A Sample-Decimation Based Fast Preamble Detection Algorithm

Haining Zhang

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 in Electrical Engineering

Concordia University

Montreal, Quebec, Canada

March, 2008

Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-40899-5
Our file Notre référence
ISBN: 978-0-494-40899-5

ABSTRACT

A Sample-Decimation Based Fast Preamble Detection Algorithm

Haining Zhang

Random access is a commonly used multiple access scheme that allows multiple users to share the same resource in a distributed fashion. In a Universal Mobile Telecommunication System (UMTS), the preamble of a random access channel (RACH) message is used by a mobile user to signal the base station for requesting network access or short data packets transportation. The base station is responsible in a timely fashion for detecting the preambles and informing the user whether the request has been granted or denied through the acquisition indication channel (AICH). Preamble detection is one of the most computationally intensive functional units of a base station. It has attracted many research attentions and investments in the past a few decades.

The drawback of the existing preamble detection (PD) algorithms for UMTS base-station is that either their computational complexity is high or the detection accuracy is low. The conventional full search PD algorithm gives the best result in terms of the detection probability, but its complexity is high. On the hand, the parallel-serial code phase detector PD algorithm provides a reduced computational complexity, but the detection accuracy becomes low. In this thesis, a sample-decimation based preamble detection technique is proposed in order to substantially reduce the computational complexity and at the same time retain a high detection accuracy. The proposed algorithm comprises two stages. Delay hypotheses or delay offsets which are unlikely to have a strong correlation power between the antenna samples and the locally generated preamble replica are

identified and discarded in the first stage. The second stage operates on the remaining offsets and employs all the antenna samples within the preamble signal.

Extensive computer simulations are conducted under different levels of additive white Gaussian noise interferences. The results show that the proposed algorithm has a detection performance very close to that of the conventional full search PD algorithm, while at the same time it reduces the computational complexity by more than sixty percent.

ACKNOWLEDGEMENTS

I wish to express my deep gratitude to my thesis supervisors, Dr. M. Omair Ahmad and Dr. M.N.S. Swamy, for their guidance throughout the years of my graduate studies at Concordia University. I would like to thank them for their advice, insight, and care. Pursuing a Mater's degree in Electrical and Computer Engineering was not only a long cherished dream of mine but also a great challenge as well. It would have been impossible for me to complete my graduate studies without the inspiration and support of my supervisors.

I am indebted to my uncle, Dr. Jiajun Zhang and his wife Dr. Li Zhang, who selflessly provided me with all their help and support, technical or otherwise, during the entire course of my graduate studies and my life in North America.

I would like to thank my cousin, Ms. Min Yang, my friends, Mr. Feng Wan and Mr. Marc-André Laverdière for their sincere advice, help and company during these years of study at Concordia University.

Lastly, and most importantly, I wish to thank my beloved parents, Guiling Xu and Jiafu Zhang. It is their unconditional love, sacrifices and understanding that have made all the things possible in my life. To them I dedicated this thesis.

To

my uncle and aunt

and

my dearest parents

Table of Contents

List of Figures

List of Tables

List of Symbols

P_s	signature s
$C_{sig,s}$	PRACH signature code for signature s
$S_{r\text{-}pre,n}$	nth PRACH preamble scrambling code
$S_{pre,n,s}$	PRACH preamble code for nth preamble scrambling code and signature s
i	sample index
j	chip index
$X(i)$	received antenna signal at sample i
$X_1(i)$	low-pass filtered antenna signal at sample i
$X_2(i)$	sample-decimated antenna signal at sample i
$Y_{n,s}(j)$	Locally generated code sequence at chip j
$Y_{n,s,1}(j)$	low-pass filtered code sequence at chip j
$Y_{n,s,2}(j)$	sample-decimated code sequence at chip j
W	search window size in samples
$\overline{p}$	average of all powers in the search window
M	non-coherent accumulation length
M_2	non-coherent accumulation length in the sample-decimated domain
N	Coherent accumulation period length
N_2	Coherent accumulation period length in the sample-decimated domain
o	delay offset in samples with respect to the search window origin
o_{org}	offset in the original domain
o_{dec}	offset in the sample-decimated domain
$P_{n,s}(o)$	Signal power at the delay offset o for scrambling code n and signature s
U	number of the finalists in the first stage
v	set of delay offsets to be evaluated in the second stage
α	scale factor
β	coefficient of detection threshold
θ	Detection threshold

List of Acronyms

2G	Second Generation
3G	Third Generation
3GPP	Third Generation Partnership Project
ASC	Access Service Class
ACK	Acknowledgement
AI	Acquisition Indication
AS	Access Slot
AWGN	Additive White Gaussian Noise
AICH	Acquisition Indication Channel
BCH	Broadcast Channel
CDMA	Code Division Multiple Access
EDGE	Enhanced data rates for GSM evolution
FD	Finger De-spreader
FSFC	Full Search with Full Contribution
FSPC	Full Search with Partial Contribution
GSM	Global System for Mobile
GPRS	General Packet Radio Service
ITU	International Telecommunication Union
MRC	Maximal Ratio Combining
MPS	Multi-path Search
OSF	Over Sampling Factor
PN	Pseudo-noise
PD	Preamble Detection
PDP	Power Delay Profile
P-CCPCH	Primary Common Control Physical Channel
P-SCH	Primary Synchronization Channel
RACH	Random Access Channel
PRACH	Physical Random Access Channel

PSFC	Partial Search with Full Contribution
PSPC	Partial Search with Partial Contribution
RRC	Radio Resource Controller
SR	Symbol Rate
SFN	System Frame Number
SNR	Signal-to-Noise Ratio
UMTS	Universal Mobile Telecommunication System
UE	User Equipment
WCDMA	Wideband Code Division Multiple Access

Chapter 1 Introduction

1.1 General

In the past two decades, mobile wireless communications have experienced a tremendous growth. From the first generation analog mobile wireless networks and phones introduced in the 1980s to the second generation (2G) digital mobile wireless communication systems, network capacity and user bandwidth have all grown substantially. The global system for mobile (GSM) communications [1] and the code division multiple access (CDMA) based IS-95 [2] are the most commonly used technologies in the 2G systems. To meet the increasing market demand for high bandwidth data services such as web browsing and internet TV, the third generation (3G) wireless telephone technologies have been developed [3]. Compared to the 2G systems, the 3G systems can support a large number of voice and data users at a higher data rate and a lower incremental cost. Moreover, the efficient spectrum utilization of the 3G wireless technologies makes it possible for the system to support a wide range of wireless multimedia services and to allow a global roaming among the different 3G wireless systems [4].

One of the most popular radio air interfaces is the wideband CDMA (WCDMA) [5]-[8]. The requirements for 3G mobile networks with IMT-2000 standard are defined by the International Telecommunication Union (ITU). The Third Generation Partnership Project (3GPP) has defined a mobile system that fulfills the IMT-2000 standard. This system is called the Universal Mobile Telecommunication System (UMTS) [9], [10]. The UMTS uses WCDMA as the underlying air interface.

Despite of the promising features of the 3G technologies, the initial deployment cost of the 3G systems has delayed a massive market production and deployment of the UMTS networks, and has challenged the academic community and the mobile industry to come up with more efficient and effective implementation techniques. In the mean time, the 2G systems have evolved into 2.5G by adapting themselves into enhanced data rates for GSM evolution (EDGE) and general packet radio service (GPRS) [11] techniques. Therefore, 3G mobile wireless communication remains a research attraction for researchers in the fields of telecommunications, semiconductor and digital signal processing.

In a UMTS system, the base station is known as node-B. The mobile user is referred to as user equipment (UE). The downlink or forward link is defined as the direction from node-B to UE, whereas the uplink direction or reverse link is defined as the direction from the UE to the node-B. A radio link between node-B and UE comprises both a forward link and a reverse link. These concepts are illustrated in Figure 1.1.

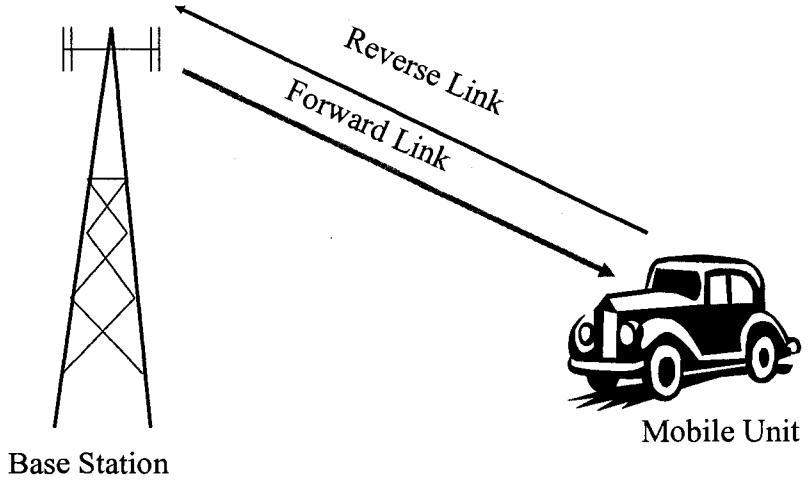

Figure 1.1 Illustration of a base station and a user equipment.

To establish a radio link between a node-B and a UE, the UE needs to follow the procedures defined in the 3GPP standard [13]. It sends a random access channel (RACH) message to the node-B to request a call. The node-B has a preamble detector (PD) that continuously monitors the RACH requests in the entire cell by detecting a preamble signal, which is sent immediately before the RACH message. Upon a positive detection, the node-B grants the request by sending an acknowledgement (ACK) message through the downlink acquisition indication (AI) channel. Upon receiving the ACK message, the UE proceeds with the RACH message transmission and radio link setup. The PD provides an estimate of the location of the UE in the cell. The multi-path searcher takes the estimate of the location of the UE as the initial position for the search window, and continuously finds and updates the locations of the fingers during the lifespan of the radio

link connection. The finger de-spreader (FD), on the other hand, is responsible for descrambling and de-spreading the fingers, and then combining them to provide the resultant symbols for further symbol rate (SR) processing. The widely-used technique for finger combining is called maximal ratio combining (MRC) [14]. The symbol rate processing is responsible for decoding the channel coding and producing the final binary values of the information bits.

In order to inform the users within a given cell the system information, the base station sends out the access service class (ASC) through the broadcast channel (BCH), which is a downlink common channel. The ASC specifies which signatures and access slots in the cell are available for the UE to use. There could be as many as 16 signatures in an ASC. From the node-B perspective, the preamble detector needs to check all the signature in the ASC at every access slot permitted by the ASC. In addition, since the UE could be anywhere in the cell, the PD needs to search the entire cell to synchronize with the UE scrambling sequence, leading to a very large search window. Further, the PD needs to compensate for the phase rotation due to the Doppler effect and carrier frequency shift. Therefore, it is well-known that the preamble detection is the most computationally intensive unit in the base station.

Developing a fast preamble detection technique that requires a small amount of correlation computation and at the same time retains the detection accuracy of the full search algorithm is a challenging problem, which is of significant importance to the telecommunication industry.

1.2 Literature Survey

Ever since the introduction of spread spectrum concepts for communications [12]-[16] over thirty years ago, Pseudo-noise (PN) code synchronization has been the focus of attention for many researchers [17]-[33]. The serial search is the most widely used approach for PN code synchronization [28]-[33]. The simplest serial search is the fixed dwell time method [17]. With this method, each of the possible PN code alignments, also known as delay offsets or delay hypotheses, is examined for a fixed period of time in a serial fashion. This fixed period of time is called correlation length. This method is called single dwell serial search, also referred to as full search with full contribution in this thesis. This method is easy to implement in the sense that it employs a fixed correlation length and exhaustively computes/examines all possible delay offsets.

However, this single dwell method is computationally intensive. In an attempt to reduce the computational complexity, multi-dwell serial search has been proposed [31], [34]. In essence, both the preamble detection (PD) and multi-path search (MPS) used in UMTS are PN code synchronization techniques. The problem with the preamble detection, however, is that the detection has to be completed within a fixed time period and has to be based on a burst of samples. That is why preamble detection is also called burst synchronization. While multi-dwell search is applicable to applications such as satellite communication synchronizations and multi-path search in UMTS, it is not very suitable for the preamble detection used in UMTS. To the best of the investigator's knowledge, no application of the multi-dwell search for preamble detection has been reported in the literature.

Recently, Sheen *et al.* proposed a parallel-serial code phase detector [35] suitable for UMTS WCDMA base station. This method is similar to the single dwell search in that the period of the time used for calculating the correlation is also fixed. However, it differs from the single dwell serial search in that not all the antenna samples are used for the detection. Specifically, the parallel-serial code phase detector can be configured to use part of the 4096 chips of preamble signal for the correlation. If we call the single dwell serial search using all the 4096 chips of the preamble signal as full-search with full contribution (FSFC), then we can call Sheen's algorithm as full-search with partial contribution (FSPC). By controlling the correlation length of the FSPC algorithm, one can make trade-offs between complexity and detection performance. Thus, the detection accuracy of this method could be expected to become close to that of the single dwell method (FSFC), while at the same time the computational complexity very close to that of the single dwell method.

So far, no fast acquisition method has been reported in the literature that has detection accuracy close to that of the single dwell method yet with substantially less computational complexity.

1.3 Motivation, Scope and Organization of the Thesis

Efficient random access is one of the key designs in WCDMA cellular systems. Detection of the preamble of the RACH message is the sole functionality of the base station. Fast burst synchronization is essential in random access in order to avoid excessive access delay and frequent retransmissions that may reduce the overall system capacity. Compared to the multi-path searcher and finger de-spreading, the preamble detection is

computationally the most intensive unit in a node-B. Thus, the effectiveness of implementing the PD directly affects the cost and performance of a node-B system. Developing new and effective PD algorithms remains a challenging research problem and any possible innovations regarding PD is probably a well-guarded industry secret [36], [37]. Therefore, developing an efficient PD algorithm for the base station is essential for a cost effective 3G UMTS network.

Until now, the most popular PD search is the single dwell serial search. This can provide the best detection accuracy, since all possible alignments are exhaustively searched with the contributions from all the received antenna samples. With the high speed correlation accelerators from Texas Instruments [38], [39], complex valued sample by sample correlation can be performed with a speed of 2048 times the chip rate. In other words, such a device has an equivalent of 2048 correlators running in parallel at the chip rate speed. However, even such a powerful chip can only support fewer than 64 mobile users in practice. The demand for an effective PD search algorithm still remains very high.

The multiple dwell search method has undoubtedly reduced the computational complexity of the single dwell search method. But it is not suitable for PD applications. The parallel-serial phase detector based algorithm can help to reduce the computational complexity, but its detection accuracy becomes very low while the savings in the computational complexity remain moderate to high.

The work of this thesis is an attempt to develop a technique for preamble detection that would reduce the computational complexity and at the same time provide a high detection accuracy. It is known from the 3GPP standard that the antenna streams are pre-

processed by a pulse shaping filter before transmission [40]. Such a filtering, along with channel fading and other interferences, actually increases the correlation among adjacent antenna samples. That is, the information contained in the adjacent antenna samples are more inter-related. The proposed sample-decimation based PD algorithm exploits the feature of the sample correlations and provides fast detection speed by performing the initial search only on the sample-decimated antenna stream and the corresponding locally generated preamble signal replica. Generally speaking, a preamble detection algorithm consists of calculating the power delay profile and detecting the presence of a true preamble signal based on the delay profile. In this research, however, we focus on calculating the power delay profile that constitutes the major part of the preamble detection. It would be shown that the proposed algorithm yields almost the same power delay profile as that obtained by using the single dwell method while the overall computational load is substantially reduced.

This thesis is organized as follows: The basic concepts of preamble signal, correlation, coherent accumulation, non-coherent averaging, search window, and power delay profile are reviewed in Chapter 2. In Chapter 3, a sample-decimation based algorithm, with an objective of achieving a high detection accuracy and low computational complexity, is developed. In Chapter 4, a simulation study of the proposed algorithm in comparison with the single dwell method and the parallel-serial phase detector method is carried out. Chapter 5 concludes the thesis by highlighting the contributions of this study and by providing some suggestions for future investigation.

Chapter 2 The Fundamentals of a Preamble Detector

There are many good references available concerning the fundamentals of WCDMA and UMTS systems [5], [6], [10]. In this chapter, we focus our attention on the background knowledge of the working principles of random access process employed in the UMTS mobile wireless network [41].

2.1 Uplink Radio Frame Structure

At the base station, the radio frequency signal is first demodulated from the carrier frequency signal into a baseband signal. Then this signal is sampled and digitized. This digital sequence will be loosely referred to as antenna signal in subsequent discussions. The signal received at the base station in essence is a delayed version of the transmitted signal by the mobile.

To facilitate the subsequent discussions, let's first describe the structure of the antenna signal. An antenna signal (sequence) consists of many radio frames. A radio frame is

10 ms long, which is divided into 15 slots. Each slot is further divided into 2560 chips. Thus, the duration of one chip period is 0.26 μs. If the over sampling factor is OSF=2, then each antenna sample is 0.13 μs in duration. The frame structure is illustrated in Figure 2.1.

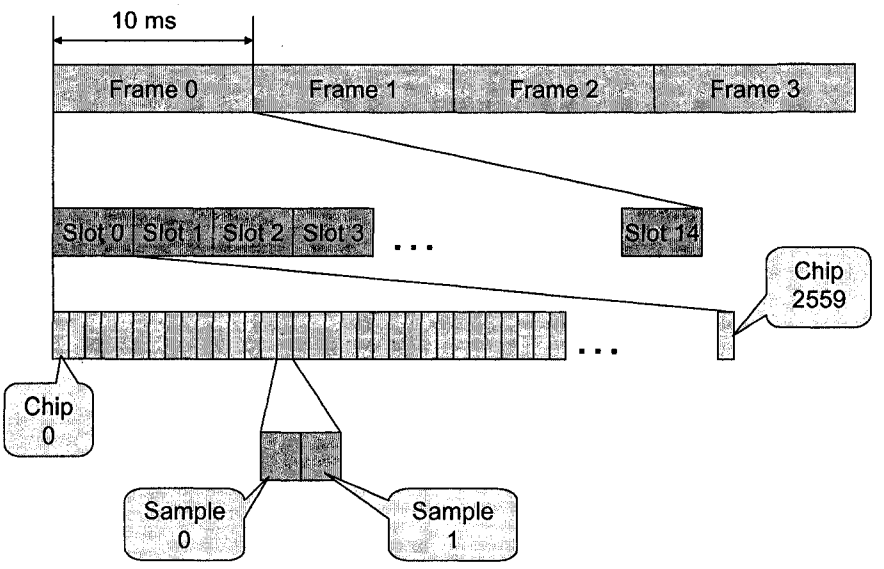

Figure 2.1 Frame structure in UMTS.

An access slot (AS) is equal to two system slots, thus having 5120 chips. In other words, two radio frames compose one access frame which comprises 15 access slots. Since each frame has 15 slots, which is an odd number, access slot 8 straddles over the frame boundary as expected. The concepts of access slot and access frame are illustrated in Figure 2.2.

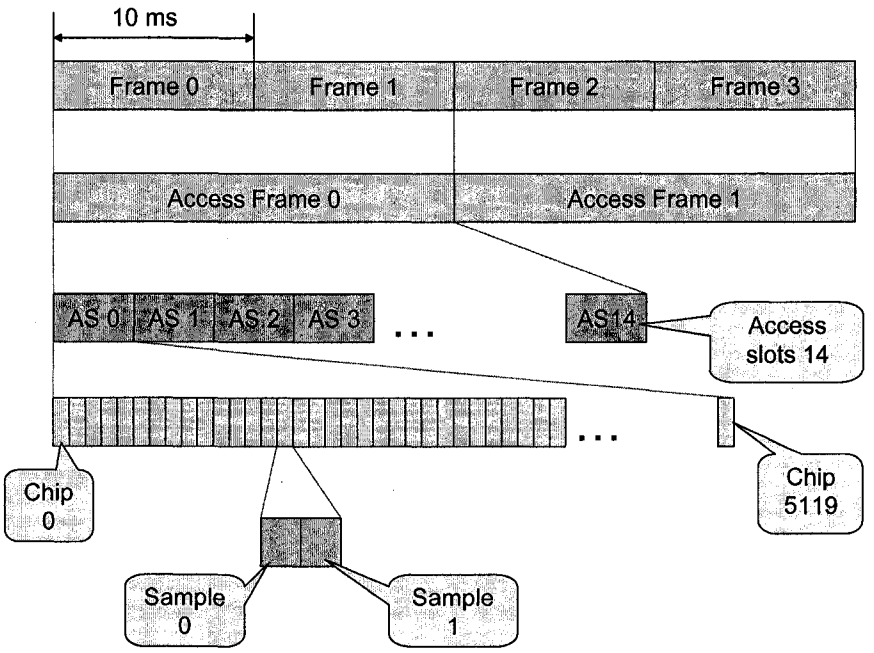

Figure 2.2 The structures of an access slot and an access frame.

2.2 Downlink Access Slots and AICH Channel

The primary common control physical channel (P-CCPCH) serves as the time reference for all the uplink and downlink physical channels and signals. Some channels are aligned with P-CCPCH such as the acquisition indication channel (AICH), and the primary synchronization channel (P-SCH). Figure 2.3 shows the timing relationships among the various physical channels and signals according to [42]. It can be seen that for every two radio frames with system frame number (SFN) modulo 2 = 0 and 1, there are 15 AICH access slots. Access slot #0 corresponds to the frame boundary with SFN modulo 2 = 0.

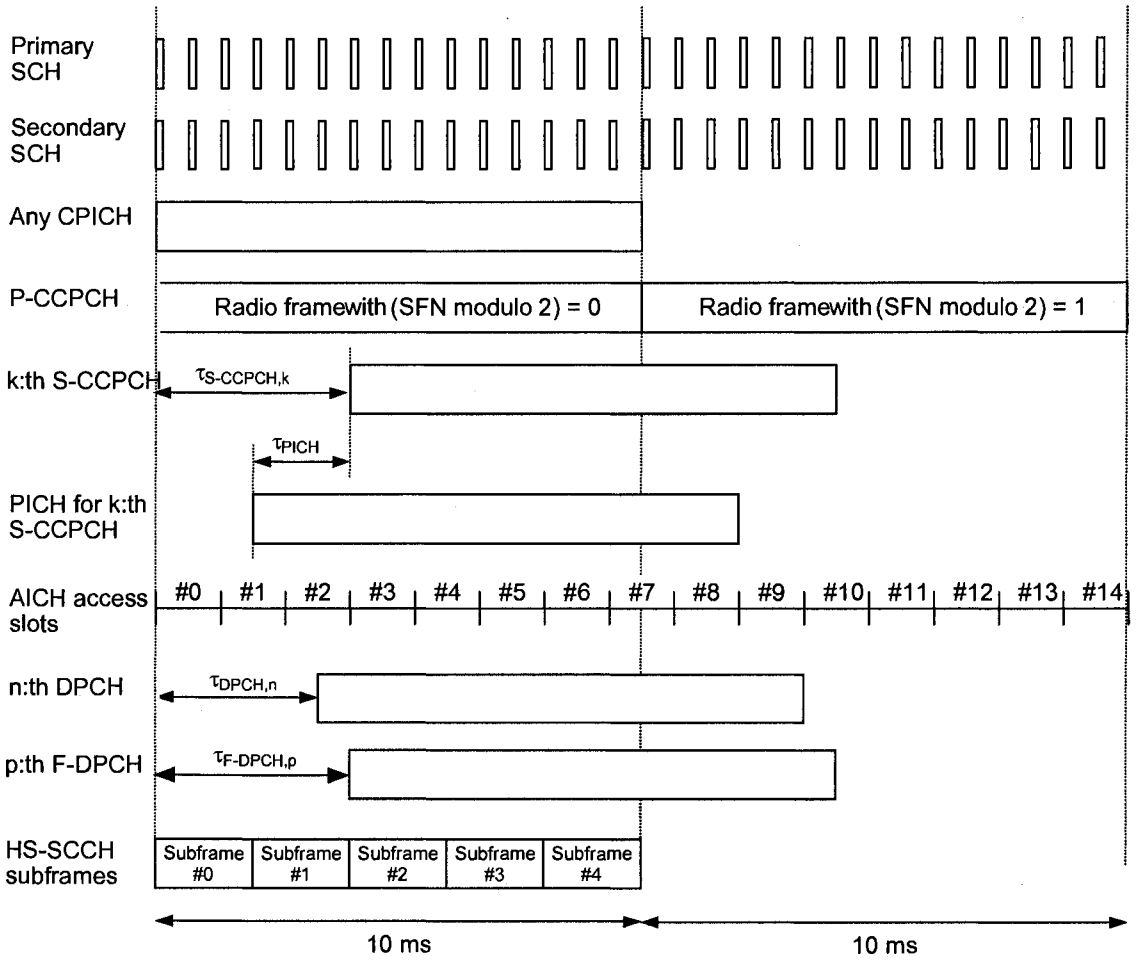

Figure 2.3 Access slot timing of downlink physical channels.

2.3 Time Relationship between AICH and PRACH

The physical random access channel (PRACH) is an uplink shared physical channel. It carries the preamble signal and the RACH message. The timing relationship between AICH and PRACH is shown in Figure 2.4. As can be seen, the preamble signals are sent

in the PRACH access slots which have fixed time offset with respect to the corresponding AICH access slots. For a given AICH access slot, the corresponding PRACH access slot is ahead in time by a period $\tau_{p\text{-}a}$. In other words, after sending a preamble signal at a given PRACH access slot, the UE would expect a response from the node-B $\tau_{p\text{-}a}$ time units later. When node-B sets the configuration parameter AICH_Transmission_Timing equal to 0, $\tau_{p\text{-}a}$ = 7680 chips. When node-B sets the configuration parameter AICH_Transmission_Timing equal to 1, $\tau_{p\text{-}a}$ = 12800 chips.

The RACH message itself is typically 10 ms or 20 ms long depending on the upper layer configuration control messages. Here upper layer refers to MAC layer that is above the physical layer where the preamble detection takes place.

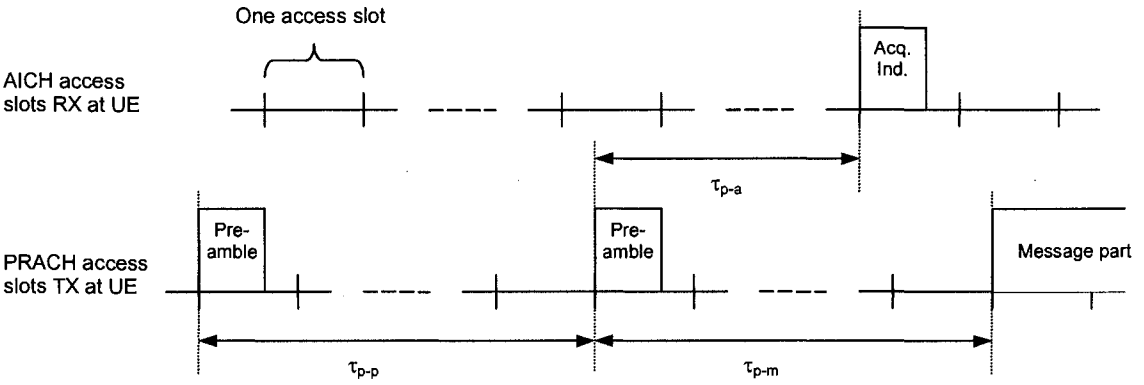

Figure 2.4 Timing relationship between AICH and PRACH.

2.4 RACH Sub-channel

A RACH sub-channel is a set of uplink access slots, i.e., the PRACH access slots. There are 12 RACH sub-channels. Any one of the uplink access slots must belong to one of the 12 sub-channels. Every two system (radio) frames comprise 15 access slots. Thus, eight system frames contain 60 access slots, giving each sub-channel 5 access slots within every 8 system frames. In other words, eight system frames is the period for resource partitioning of the uplink access slots. It can be seen from Table 2.1 that sub-channel 0 contains access slots 0, 12, 9, 6, and 3.

Table 2.1 Available access slots for the sub-channels

P-CCPCH SFN mod 8	Sub-channel number											
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
0	0	1	2	3	4	5	6	7				
1									8	9	10	11
	12	13	14									
2				0	1	2	3	4	5	6	7	
3												8
	9	10	11	12	13	14						
4							0	1	2	3	4	5
	6	7										
5			8	9	10	11	12	13	14			
6										0	1	2
	3	4	5	6	7							
7						8	9	10	11	12	13	14

The relationship between the access slots and the sub-channels is also illustrated in Figure 2.5. The access slots are numbered from 0 through 14 within every two system frames. The sub-channels are numbered from 0 through 11. As seen from this figure, eight system frames form a complete period. The access slots belonging to sub-channel 0 are marked as gray in the figure. This corresponds to the second column of Table 2.1.

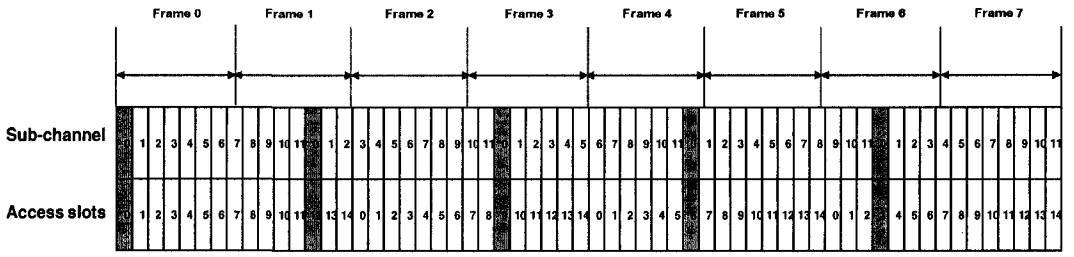

Figure 2.5 Relationship between access slots and sub-channels.

2.5 Preamble Signatures

A signature is a 16-bit code. There are a total of 16 signatures in UMTS. The preamble signature corresponding to a signature s consists of 256 repetitions of a signature of length 16, $P_s(n)$, n=0…15. This is defined as follows:

$$C_{sig,s}(i) = P_s(i \text{ modulo } 16),\ i = 0, 1, \ldots, 4095 \qquad (2.1)$$

where $C_{sig,s}(i)$ is the value of the preamble signature at chip i corresponding to the signature $P_s(.)$. The signature $P_s(.)$ is from the set of 16 Hadamard codes of length 16. These are listed in Table 2.2. It can be seen that signature 0 has all 1's in it.

Table 2.2 Signatures

Preamble signature	Value of n															
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
$P_0(n)$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
$P_1(n)$	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1
$P_2(n)$	1	1	-1	-1	1	1	-1	-1	1	1	-1	-1	1	1	-1	-1
$P_3(n)$	1	-1	-1	1	1	-1	-1	1	1	-1	-1	1	1	-1	-1	1
$P_4(n)$	1	1	1	1	-1	-1	-1	-1	1	1	1	1	-1	-1	-1	-1
$P_5(n)$	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1
$P_6(n)$	1	1	-1	-1	-1	-1	1	1	1	1	-1	-1	-1	-1	1	1
$P_7(n)$	1	-1	-1	1	-1	1	1	-1	1	-1	-1	1	-1	1	1	-1
$P_8(n)$	1	1	1	1	1	1	1	1	-1	-1	-1	-1	-1	-1	-1	-1
$P_9(n)$	1	-1	1	-1	1	-1	1	-1	-1	1	-1	1	-1	1	-1	1
$P_{10}(n)$	1	1	-1	-1	1	1	-1	-1	-1	-1	1	1	-1	-1	1	1
$P_{11}(n)$	1	-1	-1	1	1	-1	-1	1	-1	1	1	-1	-1	1	1	-1
$P_{12}(n)$	1	1	1	1	-1	-1	-1	-1	-1	-1	-1	-1	1	1	1	1
$P_{13}(n)$	1	-1	1	-1	-1	1	-1	1	-1	1	-1	1	1	-1	1	-1
$P_{14}(n)$	1	1	-1	-1	-1	-1	1	1	-1	-1	1	1	1	1	-1	-1
$P_{15}(n)$	1	-1	-1	1	-1	1	1	-1	-1	1	1	-1	1	-1	-1	1

2.6 Access Service Class

In a UMTS network, mobile users are managed through access service class (ASC). Each

user is associated with an ASC which specifies the set of signatures and the set of sub-channels available for this class. Figure 2.6 shows an example of ASC in which ASC0 has signature 1 in it and has all the sub-channels. On the hand, the ASC1 has signatures 4 and 5 and sub-channels 2 and 3 in it.

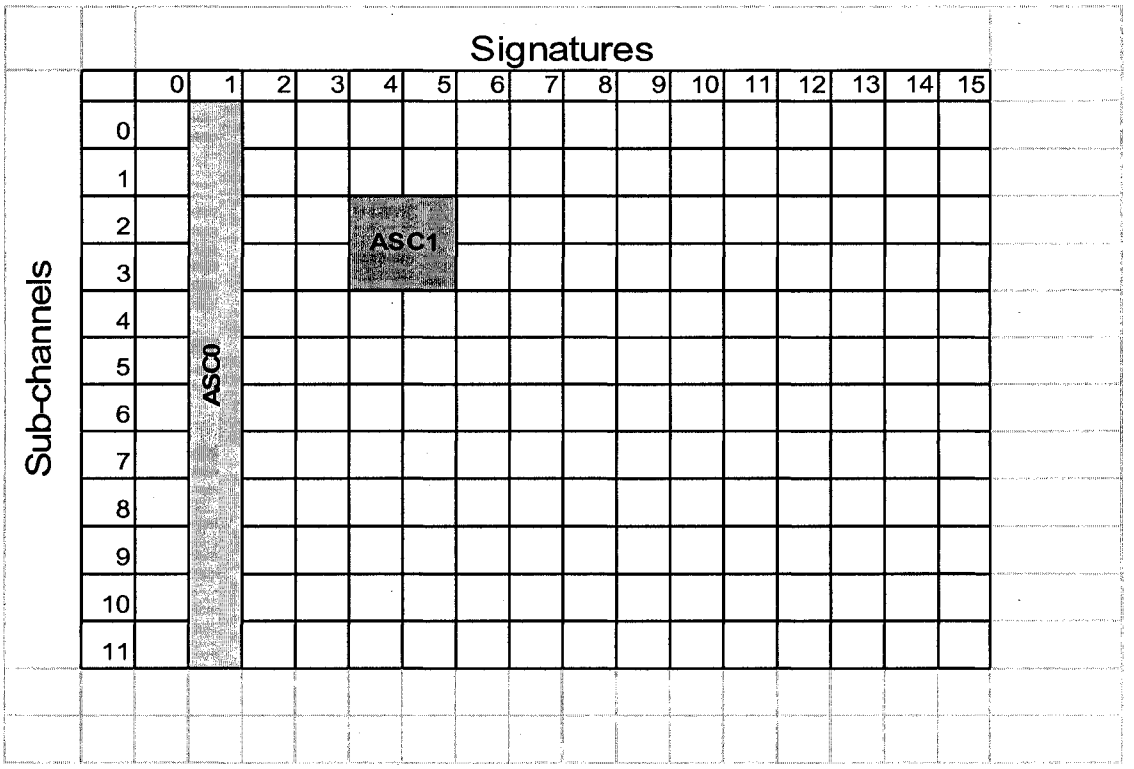

Figure 2.6 Access service class.

2.7 Random Access Procedure in UMTS

Reference [41] describes in detail the complete random access procedure. For the sake of completeness; however, a simplified version of the procedure is summarized below.

Before the physical random-access procedure is initiated, the physical layer, also known as Layer 1, receives the following information from the higher layer which is the radio resource controller (RRC):

The preamble scrambling code.

The message length in time, either 10 ms or 20 ms.

The AICH_Transmission_Timing parameter, 0 or 1.

The sets of available signatures and RACH sub-channels for each ASC.

The power-ramping factor, Power Ramp Step.

The parameter Preamble Retrans_Max.

The initial preamble power, Preamble_Initial_Power.

The physical random-access procedure is performed as follows:

1 Derive the available uplink access slots in the next full access slot set for the set of available RACH sub-channels within the given ASC. Randomly select one access slot among the ones previously determined. If there is no access slot available in the selected set, randomly select one uplink access slot corresponding to the set of available RACH sub-channels within the given ASC from the next access slot set.

2 Randomly select a signature from the set of available signatures within the given ASC.

3 Set the Preamble Retransmission Counter to Preamble Retrans_Max.

4 Transmit a preamble using the selected uplink access slot, signature and preamble transmission power.

5 If no positive or negative acquisition indicator (AI $\neq$ +1, –1) corresponding to the selected signature is detected in the downlink access slot corresponding to the selected uplink access slot:

(a) Select the next available access slot in the set of available RACH sub-channels within the given ASC.

(b) Randomly select a new signature from the set of available signatures within the given ASC.

(c) Increase the Commanded Preamble Power by ΔP_0 = Power Ramp Step in dB. If the Commanded Preamble Power exceeds the maximum allowed power by 6 dB, the UE may pass L1 status ("No ack on AICH") to the higher layers (MAC) and exit the physical random access procedure.

(d) Decrease the Preamble Retransmission Counter by one.

(e) If the Preamble Retransmission Counter > 0 then repeat from step 5. Otherwise pass L1 status ("No ack on AICH") to the higher layers (MAC) and exit the physical random access procedure.

6 If a negative acquisition indicator corresponding to the selected signature is detected in the downlink access slot corresponding to the selected uplink access slot, pass L1 status ("Nack on AICH received") to the higher layers (MAC) and exit the physical random access procedure.

7 If positive acquisition indicator is detected, transmit the random access message three or four uplink access slots depending on the AICH transmission timing parameter after the uplink access slot of the last transmitted preamble. Pass L1 status "RACH message transmitted" to the higher layers and exit the physical random access procedure.

Figure 2.7 shows an example of the RACH procedure described above. The UE first sends a preamble at access slot 0 with the minimal transmission power and a randomly selected signature from the given ASC, and expects a response from the node-B at AICH access slot 0. The UE sends another preamble with increased power at PRACH access slot 3 with another randomly selected signature from the ASC, since no response (ACK or NACK) is detected at AICH access slot 0. This retransmission is repeated at PRACH access slots 6 and 9, since no response from node-B is detected at AICH access slots 3 and 6. Finally, at AICH access slot 9, an ACK (AI=1) is detected by the UE. Thus, UE starts to transmit RACH message at PRACH access slot 12.

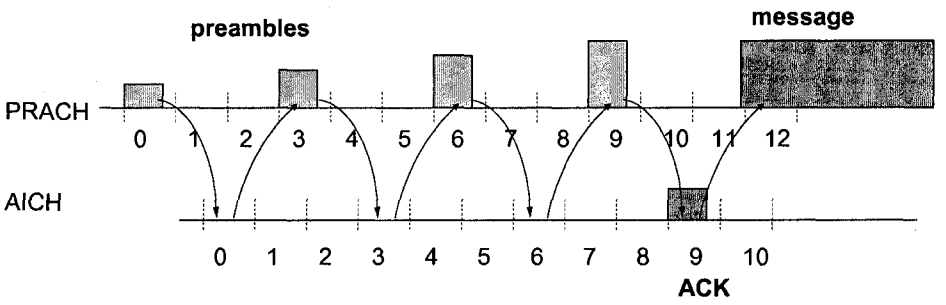

Figure 2.7 Power ramping scheme in RACH procedure.

2.8 Preamble Signals

The preamble signals sent by a UE for a given scrambling code and signature are generated using the formulas given in the following sub-sections [43].

2.8.1 Long scrambling sequence

The long scrambling sequences $c_{long,1,n}$ is constructed from the position wise modulo 2 sum of 38400-chip segments of two binary m-sequences generated by means of two generator polynomials of degree 25. Let x and y be the two m-sequences respectively. The x sequence is constructed using the primitive polynomial $X^{25}+X^3+1$. The y sequence is constructed using the polynomial $X^{25}+X^3+X^2+X+1$. The resulting sequences thus constitute segments of a set of Gold sequences [43].

Let $n_{23} \ldots n_0$ be the 24-bit binary representation of the scrambling sequence number n with n_0 being the least significant bit. The x sequence depends on the chosen scrambling

sequence number n and is denoted x_n, in the sequel. Furthermore, let $x_n(i)$ and y(i) denote the i th symbol of the sequence x_n and y, respectively.

The m-sequences x_n and y are constructed as follows:

Initial conditions:

$$x_n(0)=n_0\ ,x_n(1)=n_1\ ,\ \ldots\ =x_n(22)=n_{22}\ ,x_n(23)=n_{23},\ x_n(24)=1 \tag{2.2}$$

$$y(0)=y(1)=\ldots=y(23)=y(24)=1 \tag{2.3}$$

The recursive relations for the symbols are given by

$$x_n(i+25)=x_n(i+3)+x_n(i)\ \text{modulo 2},\ i=0,\ldots,\ 2^{25}-27 \tag{2.4}$$

$$y(i+25)=y(i+3)+y(i+2)+y(i+1)+y(i)\ \ \text{modulo 2},\ i=0,\ldots,\ 2^{25}-27 \tag{2.5}$$

Define the binary Gold sequence z_n by:

$$z_n(i)=x_n(i)+y(i)\ \text{modulo 2},\ \ i=0,\ 1,\ 2,\ \ldots,\ 2^{25}-2 \tag{2.6}$$

The real valued Gold sequence Z_n is defined by

$$Z_n(i)=\begin{cases}+1 & \text{if } z_n(i)=0\\ -1 & \text{if } z_n(i)=1\end{cases}\quad \text{for } i=0,1,\ldots,2^{25}-2 \tag{2.7}$$

Finally, the real-valued long scrambling sequences $c_{long,1,n}$ is defined as follows:

$$c_{long,1,n}(i)=Z_n(i),\ \ i=0,\ 1,\ 2,\ \ldots,\ 2^{25}-2 \tag{2.8}$$

2.8.2 Preamble scrambling code

The scrambling code for the PRACH preamble part is constructed from the long scrambling sequences. In total, there are 8192 PRACH preamble scrambling codes. The nth preamble scrambling code, n = 0, 1, ..., 8191, is defined as

$$S_{r\text{-}pre,n}(i) = c_{long,1,n}(i),\ i = 0,\ 1,\ \ldots,\ 4095 \qquad (2.9)$$

It should be noted here that the preamble scrambling code is only 4096-chip long even though the long scrambling sequence $c_{long,1,n}(.)$ is $(2^{25}-1)$-chip long.

2.8.3 Preamble signals

The random access preamble signal or code, $C_{pre,n}$, is a complex valued sequence. It is built from a preamble scrambling code $S_{r\text{-}pre,n}$ and a preamble signature $C_{sig,s}$ as follows:

$$C_{pre,n,s}(k) = S_{r\text{-}pre,n}(k) \times C_{sig,s}(k) \times e^{j(\frac{\pi}{4}+\frac{\pi}{2}k)},\ k = 0,\ 1,\ 2,\ 3,\ \ldots,\ 4095 \qquad (2.10)$$

where k=0 corresponds to the chip transmitted first in time and $S_{r\text{-}pre,n}$ and $C_{sig,s}$ are defined in (2.9) and (2.1), respectively.

2.9 Preamble Detection at Node-B

The objective of a preamble detector is to find the presence of a valid preamble signal in the antenna stream. The detector is responsible for checking all the access slots that are

broadcasted to the users in their ASC. For each permissible access slot, all the signatures permitted should be checked.

2.9.1 Correlation

The principle of the preamble detection is built on the assumption that the antenna stream of the node-B contains a time delayed version of the transmitted signal at the UE with added transmission channel impairments including fading and additive white Gaussian noise. If the receiver at the node-B could locally generate a replica of the transmitted signal by the UE, then the correlation between the received antenna stream and the complex conjugate of the replica will give a strong autocorrelation power, signifying the presence of an expected preamble signal. If the expected signature is not in the preamble signal or the alignment between the replica and the received signal is not correct, the correlation will give a weak power.

Mathematically, the correlation operation can be expressed as follows. The received antenna signal is denoted as $X(i) = I_X(i) + Q_X(i)$, where the index i is the sample position. The over-sampling factor, OSF=2, gives a total of 8192 samples for a complete preamble signal code of 4096 chips as shown in Figure 2.8.

Figure 2.8 An illustration of over-sampling of an antenna stream.

The locally generated code denoted as $Y_{n,s}(j) = I_Y(j) + Q_Y(j)$, j = 0, 1, 2, 3, …, 4095 is the complex conjugate of $C_{pre,n,s}(j)$ which denotes chip j of the preamble signal with preamble scrambling code n and signature s.

$$Y_{n,s}(j) = C^*_{pre,n,s}(j), \qquad j = 0, 1, 2, 3, \ldots, 4095 \tag{2.11}$$

The correlation between X and Y can be expressed as follows:

$$accIQ_{n,s}(o) = \sum_{j=0}^{4095} X(2j+o)Y_{n,s}(j) \tag{2.12}$$

where o is the offset in samples between the input sample stream and the locally generated code.

Since both X and Y are complex values, $accIQ_{n,s}(o)$ is also complex-value. To find the power of the correlation, we take the square of the amplitude of $accIQ_{n,s}(o)$ as follows:

$$P_{n,s}(o) = \left|accIQ_{n,s}(o)\right|^2 \tag{2.13}$$

Due to carrier frequency shift and the Doppler effect, there is a phase rotation between the received signal and the local replica of transmitted signal. Moreover, the phase rotation changes over time. This has a negative effect on the correlation towards the overall power accumulation when the correlation length (period) is too long. Therefore, in practice, the complete accumulation period is divided into several smaller

accumulation intervals. At the end of each of the intervals, the power of the intermediate result is calculated, and the total accumulation is obtained as follows:

$$P_{n,s}(o) = \sum_{m=0}^{M-1} \left| \sum_{j=0}^{N-1} X(2mN + 2j + o) Y_{n,s}(mN + j) \right|^2 \qquad (2.14)$$

where MN = 4096.

The inner summation is called coherent accumulation, and the outer summation as non-coherent accumulation. In this case, the coherent accumulation length is N, and that of the non-coherent M.

2.9.2 Search window

As stated earlier, the received signal is a time delayed version of the original transmitted signal. The amount of delay is determined by the distance between the user equipment and the base station. Quantitatively, delay by 1 chip in time corresponds to 0.26 μs × 300000000 m/s = 78 m in distance of propagation, since the radio wave travels at the speed of light. The uncertainty region of the delay in time is called the search window for the preamble detection. The node-B has to search the entire cell to determine if there is a preamble existing in the cell, since it has no *a priori* knowledge of the location of the UE. Therefore, the physical cell size determines the search window size of the PD. Table 2.3 gives the relationship between the physical size of a cell and the search window size of the PD.

Table 2.3 Cell sizes and search window sizes

Cell size (km)	Search window size (in chips)	Search window size (in samples)
5	64	128
10	128	256
20	256	512
40	512	1024
80	1024	2048
160	2048	4096
320	4096	8192

2.9.3 Power delay profile

Within the search window, the node-B needs to find out the correlation power at every possible delay location in time. Each of the delay location is called a delay offset with respect to the origin of the search window. The resulting relationship between the delay offsets and the correlation powers is called the power delay profile (PDP) for a given signature. If the expected signature does exist in the received antenna signal, then the PDP would contain a peak power at some delay offset.

2.10 Thresholding

There are many complicated methods to decide how to select and apply the threshold on the PDP. No attempt is made to study all the existing thresholding methods since the focus of this thesis is to develop an algorithm for a fast calculation of the PDP. However, a commonly used method that is employed in this thesis is briefly introduced here.

The threshold θ is defined as the average of powers at all the offsets within the search window and it is given by

$$\theta = \frac{\alpha}{W} \sum_{o=0}^{W-1} P_{n,s}(o) \tag{2.15}$$

where W is the search window size in samples and α is the scale factor.

2.11 Probability of Detection and Probability of False Alarm

This subsection defines the concepts of the probability of detection and probability of false alarm. A good detector should have a high probability of detection and a low probability of false alarm at the same time.

(a) Probability of detection

The probability of detection is defined as the ratio between the number of correct detections and the total number of tests conducted.

(b) Probability of false alarm

The probability of false alarm is defined as the ratio between the number of times a preamble is detected and the total number of tests conducted when the signature is not present in the preamble.

2.12 Summary

RACH preamble detection is an important function of the node-B in a UMTS network. The basic system level knowledge that is related to RACH preamble detection has been introduced in this chapter. Fundamental concepts such as frame structure, access slot, signature, access service class have been defined. The cell size of a node-B decides the search window size of the preamble detection. The correlation length affects the computational load and detection accuracy. To assess the performance of a RACH preamble detector, both the probability of detection and probability of false alarm have to be taken into consideration.

Chapter 3 The Proposed Sample-Decimation Based Algorithm

3.1 Introduction

As described in Chapter 2, the computational complexity of a preamble detector is affected by the cell size and the correlation length. The cell size is decided by the network deployment needs, and thus, cannot be controlled by a preamble detector algorithm. The correlation length, on the other hand, could be reduced from the maximal length of 4096 chips. However, this would result in affecting the detection accuracy.

For a macro base station, where the cell size is typically 10 km or larger and the number of UEs in a cell is 200 or more, the chance to have many users to send the preamble in the same access slot is very slim. This means, most of the delay offsets will correspond to very low correlation power. In other words, most of the delay hypotheses are wrong. It is, therefore, desirable to quickly discard the majority of the delay hypotheses (offsets) that are unlikely to correspond to a strong correlation power. For those offsets that

have strong correlation power, all the available antenna samples of 4096 chips should be used to perform the correlation, thus giving the best detection accuracy.

In order to exploit the fact that most of the delay offsets are not going to lead to strong correlation powers, one can perform the calculation of the PDP in two stages. The first stage performs a coarse search in the entire cell, and gives a list of offsets that are likely to have a strong correlation power. The second stage then performs a fine search only on those offsets that are selected by the first stage. The second stage utilizes the full correlation length when the PDP is calculated, thus preserving the accuracy of the PDP calculation on those selected offsets.

To facilitate the description of the proposed algorithm, the following terms and their abbreviations need to be introduced. As shown in Table 3.1 full search with full contribution (FSFC) refers to the conventional single dwell serial search method, where correlation powers of all the delay offsets are calculated with a correlation length equal to 4096 chips. Full search with partial contribution (FSPC) refers to the search, where correlation powers of all the delay offsets are calculated with a correlation length smaller than 4096 chips. Partial search with full contribution (PSFC) is refers to the search, where correlation powers of some of the delay offsets are calculated with a correlation length equal to 4096 chips. Finally, partial search with partial contribution (PSPC) refers to the search, where correlation powers of some of the delay offsets are calculated with a correlation length smaller than 4096 chips.

Table 3.1 Search mode and correlation mode

		Search mode	
		Full	Partial
Correlation (contribution) mode	Full	FSFC	PSFC
	Partial	FSPC	PSPC

3.2 Proposed Algorithm

The proposed algorithm consists of two stages. In the first stage, a sample-decimated domain is created by downsampling the input antenna samples and the locally generated replica of the transmitted signal, also known as locally generated code. Then, an FSFC is performed in the sample-decimated domain. The PDP in the sample-decimated domain is then used to indicate as to which offsets are likely to have strong correlation powers. This helps to narrow down the potential locations of the preamble with small amount of computation. Then, in the second stage, the correlation between the input sample stream and the locally generated code is performed in the original sample domain, but, the correlation is restricted in the vicinities of those locations that are likely to have strong correlation powers. Thus, second stage performs a PSFC. Details of the algorithm are described in the following sub-sections.

3.2.1 Correlation in the sample-decimated domain

The first step is to generate sample-decimated antenna samples and locally-generated code. But before downsampling the antenna samples and locally generated code, a low pass filtering is performed on the original antenna samples and locally generated code. This helps to increase the association or correlation among adjacent samples, and thus gives better correlation results in the sample-decimated domain.

(a) Preprocessing

The original input antenna samples are represented by {X(i)}. The over sampling factor (OSF) is equal to 2, thus there are two samples in each chip period, and 8192 input antenna samples would correspond to 4096 chips in time.

The input sample stream is first filtered with a low pass filter as follows:

$$X_1(i) = \frac{1}{9}(X(i-2) + 2X(i-1) + 3X(i) + 2X(i+1) + X(i+2)) \qquad (3.1)$$

where i = 2,3, ..., 8192+W-3, and for other values of i, $X_1(i) = 0$, W bring the search window length in samples.

It should be noted that in a real system, the input antenna stream contains a constant flow of samples at the chip rate speed with 3.84×10^6 chips per second. In describing the proposed algorithm, however, only the first 8192+W samples are of concern.

Next, the filtered sample stream is down sampled with a 4:1 ratio as follows:

$$X_2(k) = X_1(4k+1),\ k = 0,\ 1,\ \ldots,\ 2048+W/4-1 \tag{3.2}$$

Similarly, the locally generated code which is 4096 in length is preprocessed as follows:

$$Y_{n,s,1}(j) = \frac{1}{4}(Y_{n,s}(j-1) + 2Y_{n,s}(j) + Y_{n,s}(j+1)) \tag{3.3}$$

where j = 1, 2, ..., 4094, and $Y_{n,s,1}(0) = 0$, $Y_{n,s,1}(4095) = 0$.

The filtered code is then down sampled at a 2:1 ratio as follows:

$$Y_{n,s,2}(k) = Y_{n,s,1}(2k+1),\ k = 0,\ 1,\ \ldots,\ 2047 \tag{3.4}$$

It should be noted that the original {X(i)} sequence is in the half-chip resolution, whereas the original code sequence {Y(j)} is in the chip resolution. That is why, 4:1 down sampling is performed on the filtered antenna sample sequence and 2:1 down sampling is performed on the filtered code sequence. After the processing, $\{X_2(k)\}$ and $\{Y_{n,s,2}(k)\}$ are in the 2-chip per sample resolution.

(b) Correlation

We refer to $X_2(k)$ and $Y_{n,s,2}(k)$ as being in the sample-decimated domain. The correlation between $X_2(k)$ and $Y_{n,s,2}(k)$ is then performed as follows:

$$P_{n,s}(o) = \sum_{m=0}^{M_2-1} \left| \sum_{j=0}^{N_2-1} X_2(mN_2 + j) Y_{n,s,2}(o + mN_2 + j) \right|^2 \tag{3.5}$$

where $M_2N_2 = 2048$, o = 0, 1,.., W/4-1. There are different possible choices for the value of M_2 and N_2. One possible choice is $M_2 = 4$ and $N_2 = 512$.

(c) Sorting

The correlation operation mentioned above gives the PDP in the sample-decimated domain. These offset-power pairs are then sorted according to the magnitude of the power. Then the U pairs corresponding to the U largest powers are selected as the finalists of the first stage. The value of U is programmable. These candidates can be expressed by the set $\{(o_1,P_1), (o_2,P_2),\ldots, (o_k,P_k) ,\ldots, (o_U,P_U)\}$, where (o_k,P_k) represents the power at offset o_k as $P_k=P_{n,s}(o_k)$.

All the steps involved in the first-stage processing are illustrated in Figure 3.1. Both the input antenna signal and the locally generated code sequence are first low-pass filtered. It should be noted that the antenna signal X is over-sampled with OSF=2 as the input. Thus, every two samples correspond to 1 chip in time. On the other hand, the locally generated signal Y not over-sampled, i.e., OSF=1.

After the low-pass filtering, the signals are down-sampled to generate the sample-decimated signals. In the sample-decimated domain, every signal sample corresponds to two chips in time, or 0.52 μs. This holds true for both the decimated antenna samples and the decimated code. The original search window size is W corresponding to W/2 chips in time. Now, the search window size in the decimated-sample domain is W/4, which still corresponds to W/2 chips in time.

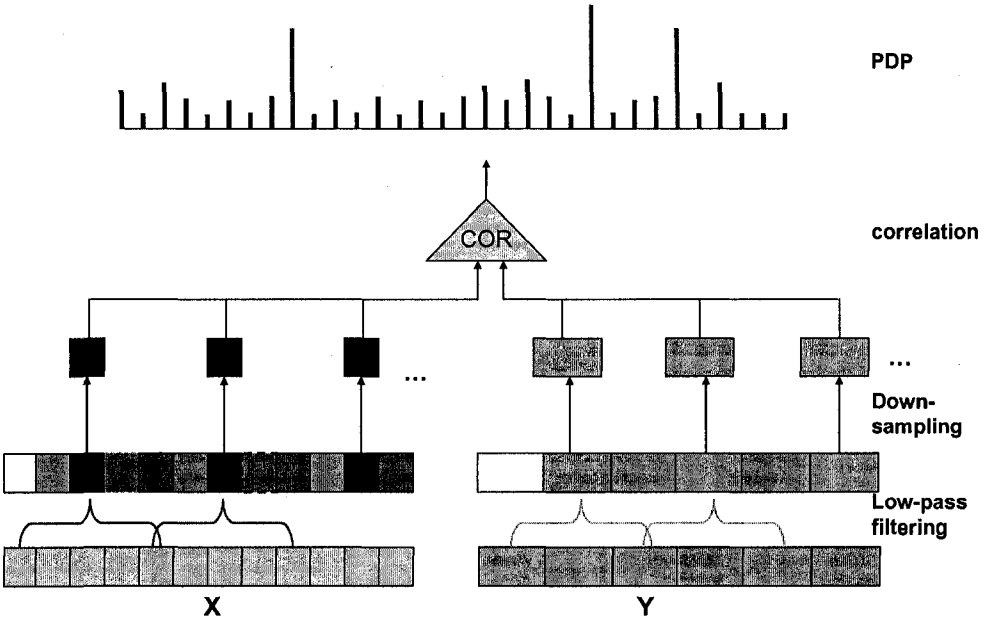

Figure 3.1 Operations performed in the first stage.

Performing correlations between two decimated signals within the corresponding search window produces the PDP in the decimated signal domain.

Finally, if U is set to 3 in the illustration of Figure 3.1, then the finalists would consist of $\{(8,P_8), (22,P_{22}),(26,P_{26})\}$.

(d) Average

During the process of sorting the PDP, the average of all the powers in the search

window is also calculated, and it is given by

$$\overline{P} = \frac{4}{W} \sum_{o=0}^{\frac{W}{4}-1} P_{n,s}(o) \tag{3.6}$$

where W is the search window size in samples in the original sample domain. In the sample-decimated domain, the search window size is actually $\frac{W}{4}$.

3.2.2 Correlation in the original sample domain

In the second stage of the proposed algorithms, correlations are performed in the original sample domain. But instead of performing the correlation for all the offsets in the search window, only offsets centered around the finalists of the first stage are examined for the possible existence of a preamble. According to the definitions introduced in Section 3.1, this is a PSFC operation.

(a) Mapping of delay offsets

The delay offsets of the finalists in the first stage in the sample-decimated domain and the original sample domain are related as

$$o_{org} = 4o_{dec} \tag{3.7}$$

where o_{org} stands for the offset in the original sample domain and o_{dec} the offset in the sample-decimated domain.

To continue the example given in the previous section, the offsets of the three finalists in the sample-decimated domain are mapped into the offsets 32, 88 and 104 in the original sample domain, respectively.

(b) Search range

Due to the coarse resolution in the decimated domain, the PDP calculation in the decimated domain could have ±1 sample error. Therefore, after the mapping, the offsets to be evaluated in the original domain are determined by including the two offsets on both sides of the mapped offset. Specifically, for every finalist in the first stage with offset o_{dec}, the following five offsets in the original domain, ($4o_{dec}-2$, $4o_{dec}-1$, $4o_{dec}$, $4o_{dec}+1$, $4o_{dec}+2$), are evaluated.

Generally speaking, if the U finalists in the first stage are represented by the set $\{(o_1,P_1), (o_2,P_2),\ldots, (o_k,P_k) ,\ldots, (o_U,P_U)\}$, then the second stage would perform a correlation on the following offsets in the original sample domain: v = ($4o_1$-2, $4o_1$-1, $4o_1$, $4o_1+1$, $4o_1+2$, $4o_2$-2, $4o_2$-1, ... , $4o_U+2$). In other words, a total of 5U offsets are searched in the original domain.

To continue with the example given in the previous section, the following offsets in the original sample domain: v = (30, 31, 32, 33, 34, 86, 87, 88, 89, 90, 102, 103, 104, 105, 106), are evaluated.

(c) Correlation

Finally, a correlation between the antenna signal {X} and the locally generated code sequence {Y} is performed on the offsets given in v.

(d) Detection

Using the power average $\overline{P}$ as calculated in the first stage according to (3.6), we can set the detection threshold θ as

$$\theta = \beta \overline{P} \tag{3.8}$$

where β is a scale coefficient. This scale factor is determined by a predefined probability of false alarm on the training test vectors. A positive detection is declared if the largest correlation power is larger than the threshold θ.

3.3 Computational Complexity of the Proposed Algorithm

In order to appreciate the amount computational savings that the proposed algorithm offers, in this section the analyses of the computational complexities of both the proposed algorithm and the full search algorithm are carried out.

3.3.1 Computational load of the full search algorithm

According to (2.14), for correlation length of COR_LEN=4096 chips, coherent length N =512 and non-coherent length M=8 and search window of SW_LEN=256 offsets, one

needs the following numbers of complex multiplications,

$$\begin{aligned} &TOTAL\_Number\_of\_Comp\_Multiplications\_FS \\ &= SW\_LEN \times COR\_LEN \\ &=256 \times 4096 \end{aligned} \tag{3.9}$$

and the following number of square of the absolute value,

$$\begin{aligned} &TOTAL\_Number\_of\_Abs\_Absolute\_FS \\ &= SW\_LEN \times M \\ &=256 \times 8 \end{aligned} \tag{3.10}$$

and the following number of complex additions,

$$\begin{aligned} &TOTAL\_Number\_of\_Comp\_Additions\_FS \\ &= SW\_LEN \times (COR\_LEN - M) \\ &= 256 \times (4096 - 8) \end{aligned} \tag{3.11}$$

and scalar additions,

$$\begin{aligned} &TOTAL\_Number\_of\_Scalar\_Additions\_FS \\ &= SW\_LEN \times (M - 1) \\ &= 256 \times (8 - 1) \end{aligned} \tag{3.12}$$

The following approximations are valid for modern digital signal processors.

One scalar addition = one scalar multiplication,

One square of the absolute value = 3 scalar additions,

One complex addition = 2 scalar additions,

One complex multiplication = 6 scalar additions.

Given the above approximations, the equivalent total number of scalar additions for FSFC is as the follows:

$$\begin{aligned} & TOTAL\_Num\_Adds\_FS\_Equivalent \\ &= 6 \times SW\_LEN \times COR\_LEN + 3 \times SW\_LEN \times M + \\ & \quad 2 \times SW\_LEN \times (COR\_LEN - M) + SW\_LEN \times (M - 1) \\ &\approx 8 \times SW\_LEN \times COR\_LEN \end{aligned} \tag{3.13}$$

For COR_LEN=4096 chips, N=512, M=8, and SW_LEN=256, we have 8×256×4096 = 8388608 as the equivalent total number of scalar additions for FSFC.

3.3.2 Computational load of the proposed algorithm

For the proposed algorithm, the computational load comprises two parts corresponding to the two stages.

(a) The first- stage computations

In the first stage, the analysis of the computational load is similar to that of the full search except that this is in the sample-decimated domain. In this domain, the correlation length is COR_LEN_SD=2048 chips and search window consists of SW_LEN_SD=64 offsets. The task performed in the sample-decimated domain is an FSFC, for which, according to (3.13), the total number of equivalent additions needed in the first stage is given by

$$\begin{aligned} & TOTAL\_Num\_Adds\_SD\_1stStage\_Equivalent \\ &\approx 8 \times SW\_LEN\_SD \times COR\_LEN\_SD \end{aligned} \tag{3.14}$$

It can be seen that computation load of the first stage is roughly 1/8 of that needed by the full search algorithm, since SW_LEN_SD is one-quarter of SW_LEN and COR_LEN_SD is one-half of COR_LEN.

(b) The second-stage computations

In the second stage, only *5U* offsets need to be checked. This is equivalent to having a search window size of 5U in the original sample domain. Therefore, total number of equivalent additions needed in the second stage is given by

$$\begin{aligned} &TOTAL\_Num\_Adds\_SD\_2ndStage\_Equivalent \\ &\approx 8 \times 5U \times COR\_LEN \end{aligned} \tag{3.15}$$

Thus, the second stage computational load is about $\frac{5U}{SW\_LEN}$ of that needed for the full search algorithm.

The total complexity of first-stage and second-stage together is given by

$$\begin{aligned} &TOTAL\_Num\_Adds\_SD\_Equivalent \\ &\approx 8 \times SW\_LEN\_SD \times COR\_LEN\_SD \\ &\quad + 8 \times 5U \times COR\_LEN \end{aligned} \tag{3.16}$$

$$= 8 \times 64 \times 2048 + 8 \times 5U \times 4096$$

3.3.3 Computational savings

Based on the analyses given above, the ratio of the computational loads of the proposed

algorithm and the FSFC is roughly given by

$$R = \frac{1}{8} + \frac{5U}{SW\_LEN} = \frac{1}{8} + \frac{5U}{256} \tag{3.17}$$

This is a substantial saving given a moderate value of U. For example, if U = 10, then R = 1/8 + 50/256 = 0.32. That is, the proposed algorithm offers a saving 68% in the computation load over that of the full search when U=10.

It should be pointed out that the sorting and low-pass filtering needed in the first stage of the proposed algorithm is negligible in terms of computational load compared to that needed for the correlation.

3.4 Summary

In this chapter, the proposed algorithm has been presented. The rationale behind this algorithm is that most of the delay offsets in the search window usually do not lead to a strong correlation power. Thus, they are not very useful for preamble detection. Thus, these offsets can be eliminated as soon as possible. Transforming the original samples into the sample-decimated domain makes it possible to discard those offsets with a very little computational cost. The samples in the decimated domain are still inter-related even though the precision of the delay hypothesis becomes low.

The partial search with full contribution performed in the second stage helps to retain the detection accuracy. For offsets passed on to the second stage, power computations are the

same as when performed FSFC. This is the reason for the proposed algorithm to provide a detection accuracy very close to that of the FSFC with a reasonable value of U.

The computational savings of the proposed algorithm are obvious. The question is, how well it could perform in terms of detection accuracy. This question can be best addressed through extensive computer simulations under various levels signal-to-noise ratio. The following chapter presents the simulation results.

Chapter 4 Simulation Study

4.1 Introduction

Extensive computer simulations are conducted to study the performance of the proposed algorithm presented in Chapter 3. Two important aspects of assessing the performance of preamble detection are the detection accuracy and the computational complexity.

The traditional single dwell serial search referred to in this thesis as the full search with full contribution is used as the baseline. This is the brute-force way of obtaining the power delay profile. It provides the best detection accuracy, but it is computational expensive.

Another existing algorithm [35] reported in the literature is also implemented in this study. This algorithm also performs a full search. However, it uses only part of the available 4096 chips of the preamble signal for the correlation. In this thesis, we refer to it as full search with partial contribution.

The proposed algorithm and the other two algorithms are implemented under the same condition. The majority of the simulation results are presented in a statistical format. In

order to explain how these statistics are obtained, an illustrative example is given first to show the process step by step.

4.2 An Illustrative Example

This section presents an example to illustrate the entire process of the proposed method.

The input antenna stream samples are shown in Figure 4.1. There is a total of 8192 + 256

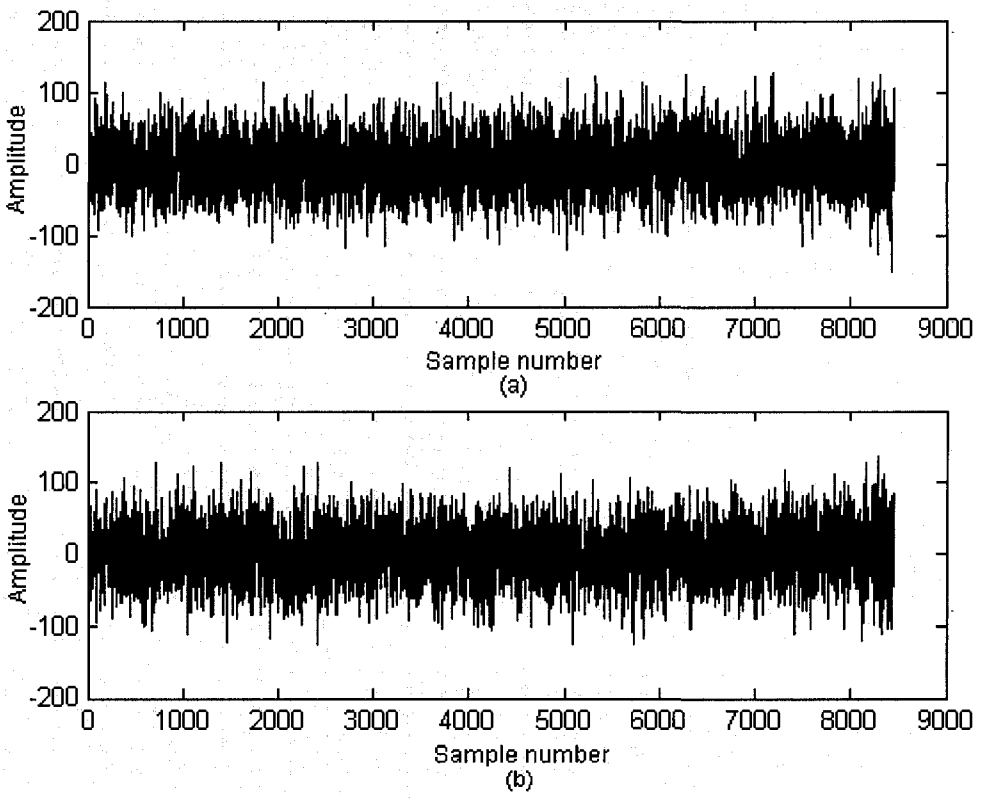

Figure 4.1 Antenna samples. (a) Real part. (b) Imaginary part.

samples in the sequence, corresponding to 4096 + 128 chips, since the sampling rate is 2 samples/chip. The search window size is 256 samples. It should be noted that the input samples are complex valued samples. Figure 4.1 plots the real (In-phase) and imaginary (quadrature) parts separately. The real and imaginary parts of a sample are also called I and Q parts, respectively.

To have a closer look of the antenna samples, Figure 4.2 shows the first 128 samples of the antenna sequence shown in Figure 4.1. This is the preamble signal with -11 dB additive white Gaussian noise (AWGN) added to it.

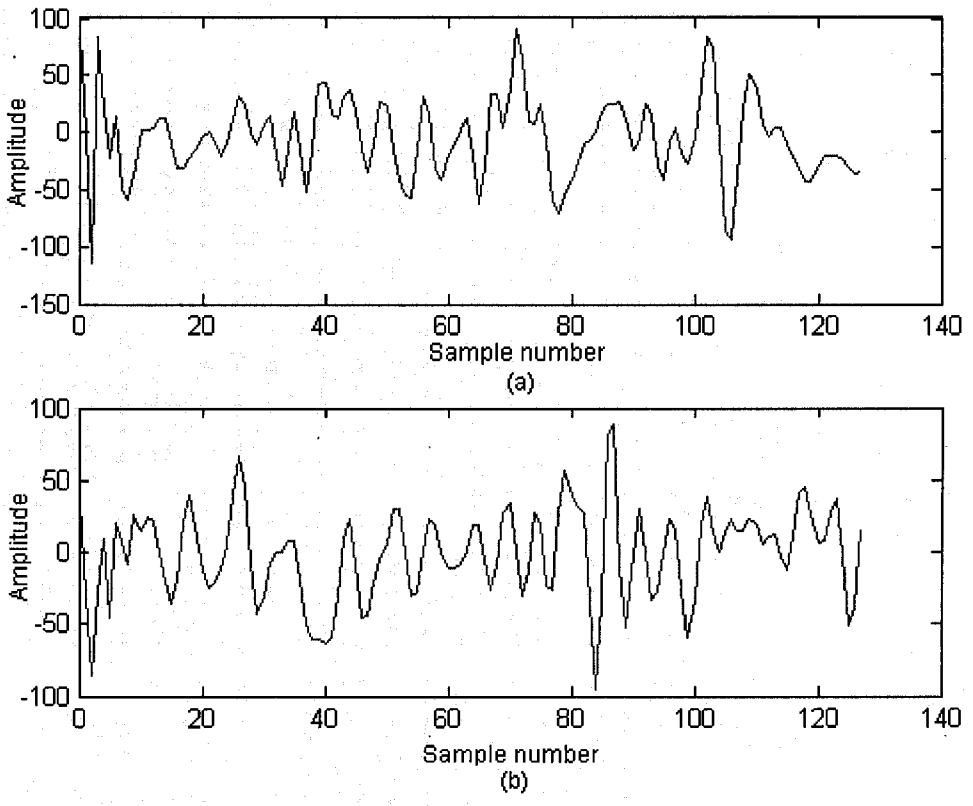

Figure 4.2 The first 128 antenna samples. (a) Real part. (b) Imaginary part.

The locally generated code sequence is, in essence, the replica of the antenna sequence transmitted by the UE at the transmitter side. The node-B, i.e., the receiver, is able to generate this replica due to the fact all the elements needed to generate this replica are available to node-B. These elements are the signature id and the scrambling code number. The Locally generated code is 4096 chips long. There are 4096 samples in the locally generated code sequence since there is one sample per chip. This sequence is shown in Figure 4.3. As seen in the figure, the amplitude of the signal is 0.707.

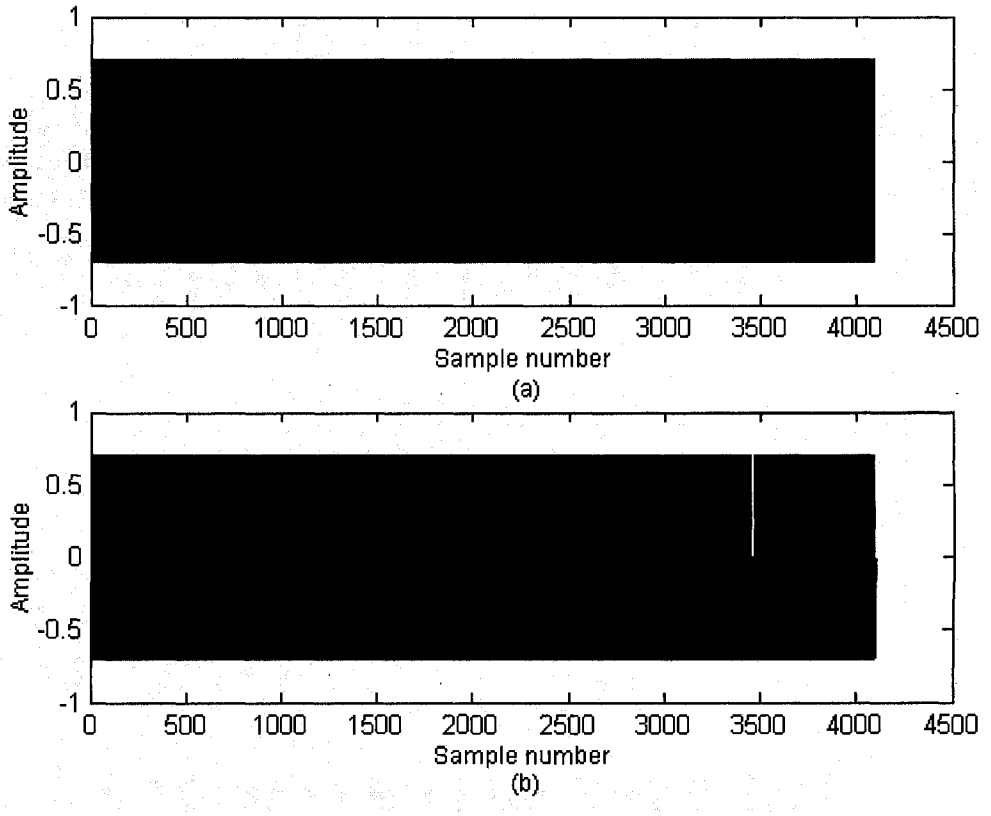

Figure 4.3 Locally generated code sequence. (a) Real part. (b) Imaginary part.

Since it is difficult to see the values of the samples given the scale of the x-axis, the first 128 samples of this sequence are shown in Figure 4.4. The locally generated code sequence used in this example is generated using the scrambling code number 0 and signature id=1. This code is then correlated with the received antenna sample stream to generate the PDP.

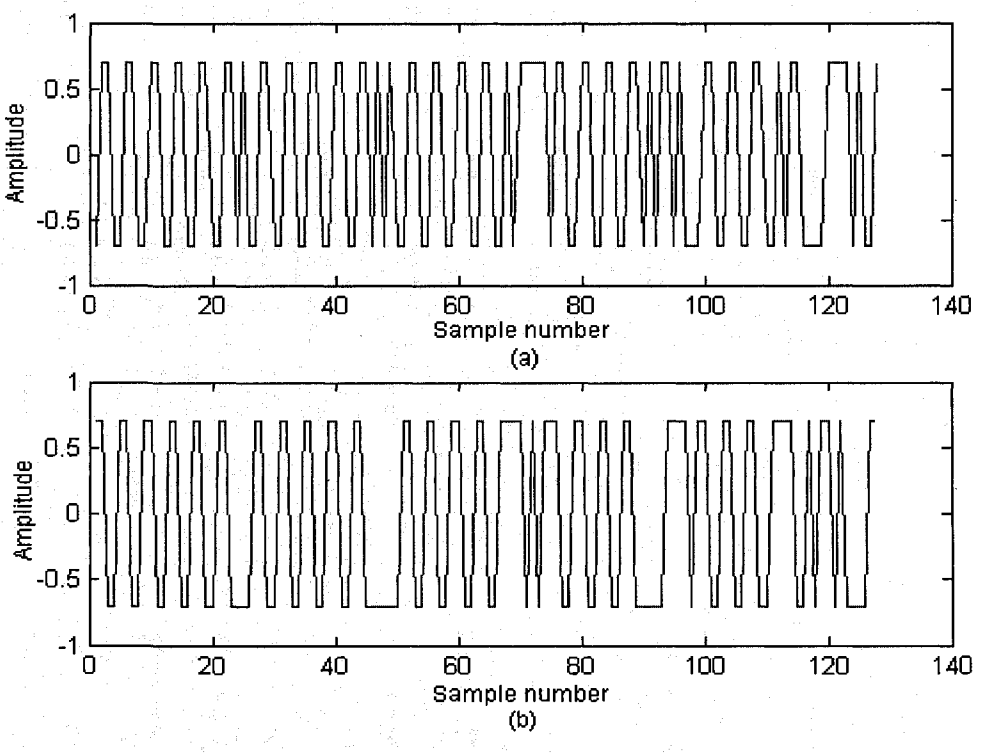

Figure 4.4 The first 128 chips of the locally generated code sequence. (a) Real part. (b) Imaginary part.

The PDP refers to the relationship between the delay offsets in the search window and the corresponding correlation powers. In other words, PDP is a correlation power as a function of the delay offset. For a given delay offset, also known as delay

hypothesis, the correlation power is the result of a correlation between the current antenna sequence in the input buffer and the locally generated code sequence.

For search window of 256 offsets, the coherent correlation length of 512, and the non-coherent correlation length of 8, the resulting PDP is shown in Figure 4.5.

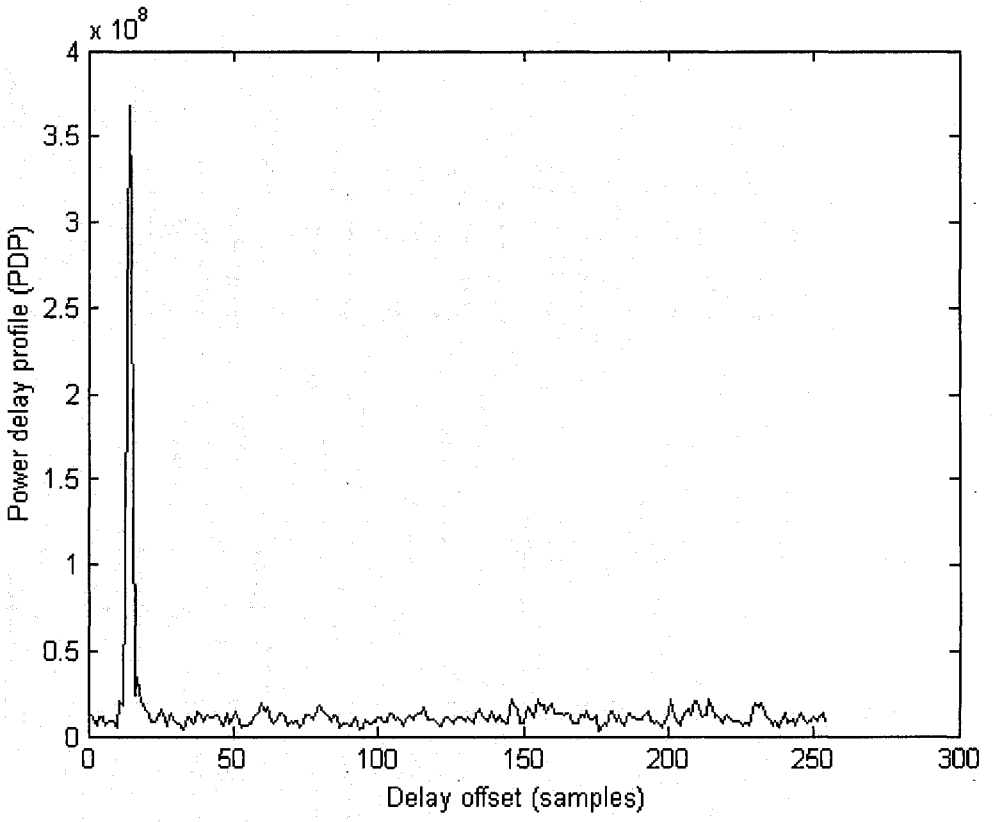

Figure 4.5 power delay profile in the original sample domain.

It can be seen from the PDP figure that there exists a strong correlation power in the vicinity of the offset of value 14. This result is obtained using the conventional FSFC method. In the following, we will show that the proposed method can reach the same conclusion about the location and magnitude of the strongest correlation power.

The low-pass filtered antenna samples are shown in Figure 4.6.

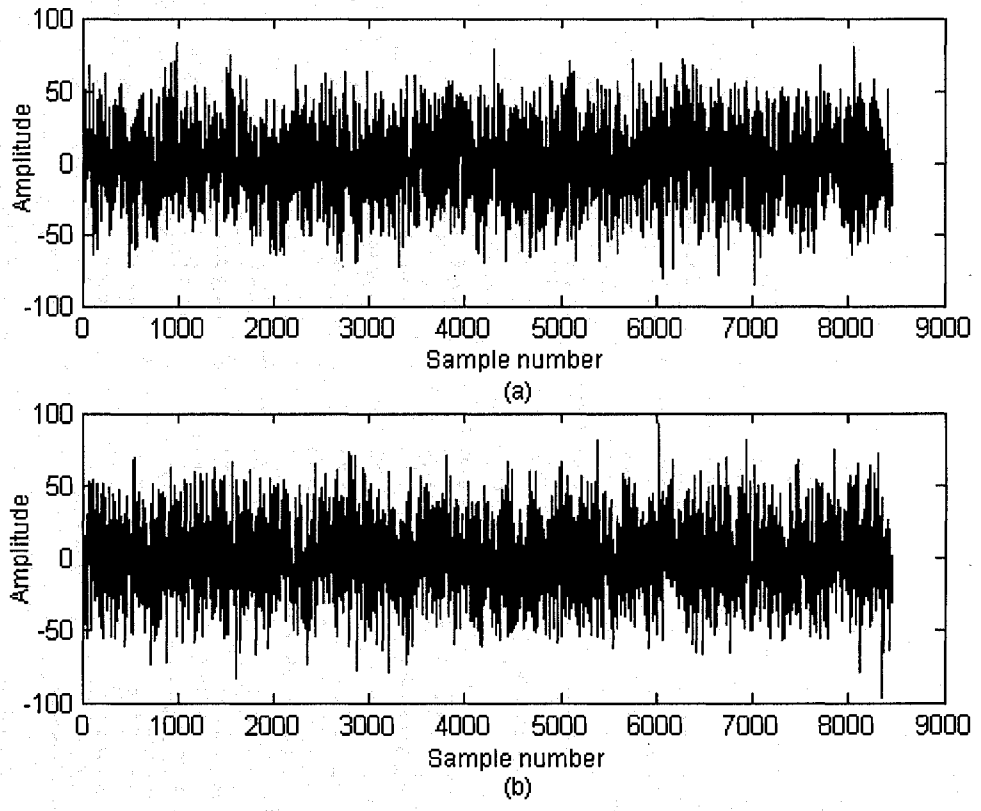

Figure 4.6 Low-pass filtered antenna samples. (a) Real part. (b) Imaginary part.

The first 128 samples of the low-pass filtered antenna samples are shown in Figure 4.7. A closer comparison between Figure 4.2 and Figure 4.7 indicates that after the filtering, the resulting curve becomes more "smooth", and the sample values are more inter-related.

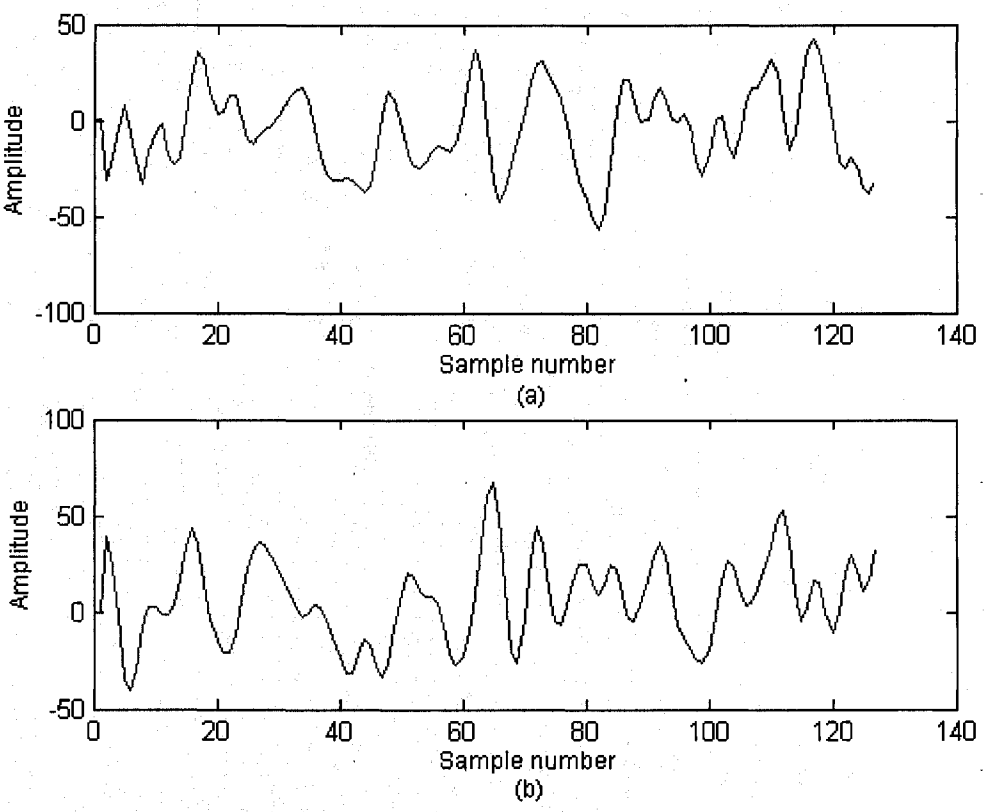

Figure 4.7 The first 128 samples of the low-pass filtered antenna samples. (a) Real part. (b) Imaginary part.

Similarly, the locally generated code is also processed using a low pass filter. The result samples are shown in Figure 4.8 with the first 128 samples shown in Figure 4.9. Due to

the averaging effect of the low pass filtering, the resulting filtered samples are more inter-related which can be seen by comparing Figure 4.4 and Figure 4.9.

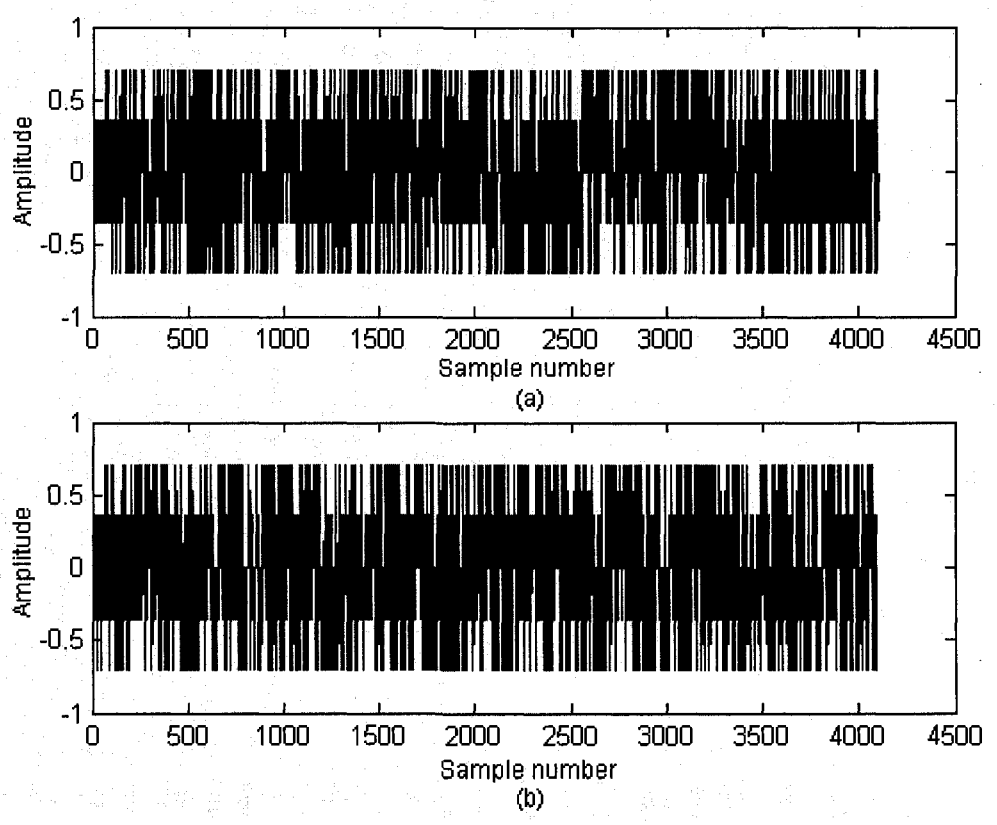

Figure 4.8 Low-pass filtered locally-generated code samples. (a) Real part. (b) Imaginary part.

After preprocessing, the antenna samples and locally generated code are down-sampled into the sample-decimated domain. The decimated antenna samples are shown in Figure 4.10. The decimation ratio for the antenna samples is 4:1, giving a total of (8192+256)/4 = 2048 + 64 samples in the sample-decimated domain, as shown in Figure 4.10. To show

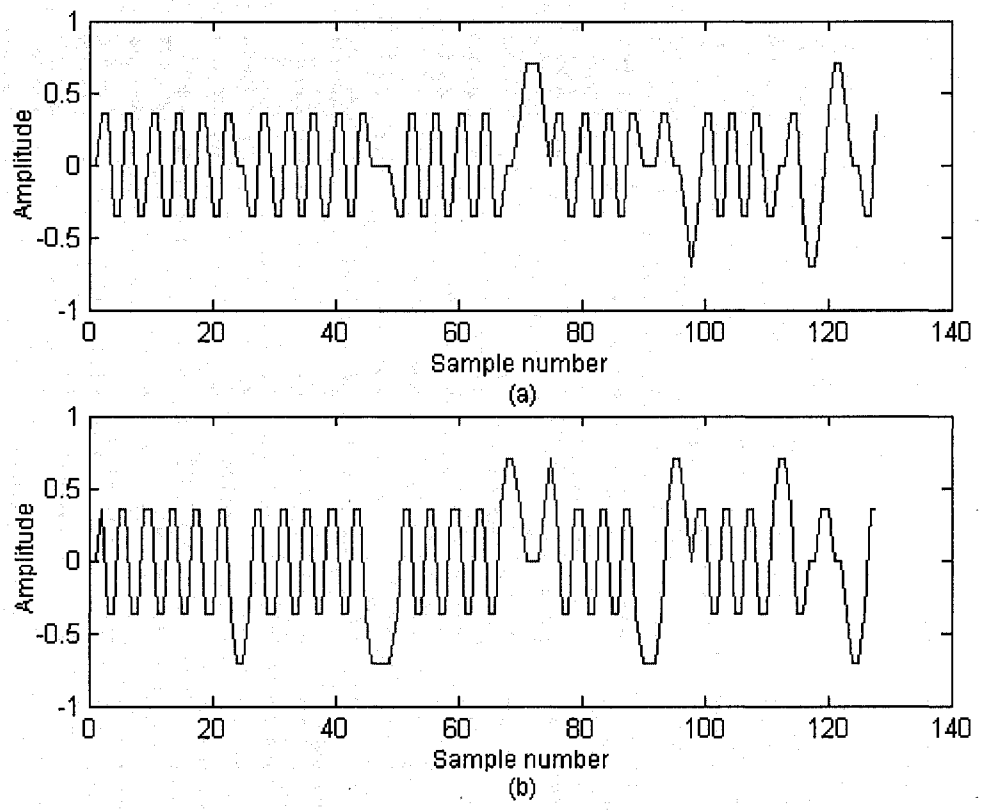

Figure 4.9 The first 128 samples of the sequence in Figure 4.8. (a) Real part. (b) Imaginary part.

the details of the sample sequence, the first 128 samples of the sequence is shown in Figure 4.11. It should be noted that after the decimation, each offset in the search window corresponds to 2 chips in time.

Similar operation is performed on the locally generated code. The difference is that the down sampling rate is 2:1. But this leads to same resolution of the offsets, that is, each offset represents two chips in time.

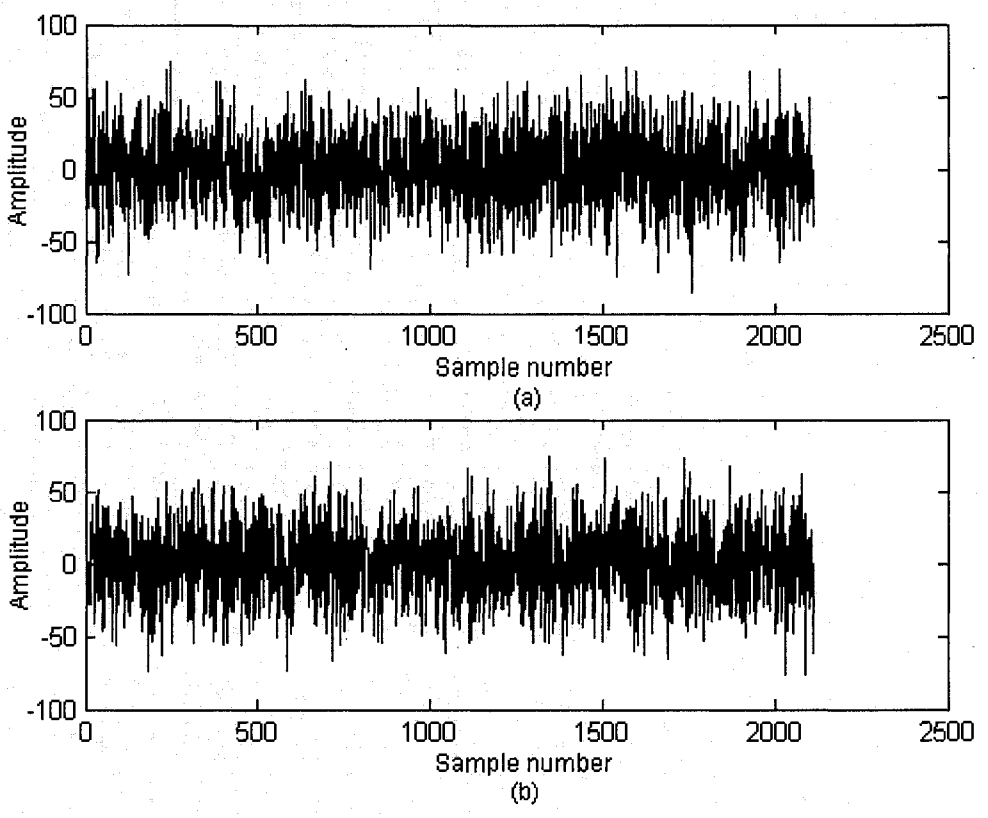

Figure 4.10 Decimated antenna samples. (a) Real part. (b) Imaginary part.

The decimated code samples sequence and the first 128 samples of the sequence are shown in Figure 4.12 and Figure 4.13, respectively.

After the decimation, an FSFC is performed in the decimated domain. The resulting PDP in the decimated domain is shown Figure 4.14. It should be noted that the search window size is reduced to 64 offsets even though the search window still corresponds to 256 chips in time. It can be seen that the peak of the correlation is located around the offset of 4.

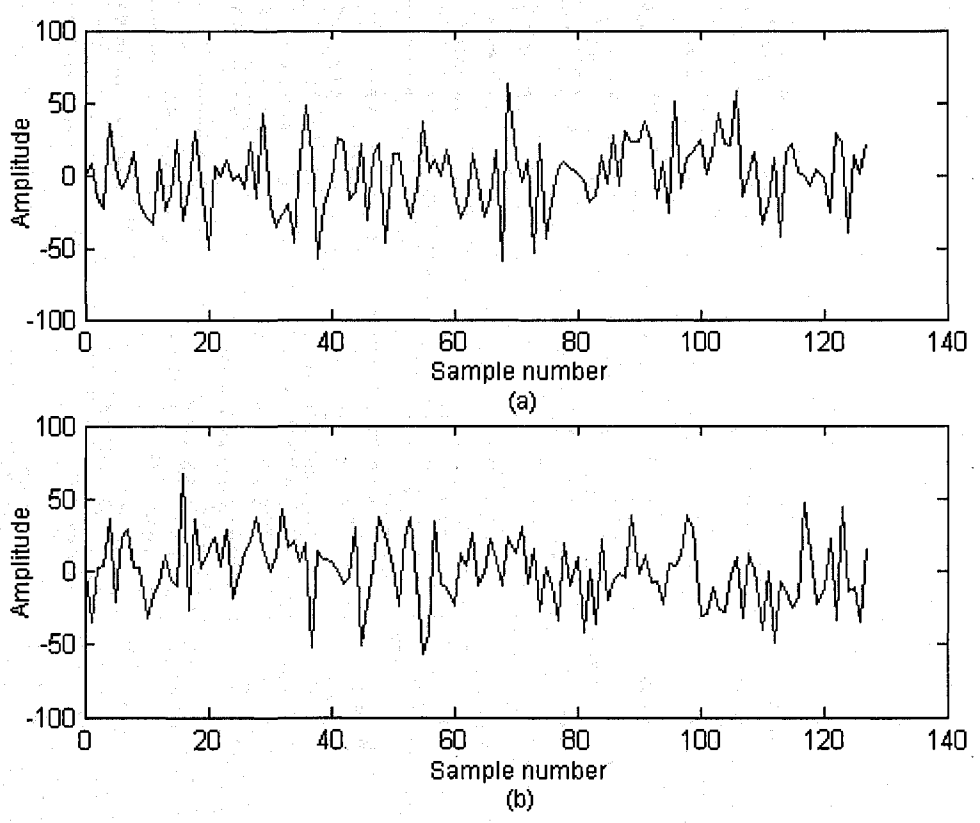

Figure 4.11 The first 128 samples of the decimated antenna signal. (a) Real part. (b) Imaginary part.

If only one finalist is chosen from the first stage, i.e., U = 1, then the corresponding offsets in the original sample domain to be searched are v=(14,15,16,17,18).

In the second stage, a PSFC is performed on the five candidate offsets in the original sample domain. The resulting correlation powers are shown in Figure 4.15. Two points can be made from the results of this figure when compared with that of Figure 4.5: The

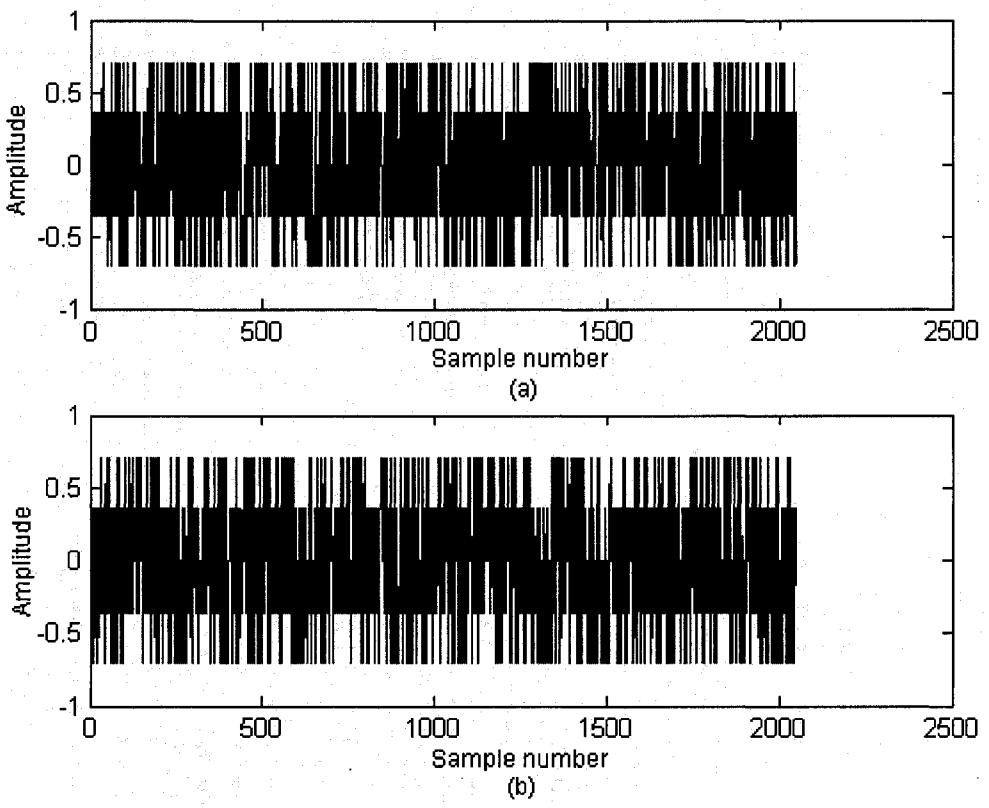

Figure 4.12 Locally generated decimated code sequence. (a) Real part. (b) Imaginary part.

PDP shown in Figure 4.15 has only five offsets and the corresponding powers, whereas the one in Figure 4.5 has all the offsets and their corresponding powers. The second observation is that power values for the offsets present in Figure 4.15 are the exactly the same as the corresponding values in Figure 4.5.

These results suggest that as long as the strongest delay offset is among the finalists, it makes no difference as to how many offsets are among the finalists. More importantly,

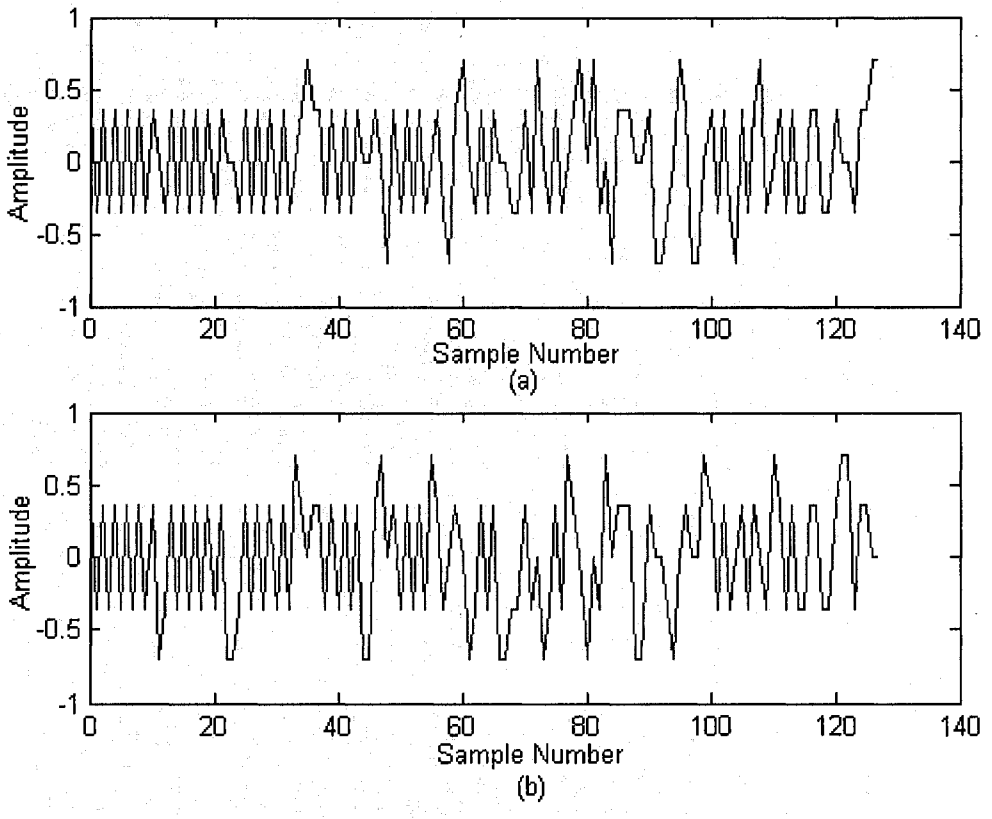

Figure 4.13 The first 128 samples of the decimated code. (a) Real part. (b) Imaginary part.

the fact that these finalists in the proposed algorithm have correlation powers the same as in the conventional FSFC method suggests that the proposed method should have a detection accuracy very close to that of the conventional FSFC method.

4.3 Simulation Results

In order to study the performance of the proposed algorithm, extensive simulations have been performed under various signal-to-noise ratios. For the purpose of comparison, the

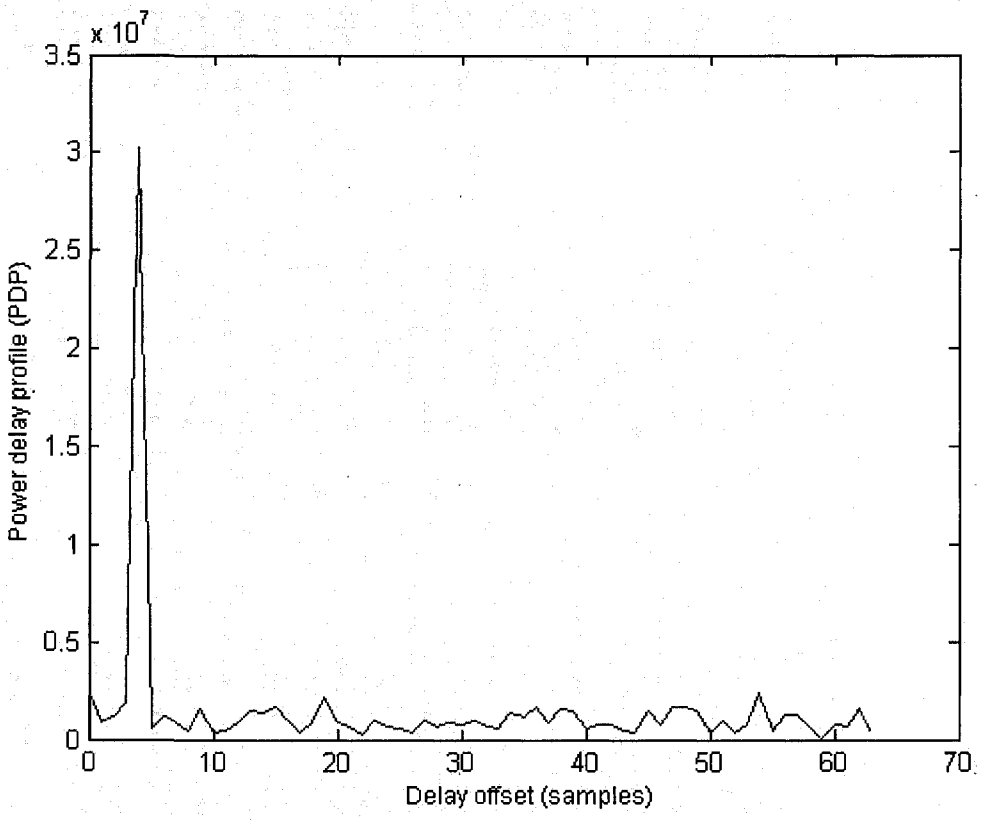

Figure 4.14 Power delay profile in the decimated domain.

FSFC and FSPC methods are also implemented to serve as a baseline in terms of computational complexity and detection accuracy.

4.3.1 Test vector suite

According to the 3GPP standard, the preamble signal is 4096 chips long. With over sampling factor of 2, there are 8192 samples in the preamble signal. The signal is first

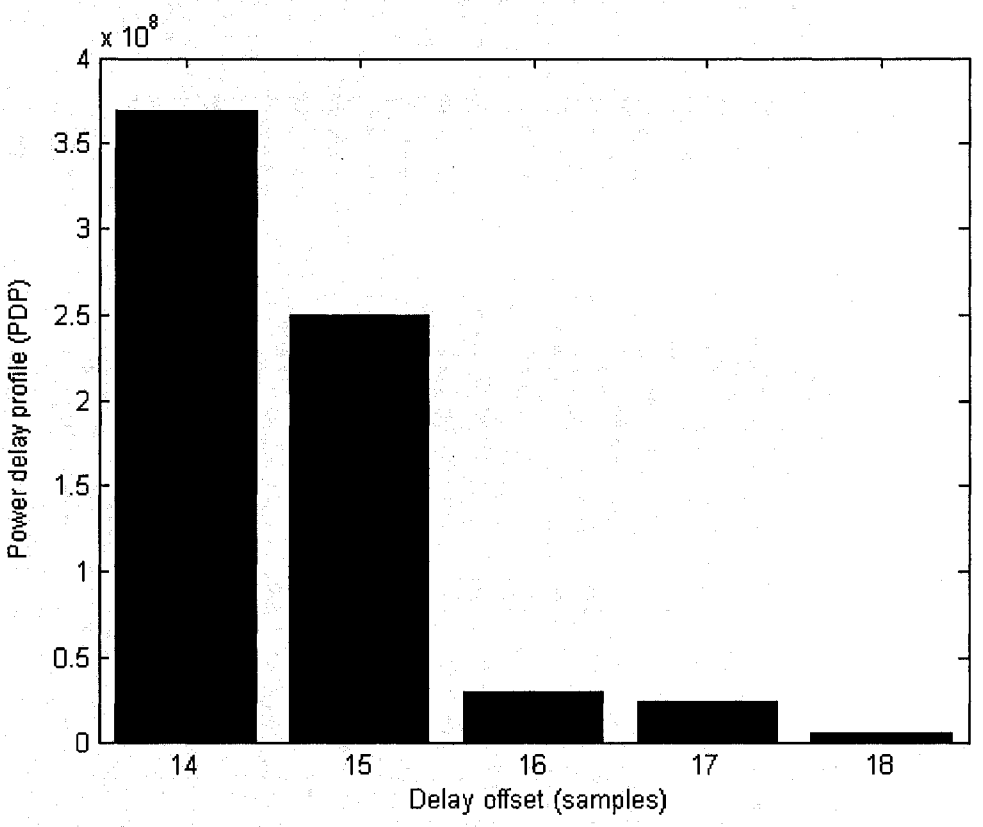

Figure 4.15 Partial PDP of the illustrative example.

generated according to (2.10). Then a random noise with a normal distribution is added to the signal. According to 3GPP standard, the range of the nominal signal-to-noise (SNR) is between -25 dB and -1 dB for PD algorithms conformance tests [44]. In order to have a good test coverage, the SNR values are chosen in the range -25 dB and -15 dB with an incremental step size of 1 dB. For each of SNR value, 1000 test vectors are randomly generated, and for each of the test vectors, the signature is randomly chosen between 0 and 15 inclusively. The scrambling code number is randomly chosen between

0 and 256, and the delay offset is randomly chosen between 0 and 255. All these parameters are summarized in Table 4.1.

Table 4.1 Test vector suite parameters

Parameter	**Range**
SNR	[-25 dB, -15 dB]
Delay offset	[0, 255]
Signature number	[0, 15]
Scrambling code number	[0, 256]

The search window size is 256 corresponding to 128 chips in time. Each test vector has a fixed length of 8192+256 samples, among which 8192 are noise-corrupted preamble signal samples and the remaining 256 are pure random noise samples. Part of the 256 samples are in the beginning of the test vector and the remaining at the very tail end of the main preamble signal body. The specific position and length of the samples depend on the offset value, illustrated in Figure 4.16.

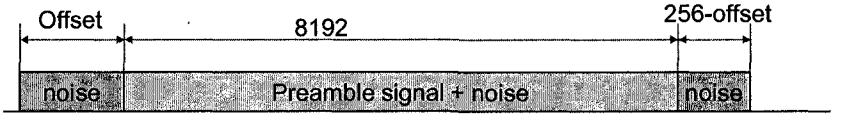

Figure 4.16 Test vector with a delay offset.

4.3.2 Simulation statistics

For each of the algorithms, the threshold is first determined by keeping the false alarm rate below 10^{-3}. This is done by trial-and-error using a set of 1000 preambles at different levels of SNR with no signatures in them. The threshold is adjusted until a false alarm rate is just below 10^{-3} is achieved. To make a fair comparison, all the three algorithms implemented in this study are made to go through the same process for choosing a proper threshold so that they all have the same false alarm rate.

Once the threshold value is fixed, the detection is carried out by using the peak power of the PDP to compare it with the threshold. If the peak power of the PDP is above the threshold, and the signature used in the detector matches with the signature contained in the antenna stream, then it is considered a successful detection; otherwise, it is not. It is assumed that there are 16 signatures in the ASC.

For the proposed algorithm, we choose U = 13, thus, the proposed algorithm uses about 38% of the computation load as that of the FSFC algorithm according to (3.17). For the FSPC algorithm, we choose the total correlation length to be 3/4 of that of the FSFC algorithm. This results in the computation load of the FSPC algorithm to be 75% of that of the FSFC algorithm. The computational complexities of the three algorithms are summarized in Table 4.2.

Table 4.2 Relative computational complexities of three algorithms with U=13

FSFC	Proposed method (U=13)	FSPC
100%	38%	75%

In the table, FSFC is used as the baseline for the computation load. In comparison to the computational load of FSFC, the computation load of FSPC is 75%, and that of the proposed algorithm with U=13 only 38%. The following shows what detection performance we get for the given computation loads.

Table 4.3 shows the statistical results of the tests. The detection accuracy is defined as the ratio of the number of successful detections to the total number of trials (tests) conducted. As it can be seen from this table, the three algorithms perform the same when the SNR is above -18 dB. This is understandable, since when the signal is strong enough, any of the fast algorithms could detect the preambles correctly just like the FSFC algorithm. However, as the SNR decreases, it becomes more and more difficult to detect the preamble correctly. The detection accuracy of the FSFC starts to drop monotonically. This holds true for both the proposed and FSPC algorithms as well. However, what is worth noting is that the detection accuracy of the proposed algorithm is always better than that of the FSPC algorithm. This is consistent in the entire range of the SNR range tested.

To provide a visual comparison, the test results of Table 4.3 is used to plot in Figure 4.17 the probability of detection as a function SNR for the three algorithms. It can be seen from this figure that the proposed algorithm provides a much better detection accuracy than that provided by the FSPC algorithm does, while its computational load is only 38% of that of FSFC, and nearly one-half of that of FSPC.

Table 4.3 Probability of detection for three algorithms with U=13 under various SNR values

SNR(dB)	FSFC	Proposed method (U=13)	FSPC
-25	0.104000	0.066000	0.055000
-24	0.245000	0.180000	0.140000
-23	0.488000	0.379000	0.287000
-22	0.733000	0.615000	0.516000
-21	0.945000	0.846000	0.773000
-20	0.988000	0.956000	0.925000
-19	0.999000	0.993000	0.992000
-18	1	1	1
-17	1	1	1
-16	1	1	1
-15	1	1	1

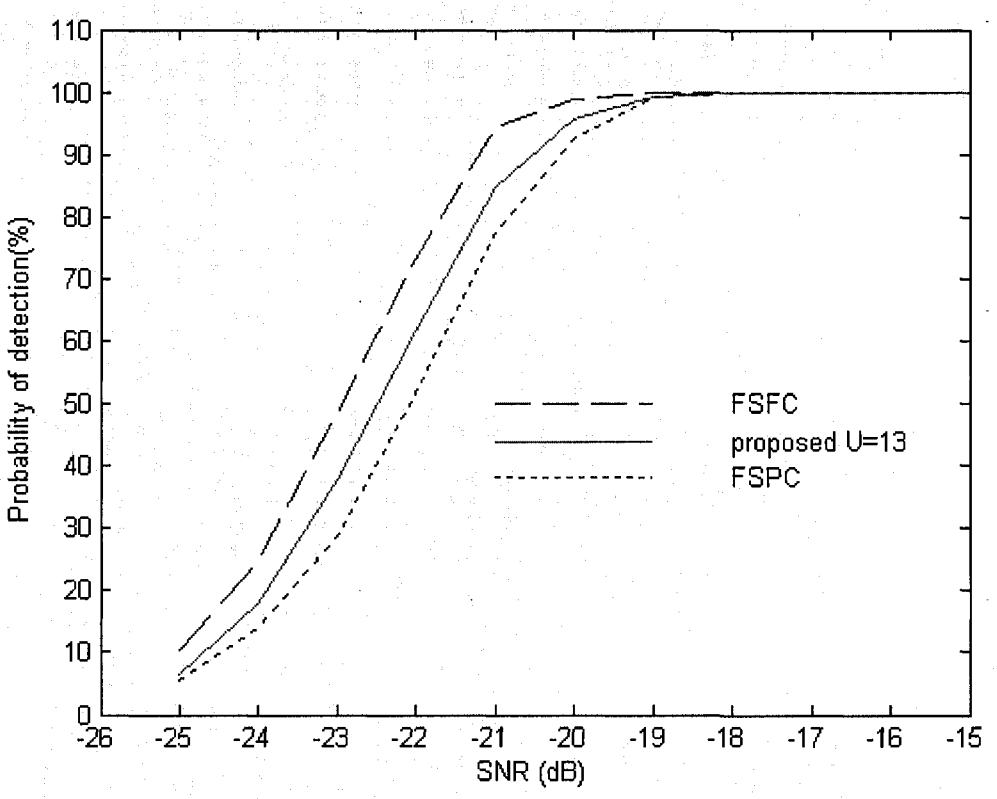

Figure 4.17 Detection performance with $U = 13$.

If for the proposed algorithm, one is willing to tolerate the same computational cost as that of FSPC by setting U=32 (see Table 4.4), it would be possible to improve its detection probability even further. This is clearly seen from the test results in Table 4.5 and the corresponding plot in Figure 4.18. Thus, a trade off between the computational load and the detection accuracy of the proposed algorithm can be achieved by varying the value of the parameter U.

Table 4.4 Relative computational complexities of three algorithms with U=32

FSFC	Proposed method (U=32)	FSPC
100%	**75%**	**75%**

Table 4.5 Probability of detection for three algorithms with U=32 under various SNR values

SNR(dB)	FSFC	Proposed method (U=32)	FSPC
-25	0.104000	0.070000	0.055000
-24	0.245000	0.189000	0.140000
-23	0.488000	0.395000	0.287000
-22	0.733000	0.645000	0.516000
-21	0.945000	0.874000	0.773000
-20	0.988000	0.971000	0.925000
-19	0.999000	0.996000	0.992000
-18	1	1	1
-17	1	1	1
-16	1	1	1
-15	1	1	1

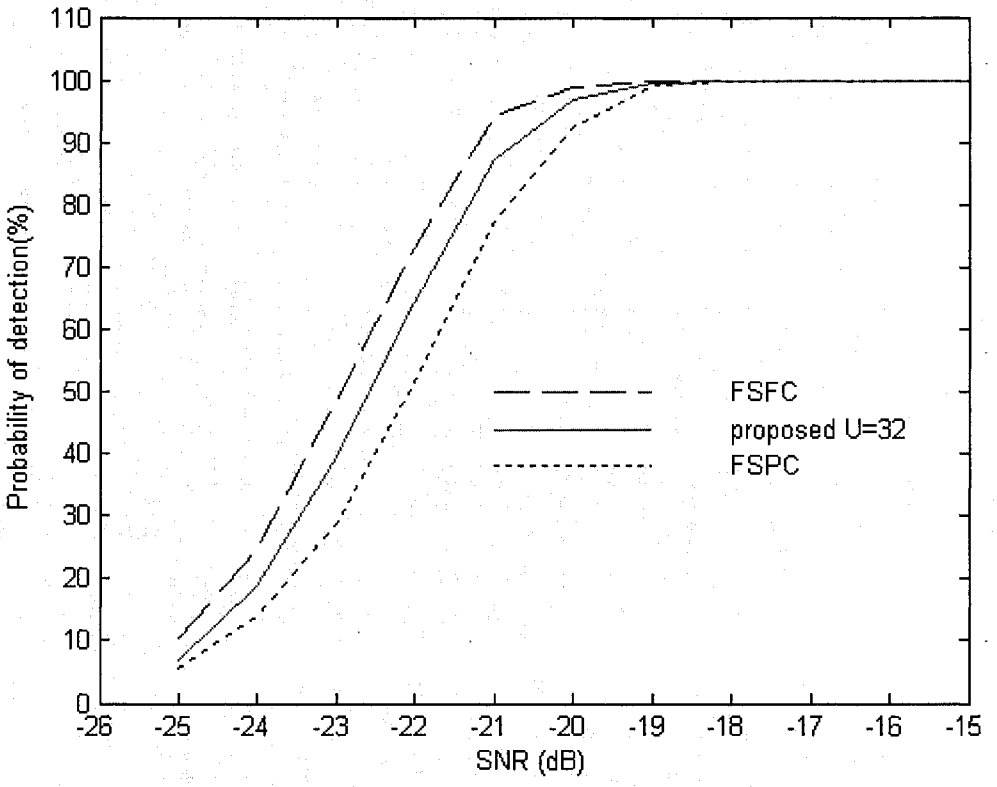

Figure 4.18 Detection performance with $U = 32$.

4.4 Summary

In this chapter, extensive computer simulations have been carried to verify the performance of the proposed algorithm. To explain the steps of the proposed algorithm in detail, first, an illustrative example has been considered. The FSFC, FSPC and proposed algorithm have been implemented to provide the relative performance of the three algorithms.

The probability of detection alone cannot be used to determine the detection performance of a PD algorithm. In order to provide a fair comparison, the threshold value for the three algorithms are chosen in such a way that they all yield the same probability of false alarm, which has been chosen as 10^{-3}.

The simulation results have shown that the proposed algorithm outperforms the FSPC algorithm. It can have a detection accuracy very close to that of the FSFC while still providing a huge savings in terms computational load. The reason for the superior performance of the proposed algorithm is as follows: In the first stage of the algorithm, the large number of unlikely offsets are discarded, and they are not used in the second stage for the correction computation, thus providing huge savings in the computational cost. In the second stage, the remaining offsets, however, provide the same values of the correlation power as that provided by FSFC, which results in a high detection accuracy.

Chapter 5 Conclusion

5.1 Concluding Remarks

Random access is an important function of a node-B in a UMTS system. In essence, preamble detection for the random access is a burst synchronization issue. The node-B needs to perform the detection within a limited time on a fixed length of preamble signal. Since there is no *a priori* knowledge of the location of the UE, the node-B needs to search the entire cell for the possible existence of a preamble, thus requiring a search window size equivalent to the cell size. To make the situation even worse, there are up to sixteen possible signatures that need to be detected. This makes the preamble detection computationally the most intensive task in the node-B channel card. Therefore, developing a fast and reliable preamble detection technique is essential for an efficient UMTS system.

In this thesis, a new preamble detection algorithm is developed. The proposed algorithm decomposes the task of the preamble detection into two stages. In the first stage, offsets within the search window are evaluated according to their correlation powers. Those with

low correlation powers are excluded from further consideration. This is achieved by transforming the antenna stream sequence and the locally generated code into a sample-decimated domain, and performing a full search full contribution in that domain. In the second stage, the remaining offsets are re-evaluated using a partial search full contribution. The correlation powers of those remaining offsets are the same as those obtained in the full search full contribution, since all the antenna samples and the locally generated code are used for the calculation of the correlation powers. This, in essence, is the reason for the proposed algorithm to provide a detection accuracy very close to that of the conventional FSFC algorithm.

Extensive computer simulations have been conducted to show that the proposed algorithm outperforms the existing FSPC algorithm under a wide range of SNR values.

5.2 Scope for Future Work

The following aspects of the proposed algorithms can be further investigated in the future.

- The proposed algorithm has been tested under AWGN channel. It would be desirable to test the proposed algorithm with other types of channel environments, such as the one with Rayleigh fading model.
- It would be practically very useful to implement the proposed algorithm on a fixed point digital signal processor such as Texas Instruments TMS320C6482.

- In this thesis, only a simple thresholding method has been used. Further investigation could be undertaken on the way of choosing a proper threshold to further improve the performance of the proposed algorithm.

References

[1] S. Redl, M. Weber, and M. Oliphant, An introduction to GSM, The Artech House Publisher, 1995.

[2] J. S. Lee and L. E. Miller, CDMA Systems Engineering Handbook, Artech House Publishers, 1998.

[3] R. Prasad, W. Mohr, and W. Konhauser Editors, Third Generation Mobile Communication Systems, Artech House Publisher, 2000.

[4] Trillium Digital System, Inc. "Third Generation (3G) Wireless White Paper," Mar. 2000.

[5] H. Holma and A. Toskala, WCDMA for UMTS, Third Edition, John Wiley & Sons, Ltd. 2004.

[6] R. Tanner, and J. Woodard, WCDMA Requirements and Practical Design, John Wiley & Sons, Ltd, 2004.

[7] S. Yang, 3G CDMA2000 wireless system engineering, The Artech House Publisher, 2004.

[8] C. Smith, 3G wireless networks, second edition, McGraw-Hill Osborne Media, 2006.

[9] C. Chevallier, C. Brunner, A. Caravaglia, K. P. Murray, and K. R. Baker, WCDMA (UMTS) Deployment Handbook, Planning and Optimization Aspects, John Wiley & Sons, 2006.

[10] P. Lescuyer, UMTS Origins, Architecture and the Standard, Springer, 2004.

[11] T. Halonen, J. Romero, and J. Melero, GSM, GPRS, and EDGE Performance, John Wiley & Sons, Ltd. 2002.

[12] R. Rice and P. E. Green, "A communication technique for multipath channels," *Proc. IRE*, vol. 46, pp. 555-570, Mar. 1958.

[13] 3GPP TS 25.214, "Physical layer procedures (FDD)", version 6.0.0, 2006.

[14] J. G. Proakis, Digital Communications, Fourth Edition, McGraw-Hill Companies, Inc. 2001.

[15] R. C. Dixon, Spread Spectrum Systems, Wiley Publishers, 1976.

[16] R. Scholtz, "The spread spectrum concept," *IEEE Trans on Communications*, vol. 25, no. 8, pp. 748-755, Aug. 1977.

[17] A. Polydoros and C. Weber, "A unified approach to serial search spread-spectrum code acquisition – Part 1: General theory," *IEEE Trans on Communications*, vol. 32, no. 5, pp. 542- 549, May 1984.

[18] A. Polydoros and C. Weber, "A unified approach to serial search spread-spectrum code acquisition – Part II: a matched filter receiver," *IEEE Trans on Communications*, vol. 32, no. 5, pp. 550 - 560, May 1984.

[19] J. K. Holmes and C. C. Chen, "Acquisition time performance of PN spread spectrum systems," *IEEE Trans. Communications*, vol. COM-25, pp. 778-783, Aug. 1977.

[20] U. Madhow and M. Pursley, "mathematical modeling and performance analysis for a two-stage acquisition scheme for DS-CDMA," *IEEE Trans. Communications*, vol. 43, pp. 2511-2520, Sept. 2005.

[21] J. Kim, S. V. Saarin, M. Yasunaga, and H. Oh, "Robust noncoherent PN-code acquisition for CDMA communication systems," *IEEE Trans. Veh. Technol.*, vol. 50, no. 1, pp. 278-286, Jan. 2001.

[22] D. M. DiCarlo and C. L. Weber, "Multiple-dwell serial acquisition of direct-sequence code signals," *IEEE Trans. Communications*, vol. COM-31, pp. 650-659, May 1983.

[23] A. Polydoros and C. L. Weber, "Worst-case considerations for coherent serial acquisition of PN sequences," *in Proc. Nut. Telecommun. Conf.*, Houston, TX, pp. 24.6.1-24.6.5, Dec. 1980.

[24] L. Milstein, J. Gevargiz, and P. Das, "Rapid acquisition for direct sequence spread spectrum communications using parallel SAW convolvers," *IEEE Trans. Communications*, vol. COM-33, pp. 593-600, July 1985.

[25] A. Polydoros and C. L. Weber, "Rapid acquisition techniques for direct-sequence spread spectrum systems using an analog detector," *in Proc. Nut. Telecommun. Conf.*, New Orleans, LA, pp. A7.1.1-A7.1.5, Dec. 1981.

[26] J. K. Holmes and K. T. Woo, "An optimum PN code search technique for a given a priori signal location density," *in Proc. Nut. Telecommun. Conf.*, Birmingham, AL, pp. 18.6.1-18.6.5, Dec. 1978.

[27] A. Weinberg, "Search strategy effects on PN acquisition performance," *in Proc. Nut. Telecommun. Conf.*, New Orleans, LA, pp. F1.5.1-F1.5.5, Dec. 1981.

[28] R. L. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of spread-spectrum communications – A tutorial," *IEEE Trans. Communications*, vol. COM-30, pp. 855-884, May 1982.

[29] G. F. Sage, "Serial synchronization of pseudo-noise systems," *IEEE Trans. Communications*, vol. COM-12, pp. 123--127, Dec. 1964.

[30] J. K. Holmes and C. C. Chen, "Acquisition time performance of PN Spread spectrum systems," *IEEE Trans. Communications*, vol. COM-25, pp. 778-783, Aug. 1977.

[31] D. M. Dicarlo, "Multiple dwell serial synchronization of pseudo-noise signals," Ph. D. dissertation, University of Southern California, Los Angeles, May 1979.

[32] P. M. Hopkins, "A unified analysis of pseudo-noise synchronization by envelope correlation," *IEEE Trans. Communications*, vol. COM-25, pp. 770-778, Aug. 1977.

[33] D. M. DiCarlo and C. L. Weber, "Statistical performance of single-dwell serial-synchronization systems," *IEEE Trans. Communications*, vol. COM-28, pp. 1382-1388, Aug. 1980.

[34] D. M. Dicarlo and C. L. Weber, "multiple dwell serial search: Performance and application to direct sequence code acquisition," *IEEE Trans. Communications*, vol. COM-31, pp. 650-655, May 1983.

[35] W. Sheen, C. Tseng, and J. Ho, "Burst synchronization of slotted random access with preamble power ramping in the reverse link of CDMA systems," *IEEE Trans. Wireless Communications*, vol. 2, no. 5, Sept. 2003.

[36] S. Dick, "Random access channel preamble detection," US Patent WO 00/36761 A3, Nov. 2000.

[37] H. Sahlin, "Preamble detection," US patent, WO 2007/010331 A1, Jan. 2007.

[38] S. Sriram, K. Brown, et al, "A 64 Channel Programmable Receiver Chip for 3G Wireless Infrastructure," *IEEE 2005 Custom Integrated Circuits Conference*, pp. 59-62, Sept. 2005.

[39] TMS320TCI6488 Datasheet, www.ti.com, 2007.

[40] 3GPP TS 25.101, "User equipment radio transmission and reception (FDD)," v7.8.0, June 2007.

[41] 3GPP TS 25.214, "Physical layer procedures (FDD)", v7.5.0, May 2007.

[42] 3GPP TS 25.211, "Physical channels and mapping of transport channels onto physical channels (FDD)," V7.2.0, May 2007.

[43] 3GPP TS 25.213, "Spreading and modulation (FDD)," V7.2.0, May 2006.

[44] 3GPP TS 25.141, "Base station conformance testing", v7.9.0, Sept., 2007.