INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA

UMI®
800-521-0600

Virtual Matched Filtering
A New Hybrid CDMA Code Acquisition Technique Under Doppler And Higher Loads

Haiying Zhu

A Thesis
In the Department of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Applied Science at
Concordia University
Montreal, Quebec, Canada

March 1999

ABSTRACT

Virtual Matched Filtering A New Hybrid CDMA Code Acquisition Technique Under Doppler And Higher Loads

Haiying Zhu

A new technique for spreading code acquisition is presented. The technique utilizes a multiplicity of communication and processing techniques namely, concatenated short signature coding, Reed Solomon adaptive error / erasure decoding, and pipelining to provide the minimum possible acquisition time in severe overload CDMA and carrier offsets conditions. The overall implementation is equivalent to a very long physical matched filter but is relatively immune to the Doppler degradation. These features would be shown by analysis results. We also propose two strategies in conjunction with the new virtual matched filtering, and compare their performance to known code acquisition techniques.

Dedicated to My Parents and My Whole Family for Their Love and Encouragement

ACKNOWLEDGEMENTS

At the outset I would like to express my most sincere gratitude to my thesis supervisor, Dr. A. K. Elhakeem, for his guidance, support, patience and encouragement during the entire course of this thesis. This work was born from his ideas and his intellectual properties, also he provided constructive suggestions to enable me to demonstrate this new technique by mathematical analysis and computer program. I am deeply touched by his scientific attitude toward knowledge and rich experiences in the area of communication. More important to me, he doesn't only give me so much guidance in the research work, but also enlightens me on the love of knowledge. Here I would like to express my greatest admiration to my supervisor again!

I give my special thanks to Yan Xin Qi Gong which helps me both spiritually and physically. It is an important part in my life.

Also this is dedicated to Sun Luck who is always like a shining star when I am in the dark. Among others, I'd like to thank the great group of friends I have in Montreal who inspired me during the course of this work. Finally I would like to express my thanks to my wonderful family, my parents, my sister, brother, antis and uncles. They are always standing besides me and helping with love and encouragement during the entire period of the study of my Masters.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

For the sake of simplicity, in this list, the included symbols are within the range of the VMF, namely chapter 3, 4, 5 and 6.

n_1	Number of stages in shift register of short Gold Code
n_2	Number of stages in shift register of back-to-back shift register
N	Short Gold Code length
N_1	BCH code word length
d	Hamming distance of BCH code words
q	Sampling rate per chip
M	Number of EDMs
M'	Number of matched filters in M parallel search
U	Maximum number of users
u	Current number of users
E_{max}	Maximum number of erasures
E	Adaptive number of erasures
T_c	PN code chip duration
$\overline{T_s}$	Sampling time (equals to T_c/q)
V	Voltage output of short code matched filter
ω_c	Carrier frequency assumed for all users
Δf_c	Doppler frequency offset of the intended user

Δ_i	Carrier and Doppler offset of user i
α	Signal to interference ratio
ϕ_1	Intended user phase angle
ϕ_i	Interference user phase angle
P	Signal power
r(t)	Received signal
$g_1(t)$	Signal of intended user in one code chip
$g_I(t)$	Interference of i th user
$a_x(t)$	Inphase received signal
$a_y(t)$	Quadrature component of the received signal
n(t)	Additive White Gaussian Noise
$\frac{N_0}{2}$	Power spectrum density of AWGN
R_{xy}	Gold Code three valued cross correlation
$\bar{l}$	Mean value of Gold Code three valued cross correlation
$\overline{l^2}$	Mean-square value of Gold Code three valued cross correlation
T_i	Integration time to investigate one search cell
T_{fa}	False alarm penalty time
T_p	Software processing time of BCH decoder
P_D	Probability for detection
P_M	Probability for miss
P_{fa}	Probability for false alarm
R	Uncertainty region

ζ	Threshold
$\bar{s}$	Mean of the signal
σ_s	Standard variance of the of the signal
$\bar{n}$	Mean of the interference
σ_n	Standard variance of the interference
Q(x)	Probability of the random variable X is greater than a given value x
ε_s	Probability of erasure under signal presents
ε_n	Probability of erasure under signal absents
P_{ss}	Decision error of heavy interference with erasure under signal presents
P_{es}	Decision error of light interference without erasure under signal presents
P_{sn}	Decision error of heavy interference with erasure under signal absents
P_{en}	Decision error of light interference without erasure under signal absents
P_{ws}	BCH decoding error under signal presents
P_{wn}	BCH decoding error under signal absents
$\overline{T_{s1}}$	Average acquisition time of VMF for strategy 1 under processing time is 0
$\overline{T_{s2}}$	Average acquisition time of VMF for strategy 2 under processing time is 0

$\overline{T_{s11}}$ Average acquisition time of VMF for strategy 2 under processing time $\neq 0$

$\overline{T_{s4}}$ Average acquisition time of serial search

$\overline{T_{s5}}$ Average acquisition time of long matched filter

$\overline{T_{s6}}$ Average acquisition time of parallel matched filter

Chapter 1

Introduction to Spread Spectrum Communication

In this chapter, a brief introduction to the Spread Spectrum Systems ---- the Direct Sequence, Frequency Hopping, Time Hopping, Hybrid Spread Spectrum System and Chirp System will be presented. Among these, the important property of Gold Code cross-correlation function is emphasized. These will be followed by a brief summary of the layout of this thesis.

1.1 Spread Spectrum Communications

Spread Spectrum Communications is a modulation and demodulation technique in which the transmission bandwidth employed is much greater than the minimum bandwidth required to transmit the digital information. A number of modulation techniques use a transmission bandwidth much larger than the minimum required for data transmission, but are not spread spectrum modulations. To be classified as a spread spectrum sys tem, the modem must have the following characteristics.

- Bandwidth occupancy by the transmitted signal must be much larger than the information bit rate.
- Demodulation must be accomplished in part by correlation of the received signal with a replica of the signal used in the transmitter to despread the information signal
- The codes used for spreading the signal must be maximal length sequences, which are a type of pseudo noise codes (PN sequences).

Some of the salient features of spread spectrum techniques are

- Interference rejection capability.
- Multiple access.
- Improved spectral efficiency.

Spread spectrum techniques may be broadly classified as

1. Direct Sequence (DS)

2. Frequency Hopping (FH)
3. Time Hopping (TH)
4. Chirp
5. Hybrid Systems (which combine two or more of the above techniques)

1.1.1 Direct Sequence Systems (DS)

Bandwidth spreading by direct modulation of a data modulated carrier by a wide-band spreading signal or code is called Direct Sequence (DS) spread spectrum. Fig. 1.1 and Fig. 1.2 show the block diagram of a simplified direct sequence transmitter and receiver system. The idea behind spread spectrum is to transform a signal with bandwidth B into a noise-like signal of much larger bandwidth B_{ss}.

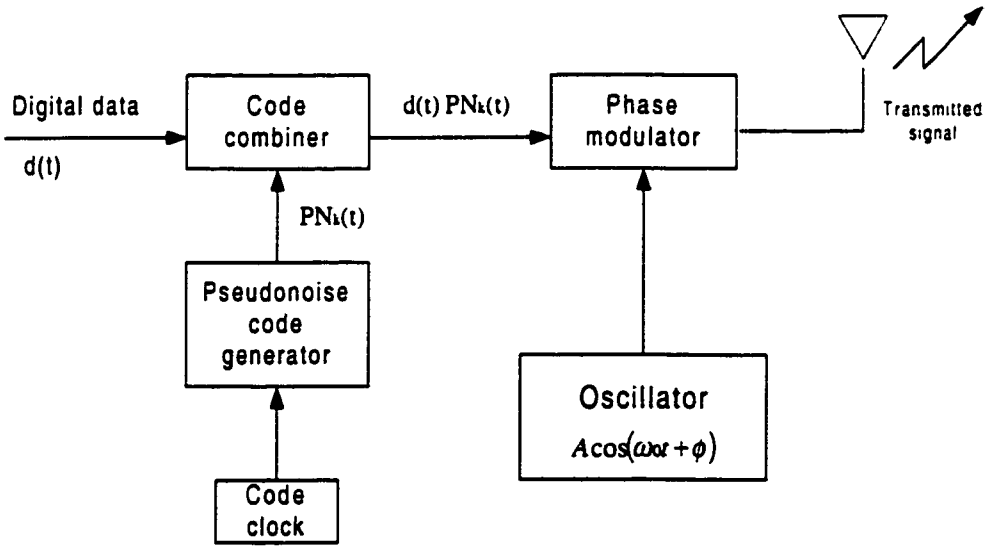


Fig. 1.1 Block diagram of a direct sequence transmitter system.

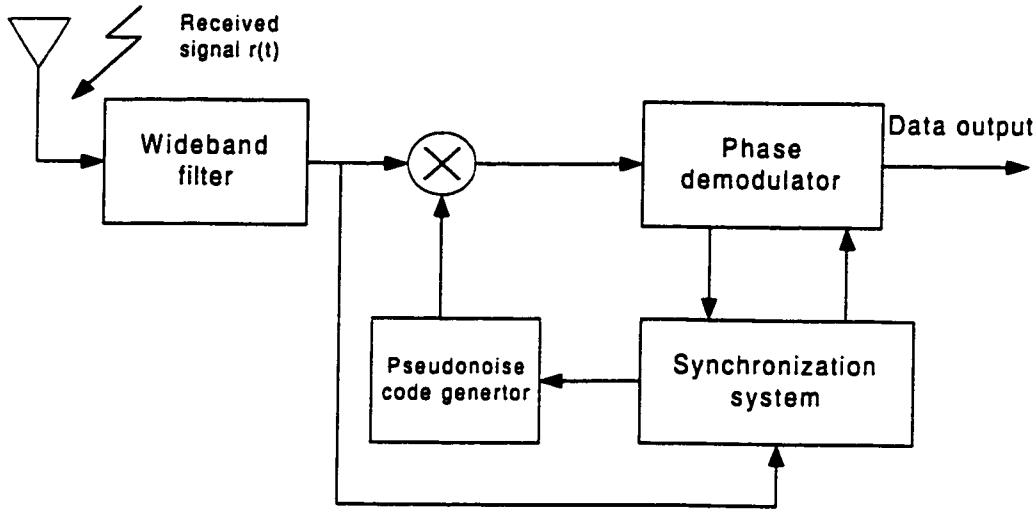


Fig. 1.2 Block diagram of a direct sequence receiver system.

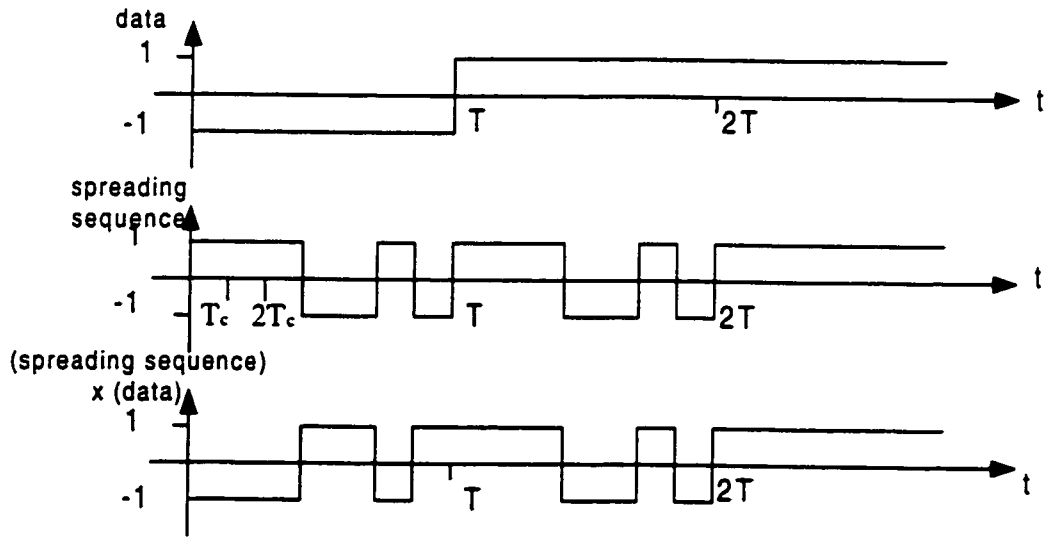


Fig. 1.3 Time waveforms involved in generation a direct sequence signal

We see from Fig. 1.3 that the spreading is accomplished by multiplying the modulated information-bearing signal by a binary $\{\pm 1\}$ base band code sequence waveform, $PN_k(t)$, where T_c, duration of a chip period, is much less than T_s, duration of a data bit period. The processing gain of the DS system (processing gain is the performance improvement achieved through use of spread spectrum techniques over techniques without spread spectrum) is equal to the number of chips (pulses of code sequence) in a symbol interval

$$N = \frac{T_s}{T_c} \tag{1.1}$$

This is also equal to B_{ss}/B, which is typically 10-30 dB. Hence the power of the radiated spread spectrum signal is spread over 10-1000 times the original bandwidth, while its power spectral density is correspondingly reduced by the same amount. The other requirement of the spread spectrum signal is that it be "noise-like". That is each spread spectrum signal should behave as if it were "uncorrelated" with every other spread signal using the same band. In practice, the cross correlation used need not be zero, i.e. the variability of differential propagation delay deference between the paths from the central station to the various users may force these codes to practically become semi-orthogonal rather than completely orthogonal.

The code sequence may be thought of as being (pseudo) randomly generated so that each binary chip can change (with probability ≡ ½) every T_c sec. Thus the signal for the *i*th transmitter is:

$$S_i(t) = Ad(t)PN_i(t)\cos(\omega_0 t + \phi) \tag{1.2}$$

where d(t) is the data modulation (assumed to be ± 1 for BPSK signaling), A is the amplitude of the BPSK waveform, $f_0 = \frac{\omega_0}{2\pi}$ is the carrier frequency, and ϕ is a random phase. Since T_c is much less than T_s, the ratio of the spread bandwidth, B_{ss}, to the unspread bandwidth, B, is given by $B_{ss}/B = T_s / T_c = N$, the processing gain. It is clear that a receiver with access to PN_I (t), and synchronized to the spread spectrum transmitter, can receive the data signal, d(t), by a simple correlation. That is, in the interval [0,T], if the date symbol is d_1, which can take on values ± 1, then

$$\frac{2}{T}\int_0^T PN_i(t)S_i(t)\cos(\omega_0 t + \phi)dt = Ad_1 = \pm A \tag{1.3}$$

A properly designed spreading sequence may have the following properties [8]:

a) In a long sequence, about half the chips will be +1 and half will be −1;

b) A run of length r chips of the same sign will occur about 2^{-r} *l* times in a sequence of *l* chips;

c) The autocorrelation of the sequence $PN_I(t)$ and $PN_I(t+\tau)$ will be very small except in the vicinity of $\tau = 0$;

d) The crosscorrelation of any two sequences $PN_I(t)$ and $PN_I(t+\tau)$ will be small.

1.1.2 Gold Code auto-correlation function and CDMA

An important class of sequences called Gold Code is well known to exhibit properties a),b) and c). The code length L of any pseudoniose (PN) code generator is dependent upon the number of shift registers K,

$$L = 2^K - 1 \tag{1.4}$$

In particular, the auto-correlation function,

$$R_i(\tau) = \frac{1}{T_q}\int_0^{T_q} PN_i(t)PN_i(t+\tau)dt \tag{1.5}$$

is given by

$$R_i(\tau) = \begin{cases} 1 - \dfrac{\tau}{T_q}\left(1 + \dfrac{T_c}{T_q}\right); & 0 \le \tau \le T_c \\ -\dfrac{T_c}{T_q}; & T_c \le \tau \le (N-1)T_c \\ \tau - \dfrac{T_q - T_c}{T_c}\left(1 + \dfrac{T_q}{T_c}\right) - \dfrac{T_q}{T_c}; & (N-1)T_c \le \tau \le NT_c \end{cases} \tag{1.6}$$

where $T_q = NT_c$ is the period of the sequence and $R_i(\tau)$ is also periodic with period T_q. The enhancement in performance obtained from a DS spread spectrum signal through the processing gain and coding gain can be used to enable many DS spread spectrum signals to occupy the same channel bandwidth provided that each

signal has its own distinct PN sequence. Thus it is possible to have a large community of relatively uncoordinated users transmit messages simultaneously over the same channel bandwidth. This type of digital communication in which each user (transmitter-receiver pair) has a distinct PN code for transmitting over a common channel bandwidth is called either code division multiple access (CDMA) or spread spectrum multiple access (SSMA).

In the demodulation of each PN signal, the signals from the other simultaneous users of the channel appear as an additive interference. The level of interference varies depending on the number of users at any given time. A major advantage of CDMA is that a large number of users can be accommodated if each transmits messages for a short period of time. In such a multiple access system it is relatively easy either to add new users or to decrease the number of users without disrupting the system.

If we assume that all signals have identical average powers, the desired signal-to-interference power ratio at a given receiver, neglecting AWGN and before despreading, is

$$\frac{S_{av}}{I_{av}} = \frac{S_{av}}{(N_u - 1)S_{av}} = \frac{1}{N_u - 1} \tag{1.7}$$

where N_u is number of simultaneous users.

Even all the users have same power, we could still detect the intended user by multiply the identical phase of the code which is different from each user at the receiver end in the despreading process.

Since CDMA capacity is only interference limited (unlike FDMA and TDMA capacities which are primarily bandwidth limited), any reduction in interference converts directly and linearly into an increase in capacity. Voice activity and spatial isolation were shown to be sufficient to render CDMA capacity at least double that of FDMA and TDMA under similar assumptions for mobile satellite applications.

In Fig. 1.4, a model of DS-CDMA system is presented. There are many attributes of direct sequence CDMA which are of great benefit to the cellular system:

- Voice activity cycles;
- No equalizer needed;
- One radio per site;
- Soft handoff;
- No guard time in CDMA;

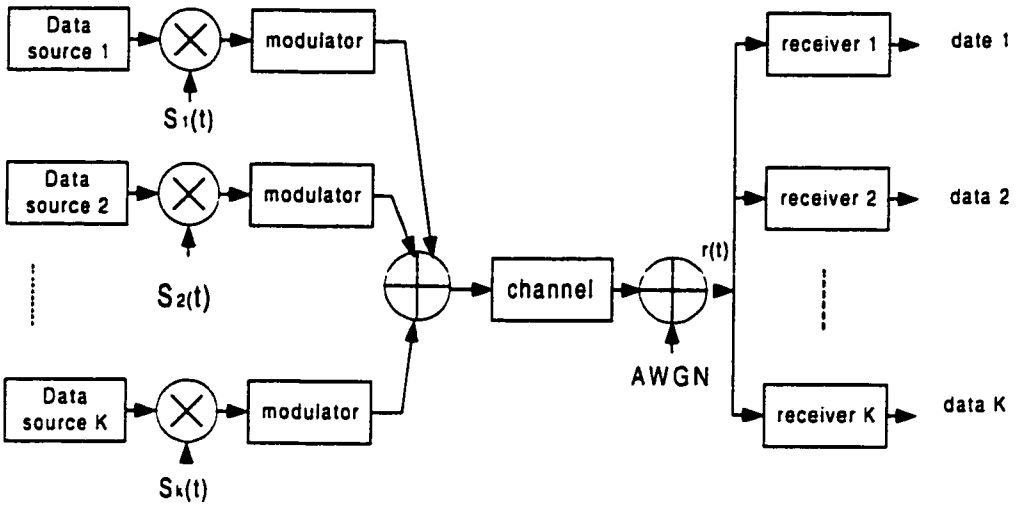


Fig 1.4 Model of DS-CDMA system

- Sectorization for capacity;
- Less fading;
- Easy transition;
- Capacity advantage;
- No frequency management or assignment needed;
- Soft capacity;
- Coexistence or ability to overlay;
- For micro-cell and in-building systems.

Some of the important components of the DS/CDMA system are

- System Features:

Frequency assignment.

System Pilot Acquisition.

Mobile station assisted soft handoff.

Variable data rate Vocoder.

- Link waveform:

  Forward Link Waveform.

  Reverse Link Waveform.

- Network and Control:

  CDMA Power Control.

  The Sync Channel.

  Framing and Signaling.

  Service Options.

  Authentication, Encryption and Privacy.

- System Functional Description:

  MTSO Functions.

  CDMA Equipment Design.

  Dual Mode Mobile Station/Portable.

There are also disadvantages of CDMA:

- Near-far effect;
- Self-jamming;
- Stringent synchronization requirements.

1.1.3 Frequency Hopping Systems (FH)

In frequency hopping the modulated signal is first generated, then the spread spectrum technique consists of changing pseudo-randomly the center frequency of the transmitted signal every T_H sec, according to the output from a PN generator, so that the hop rate is

$$f_H = \frac{1}{T_H} \tag{1.8}$$

Figure 1.5 shows the block diagrams of the transmitter for a frequency-hopped spread spectrum system.

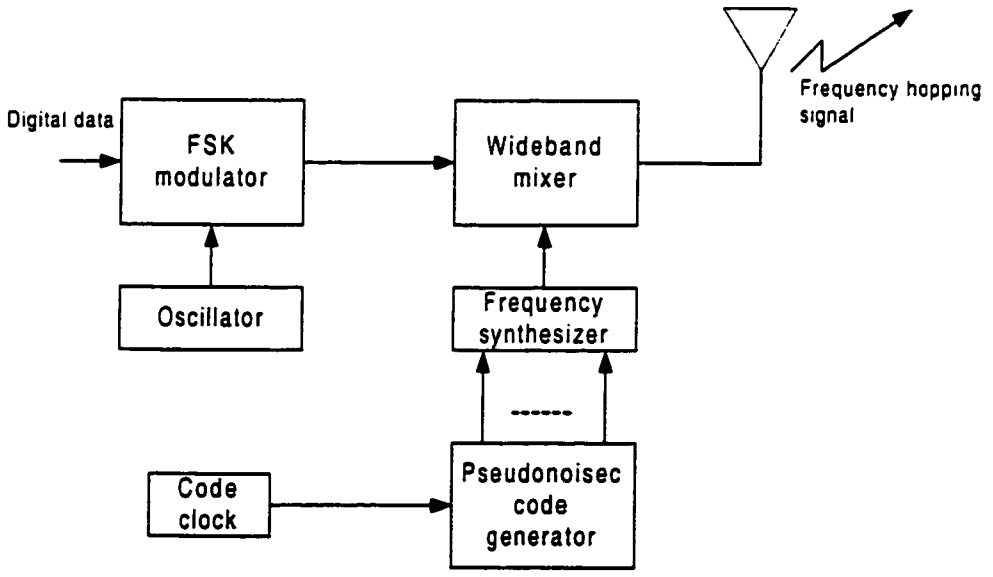


Fig 1.5 Block diagram of the transmitter of a frequency hopping system

The modulation is either binary or M-ary FSK. For example, if binary FSK is employed, the modulator selects one of two frequencies corresponding to the transmission of either a 1 or a 0. The resulting FSK signal is translated in frequency by an amount that is determined by the output sequence from the PN generator which, in turn, is used to select a frequency that is synthesized by the frequency synthesizer. This frequency is mixed with the output of the modulator and the resultant frequency-translated signal is transmitted over the channel. For example, m bits from the PN generator may be used to specify 2^m-1 possible frequency translations.

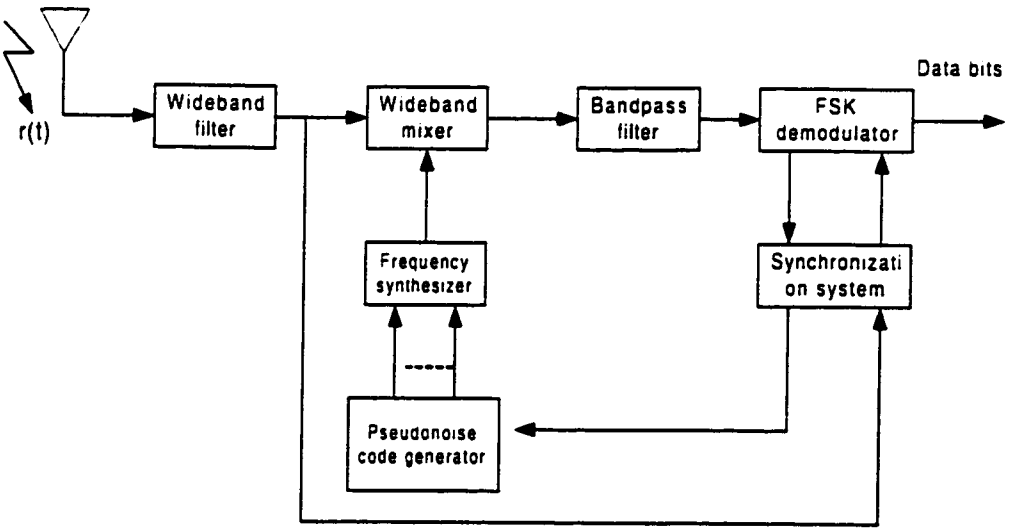


Fig 1.6 Block diagram of the receiver of a frequency hopping system

At the receiver, as shown in figure 1.6,we have an identical PN generator, synchronized with the received signal, which is used to control the output of the frequency synthesizer. Thus the pseudo-random frequency translation introduced at the transmitter is removed at the receiver by mixing the synthesizer output with the received signal. The resultant signal is demodulated by means of an FSK demodulator. A signal for maintaining synchronism of the PN generator with the frequency-translated received signal is usually extracted from the received signal.

Although PSK modulation gives better performance than FSK in an AWGN channel, it is difficult to maintain phase coherence in the synthesis of the frequencies used in the hopping pattern and also in the propagation of the signal over the channel as the signal is hopped from one frequency to another over a wide bandwidth. Consequently FSK modulation with noncoherent detection is usually employed with FH spread spectrum signals.

Frequency hopping could be done slowly (one hop per many symbols) or fast (many hops per symbol). Due to limitation of today's technology, the FH is using a slow hopping pattern. Fast frequency hopping is employed in anti-jamming (AJ) applications when it is necessary to prevent a type of jammer, called a follower jammer, from having sufficient time to intercept the frequency and retransmit it along with adjacent frequencies so as to create interfering signal components. However, there is a penalty incurred in subdividing a signal into

several frequency-hopped elements because the energy from these separate elements is combined noncoherently. Consequently the demodulator incurs a penalty in the form of a noncoherent combining loss. The total frequency spread B_{ss} is equal to the total number N of distinct frequencies used for hopping times the bandwidth occupied by each frequency.

For slow frequency hopping, $N/T=B_{ss}$, or $N=B_{ss} / B$ where $B \cong \frac{1}{T}$ and T is the duration of the data symbol. The spectrum spreading in FH is measured by the processing gain, PG, as

$$PG = 10\log N(dB) \tag{1.9}$$

To make the FH signal "noise-like", the frequency hopping pattern is driven by a "pseudo-random" number generator having the property of delivering a uniform distribution for each frequency that is "independent" on each hop.

For a Direct Sequence (DS) spread spectrum system, binary PSK (binary FSK) or M-ary PSK (M-ary FSK) may be used for transmission but M-ary transmission has the advantage that more data per chip (hop) can be transmitted.

Frequency hopping system advantages:

- Greater amount of spreading;

- Can be programmed to avoid portions of the spectrum (not requiring a contiguous band);
- Relatively short acquisition time;
- Less affected by near-far problem for highly loaded system.

Disadvantages of a FH system:

- Complex frequency synthesizer;
- Not useful for range and range rate measurement;
- Error correction required.

1.1.4 Time Hopping Systems (TH)

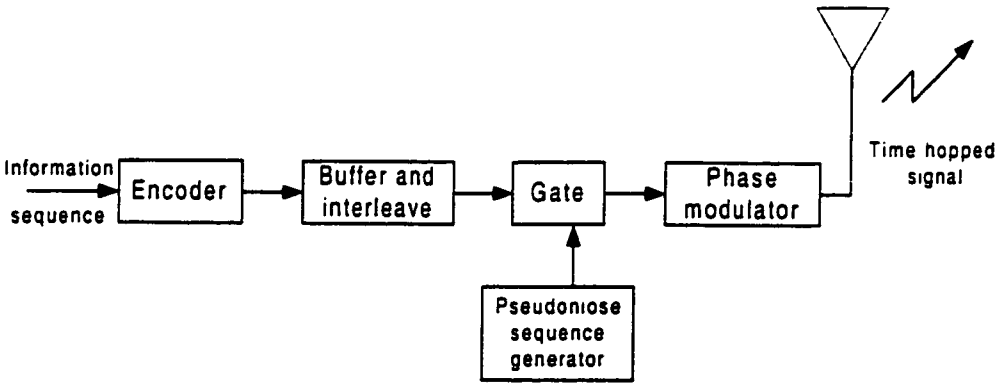


Fig 1.7 Generic transmitter of a time hopping system

In time hopping systems, spreading of the spectrum is achieved by compressing the information signal in the time domain. That is the time hopping systems control their transmission time and period with a code sequence in the same way that frequency hoppers control their frequency. The message bit period, T, may be divided into a number of non-overlapping subintervals. One of the subintervals is selected pseudo-randomly each bit interval, and a pulse is transmitted which conveys the value of the bit. The subinterval selected, τ, is independent of the bit value and is independent bit-to-bit. The position of the pulse is uniformly distributed over the bit interval, but could be predicted by a properly synchronized, friendly receiver. In a sense, TH uses time slots or subintervals in a way which is analogous to the manner FH uses frequency cells. However, in TH the power level is increased to hold the energy per bit constant. Consequently, like FH, the signal power spectral density for TH signals is about the same as a conventional signal having the same energy per bit.

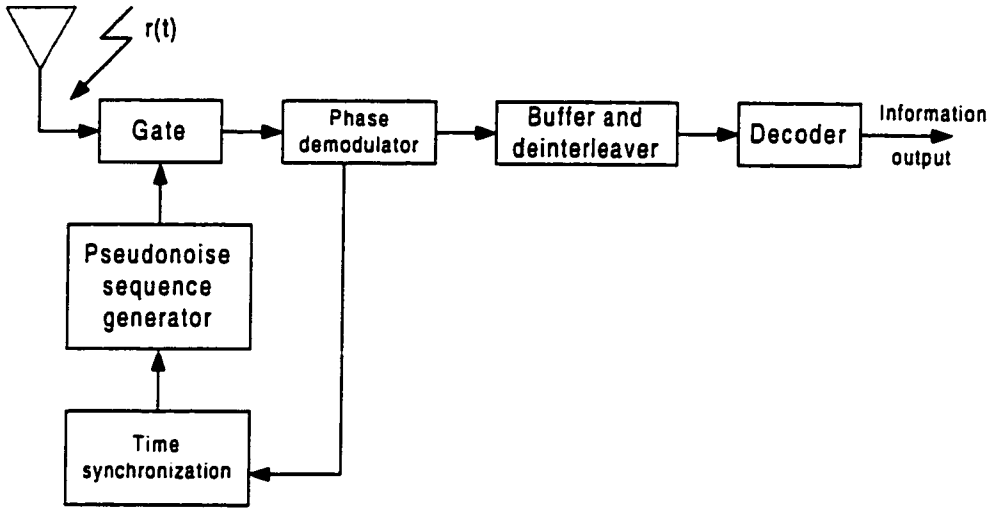


Fig 1.8 Generic receiver of a time hopping system

In the TH scheme described above, the pulse rate in TH is the same as the bit rate of the message it transmits. Rather than transmit one pulse in every bit interval, it is also possible to store up bits over several intervals and then transmit several bits together.

Figure 1.7 and 1.8 show the block diagram of a time hopping system. Interference among simultaneous users in a time hopping system can be minimized by coordination the times at which each user can transmit a signal, which also avoids the problem of very strong signals at a receiver swamping out the effects of weaker signals. In a non-coordinated system, overlapping

transmission bursts will result in message errors, and for this it will normally require the use of error-correction coding to restore the proper message bits.

Advantages of a time hopping system:

- High bandwidth efficiency;
- Implementation simpler than FH.

Disadvantages of a time hopping system:

- Long acquisition time;
- Error correction needed.

1.1.5 Hybrid spread spectrum systems

Hybrid spread spectrum systems are made up by combining two or more of, direct- sequence, frequency-hopping, time-hopping to offer certain advantages of a particular technique while avoiding the disadvantages, or to offer very wide-band and / or very high processing gain, or to combine some of the advantages of two or three types of systems in a single system, and
minimize the disadvantages of those types. Many different hybrid combinations are possible. Some of these are:

- DS /FH
- DS /TH

- FH /TH
- DS /FH /TH

In this thesis we will concentrate on only one hybrid system, namely DS /FH system.

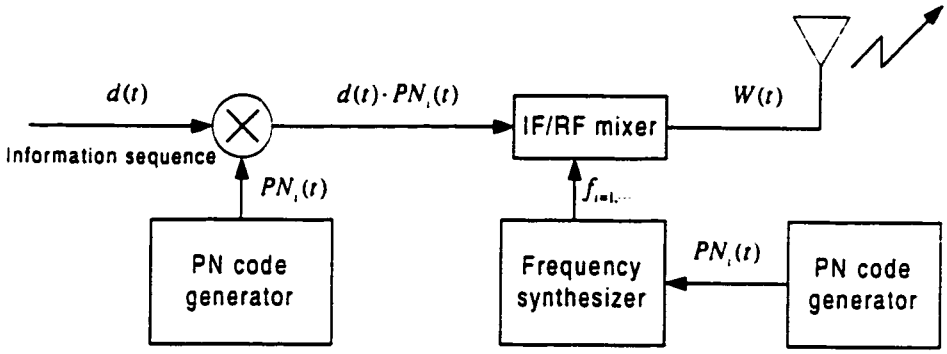


Fig. 1.9 Block diagram of a typical DS / FH transmitter system

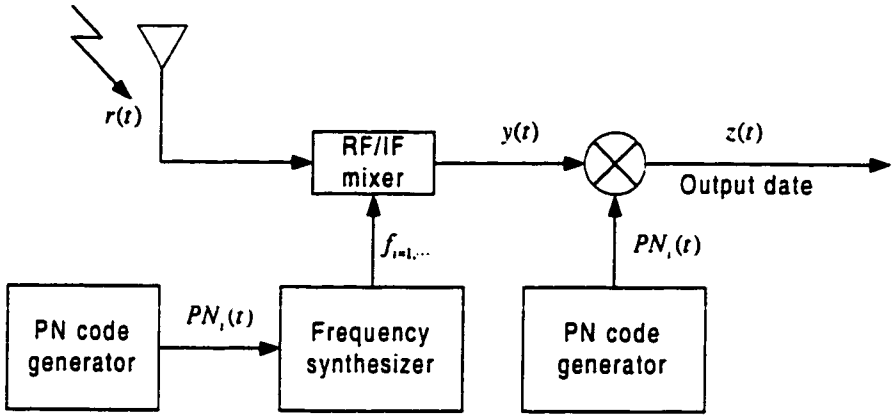


Fig. 1.10 Generic receiver of a DS /FH system

Figure 1.9 and 1.10 show the block diagrams of a typical DS /FH system. In this system PN code is used to spread the signal to an extent limited by either code generator acquisition time or speed and the frequency hopping would be used to increase the frequency spread. The DS code rate is normally much faster than the rate of frequency hopping, and the number of frequency channels available usually much smaller than the number of code chips. The signal transmitted on a single hop consists of a DS spread spectrum signal which is demodulated coherently. However, the received signals from different hops are combined non-coherently (envelope or square-law combining). Since coherent detection is performed within a hop, there is an advantage obtained relative to a pure FH system. The hybrid DS /FH processing gain is a function of

$$BW_{RF} / R_{inf\,o} \tag{1.10}$$

where BW_{RF} is the RF bandwidth of the transmitted signal and R_{info} is the bit rate of the information. A high processing gain is easily achieved by the FH technique, and the signal keeps all the great advantages of the direct sequence modulation as the resistance to narrow band interference, and multiple path rejection. However, the price paid for the gain in performance is an increase in complexity, greater cost, and more stringent timing requirements.

1.1.6 Chirp Systems

This technique is one of the only types of spread spectrum modulation that does not necessarily employ coding but does use a wider bandwidth than the minimum required, so that it can realize the processing gain. This form of spread spectrum has found its main application in radar and is also applied in communications. Chirp transmissions are characterized by pulsed RF signals whose frequency varies in some known way during each pulse period. The advantage of these transmissions for radar is that significant power reduction is possible. The receiver used for chirp signals is a matched filter, matched to the angular rate of change of the transmitted frequency-swept signal.

1.2 Thesis Outline

The information and work presented in this thesis is organized as follows:

In chapter 1, a brief introduction of Spread Spectrum is presented, namely, Direct Sequence System, Frequency Hopping System, Time Hopping System, Hybrid System and Chirp System. Among these, the important property of Gold Cold auto-correlation function is emphasized.

In chapter 2, recount some of the acquisition schemes used for PN synchronization. A detailed discussion of the Single Dwell serial search scheme is presented and brief discussions of the Multiple Dwell; PN search procedures for Non Uniformly Distributed signal location; and the Matched Filter Detection technique are presented.

In chapter 3, we proposed a new hybrid acquisition scheme called Virtual Matched Filter. We give a detailed description of the scheme and its various parameters are presented. Certain salient aspects of the acquisition scheme are recounted here as they reflect the advantages of our VMF system.

In chapter 4 and 5, we give the analysis of our system. The goal is to obtain the acquisition time and the performance. We begin with the detection of the short Gold Code, followed by evaluation of the BCH decoding error, then verification time of the long signature code is reached.

In chapter 6, we compare our VMF with serial search, long matched filter and parallel matched filter. Results are shown in figures with respect to various parameters. Right through these discussions, we could see the property of our VMF and the advantages compared with other systems.

In chapter 7, summary, conclusion and suggestions for future work is presented. And the appendix shows the program in Fortran which are employed in the computation in the previous process.

Chapter 2

PN Code Acquisition Schemes

In this chapter, we begin with an introduction to the different types of acquisition schemes. We emphasis on Serial Search and Matched Filter approaches. In Serial Search, a detailed description of the single dwell serial search will be developed using the state transition diagram or Markov Chain acquisition model. The acquisition time performance of the single dwell search scheme will be discussed. This is followed by the multiple dwell serial search – as an extension of the single dwell search scheme. Matched filter is also developed in detail with its time performance. Then appears a brief look at the modified sweep strategy for non uniformly distributed signal location.

2.1 Introduction

Initial spreading waveform synchronization is an extremely important problem in spread-spectrum. In fact, overall system performance is often limited by the performance of the synchronizer. The simplest synchronization scheme is to sweep the receiver spreading waveform phase until the proper phase is sensed. Stepped serial search over all potential waveform phases will usually achieve lower synchronization times than the swept search. After each phase step in a stepped serial search system, the correctness of the phase must be evaluated. This is accomplished by attempting to despread the received waveform using the trial spreading waveform phase. When the phase is correct, the input signal spectrum is collapsed and energy appears at the output of a narrow-band filter.

The primary function of a direct sequence (DS) spread spectrum receiver is to despread the received pseudo-noise (PN) code. The process involves generating a local replica of the PN code in the receiver and then synchronizing this local PN signal to the received one. The process of synchronizing the local and received PN signals is accomplished in two stages. Initially, a coarse alignment of the two PN signals is produced within a small residual relative timing offset. This process of bring incoming PN code has been acquired, a fine synchronization system takes over and continuously maintains the best possible waveform alignment by

means of a closed loop operation. The process of maintaining the two codes in fine synchronism is referred to as PN tracking.

2.2 Serial search

A widely used technique for initial synchronization is to search serially through all potential code phases and frequencies until the correct phase and frequency are identified. Each reference phase/ frequency is evaluated by attempting to despread the received signal. If the estimated code phase and frequency are correct, despreading will occur and will be sensed. If the estimated code phase or frequency is incorrect, the received signal will not be despread, and the reference waveform will be stepped to a new phase/ frequency for evaluation. This technique is called serial search. There are two important probabilities in this scheme, one is P_{fa}, the probability of false alarm, the other is P_d, the probability of correct detection. They are both function of evaluation (integration) time and signal to noise ratio.

Serial search acquisition schemes have been referred to as low decision rate detectors since a large number of spreading code symbols must typically be

received to make a correct/incorrect decision on the hypothesized reference spreading code phase/frequency.

2.2.1 Single-dwell serial acquisition

A simple model of a single dwell serial PN acquisition system is illustrated in Fig. 2.1. The model employs a standard type, Non-Coherent (square-law) detector. Briefly, the received signal plus noise is actively correlated with a local replica of the PN code and then passed through a bandpass filter. The output of the filter is then square-law envelope detected with the detector output being integrated for a fixed time duration, τ_d (the dwell time), and then compared to a preset threshold.

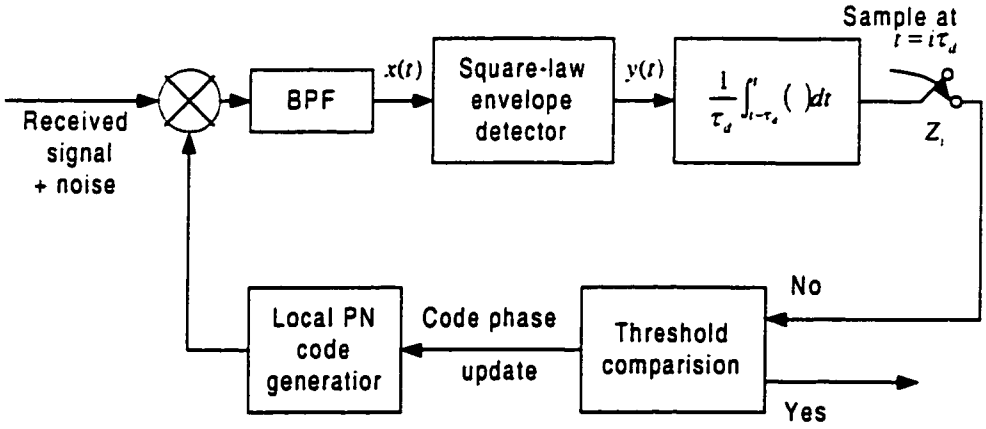


Fig. 2.1 Block diagram of a single dwell PN acquisition system with Non-Coherent detection

1. State-transition Diagram

The state-transition diagram is also called Markov Chain. We use this to illustrate the acquisition process and it results from how the detector output is processed. If the detector output is above the preset threshold, then a *hit* is declared. If this hit actually represents a true hit (i.e., the correct code phase has been determined), then the code has been officially acquired and the search comes to an end. If the hit is a false alarm, then the verification process cannot be consummated and the search must continue. In general the verification is characterized by an extended dwell time (e.g., $K\tau_d.K>>1$) assumed to be fixed and an entering into the code tracking loop mode. Since a true hit corresponds to a single code phase position, which can occur only once per search through the code, the time interval $K\tau_d$ sec can be regarded as the *Penalty time* associated when a false alarm is declared. If the detector output falls below the preset threshold, then the local PN code generator steps to its next position and the search proceeds. Thus, at each test position, one of the two events can occur, namely, a false alarm i.e., and indication that acquisition has occurred when the PN codes are actually misaligned or no false alarm. The former event can occur with a probability of P_{fa} whereas the latter can occur with a probability of $(1-P_{fa})$, hence the Markov Chain model.

Furthermore, at the true code phase position, either a correct detection can happen, i.e., an indication that acquisition has occurred when the PN codes are indeed aligned, with probability P_D, or no detection occurs, with probability $(1-P_D)$. In the absence of any a-priori information regarding the true code phase position, the uncertainty in misalignment between the received PN code and the local replica of it could be as much as a full code period. For long PN codes, the corresponding time uncertainty to be resolved could typically be quite large. In order to represent a reasonable compromise between the time required to search through this code phase uncertainty region and the accuracy within which the final alignment position is determined, the amount, which the local PN code generator is stepped in position as the search proceeds, must be judiciously chosen. It is typical in practice to require that the received and the local PN code signals be aligned to within one-half a code chip period $T_c/2$ before relinquishing control to the fine synchronization system (tracking loop). In accordance with this requirement, the time delay of the local PN code signal would be retarded (or advanced) in discrete steps of one half a chip period and a check for acquisition made after each step. If T_u represents the uncertainty time to be resolved, and we have

$$T_u = N_u T_c \tag{2.1}$$

then $q=2T_u$ would be the number of possible code alignments (these are referred to as *cells*) to be examined during each search through the uncertainty region.

The time to declare a true hit, T_{ACQ} (which is generally referred to as the *Acquisition Time*), is a random variable and generally depends upon the initial code phase position of the local PN generator relative to that of the received code. The probability density function gives a complete statistical description of this random variable. The determination of the PDF would ultimately allow computation to the system, but one is often content with measuring performance in terms of the first two moments of the PDF of T_{ACQ} namely the *mean* acquisition time $\overline{T_{ACQ}}$, and the acquisition time *variance* σ^2_{ACQ}.

2. Serial PN acquisition system: Single Dwell

In the following presentation of the performance of the acquisition time of a single dwell system, it will be assumed, for the sake of simplicity, that no code Doppler is present in the received signal and that the probability P_D is a constant (i.e., time invariant) which is equivalent to assuming that only one cell corresponds to a "correct" code alignment. In the absence of *a-priori* knowledge concerning the relative code phase position of the received and

locally generated codes, the local PN generator is assumed to start the search at any code phase position with equal probability. Stated mathematically, the probability P_1 of having the signal present (true hit) in the first cell searched is 1/q and the probability of not being present there is 1-1/q. For example, if the number of code chips to be searched is denoted by N_u and the search proceeds in half chip increments, then the number of cells to be searched is $q=2N_u$ and $P_1=1/2N_u$.

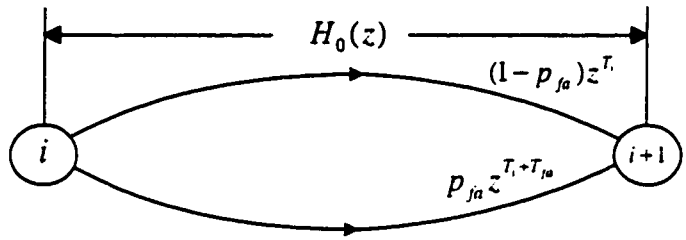


Fig 2.2 State-transition diagram of single-dwell system

Fig. 2.2 illustrates the generating function flow graph for the state-transition diagram which characterizes the acquisition process of the single dwell system. As is customary in such flow graphs, each branch is labeled with the product of the transition probability associated with going from the branch to the node at its terminating end, and an integer power of a parameter denoted by z. The parameter z is used to mark time as one proceeds through the graph and its power represents the number of time units (dwell times) spent in

traversing that branch. Furthermore, the sum of the branch probabilities (letting z=1) emanating from each node equals unity.

One can show, using standard signal flow graph reduction techniques [8], that the generating function is given by

$$U(z) = \frac{(1-\beta)z}{1-\beta z H^{q-1}(z)}\left(\frac{1}{q}\sum_{i=0}^{q-1} H^{i}(z)\right) \tag{2.2}$$

where $\beta = 1 - P_D$ *and* $H(z) = P_{fa} z^{k+1} + (1 - P_{fa})z$.

The mean acquisition time $\overline{T_{ACQ}}$ is obtained by differentiating U(z) with respect to z and evaluating the result at z=1.

$$\overline{T_{ACQ}} = \left[\frac{\partial}{\partial z} U(z)\right]_{z=1} \tag{2.3}$$

considerable algebraic simplification is experienced when U(1)=1, and the RHS of equation (2.3) can be expressed as

$$\frac{\partial}{\partial z} U(z)\Big|_{z=1} = \frac{\partial}{\partial z} \ln U(z)\Big|_{z=1} \tag{2.4}$$

since

$$\frac{\partial}{\partial z} \ln U(z)\Big|_{z=1} = \frac{1}{U(z)} \frac{\partial}{\partial z} U(z)\Big|_{z=1} = \frac{1}{U(z)}\Big|_{z=1} \cdot \frac{\partial}{\partial z} U(z)\Big|_{z=1} \tag{2.5}$$

therefore, equation (2.3) can be written as

$$\overline{T_{ACQ}} = \frac{\partial}{\partial z} \ln U(z)\Big|_{z=1} \tag{2.6}$$

Since equation (2.2) satisfies the condition that U(1)=1, applying eqn (2.6), one arrives at the following result after some algebra,

$$\overline{T_{ACQ}} = \frac{2+(2-P_D)(q-1)(1+KP_{fa})}{2P_D}\tau_d \tag{2.7}$$

where K is the number of penalty time units due to false alarm, and for q>>1 (the number of cell to be searched is large, which is of practical interest), equation (2.7) could be simplified to

$$\overline{T_{ACQ}} = \frac{(2-P_D)(1+KP_{fa})}{2P_D}(q\tau_d) \tag{2.8}$$

The variance of the acquisition time determined from the first two derivatives of U(z) as

$$\sigma^2_{ACQ} = \left[\frac{\partial^2 U(z^{\tau_d})}{\partial z^2} + \frac{\partial U(z^{\tau_d})}{\partial z} - \left(\frac{\partial U(z^{\tau_d})}{\partial z}\right)^2\right]_{z=1} \tag{2.9}$$

but since U(1)=1, one can use the equivalent relation

$$\sigma^2_{ACQ} = \left[\frac{\partial^2 \ln U(z^{\tau_d})}{\partial z^2} + \frac{\partial \ln U(z^{\tau_d})}{\partial z}\right]_{z=1} \tag{2.10}$$

It can be shown that for q >>1 and K<<q then,

$$\sigma^2_{ACQ} = \tau_d^2(1+KP_{fa})^2 q^2\left(\frac{1}{12} + \frac{1}{P_D^2} - \frac{1}{P_D}\right) \tag{2.11}$$

It should be noted that the simple dwell system discussed is not restrained to a Non-Coherent bandpass detector, the above results apply equally to a single

dwell system with a coherent detector, the only difference being the interrelation of the parameters τ_d, P_{fa} and P_D for the detector. The above analytical results for the mean and variance of the acquisition time can also be obtained by a simple heuristic approach, a detailed description of which can be found in [9].

2.2.2 Serial PN acquisition system: Multi – Dwell

Fig. 2.3 is the state-transition diagram of a typical multiple-dwell detector, compare this with Fig. 2.2, we could see that it is much more complicated. We know that there is only one correct cell within the uncertainty region. Thus most of the cells evaluated by the energy detector are noise alone. An energy detection scheme that is capable of rejection these incorrect phase cells rapidly while not letting P_{fa} become so large that false-alarm penalty time dominates synchronization time is desirable.

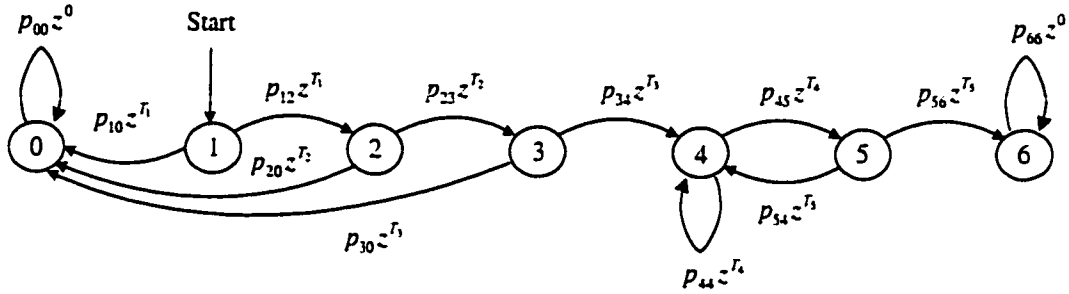


Fig. 2.3 State-transition diagram of a typical multiple-dwell detector

The multiple-dwell detection scheme, discussed in this section, accomplishes this using multiple evaluations of the same phase cell. The first evaluation is very short and results in immediate rejection of many incorrect cells. The short integration time of this first evaluation also results in a high false-alarm probability. When a false alarm occurs on the first evaluation, a second evaluation of the same cell begins, using a longer integration time. This second evaluation is capable of rejecting most of the false alarms of the first evaluation. The second evaluation may be followed by a third, fourth, or as many as desired to achieve a particular performance level. Since a particular phase cell may be rejected after one or more integration, the time required to evaluate a cell is a

random variable. All integration times and thresholds as well as the logic followed by the multiple-dwell detection scheme are chosen to yield minimum average synchronization time. Single-dwell is just a special case of the multiple-dwell systems.

Multi-dwell techniques are a generalization of the single swell serial PN acquisition with additional threshold testing that does not constrain the examination interval per cell to a constant interval of time. Nevertheless, this scheme falls into the class of fixed dwell time PN acquisition systems because of the fact that the variation in integration time is achieved here by allowing the examination interval to consist of a series of fixed short dwell periods (each generally longer than its predecessor) with a decision being made after each. Allowing the integration time in a given cell examination interval to increase towards its maximum value in *discrete steps*, permits dismissal of an incorrect alignment earlier than would be possible in a single dwell system which is constrained to always integrate over the full examination interval. Since most of the cells searched correspond to incorrect alignments, the ability to quickly eliminate them produces a considerable reduction in acquisition time (particularly in the case of long codes).

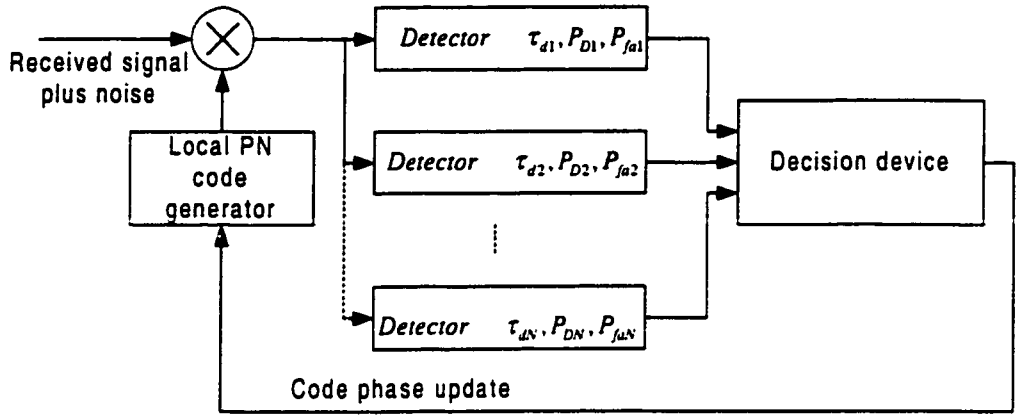


Fig. 2.4 The N-dwell serial synchronization system with half chip search

Fig. 2.4 illustrates the N-dwell serial synchronization system with half chip search. The received signal plus noise is firstly multiplied by the local replica of the PN code and this is applied to each of N non-coherent detectors. The *i*th detector, $i = 1,2,\cdots N$ is characterized by a detection probability P_{Di}, a false alarm probability P_{fai}, and a dwell time τ_{di}. If it is assumed that the detector dwell times are ordered such that $\tau_{d1} \leq \tau_{d2} \leq \cdots \leq \tau_{dN}$ then the decision to continue or stop the search at the present cell is made by sequentially examining the N-detector outputs (starting with the first) and applying the following algorithm:

- If all the N detectors (tested in succession) indicate that the present cell is correct, i.e., each produces a threshold crossing, then the decision is mode to stop the search.
- If any one detector fails to produce a threshold crossing (i.e., fails to indicate that the present cell is correct), then the decision is made to continue the search, the delay τ of the local PN generator is retarded by the chosen phase update increment, and the next cell is examined. Thus as soon as one detector indicates that the codes are misaligned, the search may move on without waiting for the decisions of the remaining detectors.

The power of the N-dwell system over the single dwell system lies in the fact that the maximum time to search a given cell is τ_{dN}, the minimum time is τ_{d1}, thus most of the cells can be dismissed after a dwell time τ_{dk}, $k \leq N$, whereas the single dwell system requires that each and every cell be examined for a time equivalent of τ_{dN}.

2.3 Matched Filter and Parallel Matched Filter

A highly efficient method of initial synchronization is to matched filter detect the received signal. The matched filter is designed to output a pulse when a particular sequence of code symbols is received. When this pulse is sensed, the receiver code generator is started using an initial condition corresponding to the received code phase and synchronization is complete. This technique requires matched filters with extremely large time-bandwidth products. In the usual case where the spreading code period is long, the matched filter must also be easily programmable. Foe example, the spreading code period may be longer than the entire mission being carried out. Thus a matched filter designed for a single sequence of code symbols may never receive these symbols. A programmable matched filter can be programmed for a particular code phase which is dependent on an estimate of the propagation delay. The requirement for programmability has led many investigations on high-speed digital correlators as well as programmable convolvers for this application.

In this method, each load is evaluated by attempting to despread the received signal just as was described for serial search techniques. This technique is called rapid acquisition by sequential estimation (RASE). A slightly modified version of this technique performs a preliminary evaluation to attempting the full correlation

despreading evaluation. This modified technique is called recursion aided RASE (RARASE).

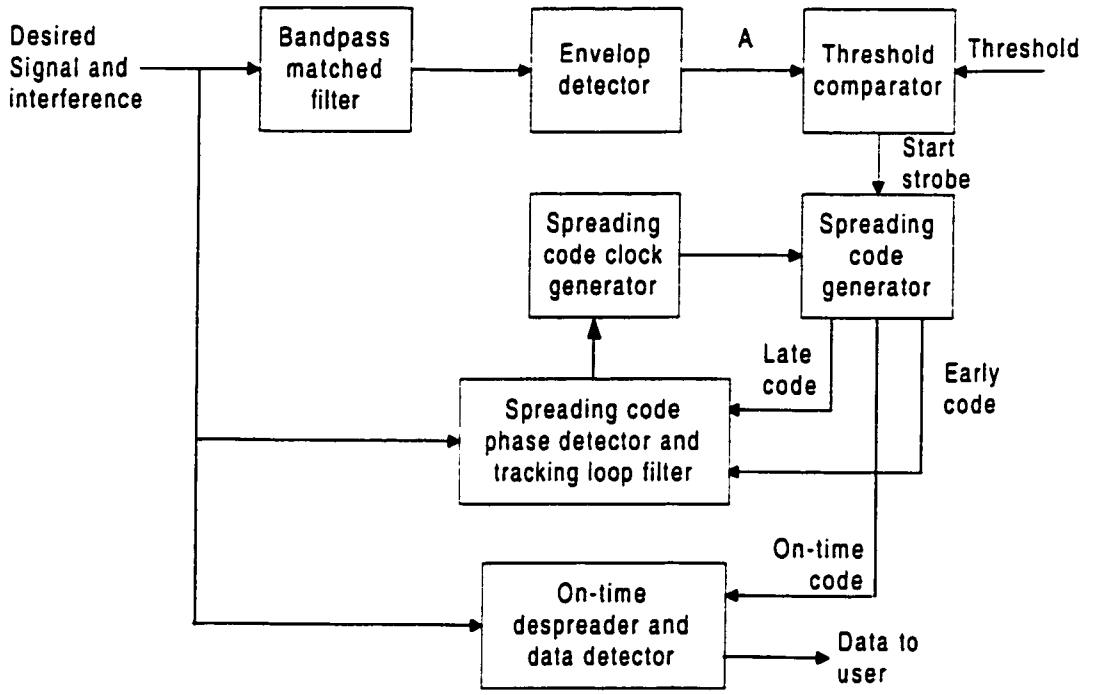


Fig. 2.5 Direct-sequence spread-spectrum receiver using matched filter code acquisition

Fig. 2.5is a top-level block diagram of a direct sequence spread-spectrum receiver which uses a matched filter synchronization processor. The received signal, including the desired signal and interference is input to the usual spreading code tracking loop and to the on-time despreader for data detection. In addition, the

received signal is input to a bandpass filter, which is matched to a segment of the direct-sequence-spread waveform. When the waveform segment to which the filter is matched is received, the matched filter produces an output pulse. The synchronizer detects this pulse using an envelope detector followed by a threshold comparator. The spreading code generator is started at the correct phase when the matched filter output pulse is sensed. The starting phase for the code generator is a function of the selected matched filter waveform segment and the matched filter and detection delays. The code generator starting phase is calculated prior to the start of the synchronization process and in some applications may be a constant set in the receiver design.

The average acquisition time for matched-filter synchronizers is calculated using the generalized analysis which are discussed in detail in reference [8], here we employ some of the results. The average acquisition time is given by

$$\overline{T_s} = \left[\frac{d}{dz} H(z)\right]_{z=1} \tag{2.12}$$

where H(z) is defined as a function to represent the sum over all path of the branch label products of the state-transition diagram which are mentioned in reference [8]. Consider the synchronizer of Fig. 2.5 and assume that the envelope of the matched filter output is compared to a threshold after each sampling interval T_s. If the envelope exceeds the threshold, the receiver code

generator is started and the code tracking loop is closed. If the threshold crossing was due to a correct detection of the synchronization waveform segment, synchronization is complete. If, however, the threshold crossing was due to a false alarm, the synchronization process is delayed by a false alarm penalty time T_{fa}, after which the synchronization process continues. If the envelope does not exceed the threshold, the synchronizer waits T_s seconds and samples the envelope detector output again.

Compare this process to the serial search synchronizer, the matched filter synchronizer evaluates one code phase each T_s seconds while the serial search synchronizer evaluates one code phase each T_l seconds. During the time T_s between envelope detector output samples, the received spreading waveform advances 1/N chips, where N is the sampling rate, it is greater than the code chip rate. Thus the serial search code-phase step size (usually ½ chip) corresponds to the step size of 1/N chops in the matched filter synchronizer. Thus the circular state diagram for the matched filter synchronizer has MN+2 states, where M is the code-phase initial uncertainty in chips. So, H(z) for the matched filter is:

$$H(z) = P_D z^{T_s} \frac{\sum_{i=1}^{MN} (1/MN)\left[P_{fa} z^{(T_l+T_{fa})} + (1-P_{fa}) z^{T_s}\right]^{MN-i}}{1-(1-P_D) z^{T_s}\left[P_{fa} z^{(T_l+T_{fa})} + (1-P_{fa}) z^{T_s}\right]^{MN-1}} \qquad (2.13)$$

Taking the derivative of the above equation with respect to z and setting z=1 yields the average synchronization time. After some straightforward manipulations, the result is

$$\overline{T_s} = \frac{T_s}{P_D} + \frac{2-P_D}{2P_D}(MN-1)(T_s + T_{fa}P_{fa}) \tag{2.14}$$

For $MN >> 1, P_D \approx 1, P_{fa} \approx 0$, this reduce to

$$\overline{T_s} \approx \frac{T_c}{N} + \frac{MT_c}{2} \tag{2.15}$$

where $T_c=NT_s$ has also been used.

Modern matched filter synchronization systems are most often implemented digitally. The digital implementation of the synchronizer uses a lowpass implementation of the bandpass matched filter. Fig. 2.6 and Fig. 2.7 illustrates both the bandpass and lowpass matched filters and the associated envelope detectors in analog form. These two implementations are equivalent and produce identical outputs A(t) when the inputs r(t) are identical.

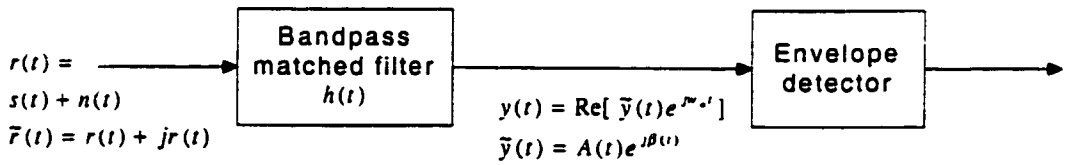


Fig. 2.6 Bandpass matched filter implementation

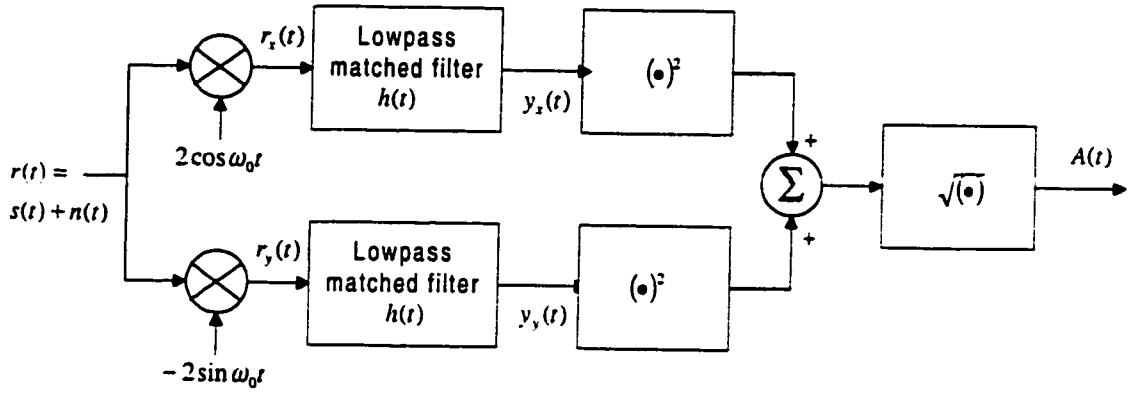


Fig. 2.7 Lowpass matched filter implementation

Consider an arbitrary narrow-band (with respect to carrier frequency) input signal r(t), which may include both the signal of interest and interference. The complex envelope of this signal is denoted $\tilde{r}(t) \equiv r_x(t) + jr_y(t)$. The reference carrier for this complex envelope is ω_0, so that

$$r(t) = r_x(t)\cos\omega_0 t - r_y(t)\sin\omega_0 t \tag{2.16}$$

The band-pass matched filter is defined by either its band-pass impulse response or its complex envelope. Observe that the complex envelope is a real signal, implying that the band-pass filter is symmetric with respect to its center frequency ω_0. The complex envelope of the band-pass filter output is

$$\begin{aligned}\tilde{y}(t) &= \int_{-\infty}^{t} \tilde{r}(t)\tilde{h}(t-\lambda)d\lambda \\ &= y_x(t) + jy_y(t) \\ &= A(t)e^{j\beta(t)}\end{aligned} \tag{2.17}$$

where

$$A(t) = \sqrt{y_x(t)^2 + y_y(t)^2} \tag{2.18}$$

By definition, the envelope detector output is A(t). Since the complex envelope $\tilde{h}(t)$ is real, (2.17) may also be written

$$\begin{aligned}\tilde{y}(t) &= \int_{-\infty}^{t} [r_x(\lambda) + jr_y(\lambda)]\tilde{h}(t-\lambda)d\lambda \\ &= \int_{-\infty}^{t} r_x(\lambda)\tilde{h}(t-\lambda)d\lambda + j\int_{-\infty}^{t} r_y(\lambda)\tilde{h}(t-\lambda)d\lambda\end{aligned} \tag{2.19}$$

Now consider the low-pass circuit of Fig. 2.7. Ignoring the sum frequency terms. The outputs of the upper and lower mixers of Fig. 2.7 are $r_x(t)$ *and* $r_y(t)$ respectively. The impulse response of either of the low-pass matched filters is $\tilde{h}(t)$. Thus the outputs of the low-pass filters are

$$\int_{-\infty}^{t} r_x(\lambda)\tilde{h}(t-\lambda)d\lambda = y_x(t) \tag{2.20}$$

$$\int_{-\infty}^{t} r_y(\lambda)\tilde{h}(t-\lambda)d\lambda = y_y(t) \tag{2.21}$$

so that the output A(t) of Fig. 2.7 is exactly the same as the output A(t) of Fig. 2.6 which was calculated in (2.18).

With the equivalence of the band-pass and low-pass implementations of the matched filter established, consider only the upper channel of the low-pass implementation. The output of this filter is the real part of (2.19).

$$\tilde{y}_x(t) = \int_{-\infty}^{t} r_x(\lambda)\tilde{h}(t-\lambda)d\lambda \tag{2.22}$$

For the impulse response and considering the output only at time $t = nT_s$,

$$\begin{aligned}\tilde{y}_x(nT_s) &= \int_{nT_s - KNT_s}^{nT_s} r_x(\lambda)c\left(KNT_s + \lambda - nT_s\right)d\lambda \\ &= \sum_{m=0}^{KN-1} \int_{(n-KN+m)T_s}^{(n-KN+m+1)T_s} r_x(\lambda)c\left(KNT_s + \lambda - nT_s\right)d\lambda \\ &= \sum_{m=0}^{KN-1} c(m)\int_{(n-KN+m)T_s}^{(n-KN+m+1)T_s} r_x(\lambda)d\lambda\end{aligned} \tag{2.23}$$

This equation provides the basis for the digital implementation of the matched filter envelope detector illustrated in Fig. 2.8. In this figure, a sampling rate of twice the spreading code chip rate has been assumed.

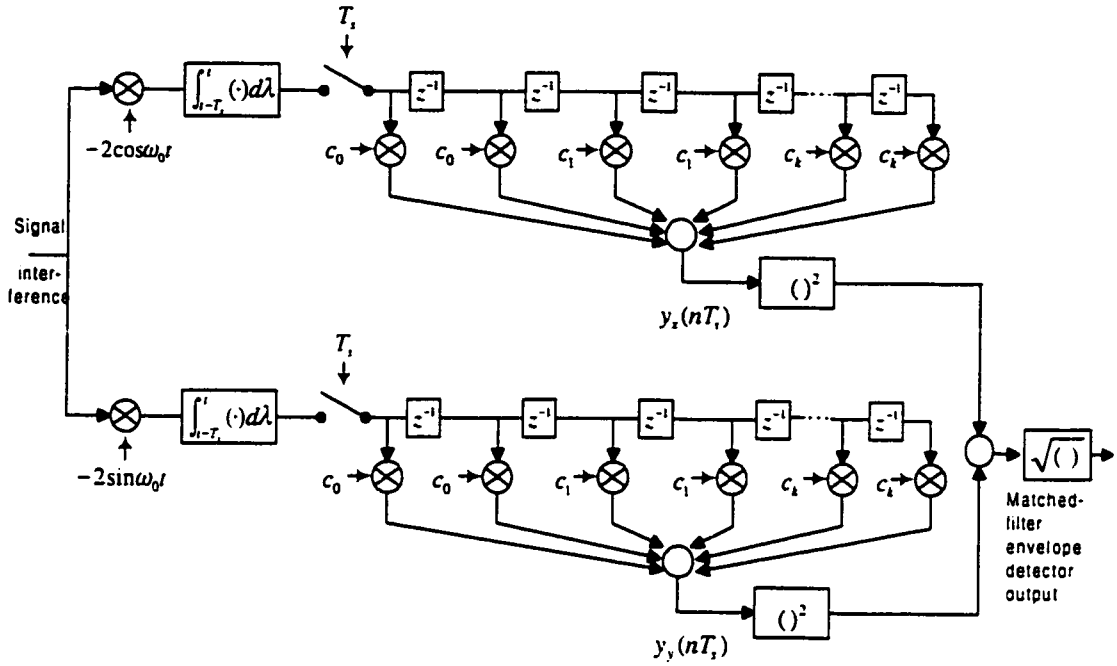


Fig 2.8 Digital implementation of non-coherent matched filter

The degradation due to carrier frequency error is the principal issue limiting the use of matched filter synchronizers. Matched filter coherent integration times also limited by data transitions. Techniques are available that mitigate but do not completely solve these problems. A brute force solution to carrier frequency error problem is, of course, to use a bank of matched filter synchronizers. Another means of mitigating the problem is to partition the matched filter coherent integration time and to non-coherently combine the results of the shorter coherent integration. This problem is acknowledged by assuming that the matched filter integration time will be shorter than that which would be possible with a non-

coherent stepped-serial-search system. This shorter integration time results in detection and false-alarm probabilities which are not acceptable at the system level. To solve this problem a multiple-dwell system is proposed in which detection by the matched filter detector triggers a coincidence detector to validate the matched filter hit. Ignoring the carrier frequency error issue, matched filter synchronizers are capable of improving the synchronization time of direct-sequence spread-spectrum system by orders of magnitude and therefore it is extremely important.

Based on the matched filter, here is the parallel matched filter. This is nothing but dividing the matched filter into several parts, each concentrating on one part of the phase, and these parallel parts are working separately and independently, and also each one outputs the epoch time of their own. Thus the acquisition time is shortened. It depends on how many parallel matched filter the system has, the more is this, the shorter is the acquisition time.

2.4 Modified Sweep Strategies for Non-Uniform Distribution Signal

All the results up to this point have been derived assuming that the received spreading waveform phase is uniformly distributed over a particular uncertainty region. Suppose now that the received phase distribution is defined by p(n), the probability that the received phase is within the *n*th phase cell. When p(n) is known, the sweep strategy should be modified to search the most likely phase cells first and then the less likely cells. For example, if the received phase distribution were Gaussian, a reasonable search strategy would be to search the cells within one standard deviation of the most likely cell first and the expand the search to cells with two standard deviations, and so on. Note that p(n) is a discrete distribution and therefore cannot be Gaussian. The distribution of the received phase is however, continuous, the function p(n) is easily derived from the continuous density and the cell boundaries. Fig. 2.9 illustrates the Gaussian distribution and the search strategy proposed by [10]. Observe that the received phase density has been truncated at three standard deviations.

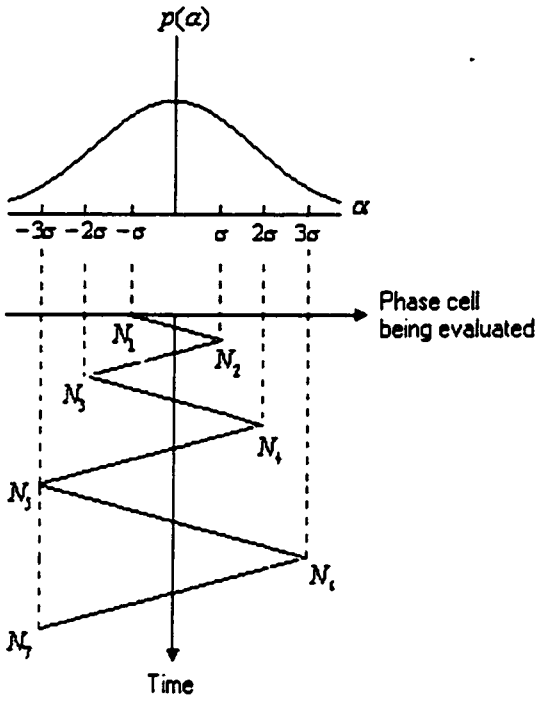


Fig. 2.9 Received spreading waveform phase probability density and possible sweep strategy

The average synchronization time for the system is calculated using

$$\overline{T_s} = \sum_{n,j,k} \Pr(n,j,k) \tag{2.24}$$

where n,k,j are the particular location for the correct phase, the particular number of false alarms and the number of missed detentions of the correct phase respectively. The proper limits for all summations for the three variables n, j and

k must be determined. If ΔT represents the time uncertainty region, Δt represents the phase step size, then $\frac{\Delta T}{\Delta t}$ is an integer and represents the number of cells to be searched. The limits for the first summation are then $-\frac{C}{2} \le n \le +\frac{C}{2}$ where cell number zero is at the center of the uncertainty region. The discrete probability of the n^{th} phase cell being correct is

$$p(n) = A\exp(-\frac{n^2}{2T^2}) \qquad -\frac{C}{2} \le n \le +\frac{C}{2} \tag{2.25}$$

where A is a constant calculated from

$$\sum_{n=-C/2}^{+C/2} p(n) = 1, \quad T = C/6 \tag{2.26}$$

Let each pass through the uncertainty region be numbered using the variable "b". Furthermore let N_b, and $N_{(b+1)}$ denote the starting and ending cell number for the *b*th pass. For the strategy of Fig. 2.9, N_1=-C/6, N_2=C/6, N_3=-C/3, N_4=C/3, and so on, where all fractions are rounded to the nearest integers. Let f(n) denote the number of the pass that evaluated call "n" for the first time. For example if C/6<n<C/3, the cell will not evaluated until the third sweep. The limits of the sum over "j" are $0 \le j \le \infty$, the variable "k" is the total number of false alarms in all incorrect phase cells is denoted by K(n, j) and is given by

$$K(n, j) = \sum_{b=1}^{f(n)+j-1} \left|N_{b+1} - N_b\right| + \left|n - N_{f(n)+j}\right| - j \tag{2.27}$$

The false alarm can occur in any order within the K(n, j) incorrect phase cells and the number of false alarms is binomially distributed. Combining all of the above information yields

$$P(n,j,k) = A\exp\left(-\frac{n^2}{2T^2}\right)P_D(1-P_D)^j\binom{K(n,j)}{k}P_{fa}^k(1-P_{fa})^{K(n,j)-k} \qquad (2.28)$$

and

$$\overline{T_s} = \sum_{n=-C/2}^{n=C/2}\sum_{j=0}^{\infty}\sum_{k=0}^{K(n,j)}[K(n,j)+j]T_i + kT_{fa}A\exp\left(-\frac{n^2}{2T^2}\right)P_D(1-P_D)^j\binom{K(n,j)}{k}P_{fa}(1-P_{fa})^{K(n,j)-k} \qquad (2.29)$$

The above equation has been evaluated in [10]. Modified sweep strategies have been used for many years in radar. Their use results in reduced average synchronization time when the distribution of the received phase is non-uniform.

Chapter 3

Concatenated Signature Coding

And The New VMF Description

This chapter begins with an introduction to the construction of the concatenated signature code. Based on this code, a new code acquisition technique, namely, Virtual Matched Filter is developed. Here a classic noncoherent code acquisition method is presented, this shows the advantages of the new scheme from the other side. Particular emphasis is laid on Epoch Detection Model, since it is the critical part to get a better understanding of the new system. Two strategies are employed in the analysis and the pipeline concept is introduced which plays an active role in minimizing the code acquisition time.

3.1 Introduction

Under the urge of applications in communication area such as CDMA Macrocellular system, Global Positioning system, Low Orbit Satellite system, or deep space system, minimization the code acquisition time under heavy load and severe Doppler becomes a great challenge. In our VMF, at User Transmitter, we select 2 short gold code which have good cross-correlation properties, and they are encoded with BCH FEC which yields fine results in error control and correction, then concatenated by LFSR, this is the generation of our signature code. At User Receiver, two short code MFs are followed by paralleled EDMs, through the pipeline operation of the best sequence selector, dynamic erasure BCH decoder and back to back shift register in each of the EDMs, the whole new technique resembles a MF of much shorter physical length than total code length, namely, Virtual MF for the new scheme. In the process, minimization of the acquisition time is achieved.

3.2 Structure of the concatenated signature code

3.2.1 Generation of two short gold code for each user

Each user transmitter (UT) is assigned 2 short Gold codes and two identifying Reed Solomon (RS) or BCH code words, for communications and signal

identifications at user receiver (UR). The short code Gold length = $(2^{n_1} - 1)$ chips = $(2^{n_1} - 1)$ Tc sec, where Tc is the code chip duration. At the UT, the two UT assigned, RS (or BCH) unique data words are fed bit by bit (round robin fusion) into the 2 RS (or BCH) encoders at a clock rate = 1 / $((2^{n_1} - 1)$ Tc). The concatenated signature code generation is shown in Fig.3.1. and Fig.3.2 represents the construction of the code.

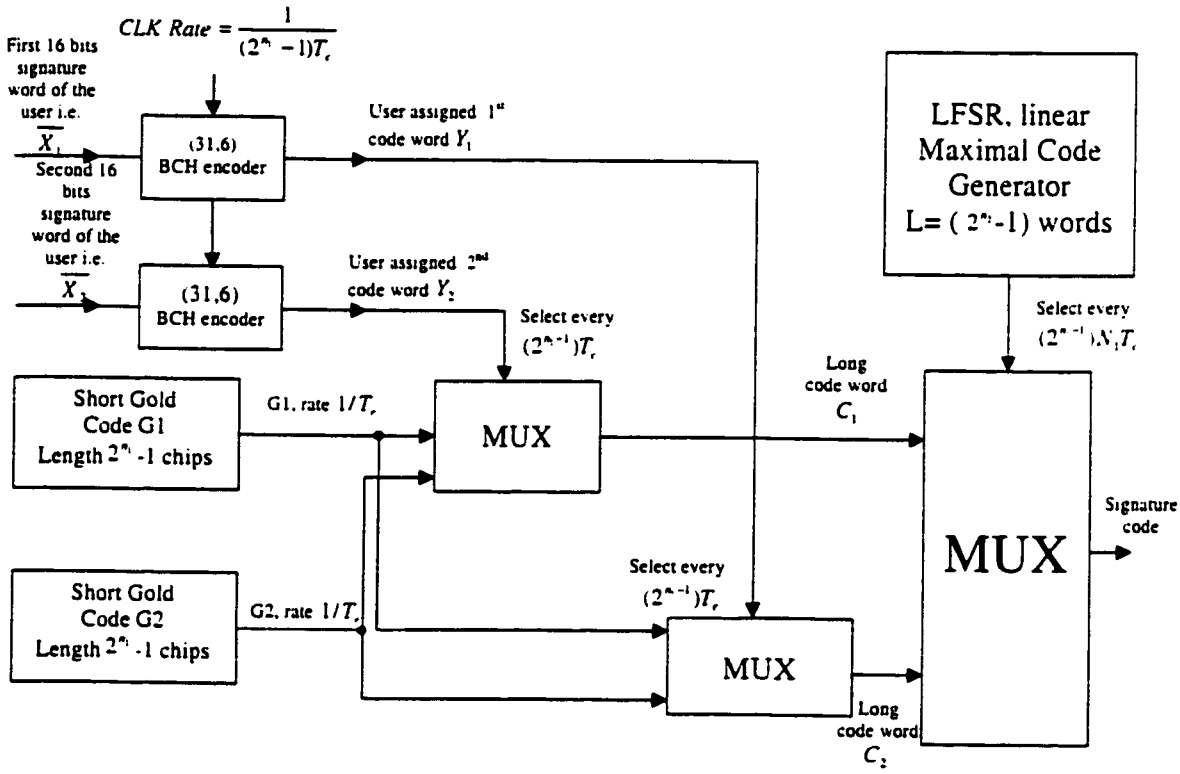


Fig. 3.1 Concatenated signature code generation for Virtual Matched Filter acquisition

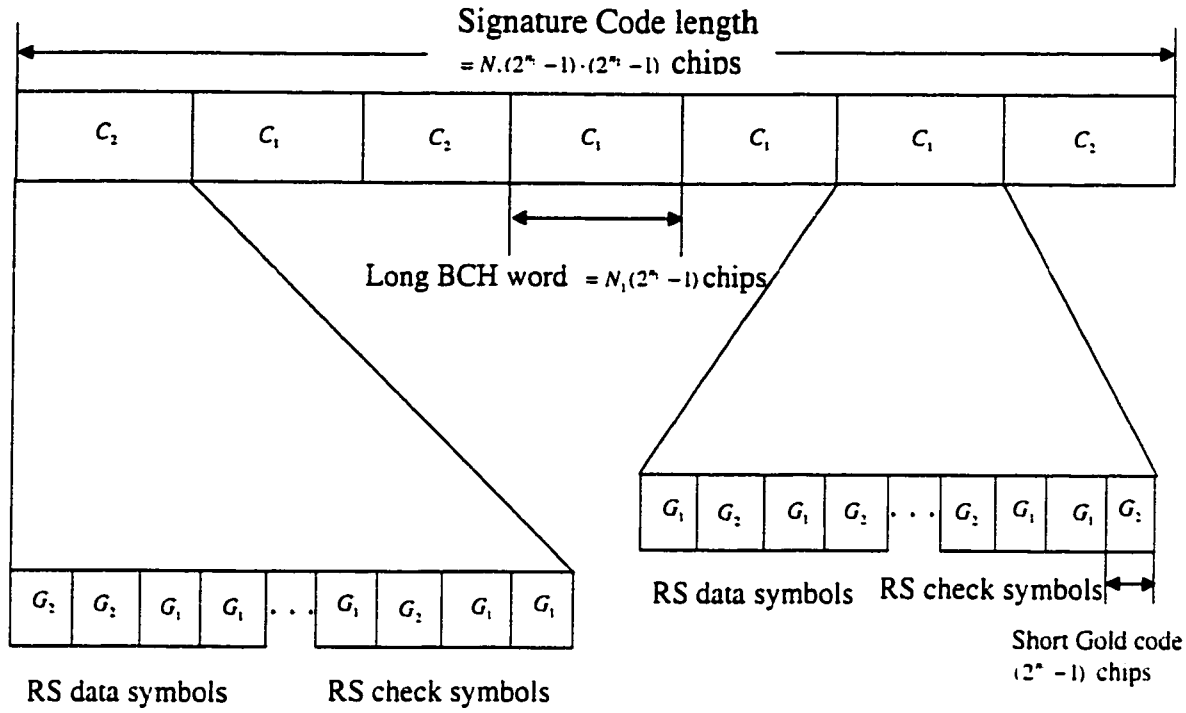


Fig. 3.2 Concatenated signature codes using short Gold Cold and BCH FEC encoding

3.2.2 BCH FEC encoding

The special case of (N_1, K_1) Binary BCH FEC encoding is taken in this paper without loosing generality (other RS or other block codes could also be used). The 2 outputs of the UT assigned 2 short Gold codes generators are both fed to two multiplexers (switches) whose selects (controls) are nothing but the output bits of two BCH encoders above, i.e., $\overline{Y_1}$,$\overline{Y_2}$. These BCH code words bits are

clocked at a rate = $1/((2^{n_1}-1)Tc)$ while the generated bits of the two short codes G1, G2 are fed at a rate of (1/Tc) to the multiplexers, which means that each of the two resulting long code words C1, C2 (Fig.3.2) consist of $(2^{n_1}-1)\cdot N_1$ code chips. Selecting C1=111111 .. i.e. all one's and C2 =0000.. i.e. all zeros BCH words seems to yield a good distance in between for protection against the effects of CDMA interference , Doppler ..etc. For simplicity, ease of erasure correction and implementations, we take the (31,6) BCH code which can correct up to t =7 bits per BCH word, this low (6/31) rate encoder many seem as a waste of capacity for data demodulation purposes. However during the code acquisition periods no data is transmitted and the FEC code rate is sacrificed for better code detection and false alarm probabilities...etc. as will be seen shortly. Worth mentioning at this point that this FEC has nothing to do with data FEC (Viterbi, RS, concatenated, Turbo, ... etc.) the UT typically uses on top to correct for data symbols errors on the transmission channel (following the acquisition process).

3.2.3 Back to back LFSR

Finally the short Linear Maximal (LFSR) code bits selects every $(2^{n_1}-1)\cdot N_1$ Tc sec., one of the long code words C1 or C2 as the signature code to be mixed with the UT data bits..., etc. This long signature code is a random succession of C1 and C2 long code words, and has a total length of CL = $(2^{n_1}-1)\cdot N_1 \cdot (2^{n_2}-1)$ code

chips, where n_2 is the Linear Maximal shift register length. For example with $n_1 = 9, n_2 = 5, N_1 = 31$, one obtains CL =long signature code length =511×31×31 =491071 code chips, with $n_1 = 9, n_2 = 7, N_1 = 63$, CL=4120704, with $n_1 = 9, n_2 = 6, N_1 = 127$, CL=4153408, with $n_1 = 9, n_2 = 5, N_1 = 255$, CL=4169760.

3.3 New code acquisition technique------ VMF description

3.3.1 Security Properties

The pseudorandomness of the presented long signature code is guaranteed by the randomization of the locations of the G1,G2 short Gold codes (under the control of the BCH code words), and further randomization of the resulting long words C1,C2 (under the control of the LFSR code bits).In this work we don't further investigate the long term crosscorrelation and power spectral properties of our long signature code. However most importantly, it has the very much desirable property of well predicted short range crosscorrelation ,i.e. of the familiar Gold code (over one UT date bit of duration $T_b = (2^{n_1} - 1).T_c$) as opposed to the TIA short pilot transmission code of length (2^{15}-1) code bits.

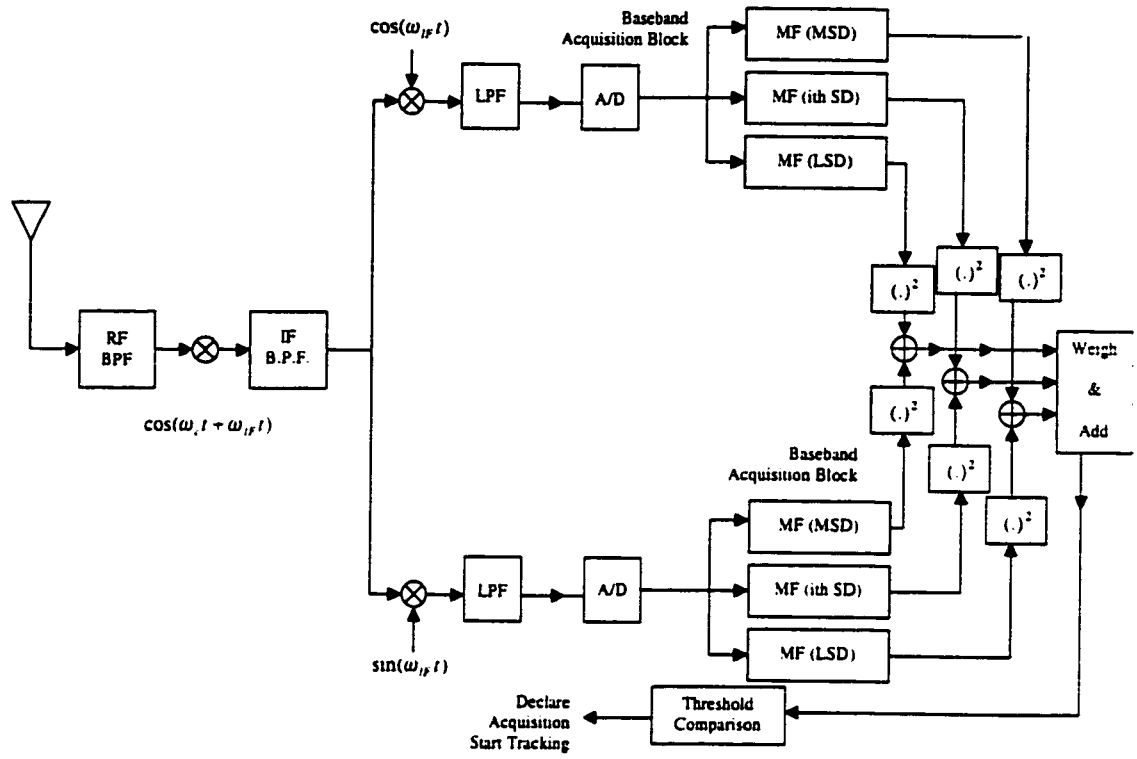


Fig. 3.3 A general practical non-coherent MF acquisition

(Notice that parallel matched filters here don't imply parallel acquisition. They are merely introduced for better soft decisions. Typically n = number of A/D bit = 3 or 4 means 8 or 16 quantization levels. Most acquisition analysis merely treat one of the MF above means one bit quantization only.)

3.3.2 Classic noncoherent code acquisition method

Prior to introducing our new code acquisition technique and for completion purposes, we introduce Fig.3.3 in which shows a classic noncoherent code

acquisition technique using two quadrature branches, A/D converters, banks of binary matched filters, and squaring devices.

It is easy to see that this practical arrangement cancels the carrier phase but not the Doppler or carrier offsets, also A/D is known here to achieve better resolution, better code detection and false alarm probabilities on the expense of more involved implementation.

3.3.3 Layout of the VMF

Taking only one of the I&Q of channels of Fig.3.3., concentrating only on the baseband (encircled) part ,assuming a one bit A/D and incorporating all the new signature coding acquisition details, we obtain Fig.3.4. The above reductions are meant to simplify the descriptions and the subsequent analysis without loosing generality.

In Fig.3.4., the baseband signal received at the UR is sampled at a rate 1/(Tc /q), typically q=2, This is followed by 2 short digital MFs (G1, G2), Each (Ts =Tc /q) sec the outputs of the 2 MFs are compared, the identity of the largest and the associated magnitudes are fed to the next stage (gate function). Thus a typical input to the gate function consists of 2 numbers like (1,30) or (0,3), where 1 denotes G1, 0 denotes G2 and 30 are the associated voltage outputs of the two

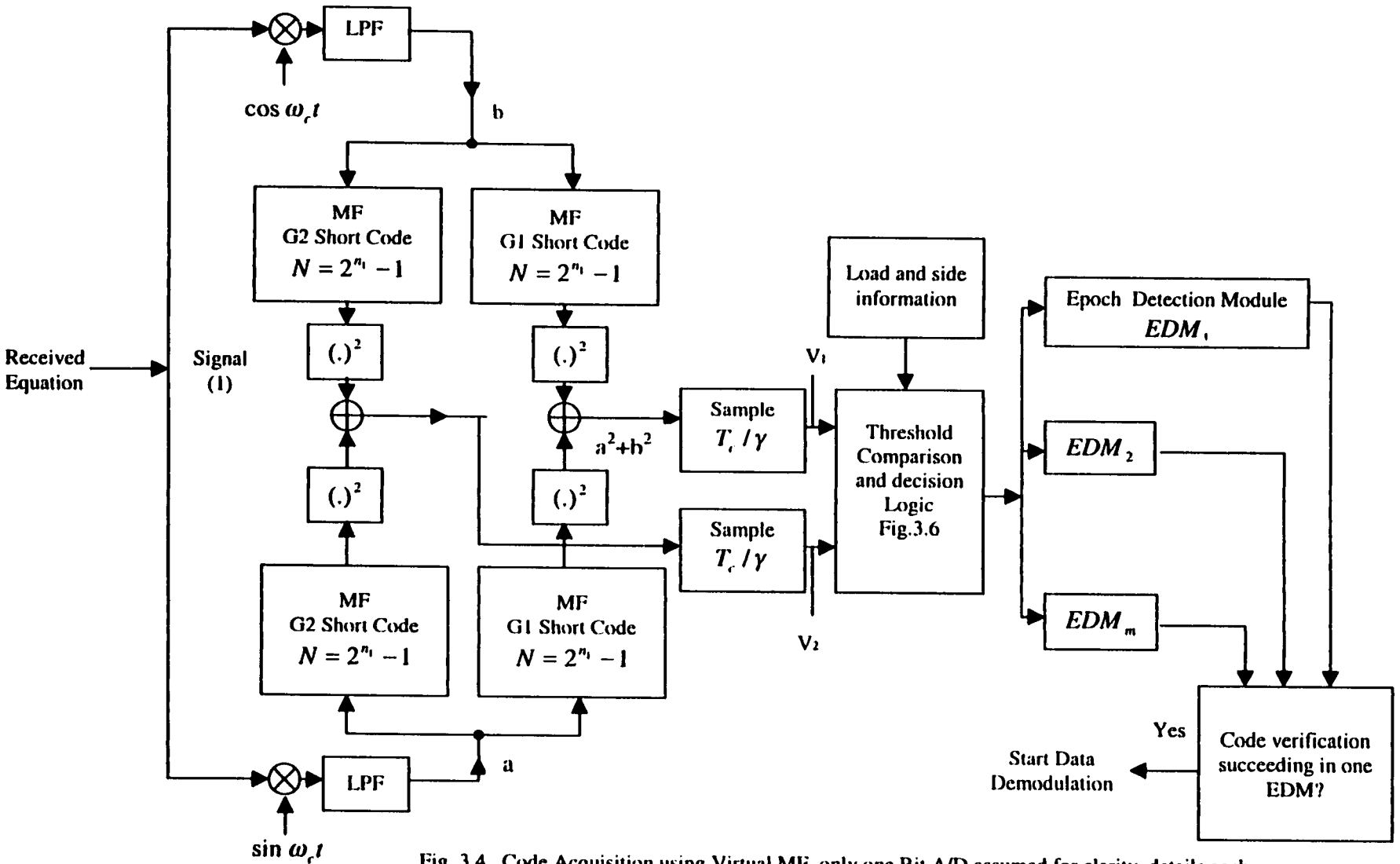
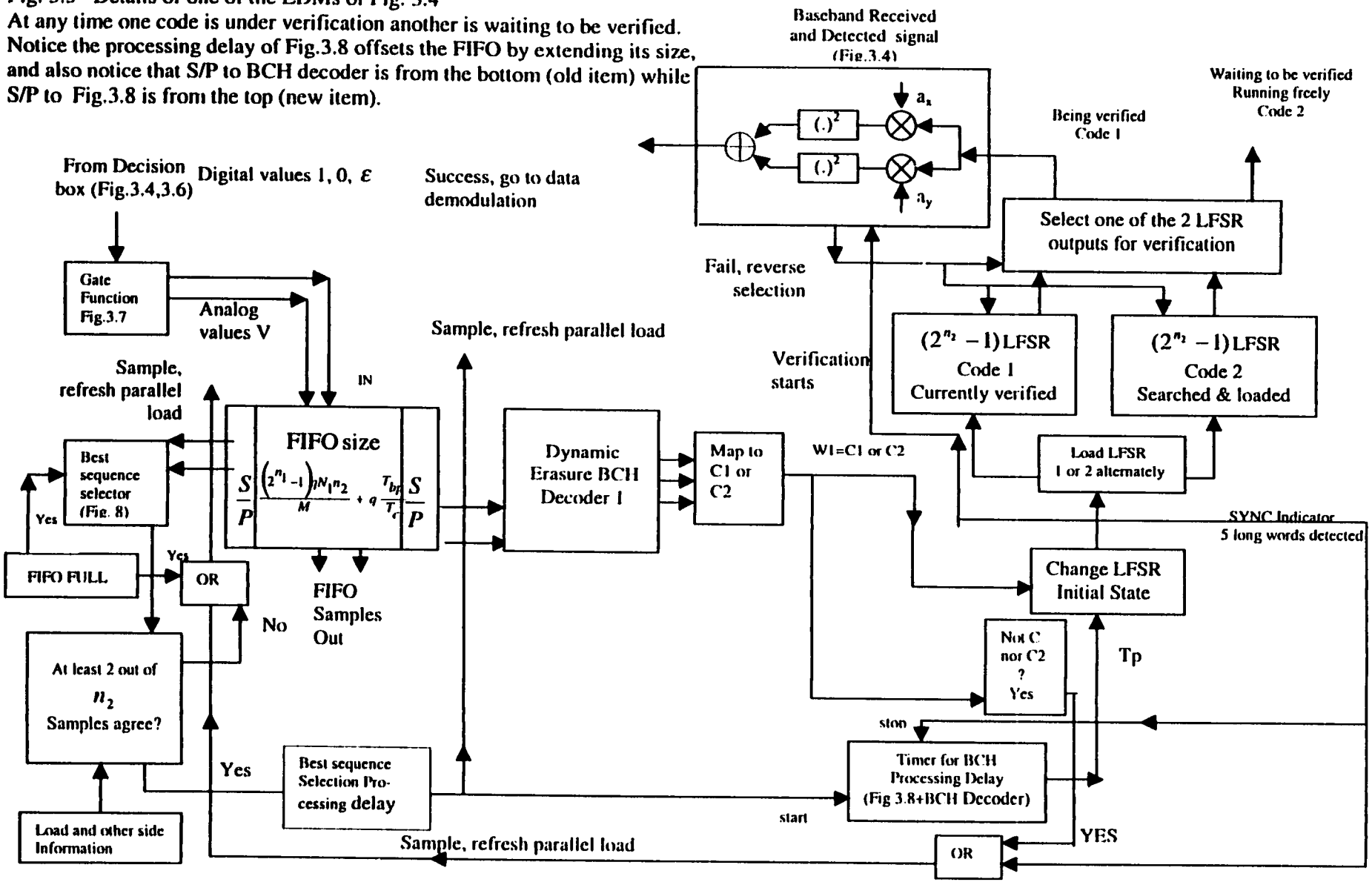
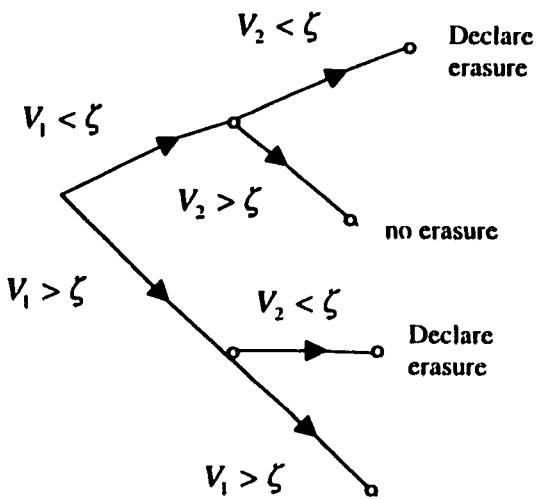


Fig. 3.4 Code Acquisition using Virtual MF, only one Bit A/D assumed for clarity, details each EDM is shown in Fig.3.5

Fig. 3.5 Details of one of the EDMs of Fig. 3.4
At any time one code is under verification another is waiting to be verified. Notice the processing delay of Fig.3.8 offsets the FIFO by extending its size, and also notice that S/P to BCH decoder is from the bottom (old item) while S/P to Fig.3.8 is from the top (new item).



E_s
Erasure Selection

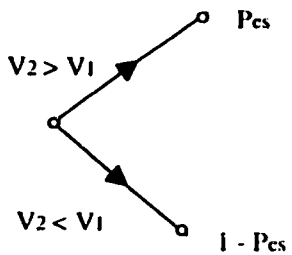


Assume G1 is the received short code.

$$E_s = P(V_1 < \zeta_1) \cdot P(V_2 < \zeta_1) + P(V_1 > \zeta) \cdot P(V_2 > \zeta)$$

Erasure declared when both G1, G2 MF outputs are too high or too low (both) Note the independence of ε, P_{es}.

P_{es}
Decision error (no erasure)
Light interference

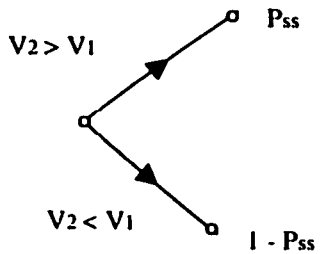


Assume G1 is the received short code.

P_{es} = P (V2 > V1 / no erasure)

$$= P(V_2 > \zeta) \cdot P(V_1 < \zeta)$$

P_{ss}
Decision error
Under erasure
Heavy interference



Assume G1 is the received short code.

P_{ss} = P (V2 > V1 / erasure)

=P (V2 > V1)

Fig. 3.6 Details of Decision Box of Fig.3.4., V1, V2, are 2 G1, G2 MF's ζ is the threshold. Short codes are declared as erasures if both V1, V2 are above or below threshold, other short codes are corrected. Pre correction decision error is Pe or Ps in low and high interference respectively.

MFs (Fig. 3.5). However if the magnitudes of the 2 MFs are both lower than a certain threshold or both higher than a second threshold, an erasure will occasionally replace 1 or 0 values according to the flow charts of Fig.3.6. More of these details will be described in the subsequent figures and analysis.

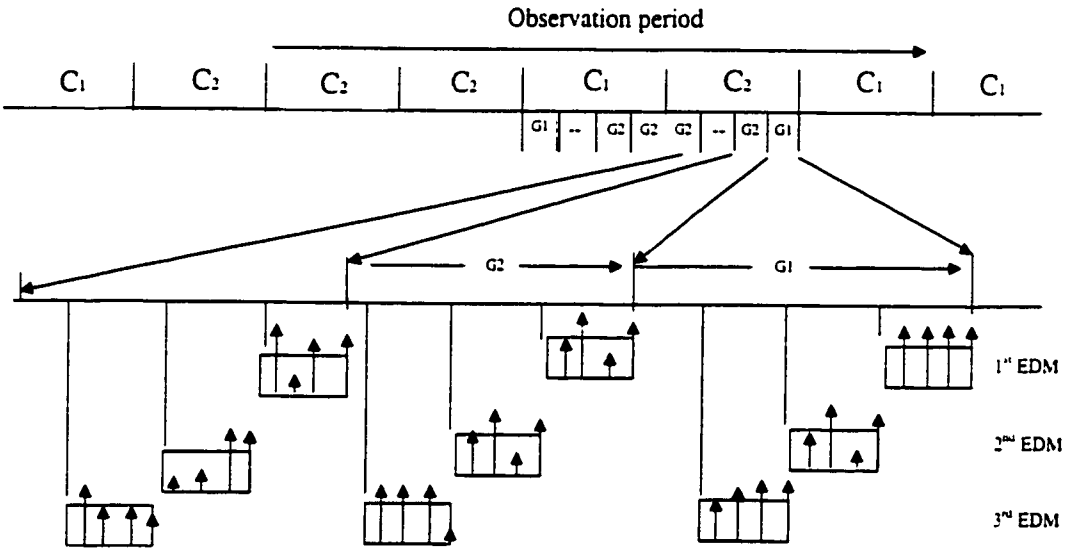


Fig. 3.7 Detection time domains of each epoch detection software module (3 out of 4 shown in figure) Arrows indicate magnitudes and occurrence times of each G1 or G2 matched filter outputs. In the figure, each module can have 0,1,2,3, ... ,

$\frac{(2^{n_1}-1)}{M} = \frac{16}{4} = 4$ pulses per window per short code (G1 or G2) spaced by T_c/q.

Each module selects the best pulse sequence out of 4 possible.

3.3.4 Role of EDM

However we back now to Fig.3.4, where we see a bank of M code epoch detection modules (EDMs) (which can be conveniently implemented in software), the details of one of these is shown in Fig.3.5. Each of these modules tries to find a legitimate sequence of G1, G2 short codes by searching over its time window as dictated by the gate function described in Fig.3.7 This Gate function is implemented by the Decision logic box of Fig.3.6. For example (EDM)1 processes the following output samples of the decision logic box of Fig.3.4, see also Fig.3.7) where $T_s = \frac{T_c}{q}$ and V is the Maximum of V_1 . V_2

$$| V(0 \cdot T_s) |, | V(1 \cdot T_s) |, |V(2 \cdot T_s)|, \ldots\ldots, | V((q \cdot 2^{n_1} / M) - 1) \cdot T_s |$$

$$\left|V((0 + q \cdot (2^{n_1}))T_s)\right|, \left|V((T_s + q \cdot (2^{n_1}))T_s)\right|, \ldots\ldots, \left|V(((q \cdot 2^{n_1} / M) - 1)T_s + q \cdot (2^{n_1}))T_s)\right|$$

...

$$\left|V((0 + q \cdot (2^{n_1}) \cdot N_3)T_s)\right|, \left|V((T_s + q \cdot (2^{n_1}) \cdot N_3)T_s)\right|, \ldots\ldots, \left|V(((q \cdot 2^{n_1} / M) - 1)T_s + q \cdot (2^{n_1}) \cdot N_3)T_s)\right|$$

where $N_3 = N_1 - 1$

The above samples resemble a matrix whose columns will be compared by the following block (Best Sequence Selector of Fig.3.5, detailed in Fig.3.8). This block essentially sums all absolute voltages of each column, compares them and

selects the column that gives the maximum. This yields also the epoch time of the maximum vector out of the possible times for $(EDM)_1$:

i.e. 0 *or* T_s *or* $2T_s$ *or* .....*or*.....$(((q \cdot 2^{n_1}/M)-1)T_s)$

The subsequent BCH Decoder read the identities (1 or 0 or ε) detected by earlier Blocks(1 denotes G1output greater than G2, 0 denotes the contrary, and ε denotes an erasure condition as described in Fig.3.6).

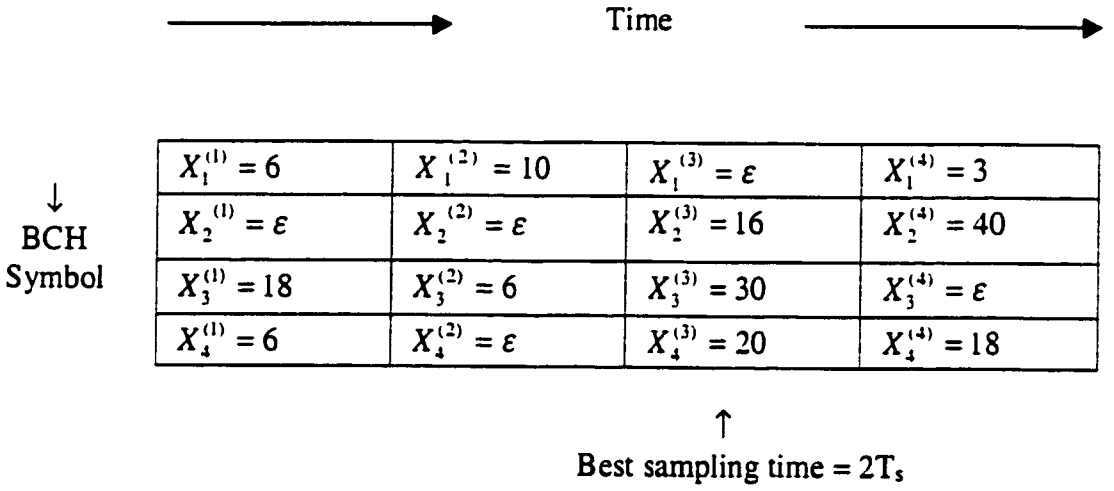


Fig. 3.8 Determination of the best sampling time.

(That is one out of $\left(\frac{2^{n_1}\cdot q}{M}-1\right)T_s$ for each EDM, $X_j^{(i)}$ denotes values of V matrix. (I=sampling time, i=0, ... , $\frac{\left(2^{n_1}\cdot q-1\right)}{M}T_s$ and j = BCH symbol identity within the BCH code word, j = 1,2,...,(N_1-1)) Here we assume $N_3=N_1-1=4$ for simplicity. In the figure, $i=2T_s$, corresponding to the third column, this yields best sampling time: $\sum_j X_j^{(i)} = 16+30+20 = 66$ is the maximum compared to all other column.)

The BCH decoder is also fed by the best $|V|$ (out of a possible $(2^{n_1}\cdot q/M)$ vectors) of the above maximum likelihood sequence time determination process (Fig.3.7, Fig.3.8).

Before describing the manner in which these are used by the BCH decoder, we notice that the length of the short Gold codes has been increased by one for clarity only, and that the times in the $|V|$ matrix above corresponds only to $(EDM)_1$. The computations are similar for other EDMs., and clarity as well as analysis convenience are the motivations here. In a typical example the matrix of $1,0,\varepsilon$ values fed to the RS decoder is shown below:

	$q \cdot \frac{(2^{n_1})}{M}$ Samples					
$N_1 = 31$ BCH Symbols	ε	ε	1	0	1	---
	1	1	ε	1	1	---
	1	0	1	0	ε	---
	0	ε	0	1	ε	---
	---	---	---	---	---	---
	---	---	---	---	---	---

Assuming that the third vector of the corresponding $|V|$ array gave the maximum, the best sequence selector Box (Fig.3.7, Fig.3.8) then samples the corresponding third vector of the above BCH symbol matrix *i.e.* $[1\ \varepsilon\ 1\ 0 \ldots]^T$ and feed it to the following BCH decoder trying to decode the received BCD symbols to one of the assigned BCH code words, C1 or C , however the received signal will occasionally decode to neither C1 nor C2, as will follow shortly. The above processes are repeated n_2 times ($n_2 = 5$ for example) i.e. the BCH decoder works hard to decode n2 =5 consecutive BCH words (of 31 BCH symbols each) and

correct possible errors and/or erasures in an adaptive manner to be described shortly).

Needless to say, the n_2 consecutive BCH words should all coincide in epoch time! i.e. all should correspond for example to the third vector of each of the $|V|$ samples arrays (shown before), that come from the Gate and Best sequence boxes (Fig.3.7, Fig.3.8), otherwise the acquisition trial is aborted. However, we opt to relax this strict epoch conditioning, especially in face of the severe Doppler and CDMA overload conditions anticipated in many networks scenarios, and select the vector location (and hence time epoch) where at least two best vectors of two consecutive $|V|$ arrays agree .If non of the n_2 = 5 detected arrays agree on the location of the best $|V|$ vector that gives the maximum over other vectors in the same array, the EDM skips this acquisition trial and declare a signature code miss. In the latter case, the "No two arrays agree" box (Fig.3.5) forces the sampling and feeding of a fresh. $2^{n_1} \cdot q \cdot N_1 \cdot n_2 / M$ samples of the received signal to the FIFO. Each sample consists of voltage values (V_1 or V_2), and (1 or 0 or ε).The newest of these ,i.e. $2^{n_1} \cdot q..W / M$ sample correspond to T_{ps} ,i.e. the processing time of (Fig.3.7, Fig.3.8). Notice the effect of this processing delay within the array, it makes the digital FIFO add extra stages, thus the best sequence box (Fig.3.8) samples the top (fresh) part of FIFO while the output to BCH decoder

corresponds to the old (aging) stages of the FIFO (bottom of FIFO) because of this processing delay of Fig. 3.8, and so pipeline operation is achieved.

The smaller the T_{ps} , the less will be W and the new fresh samples to feed the FIFO, till we reach the ideal case where, $T_{ps} = T_s$, in which case the FIFO is clocked at a fraction of the chip rate, and ideal Matched filtering comes out.

The acquisition trials are repeated till at least 2 out of $n_2 = 5$ $|V|$ arrays will agree on the location (and hence epoch time) of the best Vector of $|V|$, in which case a success signal emerging from the "best Sequence selector" box (Fig.3.5) will initiate the BCH decoder to work , triggers their feeding with samples from the FIFO and also sets a " Decoding and processing duration" timer (say DC). This timer is reset once $n_2 = 5$ long BCH code words each belonging to the set {C1, C2} are decoded, thus leading to the loading of the subsequent n_2 stages $LFSR_2$ (Fig.3.5) with the appropriate initial conditions(1 corresponding to C1 detected , and 0 corresponding to C2 detected), at that time the contents of the DC will be an indication of the BCH decoding processing times of n_2 long BCH words (C1 or C2) . Accordingly, this DC value will be used to shift the initial state of $LFSR_2$ appropriately so as to account for the mentioned decoding and processing delays.

The success signal above also initiates the best sequence box (Fig.3.7, Fig.3.8) to process fresh $|V|$ samples, while the BCH decoder is trying to find n_2 correctly decoded long BCH words as above Parallelism is the motivation here, so as to save processing, and hence precious acquisition times. The BCH decoder receives N_1 BCH symbols ($1 \, or \, 0 \, or \, \varepsilon$) from the FIFO under the control of the "Best Sequence" box (Fig.3.5, Fig.3.7 and Fig.3.8), tries to correct the errors and erasures and yields either BCH code word C1 or C2, or any of the other 29 words in the BCH code dictionary (called Z).

3.3.5 Two strategies in BCH decoding technique

Ideally, under correct signature code epoch presence the BCH decoder yields one of the BCH code words (C1 or C2). Recall (Fig.3.1, Fig.3.2) the signature code consists of a pseudorandomly selected sequence of C1,C2 long BCH words. However due to the high CDMA load and/or fading , Doppler, etc...,and assuming C1 is the current correct BCH word , the BCH decoder output may become C1 or C2 or Z.

Fig. 3.9a shows decisions made in these and other cases, as well as the associated events probabilities, in this approach, we call it strategy 1.

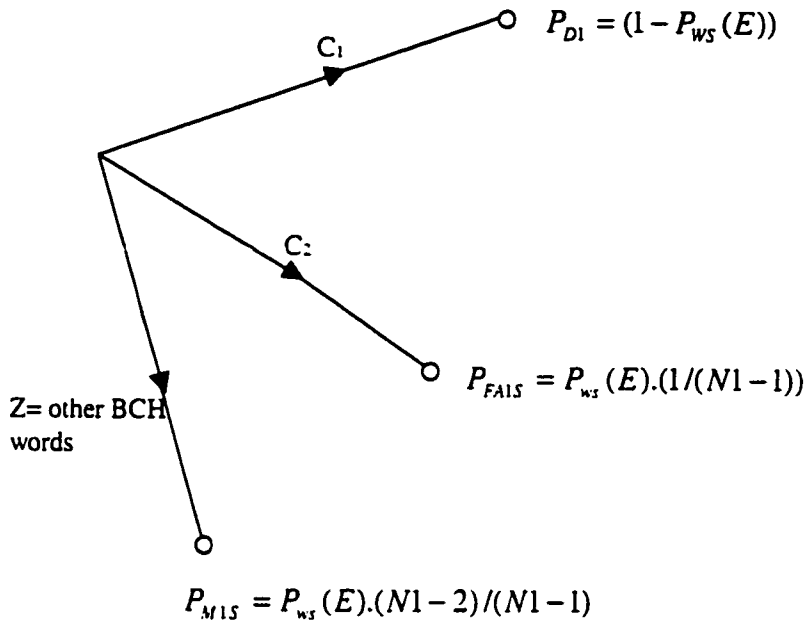


Fig. 3.9a Possible outcomes of the BCH decoding process (stategy 1) under correct signature code epoch presence. C1 is assumed to be the transmitted BCH word (without loosing generality). Please note that we have False Alarm probobility even under correct code presence in this figure. Uniform decoding error distribution over the (N_1-1) word decoding possibilities is assumed given that a BCH word error has occurred.

Of particular importance in this figure (for description purposes) is the last outcome (the BCH decoder yields Z = neither C1 nor C2), in this case the acquisition trial is aborted by the EDM in question, a miss is declared, the DC

timer is reset and the BCH decoder is started again by fresh $(1,0,\varepsilon)$ samples coming from the FIFO. Note, these are loaded in parallel and were prepared in advance (while the BCH decoder was decoding and so there is no time is wasted here (Pipelining concept). In regard to the last leaf of Fig.3.9a, strategy 1 prefers to miss the signature code epoch rather than breaking the tie, randomly guessing C1 or C2, and ending with a costly false alarm in case C1 selected while the correct received word was C2.

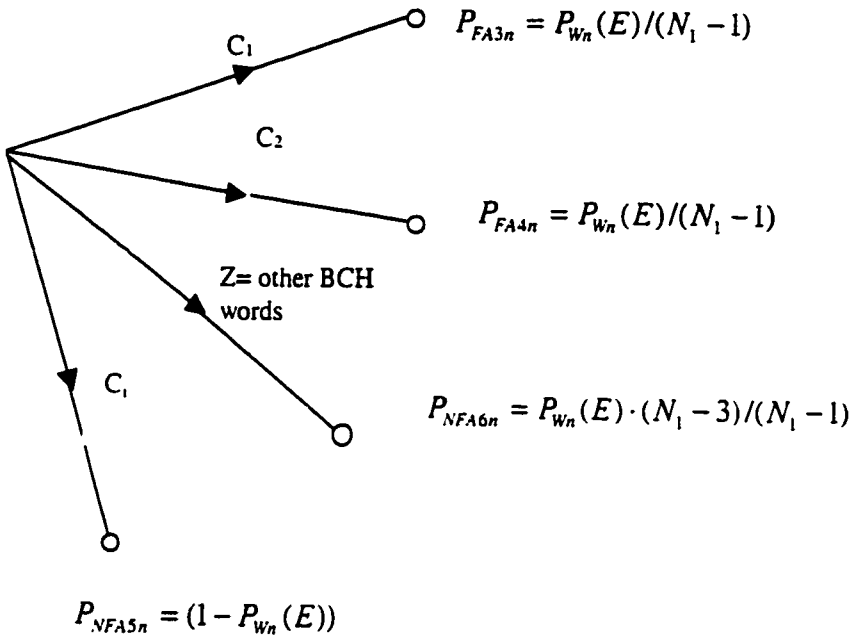


Fig. 3.9b Possible out comes of the BCH decoding process (strategy 1) under correct signature code epoch absence.

C_i is assumed to be the transmitted BCH word (i=3,4,..,31). Actually C_1 or C_2 is transmitted, but due to lack of correct timing (epoch absence), the BCH decoder thinks that C_3 or C_4 or ... was transmitted. Of course under code epoch absence, BCH decoder should never say C_1 or C_2, otherwise we get P_{FA} as above. Uniform decoding word error distribution over the (N_1-1) possibilities is assumed given that a BCH word error has occurred.

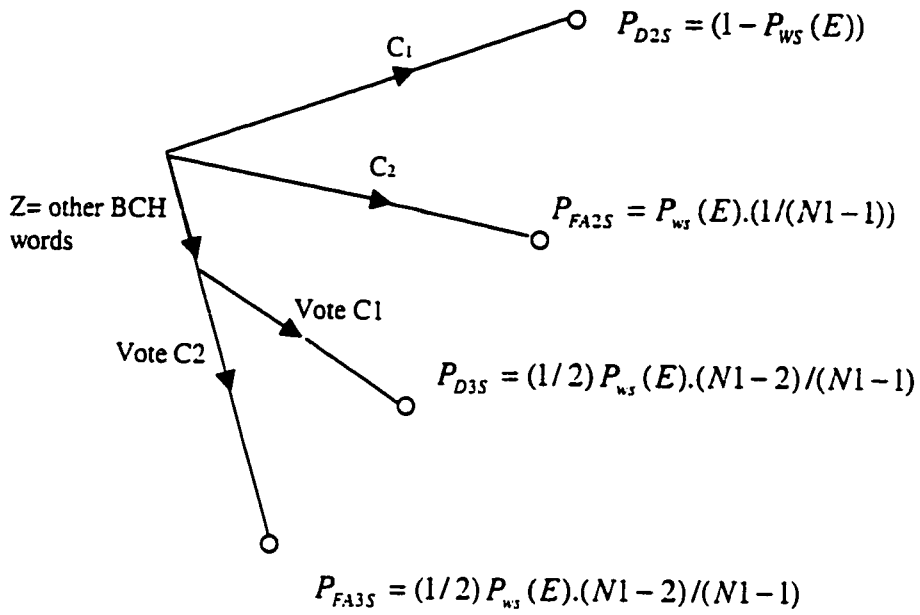


Fig. 3.10 Possible outcomes of the BCH decoding process (strategy 2) under correct signature code epoch presence.

C1 is assumed to be the correct BCH word. On the absence of code epoch, P_{FA2}=1 is always true according to strategy 2.

In a different approach, we call it strategy 2, as shown in Fig.3.9b, it is possible to randomly select C1 or C2 as the outcome instead of aborting the acquisition trial

(in the case of the last outcome of the tree of Fig.3.9a). Logically to expect that Strategy2 leads to more code acquisition False alarm P_{FA}. This strategy 2 is shown in Fig.3.10. In contrast strategy1 leads to more P_M but less P_{FA}. Both strategies will be explained in more detail in the analysis section.

3.3.6 Pipeline concept in code acquisition

Back to Fig.3.5, we see that the "select" box following the BCH decoder accumulates n_2 =5 sequence of C1 or C2 long BCH code words, maps C1 to 1, maps C2 to 0 and load one of the back to back registers (LFSR1) of length n_2=5 with these bits and throw it into the code verification mode (tracking).

Simultaneously, a SYNC indicator is sent back to reset the DC timer, and refresh the FIFO with new data (shift in). While code verification takes place in each EDM according to LFSR1(see also Fig3.1, Fig3.2), new samples of the arriving signal are processed according to Fig.3.6, Fig.3.7 and Fig.3.8.

If a new code epoch is then found, LFSR2 will be loaded with the new Initial state (n_2=5 bits corresponding to the newly decoded long BCH words), after which the new signature code corresponding to LFSR2 will run freely, waiting for its turn for verification. This means that the two signature code generators running

according to LFSR1 and LFSR2 will be busy most of the time. One verifies an old code epoch while the other one is being loaded or in its way to be loaded or actually loaded and running with a new discovered code initial state, ready in case the old code epoch fails the verification trial (all in the same EDM).

Fig.3.11 show that in each (EDM), while one LFSR is involved in code tracking, another one is looking for a new code epoch. Also in each EDM, new data may not enter the BCH decoder while the n_2 stages LFSR is loaded with the BCH decoding outcomes.

Also, all EDMs are operating independently and with larger M, new code epoch will be more frequently found and as one EDM returns from false tracking, another one or more EDMs will have another code epoch to be verified or finally one would have verified the right code epoch, i.e. code epochs can come at any time from any of the EDMs.

This is the reason behind the right most box of Fig.3.4 which initiates the information data demodulation process once one of the M EDMs verifies the right code. We notice that the CD content of a certain EDM at the end of the SYNC trial (once 5 C1 or C2 long BCH words has been detected) and loaded into one LFSR is used to alter these initial conditions corresonding to the BCH decoding

delay encountered (which might slightly vary depending on the applicable BCH word received).

Also the same SYNC signal resets the CD contents, and the n_2 stages LFSR is then shifted in a normal fashion under its (T_c) duration clock. Parallel in time a previously discovered code epoch is being verified (for a time period T_{fa}) and once code verification fails, the rules of the 2 back to back LFSRs (Fig.3.5) are exchanged and the freely running LFSR goes to verification, while the other LFSR waits (Fig.3.5, Fig.3.11) for the BCH decoder to find a new code epoch. Based on all the above, it is easily seen that the whole new technique in this work resembles a Matched Filter of physical length: $(2^{n_1}-1)\cdot N_1 \cdot n_2 \cdot qT_s$, trying to find the code epoch of the much longer total code of length $(2^{n_1}-1)\cdot N_1 \cdot (2^{n_2}-1)\cdot qT_s$, also the actual physical MFs ,i.e. G1, G2 lengths are only$(2^{n_1}-1)$, thus the name Virtual MF for the new scheme .

The parallelism of the EDMs above and the pipeline operation imply that the search time of one search cell of the code uncertainty region is approximately T_i =0, however, as in the analysis section, a worst case value will be inserted instead of 0.

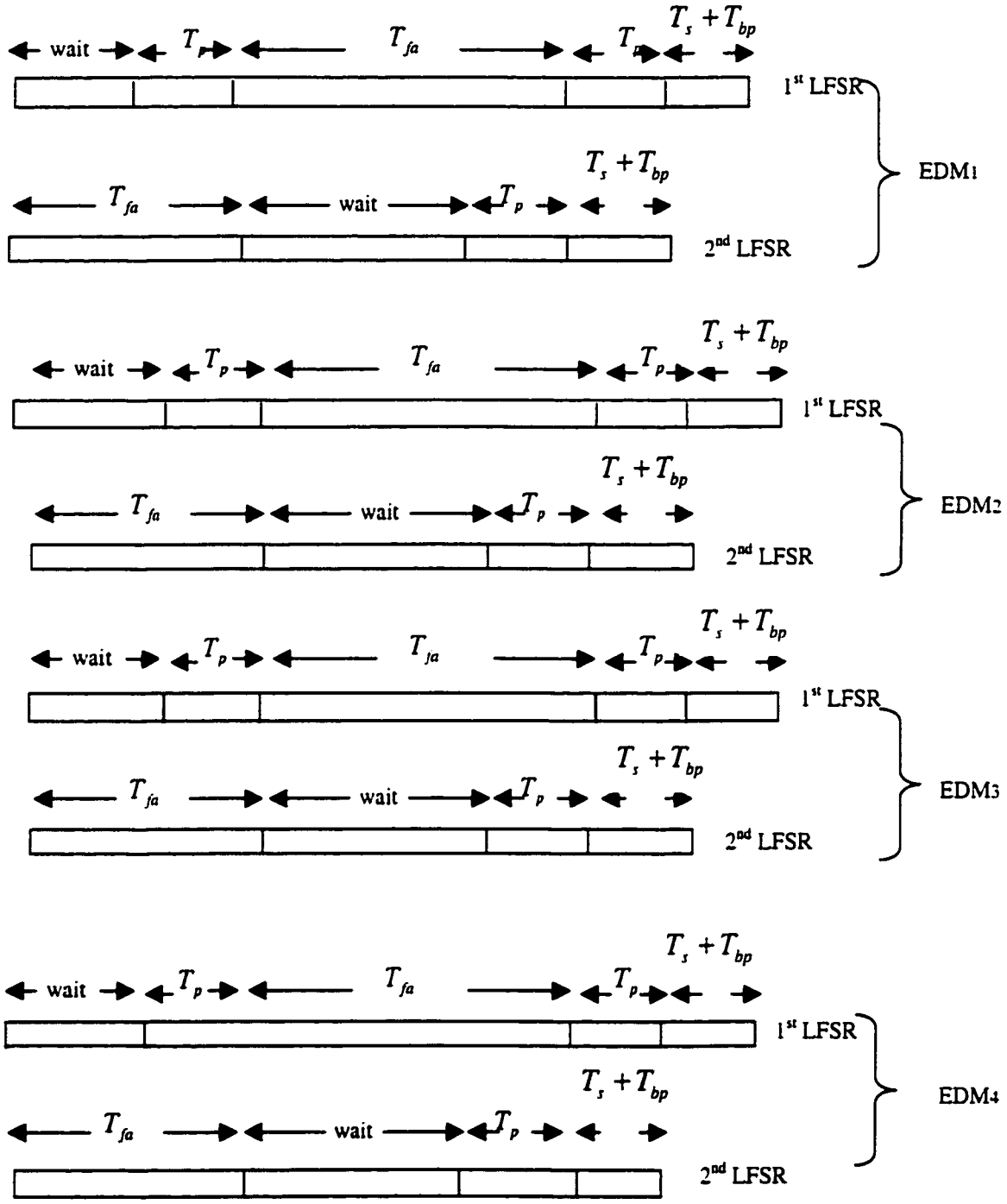


Fig. 3.11 Time sequence for EDM

(Fig. 3.11 Each EDM software samples V(t) only at the portion of $\frac{(2^{n_1}-1)}{M}.T_c.q$ of each short code length, so uncertainty region = C/2 and probability of alignment = 1/(C/2)=(2/C)=α_l ,$T_s=\frac{(2^{n_1}-1)N_1.n_2}{q}T_c$, Tp = BCH processing time, T_{fa} =false alarm penalty, T_{bp}= best sequence selection processing time. Note that a new found code epoch on one LFSR of one of the EDMS has to wait till the previous code epoch of the other LFSR of the same EDM is verified in T_{fa} sec as shown above, please also notice that T_s+T_{bp} exists only at turn off, with new trials we have only T_p.).

Chapter 4
New hybrid acquisition scheme

Based on the new VMF system design described in the previous chapter, this chapter develops the performance and the acquisition time of our system theoretically. It begins with the detection of the short Gold Code, followed by evaluation of the performance of the VMF and BCH decoding error, both under signal condition and pure noise condition respectively. Finally the verification time of the long concatenated signature word is obtained.

4.1 Introduction

In the previous chapter, an overview of our VMF system at the UR is given in Fig.3.4(p.62), to obtain the desired acquisition time, we have to start with the detection of the short code matched filter, in this part, we take the advantage of the Gold Code three valued cross correlation property. Second the error probability in detection one short Gold Code is obtained, in this case, the analysis resort to the Gaussian assumption. Third, we go further to the performance of the BCH decoder, and here we propose to use the idea of erasure plus error correction to ensure our VMF system performs well both under light and heavy interference environment. Finally, acquisition time of our VMF is achieved, under the discussion of strategy one and strategy two as defined in chapter two. It is worth to mention that all our analysis for the VMF is in the worse case without loosing generality, in the real case, it would perform much better as being described also in this chapter.

4.2 VMF Acquisition Scheme

4.2.1 Detection of the short Gold Code

We start with the Virtual Matched Filter acquisition technique (VMF), where the analysis emphasizes the performance of one of the EDMs. Being all similar, operating in parallel, and due to Matched Filtering, the uncertainty region reduces

to $R_1 = (2^{n_1} - 1) \cdot q / M$ cells from the much larger counter part of classic serial search, i.e. $R_4 = (2^{n_1} - 1) \cdot N_1 \cdot (2^{n_2} - 1) \cdot q$ cells. The average acquisition time of the whole signature code (of length $(2^{n_1} - 1) \cdot N_1 \cdot (2^{n_2} - 1).T_c$ sec) is the same as that yielded by any one of the EDM's (due to parallelism). The evaluation of this goes in steps as follows:

Assuming for analysis convenience a fading free DS/PSK user signal, one bit A/D (Fig.3.3 on p.60), The total multiaccess (CDMA) received signal (Fig.3.4 on p.62) can be described as,

$$r(t) = \sqrt{2\alpha P} g_1(t) \cos((\omega_c + \Delta\omega_c)t + \phi_1) + \sum_{i=2}^{U} \sqrt{2P}\, d_i(t) g_i(t) \cos((\omega_c)t + \Delta i t + \phi_i) + n(t) \qquad (4.1)$$

where $g_1(t)$ *and* $g_i(t)$ are the (Tc sec duration) code chips of the 1st (intended) and the ith (interfering) users respectively, ω_c is the carrier frequency assumed the same for all users. The intended 1st user has a combined Doppler and frequency error of $\Delta\omega_c$ angles, U-1 is the number of interfering CDMA users, Δi is the combined carrier and Doppler offset of user i , ϕ_1, ϕ_i are the random phases of the intended and interfering users, and α is the intended user to interferenc power ratio. P is the *i*th user signal power assumed the same for all users, $n(t)$ *is AWGN of PSD* = $No/2$. The first user signal has data bits of all 1s (during acquisition), while the other U-1 users are assumed in data transmission mode.

Following the $\cos(\omega_c t)$ *and* $\sin(\omega_c t)$ mixing, LP.F., one obtains the two following signals at the I &Q arms (Fig.3.4 on p.62),

$$a_x(t) \equiv \alpha\sqrt{2P}\frac{g_1(t)}{2}\cos(\Delta\omega_c t + \phi_1) + \sum_{i=2}^{U} \sqrt{2P}\frac{d_i(t)g_i(t)}{2}\cos(\phi_i + \Delta i t) + n_1(t)$$

$$a_y(t) \equiv \alpha\sqrt{2P}\frac{g_1(t)}{2}\sin(\Delta\omega_c t + \phi_1) + \sum_{i=2}^{U} \sqrt{2P}\frac{d_i(t)g_i(t)}{2}\sin(\phi_i + \Delta i t) + n_2(t) \quad (4.2)$$

where $n_1(t) = n(t)\cos(\omega_c t)$ *and* $n_2(t) = n(t)\sin(\omega_c t)$. And Δi is the Doppler offset of user 'i' from intended user, $\Delta_i = \Delta f_c + \psi$ where ψ is a constant frequency offset (e.g. 2Hz).

Feeding both a(t), b(t) to the respective G1 and G2 matched filters, assuming that G1 is the current short code transmitted by the 1st intended user, assuming also that the arriving signal (eqn.1) code phase, (time of arrival) matches that of the G1 MF., one obtains by the known equivalence of active correlation and MF.,

$$a = \int_0^{NTc} \alpha\frac{\sqrt{2P}}{2}\cos(\Delta\omega_c t + \phi_1)dt + \sum_{i=2}^{U} \sqrt{2P}\int_0^{NTc} \frac{d_i(t)g_i(t)g_1(t)}{2}\cos(\phi_i + \Delta_i) + n_1(t)g_1(t)\,dt$$

$$b = \int_0^{NTc} \alpha\frac{\sqrt{2P}}{2}\sin(\Delta\omega_c t + \phi_1)dt + \sum_{i=2}^{U} \sqrt{2P}\int_0^{NTc} \frac{d_i(t)g_i(t)g_1(t)}{2}\sin(\phi_i + \Delta_i) + n_2(t)g_1(t)\,dt \quad (4.3)$$

where $N = (2^{n_1} - 1)$ short Gold code length of G1, G2.

Following the integration, the squaring and summing operations (Fig.3.4 on p.62) one gets, (neglecting the presence of data bits and AWGN,

$$e_{1z} = a^2 + b^2 = \frac{\alpha^2 P}{2} N^2 T_c^2 [\frac{\sin(2\pi N \Delta f_c T_c)}{2\pi N \Delta f T_c}]^2 + \frac{(U-1)P}{2} N^2 T_c^2 R^2_{xy}$$
$$\bullet \left[\left(\frac{\sin(2\pi N \Delta_t T_c)}{2\pi N \Delta_t T_c} \right)^2 + \left(\frac{\sin^2(\pi N \Delta_t T_c)}{\pi N \Delta_t T_c} \right) \right] + \frac{\alpha^2 P N^2}{2} T_c^2 [\frac{\sin^2(\pi N \Delta f_c T_c)}{\pi N \Delta f_c T_c}]^2$$

$$= A\{\{\alpha^2 B^2 + (B'^2 + D'^2)(U-1)R_{xy}^2 + \alpha^2 D^2\} \} \quad (4.4)$$

$$where\ A = \frac{P}{2} N^2 T_c^2 \quad , B = \frac{\sin 2\pi N \Delta f_c T_c}{2\pi N \Delta f_c T_c} , D = \frac{\sin^2 \pi N \Delta f_c T_c}{\pi N \Delta f_c T_c} , B' = \frac{\sin 2\pi N \Delta_t T_c}{2\pi N \Delta_t T_c} ,$$
$$D' = \frac{\sin^2 \pi N \Delta_t T_c}{\pi N \Delta_t T_c}$$

also where $\Delta f_c = \Delta\omega_c / 2\pi, and\ R_{xy}$ is the Gold Code three valued cross correlation between G1 and any member of the Gold Code family [8], i.e.(for n=9 ,i.e. odd), and α^2 is intended signal to interference power ratio.

$$R_{xy} = \{\{-(2^{(n_1+1)/2}+1)/N\} \} = l_1 \quad \text{with probability .25}$$

$$=-1/N = l_2 \qquad \text{with probability .5} \qquad (4.5)$$

$$R_{xy} = \{\{(2^{(n_1+1)/2}-1)/N\} \} = l_3 \quad \text{with probability .25}$$

Repeating the same processing for the G2 MF, where the code epoch signal is absent, one gets

$$e_{2z} = a^2 + b^2 = (U-1)\frac{PN^2T_c^2}{2}R_{xy}^2(B^2+D^2) \tag{4.6}$$

We notice that in equation (6), we have used a different value for R_{xy} which is not necessarily equal all the time to R_{xy} of equation (4) (both are actually random variables), but both take 3 valued form in equation (5) in a random fashion.

Also we notice the presence of R_{xy}^2 in the 1st term due to passage of received code G1 through MF G2. Whenever the arriving signal code epoch matches with one of the 2 MFs (say G1 as above), we get equation (4) or (6) above corresponding to detection or miss conditions as will follow later. However, in the case where the received signal short code epoch does not match with both MFs G1 or G2 (due to time of arrival of received signal code with respect to the MFs), the output of e_{1z} and e_{2z} will both be represented by eqns (6) (We call this condition False Alarm).

Interestingly, with the very ideal situations where $\Delta f_c = R_{xy}\big|_{x\neq y} \equiv o$, one gets,

$$\begin{aligned} e_{1z} &= \frac{P}{2}N^2T_c^2 = \frac{E_b}{2}.(NT_c) \\ e_{2z} &= 0 \end{aligned} \tag{4.7}$$

as should be.

4.2.2 Performance of the VMF acquisition scheme

Defining $V_1 = e_{1z} / A$, $V_2 = e_{2z} / A$, the erroneous error probability in detecting one short Gold Code section (G1) is given by

$$P_{ss} = P\{V_2 > V_1\}$$

$$= \text{Prob.}\left\{R'^2_{xy}(U-1)(B'^2+D'^2) > (\alpha^2 B^2 + (B'^2+D'^2)(U-1)R_{xy}^2 + \alpha^2 D^2)\right\} \quad (4.8)$$

so far we have assumed that R_{xy} is the same for all CDMA interfering U terms in eqns (4.1)-(4.8), Returning now to the more general mode where this is not true, and noting the three levels of R_{xy} , i.e., l_1, l_2, l_3 for each interfering CDMA user, one gets

$$P_{ss} = \sum_{j_1=1}^{3} \sum_{j_2=1}^{3} \cdots \sum_{j_U=1}^{3} \prod_{i=1}^{U} P_{j_i} \quad \textit{where } P_{j_i} \textit{ is the probability of having } R_{xy} = l_{j_i} \quad (4.9)$$

Conditional on the $(j_1, j_2, \ldots, j_U)$ tupple being selected to satisfy eqn 8, which then changes to:

$$\left(\sum_{i=2}^{U} l'^2_{j_i}\right) > (\alpha^2 B^2 + (B'^2 + D'^2)\sum_{i=2}^{U} l^2_{j_i} + \alpha^2 D^2) \quad (4.10)$$

However computing equation (9) subject to (10) takes times of the order of (number of levels of R_{xy} raised to the number of users U, i.e. 3^{60} for example. In this case analysis would resort to the Gaussian assumption i.e. we assume both sides of the inequality (10) are distributed as Gaussian i.e. (10) is replaced by

$$Prob.\quad N(\bar{n},\sigma_n^2) > N(\bar{s},\sigma_s^2) \tag{4.11}$$

where $\bar{n}, \bar{s}, \sigma_s^2, \sigma_n^2$ are the means and variances of the respective terms in (10). Finally we get

$$P_{ss} = P(V_2 > V_1, o < V_1 < \infty, o < V_2 > \infty) \le Q(\frac{\bar{s} - \bar{n}}{2\sigma_1}) \tag{4.12}$$

Detail of deducing of P_{ss} is given in the next chapter. The next chapter also contains P_{es} which is similarly defined as P_{ss} (see Fig.3.6 on p. 64) before decoding, and

$$P_{sn} \le Q(\frac{\bar{n} - \bar{n}}{2\sigma_1}) \le 0.5 \tag{4.13}$$

However $P_{es}<P_{ss}$ due to the use of side information in the process of declaring erasures for BCH decoding. The probability ε of decoding an erasure is also derived in the next chapter. Erasures are declared as per Fig.3.6 on p. 64. If the arriving G1 or G2 short code signal is declared as erasure (as per Fig.3. 6 on p. 64) , the remaining received short codes sections of the current BCH word will suffer decision error with Probability $P_{es}<P_{ss}$ (Presumably since the short codes received in heavy interference has been erased). If these short Gold codes received in heave interference and/or Doppler are not erased then the BCH decoder will try to correct them anyhow, but the associated decision error $P_{ss}>P_{es}$ (both before decoding). The next chapter also shows the derivation of P_{sn}, P_{en}, in the case where both the G1, G2 MFS are receiving other short codes that don't

match their impulse Response, i.e. only interference present in both filters and/or unsynchronized G1 or G2 short codes (i.e. codes signal absent condition).

As mentioned before, the best sequence selector (seeFig.3.8, Fig.3.4 on p. 67 and 62) chooses one of the $2^{n_1}.q/M$ vectors of 31 components (1,0, ε) based on the comparison of the actual sampled voltage values. For the purposes of this analysis we assume Fig.3.8 on p. 67, to perform ideally.

4.2.3 Evaluation of BCH word decoding error

Returning back to the subsequent BCH decoder, we propose to use the same idea of erasure plus error correction [16] where the maximum number of erased symbols (short Gold Codes in our work) denoted E is selected adaptively based on the current load u, i.e.

$$E = \left[\frac{u}{U}\right] E_{\max}$$

Where u, U are the current and maximum number of overlapping packets (users), while E and Emax are the adaptive and maximum number of erasures respectively . For our selected (31,6) BCH code the design distance =2t+1=15.This code can correct any combination of erasures and errors such that $(2e+\varepsilon) \le d-1=14$. With e=0, we get Emax =14 ,however we apt to a lower value Emax =6 to allow for 4 occasional errors (e) correction ,and obtain,

$$E = \left[\frac{u E_{\max}}{U} \right] \tag{4.14}$$

where [x] refers to the integer part of (x)

It is seen that the BCH word decoding error for the BCH decoder (assumed to be currently receiving the synchronized code epoch signal present)) is given by:

$$P_{ws}(E) = \Pr ob.(2e + \varepsilon > d - 1)$$

$$= \sum_{j=0}^{E} \binom{N_1}{j} (\varepsilon_s)^j (1-\varepsilon_s)^{N_1-j} \sum_{2i+j \geq d,\, 0<i<N_1-j} \binom{N_1-j}{i} P_{es}^i \, (1-P_{es})^{N_1-i-j}$$

$$+ \sum_{j=E+1}^{N_1} \binom{N_1}{j} (\varepsilon_s)^j (1-\varepsilon_s)^{N_1-j} \underbrace{\sum_{i_1=0}^{N_1-j} \sum_{i_2=0}^{j-E}}_{2(i_1+i_2)+E \geq d} \binom{N_1-j}{i_1} \binom{j-E}{i_2} P_{es}^{i_1} (1-P_{es})^{N_1-i_1-j} P_{ss}^{i_2} (1-P_{ss})^{j-E-i_2}$$

(4.15)

The first two summations deal with the case the number of errors declared is less than the adaptive value allowed at the given load (given by Eqn.4.14). In this case an additional i errors (with P_{es} probability in light interference as per Fig.3.6 on p. 64 and the next chapter) can also be corrected out of (N_1–j) remaining Gold symbols (short codes) provided that $2i+j < d$. Now if $2i+j \geq d$, we get BCH word decoding errors. The following summations (Eqn.4.15) show other scenarios where the number of actual erasures (short codes in heavy interference) exceeds the adaptive limit set in Eqn.4.14, in which case the excess erasures will be treated as regular errors in heavy interference (with Prob. P_{ss} which is higher than

P_{es} and both are given in the next chapter) to be corrected in addition to the regular errors (with Prob. P_{es}) _.

4.2.4 Error probabilities under noise condition

We notice the presence of the s subscript to P_{es} , P_{ss} in eqn (4.15) to reflect the signal condition i.e. MF G1 is now receiving the right short code (G1) and MF G2 is receiving anything but G2 .

In addition to P_{ws} (E), will need for further development P_{wn} (E), which is the BCH word decoding error under interference only (code epoch absent in both G1, G2 MFs) This is a function of P_{en}, P_{sn}.

$$P_{en} = Q(\frac{\zeta - \bar{n}}{\sigma_n}).(1 - Q(\frac{\zeta - \bar{n}}{\sigma_n})) \tag{4.16}$$

$$P_{sn} \le Q(\frac{\bar{n} - \bar{n}}{2\sigma_1}) \le 0.5 \tag{4.17}$$

Actually this is the decoding word error of the other BCH decoder not containing the right code epoch (since at any time the user either transmits BCH word C1 or C2) P_{wn} (E) is obtained again from (4.15), by replacing P_{ss} , P_{es} by P_{sn} and P_{en} respectively, all of these probabilities are evaluated in the next chapter (see also Fig.3.6 on p. 64)

Needless to say that the assumption that all types of decision errors, are independent from short code to another has been utilized in (4.15).

4.2.5 Verification of the long concatenated signature code

The next box to be analyzed in Fig.3.4 on p. 62 is the long symbols (C1 or C2 BCH words) select box where details are given in the flow chart of Fig.3.9a, Fig.3.9b and Fig.3.10 on p. 73, 75 and p.76.

We assume without loosing generality that C1 is the currently received (BCH word or long symbol) and all its received sections (31=N1 received short Gold codes) have locked (has correct code epoch) to their respective (G1 or G2) MFs .This will further lead to the final signature code epoch detection condition (if n_2=5 such consecutive locked BCH words have been received) . In the mean time, and while the (C1 or C2) select box tries to find n_2=5 consecutive BCH symbols to load the LFSR with , errors could occur leading to missing the right signature code epoch, for example if the applicable EDM detects C1 C2 C2 C2 Z instead of its assigned word C1 C2 C1 C2 C1 where Z means the BCH decoder says (neither C1, nor C2) , (Fig.3.9a on p. 73). In this case the acquisition trial is aborted and a signature code epoch miss is declared (with Probability P_M). As

mentioned before, with strategy 1 of Fig.3.9a on p.73, the third leaf of the decision tree of Fig.3.9a leads to P_M.

To further illustrate we take the 1st leaf of the decision true of Fig.3.9 a. (that deals with the case where the received code phase matches with C1 without loosing generality). For this leaf the BCH decoder detects C1, this event probability is given by $P_{D1S} = (1-P_{ws}(E))$. The second leaf of Fig.3.9.a leads to False alarm, since C2 is detected instead of the assumed C1, the probability of this event is:

$P_{FA1S} = P_{ws}(E).\ (1/(N1-1))$. The third leaf leads to miss as mentioned before with probability: $P_{M1S} = P_{ws}(E).\ ((N1-2)/(N1-1))$ However, with strategy 2 (and corresponding to the last Z leaf shown in Fig.3.10 on p.76), we select randomly C1 or C2. In this case, the tie between C1, C2 selection is broken by a probability of ½ and finally two outcomes emerge from leaf 3, the first leading to the correct decision (C1) with probability equal to $P_{D3S} = P_{ws}(E).\ ((N1-2)/(N1-1)).\ (1/2)$

The other outcome of the third leaf of Fig.3.10 on p.76 contributes to the code False Alarm probability, i.e. $P_{FA3S} = P_{ws}(E).\ ((N1-2)/(N1-1))\ .(1/2)$ The first and second leafs of Fig.3.10 lead to correct epoch detection and false alarm respectively with probabilities equal to

$$P_{D2S} = (1-P_{ws}(E)),\ \text{and}\ P_{FA2S} = P_{ws}(E).\ (1/(N1-1)) \qquad (4.18)$$

It is easy to see that with strategy 2 the EDM will never miss!! (i,e under the correct code epoch presence P_M =0), the EDM will have either Detection or False Alarm conditions.

If the signature code epoch is absent, we have a problem with strategy 2 since False alarms will be more frequent.(with strategy 1 ,Fig.3.9b on p. 75 shows that many leaves of the tree lead to False Alarm avoidance) . In Fig.3.9b.(correct signature code epoch absent) the values of the four leaves probabilities are given by:

$$P_{FA3n} = P_{wn}(E) \,.\, (1/N_1\text{-}1) \qquad (4.19)$$

$$P_{FA4n} = P_{wn}(E).\ (1/N_1\text{-}1) \qquad (4.20)$$

$$P_{NFA5n} = (1 - P_{wn}(E)) \qquad (4.21)$$

$$P_{NFA6n} = P_{wn}(E).(N_1\text{-}3)/(N_1\text{-}1) \qquad (4.22)$$

Probabilities contributing to the False alarm (PFA) in both figures (Figs. 3.9a, 3.9b) will have to be averaged somehow! ,i.e., for strategy 1

$P_{FA1} = r\,P_{FA1S} + (1\text{-}r)(P_{FA3n} + P_{FA4n}) = P_{wn}(E).2/(N_1\text{-}1)$,(r approximately negligible)

(4.23)

Where r is the signature code alignment probability, i.e. $r = 1/R_1 = 1 / ((2^{n_1} - 1)\cdot q/M)$ assumed 0 in the approximation above.

While for strategy 2, one gets the following averaged FA probability:

$$P_{FA2} = r.\ (P_{FA2S} + P_{FA3S}) + (1-r) \times 1 \qquad (4.24)$$

Which is very high (approximately equal to 1-r)

Finally, since in Fig.3.5 on p. 63, $n_2 = 5$ long symbols (C1, C2) are loaded into the LFSR and code verification is tried (of course after mapping C1 to 1 C2 to 0), the code acquisition trial succeeds if the right $n_2 = 5$ BCH words are loaded into the LFSR this fact yields the overall signature code detection probability i.e.,.

$$P_{D1}^{'} = (P_{D1S})^{n_2} \qquad (4.25)$$

However overall signature code epoch Miss and False alarm may also result, i .e.

$$P_{M1}^{'} = 1 - (1 - P_{M1S})^{n_2} \qquad (4.26)$$

$$P_{F1}^{'} = (P_{FA1})^{n_2} \qquad (4.27)$$

Back to Fig.3.4, Fig.3.5 on p. 62 and p.63, we see that each EDM facing on average an acquisition time $\overline{T_{S1}}$ whose analysis resembles that of an equivalent serial search process (Recall our technique includes MFs !! FEC, parallel processing, and the equivalence is in the analysis concept only as will be reflected by our Detection and false alarm probabilities which are much better than that of the serial classic search) and by the generalized circular state diagram [8] one obtains ,

$$\overline{T_{S1}} \equiv \frac{T_i}{P_{D1}^{'}} + \left(\frac{2 - P_{D1}^{'}}{2P_{D1}^{'}}\right) (R_1 - 1)\ (T_i + P_{F1}^{'}\ T_{fa}) \qquad (4.28)$$

where T_{fa} = false alarm penalty time selected as $\propto R_1.n_2.M = 2.R_1.n_2.M.T_c$ *for example and* T_i *is* the time to investigate one search cell of R_1 i.e.,

T_i = maximum of (T_p, T_s) where T_p = software processing time of BCH decoding of n_2 BCH words plus the best sequence ,and other processing boxes of Fig.3.5 on p. 63.

These could be much less than $.R_1.n_2.M.T_c/q$, i.e the actual corresponding received signal record length since the BCH decoder is loaded and its results loaded into the LFSR in parallel not in series. Notice also the FIFO of Fig.3.5 on p. 63 has all the necessary samples corresponding to the n_2 BCH words available for parallel loading to the single or multiple (if one wishes to cut the processing time) BCH decoders. In this work we take $T_i = R_1 \cdot n_2 \cdot M \cdot T_c / 8q$. Here we notice that this is actually a worst case design, recall we have two LFSR working in parallel, one verifying one discovered code epoch while the MFs, FIFOs, etc... are trying parallel in time to find another code epoch, this implies that T_i, i.e. one cell evaluation time is included in T_{fa}, i.e. the verification time of the previously discovered code epoch .So one can safely say T_i =0 with our VMF, but for classic code search schemes the two processes mentioned run series in time and so T_i is

not equal to 0. T_s = time to feed in new data record to the G1, G2, MFs i.e. $T_s = (2^{n_1} - 1)T_c$ sec.

In this work we look at a different strategy (strategy 2 of Fig.3.10 on p. 76 while Figs 3.9.a.b correspond to strategy 1) . In strategy 2, no miss is allowed, i.e the C1, C2 selector box of figures always dump C1 or C2.

This means that with strategy 2, one sees a succession of false alarms followed by the final successful trial, this is represented by a geometrical distribution, adding the average of which to the T_i cell evaluation time , we get the average acquisition time of strateg2 ,i.e.

$$\overline{T_{S2}} = T_i + \sum_{j=1}^{\infty} j.T_{fa}.(1 - P_{D2}^{-}.r)^{j}(P_{D2}^{-}.r) = T_i + T_{fa}.(1 - (P_{D2}^{-}.r))/(P_{D2}^{-}.r) \qquad (4.29)$$

where $P_{D2}^{-} = (P_{D2S} + P_{D3S})^{n_2}$ *and* $P_{FA}^{-} = 1 - r$, and each acquisition trials take T_{fa} sec. Except the last trial that takes T_i sec. only .

Chapter 5
Analysis of the VMF

In Chapter 4, for the sake of completeness, we directly give the results without detailed implementation. In this chapter, we introduce a specified analysis of the proposed hybrid acquisition scheme. We go deeply into the equations step by step. Certain assumptions are made in the analysis of the scheme to simplify the complexity involved in developing the expressions. These rigorous deducing would help us to get a better understanding of the VMF. Also it is necessary to show the advantage of our VMF among other systems. Finally numerical results of this model are presented.

5.1 Evaluation of mean and variance of signal and noise of our VMF

Evaluating the mean and variance of each of the 2 sides of the inequality in (4.10), using the instantaneous load u rather than the maximum load U, and utilizing the statistics of the Gold Code cross correlation, Eqn.(4.5), we get,

$$\begin{aligned}&(\sum_{i=1}^{u} l_{ji})^2 \\ &= (l_{j1} + l_{j2} + \ldots + l_{ju})^2 \\ &= \underbrace{l_{j1}^2 + l_{j2}^2 + \ldots + l_{ju}^2}_{u} + \underbrace{2(l_{j1}l_{j2} + l_{j1}l_{j3} + \ldots + l_{j1}l_{ju})}_{u-1} + \ldots + 2l_{j(u-1)}l_{ju}\end{aligned} \tag{5.1}$$

hence, we could get:

$$\overline{l_{ji}} = -1/N \tag{5.2}$$

$$\overline{l_{ji}^2} = (2^{n_1} + 1)/N^2 = (N+2)/N^2 \tag{5.3}$$

where $N = 2^{n_1} - 1$.

From the equation (4.4) in the previous chapter, and after normalization we could get

$$\bar{s} = E\{e_{12}\} = E\{\alpha^2 B^2 + \alpha^2 D^2 + (B^{-2} + D^{-2})(u-1)R_{xy}^2\} = \alpha^2 B^2 + \alpha^2 D^2 + (B^{-2} + D^{-2})(u-1)\overline{l^2} \tag{5.4}$$

In the same way from equation (4.6), we could have

$$\bar{n} = E\{e_{22}\} = (B^{-2} + D^{-2})(u-1)\overline{l^2} \tag{5.5}$$

The next step is to calculate the variance of signal and noise, we assume that they are the same. From equation (4.3) and (4.4), we have (Fig. 3.4)

$$
\begin{aligned}
&a^2+b^2= \\
&\alpha^2B^2+\alpha^2D^2+\left(B'^2+D'^2\right)\sum_{i=2}^{u}R_i^2 \\
&+\sum_{i=2}^{u}\left[\cos(\phi-\phi_i)+\cos(\phi+\phi_i)\right]\frac{B'\alpha BR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}\left[\cos(\phi_i-\phi_j)+\cos(\phi_i+\phi_j)\right]\frac{B'B'R_iR_j}{2} \\
&+\sum_{i=2}^{u}\left[\cos(\phi-\phi_i)-\cos(\phi+\phi_i)\right]\frac{D'\alpha DR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}\left[\cos(\phi_i-\phi_j)-\cos(\phi_i+\phi_j)\right]\frac{D'D'R_iR_j}{2} \\
&+\sum_{i=2}^{u}-\left[\sin(\phi-\phi_i)+\sin(\phi+\phi_i)\right]\frac{B'\alpha DR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}-\left[\sin(\phi_i-\phi_j)+\sin(\phi_i+\phi_j)\right]\frac{B'D'R_iR_j}{2} \\
&+\sum_{i=2}^{u}\left[\sin(\phi-\phi_i)-\sin(\phi+\phi_i)\right]\frac{D'\alpha BR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}\left[\sin(\phi_i-\phi_j)-\sin(\phi_i+\phi_j)\right]\frac{D'B'R_iR_j}{2} \\
&+\sum_{i=2}^{u}\left[\cos(\phi-\phi_i)+\cos(\phi+\phi_i)\right]\frac{D'\alpha DR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}\left[\cos(\phi_i-\phi_j)+\cos(\phi_i+\phi_j)\right]\frac{D'D'R_iR_j}{2} \\
&+\sum_{i=2}^{u}\left[\cos(\phi-\phi_i)-\cos(\phi+\phi_i)\right]\frac{B'\alpha BR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}\left[\cos(\phi_i-\phi_j)-\cos(\phi_i+\phi_j)\right]\frac{B'B'R_iR_j}{2} \\
&+\sum_{i=2}^{u}\left[\sin(\phi-\phi_i)+\sin(\phi+\phi_i)\right]\frac{D'\alpha BR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}\left[\sin(\phi_i-\phi_j)+\sin(\phi_i+\phi_j)\right]\frac{D'B'R_iR_j}{2} \\
&+\sum_{i=2}^{u}-\left[\sin(\phi-\phi_i)-\sin(\phi+\phi_i)\right]\frac{B'\alpha DR_i}{2}+\sum_{i=2}^{u}\sum_{j=2\neq i}^{u}-\left[\sin(\phi_i-\phi_j)-\sin(\phi_i+\phi_j)\right]\frac{D'B'R_iR_j}{2}
\end{aligned}
$$

Variance of the first sum in the above equation is

$$\left(B'^2+D'^2\right)^2\frac{u-1}{N^2} \tag{5.6}$$

We note that the expected values of all cross terms involving different lines of the above equation are all zeros, now evaluating the variance of the other terms we abtain (approximately):

$$\frac{u-1}{4}\left(2\alpha^2B'^2B^2+2\alpha^2D'^2D^2+2\alpha^2B'^2D^2+2\alpha^2D'^2B^2+8\alpha^2BB'DD'\right)/N$$
$$+\frac{(u-1)(u-2)}{4}\cdot\left(2B'^4+2D'^4+4B'^2D'^2\right)\cdot\left[\overline{R_i^2}\cdot\overline{R_j^2}-\left(\overline{R_i}\cdot\overline{R_j}\right)^2\right] \quad (5.7)$$

Since $\left[\overline{R_i^2}\cdot\overline{R_j^2}-\left(\overline{R_i}\cdot\overline{R_j}\right)^2\right]=\frac{(N+2)}{N^4}-\frac{1}{N^4}\cong\frac{1}{N^2}-\frac{1}{N^4}\cong\frac{1}{N^2}$, so that we obtain

$$\sigma_S^2=\sigma_n^2=\left(B'^2+D'^2\right)^2(u-1)/N^2+\frac{u-1}{4}$$
$$\cdot\left(2\alpha^2B'^2B^2+2\alpha^2D'^2D^2+2\alpha^2B'^2D^2+2\alpha^2D'^2B^2+8\alpha^2BB'DD'\right)/N$$
$$+\frac{(u-1)(u-2)}{4}\left(2B'^4+2D'^4+4B'^2D'^2\right)/N^2 \quad (5.8)$$

where we have assumed the independence of the different Gold Code cross correlation i.e. $E(l_1l_2)=E(l_1)E(l_2)$ similarly $E(l_x^3l_y)=E(l_x^3).E(l_y)$, and $E(l_x^2l_y^2)=E(l_x^2).E(l_y^2)$ and so on.

It is not difficult to check that $\sigma_n^2=\sigma_s^2$ and hence from the optimal detection theory of ASK signals, the optimum decision threshold for equation (4.10) and (4.11) is given by $\zeta=\left(\frac{\bar{s}+\bar{n}}{2}\right)$. It is also clear that ζ is a function of the instantaneous user load u, because of the load adaptive erasure policy used (same as adaptive declaration of erasures of Fig.3.6 and equation (4.14)).

5.2 Developing of the performance under signal epoch presence and absence

Now we try to find $\varepsilon_s, P_{es}, P_{ss}$ of Fig.3.6, all under signature code signal epoch presence, where V_1 *and* V_2 are respectively the outputs of the Gold codes MFs containing the signal (completely matching the arriving short Gold code) and not containing the signal i.e.

$$\varepsilon_s = P(V_1 > \zeta).P(V_2 > \zeta) + P(V_1 < \zeta).P(V_2 < \zeta)) \tag{5.9}$$

Now,

$$P(V_1 > \zeta) = \frac{1}{\sqrt{2\pi}\sigma_s}\int_{\zeta}^{\infty} e^{-(v_1-\bar{s})^2/\sigma_s^2}\, dv_1 = Q(\frac{\zeta-\bar{s}}{\sigma_s}) \tag{5.10}$$

Also

$$P(V_2 > \zeta) = \frac{1}{\sqrt{2\pi}\sigma_n}\int_{\zeta}^{\infty} e^{-(v_2-\bar{n})/2\sigma_n}\, dv_2 = Q\left(\frac{\zeta-\bar{n}}{\sigma_n}\right) \tag{5.11}$$

So

$$\varepsilon_s = Q\left(\frac{\zeta-\bar{s}}{\sigma_s}\right).Q\left(\frac{\zeta-\bar{n}}{\sigma_n}\right) + (1-Q\left(\frac{\zeta-\bar{s}}{\sigma_s}\right).(1-Q\left(\frac{\zeta-\bar{n}}{\sigma_n}\right)) \tag{5.12}$$

Similarly , under signature code epoch absence (interference only, no acquisition resulting), we Get,

$$\varepsilon_n = (Q(\frac{\zeta-\bar{n}}{\sigma_n}))^2 + (1-Q\ (\frac{\zeta-\bar{n}}{\sigma_n}))^2 \tag{5.13}$$

Similarly, the probability of Gold code decision error, when there is no erasure declared, i.e. no high interference and the signature code epoch signal is present is given by (see Fig.3.6),

$$P_{es} = P(V_2 > \zeta\ ).P(V_1 < \zeta) = Q(\frac{\zeta - \bar{n}}{\sigma_n})(1 - Q(\frac{\zeta - \bar{s}}{\sigma_s})) \qquad (5.14)$$

Similarly, in the code epoch absence, the Gold code decision error probability is:

$$P_{en} = Q(\frac{\zeta - \bar{n}}{\sigma_n}).(1 - Q(\frac{\zeta - \bar{n}}{\sigma_n})) \qquad (5.15)$$

The probability of Gold code decision error under erasure condition is given by (Fig.3.6) follows from elementary ASK demodulation. In this case the two Gold MFs outputs have to be compared to each other (notice in all the previous probabilities cases, these outputs were compared to a threshold and to each other.

$$P_{ss} = P(V_2 > V_1\ , o < V_1 < \infty\ , o < V_2 > \infty) \le Q(\frac{\bar{s} - \bar{n}}{2\sigma_s}) \qquad (5.16)$$

Similarly, under the signature code epoch absence,

$$P_{sn} \le Q(\frac{\bar{n} - \bar{n}}{2\sigma_n}) \le 0.5 \qquad (5.17)$$

Inserting $\varepsilon_s, P_{es}, P_{ss}$ into equation (4.15), one obtains $P_{ws}(E)$. Repeating the same with $\varepsilon_n, P_{en}, P_{sn}$ into equation (4.15) obtains $P_{wn}(E)$.

5.3 Computation of other probabilities and acquisition time

One now proceeds with $P_{ws}(E), P_{wn}(E)$ to equation (4.16) and the rests to compute the remaining probabilities and acquisition times.

We will also need for comparison purposes P_{fa}, P_d, P_m of serial, long Physical M.F, and M parallel MFs. These probabilities will be derived in the same Doppler, CDMA interference environments as our VMF, however one should expect values worse than these of VMF due to the FEC utilization of RS decoding of our VMF and the elimination of Doppler effects due to the use of short Gold codes. For example a serial search scheme that corresponds to our virtual MF would have T_{i4} = search integration time $= R_1 \cdot N_1 \cdot n_2 \cdot M \cdot T_c / q$ and from the standard Gaussian analysis one gets,

$$P_{D4} = Q(\frac{\zeta - \bar{s}}{\sigma_s}) \tag{5.18}$$

$$P_{F4} = Q(\frac{\zeta - \bar{n}}{\sigma_n}) \tag{5.19}$$

where $\zeta, \bar{n}, \bar{s}$ has the same formats above , however due to the longer T_{i4} of serial research $N.T_c = (2^{n_1} - 1).T_c$ is replaced by $(2^{n_1} - 1)N_1.n_2.T_c$ and N by

$(2^{n_1}-1)N_1.n_2$ in all relevant equations of this chapter and text ,in particular the Doppler related constants A,B,D,B',D' and $\zeta, \bar{n}, \bar{s}$.

We also notice that $\zeta, \bar{n}, \bar{s}, \sigma_s, \sigma_n$ all get multiplied by the same constant A (equation 4.4) thus leaving P_{F4} and P_{D4} unchanged as in equation (5.18) and (5.19).

For the long Physical and M parallel MF cases the developments of the probabilities above are similar and they give equal values since we have intentionally made the length of each MF compatible with the serial search T_{1s} (For analysis convenience and easy comparisons to our VMF.

Chapter 6

Comparison VMF with other system and discussion

In this chapter, we compare our VMF with serial search, long mated filter and parallel matched filter. The acquisition time of those systems are given for the purpose of illustration. We have obtained the expressions for our VMF in the previous chapter, and the program to calculate those equations is attached in the appendix. Here we use the results of the program and show it in figures to explain our VMF more straightforwardly and more easy-understanding rather than the theoretical equations in the previous chapter. Based on these comparisons, we discuss the properties and many advantages of our VMF among other systems.

6.1 Acquisition time for other systems

6.1.1 Classic serial search approach

We have already introduced the classic serial search in the first chapter, and here for the purpose of comparison with our VMF, we should give the expression of the acquisition time for this classic scheme. In this work, we compare our results of different schemes under the same Environment of Doppler, CDMA load. etc. For serial search,

$$\overline{T_{S4}} \cong \frac{T_{i4}}{P_{D4}} + \left(\frac{2 - P_{D4}}{2P_{D4}}\right)(R_4 - 1)(T_{i4} + P_{F4}\, T_{fa}) \qquad (6.1)$$

Where $T_{fa} = 2R_4 n_2 M T_c$, $T_{i4} = R_1 N_1 n_2 M T_c / q$ sec, which is about is a fraction $= n_2/(2^{n_2} - 1)$ of the total signature code length and the remaining probabilities are functions of the effective integration time (T_{i4}), and T_{fa} as derived in the previous chapter.

6.1.2 Long physical Matched Filter and Parallel Matched Filter

For a long physical M.F of length $T_{i4} = .R_1 N_1 .n_2 .M .T_c / q$ sec. i.e. . $.R_1 N_1 .n_2 .M./q$ code chips which is a fraction of the total signature code length as above ,the average acquisition time is given by [8]

$$\overline{T_{S5}} \equiv \frac{T_c/q}{P_{D4}} + (\frac{2-P_{D4}}{2P_{D4}})(R_4 - 1)(T_c/q + P_{F4} T_{fa}) + T_{i4} \qquad (6.2)$$

where we notice the evaluation time reduces to T_c/q (Blessing of MF !), and the initial full up time is T_{i4}. R_4 *and* T_{fa} *remain the same as for serial search* , the remaining probabilities remaining the same as above since we have assumed the length of the MF to be equal to the integration time of the serial search and are functions of the effective integration time (length of the MF $= T_{i4}$), and T_{fa} as derived in the previous chapter.

For a M parallel Matched Filtering, it has the same structure as the Matched Filter except that it has been separated by several shorter filters. Hence the acquisition time is

$$\overline{T_{S6}} \equiv \frac{T_c/q}{P_{D4}} + (\frac{2-P_{D4}}{2P_{D4}})(R_6 - 1)(T_c/q + P_{F4} T_{fa}) + \frac{T_{i4}}{M} \qquad (6.3)$$

where $R_6 = R_4 / M^{'}$, $T_{i4} = R_1 \cdot N_1 \cdot n_2 \cdot M \cdot T_c / q$ is the corresponding length of each of the $M^{'}$ MFs and ($M^{'}$. T_{i4}) is less than or equal to (CL.T_c) = Total signature code length (or resources will be wasted). The remaining probabilities have been evaluated in the previous chapter.

6.2 Discussion

6.2.1 Assumptions of the systems

Here we have assumed that the serial search, long and parallel MFs still use Gold Code (but a longer one) and faces the same load u and Doppler environment as our VMF, also the same Δf_c, and the same optimal load adaptive threshold ζ !! Finally the $M^{'}$ parallel MFs case will be of the same order of complexity as our VMF, even worse since it relies on longer physical hardware

Implemented MFs, in contrast to our VMF which needs only 2 much shorter Gold MFs, and leaves it to software to take care of the remaining processing.

6.2.2 Properties of the VMF

On the other hand, the new Hybrid scheme is computation intensive, yet it is fault tolerant since Doppler and carrier shifts are neutralized (Powerful parallel BCH FEC Codes are used), but high speed software processing is needed to enable the

parallel operation of say 16 such EDMs and still cut the processing time T_p and its effect on the Acquisition time as already been shown.

One may notice that the whole arrangement in Fig. 3.4 on page 62 and Fig. 3.5 on page 63 is approximately equivalent to a long digital Physical M.F of length $((2^{n_1}-1)\cdot N_1\cdot(2^{n_2}-1)\cdot qT_s)$ though it does not suffer the severe Doppler frequency offset problems that this filter has. A long MF of this length would completely loose the signal (as per the analysis section).

Also this long MF would not have the true error correction capacity of the ((31,6)) BCH code used in our scheme which can shorten the acquisition time under large CDMA loads. Interestingly, breaking the shorter IS-95 pilot code of length $(2^{15}-1)$ chips into smaller shorter codes and trying our BCH codes or alternate ideas would not work! Still one needs many shorter physical different MF, each matched to a small section of the IS 95 short code which multiplies the left part (the G1, G2 MFs) of Fig. 3.5 on page 63 by many orders of magnitude. Our idea of constructing the long code from shorter Gold codes is amenable to schemes as that of Fig.3.4 on page 62 and Fig. 3.5 on page 63.

6.2.3 VMF Results and comparison with other system

The following figures are under different system parameters and comparison with other systems. Those results are based on the equations given in this thesis. To get a clear and easy visual figure, computer program is employed and the software language appears in the Appendix.

All figures in this chapter are obtained results, the unit of acquisition time is in seconds.In this group of figures, we want to compare T_{s1}, T_{s2}, T_{s11}, T_{s4}, T_{s5}, T_{s6} under these parameters:

$n_1=5$, $n_2=8$, $N_1=63$, d=9, U=35, $E_{max}=3$

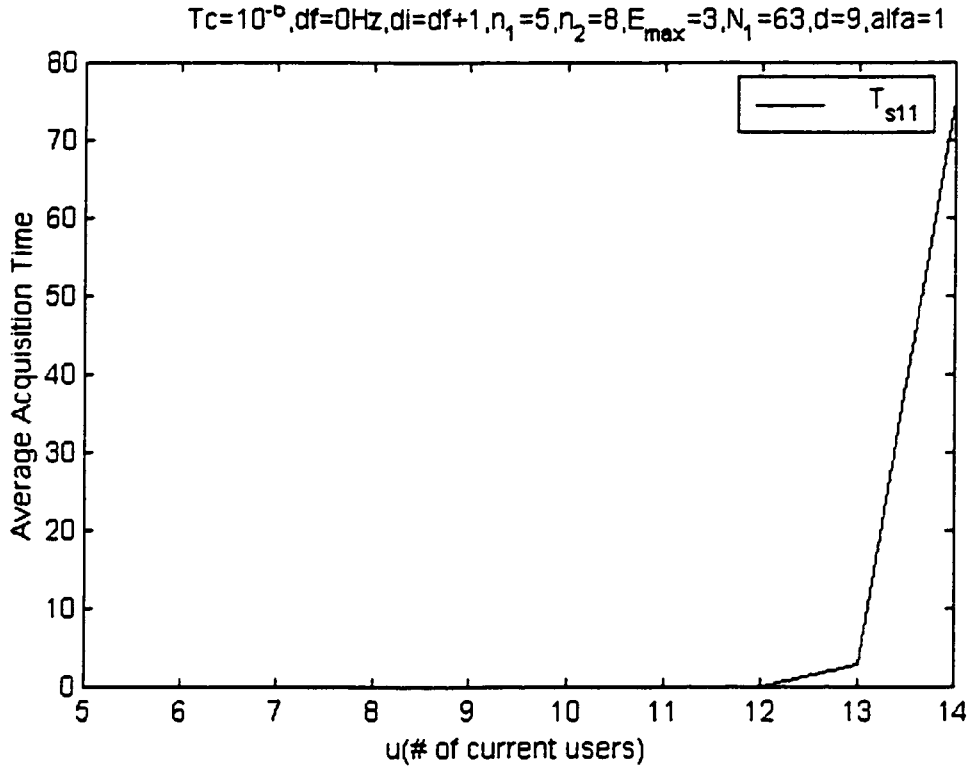


Fig. 6.1 T_{s11} under Doppler is 0

- T_{s11} ($T_I \neq 0$) is larger than T_{s2} and T_{s1} ($T_I=0$) but smaller than T_{s4}, T_{s5}, and T_{s6}. The reason is that we consider the processing time of BCH decoder in this case, and actually the processing time is included in the false alarm time in our VMF, so this is the worst case, even it is the worse case, it is still better than other systems.

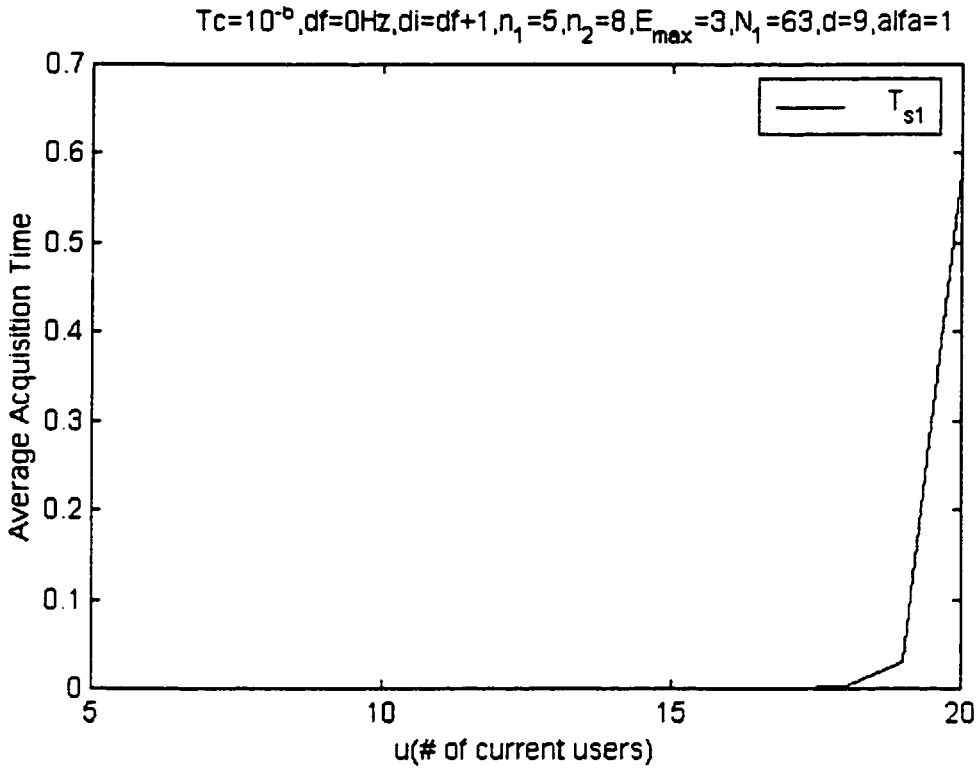


Fig. 6.2 T_{sl} under Doppler is 0

- This is our strategy 1, assuming processing time is 0 (T_l=0) which is the reasonable ideal case. For 0<u<15, the result is excellent.

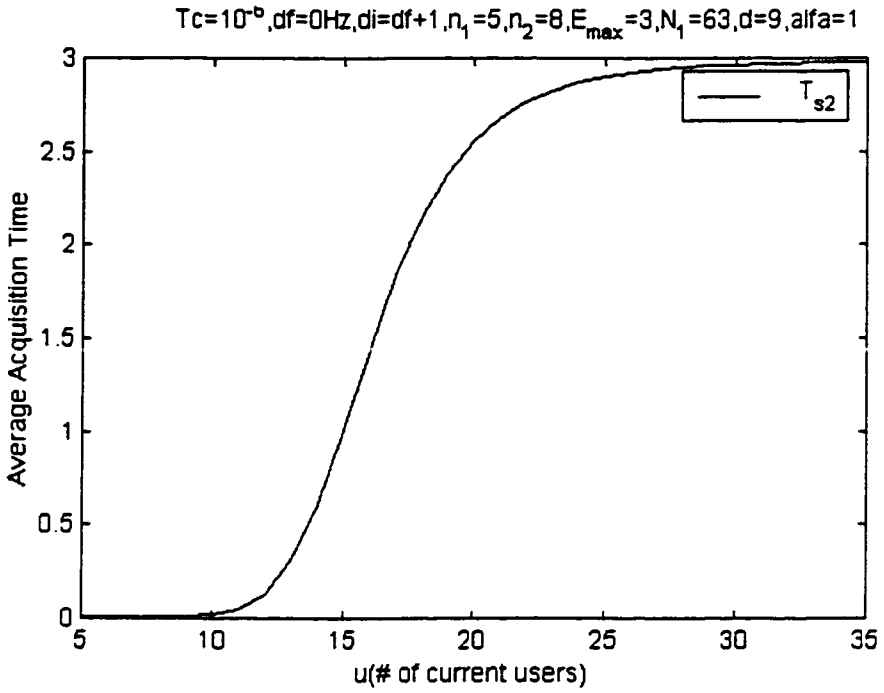


Fig. 6.3 T_{s2} under Doppler is 0

- This is the case when processing time is 0 while using strategy 2, the result is also excellent for $0<u<15$. When compare this with T_{s1}, we could see the advantage of strategy two is that it works well under heavy load, the reason is that the probability of false alarm is higher but the false alarm penalty time is already absorbed by our back to back shift registers. Recall in strategy 1, when u is larger, P_{es}, P_{ss} approach ½, and time is wasted in rejecting all these search cells. In strategy 2, we flip a coin and obtain code sequence even if many BCH words are in error (by erasure and error correction capability of he dynamic erasure based BCH codes). Thus solving some of the cases that have been otherwise wasted by strategy 1.

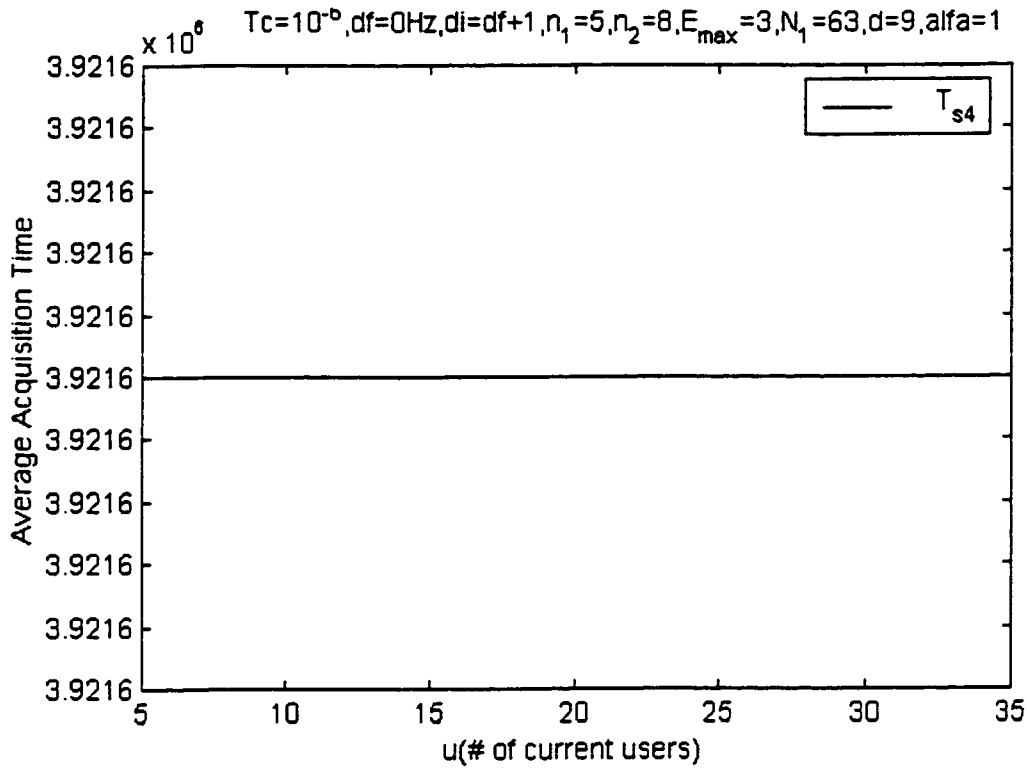


Fig. 6.4 T_{s4} under Doppler is 0

- This is the serial search, and even under no Doppler offset, this method doesn't work (T is huge and unpractical as u exceeds 4 users). When code is long, the uncertainty region is large, so this method is seldom used in reality.
- It doesn't change with number of users, this is reasonable according our equations, since Doppler is 0. This is also true for system 5 and 6 as will be seen next page. But when Doppler is not equal to zero, it will change with the number of users.
- The reason is that even a small number of users is enough to cause severe P_{fa}, even if $\Delta f = 0$.

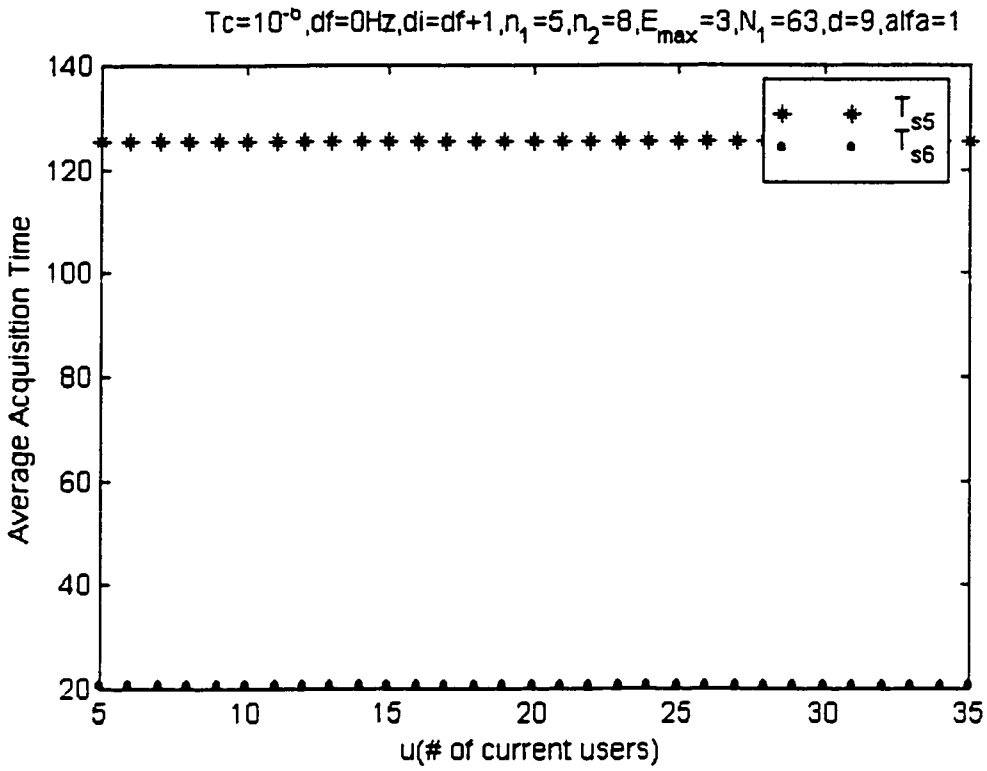


Fig. 6.5 T_{s5} and T_{s6} under Doppler is 0

- Due to the same reason they don't change with number of users as the previous figure.
- MF and parallel MF are much better than serial search, and tolerable, so these methods are more practical.
- But it is worse than our VMF, our new hybrid scheme is better compared with classic schemes as those ones.
- We should also notice that the number of parallel filter is 6, so T_{s6} is almost 1/6 of T_{s5}, parallel filter has better effecting than a single matched filter.

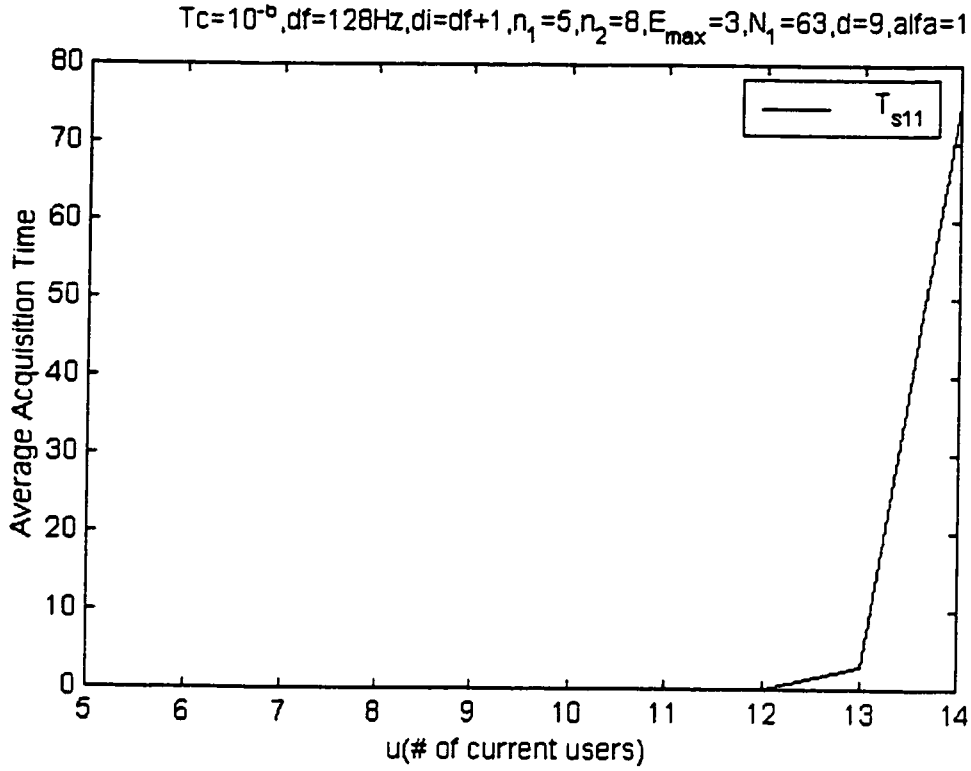


Fig. 6.6 T_{s11} under Doppler is 128

- Compare this with Fig. 6.1, all the parameters are the same, except Doppler, from these two figures, it is clear that Doppler could severely decrease our system. This is sometime the crucial point in mobile system design.
- Fortunately our VMF could tolerate appreciable Doppler value. From the above figure, you could find that when system is not heavily loaded, u<12, our VMF still works, even this T_{s11} is the worse case among our schemes.

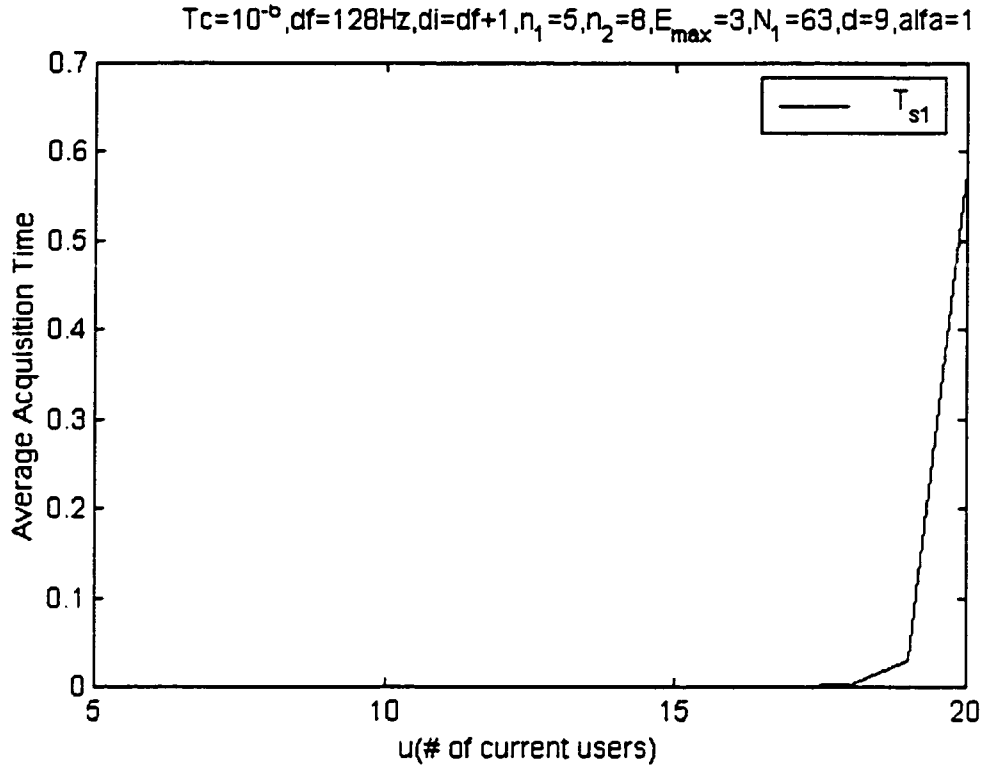


Fig. 6.7 T_{s1} under Doppler is 128

- From this figure we could see that even Doppler is heavy, our ideal system (processing time is equal to 0 which could be the real case) works very well, almost no notable delay of the acquisition time. Our VMF is in some way 'Doppler resistant'.
- For T_{s2}, it is almost the same good situation as T_{s1}, so it is unnecessary to display that figure.

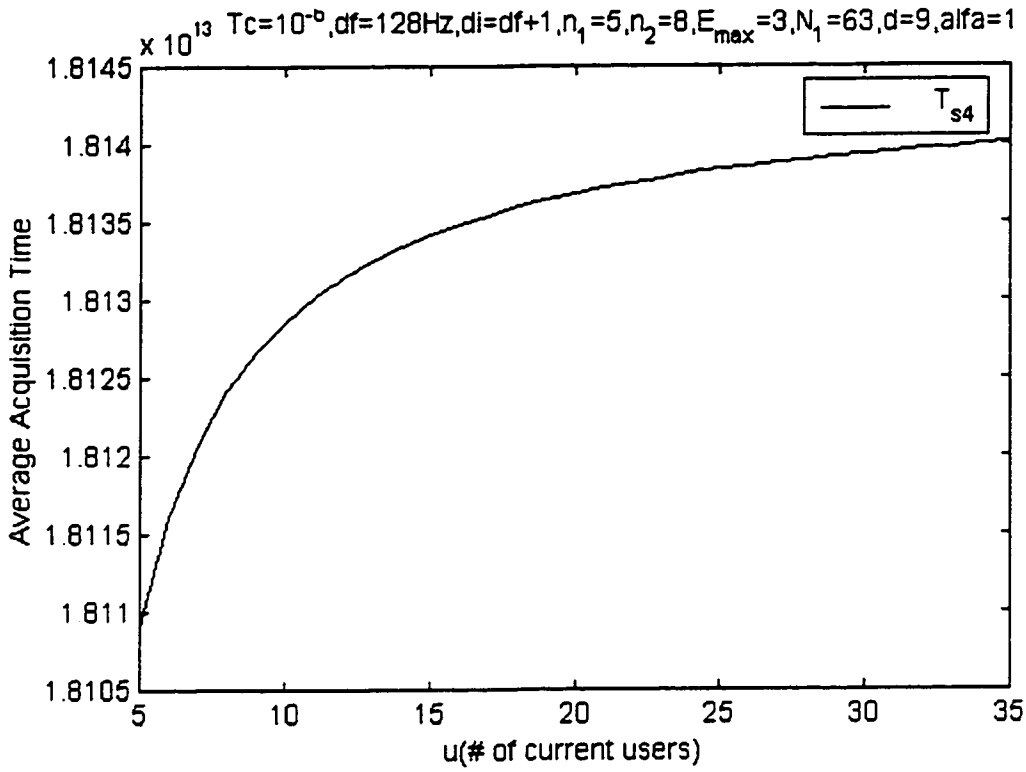


Fig. 6.8 T_{s4} under Doppler is 128

- Comparing this with Fig. 6.4, of course it is just impracticaly large under such heavy Doppler.
- Also in Fig. 6.4, it doesn't change with users, since Doppler is 0. Now when Doppler isn't equal to 0, the acquisition time increases monotonously with the number of users.

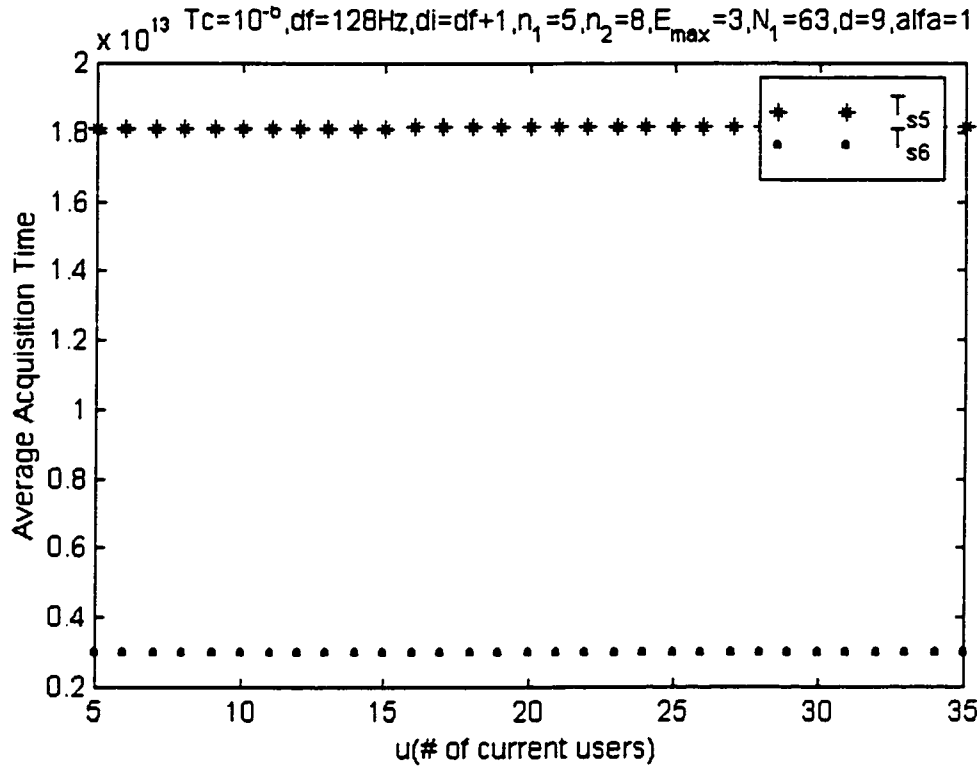


Fig. 6.9 T_{s5} and T_{s6} under Doppler is 128

- Even Matched Filter and Parallel filter are unable to carry such heavy Doppler, so our VMF is the only survivor technique under high Doppler.
- It is worth mention that T_{s5} and T_{s6} really changes with u, but too slight to be distinguished in this figure. The following table 6.1 is just the row data of this figure, it means that T_{s5}, T_{s6} mainly depend on Doppler, increasing u adds little to the increase of T_{s5} and T_{s6} as obtained.

Table 6.1 Variation of T_{s5} and T_{s6} VS number of users of Fig. 6.9

Users #	T_{s5}	T_{s6}
9	.181267E+14	.302111E+13
10	.181286E+14	.302143E+13
11	.181302E+14	.302169E+13
12	.181314E+14	.302208E+13
13	.181325E+14	.302223E+13
14	.181334E+14	.302236E+13
15	.181342E+14	.302248E+13
16	.181349E+14	.302257E+13
17	.181354E+14	.302266E+13
18	.181360E+14	.302274E+13
19	.181364E+14	.302280E+13
20	.181368E+14	.302287E+13
21	.181372E+14	.302292E+13
22	.181375E+14	.302297E+13
23	.181378E+14	.302302E+13
24	.181381E+14	.302306E+13
25	.181384E+14	.302306E+13
26	.181386E+14	.302310E+13
27	.181388E+14	.302314E+13
28	.181390E+14	.302317E+13
29	.181392E+14	.302320E+13
30	.181394E+14	.302323E+13
31	.181395E+14	.302326E+13
32	.181397E+14	.302328E+13

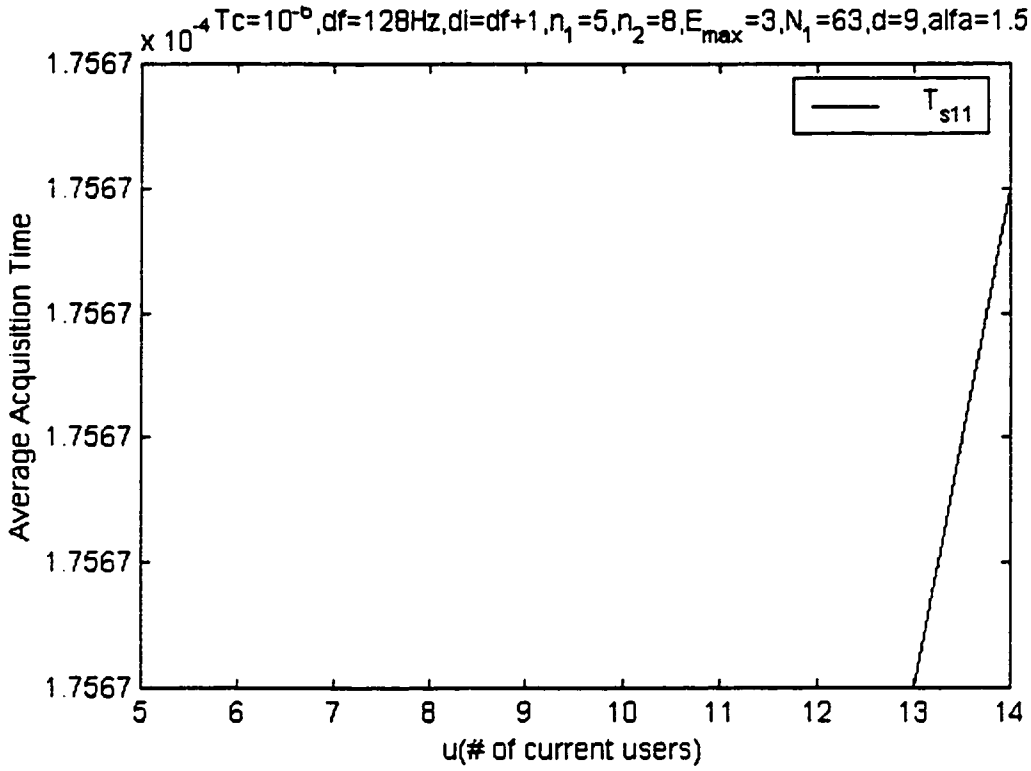


Fig. 6.10 T_{s11} under Doppler=128, and α=1.5

- α^2 is signal to interference power ratio, it should play an important part in every system. Here we set $\alpha = 1.5$, and use our T_{s11} as a typical example. In fact, as read from the figure, this ratio improves our system almost 1000 times under Doppler=128. For both T_{s1} and T_{s2} , a moderate value of α as this leads to appreciable acquisition time improvement.

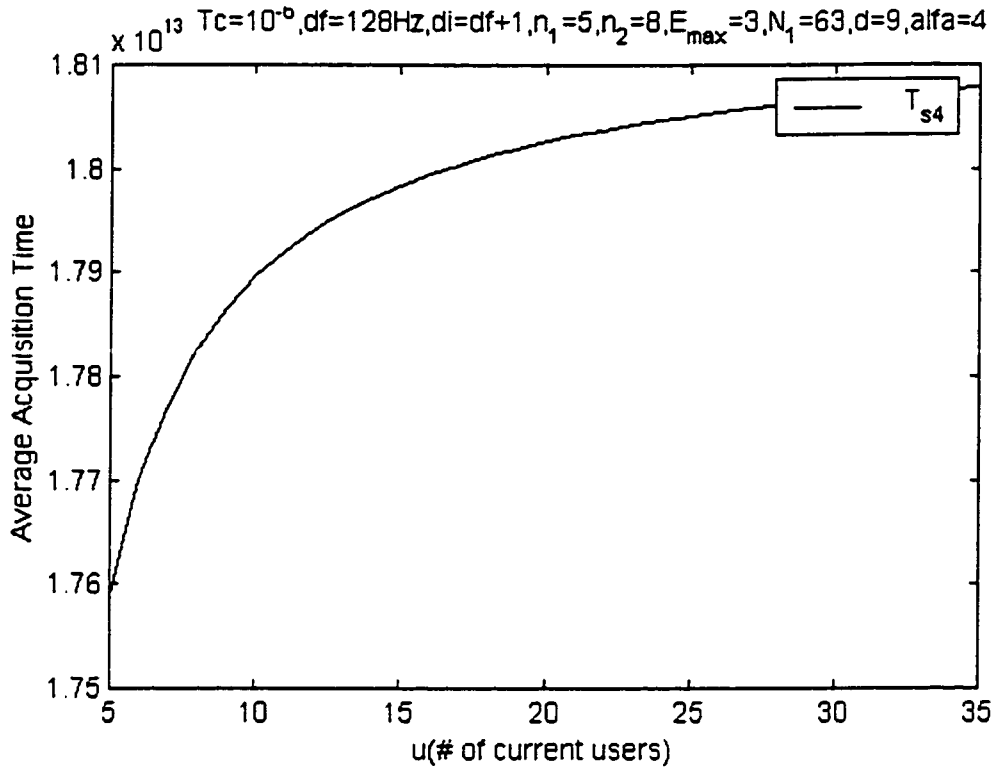


Fig. 6.11 T_{s4} under Doppler is 128, α=4

- α is a nice parameter to improve the performance. But as shown in the above figure, it is difficult to improve serial search, it just slightly improves the system, but too slight to be noticed, even we set α=4 which is much larger than what we used in our VMF, T_{s4} is still impracticaly high.
- This tells us that our VMF is much more constructively sensitive to α than other systems.

In this figure, we change N_1, but the whole code length would keep the same, namely:

$n_1=5$, $n_2=7$, $N_1=127$, d=19, U=35, $E_{max}=9$

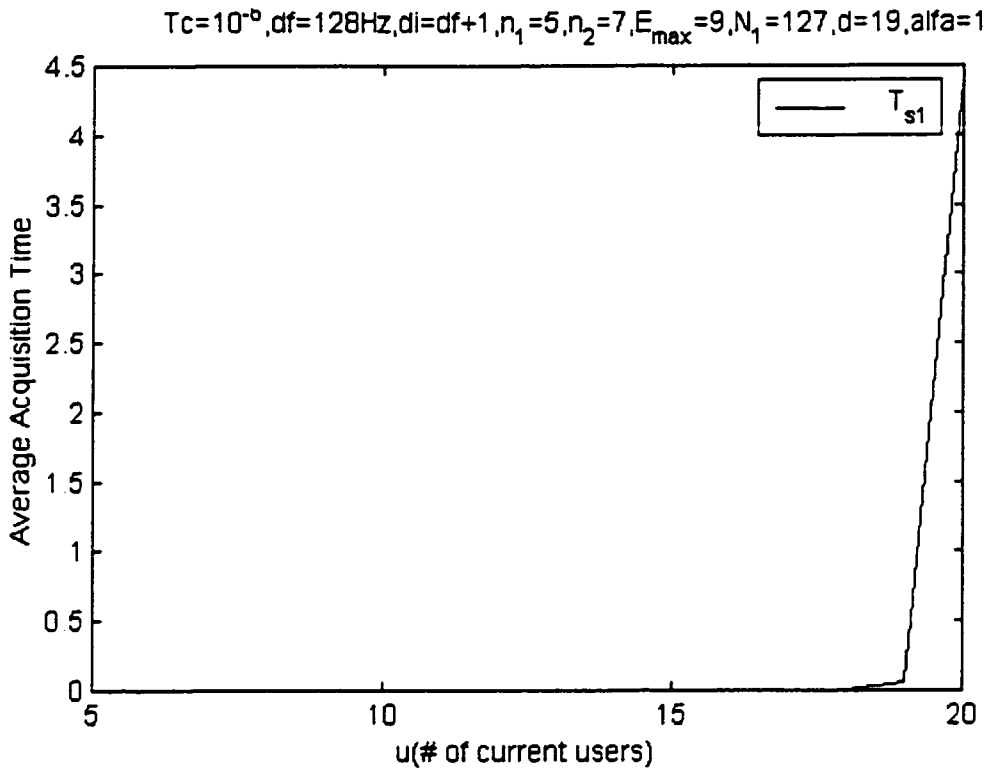


Fig. 6.12 T_{s1} under larger N_1

- In this group of parameters, we keep the whole signature code length as before, but we reassign the whole length among n_1, n_2 and N_1 values. Now we increase N_1 to 127, comparing this figure with Fig. 6.7, we can see that the acquisition time increased. The reason is that when N_1 increase, length of BCH code word become large, and accordingly the performance of decoding become worse.

In this figure, we change T_c and see its effect on our VMF.

n_1=5, n_2=8, N_1=63, d=9, U=35, E_{max}=3

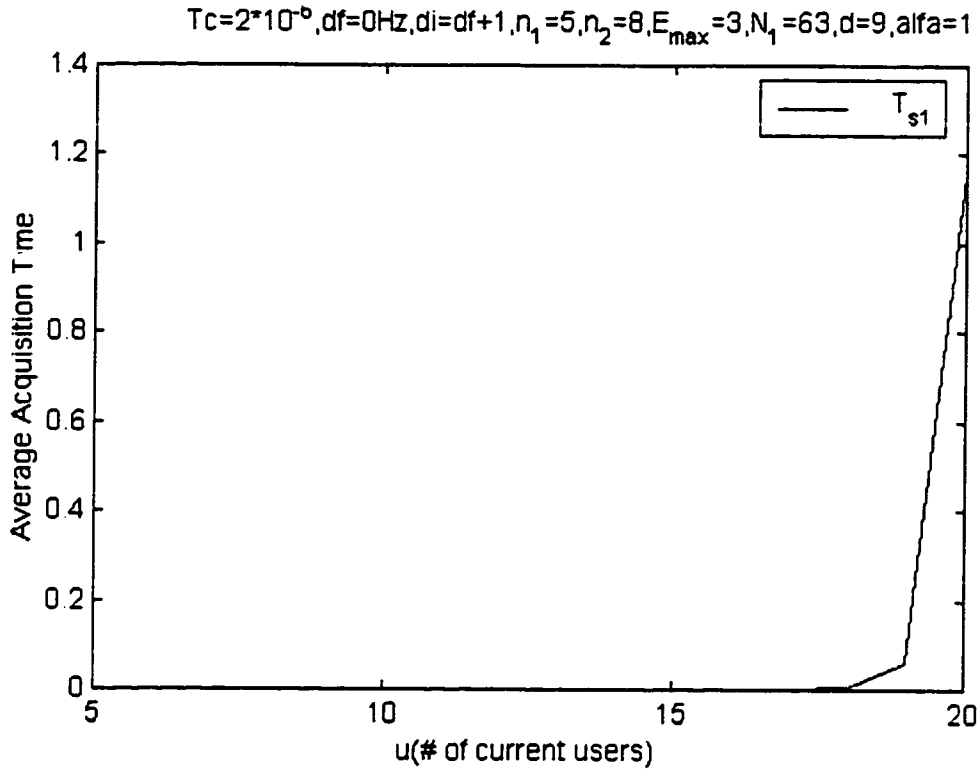


Fig. 6.13 Effect of T_c on VMF

- Here we double the chip duration of T_c to its effect on our VMF.
- Compare this figure with Fig. 6.2, it is clear that when T_c increases, the acquisition time also increase.

Chapter 7

Summary and Conclusions

This is the final chapter of the thesis, and here we recount the salient features of our VMF, based on these features, also based on much equation deduction and computation in the previous chapters, we could see the advantages of our VMF compared with other systems. Summary and conclusions are made in this chapter. Since this is a new scheme, there must be some shortcomings, suggestions for future research is also presented.

Summary

In chapter 1 of this thesis, we have given a short general introduction to Spread Spectrum System, namely, Direct Sequence System, Frequency Hopping System, Time Hopping System, Hybrid Spread Spectrum System and Chirp System.

In chapter 2, we gave a detailed review and discussion of the basic spreading code acquisition scheme, namely, serial search scheme and matched filter detection technique.

In chapter 3, a new hybrid acquisition scheme called virtual Matched Filter was presented based on the ideas and intellectual properties of Prof. A. K. Elhakeem. We gave a detailed description of the scheme and its various parameters. Certain salient aspects of the acquisition scheme are recounted here as they reflect the advantages of our VMF system.

- The basic constituents of the concatenated signature code are the two short gold codes which were encoded by the BCH FEC codes. The basic motivation of this structure is to decrease the impact of Doppler and carrier offsets on the length of acquisition time.
- The use of BCH word gives our system the ability to correct errors on the transmission channel, with this powerful tool, the performance of our system

was improved, i.e., the acquisition time was minimized compared to classical systems.

- The role of the M code epoch detection modules, namely, EDM is very important in our system. Each of these modules tries to find a legitimate sequence of G1, G2 short code by searching over its time window. As soon as one of the EDMs finds the epoch time, it leads to code acquisition trials. This is the parallelism concept employed in our system, the more the number of EDM, the shorter is the acquisition time.
- The back to back shift register used following the BCH decoder at the UR (user receiver) is another utility of the parallelism concept. This reduces the search time for the new sending code cell since this is achieved while the other LFSR is tracking the wrong code cell.
- Two strategies are given in BCH decoding technique, strategy 1 leads to more miss probabilities but less false alarm, while strategy 2 leads to more code acquisition false alarm. Under light user load, strategy 1 is better, but when user number increase, strategy 2 would become better.
- Another feature of our system is that it needs fast software processing within the best sequence selector. It is one of the requirements for our VMF system.

In chapter 4 and chapter 5, we gave the analysis of our system. The goal is to obtain the acquisition time and the performance. We begin with the detection of the short Gold Code, followed by evaluation of the performance of the BCH decoding error, then verification time of the long signature code is reached. Here are the main issues.

- In detection of the short Gold Code, DS/PSK is assumed. The intended user has Doppler offset and carrier phase which is the real case, and other users has a random phase and Doppler, this is also what happens in the real world. Both are counted in our scheme.
- We take advantage of the Gold Code three valued cross correlation property to calculate the probability of detection, under both signal presents and absence
- In the process of evaluation of the BCH word decoding error, we use the idea of erasure plus error correction where the maximum number of erased symbols (short Gold Code) is selected adaptively based on the current load. This idea is based on the consideration that the interference environment may be either light or heavy. This would give better results.
- Worth mention is that during the evaluation of the BCH decoding error, we apply the center limit theory and use Normalized Gaussian approximation instead of binomial. However, our program also considers correction terms due to this approximation (Appendix).

- The acquisition time is thus given based on the previous steps. For our VMF, the acquisition time is shorter in general.

Conclusions

In chapter 6, we compare our VMF with serial search, long matched filter and parallel matched filter. Results are shown in the figures under the various parameters. Right through these discussions, we could see the property of our VMF and the advantages compared with other systems. Here is the main conclusion.

- For our VMF, when processing time is counted, the acquisition time is longer. Actually this happens only at the beginning, it is just fill up time or warm up time, after that result would be much better.
- Under heavy load, strategy two is much better than strategy one. It means that if we choose to use strategy two in our VMF, our system could accommodate more users.
- When code length is long, serial search is almost failure, matched filter, especially parallel matched filter has a chance to play when Doppler is not heavy.

- Our VMF could survive even under high Doppler, this is the most salient feature which makes our system on the top of the others. In today's world, high speed in mobile communication causes much Doppler effect, and this fault tolerable system is just to meet today's requirements.
- Our VMF is more sensitive to Signal-to-Noise Ratio than other systems. This means even when the environment is worse, we still have the resource of Power, meanwhile, even this valuable way could not make effect to other systems.
- For our VMF, at the same total code length, BCH code word length has much effect on our system, the longer the BCH code word, the longer for the acquisition time.
- For our VMF, the code chip duration T_c has its effect. When the chip duration is increased, the acquisition time is also increased.

Suggestions for future work

The above is the summary and conclusions. There are other features which were not explained in detail in this thesis. Also there remains some questions which needs further study and investigation. Here is the suggestion for future work.

- More study is needed to optimize the adaptive threshold in our VMF, threshold is always an important part in all CDMA systems.

- Effects of best sampling time array of V matrix.
- More variety of combination of value of n_1, n_2 and N_1 for the same code length should be tried in order to get the optimum value.
- Get the optimizing point between acquisition time and the capacity of our system.
- Study of the analytical model of the acquisition scheme in a Rayleigh fading channel and other fading channels. In this thesis, we assume the channel is ideal, so it is important to study its feature in various channels.
- Security aspects of our concatenated shorter code should be studied.
- Bit error of our CDMA system under Doppler, higher loads, Reyleigh fading channel should be studied.
- Simulation studies of this VMF scheme, including channel generation, noise generation, computation and comparing with other systems under various environment, such as high speed (Doppler), larger multiple access interference, etc. Also we should develop and implement the acquisition scheme with dynamically varying parameter.

References

[1] A. Salmasi and K.S. Gilhousen, "On the system design aspects of code division multiple access (CDMA) applied to digital cellular and personal communications networks" Proc. IEEE Veh. Technol. Conf, vol. 41, pp. 57-62, May 1991.

[2] A.M. Viterbi and A.J. Viterbi, "Erlang capacity of a power controlled CDMA system," IEEE J. Selected Areas Commun., vol. 11, pp. 892-900, August 1993.

[3] K.S. Gilhousen, I.M. Jacobs, R. Padovani, A.J. Viterbi, and L. A. Weaver, "On the capacity of a cellular CDMA system," IEEE Trans, Veh. Technol., vol. 40,pp. 303-312, May 1991.

[4] Telecommunications Industry Association, Washington D.C., TIA/EIA/IS-95 Interim Standard, Mobile Station-Base Station Compatibility Standard for Dual Mode Wideband Spread Spectrum Cellular System, July 1993.

[5] G.R. Cooper, R. W. Nettleton, and D.P. Grybos, "Cellular land-mobile radio- why spread spectrum?", IEEE Commun, Mag., vol. 17, pp. 17-24, March 1979.

[6] A.J. Viterbi, A. M. Viterbi, and E. Zehavi, "Performance of power controlled wideband terrestrial digital communication," IEEE Trans, Commun., vol. 41, pp. 559-569, April 1993.

[7] J.W. Ketchum, "Down-link capacity of direct sequence CDMA for application in cellular systems," IEEE Symp. Spread Spectrum Tech, Appli., pp. 151-157, September 1990.

[8] Roger L.Peterson, Rodger E. Ziemer and Davud E. Borth, Introduction to Spread-Spectrum Communications, Prentice Hall Inc, 1995.

[9] M. K. Simon, J. K. Omura, R. A. Scholtz, abd B. K. Levitt, Spread Spectrum Communication Handbook. McGraw-Hill, Inc, 1994.

[10] W. R. Braun, " Performance Analysis for the Expanding Search PN Acquisition Algorithm," IEEE Trans. Commun., vol. COM-30, pp. 424-435, March 1982.

[11] R.B. Ward, "Acquisition of Pseudonoise Signals by Sequential Estimation," IEEE Trans, Commun. Technol., Vol. CT-13, pp.475-483, December, 1965.

[12] R.B. Ward and K.P. Yiu, "Acquisition of Pseudonoise Signals by Recursion -Aided Sequential Estimation," IEEE Trans. Commun., Vol COM-25, pp.784-794, August 1977.

[13] Michael B. Pursley, "The Derivation and Use of Side Information in Frequency Hop Spread Spectrum Communications.," IEICE Trans., Commun., vol. E76-B, No. 8, pp.814-823, August 1993.

[14] Michael Georgiopoulos, "Packet Error Probabilities in Frequency – Hopped Spread Spectrum Packet Radio Networks----Memoryless Frequency-Hopping Patterns Considered," IEEE Trans on Commun, vol. 36, No. 6, pp. 720-723, June, 1988.

[15] Manjunath V. Hegde and Wayne E. Stark, "On the Error Probability of Coded Frequency-Hopped Spread Spectrum Multiple-Access Systems," IEEE Trans on Commun., vol. 38, No. 5, pp. 571-573, May, 1990.

[16] Rui-Hua Dou and Laurence B. Milstein, "Erasure and Error-Correction Decoding Algorithm for Spread Spectrum Systems with Partial-Time Interference," IEEE Trans. On Commun., vol. COM-33, No. 8, pp.858-862, August 1985.

[17] William W. Wu and Robert Peile, "Coding for Satellite Communication", IEEE Journal on selected areas in commun., vol. SAC-5, No. 4, pp 724-748, May 1987.

[18] Wayne E. Stark, "Coding for Frequency-Hopped Spead-Spectrum Communication with Partial-Band, Interference----Part two: Coded Performance," IEEE Trans on Commun., vol. COM-33, No. 10, pp. 1045-1057, October 1985.

[19] Laurence B. Milstein and Sorin Davidovici and Donald L. Schilling, "Coding and Modulation Techniques for Frequency Hopped Spread Spectrum Communications Over a Pulse-Burst Jammed Rayleigh Fading

Channel," IEEE Journal on selected areas in commun. vol. SAC-3, No. 5, pp. 644-651, September 1985.

[20] Kebing Yang and Gordon L. Stuber, "Throughput Analysis of a Slotted Frequency-Hop Mutiple-Access Network," IEEE Journal on selected areas in commun., vol. 8, No. 4, pp. 588-602, May 1990.

[21] Stephen B. Wicker, Error Control System for Digital Communication and Storage, Prentice-Hall Inc., 1995.

[22] A.K. Elhakeem and V. Doradla, "Hybrid acquisition schemes for CDMA systems: A new perspective," Canadian Conference for Electrical and Computer Engineering, CCECE "97, May 1997.

[23] A.K. Elhakeem and V.Doradla, " A new gybrid acquisition scheme for CDMA systems employing short concatenated codes," 7th Virginia Tech/MPRG symposium on Wireless Personal Communications, June 11-13 1997.

[24] D.M. DiCarlo and C.L. Weber, "Multiple dwell serial search: Performance and applications to direct sequence code acquisition," IEEE Trans. Commun., vol. COM-31, No. 5, pp. 650-659. May 1983.

[25] E.Sourour and S.C. Gupta, "Direct sequence spread spectrum parallel acquisition in nonselective and frequence selective Rician fading channels," Ieee Journal on Selected areas in Communications, vol. 10, No. 3, pp. 535-544, April 1992.

[26] E. Sourour and S.C. Gupta, "Direct swquence spread spectrum parallel acquisition in a fading mobile channel," IEEE Trans. On Commun., vol. 38, No. 7, pp. 992-998, July 1990.

[27] V.Jovanovic, "Analysis of Strategies for Serial Search Spread-Spectrum Code Acquisition," IEEE Trans. Commun., vol. COM-36, pp. 1208-1220, August 1988.

[28] J. K. Holmes and C.C. Chen, "Acquisition Time Performance of PN Spread Spectrum Systems," IEEE Trans. Commun., vol. COM-25, pp. 778-784, August 1977.

[29] V. Jovanovic, " On the Distribution Function of the Spread-Spectrum Code Acquisition Time," IEEE J. Selected Areas Commun., vol. 10, pp. 760-769, May 1992.

[30] A. Weinberg," Generalized Analysis for the Evaluation of Search Strategy Effects on PN Acquisition Performance," IEEE Trans, Commun., vol. COM-31, pp. 37-49, January 1983.

[31] R. E. Ziemer and W. H. Tranter, Principles of communications, Second Edition, Houghton Mifflin, 1985.

[32] D.A. Shnidman, " Evaluation of the Q-Function," IEEE Trans. Commun., vol. COM-22, pp. 746-751, March 1974.

[33] W. Feller, An Introduction to Probability Theory and Its Applications, New York: Wiley, 1950.

[34] G. D.Forney, "On Decoing BCH Codes," IEEE Trans. Inf. Theory, IT-11, pp. 549-557, October 1965.

[35] J.L. Massey, " Shift-Register Synthesis and BCH Decoding," IEEE Trans. Inf. Theory, IT-15, pp. 122-127, January 1969.

Appendix

Here we include the Fortran code that was written for the purpose of computation the equations in the thesis.

```
$debug
program Acquisition Time


c parameters: n1=5,                    length of short code G1,
c             n2=7,                    length of short code G2,
c             Tc=1.e-6                 chip duration,
c             N1=127,d=19,Emax=9       BCH code word length & property,
c             u=1~35,                  number of current useser,
c             df=(10 values):0,2,4,8,16,32,64,128,256,512
c             q=2                      number of samples per Tc
c             M=6                      number of EDM
c             M'=6                     number of M.F. in M parallel search
```

```
c    'switch' is just a switch-on and off for system 1,2 and system 4,5,6.
     integer*4 n1,n2,n,bign1,smalld,smallq,m,mp,bigu,emax,e,u
     integer*4 nx,i1,m2,i2,i3,switch
     real*8 t1,t2,pi,tc,df,alfa,
    $              ti,pd1p,r1,pf1p,tfa1,ts1bar,
    $              pd2p,ts2bar,
    $              ti1,ts11bar,
    $              ti4,pd4p,r4,tfa4,pf4p,ts4bar,ts5bar,ts6bar,
    $              pd1s,pfa1,pd2s,pd3s,ci,sbar,sgms,nbar,sgmn,
    $             epss,pes,pss,epsn,pen,psn,pws,pwn
     real*8 ts6barf,ts5barf,ts4barf,ts2barf,ts1barf,q
     parameter(m2=5,nx=10,pi=3.141592654d0)
     data t1,t2/1.e-6,5.e-6/


     open(1,file='bp.dat')
     open(2,file='dp.dat')
     open(11,file='b.dat')
     open(13,file='d.dat')
    open(15,file='sbar.dat')
    open(17,file='sgms.dat')
    open(18,file='nbar.dat')
```

```
open(19,file='sgmn.dat')
open(20,file='ci.dat')
open(21,file='epss.dat')
open(22,file='pes.dat')
open(23,file='pss.dat')
open(24,file='pws.dat')
open(25,file='pd1s.dat')
open(26,file='pd1p.dat')
open(27,file='epsn.dat')
open(28,file='pen.dat')
open(29,file='psn.dat')
open(30,file='pwn.dat')
open(31,file='pf1p.dat')
open(34,file='pd4p.dat')
open(35,file='pf4p.dat')
open(32,file='ts1bar.dat')
open(33,file='ts2bar.dat')
open(36,file='ts11bar.dat')
open(40,file='ts4bar.dat')
open(41,file='ts5bar.dat')
open(42,file='ts6bar.dat')
```

```
      open(50,file='pd2p.dat')
      open(51,file='pfa1.dat')
      open(52,file='ti.dat')
      open(53,file='r1.dat')
      open(54,file='tfa1.dat')


smallq=2
m=6
mp=6
write(*,*)'n1=?'
read(*,*)n1
write(*,*)'n2=?'
read(*,*)n2
write(*,*)'N1=?'
read(*,*)bign1
write(*,*)'d=?'
read(*,*)smalld
write(*,*)'alpha=?'
read(*,*)alfa
write(*,*)'U=?'
```

```
read(*,*)bigu
write(*,*)'Emax=?'
read(*,*)emax
write(*,*)'Input data read OK. '

c this loop is related with "u"
    do i3=5,14
       u=i3
       e=nint(float(u)*(emax/bigu))
       if (e .eq. 0) e=1
c this loop is related with "Tc"
                do i1=0,0
                   tc = t1+float(i1)/float(nx)*(t2-t1)
c this loop is related with "df"
                                do i2=7,7
                                   df=2.0**i2

c This is to calculate ts1bar,ts2bar,ts11bar,
                                n=(2**n1-1)
                                switch=12
                                call value(n,i2,df,pi,tc,u,e,bign1,n2,smalld,alfa,
```

```
$                    pd1s,pfa1,pd2s,pd3s,ci,sbar,sgms,nbar,sgmn,
$                    epss,pes,pss,epsn,pen,psn,pws,pwn,switch)

                  ti=0.
                  pd1p =pd1s**n2
                  r1=(n+1-1.)*smallq/m
                  pf1p =pfa1**n2
                  tfa1=2.*r1*n2*m*tc
                  ts1bar=ts1barf(ti,pd1p,r1,pf1p,tfa1)

                  pd2p=(pd2s+pd3s)**n2

                  ts2bar=ts2barf(ti,tfa1,pd2p,r1)

                  ti1=r1*n2*m*tc/(8*smallq)
                  ts11bar=ts1barf(ti1,pd1p,r1,pf1p,tfa1)

c This is to calculate ts4bar,ts5bar
                  n=(2**n1-1)*bign1*n2
                  switch=456
                  call value(n,i2,df,pi,tc,u,e,bign1,n2,smalld,alfa,
```

```
$                    pd1s,pfa1,pd2s,pd3s,ci,sbar,sgms,nbar,sgmn,
$                    epss,pes,pss,epsn,pen,psn,pws,pwn,switch)

                  ti4=r1*bign1*n2*m*tc/smallq
                  r4=(n+1-1.)*bign1*(2**n2-1.)*smallq
                  tfa4=2.*r4*n2*m*tc
                  pd4p=q((ci-sbar)/sgms)
                  pf4p=q((ci-nbar)/sgmn)

                  ts4bar=ts4barf(ti4,tfa4,pd4p,r4,pf4p)
                  ts5bar=ts5barf(ti4,tc,smallq,tfa4,pd4p,r4,pf4p)

c This is to calculate ts6bar
                  n=(2**n1-1)*bign1*n2/mp
                  switch=456
                  call  value(n,i2,df,pi,tc,u,e,bign1,n2,smalld,alfa,
$                    pd1s,pfa1,pd2s,pd3s,ci,sbar,sgms,nbar,sgmn,
$                    epss,pes,pss,epsn,pen,psn,pws,pwn,switch)

                  ts6bar=ts6barf(ti4,tc,smallq,tfa4,pd4p,r4,pf4p,mp)
c this is to output ts1bar
```

```
                write(*,1002) u, ts1bar
                write(32,1002) u,ts1bar
c this is to output ts2bar
                write(*,1002) u, ts2bar
                write(33,1002) u,ts2bar
c this is to output ts11bar
                write(*,1002) u, ts11bar
                write(36,1002) u,ts11bar
c this is to output ts4bar
                write(*,1002) u, ts4bar
                write(40,1002) u,ts4bar
c this is to output ts5bar
                write(*,1002) u, ts5bar
                write(41,1002) u,ts5bar
c this is to output ts6bar
                write(*,1002) u, ts6bar
                write(42,1002) u,ts6bar
              enddo
            enddo
          enddo
            close(1)
```

close(2)

close(11)

close(13)

close(15)

close(17)

close(18)

close(19)

close(20)

close(21)

close(22)

close(23)

close(24)

close(25)

close(26)

close(27)

close(28)

close(29)

close(30)

close(31)

close(32)

close(33)

```
              close(34)
              close(35)
              close(36)
              close(37)
              close(40)
              close(41)
              close(42)

1002          format(1x,I3,5(e12.6))
              end

c************ Calculate Common Parameters ***************************
      subroutine value(n,i2,df,pi,tc,u,e,bign1,n2,smalld,alfa,
     $                 pd1s,pfa1,pd2s,pd3s,ci,sbar,sgms,nbar,sgmn,
     $                 epss,pes,pss,epsn,pen,psn,pws,pwn,switch)
      integer*4 n,i2,u,e,bign1,n2,smalld,switch
      real*8 df,di,pi,tc,lbar,lsqbar,b,bp,d,dp,sbar,sgms,nbar,sgmn,ci,
     $     epss,q,q1,q2,pes,pss,pws,pd1s,pd1p,pd2s,pd3s,epsn,
     $     pen,psn,pwn,pfa1,pwscf,alfa

              lbar=-1./n
```

```
      lsqbar=(n+1+1.)/float(n)**2


      if (i2.eq.0) then
         df=0
         b=1
         d=0
      else
         b=sin(2.*pi*float(n)*df*tc)/(2.*pi*float(n)*df*tc)
         d=(sin(pi*n*df*tc))**2/(pi*n*df*tc)
      end if


      di=df+1
      bp=sin(2.*pi*float(n)*di*tc)/(2.*pi*float(n)*di*tc)
      dp=(sin(pi*n*di*tc))**2/(pi*n*di*tc)


      sbar = alfa**2*b**2+alfa**2*d**2+(bp**2+dp**2)*lsqbar*(u-1)
      sgms = sqrt( (bp**2+dp**2)*(u-1)**2*lbar**2+(u-1)
     $     /8.*2*alfa**2*((bp*b)**2+(dp*d)**2+(bp*d)**2+(b*dp)**2)
     $     /n+(u-1)*(u-2)/8*(2*bp**4+2*dp**4+4*bp**2*dp**2)/n**2 )
      nbar =(bp**2+dp**2)*(u-1)*lsqbar
      sgmn =sgms
```

```
      ci =(sbar+nbar)*0.5
      if(switch .eq. 456) goto 1009

c calculate epss
      q1 =q((ci-sbar)/sgms)
      q2 =q((ci-nbar)/sgmn)
      epss =q1*q2+(1.-q1)*(1.-q2)

c calculate pes
      pes =q2*(1.-q1)
c calculate pss
      pss=q((sbar-nbar)/(2.*sgmn))
c based on "sbar,sgms, nbar, sgmn, ci", calculate "epsn,pen,psn",
      epsn =q2**2+(1.-q2)**2
      pen =q2*(1.-q2)
      psn =q((nbar-nbar)/(2.*sgmn))
      pws =pwscf(e,bign1,smalld,epss, pes,pss)
      pd1s =1.-pws
      pd1p =pd1s**n2
      pd2s=1.0-pws
      pd3s=0.5*pws*(bign1-2)/(bign1-1)
```

```
                pwn =pwscf(e,bign1,smalld,epsn, pen,psn)
                pfa1 =pwn*2./(bign1-1)

1008            format(1x,I3,5(e12.6))
1009            end
c *******Function Block of pwsc (Nomolized Gaussian Distribution)*********
                function pwscf(e,bign1,smalld,epss, pes,pss)
                real*8 pwscf,tol
                integer*4 e,bign1, smalld,j1,i, j2,i1,i2
                real*8 epss,pes, pss, s12, s2,  s345, s45
                real*8 mean1,variance1,z1,p1,sump1,
     $                mean2,variance2,z2,p2,sump2,
     $                mean3,variance3,z3,p3,sump3,
     $                mean4,variance4,z4,p4,sump4,
     $                mean5,variance5,z5,p5,sump5
                parameter(pi=3.141592654d0,tol=1.e-20)
c first and second summation loop:
                s12=0.0
                sump1=0.0
                do j1=0,e
                   s2=0.0
```

```
                sump2=0.0
                   do i=0,bign1-j1
                      mean2=(bign1-j1)*pes
                      variance2=sqrt((bign1-j1)*pes*(1.-pes))
                            if(variance2.lt.tol)then
                               p2 = 0.0
                            else
                               z2=(i-mean2)/variance2
                               p2=(1./sqrt(2.*pi))*exp(-z2**2/2.)/variance2
                            endif
                      sump2=sump2+p2
                            if( (2*i+j1).lt.smalld ) goto 7000
                            s2=s2+p2
7000            continue
                enddo

                mean1=bign1*epss
                variance1=sqrt(bign1*epss*(1.-epss))
                if(variance1.lt.tol)then
                  p1=0
                else
```

```
            z1=(j1-mean1)/variance1
            p1=1./sqrt(2.*pi)*exp(-z1**2/2.)/variance1
          endif
          sump1=sump1+p1
          if(sump2 .eq. 0) goto 9000
          s12=s12+p1*s2/sump2
        enddo
c third, forth and fifth summation loops:
        s345=0.0
        sump3=0.0
        do j2=e+1,bign1-1
          s45=0.0
          sump4=0.0
            do i1=0,bign1-j2
              mean4=(bign1-j2)*pes
              variance4=sqrt((bign1-j2)*pes*(1.-pes))
              if(variance4.lt.tol)then
                p4=0
              else
                z4=(i1-mean4)/variance4
                p4=1./sqrt(2.*pi)*exp(-z4**2/2.)/variance4
```

```
                endif


                sump4=sump4+p4
                sump5=0.0
                  do i2=0,j2-e
                    mean5=(j2-e)*pss
                    variance5=sqrt((j2-e)*pss*(1.-pss))
                    if(variance5.lt.tol)then
                         p5=0
                    else
                         z5=(i2-mean5)/variance5
                         p5=1./sqrt(2.*pi)*exp(-z5**2/2.)/variance5
                    endif
                    sump5=sump5+p5
                    if ((2*(i1+i2)+e).lt.smalld) goto 8000
                      s45=s45+p4*p5
8000                  continue
                  enddo
              enddo
              if(sump4 .eq. 0) goto 9000
              if(sump5 .eq. 0) goto 9000
```

```
                s45=s45/sump4/sump5


                mean3=bign1*epss
                variance3=sqrt(bign1*epss*(1.-epss))
                if(variance3.lt.tol)then
                   p3=0
                else
                   z3=(j2-mean3)/variance3
                   p3=1./sqrt(2.*pi)*exp(-z3**2/2.)/variance3
                endif


                sump3=sump3+p3
                s345=s345+p3*s45
             enddo
             if((sump1+sump3).eq.0) goto 9000
             pwscf=(s12+s345)/(sump1+sump3)
             goto 9001
9000         pwscf=0
9001         end
```

```
c****************** Function Block of Ts1bar**************************
                function ts1barf(ti,pd1p,r1,pf1p,tfa1)
                real*8 ts1barf
                real*8 ti,r1,tfa1,pd1p,pf1p
                ts1barf=(ti/pd1p)+(2.-pd1p)/(2*pd1p)*(r1-1.)*(ti+pf1p*tfa1)
                end

c ****************** Function Block of Ts2bar **************************
                function ts2barf(ti,tfa1,pd2p,r1)
                real*8 ts2barf
                real*8 ti,tfa1,pd2p,r1
                ts2barf=ti+tfa1*(1.-pd2p/r1)/(pd2p/r1)
                end

c ********************Function Block of Ts4bar ************************
                function ts4barf(ti4,tfa4,pd4p,r4,pf4p)
                real*8 ts4barf
                real*8 ti4,tfa4,pd4p,r4,pf4p
                ts4barf=(ti4/pd4p)+((2.-pd4p)/(2*pd4p))*(r4-1.)*(ti4+pf4p*tfa4)
                end
```

```
c ******************** Function Block of Ts5bar ************************
                function ts5barf(ti4,tc,smallq,tfa4,pd4p,r4,pf4p)
                real*8 ts5barf
                real*8 ti4,tc,tfa4,pd4p,r4,pf4p
                integer*4 smallq
                ts5barf=(tc/smallq)/pd4p+(2.-pd4p)/(2.*pd4p)*(r4-1.)*
     $                  (tc/smallq+pf4p*tfa4)+ti4
                end

c ********************** Function Block of Ts6bar **********************
                function ts6barf(ti4,tc,smallq,tfa4,pd4p,r4,pf4p,mp)
                real*8 ts6barf
                real*8 ti4,tc,tfa4,pd4p,r4,pf4p
                integer*4 smallq,mp
                ts6barf=(tc/smallq)/pd4p+(2.-pd4p)/(2.*pd4p)*(r4/mp-1.)*
     $                  (tc/smallq+pf4p*tfa4)+ti4/mp
                end

c************************ Q function block ****************************
                function q(x)
                real*8 q,x,erfcc
```

```
      q = 0.5*erfcc(x/1.4142135)
      end

c******************** erfcc (x) function block ************************
      FUNCTION erfcc(x)
      real*8 erfcc,x
      real*8 t,z
      z=abs(x)
      t=1./(1.+0.5*z)
      erfcc=t*exp(-z*z-1.26551223+t*(1.00002368+t*(.37409196+t*
      *(.09678418+t*(-.18628806+t*(.27886807+t*(-1.13520398+t*
      *(1.48851587+t*(-.82215223+t*.17087277)))))))))
      if (x.lt.0.) erfcc=2.-erfcc
      return
      END
```