\underline{x})$$

For sufficiently large λ it can be shown that the criterion for success

$$F(\underline{x} + \underline{\delta}) < F(\underline{x}) \qquad (5.52)$$

is satisfied. Moreover, if the term λI overwhelms J^TJ, the minimization approaches the steepest descent search.

The Powell least squares algorithm (POS) is initiated by providing an estimate $\underline{x}$ of the minimum of $F(\underline{x})$, and also an approximation to the Jacobian matrix J, which is obtained by calculating

$$J_{kj} = \partial f_k(\underline{x})/\partial x_j \approx \{f_k(x_1,x_2,\ldots,x_j + \varepsilon, x_{j+1},\ldots,x_n)\}/\varepsilon$$

Furthermore, the matrix J^{-1}, a matrix Ω of n directions in the space of the variables and a step length Δ are required.

The calculation of the correction vector $\underline{\delta}$ is carried out by predicting both, the Newton correction $\underline{q}$ and the steepest descent direction $\underline{g}$

$$\underline{q} = -J^{-1}(\underline{x}) \; \underline{f}(\underline{x})$$

$$\underline{g} = -J^{T}(\underline{x}) \; \underline{f}(\underline{x})$$

Because of the good convergence properties of the Newton iteration it would be desirable to let $\underline{\delta} = \underline{q}$, provided the norm $\|\underline{q}\| < \Delta$. The purpose of Δ is to limit the size of $\underline{\delta}$. If $\|\underline{q}\| > \Delta$, $\underline{\delta}$ is set equal to a positive multiple of the gradient $\underline{g}$ such that $\|\underline{\delta}\| = \Delta$. However, if neither of the inequalities is satisfied, $\underline{\delta}$ is assigned a value which is a linear combination of $\underline{g}$ and $\underline{q}$. The value of Δ is reduced if $F(\underline{x} + \underline{\delta}) \geq F(\underline{x})$. However, Δ cannot become smaller than some pre-assigned small quantity, ε, which ensures an adequate Jacobian approximation. The algorithm includes an additional feature to increase Δ if the inequality (5.52) is satisfied. In general, however, Δ is bounded by a minimum and maximum value.

The POS algorithm uses an updating procedure for the matrices J and J^{-1}. The revision formulae are given by

$$J^{(k+1)} = J^{(k)} + (\underline{\kappa} - J^{(k)}\,\underline{\delta})\,\underline{\delta}^T/\|\underline{\delta}\|^2 \qquad (5.53)$$

where

$$\underline{\kappa} = \underline{f}(\underline{x} + \underline{\delta}) - \underline{f}(\underline{x})$$

and

$$(J^{-1})^{(k+1)} = (J^{-1})^{(k)} + (\underline{\delta} - (J^{-1})^{(k)}\,\underline{\kappa})\,\underline{\delta}^T(J^{-1})^{(k)}/(\underline{\delta}^T(J^{-1})^{(k)}\underline{\kappa}) \qquad (5.54)$$

Special steps are taken to avoid the situation where $(\underline{\delta}^T(J^{-1})^{(k)}\underline{\kappa})$ is zero.

The POS algorithm includes a device to maintain sufficient 'linear independence' in the directions $\underline{\delta}$. Powell defines the vector $\underline{\delta}$ to be independent of a set of directions $(\underline{d}_1, \underline{d}_2, \ldots, \underline{d}_n)$ if the angle between $\underline{\delta}$ and some vector in the space spanned by the directions is not less than thirty degrees. The algorithm therefore ensures that for most iterations $(k > 2n)$, the directions $\underline{\delta}^{(k-2n)}, \underline{\delta}^{(k-2n+1)}, \ldots, \underline{\delta}^{(k)}$ span the full space of the variables, where $\underline{\delta}^{(k)}$ is the correction vector on the kth iteration. This is accomplished by a matrix Ω whose columns are the n orthogonal vectors $(\underline{d}_1, \underline{d}_2, \ldots, \underline{d}_n)$ and a vector $\underline{\omega}$ which is used to store the history of the previous iterations.

The calculation of an iteration of the POS algorithm can now be summarized, with respect to Fig. 5.4, in the following major steps:

1. Initialize the iteration by calculating the gradient $\underline{g}$ and the Newton correction $\underline{q}$ to obtain $\underline{\delta}$, where $\|\underline{\delta}\| < \Delta$.

2. If $\underline{\delta}$ is sufficiently independent, the functions $\underline{f}(\underline{x} + \underline{\delta})$ are computed and Δ is revised.

3. If $\underline{\delta}$ is linearly dependent, it is set to some pre-assigned value. The functions $\underline{f}(\underline{x} + \underline{\delta})$ are calculated and after revision of J the iteration is continued from step 1.

4. If $F(\underline{x} + \underline{\delta}) < F(\underline{x})$, $\underline{x}$ is replaced by $(\underline{x} + \underline{\delta})$, otherwise the old value remains.

5. If $\|\underline{\delta}\| < \varepsilon$, the search is continued from step 3. Otherwise the Jacobian is updated and a new iteration is started by going back to step 1.

Powell suggested termination of the search when either of the conditions is satisfied:

(i) $F(\underline{x}) \leqslant \varepsilon_1$, where ε_1 specifies the accuracy of the solution.

(ii) If (n+4) iterations fail to improve the function $F(\underline{x})$.

(iii) If $\|\underline{\delta}\| < \varepsilon$ and further iterations do not decrease $F(\underline{x})$.

(iv) If a stationary value of $F(\underline{x})$ is predicted, i.e. $\nabla F(\underline{x}) \simeq 0$.

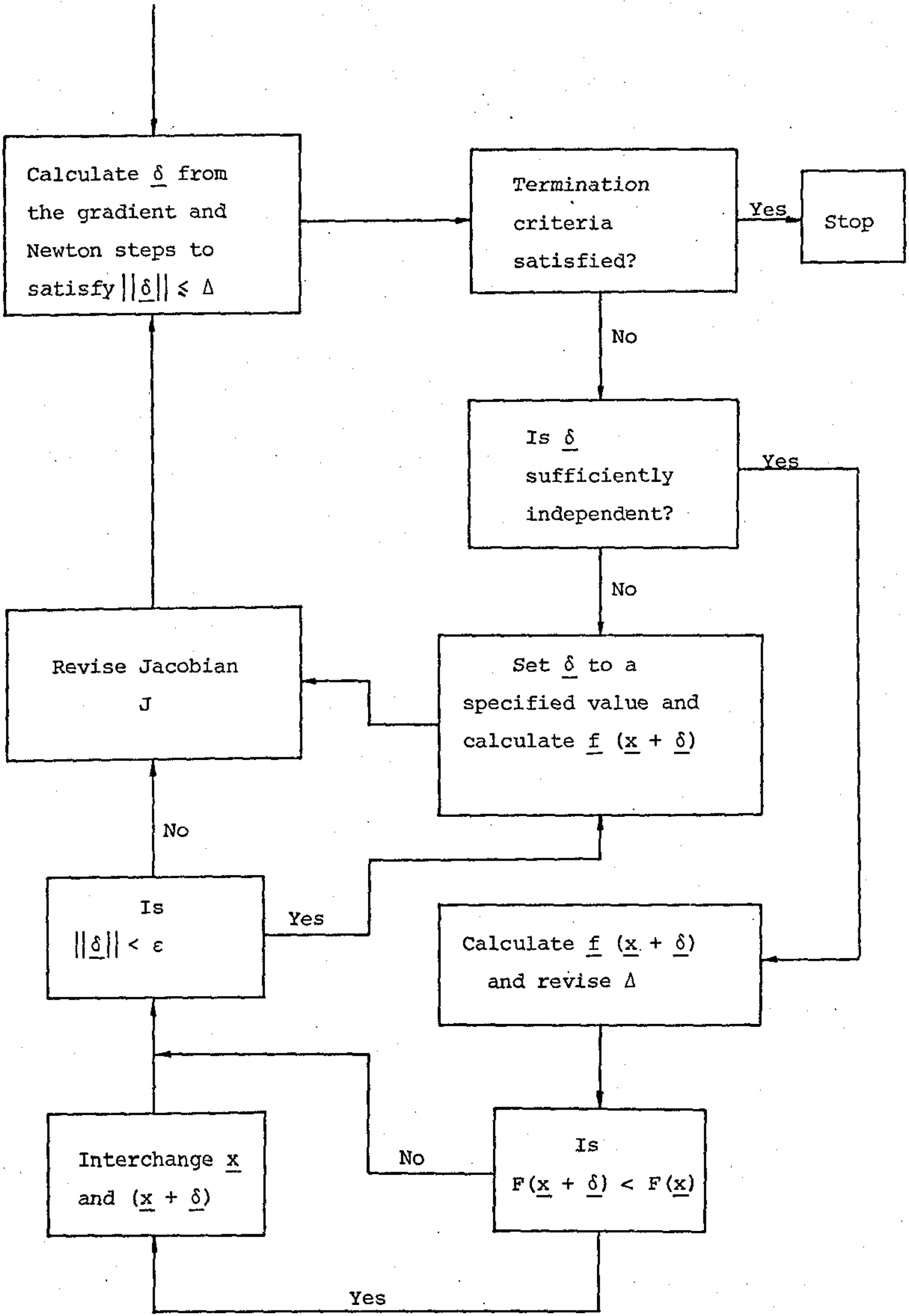

Calculate $\underline{\delta}$ from the gradient and Newton steps to satisfy $||\underline{\delta}|| \leq \Delta$
Termination criteria satisfied?
Yes
Stop
No
Is $\underline{\delta}$ sufficiently independent?
Yes
No
Revise Jacobian J
Set $\underline{\delta}$ to a specified value and calculate $\underline{f}$ $(\underline{x} + \underline{\delta})$
No
Is $||\underline{\delta}|| < \varepsilon$
Yes
Calculate $\underline{f}$ $(\underline{x} + \underline{\delta})$ and revise Δ
Interchange $\underline{x}$ and $(\underline{x} + \underline{\delta})$
No
Is $F(\underline{x} + \underline{\delta}) < F(\underline{x})$
Yes

Fig. 5.4 An iteration of Powell's sum of squares (POS) method.

CHAPTER 6

SYNTHESIS OF UNIFORM SEQUENCES USING NUMERICAL OPTIMIZATION TECHNIQUES

6.1 Introduction

The resolution performance of linear and non-linear FM type sequences (Chapter 4) may be quite satisfactory for a number of different applications. However, a better clutter rejection can be obtained by reducing the sidelobes even further using numerical means. The application of unconstrained minimization algorithms to the sequence design problem raises the question as to which of the currently favoured methods described in the preceding chapter is most suitable. Most of the unconstrained search methods work well for problems with a small number of variables. Unfortunately, very little information about the relative efficiency of various algorithms, when applied to functions of medium to high dimensionality, is available at present. In signal design problems the dimensionality of the objective function is usually large $(N > 30)$. In addition in this particular case trigonometric equations become very ill-conditioned (highly non-linear, multimodal and oscillatory) as the number of variables increases. It is therefore essential to consider first the efficiency of the various minimization methods described previously so as not to seriously limit the feasible number of variables that can be handled and thus to restrict the sequence length.

This chapter also shows that optimization techniques can be useful in the synthesis of binary phase coded signals. This class of waveform has the advantage that simple and efficient decoders can be built. In addition the possibility of designing pairs of phase coded sequences with low ACF sidelobes and small crosscorrelation is investigated. The use of such sequences in time-division multiple-access (TDMA) systems is of wide interest.

6.2 Basis for Comparison of Numerical Algorithms

Before evaluating the relative effectiveness of the various algorithms, some remarks concerning the coding and the criteria to use in the evaluation of the effectiveness are appropriate.

The programs used in the investigation were:

(i) The pattern search method of Hook and Jeeves, hereafter termed PAT.

(ii) Powell's linear search method employing two different unidimensional searches, the golden section and the Coggin-Powell search, hereafter termed POG and POC respectively.

(iii) The simplex method of Nelder and Mead, hereafter termed NM.

(iv) Powell's sum of squares method termed POS.

The programs used were, as far as possible, definite versions of each algorithm described in Chapter 5. The searches POC, POG and NM were coded as given by Himmelblau[57]. Algorithm POS uses the program published by Powell[70], and PAT was written from the flow diagram, Fig. 5.3. All algorithms were coded in Fortran IV and run on the ICL 1904A computer system at L.U.T. The experimental evaluation of the algorithms has been done by minimizing $F_4(\underline{\phi})$ (Eq. 5.34)) of varying dimensionality and for different termination criteria.

The performance of each algorithm depends in practice on the values of certain preset parameters such as the initial step size, starting point, accuracy required as well as the precise coding. Computing time and location of the minimum in particular are very sensitive to these factors and to the termination criteria.

The criteria considered in the assessment of the unconstrained algorithms are:

(i) Number of function evaluations

(ii) Computation time to termination (for a specified accuracy)

In order to compare the performance of the various methods, under conditions as nearly identical as possible, they were used to solve a set of signal design problems, starting each problem with the same initial conditions. However, since the course of computation varied with each method, it was impossible to terminate the calculations at the same point; the accuracy of the final solution varied from method to method. The degree of precision usually depended on the termination criterion. For POG, POC and PAT the termination criterion ε was chosen to be a fractional change in $F_4(\underline{\phi})$:

$$\varepsilon \leqslant \left| \frac{F_4(\underline{\phi}^{(k+1)}) - F_4(\underline{\phi}^{(k)})}{F_4(\underline{\phi}^{(k)})} \right| \qquad (6.1)$$

For the NM algorithm the iteration was terminated when

$$\varepsilon \leqslant \left\{ \frac{1}{N+1} \sum_{i=1}^{N+1} \left[F_4(\underline{\phi}_i^{(k)}) - F_4(\underline{\phi}_{N+2}^{(k)}) \right]^2 \right\}^{\frac{1}{2}} \qquad (6.2)$$

whereas the POS algorithm was terminated when either the norm of the correction vector $\|\underline{\delta}\|$ was smaller than a specified small number ε and/or a stationary value of $F_4(\underline{\phi})$ was predicted, i.e.

$$\|\underline{\delta}\| < \varepsilon$$

or

$$\nabla F_4(\underline{\phi}) \simeq 0 \qquad (6.3)$$

The number of function evaluations of $F_4(\underline{\phi})$ to reach a certain precision is in itself a not too satisfactory measure of efficiency for algorithms having widely differing strategies. Consequently, a second criterion, the computation time to execute an algorithm, is alternatively cited as a measure of the effectiveness of a method. Although the computation time is not necessarily the best criterion, because the measured time depends on the type of computer and the method of coding of the algorithm, it often has to serve, in lieu of a better measure.

6.2.1 Evaluation of Non-linear Programming Methods

The performance of the non-linear unconstrained optimization algorithms is summarized in Table 6.1 to 6.5 where the following notations have been adopted:

N = number of variables (sequence length)

M = total number of function evaluations

ITR = number of iterations

ε = termination criterion

F_1 = initial value of $F_4(\underline{\phi})$

F_2 = final value of $F_4(\underline{\phi})$

S_{max} = peak sidelobe level of ACF, $\max_k |r(k)|$

$E_s(\%)$ = sidelobe energy ratio in %; $E_s(\%) = 10^2 \; (\sum_{k=1}^{N-1} |r(k)|^2)/E^2$

R_c = mean convergence rate, $1/M \; (F_1 - F_2)$

T(mil) = execution time in millunits (1 millunit ≈ 0.5 seconds)

For the POS algorithm the number of function evaluations to set up the Jacobian matrix J were disregarded. All algorithms were initiated at the point

$$\phi_i = \pi i^2/N \qquad i = 0,1,2,\ldots,(N-1)$$

and the initial step size, except for POS, was chosen to be 0.1. The selection of the parameters ε and Δ for POS is quite critical in view of the effectiveness of the method. In particular ε must be so small that for $\|\underline{\delta}\| < \varepsilon$, $f_k(\underline{\phi} + \underline{\delta})$, $k = 1,2,\ldots,N$ is nearly a linear function of $\underline{\delta}$. The parameter Δ is usually set to a generous estimate of the distance of the solution from the initial guess. Using the Euclidean metric the distance between $\underline{\phi}$ and $\hat{\underline{\phi}}$ is

N	ε	M	ITR	F_1	F_2	S_{max}	E_s(%)	R_c	T(mil)
5	0.1	35	1	2.292	1.866	1.001	9.194	0.007	2
10	0.1	206	3	18.751	4.862	1.054	6.060	0.055	23
15	0.1	292	3	42.632	9.158	1.174	4.197	0.093	64
15	0.05	531	6	-	7.289	1.099	3.919	0.059	104
15	0.01	598	7	-	7.062	1.090	3.898	0.054	115
15	0.005	658	8	-	6.994	1.083	3.886	0.050	124
15	0.001	894	12	-	6.906	1.065	3.842	0.038	163
20	0.1	133	1	80.855	52.049	1.842	6.151	0.139	71
25	0.1	616	4	141.409	19.281	1.383	2.892	0.167	319
30	0.1	362	2	216.681	121.378	2.269	4.815	0.195	307
30	0.05	362	2	-	121.378	2.269	4.815	0.195	307
30	0.01	1616	11	-	88.945	1.906	4.998	0.074	1046
30	0.005	3471	25	-	51.847	1.590	3.727	0.046	2146
40	0.1	264	1	435.491	251.918	2.523	4.299	0.430	498
50	0.1	2007	7	752.341	57.130	1.336	1.821	0.314	3489

Table 6.1 Performance of POC algorithm.

N	ε	M	ITR	F_1	F_2	S_{max}	E_s (%)	R_c	T(mil)
5	0.1	49	1	2.292	1.924	1.000	9.407	0.005	3
10	0.1	253	3	18.751	4.802	1.049	5.861	0.048	26
15	0.1	426	4	42.632	10.192	1.172	4.316	0.067	88
15	0.05	495	5	-	9.221	1.153	4.288	0.062	97
15	0.01	996	13	-	7.048	1.041	3.898	0.034	184
15	0.005	1039	14	-	6.996	1.042	3.873	0.033	194
15	0.001	1326	19	-	6.918	1.057	3.817	0.026	245
20	0.1	160	1	80.855	51.395	1.911	6.024	0.133	74
25	0.1	711	4	141.409	22.895	1.491	2.865	0.148	355
30	0.1	237	1	216.681	138.557	2.301	4.847	0.245	223
30	0.05	366	2	-	126.968	2.274	4.842	0.199	292
30	0.01	3300	27	-	55.499	1.637	3.923	0.048	2024
30	0.005	3590	30	-	54.250	1.643	3.858	0.044	2243
40	0.1	330	1	435.491	249.315	2.577	4.254	0.404	540
50	0.1	1943	7	752.341	97.162	1.784	2.123	0.313	3341

Table 6.2 Performance of POG algorithm.

N	ε	M	ITR	F_1	F_2	S_{max}	E_s(%)	R_c	T(mil)
5	0.1	11	1	2.292	2.171	1.000	11.005	0.004	1
10	0.1	258	72	18.751	5.102	1.030	6.104	0.053	25
15	0.1	490	147	42.632	9.017	1.250	5.541	0.068	93
15	0.05	598	180	-	8.418	1.219	4.167	0.057	113
15	0.01	701	209	-	8.319	1.163	4.172	0.049	133
15	0.005	723	215	-	8.312	1.154	4.191	0.047	136
15	0.001	2839	857	-	7.222	1.069	3.930	0.012	533
20	0.1	736	222	80.855	46.765	1.765	6.263	0.046	230
25	0.1	1206	370	141.409	23.197	1.543	2.953	0.098	559
30	0.1	965	292	216.681	125.488	2.238	4.995	0.094	623
30	0.05	1594	488	-	120.252	2.281	4.872	0.060	1027
30	0.01	3143	973	-	116.329	2.315	5.055	0.032	2020
30	0.005	3295	1018	-	116.273	2.314	5.055	0.030	2122
40	0.1	2176	676	435.491	202.082	2.522	4.011	0.107	2393

Table 6.3 Performance of NM algorithm.

N	M	F_1	F_2	S_{max}	E_s(%)	R_c	T(mil)
5	36	2.292	1.795	1.000	9.331	0.014	2
10	126	18.751	4.809	1.050	6.045	0.111	18
15	115	42.632	8.537	1.240	4.120	0.296	34
20	80	80.855	52.526	2.000	5.947	0.354	37
25	269	141.409	18.444	1.480	2.537	0.457	205
30	92	216.681	134.628	2.380	5.093	0.892	94
40	281	435.491	209.542	2.509	3.961	0.804	484
50	729	752.342	73.704	1.644	2.076	0.931	2017
60	671	1162.701	294.119	2.384	2.758	1.294	2560

Table 6.4 Performance of POS algorithm.

(ε = 0.001, Δ = 3.0)

N	ε	M	ITR	F_1	F_2	S_{max}	E_s(%)	R_c	T(mil)
5	0.1	53	5	2.292	1.793	1.00	9.186	0.009	1
5	0.01	74	7	-	1.793	1.000	9.165	0.007	2
10	0.1	134	7	18.751	4.677	1.058	5.867	0.105	6
10	0.01	200	10	-	4.664	1.043	5.856	0.070	8
15	0.1	197	7	42.632	8.522	1.195	4.198	0.173	12
15	0.05	286	10	-	7.579	1.177	4.055	0.123	16
15	0.01	503	17	-	7.016	1.086	3.939	0.071	28
15	0.005	592	20	-	6.935	1.067	3.880	0.060	31
15	0.001	754	25	-	6.909	1.066	3.861	0.047	39
20	0.1	179	5	80.855	46.062	1.833	6.327	0.194	15
20	0.01	585	15	-	41.517	1.687	6.301	0.070	40
25	0.1	358	8	141.409	17.648	1.434	2.730	0.346	34
30	0.1	316	6	216.681	115.619	2.288	4.778	0.320	36
30	0.05	378	7	-	112.359	2.240	4.827	0.276	43
30	0.01	1508	27	-	79.614	1.819	4.740	0.091	159
30	0.005	2846	50	-	50.300	1.530	3.785	0.058	295
30	0.001	3136	55	-	50.015	1.503	3.791	0.053	326
40	0.1	420	6	435.491	188.575	2.421	3.941	0.588	64
40	0.01	2782	37	-	82.997	1.792	3.092	0.127	369
50	0.1	1072	12	752.341	55.366	1.538	1.795	0.650	190
50	0.01	2573	27	-	47.571	1.479	1.698	0.274	421
60	0.1	1059	10	1162.701	244.701	2.068	2.697	0.867	228
60	0.01	4671	41	-	93.575	1.962	1.855	0.229	908
70	0.1	1365	11	1701.352	351.538	2.508	2.616	0.989	340
70	0.01	5358	40	-	142.404	1.782	1.809	0.291	1204
80	0.1	1700	12	2363.900	274.297	2.138	1.910	1.229	482
80	0.01	4111	27	-	231.327	1.984	1.825	0.519	1063
90	0.1	1249	8	3163.479	972.081	3.320	2.697	1.755	408
90	0.01	7793	45	-	199.688	1.722	1.446	0.382	2253
100	0.1	2506	14	4088.136	333.660	2.158	1.509	1.498	874

Table 6.5 Performance of PAT algorithm.

$$d(\underline{\phi},\hat{\underline{\phi}}) = \{\sum_{i=o}^{N-1} (\phi_i - \hat{\phi}_i)^2\}^{\frac{1}{2}} \qquad (6.4)$$

Preliminary trials have shown that $\varepsilon = 0.001$ and $\Delta = 3.0$ give consistently good results for a variety of problems with different dimensionality.

The investigations have shown that PAT was the best, and the other methods, exclusive of the POS and NM algorithms, were of roughly the same effectiveness. Quite obviously accurate minimization along each search direction for the univariate search type algorithms can be expensive in function evaluations. The results in Table 6.1 and Table 6.2 show that a high accuracy is not required in most cases, and thus computing time can be kept low. A value of the termination criterion $\varepsilon = 0.01$ to 0.005 usually suffices. For example with values of $\varepsilon = 0.01$ and $N = 15$ the minimum obtained with POC is 7.062 while for $\varepsilon = 0.001$ it is 6.906. The effect of such a small decrease in the objective function on the sidelobe levels and sidelobe energy is negligible. To obtain a significant improvement in sidelobe performance the reduction of $F_4(\underline{\phi})$ must be at least of the order of 20% to 30% (see for example $N = 30$). However, if the accuracy is too low failure to obtain function decreases can occur.

The golden section univariate search is clearly less reliable in that the weak search along a line until a minimum is bracketed frequently fails to terminate in a reasonable number of iterations. Fig. 6.1 shows the number of function evaluations required to optimize $F_4(\underline{\phi})$ for $N = 15$ and various degrees of precision. If the evaluation of the objective function takes any significant amount of time, then the PAT search is preferable.

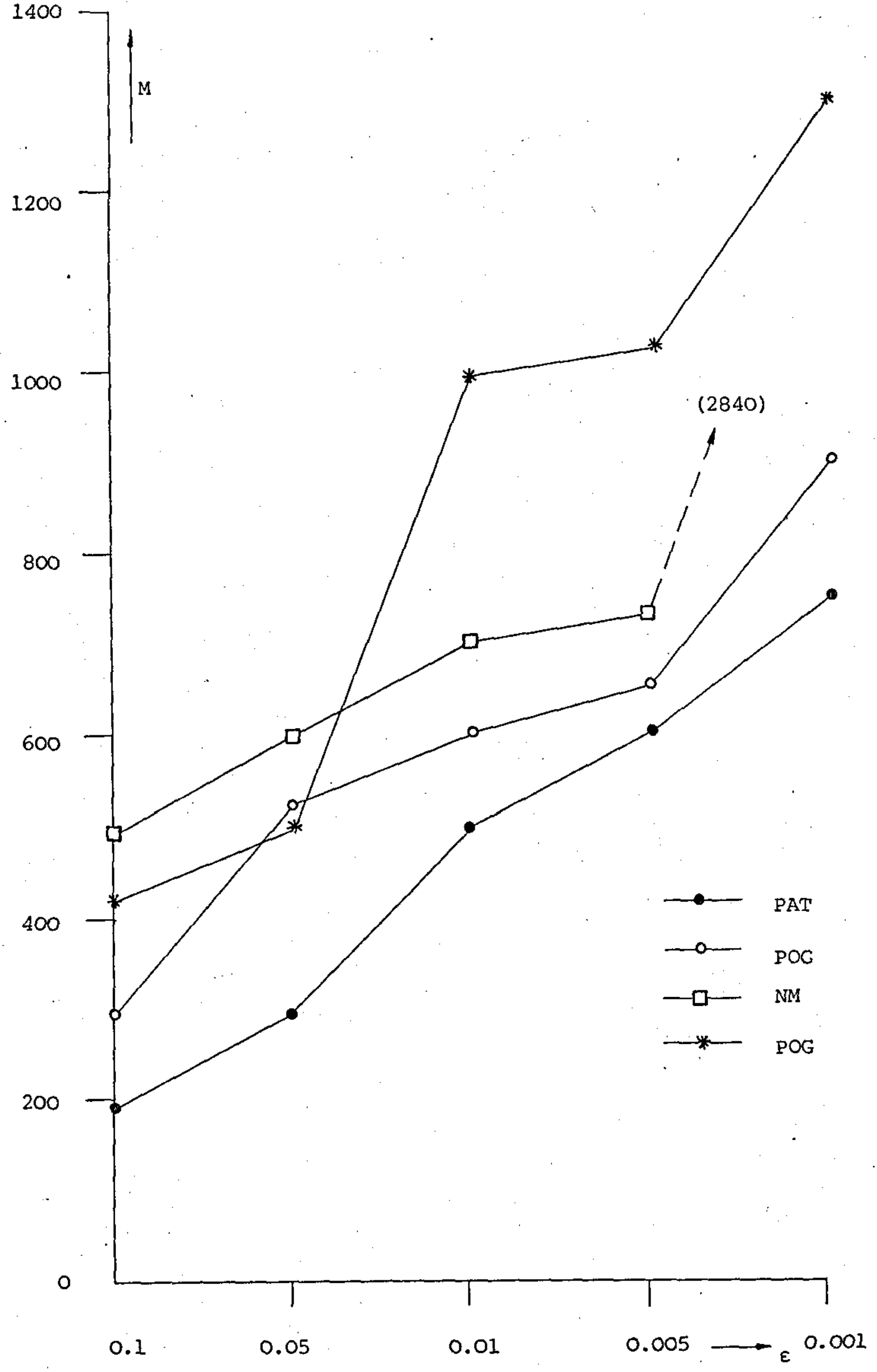


Fig. 6.1 Number of function evaluations to minimise $F_4(\underline{\phi})$ for N = 15.

Attempts to compare quite different algorithms on the basis of the number of functional evaluations can be less satisfactory and is often misleading. Consequently, times to execute a particular problem rather than the number of function evaluations has been selected as being the single most suitable measure of effectiveness. The execution times, measured in millunits (1 millunit ≈ 0.5 seconds), were determined on an ICL 1904A digital computer and a subset of the results are shown in Fig. 6.2. It can be seen that the linear search algorithms POG, POC and the NM method perform similarly and that PAT is clearly superior.

It is important to inquire into how these techniques behave as the dimensionality of the objective function increases. To keep the computation time within reasonable limits the tests were carried out in most cases with $\varepsilon = 0.1$ and are shown in Fig. 6.3 and Fig. 6.4. The unidimensional searches POG, POC and NM were not tested for $N > 40$ because of excessive computation time. As various test problems were solved it became increasingly apparent that the NM method was the most inefficient, even for problems of moderate dimensionality and accuracy. Fig. 6.3 shows that all methods tested perform well for problems with a small number of variables ($N \leqslant 15$). However, the linear search, and in particular the NM algorithm, clearly become less competitive as N increases. This is characteristic for search methods which locate a minimum by conducting N linear searches along a line which requires a large number of function evaluations. Tables 6.1 to 6.5 demonstrate that although the number of function evaluations for POG, POC and PAT are of the same order, the PAT method is superior in execution time. This is basically due to the fact that for an exploratory search in the direction of the coordinate axes the ACF need not be recalculated but can be updated for each individual variable change $\phi_i^{(1)} = \phi_i^{(0)} + \Delta\phi_i^{(0)}$ according to

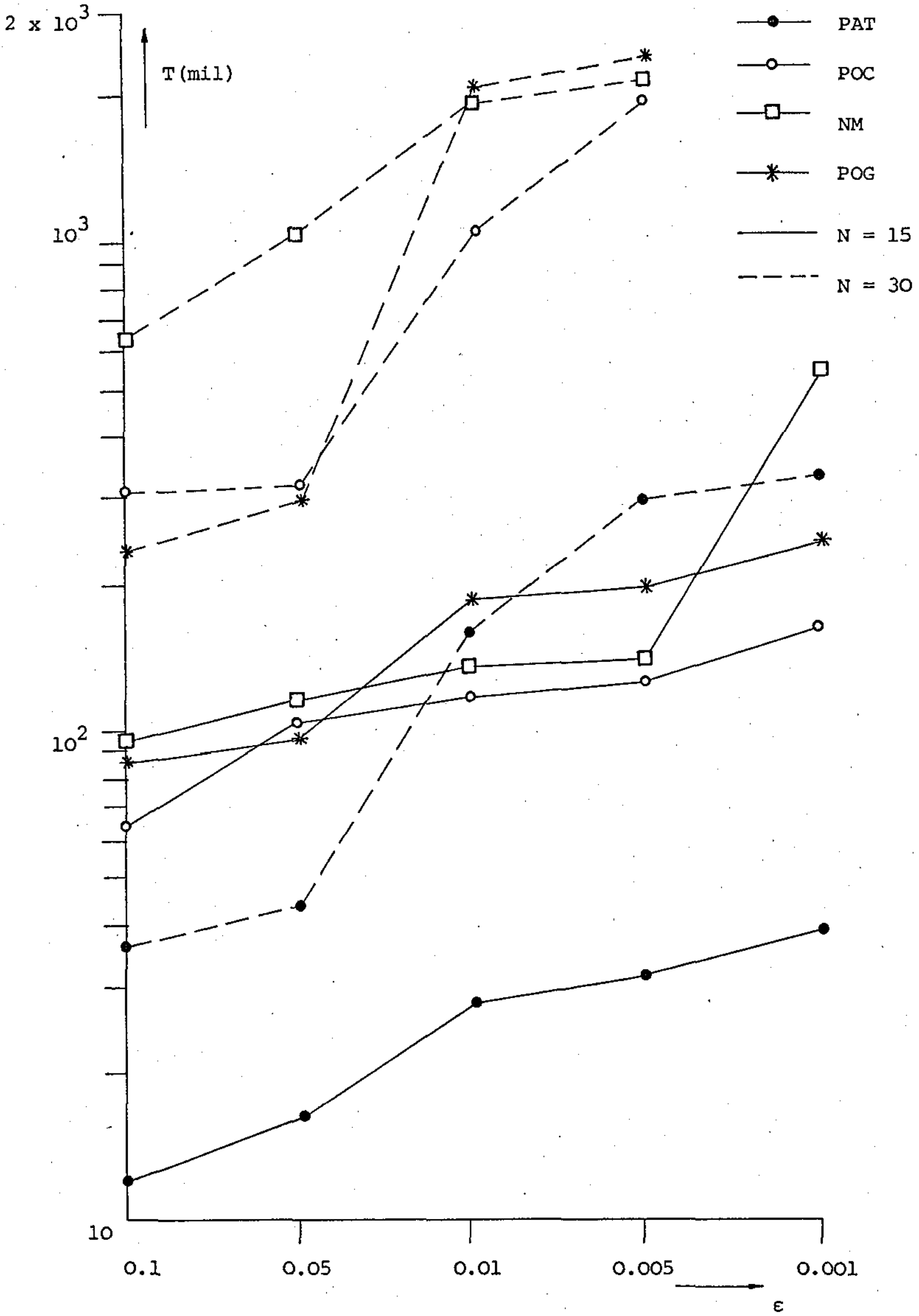


Fig. 6.2 Execution time to minimize $F_4(\underline{\phi})$
(1 milliunit ≃ 0.5 seconds).

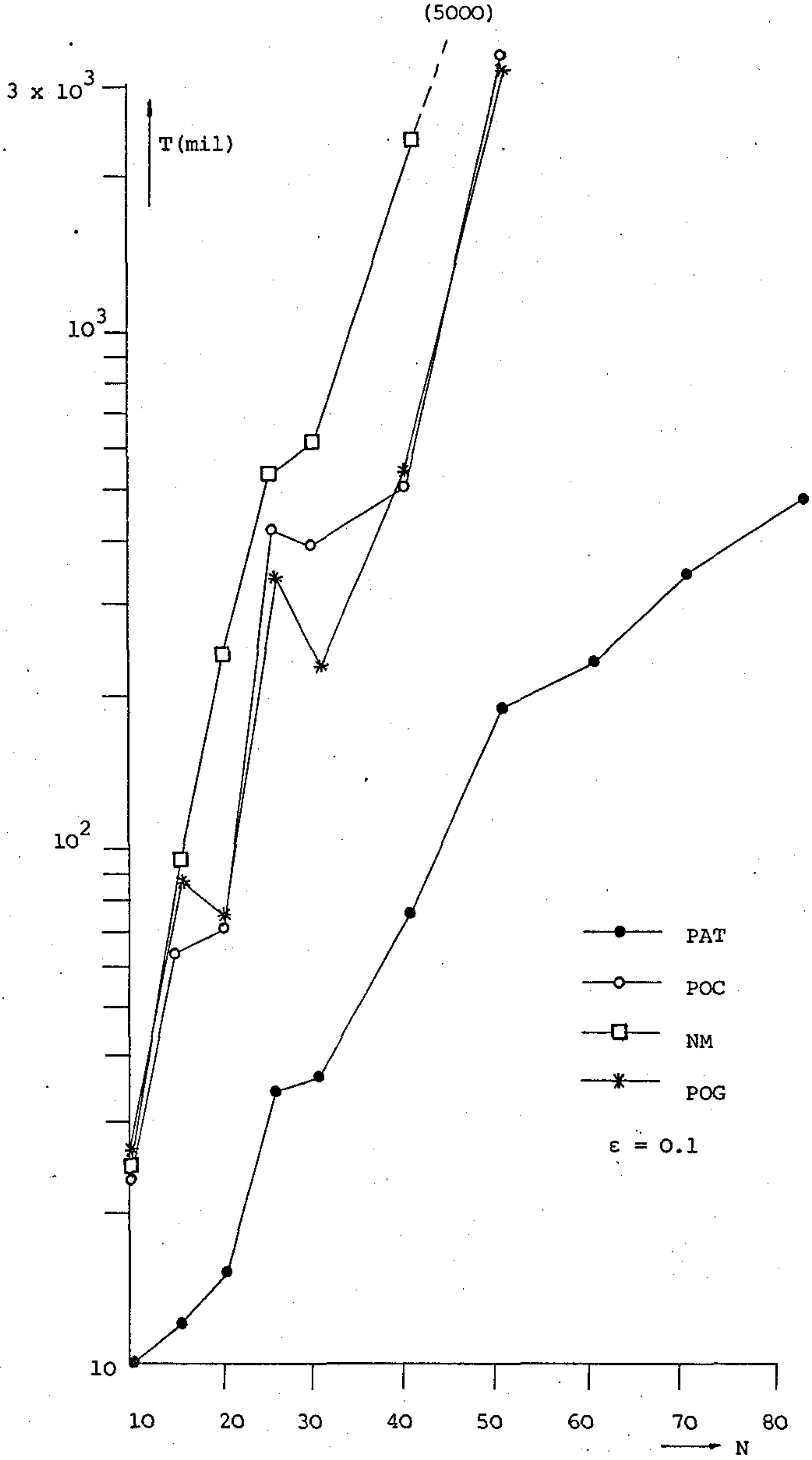


Fig. 6.3 Influence of dimensionality on the minimization algorithms.

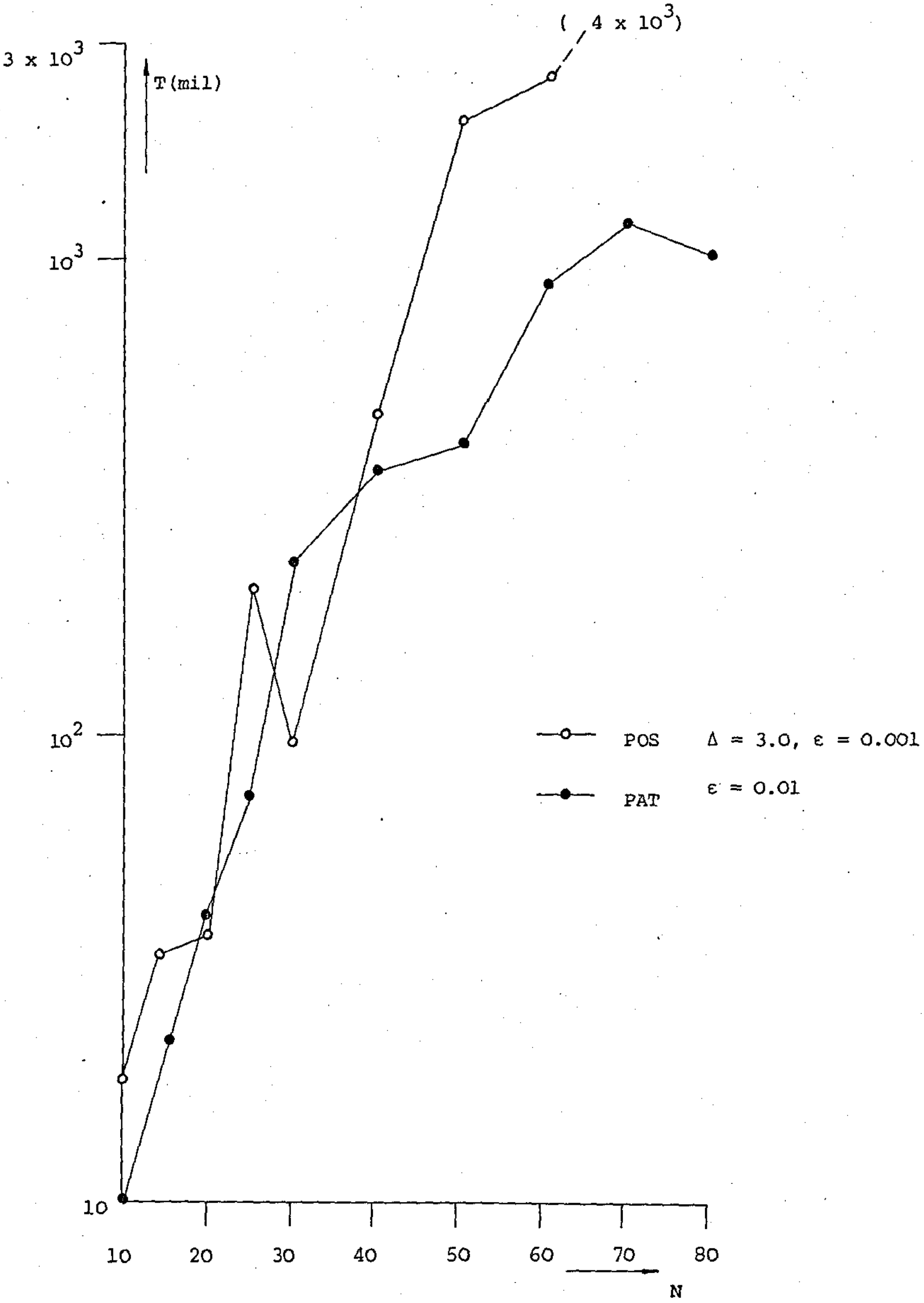


Fig. 6.4 Influence of dimensionality on the minimization algorithms.

$$\Delta r(k) = \begin{cases} a_o(i-k)\left[a_1^*(i) - a_o^*(i)\right] + \left[a_1(i) - a_o(i)\right] a_o^*(i+k) ; & i-k \geqslant 0, i+k \leqslant N \\ \left[a_1(i) - a_o(i)\right] a_o^*(i+k) & ; \ i-k < 0, i+k \leqslant N \\ a_o(i-k)\left[a_1^*(i) - a_o^*(i)\right] & ; \ i-k \geqslant 0, i+k > N \end{cases} \qquad \text{(6.5}$$

where $a_o(i) = \exp(j\ \phi_i^{(0)})$ and $a_1(i) = \exp(j\ \phi_i^{(1)})$

Unfortunately, a similar time-saving rule cannot be found for algorithms that search along a line which is not necessarily parallel to the coordinate axes. Although it is possible to use FFT methods to compute the ACF or even to optimize in the frequency domain by minimizing an objective function of the form

$$F_p(\underline{\phi}) = \sum_{n=0}^{N-1} \left| |S(n,\underline{\phi})|^2 - N \right|^p \qquad (6.6)$$

where $|S(n,\underline{\phi})|^2$ are the power spectrum samples of the sequence s(n), it was found that such an approach does not offer an increase in computational speed for sequences of length $N < 200$. Moreover, since the objective is to design sequences having small ACF sidelobes, it is preferable to work in the time domain rather than the frequency domain.

The Figures 6.3 and 6.4 can provide only qualitative evidence concerning the effectiveness of the respective algorithms when applied to higher dimensional problems. However, it is evident that PAT is distinctly superior to all other methods considered. The POS algorithm performs reasonably well if the initial parameters are chosen carefully (occasional premature termination occurred; see for example Fig. 6.5). In no case, however, was it superior to the PAT method. In addition computer storage can be a serious limitation on the feasible dimensionality of the objective function for POS. In comparing the two variants of linear search POG and POC, it is seen that they behave almost identically. The NM method on the other hand is an unsuitable strategy for this type of application.

Examination of the results presented so far can give only a fragmentary picture of the relative effectiveness of the algorithms, because of the different search strategies and termination criteria. To reduce the mass of data that can be produced by solving test problems, the objective function $F_4(\underline{\phi})$ versus the number of function evaluations is plotted in Fig. 6.5. Such figures provide an empirical measure of the rate of convergence of the respective algorithms. A figure of merit representing the mean convergence rate R_c was chosen to be

$$R_c = 1/M \ (F_1 - F_2)$$

where M is the total number of function evaluations and F_1, F_2 the initial and final values of $F_4(\underline{\phi})$.

The algorithms show a rapid reduction of the objective function in the first few iterations followed by a relatively slow convergence to the minimum. This is not surprising in view of the multimodal and ill-conditioned character of transcendental equations (a different search direction has to be employed at almost every iteration).

The results of this study should not be used as a decisive comparison of the various unconstrained minimization methods. However, the following general conclusions seem to be appropriate in view of the results obtained. The PAT algorithm presents the most consistent and efficient behaviour among the group of techniques investigated. If a reasonably good estimate of the initial parameters is available POS is a good second best in effectiveness. The performance of the algorithm may thus be classified in qualitative terms as

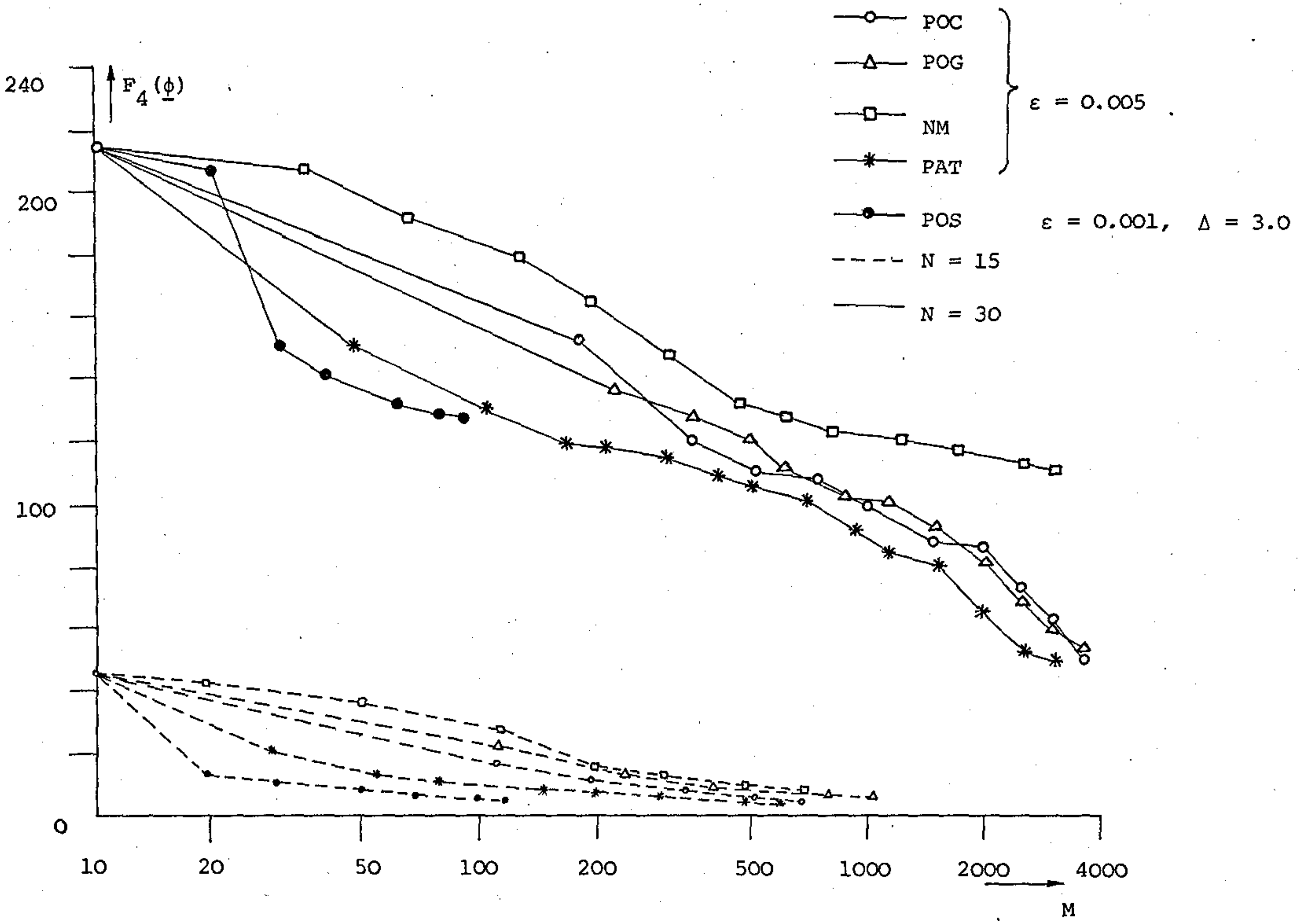


Fig. 6.5 Comparison of the unconstrained optimization algorithms.

(i)	PAT	superior
(ii)	POS	good
(iii)	POG,POC	fair
(iv)	NM	not suitable

Consequently, the PAT method has been selected for the subsequent sequence synthesis problems as the most suitable algorithm.

6.3 Uniform Complex Pulse Trains

In view of the discussion in the previous section the PAT algorithm was used to synthesize uniform (constant amplitude) complex codes[73]. Table 6.6 shows the properties of the best codes of various lengths so far obtained by the use of numerical minimization. The phases of the elements of the initial codes were chosen randomly, except for the codes of length 9, 16, 25, 36, 64, 81, and 100, where the corresponding Frank code was used as the initial code. A value of 4 was used for p, except in the cases of codes of length 4 to 11, where a value of 2 was used. For comparison Table 6.6 also shows the properties of other known codes.

It is of interest to find that uniform complex codes with largest sidelobe unity exist up to a length of at least 18*. Even for much larger lengths, codes exist with peak sidelobe values less than 2. For example the peak sidelobe of the sequence of length N = 100 is only 1.44 or -37 dB. (A subset of these codes is given in Appendix E). It is also of some interest to note that numerical optimization yields a significant improvement in both peak sidelobe and sidelobe energy performance on the Frank codes of the lengths given here. It is also possible to improve on the performance of the Barker codes. The energy

*A code of length 19 was found later using the method described in Chapter 7.

Table 6.6 Autocorrelation sidelobes of codes obtained by numerical optimization where; (B) Barker, (G) Golomb and Scholtz, (S) Scholtz, (C) Carley, (D) Develet and (F) Frank

Code length	Peak sidelobe		Energy ratio	
N	$\max_k \lvert r(k) \rvert$		E_s(%)	
	prior	new	prior	new
4	(B) 1.00	1.00	12.50	9.38
5	(B) 1.00	1.00	8.00	8.01
6	(G) 1.00	1.00	13.89	13.89
7	(B) 1.00	1.00	6.13	3.32
8	(G) 1.00	1.00	7.81	4.32
9	(G) 1.00	1.00	2.41	1.32
10	(G) 1.00	1.00	8.64	4.70
11	(B) 1.00	1.00	4.14	2.69
12	(G) 1.00	1.00	6.25	4.71
13	(B) 1.00	1.00	3.55	2.87
14	(S) 1.00	1.00	5.61	3.76
15	(C) 1.00	1.00	3.11	2.38
16	(D) 1.00	1.00	4.73	3.17
17		1.00		3.32
18		1.00		3.16
19		1.08		3.34
20		1.14		3.12
21		1.28		3.12
23		1.20		2.61
25	(F) 1.62	1.17	4.42	2.22
27	(F) 1.62	1.39		3.49
29		1.50		3.80
31		1.31		3.27
33		1.69		4.32
35		1.66		3.82
36	(F) 2.00	1.46	3.86	2.66
37		1.58		3.46
39		1.84		4.33
41		1.93		4.63
43		1.83		3.28
49	(F) 2.25	1.49	3.10	1.62
64	(F) 2.61	1.67	2.80	1.43
81	(F) 2.88	1.89	2.37	1.59
100	(F) 3.24	1.86	2.20	1.44

ratio for longer sequences ranges from 1.5% to 4.5% which gives r.m.s. sidelobes of the order of $0.12\sqrt{N}$ to $0.21\sqrt{N}$ *.

A typical ACF and zero pattern of a uniform complex code when the initial phase is chosen in a quadratic fashion is depicted in Fig. 6.6. The uniformity of the residues clearly shows the ability of the ℓ_4-measure to reduce single large spikes. If the index p in the ℓ_p-norm is increased one departs further and further away from the mean-square approximation and approaches the minimax or Chebyshev solution for sufficiently large p ($p \simeq 10$). This is illustrated in Fig. 6.7. Hence a value $p = 4$ seems to be a reasonable compromise between self-clutter rejection and peak sidelobe suppression.

In all cases no convergence problems were experienced. However, due to the multimodal character of the objective function the located extremum is most likely a local minimum. There is no method to sidestep this problem other than to begin the search at several different random starting points and to select the best minimum.

6.3.1 Effects of Phase Quantization on the ACF

In general the phase elements of uniform complex codes may have a continuum of values. From practical considerations, however, one could subdivide the phase interval 2π into M increments and select the multiple of $2\pi/M$ which is closest to the actual value of the phase step. The resulting phase error or 'quantization noise' will, of course, increase the overall sidelobe level of the ACF.

* The r.m.s. sidelobe is calculated as; r.m.s. sidelobe = $\{E_s N^2/(N-1)\}^{\frac{1}{2}}$

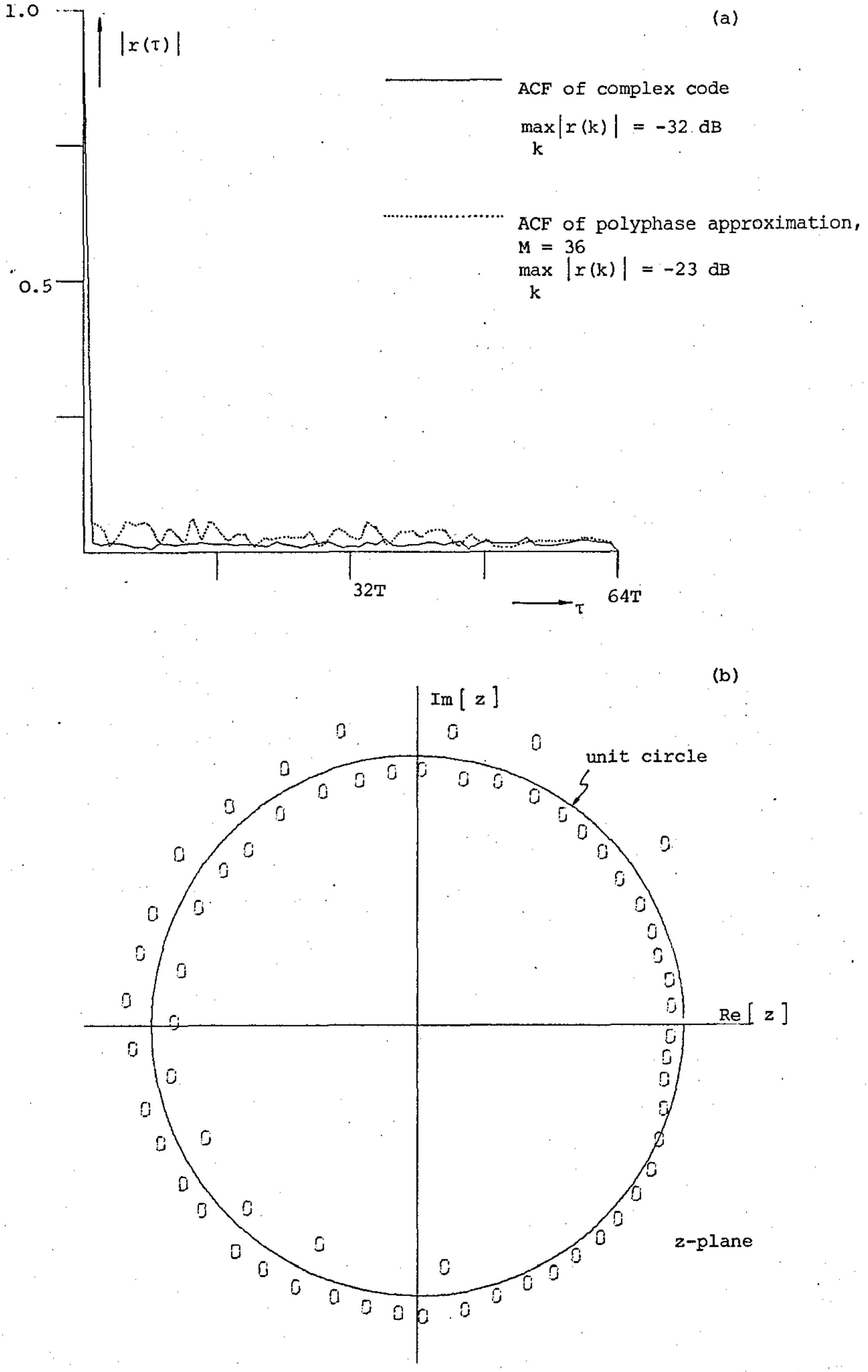


Fig. 6.6 (a) ACF of 64-element complex code and its polyphase approximation
(b) Zero pattern of 64-element complex code.

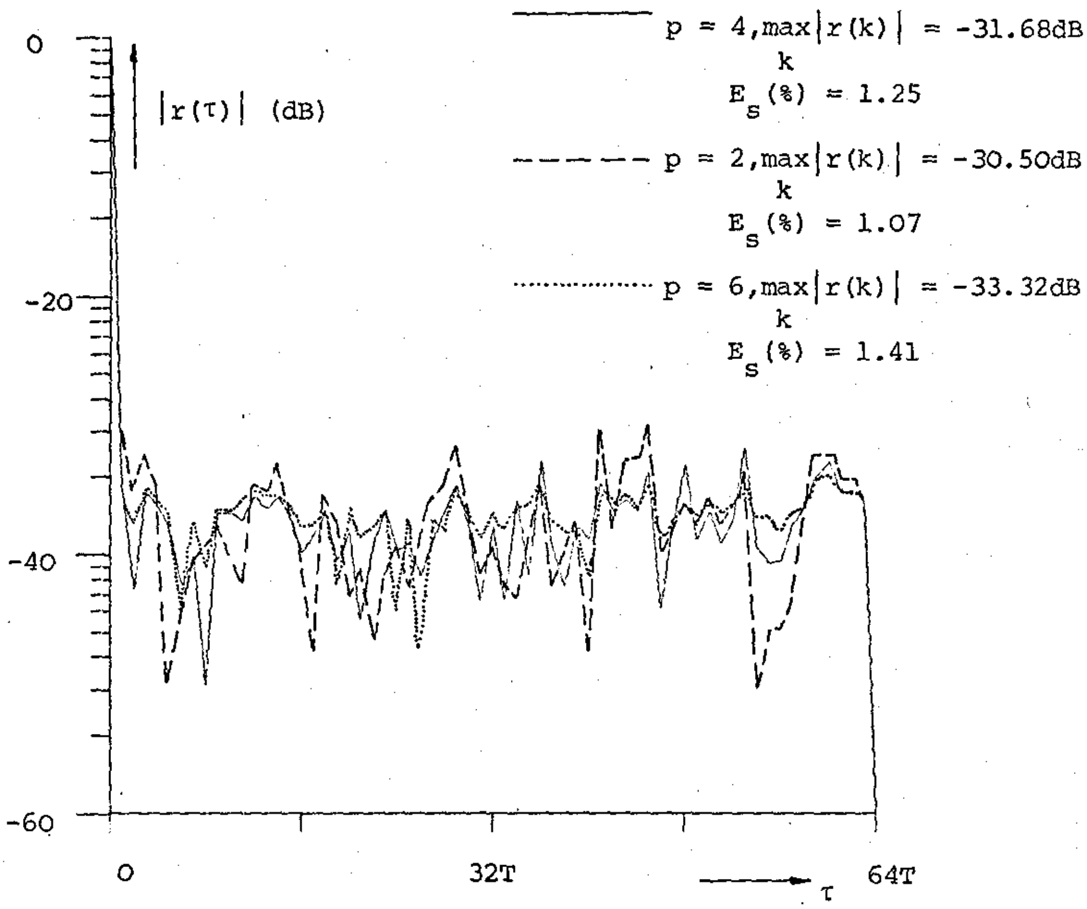


Fig. 6.7 ACF of 64-element uniform complex code for various values of p.

An estimate of the quantization noise of such a polyphase approximation can be obtained by considering the expression

$$\varepsilon^2 = \sum_{k=-(N-1)}^{N-1} |r_1(k) - r_2(k)|^2 \tag{6.7}$$

where $r_1(k)$ and $r_2(k)$ denote the ACF's of the actual sequence, $s_1(n)$, and its quantized version, $s_2(n)$, respectively.

Using Parseval's theorem the above expression can be written in its equivalent form

$$\varepsilon^2 = 1/W \int_{-W/2}^{W/2} \{|S_1(f)|^2 - |S_2(f)|^2\}^2 \, df \tag{6.8}$$

Expanding the expression in the brackets yields

$$\varepsilon^2 = 1/W \int_{-W/2}^{W/2} \{(|S_1(f)| + |S_2(f)|) \; (|S_1(f)| - |S_2(f)|)\}^2 \, df \tag{6.9}$$

For signals with good ACF's the approximation $|S_1(f)|^2 \approx N$ holds reasonably well. Moreover, the quantized signal, $s_2(n)$, should differ very little from $s_1(n)$, i.e.

$$|S_2(f)| \approx |S_1(f)| \approx \sqrt{N} \tag{6.10}$$

Consequently, it is the difference term in Eq. (6.9) which is of importance. Substituting the above approximations into Eq. (6.9) leads to

$$\varepsilon^2 = 8\,N/W \int_{-W/2}^{W/2} (N - |S_1(f)| \; |S_2(f)|) \, df \tag{6.11}$$

After applying the inequality

$$\left| \int_{-W/2}^{W/2} S_1(f) \; S_2^*(f) \; e^{j2\pi fkT} \, df \right| \leq \int_{-W/2}^{W/2} |S_1(f)| \; |S_2(f)| \, df$$

which is equivalent to

$$|r_{12}(k)| = |\sum_{n=o}^{N-1-|k|} s_1(n)\ s_2^*(n-k)| \leq 1/W \int_{-W/2}^{W/2} |S_1(f)||S_2(f)|\ df \quad k = 0, \pm 1, \ldots\ldots, \pm(N-1) \tag{6.12}$$

yields

$$\varepsilon^2 \leq 8\ N\ (N - \max_k |r_{12}(k)|) \tag{6.13}$$

A particular value of interest is obtained for zero time shift by letting $k = 0$ in (6.13)

$$\varepsilon^2 \leq 8\ N\ (N - |r_{12}(0)|) \tag{6.14}$$

Hence the mean-square error or quantization noise of two ACF's depends on the crosscorrelation of the actual and quantized signal for zero time shift.

If the phase coded signals $s_1(n)$ and $s_2(n)$ are given by

$$s_1(n) = e^{j\phi(n)} \tag{6.15}$$

and

$$s_2(n) = e^{j(\phi(n) + \Delta\phi(n))} \tag{6.16}$$

where $\Delta\phi(n)$ is the phase quantization error, the crosscorrelation for $k = 0$ is according to (6.12)

$$|r_{12}(0)| = |\sum_{n=o}^{N-1} e^{-j\Delta\phi(n)}| \tag{6.17}$$

For small values of $\Delta\phi(n)$ $(\Delta\phi^2(n) << 6)$ the exponential expression can be approximated reasonbly well by the first few terms of a Taylor series

$$e^{-j\Delta\phi(n)} \approx 1 - \tfrac{1}{2}\Delta\phi^2(n) - j\Delta\phi(n)$$

Substituting this approximation into Eq. (6.17) yields

$$|r_{12}(0)| = |\sum_{n=o}^{N-1} \{(1 - \tfrac{1}{2}\Delta\phi^2(n)) - j\Delta\phi(n)\}|$$

$$|r_{12}(0)| = \{[\sum_{n=0}^{N-1} (1 - \tfrac{1}{2}\Delta\phi^2(n))]^2 + [\sum_{n=0}^{N-1} \Delta\phi(n)]^2\}^{\frac{1}{2}} \tag{6.18}$$

The second term in Eq. (6.18) is certainly greater or equal to zero. Thus,

$$|r_{12}(0)| \geq \sum_{n=0}^{N-1} [1 - \tfrac{1}{2}\Delta\phi^2(n)] \tag{6.19}$$

If now $\Delta\hat{\phi}$ denotes the maximum phase quantization error, Eq. (6.19) becomes

$$|r_{12}(0)| \geq \sum_{n=0}^{N-1} (1 - \tfrac{1}{2}\Delta\hat{\phi}^2) \tag{6.20}$$

Dividing the phase interval 2π into M increments gives

$$\Delta\hat{\phi} = \pi/M$$

Hence

$$|r_{12}(0)| \geq N(1 - \frac{\pi^2}{2M^2}) \tag{6.21}$$

Combining Eq. (6.14) with Eq. (6.21) leads to

$$\varepsilon^2 \leq 4\pi^2 N^2/M^2 \simeq 40 N^2/M^2 \tag{6.22}$$

Using the above relationship it is possible to determine the order of the polyphase approximation for a given quantization noise ε^2. For example, if ε^2 should not exceed $0.01 N^2$ a value of

$$M \geq (4000)^{\frac{1}{2}} \simeq 64$$

or $\Delta\hat{\phi} = 2.5^{\circ}$ is required. The degradation of the ACF due to this type of quantization error is indicated in Fig. 6.6(a) for a maximum phase error of 5° (M = 36).

6.4 Binary Coded Pulse Trains

A relatively large group of signals which have received special attention are certain binary sequences (Group II, Chapter 2). Such signals have been extensively considered for improving ambiguity and resolution in radar and for solving special problems in the field of communications[5,7,8,14].

The attractive feature of binary coding is that a number of simple, efficient, and flexible decoders can be built (a pair of shift registers can be used as a tapped delay line pulse-compressor). The aperiodic ACF of a N element binary sequence c(n) can be written as

$$r(k) = \sum_{n=o}^{N-1-k} c(n)\ c(n+k) \tag{6.23}$$

where $c(n) = \pm 1 \quad n = 0,1,2,\ldots.,N-1$

A class of binary sequences whose ACF's satisfy the conditions

$$r(k) = \begin{cases} N & \text{if } k = o \\ 0 & \text{if } N-k \text{ is even} \\ \pm 1 & \text{if } N-k \text{ is odd} \end{cases} \tag{6.24}$$

are called Barker sequences or perfect words[8]. Barker sequences exist for length N = 2,3,4,5,7,11,13. Turyn[9] has shown that there are no other binary sequences with this property for $13 < N \leq 6,084$ and that it is unlikely any will exceed 6,084. The limitation to a maximum length of 13 is a serious one in radar detection. Consequently, considerable effort has been devoted to the problem of finding longer binary sequences which, if not optimum, are at least satisfactory for a given application[6,10,11,14].

It is well known that by choosing a large number of elements c(n) randomly, sequences whose r.m.s. sidelobe levels are of the order $\sqrt{N}$ can be found. However, it can be expected that in the statistical synthesis a large number of sidelobes will exceed $\sqrt{N}$. It will be shown that by proper choice of the sequence both average and peak sidelobe can be held at a lower value and thus yield a better range resolution and clutter rejection. However, nothing is known about how small the maximum peak sidelobe might be in the best cases. At present there is apparently no solution, other than an exhaustive search, to this general problem and good binary sequences which approach the Barker codes have been found only by trial and error.

The major difficulty in synthesizing binary sequences is the discrete nature of the amplitude and phase. In formulating the synthesis problem using optimization techniques an additional set of constraints has to be included. Thus the minimization problem which is equivalent to solving the system on non-linear Eq's. (6.23) may be written as

$$\min_{\underline{\phi}} F_p(\underline{\phi}) \qquad \underline{\phi} \in E^N \qquad (6.25)$$

subject to $\phi_i = 0,\pi \qquad i = 0,1,\ldots.,N-1$

Consequently, the major objective in subsequent sections will be to describe the various ways of obtaining binary coded waveforms using numerical methods, to study their properties, and to consider where they are suitable.

6.4.1 Synthesis Using Element Complementation

Probably the most obvious method of synthesizing binary sequences is to choose a number of sequences, perhaps at random, and to observe their ACF's. This method can be improved significantly by adopting a search strategy which, starting from an initial code, produces a succession of progressively better codes. This involves minimization of a measure of the sidelobes such as that given by Eq. (6.25) over a set of discrete points in multi-dimensional space. To be more specific, the problem is to minimize a function of N discrete variables over a set, S, of 2^N points in N-dimensional space. Any attempt to compute all possible functional values becomes unfeasible, even for moderately large N $(N > 20)$, as 2^N increases exponentially. Hence, the search will have to be restricted to a subset S_1 of $S (S_1 \subset S)$. In addition the iterative method must take the discrete nature of the variables into account and, moreover, must be economical with respect to the volume of computations.

A simple search strategy is to take the current sequence, changing one of its elements to either +1 or -1, and to evaluate the ACF. If the measure of the sidelobes is reduced the modification is retained and the new code is subjected to further modifications. This is done iteratively until changes of the elements do not yield a further reduction in the objective function. Such a method was proposed by Vakman[11] and later adopted by Indiresan[74] and referred to as the element complementation technique.

Vakman suggested choosing the initial starting sequences to be of the form

$$c(n) = \text{sgn} \;\{\cos(\frac{\pi}{2} \;\frac{(n-1)^2}{(N-1)} - \frac{\pi}{4})\} \qquad (6.26)$$

where $$\text{sgn}\; x = \begin{cases} 1 & x \geqslant 0 \\ -1 & x < 0 \end{cases}$$

It can be shown that these sequences, referred to as Vakman codes, are binary approximations to a LFM waveform. Such codes have an estimated r.m.s. sidelobe of $\sqrt{N}$. In addition odd length Vakman codes satisfy the reflection condition*

$$c(\frac{(N+1)}{2} + n) = (-1)^n \;\; c(\frac{(N+1)}{2} - n) \qquad (6.27)$$

It is easy to verify that sequences which satisfy this condition also

*Another type of code which also satisfies (6.27) will be discussed in Chapter 7.

satisfy $r(k) = 0$ for $(N-k)$ even, since the terms in $r(k)$ cancel in pairs (first and last, etc.) A typical example of a Vakman code ACF of length $N = 101$ is shown in Fig. 6.8.

A search program based on the element complementation method was written in Fortran to find sequences with minimum sidelobe peaks. The initial starting points were:

(i) Vakman code

(ii) randomly chosen sequence

In addition the sequences were also tested to see if cyclic shifts would improve the objective function. It is apparent from Eq. (6.23) that $\max_k |r(k)|$ cannot change by more than two units when the initial bit is moved one place, the length of the sequence remaining fixed.

In the binary case the updating of the ACF (Eq.(6.5)) after each element change reduces to

$$\Delta r(k) = \begin{cases} -2c(m)\left[c(m+k) + c(m-k)\right] & ;\ m-k \geq 0 \text{ and } m+k \leq N-1 \\ -2c(m)\ c(m+k) & ;\ m-k < 0 \text{ and } m+k \leq N-1 \\ -2c(m)\ c(m-k) & ;\ m-k \geq 0 \text{ and } m+k > N-1 \end{cases} \tag{6.28}$$

where $c(m)$ is the old value of the mth element. A similar expression which economises computation time can be obtained for the cyclic shifts, since such an operation results in merely removing and adding bits of the sequence.

Table 6.7 shows the binary sequences obtained when optimizing Vakman codes with respect to the criteria

$$\sum_k |r(k)|^2,\ \sum_k k|r(k)|^2,\ \sum_k |r(k)|^4,\ \sum_k k|r(k)|^4$$

For comparison purposes the signals obtained by Vakman are also given together with the sidelobe energy ratio E_s. (Some of the sequences are appended).

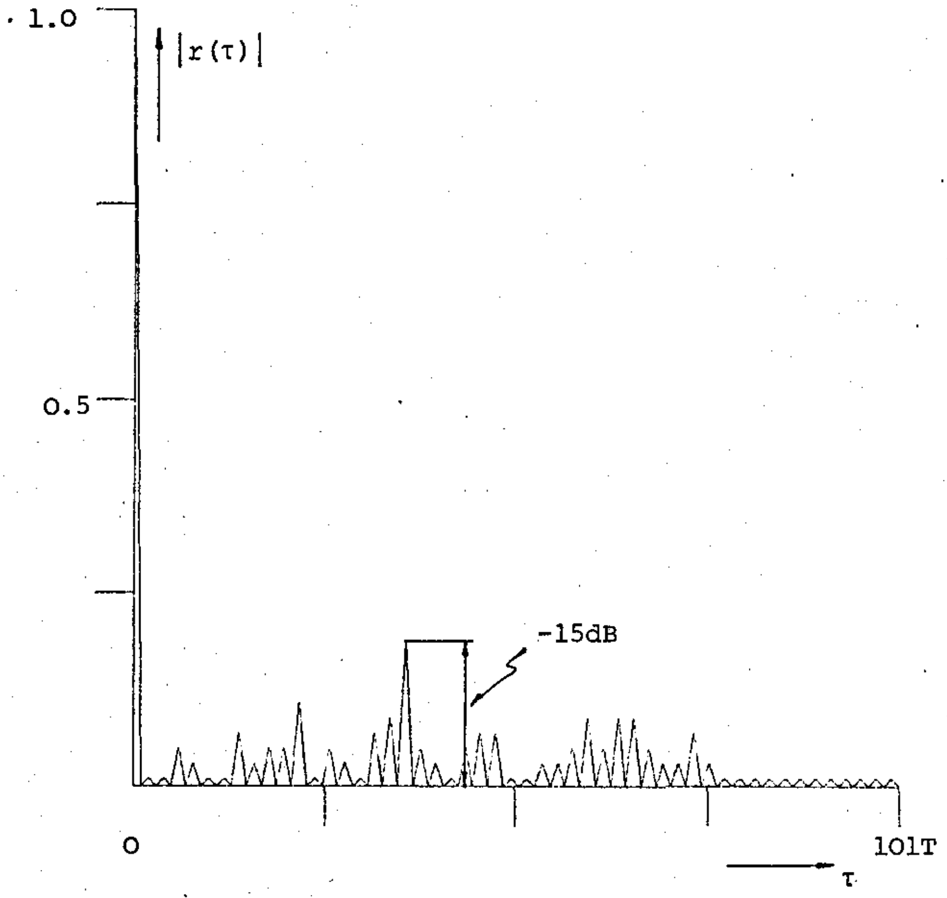


Fig. 6.8 ACF of 101-element Vakman code.

Code Length	FUNCTIONALS									
N	$\Sigma\lvert r(k)\rvert^2$			$\Sigma k\lvert r(k)\rvert^2$		$\Sigma\lvert r(k)\rvert^4$			$\Sigma k\lvert r(k)\rvert^4$	
	S_{max}	E_s(%)	(V)S_{max}	S_{max}	E_s(%)	S_{max}	E_s(%)	(V)S_{max}	S_{max}	E_s(%)
13	1	3.55	1	1	3.55	1	3.55	1	1	3.55
19	3	13.57	3	5	19.11	3	11.35	3	3	13.57
23	3	11.15	3	3	11.15	3	11.15	4	3	11.15
31	9	11.55	9	5	12.38	4	16.50	4	6	21.54
37	5	9.78	5	5	10.08	5	11.25	5	5	11.25
41	5	6.90	5	12	17.13	5	15.94	6	6	15.71
43	5	13.25	6	7	12.39	5	16.27	5	6	15.80
47	4	8.28	7	8	10.46	5	14.26	4	7	14.80
53	6	10.18	6	9	14.88	5	12.31	5	7	13.46
59	7	11.52	9	9	10.60	8	16.23	8	8	21.06
61	7	9.62	7	9	13.06	6	10.37	7	8	10.18
63	7	8.64	7	7	8.64	7	8.64	7	7	8.64
67	8	9.91	7	8	10.54	7	15.26	7	7	15.44
71	9	8.86	7	11	13.31	9	12.51	9	7	12.52
73	8	11.03	8	8	12.76	7	11.33	7	9	14.71
79	8	11.39	7	12	11.97	8	16.65	8	9	14.73
91	9	13.83	10	15	16.19	9	15.28	9	12	24.21
93	13	8.11	13	9	10.52	8	10.05	8	11	13.58
95	9	12.04	9	17	13.91	8	13.28	8	11	15.95
95	10	14.03	9	13	15.18	9	19.11	10	13	20.87
99	13	10.05	13	19	15.36	9	10.86	9	11	15.56
101	9	10.64	8	15	15.27	9	15.82	9	8	11.90
103	13	11.49	11	15	13.49	9	15.45	9	11	13.49
105	13	10.77	13	16	16.33	9	13.68	9	9	14.66
107	13	9.82	11	13	13.29	9	12.27	9	11	15.84
109	11	10.76	13	15	12.54	9	15.30	9	11	17.29
111	10	10.56	13	11	13.89	9	14.53	10	12	16.55
113	9	9.02	9	11	9.24	9	14.31	9	11	14.79
115	10	10.71	10	18	12.86	9	11.86	9	13	16.04
117	10	10.36	11	13	12.29	12	13.03	12	10	13.69
119	9	12.82	11	15	12.62	9	13.80	10	10	16.15
121	9	11.80	12	12	13.22	10	13.82	11	12	16.12
123	11	11.77	14	17	14.52	10	14.49	10	13	18.70
125	11	13.24	11	16	10.92	7	10.25	7	12	17.22
251	19	10.93	17	32	15.08	13	12.19	13	20	15.41
253	21	10.42	21	28	13.78	13	13.28	13	24	16.44
255	19	10.22	16			14	12.09	15		
257	16	10.50	18			15	15.40	15		
259	17	13.17	17			13	12.97	13		
261	27	9.37	27			15	12.54	15		
299	18	11.27	18			17	16.25	17		
301	18	9.97	19			18	11.40	16		
303	21	10.40	23			16	13.76	15		
305	17	9.74	19			16	13.62	16		
503	25	10.79	27			22	13.28	22		
511	25	11.32	28			22	14.39	23		
513	23	9.14	25			21	14.04	21		

Table 6.7 Binary sequences obtained with Vakman code as initial starting point, (V) indicates best Vakman sequences.

Having found a good sequence it is often helpful to remove or to adjoin bitsto reduce the sidelobes even further. In the worst case such changes may increase the sidelobe level by an amount equal to the number of insertions or removals. The search covered a chosen range of initial sequence length and codes of every length were tested starting from a specified minimum up to a length exceeding the minimum by a fixed amount (at present 32 bits). Fig. 6.9 shows the variation in the maximum sidelobe for successively adding or removing bits. It can be seen that a number of sequences of various lengths exist which have the same maximum sidelobe level.

In some cases better codes than those obtained by Vakman have been found. As expected minimization of a ℓ_4-measure yields better results particularly for large compression codes. The energy ratio E_s varies between 10% and 16% of the mainlobe energy, which gives a r.m.s. sidelobe of approximately 0.3 $\sqrt{N}$ to 0.4 $\sqrt{N}$. As mentioned previously a non-negative weighting of the objective function enables control of the sidelobe distribution along the delay axes of the ACF. A designer may feel that it is worthwhile reducing the sidelobes which are further away from the main peak, at the expense of an increase in the close-in sidelobes, by minimizing a functional of the form

$$F_p = \sum_{k=1}^{N-1} k \, |r(k)|^p \qquad (6.29)$$

This is illustrated in Fig. 6.10.

On the other hand, when a binary sequence is used for resolving closely spaced targets or observing missiles in the presence of tank fragments or decoys for example, it is desirable to have low residues near the mainlobe. However, ambiguities can be tolerated in range if they are sufficiently distantfrom the expected target position. In this situation a weighting function of the form

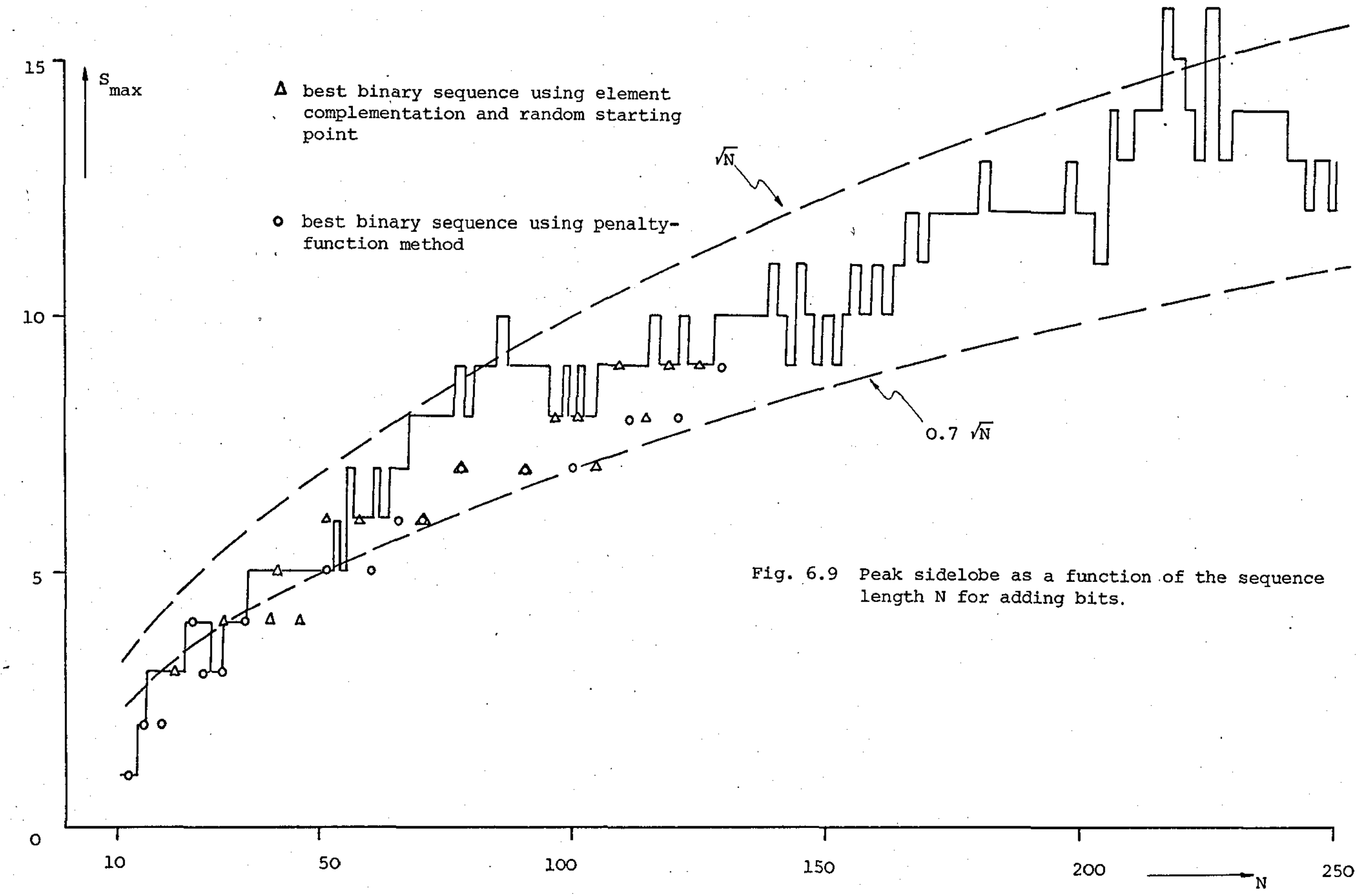


Fig. 6.9 Peak sidelobe as a function of the sequence length N for adding bits.

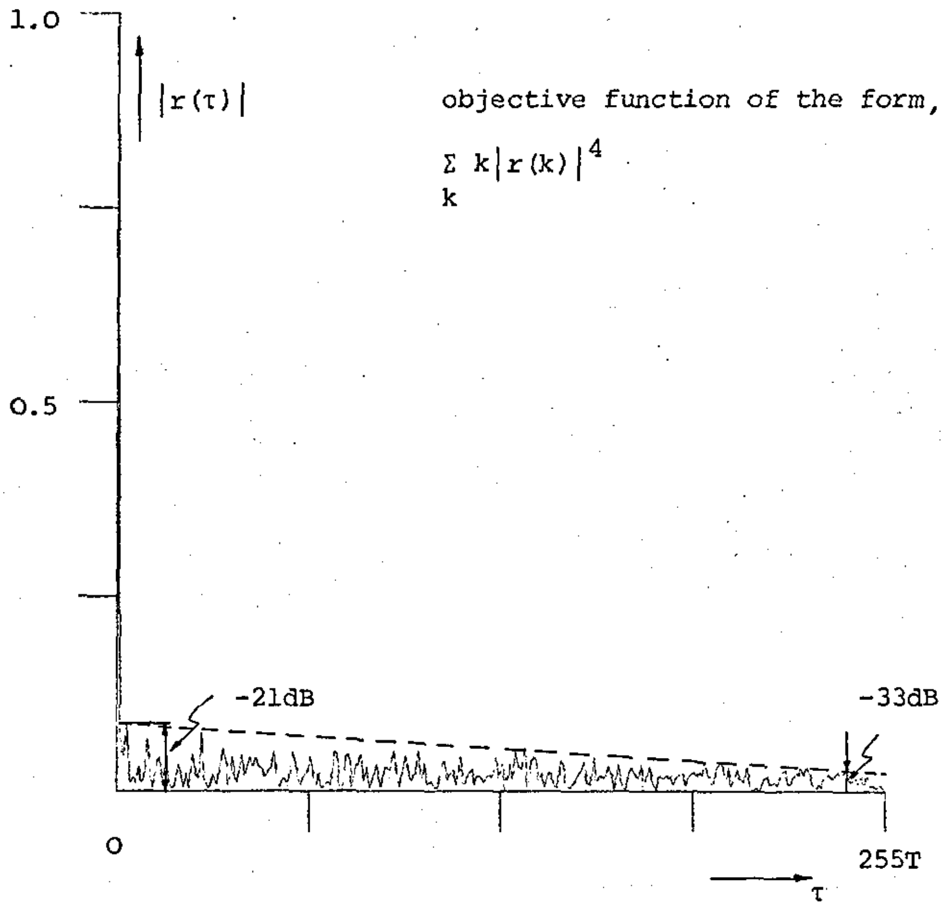


Fig. 6.10 ACF of 255-element binary sequence for linear weighting sequence.

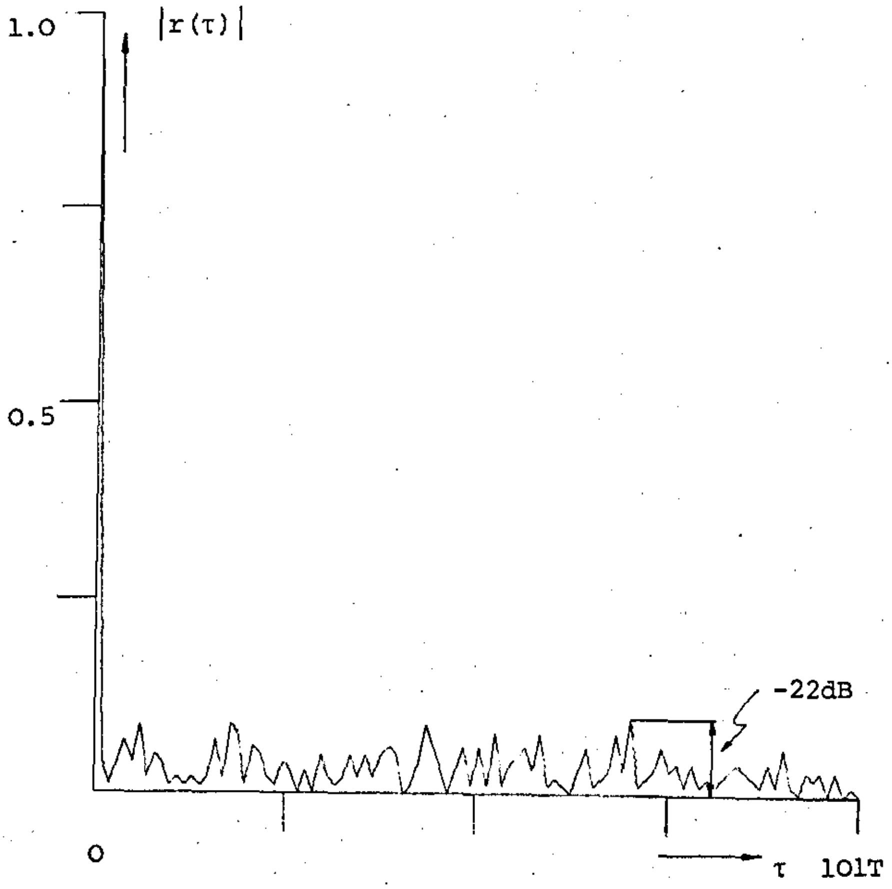


Fig. 6.11 ACF of optimum 101-element binary sequence with random initial starting point.

$$w(k) = N - k$$

or

$$w(k) = \begin{cases} 1 & , \quad k \leqslant N_1 \\ 0 & , \quad N_1 < k \leqslant N-1 \end{cases}$$

where N_1 denotes the desired clear region, could be used. For example sequences of length $N > 200$ whose mainlobe-to-sidelobe ratio is 200:1 or 46 dB within 10 segment length of the main peak were found. Thus with a proper choice of the weighting function it is possible to obtain sequences with clear regions. However, an increase in the maximum and average sidelobe levels is usually observed. In addition, to obtain this degree of discrimination the decoder must be highly linear and have a large dynamic range.

An extensive computer search has shown that starting with a randomly chosen initial sequence instead of a Vakman code can give appreciably better results. Table 6.8 shows that binary sequences with lower sidelobe peaks than those obtained by Vakman were found. The r.m.s. level is about $0.4\sqrt{N}$ and $\max\limits_{k} |r(k)|$ ranges from $0.6\sqrt{N}$ to $\sqrt{N}$ but generally does not exceed $\sqrt{N}$ as illustrated in Fig. 6.9. A typical ACF of such a sequence is depicted in Fig. 6.11.

It has been observed that there are a number of good binary sequences which have the lowest sidelobe known for a given length. In general, however, they will differ in sidelobe energy performance.

The element complementation method requires a computation time which increases almost linearly with the code length N. In most cases the optimum was found in about 3 to 4 iterations*. On an ICL 1904A digital computer good binary sequences of length as large as N = 900

*An iteration consists of complementing the N bits of the sequence plus N cyclic shifts.

Code length	Functional: $\sum_k \lvert r(k) \rvert^4$		Best Vakman sequence
N	$\max_k \lvert r(k) \rvert$	E_s (%)	$\max_k \lvert r(k) \rvert$
13	1	3.55	1
19	3	11.36	3
23	3	11.91	3
31	4	10.72	4
37	4	11.83	5
41	4	12.61	6
43	5	12.17	5
47	4	8.28	5
53	6	15.88	5
59	6	14.62	8
61	6	10.91	7
63	6	11.96	7
67	6	12.77	7
71	6	9.26	7
73	7	13.51	8
79	7	12.99	8
91	7	16.15	9
93	7	15.52	8
95	8	13.60	8
97	8	13.52	10
99	8	15.89	9
101	8	13.16	9
103	8	14.58	9
105	7	13.61	9
107	8	12.90	9
109	9	12.40	9
111	9	14.34	10
113	9	12.88	9
115	8	14.65	9
117	8	14.83	12
119	9	15.62	10
121	9	15.46	11
123	9	11.72	10
125	9	15.83	7
251	14	14.41	13
253	15	14.92	13
255	14	14.44	15
257	15	14.76	15
259	13	14.66	13
299	15	15.09	17

Table 6.8 Best binary sequences obtained with random initial starting point.

were found in less than 30 minutes. On a time-sharing computer a search program of this kind runs at low priority very cheaply, since both input and output data are extremely small.

The element complementation method, although simple and efficient, is not without its disadvantages. First, no local explorations of the function to be minimized can be obtained. Secondly, one has little or no control over the rate of convergence. Therefore, a method which, to some extent, overcomes these difficulties will now be described.

6.4.2 A New Technique of Synthesizing Binary Sequences

Because of the lack of suitable methods for solving non-linear equations with integer variables it is necessary to accept formulations in continuous variables with 'rounding off' the resulting solution or, alternatively, to adopt a search strategy of the type described previously. Although this may be satisfactory for problems with small dimensions, it may lead to results quite distant from the optimum if the number of variables increases. Therefore, a new approach to the problem, using non-linear integer programming, is suggested[75].

In recent years a large number of programming methods for solving linear integer problems have been proposed by many authors[76,77]. However, in the field of non-linear integer programming little progress seems to have been made. An interesting approach to this problem, utilizing penalty-functions, was suggested by Gisvold and Moe[78]. Here the possibility of using penalty-function techniques to synthesize binary sequences with low sidelobes is investigated.

The minimization of a measure of the sidelobes such as

$$F_4 = \sum_{k=1}^{N-1} \left| \sum_{n=o}^{N-1-k} c(n)\ c(n+k) \right|^4 \qquad (6.30)$$

subject to the constraints

$$c(n) = \underline{+}1 \qquad n = 0,1,2,\ldots,N-1$$

can be regarded as a non-linear constrained optimization problem. By making the variable substitutions

$$\cos x_n = c(n)$$

Eq. (6.30) can be written as

$$F_4(\underline{x}) = \sum_{k=1}^{N-1} \left| \sum_{n=o}^{N-1-k} \cos x_n \cos x_{n+k} \right|^4 \qquad (6.31)$$

$$x_n = 0, \pi \qquad n = 0,1,2,\ldots,(N-1) \qquad (6.32)$$

The optimum realizable approximation to the desired ACF of Eq. (6.24) is given by the set of phases $\{x_n\}$ that minimize the cost functional $F_4(\underline{x})$ subject to the conditions (6.32). The essential idea of the penalty-function approach is to transform the constrained non-linear problem into a sequence of unconstrained problems by adding the functions of the constraints to the objective function and deleting the constraints as such. Formally this can be written as

$$\text{Minimize:} \quad P(\underline{x},\underline{s}) = F_p(\underline{x}) + \sum_{i=1}^{m} s_i \, H\left[h_i(\underline{x}) \right] \qquad (6.33)$$

where $P(\underline{x},\underline{s})$ is a generalized augmented function, $s_i > 0$ are weighting factors, and $H\left[h_i(\underline{x})\right]$ is a function of the equality constraints $h_i(\underline{x})$. A typical choice for $H\left[h_i(\underline{x})\right]$ is

$$H\left[h_i(\underline{x}) \right] = h_i^2(\underline{x}) \qquad (6.34)$$

Using this method, the problem represented by Eq!s (6.31) and (6.32) can be transformed into the unconstrained minimization problem

$$\text{Minimize:} \quad P(\underline{x},s^{(k)}) = F_4(\underline{x}) + (s^{(k)})^{-\frac{1}{2}} Q(\underline{x}) \qquad (6.35)$$

The term $Q(\underline{x})$ is the discretization penalty-function and $s^{(k)} > 0$ is a weighting factor. This function has the property

$$Q(\underline{x}) = \begin{cases} 0 & \text{if } x_n = 0,\pi \\ \Lambda > 0 & \text{if } x_n \neq 0,\pi \end{cases} \qquad (6.36)$$

In principle $Q(\underline{x})$ could have any form as long as it is continuous and the requirement (6.36) is met. A suitable criterion seems to be

$$Q(\underline{x}) = \sum_{n=o}^{N-1} (\sin^2 x_n)^{\gamma} \tag{6.37}$$

where $\gamma \geq 1$.

The factor γ allows for shaping the penalty-function, while the amplitude is controlled by the weighting coefficient $s^{(k)}$.

The augmented function $P(\underline{x},s^{(k)})$ is now minimized for a sequence of decreasing values of weighting factors $s^{(k)}$, such that, for $k \to \infty$,

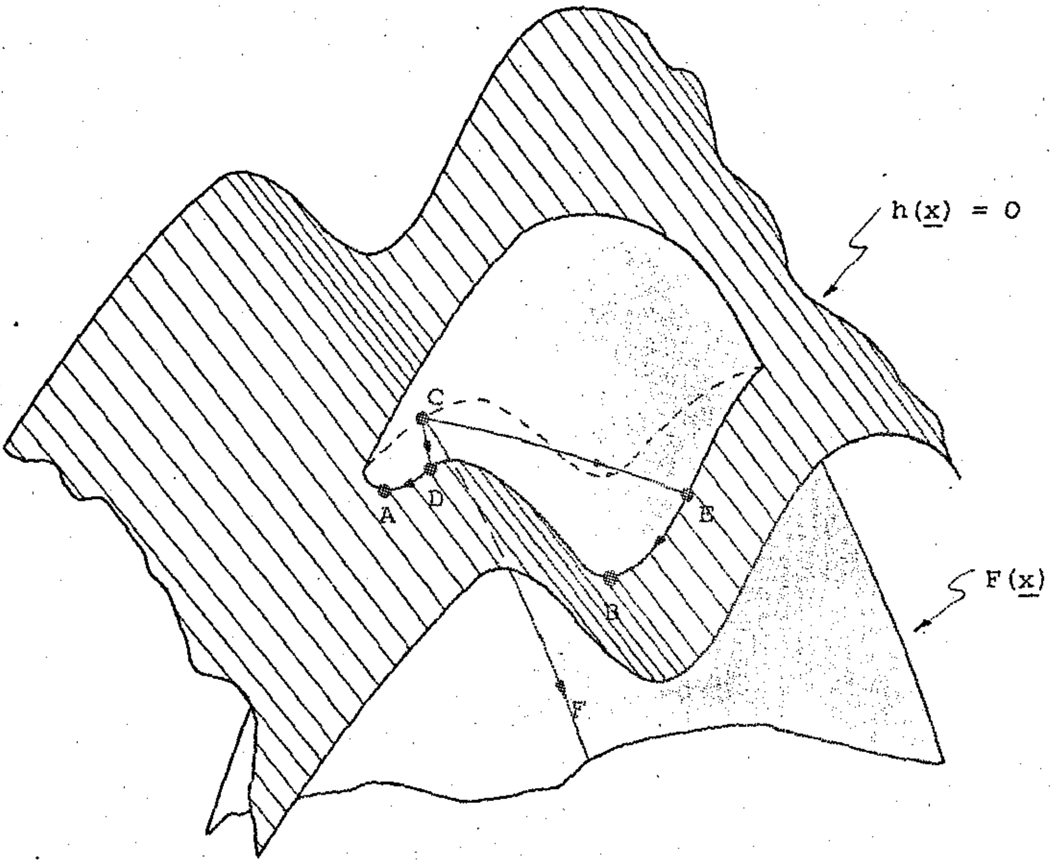
$$\min\{P(\underline{x},s^{(k)})\} \to \min\{F_4(\underline{x})\} \tag{6.38}$$

and

$$Q(\underline{x}) \to 0 \tag{6.39}$$

In other words, the effects of the constraints in the augmented function $P(\underline{x},s^{(k)})$ on the value of the function is gradually diminished as the search continues, and completely removed at the limit, so that the value of the augmented function converges to the same value as $F_4(\underline{x})$ and the extremum of $P(\underline{x})$ is the same as that of $F_4(\underline{x})$. In practice the required number of cycles is rather small, i.e. k_{max} is between ten and twenty.

The choice of the initial $s, s^{(o)}$, the method of reducing s, and the choice of $Q(\underline{x})$ all have considerable influence on the effectiveness of the method. If $s^{(o)}$ is too small, the initial minimization will drive $\underline{x}$ to the minimum of $F_4(\underline{x})$ itself, a point unlikely to be the constrained minimum (path CF, Fig. 6.12). On the other hand, if $s^{(o)}$ is too large the search tends to converge prematurely in the first few stages to some non-optimal solution such as that illustrated in Fig. 6.12 (CDA). The sequence of $\{s^{(k)}\}$ has been chosen according to the method suggested by Fiacco and McCormick[79].



Path CDA $s^{(o)}$ too large

Path CEB $s^{(o)}$ moderate

Path CF $s^{(o)}$ too small

Fig. 6.12 Influence of weighting parameter $s^{(o)}$ on the path of optimization (A,B local minima).

$$s^{(k+1)}/s^{(k)} = 1/c$$

where $c \approx 4$ and $s^{(o)} = 1$

The advantage of this method is that in principle any of the well known search algorithms can be used to minimize $P(\underline{x},s^{(k)})$. However, the study described in Section 6.1 has shown that pattern search techniques are particularly suitable for this type of application. Such a search algorithm was written in Fortran and the flow diagram is given in Fig. 6.13.

In deciding upon a convergence criterion for the constrained equations, account must be taken of the magnitudes of the quantities being dealt with. Therefore, the search was terminated when the sum of the squares of the violated equality constraints was less than some small positive number ε, i.e.

$$Q(\underline{x}) < \varepsilon$$

Preliminary investigations have shown that a value of 0.05 N for ε appeared to be adequate.

The binary sequences obtained, starting with an initial set of phases

$$x_n = \frac{\pi n^2}{2(N-1)} \qquad n = 0,1,2,\ldots.,N-1$$

are given in Table 6.9.

A comparison of the results with known better codes [10,80] (Table 6.9) indicate that the proposed method gives satisfactory results. In addition this method has a number of advantages over other design techniques. First, no exhaustive and thus time-consuming search procedure is required. For example, sequences of length $N \approx 100$ were obtained in less than 5 min on an ICL 1904A computer. This is

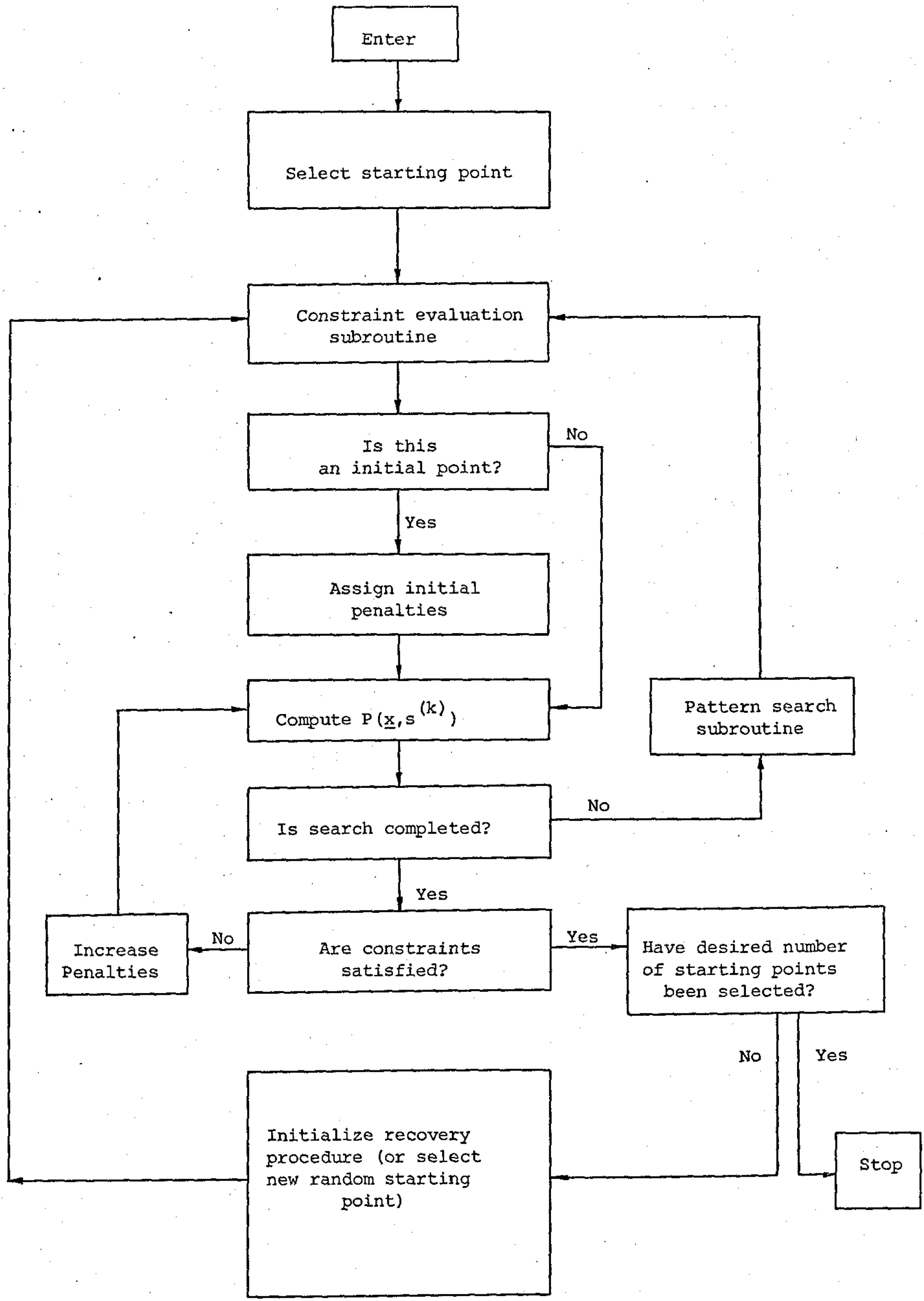

Enter
Select starting point
Constraint evaluation subroutine
Is this an initial point?
No
Yes
Assign initial penalties
Compute P(x,s(k))
Pattern search subroutine
Is search completed?
No
Yes
Increase Penalties
No
Are constraints satisfied?
Yes
Have desired number of starting points been selected?
No
Yes
Initialize recovery procedure (or select new random starting point)
Stop

Fig. 6.13 Flow diagram of penalty-function algorithm.

Code length	Penalty-function method		Improved penalty-function method		Best known signal
N	$\max_k \lvert r(k) \rvert$	E_s (%)	$\max_k \lvert r(k) \rvert$	E_s (%)	$\max_k \lvert r(k) \rvert$
13	1	3.55	1	3.55	1*
17	3	13.84	2	11.07	2*
19	3	15.79	2	10.25	2*
21	3	12.25	3	12.25	2*
23	3	11.91	3	15.69	3*
25	4	10.89	4	10.24	2*
27	3	8.37	3	8.37	3*
31	3	6.97	3	6.97	3*
33	4	12.49	4	11.02	3*
35	4	10.53	4	10.53	3*
37	4	10.67	4	10.08	3*
41	4	14.52	4	6.66	4
47	5	11.54	4	8.29	4
53	6	14.88	5	10.89	5
61	6	10.80	5	12.20	5
67	6	11.43	6	11.43	5
71	6	11.80	6	11.80	5
79	7	9.53	7	8.76	6
91	7	12.38	7	12.38	
97	8	13.05	8	13.05	
101	8	11.94	7	10.29	
113	8	9.15	8	9.15	
121	9	12.68	8	9.54	
128	10		9	11.33	

Table 6.9 Binary sequences obtained using penalty-function techniques.
(*minimum possible sidelobe found by exhaustive search[80]).

considerably less than the time required using the method presented by Boehmer[10], but is comparable in efficiency with the element complementation technique. An additional increase in efficiency can be achieved by limiting the number of moves at a given penalty level. It has been found that when the number of function evaluations exceeds 30xN to 50xN, only small changes in the variables are made. Even though convergence has not been reached, the variables are close enough to their final values to move on to the next penalty level. Owing to the increased efficiency relatively long sequences can be handled in a reasonable computation time. Another advantage is that no prior information of the structure of the function to be minimized is required. However, since the function is multimodal, the search procedure most likely converges to a local extremum.

The problem of premature convergence to a 'false optimum' can be tackled by either beginning the search at several different random starting points or even better, to initialize a recovery procedure. Such a procedure is based on the observation that a reduction of the penalty weighting factor $s^{(k)}$ will cause the minimum point to move away from the constraint boundary. If the reduction of $s^{(k)}$ is large enough, this will cause the search procedure to converge to a different (local) minimum.

6.4.3 Improved Penalty-Function Method

Although the minimization of $P(\underline{x},s^{(k)})$ seems promising in view of the results obtained, the method is not without its disadvantages; the weighting factor $(s^{(k)})^{-\frac{1}{2}}$ becomes very large as the search progresses. This causes the function that is minimized to be very sensitive to variable changes. Particularly in cases where the objective function is ill-conditioned (deep narrow valleys) difficulties arise because the location of the minimum usually requires a large number of steps.

An improved penalty-function method which to some extent overcomes this problem was suggested by Powell[81]. Powell's modified augmented function can be written in the general form

$$\text{Minimize:}\quad P(\underline{x},\underline{s},\underline{\nu}) = F_p(\underline{x}) + \sum_{i=1}^{m} s_i (h_i^2(\underline{x}) + \nu_i)^2 \tag{6.40}$$

where $(s_1, s_2, \ldots, s_m)$ are the usual penalty weighting factors and $(\nu_1, \nu_2, \ldots, \nu_m)$ are an additional set of parameters. The main difference between Eq. (6.33) and Eq. (6.40) is the introduction of the parameters $\underline{\nu}$. Because of these parameters it is usually satisfactory to use moderate values of $\underline{s}$.

The method is based on the fact that if the values of the variables $\underline{x}$ which minimize $P(\underline{x},\underline{s},\underline{\nu})$ are $\underline{\xi}(\underline{s},\underline{\nu})$, then $\underline{\xi}(\underline{s},\underline{\nu})$ is a solution to the constrained problem:

$$\text{Minimize:}\qquad F_p(\underline{x}) \tag{6.41}$$

subject to

$$h_i^2(\underline{x}) = h_i^2[\underline{\xi}(\underline{s},\underline{\nu})] \qquad i = 1,2,\ldots,m$$

This implies that values of the parameters $\underline{s},\underline{\nu}$ can be found such that

$$h_i^2[\underline{\xi}(\underline{s},\underline{\nu})] = 0 \qquad i = 1,2,\ldots,m \tag{6.42}$$

Eq. (6.42) represents a system of non-linear equations which, in principle have to be solved for the unknown parameters $\underline{\nu}$ for fixed values of $\underline{s}$. The solution of this set of equations could turn out to be difficult and time-consuming. Fortunately, it can be shown[81] that if the factors s_i are sufficiently large an iterative adjustment such as

$$\nu_i^{(k+1)} = \nu_i^{(k)} + [h_i^2(\underline{\xi})]^{(k)} \qquad i = 1,2,\ldots,m \tag{6.43}$$

is adequate. In addition by choosing the sequence $\{s_i\}$ sufficiently large the iteration (6.43) can be made to have linear convergence at as fast a rate as is required.

Using the modified method the binary signal design problem can now be formulated as

$$\text{Minimize:} \quad P(\underline{x},s^{(k)},\nu^{(k)}) = F_4(\underline{x}) + s^{(k)} \{Q(\underline{x}) + \nu^{(k)}\}^2 \tag{6.44}$$

The new augmented function is now minimized applying Eq. (6.43) for a fixed value of $s^{(k)}$, unless $c^{(k)} = Q(\underline{x})$ either fails to converge or converges to zero too slowly. In this case $s^{(k)}$ is increased in order to improve the rate of convergence. A flow diagram of the algorithm is shown in Fig. 6.14. The search process is started with the initial parameters $s^{(o)} = 1$, $\nu^{(o)} = 0$, and $c^{(o)} = K$, where K is some positive number exceeding $Q(\underline{x})$. The correction (6.43) is applied if the required convergence

$$c^{(k)} = 1/\alpha \; c^{(k-1)}$$

is obtained, otherwise the value of $s^{(k)}$ is increased by a factor β. Powell suggested increasing $s^{(k)}$ so that $s^{(k)}\nu^{(k)}$ remains unchanged. This seems reasonable because at the optimum solution the final value of $s^{(k)}\ \nu^{(k)}$ is independent of the parameters.

The sequences obtained using the modified penalty-function method for for $\gamma = 1$ and starting with the same initial set of phases $\{x_n\}$ as before are given in Table 6.9. The codes indicated by (*) are known to be the best possible[80]. A typical ACF of a binary sequence of length N = 101 is depicted in Fig. 6.15.

As mentioned previously the new method is less sensitive to changes in the variables as the constraints are approached. Consequently, sequences with better sidelobe and sidelobe energy performance were found in many cases. Moreover, the ability to increase $s^{(k)}$ can achieve linear convergence at a rate given by (6.43) which makes the modified algorithm clearly superior in efficiency to other design techniques. A factor of $\alpha = 1/4$ and $\beta = 4$ was chosen in this case but

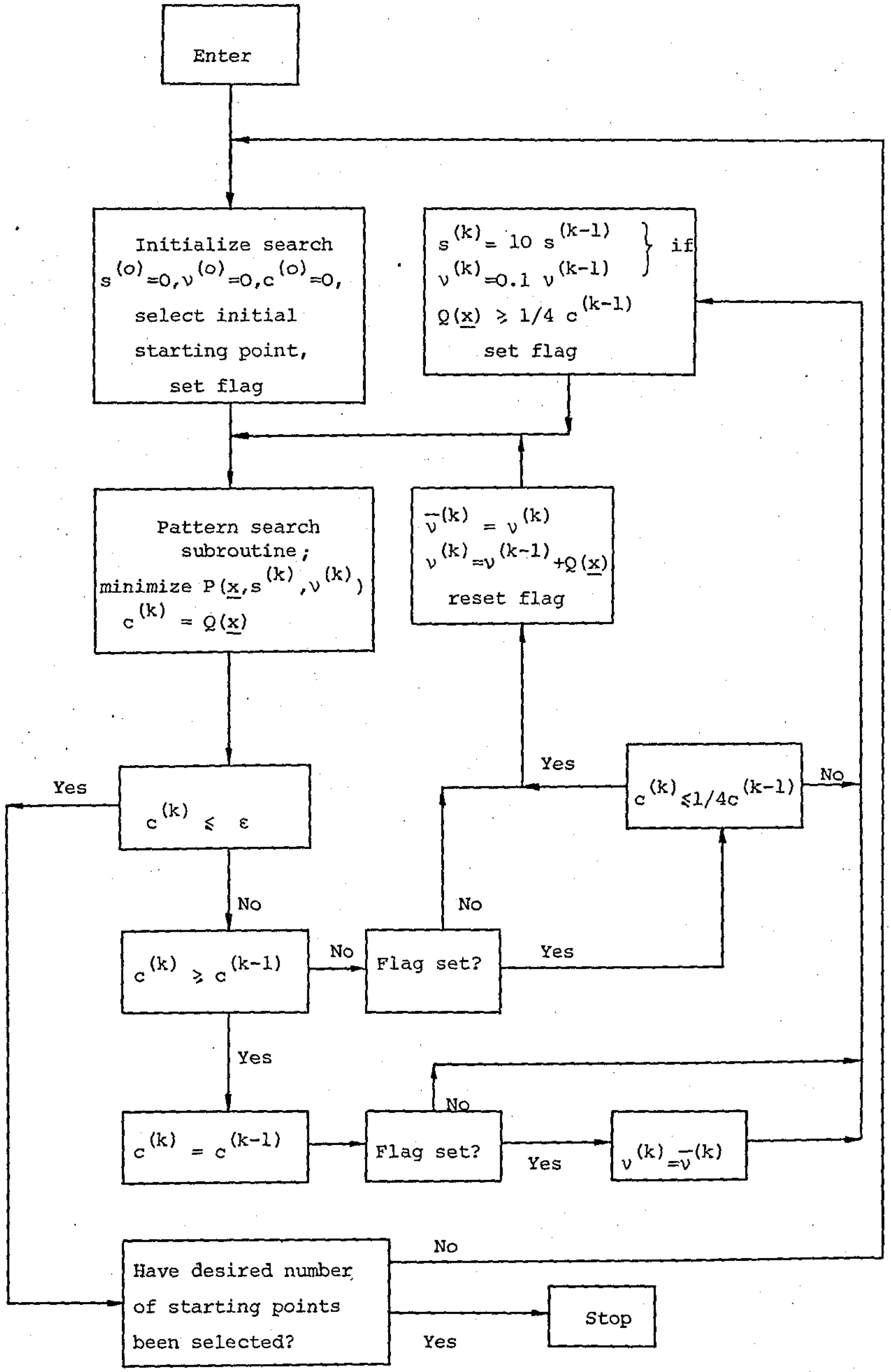

Enter
Initialize search $s^{(o)}=0, \nu^{(o)}=0, c^{(o)}=0$, select initial starting point, set flag
$s^{(k)} = 10\ s^{(k-1)}$
$\nu^{(k)}=0.1\ \nu^{(k-1)}$
if
$Q(\underline{x}) \geqslant 1/4\ c^{(k-1)}$
set flag
Pattern search subroutine; minimize $P(\underline{x}, s^{(k)}, \nu^{(k)})$
$c^{(k)} = Q(\underline{x})$
$\bar{\nu}^{(k)} = \nu^{(k)}$
$\nu^{(k)} = \nu^{(k-1)} + Q(\underline{x})$
reset flag
Yes
$c^{(k)} \leqslant \varepsilon$
Yes
$c^{(k)} \leqslant 1/4 c^{(k-1)}$
No
No
No
$c^{(k)} \geqslant c^{(k-1)}$
No
Flag set?
Yes
Yes
No
$c^{(k)} = c^{(k-1)}$
Flag set?
Yes
$\nu^{(k)} = \bar{\nu}^{(k)}$
No
Have desired number of starting points been selected?
Stop
Yes

Fig. 6.14 Flow diagram of improved penalty-function method.

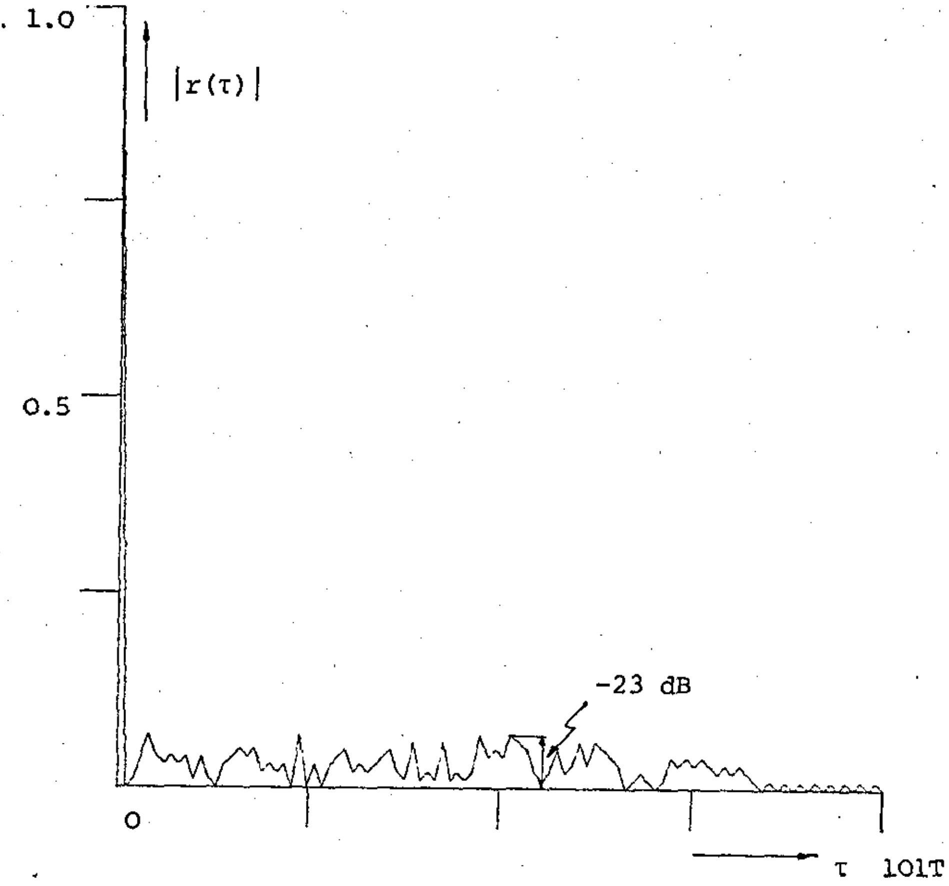


Fig. 6.15 ACF of 101-element binary sequence using penalty-function method.

it is possible to obtain faster convergence if desired. However, increasing the rate of convergence seems to make the unconstrained minimization problem more difficult (premature termination) and in practice the best choice of the factors α, β must be determined experimentally.

At this point it is of interest to note that penalty-function techniques can easily be modified to synthesize polyphase codes, i.e. uniform complex codes whose phases are integer multiples of a basic phase angle $x_o = 2\pi/M$.

Initial trials to find good 6-phase codes using a quantization penalty-function

$$Q(\underline{x}) = \sum_{n=o}^{N-1} \sin^2(x_n M/2) \tag{6.45}$$

where M = 6 are shown in Table 6.10.

Non-linear integer programming is a research area which has, so far, received very little attention. However, the results of this study are very encouraging and show that penalty-function techniques provide a useful tool in tackling problems with integer variables.

6.4.4 Other Binary Sequences

The performance of a binary sequence can be improved significantly using numerical techniques. The attainable energy ratios and peak sidelobes are of the order of 20% and $\sqrt{N}$ respectively. The still relatively large residues might be the limiting factor for particular applications such as precision trackers. One approach to try and improve the range resolution and yet to maintain the binary nature of the sequence is to introduce zeros, that is by setting a number of elements in the sequence equal to zero. The resulting sequence can be regarded as having three possible levels, namely ± 1 or 0, and is sometimes referred to as interrupted binary sequence or ternary code.

Code length	Peak sidelobe	Energy ratio
N	$\max_k \lvert r(k) \rvert$	E_s (%)
15	1.73	8.00
16	1.73	7.81
17	1.73	6.57
18	1.73	7.72
19	2.00	8.03
20	1.73	7.25
21	2.00	7.94
22	2.00	6.82
23	2.00	4.35
24	2.00	8.16
25	2.00	6.72
26	2.00	6.21
27	2.65	7.41
28	2.65	8.42
29	3.46	9.99
30	3.00	8.78

Table 6.10 Six-phase codes obtained using penalty-function techniques.

The inevitable loss in transmitted signal energy associated with ternary codes can be kept small provided that the number of zeros, L, is small compared to the code length N. For L = 0.1 N, a reduction of 0.4 dB in SNR is obtained. If L is large ($L \approx 3N/4$) the code becomes energy inefficient and approaches the staggered pulse train properties[22]. Hence the performance of ternary codes is somewhere between these two extremes, depending on the number of zeros, L.

The question which arises now is how much can the performance of a code be improved by allowing the sequence elements to take on values $\pm 1, 0$? For an estimation of the r.m.s. sidelobe level consider a random sequence c(n) whose elements can assume the values -1, 0, +1, with probabilities p(-1), p(0), p(1). It is reasonable to assume a symmetric probability distribution, that is

$$p(-1) = p(1) = w$$

and

$$p(0) = 1-2w$$

The ACF is given by

$$r(k) = \sum_{n=o}^{N-1-k} c(n)\, c(n+k) \qquad (6.46)$$

The elements c(n) are independent random variables. Making the substitution $N_k = N-k$, Eq. (6.46) can be written as

$$r(k) = \sum_{n=o}^{N_k-1} q(n) \qquad (6.47)$$

where $q(n) = c(n)\, c(n+k)$ is a random variable taking on values -1, 0, 1, with probabilities w, 1-2w, w. Each q(n) is independent ($k \neq 0$) with zero mean and variance $\sigma^2 = 2w$. The probability distribution of the sum of two random variables is the convolution of their individual distribution. Having N_k terms in (6.47) this operation has to be performed N_k times. For N_k reasonably large the central limit theorem applies[44]. The distribution will be Gaussian with zero mean and variance σ_k^2, where σ_k^2 is simply the sum of the variances σ^2, that is

$$\sigma_k^2 = N_k \sigma^2 = 2wN_k$$

$$= 2w(N-k)$$

and $$P_r(r(k)) \simeq (2/\pi \sigma_o^2)^{\frac{1}{2}} e^{-r(k)^2/2\sigma_k^2} \qquad (6.48)$$

where $$\sigma_o = (2wN)^{\frac{1}{2}}$$

The r.m.s. value of the kth sidelobe is

$$\sigma_k = \sigma_o (1 - k/N)^{\frac{1}{2}}$$

It is clear that the r.m.s. value decreases with distance from the main peak.

$$\sigma_k \simeq \sigma_o \quad \text{for k small}$$

$$\sigma_k \simeq \sqrt{2w} \quad \text{for k large} \qquad (k < (N-1)) \qquad (6.49)$$

For the strictly bipolar case $w = 1/2$ and $\sigma_o = \sqrt{N}$, which is the familiar result. To obtain small r.m.s. sidelobe values Eq. (6.48) indicates that w should be small. However, the smaller w the greater the loss in transmitted signal energy. For $w = 1/4$ the decrease in SNR is expected to be about 3 dB. The r.m.s. sidelobe on the other hand will be reduced to $0.7 \sqrt{N}$. This is not so significantly better in comparison to $\sqrt{N}$ for the strictly binary case and in general to obtain low sidelobes one must be prepared to introduce more than N/2 zeros. However, it is shown that with a proper choice of the zero positions some improvement in peak sidelobe is obtained with little loss in energy performance.

The problem is to find the optimum zero positions which yield the maximum reduction in sidelobe energy and sidelobe level. The search procedure can be carried out in a similar manner as described in Section 6.4.1 by a simple modification of the basic element complementation method.

A subset of the results is given in Table 6.11 for a relatively small number of zeros. In many cases the peak sidelobe is reduced by three units while suffering a loss in SNR of less than 1 dB. For further improvements more zeros have to be introduced at the expense of the energy performance of the code.

An interesting feature of ternary codes is their use in a multiplex pulse-compression system[82]. If one of the zero elements of a ternary code, C, is changed to +1 or -1, resulting in the sequences C_+ and C_- respectively, it can easily be seen from Eq. (6.46) that the coherent summation of their individual ACF's is given by

$$r_C(k) = \tfrac{1}{2}\{r_{C_+}(k) + r_{C_-}(k)\} \tag{6.50}$$

For a ternary code with L zeros, 2^L sequences with this property can be found.

Another technique to reduce the time sidelobes of a finite-length binary sequence has been reported by Golay[83]. Golay has found binary sequences of length 2^n which have the property that the sum of their individual ACF is zero, except at zero time shift. Thus, if two complementary codes can be transmitted simultaneously and their matched output added vectorially (for zero doppler shift), there will be no residues.

In practice, however, a problem exists with this approach in a clutter environment. If the two transmitted codes are separated in

Code length	Binary sequence		Ternary sequence			
N	$\max_k \lvert r(k) \rvert$	E_s(%)	L	$\max_k \lvert r(k) \rvert$	E_s(%)	Loss (dB)
19	3	11.36	1	2	7.72	-0.24
23	3	11.15	3	2	6.00	-0.61
31	4	16.55	2	3	10.46	-0.29
37	5	11.25	2	3	8.25	-0.24
41	6	16.66	5	3	9.57	-0.57
43	5	16.28	6	3	6.94	-0.77
47	5	14.26	8	3	3.94	-0.81
53	5	12.32	4	4	9.00	-0.34
59	8	16.23	8	4	11.34	-0.63
61	7	13.81	6	4	8.49	-0.45
63	7	8.64	2	5	9.03	-0.14
67	6	14.44	3	5	12.65	-0.20
101	9	15.82	9	6	11.32	-0.41
103	9	15.45	10	6	9.74	-0.44
105	9	13.68	10	5	9.14	-0.44
107	9	12.27	7	6	10.20	-0.29
119	9	13.80	7	7	10.01	-0.30
121	10	13.82	6	7	12.26	-0.26
125	7	10.25	5	6	6.67	-0.25
251	13	10.10	9	11	8.64	-0.29

Table 6.11 Performance of ternary sequences
(L denotes number of zero elements).

frequency, there will be decorrelation of the clutter echoes and little cancellation. If the codes are separated in time and the clutter is extensive, the sidelobes will be temporally or spatially decorrelated. Therefore, the primary use of this technique would be to prevent the time sidelobes of a large target (or large point clutter) from obscuring a smaller target. An interesting application of complementary codes is in long-range demonstration radars (LRDR) where the objective is to detect very slowly moving targets such as cars, humans, enemy soldiers, etc. in large amounts of ground clutter[31].

This class of signals and their relationship to other types of codes have been further investigated by several authors[84,85]. In particular Welty[84] described a simple recursive method to synthesize a whole family of (2^n-1) complementary sequences, using orthogonal codes. Welty has shown that $\max_k |r(k)|$ of such codes cannot exceed an upper bound given by 2^{n-1}. Another interesting property found while studying these codes was that all complementary sequences of equal length $N = 2^n$ seem to have the same sidelobe energy, that is

$$\sum_{k=1}^{N-1} |r_i(k)|^2 = \text{const.} \qquad i = 0,1,2,\ldots,2^n-1 \tag{6.51}$$

where the subscript, i, indicates the ith complementary code. Unfortunately, no mathematical proof has yet been formulated, but for a large number of codes this has been verified by computer experiments.

Fig. 6.16 illustrates the ACF of a complementary code of length $N = 128$. It is noted that for $(N-k)$ even, $r(k) = 0$. Moreover, the sidelobe energy of this type of sequence was found to be of the order of $0.25\ N^2$.

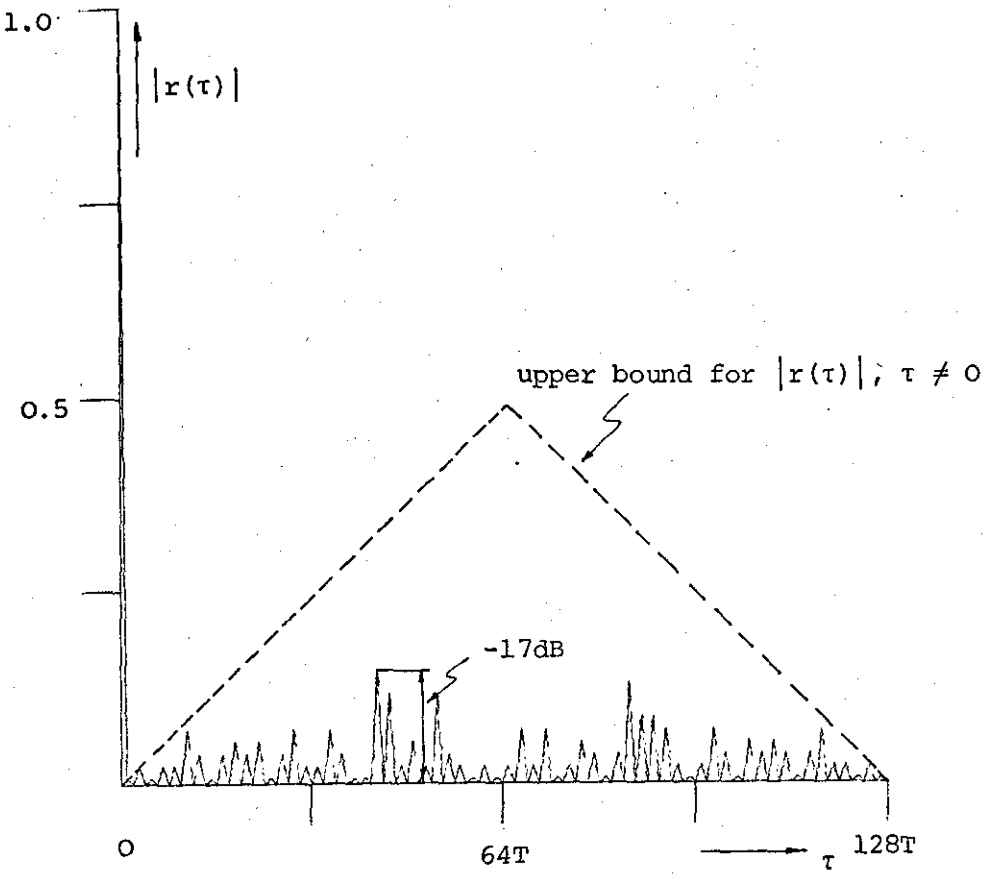


Fig. 6.16 ACF of 128-element complementary code.

This type of sequence could be used as an initial starting point in the element complementation search algorithm. However, if the structure of the code is violated, the number of residues will be equal to $2(N-1) \simeq 2N$. Hence, it is expected that the r.m.s. sidelobe level will decrease by $\sqrt{2}$ to approximately $0.35\sqrt{N}$.

6.5 Uniform Sequences with Low Autocorrelation Sidelobes and Small Crosscorrelation

Much attention has been paid by many authors to the construction of binary sequences having ACF's as small as possible away from the coincidence peak. However, little is known about sequences with small crosscorrelation. Such sequences have many practical applications. For example, they may be used as address codes in a time-division multiple-access (TDMA) system, where information from several data sources is to be transmitted over a channel[86].

It has been shown that for most good binary sequences of length N ($N > 13$), the attainable sidelobe levels are approximately $\sqrt{N}$. The mutual crosscorrelation peaks, however, of sequences of the same length tend to be much larger and are usually in the order of $2\sqrt{N}$ to $3\sqrt{N}$. Consequently, the objective in this section is to find pairs of binary (or complex) sequences of length N with autocorrelation sidelobes and crosscorrelation peak values both of approximately $\sqrt{N}$.

6.5.1 Statement of the Problem

The more complicated problem of finding a pair of uniform pulse-compression codes which besides having small sidelobes also have small crosscorrelation, can be approached using optimization techniques[87]. The ACF and crosscorrelation of two arbitrary sequences of length N are given by

$$r_1(k) = \sum_{n=o}^{N-1-|k|} a(n)a^*(n+k) \tag{6.52}$$

$$r_2(k) = \sum_{n=o}^{N-1-|k|} b(n)b^*(n+k) \tag{6.53}$$

$$r_{12}(k) = \sum_{n=o}^{N-1-|k|} a(n)b^*(n+k) \tag{6.54}$$

where $k = 0, \pm 1, \pm 2, \ldots, \pm(N-1)$

In the binary case, sequences which result in the best possible autocorrelation and crosscorrelation must satisfy the conditions

$$\underset{k \neq o}{r_1(k)},\ \underset{k \neq o}{r_2(k)},\ r_{12}(k) = \begin{cases} 0 & ;\ N-k \text{ is even} \\ \pm 1 & ;\ N-k \text{ is odd} \end{cases} \tag{6.55}$$

The above conditions form a system of 2(N-1) non-linear equations in the unknowns a(1), a(2),....,a(N-1) and b(1), b(2),....,b(N-1) (one of the unknowns of each sequence can be chosen arbitrarily, i.e., a(0) = b(0) = 1). It will be shown later that it is impossible to satisfy (6.55) simultaneously, except for the two trivial cases N = 1,2. Again an approximate solution to this set of equations is sought using numerical methods.

As mentioned previously one of the most important steps in the optimization of any design or process is the choice of the optimization criterion. For pulse compression sequences the properties of concern are the total sidelobe energy and the peak sidelobe. For a set of sequences, however, an additional important criterion is the peak magnitude and energy of their mutual crosscorrelation function. This may be of concern in satellite communication systems such as TDMA where the problem of unique word synchronization requires sequences which not only have good autocorrelation properties, but also should have as little crosscorrelation as possible[86].

Therefore, what is required is the minimization of a suitable measure which characterizes the 'goodness' of a pair of sequences. Any such measure is to some extent arbitrary but in general will be of the form $F(|r(k)|)$. For any particular pair of sequences the performance index for each individual correlation function is defined as

$$F_1 = \sum_{k=1}^{N-1} |r_1(k)|^4 \tag{6.56}$$

$$F_2 = \sum_{k=1}^{N-1} |r_2(k)|^4 \tag{6.57}$$

and

$$F_{12} = \sum_{k=-(N-1)}^{N-1} |r_{12}(k)|^4 \tag{6.58}$$

Again only one half of the ACF is considered, since it is an even function.

The optimum sequences $a(n)$ and $b(n)$ are determined from the conditions

$$F_1 + F_2 = \min \quad \text{and} \quad F_{12} = \min \tag{6.59}$$

This can be accomplished by minimizing a linear combination of the performance indices

$$\min F = F_1 + F_2 + \lambda F_{12} \tag{6.60}$$

where $\lambda \geqslant 0$ is a weighting parameter. The problem of minimizing F is one of minimizing a function of $2(N-1)$ discrete variables, which can assume only two values ± 1, over a set of $2^{2(N-1)}$ points. In addition a train of values of weighting parameters are required. If λ is large, minimization results in sequences with good crosscorrelation but poor autocorrelation. For small $\lambda (\lambda < 1)$, sequences with optimum ACF's are obtained.

The choice for the best value of λ depends on the specific application as well as the knowledge of the behaviour of the functional F, including the interactions between the performance indices. Obviously, F_1 and F_2 are independent of each other. This is, however, not the case for F_{12}.

6.5.2 Bounds on the Crosscorrelation Energy

When minimizing a functional F of the form given by (6.60) there are certain limits on the possible reduction in sidelobe and crosscorrelation energy and their peak values. The crosscorrelation energy can be estimated by considering a simple but probably not so well known relationship (Appendix D).

$$\sum_{k=-(N-1)}^{N-1} |r_{12}(k)|^2 = \sum_{k=-(N-1)}^{N-1} r_1(k)\ r_2^*(k) \qquad (6.61)$$

This is a remarkable result since it means that the total energy of the crosscorrelation function is equal to the sum of the product of the individual ACF's. For low autocorrelation sidelobes the right side of (6.61) is approximately N^2, hence

$$E_{12} = \sum_{k=-(N-1)}^{N-1} |r_{12}(k)|^2 \approx N^2 \qquad (6.62)$$

Using Eq. (6.61) it is easy to verify that binary sequences satisfying the ideal conditions (6.55) do not exist for $N > 2$.

The total sidelobe and crosscorrelation energy is defined as

$$E_t = \sum_{k=1}^{N-1} \{|r_1(k)|^2 + |r_2(k)|^2\} + \sum_{k=-(N-1)}^{N-1} |r_{12}(k)|^2 \qquad (6.63)$$

$$E_t = E_1 + E_2 + E_{12}$$

A lower bound on E_t is obtained by examining the expression

$$d_1^2 = \sum_{k=-(N-1)}^{N-1} |r_1(k) + r_2(k)|^2$$

expanding d_1^2 and using (6.61) leads to

$$d_1^2 = \sum_{k=-(N-1)}^{N-1} \{|r_1(k)|^2 + |r_2(k)|^2 + 2|r_{12}(k)|^2\}$$

Since $r_1(O) = r_2(O) = N$, one obtains

$$E_t = N^2 + \sum_{k=1}^{N-1} |r_1(k) + r_2(k)|^2 \qquad (6.64)$$

hence, $$E_t \geqslant N^2 \qquad (6.65)$$

Similarly, an upper bound for E_t and E_{12} can be derived by expanding

$$d_2^2 = \sum_{k=-(N-1)}^{N-1} |r_1(k) - r_2(k)|^2$$

which leads to

$$E_{12} \leqslant N^2 + E_1 + E_2 \qquad (6.66)$$

Thus, $$E_t \leqslant N^2 + 2(E_1 + E_2) \qquad (6.67)$$

It can be shown using Eq. (6.64) that the minimum value of E_t is obtained for complementary sequences. Hence,

$$E_t = N^2$$

and since $$E_1 = E_2$$

$$E_{12} = N^2 - 2E_1$$

It has been shown that the r.m.s. sidelobe levels of good binary sequences are about $0.4\sqrt{N}$, i.e. $E_1 = E_2 \simeq 0.15N^2$. Using Eq. (6.62) and substituting these values into Eq. (6.63), it is to be expected that E_t will be a reasonably constant quantity of approximately

$$E_t \approx 1.3\ N^2 \tag{6.68}$$

and for uniform complex codes

$$E_t \approx 1.06\ N^2 \tag{6.69}$$

Hence, minimization of the performance index F results merely in a redistribution of the total energy E_t. This indicates that improved crosscorrelation properties are obtained only at the expense of an increase in autocorrelation sidelobe level. For a given form of the functional F the energy distribution will depend on the weighting parameter λ. For many applications it is desirable to distribute the energy E_t evenly between the individual correlation functions. Hence,

$$E_1 + E_2 \approx E_{12} \tag{6.70}$$

where $E_1 = E_2 \approx 0.3\ N^2$, and with respect to Eq. (6.68), $E_{12} \approx 0.7\ N^2$. For complex codes these quantities are; $E_1 = E_2 \approx 0.25\ N^2$ and $E_{12} \approx 0.56\ N^2$. Such an energy distribution can be achieved by minimizing an equally weighted performance criterion of the form

$$F = \sum_{k=1}^{N-1} \{|r_1(k)|^4 + |r_2(k)|^4\} + \sum_{k=-(N-1)}^{N-1} |r_{12}(k)|^4 \tag{6.71}$$

6.5.3 <u>An Estimate of a Bound for the Minimum Peak Crosscorrelation Value</u>

The crosscorrelation for the evenly distributed binary case is $0.7\ N^2$. To minimize the $\max_k |r_{12}(k)|$ the best solution would be for all the values of $r_{12}(k)$ to be of equal magnitude, say, m. Then

$$m^2 = E_{12}/(2N-1) \tag{6.72}$$

However, the magnitudes of $r_{12}(k)$ cannot all be the same and will be distributed at integer values between a minimum and maximum peak value

($\pm m$), in some manner. By assuming different probability distributions for the values of $r_{12}(k)$, bounds for the maximum value of m can be established[88]. Assuming a uniform distribution* $r_{12}(k)$ will have equal probability $P_r(r_{12}(k))$ of having any level between $+m$ and $-m$, where

$$P_r\left[r_{12}(k)\right] = 1/(2m+1) \tag{6.73}$$

Hence,

$$E_{12} = \sum_{k=-(N-1)}^{N-1} r_{12}^2(k) = (2N-1) \sum_{r_{12}(k)=-m}^{m} r_{12}^2(k)P_r\left[r_{12}(k)\right]$$

$$= 2(2N-1)/(2M+1) \sum_{r_{12}(k)=o}^{m} r_{12}^2(k)$$

$$E_{12} = (2N-1)(m+1)m/3 \simeq 2Nm^2/3$$

thus

$$m \simeq (3E_{12}/2N)^{\frac{1}{2}} \tag{6.74}$$

Substituting E_{12} from Eq.(6.70) an approximate estimate of $\min(\max_k|r_{12}(k)|)$ is obtained

$$m \simeq \sqrt{N} \tag{6.75}$$

For complex codes, however, it should be possible to approach the bound (6.72), since these codes have a much wider degree of coding freedom.

*Strictly speaking the crosscorrelation values $r_{12}(k)$ tend to have a normal distribution for very large sequences. However, a uniform probability distribution may be an adequate assumption to obtain a first order estimate of $\min(\max_k|r_{12}(k)|)$.

6.5.4 Results of the Synthesis

The element complementation and pattern search methods were used to minimize F in the binary and complex case respectively. (It should be noted that the number of variables are now doubled). There are, however, a number of ways of applying the optimization procedure. One approach is first to change all the elements of sequence a(n), keeping b(n) fixed, that is minimization of

$$F = C_b + \sum_{k=1}^{N-1} |r_1(k)|^4 + \sum_{k=-(N-1)}^{N-1} |r_{12}(k)|^4 \qquad (6.76)$$

is carried out, where

$$C_b = \sum_{k=1}^{N-1} |r_2(k)|^4$$

is now a constant term. The next step is to repeat the process for sequence b(n) while a(n) remain unchanged, i.e.

$$F = C_a + \sum_{1}^{N-1} |r_2(k)|^4 + \sum_{-(N-1)}^{N-1} |r_{12}(k)|^4 \qquad (6.77)$$

This is done iteratively until the (local) minimum is reached.

An extensive computer search was carried out using the element complementation technique for $\lambda = 0$ and $\lambda = 1$. The search was started with a randomly chosen pair of binary sequences. Some of the results are summarized in Table 6.12 for various values of N, where e_1, e_2, and e_{12} are the normalized energy ratios given by

$$e_{1,2} = 10^2 \, E_{1,2}/N^2$$

$$e_{12} = 10^2 \, E_{12}/N^2$$

For $\lambda = 0$ on the r.m.s. and peak values of the autocorrelation sidelobes are around $0.4\sqrt{N}$ and $\sqrt{N}$ respectively. These values correspond to those obtained in Section 6.4.1. It should be noted that the crosscorrelation

Sequence length	Peak sidelobe		Peak cross-correlation	Energy ratio (%)		
N	$\max_k \|r_1(k)\|$	$\max_k \|r_2(k)\|$	$\max_k \|r_{12}(k)\|$	e_1	e_2	e_{12}
$\lambda = 0$						
19	3	3	8	11.36	11.36	104.99
23	3	3	9	11.15	11.91	88.28
31	4	4	10	16.15	10.72	97.29
37	5	4	10	11.25	11.83	97.37
41	5	5	13	15.94	11.66	97.14
43	6	5	13	18.88	12.17	108.11
53	7	6	24	11.32	15.88	104.41
59	8	6	18	16.23	14.62	102.82
61	6	7	18	10.37	12.63	108.60
63	7	6	20	8.64	13.98	104.69
67	7	6	18	19.00	11.96	102.27
73	8	7	21	13.66	13.89	96.47
79	8	8	19	16.65	13.51	93.88
91	10	7	18	16.19	16.15	97.08
93	8	8	24	12.28	16.26	100.42
101	11	8	22	12.57	14.49	98.59
103	9	9	24	16.69	13.75	101.64
107	10	9	24	14.79	13.70	97.12
109	10	9	26	14.70	12.44	104.98
$\lambda = 1$						
19	3	4	5	21.33	25.76	67.31
23	5	5	6	24.01	24.76	77.69
31	4	5	7	16.55	24.87	86.47
37	7	7	8	27.03	30.83	70.78
41	7	6	9	26.89	29.74	61.45
43	6	7	9	21.47	21.90	77.61
53	10	9	9	32.54	28.98	67.25
59	10	8	11	25.54	24.27	70.07
61	9	10	10	29.72	28.86	66.57
63	10	9	10	31.12	26.68	70.32
67	9	9	11	25.24	27.47	74.83
73	10	11	11	29.50	27.17	67.95
79	9	9	12	20.56	23.89	76.77
91	10	12	13	24.36	29.09	67.47
93	11	12	13	25.46	25.37	71.28
101	9	12	15	26.72	30.13	71.45
103	9	13	14	16.69	24.46	78.57
107	14	13	15	23.45	29.95	72.21
109	14	12	14	27.59	29.37	72.06

Table 6.12 Correlation properties obtained using numerical optimization.

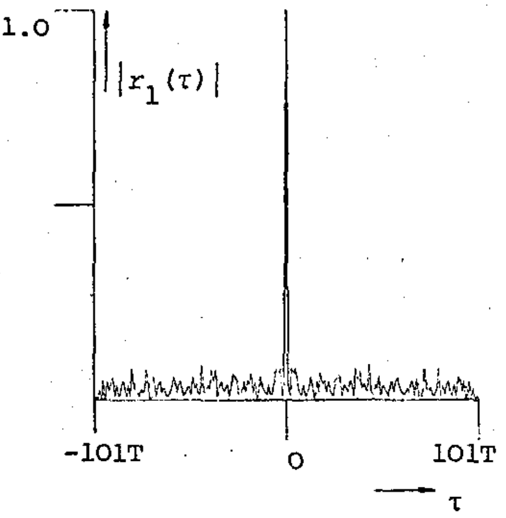
energy for this case is approximately N^2, which confirms the assumption derived from Eq. (6.62). The results in Table 6.12 show quite clearly that sequences having optimum ACF do not give smallest crosscorrelation values and thus appear to be correlated, even though these sequences were obtained using different source logic. In general one would expect peak values about twice the maximum autocorrelation sidelobes since the number of different crosscorrelation levels is ordinarily twice the number of autocorrelation levels. Table 6.12 also shows good agreement with the sought after energy distribution when a measure given by (6.71) is minimized. Moreover, the crosscorrelation peak values have decreased considerably to approximately $1.3\sqrt{N}$ as compared to $2\sqrt{N}$ to $3\sqrt{N}$ for $\lambda = 0$. However, as expected, this improvement is achieved at the expense of an increase of the maximum autocorrelation sidelobes which are, except for small values of N, usually of the same order. The uniform character of the sidelobes and crosscorrelation values reflects the equal weighting of the performance indices. The r.m.s. value of the sidelobes and crosscorrelation is about $0.6\sqrt{N}$, assuming average energies of $0.3\ N^2$ and $0.7\ N^2$ respectively*. In Fig. 6.17 a representative graph of two sequences of length $N = 101$ and the magnitude of their correlation functions is shown. In all cases the minimum crosscorrelation peak values obtained have been found to be greater than the minimum bound estimated using Eq. (6.75) as shown in Fig. 6.19.

The results for the uniform complex codes are given in Table 6.13. Again the autocorrelation and crosscorrelation peak values are of the same order ($\simeq 0.8\sqrt{N}$) and the r.m.s. values are approximately $0.5\sqrt{N}$ to $0.6\sqrt{N}$. It is noted that there is not a great difference between the r.m.s. and peak values which indicates that the residues are very uniform in magnitude. This is illustrated in Fig. 6.18 for a pair of complex

*The r.m.s. value for the one-sides autocorrelation sidelobes is given by $(E_{1,2}/(N-1))^{\frac{1}{2}}$, while for the crosscorrelation it is $(E_{12}/(2N-1))^{\frac{1}{2}}$.

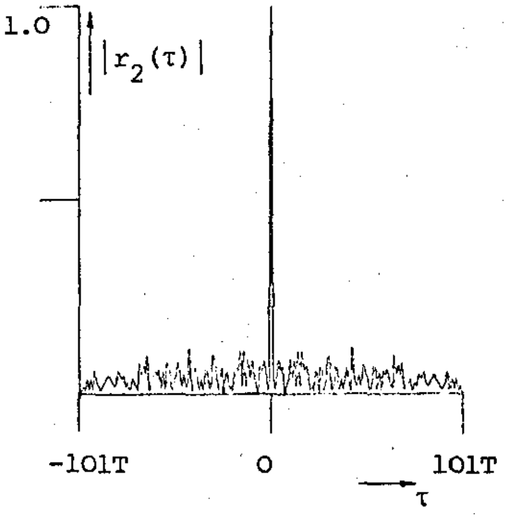
(a) Sequence a(n):

```
+ - - + +   - - + + +
+ + - + +   + - + + -
+ + - + +   - + + - +
- - - - +   - - + - +
- + + + -   - + - + -
+ - + - +   - - - + +
+ + + + +   - + - - -
- + + + +   - - - + +
+ + - - +   + + - - +
+
```



(b) Sequence b(n):

```
+ - - + -   - - + + +
- - - + -   + - - + -
+ + + + -   - + - + -
- + + - +   - + - + +
+ + - + -   - - - - +
+ - + + +   - - + + -
+ + - - -   + - - - +
- - + - +   - + + + +
+ + + + +   + + + - -
+ + - + -   - + + - -
+
```



(c)

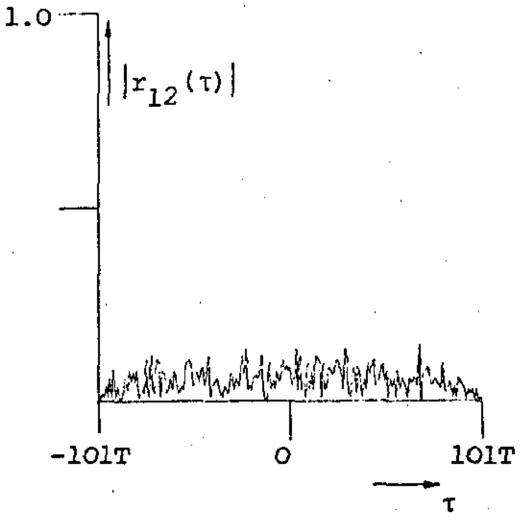


Fig. 6.17 Correlation functions of two binary sequences of length N = 101;

(a) $\max_k |r_1(k)| = -21$ dB

(b) $\max_k |r_2(k)| = -19$ dB

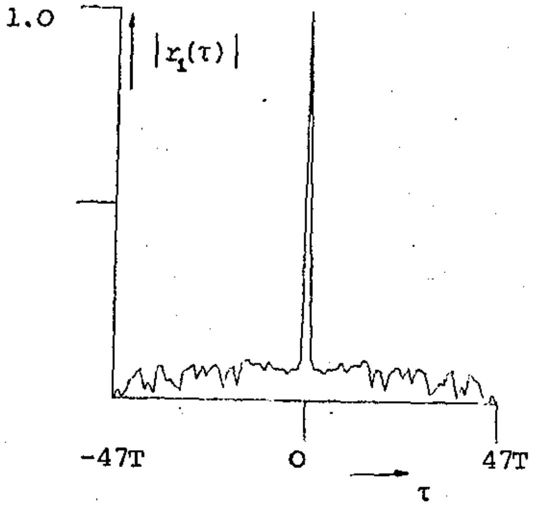
(c) $\max_k |r_{12}(k)| = -18$ dB

Sequence length	Peak sidelobe		Peak cross-correlation	Energy ratio (%)		
N	$\max_k \|r_1(k)\|$	$\max_k \|r_2(k)\|$	$\max_k \|r_{12}(k)\|$	e_1	e_2	e_3
$\lambda = 1$						
9	1.627	2.327	2.560	18.17	27.76	61.06
11	2.199	2.446	2.418	24.31	28.78	55.78
15	2.865	2.678	2.925	25.49	22.78	57.43
17	2.763	2.952	3.082	21.93	25.67	58.26
19	3.376	3.001	3.385	21.74	22.75	61.69
20	3.120	3.251	3.381	24.12	23.66	57.01
21	3.086	3.481	3.375	23.91	23.60	58.69
23	3.300	3.472	3.535	24.14	22.19	58.68
25	3.280	3.925	3.869	23.19	25.47	58.34
27	3.441	3.498	4.047	22.08	23.88	60.66
29	3.725	3.611	4.039	24.58	22.64	57.28
30	3.893	3.804	4.008	23.17	23.17	60.71
31	3.580	3.612	4.367	19.59	22.70	62.92
34	3.953	4.409	4.477	23.60	25.08	56.56
35	4.531	4.480	5.146	22.97	23.51	62.07
41	4.565	4.873	5.125	20.40	26.12	59.50
43	4.732	4.900	5.814	21.56	24.05	61.29
47	5.388	7.821	6.652	22.20	33.11	59.03

Table 6.13 Optimum correlation functions obtained for uniform complex codes.

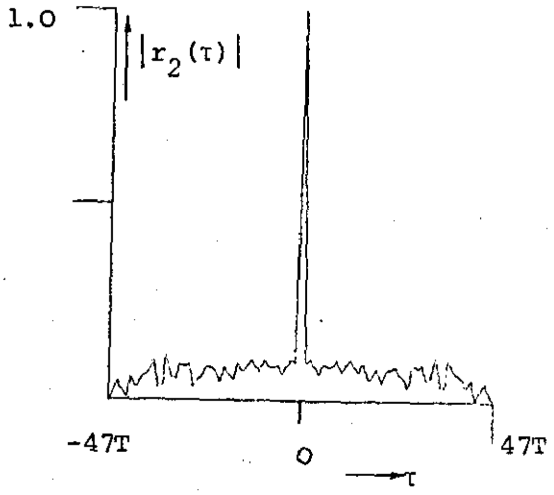
(a) Sequence a(n), (radians):

-2.094	1.682	-2059	0.650	2.237
0.763	-1.325	-1.737	0.231	-2.841
-2.724	-0.735	2.973	1.650	2.698
-0.101	-1.948	2.596	2.406	-2.052
0.150	1.779	-1.885	-2.823	-0.141
-2.170	-1.097	1.179	0.028	-2.907
-2.949	2.012	-2.273	-1.338	-2.719
-2.705	0.504	-2.073	1.305	0.801
-3.133	-2.797	1.158	-2.220	-1.209
-1.804	-2.619			



(b) Sequence b(n), (radians):

-1.977	-2.472	2.297	-2.862	2.322
-3.020	2.395	-1.753	-2.304	2.029
2.910	-1.287	1.734	-2.413	-1.271
0.841	0.938	0.183	0.448	0.241
-0.521	0.473	0.415	2.749	0.665
-2.124	1.076	-2.937	3.094	-1.484
0.347	-0.111	1.022	3.099	2.600
-1.965	0.679	0.773	0.089	2.691
-1.22.	2.982	-2.440	-2.033	2.708
1.627	-0.105			



(c)

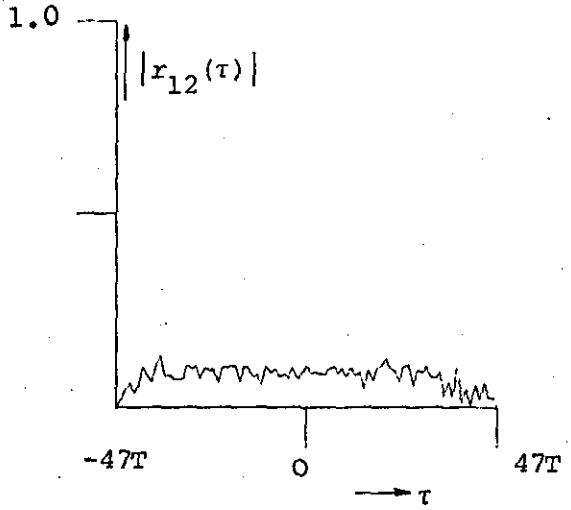


Fig. 6.18 Correlation functions of two uniform complex codes of length N = 47;

(a) $\max_k |r_1(k)| = -19\text{dB}$

(b) $\max_k |r_2(k)| = -16\text{dB}$

(c) $\max_k |r_{12}(k)| = -17\text{dB}$

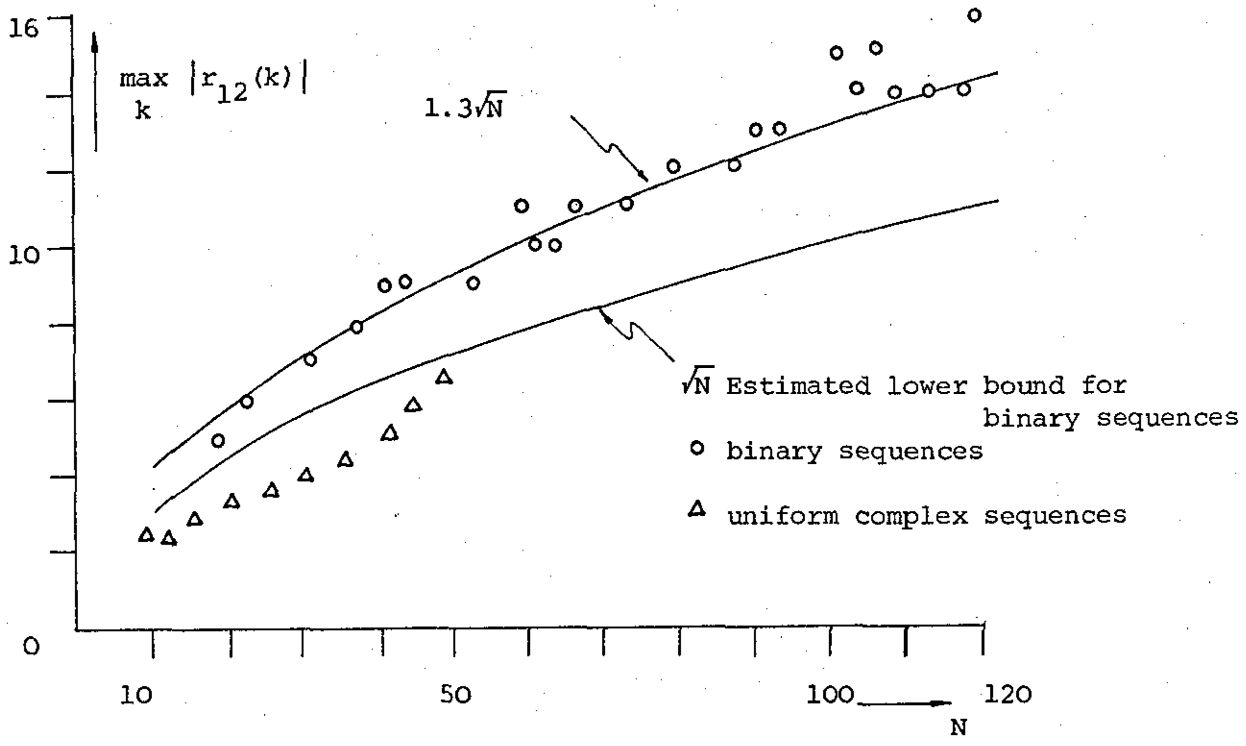


Fig. 6.19 Crosscorrelation peak value as a function of N.

sequences of length N = 4 7.

It was observed that the computer program converged in most cases in one or two iterations (six to ten in the complex case) and optimum binary sequences were found for N as large as 250 in less than 20 min using an ICL 1904A digital computer.

It has been shown that binary sequences whose largest autocorrelation sidelobes and crosscorrelation do not exceed unity do not exist for length $N > 2$. It can easily be verified that this also holds for uniform complex codes for $N > 3$. However, adopting a numerical optimization technique it is possible to find sequences which are satisfactory for most practical applications. With a proper choice of the λ-weighting parameter, a significant improvement of the crosscorrelation peak value can be achieved at the expense of only a relatively small increase in peak sidelobe level. Sequences with near uniform residues of approximately $1.3\sqrt{N}$ in the binary and $0.8\sqrt{N}$ in the complex case have been obtained. It has been shown that minimization merely results in a redistribution of the energies contained in both the autocorrelation and crosscorrelation functions. Moreover, assuming a uniform probability distribution of the crosscorrelation values a lower bound on the peak value has been estimated.

The synthesis method presented here has been restricted to pairs of uniform sequences. A more general problem would, therefore, aim at finding not just pairs, but finite sets of binary or complex codes having good autocorrelation and crosscorrelation properties.

6.6 Summary

In this chapter the application of numerical methods to the design of phase coded pulse trains has been investigated. The objective was to study the range resolution and clutter rejection performance of these discrete coded waveforms. Sequences which optimize the correlation properties, defined by suitable cost functionals, were considered. Without prior information about the radar environment the choice of such a measure of 'best' is to some extent arbitrary. It has been shown that minimizing an ℓ_4-measure of the residues has the desirable effect of reducing the peak sidelobes as well as the sidelobe energy. However, such a measure is highly non-linear and multimodal and at present there is no criterion to tell whether the obtained extremum is a global minimum.

It is difficult if not impossible to predict how various numerical algorithms perform when applied to this type of objective function. While there are a number of efficient optimization techniques for problems with small dimensionality ($N < 15$) not many seem to be able to handle functions with a large number of variables ($N > 50$). A study of the efficiency of several different algorithms has shown that pattern search techniques are particularly suitable for this type of application.

It was found that uniform complex codes with largest sidelobe not exceeding unity (Barker code property) exist up to a length of at least $N = 18$. Even for a much larger length, $N \simeq 100$, it is possible to find sequences with peak sidelobes less than two.

The design of binary sequences was approached by transforming the synthesis problem into a sequence of unconstrained minimization problems using penalty-function techniques. This method of handling discrete variable problems looks very encouraging in the light of the results obtained. Sequences whose sidelobe performance is equal or only marginally inferior than that of the best known binary codes were found.

A comparison of various phase coded waveforms is illustrated in Fig. 6.20.

An improved penalty-function method has been described which has certain advantages such as less sensitivity to parameter variations during the search process as well as direct control of the rate of convergence. The latter is important for large scale problems. This method has also been found useful in designing poly-phase coded sequences.

Finally, pairs of phase coded sequences having low autocorrelation sidelobes and small mutual crosscorrelation have been designed. These sequences are particuarly useful in spread spectrum multiple-access systems. It has been shown, however, that improved crosscorrelation properties can only be obtained at the expense of an increase in the autocorrelation sidelobes.

Although numerical methods have proved to be successful they are not without their weaknesses. From the theoretical point of view the most serious objection is that the results are not unique. Thus for a given design objective, the sequence to which the procedure converges is dependent on the initial choice of the starting sequence. In some cases a vector with a quadratic phase (QP) could be a good initial choice, whilst in others the uncoded vector may serve as a suitable starting point. Thus, for each individual application the initial vector would have to be chosen judiciously. Furthermore, when very large pulse trains with hundreds or thousands of pulses are required difficulties of a computational nature arise. These difficulties can be overcome to some extent by trying to generate longer sequences by combining short ones[15]. Unfortunately, the ACF's of good short sequences do not reveal any pattern which would suggest a plausible rule for their construction.

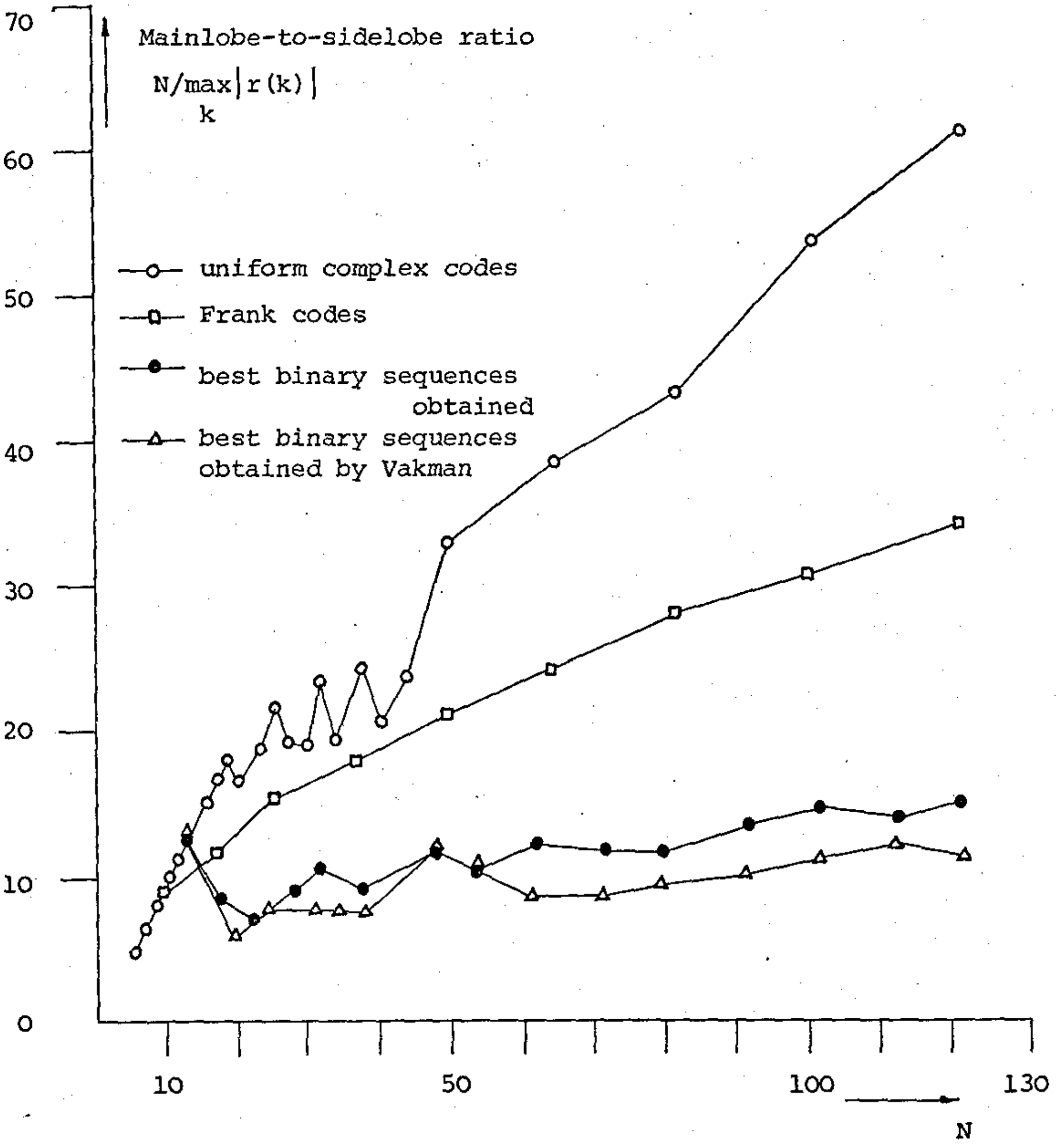


Fig. 6.20 Comparison of sidelobe performance of various phase coded sequences.

Although various techniques to obtain combination codes exist[15]*, no* one has yet found a construction method which diminishes $\max\limits_{k}|r(k)|$. Other problems such as the number of phase quantization levels necessary to achieve the desired performance of uniform complex codes and the possibility of real-time adaption must be considered for each case individually.

Optimization, which forms an integral part of system design, is made possible by proper use of algorithms and computers. Keeping Hamming's motto[89] that 'the purpose of computing is insight not numbers' in mind, the problem should be formulated and an optimization method selected in such a manner that the results will give information beyond mere numerical values of the best attainable design vector.

CHAPTER 7

AMPLITUDE AND PHASE MODULATED PULSE TRAINS

7.1 Introduction

Most modern high performance radars use travelling-wave tube amplifiers to obtain coherent transmission. As pointed out previously these tubes work most efficiently under constant amplitude conditions. Moreover, good amplitude modulation (AM) is difficult (and very expensive) to achieve with these devices. Therefore only purely phase modulated pulse trains have been considered so far. However, with the emergence of solid-state microwave sources efforts are being made to replace the relatively large and expensive vacuum devices by low power solid-state elements and the waveguide elements by planar circuits. With these new components the size and costs are reduced and a number of commercial applications become feasible.

The contribution of solid-state devices has also been significant in areas where performance was previously inadequate. For example, it is much easier to use any form of modulation with solid-state components. Although AM is not an efficient method to provide the large time-bandwidth required for good radar performance, it does provide a means of improving the resolution capability. Consequently, this chapter treats the problem of finding energy efficient amplitude and phase modulated (a.m.ph.m) pulse trains.

7.2 Huffman Sequences

Huffman[20] has shown that it is possible to derive sequences a(n) of any arbitrary length (N+1) whose ACF's have the property

$$r(k) = \sum_{n=o}^{N-|k|} a(n)\, a^*(n+k) = \begin{cases} E & ; \quad k = o \\ 0 & ; \quad |k| \neq 0,N \\ r(N) & ; \quad |k| = N \end{cases} \tag{7.1}$$

where E is the energy of the sequence.

This ACF is zero for all time shifts except for the unavoidable end sidelobes. Without loss of generality the end sidelobes r(N) can be set to unity. Using the familiar ZT notation, the ACF of a Huffman code reduces to

$$\begin{aligned} R(z) &= z^{-N} A(z)\, A^*(1/z) \\ &= 1 + E\, z^{-N} + z^{-2N} \end{aligned} \tag{7.2}$$

where $A(z) = \sum_{n=o}^{N} a(n)\, z^{-n} = a(o) \prod_{i=1}^{N} (1 - z^{-1} z_i)$

For a pulse train A(z) to have the property (7.1) Huffman[20] showed that the roots z_i of A(z) must lie at equal angular intervals ($2\pi/N$) in the complex z-plane on either of two origin centered circles whose radii are given by

$$\begin{aligned} X &= (E/2 + [(E/2)^2 - 1]^{\frac{1}{2}})^{1/N} \\ 1/X &= (E/2 - [(E/2)^2 - 1]^{\frac{1}{2}})^{1/N} \end{aligned} \tag{7.3}$$

Since the polynomial A(z) has N roots, there are 2^N possible root patterns and thus 2^N Huffman codes with the same ACF that can be derived for a given energy E and sequence length (N+1). Although some inefficiency in the use of transmitter power may be acceptable in order to obtain the impulse-like property (7.1), it would certainly be wasteful not to seek the most efficient sequence for an application. A figure of merit for the energy distribution of a sequence is the energy ratio defined by

$$\rho = E/\{(N + 1) \max_{n} |a(n)|^2\} \leqslant 1 \tag{7.4}$$

The maximum value of ρ is attained for purely phase modulated pulse trains such as binary sequences. The energy ratio depends on two independent variables; namely, the total energy E of the coded waveform and the magnitude of the largest coefficient of A(z), denoted by

$\max_n |a(n)|$. By referring to Eq. (7.3) it can be easily verified that the energy E is given by

$$E = X^N + X^{-N} \tag{7.5}$$

and thus is a function of the radius X only. The maximum amplitude $\max_n |a(n)|$, however, depends on the particular choice of the N zeros for A(z) as well as the radius. (A general expression for the coefficients of a polynomial in terms of its zeros is given by Eq. (7.11)). Unfortunately, a mathematical method has not been found which leads, without trial and error, to the most efficient Huffman code.

The design of a Huffman sequence in general requires the choice of the code length (N+1), the circle radius X and the zero pattern for which the magnitude sequence $|a(n)|$ is most uniformly distributed. This, however, remains an unsolved problem. Direct evaluation of all 2^N possible root patterns, for a given radius X and N, is not feasible if N is large (N > 20). This remains so even if account is taken of the zero patterns formed from others by rotation in the z-plane through an angle ϕ or by other transformations for which the energy ratio is invariant[90]. A general trial and error procedure is to choose a root pattern, perhaps at random, and to compute the sequence for a succession of values for the radius X. Other methods devoted to this problem have been suggested by Ackroyd[91]. An alternative approach for designing efficient Huffman codes is to use a 'good' but non-Huffman code to suggest a zero pattern for the Huffman sequence. A good code in this context is simply an energy efficient sequence, preferably $\rho = 1$, whose ACF has low sidelobes. For example the uniform complex sequences described in Chapters 4 and 6 are such codes.

The optimum choice of the radius and hence the energy E does not appear to be a great problem since the constraint $|a(0)\ a^*(N)| = 1$ and the desired uniformity of the amplitudes, i.e. $|a(0)| \approx |a(1)| \approx \ldots.$

... ≃ $|a(N)|$ implies $E \simeq (N+1)$. Thus using $E = (N + 1)$ as an initial value, computer search techniques can provide an optimum radius X for any given zero pattern. However, the tolerable peak-to-sidelobe ratio sets a lower limit to the minimum value of X that can be used. Using Eq. (7.2) this ratio is given by

$$E/|r(N)| = (1 + X^{2N})/X^{N} \tag{7.6}$$

A convenient way of representing a zero pattern is by its binary equivalent (b(O), b(1), ..., b(N)) obtained as follows. Starting at an arbitrary point on the unit circle, for example where arg(z) = O, and proceeding counter-clockwise, a root of A(z) is represented by a + 1 if it occurs on the circle with radius X, and by -1, if it occurs on the circle with radius 1/X. For example, the zero pattern of A(z) shown in Fig. 7.1 has the binary representation (-1,1,1,-1,-1,-1,-1,-1, 1,-1,-1,1,1,1,1,1). Thus, except for an arbitrary constant, a Huffman sequence is completely defined by its radius X and the binary sequence b(n). It can be easily verified using Eq. (7.2) that with $|a(O)\ a^*(N)| = 1$ and $|a(N)/a(O)| = |z_1 z_2 \ldots\ldots z_N|$, the relationship between A(z) and the binary representation becomes

$$A(z) = X^{-\frac{1}{2}\sum\limits_{i} b(i)} \prod_{i=1}^{N} (1 - z^{-1}\ X^{b(i)}\ e^{j2\pi i/N}) \tag{7.7}$$

As noted earlier the O,π phase requirement may restrict the ACF of a sequence that could otherwise be achieved. Conversely, it is conjectured that real Huffman codes will not result in the optimum amplitude distribution. Hence, it is to be expected that the zero patterns that lead to complex codes will probably provide superior energy ratios, since the elements of these codes have essentially two degrees of freedom; namely, the amplitude and phase. Fig. 7.1 shows the zero pattern which results from factorization of the corresponding z-transform of the uniform complex code of length N + 1 = 17. This

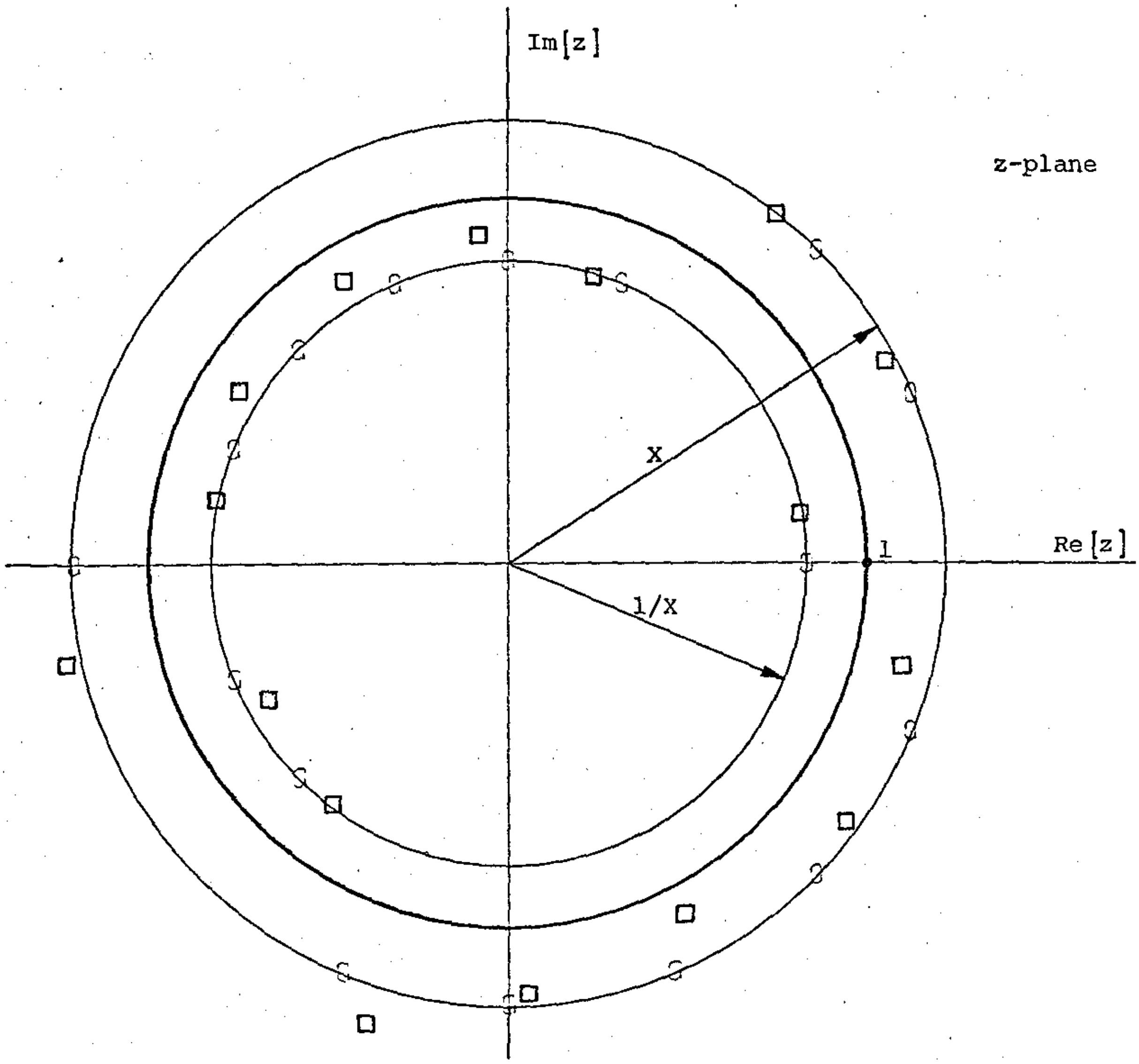


Fig. 7.1 Zero patterns of 17-element uniform complex code (□) and corresponding Huffman code (O).

pattern is similar to a Huffman type pattern in that only relatively small changes in the root positions are required to transform it into a Huffman code zero pattern. Hopefully such a modification will cause little reduction in the energy efficiency of the resulting Huffman code.

A subset of the results using this method is displayed in Table 7.1. (Some energy efficient Huffman codes are listed in Appendix E). Given the zero pattern the codes were compared under two distinct criteria. The first criterion is the energy ratio ρ and the second is a normalized mean-square error defined by

$$\varepsilon = \frac{1}{E} \sum_{n=o}^{N} \{|a(n)| - (E/(N+1))^{\frac{1}{2}}\}^2 \tag{7.8}$$

Although this scheme may be acceptable for designing Huffman codes of moderately large lengths (N+1 ≃ 100), two major difficulties arise for longer sequences. The first difficulty is the computational one of factorizing polynomials of degrees larger than 100*. The second problem is that with increasing sequence length the zero patterns of the uniform complex codes depart quite considerably from that of a Huffman type pattern. It is therefore no longer possible to make only relatively small changes in the zero positions. Consequently other methods for designing efficient Huffman sequences must be sought. As mentioned previously Ackroyd introduced an interesting technique, using the stationary phase principle, which is applicable for an arbitrary length sequence.

Another method which immediately comes to mind is a random selection of the zero pattern. In fact Huffman suggested that in order to maximize the energy ratio ρ, the roots should be chosen in a random

*At present standard root-finding algorithms can handle polynomials of degrees up to 100.

Sequence length	Energy efficiency	Mean-square error	Radius
$N+1$	ρ	ε	x_ρ
5	0.842	0.046	1.420
6	0.584	0.158	1.360
7	0.750	0.052	1.310
8	0.585	0.067	1.290
9	0.942	0.001	1.304
10	0.617	0.048	1.360
11	0.579	0.047	1.325
12	0.509	0.143	1.450
13	0.546	0.098	1.180
14	0.629	0.068	1.200
15	0.681	0.053	1.190
16	0.505	0.074	1.160
17	0.595	0.077	1.160
18	0.552	0.094	1.160
19	0.465	0.125	1.230
20	0.503	0.072	1.140
21	0.433	0.105	1.120
22	0.530	0.099	1.140
23	0.460	0.103	1.153
25	0.606	0.054	1.199
27	0.405	0.129	1.110
31	0.479	0.074	1.101
35	0.323	0.173	1.075
41	0.347	0.102	1.132
45	0.469	0.098	1.110
51	0.378	0.130	1.157
55	0.323	0.173	1.147
61	0.359	0.118	1.116
65	0.368	0.134	1.127
71	0.449	0.149	1.148
75	0.378	0.161	1.140
81	0.376	0.159	1.117
85	0.273	0.381	1.099
91	0.332	0.174	1.046
95	0.178	0.186	1.050
100	0.196	0.180	1.068

Table 7.1 Codes which maximize ρ.

fashion with approximately half the zeros on each circle. He then conjectured further that the optimum energy ratio should be proportional to $(N+1)^{-\frac{1}{2}}$. In general correlation properties are used to judge the randomness of a sequence, that is a sequence is uncorrelated with itself, i.e. random, if its ACF has uniformly low sidelobes. Hence, the optimum binary sequences derived in Chapter 6 could be used as desirable generating patterns. A comparison of the maximum energy ratios obtained for Huffman codes derived from zero patterns of different phase coded sequences is shown in Fig. 7.2. It is evident that for the sequence length given here linear FM type zero patterns (see Fig. 4.19) tend to give more efficient Huffman codes. However, in both cases ρ seems to approach a certain limit ($\rho \simeq 2.3/\sqrt{N}$ for linear FM type zero patterns, and $\rho \simeq 1.8/\sqrt{N}$ for random root selection) as (N+1) increases. Moreover, the error ε tends to approach the same limit. These results clearly support the conjecture made by Huffman. However, to predict any asymptotic behaviour more data would be needed.

7.3 A New Approach to the Signal Design Problem using Parameter Variational Techniques.

Generally speaking the design of pulse trains whose ACF's satisfy the condition (7.1) subject to constraints requires the solution of a set of highly non-linear equations. A closed form colution is only known for Huffman sequences. However, there exists the problem of selecting a zero pattern which results in the most energy efficient pulse train. All known Huffman sequences of length (N+1) > 3 are amplitude and phase modulated and thus to achieve the impulse-like property (7.1) a reduction in power utilization has to be accepted. Even for the optimum solutions (for a given zero pattern) there is a considerable loss in signal energy ($\rho \simeq 2.3/\sqrt{N+1}$). It is believed that there are no uniform Huffman codes (i.e. $|a(n)| = 1$ for all n) for (N+1) > 3, but their nonexistence has

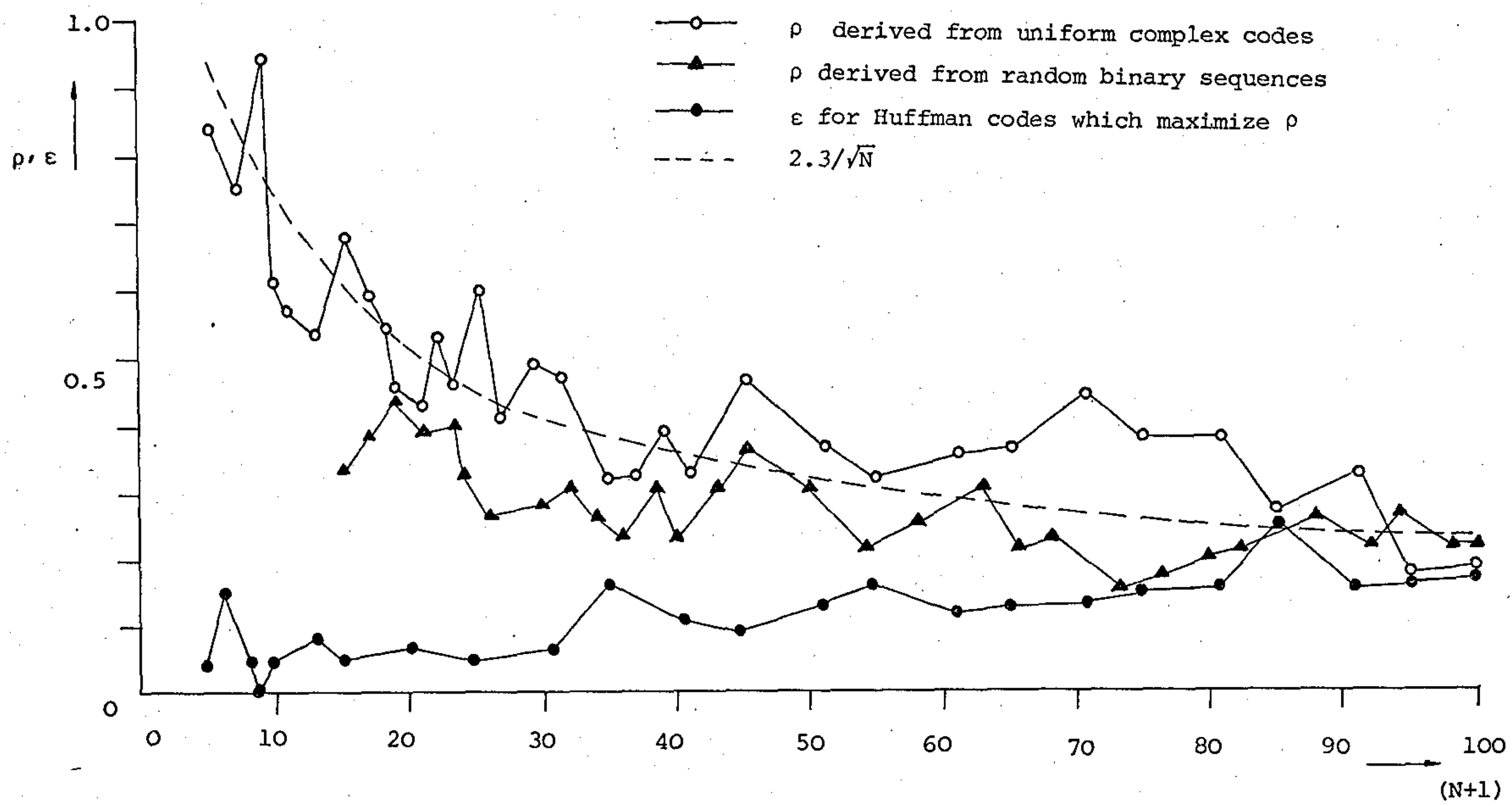


Fig. 7.2 Comparison of Huffman codes derived from different phase coded pulse trains.

not yet been proved mathematically. However, it is possible to state two necessary conditions for uniform Huffman codes. One follows directly from Eq. (7.3) which defines the circle radius

$$X = \{\frac{N+1}{2} + \sqrt{(\frac{N+1}{2})^2 - 1}\}^{1/N} \tag{7.9}$$

since E = N+1. The radius approaches unity as N increases since

$$\lim_{N \to \infty} X = \lim_{N \to \infty} (N+1)^{1/N} = 1 \tag{7.10}$$

In addition uniform Huffman codes, if they exist, must be of odd length. This can be easily verified by considering the relationship between the zeros z_i and the coefficients of the polynomial A(z)

$$\begin{array}{l} z_1 + z_2 + \ldots\ldots + z_N = a(1)/a(0) \\ z_1z_2 + z_1z_3 + \ldots\ldots\ldots + z_{N-1}z_N = a(2)/a(0) \\ \vdots \qquad\qquad\qquad \vdots \\ z_1z_2z_3 \ldots\ldots z_i + \ldots\ldots + z_{N-i+1} \; z_N = (-1)^i \; a(i)*a(0) \\ \vdots \qquad\qquad\qquad \vdots \\ z_1z_2 \ldots\ldots z_N = (-1)^N \; a(N)/a(0) \end{array} \tag{7.11}$$

or in condensed form

$$\sum z_1z_2z_3 \ldots\ldots z_i = (-1)^i \; a(i)/a(0)$$
$$(i = 1,2, \ldots, N-1, N)$$

Since all the zeros are of the form

$$z_i = (X \text{ or } \frac{1}{X}) \exp j(2\pi i/N)$$

and with the conditions $|a(n)| = 1$ for all n, it follows that

$$\left| \prod_{i=1}^{N} z_i \right| = 1 \tag{7.12}$$

Eq. (7.12) can only be satisfied if half of the N roots lie on either circle in the z-plane. Therefore N must be an even number. In other words, if uniform Huffman codes exist they must be of odd length.

The loss in signal energy of Huffman codes is mainly attributed to the fact that all except the end sidelobes are suppressed. In most applications a complete suppression of the residues is not required, provided they can be kept to a specified low level. Therefore, if the condition (7.1) can be relaxed it is expected that there might be a substantial increase in pulse train energy efficiency.

The procedure for synthesizing pulse trains with a specified (realizable) ACF has been treated in Chapter 3. However, even if it is possible to find a suitable factorization of the ACF, it seems impossible in practice to determine whether the resulting polynomial will have all its coefficients of unit magnitude.

In Chapter 6 an attempt has been made to solve the system of equations (7.1) subject to amplitude constraints by minimizing a ℓ_p-measure of the response sidelobes. Because of the highly multi-modal character of the objective function (constraints introduce additional extrema, see for example Fig. 6.9), the effectiveness of such an approach depends largely on the initial estimate of the solution. In the search for a solution of the constrained problem it is therefore desirable to develop a method which is to some extent independent of the need for a 'good' starting point. Consequently, a new approach to the signal design problem, using parameter variational techniques, is considered. The method is based on the idea described by Freudenstein and Roth[92].

7.3.1 The Method in General

The problem considered here is to find a solution to the set of non-linear equations

$$f_j(\underline{x}) = 0 \qquad j = 1,2,\ldots\ldots,n \tag{7.13}$$

where $\underline{x}$ is a n-dimensional column vector of the independent variables and the functions $f_j(\underline{x})$ are of the form

$$f_j(\underline{x}) = \sum_{i=1}^{m} \alpha_{ji}\, v_{ji}\ (\underline{x}) \tag{7.14}$$

where the α_{ji}'s are parameters.

Introducing a set of parameters β_{ji} a new system of equations, $y_j(\underline{x})$ belonging to the same family, can be derived from (7.14) such that

$$y_j^{(O)}(\underline{x}) = \sum_{i=1}^{m} \beta_{ji}^{(O)}\ v_{ji}(\underline{x})\ ,\ j = 1,2,\ldots,n \tag{7.15}$$

where in general $\beta_{ji}^{(O)} \neq \alpha_{ji}$

With a suitable choice of the parameters in conjunction with a particular set of values $\beta_{ji}^{(O)}$ say, it is possible to find analytically one or more solutions of (7.15). Therefore, these derived equations may be any set, belonging to the same class, with at least one known solution, denoted by $\underline{x}^{(O)}$. The Eq.'s (7.15) are now deformed into the Eq.'s (7.14) in a finite number of small increments in the parameters β_{ji} such that

$$y_j^{(k)}(\underline{x}) = \sum_{i=1}^{m} \beta_{ji}^{(k)}\ v_{ji}(\underline{x})\ ,\ k = 1,2,3,\ldots,M \tag{7.16}$$

with

$$y_j^{(M)}(\underline{x}) = f_j(\underline{x})$$

and

$$\beta_{ji}^{(k)} = \beta_{ji}^{(O)} + (\alpha_{ji} - \beta_{ji}^{(O)})\ k/M$$

As the β_{ji} take on different values, the zeros of $y_j^{(k)}(\underline{x}) = 0$ will move in $\underline{x}$-space. Differentiating (7.15) with respect to one of the parameters, β_{ji}, the above deformation may be described in matrix notation as

$$J\ \partial/\partial\ \beta_{ji}(\underline{x}) + \partial/\partial\ \beta_{ji}(\underline{y}) = 0 \qquad j = 1,2,\ldots,n \quad i = 1,2,\ldots,m \tag{7.17}$$

where J is the Jacobian matrix (Chapter 5) with the elements $J_{rs} = \partial y_r(\underline{x})/\partial x_s$, and $\underline{y}^T$ denotes the row vector $(y_1(\underline{x}), y_2(\underline{x}), \ldots. y_n(\underline{x}))$. Solving Eq. (7.17) for $\partial/\partial\beta_{ji}(\underline{x})$

$$\partial/\partial\beta_{ji}(\underline{x}) = -J^{-1}\ \partial/\partial\beta_{ji}(\underline{y}) \tag{7.18}$$

Hence, it can be seen that the task of tracing $\underline{x}$ from $\underline{x}^{(O)}$ to $\underline{x}^{(M)}$ is equivalent to the solution of (nxm) first order differential equations in the interval $\beta_{ji}^{(O)} \leqslant \beta_{ji} \leqslant \beta_{ji}^{(M)}$; $j = 1,2,\ldots,n$; $i = 1,2,\ldots,m$ and subject to the initial conditions $\underline{x}^{(O)}$ and $\beta_{ji}^{(O)}$.

The straightforward way to obtain the desired solution is the approach suggested by Eq. (7.18). Alternatively a method which utlizes any of the available search techniques (see Chapter 5) is by solving the M sets of equations as follows. The known solution $y_j^{(O)}(\underline{x}) = 0$ is used as an initial estimate for the iterative solution of $y_j^{(1)}(\underline{x}) = 0$. This solution in turn is then used as an initial approximation of the root of $y_j^{(2)}(\underline{x}) = 0$, and so forth, until the solution $y_j^{(M)}(\underline{x}) = f_j(\underline{x})$ is obtained. Hence, simultaneous tracing of $\underline{x}$, such that $y_j^{(k)}(\underline{x}) = 0$ is always satisfied will, subject to conditions discussed below, yield a zero of (7.13). In other words, instead of solving the original set of Eq.'s (7.14), one starts with a different set of equations (same function of $\underline{x}$ but different parameters), whose solution is known. A succession of searches is then performed, each time incrementing the parameters, until the correct equations and their solution is found.

The obvious choice in tracing a zero from $y_j^{(k)}(\underline{x}) = 0$ to $y_j^{(k+1)}(\underline{x}) = 0$ is the generalized Newton-Raphson iteration (see Chapter 5).

$$\Delta x_s' = - \sum_{j=1}^{n} J_{sj}^{-1} y_j(\underline{x}) \qquad s = 1,2,\ldots,n \tag{7.19}$$

Thus at each stage two different increments are performed. One is the usual change in the independent variables $\underline{x}$ and the other is the change in the parameters β_{ji}.

The condition for convergence is obviously $|J| \neq 0$ (no singularities) and, moreover, convergence at each step, i.e. the deformation step size must be suited to the radius of convergence of the iteration technique employed. Furthermore, it is required that the functions $y_j^{(0)}(\underline{x})$ are such that:

(i) $$y_j(t,\underline{x}) = \sum_{i=1}^{m} \{\beta_{ji}^{(0)} + (\alpha_{ji} - \beta_{ji}^{(0)})t/M)\} \, v_{ji}(\underline{x}) \tag{7.20}$$

is continuous in the interval $0 \leqslant t \leqslant M$. Note that for convenience the discrete increments k/M have been substituted by the continuous variable t/M.

(ii) The zeros of (7.20) denoted by $Z(t)$ are continuous in the same interval $0 \leqslant t \leqslant M$. (7.21)

The condition for a continuous function of the roots, $Z(t)$, is illustrated in Fig. 7.3 for a single equation $f(x) = 0$. Since the root $Z_1(0)$ does not continuously approach $Z(M)$, the method fails. However, a choice of $Z_3(0)$ would succeed. It is also noted that if the Newton iteration, or any other iteration method, would be applied to $f(x)$ for $x < A$, the method would certainly fail to locate a root of $f(x)$.

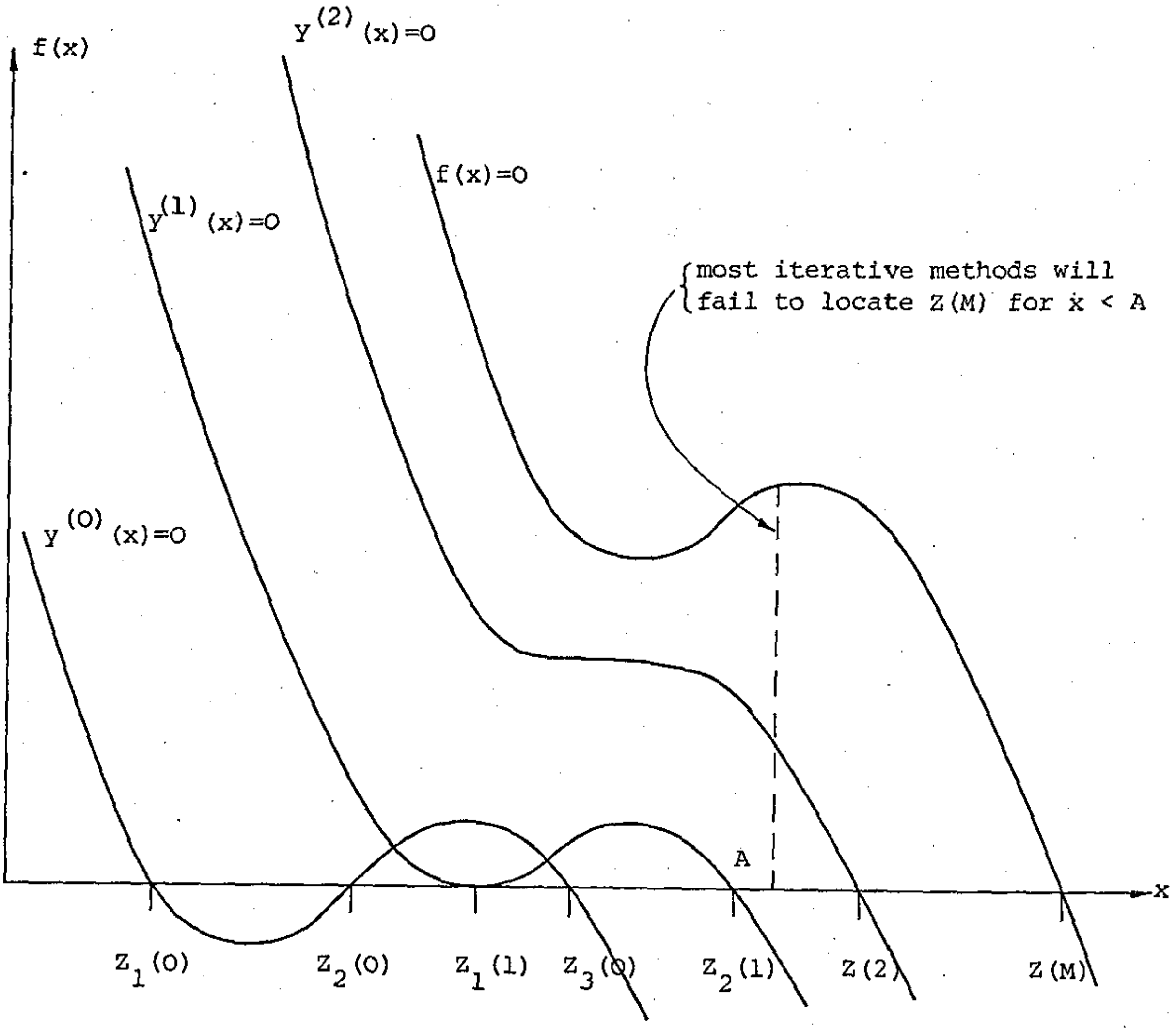


Fig. 7.3 The continuity of the roots Z(t) is violated if the parameter perturbation procedure is started from $Z_1(0)$ or $Z_2(0)$, however a start from $Z_3(0)$ will converge to Z(M).

7.3.2 Application of the Method to the Signal Design Problem

The parameter variation method is now applied to find a solution of

$$F_p(\underline{\phi}) = \sum_{k=1}^{N-1} |r(k,\underline{\phi})|^p = 0 \tag{7.22}$$

where

$$r(k,\underline{\phi}) = \sum_{n=o}^{N-k} \exp(j[\phi(n) - \phi(n+k)]) \tag{7.23}$$

By the introduction of a set of real parameters $\beta(n)$, $n = 0,1,2,\ldots,N$ a new system of equations is derived

$$r(k,\underline{\phi},\underline{\beta}) = \sum_{n=o}^{N-k} \beta(n)\ \beta(n+k)\ \exp(j[\phi(n) - \phi(n+k)]) \tag{7.24}$$

and

$$F_p(\underline{\phi},\underline{\beta}) = \sum_{k=1}^{N-1} |r(k,\underline{\phi},\underline{\beta})|^p = 0 \tag{7.25}$$

where $p = 4$.

From the foregoing it should be evident that in general there is no formal way of introducing the parameters. However, it has been shown in the preceeding section that Huffman codes satisfy Eq. (7.25). Since there are 2^N Huffman sequences for a given length $(N+1)$, 2^N roots of (7.25) can be established. Thus, if the sequence $a(n)$ denotes a Huffman code, a choice of the parameters $\underline{\beta}^{(0)}$ and the initial phase vector $\underline{\phi}^{(0)}$ such that

$$\beta^{(0)}(n) = |a(n)| \tag{7.26}$$

and $$\phi^{(0)}(n) = \arg[a(n)] \qquad n=0,1,\ldots,N$$

will furnish a solution of (7.25). It is noted that if the parameters are introduced as indicated above, the $r(k,\underline{\phi},\underline{\beta})$ still represent essentially an ACF.

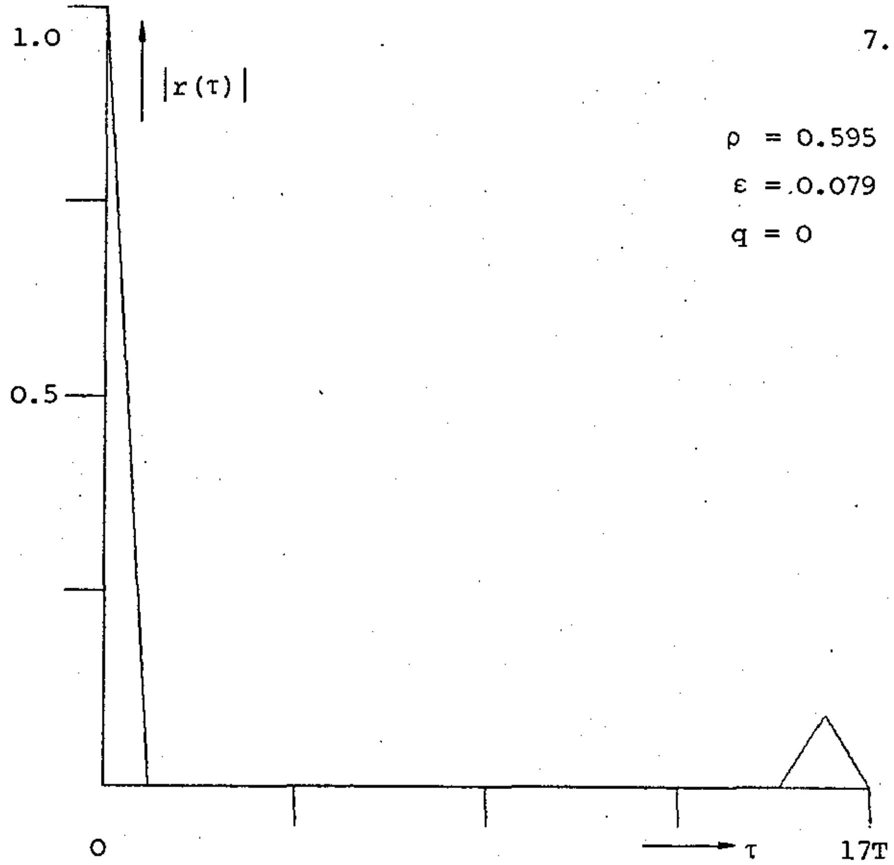
The parameter variation method was coded in Fortran and applied to a number of Huffman codes with high energy efficiency. The tracing of a root from the kth to the (k+1)th deformed system of equations was carried out using the modified Newton-Raphson iteration (Powell's sums of squares method, Chapter 5), and the efficient pattern search technique. Both methods failed to locate a root of the objective function at successive increments because the given set of equations (7.24) do not satisfy the sufficiency conditions (7.21). This can easily be seen by considering a given Huffman code $A(z)$ with optimum energy ratio ρ_{opt} for a specified zero pattern. Evidently every solution which satisfies (7.25) exactly must be of the Huffman type. If the coefficient magnitudes $|a(n)|$ of $A(z)$ are incremented by $\Delta(n)$ and if the resulting sequence $|a'(n)|$ should satisfy (7.25), the zero pattern of its z-transform, $A'(z)$, must be of the Huffman type. However, the zeros of a polynomial are continuous functions of its coefficients (Eq. (7.11)). Consequently, since $\Delta(n)$ can be any arbitrary small value, the zeros of $A'(z)$ and $A(z)$ must be very close. It can be shown that the variation of the ith zero of $A'(z)$ is approximately given by[93]

$$\Delta z_i = - \zeta_i \sum_{n=o}^{N} \Delta(n)\ z_i^{-n} \tag{7.27}$$

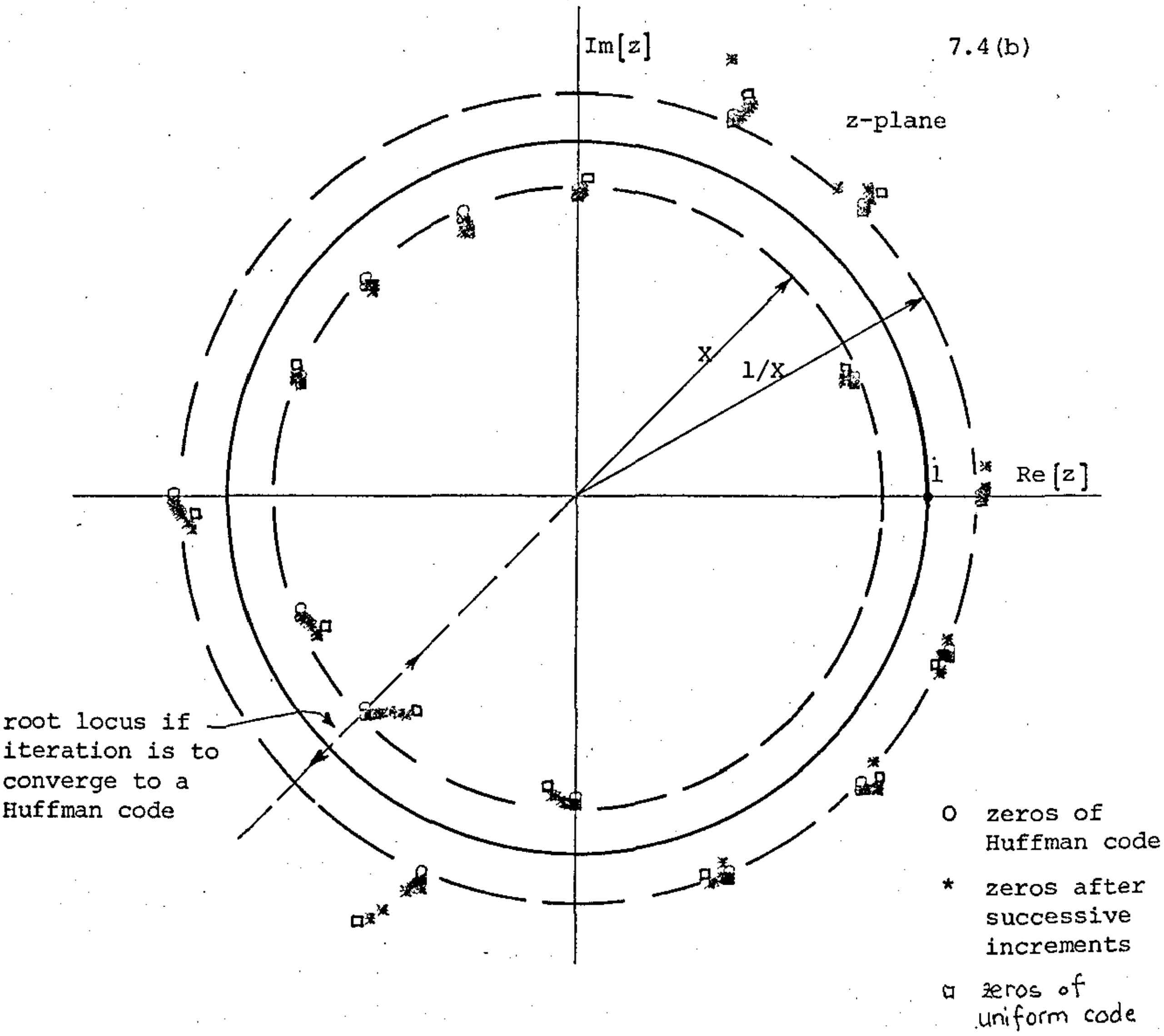
where

$$\zeta_i = (z - z_i)/A(z)\Big|_{z=z_i}$$

Since the zero pattern of $A'(z)$ must be of the Huffman type, if the iteration is to converge to a root of (7.25), it follows that its zeros can only move radially (ignoring a possible constant phase shift), as indicated in Fig. 7.4(b). Incrementing the coefficients of $A(z)$ must, however, always increase the energy ratio ρ, hence $\rho' > \rho_{opt}$ which is not possible. Hence, the roots of the objective function (7.25) cannot be continuous functions of the parameters $b(n)$. This conclusion


1.0
|r(τ)|
ρ = 0.595
ε = 0.079
q = 0
0.5
0
τ
17T

7.4(a)


Im[z]
z-plane
X
1/X
1
Re[z]
root locus if iteration is to converge to a Huffman code
O zeros of Huffman code
* zeros after successive increments
□ zeros of uniform code

7.4(b)

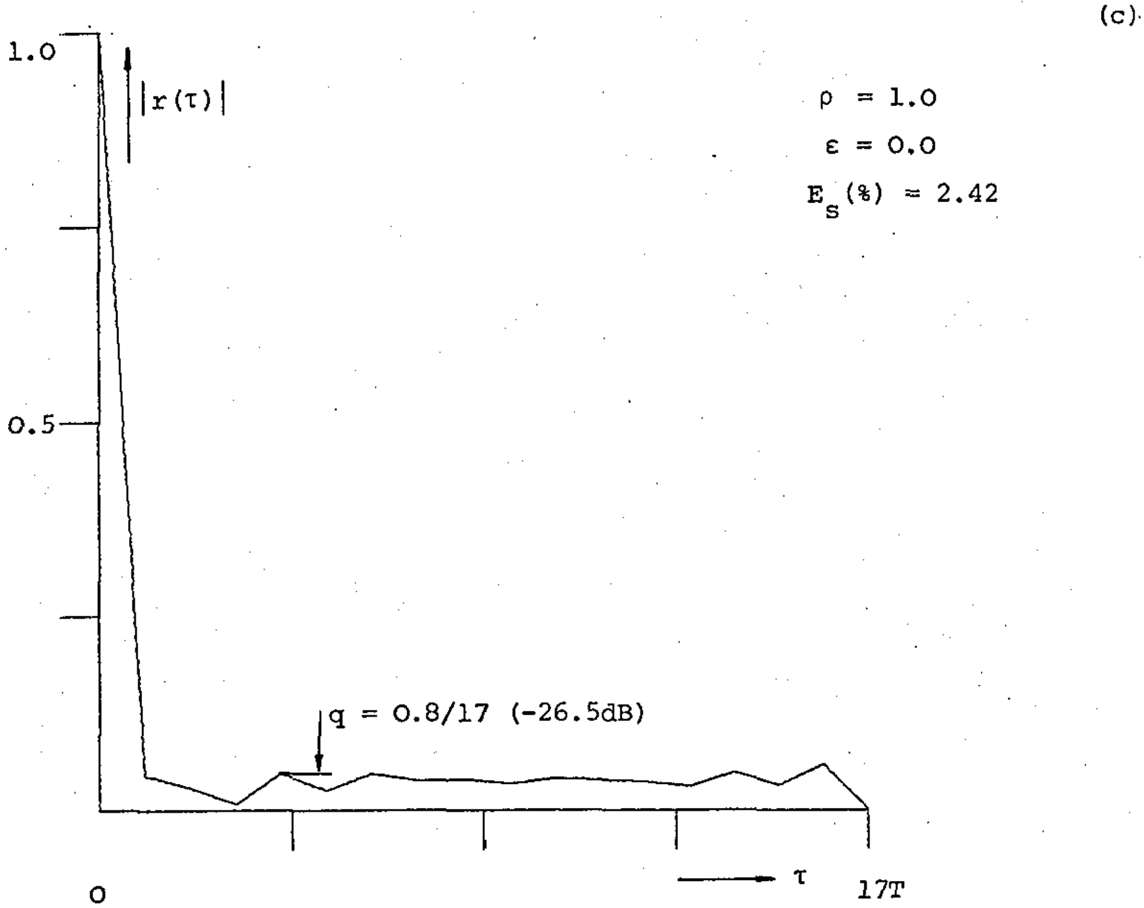


Fig. 7.4 (a) ACF of 17-element Huffman code
(b) root locus for successive increments
(c) ACF of resulting uniform complex code.

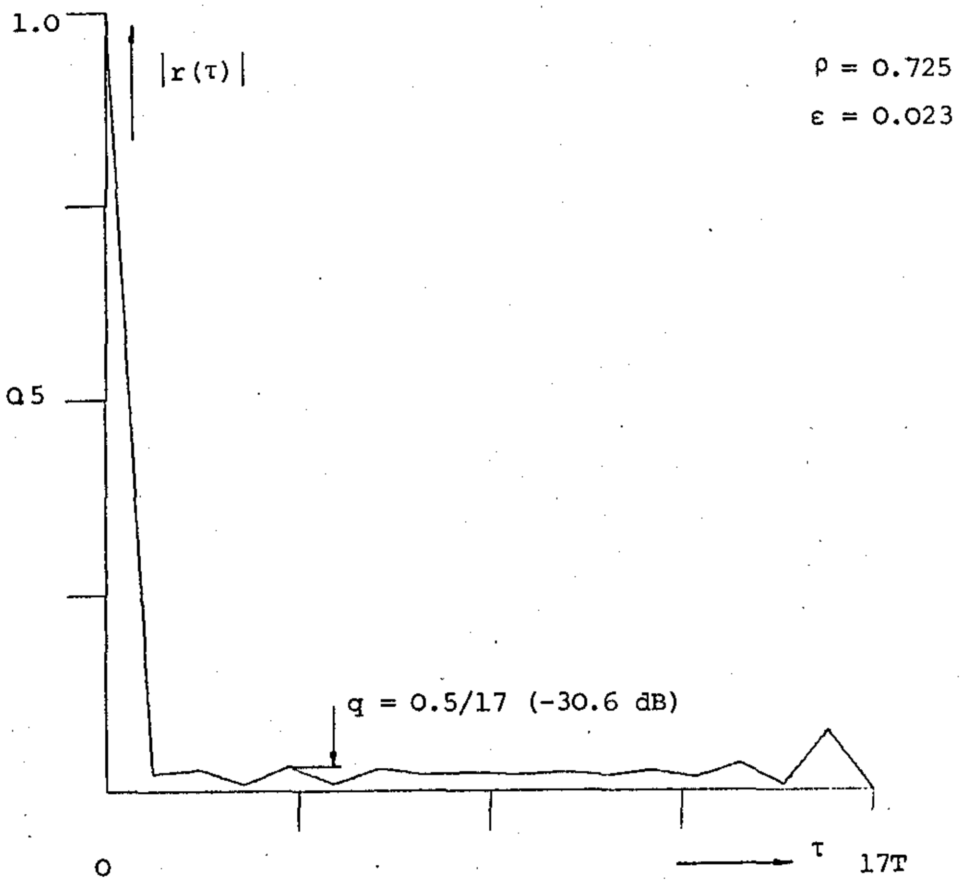


Fig. 7.5 ACF of 17-element quasi-Huffman code.

could also have been drawn directly from the observation that the solution set of (7.1) is discrete (2^N possible solutions). Fig. 7.4(b) shows the actual locus of the zeros, starting with a 17-element Huffman code.

In addition the convergence of least squares and Newton methods is limited to regions for which the condition rank (J) = N is satisfied. As pointed out in Section 5.5 the Jacobian J becomes very ill-conditioned for trigonometric functions when the number of variables increases. Hence, singularities ($|J| = 0$) in the ($\underline{\phi}$,$\underline{\beta}$)-space are very likely to be encountered. In principle various schemes exist to overcome the problem of a vanishing Jacobian. For example, the increments in the parameters can be made selectively unequal until the value of J is greater than some predetermined minimum. Another approach is to restart the procedure with a different set of equations. However, none of these methods can guarantee convergence for the simple reason that there might be no solution to the problem. Moreover, in practice it is usually not known a priori whether a given set of equations satisfy the conditions for convergence, since the knowledge of these conditions implies the knowledge of the solution itself.

Since the iterations do not converge to an exact solution, 2(N-1) additional sidelobes will, in general, be introduced. However, the peak magnitude of these additional residues can be kept to a specified low value q, and the iteration may be stopped as soon as this value is exceeded by one or more of the sidelobes as shown in Fig. 7.5. The improvement in energy and error performance and the increase in sidelobe level for successive increments in the parameters starting from q = 0 (Huffman case) is illustrated in Fig. 7.6.

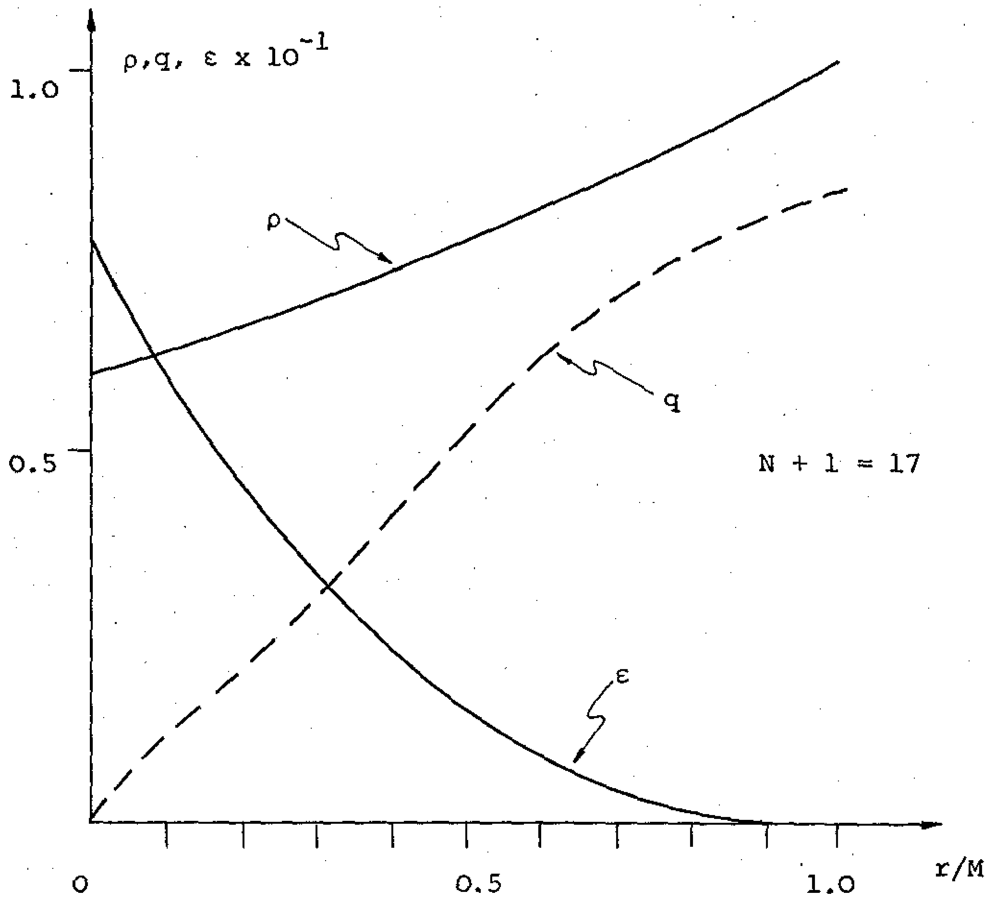


Fig. 7.6 Energy, error and sidelobe performance at successive increments.

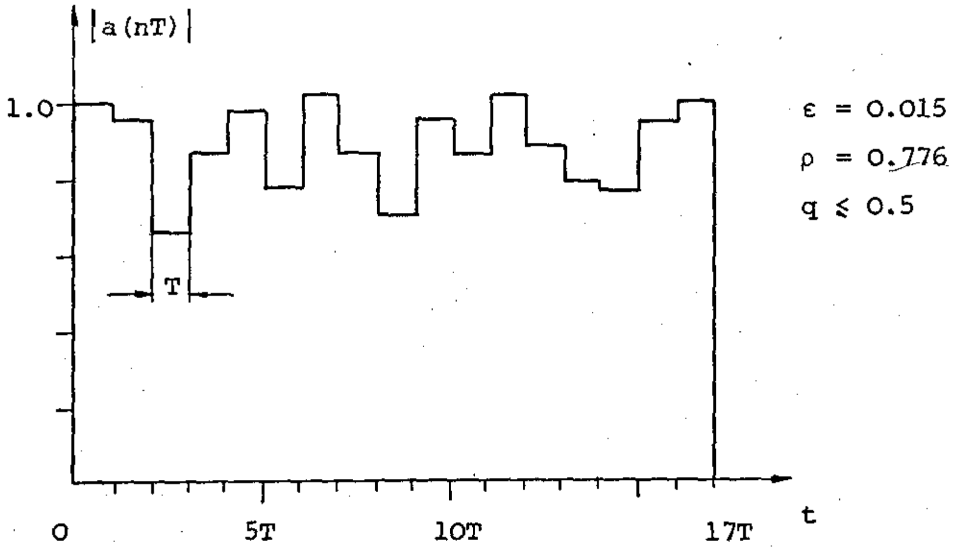


Fig. 7.7 Envelope of quasi-Huffman code of length N + 1 = 17.

In all cases a considerable increase in power utilization is obtained, Fig. 7.8, even for moderate values of q ($q \lesssim 1$). For example the 17-element sequence depicted in Fig. 7.7 shows an increase in pulse train energy efficiency of 24% for $q \lesssim 0.5$. This is significant because the energy of the added time sidelobes is only 0.9% of the mainlobe energy with a maximum value of the q time sidelobes equal to - 30dB.

Although no precise formulation for the design of Huffman sequences can be made, the following remarks may serve as a guide line to their synthesis. It is well known that if the zeros of a polynomial A(z) are all within the unit circle the polynomial is called minimum delay, and its energy is concentrated towards the beginning. Conversely, the maximum delay polynomial has all its zeros outside the unit circle and consequently its energy is concentrated towards the end. Therefore, when half of the zeros lie on either circle, the envelope of the pulse train tends to be relatively symmetric about the mid-pulse. Moreover, the energy ratio ρ is very sensitive to the radius X. In Fig. 7.9, ρ is plotted as a function of X for a 17-element sequence. It can be seen that for this particular zero pattern (Fig. 7.1), ρ peaks quite sharply around the value where $E \approx N+1$. It is also noted that the maximum energy ratio does not correspond to the minimum error. The resulting magnitude of the complex envelope of the pulse train for the optimum radius X_ρ(opt) is depicted in Fig. 7.10. In addition it was observed that for relatively large X ($X > X_\rho$(opt)) with half the zeros inside the unit circle, the energy tends to be concentrated in the centre of the pulse train. As X approaches unity, the energy becomes more concentrated at the extremes of the pulse train.

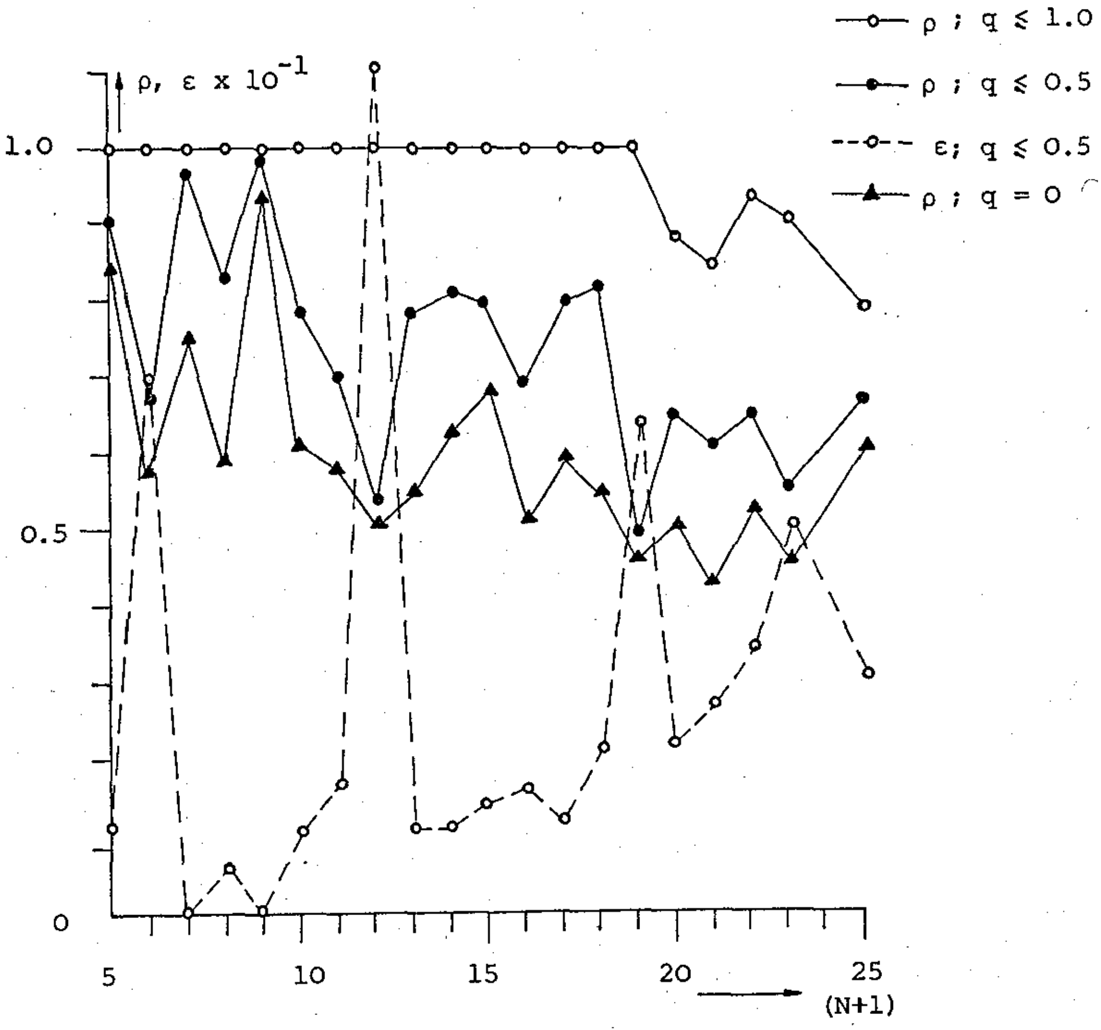


Fig. 7.8 Energy and error performance of quasi-Huffman codes.

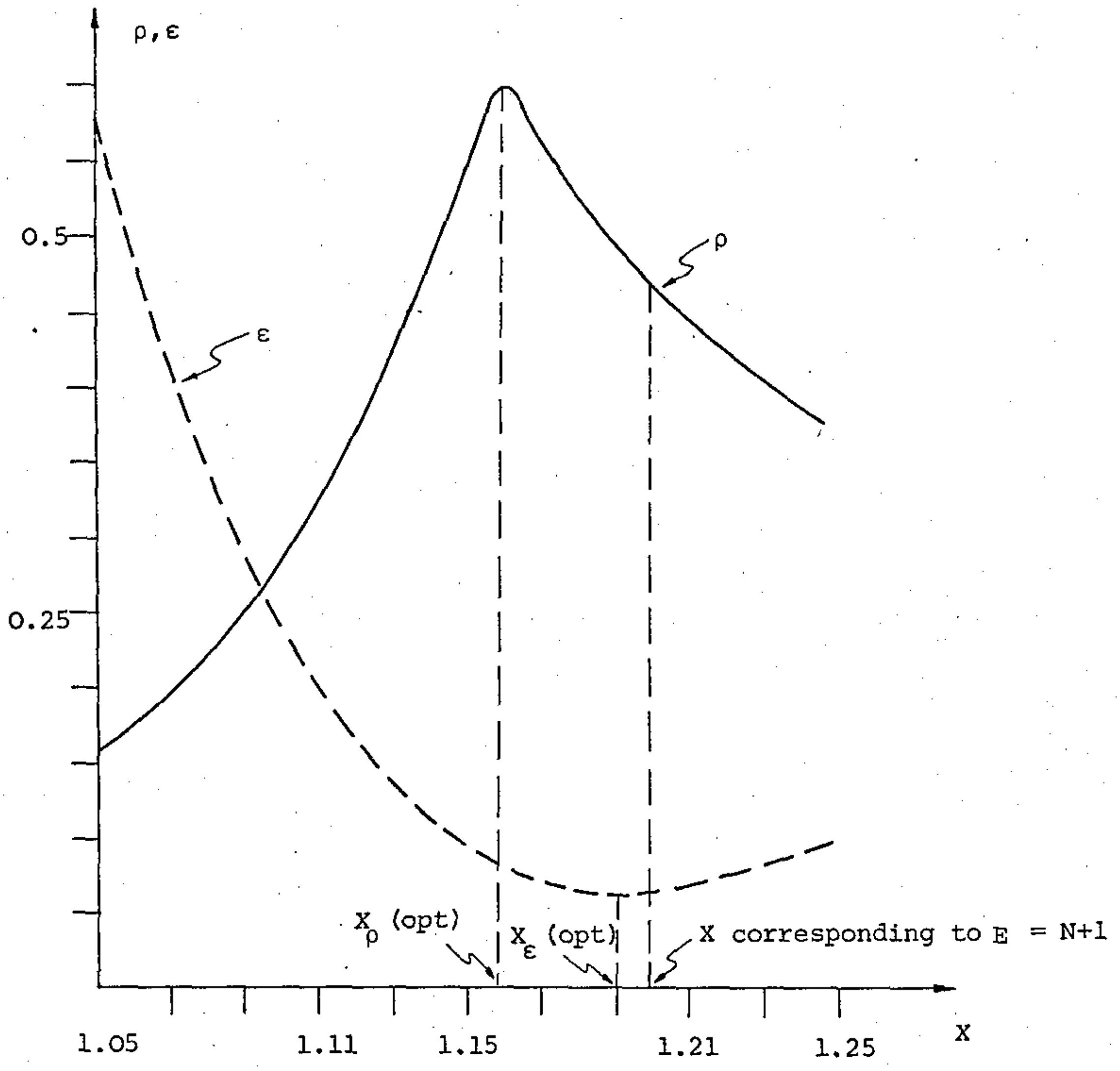


Fig. 7.9 Energy efficiency ρ and mean-square error ε as a function of the circle radius X for N+1 = 17.

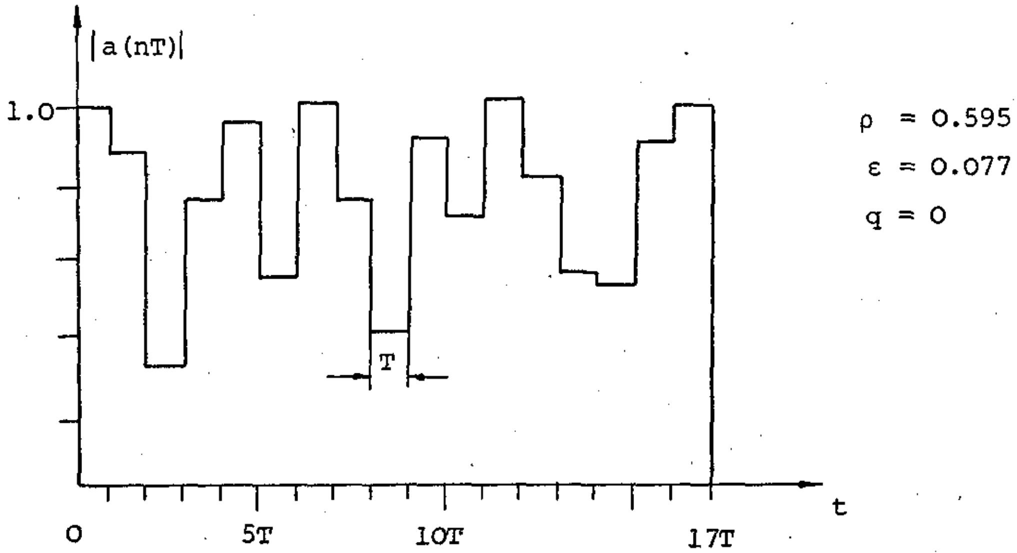


Fig. 7.10 Envelope of 17-element Huffman code for X_ρ (opt).

7.4 Bessel Sequences

A different class of a.m.ph.m. pulse trains having low ACF sidelobes can be generated using Bessel functions. The nth order Bessel functions, $J_n(x)$, of the first kind and of argument x are defined as the Fourier coefficients of the function[94]

$$S(\theta) = e^{-jx \sin \theta} \tag{7.28}$$

The function $S(\theta)$ is periodic with a period of 2π. Making the variable transformation $\theta = 2\pi f/W$, the FT relationship

$$\sum_{n=-\infty}^{\infty} J_n(x)\ \delta(t-nT) \leftrightarrow S(2\pi f/W) \tag{7.29}$$

can be written as

$$J_n(x) = 1/W \int_{-W/2}^{W/2} e^{-j(x \sin 2\pi f/W - 2\pi n f/W)} df \tag{7.30}$$

and

$$S(f) = e^{-jx \sin 2\pi f/W} = \sum_{n=-\infty}^{\infty} J_n(x)\ e^{-j2\pi n f/W} \tag{7.31}$$

Therefore, for any given argument x, the mth element of a Bessel code is simply the mth order Bessel function of the first kind, $J_m(x)$. The Bessel functions can thus be regarded as being generated by sampling the complex envelope of a bandlimited continuous signal at the Nyquist rate $1/T = W$. With this notation Eq. (7.29) has a ZT given by

$$S(z) = \sum_{n=-\infty}^{\infty} J_n(x)\ z^{-n} \tag{7.32}$$

Since S(f) has the property of complex conjugate symmetry, the coefficients $J_n(x)$ are all real. Moreover, Bessel functions of integral order satisfy the relationships

$$J_n(-x) = (-1)^n J_n(x) \tag{7.33}$$

and

$$J_{-n}(x) = (-1)^n J_n(x) \tag{7.34}$$

Incidentally, it may be noted that the relationship (7.34) is the reflection condition (Eq. (6.27)) discussed in Section 6.4.1 which causes the ACF to be zero for odd time shifts.

The ACF of the sequence $J_n(x)$ (x fixed) is by definition (Eq. (2.35))

$$r(k) = \sum_{n=-\infty}^{\infty} J_n^*(x)\ J_{n+k}(x) \tag{7.35}$$

$$k = 0, \underline{+}1, \underline{+}2, \ldots\ldots.$$

Substituting Eq. (7.30) into the above expression results in

$$r(k) = T \sum_{n=-\infty}^{\infty} J_{n+k}(x) \int_{-\frac{1}{2}T}^{\frac{1}{2}T} S^*(f)\ e^{-j2\pi nfT}\ df$$

changing the order of integration gives

$$r(k) = T \int_{-\frac{1}{2}T}^{\frac{1}{2}T} \sum_{n=-\infty}^{\infty} J_{n+k}(x)\ e^{-j2\pi nfT}\ S^*(f)\ df$$

letting m = n+k

$$r(k) = T \int_{-\frac{1}{2}T}^{\frac{1}{2}T} |S(f)|^2\ e^{j2\pi kfT}\ df$$

$$= T \int_{-\frac{1}{2}T}^{\frac{1}{2}T} e^{j2\pi kfT}\ df = (\sin k\pi)/k\pi = \begin{cases} 1 & , \ k = 0 \\ 0 & , \text{ otherwise} \end{cases} \tag{7.36}$$

Hence the ACF consists of a single spike of unit magnitude at k = 0. From Eq. (7.35) it follows immediately that

$$r(0) = E = \sum_{n=-\infty}^{\infty} J_n^2(x) = J_o(x) + 2 \sum_{n=1}^{\infty} J_n^2(x) = 1 \tag{7.37}$$

Although the summation in Eq. (7.35) is infinite, the number of Bessel functions with significant magnitude is limited to a finite number. The magnitude of each term depends on the argument x (in FM x is known as the modulation index) and, moreover, the number of significant terms increases with x. The question now arises of just how many terms are important. For a given argument, x_o, a typical high-order (n >> 1) Bessel function, $J_n(x)$, is essentially zero for $(n - 2) > x_o$ (Appendix C). Hence, truncation with $J_{n+2}(n)$ as the highest-order significant factor encompasses most of the significant components. The sequence length N of a Bessel code for a given argument x_o is therefore given by

$$N \approx 2(x_o + 2) + 1$$

In Fig. 7.11 the number of significant terms, disregarding higher-order Bessel functions of magnitude less than 0.02, is plotted as a function of the argument x. Provided the condition (7.34) is not violated such a truncation will only introduce ACF sidelobes for even time shifts. Moreover, it has been found that the magnitude of the sidelobes is not greater than about -35 dB for the sequence lengths shown in Fig. 7.11. A typical Bessel code of length N = 61 and its ACF is shown in Fig. 7.12 and Fig. 7.13 respectively.

The energy efficiency ρ and the mean-square error ε of this type of pulse train is shown in Fig. 7.14 as a function of the sequence length N. Comparing Fig. 7.2 with Fig. 7.14 it can be seen that for relatively short length sequences (N < 31), Huffman codes are clearly superior in energy performance. This is not surprising, since complex sequences have essentially 2N degrees of freedom (amplitude and phase can be coded independently) whereas Bessel codes have only N degrees. However, for larger lengths (N > 31) the difference in efficiency ρ and error ε is marginal, which suggests that the use of complex Huffman codes for these lengths does not provide any significant advantage in energy efficiency.

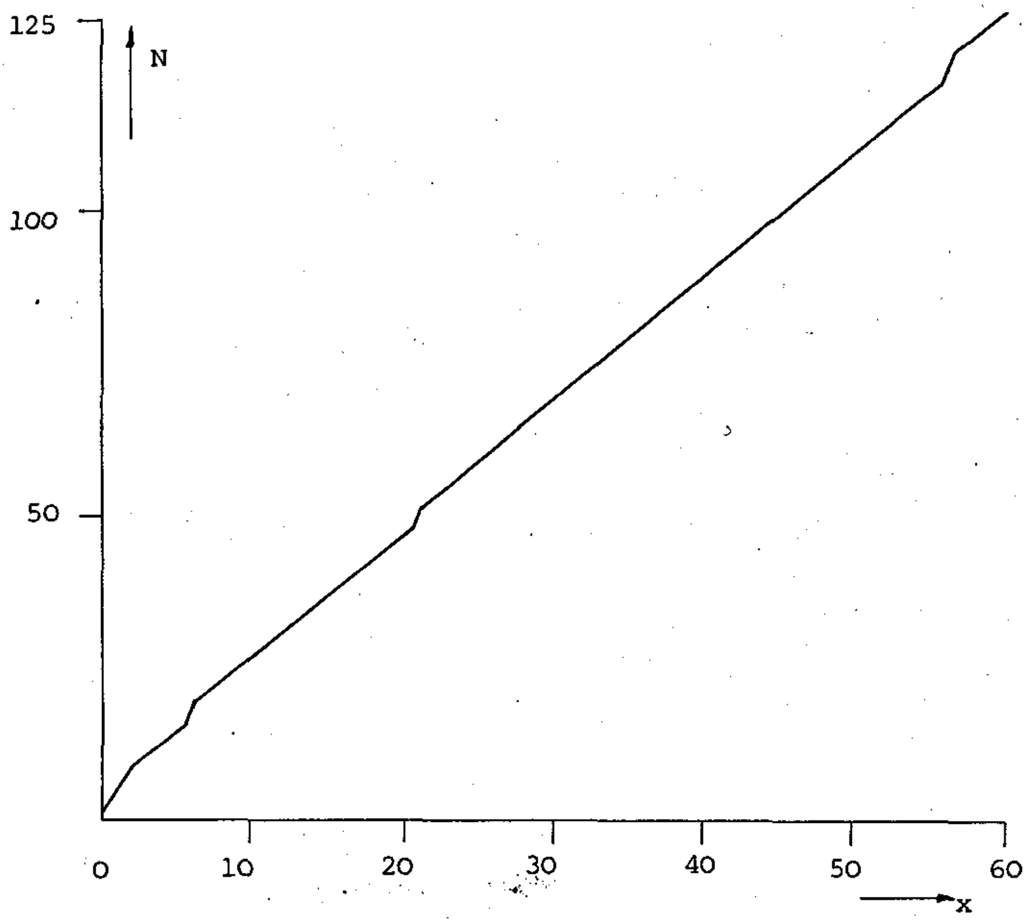


Fig. 7.11 Number of significant Bessel functions as a function of the argument x.

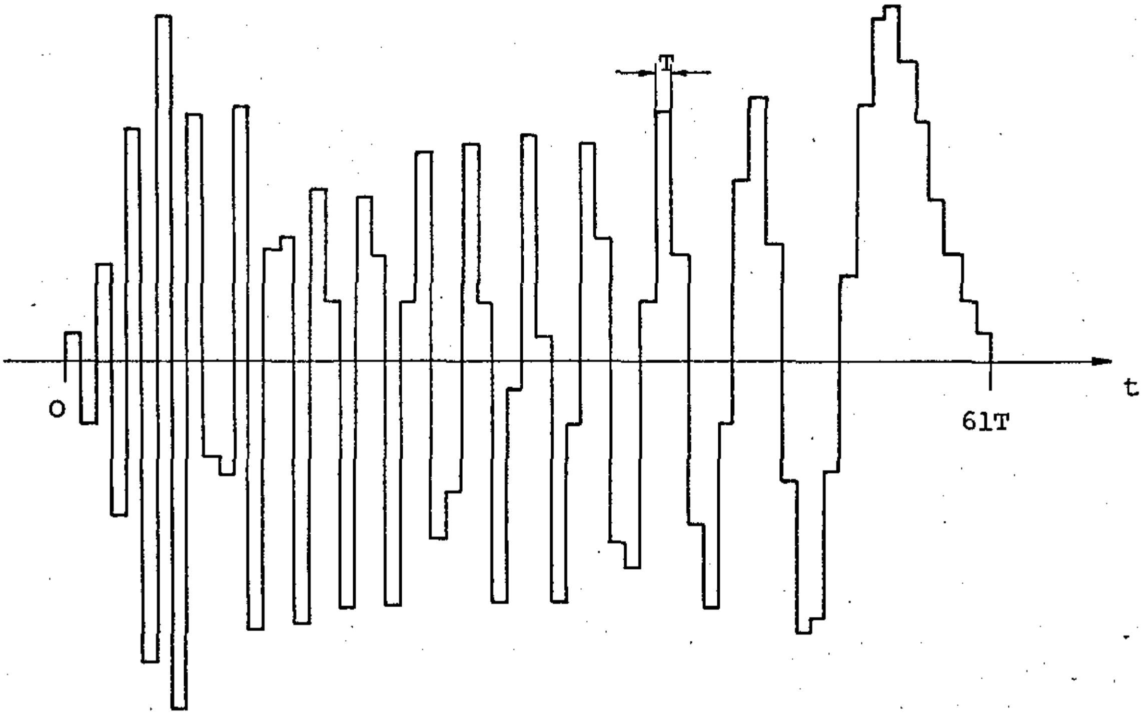


Fig. 7.12 Waveform of 61-element Bessel code.

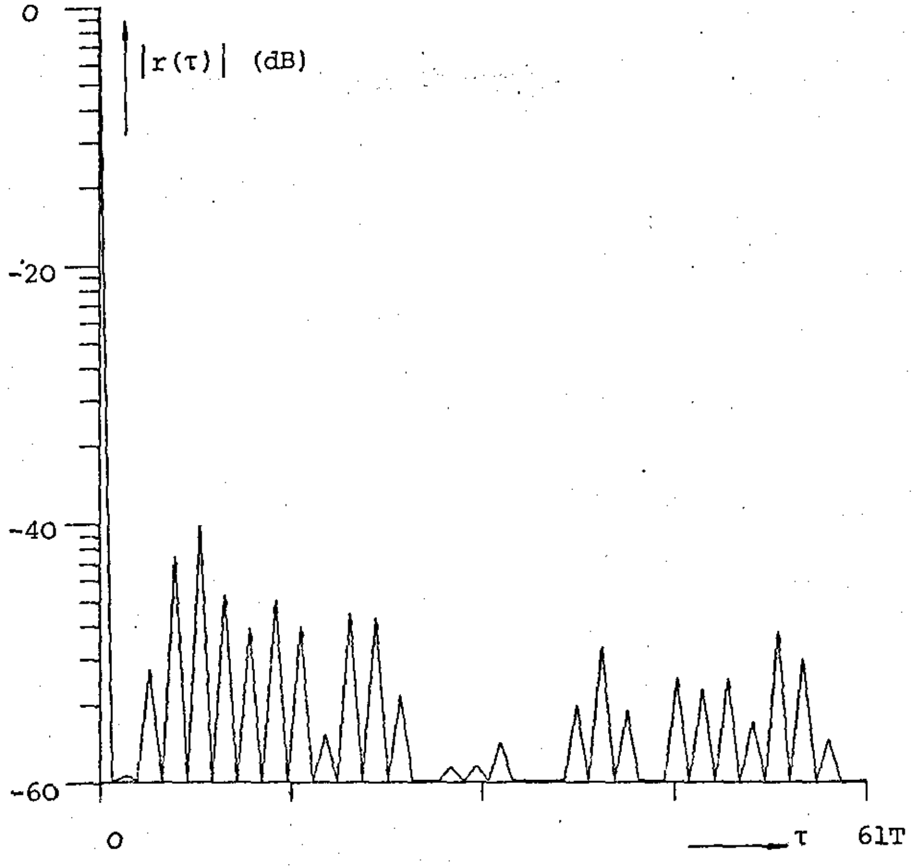


Fig. 7.13 ACF of 61-element Bessel code.

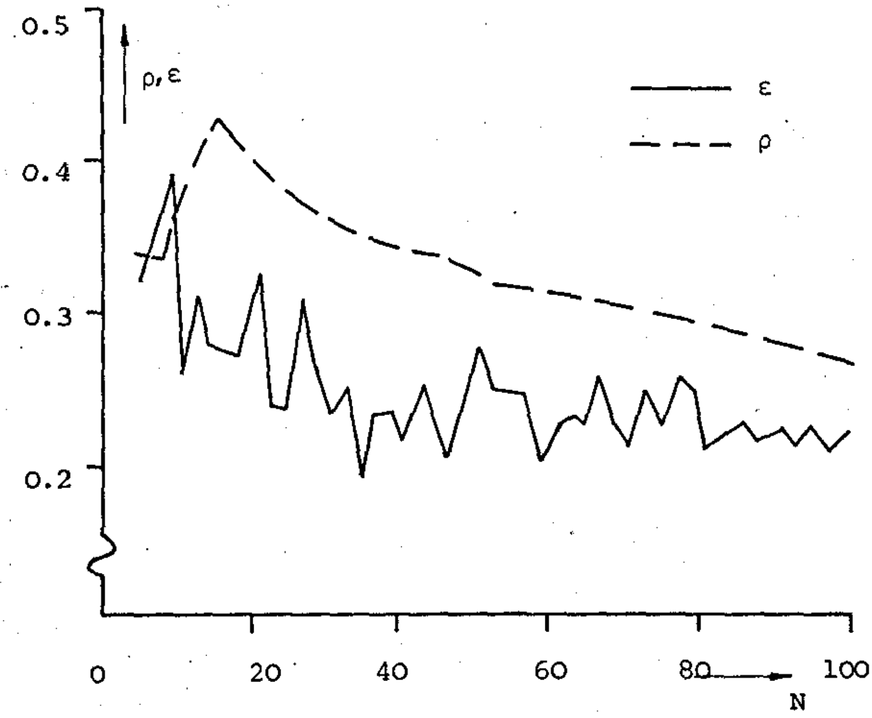


Fig. 7.14 Energy and error performance of Bessel codes as a function of N.

Another interesting property of Bessel codes is their zero pattern which is obtained by factorizing Eq. (7.32). A typical zero pattern is shown in Fig. 7.15. A distinguishing feature of Bessel code zero patterns is the 'horse-shoe' appearance of their root loci as compared to the zeros of QP codes which tend to lie on a spiral (see Fig. 4.19).

Alternatively, Bessel codes could provide an initial starting sequence in the design of uniform or quasi-uniform complex codes using the parameter variational technique described in the preceeding section. For example, Fig. 7.16 shows the root locus of the zeros when a Bessel code of length $N = 21$ was used. The resulting uniform complex code had a peak sidelobe level of 1.03 or -26 dB.

7.5 Summary

In some circumstances the use of pulse trains, other than purely phase modulated ones, may be precluded due to the expense incurred in providing AM. However, the additional expense of encoding and decoding in amplitude and phase may be justified for radars that must cope with land clutter or operate in a dense-target environment.

The very low sidelobes that can be achieved with a.m.ph.m. pulse trains such as Huffman codes ($q = 0$), quasi-Huffman codes ($q \lesssim 1$), or Bessel codes make their use particularly attractive in systems requiring a large dynamic range. Moreover, the excellent self-clutter rejection performance is obtained without sacrificing close-target separability (no mainlobe widening). This property is, however, achieved at the expense of a significant reduction in power utilization at the transmitter. In addition it has been shown that complex Huffman codes and Bessel codes of lengths $N > 31$ are comparable in energy efficiency ρ and mean-square error ε. Since Bessel codes are real (phase is 0 or π), this implies that for longer sequences ($N > 31$) the phase modulator may be replaced by a switching phase inverter without any significant loss in energy performance.

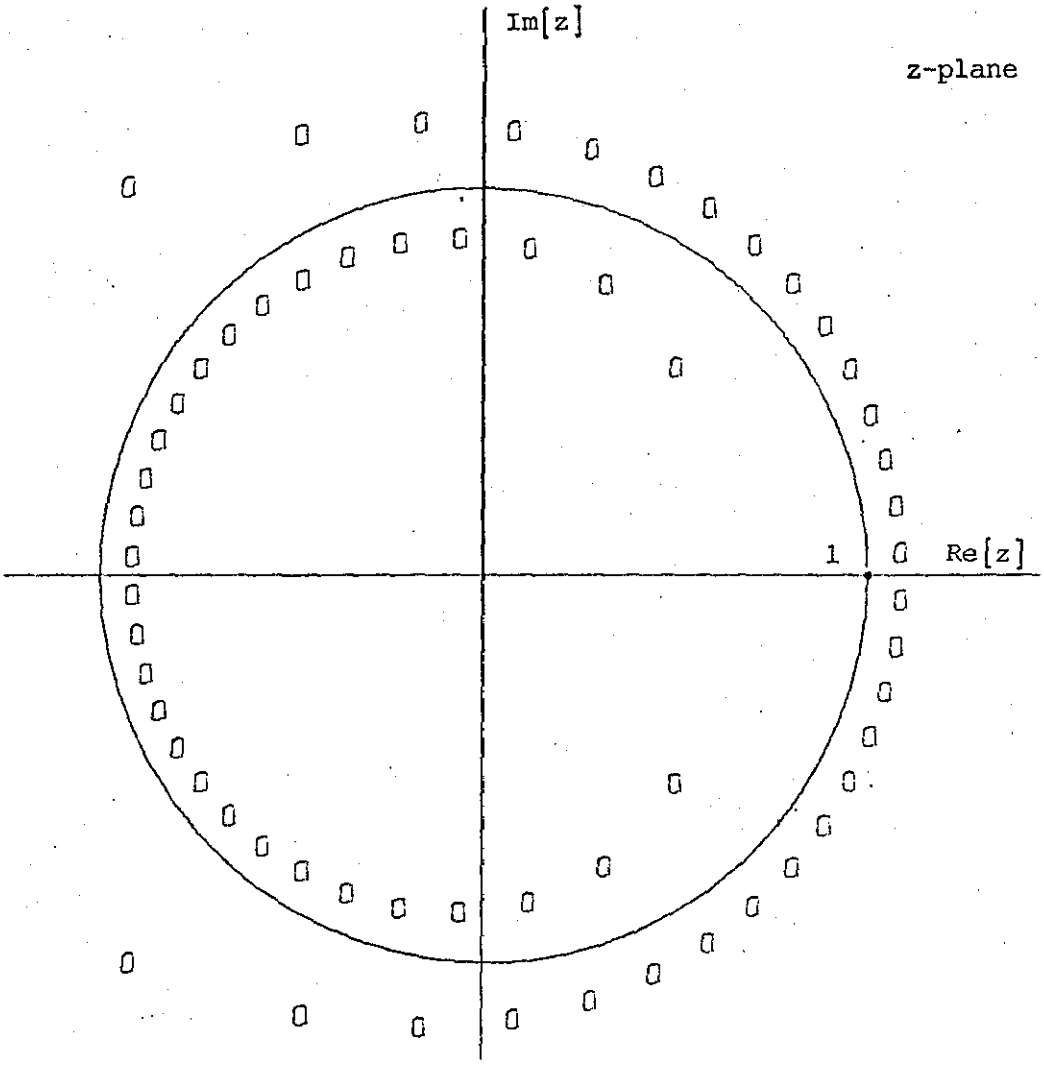


Fig. 7.15 Zero pattern of 61-element Bessel code.

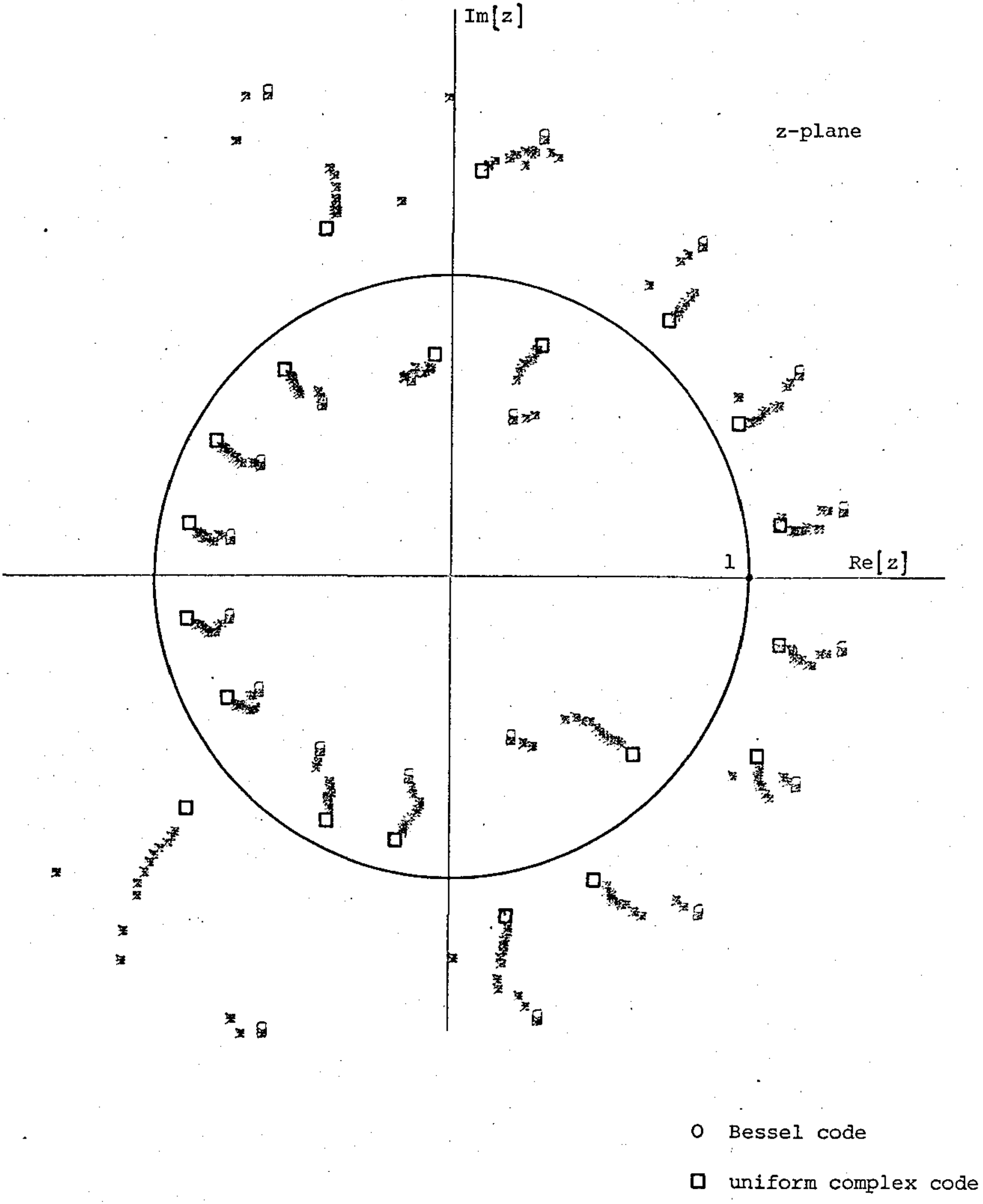


Fig. 7.16 Root locus for successive increments, starting with 21-element Bessel code.

The loss in energy efficiency of Huffman and Bessel codes is mainly attributed to the high degree of sidelobe suppression. In most applications a complete suppression of the sidelobes is not required, provided they can be kept to a specified low level. Consequently, a method which trades sidelobe performance against energy efficiency has been developed using parameter variational techniques.

Although the reduction in sidelobe levels can be achieved by other means to be considered in the next chapter, this method of tackling the signal design problem has some useful advantages. First, the algorithm provides a starting point which may lead to improved numerical methods for a number of related signal or filter design problems. Secondly, with a suitable choice of q and the parameters b(n), sequences with high energy efficiency and discrimination ratio can be obtained. Moreover, the algorithm is less sensitive to the initial starting point and, therefore, generally performs better when applied to multimodal objective functions. In many cases the best uniform complex codes (Section 6.3) were obtained in only one computer run whilst other methods required a number of runs, each time starting at a different initial point. For example, an additional complex sequence having the Barker code property (all sidelobes $\leqslant 1$) was found for $N+1 = 19$ (Appendix E). However, the requirement for a succession of solutions of the M sets of equations may result in excessive computer time for large dimensional problems.

CHAPTER 8

SIDELOBE REDUCTION FILTERS

8.1 Introduction

In a dense-target environment or in situations where there are large undesired scatterers (point clutter), it is often desirable to reduce the time sidelobes of phase coded sequences to a prescribed low level. In principle, there is no difference between the problems of resolving a target in the interference from other targets and the detection of a target in clutter. In previous chapters the reduction of the range sidelobes of the compressed pulse has been of much concern in the application of matched filter techniques to radar systems. In fact the mutual interference between targets or self-clutter imposes rather fundamental limitations on resolution performance. So far, the reduction of the range sidelobes has been approached via waveform design. However, the attainable sidelobe levels for phase coded pulse trains might be inadequate for specific applications. Although it is possible to use a.m.ph.m. pulse trains such as Huffman codes to obtain the desired degree of discrimination, in most high power radar systems this method is not readily available. Thus, the designer has to resort to other sidelobe reduction techniques.

In principle sidelobe reduction can be achieved by either

(i) amplitude- or phase-weighting in the frequency domain, or,

(ii) amplitude- or phase-weighting in the time domain.

The weighting may be accomplished at the transmitter or receiver or at both. Furthermore, the shaping can be performed at the RF, IF, or video stages.

Sidelobe suppression in the frequency domain requires the design of a filter such that the spectrum of the filtered waveform has a linear phase-frequency relationship and that the spectrum magnitude be proportional to one of the many available weighting functions such as Taylor or Chebyshev[22] (see also Chapter 4). The rapid advance of digital hardware, along with the pipeline FFT configuration, does permit practical realization of the required transfer function as depicted in Fig. 2.9. However, since the objective here is to design a filter for specified sidelobe levels it is preferable to work in the time domain rather than the frequency domain.

Amplitude weighting may be introduced into a pulse-compression system either entirely at the receiver, entirely at the transmitter, or at both simultaneously. Equal weighting at both the transmitter and receiver is equivalent to altering the transmitted waveform. In this case the system is still considered as matched. However, it can be shown that in the peak power limited case, the SNR which assumes weighting at the receiver alone is greater or equal to the SNR which assumes matched weighting at the transmitter and receiver[35]. An additional reason for unilateral weighting at the receiver is due to the advantage of operating the transmitter at its peak power limit (no expensive amplitude modulators required). Furthermore, amplitude weighting solely at the receiver can be maintained conveniently, due to the accessibility of the components and the low power levels involved. For these reasons it is henceforth assumed that weighting is performed solely at the receiver at the expense of a lower SNR.

A convenient way of weighting is at the IF stage in the time domain. Most of the sidelobe reduction techniques which have been proposed depend on cascading a weighting filter (tapped delay line) after the MF or by providing a suitable band shaping network[95]. However, instead of placing a sidelobe reduction filter after the MF it is probably more straightforward to design a mismatched filter (MMF) under some conditions of optimality. The amount of mismatch from the matched conditions is usually characterized by the loss factor L_s, given by

$$L_s = \frac{\text{SNR (weighted)}}{\text{SNR (matched)}} .$$

For an input sequence a(n) of length (N+1) and a filter weighting sequence h(n) of length (M+1), L_s becomes

$$L_s = \frac{\left| \sum_{n=o}^{M} h(n)\ a(j-n) \right|^2}{\sum_{n=o}^{N} |a(n)|^2 \sum_{n=o}^{M} |h(n)|^2} \leqslant 1 \tag{8.1}$$

where $j \geqslant N$ denotes the time delay for which the output is maximum. (It is assumed that $(M+1) \geqslant (N+1)$). The basis for Eq. (8.1) is that for coherent summation signal components add as voltage levels while noise components add as power levels.

The problem in selecting an appropriate weighting function for a pulse-compression system is to find out which finite spectrum shape can produce the desired waveform under some criterion of optimization.

8.2 Inverse Filters

The reduction of the sidelobe interference can be accomplished through the use of inverse or deconvolution filters. These filters (equalizers) have been of much concern in removing intersymbol interference in communications and they are also of interest in spectrum analysis. The ideal inverse, h(n), of a code sequence, a(n), is defined as

$$a(n) * h(n) = \delta(n-m) \tag{8.2}$$

where $\delta(n)$ is the Kronecker delta function. In terms of the ZT, the inverse relationship is given by

$$A(z)\ H(z) = z^{-m} \tag{8.3}$$

Solving for the unknown filter transfer function, H(z), leads to

$$H(z) = z^{-m}/A(z) \tag{8.4}$$

The factor z^{-m} representing a delay of mT seconds is immaterial for the present discussion and may be ignored. Hence,

$$H(e^{j2\pi fT}) = 1/A(e^{j2\pi fT}) \tag{8.5}$$

It can be seen from the above expression that the ideal inverse filter and the filter matched to A(z) differ only in their gain characteristics, the phase being identical.

The problem of solving Eq. (8.2) or Eq. (8.3) is equivalent to solving a Fredholm integral equation of the first kind*. The inherent difficulty in solving this type of equation is that the convolution operator may not have an inverse, let alone a bounded inverse[96]. This leads, in general, to non-unique, widely oscillating unstable solutions and the success in solving Eq. (8.3) to a close approximation will depend largely on the properties of A(z).

If A(z) has all its zeros inside the unit circle in the z-plane (minimum delay), the inverse filter, H(z), is stable and can be realized either by a recursive filter, Fig. 8.1, or it can be closely approximated by a transversal filter, Fig. 8.2. However, it has been shown in the preceeding chapter that sequences with high energy efficiency must have approximately half of their zeros inside and half outside the unit circle. The inverse filter will thus have poles outside the unit circle and consequently will be unstable. Hence H(z) will not be physically realizable for the input signals of interest.

* In general the linear integral equation may be written as

$$h(x)\ f(x) + \int_a^b K(x,y)\ f(y)\ dy = g(x), \quad a \leqslant x \leqslant b$$

For $h(x) \equiv 0$ the equation is of the first kind.

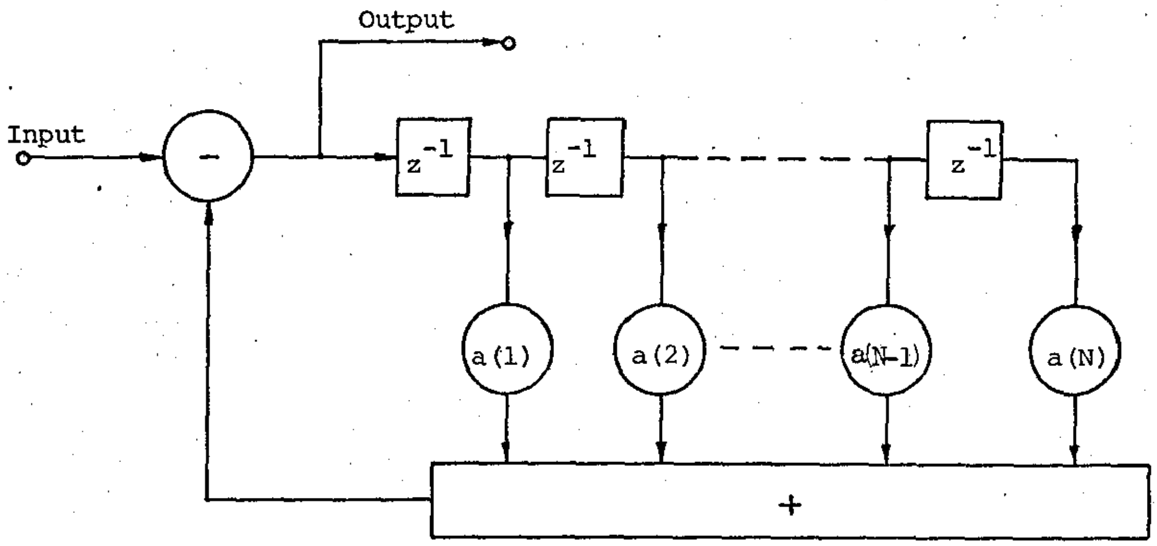


Fig. 8.1 Recursive digital filter, $1/A_{min}(z)$.

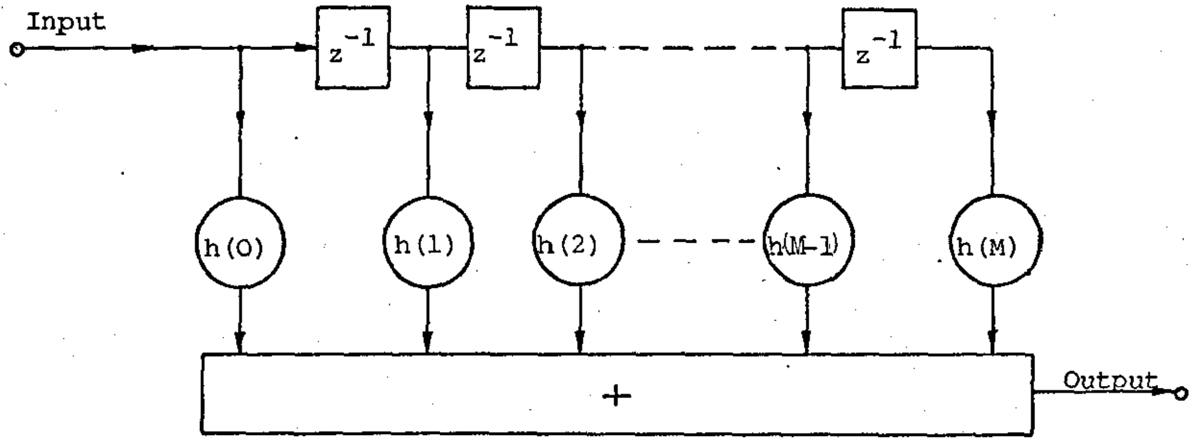


Fig. 8.2 Transversal digital filter approximating $1/A_{min}(z)$.

In order that the filter can be realized, some approximation must be made to accomplish inverse filtering. Instead of the exact inverse, a solution is sought such that

$$a(n) * h(n) = g(n-m) \tag{8.6}$$

where $g(n)$ is the output signal specified to be narrow peaked. It is shown below that under certain conditions, and provided m is large enough, a good approximation to $\delta(n-m)$ can be obtained which is physically realizable.

One approach which can be taken to obtain a physically realizable filter is to factorize A(z) into its minimum delay and maximum delay polynomials, denoted by $A_{min}(z)$ and $A_{max}(z)$ respectively. (It is assumed that no zeros are of unit magnitude). The minimum delay part of the inverse filter $1/A_{min}(z)$ can be readily realized, since all its zeros are inside the unit circle. The maximum delay factor

$$1/A_{max}(z) = 1/\sum_{n=o}^{s} a'(n)\ z^{-n} \tag{8.7}$$

can be expanded into a convergent series in positive powers of z by long division of

$$1/A_{max}(z) = z^{s}(\sum_{n=o}^{s} a'(s-n)\ z^{n})^{-1} \simeq z^{s} \sum_{n=o}^{m} h'(n)\ z^{n} \tag{8.8}$$

If the series is truncated after the mth term and delayed by $z^{-(m+s)}$, an approximate realizable transfer function delayed by (m+s)T seconds is obtained. Hence for reasonably large m

$$A(z)\ H(z) \simeq z^{-(m+s)} \tag{8.9}$$

Although in principle the minimum delay part $1/A_{min}(z)$ can be realized by a recursive (feedback) filter it is not normally used because of the risk of instability. Therefore, the inverse filter is composed of two cascaded transversal filters each of which is truncated at a suitable point. This approximation will cause errors preceeding the main pulse, due to the truncation of $1/A_{max}(z)$, and errors trailing the main pulse, due to truncation of $1/A_{min}(z)$. It can be shown that an inverse filter with a total of (M+1) taps give rise to truncation errors of approximately the same order when

$$(|z|_{max})^{M-r} = (1/|z|_{min})^{r}$$

where $|z|_{min}$ is the smallest root > 1, and $|z|_{max}$ is the largest root < 1. However, to obtain the smallest error for a given number of taps it is necessary to try different combinations of the individual filter length, (M+1-r), and, (r+1), respectively.

As an example, consider the 17-element uniform complex code (Table 8.1), whose zero pattern is given in Fig. 8.3, as the input sequence. Cascading two 24-tap transversal filters corresponding to truncated versions of $1/A_{min}(z)$ and $1/A_{max}(z)$, respectively, results in an inverse filter whose zeros are shown in Fig. 8.4. Table 8.1 gives the weighting sequence of the inverse filters of lengths 31, 39, and 47. The peak sidelobe level and the decrease in SNR of these filters is given in Table 8.2. It is of interest to note that for a given input sequence there exists an optimum number of filter tap gains for a minimum loss in signal detectability. For example, the filter of length 47 in Table 8.2 is clearly superior in both sidelobe and detection performance as compared to the filter of length 39. Fig. 8.5 shows the normalized response of the inverse filter of length 47 with, for comparison, the matched filter response. Since some of the zeros of the ZT of the input sequence A(z) are relatively close to the unit circle (Fig. 8.1), large truncation errors occur (-15 dB) for filters of length $(M+1) \leq 39$.

This method of designing an inverse filter requires the factorization of A(z) which becomes increasingly onerous as the sequence length exceeds

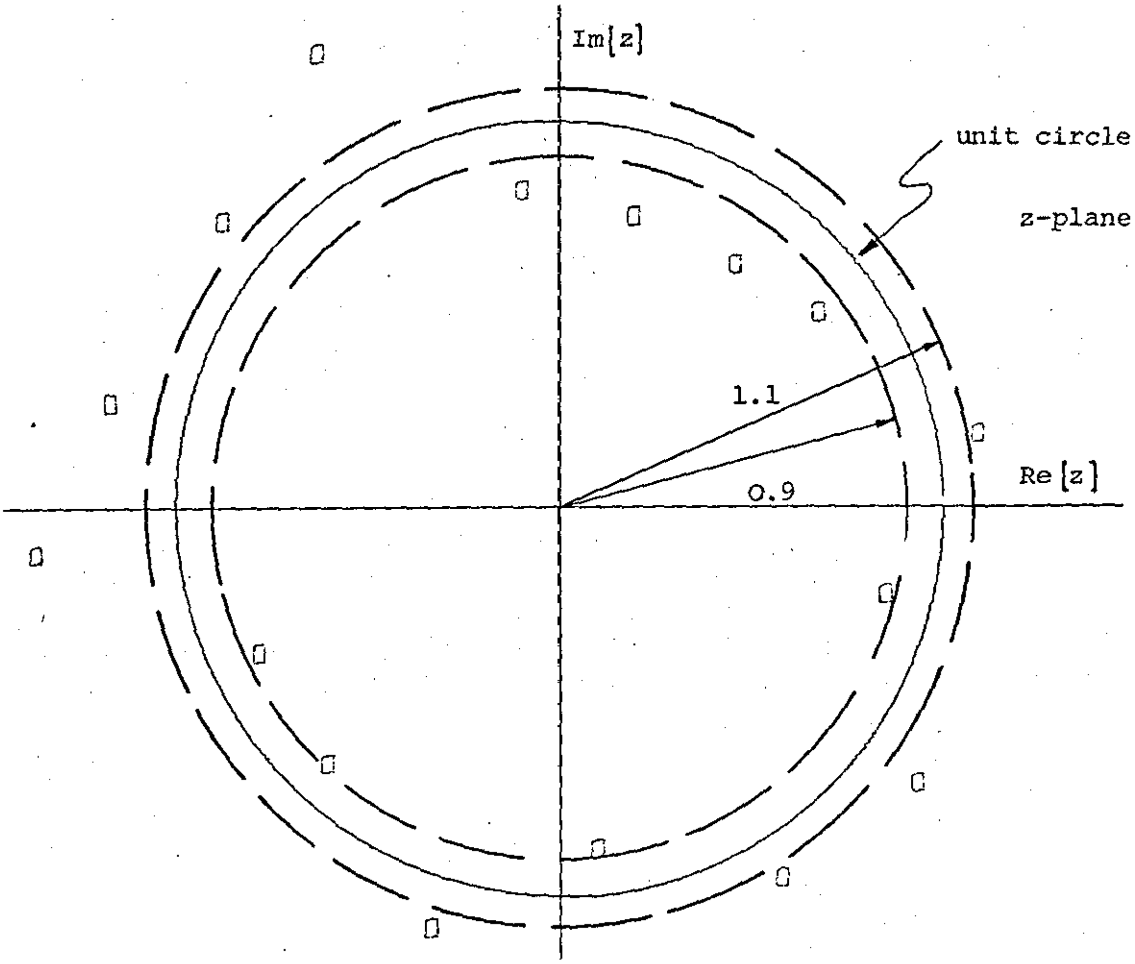


Fig. 8.3 Zero pattern of 17-element input sequence.

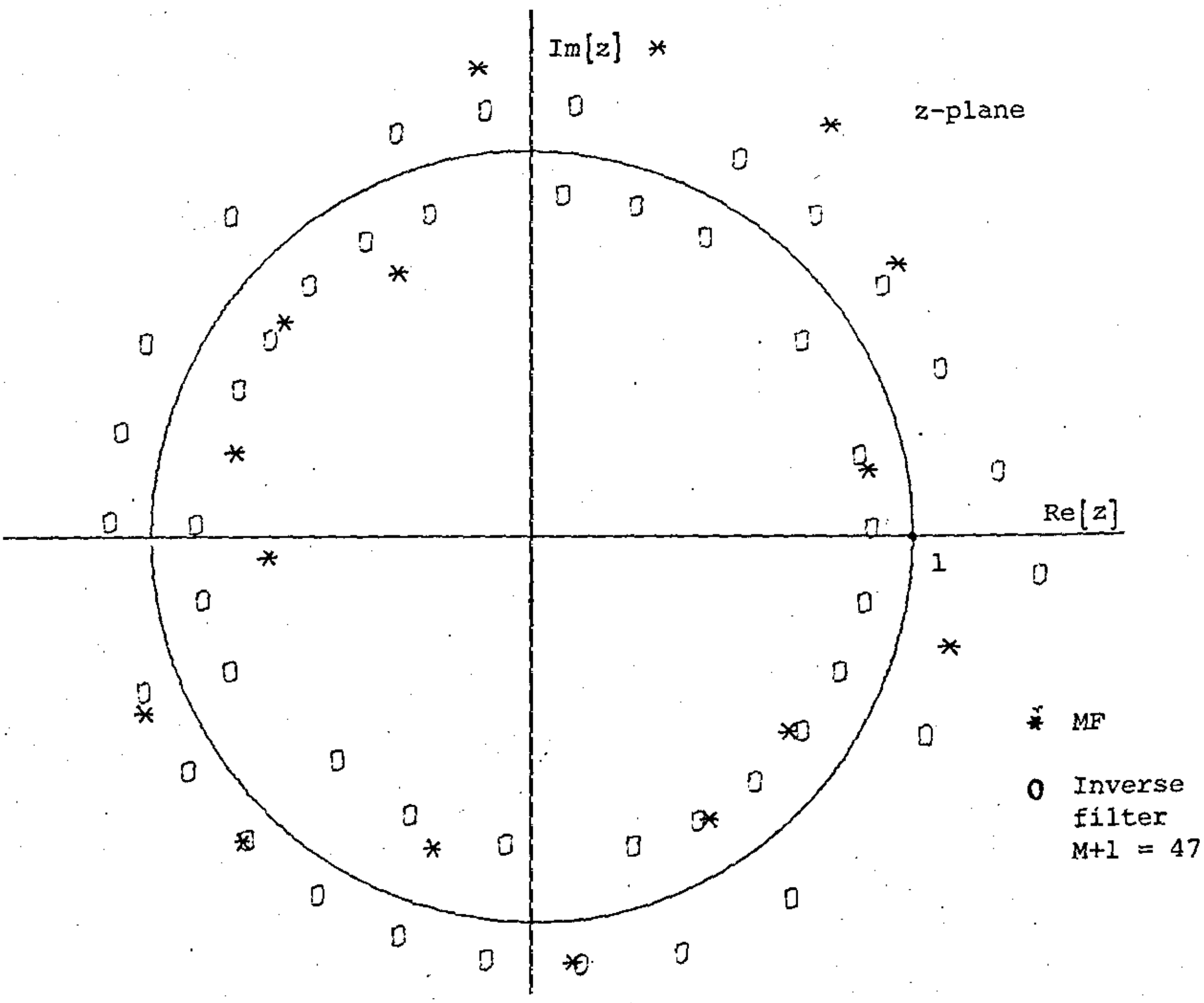


Fig. 8.4 Zero pattern of inverse filter and MF.

Input sequence of length N+1 = 17		Inverse filters of length					
		M+1 = 31		M+1 = 39		M+1 = 47	
Re [a(n)]	Im [a(n)]	Re [h(n)]	Im [h(n)]	Re [h(n)]	Im [h(n)]	Re [h(n)]	Im [h(n)]
0.216	0.110	-0.008	-0.053	0.061	0.015	-0.060	-0.005
0.227	0.085	0.144	0.057	-0.052	0.004	-0.049	-0.000
-0.158	0.184	0.007	0.019	-0.051	-0.085	0.010	0.011
-0.166	0.177	-0.123	0.000	-0.039	-0.054	-0.011	-0.023
-0.048	-0.238	0.007	-0.099	0.012	-0.015	0.025	-0.012
0.243	0.002	-0.021	-0.062	0.092	-0.026	-0.014	-0.013
0.186	-0.155	0.098	0.005	-0.019	-0.059	-0.008	-0.062
-0.204	-0.131	0.244	0.147	-0.024	-0.036	-0.028	0.009
0.242	-0.017	0.047	-0.209	0.030	-0.023	-0.015	-0.008
0.215	-0.113	0.232	-0.090	-0.039	0.025	0.057	-0.032
0.038	-0.240	-0.232	0.036	0.037	-0.014	-0.008	-0.059
-0.120	-0.211	0.134	-0.147	0.206	0.142	-0.017	-0.044
0.169	0.174	-0.127	0.234	0.048	-0.210	0.031	-0.026
-0.240	-0.037	0.031	0.197	0.225	-0.097	-0.031	0.019
0.227	0.085	0.239	0.149	-0.223	0.029	0.041	-0.011
0.031	0.241	0.289	0.018	0.144	-0.149	0.211	0.148
0.188	-0.153	-0.206	0.111	-0.121	0.247	0.051	-0.213
		0.212	0.214	0.036	0.208	0.221	-0.099
		0.194	-0.019	0.249	0.150	-0.223	0.032
		-0.055	0.284	0.286	0.018	0.144	-0.150
		-0.095	-0.131	-0.208	0.117	-0.122	0.243
		-0.198	-0.213	0.217	0.209	0.037	0.205
		0.256	-0.061	0.193	-0.022	0.246	0.147
		0.138	-0.129	-0.061	0.281	0.284	0.019
		-0.025	0.078	-0.092	-0.129	-0.205	0.115
		0.028	-0.001	-0.204	-0.210	0.217	0.209
		-0.012	0.010	0.259	-0.064	0.193	-0.023
		0.004	0.021	0.149	-0.122	-0.056	0.278
		-0.048	-0.071	-0.020	0.057	-0.094	-0.132
		-0.064	-0.062	0.026	-0.013	-0.203	-0.209
		-0.067	0.027	-0.028	0.014	0.261	-0.059
				0.004	0.021	0.154	-0.126
				-0.040	-0.066	-0.019	0.056
				-0.070	-0.036	0.027	-0.013
				-0.029	0.004	-0.028	0.013
				0.019	0.008	-0.001	0.024
				-0.035	-0.034	-0.042	-0.046
				0.031	0.036	-0.050	-0.034
				0.005	-0.013	-0.019	0.010
						0.012	-0.009
						-0.021	-0.021
						0.002	0.018
						0.000	0.004
						-0.018	0.029
						0.023	0.006
						-0.023	-0.013
						-0.008	0.034

Table 8.1 Weighting sequences of inverse filters.

Length of filter sequence	Detection loss	Sidelobe energy	Peak sidelobe
M+1	L_s(dB)	E_s(dB)	S_{max}(dB)
27	-0.825	-7.642	-14.780
29	-0.493	-10.664	-15.040
31	-0.548	-11.326	-14.951
39	-0.416	-16.437	-21.111
47	-0.338	-24.983	-30.485
49	-0.346	-22.811	-30.012
55	-0.352	-22.611	-27.652
63	-0.341	-26.642	-32.680
71	-0.335	-33.613	-39.451
79	-0.334	-39.232	-45.721

Table 8.2 Performance of inverse filters.

$$L_s(\text{dB}) = 10 \log(L_s),$$

$$E_s(\text{dB}) = 10 \log \Big(\sum_{\substack{n \\ n\neq j}} |g(n)|^2/|g(j)|^2\Big),$$

$$S_{max}(\text{dB}) = 20 \log \big(\max_{n\neq j}|g(n)|/|g(j)|\big),$$

(Input sequence 17-element uniform complex code (Table 8.1))

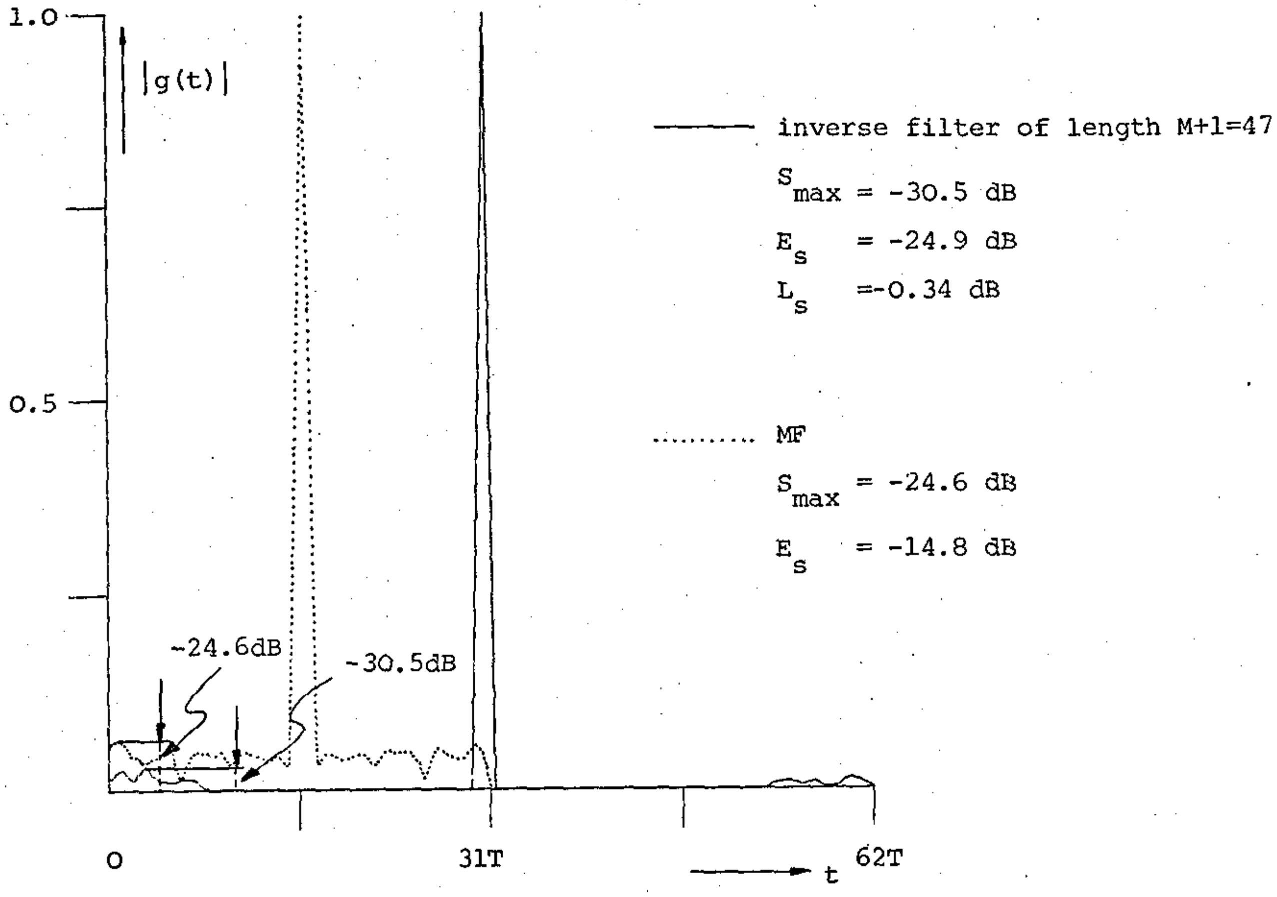


Fig. 8.5 Response of inverse filter and MF to 17-element uniform complex code (Table 8.1).

a hundred. In addition if one or more roots of A(z) lie close to or on the unit circle, that is the magnitude of any root z_j lies in the range $0.9 < |z_j| < 1.1$ (Fig. 8.1), an adequate inverse filter cannot be achieved in a reasonable number of taps. (In fact if one of the roots lies on the unit circle the ideal inverse filter will have an unbounded mean-square response to white noise). It can be seen from Eq. (7.10) that the zeros of the signals of interest tend to cluster around the unit circle as N increases, Fig. 4.19 and Fig. 6.6(b). Consequently, this method should only be applied to sequences where the zeros are well removed from the unit circle.

Another but similar approach of obtaining an approximate inverse filter follows from the Fourier transform relationship Eq. (8.5)

$$A(f)\ H(f) = 1$$

The spectrum A(f) is evaluated by computing the DFT of the sequence a(n) which has been augmented with a suitable number of K zeros, i.e.

$$\mathrm{DFT}\ \{(a(0),\ a(1),\ldots,a(N),\overbrace{0,0,\ldots0)}^{K}\} = (A(0),A(1),\ldots,A(N+K))$$

The reciprocal of the resulting Fourier coefficients, A(n), that is; (1/A(0),1/A(1),....,1/A(N+K)), where care is taken so as not to divide by zero, represent the sampled values of the transfer function of the inverse filter. By taking the IDFT, an aliased version of the ideal inverse filter sequence is obtained.

Inverse filters designed by either of these two methods described above result in a truncated or folded version of the ideal physically non-realizable weighting sequence. However, it is often required to achieve the best degree of sidelobe suppression for a given filter length. This can be accomplished through the use of the more sophisticated optimization techniques described subsequently.

8.3 Unconstrained Optimum Sidelobe Reduction Filters

The problem of finding the (M+1) unknown filter coefficients, h(n), for a given input sequence, a(n), of length (N+1) and for a specified output sequence, d(n), requires the solution of the set of (M+N+1) linear equations.

$$d(k) = \sum_{n=o}^{M} h(n)\, a(k-n) \qquad (8.10)$$

$$k = 0,1,2,\ldots.,(M+N)$$

where for convenience the energy of the input sequence is normalized to unity

$$E = \sum_{n=o}^{N} |a(n)|^2 = 1 \qquad (8.11)$$

Since there are more equations than unknowns no exact solution exists. Consequently, it is required to relax the notion of a solution for such an overdetermined problem. Thus, a solution is sought which is of the form

$$a(n) * h(n) = d(n) + \varepsilon(n) = g(n) \qquad (8.12)$$

where $\varepsilon(n)$ denotes an error or residue sequence representing the difference between the actual output g(n) and the desired output d(n). It is often convenient to represent linear systems of equations in compact matrix notation. The above equations may thus be written as

$$A\, \underline{h} = \underline{d} + \underline{\varepsilon} \qquad (8.13)$$

and

$$\underline{\varepsilon} = \underline{d} - \underline{g}$$

where $\underline{h}$, $\underline{d}$, $\underline{g}$, and $\underline{\varepsilon}$ represent the column vectors

$$\underline{h} = \text{col}\ (h(0),h(1),\ldots.,h(M))$$

$$\underline{d} = \text{col}\ (d(0),d(1),\ldots.,d(M+N))$$

$$\underline{g} = \text{col}\ (g(0),g(1),\ldots.,g(M+N))$$

$$\underline{\varepsilon} = \text{col}\ (\varepsilon(0),\varepsilon(1),\ldots.,\varepsilon(M+N))$$

and A is a (M+N+1) x (M+1) matrix whose ith column is given by

$$A_{i-1} = \text{col}\ (\overbrace{0,0,\ldots\ldots,0}^{i-1},a(0),a(1),\ldots\ldots,a(N),\overbrace{0,\ldots,0}^{M+1-i}) \tag{8.14}$$

$$i = 1,2,\ldots\ldots,M+1$$

The problem of finding the best approximation is to select a filter weighting sequence h(n) such that a functional $F(|\varepsilon(n)|)$ defined on the error sequence $\varepsilon(n)$ is minimized. Adopting the ℓ_p-norm from Chapter 5 the optimization problem may be formulated as

$$\min F_p = \sum_{n=o}^{M+N} |\varepsilon(n)|^p \qquad p \geqslant 1 \tag{8.15}$$

For p = 2 the classical minimum mean-square error criterion is obtained.

$$\min F_2 = \sum_{n=o}^{M+N} |\varepsilon(n)|^2 = \underset{\sim}{\tilde{\varepsilon}}\ \underset{\sim}{\varepsilon} = \|\underline{g} - \underline{d}\|^2 \tag{8.16}$$

where $\underline{\tilde{\varepsilon}}$ denotes the complex conjugate transpose of $\underline{\varepsilon}$.
The formulation of the problem in a least square sense has the great advantage that with the proper interpretation all the results of linear vector space theory become available.

Deconvolution filters designed on the minimum mean-square criterion have been widely used in the theory of prediction, spectral smoothing, and resolution of seismic data[45,97]. However, little attention seems to have been payed to using these filters for sidelobe suppression in radar systems[98]. A brief description of a slightly more general development of the minimum mean-square error filter is given below.

8.3.1 Minimum Mean-Square Filter

The formulation of the least squares approximation problem in complex vector spaces (Hermitean vector spaces) is basically the same as in Eucledian vector spaces. However, it should be noted that the scalar product of two vectors $\underline{a}$ and $\underline{b}$, in a Hermitean vector space satisfies the relationship

$$(\underline{a},\underline{b}) = (\underline{b},\underline{a})^*$$

Minimizing the quadratic form F_2 is equivalent to minimizing $\|\underline{g} - \underline{d}\|$ which is the length of the error vector $\underline{\varepsilon}$. The vector may be represented as a point in n-dimensional Hermitean vector space, and its length is the distance of this point from the origin. Furthermore, it can be seen from Eq. (8.14) that the (M+1) column vectors $\{A_i\}$ are linearly independent. Hence, the (M+N+1) x (M+1) matrix A is of rank (M+1). Since the output vector $\underline{g}$ is a linear combination of the (M+1) vectors $\{A_i\}$ it must lie in the (M+1)-dimensional subspace spanned by these vectors. Using the projection theorem[99], $\|\underline{\varepsilon}\|$ is minimum when $\underline{g}$ is the orthogonal projection of $\underline{d}$ onto the (M+1)-dimensional subspace. Therefore, the inner product of the (M+1) rows of A and the error vector $\underline{\varepsilon}$ must be zero

$$\tilde{A}\,\underline{\varepsilon} = \tilde{A}(\underline{g} - \underline{d}) = \tilde{A}(A\,\underline{h} - \underline{d}) = 0$$

$$\tilde{A}\,A\,\underline{h} = \tilde{A}\,\underline{d}$$

$$\underline{h} = (\tilde{A}\,A)^{-1}\,\tilde{A}\,\underline{d} \tag{8.17}$$

and

$$\underline{g} = A\,(\tilde{A}\,A)^{-1}\,\tilde{A}\,\underline{d} \tag{8.18}$$

The matrix $(\tilde{A}\,A)^{-1}\,\tilde{A}$ is known as the generalized inverse (Chapter 5) and $A(\tilde{A}\,A)^{-1}\,\tilde{A}$ is the projection matrix of the columns of A. The solution of Eq. (8.13) can be regarded as the discrete equivalent of the Wiener Hopf equation encountered in prediction and smoothing[97].

The (M+1) x (M+1) square matrix $R = \tilde{A} A$ has the elements

$$\{R_{iK}\} = \sum_{n=o}^{M+N} a(n-k)\ a^*(n-i) \qquad (8.19)$$

$$i = 0,1,\ldots,M$$

$$k = 0,1,\ldots,M$$

which are the autocorrelation coefficients r(i-k) of the input sequence a(n). The positive definite matrix R is Hermitean, that is $\tilde{R} = R$, since $r(k) = r^*(-k)$. Furthermore, all the elements on any diagonal of the autocorrelation matrix are equal and thus R is a Tőplitz matrix. This property can be exploited to simplify the calculation of R^{-1}, 45.

Similarly the column vector $\underline{c} = \tilde{A}\ \underline{d}$ having the elements

$$c(k) = \sum_{n=o}^{M+n} d(n)\ a^*(n-k) \qquad (8.20)$$

$$k = 0,1,\ldots,M$$

represents the crosscorrelation between the input sequence and the desired output sequence.

The desired output $\underline{d}$ may be specified as the ideal sequence consisting of a single unit spike delayed by jT seconds denoted as

$$\underline{d}_j = \text{col}\ (\overbrace{0,0,\ldots,0}^{j},1,\overbrace{0,\ldots,0,0}^{M+N-j}) \qquad (8.21)$$

In this case the crosscorrelation vector $\underline{c}$ becomes simply

$$\underline{c} = \tilde{A}\ \underline{d}_j = \text{col}\ (\tilde{A}_j) \qquad (8.22)$$

To obtain the optimum least square filter, the weighting sequence h(n) must be determined for each delay jT in the range from 0 to (M+N)T. However, it can be shown that for minimum delay input sequences the spike position j should be chosen to be zero while for maximum delay inputs the least energy error occurs when $j = (M+N)$. Consequently, for mixed delay inputs the spike position has to be chosen at some intermediate time delay. Moreover, for input sequences with good autocorrelation properties the diagonal elements of the matrix R become dominant. Therefore, it can be argued that the value for j should be between N and M.

From Eq. (8.12) and for sufficiently small values of $\|\underline{\varepsilon}\|^2$ the ZT of the output sequence becomes

$$A(z)\ H(z) = G(z) \simeq z^{-j} \tag{8.23}$$

The solution of Eq. (8.17) provides the filter coefficients which minimize the mean-square error. This criterion is, however, not necessarily equal to the sidelobe energy ratio

$$E_s = \sum_{\substack{n=o \\ n\neq j}}^{M+N} |g(n)|^2 \ / \ |g(j)|^2 \tag{8.24}$$

which represents the signal-to-clutter ratio and is a better measure of the resolution capability (see Eq. (5.26)). In a dense uncorrelated target environment, the self-clutter power at the filter output is proportional to E_s. However, it can be easily verified that the mismatched filter designed to minimize $\|\underline{\varepsilon}\|^2$ subject to the constraint

$$|g(j)| = 1$$

also minimizes E_s. Thus the filter coefficients h(n) obtained by solving Eq. (8.17) should be multiplied by a factor $1/g(j)$, to adjust the $(j+1)$th component of $\underline{g}$ to unity.

The minimum error energy for the optimum solution of Eq. (8.17) denoted by $\underline{\hat{h}}$ is given by

$$\|\underline{\varepsilon}\|^2_{\min} = \|\underline{d} - \underline{g}\|^2$$

$$= \tilde{\underline{d}}\,\underline{d} - \tilde{\underline{d}}\,\underline{g} - \tilde{\underline{g}}\,\underline{d} + \tilde{\underline{g}}\,\underline{g}$$

Substituting the expressions

$$\tilde{A}\, A\, \hat{\underline{h}} = \tilde{A}\, \underline{d} = \underline{c}$$

and

$$A\, \hat{\underline{h}} = \underline{g}$$

leads to

$$\|\underline{\varepsilon}\|^2_{\min} = \tilde{\underline{d}}\,\underline{d} - \tilde{\underline{c}}\,\hat{\underline{h}} \tag{8.25}$$

For $\underline{d} = \underline{d}_j$

$$\|\underline{\varepsilon}\|^2_{\min} = 1 - \tilde{\underline{c}}\,\hat{\underline{h}} \tag{8.26}$$

Therefore, $\hat{\underline{h}}$ has the same phase as $\underline{c}$ and since

$$\underline{c} = \tilde{A}\, \underline{d}_j = \text{col}\ (\tilde{A}_j)$$

$\hat{\underline{h}}$ will have the same phase as the MF for $N \leqslant j \leqslant M$. Using the above expressions the mismatch loss L_s becomes

$$L_s = \|A_j^T\, \hat{\underline{h}}\|^2 \ / \ \|\hat{\underline{h}}\|^2 = \frac{|A_j^T\ (\tilde{A}\ A)^{-1}\ \tilde{A}_j|^2}{\|(\tilde{A}\ A)^{-1}\ \tilde{A}_j\|^2} \tag{8.27}$$

8.3.2 Filter Design Examples

The advantage of using a minimum mean-square error criterion ($p = 2$), is that a unique solution for the optimum filter coefficients can be found by solving a set of linear equations

$$\partial F_2/\partial h^*(i) = 0 \qquad i = 0,1,2,\ldots,M$$

If the design objective is to minimize the self-clutter power, such a filter is optimum. However, in situations where the peak sidelobes introduce target masking problems a performance measure with $p > 2$ is preferable. Moreover, in many cases a minimax or Chebyshev solution ($p \to \infty$)

$$\min \quad F = |\varepsilon| \tag{8.28}$$

subject to

$$|\underline{g} - \underline{d}| < |\varepsilon|$$

may be desired. Near minimax solutions can be achieved using the least ℓ_p-approximation by successively minimizing the objective function, F_p, for increasing values of the index p, i.e. p = 2,4,6,...,etc. (Chapter 5). However, unless special precautions are taken to avoid ill-conditioning, the use of values greater than about ten for p is not practical.

For $p > 2$ the partial derivatives of the objective function

$$\nabla F_p = \sum_{n=o}^{M+N} \mathrm{Re}\{p|\varepsilon(n)|^{p-2} \varepsilon(n)\nabla\varepsilon^*(n)\} \tag{8.29}$$

are no longer linear functions in the variables h(n). Consequently, to find the optimum filter weighting sequence the objective function, F_p, has to be minimized using non-linear optimization techniques. A method which has proved to be quite efficient for this task is the pattern search algorithm (PAT) already available from Chapter 5.

In order to compare the sidelobe reduction filters designed for p = 2,4 it is supposed that the 17-element uniform complex code (Table 8.1) is applied to the digital filter. In addition, before proceeding with the optimization procedure, there are two variables that must be specified, namely, the length of the filter weighting sequence, M+1, and the output spike location, j. Therefore, the first problem to consider is that of determining the optimum spike position of the output sequence for a given filter length. Fig. 8.6 shows a plot of the peak sidelobe and sidelobe

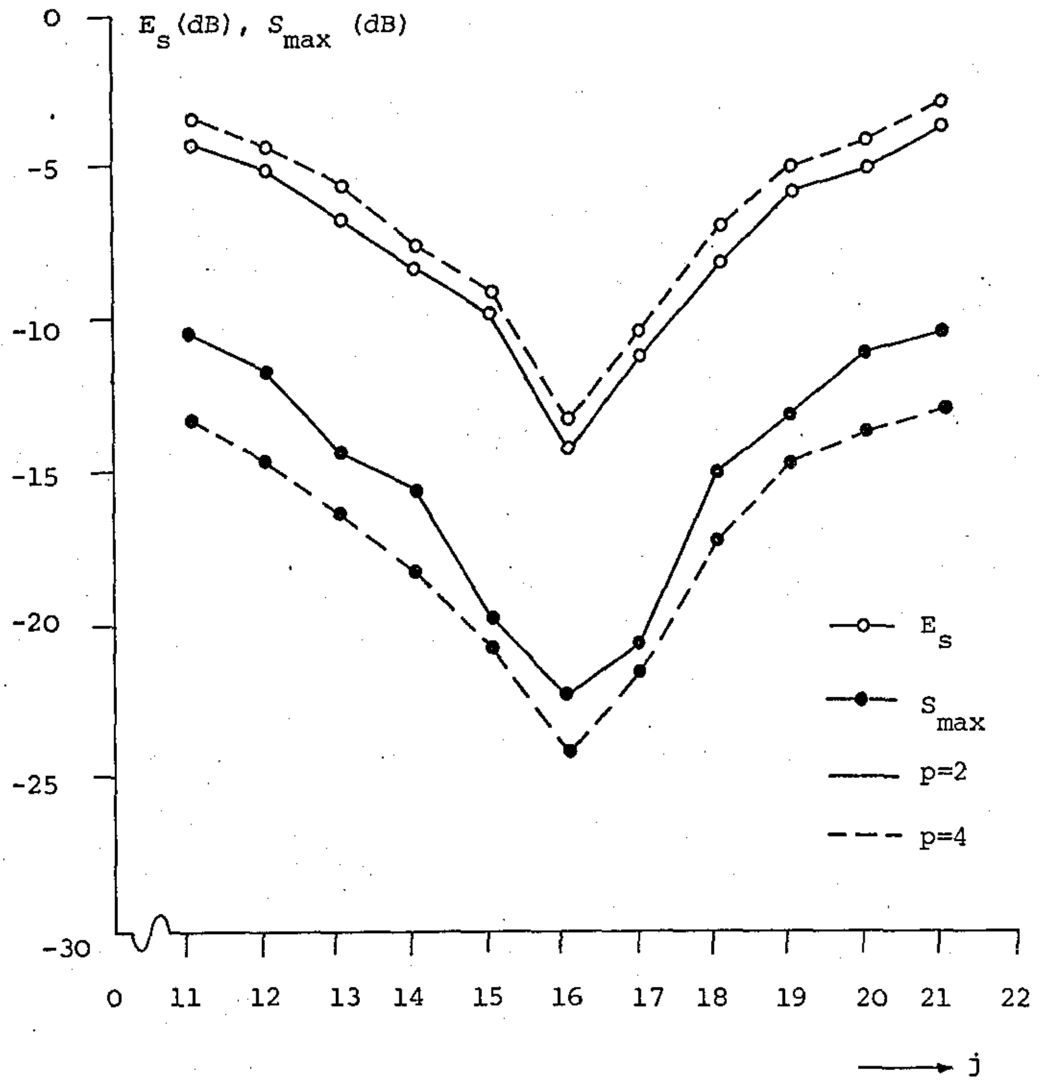


Fig. 8.6 Peak sidelobe and energy performance as a function of the spike position j for a MMF of length M+1 = 17.

energy ratio versus the spike position, j, for p = 2 and 4 and a filter length of M+1 = 17. For either of these two criteria the best output pulse location is for j = 16. The optimum filter coefficients which yield an output pulse at 16T seconds are given in Table 8.3. The range of the coefficients in both cases is of the same order of magnitude.

In Fig. 8.7 and Fig. 8.8 the peak sidelobe, the sidelobe energy and the detection loss are plotted on a decibel scale as a function of the filter length for p = 2 and 4. As an example, Table 8.4 presents the filter weighting sequences of length 32 for the two different error criteria. These filters have an optimum spike position at 25T seconds and give peak sidelobe levels of -30.4 dB and -31.5 dB respectively. Since the input sequence has already been optimised with respect to p = 4, the difference in performance of the MMF's for the two design objectives is small.

As mentioned previously the loss in SNR when a MF is replaced by a MMF is small provided the input signal has good autocorrelation properties. For example, a MMF of length 32 (Table 8.4) causes a detection loss of only 0.27 dB. Moreover, for p = 2 the MMF and the MF differ only in their gain characteristics. For p > 2, however, the filter weighting sequence has to be optimized in amplitude and phase. This is done by alternatively minimizing F_p with respect to the amplitude and phase, using the PAT algorithm, until the procedure converges to some optimum solution.

In summary, mismatching the receiver filter results in a small loss in SNR of the target return if the transmitted waveform has been optimized for matched conditions. However, to obtain improvements of about 6 dB in peak sidelobe suppression and 8 dB in clutter rejection over the MF, it is necessary to use a weighting sequence which is roughly

Mismatched filter of length M + 1 = 17			
p = 2		p = 4	
Re[h(n)]	Im[h(n)]	Re[h(n)]	Im[h(n)]
0.212	0.152	0.185	0.142
0.039	-0.217	0.031	-0.210
0.218	-0.088	0.205	-0.086
-0.223	0.031	-0.240	0.039
0.136	-0.155	0.140	-0.159
-0.130	0.245	-0.131	0.241
0.030	0.197	0.027	0.200
0.233	0.149	0.232	0.147
0.286	0.020	0.284	0.028
-0.204	0.111	-0.215	0.114
0.219	0.195	0.222	0.179
0.200	-0.027	0.211	-0.019
-0.060	0.273	-0.054	0.269
-0.105	-0.134	-0.134	-0.136
-0.197	-0.212	-0.195	-0.223
0.264	-0.071	0.249	-0.087
0.165	-0.124	0.179	-0.093
L_s = -0.124 dB E_s = -14.047 dB S_{max} = -22.376 dB		L_s = -0.089 dB E_s = -13.946 dB S_{max} = -24.041 dB	

Table 8.3 Mismatched filter weighting sequences for p = 2, 4 and optimum spike position j = 16.

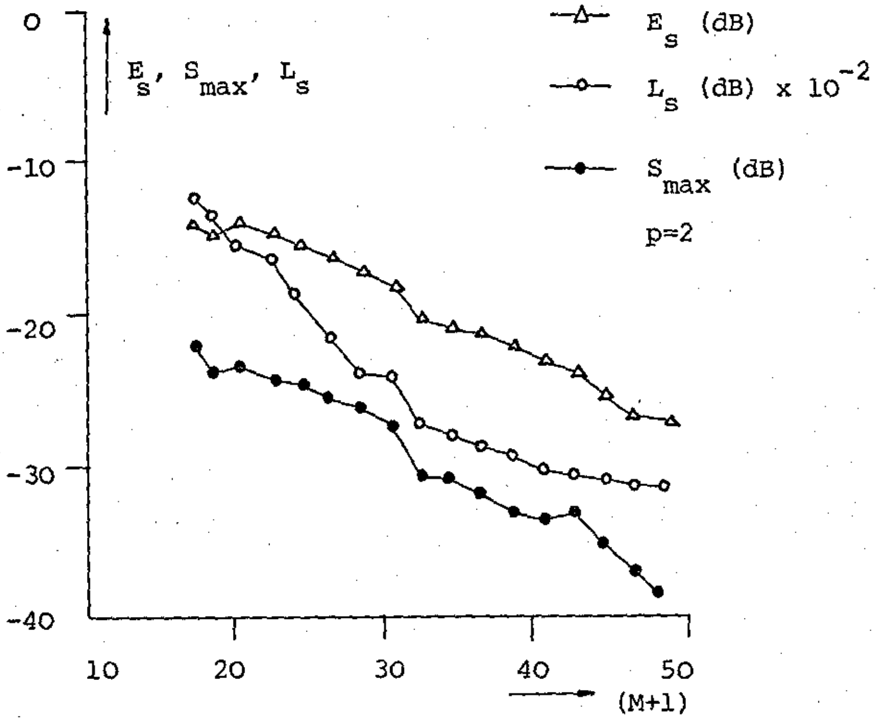


Fig. 8.7 Sidelobe and detection performance as a function of the filter length for p = 2.

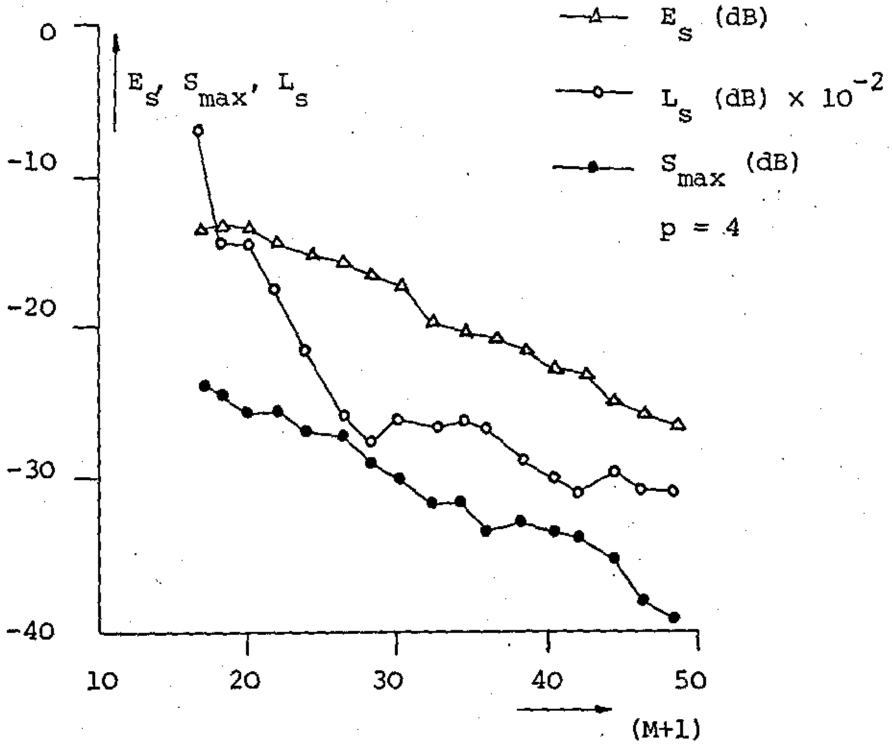


Fig. 8.8 Sidelobe and detection performance as a function of the filter length for p = 4.

Mismatched filter of length M + 1 = 32			
p = 2		p = 4	
Re[h(n)]	Im[h(n)]	Re[h(n)]	Im[h(n)]
-0.004	-0.065	0.002	-0.066
-0.029	0.006	-0.029	0.006
-0.008	-0.009	-0.009	-0.009
0.053	-0.032	0.044	-0.028
-0.009	-0.055	-0.012	-0.050
-0.117	-0.040	-0.015	-0.036
0.027	-0.022	0.018	-0.017
-0.031	0.020	-0.029	0.024
0.039	-0.010	0.033	-0.013
0.206	0.145	0.190	0.138
0.047	-0.210	0.042	-0.204
0.221	-0.098	0.218	-0.101
-0.225	0.033	-0.229	0.033
0.146	-0.152	0.148	-0.154
-0.126	0.241	-0.130	0.243
0.036	0.209	0.033	0.213
0.247	0.147	0.246	0.147
0.284	0.021	0.283	0.021
-0.205	0.116	-0.205	0.113
0.216	0.207	0.215	0.200
0.194	-0.022	0.202	-0.025
-0.056	0.277	-0.058	0.281
-0.097	-0.131	-0.104	-0.135
-0.202	-0.211	-0.201	-0.218
0.261	-0.062	0.259	-0.067
0.153	-0.125	0.147	-0.120
-0.019	0.059	-0.019	0.065
0.027	-0.015	0.033	-0.016
-0.033	0.015	-0.037	0.016
0.002	0.025	0.004	0.025
-0.043	-0.052	-0.048	-0.049
-0.053	-0.030	-0.055	-0.028
L_s = -0.275 dB E_s = -20.601 dB S_{max} = -30.408 dB		L_s = -0.266 dB E_s = -20.027 dB S_{max} = -31.594 dB	

Table 8.4 Weighting sequence of MMF for p = 2,4 and optimum spike position j = 25.

twice the length of the input sequence. These improvements can be obtained at a cost of about 0.3 dB in SNR. If the filter length is made larger its behaviour approaches that of an ideal inverse filter. However, increased filter length causes an additional degradation in noise performance. It is evident that the design of MMF's using ℓ_p-measures is clearly superior to the approach described in the previous section where a 47-tap filter was necessary to achieve the same sidelobe suppression (≃ -30 dB) as the MMF of length 32 and p = 4.

Sidelobe filters are particularly useful in situations where, for some reason, the use of an optimum waveform and a MF receiver is not possible. However, in these circumstances the degradation in SNR might be quite severe. Nevertheless, it is conceivable that a good compromise between improvement in resolution and degradation in detection performance can be found.

8.4 Constrained Mismatched Filters

Although the range sidelobes can be reduced to an arbitrary low level with a MMF of adequate length, the inevitable degradation in detection performance might be unacceptable. The noise enhancement is particularly aggravated in situations where, for some reason, it is not possible to transmit a waveform which is suitable for the task.

The reduction in noise performance of MMF's is caused by the enhancement of the spectrum outskirts. The resulting widening of the spectrum improves resolution but increases the output noise and thus degrades the SNR. The two conflicting requirements of achieving the best SNR and resolution at the output of a MMF can, therefore, not be met simultaneously. In any practical situation noise is always present and it would certainly be unrealistic to neglect the thermal noise when considering range sidelobe reduction techniques.

8.4.1 Design of Mismatched Filters in the Presence of Noise

For a system which is subjected to noise some performance measure for the MMF must be introduced which compromises the resolution and detection performance. This is equivalent to applying one or more constraints, reflecting the loss in SNR, to the filter weighting coefficients. In principle, the constrained ℓ_p-approximation problem may thus be formulated as finding a set of filter coefficients, h(n), which minimize a quadratic (Hermitean) form

$$Q = \underline{\tilde{h}}\ S\ \underline{h} \tag{8.30}$$

subject to the condition

$$F_p = \sum_{n=o}^{M+N} |\varepsilon(n)|^p = \text{const.} \tag{8.31}$$

If λ denotes an undetermined multiplier, the required solution may be obtained by minimizing

$$\min\ \{F_p + \lambda Q\} \tag{8.32}$$

The constrained minimum mean-square criterion, a special case of the above expression, is given by

$$F_2 + \lambda\ Q \tag{8.33}$$

Thus

$$\min\ \{\ \widetilde{(A\ \underline{h} - \underline{d})}(A\ \underline{h} - \underline{d}) + \lambda\ \underline{\tilde{h}}\ S\ \underline{h}\ \} \tag{8.34}$$

The solution of the quadratic constrained approximation problem is furnished by the set of linear equations.

$$\partial/\partial h^*(i)\{\ \widetilde{(A\ \underline{h} - \underline{d})}(A\ \underline{h} - \underline{d}) + \lambda\ \underline{\tilde{h}}\ S\ \underline{h}\ \} = 0 \qquad i = 0,1,2,\ldots,M \tag{8.35}$$

which reduces to

$$\partial/\partial h^*(i)\ \{\underline{\tilde{h}}\,\tilde{A}\,A\,\underline{h} - \underline{\tilde{d}}\,A\,\underline{h} - \underline{\tilde{h}}\,A\,\underline{d} + \lambda\,\underline{\tilde{h}}\,S\,\underline{h}\} = 0$$

$$i = 0,1,2,\ldots.,M$$

After some straightforward algebraic steps

$$\tilde{A}\,A\,\underline{h} - \tilde{A}\,\underline{d} + \lambda\,S\,\underline{h} = 0$$

or

$$\underline{h} = (\tilde{A}\,A + \lambda\,S)^{-1}\,\tilde{A}\,\underline{d} \tag{8.36}$$

It is noted that for $\lambda = 0$ the generalised least squares solution Eq.(8.17) is obtained.

The above expression is quite general in that nothing has as yet been assumed about the matrix S. Several quantities may be selected for S, but to be practically useful it must have some a *priori* plausibility[96]. It is evident that for this particular application S should reflect the degradation in SNR or at least should give an indication to what degree the system is affected by noise.

8.4.2 Choice of Performance Measure

A suitable choice for the quadratic form Q in white gaussian noise could be the detection loss factor L_s given by Eq. (8.1). Alternatively, a criterion which is equivalent to L_s is the sum of the square departures from matched conditions

$$Q = \sum_{n=o}^{M} |h(n) - a^*(j-n)|^2 = ||\underline{h} - \underline{a}_r^*||^2 \tag{8.37}$$

where $j \geqslant N$ is the spike position and $\underline{a}_r$ denotes the time reversed and shifted version of the vector $\underline{a}$

$$\underline{a}_r = \text{col}\ (\overbrace{0,0,\ldots.,0}^{j-N},\ a_N, a_{N-1},\ldots.a_o,\overbrace{0,0,\ldots,0}^{M-j})$$

As shown below both criteria lead to the same optimality conditions and the optimum is achieved when $\underline{h} = \underline{a}_r^*$.

The equivalence of the two criteria L_s and Q can be established by differentiating each criterion with respect to the filter coefficients h(n), i.e.

$$\partial/\partial h^*(i)(L_s) = \partial/\partial h^*(i) \left[\frac{\left|\sum_n h(n)\ a(j-n)\right|^2}{\sum_n |h(n)|^2} \right] = 0$$

$$i = 0,1,\ldots,M$$

which leads to

$$\frac{\left(\sum_n h(n)\ a(j-n)\right)^*}{\sum_n |h(n)|^2} = \frac{a^*(j-i)}{h(i)} \qquad i = 0,1,\ldots,M \tag{8.38}$$

It should be noted that the first criterion, L_s, gives a result which is independent of the filter gain. The second criterion, however, is gain dependent. Introducing an arbitrary gain factor α, the partial derivatives of Q are given by

$$\partial/\partial h^*(i)\ (Q) = \partial/\partial h^*(i)\ \left\{\sum_n |\alpha h(n) - a^*(j-n)|^2\right\} = 0$$

$$\alpha = \frac{a^*(j-i)}{h(i)} \qquad i = 0,1,\ldots,M \tag{8.39}$$

Hence, both criteria lead to the same optimality conditions if α is set equal to

$$\alpha = \frac{\left(\sum_n h(n)\ a(j-n)\right)^*}{\sum_n |h(n)|^2}$$

Adopting the latter criterion the original set of Eq.'s (8.13) may now be written in terms of the difference $(\underline{h} - \underline{a}_r^*)$ as the unknown vector and solved subject to the constraint that $\|\underline{h} - \underline{a}_r^*\|^2$ be a minimum. It can easily be verified that in this case S becomes the identity matrix I. Thus,

$$(\tilde{A}\, A + \lambda\, I)(\underline{h} - \underline{a}_r^*) = \tilde{A}(\underline{d} - A\, \underline{a}_r^*)$$

which leads to

$$\underline{h} = (\tilde{A}\, A + \lambda\, I)^{-1}\, (\tilde{A}\, \underline{d} + \lambda\, \underline{a}_r^*) \tag{8.40}$$

It can be seen from the above expression that the MF is approached for large values of λ.

There are several other suitable quantities which could be chosen for the quadratic form Q. For example, it might be required to minimize the power of the filtered noise, in which case Q becomes

$$Q = \tilde{\underline{h}}\, \tilde{V}\, V\, \underline{h}$$

where $S = \tilde{V}\, V$ is the autocorrelation matrix of the input noise samples[45].

Another interesting interpretation for the matrix S is obtained from duality considerations in time and frequency.

$$g(t) = \int_{-\infty}^{\infty} a(t,\tau)\; h(\tau)\; d\tau$$

$$g(t) = \int_{-\infty}^{\infty} A(f,t)\; H(f)\; df$$

Hence the original Eq's. (8.13), repeated here for convenience

$$A\, \underline{h} = \underline{d} + \underline{\varepsilon} = \underline{g}$$

can be written as

$$A_f\, \underline{H} = \underline{d} + \underline{\varepsilon}' = \underline{g} \tag{8.41}$$

$$\underline{h} = Y\, \underline{H} \tag{8.42}$$

where the (M+1) x (M+1) square matrix Y is the IDFT matrix given by Eq. (2.18). The column vector $\underline{H}$ thus represents the (M+1)-point DFT of $\underline{h}$. It follows from the above expression that the (M+N+1) x (M+1) matrix A_f is given by

$$A_f = A\ Y \tag{8.43}$$

Adopting a quadratic from such as

$$Q = \tilde{\underline{H}}\ B\ \underline{H} \tag{8.44}$$

results in a constrained least squares solution for the Fourier coefficients H(n)

$$\underline{H} = (\tilde{A}_f\ A_f + \lambda\ B)^{-1}\ \tilde{A}_f\ \underline{d} \tag{8.45}$$

Substituting Eq. (8.43) into Eq. (8.45) leads to

$$(\tilde{Y}\ \tilde{A}\ A\ Y + \lambda\ B)\ \underline{H} = \tilde{Y}\ \tilde{A}\ \underline{d}$$

$$(\tilde{A}\ A\ Y + \lambda\ \tilde{Y}^{-1}\ B)\ \underline{H} = \tilde{A}\ \underline{d}$$

and with

$$\underline{H} = Y^{-1}\ \underline{h}$$

yields

$$\underline{h} = (\tilde{A}\ A + \tilde{Y}^{-1}\ B\ Y^{-1})^{-1}\ \tilde{A}\ \underline{d} \tag{8.46}$$

Hence

$$S = \tilde{Y}^{-1}\ B\ Y^{-1}$$

and since $Y^{-1} = Y^*,\ Y^T = Y$

$$S = Y\ B\ Y^*$$

The weighting matrix S can thus be regarded as a unitary transformation (rotation) of a filtering matrix B. This relationship could be useful in applications where the filter coefficients $\underline{h}$ should have certain spectral properties. For example, if clutter interference at low frequency (ground clutter) is to be suppressed, it could be envisaged using a diagonal matrix B, where the diagonal elements (eigenvalues of S) can be regarded as the rejection factors at that frequency. This is illustrated in Fig. 8.9.

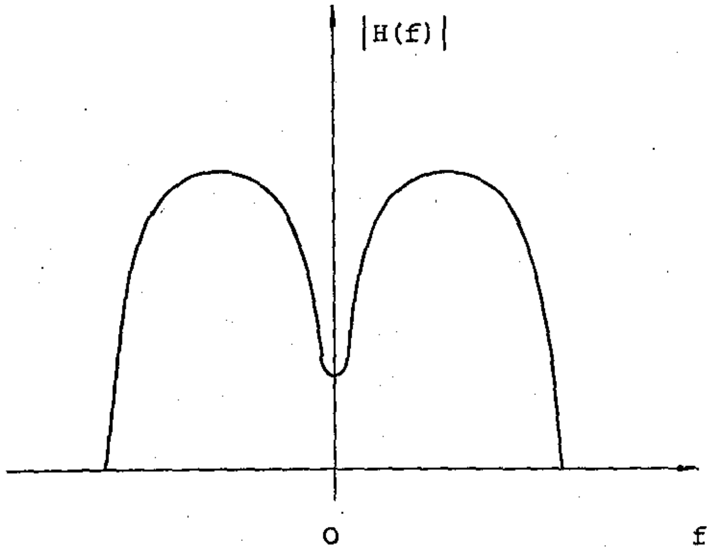
8.4.3 Relationship Between the Unconstrained and Constrained Mismatched Filter

So far the determination of λ has not been described. In principle, the parameter λ is implicitly determined once a specified value, e, has been assigned to $\|\underline{\varepsilon}\|^2$. However, in practice it is usually necessary to obtain solutions for several values of λ. This is done in an iterative manner by varying λ until a solution is obtained for which

$$\|\underline{\varepsilon}\|^2 \approx e \quad .$$

Since there may be limits to the acceptable loss in SNR, the iterative process should be initiated by choosing an appropriate value of λ, keeping in mind that for large values of λ ($\lambda > 2$) the MF is obtained at each iteration. Hence, for large λ's the filter departs slowly from matched conditions. The rate of departure is controlled by the rate of decrease of λ at each iteration. In Fig. 8.10 the peak sidelobe, sidelobe energy and the loss factor are plotted as a function of λ for a MMF of length 32. The constrained solution for $p > 2$ can be obtained in a similar manner by minimizing (8.32) using the PAT algorithm.

There exists a simple relationship between the solution of the unconstrained and constrained mean-square problem denoted by $\underline{h}$ and $\underline{h}'$ respectively. It is clear from linear vector space theory[99] that any vector lying in the subspace spanned by the (M+1) column vectors $\{A_i\}$ can be expressed as a linear combination of the orthogonal and linearly independent eigenvectors $\underline{e}_i$, $i = 0,1,2,\ldots,M$ of the positive definite



$$B = \begin{bmatrix} 1.0 & & & & & & & & & & \\ & 1.0 & & & & & & & \mathbf{0} & & \\ & & 0.9 & & & & & & & & \\ & & & 0.7 & & & & & & & \\ & & & & 0.5 & & & & & & \\ & & & & & 0.1 & & & & & \\ & & & & & & 0.0 & & & & \\ & & & & & & & \ddots & & & \\ & & & & & & & & 0.0 & & \\ & & \mathbf{0} & & & & & & & 1.0 & \\ & & & & & & & & & & \ddots \\ & & & & & & & & & & & 1.0 \end{bmatrix}$$

Fig. 8.9 Filter transfer function and corresponding filtering matrix.

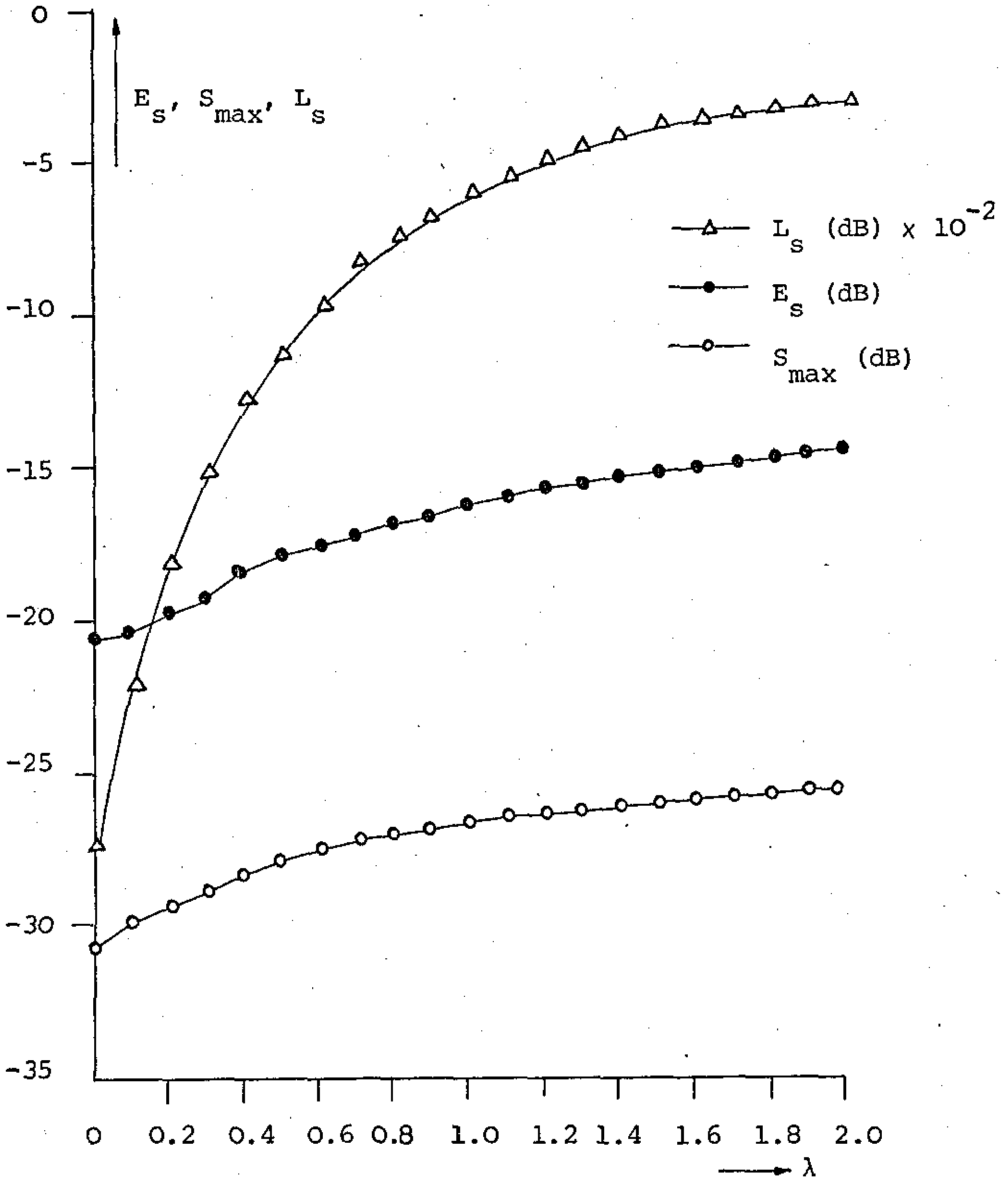


Fig. 8.10 Constrained MMF as a function of λ
(Filter length M+1 = 32).

matrix $\tilde{A}$ A. Thus

$$\underline{h} = \xi_o \underline{e}_o + \xi_1 \underline{e}_1 + \ldots\ldots + \xi_M \underline{e}_M = \sum_{i=o}^{M} \xi_i \underline{e}_i \tag{8.47}$$

$$\underline{h}' = \xi_o' \underline{e}_o + \xi_1' \underline{e}_1 + \ldots\ldots + \xi_M' \underline{e}_M = \sum_{i=o}^{M} \xi_i' \underline{e}_i \tag{8.48}$$

Consequently,

$$(\tilde{A} A + \lambda S) \underline{h}' = (\eta_o I + \lambda S) \xi_o' \underline{e}_o + \ldots\ldots + (\eta_M I + \lambda S)\xi_M' \underline{e}_M \tag{8.49}$$

and

$$\tilde{A} A \underline{h} = \eta_o \xi_o \underline{e}_o + \ldots\ldots\ldots + \eta_M \xi_M \underline{e}_M \tag{8.50}$$

where the η_i are the eigenvalues of $\tilde{A}$ A.

If S becomes the identity matrix I, then the relationship between the coefficients ξ_i and ξ_i' is given by

$$(\eta_i + \lambda) \xi_i' = \eta_i \xi_i$$

or

$$\xi_i' = (\frac{1}{1 + \lambda/\eta_i}) \xi_i \tag{8.51}$$

Eq. (8.51) clearly shows the smoothing or filtering effect of the constraints. Components corresponding to small eigenvalues η_i are greatly suppressed, while components where $\eta_i >> \lambda$ are practically unaffected. This supports the previous observation (Fig. 8.10) that the properties of the constrained solution vary slowly with λ.

The solution of Eq. (8.36) requires matrix multiplication and inversion operations which could result in considerable computational effort for large dimensional problems (matrix inversions usually require a computation time proportional to $(M+1)^3$). However, a transformation of the least squares problem into the frequency domain in conjunction with the FFT algorithm, can offer an increase in computational speed and a reduction in computer storage requirements. In general the constrained least squares problem is of the form

$$\underline{h} = (\tilde{A}\ A + \lambda\ \tilde{C}\ C)^{-1}\ (\tilde{A}\ \underline{d} + \lambda\ \underline{u}) \qquad (8.52)$$

where $S = \tilde{C}\ C$ is a Hermitean matrix and $\underline{u}$ is a vector of interest.

It can be seen easily that the matrix products $\tilde{A}A\underline{h}$ and $\tilde{C}C\underline{h}$ perform a double convolution, since $A\underline{h}$ and $C\underline{h}$ represent convolution operations. Using the convolution theorem of basic transform theory, Eq. (8.52) can be written as

$$\{A^*(n)\ A(n) + \lambda\ C^*(n)\ C(n)\}\ H(n) = A^*(n)\ D(n) + \lambda\ U(n)$$

hence

$$H(n) = \frac{A^*(n)\ D(n) + \lambda\ U(n)}{|A(n)|^2 + \lambda|C(n)|^2} \qquad (8.53)$$

where A(n), C(n), D(n), U(n) and H(n) now represent the DFT's of the corresponding vectors (sequences) a(n), c(n), d(n), u(n) and h(n) respectively. The filter coefficients h(n) are simply given by taking the IDFT of Eq. (8.53). It should be pointed out, however, that the sequences have to be augmented with the proper number of zeros to avoid significant aliasing.

The main advantage of the frequency domain approach is the use of FFT methods for computational work and the possibility of computing filter weighting sequences of large length.

8.5 Summary

If processor complexity is not of overriding concern, the use of a MMF may be justified in those cases where improvements in clutter performance are of significant magnitude. In general, there exists a trade-off between resolution and detection performance which sets practical limits on the sidelobe suppression that can be obtained. However, the degradation in SNR is usually small if the input waveform is optimized for matched conditions. Moreover, as pointed out by Rihaczek[35] the approach of optimum waveform design for a MF receiver also implicitly solves the problem of waveform design for the optimum filter in the presence of clutter. Therefore, whenever possible it is preferable to transmit a wider spectrum to achieve the desired resolution rather than widening the spectrum of the receiver filter. In addition to achieve the full benefits of the MMF it is necessary to reduce the tolerances of the receiver gain and phase characteristics. Rummler[38] has shown that the r.m.s. of amplitude and phase errors must be kept within 0.2 dB and 1° respectively to maintain low sidelobe levels. In most applications, however, the suppression of the sidelobes much below 32 dB does not seem practical due to probable phase errors in the processor, transient effects and the degradation due to slight target-doppler shifts.

CHAPTER 9

COMBINED RANGE AND RANGE RATE RESOLUTION

9.1 Introduction

The principal aim throughout this work has been to design waveforms for a radar environment where the relative doppler spread of the targets is negligible. However, when there is significant doppler shift, the reflections from a target are no longer replicas of the transmitted waveform and the MF response in time and frequency has to be considered.

The following brief discussion of combined range and range rate (doppler) resolution of a signal is included for completeness. Detailed analyses of resolution in range and range rate of general types of waveforms can be found in many contemporary books[22,23]. Moreover, consideration will only be given to properties pertinent to the types of waveforms described in the preceeding chapters.

Although, in principle, the numerical optimization methods developed to design pulse trains having good range resolution can be extended to the more general case of range and doppler resolution, the computational effort involved for longer sequences is quite formidable even for modern computers. For this reason waveform synthesis for range and velocity resolution is commonly done by trial and judicious use of available information (e.g. ambiguity function).

It has been shown (Chapter 2) that the range resolution property of a signal depends on the shape of its spectrum envelope. Based on the time-frequency duality it can be argued, therefore, that resolution in range rate depends only on the envelope of the signal in the time domain. Consequently, combined range and velocity resolution depends on the complete waveform structure in time and frequency. Hence signals with good resolution in one parameter may perform very poorly when combined resolution in both parameters is required.

For combined resolution in range and velocity the waveform must be investigated in terms of the complete MF response in delay and doppler. This generalized response is given by Woodward's ambiguity function (ABF) which has already been introduced in Chapter 2 and is repeated here for convenience

$$|\chi(\tau,\nu)| = \left| \sum_{n=-\infty}^{\infty} s(nT + \tau)\; s^*(nT)\; e^{j2\pi\nu(nT + \tau)} \right| \tag{9.1}$$

It is noted that in the literature the terms $\chi(\tau,\nu)$, $|\chi(\tau,\nu)|$ and $|\chi(\tau,\nu)|^2$ are often used synonymously for the ABF.

The ABF plays a central part in the analysis of combined resolution. This is so because the width of the main response peak of the ABF serves as a measure for close-target visability in range-doppler, while the low-level response and subsidiary spikes give an indication of the self-clutter and target masking problem by mutual interference. Since the volume of the ABF over the entire (τ,ν)-plane is constant, the signal design problem for combined range and velocity resolution may therefore be regarded as shifting the unavoidable ambiguity (volume) to those parts of the (τ,ν)-plane where it causes least interference for a given environment and application.

Some of the general resolution properties of the various types of pulse trains considered previously are discussed subsequently using the ABF description.

9.2 Ambiguity Function of Pulse Trains

The ambiguity function (ABF) of a QP code given by

$$s(nT) = e^{-j\pi(W/T_s)(nT)^2} \tag{9.2}$$

can readily be obtained by substituting Eq. (9.2) into Eq. (9.1)

$$|\chi(\tau,\nu)| = \left| \sum_{n=-\infty}^{\infty} e^{-j\{2\pi(W/T_s)\tau nT - 2\pi\nu nT\}} \right| \qquad (9.3)$$

For a sequence of length N, ($T_s = NT$), and time shifts $\tau = kT$, Eq. (9.3) becomes

$$|\chi(kT,\nu)| = \left| \sum_{n=o}^{N-1-|k|} e^{-j\{2\pi(W/N)knT - 2\pi\nu nT\}} \right| \qquad (9.4)$$

The above summation is a geometric progression in n and can be written in closed form (see Eq. (4.16)),

$$|\chi\ (kT,\nu)| = \left| \frac{\sin\{\pi[(WT/N)k - \nu T](N-|k|)\}}{\sin\{\pi(WT/N)k - \nu T)\}} \right| \qquad (9.5)$$

Furthermore, if T is the Nyquist sampling interval (T=1/W), and if the doppler shift ν is expressed as multiples of 1/NT, Eq. (9.5) reduces to

$$|\chi(kT,\ell/NT)| = \left| \frac{\sin[\pi(k-\ell)(1 - |k|/N)]}{\sin[\pi/N\ (k-\ell)]} \right| \qquad (9.6)$$

For $\ell = 0$ the zero-doppler cross-section of the ABF is obtained

$$|\chi(kT,0)| = \left| \frac{\sin[\pi k(1 - |k|\ /N)]}{\sin(\pi k/N)} \right| \qquad (9.7)$$

while the zero-delay cross-section is given by letting $k = 0$

$$|\chi(0,\ \ell/NT)| = \left| \frac{\sin\ (\pi\ell)}{\sin(\pi\ell/N)} \right| = \begin{cases} N & ;\quad \ell = 0 \\ 0 & ;\ \text{otherwise} \end{cases} \qquad (9.8)$$

It is noted that the ABF is periodic along the doppler axes with a period of 1/T.

Since the effects due to delay and doppler appear as a difference term, $(k - \ell)$ in Eq. (9.6), it is not possible to separate targets along the line where $k = \ell$. This is illustrated in Fig. 9.1(a). Along the ridge $k = \ell$ in the (τ,ν)-plane reduces Eq. (9.6) to

$$|\chi(kT, \ell/NT)| = (1 - |k|/N)N \tag{9.9}$$

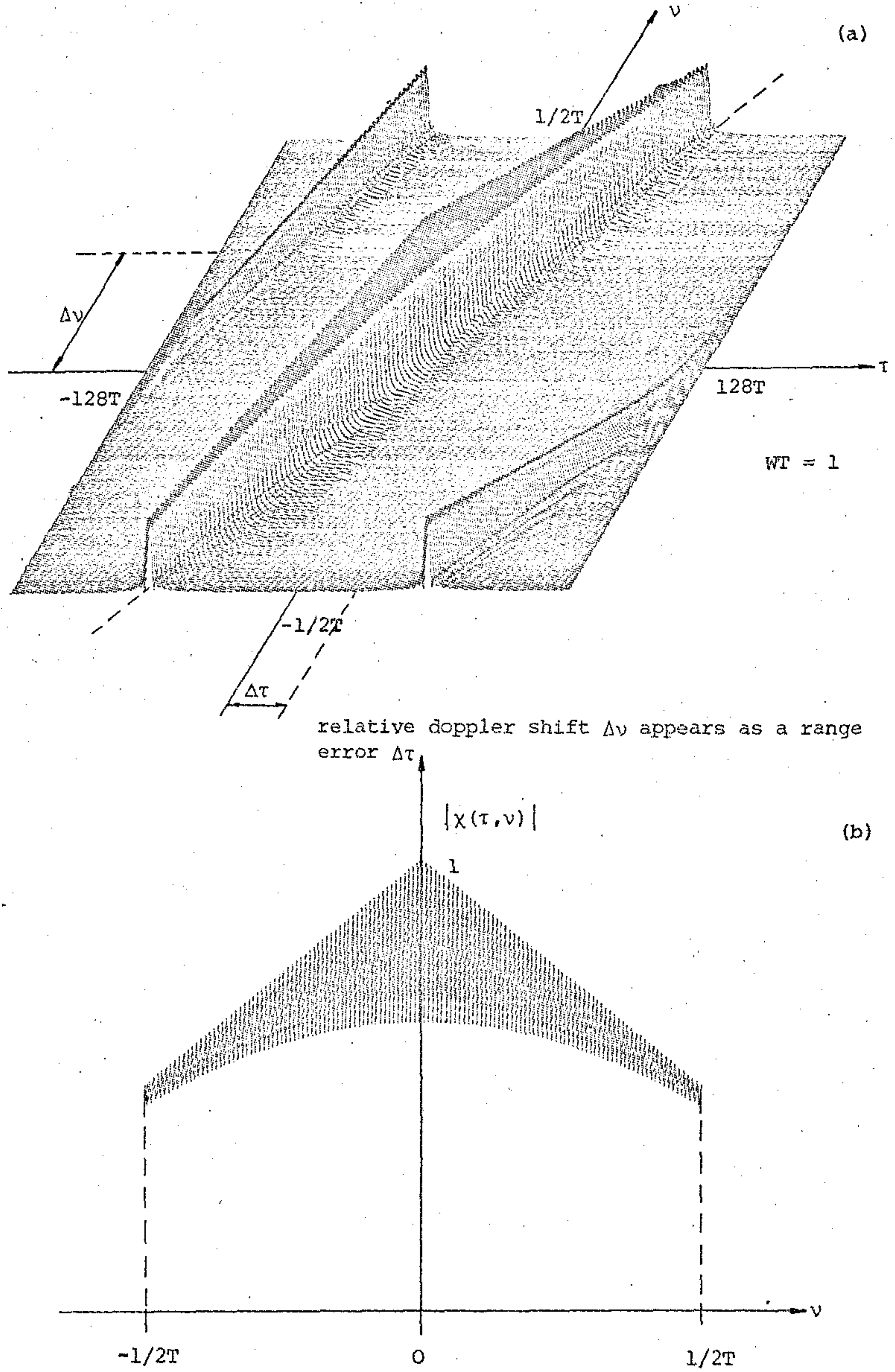
as shown in Fig. 9.1(b).

The diagonal ridge of the ABF of QP codes will almost certainly intersect extended clutter areas. However, since the clutter power output is proportional to the width of the ridge (3-dB points) this type of interference is minimized for high compression ratios (time-bandwidth products). In addition this type of ABF can be advantageous in the presence of small doppler shifts since there is little loss in clutter rejection as compared to other codes (see for example Fig. 9.2). Furthermore, the relatively slow decrease in the ACF for small doppler shifts (Fig. 9.1(b)) may result in considerable hardware savings. However, a doppler shift will be interpreted as a range error, as illustrated in Fig. 9.1(a).

Pulse trains whose ACF decreases at a faster rate for small doppler shifts are the non-linear FM type approximations discussed in Chapter 4. It can be seen from Fig. 9.2 to Fig. 9.4 that the ABF's of these waveforms still basically exhibit the ridge-like structure which suggests a relatively strong range-doppler coupling. However, the linear FM property is more and more eliminated as the order of the spectrum tapering, n, (Eq.(4.39)) increases.

For certain applications the inability to resolve targets in range and velocity along the ridge might be unacceptable. In these circumstances a signal whose ABF approaches that of a single strong spike (thumbtack) as shown in Fig. 9.5 and Fig. 9.6 might be adequate*.

*It is noted that sequences whose length N is not an integral power of two, i.e. $N \neq 2^m$, have been augmented with a suitable number of zeros for the use of FFT algorithms.



relative doppler shift Δν appears as a range error Δτ

Fig. 9.1 (a) ABF of 128-element QP code

(b) Peak response as a function of doppler shift ν.

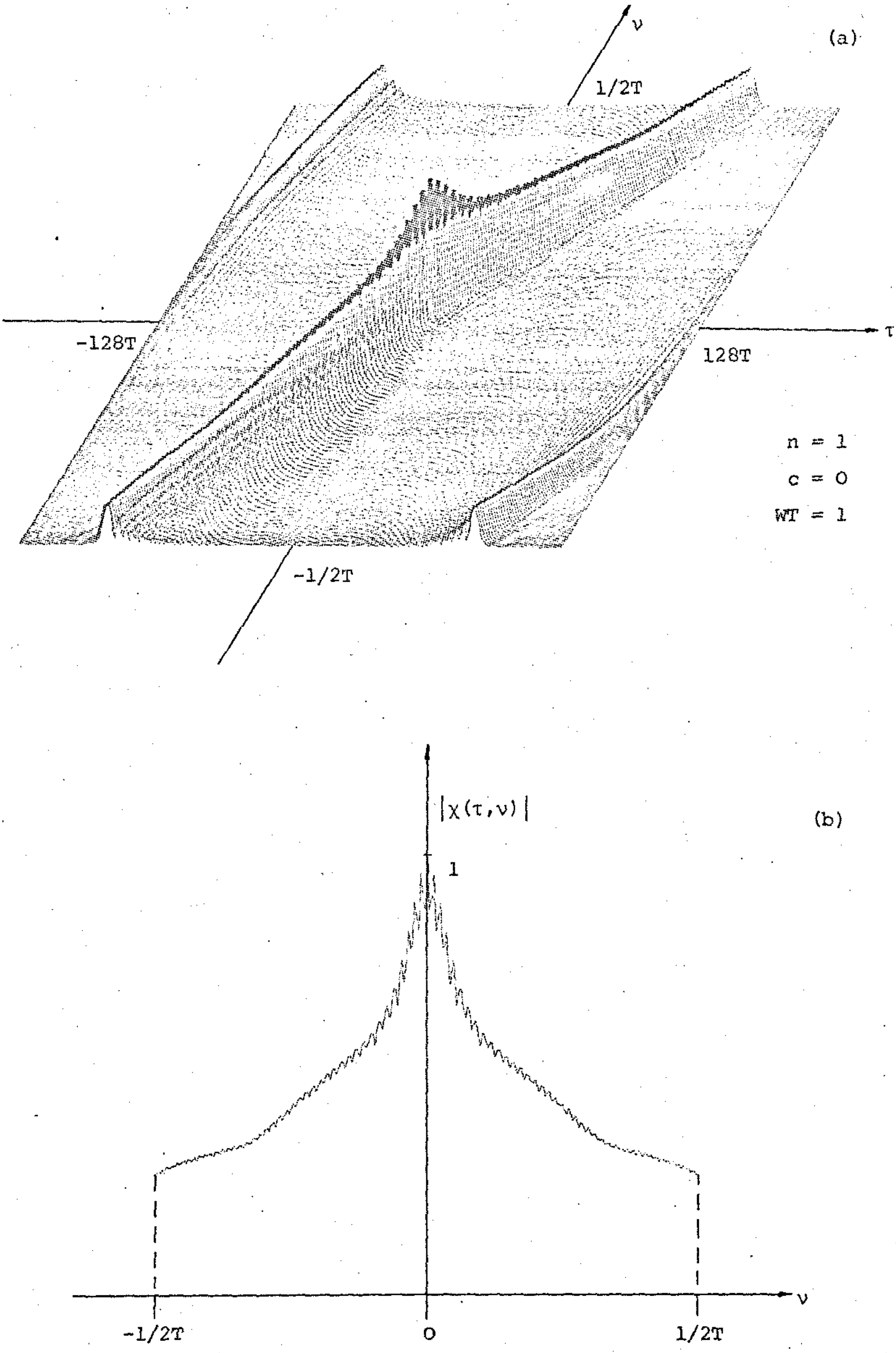


Fig. 9.2 (a) ABF of 128-element non-linear FM type pulse train for n = 1.
(b) Peak response as a function of doppler shift ν.

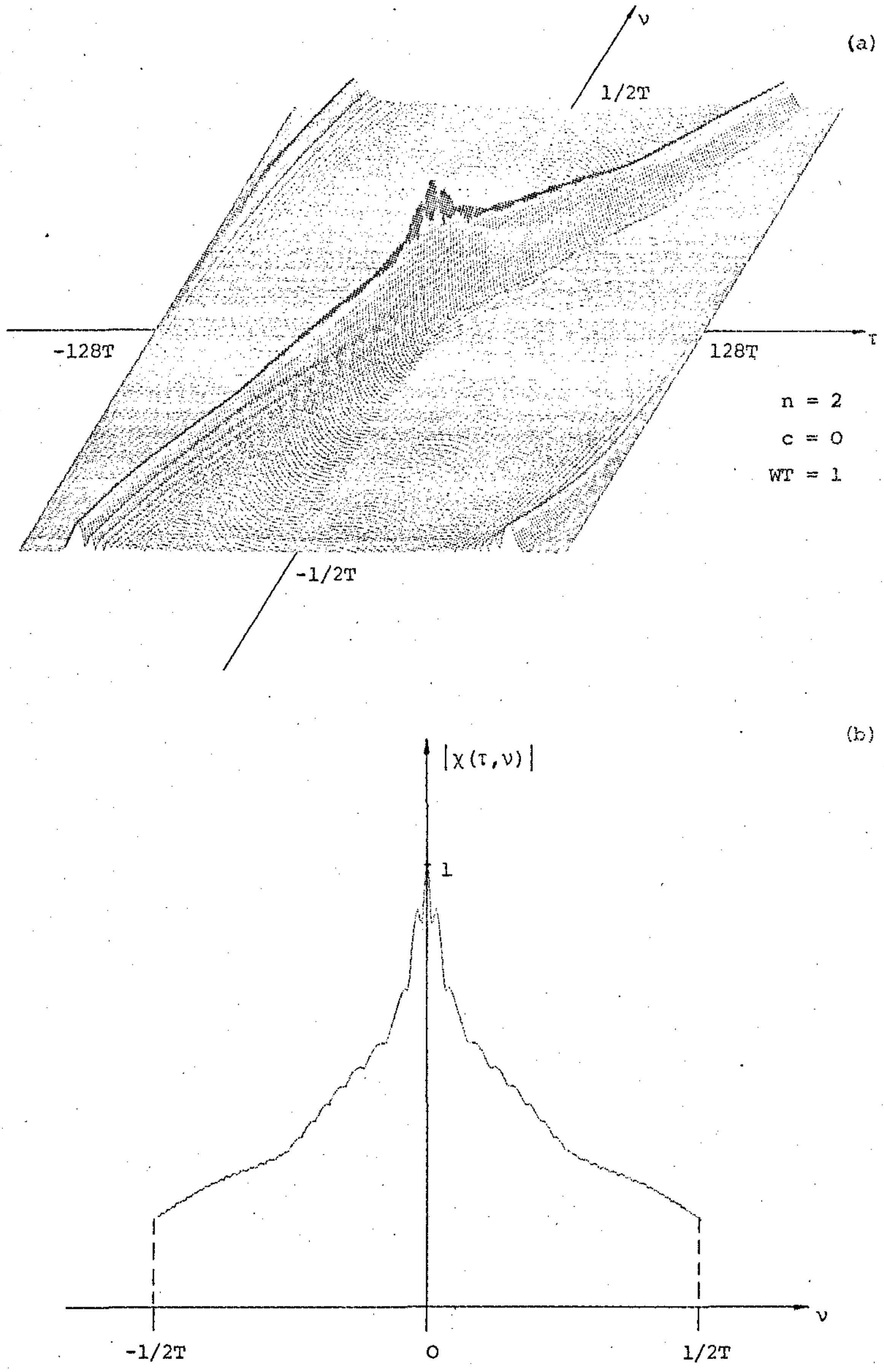


Fig. 9.3 (a) ABF of non-linear FM type pulse train for n = 2
(b) Peak response as function of doppler shift ν.

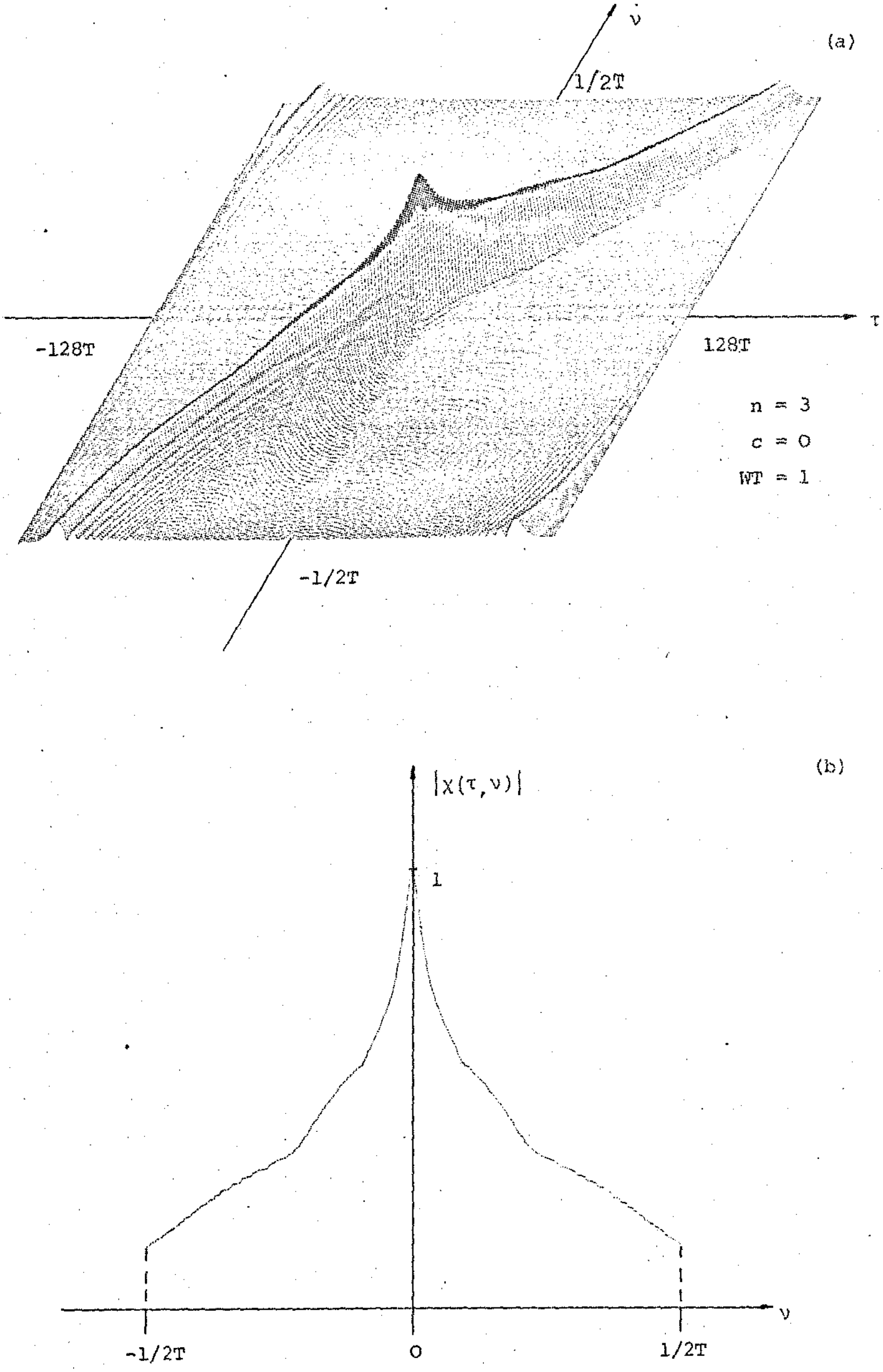


Fig. 9.4 (a) ABF of 128-element non-linear FM type pulse train for n = 3

(b) Peak response as a function of doppler shift ν.

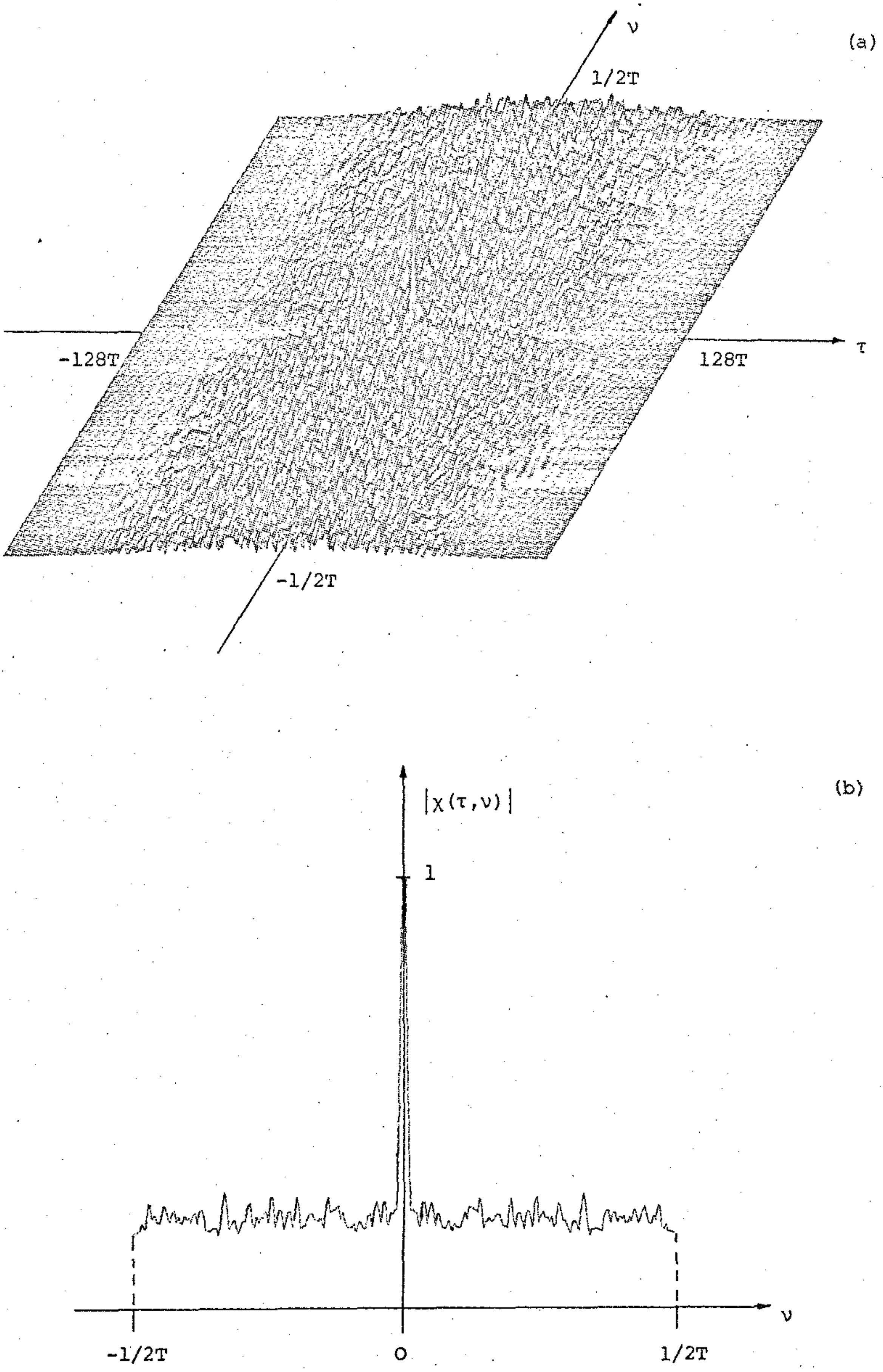


Fig. 9.5 (a) ABF of 108-element Huffman code derived from a random zero pattern.

(b) Peak response as a function of doppler shift ν.

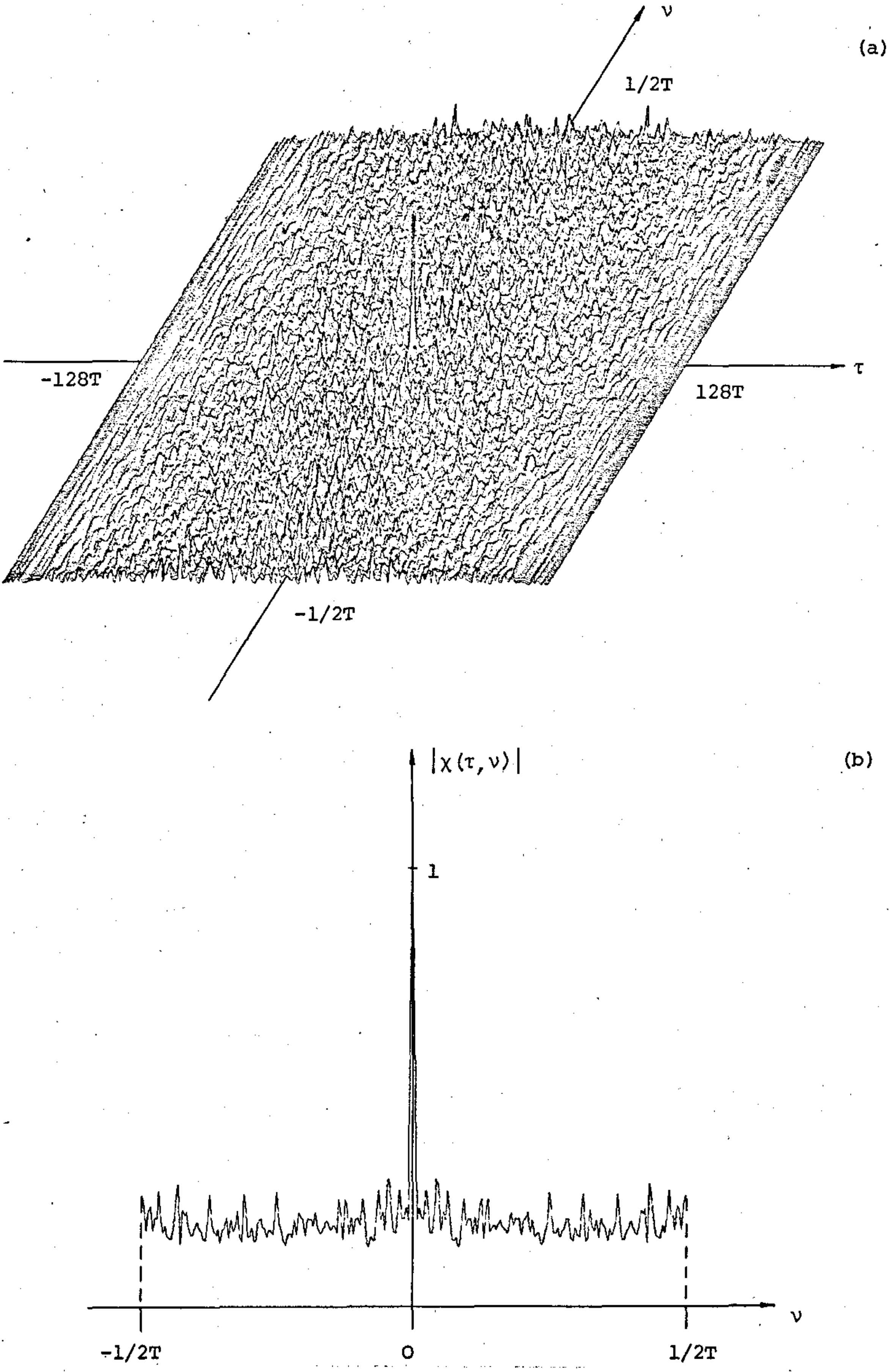


Fig. 9.6 (a) ABF of optimum 128-element binary sequence when the initial sequence is chosen randomly.

(b) Peak response as a function of doppler shift ν.

However, the expense of implementing many doppler channels may be prohibitive.

The choice of a thumbtack ABF may be justified for high close-target resolution in the absence of any prior information of the target environment. The close-target resolvability is, however, achieved at the expense of introducing self-clutter. Therefore, if the target space is confined to a narrow region, there is no reason to spread the volume of the ABF uniformly over the (τ,ν)-plane. Moreover, if visibility of small targets is of overriding importance, it is preferable to choose an ABF whose volume is concentrated in strong spikes or a narrow ridge. Such an ABF trades uniform poor visibility for weak targets (thumbtack ABF) against good visibility for most targets and extremely poor visibility for some targets.

Fig. 9.7 illustrates the relatively large increase in sidelobe levels off the delay axes for an optimum binary sequence of length 128 derived from a Vakman code. The residual diagonal ridges are not surprising, since the bit polarities of Vakman codes vary in a quadratic fashion (see Eq. 6.26). The concentration of the ambiguity along the ridges could be advantageous in certain clutter environments. However, resolution along these ridges is very poor. Fig. 9.6, on the other hand, shows the ABF of a 128-element binary code derived from a randomly chosen sequence. It can clearly be seen that noise-like waveforms, Fig. 9.5 and Fig. 9.6, are inherently suited to approximate thumbtack ABF's. In general, however, binary sequences are usually better suited to improving range resolution rather than velocity resolution.

The behaviour of the ABF of uniform complex codes is essentially no different from that of a binary sequence, Fig. 9.6, if the phase shifts are chosen randomly. However, if the initial sequence is a Frank code, for example, the resulting ABF will have the shape as shown in Fig. 9.8.

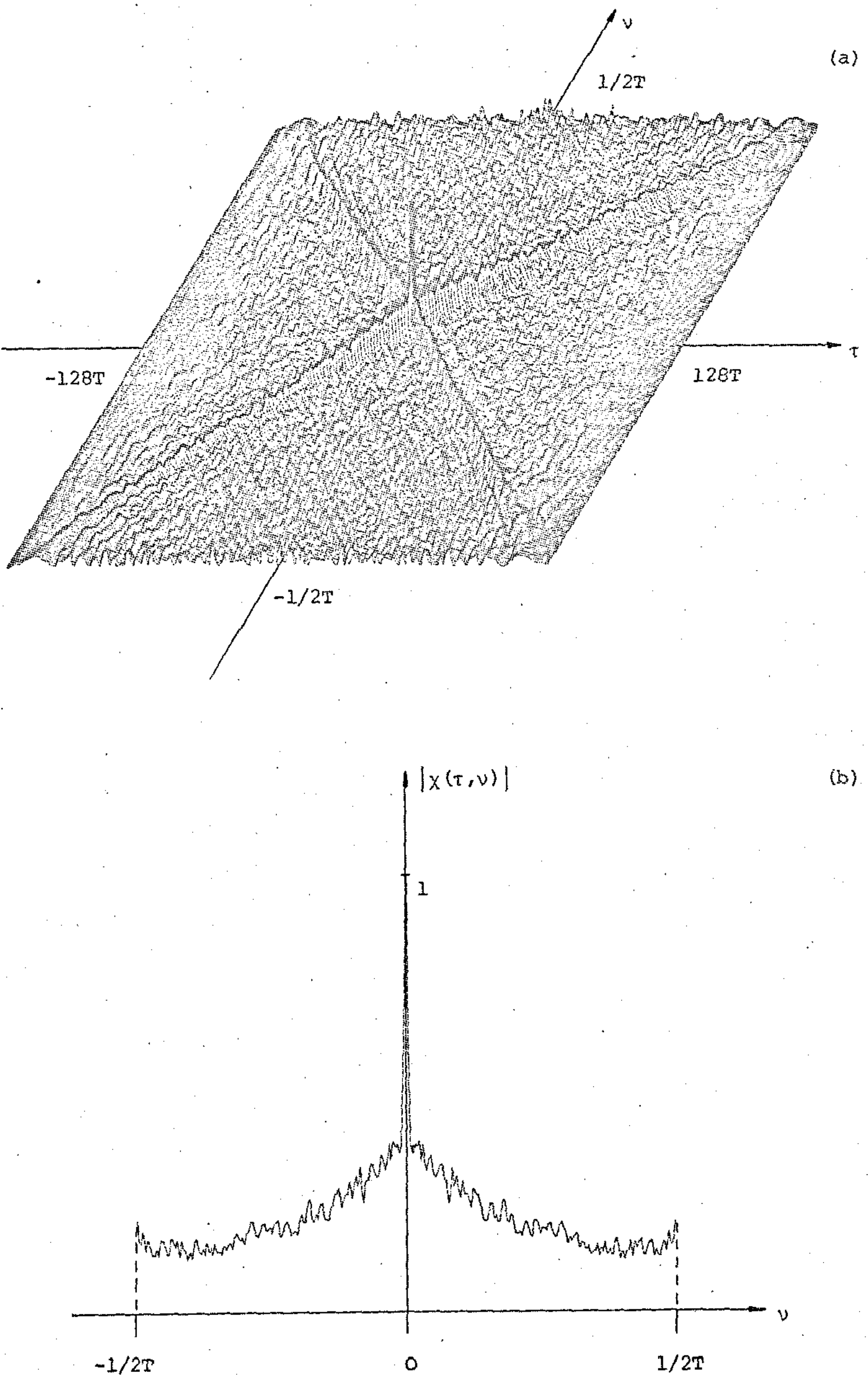


Fig. 9.7 (a) ABF of optimum 128-element binary sequence when the initial sequence is a Vakman code.

(b) Peak response as a function of doppler shift ν.

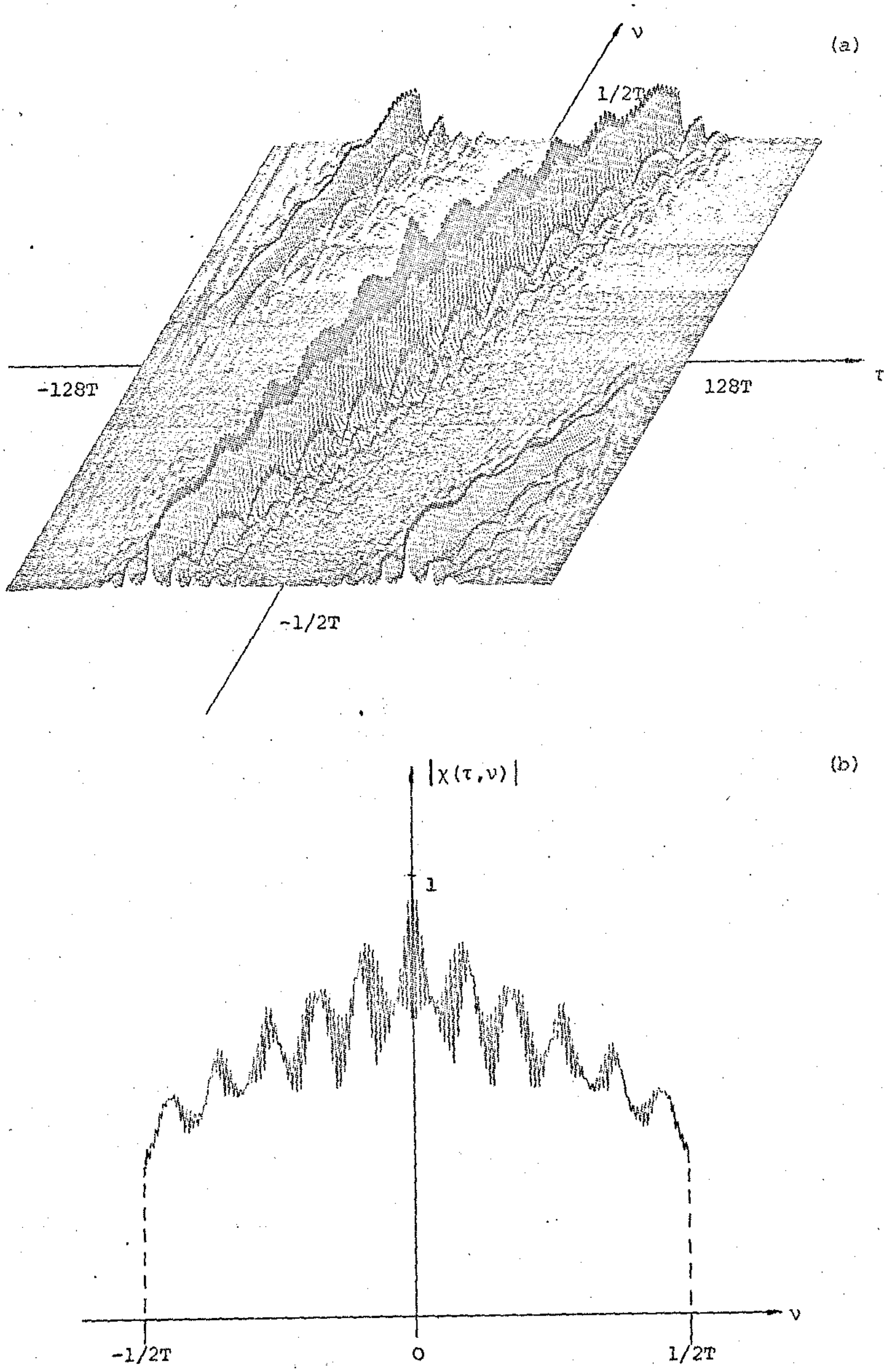


Fig. 9.8 (a) ABF of optimum 121-element uniform complex code when the initial sequence is a Frank code.

(b) Peak response as a function of doppler shift ν.

Thus the optimization procedure usually does not destroy the basic underlying properties of the ABF of the initial waveform.

ABF's of other codes, Fig. 9.9 and Fig. 9.10, show similar residual ridge-like structures which suggest a certain degree of range-doppler coupling. This may or may not lead to resolution problems and will have to be considered for each individual application.

In summary, all of the waveforms discussed in previous chapters are optimum for some particular clutter environment. The different pulse trains yield a wide variety of ABF shapes. The contours may be of the diagonal ridge structure as for linear FM type signals, or may consist of a single strong spike surrounded by a low level pedestal for noise-like waveforms, or various combinations of these basic structures. The waveforms have different tolerances to doppler shifts. This may be exploited for hardware savings if ambiguity in the range-doppler coverage can be tolerated.

Another advantage of discrete coding which has not been mentioned is the flexibility of eliminating the range-doppler ambiguity of QP codes, for example, by simply scrambling the order of the sub-pulses. Moreover, the variations possible with discrete coded pulse trains are virtually unlimited in that the phase, amplitude, frequency and time of transmission of each sub-pulse can be varied. The resulting multi-function capability and adaptability to a particular target environment is clearly one of the most attractive features of discrete coding.

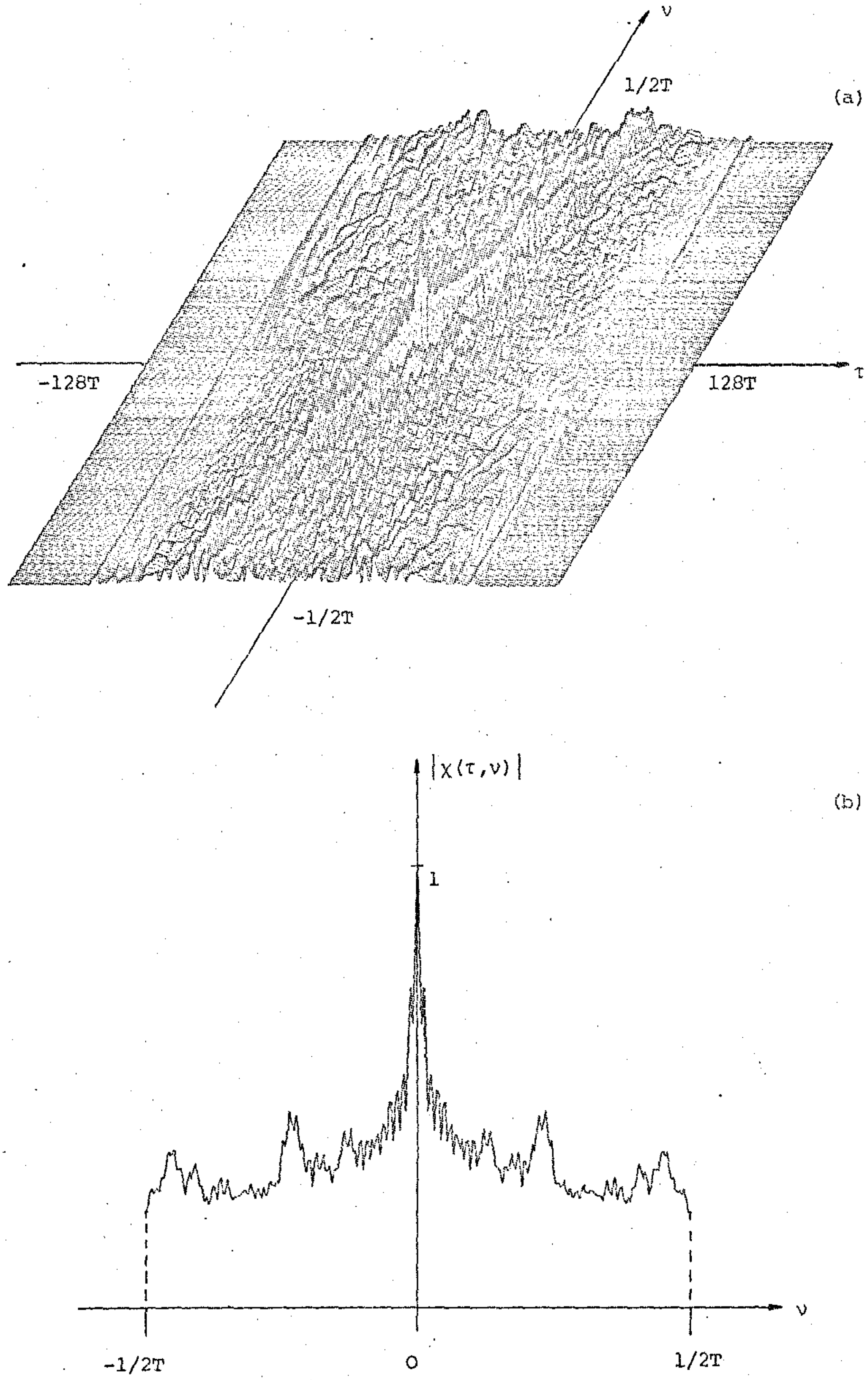


Fig. 9.9 (a) ABF of a 91-element Huffman code derived from a LFM type zero pattern

(b) Peak response as a function of doppler shift ν.

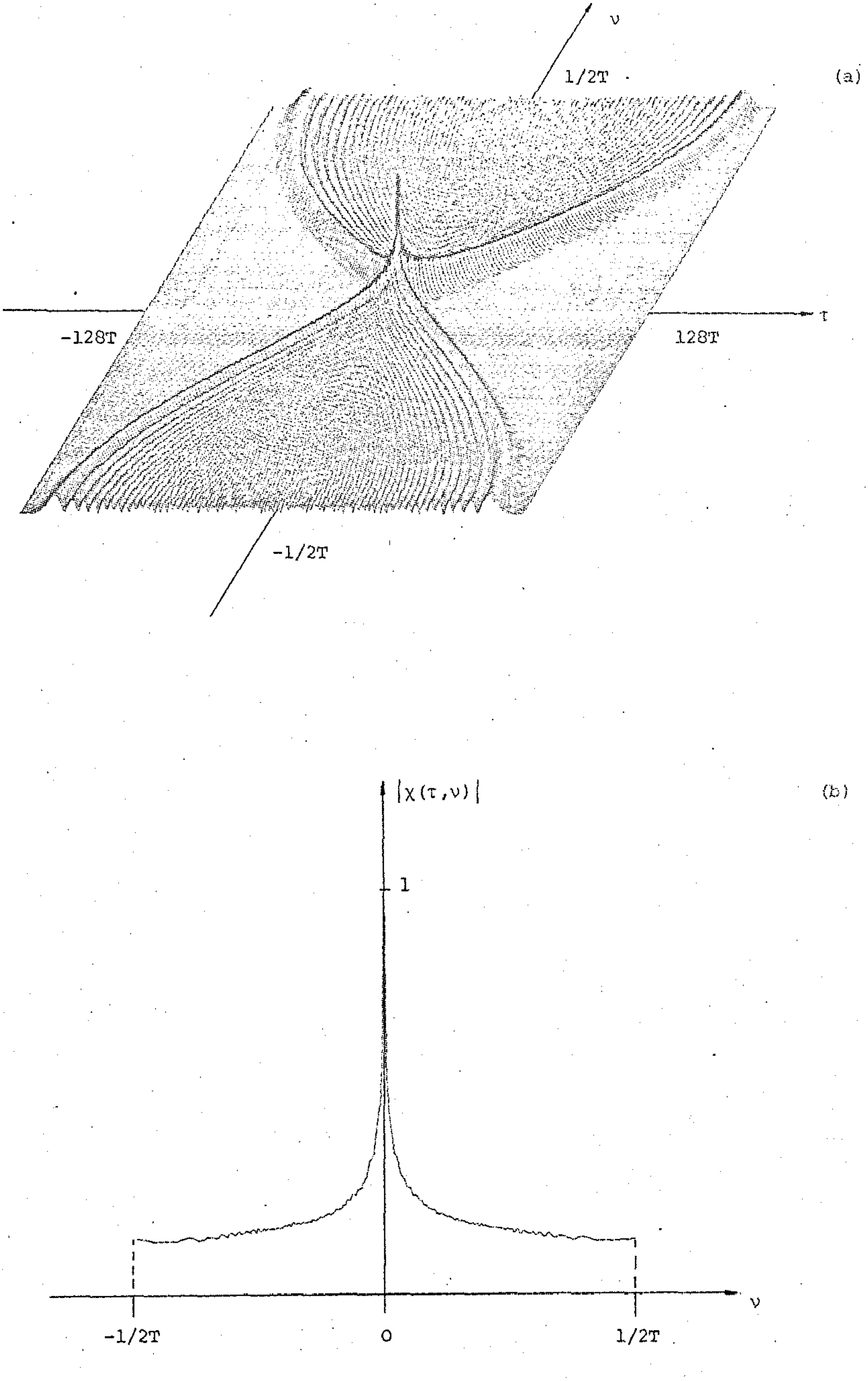


Fig. 9.10 (a) ABF of 121-element Bessel code

(b) Peak response as a function of doppler shift ν.

CHAPTER 10

DISCUSSION AND CONCLUSIONS

This thesis has reported investigations into discrete coding techniques for improving range resolution and clutter performance of radar systems. The waveforms considered in this work, besides representing an interesting mathematical area, are also of practical significance in related fields such as sonar, navigation and communications. The study is focussed on the properties of the coded waveforms as modulating functions of a carrier signal. The design of hardware structures of radar processors has not been considered, since there are generally a number of ways available to implement near optimum receivers for a given waveform. Cost, complexity and reliability are usually the bounds set on processor design rather than physical realizability. These problems would have to be considered for each application individually.

The waveform design approach using discrete coding offers a degree of flexibility and has many advantages in terms of waveform shaping and processor implementation. In general discrete coding of a N-element pulse train provides 3N degrees of coding freedom. The variations possible with such waveforms is virtually unlimited in that the phase, amplitude, frequency and time of transmission of each sub-pulse can be varied. This is clearly in contrast to analogue waveforms which depend on one, or possibly two parameters. The inherent multi-function capability of discrete coded waveforms is compatible with the requirements of modern phased array radars. In addition amplitude and phase modulated pulse trains are particularly well suited to digital implementation.

Throughout this work digital processing has been assumed. The application of digital processing techniques to radar becomes more practical as compactness, cheapness and operational speed of digital microcircuits continue to increase. Although modern optical processing techniques sometimes provide an attractive alternative, the use of a digital method with its inherent flexibility and reliability offers many advantages.

To mention but a few, it simplifies pulse compression and real-time multi-dimensional analysis of input data in range, doppler, bearing, etc. Furthermore, it also offers considerable advantages in post-detection and display processing. In addition the use of digital processors will in many cases reduce future system modifications to easy and inexpensive software changes, rather than requiring costly hardware replacement. However, the complexity and the amount of data to be handled by modern high performance radar processors can easily reach critical limits. Current developments in emitter-coupled logic technology indicate that real-time processing of signals with 40 MHz bandwidth seem attainable.

An attempt has been made in this work to present the results of various design objectives. The assumption of a matched filter receiver, underlying most of the work, is not a serious limitation on the applicability of the results, since in practice very little prior information about the target environment is usually available. Therefore, the investigations were concentrated on the autocorrelation function properties of the various types of pulse trains considered. The principal aim has been to design pulse trains,subject to a fixed amplitude constraint, whose autocorrelation sidelobes are as low as possible. Fixed amplitude constraints arise from practical radar considerations of operating the transmitter at its optimum peak power point. The problem, therefore, was to find particular phase codes that lead to suitable resolution properties. For some applications, especially digital implementation, the design objective may be to approximate the response characteristics of a given analogue waveform. It has been shown in Chapter 4 that virtually all the desired properties of analogue waveforms can be retained if the sampling interval is chosen properly.

The design methods based on numerical optimization developed in this thesis have shown that phase coded pulse trains can improve the range resolution and clutter rejection performance of a radar system. However, the full benefits of these codes can only be obtained with increased stringency of the transmitter and receiver tolerances. The additional expense in system complexity may be justified in situations where the improvements in sidelobe and clutter performance are of significant magnitude as compared to other waveforms which are simpler to generate and process.

Signal design via optimization techniques is evidently no panacea. From the theoretical point of view the most serious objection to optimization procedures is that the results are not unique. Thus for a given design objective the sequences to which the process converges is dependent on the initial choice of the starting sequence. Moreover, the usefulness of such an approach to signal design depends largely on the optimization algorithms available. While there are a number of efficient optimization techniques for problems with small dimensionality ($N < 15$) not many seem to be able to handle non-linear functions with a large number of variables ($N > 50$). In particular, there is almost a complete lack of methods for non-linear integer problems. Nevertheless, with the development of numerical algorithms and improvements in computing facilities, computer-aided design techniques will attract increasing attention in signal synthesis problems for a variety of applications. However, the problem should be formulated and an optimization method selected in such a manner that the results will give information beyond mere numerical values of the best attainable design vector.

Self-clutter interference introduced by inadequacies in the matched filter response impose rather fundamental limitations on weak-target visibility. Moreover, the resolution problems caused by self-clutter and undesired objects are in principle no different in that both impede resolution in the same manner. If processor complexity is not of utmost concern, the use of amplitude and phase modulated pulse trains may proffer an improved performance. Alternatively, self-clutter suppression may be achieved by sidelobe reduction filters. Although mismatching the receiver filter may be a useful means of adopting the waveform to a particular target environment, it should not be used as a primary method to improve resolution. Hence, the proper approach to clutter suppression, except in a few special cases, is via waveform design and matched filtering.

In situations where the relative doppler spread of the targets cannot be neglected, the matched filter response has to be investigated in terms of the ambiguity function. In principle, the methods developed could be applied to the more general problem of designing pulse trains suitable for resolving targets in range and range rate. However, even with modern computers, the search for phase codes having good resolution properties in both range and velocity is so expensive that it would have to be restricted to relatively short sequences ($N \simeq 20$) and small areas of the range-doppler plane. For these reasons waveform synthesis for range and range rate resolution usually consists of a trial and error procedure and a judicious use of available information.

In summary, the signal design problem has in general defied solutions by all means other than exhaustion. In particular, no concise set of necessary and sufficient conditions has been formulated by which signals with specified properties can be synthesized. Signal theory, the basis for many technical advances, is far from being complete and its further development is, therefore, of fundamental importance.

As a final remark, the design methods developed in this thesis are of general interest. With appropriate modifications they can be applied to a variety of signal design and filtering problems. Furthermore, it is hoped that the results of this study may give some insight into the problems encountered in related areas of signal theory.

APPENDIX A

A.1 Principle of Stationary Phase[22]

The principle of stationary phase is a proposition which has found wide application in many branches of electrical engineering and physics. To arrive at the concept of stationary phase consider the expression

$$I = \int_{-\infty}^{\infty} s(t)\ e^{j\phi(t)}\ dt \qquad (A.1)$$

in which s(t) is a slowly varying function of t, while $\exp(j\phi(t))$ goes through a large number of cycles. Under these conditions the value of the integral (A.1) will be small, since the negative and positive portions of successive cycles tend to cancel each other. However, at the stationary value of $\phi(t)$, i.e.

$$d/dt\ \{\phi(\xi)\} = 0 \qquad (A.2)$$

a significant contribution may be made to the value of I in the neighbourhood of the stationary phase points ξ which satisfy condition (A.2). This is illustrated in Fig. A.1.

Near the stationary phase point ξ, the function $\phi(t)$ can be represented by the first few terms of a Taylor series

$$\phi(t) \approx \phi(\xi) + 1/2\ \phi''(\xi)(t-\xi)^2 \qquad (A.3)$$

(The linear term is absent due to condition (A.2)).

Therefore the contribution to the integral (A.1) in the neighbourhood Δ is given by

$$I \approx s(\xi)\ e^{j\phi(\xi)} \int_{\xi-\Delta}^{\xi+\Delta} e^{j\frac{1}{2}\phi''(\xi)(t-\xi)^2}\ dt$$

The term $s(\xi)$ is assumed to be constant for the integration.

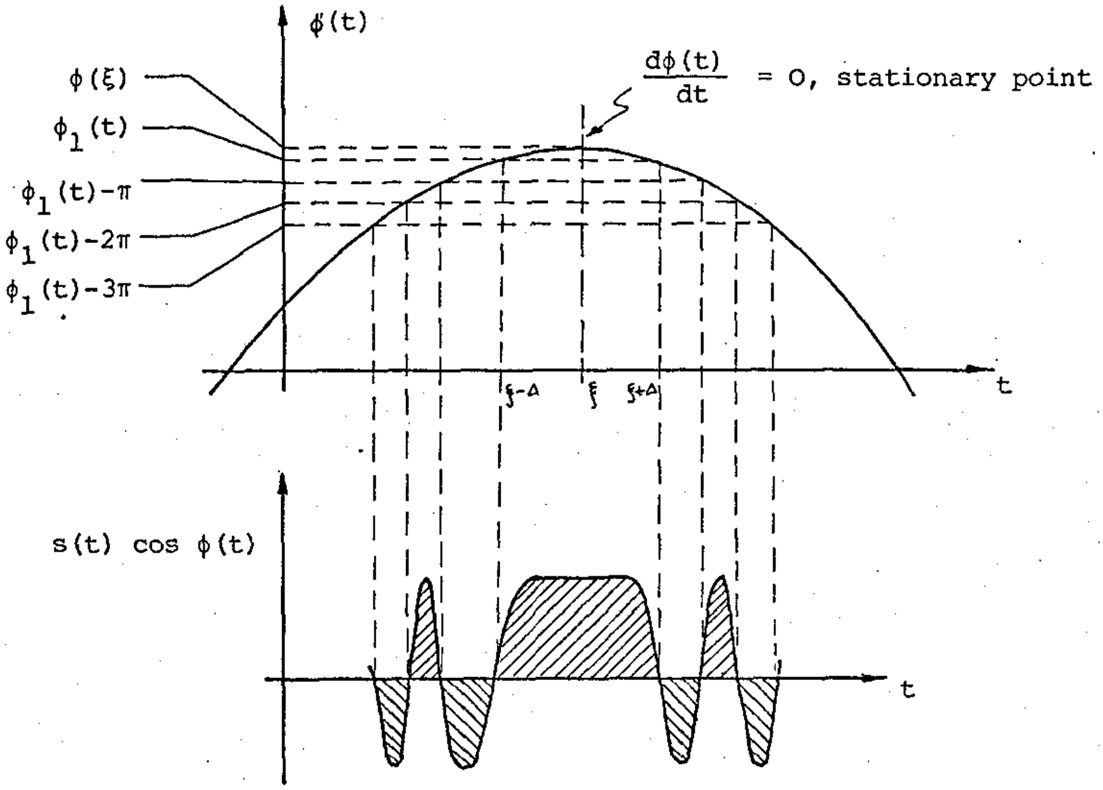


Fig. A.1 Stationary phase criterion.

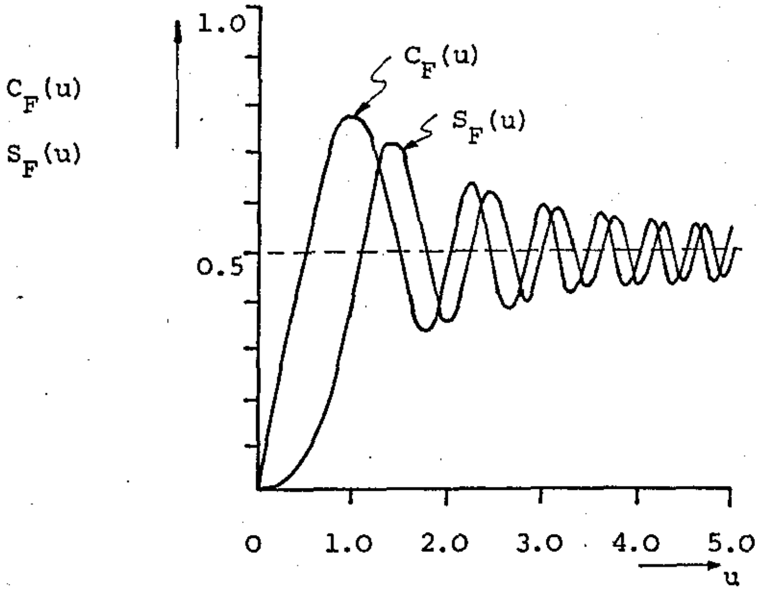


Fig. A.2 Sine and cosine Fresnel integrals.

Making the variable transformations

$$t - \xi = \zeta \quad \text{and} \quad \frac{\phi''(\xi)}{2}\,\zeta^2 = \frac{\pi}{2}\,\eta^2$$

so that

$$d\zeta = \sqrt{\pi}\,\left(|\phi''(\xi)|\right)^{-\frac{1}{2}}\,d\eta$$

leads to

$$I \simeq 2\sqrt{\pi}\,\frac{s(\xi)}{\sqrt{|\phi''(\xi)|}}\,e^{j\phi(\xi)}\int_{o}^{(|\phi''(\xi)|/\pi)^{-\frac{1}{2}}} e^{j\frac{\pi}{2}\eta^2}\,d\eta \qquad \text{(A.4)}$$

The integration limits may be arbitrarily extended without substantially affecting the value of I, since only the stationary points give significant contributions. The above Fresnel type integral (see Section A.2) may thus be approximated by

$$\frac{1}{\sqrt{2}}\,\exp(j \pm \pi/4)$$

Hence

$$I \simeq s(\xi)\,\sqrt{\frac{2\pi}{|\phi''(\xi)|}}\;e^{j(\phi(\xi)\ \pm\ \pi/4)} \qquad \text{(A.5)}$$

The plus sign applies when $\phi''(\xi) > 0$, and the minus sign when $\phi''(\xi) < 0$. If Eq. (A.2) has several points of stationary phase, then it is necessary to take the sum of terms similar to (A.5).

A.2 Fresnel Integrals[94]

The complex Fresnel integral denoted by Z(u) is defined as

$$Z(u) = C_F(u) + j\,S_F(u) = \int_o^u e^{j(\pi/2)x^2}\,dx \qquad \text{(A.6)}$$

where

$$C_F(u) = \int_o^u \cos\left(\frac{\pi}{2}\,x^2\right)\,dx \qquad \text{(A.7)}$$

$$S_F(u) = \int_o^u \sin(\frac{\pi}{2} x^2)\ dx \tag{A.8}$$

These integrals cannot be solved in closed form. The values of $C_F(u)$ and $S_F(u)$ are plotted as a function of u in Fig. A.2. It is noted that for large values of u both functions converge to the value 0.5.

It can be shown that Fresnel integrals are odd functions of the variable u, i.e.

$$Z(u) = -Z(-u)$$

$$C_F(u) = -C_F(-u)$$

and

$$S_F(u) = -S_F(-u)$$

Consider now the chirp signal

$$s(t) = e^{j\mu t^2} \qquad , \quad 0 \leqslant t \leqslant T_s$$

where

$$\mu = \pi W/T_s$$

The spectrum of s(t) is given by the Fourier integral

$$S(f) = \int_o^{T_s} s(t)\ e^{-j2\pi ft}\ dt$$

$$= \int_o^{T_s} e^{j(\mu t^2 - 2\pi ft)}\ dt$$

Completing the square of the exponent yields

$$S(f) = e^{-j(\pi f)^2/\mu} \int_o^{T_s} e^{j(\sqrt{\mu}t - \pi f/\sqrt{\mu})^2}\ dt$$

Changing the variables by defining

$$\frac{\pi}{2} \eta^2 = (\sqrt{\mu}t - \pi f/\sqrt{\mu})^2$$

and

$$dt = \sqrt{\frac{\pi}{2\mu}}\ d\eta = \sqrt{\frac{T_s}{2W}}\ d\eta$$

leads to

$$S(f) = \sqrt{\frac{T_s}{2W}}\; e^{-j(\pi f)^2/\mu} \int_{u_1}^{u_2} e^{j(\pi/2)\,\eta^2}\, d\eta$$

where

$$u_1 = -f\,\sqrt{\frac{2T_s}{W}}$$

$$u_2 = \sqrt{2}\,\{\sqrt{WT}_s - f\sqrt{\frac{T_s}{W}}\}$$

Using Eq. (A.6) the spectrum of the linear FM signal is given by

$$S(f) = \sqrt{\frac{T_s}{2W}}\; e^{-j(\pi f)^2/\mu} \Big[\{C_F(u_2) - C_F(u_1)\} + j\{S_F(u_2) - S_F(u_1)\}\Big] \qquad \text{(A.9)}$$

APPENDIX B

Bessel Functions[94]

Bessel functions are of considerable importance in many branches of science and engineering. For the reader which is unacquainted with them, some of their fundamental properties and relationships are listed below.

The nth order Bessel functions, $J_n(x)$, of the first kind and of argument x are defined as the Fourier coefficients of the function

$$S(\theta) = e^{jx \sin \theta} \tag{B.1}$$

Consequently,

$$J_n(x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j(x\sin\theta + n\theta)} \, d\theta \qquad n = 0, \underline{+}1, \underline{+}2, \ldots.. \tag{B.2}$$

Bessel functions behave very much like damped sine waves. However, while all sin and cos waveforms are similar in shape, each different type of Bessel function has a different shape. Graphs of some Bessel functions of the first kind are shown in Fig. B.1.

There are certain relations between trigonometric functions and Bessel functions that are of interest in modulation theory.

$$\cos(x\sin\theta) = J_o(x) + 2[J_2(x) \cos 2\theta + J_4(x) \cos 4\theta + \ldots..]$$

$$\sin(x\sin\theta) = 2\{J_1(x) \sin\theta + J_3(x) \sin 3\theta + J_5(x) \sin 5\theta + \ldots.\}$$

$$\cos(x\cos\theta) = J_o(x) - 2\{J_2(x) \cos 2\theta - J_4(x) \cos 4\theta$$

$$J_6(x) \cos 6\theta - J_8(x) \cos 8\theta + \ldots..\}$$

$$\sin(x\cos\theta) = 2\{J_1(x) \cos\theta - J_3(x) \cos 3\theta + J_5(x) \cos 5\theta$$

$$- J_7(x) \cos 7\theta + \ldots..\}$$

Fig. B.2 shows Bessel functions of constant argument and variable order n. The above relationships show that the Bessel functions of integral order are particularly important in modulation theory. In addition for integral order n, the following relations hold

$$J_n(-x) = (-1)^n J_n(x)$$

$$J_{-n}(x) = (-1)^n J_n(x)$$

Another relationship which has been found useful in Chapter 4 is

$$e^{j\, x\sin\theta} = \sum_{n=-\infty}^{\infty} J_n(x)\, e^{jn\theta} = J_o(x) + \{J_1(x)\, e^{j\theta} + J_{-1}(x)\, e^{-j\theta}\} + \{J_2(x)\, e^{j2\theta} + J_{-2}(x)\, e^{-j2\theta}\} + \ldots \qquad \text{(B.3)}$$

Like trigonometric functions Bessel functions can be expanded in power series. Thus,

$$J_n(x) = \frac{x^n}{2^n n!}\{1 - \frac{x^2}{2^2(n+1)} + \frac{x^4}{2\cdot 2^4(n+1)(n+2)} + \ldots\ldots\ldots \qquad \text{(B.4)}$$

$$+ \frac{(-1)^p\, x^{2p}}{p!\; 2^{2p}(n+1)(n+2)\;\ldots\ldots(n+p)} + \ldots\ldots\ldots\ldots \}$$

Furthermore, for very large values of the argument x, Eq. (B.4) reduces to

$$J_n(x) = \sqrt{\frac{2}{\pi x}}\; \cos\,(x - n\pi/2 - \pi/4) \qquad \text{(B.5)}$$

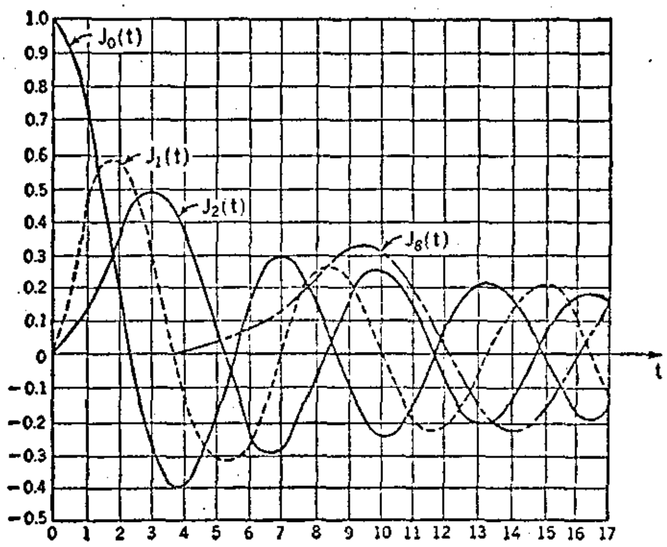


Fig. B.1 Some Bessel functions of the first kind.

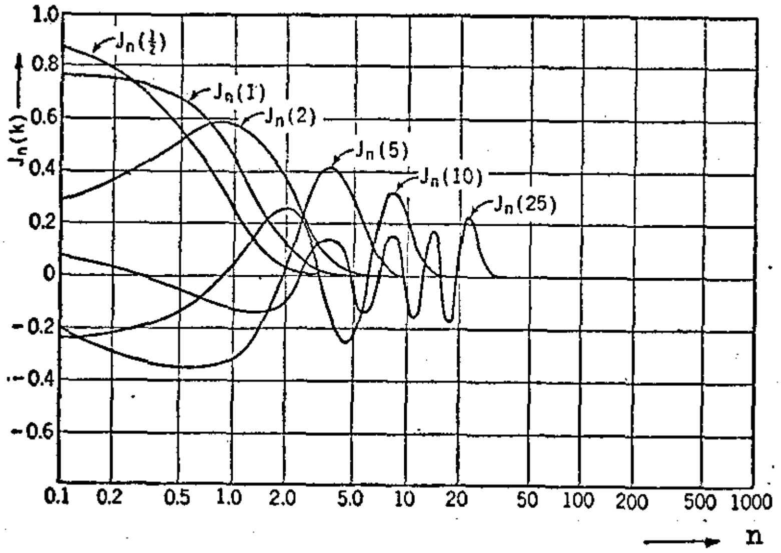


Fig. B.2 Bessel functions of constant argument and variable order n.

APPENDIX C

Group Delay Computation of Discrete Sequences

The group delay $\tau_g(f)$ of an analogue waveform s(t) is defined as

$$\tau_g(f) = \frac{1}{2\pi} \; d\theta(f)/df \tag{C.1}$$

where $\theta(f)$ is the phase of the spectrum of s(t), i.e.

$$\theta(f) = \arg[S(f)]$$

and $$S(f) = FT[s(t)]$$

The spectrum of a discrete waveform s(nT), n = 0,1,2,...,N-1 is according to Eq. (2.14)

$$S(f) = \sum_{n=o}^{N-1} s(n) \; e^{-j2\pi fnT} \tag{C.2}$$

or in ZT

$$S(z) = \sum_{n=o}^{N-1} s(n) \; z^{-n} \tag{C.3}$$

The phase of S(z) is now given by

$$\arg[S(z)] = \text{Im} \; \{\ln[S(z)]\}$$

Hence, the group delay $\tau_g(f)$ is obtained by differentiating the above expression with respect to f

$$\tau_g(f) = \frac{1}{2\pi} \; \text{Im} \; \{ \; d/df \; [\ln S(z)] \}$$

$$= \frac{1}{2\pi} \; \text{Im} \; \{ \; d/dz \; [\ln S(z)] \; dz/df \}$$

$$= - \frac{1}{2\pi} \; \text{Im} \; \{ \frac{S'(z)}{S(z)} \; z(j2\pi T) \}$$

where (') denotes differentiation with respect to z.

From Eq. (C.3)

$$S'(z) = - \sum_{n=o}^{N-1} n \; s(n) \; z^{-(n+1)}$$

Thus

$$\tau_g(f) = T \text{ Real} \left[\frac{\sum_{n=o}^{N-1} n\ s(n)\ z^{-n}}{\sum_{n=o}^{N-1} s(n)\ z^{-n}} \right]$$

For computational work the FFT can be used to evaluate the above expression. Hence

$$\tau_g(n/NT) = T \text{ Real} \left[\frac{\text{FFT} \quad \{n\ s(n)\}}{\text{FFT} \quad \{s(n)\}} \right] \qquad (C.4)$$

which is simply the real part of the DFT of the 1st moment of the sequence s(n), divided by the DFT of s(n). The sampling interval T is usually set equals to unity for convenience.

APPENDIX D

Energy Relationship between the Autocorrelation Function and Crosscorrelation Function of Two Sequences[88].

Consider two arbitrary N-element sequences a(n) and b(n). Their ZT's are given by

$$A(z) = \sum_{n=o}^{N-1} a(n)\ z^{-n} \tag{D.1}$$

$$B(z) = \sum_{n=o}^{N-1} b(n)\ z^{-n} \tag{D.2}$$

The autocorrelation and crosscorrelation functions of these sequences are defined as

$$r_1(k) = \sum_{n=o}^{N-1-|k|} a(n)\ a^*(n+k)$$

$$r_2(k) = \sum_{n=o}^{N-1-|k|} b(n)\ b^*(n+k)$$

and

$$r_{12}(k) = \sum_{n=o}^{N-1-|k|} a(n)\ b^*(n+k)$$

where k ranges from $-(N-1)$ to $(N-1)$.

The ZT's of the above expressions, neglecting the unessential delay factor z^{-N}, are according to Eq. (2.36)

$$R_1(z) = \sum_{k=-(N-1)}^{N-1} r_1(k)\ z^{-k} \qquad = A(z)\ A^*(1/z)$$

$$R_2(z) = \sum_{k=-(N-1)}^{N-1} r_2(k)\ z^{-k} \qquad = B(z)\ B^*(1/z)$$

and

$$R_{12}(z) = \sum_{k=-(N-1)}^{N-1} r_{12}(k)\ z^{-k} \qquad = A(z)\ B^*(1/z)$$

Similarly, the autocorrelation function of the sequence $r_{12}(k)$, denoted by $\psi(m)$ is given by

$$\psi(m) = \sum_{k=-(N-1)}^{N-1-|m|} r_{12}(k)\ r^*_{12}(k+m) \tag{D.3}$$

$$m = 0, \underline{+}1, \underline{+}2, \ldots\ldots, \underline{+}(2N-2)$$

The ZT of the autocorrelation function, $\psi(m)$, is simply

$$\Psi(z) = \sum_{m=-2(N-1)}^{2(N-1)} \psi(m)\ z^{-m} = R_{12}(z)\ R^*_{12}(1/z) \tag{D.4}$$

using the properties (3.5) and (3.6) leads to

$$\Psi(z) = A(z)\ B^*(1/z)\ A^*(1/z)\ B(z)$$

$$\Psi(z) = R_1(z)\ R_2(z) \tag{D.5}$$

Since the autocorrelation is complex conjugate symmetric with respect to zero time shift, i.e.

$$R_2(z) = R^*_2(1/z)$$

With the above expression, Eq. (D.5) can be written in the form

$$\Psi(z) = R_1(z)\ R^*_2(1/z) = R_{12}(z)\ R^*_{12}(1/z) \tag{D.6}$$

Hence, the autocorrelation function of the crosscorrelation of the two sequences, a(n) and b(n), is equal to the crosscorrelation of the autocorrelation functions of the individual sequences. Moreover, two polynomials of the same order are equal if they have identical coefficients, i.e.

$$\psi(m) = \sum_{k=-(N-1)}^{N-1-|m|} r_{12}(k)\ r_{12}^{*}(k+m) = \sum_{k=-(N-1)}^{N-1-|m|} r_{1}(k)\ r_{2}^{*}(k+m)$$

In particular, for zero time shift (m = O)

$$\psi(O) = \sum_{k=-(N-1)}^{N-1} \left|r_{12}(k)\right|^{2} = \sum_{k=-(N-1)}^{N-1} r_{1}(k)\ r_{2}^{*}(k) \qquad (D.7)$$

which is the result used in Chapter 6.

Appendix E

List of Sequences

E.1 Uniform Sequences

Binary Sequences (+ = 1, - = -1)

N	sequence
17	+ + - - + + + + + - + - + - - + -
19	- + - + + - + + - + + + - + + + - - -
21	+ - - - + + + - - + - - + - + - + + + + +
31	+ + + + + + + - - - + + + - - + + - + + - - + - + - + + - + -
41	- - + - + + - + - - + - + + + - - - - + + + - + + + + - - - + - - - + - - - + - -
53	+ - - + - - + - + + + + + + - - + + + + + + + - - + + + - - - - + - + - - + + + - + - + - + - - - + - - +
61	- - - - - + - - - - - + + + + - - - + - + - + - + - - + - + + - + - - + + - - + + + + + - - + + - - + + + + - + + - - - +
71	+ + + + + + + + + + - - - - - + + + + - - - - + + - - + + + + - - + + - - + + - - + + - + + - + + - + - - + - - - - + - + + - + - + - + - + -
91	+ + + + + + + + + + - + - + - + - + + + - + + - - - + - - + - - - + + - - + + + + - - - + + + + - - + - - - - + - - - + + + - - + - - + + + + - - - + - - + - + + - - + + - + + - + -
101	+ + + + + + + + + + + - - - - - - - + + + + + - - - - + + + + - - - + + + - - + + + - - + + - - - + + - + + - - + + - - + - - + - - + - - + - - + - - + - + - - + - + - - + - + - + + - + - + - + - + - +

N	sequence
128	+ + + + + + + + + + + + + - - - - - - +
	+ + + + + - - - - - + + + + - - - + + +
	+ - - - + + + - - + + + - - + + + - - +
	+ - - + + - - + + - - + + - - + - - + +
	- + + - + + - + + - + + - + - + + - + -
	- + - + + - + - + - + - - + - + - + - +
	- + - - - + - +
253	+ + + - + - + - + + + + + + + + + + + +
	- - - - - - - - - - - - + + + + - + - -
	- - - - + + + + - - - - - - - + + + + -
	- - - - + + + - - - - + + + - - - - + +
	+ - - - + + + - - - + + - - + + + + - -
	+ + - - - + + + - - + + - - + + + - - +
	+ - - + + - - + + - - + + - - + + - + +
	- - + + - - + - - + + - + + - + + - - +
	- - + + - + + - + + - + + - + - - + - -
	+ - - + - + + - - - - + - + + - + - + +
	- + - + + - + - + + - + - + - - + - + -
	+ - + + - + - + - + - + - - + - + - + -
	+ - + - + - + - + - + - +
305	+ + + + + - + - + - + + + - - + + - + +
	+ + - - - - + + - - + + + - + + - - + +
	- - - - - - + + + + + - - - - + + - + -
	+ - - - - + + - - - + + - + + - - - - +
	+ - - - - + - - - + + + + + - + + + - +
	- + + - - - + + - - - - + - + + - - + -
	- - + - - - + + + + + + + - + - + - - +
	+ - - - + - + - + - + - - + - - + + + +
	+ - - + + - - + - + + - - + + - + - - +
	+ - + - - - - + + + - - - - - + + + + -
	+ + + - - + - + + + + - + - - + - + - +
	+ - - + - - + - + - + - + + - + - - + -
	- + - - + + + + + + - + + + - + - + + +
	+ + + + - + + - + - + - + + + + - + + +
	+ + - + + - + + - - - - + - - + - - - +
	- + + - +

Uniform complex codes

N			sequence $\phi(n)$		(radians)
13	1.617	-0.433	1.622	-0.376	1.228
	-0.173	-0.817	2.815	1.254	1.162
	2.885	-1.860	-0.768		
15	0.771	0.248	-0.717	-0.378	-0.803
	2.713	0.908	-0.597	1.852	2.091
	-2.980	-0.655	-3.100	-0.500	1.395
17	-0.061	2.653	-1.021	1.484	2.470
	-0.784	-2.522	1.659	-1.177	-1.185
	-0.775	1.259	1.198	1.660	0.029
	-0.346	-0.250			
18	2.082	-2.408	-1.597	0.332	-2.568
	-0.153	-0.198	1.297	0.228	2.301
	-0.405	0.140	-0.942	0.539	-1.593
	-2.741	2.522	0.344		
19	0.000	-2.928	2.134	-0.570	-1.802
	1.709	-2.467	-0.034	-0.855	1.308
	3.028	1.305	2.675	-1.819	-1.452
	-2.688	-2.466	2.558	-3.142	
25	2.545	2.825	1.945	2.625	-1.337
	-0.457	-1.137	-1.175	-2.177	-0.937
	1.105	3.025	-1.337	-1.497	-2.657
	1.505	-0.135	-2.617	-2.897	0.025
	-2.777	0.425	-2.417	1.505	-2.217
31	1.825	-0.055	-1.217	2.065	-1.818
	2.145	1.585	-0.338	1.465	-0.455
	-1.828	1.785	2.265	0.382	-2.698
	1.385	0.745	0.865	-1.055	0.425
	0.145	1.945	4.545	-0.215	2.145
	2.465	2.585	-2.295	-2.868	-1.698
	0.225				

N			sequence $\phi(n)$		(radians)
49	0.040	0.400	0.280	0.240	-0.040
	-0.240	-0.160	0.080	0.698	1.875
	3.133	-2.973	-2.075	-1.178	0.280
	2.075	-2.253	-0.538	1.178	2.253
	-2.275	0.000	2.933	-0.898	1.475
	-2.075	0.698	3.030	-0.120	-2.853
	1.258	-1.635	1.355	-1.138	2.373
	0.160	-1.595	3.053	1.418	-1.098
	-3.093	1.915	0.800	-0.022	-1.075
	-1.693	-3.110	2.515	0.858	
64	0.040	0.480	0.400	0.240	0.360
	0.120	0.200	0.000	0.280	1.065
	2.071	-2.966	-2.542	-2.036	-1.531
	-0.585	0.720	1.971	-2.822	-0.771
	0.920	1.731	-2.862	-1.371	0.240
	2.796	-0.611	1.345	-2.622	-0.145
	1.571	-2.516	0.040	-3.142	0.200
	-3.022	-0.200	2.980	-0.040	2.660
	-0.360	-2.716	1.491	-0.545	-3.062
	0.265	-1.971	1.876	-0.080	-1.811
	-3.062	1.691	0.200	-2.091	2.340
	1.451	0.320	-0.265	-1.011	-1.836
	-2.662	2.956	1.851	0.385	
81	-0.440	0.160	0.360	0.120	0.200
	0.080	0.080	0.080	-0.120	-0.080
	0.538	1.276	2.414	3.112	-2.432
	-2.094	-1.576	-1.018	0.000	1.396
	2.552	-1.974	-0.098	0.898	2.054
	-3.152	-1.636	-0.160	1.654	-1.854
	0.200	2.334	-2.054	-0.080	2.134
	-2.454	-0.080	2.792	-0.298	2.574
	-1.116	1.556	-1.614	0.658	-2.992
	-0.160	-2.672	0.738	-2.054	1.756
	-1.196	2.294	-0.658	2.352	-0.040
	-2.414	1.894	0.080	-1.734	2.454
	-0.280	-2.374	1.814	-0.240	-1.596
	-2.952	2.174	0.818	-0.498	-2.334
	2.352	1.076	0.080	-0.698	-1.116
	-1.894	-2.512	2.912	2.454	1.676
	0.418				

N		sequence $\phi(n)$		(radians)	
100	0.040	0.520	0.600	0.640	0.720
	0.440	0.640	0.320	0.600	0.600
	0.720	0.869	1.297	2.285	-2.891
	-1.680	-2.353	-1.565	-0.777	-0.229
	0.480	1.617	2.793	-2.473	-1.177
	0.680	1.937	2.593	-2.633	-1.497
	-0.040	1.405	-2.793	-0.589	1.137
	2.662	-1.617	0.349	2.073	-2.205
	-0.320	2.153	-1.137	1.857	-2.073
	0.360	2.473	-1.097	1.497	-2.553
	0.040	-2.942	0.440	-2.142	0.840
	-2.782	0.000	3.062	0.240	-2.742
	0.280	-2.433	1.497	-1.057	2.873
	0.680	-2.073	1.817	-0.817	1.793
	-0.560	-2.165	2.033	0.509	-1.057
	-2.982	0.817	-1.269	-2.873	1.485
	-0.440	-1.977	-2.873	2.673	1.537
	0.320	-1.337	-3.033	1.835	1.057
	0.120	-0.549	-0.897	-1.485	-1.993
	-2.702	2.873	2.365	1.457	0.109

E.2 Pairs of Sequences

Binary Sequences (+ = 1, - = -1)

N	sequence
19	+ - - + - + - - - - + - + + + - + + +
19	+ + - - + + + + + + + - + - - + + - +
23	+ + - - - - + - - - - + - + + + - + + - + + -
23	- + + - - + + - + - + - - - - + + + + - - - -
31	+ - + + + + - - - - - + + - - - + - - + - - + - - + - + - + -
31	- + - - - - + + - + - - - + - - - - + + + - + - + + - + + - -
53	- - + - - - - - - + + - - + + + + - - + - - + + - + + - + - + - - + + - + + - - - + - - + - + - + - - - -
53	+ + + + - + - + - - + - + - - + + - + - + - + - + + + - - + + + + - - - - - + - - + - - - - - + + - - - -
61	+ + + + + - + + - - - - + + + - - - + + - + + + - + - + + - - - - + - + + + - - - - + - + - + + - + + + + + + + - + - - -
61	+ - - - - - + + - + - + + + - + - - + + - - + - - + + - - - + - - - - + - - + - - - - - - - + - + - + - + + - + + - + - +

N	sequence
73	+ + - - + + - + - + - - + - - + - + + - + - + - - + + + + + - + + - - + - + + - - - - - + - + + + + - + - - - + + - - - + + + - - + + + - + + - -
73	+ - + - + - - - + - - - + + + + + + + - - + - - + + - - + + + + - - - - - + + - + - - - - + - - - + - - - - - - + - - - - + - + + - - + - + + + -
91	- + - - - + - + - - + - - + - + + - - + + - - + - + + - + - + - - + + - + + + - - + + - + + + - - + + - - - - - - - + - - - + - + + + + + + - + + + - - - + + + - - + + + + - - - - -
91	- - + + - - + - + + + - + + + + - - + - + + + - + + + + - + - + + + + - + - - + + + - + - - + - + + + - - - + - + - + - + - - + - + - - - - - + - + + - + - - - - - + + - + + + - - -
101	+ - - + + - - + + + + + - + + + - + + - + + - + + - + + - + - - - - + - - + - + - + + + - - + - + - + - + - + - - - + + + + + + + - + - - - - - - + + + + - - - - + + + + - - - + + + + - - + + + - - + +
101	- + + - + + + - - - + + + - + - + + - + - - - - + + - + - + + - - + - + - + - - - - + - + + + + + - - + - - - + + - - + - - + + + - + + + - + + - + - + - - - - - - - - - - - - + + - - + - + + - - + + -

Uniform complex codes

N		sequence $\phi(n)$		(radians)	
9	1.507	-1.701	-0.450	-2.704	-0.185
	-1.773	2.409	-2.470	-2.048	
9	1.328	0.879	-2.107	-1.750	2.712
	2.760	-1.388	-2.120	-0.456	
11	2.809	-1.025	-3.087	2.294	0.400
	-2.003	-2.912	-2.229	1.699	2.801
	-2.214				
11	-0.221	0.323	-0.661	1.142	-2.771
	2.882	-1.990	-2.533	1.009	2.827
	-0.705				
21	0.499	-1.250	-0.402	-2.937	-2.090
	2.940	-0.998	-1.053	2.591	-1.049
	-2.389	2.335	2.993	0.050	-1.495
	1.444	1.298	-1.948	-1.495	-1.343
	2.691				
21	3.100	1.607	1.157	2.092	1.556
	2.891	-3.044	-1.740	-2.035	-0.590
	1.741	-1.585	0.543	-1.399	2.498
	0.211	-0.037	-1.800	-1.337	-0.003
	2.742				
31	-1.094	3.133	-2.758	1.000	-2.146
	0.763	-2.575	-1.136	-0.170	-1.192
	-1.873	2.548	-2.811	2.499	-0.102
	0.249	-2.098	0.996	0.307	1.198
	-0.350	2.579	0.315	-1.074	2.409
	-2.070	-2.348	-0.871	-1.622	0.777
	2.034				
31	1.313	2.661	-0.488	-1.620	-1.655
	1.205	0.827	-2.329	2.151	-2.082
	-1.447	1.908	2.714	-1.259	-0.354
	1.381	-1.828	-2.673	1.847	1.622
	2.522	0.812	1.596	1.931	2.679
	-3.004	2.360	-0.838	2.234	-2.562
	1.262				

N		sequence $\phi(n)$		(radians)	
41	2.256	2.433	-0.559	0.650	1.338
	1.363	-1.775	-0.836	0.380	-1.041
	-1.824	2.249	1.172	2.399	0.748
	1.549	-2.547	-0.255	1.657	-1.001
	0.300	-2.704	1.115	-0.423	1.659
	-0.970	-0.048	0.579	-1.321	2.026
	3.033	1.263	-0.939	-1.938	2.064
	-0.905	2.438	1.227	-3.029	0.351
	1.651				
41	-0.697	1.908	2.714	-1.059	-0.904
	-0.069	-2.127	2.911	0.647	1.771
	2.021	0.713	0.595	-2.652	2.178
	3.079	0.661	-1.138	1.585	2.521
	2.762	-1.560	-2.512	-1.317	0.297
	-1.259	-0.072	-0.726	-1.085	1.999
	-0.235	-1.374	0.025	-2.187	-2.139
	3.000	-1.153	-2.211	-0.778	1.450
	1.251				
47	-2.094	1.682	-2.059	0.650	2.237
	0.763	-1.325	-1.737	0.231	-2.841
	-2.724	-0.735	2.973	1.650	2.698
	-0.101	-1.948	2.596	2.406	-2.052
	0.150	1.779	-1.885	-2.823	-0.141
	-2.170	-1.097	1.179	0.028	-2.907
	-2.949	2.012	-2.273	-1.338	-2.719
	-2.705	0.504	-2.073	1.305	0.801
	-3.133	-2.797	1.158	-2.220	-1.209
	-1.804	-2.619			
47	-1.977	-2.472	2.297	-2.862	2.322
	-3.020	2.395	-1.753	-2.304	2.029
	2.910	-1.287	1.734	-2.413	-1.271
	0.841	0.938	0.183	0.448	0.241
	-0.521	0.473	0.415	2.749	0.665
	-2.124	1.076	-2.937	3.094	-1.484
	0.347	-0.111	1.022	3.099	2.600
	-1.965	0.679	0.773	0.089	2.691
	-1.221	2.982	-2.440	-2.033	2.708
	1.627	-0.105			

Six-phase codes

N		sequence $\phi(n)$		(radians)	
17	0.000	1.047	1.047	2.054	2.054
	-2.054	1.047	2.054	-1.047	2.054
	0.000	3.142	1.047	0.000	-1.047
	-2.054	3.142			
18	2.054	3.142	-2.054	-2.054	-2.054
	0.000	3.142	-2.054	1.047	-2.054
	2.054	-1.047	2.054	1.047	-1.047
	-1.047	3.142	2.054		
19	1.047	2.054	3.142	-2.054	-2.054
	1.047	-2.054	-2.054	1.047	-2.054
	0.000	-2.054	1.047	-1.047	-1.047
	-2.054	-2.054	3.142	2.054	
20	2.054	2.054	1.047	0.000	1.047
	1.047	1.047	3.142	3.142	-2.054
	0.000	3.142	-2.054	1.047	1.047
	-1.047	2.054	-1.047	3.142	1.047
21	0.000	1.047	1.047	1.047	2.054
	3.142	-1.047	1.047	3.142	1.047
	3.142	0.000	3.142	1.047	-1.047
	3.142	1.047	0.000	-1.047	-1.047
	-2.054				
22	-2.054	-1.047	-1.047	-1.047	0.000
	1.047	3.142	-1.047	0.000	3.142
	0.000	3.142	-1.047	2.054	0.000
	-2.054	2.054	0.000	-1.047	-2.054
	-2.054	3.142			

N		sequence $\phi(n)$		(radians)	
23	3.142	3.142	2.054	1.047	0.000
	0.000	1.047	2.054	1.047	2.054
	3.142	-2.054	0.000	2.054	-2.054
	2.054	-2.054	0.000	3.142	1.047
	-1.047	3.142	0.000		
24	2.054	2.054	2.054	0.000	0.000
	0.000	1.047	1.047	1.047	2.054
	3.142	-2.054	1.047	1.047	-2.054
	-1.047	2.054	-1.047	1.047	-2.054
	2.054	0.000	-2.054	2.054	
25	3.142	3.142	2.054	1.047	1.047
	1.047	1.047	2.054	3.142	-2.054
	-2.054	1.047	1.047	-2.054	-2.054
	1.047	2.054	-1.047	1.047	-2.054
	1.047	-2.054	2.054	0.000	-2.054
26	-2.054	-1.047	-1.047	0.000	0.000
	1.047	3.142	-2.054	0.000	2.054
	-2.054	1.047	-2.054	1.047	3.142
	1.047	-1.047	3.142	1.047	-1.047
	3.142	2.054	1.047	0.000	0.000
	-1.047				
28	0.000	0.000	1.047	1.047	2.054
	3.142	-1.047	0.000	2.054	3.142
	0.000	2.054	-2.054	1.047	-2.054
	1.047	-2.054	2.054	-1.047	-2.054
	2.054	0.000	-2.054	-2.054	3.142
	2.054	2.054	1.047		
30	0.000	1.047	1.047	1.047	2.054
	2.054	-2.054	-1.047	0.000	2.054
	-2.054	0.000	3.142	-1.047	2.054
	-1.047	3.142	0.000	3.142	1.047
	-1.047	3.142	1.047	0.000	-2.054
	3.142	3.142	2.054	2.054	1.047

E.3 Non-uniform Sequences

Huffman codes

N		sequence a(n)				
5	Re [a(n)] :	1.000	0.716	0.000	0.716	-1.000
	Im [a(n)] :	0.000	-0.716	-0.512	0.716	0.000
7	Re a(n)] :	1.000	-0.273	0.775	0.628	0.925
		-0.273	-1.000			
	Im [a(n)] :	0.000	0.473	-0.620	0.537	0.361
		-0.473	0.000			
9	Re [a(n)] :	1.000	0.380	0.208	-0.866	0.885
		-0.268	0.904	0.380	-1.000	
	Im [a(n)] :	0.000	0.917	0.904	0.486	-0.178
		0.956	-0.208	-0.917	0.000	
15	Re [a(n)] :	1.000	0.505	-0.234	-0.251	-0.939
		0.687	0.174	-0.782	-0.174	0.687
		0.939	-0.251	0.234	0.505	-1.000
	Im [a(n)] :	0.000	-0.850	-0,430	-1.032	-0.408
		0.437	-0.603	0.000	-0.603	-0.437
		-0.408	1.032	-0.430	0.850	0.000
17	Re [a(n)] :	1.000	-0.811	0.076	0.716	-0.586
		0.132	-1.022	0.639	-0.307	-0.701
		-0.692	0.824	0.497	0.429	-0.429
		-0.811	-1.000			
	Im [a(n)] :	0.000	0.389	-0.316	0.215	0.757
		-0.522	0.164	0.383	-0.259	-0.573
		-0.129	-0.599	-0.613	-0.353	-0.316
		-0.389	0.000			
20	Re [a(n)] :	1.000	-0.104	-0.154	0.417	0.697
		-0.668	0.616	-0.168	0.529	-0.310
		-0.576	-0.508	0.014	0.774	0.879
		0.595	-0.333	0.571	-0.119	-1.140
	Im [a(n)] :	0.000	0.816	-0.542	-0.097	0.754
		0.963	0.037	0.687	-0.368	0.792
		0.326	0.578	0.391	-0.596	-0.118
		-0.919	0.541	0.424	-0.930	0.000

N				sequence a(n)		
25	Re[a(n)]	:	1.000	0.200	-0.845	-1.510	-1.699
			-1.817	0.002	1.013	-1.691	-1.726
			1.855	-1.765	-0.087	-1.346	0.963
			-2.022	-0.238	0.304	-1.405	-0.674
			0.810	1.592	1.413	0.200	-1.000
	Im[a(n)]	:	0.000	1.516	1.363	0.741	-0.045
			-1.184	-1.405	0.043	0.702	-1.133
			-1.287	1.384	-1.712	1.794	-2.043
			0.419	1.815	-0.968	-0.002	2.061
			1.494	0.544	-0.758	-1.516	0.000
45	Re[a(n)]	:	1.000	0.108	-1.012	-1.201	-1.269
			-1.283	-0.883	-0.723	0.396	0.921
			0.347	-0.797	-1.735	1.141	1.756
			-1.646	0.255	0.538	-1.712	1.483
			-0.218	0.439	-1.426	0.199	0.020
			0.776	-1.512	0.490	-0.996	-0.393
			1.260	-1.432	-1.550	0.182	0.409
			-0.573	-1.420	-0.827	-0.144	0.599
			1.062	1.261	1.249	0.108	-1.000
	Im[a(n)]	:	0.000	1.507	0.988	0.407	-0.009
			-0.566	-0.955	-1.447	-1.380	-0.327
			0.463	0.182	-1.317	-1.827	0.584
			0.591	-1.539	0.128	-0.133	-1.378
			0.812	-1.650	1.387	-1.695	0.840
			-1.870	0.814	-0.256	-1.200	1.705
			-1.355	-1.609	1.530	0.797	-0.409
			-0.792	0.213	1.390	1.293	1.268
			0.694	0.129	-0.663	-1.507	0.000
55	Re[a(n)]	:	1.000	0.110	-2.698	-3.414	-1.666
			-1.138	-0.671	0.454	1.688	4.054
			6.907	7.635	2.392	-5.477	-2.246
			7.530	3.934	-2.321	3.691	3.062
			-3.288	1.931	2.307	-0.514	5.750
			-2.953	2.227	1.874	1.670	-1.078
			4.089	-1.131	4.748	0.782	-0.626
			4.647	-1.947	0.084	8.976	0.949
			-6.232	0.237	4.698	2.437	-1.393
			-4.271	-5.537	-5.173	-2.455	0.062
			1.127	3.059	2.906	0.110	-1.000

N		sequence a(n)				
55	Im[a(n)] :	0.000	2.370	1.632	-1.058	-2.415
		-2.454	-3.174	-3.851	-4.177	-4.279
		-1.992	3.175	7.252	3.362	-5.406
		-3.046	5.632	1.300	-2.237	5.383
		2.093	-3.125	3.523	-1.025	-1.826
		9.016	-6.935	9.526	-8.836	7.807
		-3.473	1.042	3.295	-3.970	1.746
		-0.823	-3.652	5.318	-0.270	-9.593
		-1.559	4.356	-0.896	-5.207	-5.419
		-3.770	-1.070	2.114	4.109	3.711
		3.003	1.852	-1.109	-2.370	0.000
65	Re[a(n)] :	1.000	0.102	-2.962	-3.343	-2.417
		-0.578	1.236	0.812	1.410	3.711
		7.362	8.345	3.838	-1.967	-4.508
		0.016	4.916	4.690	-0.076	-3.279
		5.487	1.246	-9.207	5.846	4.617
		-4.121	3.731	-2.440	1.672	0.201
		-0.327	2.656	-1.406	2.924	-1.400
		2.548	0.932	0.025	2.416	-3.599
		8.642	-0.271	-3.169	8.054	-1.338
		-2.529	5.082	6.569	1.670	-3.940
		-1.448	3.529	5.490	1.595	-3.542
		-5.505	-5.380	-3.868	-3.133	-0.780
		1.673	3.026	3.129	0.102	-1.000
	Im[a(n)] :	0.000	2.470	1.544	-1.043	-2.890
		-3.872	-3.070	-2.488	-3.773	-4.317
		-2.422	3.068	7.300	6.014	0.670
		-3.863	-1.037	3.613	5.240	0.445
		-1.786	8.812	0.882	-7.226	7.152
		0.921	0.574	4.895	-3.160	6.587
		-4.895	6.692	-4.906	7.591	-3.293
		6.664	-4.666	3.711	-2.298	4.516
		3.909	-8.335	6.834	0.976	-6.924
		4.364	3.921	-2.587	-6.392	-2.977
		2.629	1.926	-3.010	-7.248	-6.169
		-2.718	0.277	1.868	3.066	4.391
		3.573	1.717	-1.042	-2.470	0.000

N		sequence a(n)				
81	Re[a(n)] :	1.000	0.171	-4.026	-4.971	-1.900
		1.669	3.191	4.022	3.868	3.337
		7.146	10.452	9.835	3.416	-5.980
		-10.636	-8.321	2.933	8.192	-0.688
		-0.699	-0.882	7.124	-9.316	-11.854
		4.295	1.791	-4.405	-0.837	1.227
		-8.791	-0.148	1.210	-10.407	8.241
		-7.533	-2.319	5.531	-11.284	4.444
		-7.000	2.999	-2.736	-0.875	-3.465
		-2.227	3.440	-5.883	4.307	-4.372
		-1.453	1.789	-6.821	10.311	-1.287
		-9.603	13.034	1.897	-12.104	2.231
		12.599	8.764	0.343	-4.523	-2.404
		6.761	11.230	7.435	4.341	-0.032
		-4.984	-6.582	-6.376	-6.010	-4.732
		-1.453	0.848	3.411	4.114	0.171
		-1.000				
	Im[a(n)] :	0.000	2.858	1.957	-2.422	-5.711
		-5.227	-3.644	-2.443	-0.969	-2.099
		-2.560	1.766	8.616	14.297	12.468
		4.863	-4.582	-6.831	3.826	10.898
		3.087	-5.364	7.120	11.543	-6.386
		-5.043	5.074	3.106	-2.050	5.293
		3.814	-7.488	10.163	-2.671	-4.186
		10.781	-10.869	13.373	-6.528	7.876
		-6.476	9.717	-6.948	9.225	-8.461
		12.629	-1.948	1.223	4.723	-4.425
		7.851	-2.065	7.218	6.553	-10.067
		10.781	7.138	-11.017	3.558	14.121
		5.168	-5.236	-6.746	-1.600	6.377
		7.431	-0.944	-6.200	-8.419	-9.752
		-7.743	-3.896	-0.880	0.967	4.028
		5.114	4.400	3.176	-0.981	-2.858
		0.000				

N		sequence a(n)				
91	Re[a(n)] :	1,000	0.052	-0.619	-0.585	-0.420
		-0.516	-0.526	-0.661	-0.796	-0.743
		-0.139	0.154	0.250	0.562	0.081
		-0.300	-0.326	0.011	0.467	-0.187
		-0.276	-0.289	-0.364	1.103	0.778
		-0.810	-0.232	0.783	-0.074	-1.205
		0.143	0.497	-0.197	0.888	-0.047
		-0.174	0.272	-1.250	0.641	0.434
		-0.942	0.885	-0.361	0.811	-1.013
		0.204	-0.817	0.900	-0.087	0.762
		-0.955	0.314	-0.024	-1.025	0.151
		-0.246	0.431	0.343	-0.284	0.349
		-0.601	-0.599	0.659	0.013	-1.149
		-0.443	0.379	-0.425	-0.358	0.456
		0.547	0.364	-0.516	-1.210	-0.987
		-0.320	0.113	-0.062	-0.019	-0.002
		-0.064	0.236	0.366	0.266	0.205
		0.350	0.439	0.686	0.767	0.052
		-1.000				
	Im[a(n)] :	0.000	1.178	0.604	0.158	0.129
		0.064	0.066	0.029	-0.231	-0.775
		-0.863	-0.654	-0.529	-0.036	0.253
		-0.055	-0.523	-0.598	-0.011	0.216
		-0.037	-0.278	-0.896	-0.582	0.587
		0.208	-0.635	-0.005	0.624	-0.427
		-1.122	-0.079	-0.146	-0.143	0.840
		-0.126	-0.177	0.010	-1.202	0.047
		-0.034	-0.363	0.713	-0.599	0.800
		-0.944	0.643	-1.073	0.412	-0.803
		0.304	-0.013	-1.062	0.914	-0.221
		-0.049	0.518	-0.809	0.128	-0.345
		-0.654	0.728	-0.048	-1.096	-0.068
		1.018	-0.219	-0.492	0.861	0.363
		-0.139	-0.565	-1.099	-0.143	0.579
		0.708	0.411	0.175	0.533	0.384
		0.455	0.599	0.183	0.040	0.140
		0.256	0.160	0.073	-0.481	-1.178
		0.000				

Bessel codes

N	sequence				
11	-0.043	0.132	-0.309	0.486	-0.339
	-0.260	0.339	0.486	0.309	0.132
	0.043				
13	0.049	-0.132	0.281	-0.430	0.364
	0.066	-0.397	-0.066	0.364	0.430
	0.281	0.132	0.049		
15	-0.053	0.131	-0.261	0.391	-0.365
	0.047	0.328	-0.178	-0.328	0.047
	0.365	0.391	0.261	0.131	0.053
19	-0.021	0.057	-0.130	0.246	-0.362
	0.358	-0.115	-0.243	0.277	0.151
	-0.277	-0.243	0.115	0.358	0.362
	0.246	0.130	0.057	0.021	
21	0.024	-0.059	0.128	-0.234	0.339
	-0.348	0.158	0.168	-0.301	0.005
	0.300	-0.005	-0.301	-0.168	0.158
	0.348	0.339	0.234	0.128	0.059
	0.024				
25	0.027	-0.062	0.125	-0.215	0.305
	-0.327	0.204	0.055	-0.265	0.181
	0.145	-0.245	-0.090	0.245	0.145
	-0.181	-0.265	-0.055	0.204	0.327
	0.305	0.215	0.125	0.062	0.027
31	-0.032	0.065	-0.120	0.195	-0.270
	-0.300	-0.230	0.045	0.170	-0.244
	0.073	0.182	-0.195	-0.085	0.223
	0.048	-0.223	-0.085	0.195	0.182
	-0.073	-0.244	-0.170	0.045	0.230
	0.300	0.270	0.195	0.120	0.065
	0.032				

N	sequence				
41	0.036	-0.067	0.114	-0.174	0.234
	-0.267	0.236	-0.123	-0.049	0.191
	-0.199	0.043	0.154	-0.188	0.001
	0.187	-0.111	-0.135	0.158	0.098
	-0.170	-0.098	0.158	0.135	-0.111
	-0.187	0.001	0.188	0.154	-0.043
	-0.199	-0.191	-0.049	0.123	0.236
	0.267	0.234	0.174	0.114	0.067
	0.036				
61	0.023	-0.041	0.068	-0.105	0.151
	-0.197	0.227	-0.223	0.167	-0.060
	-0.070	0.168	-0.175	0.075	0.078
	-0.170	0.119	0.042	-0.161	0.106
	0.071	-0.161	0.040	0.136	-0.114
	-0.084	0.146	0.039	-0.155	-0.015
	0.156	0.015	-0.155	-0.039	0.146
	0.084	-0.114	-0.136	0.040	0.161
	0.071	-0.106	-0.161	-0.042	0.119
	0.170	0.078	-0.075	-0.175	-0.168
	-0.070	0.060	0.167	0.223	0.227
	0.197	0.151	0.105	0.068	0.041
	0.023				
71	-0.025	0.042	-0.068	0.102	-0.142
	0.183	-0.212	0.214	-0.174	0.084
	0.024	-0.128	0.174	-0.130	0.011
	0.116	-0.160	0.081	0.067	-0.154
	0.092	0.065	-0.150	0.061	0.103
	-0.134	-0.016	0.144	-0.058	-0.117
	0.104	0.084	-0.125	-0.060	0.133
	0.051	-0.133	-0.060	0.125	0.084
	-0.104	-0.117	0.058	0.144	0.016
	-0.134	-0.103	0.061	0.150	0.065
	-0.092	-0.154	-0.067	0.081	0.160
	0.116	-0.011	-0.130	-0.174	-0.128
	-0.024	0.089	0.174	0.214	0.212
	0.183	0.142	0.102	0.068	0.042
	0.025				

N	sequence				
91	-0.027	0.044	-0.067	0.096	-0.130
	0.164	-0.190	0.197	-0.176	0.120
	-0.035	-0.060	0.135	-0.157	0.110
	-0.010	-0.096	0.145	-0.103	-0.010
	0.116	-0.131	0.037	0.089	-0.133
	0.047	0.087	-0.127	0.025	0.107
	-0.108	-0.027	0.127	-0.053	-0.096
	0.105	0.045	-0.124	0.004	0.123
	-0.040	-0.113	0.062	0.104	-0.072
	-0.101	0.072	0.104	-0.062	-0.113
	0.040	0.123	-0.004	-0.124	-0.045
	0.105	0.096	-0.053	-0.127	-0.027
	0.108	0.107	-0.025	-0.127	-0.087
	0.047	0.133	0.089	-0.037	-0.131
	-0.116	-0.010	0.103	0.145	0.096
	-0.010	-0.110	-0.157	-0.135	-0.060
	0.035	0.120	0.176	0.197	0.190
	0.164	0.130	0.096	0.067	0.044
	0.027				
121	0.030	-0.045	0.065	-0.089	0.117
	-0.145	0.167	-0.178	0.169	-0.136
	0.079	-0.005	-0.070	0.125	-0.140
	0.105	-0.029	-0.060	0.121	-0.122
	0.057	0.040	-0.113	0.113	-0.037
	-0.066	0.119	-0.079	-0.027	0.109
	-0.094	-0.008	0.102	-0.094	-0.012
	0.105	-0.082	-0.035	0.111	-0.052
	-0.071	0.103	0.001	-0.104	0.062
	0.069	-0.099	-0.019	0.108	-0.027
	-0.097	0.062	0.077	-0.084	-0.056
	0.096	0.039	-0.101	-0.028	0.103
	0.025	-0.103	-0.028	0.101	0.039
	-0.096	-0.056	0.084	0.077	-0.062
	-0.097	0.027	0.108	0.019	-0.099
	-0.069	0.062	0.104	0.001	-0.103
	-0.071	0.053	0.111	0.035	-0.082
	-0.105	-0.012	0.094	0.102	0.008
	-0.094	-0.109	-0.027	0.079	0.119
	0.066	-0.037	-0.113	-0.113	-0.040
	0.057	0.122	0.121	0.060	-0.029
	-0.105	-0.140	-0.125	-0.070	0.005
	0.079	0.136	0.169	0.178	0.167
	0.145	0.117	0.089	0.065	0.045
	0.030				

REFERENCES

1. A.W. RIHACZEK, 'Radar signal design for target resolution', Proc. IEEE, Vol. 53, pp 116-128, February 1965.

2. A.W. RIHACZEK, 'Radar resolution properties of pulse trains', Proc. IEEE, Vol. 52, pp 153-164, February 1964.

3. D.E. HUNT, 'Improved multifunction array radar performance with multifrequency operations (with technology prediction for 1978)', EASCON Conv. Record, pp 52-59, September 1968.

4. W.M. SIEBERT, 'A radar detection philosophy', IRE Trans. Inform. Theory, Vol. IT-2, pp 204-221, September 1956.

5. N. ZIERLER, 'Linear recurring sequences', J. SIAM Appl.Math., Vol. 7, pp 31-48, March 1959.

6. R.M. LERNER, 'Signals with uniform ambiguity functions', IRE Trans. Nat. Conv. Record, pt. 4, pp 27-36, March 1958.

7. J.E. STORER and R. TURYN, 'Optimum finite code groups', Proc. IRE (Corr.), Vol. 46, p 1649, September 1958.

8. R.H. BARKER, 'Group synchronizing of binary digital systems', in 'Communication theory', W. Jackson (Ed.) pp 273-287, Butterworths, 1953.

9. R.J. TURYN, 'On Barker codes of even length', Proc. IEEE (Corr.), Vol. 51, p 1256, September 1963.

10. A.M. BOEHMER, 'Binary pulse compression codes', IEEE Trans. Inform. Theory, Vol. IT-13, pp 156-167, April 1967.

11. D.E. VAKMAN, P.N. SEDLETSKIY and I.Z. SHAPIRO, 'Synthesis of quantized phase-manipulated signals with good autocorrelation properties', Radio Eng. Electron. Phys. (USSR), Vol. 15, pp 606-614, April 1970.

12. L. YE. VARAKIN, 'Synthesis of phase-manipulated signals', Radio Eng. Electron. Phys. (USSR), Vol. 14, pp 690-698, May 1969.

13. R. TURYN and J. STORER, 'On binary sequences', Proc. Am. Math. Soc., Vol. 12, pp 394-399, June 1961.

14. R. TURYN, 'Sequences with small correlation', in 'Error correcting codes', H.B. Mann (Ed.), pp 195-228, John Wiley 1968.

15. E. HOLLIS, 'Comparison of combined Barker codes for coded radar use', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-3, pp 141-143, January 1967.

16. R.C. HIEMILLER, 'Phase shift codes with good periodic correlation properties', IRE Trans. Inform. Theory, Vol. IT-7, pp 254-257, October 1961.

17. R.L. FRANK, 'Polyphase codes with good nonperiodic correlation', IRE Trans. Inform. Theory, Vol. IT-9, pp 43-45, January 1963.

18. R.L. FRANK and S.A. ZADOFF, 'Phase shift pulse codes with good periodic correlation properties', IRE Trans. Inform. Theory, Vol. IT-8, pp 381-382, October 1962.

19. S.W. GOLOMB and R.A. SCHOLTZ, 'Generalized Barker sequences', IRE Trans. Inform. Theory, Vol. IT-11, pp 533-537, October 1965.

20. D.A. HUFFMAN, 'The generation of impulse equivalent pulse trains', IRE Trans. Inform. Theory, Vol. IT-8, pp S10-S16, September 1962.

21. P.M. WOODWARD, 'Probability and information theory with applications to radar', Pergamon Press 1955.

22. C.E. Cook and M. BERNFELD, 'Radar Signals', Academic Press 1967.

23. A.W. RIHACZEK, 'Principles of high-resolution radar', McGraw-Hill 1969.

24. R. PRICE and E.M. HOFSTETTER, 'Bounds on the volume and height distributions of the ambiguity function', IEEE Trans. Inform.Theory, Vol. IT-11, pp 207-214, April 1965.

25. J.R. KLAUDER, A.C. PRICE, S. DARLINGTON and W. ALBERSHEIM, 'The theory and design of chirp radars', Bell Sys. Tech. J., Vol. 39, pp 745-808, July 1960.

26. M.I. SKOLNIK, 'Introduction to radar systems', McGraw-Hill 1962.

27. M.H. ACKROYD, 'Amplitude and phase modulated pulse trains for radar', The Radio and Electronic Engineer, Vol. 41, pp 541-552, December 1971.

28. L.E. FRANKS, 'Signal theory', Prentice-Hall 1969.

29. J. DUGUNDJI, 'Envelopes and pre-envelopes of real waveforms', IRE Trans. Inform. Theory, Vol. IT-4, pp 53-57, March 1958.

30. C.W. HELSTROM, 'Statistical theory of signal detection', (2nd Ed.), Pergamon Press 1968.

31. L.R. RABINER and B. GOLD, 'Theory and application of digital signal processing', Prentice-Hall 1975.

32. J.W. COOLEY and J.W. TUKEY, 'An algorithm for machine calculation of complex Fourier series', Math. Comput., Vol. 19, pp 297-301, April 1965.

33. Special issue on the fast Fourier transform, IEEE Trans. Audio and Electroacoustics, Vol. AU-15, June 1967.

34. W.B. DAVENPORT and W.L. ROOT, 'An introduction to the theory of random signals and noise', McGraw-Hill 1968.

35. A.W. RIHACZEK, 'Optimum filters for signal detection in clutter', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-1, pp 297-299, December 1965.

36. L.J. SPAFFORD, 'Optimum radar signal processing in clutter', IEEE Trans. Inform. Theory, Vol. IT-14, pp 734-743, September 1968.

37. D.F. DELONG (Jr.) and E.M. HOFSTETTER, 'On the design of optimum radar waveforms for clutter rejection', IEEE Trans. Inform. Theory, Vol. IT-13, pp 454-463, July 1967.

38. W.D. RUMMLER, 'Clutter suppression by complex weighting of coherent pulse trains', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-2, pp 689-699, November 1966.

39. C.A. STUTT and L.J. SPAFFORD, 'A best mismatched filter response for radar clutter discrimination', IEEE Trans. Inform. Theory, Vol. IT-14, pp 280-287, March 1968.

40. J. SAKAMOTO et al., 'Coded pulse radar system', J. Faculty Eng. Univ. Tokyo, Vol. 27, pp 119-181, 1964.

41. L.W. MARTIONSON and R.J. SMITH, 'Digital matched filtering with pipelined floating point fast Fourier transforms', IEEE Trans. *Acoustics, Speech and Signal Processing*, Vol. ASSP-23, pp 222-234, April 1975.

42. B. LIU and A. PELED, 'A new hardware realization of high-speed fast Fourier transformers', IEEE Trans. Acoustics, Speech and Signal Processing, Vol. ASSP-23, pp 543-547, December 1975.

43. P.E. BLANKENSHIP and E.M. HOFSTETTER, 'Digital pulse compression via fast convolution', IEEE Trans. Acoustics, Speech and Signal Processing, vol. ASSP-23, pp 189-201, April 1975.

44. D.E. VAKMAN, '*Sophisticated signals and the uncertainty principle* in radar', Springer-Verlag 1968.

45. E.A. ROBINSON, 'Statistical communication and detection with special reference to digital data processing of radar and seismic signals', Griffin 1967.

46. J.H. MILLER, 'On the location of zeros of certain classes of polynomials with applications to numerical analysis', J. Inst. Maths. Applics., Vol. 8, pp 397-406, 1971.

47. H.B. VOELCKER, 'Toward a unified theory of modulation', Proc. IEEE, Vol. 54, pp 340-353, and pp 735-755, 1966.

48. H.M. ACKROYD, 'Computing the coefficients of high-order polynomials', Electron. Letters, Vol. 6, pp 715-717, October 1970.

49. E.L. KEY, E.N. FOWLE and R.D. HAGGERTY, 'A method of designing signals of large time-bandwidth product', IRE Int. Conv. Record, pt. 4, pp 146-155, 1961.

50. E.N. FOWLE, 'The design of FM pulse compression signals', IEEE Trans. Inform. Theory, Vol. IT-10, pp 61-67, January 1964.

51. M. BERNFELD, C.E. COOK, J. PAOLILLO and C.A. PALMIERI, 'Matched filtering,pulse compression and waveform design (four parts), Microwave J., pt 1, pp 56-64, October 1964; pt. 2, pp 81-90, November 1964; pt. 3, pp 70-76, December 1964; pt. 4, pp 73-81, January 1965.

52. D.C. CHU, 'Polyphase codes with good periodic properties', IEEE Trans. Inform. Theory, (Corr.), Vol. IT-18, pp 531-532, July 1972.

53. R. ROY and O. LOWENSCHUSS, 'Chirp waveform generation using digital samples', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-10, pp 10-16, January 1974.

54. L.I. BLUSTEIN, 'A linear filtering approach to the computation of discrete Fourier transform', IEEE Trans. Audio and Electroacoustics, Vol. AU-18, pp 451-456, December 1970.

55. E.A. SLOAN, 'Comparison of linearly and quadratically modified spectral estimates of gaussian signals', IEEE Trans. Audio and Electroacoustics, Vol. AU-17, pp 133-137, June 1969.

56. P. WOLFE, 'Review of nonlinear programming', in 'Optimization', R. Fletcher (Ed.), pp xi-xv, Academic Press 1969.

57. D.M. HIMMELBLAU, 'Applied nonlinear programming', McGraw-Hill 1972.

58. J.W. BANDLER and C. CHARALAMBOUS, 'Practical least pth optimization of networks', IEEE Trans. Microwave Theory and Techniques, Vol. MTT-20, pp 834-840, December 1972.

59. A.G. DECZKY, 'Synthesis of recursive digital filters using the minimum p-error criterion', IEEE Trans. Audio and Electroacoustics, Vol. AU-20, pp 257-263, October 1972.

60. G.C. TEMES and S.K. MITRA, 'Modern filter theory and design', John Wiley 1973.

61. H. EKBLOM and S. HENSIKSON, 'L_p-criteria for the estimation of location parameters', SIAM J. Appl. Math., Vol. 17, pp 1130-1141, November 1969.

62. M.J. BOX, D. DAVIS and W.H. SWANN, 'Non-linear optimization techniques', I.C.I. Monograph, No.5, Oliver and Boyd 1969.

63. M.J.D. POWELL, 'Finding minima of functions of several variables', in 'Numerical analysis', J. Walsh (Ed.), pp 143-157, Academic Press 1966.

64. D.J. WILDE, 'Optimum seeking methods', Prentice-Hall 1964.

65. R. HOOKE and T.H. JEEVES, 'Direct search' solution of numerical and statistical problems', J. Ass. Comp. Mach., Vol. 8, pp 212-229, 1961.

66. J.A. NELDER and R. MEAD, 'A simplex method for function minimization', Comput. J., Vol. 7, pp 308-313, 1965.

67. J.M. PARKINSON and D. HUTCHINSON, 'An investigation into the efficiency of variants on the simplex method', in 'Numerical methods for non-linear optimization', F.A. Lootsma (Ed.), pp 115-135, Academic Press 1971.

68. M.J.D. POWELL, 'A method for minimizing a sum of squares of non-linear functions without calculating derivatives', Comput. J., Vol. 7, pp 303-307, 1965.

69. R. FLETCHER, 'Generalized inverse methods for the best least squares solution of systems of non-linear equations', Comput. J., Vol. 10, pp 392-399, 1968.

70. M.J.D. POWELL, 'A Fortran subroutine for solving systems of non-linear algebraic equations', Harwell Report, AERE-R-5947, Nov. 1968.

71. K. LEVENBERG, 'A method for the solution of certain non-linear problems in least squares', Quart. Appl. Math., Vol. 2, pp 164-168, 1944.

72. D.W. MARQUARDT, 'An algorithm for least squares estimation of non-linear parameters', J. Soc. Ind. Appl. Math., Vol. 11, pp.431-441, 1963.

73. U. SOMAINI and M.H. ACKROYD, 'Uniform complex codes with low auto-correlation sidelobes', IEEE Trans. Inform. Theory, (Corr.), Vol. IT-20, pp 689-691, September 1974.

74. P.V. INDIRESAN and G.K. UTTARADHI, 'Iterative method for obtaining good aperiodic binary sequences', J. of Optimization Theory and Appl., Vol. 7, pp 90-108, February 1971.

75. U. SOMAINI, 'Binary sequences with good correlation properties', Electron. Letters, Vol. 11, pp 278-279, June 1975.

76. H.M. MARKOWITZ and A.S. MANNE, 'On the solution of discrete programming problems', Econometrica, Vol. 25, pp 87-110, 1957.

77. R.E. GOMORY, 'An algorithm for integer solutions to linear programs', Princeton-IBM Mathematical Research Project, Technical Report No.1, 1958.

78. K.M. GISVOLD and J. MOE, 'A method for non-linear mixed-integer programming and its application to design problems', ASME Trans. J. Engng. Ind., Vol. 92, pp 353-364, May 1972.

79. A.V. FIACCO and G.P. McCORMICK, 'Non-linear programming: sequential unconstrained minimization techniques', John Wiley 1968.

80. J. LINDER, 'Binary sequences up to length 40 with best possible autocorrelation function', Electron. Letters, Vol. 11, p 507, October 1975.

81. M.J.D. POWELL, 'A method for non-linear constraints in minimization problems', in 'Optimization', R. Fletcher (Ed.), pp 283-298, Academic Press 1969.

82. P.S. MOHARIR, 'Ternary Barker codes', Electron. Letters, Vol.10, pp 460-461, October 1974.

83. M.J.E. GOLAY, 'Complementary series', IRE Trans. Inform. Theory, Vol. IT-7, pp 82-87, April 1961.

84. G.R. WELTI, 'Quaternary codes for pulsed radar', IRE Trans. Inform. Theory, Vol. IT-6, pp 400-408, June 1960.

85. C.C. TSENG and C.L. LIU, 'Complementary sets of sequences', IRE Trans. Inform. Theory, Vol. IT-18, pp 644-652, September 1972.

86. W. SCHREMP and T. SEKIMOTO, 'Unique word detection in digital burst communications', IEEE Trans. Communications Technology, Vol. COM-16, pp 597-605, August 1968.

87. U. SOMAINI, 'Binary sequences with good autocorrelation and small mutual crosscorrelation properties', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-11, pp 1226-1231, November 1975.

88. N.B. CHAKRABARTI, 'Estimate of a bound for the minimum value of the crosscorrelation between two binary sequences', Electron. Letters, Vol. 6, pp 686-687, October 1970.

89. R.W. HAMMING, 'Numerical methods for scientists and engineers', McGraw-Hill 1962.

90. J.R. CAPRIO, 'Strictly complex impulse-equivalent codes and subsets with very uniform amplitude distributions', IEEE Trans. Inform. Theory, Vol. IT-15, pp 695-706, November 1969.

91. M.H. ACKROYD, 'The design of Huffman sequences', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-6, pp 790-796, November 1970.

92. F. FREUDENSTEIN and B. ROTH, 'Numerical solution of systems of non-linear equations', J. ASS. Comput. Mach., Vol. 10, pp 550-556, 1963.

93. S.K. MITRA and R.J. SHERWOOD, 'Estimation of pole-zero displacement of a digital filter due to coefficient quantization', IEEE Trans. Circuits and Sys., Vol. CAS-21, pp 116-124, January 1974.

94. M. ABRAMOWITZ and I. SEGUN, 'Handbook of mathematical functions', National Bureau of Standards, Applied Mathematics Series, Washington, D.C.: Government Printing Office 1965.

95. E.L. KEY, E.N. FOWLE and R.D. HAGGERTY, 'A method of sidelobe suppression in phase-coded pulse compression systems', M.I.T. Lincoln Lab., Lexington, Tech. Rep. 209, August 1959.

96. S. TWOMEY, 'The application of numerical filtering to the solution of integral equations encountered in indirect sensing measurements', J. of the Franklin Institute, Vol. 279, pp 95-109, February 1965.

97. E.A. ROBINSON and S. TREITEL, 'Principles of digital Wiener filtering', Geophys. Prosp., Vol. 15, pp 311-333, 1967.

98. M.H. ACKROYD and F. GHANI, 'Optimum mismatched filters for sidelobe suppression', IEEE Trans. Aerospace and Electron. Sys., Vol. AES-9, pp 214-218, March 1973.

99. W. NEF, 'Lehrbuch der linearen Algebra', Birkhauser-Verlag, 1966.