

**A Two-Stage Multi-Path Searcher  
for WCDMA RAKE Receivers**

**Min Yang**

**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 (Electrical Engineering) at  
Concordia University  
Montreal, Quebec, Canada**

**August 2007**

**© Min Yang, 2007**



Library and  
Archives Canada

Bibliothèque et  
Archives Canada

Published Heritage  
Branch

Direction du  
Patrimoine de l'édition

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file* *Votre référence*  
*ISBN: 978-0-494-34726-3*  
*Our file* *Notre référence*  
*ISBN: 978-0-494-34726-3*

**NOTICE:**

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

**AVIS:**

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

  
**Canada**

# ABSTRACT

## A Two-Stage Multi-Path Searcher for WCDMA RAKE Receivers

Min Yang

The performance of a Universal Mobile Telecommunication System (UMTS) base station RAKE receiver is largely dependent on the accuracy of the estimate of the time-delay of the different paths. A commonly employed technique in a RAKE receiver is to use a multi-path searcher (MPS) to find the number of different strong paths and the time-delay of these paths.

The drawback of the existing MPS algorithms for UMTS base-station RAKE receiver is that their computational complexity is high or the probability of detection is low. The full search with full contribution MPS gives the best result in terms of the detection probability, but its complexity is high. To reduce the complexity, a full search with partial contribution MPS has been proposed, but the detection probability of this algorithm is low. In order to solve this problem, in this thesis, a two-stage MPS, based on a dual-dwell technique is proposed. In the proposed algorithm, all the delay offsets in the search window are first evaluated for possible true paths by using only part of the slots in the correlation period. This helps to quickly narrow down the number of search candidates with only a small amount of computation needed. Then, in the second stage, the  $M$  strongest candidates from the first stage are further evaluated by using all the slots in the correlation period.

The simulation results show that the proposed algorithm has a detection performance very close to that of the conventional full search with full contribution MPS, while at the same time it has a substantially lower computational complexity.

## 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 advices, and care. Pursuing a Master's degree in Electrical and Computer Engineering was a long cherished dream for me but a great challenge as well, given my previous background. It would have been impossible for me to complete my graduate studies without their inspiration and support.

I would like to thank Dr. Jiajun Zhang who is with the Wireless Infrastructure Business Unit at Texas Instruments Inc. He provided me with many valuable suggestions during the course of my studies at Concordia University. I am especially grateful to him for spending a great deal of time in reviewing my thesis.

I would like to thank my sister, Haining Zhang, and Mr. Wan Feng and Mr. Qingling Zhang for their help during these years at Concordia University. Thanks are due to Mr. Shaikh Anowarul Fattach, and Ms. Celia Shahnaz for proofreading the final manuscript of my thesis.

I also would like to thank my friend, Howard Garrod, and his wife, Nancy, who have been so nice, generous and helpful to me throughout the time when I was in Belleville and later in Montreal. I am very grateful for their friendship and support throughout these years.

Finally, I would like to thank my beloved parents whose love, sacrifices and understanding have made all the things in my life possible.

To my uncle and aunt

and

my dearest parents

# CONTENTS

LIST OF FIGURES .....	viii
LIST OF TABLES .....	x
LIST OF SYMBOLS .....	xi
LIST OF ACRONYMS .....	xii
Chapter 1      Introduction.....	1
1.1 General.....	1
1.2 Literature Survey on Multi-path Searchers.....	6
1.3 Motivation, Scope and Organization of the Thesis.....	8
Chapter 2      The Fundamentals of a Multi-Path Searcher .....	11
2.1 Spreading and Modulation.....	11
2.1.1 Transmitter.....	11
2.1.2 Receiver... ..	12
2.1.3 Complex Representation .....	13
2.2 Digital Modulation at User Equipment.....	15
2.2.1 The Frame Structure of the Uplink Signals .....	15
2.2.2 Spreading.. ..	17
2.3 Digital Demodulation at Node-B.....	20
2.4 Principle of a Multi-path Searcher.....	21
2.4.1 Search Window.....	21
2.4.2 Correlation .....	22
2.4.3 Power Delay Profile.....	27
2.4.4 Threshold.. ..	28
2.5 Summary.....	29
Chapter 3      The Proposed Two-Stage Multi-path Searcher.....	31
3.1 The Background.....	31
3.2 First Stage Search .....	33
3.2.1 Method A.. ..	34
3.2.2 Method B.. ..	38
3.3 Second-Stage Search.....	45

3.3.1	Correlation and Accumulation.....	46
3.3.2	Thresholding .....	47
3.4	Some Practical Considerations for the Proposed Algorithm .....	49
3.5	Computation Complexities of the Proposed and Existing Algorithms.....	50
3.6	Summary.....	52
Chapter 4	Results on Computational Complexity and Simulation Study .....	54
4.1	Introduction.....	54
4.2	Computational Complexity of the Algorithms .....	56
4.3	Experimental Conditions for Simulation .....	57
4.4	Simulation Results .....	58
4.4.1	Scenario 1.....	59
4.4.2	Scenario 2.....	63
4.4.3	Results with Burst noise.....	66
4.5	Summary .....	71
Chapter 5	Conclusion and Future Work.....	73
5.1	Summary .....	73
5.2	Suggestions for Future Work.....	74
Appendix A.....		76
The Structure of Uplink DPDCH and DPCCH.....		76
Appendix B.....		78
Pilot Bit Patterns for Uplink DPCCH .....		78
References.....		80

## LIST OF FIGURES

Figure 1.1 A base station and user equipments .....	3
Figure 2.1 Block diagram of a transmitter .....	12
Figure 2.2 Block diagram of a receiver .....	13
Figure 2.3 Frames and slots .....	16
Figure 2.4 Frame structure for the uplink DPDCH and DPCCH .....	17
Figure 2.5 Code tree of the OVSF code.....	17
Figure 2.6 Uplink spreading .....	18
Figure 2.7 Illustration of the channelization process .....	19
Figure 2.8 Descrambling and despreading.....	21
Figure 2.9 Correlation process.....	26
Figure 2.10 An example of coherent and non-coherent accumulations.....	27
Figure 2.11 An example of PDP .....	28
Figure 3.1 An illustration of the PDP values after slot 0 using Method A.....	36
Figure 3.2 An illustration of the PDP values after slot 1 using Method A.....	36
Figure 3.3 An illustration of the PDP values after slot 2 using Method A.....	37
Figure 3.4 The timing relationship between the two stages using Method A.....	38
Figure 3.5 An illustration of the PDP values after slot 0 using Method B.....	40
Figure 3.6 An illustration of the PDP values after slot 1 using Method B.....	41
Figure 3.7 An illustration of the PDP values after slot 4 using Method B.....	41
Figure 3.8 An illustration of the PDP values after the first 10 slots using Method B.....	42
Figure 3.9 An illustration of the PDP values of the first stage using Method B.....	43
Figure 3.10 The timing relationship between the two stages using Method B.....	45
Figure 4.1 An illustration of Scenario 1 .....	55
Figure 4.2 An illustration of Scenario 2 .....	55
Figure 4.3 Probability of detection in Scenario 1 .....	61
Figure 4.4 Probability of false alarm for Scenario 1.....	62
Figure 4.5 Probability of detection in scenario 2.....	64
Figure 4.6 Probability of false alarm for Scenario 2.....	65



Figure 4.7 Probability of detection for scenario 1 with burst noise.....	67
Figure 4.8 Probability of false alarm for Scenario 1 with burst noise .....	68
Figure 4.9 Probability of detection for Scenario 2 with burst noise.....	69
Figure 4.10 Probability of false alarm for scenario 2 with burst noise.....	70

## LIST OF TABLES

Table 1 Combinations of the search mode and contribution mode.....	31
Table 2 PDP values of the first stage using Method A.....	37
Table 3 PDP values of the first stage using Method B.....	44
Table 4 Comparison of the computational complexity of various MPS algorithms under Scenario 1 .....	56
Table 5 Comparison of the computational complexity of various MPS algorithms under Scenario 2 .....	57
Table 6 Simulation parameters.....	58
Table 7 Probability of detection for Scenario 1 .....	60
Table 8 Probability of false alarm for Scenario 1 .....	62
Table 9 Probability of detection for Scenario 2 .....	63
Table 10 Probability of false alarm for Scenario 2 .....	64
Table 11 Percentage improvement from Scenario 1 to Scenario 2 in the detection performance of the two algorithms .....	65
Table 12 Probability of detection for Scenario 1 with burst noise.....	66
Table 13 Probability of false alarm for Scenario 1 with burst noise.....	68
Table 14 Probability of detection for Scenario 2 with burst noise.....	69
Table 15 Probability of false alarm for Scenario 2 with burst noise.....	71

## LIST OF SYMBOLS

$W$	Search window size
$N_p$	Number of pilot bits
$\tau$	Delay in number of chips
$R_i$	Received signal at chip $i$
$C_i$	Locally generated signal at chip $i$
$M$	Number of offsets selected for the second stage
$N$	Number of slots in a correlation period
$\theta$	Detection threshold
$PDP[i]$	Signal power at the delay offset $i$

## LIST OF ACRONYMS

2G	Second generation
3GPP	Third generation partnership project
ACK	Acknowledgement
AI	Acquisition indication
CDMA	Code division multiple access
DPCCH	Dedicated physical control channel
DPDCH	Dedicated physical data channel
EDGE	Enhanced data rates for global evolution
FD	Finger de-spreader
FDD	Frequency division duplex
FSFC	Full search with full contribution
FSPC	Full search with partial contribution
GPRS	General packet radio service
GSM	Global system for mobiles
ITU	International telecommunication union
LOS	Line-of-sight
MPS	Multi-path searcher
MRC	Maximum ratio combiner/combining
PCS	Personal communication service
PD	Preamble detection
PDP	Power delay profile
RACH	Random access channel
SF	Spreading factor
TDD	Time division duplex
TD-SCDMA	Time division-synchronous code division multiple access
TFCI	Transport format combination indicator
UE	User equipment
UMTS	Universal mobile telecommunication system
WCDMA	Wideband code division multiple access

# **Chapter 1 Introduction**

## **1.1 General**

With the maturity of the TDMA-based global system for mobile (GSM) communications [1] and the code division multiple access (CDMA) based IS-95 [2], mobile telephone communication has become ubiquitous around the world. These second generation (2G) wireless telephone technologies have made the global personal communication service (PCS) a reality. However, these PCS services are still very much limited to voice services. To meet the increasing demands for data services, such as internet browsing, enhanced data rates for global evolution (EDGE) and general packet radio service (GPRS) [3] have been developed and added to the existing wireless communication systems, referred to as 2.5G wireless technology. While the 2.5G systems make the data services possible, they are still very slow, since they can support data rates only up to 384 Kbps. To meet the growing demands for even higher data rates, a better spectrum efficiency, and increased use of the internet and global mobility, the third-generation (3G) wireless telephone technologies have been developed [4]. The 3G wireless systems promise a variable bit rate of up to 2 Mbps. Compared to the 2G and 2.5G systems, the 3G systems can support a large number of voice and data users at a higher data rate with 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 different 3G wireless systems [5].

According to the International Mobile Telecommunications-2000 (IMT-2000), five standard radio air interfaces have been defined, among which the three most popular ones are wideband CDMA (WCDMA) [6], CDMA2000 [8], and TD-SCDMA [9]. The International Telecommunication Union (ITU) defined the demands for 3G mobile networks with IMT-2000 standard. An organization called the Third Generation Partnership Project (3GPP) has continued the work by defining a mobile system that fulfills the IMT-2000 standard. This system is called the Universal Mobile Telecommunication System (UMTS) [10], [11]. The UMTS uses WCDMA as the underlying air interface.

In a mobile communication system, the base station, also known as node-B in UMTS, transmits as well as receives signals at the same time, so does the mobile handset which is also known as user equipment (UE). To facilitate the discussions and avoid possible misunderstanding; we define the following terminologies according to the industry conventions. The downlink, or the forward link, is defined as the direction from node-B to the UE; the uplink direction, or the reverse link, is defined as the direction from the UE to the node-B. A radio link between the node-B and the UE comprises both the forward link and the reverse link. These concepts are illustrated in Figure 1.1.

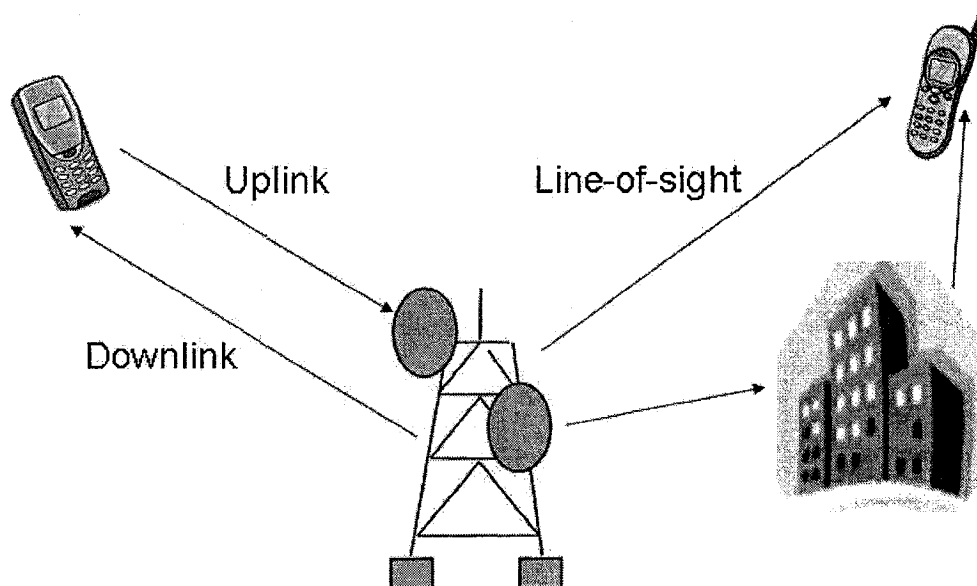


Figure 1.1 A base station and user equipments

The UMTS systems are further distinguished from each other according to the way of separation of their uplink and downlink signals. In the frequency-division duplexing (FDD) mode, the downlink and the uplink use different carrier frequencies, while transmitting at the same or different times. In the time-division duplexing (TDD) mode, on the other hand, the uplink and the downlink use the same frequency, but they can only take turns to transmit.

In wireless channels, a signal may go through different fading paths. Since each path has a different length and the signals travel at the same speed, signal arrival times for the paths differ. This is called multi-path propagation. These paths differ from each other in length of the route and channel impairments even though they propagate at the same transmission speed. Thus, the resultant signals reach the receiver with different delays

and different strengths. The simplest solution for the UE receiver would be to detect the strongest path and use its corresponding delay offset to descramble and despread the received signal. However, it has been found to be much more beneficial to combine these signals from the multiple paths. Receivers employing the multi-path combining technique is typically called RAKE receivers [12], [13]. In a RAKE receiver, a path that is used for the combination is often referred to as a finger. In a UMTS system, both the node-B receiver and the UE receiver are RAKE receivers.

To establish a radio link between a node-B and a UE, the UE needs to follow the procedures defined in the 3GPP standard [14] to send a random access channel (RACH) message to the node-B to request a call. The node-B has a preamble detector (PD) which 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 messages and radio link setup. The PD provides an estimate of the location of the UE in the cell. The MPS 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 despreader (FD), on the other hand, is responsible for descrambling and despreding the fingers, and then combining them to provide the final soft symbols for further symbol rate (SR) processing. The widely-used technique for finger combining is called maximal



ratio combining (MRC) [15]. The symbol rate processing is responsible for decoding the channel coding and producing the final binary values of the information bits.

In order for the FD to correctly descramble the antenna signal of a given path, the propagation delay of that path must be calculated accurately. The propagation delay of a path is also referred to as the position of the finger in a RAKE receiver. The functional block of a RAKE receiver that is responsible for finding all the finger positions is typically called a multi-path searcher (MPS). The propagation delay or the finger position is expressed in terms of number of chips with respect to the origin of the search window, which is derived from the initial estimate of the UE's position in the cell. The position of UE, in turn, is determined from the corresponding successful preamble detection. The nominal size of the search window expressed in terms of the number of chips is a system parameter for Node-B, this parameter is typically depends on the possible propagation delays and the Doppler effects. Furthermore, a real MPS has typically two different search phases. The initial search phase is usually called wide search, in which the size of the search window size is several times larger than the nominal size. This wide search is carried out during the initial radio link setup to find the finger positions. The second search phase is the so-called normal search phase in which the nominal window size is used. The normal search is used to track the changes of the finger position during the remainder of the radio link connection.

The multi-path searcher of a RAKE receiver faces many technical challenges. First of all, the wide search is computationally very intensive due to the large sized of the search window, making the entire MPS computationally very expensive for the receiver.

Secondly, the WCDMA channels are fast fading channels because of the Doppler effect, and the scattering, reflection and deflection of the magnetic waves. As a result, finger positions of a UE are constantly changing. Thus, the MPS must frequently update the finger positions by tracking these changes, thereby making the MPS even more computationally intensive.

## **1.2 Literature Survey on Multi-path Searchers**

In the past decade, many research efforts have been devoted to improving the performance of the MPS [16]-[38]. Some of these efforts have focused on the performance analysis and parameter optimizations including the selection of the threshold values adaptively [16]-[28]. Some other efforts have been devoted to the study of the finger placement and management [29]-[33]. Semiconductor vendors have been working hard to deliver solutions to address the issue of overwhelming computational complexity of WCDMA RAKE receivers. Freescale Semiconductor has delivered a StarCore based DSP solution for mobile handsets [34]. Recently, Texas Instruments has delivered OMAP DSP based solution for mobile handsets and TCI based ASIC plus DSP solution for base stations [35]. Most of the published literature has been focused on solution pertaining to mobile handsets or user equipments, but not many on the base station or wireless network infrastructure solutions. In other words, most of the solutions are on the downlink and not many on the uplink.

There are many differences between uplink MPS and downlink MPS. A downlink MPS handles multi-path search only for one UE, whereas the uplink MPS needs to find and track the multi-path for many UEs. Depending on the number and the size of the cell

within a Node-B, the MPS of a base station could handle more than one hundred UEs at a time. This translates into many implementation challenges such as the memory used to buffer the antenna data, and the DSP CPU cycles used to handle the processing. For example, the non-deterministic nature of the amount of computation required by the techniques proposed in [39] and [40] for the processing of the second stage becomes even more so in the worst case when all the UEs in the cell have the same transmit offset and have a very large number of the candidate offsets to process. The system designer has to employ the DSP resources for the worst case, making the system less efficient. The frame structures are also different for the uplink and downlink. Multi-path searchers for both uplink MPS and downlink use the pilot-assisted approach [41] for detecting and tracking the multi-paths. However, the pilot symbols are mapped differently in the two links. The downlink uses a common pilot channel (CPICH) [41] to allow for all the UEs in the cell to detect their multi paths. There is only one primary CPICH (P-CPICH) in the entire cell. Each slot of the P-CPICH channel carries the same 20 bits of the pilot symbols. On the other hand, for the uplink each UE uses its dedicated physical control channel (DPCCH) [41] to carry the pilot symbols. The number of pilot symbols varies between 3 and 8 depending on the slot format. Because of the differences in the frame structures, a scheme that works for the downlink may not necessarily work for the uplink. Moreover, the base station tends to suffer more from the multi-user interferences.

Kim et al. [39] have applied the double-dwell technique in MPS for downlink mobile handset RAKE receiver, where the offsets with the correlation results exceeding a certain predefined threshold are further examined for correct path in the second stage. Grayver et

al. [40] have tried to apply the double-dwell idea in a downlink mobile handset RAKE receiver, and further expanded this approach to a three-stage one by adding the so called detection stage in addition to the initial search stage and the verification stage. However, the problem with methods given in [39] and [40] is that the processing load of their second stage is data dependent not deterministic, thus making the ASIC hardware implementation or DSP implementation more challenging. To the best of the author's knowledge, so far, no attempt has been made to apply the double-dwell technique reported in [39] to the MPS of an uplink base station RAKE receiver.

### **1.3 Motivation, Scope and Organization of the Thesis**

It is well known that more than ninety percent of the cost of a wireless network infrastructure is carried by its Node-B. Inside a Node-B, the most costly component is the RAKE receiver of the physical layer baseband processing unit. A RAKE receiver is typically composed of a DSP farm that provides the largest processing power for the preamble detection, multi-path searcher (MPS) and finger despreading. Generally, the MPS is one of the most computationally intensive modules [7]. Therefore, developing an efficient MPS algorithm with a practical hardware implementation scheme for the base station is essential for a cost effective 3G UMTS systems.

The conventional full search MPS with full contribution (FSFC) [19] from all the slots of the correlation period provides the best performance in terms of the path detection accuracy, but it is also computationally the most intensive one. The FSFC method employs a single-stage correlation process, where the power delay profile values of all the possible locations within the search window are calculated simultaneously, and

become available at the same time for the processing of the path detection by the time the correlation operation is completed. In the detection process, only 5-10% of the delay paths are determined to be the true paths.

The problem with such single-stage path searchers is that they attach equal importance to all the offsets in the search window, even though the possible strong paths may exist only at a few offset locations, thus effectively wasting the computation resources on unlikely locations. This deficiency becomes even more pronounced specially when the search window size becomes large.

Bejjani, Bouquier and Cacqueray [36] have attempted to reduce the computational complexity of the MPS by introducing a scheme of full search with partial contribution (FSPC) from the slots, where only a small number of the slots in a frame are used in the correlation period. This reduces the computational complexity. But the detection accuracy is much lower than that of the FSFC technique [19], the conventional full search MPS.

This study is an attempt to develop a new MPS algorithm suitable for base station RAKE receivers. The objective of the algorithm is to provide a detection accuracy close to that of the full search MPS but with a substantially reduced computational complexity. The main idea of the new MPS algorithm is to adopt the double-dwell technique initially proposed by Pursley et al. [37] for developing an efficient PN-code acquisition scheme. The principle of the double-dwell technique is that if the correlation between the received

sequence and the locally generated sequence exceeds the given threshold, a second test using parameters with tighter bounds is performed. This principle, along with the ideas of using interleaved offsets, and a scheme of hybrid fixed and adaptive thresholding, is applied for developing the proposed uplink MPS algorithm.

This thesis is organized as follows: The basic concepts of uplink frame structure, correlation, coherent accumulation, non-coherent averaging, search window, and power delay profile related to uplink MPS are introduced in Chapter 2. In Chapter 3, a two-stage MPS algorithm, with an objective of achieving a high detection accuracy and low computational complexity, is developed. Two scenarios of selecting the number of slots used in the first stage of the algorithm are discussed. In Chapter 4, a simulation study of the proposed algorithm, and the full search algorithms with full and partial contributions from the frame slots, is undertaken. These results are analyzed to demonstrate the effectiveness of the proposed algorithm. Finally, Chapter 5 concludes the thesis by highlight the significant contribution of this thesis and by providing some suggestion for future investigation.

## **Chapter 2     The Fundamentals of a Multi-Path Searcher**

There is a considerable amount of material available in the literature concerning the principle of WCDAM and UMTS systems [6], [7], [11]. In this chapter, however, we restrict ourselves to the discussions of the part of the material constituting a background preparation for the working principle of a multi-path searcher. To simplify the discussions, we also assume the sampling rate at the receiver to be the same as the chip rate. Therefore, the word “chip” and the word “sample” are used interchangeably most of the time in the following discussion.

### **2.1 Spreading and Modulation**

#### **2.1.1 Transmitter**

According to the 3GPP standard, after the symbol rate processing [42], and the chip rate processing [41], [43], the user data is to be digitally modulated by using quadrature phase shift keying (QPSK) modulation, and then further modulated by a radio carrier as shown in Figure 2.1.

The chipping sequence collectively collects the channelization code and scrambling code. Multiplying the channelization code sequence with the repeated user data sequence is the

process of digital modulation. It is also the process of spreading where each user data bit is repeated SF times, where SF is the spreading factor, before being multiplied by the chipping sequence. After this process, the original digital signal becomes a baseband analog signal. In this specific case, the resulting signal is also a spread spectrum signal. This signal is then further modulated by the radio carrier signal to produce the final signal to be transmitted.

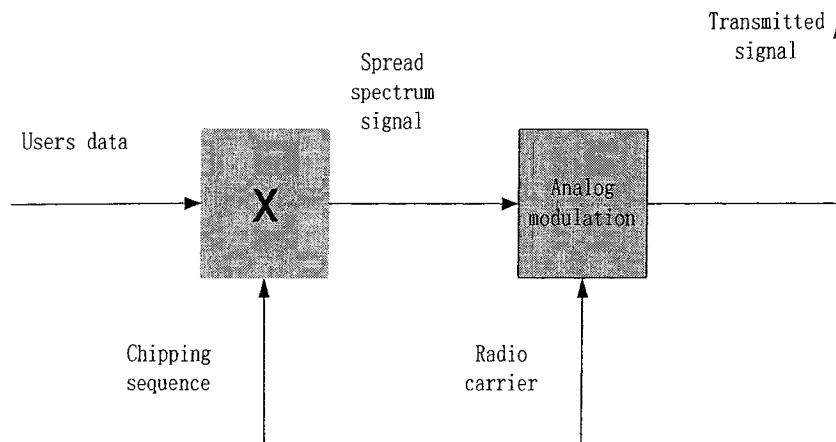


Figure 2.1 Block diagram of a transmitter

### 2.1.2 Receiver

The receiver is much more complicated than the transmitter. Even though the radio signal demodulation part is not complicated, the reverse process of the channelization and scrambling is. The functional block diagram of a receiver is shown in Figure 2.2. Each of the original user data bits is spread over many chips in the transmitter. Consequently, the receiver has to reverse the spreading process by integrating or accumulating the products of the demodulated sequence and the chipping sequence, This process is accomplished by a correlator at the receiver [44]. The output of the accumulator is a non-binary value, a



decision block is needed to decide whether this value is “1” or “0”. It is worth noting that in a real system, this decision block is not needed. This is because the subsequent processing module is the symbol rate processing that performs the reverse process of the forward error coding (FEC), and prefers non-binary values. However, such a treatment is beyond the scope of this thesis.

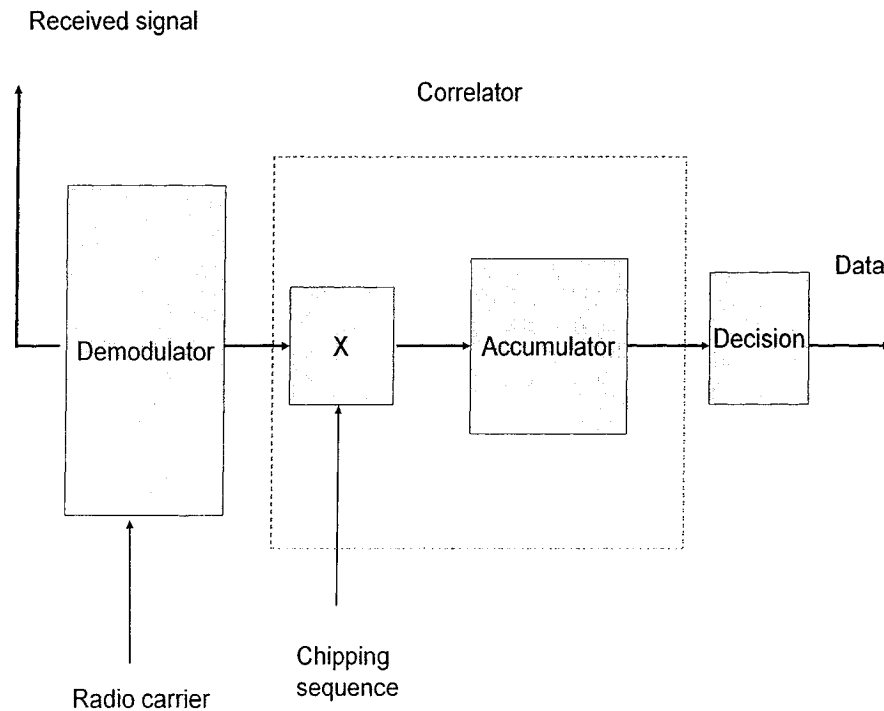


Figure 2.2 Block diagram of a receiver

### 2.1.3 Complex Representation

The above mentioned transmitting and receiving processes do not necessarily involve manipulations of complex numbers. However, using a complex representation would be much easier for discussion and processing, since the digital modulation used is the QPSK where the user data bits, i.e., the information bits, are modulated by in-phase ( $I$ ) and

quadrature ( $Q$ ) components. For the sake of completeness, the following mathematical details are adopted from [45].

The general form of the passband signal is

$$s(t) = A(t) \cos[2\pi f_c t + \theta(t)] \quad (2.1)$$

Here  $A(t)$  is the amplitude of  $s(t)$ , and  $\theta(t)$  is the phase. Varying either or both can be used to carry information. Equivalently, (2.1) can be expanded into

$$s(t) = m_i(t) \cos[2\pi f_c t] - m_q(t) \sin[2\pi f_c t] \quad (2.2)$$

where

$$m_i(t) = A(t) \cos[\theta(t)] \quad (2.3)$$

and

$$m_q(t) = A(t) \sin[\theta(t)] \quad (2.4)$$

are the in-phase and quadrature modulations, respectively.

Define the complex baseband signal (using “ $\sim$ ” to denote a complex function),

$$\tilde{m}(t) = m_i(t) + jm_q(t) \quad (2.5)$$

then the complex passband signal is defined to be

$$\tilde{s}(t) = \tilde{m}(t)e^{j2\pi f_c t} = s(t) + j\tilde{s}(t) \quad (2.6)$$

where the real part of  $\tilde{s}(t)$  is the actual transmitted signal  $s(t)$  of (2.1), and the imaginary part of  $\tilde{s}(t)$ ,

$$\tilde{s}(t) = m_i(t) \sin[2\pi f_c t] + m_q(t) \cos[2\pi f_c t] \quad (2.7)$$

is the Hilbert transform of  $s(t)$ . Thus,

$$s(t) = \text{Re } \tilde{s}(t) = \text{Re } \tilde{m}(t)e^{j2\pi f_c t} \quad (2.8)$$

The baseband signal information is carried by the  $I$  component given by (2.3) and the  $Q$  component given by (2.4). Both the scrambling and the descrambling are complex-number operations in the 3GPP standard [41], [43].

## 2.2 Digital Modulation at User Equipment

As stated earlier, the purpose of digital modulation is to convert the digital user data sample stream into a baseband analog signal. In order to have the features of power control, pilot-assisted synchronizations, etc, the physical layer also needs to send control information in addition to the user data information. In order to understand as to how these two pieces of information are organized during the digital modulation process, one needs to first understand the uplink frame structure defined by the 3GPP.

### 2.2.1 The Frame Structure of the Uplink Signals

The user data sample stream is divided into radio frames. Each frame is further divided into 15 slots as shown in Figure 2.3. After the spreading, each slot contains 2560 chips or samples, thus the entire frame consists of 38400 chips. In terms of the duration in time, one radio frame is 10ms long, and one slot has a length of 0.667 ms.

The user data is carried in or mapped to the so called dedicated physical data channel (DPDCH) [41]. A dedicated channel means that it is exclusively used for a given user during the lifespan of the radio connection. Here, channel is a logical entity defined in the 3GPP physical layer for the ease of discussions. The physical layer control information such as the pilot bits, transport format combination indicator (TFCI) bits, etc. is mapped

to the dedicated physical control channel (DPCCH). According to the standard [43], the DPDCH is mapped to the I branch and the DPCCH to the Q branch.

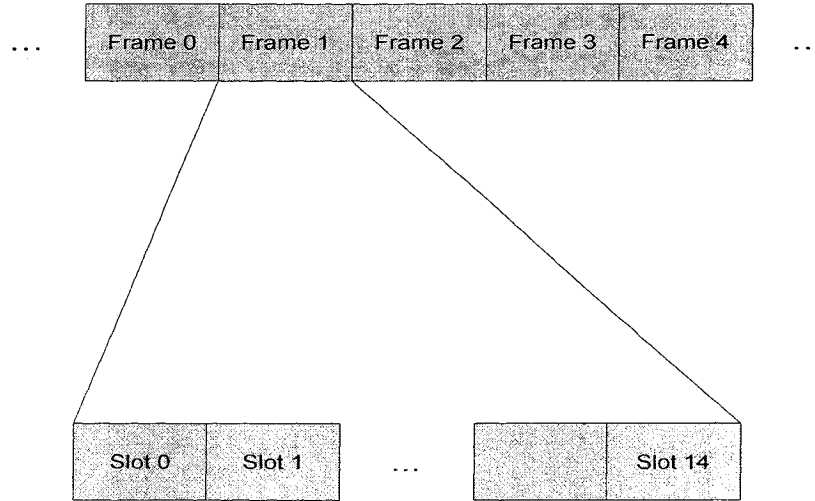


Figure 2.3 Frames and slots

As shown in Figure 2.4, the number of bits that can be mapped onto one slot of the DPCCH is fixed to 10, since DPCCH has a fixed spreading factor of 256. The number of bits that can be mapped onto one slot of the DPDCH, however, depends on the spreading factor of the UE. The spreading factor (SF) varies between 2 and 256 [41]. In the Appendix A, Table A.1 gives the exact number of bits and the corresponding spreading factor of the uplink DPDCH, and Table A.2 gives the exact number of bits for the different fields of the uplink DPCCH. The slot format number is chosen dynamically by the physical layer depending on the number of bits to be transmitted [7]. Once the number of pilot bits  $N_{\text{pilot}}$  in the DPCCH is specified based on the slot format number, one can find the pilot patterns of each slot in the uplink DPCCH through Table B.1 or Table B.2 in the Appendix B.

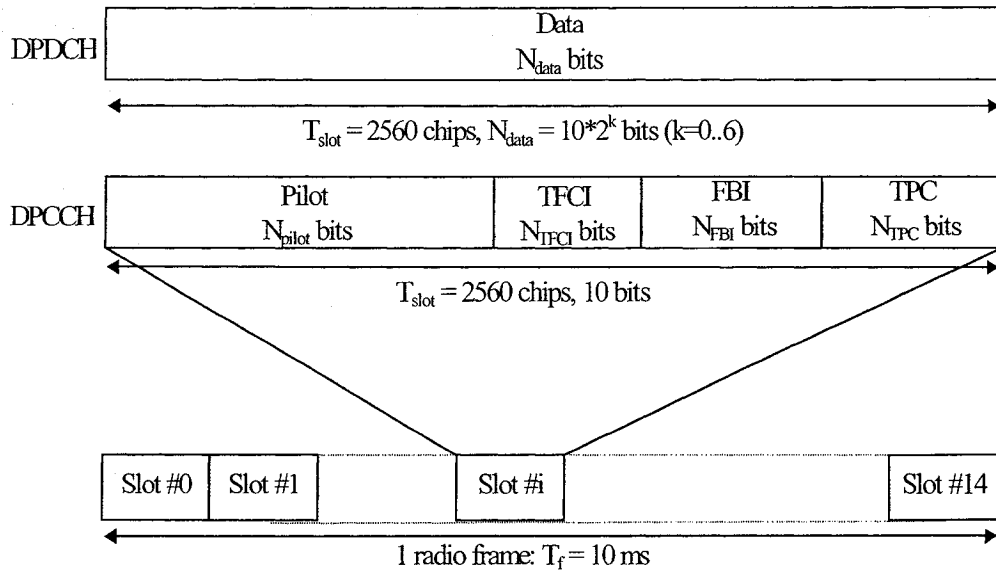


Figure 2.4 Frame structure for the uplink DPDCH and DPCCH

### 2.2.2 Spreading

The channelization code is an orthogonal variable spreading factor (OVSF) code. The code sequence is orthogonal to each other within the set as shown in Figure 2.5.

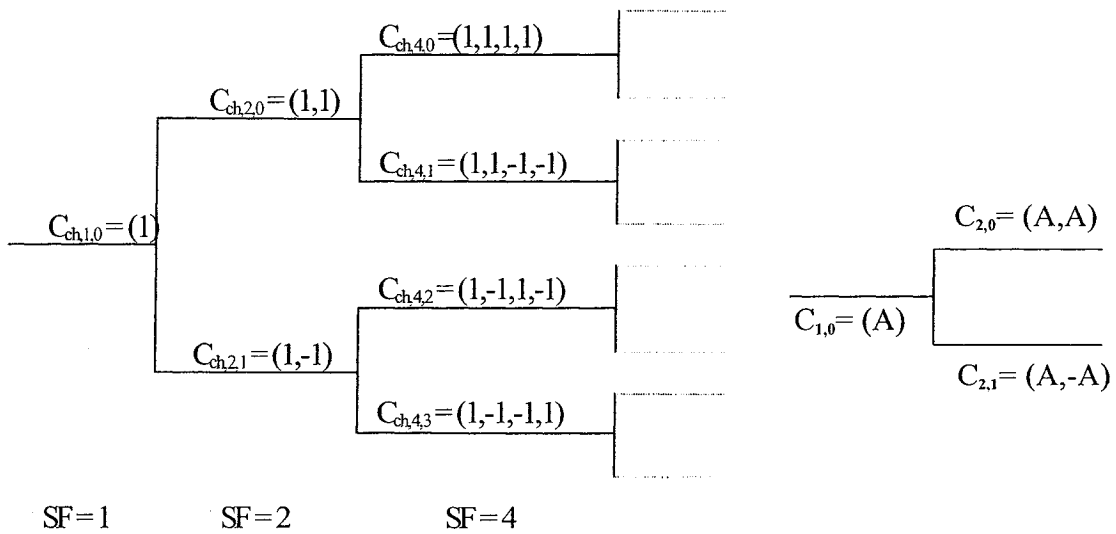


Figure 2.5 Code tree of the OVSF code

The spreading process is divided into two steps. The first one is called channelization and the second scrambling as shown in Figure 2.6.

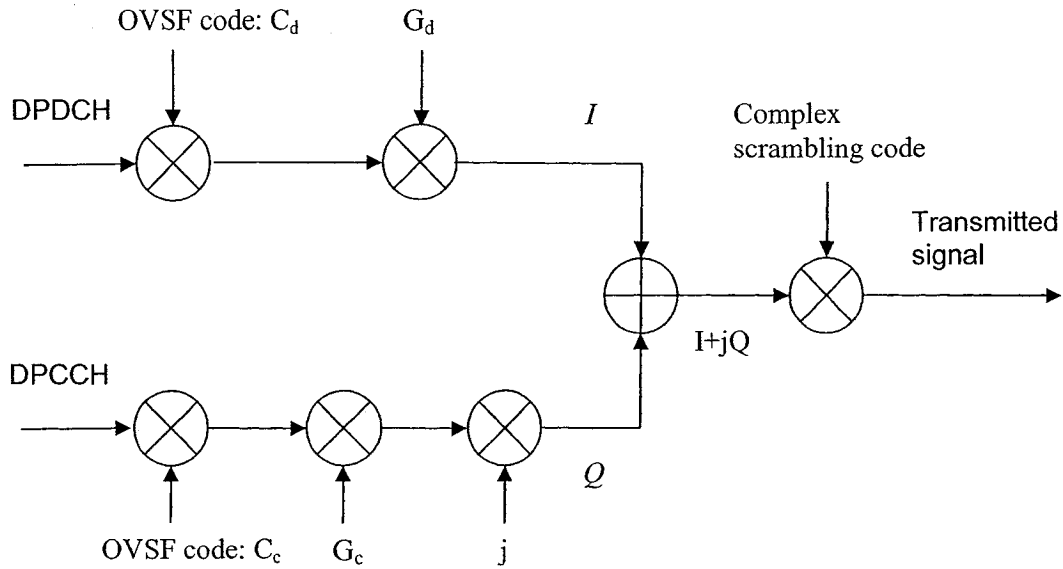


Figure 2.6 Uplink spreading

The channelization process is the process where each of the user data bits is repeated SF times, and then multiplied with the corresponding OVSF code. For example, if the DPDCH data bits are  $\{1, -1, 1, 1, 1, \dots\}$  and the OVSF code is  $\{1, 1, -1, -1\}$ , implying the  $SF = 4$ , then the resulting bits after the channelization become  $\{1, 1, -1, -1, -1, -1, 1, 1, 1, -1, -1, 1, 1, -1, -1, 1, 1, -1, -1, \dots\}$ .

This idea of channelization process is also illustrated in Figure 2.7. However, it should be pointed out that according to the standard, the user data “1” is mapped into “-1”, and “0” to “1” before the multiplication operations. As can be seen, the channelization process effectively widens the bandwidth, which is expected.

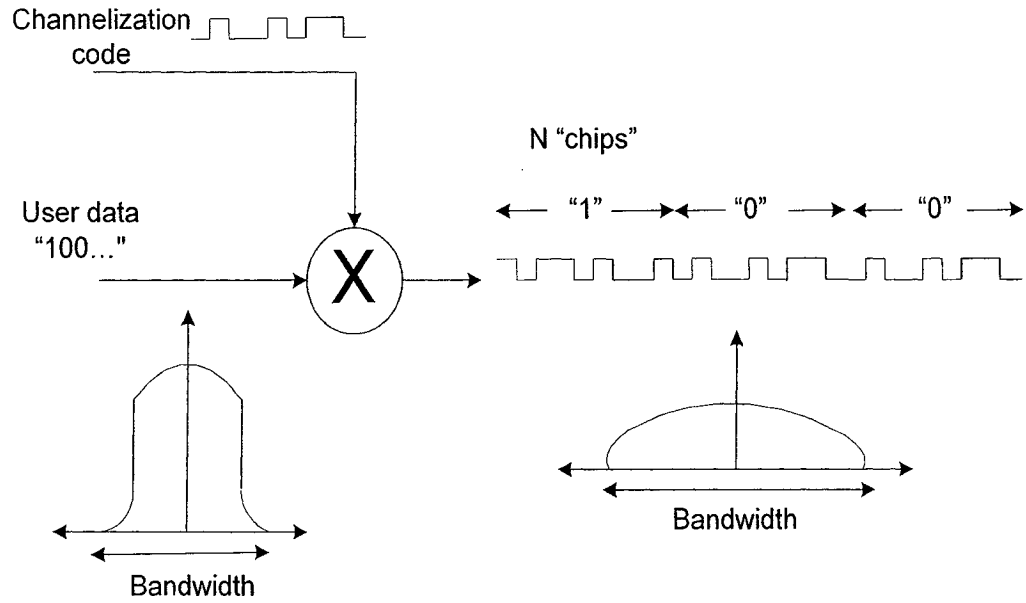


Figure 2.7 Illustration of the channelization process

The DPDCH and DPCCH use different OVSF codes, and they also have different gain values of  $G_d$  and  $G_c$ , respectively as indicated in Figure 2.6.

After the channelization, the scrambling operation is performed on both of the I and Q branches. According to the standard, the scrambling code is a sequence of complex values [43]. The basic operation of the scrambling is a complex number multiplication given by

$$(I_{sc} + jQ_{sc})(I_{data} + jQ_{data}) = (I_{sc}I_{data} - Q_{sc}Q_{data}) + j(I_{sc}Q_{data} + Q_{sc}I_{data}) \quad (2.9)$$

where  $(I_{sc}, Q_{sc})$  is the scrambling code and  $(I_{data}, Q_{data})$  is the data after the channelization.

The result of the spreading is a sequence of complex values. The pairs of the  $(I, Q)$  values are then mapped onto points on the QPSK constellation map before carrying out the analog modulation.

## 2.3 Digital Demodulation at Node-B

At the receiver side, after the analog demodulation to remove the radio carrier, the baseband signal is restored. The digital demodulation decodes the user information bits from the baseband signal. After the constellation de-mapping, the  $I$  and  $Q$  values are restored. These  $I$  and  $Q$  sequence is also loosely called the antenna data in the following discussions.

The next step to descramble the  $I$  and  $Q$  sequence using the complex conjugate of the original scrambling code. However, the problem is that the receiver does not know the starting position of the scrambling code in the antenna data. This is caused by the propagation delay due to the time it takes for the signal to travel from the UE to the Node-B. The objective of the MPS is to find the starting position of the scrambling code. Once the starting position is known, the descrambling is simply a complex multiplication between a locally generated sequence and the antenna input sequence.

After the descrambling, the resulting sequence needs to be de-spread or de-channelized. This is a process reverse of that of the channelization. Conceptually, the result of the de-spread should give back the user data bits as shown in Figure 2.8.



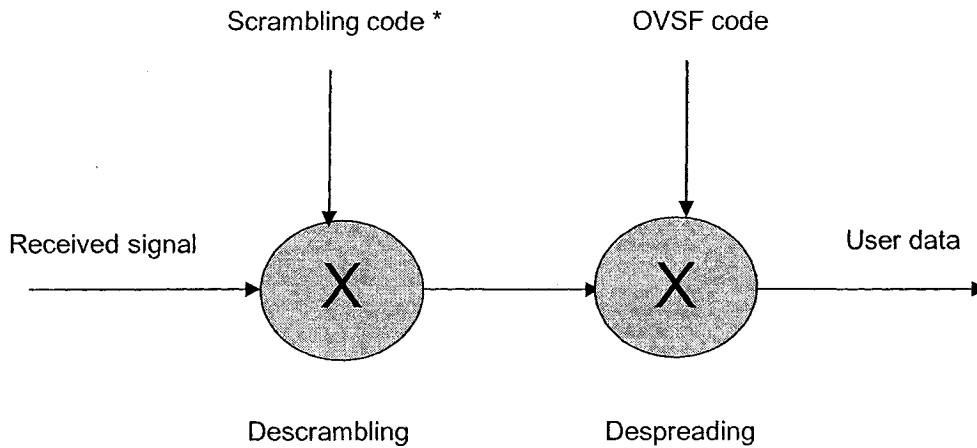


Figure 2.8 Descrambling and despreading

However, in a real system, the process is much more complicated due to the channel impairments and multipath propagations.

## 2.4 Principle of a Multi-path Searcher

The objective of an MPS is to find the starting position of the scrambling code contained in the antenna input sequence. This is achieved by exploiting the orthogonality of the OVSF code, the pseudo orthogonality of the scrambling code and the known information about the pilot bits.

### 2.4.1 Search Window

The starting point of the scrambling code at the receiver is determined by the overall propagation delay. The delay changes as the propagation path changes. Therefore, there

is an uncertainty about the delay. Consequently, there is an uncertainty about the starting point of the scrambling code. At the system level, one can be sure only about the range of such an uncertainty. Within this range, the MPS assumes every one of the possible delays, and thus will exhaustively tests each of the hypotheses. This range is called the search window. The delays are expressed in terms of the number of chips, thus the window size is also expressed in number of chips.

In a real system, the origin of the search window is provided by the result of the preamble detection. Then, the search window is used to cover the range of the region of the time delay uncertainty between the first arrived delay path and the last arrived delay path. In a WCDMA system, the chip duration at 3.84 Mcps is  $0.26 \mu s$ , which allows the radio wave

to travel  $\frac{3 * 10^8 (m/s)}{3.84 * 10^6 (chip/s)} = 78.125$  meters during a chip period. Therefore, a delay

uncertainty range of  $20 \mu s$  would result in a search window of about 80 chips.

## 2.4.2 Correlation

From the signal processing point of view, and if the antenna sequence is viewed as a random process, the path search processing is the operation of an autocorrelation of the random process at two different times. The received signal is a time delayed replica of the original transmitted signal, the receiver can locally generate the “replica” of the transmitted signal, since the pilot bits, the OVSF code and the scrambling code are all known. These three pieces of information together are referred as the locally generated CODE. This CODE is illustrated in Figure 2.2 as the chipping sequence.

Since the received signal is a spread spectrum signal, that is, the pilot bits are spread by a factor of 256, it is necessary to de-spread the signal. The accumulator or the integration shown in Figure 2.2 is used to reverse the spreading process done in the transmitter. Moreover, the purpose of the correlation is not to recover the pilot bits since they are already known to the receiver; rather it is used to find out as to which hypothesis about the actual delay is correct. If the hypothesis is wrong, then the products between the CODE and the antenna sequence will cancel one another at the accumulator due to the pseudo orthogonality of the scrambling code. The accumulator produces a peak value if the hypothesis is correct. This is the fundamental principle used in a pilot-assisted MPS. This kind of correlation approach is also sometimes called the sliding correlator approach [23], since the operation effectively slides the locally generated signal over the input signal with a range given by the search window.

#### **A. Coherent Accumulation**

The correlation of the complex valued scrambling code and the complex valued antenna sequence results in complex values. Therefore, the accumulator in Figure 2.2 is actually a vector accumulator. It adds up the contributions for both the I and the Q branches, respectively. This accumulation is also called coherent accumulation.

The length of the coherent accumulation could be one slot or a few slots depending on the design of the MPS. However, in real DSP or ASIC implementation, this coherent accumulation may be achieved through multiple smaller correlation intervals. For example, in the TCI110 ASIC from Texas Instruments [35], the correlation is done

through multiple iterations, where each iteration correlates only 64 chips. Thus, if the coherent accumulation length is one slot, it would take 40 iterations to complete it.

Mathematically, the coherent accumulation operation can be expressed as follows. The received baseband signal is a weighted sum of the various delayed versions of the original baseband signal,

$$R_i(n) = \sum_{l=1}^L S_{i-\tau_l}(n) \alpha_l(n) + m_l(n) \quad (2.10)$$

where  $R_i(n)$  is the  $i$ -th sample of slot  $n$ ,  $S_i(n)$  is the original baseband signal of slot  $n$ ,  $S_{i-\tau_l}(n)$  is  $S_i(n)$  delayed by  $\tau_l$  chips,  $L$  is the number of paths contributing to the received signal, and  $\alpha_l(n)$  is the channel coefficient of path  $l$  at slot  $n$ , and  $m_l(n)$  is a zero-mean complex Gaussian noise with a variance  $\sigma^2$ .

The locally generated signal is given by

$$C_i(n) = S c_i^*(n) C c_i(n) P_i(n) \quad (2.11)$$

where  $C_i(n)$  is the  $i$ -th sample of slot  $n$ ,  $S c_i^*(n)$  is the complex conjugate of the  $i$ -th sample of the scrambling code at slot  $n$ ,  $C c_i(n)$  is the  $i$ -th sample of the OVFSF code at slot  $n$ , and  $P_i(n)$  is  $i$ -th sample of the spread pilot pattern at slot  $n$ .

Considering that the coherent length is 1 slot, and only  $N_p$  pilot symbols were transmitted in each DPCCH slot, the correlation result  $Y_r(n)$  of the two signals,  $R_i(n)$  and  $C_i(n)$ , is the coherent accumulation over  $N$  chips, where  $N=N_p * 256$ , in one slot, and it is given by

$$Y_{\tau}(n) = \sum_{i=1}^N R_{i+\tau}(n) C_i(n) \quad (2.12)$$

where  $\tau$  is the path delay in the range  $[0, W-1]$ ,  $W$  being the search window size.  $C_i(n)$  is correlated with  $R_i(n)$  chip by chip over  $N$  chips with a delay of  $\tau$  chips.

## B. Non-coherent Averaging

The longer the correlation, the more pronounced the correlation peak, when the delay hypothesis is correct. However, due to the phase rotation caused during the transmission, the correlation products start to cancel each other when the correlation length is too long. On the other hand, it is found to be beneficial to make the path search decision based on the average power of a few coherent accumulation results. At the end of each coherent accumulation, the accumulated result is a complex value. The power of the coherent accumulation is then taken from the sum of the squares of the real and imaginary components of the complex value.

The non-coherent averaging is, in fact, implemented as the non-coherent accumulation, since the comparisons of the powers of the delays in MPS are all relative.

Mathematically, the coherent accumulation output  $Y_{\tau}(n)$  for the slot  $n$  is a vector summation, since both the locally generated signal and the input baseband signal are complex signals. A scalar value is obtained by calculating the power of  $Y_{\tau}(n)$  at the end of the coherent accumulation. Those power values are then accumulated over  $m$  slots to obtain the non-coherent accumulation result  $Z_{\tau}$  at the delay offset  $\tau$ , as given by

$$Z_{\tau} = \sum_{n=1}^m |Y_{\tau}(n)|^2 \quad (2.13)$$

Thus, the energy value of the correlated signal at the delay offset  $\tau$  in the search window is obtained by non-coherently accumulating the squared magnitude of  $Y_{\tau}(n)$  over  $m$  slots.

The entire correlation process, including the coherent and non-coherent accumulations, is illustrated in Figure 2.9.

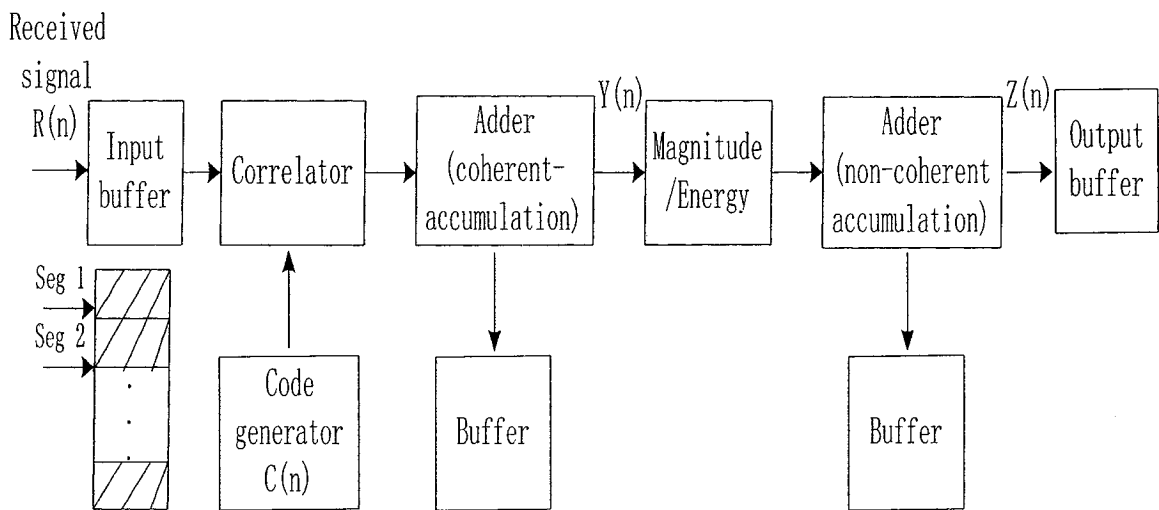


Figure 2.9 Correlation process

One set of the MPS parameters could be the coherent accumulation length of one slot and the non-coherent accumulation length of 15, which gives a total correlation length of one frame.

Breaking the entire correlation length into several coherent accumulation periods also brings the flexibility. For example, the coherent accumulation periods of a correlation do not have to be contiguous, as shown in Figure 2.10. As seen from this figure, the correlation takes contributions from slot 0, slot 2 and slot 4 and skips slot 1 and slot 3. This is one way to reduce the computations and exploit the time diversity of the antenna signal.

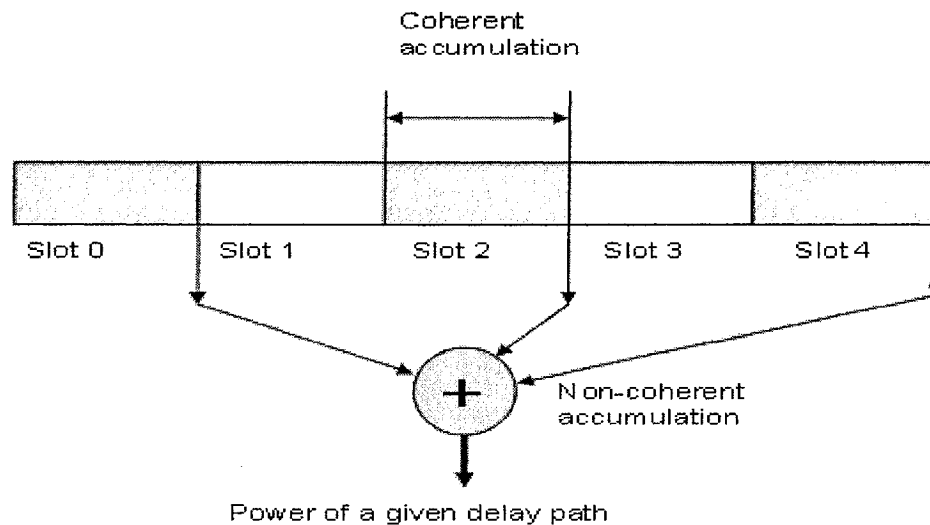


Figure 2.10 An example of coherent and non-coherent accumulations

### 2.4.3 Power Delay Profile

The power delay profile (PDP) represents the received signal power as a function of the path delay. The received signal power is calculated according to the coherent and non-coherent accumulation method described in Section 2.4.2. Each delay offset in the search window should have a corresponding power. Figure 2.11 shows an example of a PDP. It can be seen from this figure, the offset 9 corresponds to the strongest path and the offset 74 corresponds to the second strongest path.

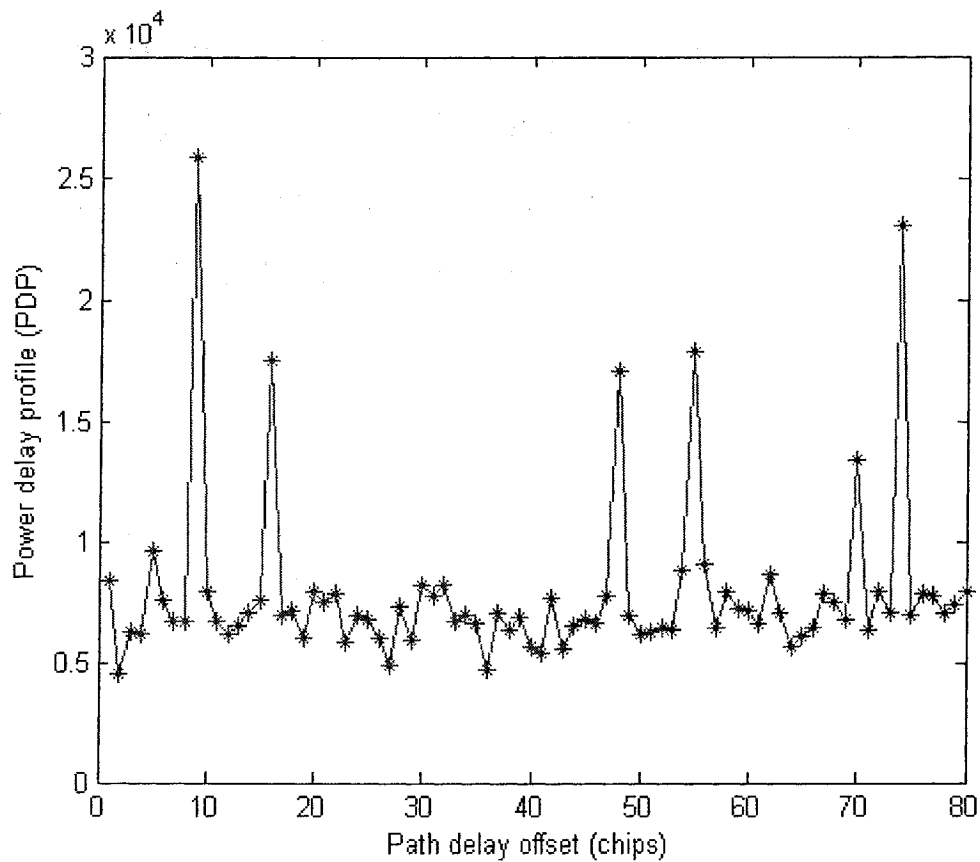


Figure 2.11 An example of PDP

#### 2.4.4 Threshold

Once the PDP is available, one can decide as to which of the offsets in the search window corresponds to a valid strong path. This is achieved by calculating a threshold. A path with a power larger than the threshold will be decided as a valid path. Depending on the level of the threshold, the PDP shown in Figure 2.11 could be said to contain 2, 5 or 6 strong paths.



Obviously, setting the proper threshold is very important for MPS. A wrong threshold could lead to a false detection of a path and destructive additions in the MRC, or misdetection of the paths.

Our research focus is not on how to find the right threshold, instead, since our research focus is on the development of an efficient way to calculate the PDP, we make use of a popular expression for calculating the threshold given by [36]

$$\theta = m_z + \alpha\sigma_z \quad (2.14)$$

where  $m_z$  and  $\sigma_z$  are the mean and variance of the noise, respectively, and  $\alpha$  is a weighting coefficient, by modifying it to suit the proposed scheme.

## 2.5 Summary

In this chapter, the basic knowledge related to MPS in a UMTS system has been reviewed. For the sake of brevity, only the key concepts and operations are discussed. These concepts include digital modulation and demodulation, correlation and power delay profile. The correlation is accomplished through two intermediate steps, namely the coherent accumulation and non-coherent accumulation. The channelization and de-channelization operations are real-number operations, while the scrambling and descrambling operations are complex-number operations.

The search window of an MPS is the uncertainty range of the multi-path delays. The fundamental idea of the MPS is to exhaustively try every possible delay offset within the search window by correlating the locally generated signal, supposedly the replica of the

original transmitted signal in an ideal situation, with the input baseband signal for a certain correlation length. When the hypothesis about the delay of a path is correct, the correlation will result in a peak power value, signifying the existence of a strong path. Ultimately however, a threshold needs to be used to decide whether or not a potential path is a true path.

# Chapter 3      The Proposed Two-Stage Multi-path Searcher

## 3.1 The Background

As stated in Section 1.3, in a full search (FS), all the offsets in a search window are examined for the existence of strong paths, and partial search (PS) checks only some of the offsets in the search window. A full contribution (FC) of a search estimates the contributions from all the antenna samples at any given delay offset in the search window, whereas a partial contribution of a search means only part of the antenna samples of the correlation period are involved in the evaluation of the power delay profile. There are four possible combinations of the search mode and the contribution mode as shown in Table 1.

Table 1 Combinations of the search mode and contribution mode

Mode of contribution \ Mode of search	Full search (FS)	Partial search (PS)
Full contribution (FC)	FSFC	PSFC
Partial contribution (PC)	FSPC	PSPC

Based on the four possible combinations of the search and contribution modes specified in Table 1, we can call the conventional MPS method [19] as the full search with full contribution (FSFC), and the fast search algorithm proposed in [36] as the full search with partial contribution (FSPC). It can be seen that the FSPC is fast, but not reliable. On the other hand, the PSFC is as reliable as the FSFC but only for the slots that are selected for evaluation by the partial search. PSFC does not evaluate all the offsets in the search window. In the proposed algorithm [47], the foresaid features of FSPC and PSFC are exploited in a two-stage scheme described below.

In the first stage, the FSPC is employed to narrow down the number of offsets from  $W$  to  $M$ . Processing of FSPC is fast, since its computational complexity is low. At the same time, as long as  $M$  is large enough, the offsets of true paths are still retained in the subset even though the FSPC cannot be guaranteed to provide the final conclusion on the true delays.

In the second stage, the PSFC is employed to re-evaluate the accumulated power at the  $M$  offsets for the entire correlation period. Consequently, every single offset in the subset of  $M$  offsets has the same power as the one that would have been calculated using the FSFC. This is the reason as to why the proposed algorithm has almost the same path detection accuracy as that given by the FSFC.

The proposed two-stage MPS algorithm can substantially reduce the computation complexity and yet maintain almost the same path detection accuracy as that of the

conventional full search approaches. The fundamental idea of this new approach is to use full search with partial contribution (FSPC) in the first stage to eliminate those locations that are impossible to represent a strong path. Then, in the second stage, a partial search with full contribution (PSFC) is conducted on the remaining candidates to obtain the same detection accuracy as that of a FSFC algorithm.

The proposed idea is applicable to multi-path searchers for any correlation lengths even though in this study for the sake of simplicity, we limit the correlation length to be that of one frame. We further restrict ourselves to the case of a base station receiver where the sampling rate is equal to the chip rate even though the proposed algorithm is applicable to cases with the over sampling rate to be 2 or higher. Thus, one sampling period is the same as one chip period. We use terms sample and chip interchangeably in the subsequent discussions.

### **3.2 First Stage Search**

The objective of the first stage is to narrow down the number of delay offsets from the number corresponding to a search window size to that corresponding to a handful locations, which require a closer examination for true delays in the second stage. In essence, this is a full search (FS) with partial contribution (PC). Every location in the search window is a possible location for a strong path. Thus, certain criterion has to be used to decide on the likelihood of a position to be a true path. The criterion used here is still a kind of power delay profile (PDP), in which these PDPs do not have the correlation contributions from all the segments of the antenna data. Using the partial contribution

from a frame is the key to reducing the computational cost. Nonetheless, these PDPs still distinguish a possible strong path from an unlikely path.

Assume that the total correlation length is  $N$  slots, and only  $K$  ( $K < N$ ) slots are used for the calculations of the PDP, if the partial contribution is performed. There are more than one way to fulfill the requirements for processing the FS and PC parts of the FSPC of the first stage. One way to do this is to use the first  $K$  slots of the  $N$  slots to calculate the PDP. This method is designated as Method A. Another way to do this is to use the  $N$  slots for the calculation of the PDP, but each slot is allowed to contribute only to a few delay offsets. This method is designated as Method B.

### **3.2.1 Method A**

In this method the correlation process is very similar to the conventional full search with full contribution except that the search stops right after the  $K$ th slot has been correlated. Using the first  $K$  slots has the best processing latency advantage even though any  $K$  slots of the  $N$  slots could be used for this purpose. This is because the second stage can start the search immediately after the correlation calculation of the first  $K$  slots has been completed, if the first  $K$  slots are used for the first-stage correlation. This advantage can be better appreciated after the discussion of Method B in Section 3.2.2. We now describe the processes of correlation and PDP accumulation.

*The  $W$  PDP accumulators corresponding to the  $W$  chips in the search window are first reset to zero.*

*For each of the  $K$  slots do the following,*

*For each of the PDP, do the following,*

- *Select the proper phase of the scrambling code according to the delay offset,*
- *Generate the corresponding pilot symbols according to the slot format number*
- *Coherently correlate and accumulate the locally generated scrambling code along with the pilot symbols with the input antenna data of the current slot*
- *From the coherent accumulation result, we obtain the power of the signal, and non-coherently add these power to the PDP accumulation.*

The final PDP values are obtained after all the  $K$  slots have been processed.

The following example illustrates the correlation and accumulation processes given that  $N=15$  slots,  $K = 3$  slots, and  $W = 80$  chips. The coherent accumulation is performed over one slot, whereas the non-coherent one is performed over three slots. That is, each final PDP value is the summation of three intermediate values each of which is the power of the coherent accumulation result. As shown in Figure 3.1, the PDP values within the search window are simply the power values of the corresponding coherent accumulation results between slot 0 antenna data and the corresponding locally generated scrambling code and the pilot symbols. It can be seen that the PDP values represent the preliminary strength of a potential path, vary depending on the location of the delay offsets.

As shown in Figure 3.2, each of the PDP values within the search window after slot 1 is the summation of two scalar numbers; one is the power of the coherent accumulation of slot 0 and the other is that of slot 1.

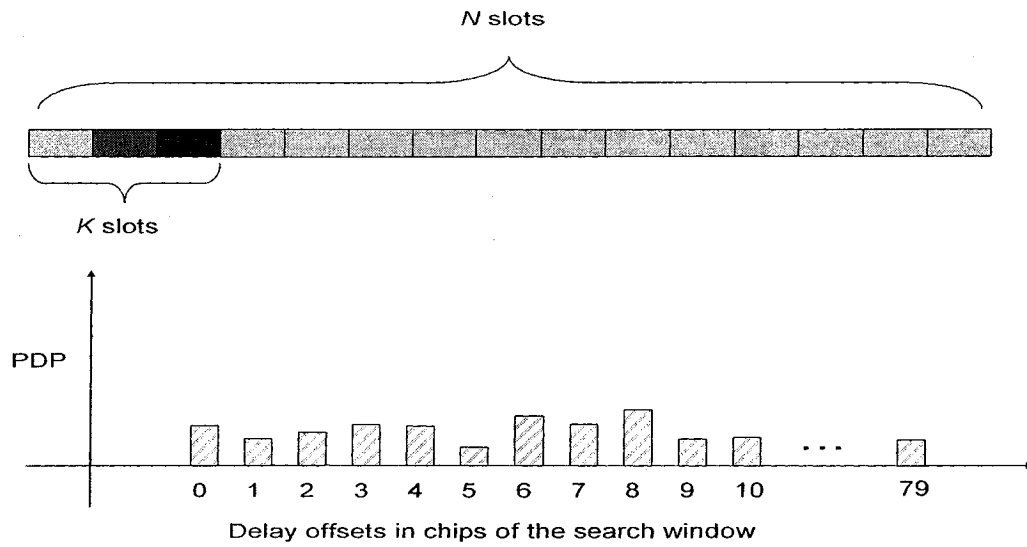


Figure 3.1 An illustration of the PDP values after slot 0 using Method A

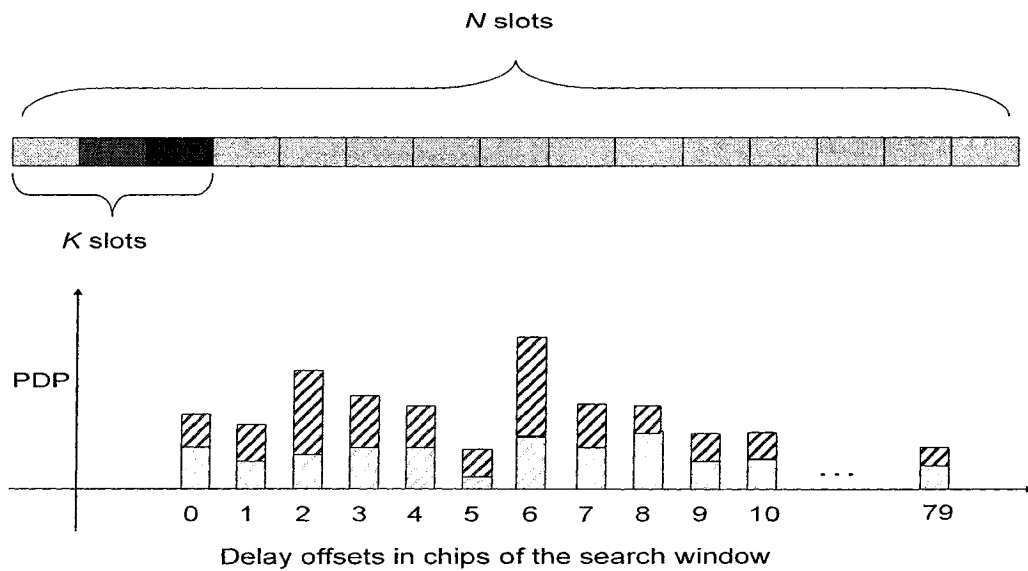


Figure 3.2 An illustration of the PDP values after slot 1 using Method A

Finally, as shown in Figure 3.3, the PDP values in the search window represents the total contribution from all the  $K = 3$  slots.



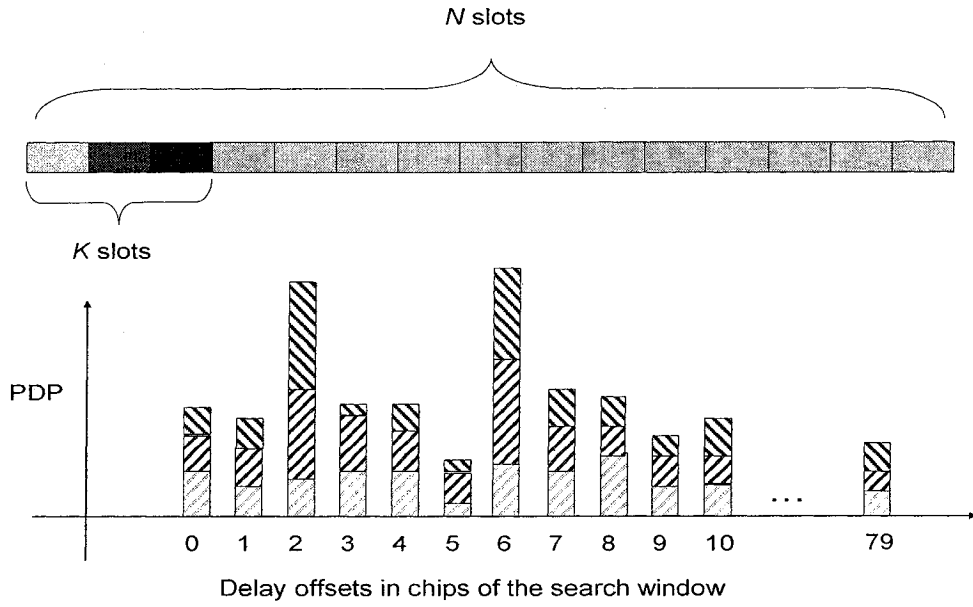


Figure 3.3 An illustration of the PDP values after slot 2 using Method A

The results of the correlation and accumulation processes for the above example are summarized in the Table 2, where  $A(i,j)$  represents the power of the coherent accumulation at offset  $i$  from the contribution of slot  $j$ , and  $PDP[i]$  represents the final power delay profile value at offset  $i$ . It can be seen from this table (as well as from Figure 3.3) that for the given example, every PDP value in the search window is result of the contributions from slots 0, 1 and 2.

Table 2 PDP values of the first stage using Method A

Delay offset (chips)	Power of the coherent accumulation of a slot			Power delay profile (PDP)
	$A(i,0)$	$A(i,1)$	$A(i,2)$	
0	$A(0,0)$	$A(0,1)$	$A(0,2)$	$PDP[0] = A(0,0) + A(0,1) + A(0,2)$
1	$A(1,0)$	$A(1,1)$	$A(1,2)$	$PDP[1] = A(1,0) + A(1,1) + A(1,2)$
2	$A(2,0)$	$A(2,1)$	$A(2,2)$	$PDP[2] = A(2,0) + A(2,1) + A(2,2)$
3	$A(3,0)$	$A(3,1)$	$A(3,2)$	$PDP[3] = A(3,0) + A(3,1) + A(3,2)$
...	...	...	...	...
79	$A(79,0)$	$A(79,1)$	$A(79,2)$	$PDP[79] = A(79,0) + A(79,1) + A(79,2)$

It should be emphasized that this method has the timing advantage. The second stage commences immediately after the first  $K$  slots have arrived, thus providing more time for the second stage to process. Figure 3.4 illustrates the timing relation between the two stage for Method A.

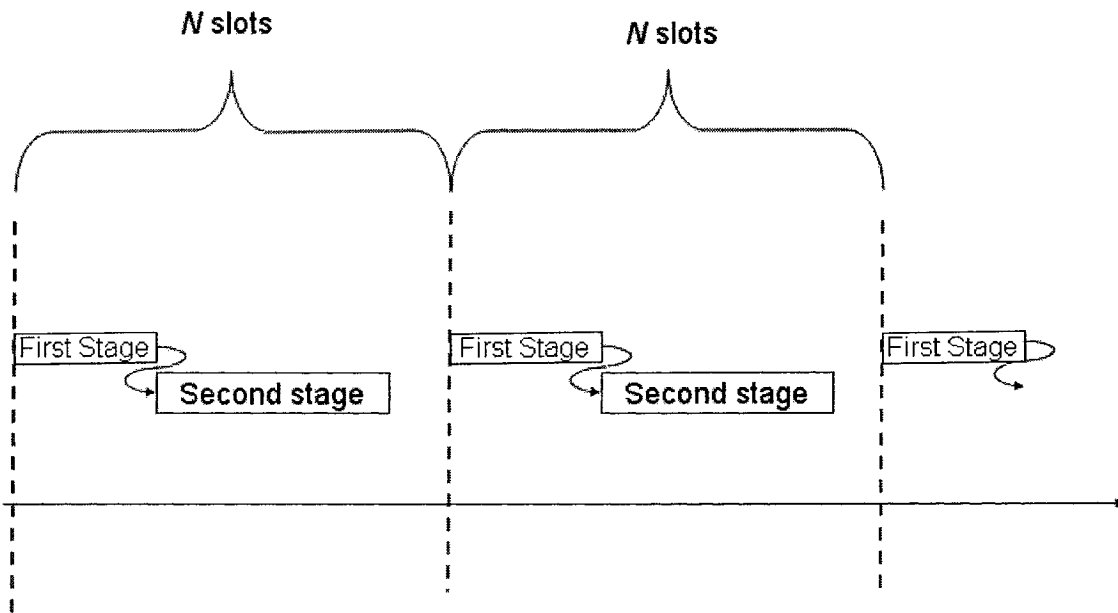


Figure 3.4 The timing relationship between the two stages using Method A

The obvious drawback of this method, however, is that it is sensitive to burst noise since only the first  $K$  slots are used for evaluating the potential candidates for the strong path. The following method overcomes this problem by exploiting the time diversity of the signal, thus is more robust in the presence of burst noise.

### 3.2.2 Method B

In this method, the principle of interleaved offsets [40] is applied in the first-stage search, since the interleaved data is more robust to a variety of interference sources, such as noise

interference and multi-path interference, especially in the case of time varying burst noise. By interleaving the same correlated data at different delay offset, we can take advantage of the time diversity to limit data errors within a reasonable range, thus achieving an improved detection probability.

To apply the interleaving idea to the first stage, one needs to select the  $K$  contributing slots that are evenly distributed among the  $N$  slots for a given offset. One way to implement this is to allow slots  $0, N/K, 2*N/K, \dots, n*N/K$  to contribute to the PDP of offset  $0$ , and slots  $1, N/K+1, 2*N/K+1, \dots, n*N/K+1$  to contribute to that of offset  $1$ , etc.

We now illustrate this method by choosing the following parameter values:  $N = 15$ ,  $K = 3$ , and  $W = 80$ . Furthermore, the coherent and non-coherent accumulation lengths are chosen to be 1 and 3 slots, respectively.

In the interleaved offset correlation, the consecutive offsets are processed one slot apart to avoid the effect of the noise interference and the PN-cross correlations. The coherent accumulation between the first slot of the received complex signals  $R_i(0)$  and the locally generated complex signals  $C_i(0)$  are performed at the delay offsets of  $0, 5, \dots, 75$ , for a search window of size  $W = 80$ . The PDP value distribution, after the first slot is processed, is shown in Figure 3.5. As seen from this figure, slot 0 contributes to the PDP accumulation for the offsets of  $0, 5, \dots, 75$ . Next, the second slots,  $R_i(1)$  and  $C_i(1)$ , are coherently accumulated at the delay offsets of  $1, 6, \dots, 76$ . After the accumulation of

correlation results from the second slot, 16 new PDP values are obtained as shown in Figure 3.6.

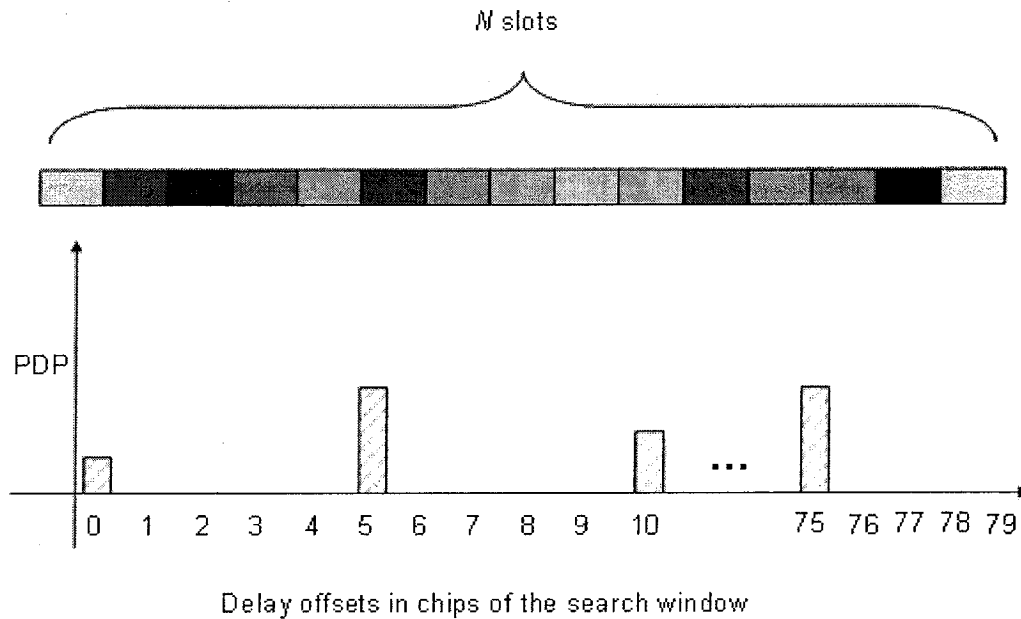


Figure 3.5 An illustration of the PDP values after slot 0 using Method B

This process is repeated until the 5th slot correlation between  $R_i(4)$  and  $C_i(4)$  is carried out for the delay offsets 4, 9, ..., 79. By doing so, the PDP values corresponding to all the 80 offsets in the search window are loaded with the correlation results obtained from the first 5 slots of the received signal. One can consider these correlation results as a first layer of power delay profile over a search window of size 80. The information in the first 5 slots are repeated and interleaved in the search window. The PDP values thus obtained are different for each slot for different delay offsets as shown in Figure 3.7.

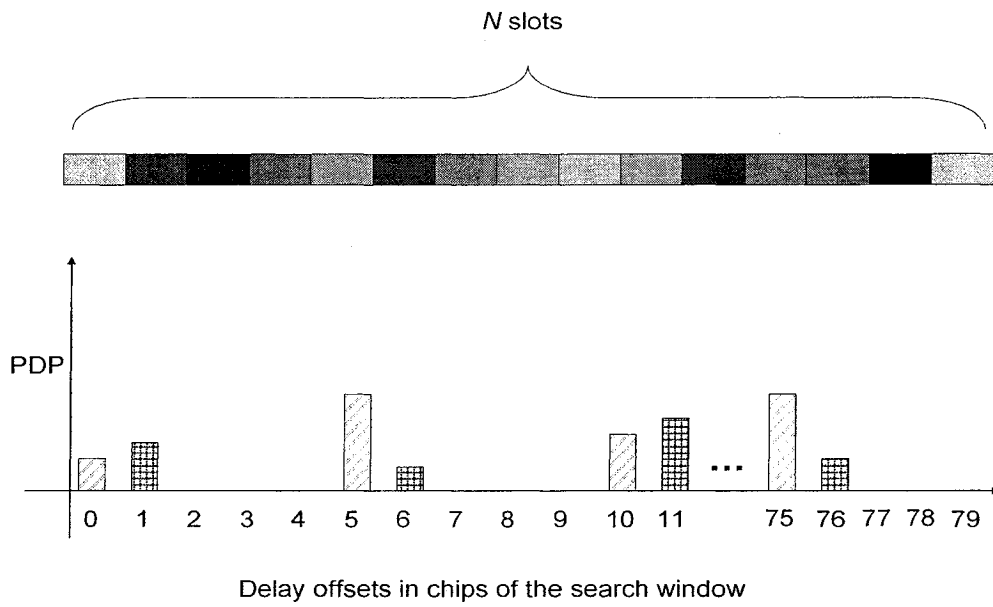


Figure 3.6 An illustration of the PDP values after slot 1 using Method B

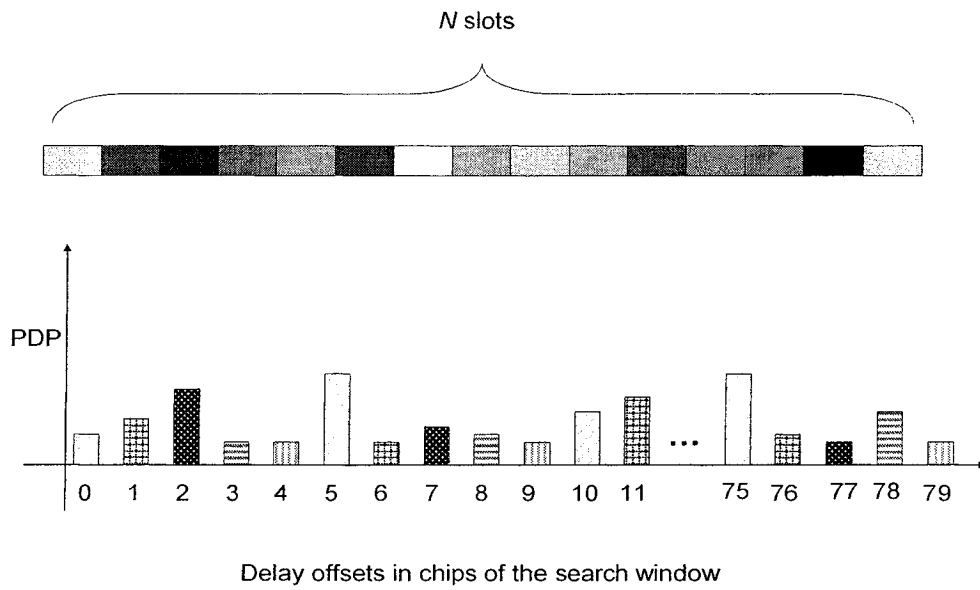


Figure 3.7 An illustration of the PDP values after slot 4 using Method B

The next 5 slots are interleaved in the same way as carried out for the first five slots, and the correlation values of these next 5 slots serve as the second layer of PDPs. The resulting PDP distribution is shown in Figure 3.8.

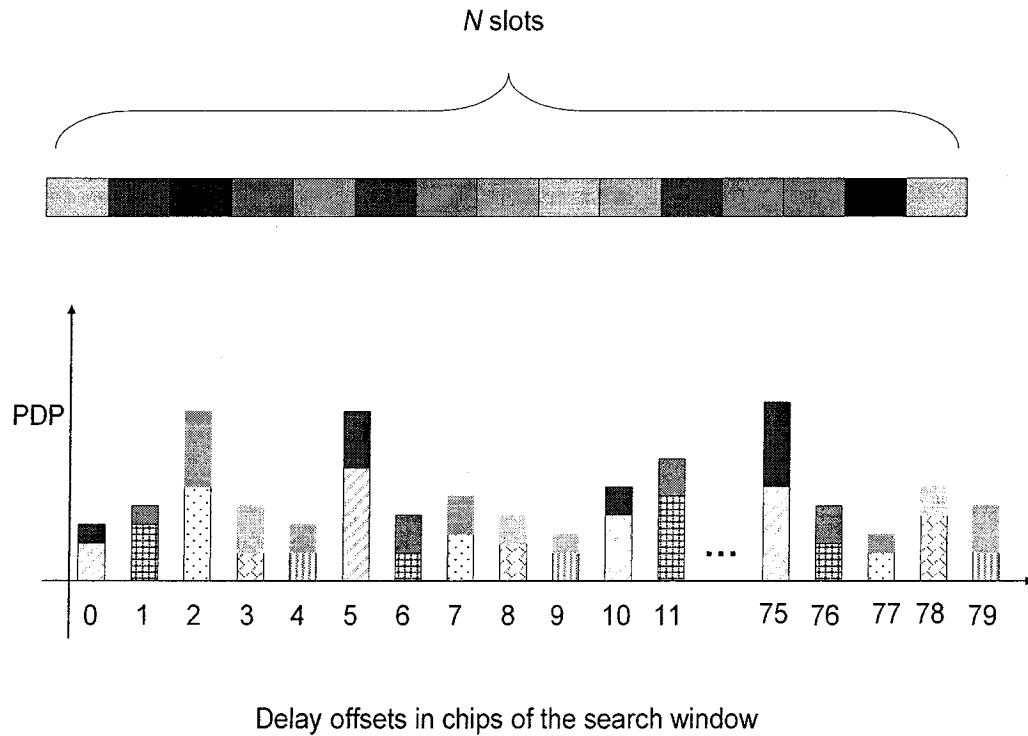


Figure 3.8 An illustration of the PDP values after the first 10 slots using Method B

Finally, the third layer of PDP values are obtained by correlating and interleaving the remaining 5 slots of the received signal over 80 offsets. The final PDP distribution of the first stage is shown in Figure 3.9.

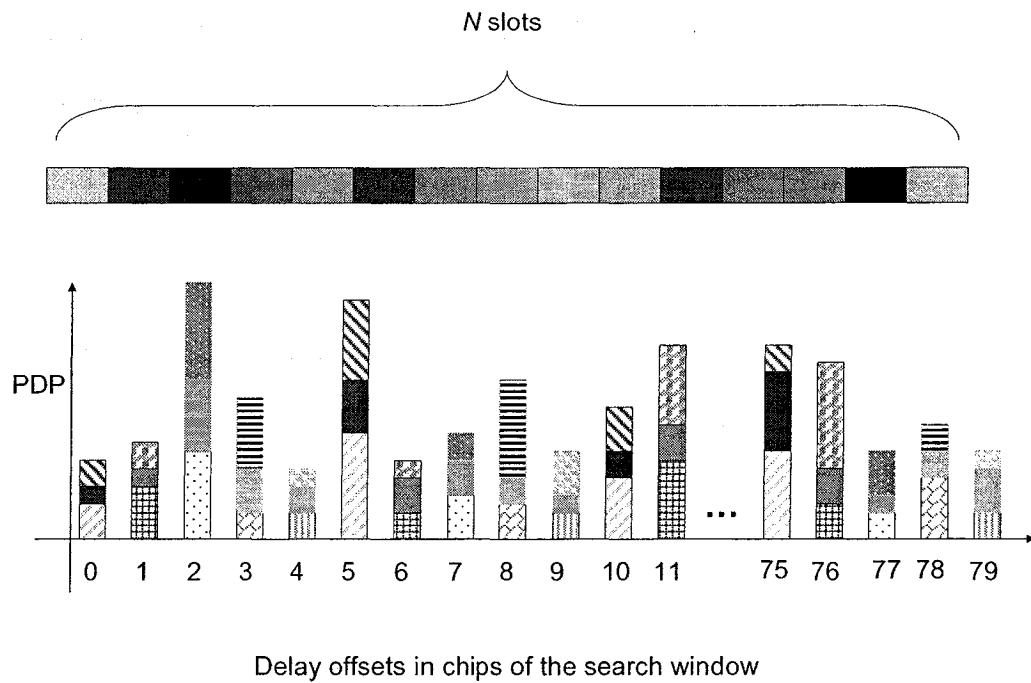


Figure 3.9 An illustration of the PDP values of the first stage using Method B

Table 3 shows how the interleaved offsets are used in the first-stage search to obtain the final PDP values at various delay offsets in the search window. It is seen from this table that the first-stage PDP values are obtained by non-coherently accumulating over three slots. Conceptually this table conveys information similar to that given by Figure 3.9. The PDP results obtained by using Method B as shown in Figure 3.9 and Table 3 would help us performing a refined search.

Table 3 PDP values of the first stage using Method B

Delay offset (chips)	Power of the coherent accumulation of a slot			Power delay profile (PDP)
0	A(0,0)	A(0,5)	A(0,10)	PDP[0] = A(0,0) + A(0,5) + A(0,10)
1	A(1,1)	A(1,6)	A(1,11)	PDP[1] = A(1,1) + A(1,6) + A(1,11)
2	A(2,2)	A(2,7)	A(2,12)	PDP[2] = A(2,2) + A(2,7) + A(2,12)
3	A(3,3)	A(3,8)	A(3,13)	PDP[3] = A(3,3) + A(3,8) + A(3,13)
4	A(4,4)	A(4,9)	A(4,14)	PDP[4] = A(4,4) + A(4,9) + A(4,14)
5	A(5,0)	A(5,5)	A(5,10)	PDP[5] = A(5,0) + A(5,5) + A(5,10)
6	A(6,1)	A(6,6)	A(6,11)	PDP[6] = A(6,1) + A(6,6) + A(6,11)
...	...	...	...	...
5 $\gamma$ (0 $\leq\gamma\leq15$ )	A(5 $\gamma$ ,0)	A(5 $\gamma$ ,5)	A(5 $\gamma$ ,10)	PDP[5 $\gamma$ ] = A(5 $\gamma$ ,0)+ A(5 $\gamma$ ,5)+ A(5 $\gamma$ ,10)
...	...	...	...	...
5 $\gamma$ +4	A(5 $\gamma$ +4,4)	A(5 $\gamma$ +4,9)	A(5 $\gamma$ +4,14)	PDP[5 $\gamma$ +4]=A(5 $\gamma$ +4,4)+A(5 $\gamma$ +4,9)+ A(5 $\gamma$ +4,14)
...	...	...	...	...
78	A(78,3)	A(78,8)	A(78,13)	PDP[78] = A(78,3)+A(78,8)+ A(78,13)
79	A(79,4)	A(79,9)	A(79,14)	PDP[79] = A(79,4)+A(79,9)+A(79,14)

In Method B, the contributions to the PDP at a given delay offset come from the input antenna slots that are evenly distributed within the N slots. This ensures that the best time diversity is utilized to combat any burst noise. For example, PDP[0] consists of the contributions from slot 0, slot 5 and slot 10. For the same offset in Method A, however, the PDP consists of the contributions from the very first three slots. On the other hand,



Method B does have a drawback compared to Method A. That is, it takes longer time to finish since it requires all the  $N$  slots to participate in the correlation and accumulation processes, the second stage cannot start until all the  $N$  slots have arrived. This effectively makes the second stage of the current correlation period ( $N$  slots) to be executed at the same time as the first stage of the next correlation period. This phenomenon is clear from the timing relation of the two stage using Method B, shown in Figure 3.10.

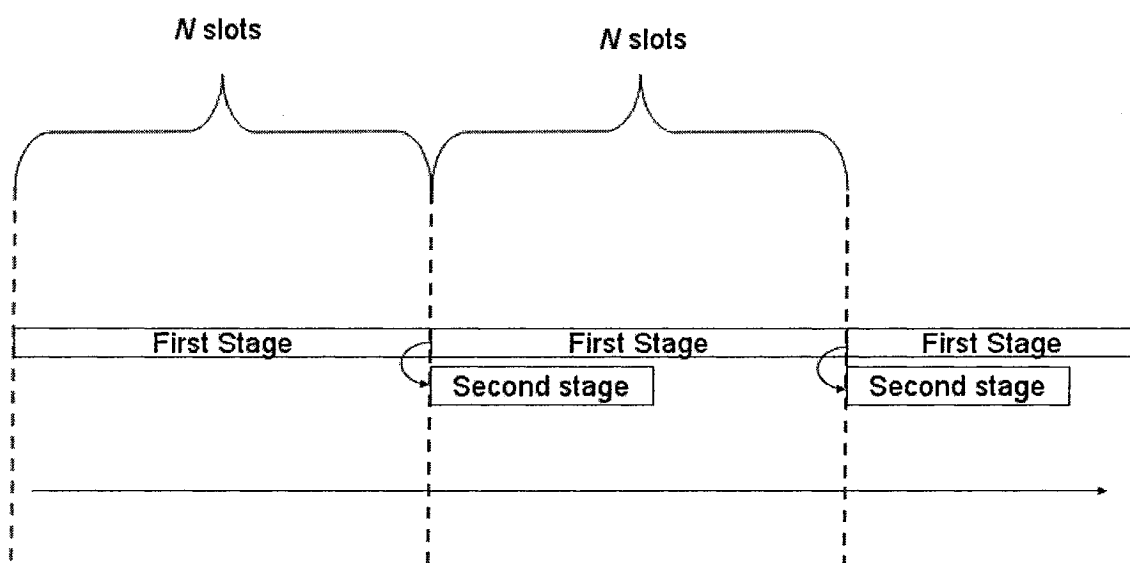


Figure 3.10 The timing relationship between the two stages using Method B

Because of the concurrency of the two stages using Method B, this method requires more memory to buffer the antenna data, since the previous  $N$  slots cannot be relinquished until the processing of second stage is completed.

### 3.3 Second-Stage Search

By using the interleaved offset method in the first-stage search, we could obtain a set of PDP values at  $W$  offsets as shown in the last column of Table 3. These PDP values seem

to serve well in determining the possible path delay offsets. However, we cannot find the actual delay paths by applying just one stage search, since the detection probability would be low when the information contained only in a few slots are used. Because of this reason, a second-stage search is needed. We would now take advantage of the selection of the offsets obtained through the first-stage search and carry out a more refined search and selection in a second stage with a narrowed down range for the search of offsets.

In the second stage, a partial search (PS) with a full contribution (FC) is performed. We choose  $M$  ( $M < W$ ) offsets corresponding to the  $M$  largest PDP values from the first-stage search as a narrowed down range for the search of offset in the second stage. That is, only the  $M$  strong candidate offsets are re-evaluated with the full contributions from all the slots within the correlation period.

### **3.3.1 Correlation and Accumulation**

For each of the  $M$  offsets, the final PDP value comprises the contributions from all the  $N$  slots. The corresponding PDP obtained in the first stage is the contribution from the  $K$  slots, thus it can be used as the starting point for the accumulation of the second stage. Thus, in the second stage only the remaining  $(N-K)$  slots need to be correlated and accumulated. Depending on Method A or B used in the first stage, the second stage has a slightly different way of carrying out the correlation operation.

If Method A is used in the first stage, then the PDP calculation of the  $M$  candidates can simply proceed as each new slot becomes available in time. The calculation ends after the

$N^{\text{th}}$  slot calculation has been completed. For example, after the first stage the PDP[0] already contains the contributions from slots 0, 1 and 2, as given by

$$PDP[0] = A(0,0) + A(0,1) + A(0,2) \quad (3.1)$$

The correlation and accumulation in the second stage will, therefore, use the contributions only from the remaining 12 slots, if  $N = 15$ :

$$PDP[0] = A(0,0) + A(0,1) + A(0,2) + A(0,3) + \dots + A(0,14) \quad (3.2)$$

If Method B is used in the first stage, then the PDP calculation of the  $M$  candidates needs to be done by adding to the buffered information concerning PDP value of the  $N$  previous slots and add to it only those slots that are not included in the PDP from the first stage. In particular, for the case of the above example, the PDP[0] would start with the following value at the beginning of the second stage,

$$PDP[0] = A(0,0) + A(0,5) + A(0,10) \quad (3.3)$$

Then, it would have the same final value as that given in Eq. (3.2), except that the order of obtaining the result is slightly different:

$$PDP[0] = A(0,0) + A(0,5) + A(0,10) + A(0,1) + A(0,2) + \dots + A(0,14) \quad (3.4)$$

### 3.3.2 Thresholding

Unlike the downlink MPS in [40], where several thresholds are calculated and used in different stages and for different detection purposes, in the proposed algorithm, no thresholding is needed in the first stage; however, it is employed in the second stage.

A threshold  $\theta$  is needed in the second stage to determine the actual delay paths. We make use of the threshold expression given by (2.14) for the proposed and existing FSFC and FSPC algorithms. However, the expression for the threshold given by (2.14) cannot be applied directly for the proposed algorithm due to the following reason.

The expression for the threshold given by (2.14) has been derived for a situation, where the number of delay offsets is large, so that one can remove  $Q$  offsets corresponding to the  $Q$  largest PDP values and obtain the mean and variance of the remaining noise paths. In the existing uplink MPS methods, the number of delay offsets is equal to  $W$ , the size of the search window. Therefore, in these cases if  $Q$  (say  $Q = 10$ ) offsets are removed, one can still obtain reasonably accurate values of the noise mean and variance, and still use (2.14) directly to calculate a value for the threshold  $\theta$ .

For the calculation of the threshold  $\theta$  to be used in the second stage of the proposed algorithm, the condition for using (2.14) is different: the PDP values of only  $M$  (eg.  $M = 12$ ) offsets, instead of  $W$  offsets, are calculated. In order to overcome this problem, we proceed as follows to determine a reasonably accurate value for the threshold  $\theta$  for the second stage.

Note that in the first stage the PDP values of all the  $W$  offsets are available. Therefore, by using (2.14), one can obtain the value of the threshold as

$$\theta = \theta_1 = m_z + \alpha\sigma_z \quad (3.5)$$

As we move from stage 1 to stage 2, the number of slots contributing to the correlation changes from  $K$  (=3) to  $N$  (=15). Therefore, the threshold expression given by (3.5) can be modified to provide expression for obtaining an approximate value of threshold for the second stage:

$$\theta = \theta_1 * \frac{N}{K} = (m_z + \alpha\sigma_z) * \frac{N}{K} \quad (3.6)$$

### 3.4 Some Practical Considerations for the Proposed Algorithm

As pointed in Section 3.2.2, Method B requires extra memory to buffer the data corresponding to  $N$  slots until the operation of the second stage is completed. Fortunately, this is not an issue for a real UMTS system due to the following reason.

In order for the Node-B to correctly de-spread the antenna data, the spreading factor has to be known before the FD can start to work on a given frame of DPDCH data. When a radio link is set up, only the minimal spreading factor is given to the physical layer at a Node-B. The UE could change the spreading factor on a frame by frame basis. Therefore, the Node-B needs to decode the spreading factor for each frame. The spreading factor is encoded by a set of TFCI bits in the DPCCH. This set of TFCI bits is encoded in one radio frame which consists of 15 slots. Since DPDCH and DPCCH are transmitted at the same time for each frame, and the SF of the DPDCH is encoded by the TFCI bits of the corresponding DPCCH, at least one frame of antenna data has to be buffered. In a real system, the buffer length is typically 18 to 20 slots since the channel processing time has to be taken into consideration. Therefore, the proposed two-stage search algorithm

effectively does not require any extra memory to buffer the antenna data. The buffering needed to resolve the TFCI decoding issue would be sufficient for this purpose.

Another point that needs to mention is the second-stage computation is not computationally very intensive, since only a few offsets need to be evaluated. Thus, the corresponding processing time is relatively very small compared to the frame length.

### **3.5 Computation Complexities of the Proposed and Existing Algorithms**

In this section, we present expressions for the computational complexities of the full search with full contribution (FSFC), full search with partial contribution (FSPC) and the proposed algorithms.

The most rudimental operation of the correlation is the complex multiplication as expressed by (2.9). However, the actual number of cycles that it takes to complete the operation in a given platform is hardware dependent. For example, 64 complex multiplications can be performed within one clock cycle, which is 32 times the chip rate in TC1110 [35]. On the hand, the hardware proposed in [40] operates in a completely different way. In order to maintain a fair comparison, we assume all the algorithms are implemented on a general-purpose computer.

Also, here we define a macro computation unit to obtain a simple metric for the measure of the computational complexities of the algorithms. We call this macro computation unit

as slot correlation, and define it as the amount of computation needed to complete the correlation of one slot of the input data sequence and one slot of the locally generated sequence. The granularity of the correlation length is one slot for all the algorithms. With one unit of computational complexity as defined above, expressions of the computational complexities of the three algorithms, namely the FSFC, FSPC and proposed algorithms, can be obtained as follows:

$$C_{FSFC} = N*W \quad (3.6)$$

$$C_{FSPC} = K*W \quad (3.7)$$

$$C_P = K*W+N*M \quad (3.8)$$

where  $N$  is the number of slots in the correlation period,  $K$  is the number of slots contributing to the correlation,  $W$  is the number of offsets in the search window, and  $M$  is the number of offsets used by the second-stage search of the proposed algorithm.

It is seen from (3.6) that for the FSFC algorithm with a correlation length of 15 slots, and a search window of 80 chips, the computation complexity is 1200 slot correlation units. For the FSPC algorithm with 5 slots (using one slot out of every three) contributing to the correlation, and a search window of 80 chips, the complexity using (3.7) is 400 slot correlation units. Effectively, this method requires only  $400/1200 = 33\%$  of the computations required by the FSFC method. Finally, for the proposed method, for the same correlation length of 15 slots and the same search window size of 80 chips, and if the first stage uses only three slots for the correlation and the second stage performs correlations on  $M=12$  offsets, the complexity as calculated using (3.8) is 384 slot

correlation units. Effectively, this method requires  $384/1200 = 32\%$  of the computations required by the FSFC method.

It should be pointed out that practically it not very objective to compare the computational complexities of the algorithms without looking at their performance in terms of the detection accuracy. The expression for the computational complexities of proposed, FSFC and FSPC algorithms presented in this section will be used in Chapter 4 when these algorithms will be compared with each other under various scenarios.

### **3.6 Summary**

The goal of this chapter has been to develop an MPS algorithm that can provide a path detection performance similar or comparable to that of the traditional full search with full contribution algorithm but with a lower computational complexity. For this purpose, a two-stage MPS algorithm has been introduced. The first stage is used to narrow down the number of the candidates that are likely to be one of the true paths. Thus, a full search with partial contribution is used in this stage. The accuracy of the PDP is not important in this stage, since this is only a preliminary search for possible true paths. Two methods (Method A and Method B), are devised for the first stage based on the way a small number of slots are selected for the search. The Method A has the advantage of requiring less processing time, but suffers from presence of burst noise in the input data sequence. The Method B, on the other hand, exploits the time diversity of the input sequence, since it is more resistant to burst noise in the input sequence, but it requires a longer processing time and potentially more memory for buffering the data. The second stage performs a



refined search, but it is limited to only a certain number of the largest peaks of the PDP of the first stage. Thus, effectively this stage makes a partial search with full contributions.

## **Chapter 4      Results on Computational Complexity and Simulation Study**

### **4.1 Introduction**

In this chapter, the proposed two-stage MPS algorithm and the existing FSFC and FSPC algorithm are assessed in terms of their computational complexity and performance accuracy. The computational complexity of the algorithm is determined in terms of the number of correlation operations, whereas the performance analysis is carried out in terms of detection and false alarm probabilities by conducting a number of simulations of the algorithms.

Both the computational complexity and the performance evaluation are carried out for two different scenarios of the number of slots contributing to the correlation operations and the size of the search range. In Scenario 1, as shown in Figure 4.1, the number of contributing slots  $K$  is chosen to be 5 in the FSPC algorithm, and the number of selected offsets  $M$  is chosen to be 12 in the second stage of the proposed algorithm. In Scenario 2, as shown in Figure 4.2, these parameter values are changed to  $K = 9$  and  $M = 40$ . The choices made for different number of slots and offsets are for the purpose of getting a similar computational complexity for the proposed and FSPC algorithms under the same scenario.

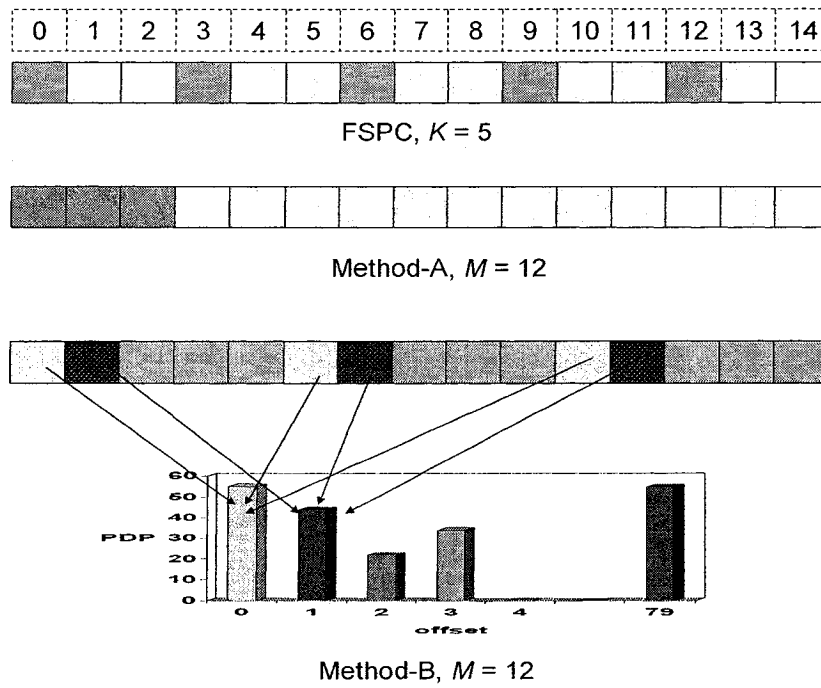


Figure 4.1 An illustration of Scenario 1

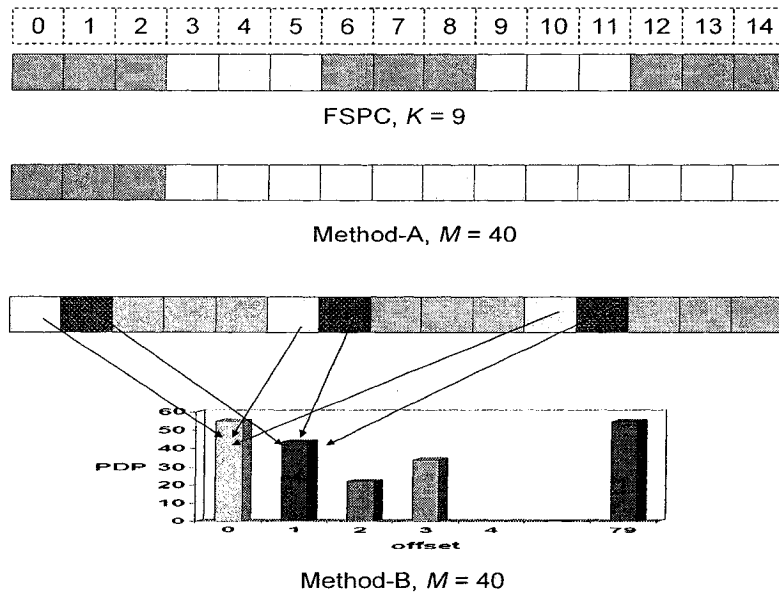


Figure 4.2 An illustration of Scenario 2

## 4.2 Computational Complexity of the Algorithms

In this section, we determine the computational complexity of the three MPS algorithms in terms of the number of correlation operations. The computational complexity derived in Section 3.5 for both proposed and existing MPS algorithms are used for this purpose.

The full search with full contribution (FSFC) algorithm is used as the baseline for the comparison. The correlation length is chosen to be one frame, and the search window size as  $W = 80$  offsets. The results of the computational complexity of the various MPS algorithms are shown in Table 4 for Scenario 1 ( $K = 5$  and  $M = 12$ ). It is seen from this table that the FSPC and proposed algorithms carry a computational load of 33% and 32% respectively of that required by the FSFC algorithm.

Table 4 Comparison of the computational complexity of various MPS algorithms under Scenario 1

Search method		Number of slots	Number of offsets	Number of slot correlations	Complexity ratio
FSFC		15	80	1200	100%
FSPC		5	80	400	33%
Proposed algorithm with $M=12$	1 <sup>st</sup> stage	3	80	384	32%
	2 <sup>nd</sup> stage	12	12		

The computational complexity of various MPS algorithms are also calculated for Scenario 2 ( $K = 9$  and  $M = 40$ ), and shown in Table 5. It is seen from this table that both the FSPC and proposed MPS algorithms carry a computational load of 60% of that required by the FSFC algorithm.

Table 5 Comparison of the computational complexity of various MPS algorithms under Scenario 2

Search method		Number of slots	Number of offsets	Number of slot correlations	Complexity ratio
FSFC		15	80	1200	100%
FSPC		9	80	720	60%
Proposed algorithm with $M = 40$	1 <sup>st</sup> stage	3	80	720	60%
	2 <sup>nd</sup> stage	12	40		

### 4.3 Experimental Conditions for Simulation

Computer simulations have been performed under the following conditions.

- The over-sampling rate is chosen to be one, that is, there is one sample in one chip period.
- Only pilot bits are used for the calculation of the power of multi paths, and a mask is applied to each slot to exclude those bits that do not correspond to the pilot bits.
- Only floating-point arithmetic is used in the simulation.
- A single-user system is assumed; thus, the antenna input data would contain only one scrambling code with multiple delays.
- Up to 6 delay paths are simulated in the tests.

- It is assumed that the Gaussian noise is applied to all the slots within the correlation period.

The parameter values used in the simulations are listed in Table 6.

Table 6 Simulation parameters

Parameter	Values
Scrambling code number	12
OVSF code for DPDCH	$C_{ch,64,16}$
OVSF code for DPCCH	$C_{ch,256,0}$
Slot format number for DPDCH	2
Slot format number for DPCCH	5
Input data length	1 frame (38400 chips)
Over-sampling rate	1
DPDCH spreading factor	64
DPCCH spreading factor	256
Digital modulation	QPSK

## 4.4 Simulation Results

All the results presented in this section are statistical averages. One thousand runs are made to determine the average value for each performance index. For example, to get the probability of detection at SNR = -16 dB for the proposed algorithm, 1000 input data sequences of 38,400 chips is generated, and a set of 6 delays (within the delay offset range of the search window  $W$ ) are randomly generated and applied to the frame data to create a multi-path environment. The multi-path signal is then corrupted by AWGN to provide various SNR levels to the transmitted signal. The process is described by (2.10). The received signal sequence is then fed into the proposed method to detect the path.

The probability of detection ( $P_d$ ) is defined as the ratio between the number of fingers correctly detected and the number of fingers that actually exists in the input data sequence. The probability of false alarm ( $P_f$ ) is defined as the probability of invalid paths that exceed the threshold and assigned to the RAKE fingers. In other words, it is the probability of selecting a path that actually does not exist

As an example of calculating one test, let us consider the randomly generated delays of  $\{4, 17, 31, 42, 48, 65\}$ , and the detected delays of  $\{4, 17, 36, 42, 48, 65\}$ . Then, the probability of detection is  $5/6 = 0.8333$ , the probability of false alarm is  $1/6 = 0.1667$ . This test is repeated 1000 times to obtain the average value for the probability of detection and probability of false alarm at certain level of SNR. The test is conducted for each level of SNR.

#### **4.4.1 Scenario 1**

Table 7 shows the results of the probability of detection for Scenario 1. As already seen from Table 4, the proposed method has about the same computational complexity as that of the FSPC method. Yet, as seen from Table 7, the former has a detection probability performance much higher than that of the latter and close to that of the FSFC method through the entire range of the SNR values used in the experiment.

Table 7 Probability of detection for Scenario 1

SNR (dB)	FSFC algorithm	FSPC algorithm with $K = 5$	Proposed algorithm with $M = 12$	
			Method A	Method B
-10	0.90917	0.82167	0.88479	0.88396
-12	0.84146	0.71667	0.81042	0.80125
-14	0.77333	0.60375	0.71375	0.72396
-16	0.68854	0.48271	0.63062	0.62583
-18	0.54667	0.32458	0.47437	0.48167

In order to provide a graphical illustration of the performance of the three algorithms, in Figure 4.3 the probability of detection is plotted as a function of SNR. As expected and also seen from both Table 7 and Figure 4.3, the proposed method cannot provide a performance as good as that obtained from the FSFC method. However, the proposed method needs only a fraction of the computations that the FSFC method would require. The smoothness of the curves in Figure 4.3 also suggests that the performance of each of the methods is consistent under the conditions of varying SNR values.

The reason for the proposed method to provide a detection accuracy higher than that achieved by using the FSPC method can be given as follows. The proposed method eliminates the impossible locations in the first stage based on the partial PDP information gathered from only three slots. The  $M = 12$  strongest paths obtained from the partial PDP evaluation of the first stage are then further assessed in the second stage through a PSFC method. For these 12 locations, the proposed method provides more information about the PDP value than the FSPC method can, since for these 12 locations, the PDP obtained by the proposed method has a value that is equivalent to that given by a FSFC method.



Another reason for superior performance of the proposed method is that it employs a better strategy of distributing its computational requirement between the first and second stages.

It is also seen from Table 7 and that both Method A and Method B of the proposed algorithm yield almost the same results for probability of detection. This is so, since there is no burst noise considered in the transmission. All the 1000 trials (tests) are generated using Gaussian white noise with the slots in the correlation period equally impaired in both the methods.

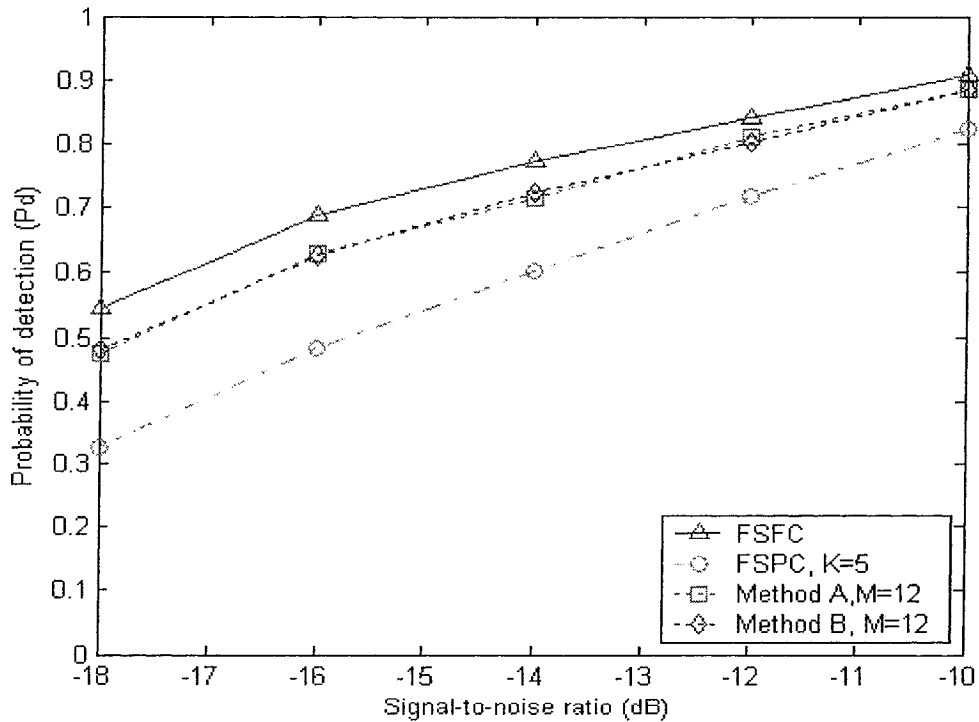


Figure 4.3 Probability of detection in Scenario 1

Another aspect of the performance comparison is the probability of the false alarm ( $P_f$ ). For Scenario 1, the simulation results are listed in Table 8 as well as plotted in Figure 4.4.

It is seen from these illustrations that the probability of false alarm for each of the methods is quite small. A close look of the results in Table 8 and Figure 4.4 suggests that the proposed method has a lower probability of false alarm compared to the FSPC method.

Table 8 Probability of false alarm for Scenario 1

SNR (dB)	FSFC algorithm	FSPC algorithm with $K = 5$	Proposed algorithm with $M = 12$	
			Method A	Method B
-10	0.0033335	0.0035415	0.0027085	0.00375
-12	0.0047915	0.0066665	0.002917	0.0033335
-14	0.008125	0.010417	0.0058335	0.0045835
-16	0.0079165	0.009792	0.004583	0.006875
-18	0.007292	0.011459	0.003542	0.004375

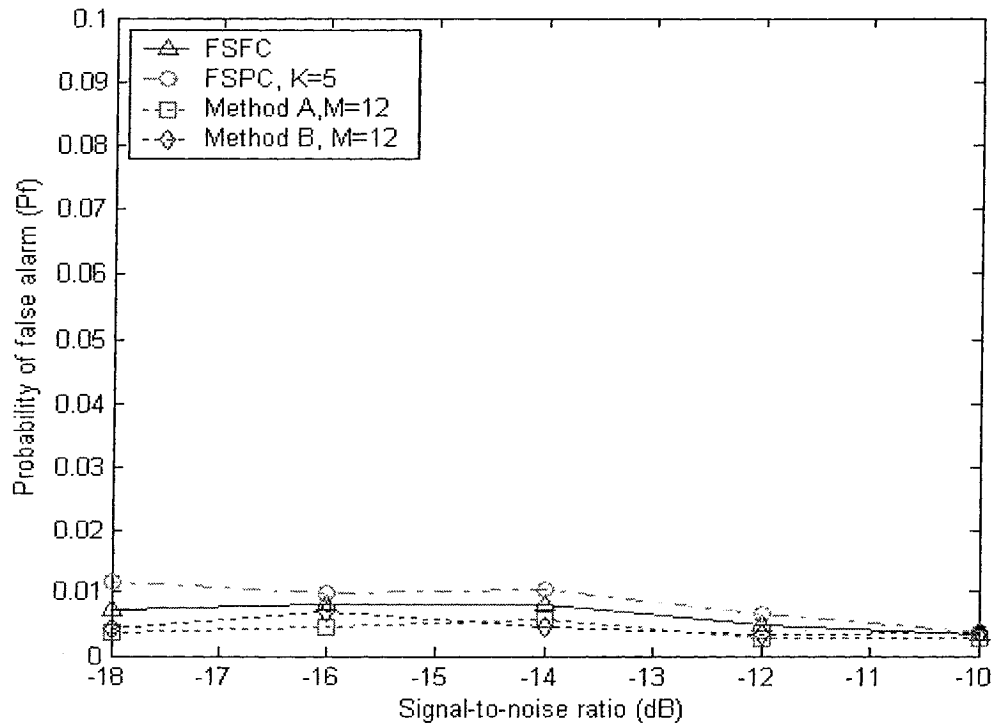


Figure 4.4 Probability of false alarm for Scenario 1

#### 4.4.2 Scenario 2

Simulation results for Scenario 2 are given in Table 9, and the corresponding curves are plotted in Figure 4.5. The results suggest that the proposed method has almost the same probability of detection as that of the FSFC method for all the SNR values considered, and the results are better than the corresponding values obtained by using the FSPC method, and yet, as already seen from Table 5, the computational complexity of the proposed algorithm is the same as that of the FSPC algorithm and only 60% of that of FSFC algorithm. The performance results also indicate that as  $M$  is increased, the performance of the proposed algorithm approaches to that of the FSFC algorithm.

As in the case of Scenario 1, the input data in this scenario is also assumed to be contaminated by the AWGN noise and free of the burst noise. As expected, the performance of Method A and that of Method B are very close to each other.

Table 9 Probability of detection for Scenario 2

SNR (dB)	FSFC algorithm	FSPC algorithm with $K = 9$	Proposed algorithm with $M = 40$	
			Method A	Method B
-10	0.90917	0.87521	0.90687	0.90354
-12	0.84146	0.78458	0.83667	0.83188
-14	0.77333	0.68729	0.76313	0.76417
-16	0.68854	0.58542	0.68083	0.68188
-18	0.54667	0.43	0.53563	0.53208

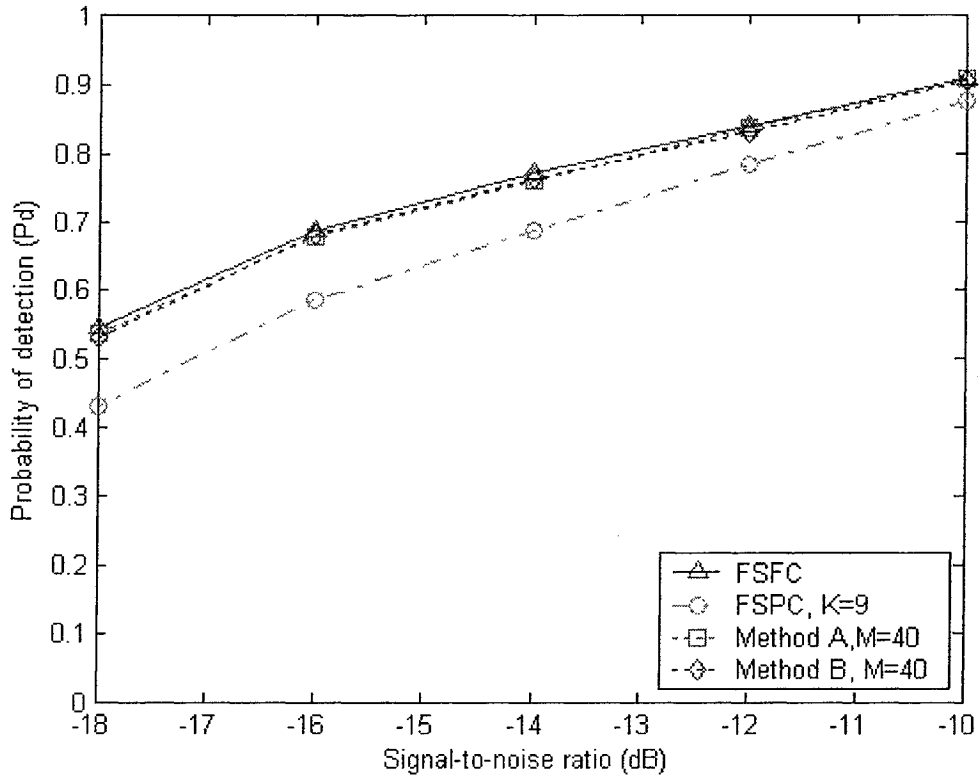


Figure 4.5 Probability of detection in scenario 2

The probability of false alarm is also calculated for Scenario 2 and given in Table 10 and Figure 4.6. From these results it is seen that the false alarm probability results for the various algorithms and for the two scenarios are similar.

Table 10 Probability of false alarm for Scenario 2

SNR (dB)	FSFC algorithm	FSPC algorithm with $K=9$	Proposed algorithm with $M=40$	
			Method A	Method B
-10	0.0033335	0.0025	0.003542	0.0033335
-12	0.0047915	0.004375	0.0047915	0.004583
-14	0.008125	0.0085415	0.006875	0.0077085
-16	0.0079165	0.007708	0.007708	0.0075
-18	0.007292	0.005417	0.0070835	0.006875

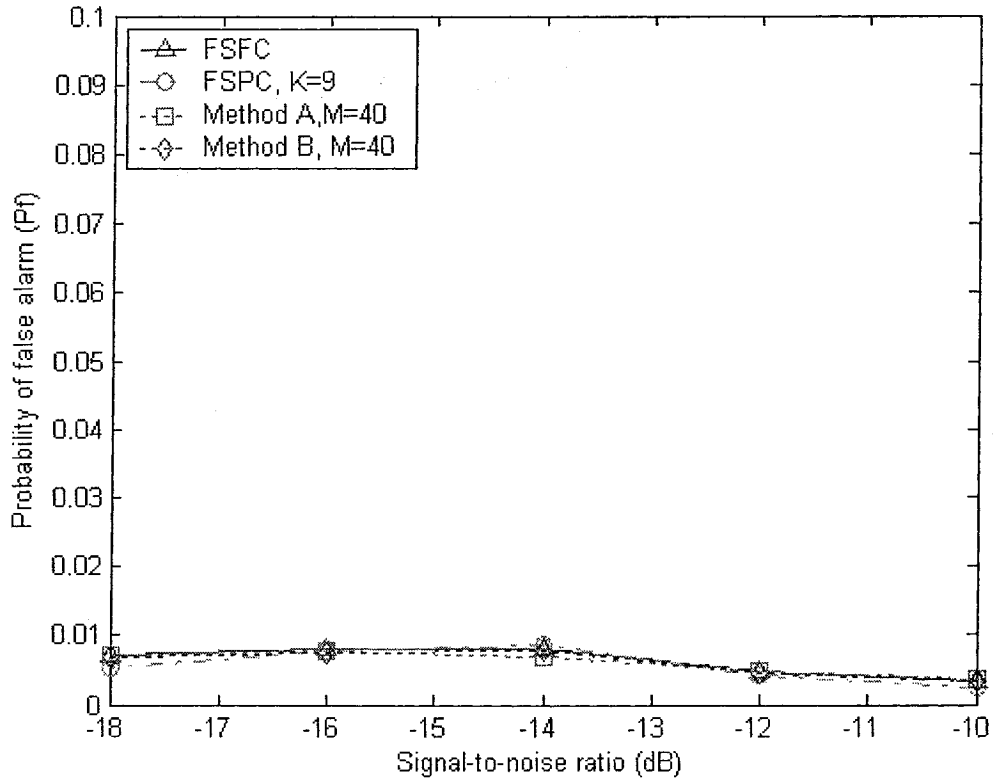


Figure 4.6 Probability of false alarm for Scenario 2

Table 11 shows the percentage improvement in the detection performance of the FSPC and the proposed (Method A and Method B) algorithms in using Scenario 1 instead of Scenario 2.

Table 11 Percentage improvement from Scenario 1 to Scenario 2 in the detection performance of the two algorithms

SNR (dB)	FSPC algorithm	Proposed algorithm	
		Method A	Method B
-10	6.52	2.5	2.22
-12	9.48	3.24	3.82
-14	13.84	6.92	5.55
-16	21.28	7.96	8.96
-18	32.5	12.9	10.5

From Table 11, it is seen that FSPC algorithm gains more in detection performance than the proposed algorithm as we switch from Scenario 1 to Scenario 2. However, it should be pointed out that, since the net performance of the proposed algorithm is higher than that of the FSPC algorithm with the same amount of complexity, the proposed algorithm should be preferable over the FSPC algorithm.

### 4.4.3 Results with Burst noise

In order to examine the performance of Method A and Method B under the presence of the burst noise, the first two slots of the correlation period are corrupted by a factor-2 scaled up version of the AWGN noise that is used to corrupt other slots.

The results for Scenario 1 are shown in Table 12 and Figure 4.7. These results clearly show that Method B is superior to Method A, but the latter is still better than the FSPC algorithm consistently.

Table 12 Probability of detection for Scenario 1 with burst noise

SNR (dB)	FSFC algorithm	FSPC algorithm with $K = 5$	Proposed algorithm with $M = 12$	
			Method A	Method B
-10	0.85959	0.71059	0.73381	0.81833
-12	0.80504	0.6227	0.64615	0.75015
-14	0.72256	0.47256	0.538	0.64763
-16	0.61174	0.34737	0.42267	0.52681
-18	0.45641	0.20793	0.29059	0.37478

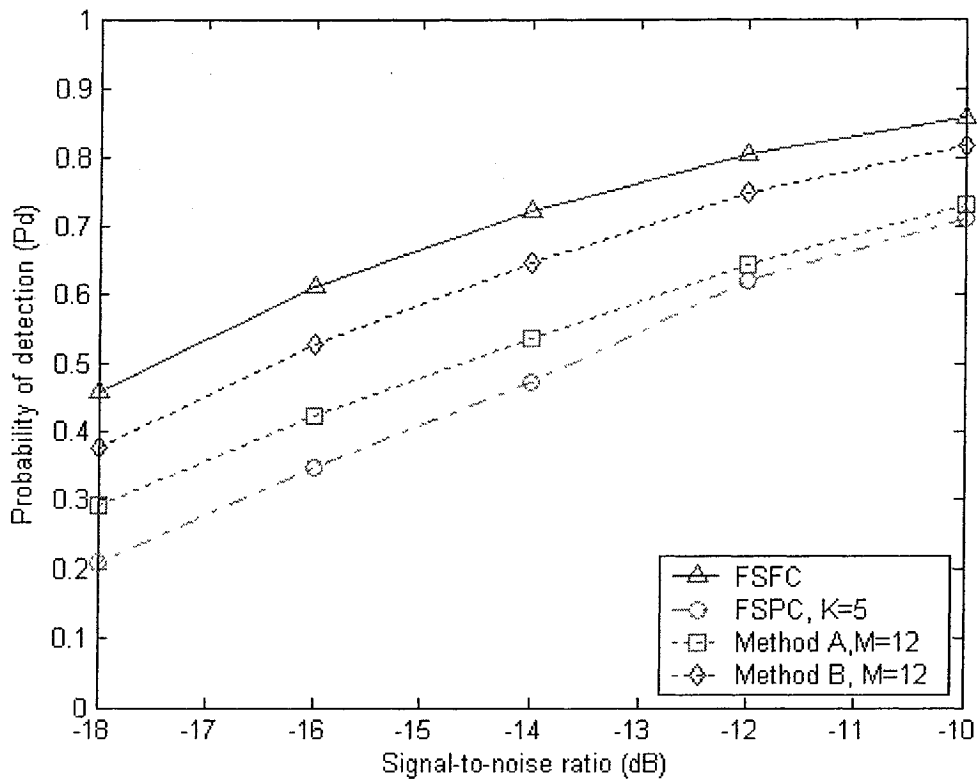


Figure 4.7 Probability of detection for scenario 1 with burst noise

The probability of false alarm for Scenario 1 with burst noise is shown in Table 13 and Figure 4.8. It is seen from the figure and the table that the Method A of the proposed method gives a false alarm probability very similar to that of FSFC method, whereas the Method B has the probability of false alarm less than 1% for all the SNR level tested, which is lower than that of the FSFC method.

The detection performance results for Scenario 2 with burst noise are given in Table 14 and Figure 4.9. Once again, these results corroborate the statement that Method A provides better results under the burst noise compared to Method B. This is because Method B, even though uses the same number of slots for each of the offsets, uses the

slots that are evenly distributed in the correlation period, making it more resistant than Method A to any burst noise in the input data sequence.

Table 13 Probability of false alarm for Scenario 1 with burst noise

SNR (dB)	FSFC algorithm	FSPC algorithm with $K = 5$	Proposed algorithm with $M = 12$	
			Method A	Method B
-10	0.0080368	0.0093336	0.0096668	0.005926
-12	0.0068518	0.011037	0.0075556	0.005
-14	0.0068148	0.010111	0.0067776	0.0037036
-16	0.011444	0.014111	0.010407	0.0060742
-18	0.011593	0.015963	0.011259	0.0065184

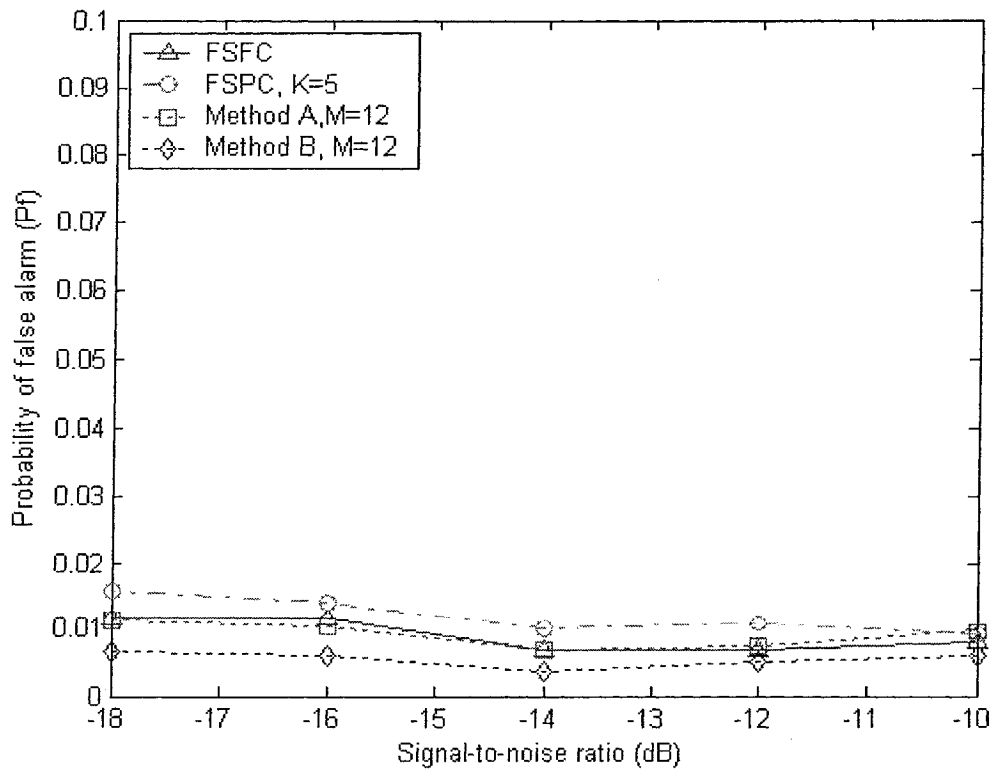


Figure 4.8 Probability of false alarm for Scenario 1 with burst noise



Table 14 Probability of detection for Scenario 2 with burst noise

SNR (dB)	FSFC algorithm	FSPC algorithm with $K=9$	Proposed algorithm with $M=40$	
			Method A	Method B
-10	0.85959	0.77985	0.83433	0.85681
-12	0.80504	0.70807	0.7743	0.79993
-14	0.72256	0.56978	0.68193	0.71456
-16	0.61174	0.44904	0.56815	0.60296
-18	0.45641	0.28733	0.41759	0.44567

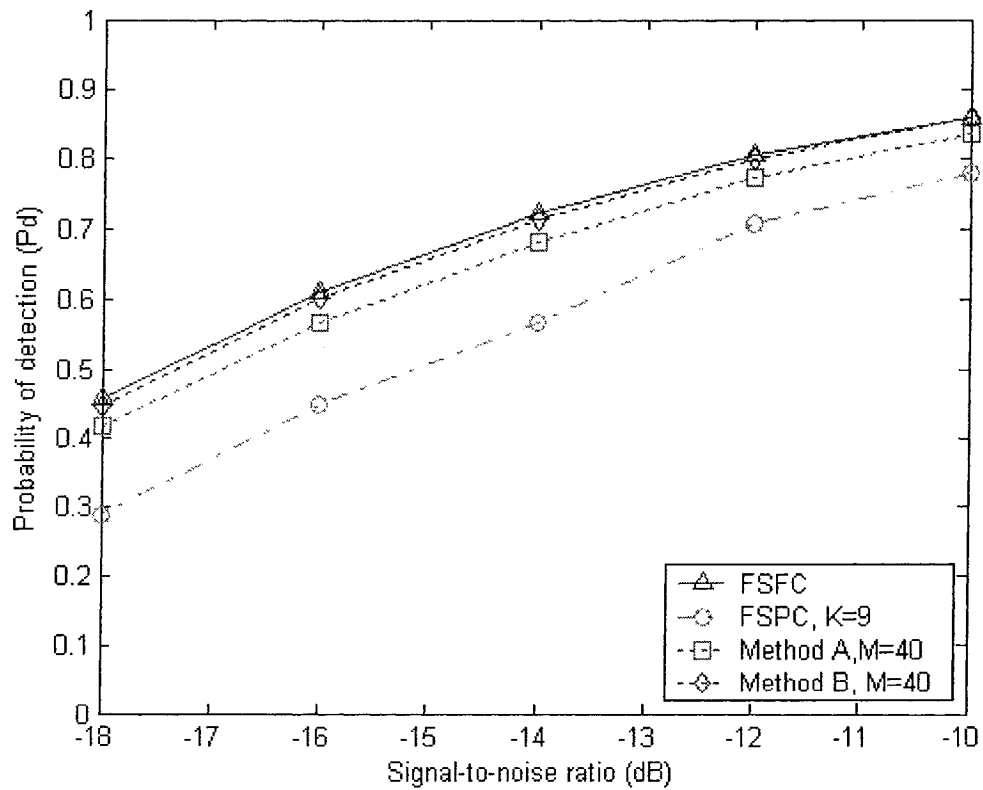


Figure 4.9 Probability of detection for Scenario 2 with burst noise

The probability of false alarm for scenario 2 with burst noise is shown in Figure 4.10 and Table 15. It can be seen from the figure as well as the table that both of the conventional algorithms and the proposed MPS algorithm have the similar false alarm probability, which shows a fair comparison between the different algorithms. By comparing probability curves in Figure 4.10 with that in Figure 4.8, we find that the false alarm probability of Method B increases about 0.2% for Scenario 2. This is due to the number of offsets selected for Scenario 2 ( $M = 40$ ) is larger than for Scenario 1 ( $M = 12$ ); therefore, a higher chance of detecting an invalid path.

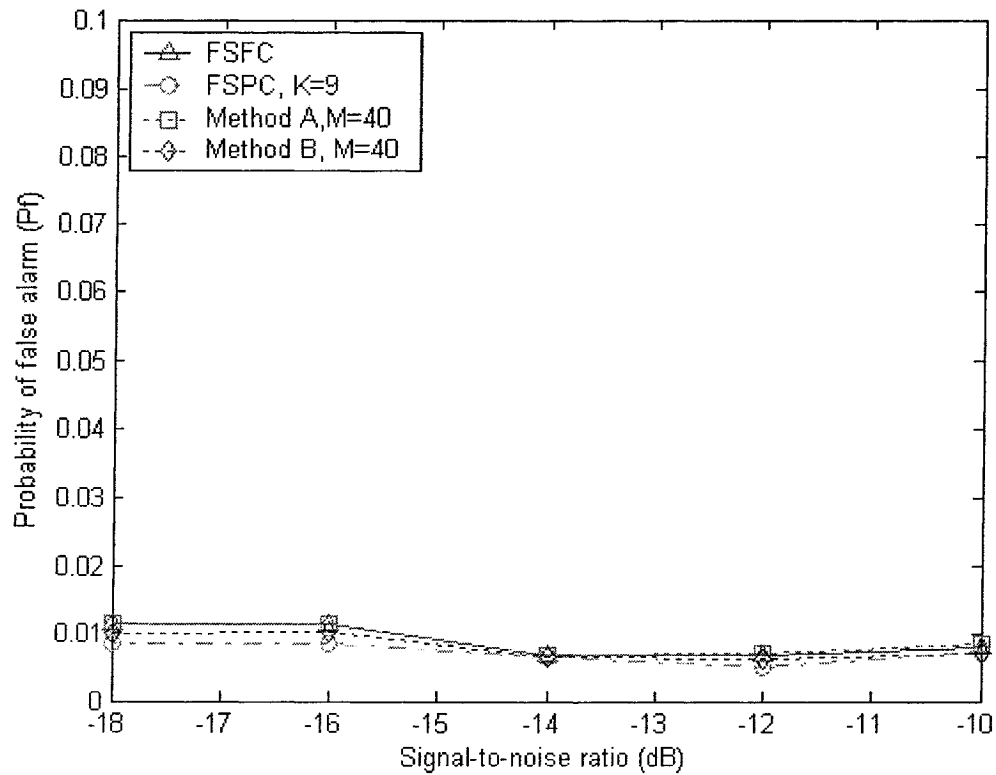


Figure 4.10 Probability of false alarm for scenario 2 with burst noise

Table 15 Probability of false alarm for Scenario 2 with burst noise

SNR (dB)	FSFC algorithm	FSPC algorithm with $K = 9$	Proposed algorithm with $M = 40$	
			Method A	Method B
-10	0.0080368	0.0074816	0.0085186	0.007074
-12	0.0068518	0.0052222	0.007037	0.0064074
-14	0.0068148	0.0066296	0.0068148	0.0065184
-16	0.011444	0.0085926	0.011482	0.010259
-18	0.011593	0.0084816	0.011482	0.010222

## 4.5 Summary

In this chapter, an analysis of the computational complexity, in terms of the number of correlations required, and the performance evaluation, in terms of the detection and false alarm probability, of the proposed algorithm have been carried out. For the purpose of this study, two scenarios, with respect to the number of slots contributing to the correlation operations and the size of the search range, have been chosen. The results of this study have then compared with that of the FSFC and FSPC algorithms. For a meaningful comparison of the performance of the proposed and FSPC algorithms, the parameter values of the two algorithms have been chosen so that both have the requirement of the same computational load.

It has been shown that the proposed algorithm is capable of providing a performance superior to that of the FSPC algorithm and almost the same as that of the FSFC algorithm at a computational load that is 60% of that of the FSFC algorithm.

Finally, in regard to the proposed algorithm, it has been shown that in comparison to Method A, Method B is more resilient in the presence of burst noise, but requires some extra memory for buffering of the data.

## **Chapter 5 Conclusion and Future Work**

### **5.1 Summary**

The multi-path searcher (MPS) of a base station RAKE receiver is responsible for finding and tracking the delay paths of all the users in the cell. The major drawback of the existing multi-path searchers of base-station RAKE receivers is that they are either computationally too intensive or unreliable due to the low detection performance.

The proposed algorithm, in the first stage, screens all the delay offsets in the search window, by obtaining the power delay profile of the signal associated with each delay offset with a contribution only from a small number of selected slots in the correlation period. Based on the way these small numbers of contributing slots are selected. Two methods, Method A and Method B, have been devised for the realization of the first stage. Thus, this process constitutes a full search with partial contribution. Consequently, the search is exhaustive for all the offsets in the search window, but the computations are not too excessive, since only a part of the slots contribute to the correlation computations of this stage. This stage helps in narrowing down the number of offsets for the second stage. In the second stage, only a much smaller number (say  $M$ ) of offsets that corresponds to a certain number of largest PDP obtained in the first stage is used. The paths corresponding to  $M$  selected offsets are re-examined by determining their power delay

profile, this time, by including contributions from all the slots in the correlation period. Thus, effectively the second-stage search is a partial search with full contribution, and therefore, capable of providing a high detection accuracy.

Extensive simulations have been conducted to assess the performance of the proposed algorithm under two different scenarios of the number of offsets selected for the second stage. The performance results are compared with that of the two conventional MPS algorithms, one with a full search and the other with a partial search. The proposed algorithm is shown to provide a probability of detection performance almost the same as that of the conventional full search with full contribution, and at the same time has a substantially reduced computational complexity.

It has been shown that proposed algorithm using Method B is more resilient to burst noise than when using Method A. This is due to the fact that Method B exploits the time diversity of the slots.

Finally, it should be pointed out that the proposed algorithm has a deterministic computational complexity, a significant feature for an ASIC or DSP implementation of an algorithm.

## **5.2 Suggestions for Future Work**

From the two scenarios chosen in this thesis, it is seen that  $M$ , the number of offsets employed in the second stage, has a bearing on the computational complexity and

detection performance of the proposed algorithm. A study can be undertaken for selecting the value of  $M$  to provide a desired balance between the two performance indices.

All the simulations have been conducted in the AWGN environment; further studies could be carried out with a fading environment, a very common phenomenon in wireless applications.

An implementation of the proposed algorithm on digital signal processor such as TMS320C6416 and TMS320C6488 would be an interesting and very rewarding exercise. In fact, TMS320C6488 has a peripheral named RAC which is a general-purpose correlation engine designed just for WCDMA applications.

## Appendix A

### The Structure of Uplink DPDCH and DPCCH

Table A.1 DPDCH fields

Slot Format # $i$	Channel Bit Rate (kbps)	Channel Symbol Rate (ksps)	SF	Bits/ Frame	Bits/ Slot	$N_{\text{data}}$
0	15	15	256	150	10	10
1	30	30	128	300	20	20
2	60	60	64	600	40	40
3	120	120	32	1200	80	80
4	240	240	16	2400	160	160
5	480	480	8	4800	320	320
6	960	960	4	9600	640	640



Table A.2 DPCCH fields

Slot Format # i	Channel Bit Rate (kbps)	Channel Symbol Rate (ksps)	SF	Bits/ Frame	Bits/ Slot	$N_{\text{pilot}}$	$N_{\text{TPC}}$	$N_{\text{TFCI}}$	$N_{\text{FBI}}$	Transmitted slots per radio frame
0	15	15	256	150	10	6	2	2	0	15
0A	15	15	256	150	10	5	2	3	0	10-14
0B	15	15	256	150	10	4	2	4	0	8-9
1	15	15	256	150	10	8	2	0	0	8-15
2	15	15	256	150	10	5	2	2	1	15
2A	15	15	256	150	10	4	2	3	1	10-14
2B	15	15	256	150	10	3	2	4	1	8-9
3	15	15	256	150	10	7	2	0	1	8-15

## Appendix B

### Pilot Bit Patterns for Uplink DPCCH

Table B.1 Pilot bit patterns for uplink DPCCH with  $N_{\text{pilot}} = 3, 4, 5$  and 6

Bit #	$N_{\text{pilot}} = 3$			$N_{\text{pilot}} = 4$				$N_{\text{pilot}} = 5$					$N_{\text{pilot}} = 6$					
	0	1	2	0	1	2	3	0	1	2	3	4	0	1	2	3	4	5
Slot #0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0
1	0	0	1	1	0	0	1	0	0	1	1	0	1	0	0	1	1	0
2	0	1	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1
3	0	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
4	1	0	1	1	1	0	1	1	0	1	0	1	1	1	0	1	0	1
5	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0
6	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	0	0
7	1	0	1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0
8	0	1	1	1	0	1	1	0	1	1	1	0	1	0	1	1	1	0
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	0	1	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1
11	1	0	1	1	1	0	1	1	0	1	1	1	1	1	0	1	1	1
12	1	0	1	1	1	0	1	1	0	1	0	0	1	1	0	1	0	0
13	0	0	1	1	0	0	1	0	0	1	1	1	1	0	0	1	1	1
14	0	0	1	1	0	0	1	0	0	1	1	1	1	0	0	1	1	1

Table B.2 Pilot bit patterns for uplink DPCCH with  $N_{\text{pilot}} = 7$  and 8

Bit #	$N_{\text{pilot}} = 7$							$N_{\text{pilot}} = 8$							
	0	1	2	3	4	5	6	0	1	2	3	4	5	6	7
Slot #0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0
1	1	0	0	1	1	0	1	1	0	1	0	1	1	1	0
2	1	0	1	1	0	1	1	1	0	1	1	1	0	1	1
3	1	0	0	1	0	0	1	1	0	1	0	1	0	1	0
4	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1
5	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0
6	1	1	1	1	0	0	1	1	1	1	1	1	0	1	0
7	1	1	0	1	0	0	1	1	1	1	0	1	0	1	0
8	1	0	1	1	1	0	1	1	0	1	1	1	1	1	0
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	0	1	1	0	1	1	1	0	1	1	1	0	1	1
11	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1
12	1	1	0	1	0	0	1	1	1	1	0	1	0	1	0
13	1	0	0	1	1	1	1	1	0	1	0	1	1	1	1
14	1	0	0	1	1	1	1	1	0	1	0	1	1	1	1

## 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, The Artech House Publishers, 1998.
- [3] T. Halonen, J. Romero, and J. Melero, GSM, GPRS, and EDGE Performance, John Wiley & Sons, Ltd. 2002.
- [4] R. Prasad, W. Mohr, and W. Konhauser Editors, Third Generation Mobile Communication Systems, Artech House Publisher, 2000.
- [5] Trillium Digital System, Inc. "Third Generation (3G) Wireless White Paper", Mar. 2000.
- [6] H. Holma and A. Toskala, WCDMA for UMTS, Third Edition, John Wiley & Sons, Ltd. 2004.
- [7] R. Tanner, and J. Woodard, WCDMA Requirements and Practical Design, John Wiley & Sons, Ltd, 2004.
- [8] S. Yang, 3G CDMA2000 Wireless System Engineering, The Artech House Publisher, 2004.
- [9] C. Smith, 3G Wireless Networks, Second Edition, McGraw-Hill Osborne Media, 2006.
- [10] C. Chevallier, C. Brunner, A. Caravaglia, K. P. Murray, and K. R. Baker, WCDMA (UMTS) Deployment Handbook, Planning and Optimization Aspects, John Wiley & Sons, Ltd, 2006.
- [11] P. Lescuyer, UMTS Origins, Architecture and the Standard, Springer, 2004.

- [12] R. Rice and P. E. Green, "A Communication Technique for Multipath Channels," in *Proc. IRE*, vol. 46, pp. 555-570, Mar. 1958.
- [13] R. C. Dixon, *Spread Spectrum Systems*, Wiley Publishers, 1976.
- [14] 3GPP TS 25.214, "Physical layer procedures (FDD)", version 6.0.0, 2006.
- [15] J. G. Proakis, *Digital Communications, Fourth Edition*, McGraw-Hill Companies, Inc. 2001.
- [16] B. Hwang, "Performance analysis of multipath searcher in WCDMA system," UMTS System Research Lab., LG Electronics Inc. 2004.
- [17] H. Elders-Boll, "WCDMA baseband design faces challenges," *Wireless Systems Design*, January 2003.
- [18] M. C. Damstra, "Path search algorithm for application in W-CDMA systems", Master's thesis, University of Twente, 2004.
- [19] S. Fukumoto, K. Okawa, K. Higuchi, M. Sawahashi, and F. Adachi. "Path search performance and its parameter optimization of pilot symbol-assisted coherent rake receiver for W-CDMA mobile radio," *IEICE Trans. Fundamentals*, vol. E83-A, no.11, pp. 2110-2119, Nov. 2000.
- [20] S. H. Won and Y. J. Kim. "Performance analysis of multi-path searcher for mobile station in W-CDMA system employing transmit diversity," *Electronics letters*, vol. 39 no.1, pp.137-139, Jan. 2003.
- [21] J. Kawamoto, et al, "Path searcher and path searching method", United States Patent Application: 0050255819, Nov., 2005.

- [22] H. Hamada, M. Nakamura, T. Kubo, M. Minowa, and Y. Oishi, "Performance evaluation of the path search process for the WCDMA system," in *Proc. IEEE 49th Vehicular Technology Conference*, vol. 2, pp. 980 – 984, May 1999.
- [23] T. Miyatani, K. Urabe, and Y. Akaiwa, "A reduced-complexity path timing detection method for DS-CDMA," *IEEE 1998 International Conference on Universal Personal Communications*, vol. 1, pp. 357-361, Oct. 1998.
- [24] M. Uzzafer, "Single correlator receiver for wideband multi-path channel," *IEEE 6th CAS Symp. On Emerging Technologies: Mobile and Wireless Comm.* pp. 421-424, 2004.
- [25] X. Lai and H. M. Kwon, "Multipath resolution with PN chip interval," *IEEE 7th Int. Symp. On Spread-Spectrum Tech. & Appl.*, pp. 746-750, Sep. 2002.
- [26] H. Elders-Boll, "Simplified interference-based threshold rule for delay selection in DS-CDMA systems," *The 11th IEEE International Symposium*, vol. 1, pp. 77-81, PPIMRC 2000.
- [27] R. Hamila, Elena S. Lohan, and M. Renfors, "Subchip multipath delay estimation for downlink WCDMA system based on teager-kaiser operator," *IEEE communications letters*, vol. 7, no.1, Jan. 2003.
- [28] E. Sourour, G. Bottomley, and R. Ramesh, "Delay tracking for direct sequence spread spectrum systems in multi-path fading channels," in *Proc. IEEE 49th, VTC*, pp. 422-426, 1999.
- [29] C. Cozzo, G. E. Bottomley, and A. S. Khayrallah. "Rake receiver finger placement for realistic channels," *IEEE Communications Society*, pp. 316-321, WCNC 2004.

- [30] D. Banerjee. and A. Paulraj, "Location-aided RAKE receiver finger assignment," in *Proc. IEEE Global Telecommunications Conference*, pp. 1043-1047, 2003.
- [31] M. Abou-Khousa, M. El-Tarhum, and A. Ghayeb, "A novel finger assignment algorithm for RAKE receivers in CDMA systems," in *Proc. IEEE Intern. Conf. Commun. (ICC)*, Paris, June 2004.
- [32] M. Abou-Khousa, A. M. Ghayeb, and M. El-Tarhum. "On multipath detection in CDMA systems," *2005 IEEE International Conference on Communications*, vol. 3. pp. 2097-2101, 2005.
- [33] B. Vejlgard, P. Mongensen, and J. Knudsen, "Grouped rake finger management principle for wideband CDMA," in *Proc. IEEE VTC*, pp. 87-91, 2000.
- [34] Kim-Chyan Gan. "Path searcher for a WCDMA rake receiver," Freescale Semiconductor, Inc. Rec.3, 2005.
- [35] S. Sriram, et al, "A 64 channel programmable receiver chip for 3G wireless infrastructure," in *Proc. IEEE 2005 Custom Integrated Circuits Conference*, pp. 59-62, Sept. 2005.
- [36] E. Bejjani, J.-F. Bouquier, and B. de Cacqueray. "Adaptive channel delay selection for WCDMA mobile system," in *Proc. of IEEE VTC '99*, pp. 203-207, Amsterdam, The Netherlands, 1999.
- [37] U. Madhow and M. Pursley, "mathematical modeling and performance analysis for a two-stage acquisition scheme for DS-SS-SSA," *IEEE Trans. Commun.*, vol. 43, pp. 2511-2520, Sept. 2005.

- [38] 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.
- [39] D. Kim, Y. Park, and W. Kim, "A design of high speed multi-path searcher using dual scrambling code generators for WCDMA," Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual, vol. 3, pp. 1685 – 1688, April 2003.
- [40] E. Grayver, et al. "Design and VLSI implementation for a WCDMA multipath searcher," *IEEE Transactions on Vehicular Technology*, vol. 54. no. 3, pp.889-902, May 2005.
- [41] 3GPP TS 25.211, "Physical Channels and Mapping of Transport Channels onto Physical Channels (FDD)," V7.2.0, May 2007.
- [42] 3GPP TS 25.212, "Multiplexing and Channel Coding", v7.5.0, May 2007.
- [43] 3GPP TS 25.213, "Spreading and Modulation (FDD)," V7.2.0, May 2006.
- [44] J. Schiller, *Mobile Communications*, Second Edition, Addison-Wesley, 2003.
- [45] R. Giltlin, J. Hayes, and S. Weinstein, *Data Communications Principles*, Springer, 1992.
- [46] L. Milstein, J. Gevargiz, and P. Das, "Rapid acquisition for direct sequence spread spectrum communications using parallel SAW convolvers, " *IEEE Trans. Communi.*, vol. COM-33, pp. 593-600, July 1985.
- [47] Min Yang, M.O. Ahmad, and M.N.S. Swamy, "A two-stage multi-path searcher for uplink WCDMA base station, " in *Proc. IEEE-MWSCAS/NEWCAS'07*, Montreal, pp. 1229 – 1232, Aug. 2007.