Simulations of Blind and Non-Blind Channel Estimation Techniques for Bluetooth Systems

Hua Wu

A Thesis in

The Department of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montréal, Québec, Canada

© Hua Wu, August 2004

Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-94715-7
Our file Notre référence
ISBN: 0-612-94715-7

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

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

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement 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

Simulations of Blind and Non-Blind Channel Estimation Techniques for Bluetooth Systems

Only few years after Bluetooth technology emerged as a new wireless technology, it has grasped the focus of top manufactures in the world and got widespread use. Indeed, it is an excellent solution to replace cumbersome cables, which used to carry information between devices. Like other wireless technologies, fading and time-varying channels greatly degrade the performance of Bluetooth communications. In the first part of this thesis, Bluetooth specifications are introduced. This is followed by channel estimation techniques applicable to Bluetooth systems. Bluetooth adopts slow frequency hopping modulation. Hence, it is possible to implement channel estimation at the hop beginning, and then compensate for the impairment during the later header and payload parts of data packets. Classical channel estimation techniques are discussed and compared in the Bluetooth case, such as LMS, LS, MMSE, and blind channel estimation algorithms. We derive the specific channel estimation equations for each channel estimation technique. In the last chapter, the simulation results of these channel estimation techniques are presented in various fading cases, i.e. slow and fast fading.

Acknowledgements

Firstly, my sincerest thanks go to my supervisor Dr. A.K. Elhakeem. His continuous guidance and helpful advices made the thesis possible. I always benefited from his rich knowledge and experiences in the wireless area when I discussed my research with him. Besides, I am most grateful for his kindness and patience with me throughout this thesis.

Furthermore, I would like to thank my friends Xiaoming Lai and Renyuan Li for their frequent suggestions in the wireless area.

Finally, I am most greatly indebted to my husband and my lovely daughter Shirley. Their understanding and encouragement supported me to complete the thesis.

Table of Contents

List of Abbreviations

ACL	Asynchronous Connectionless
AWGN	Additive White Gaussian Noise
BD_ADDR	Bluetooth Device Address
BER	Bit Error Rate
BIG	Bluetooth Special Interest Group
BT	Bandwidth Time
CAC	Channel Access Code
CRC	Cyclic Redundancy Check
DAC	Device Access Code
DIAC	Dedicated Inquiry Access Code
ETSI	European Telecommunications Standards Institute
FHSS	Frequency-Hopping Spread-Spectrum
GFSK	Gaussian Frequency Shift Keying
GIAC	General Inquiry Access Code
GMS	Global System for Mobile
HCI	the Host Controller Interface
IAC	Inquiry Access Code
IrDA	Infrared Data Association
ISM	Industrial, Scientific, and Medical
LAP	Lower Address part
L2CAP	Logical Link Control and Adaptation Protocol
LMP	Link Manager Protocol

LMS	Least Mean Square
LOS	Line Of Sight
LS	Least Square
MAC	Media Access Control
MMSE	Minimum Mean-Square Error
OLOS	Obstructed Line Of Sight
OSI	Open System Interconnection
PHY	Physical
RF	Radio Frequency
rms	root mean square
RSSI	Received Signal Strength Indication
SCO	Synchronous Connection-Oriented
SDP	Service Discovery Protocol
SNR	Signal to Noise Ratio
TCS	Telephony Control Specification
TDM	Time Division Multiplexed

List of Figures

List of Tables

Chapter 1

Background on Bluetooth Technology

The advancement of VLSI technology in the last decades has boosted widespread use of personal, portable, handheld electronic devices like laptop, palmtop, personal data assistant and cell phone. Presently, these devices hold weak capability of networking with desktop computers or other units. The reason that limits information exchange between these diversified devices is the cumbersome cable connectivity. Before we consider the connectivity between desktop computers and their portable units, one can see that the computer is already wrapped around by cables, which connect to a printer, a scanner, a keyboard. If we have multimedia computer with internet ability, there are more cables for a microphone, speakers, an external modern, and web camera. At the moment, any additional applications added to the computer entails even more cables. Let's face it, cables has become a big hassle for communications. Bluetooth wireless technology, however, is working the way to get rid of wires and provide free communications between them. It has developed a solution to using short-range radio link to replace the cables. In this way, devices can seamlessly connect and communicate to others, which are within a short range, without thinking of connectivity. The Bluetooth Special Interest Group (SIG) [17] is dedicated to a new technology of cable replacement. Some of the fundamental precepts of the creation of Bluetooth technology within the unlicensed 2.5 GHz reserved for Industrial, Scientific, and Medical (ISM) band are to enable low-cost, low power capability in most existing devices. Including not only personal units but also many industrial applications like monitoring and controlling units [24]. Eventually, it can realize connectivity of everything to

everything. Hence, Bluetooth technology absolutely has its promise and ubiquitous application areas.

1.1 Introduction to Bluetooth Wireless Technology

Bluetooth is a cable replacement technology to connect devices in very short rang area i.e. <10m. It is widely applied to wireless header set, wireless mouse, wireless microphone, as well as communication between laptop with palmtop. The kind of technology was first considered and researched by Ericsson Company. When Ericsson, Norkia, IBM, Intel and Toshiba founded the Bluetooth Special Interest Group (SIG) in May 1998, the SIG started to be responsible for the work of developing Bluetooth technology specification [1]. They released Bluetooth specification in July 1999, and then SIG was extended to nine promotion members in December 1999, which remained unchanged till now. The new four members are Microsoft, Lucent, 3Com, and Motorola. The Bluetooth specification includes the protocols that allow for the development of interactive services and applications that operate over wireless links established with interoperable radio modules. Various usage models for Bluetooth wireless technology were also envisioned by the SIG, and the specification of these models. [1]

In Figure 1.1 [4], all protocols cooperate to implement the various services required by the devices. Logically, these protocols can be grouped into three parts from lowest layer to highest layer: the transport protocols, the middleware protocols and the application part. In the following we explain the basic components of each.

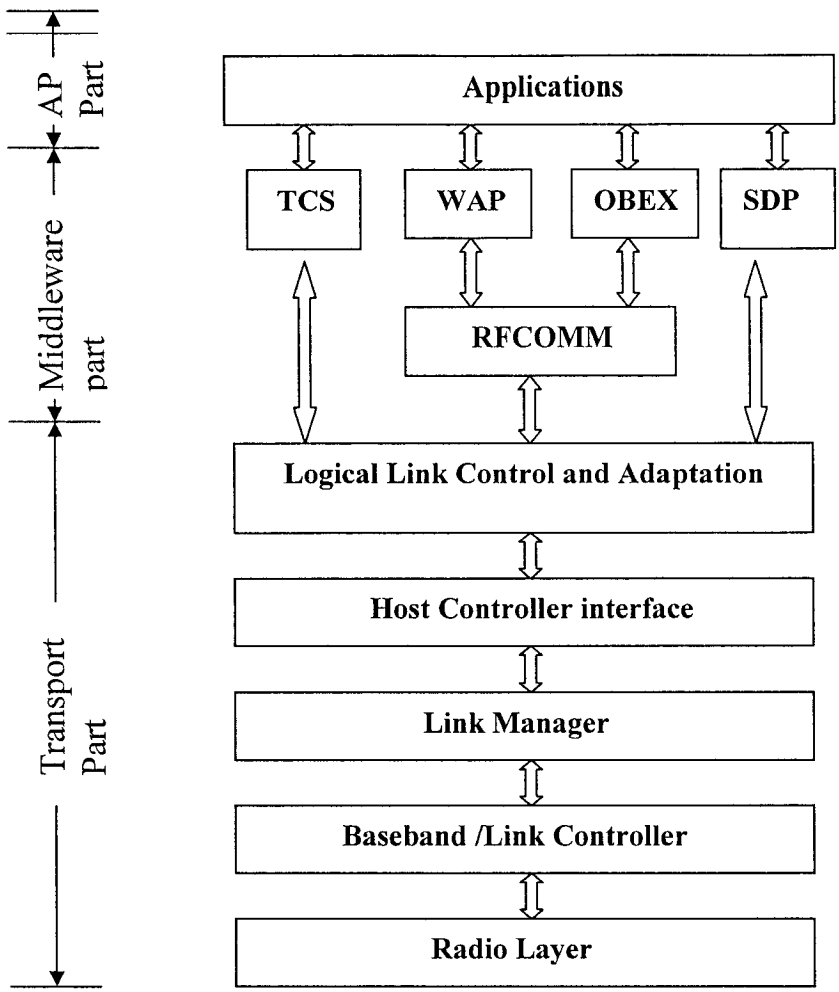


Figure 1.1 Bluetooth protocol stacks

1. 1. 1 The Transport Part of the Bluetooth Protocol Stack

The transport part is principally responsible for creation, configuration and management of both physical and logical links that provide channels for higher layer protocols and applications to pass data. It also defines the methods for Bluetooth units to find and locate each other. We note that the protocols in the transport part do not belong to the transport layer of the seven-layer Open System Interconnection (OSI) protocol models because this part primarily involves link-level transport that enable the transmitting between devices. The protocols in this part

include the radio layer, baseband layer, logical link and adaptation (L2CAP), link manager(LMP),the host controller interface(HCI).[3]

- **Radio layer**

Radio layer defines a physical channel in the air. This air channel transmits information between Bluetooth devices without visible cables. It specifies he frequency band of transmission, channel arrangement, modulation technique, power limitation, and receiver sensitivity.

- **Baseband layer**

The baseband layer enables the physical Radio Frequency (RF) link between Bluetooth units in a piconet. The Bluetooth RF system uses a time division multiplex and frequency-hopping spread-spectrum technology, in which packets are transmitted in defined time slots across defined frequencies, Synchronizing the transmission-hopping frequency and clock of different Bluetooth devices of a piconet (small collection of Bluetooth devices), is implemented on this layer via inquiry and paging procedures. [1]

This layer provides the two different kinds of physical links with their corresponding baseband packets: Synchronous Connection-Oriented (SCO) and Asynchronous Connectionless (ACL), which can be transmitted in a multiplexed manner on the same RF link. These two types of links pertain to different services requirement: time-bounded and non time-bounded. Data packets are running over the ACL link, while audio only or a combination of both audio and data are running over the SCO link. All audio and data packets can be provided with different levels of error correction and can be encrypted to ensure privacy. In addition, the link management and control messages are each allocated a special channel by baseband layer.

Packets containing audio data can be transferred between one or more Bluetooth devices, making possible various usage models. The audio data in SCO packets are routed directly to and from Baseband, and do not go through Logical Link Control and Adaptation Protocol (L2CAP). The audio model is relatively simple within the Bluetooth specification; any two devices can send and receive audio data between each other just by opening an audio link.

- **Link Manager Protocol (LMP)**

LMP is responsible for link setup and control between Bluetooth devices; it includes the control and negotiation of baseband packet sizes .It is also used for security: authentication and encryption; generating, exchanging, and checking link and encryption keys. Besides, the LMP controls the power model and duty cycles of the Bluetooth radio device, and the connection states of a Bluetooth unit in a piconet. LMP messages are filtered out and interpreted by the link manager on receive side, so they are never passed to higher layers. LMP messages have higher priority than user data. If a link manager needs to send a message, it will not be delayed by L2CAP traffic. In addition, LMP messages are not explicitly acknowledged, since the logical channel provides a reliable enough link, making acknowledgments unnecessary [1].

- **Logical Link Control and Adaptation Protocol (L2CAP)**

L2CAP provides the interface to both the baseband and several higher-level protocols. It supports higher-level protocol multiplexing, packet segmentation and reassembly, and Quality and Service (QoS). L2CAP permits higher-level protocols and applications to transmit and receive data packets up to 64 kilobytes in length. Although the baseband protocol provides the SCO and ACL link types, L2CAP is defined only for ACL links and no support for SCO links

is planned. Voice-quality channels for audio and telephony applications are usually run over baseband SCO links. However, audio data may be packetized and sent using communication protocols running over L2CAP [1].

• Host Controller Interface (HCI) Layer

In order to access Bluetooth hardware (or Bluetooth module) for a host, the specification defines the HCI layer that provides a uniform interface to transport the host's commands to the Baseband controller and link manager. It is comprised of HCI controller, HCI Transport layer and HCI driver.

1. 1. 2 The Middleware Part of the Bluetooth Protocol Stack

The protocols covered by the middleware part are Service discovery protocol (SDP), RFCOMM, a set of Infrared Data Association (IrDA) interoperability protocols, and Telephony Control Specification (TCS). These protocols primarily present standard interface to a set of applications for peer-to-peer files transferring. With these protocols, the details of link level transport protocols are hidden from applications. Therefore, most of existing applications can apply to Bluetooth. In view of the seven-layer Open System Interconnection (OSI) protocol model, this part of Bluetooth protocols stack corresponds with the transport layer and network layer of OSI. Some protocols in this part are adopted from the third part and industry standard protocol to extend the Bluetooth application. [3]

• Service Discovery Protocol (SDP)

Discovery services are an important element in the Bluetooth framework since they provide the basic for all the usage models. Using SDP, device information, services, and the characteristics of the services can be queried. Having located available services within the vicinity, the user may select from any of them. After that, a connection between two or more Bluetooth devices can be established.

- **RFCOMM**

RFCOMM is a serial line emulation protocol based on a subset of the European Telecommunications Standards Institute (ETSI)'s Technical Standard (TS) 07.10, which is also used for Global System for Mobile (GMS) communication devices. The RFCOMM protocol provides emulation of RS-232 serial ports over the L2CAP protocol. This "cable replacement" protocol emulates RS-232 control and data signals over the baseband, providing both transport capabilities for upper level services (e.g., OBEX) that use serial line as the transport mechanism. [1]

- **IrDA Interoperability Protocols**

Infrared technology is an alternative wireless communication technology making use of the invisible spectrum of light just beyond red in the visible spectrum. In fact, the standard and the specification of this technology are not specified by SIG but by Infrared Data Association (IrDA). One reason why SIG has chosen to adopt some of the IrDA protocols is that the IrDA has similarities to Bluetooth in their lower layers so that IrDA's OBEX protocols is ideally suited to transferring objects between Bluetooth devices. This does not mean that devices with Bluetooth wireless communication can communicate directly with IrDA devices, but rather it

refers to protocols that enable common applications to use either form of wireless communication. The other reason is that adopting existing standards means there is a large use of available support for the standard [4].

• Telephony Control Specification (TCS) layer

For the purpose of voice service application, the SIG develop the Bluetooth telephony control specification (TCS) to support telephony functions, including call control and group management. The TCS protocols are compatible with the international Telecommunications Union-Telecommunication (ITU_T) Q9.31 specification. They are involved all functions to start a call, such as setting up the call parameter, service connection and disconnection, and establishing audio channel to carry the call's voice content [3]. TCS might also be used to set up data calls, such as would be used with the dial-up networking profile; in this case, the call control is then carried as standard data packets over L2CAP.

• Wireless Access Protocol (WAP)

Wireless Access Protocol (WAP) is a wireless protocol that allows mobile devices to use data service and access the Internet. It uses a combination of Internet protocol such as UDP and protocols specially modified to work with mobile devices i.e. Wireless Markup Language (WML), which is based on XML, the same standard on which Hypertext Markup Language (HTML) is based.

The WAP components used above the Bluetooth protocol stack are [4]:

- WAE (Wireless Application Environment) - Provides a user interface, typically a mico-browser, which is a lightweight version of a Web browser.

- WSP (Wireless Session Protocol) – Supports the session between a WAP client and the WAP server.
- WTP (Wireless Transport Protocol) - Provides a reliable transports layer for WSP. If other layers below WSP provide a reliable service, there is no need for WTP.
- WTLS (Wireless Transport Layer security) – Provides security.

1. 1. 3 The Application Part of Bluetooth Protocol Stack

The application part resides the top of the stack to accomplish various usage scenarios. In fact, this part is not exactly defined by the SIG because it refers to the software supplied by device manufacturers, independent software vendors or others that exercises the protocol stack to accomplish some function that benefits the user of a Bluetooth device with applications. Although the middleware protocols can provide all required applications, they just provide necessary functions to accomplish the application. Users still need to develop some software to enrich its product, such as user's interface or differentiating the product with others. Hence, those protocols closely linked to applications in Bluetooth stack are not included in this part. The SIG still has released some profiles to guide this kind of software development. [3]

In this thesis, we are only interested in channel estimations techniques which only relates to the layers of radio and baseband. Therefore, the following introductions of Bluetooth will just involve these two parts.

1. 2 The Specification of Bluetooth Radio Interface

Bluetooth specification defined its physical interface by the radio layer. The SIG chooses unlicensed ISM band at 2.4 GHz for the Bluetooth carrier band because it is globally available for license-free use. This band is reserved for general use by Industrial, Scientific, and Medical (ISM) applications worldwide. Also, the ETSI ETS 300-328 in Europe or FCC CFR47 Part 15 in the U.S. specified some regulations to this band user on the power and spectral emission and interference. Although globally available, the utilizable resources of the ISM band have slight differences in different countries [4] [5]. Europe and the United States allocate 83.5 MHz to this band, but France reduces the frequency band. For this reason, 79 channels at 1MHz are defined for Europe and the U.S., and 23 channels at 1MHz for France. Table 1.1 gives the details for the frequency band allocation.

- **Frequency Hopping**

The Bluetooth devices occupy these 79 channels by a frequency-hopping spread-spectrum (FHSS) mechanism. The system first calculates a pseudorandom hopping sequence

Geograhy	Frequency Bands	RF Channels
USA, Europe and most other countries	2.400 -2.4835 Ghz	f=2402+k MHz, k=0,…,78
France	2.4465-2.4835 Ghz	f=2454+k MHz, k= 0,…,22

Table 1.1 Bluetooth Operating frequency bands

, and then shifts the frequency carrier through the 79 channels by the pseudorandom sequence at 1600 hops/second. There two advantages of adopting the FHSS technique for Bluetooth. First, it is effective against interferences from other sources in the ISM band. Since ISM band is license-free, there many other applications share the ISM band with Bluetooth and make this band busy. It is always possible for a radio channel to become temporarily blocked by an

interference source. In this case, the receiver will require the transmitter to retransmit the lost data packets. This retransmission action will operate on a new channel, which is unlikely to be blocked too. Indeed, the algorithm employed to calculate the hop sequence ensures maximum distance between adjacent hop channels in the sequence [19]. Also several active Bluetooth piconets may be with range of each other; with each piconet hopping independently with a pseudo random sequence based on each piconet's identity /access code, collisions will be minimized. This is important as all Bluetooth devices only have 79channels in which to operate [4]. Secondly, FHSS can provide secure communication. The algorithm for pseudo random sequence only related to the Bluetooth device address (BD_ADDR), then each device has its unique sequence. Without the sequence knowledge tracing the communication is almost impossible.

- **Modulation**

Bluetooth chooses Gaussian Frequency Shift Keying (GFSK) as its modulation scheme for each signaling data at the RF end. GFSK is a simple binary modulation scheme, which may be viewed as a derivative of FSK. In GFSK, the message bit stream first pass through a pre-modulation Gaussion pulse-shaping filter to generate Gaussian shaped waveform, then sent to the following FSK modulator [6]. Figure 1.2 shows the block diagram. Baseband Gaussian pulse shaping smoothes the frequency transitions, and hence stabilizes the instantaneous frequency variations over time. This has the effect of considerably reducing the sidelobe levels in the transmitted spectrum. Therefore, it allows better spectral efficiency and less inter-symbol interference.

In specification, the modulation of GFSK has a bandwidth time (BT) = 0.5 and a modulation index of between 0.28 and 0.35. The BT product is the product of adjacent signal

frequency separation (0.5 MHz) and symbol duration (1μs), A BT product of 0.5 corresponds to the minimum carrier separation to ensure [4] [11]

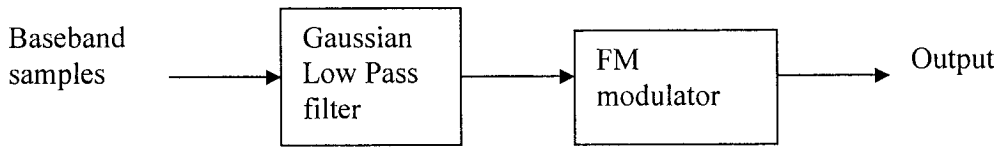


Figure 1.2 GFSK modulation block

orthogonality. A binary 1 is represented by a positive frequency deviation, and a binary 0 is represented by a negative frequency deviation.

· Transmitters

Bluetooth is a technology focusing on the application of short-range and power saving. So, it has some specifications on transmitters.

There are three power levels defined for the Bluetooth transmitter. A lower power level covers the shorter personal area within a room, and a higher power level can cover a medium range. That is described in the Table 1.2.

In order to minimize power consumption, each device can vary its transmit power on the Received Signal Strength Indication (RSSI). [11] This power control operates by a receiver. The receiver measures RSSI, then sends Link Manage Protocol (LMP) control commands back to the transmitter to require their transmit power to be reduced or increased if the RSSI value is higher or lower than that strictly necessary to maintain a satisfactory link. Equipment with power control capability can optimize the output power in a link.

Power Class	Output Power(Max)	Output Power(Min)	Transmission range
1	100 mW(20dBm)	1 mW (0dBm)	<100 m
2	2.5 mW(4 dBm)	0.25 mW (-6dBm)	<10 m
3	1 mW(0 dBm)	N/A	<10 cm

Table 1.2 Bluetooth Power Class

1.3 The Baseband Protocols of Bluetooth

The Bluetooth baseband constructs its packets based on the length of Bluetooth channel and directly passes it to the radio layer for an orderly exchange of data between devices. [5] It implements Bluetooth's physical (PHY) and media access control (MAC) processing. This includes the tasks such as a connection establishment and maintenance within one packet, asynchronous and synchronous links processing, low level timing control, authentication, channel coding and decoding.

1.3.1 Physical Channel

Bluetooth is a Time Division Multiplexed (TDM) with frequency hopping system. The channel is divided into time slots, where each slot is 625 μs in length and corresponds to an RF hop frequency. The nominal frequency hop rate is 1600 hops/s. Each piconet has a unique pseudo random hopping sequence, which is derived only from the Bluetooth device address of the master [17]; the Bluetooth clock of the master determines the phase in the hopping sequence. All Bluetooth units participating in the piconet are time- and hop-synchronized to the channel.

1.3.2 Piconets, Masters, and Slaves

The Bluetooth network supports both point-to-point and point-to-multi-point connections. Due to the Bluetooth unit's random and mobility, the network is set up with an ad hoc structure. All devices enter or leave the network constantly. So, there is no existing infrastructure for it. In an area, all devices are organized into groups of two to eight devices called piconets. Each piconet is an arbitrary collection of Bluetooth-enabled devices, which are physically close enough to be able to communicate and are exchanging information in a regular way [18] [20]. Only one master exists in a piconet to control one or up to seven slaves belonging to the piconet. Figure 1.3 illustrates the piconet structure.

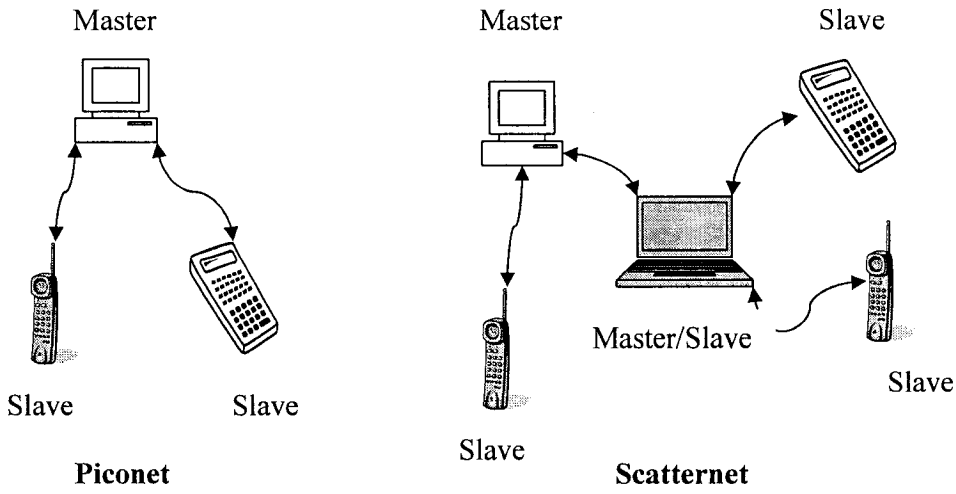


Figure 1.3 The structure of piconet and scatternet

Although Bluetooth devices communicate with peer-to-peer manner in an ad hoc network, devices act either as masters or as slaves at any one time. Usually, the master controls the piconet and initiates an exchange of data; slaves respond to the master described in Figure 1.4. The master and one of the slaves in the piconet take turns transmitting, so two or more transmitters never send at the same time within a piconet. In connection, masters send packets at

even-number slots such as 0,2,4,6, while slaves are listening for the message and receiving the packet which contains its own address. On the following slots such as 1,3,5,7, the slave which has just received the packet from master transmits to the master [23], which is depicted in Figure1.4.

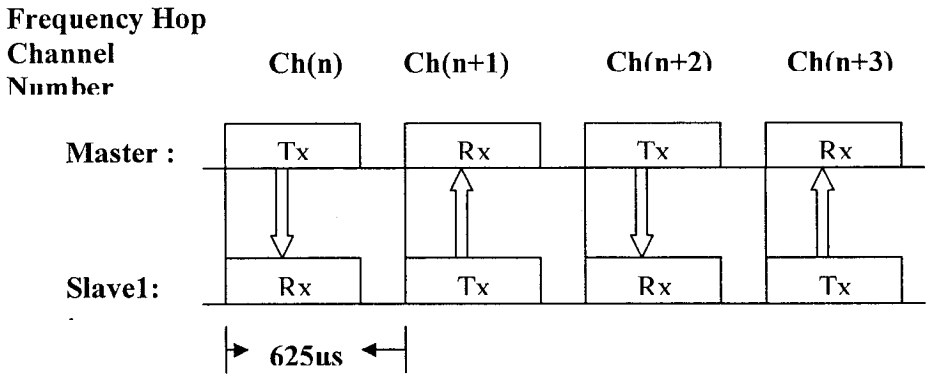


Figure 1.4. Slot timing for single-slot packet

Bluetooth also defines data packets which are 1, 3, 5 slots long. Figure 1.5 illustrates how the timing changes slightly for these multi-slot packets [2].

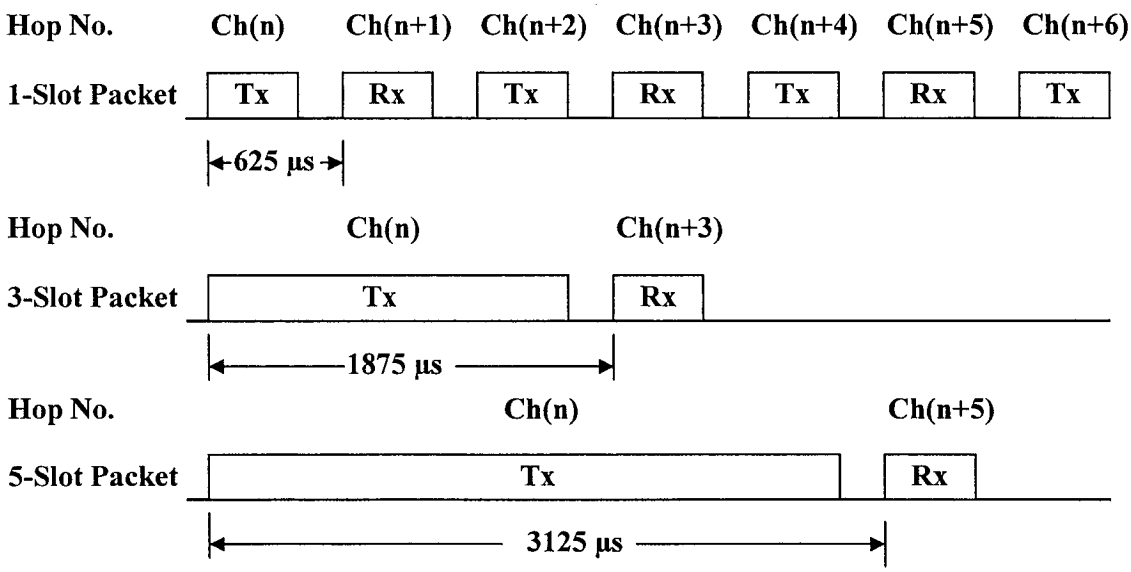


Figure 1.5 Slot timing for multi-slot packets

1.3.3 Physical Links: ACL and SCO

The links set up between masters and slaves has two types: Synchronous Connection-Oriented (SCO) link and Asynchronous Connection-Less (ACL) link. The SCO link is a point-to-point link between a master and a single slave in the piconet. It is maintained by the master using reserved slots at regular intervals. The ACL link is a point-to-multipoint link between the master and all the slaves participating on the piconet. The master establishes an ACL link on a per-slot basis to any slave in the slots not allocated for the SCO links [5].

A. SOC

A SCO link provides a symmetric link between a master and a slave with reserved channel bandwidth and regular periodic exchange of data in the form of reserved slots. Actually, it is a circuit-switched connection, which is typically used for time-bounded information such as a voice transmition [11]. A master can support up to three SCO links to the same slave or to different slaves. A slave can support up to three SCO links from the same master. Due to the time-bounded nature of SCO data, SCO packets are never retransmitted.

B. ACL

A master can establish ACL links on all slots except the slots of SCO. The ACL link provides a packet-switched connection between the master and all active slaves participating in the piconet. Both asynchronous and isochronous services are supported. Non time-bounded information is usually carried on the ACL link. Between a master and a slave only a single ACL link can exist. For most ACL packets, packet retransmission is applied to assure data integrity.

All ACL links in a piconet are controlled by the master. The master decides which slave it will transmit to base on the packet passed from L2CAP or LMP layers. A slave may only

respond with an ACL packet in the next slave-to-master slot if it has been addressed in the preceding master-to-slave slot. If the slave fails to decode the slave address in the packet header, it does not know whether it was addressed or not and so is not allowed to respond. Broadcast packets are ACL packets, which are not addressed to a specific slave and so are received by every slave [11].

1.3.4 Packet Formation

Several types of packets exist in baseband such as ACL packets and SCO packets. Of these types, the part of a packet beginning is identical, which are access code and header. Figure 1.6 illustrates the general structure of baseband packets [11].

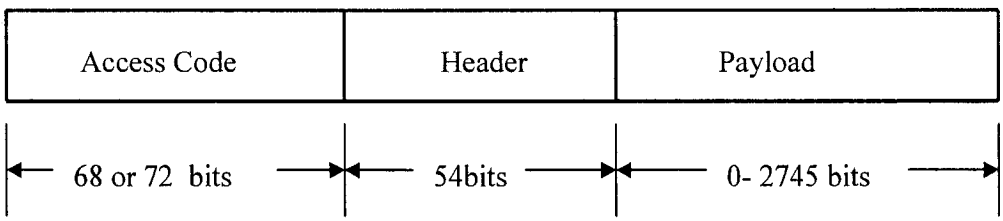


Figure 1.6 Bluetooth general packet structure

- **Access code**

When several piconets exist in the same area, the access code can identify a piconet: all packets transmitted in the same piconet are preceded by the same channel access code and each piconet has its own distinctive access code. The access code structure is shown in Figure1.7.[11]

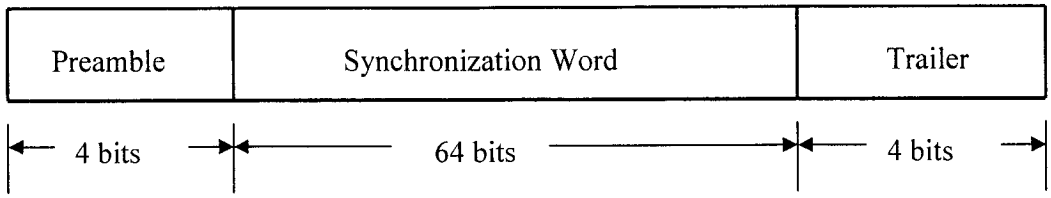


Figure 1.7 Access code structure

The access code consists of a preamble, a sync word, and possibly a trailer. The preamble is a fixed sequence of four bits either 0101 or 1010, depending on whether the first bit of the following sync word is 1 or 0. The preamble is used to detect the edges of the received data and facilitate DC compensation. If the packet header follows the access code, a trailer, the last part of the access code will be appended to the sync word. The trailer is a fixed four bits sequence similar to the preamble. It is either 1010 or 0101 depending on whether the last bit of the sync word is 0 or 1.

The sync word, which is unique for the piconet, is a 64-bit code word derived from a 24 bit Lower Address part (LAP) of the Bluetooth device address. It consists of a 34-bit BCH parity word, 24-bit LAP, and 6-bit Barker sequence. Bluetooth specification employs an algorithm to construct the sync word from the LAP, which guarantees large Hamming distance between sync words based on different addresses. In addition, it exhibits very high auto-correlation and very low cross-correlation properties which improves the timing synchronization process [2].

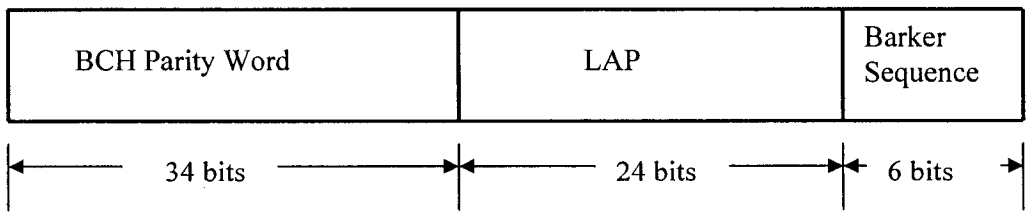


Figure 1.8 Synchronization word structure

In a piconet, since there are several operation modes for a master such as connection, paging, and inquiry, three different types of access codes are defined related to these modes [4]:

A. Channel Access Code (CAC) - The CAC is derived from the master's LAP and is used by all devices in that piconet during the exchange of data over a live connection.

B. Device Access Code (DAC)- The DAC is used for special signaling procedures, e.g., paging and response to paging. It is derived from a specific device's LAP.

C. Inquiry Access Code (IAC)-There are two variations for the IAC, general inquiry access code (GIAC) and dedicated inquiry access code (DIAC). GIAC is used by all devices during the inquiry procedures. It is to discover which other Bluetooth units are in range. The GIAC is fixed as 0x9E8B33 since no prior knowledge of anyone's LAP will exist. DIAC is common for a dedicated group of Bluetooth units that share a common characteristic, e.g. printers or cellular handset. The DIAC can be used to discover only these dedicated Bluetooth units in range. Presently, it is reserved by the specification for implementing inquiry procedures for these dedicated devices in range. DIACs use LAPs in the range ox9E8B00-ox9E8B3F.

- **Header**

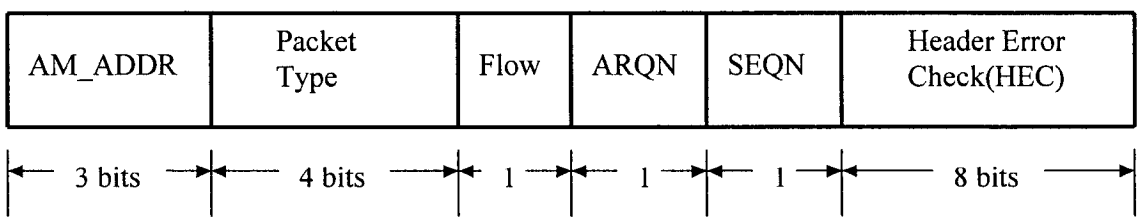


Figure 1.9 Packet Header

The packet header contains link control (LC) information. It is of length of 18 bits and includes six fields: AM_ADDR(active mobile address), Packet Type, Flow, ARQN, SEQN, and Header Error Check (HEC) [11].

- **Payload**

The actual user information and control information for the higher levels are carried on the payload. Two types of the payload are defined in Bluetooth, ACL payload and SCO payload.

A. ACL payload

ACL payload usually carries data message and has three segments: a payload header, a payload body, and a Cyclic Redundancy Check (CRC) code (only the AUX1 packet does not carry a CRC code) [4].

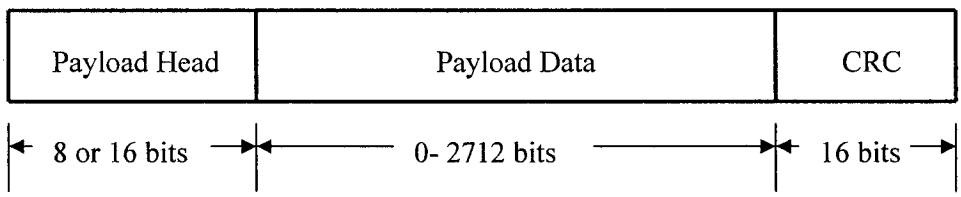


Figure 1.10 ACL payload structure

The payload header specifies logical link control information: L_CH, Flow, and Length [4].

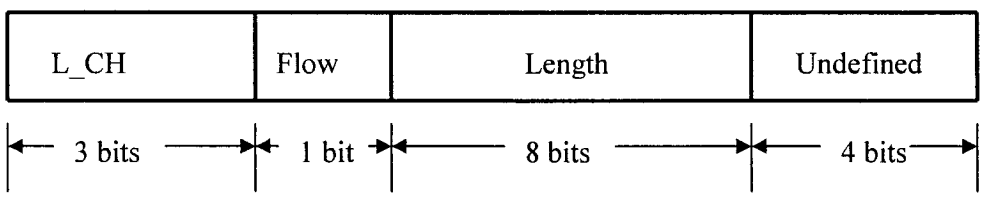


Figure 1.11 ACL payload header structure

The L_CH field value indicates the data message of this payload is the first fragment of a message or continuing fragments. L2CAP messages are segmented into several fragments to fit for the length of the baseband packets. The flow flag controls the flow at the L2CAP level. The length field indicates the number of bytes (i.e. 8-bit words) in the payload excluding the payload header and the CRC code.

The cyclic redundancy check code (CRC) is length of 16 bits. It is generated by the CRC-CCITT polynomial 210041 (octal representation).

B. SCO payload

The SCO payload has a fixed length. For the High-Quality Voice (HV) packets, the size of the payload is 240 bits; for the Data-Voice (DV) packet, the size of the payload is 80 bits. The payload header and CRC are absent [11].

Chapter 2

Small-Scale Fading over Indoor Wireless Radio Propagation Channel

Bluetooth information travels through an open-air channel from transmitter to receiver. It raises a special propagation problem due to the highly reflective and shadowing environment. When a radio wave strikes an object existing in this open environment, reflection, scattering, and diffraction will result. In this case, the radio wave from one transmitter may travel several different paths to arrive at the receiver in Figure 2.1. [2]. The receiver will collect a sum of waves with different amplitude, phase, and time delay for one incident signal. As a result, the multipath waves, combined at the receiver, will cause the received signal to vary widely in amplitude and phase. This is called multipath fading or distortion.

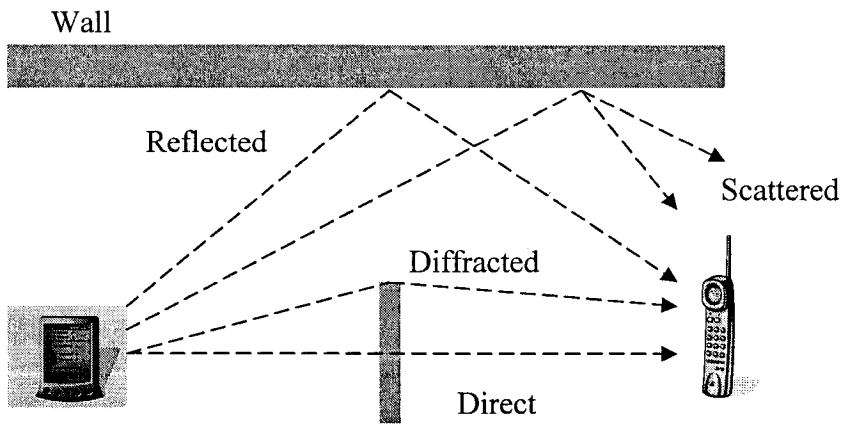


Figure 2.1 Radio wave propagation mechanisms

Multipath fading exists in almost all wireless communication indoor or outdoor. It depends on many factors such as the structure of surrounding, the motion of the mobile or surrounding objects. In the following chapter, we will primarily discuss two kinds of multipath fading: Rayleigth fading and Rice fading.

2.1 Rayleigh Fading

In Figure 2.1, the direct path may be line-of-sight (LOS) if there are no obstacles between the transmitter and receiver or obstructed LOS (OLOS). In the case of OLOS, the fluctuating amplitude of the received signal complies with a stochastic Rayleigh distribution [25]. In general, when the transmitted information experience a wireless radio multipath channel, the one path component of an impulse response over this channel can be expressed as [6] [14].

$$h_l(t) = \alpha_l \exp(j\theta_l)\delta(t-\tau_l) \qquad (2.1)$$

here l is the path index

α_l is the l^{th} path gain

θ_l is the phase shift

τ_l is the time delay of the l^{th} path

If the transmitted signal is given by

$$s(t) = u(t)\exp(j\omega_c t + \varphi) \qquad (2.2)$$

Where u (t) is the pulse signal in baseband

ω_c is the carrier frequency

φ is the carrier phase

The received signal through the multipath channel will be

$$r(t) = h(t) * s(t)$$

$$= \sum_{l=0}^{P-1} \alpha_l u(t - \tau_l) \exp(j(\omega_c (t - \tau_l) + \varphi + \theta_l)) \qquad (2.3)$$

Note that the parameters $\alpha_l, \theta_l, \tau_l$ are fluctuant over multipath channel. Rayleigh multipath fading implies that the amplitude fluctuations coefficient α_l are characterized by Rayleigh stochastic distribution [8].

2.1.1 Rayleigh Probability Density Function

Let us consider two zero-mean statistically independent normally distributed random variables n_1 and n_2, each having a variance σ_0^2, i.e., $n_1, n_2 \sim N(0, \sigma_0^2)$. Furthermore, let us derive a new random variable from n_1 and n_2 according to $\alpha = \sqrt{n_1^2 + n_2^2}$. Then, α represents a Rayleigh distributed random variable. The probability density function $p_\alpha(x)$ of Rayleigh distribution random variables α is given by [16]

$$p_\alpha(x) = \begin{cases} \frac{x}{\sigma_0^2} e^{-\frac{x^2}{2\sigma_0^2}}, & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad (2.4)$$

Rayleigh distributed random variables α have the expected value

$$E[\alpha] = \sigma_0 \sqrt{\frac{\pi}{2}} \qquad (2.5)$$

And the variance

$$\mathrm{Var}[\alpha] = \sigma_0^2 (2 - \frac{\pi}{2}) \qquad (2.6)$$

The second moment of Rayleigh distributed random variables α is

$$E[\alpha^2] = E[\alpha]^2 + \mathrm{Var}[\alpha] = 2\sigma_0^2 \qquad (2.7)$$

2. 2 Rice Fading

When multipath propagation model includes a LOS path, that is no obstacles existing between the transmitter and the receiver, the received signal amplitude varies with a Rice stochastic distribution. In this scenario, the coefficient α_l of the equation (2.3) should present a Rice stochastic distribution.

2. 2.1 Rice Probability Density Function

Similarity to Rayleigh denotation, let $n_1, n_2 \sim N(0, \sigma_0^2)$ and $\rho \in$ set of real numbers. Then, the random variable $\beta = \sqrt{(n_1 + \rho)^2 + n_2^2}$ is defined as Rice distributed random variable. The probability density function $p_\beta(x)$ of Rice distributed random variable β [16] is

$$p_\beta(x) = \begin{cases} \frac{x}{\sigma_0^2} e^{-\frac{x^2+\rho^2}{2\sigma_0^2}} I_0(\frac{x\rho}{\sigma_0^2}), & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad (2.8)$$

where $I_0(\cdot)$ denotes the modified Bessel function of 0^{th} order. For $\rho = 0$, the Rice distribution $p_\beta(x)$ results in the Rayleigh distribution $p_\alpha(x)$ described above. The first and the second moment of Rice distributed random variables β are [16]

$$E[\beta] = \sigma_0 \sqrt{\frac{\pi}{2}}\, e^{-\frac{\rho^2}{4\sigma_0^2}} \{ (1+\frac{\rho^2}{2\sigma_0^2}) I_0(\frac{\rho^2}{4\sigma_0^2}) + \frac{\rho^2}{2\sigma_0^2} I_1(\frac{\rho^2}{4\sigma_0^2}) \} \qquad (2.9)$$

and

$$E[\beta^2] = 2\,\sigma_0^2 + \rho^2 \qquad (2.10)$$

2.3 Parameters of Small-Scale Fading

Some parameters associated with the multipath channels such as delay spread, coherence bandwidth, Doppler spread, and coherence time are to be described next.

- **Delay spread**

Channel's delay spread is one of significant concerns that characterize the wireless communication performance affected by multipath channel. Because of multipath propagation, each component arrives at receiver from different path having its own time delay showed in Figure 2.2. [6]. This constructs a delay-spread picture for a multipath channel. Each multipath channel has its intrinsic delay spread property. Generally

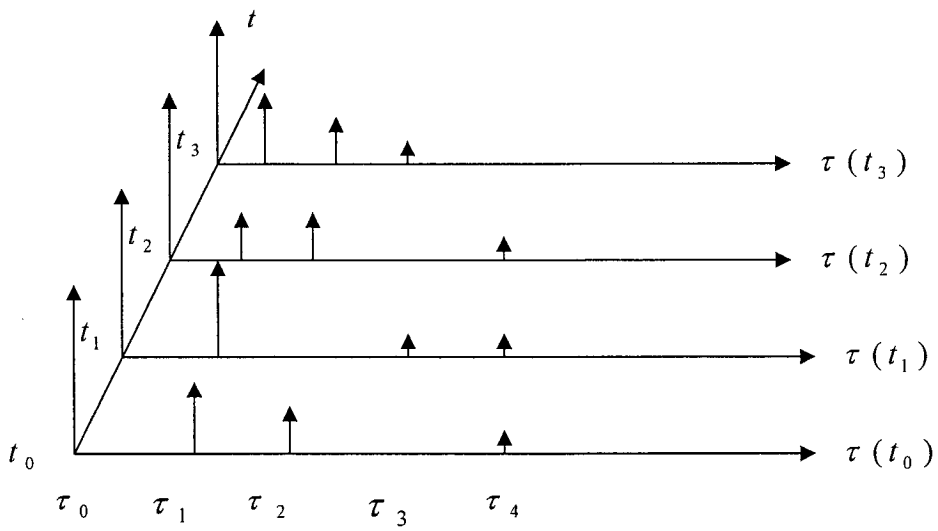


Figure 2.2 The time varying discrete-time impulse response model for a multipath radio channel

speaking, the channel time dispersive properties can be quantified in one of two ways: mean excess delay ($\bar{\tau}$) and root mean square (rms) delay spread (σ_τ). The mean excess delay is defined by the first moment of the power delay profile [6].

$$\bar{\tau} = \frac{\sum_k a_k^2 \tau_k}{\sum_k a_k^2} = \frac{\sum_k P(\tau_k)\tau_k}{\sum_k P(\tau_k)} \tag{2.11}$$

a_k^2 denotes kth path component amplitude, τ_k denotes the kth path component time delay. $P(\tau_k)$ denotes kth path component power.

The rms delay spread is the square root of the second central moment of the power delay profile.

$$\sigma_\tau = \sqrt{\overline{\tau^2} - (\bar{\tau})^2} \tag{2.12}$$

where

$$\overline{\tau^2} = \frac{\sum_k a_k^2 \tau_k^2}{\sum_k a_k^2} = \frac{\sum_k P(\tau_k)\tau_k^2}{\sum_k P(\tau_k)} \qquad (2.13)$$

- **Coherence bandwidth**

While the delay spread is a natural phenomenon caused by reflected and scattered propagation paths in the radio channel, the coherence bandwidth, B_c, is a defined relation derived from the rms delay spread. Coherent bandwidth is a measure of the range of frequencies over which the channel the channel response is *flat*, meaning that all spectral components will be passed with approximately equal gain and linear phase. If the signal spectral bandwidth is within B_c, the distortion over the channel that results in ISI is negligible. Coherent bandwidth can be expressed as follows if the frequency correlation function is above 0.9[6]

$$B_c \approx \frac{1}{50\sigma_\tau} \qquad (2.14)$$

If the definition is relaxed so that the frequency correlation function is above 0.5, then the coherent bandwidth is approximately [6]

$$B_c \approx \frac{1}{5\sigma_\tau} \qquad (2.15)$$

- **Doppler spread**

Doppler spread B_D characterizes frequency shifts among the various signal components due to the relative motion between transmitters and receivers or of objects around surrounding. It is defined as the range of frequencies over which the received

Doppler spectrum is essentially non-zero. Firstly, we define a Doppler shift f_d for a signal with carrier f_c is

$$f_d = \frac{v}{\lambda} \tag{2.16}$$

v is velocity of the receiver away from the transmitter, and λ is the carrier wavelength.

Doppler spread B_D is obtained by the maximum Doppler shift of f_d.

- **Coherence time**

The coherence time is related to the reciprocal of the maximum Doppler shift. It characterizes the time varying nature of the frequency dispersive channel in the time domain. Coherent time is the time duration over which two received signals have a strong potential for amplitude correlation. In general, we consider the channel characteristics within coherent time as be time invariant. It is derived as [6]

$$T_c \approx \frac{1}{f_m} \tag{2.17}$$

f_m is the maximum Doppler shift .

In practice, a more acceptable formula is [6]

$$T_c = \sqrt{\frac{9}{16\pi f_m^2}} = \frac{0.423}{f_m} \tag{2.18}$$

Delay spread and coherent bandwidth are parameters which describe the time dispersive nature of the channel in a local area without any information of the time varying nature of channel caused by the motion of units or objects of surrounding. On the other hand, the Doppler spread and time coherent are parameters that describe the time

varying natural of the channel in a small-scale region. Actual channels have a blend of both effects as will follow.

2.3.1 Flat Fading and Frequency Selective Fading

The bandwidth of the transmitted signal is less than the channel coherence bandwidth. Equivalently, the symbol period is much greater than the rms delay spread. In this case, the received signal will undergo *flat fading* [6] [13], meaning that all spectral components have a constant gain and linear phase shift. Assume that the B_s is the signal bandwidth and T_s is the symbol period, then a signal experiences flat fading if

$$B_s << B_c \tag{2.19}$$

and

$$T_s >> \sigma_\tau \tag{2.20}$$

If the bandwidth of the transmitted signal is greater than the channel coherence bandwidth, the received signal will experience *frequency selective fading* [6] [10]. In the case of frequency fading, certain frequency components in the received signal spectrum have greater gain than others, and then the transmitted symbols may be significantly distorted by multipath-induced ISI.

A signal experiences frequency selective fading if

$$B_s > B_c \tag{2.21}$$

and

$$T_s < \sigma_\tau \tag{2.22}$$

2.3.2 Fast Fading and Slow Fading

Flat fading and frequency selective fading are not associated with channel time variation. The amplitude may change in time according to the Rayleigh distribution. This chapter will present the time varying properties caused by Doppler spread.

When the channel impulse response varies faster than the baseband signal, that is the coherence time is less than the symbol period of the transmitted signal. We refer to this kind of fading as *fast fading* [6]. Fast fading satisfies

$$T_s > T_c \tag{2.23}$$

and

$$B_s < B_D \tag{2.24}$$

If the channel impulse response is slower than the baseband signal, the coherence time is greater than the symbol period of the transmitted signal. Under this condition, the channel experiences *slow fading* [6]. The channel may be assumed to be static over the coherence time period.

A signal undergoes slow fading if

$$T_s << T_c \tag{2.25}$$

and

$$B_s >> B_D \tag{2.26}$$

This should be noted that when a channel is specified as a fast or slow fading channel, this does not specify whether the channel is flat fading or frequency selective fading.

Chapter 3

Derivation of Blind and Classic Non-Blind Channel Estimation Algorithms

In wireless communications, the received information may be distorted through multipath channels, which was discussed in Chapter 2. To compensate for this kind of channel impairments, channel estimation should be one of the efficient solutions [22] [29]. In the thesis, two kinds of channel estimations techniques are presented and compared, i.e. non-blind channel estimation technique and blind channel estimation technique. Non-blind approaches require training sequences for channel estimations, which is known to both the transmitter and receiver. Blind channel estimation technique only utilizes the underlying characteristics of the signal structure to estimate channels without training sequences. For our Bluetooth application case, the algorithms to these techniques will be derived next.

3.1 Classic Non-Blind Channel Estimation Algorithms

In the thesis, three classical estimation techniques are involved in non-blind channel estimations, i.e. Least Square (LS), Minimum Mean-Square Error (MMSE), and Least Mean Square (LMS).

3.1.1 Derivation of Least Square (LS) Algorithm

Consider Least Squares channel estimation block in Figure 3.1

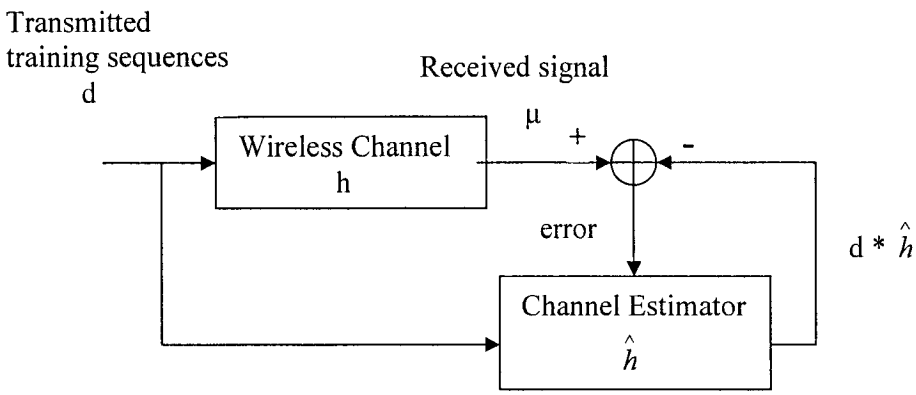


Figure 3.1 Least Squares channel estimation block

Over a finite number of time samples, LS algorithm is to minimize the error in Figure 3.1 between the estimated output and a received signal. Therefore, the result of estimated channel $\hat{h}$ should best describe the actual channel h. In Least Squares, a set of P snap shots of the data vector, d, are collected. We define LS cost function as, [26] [12]

$$J(\hat{h}) = \sum_{m=0}^{P-1} |\mu_m - \sum_{j=0}^{N-1} \hat{h}_j d_{m-j}|^2 \qquad (3.1)$$

m is time index

N is the number of taps of the actual channel h, $h = [h_0, h_1 \ldots h_{N-1}]$

P is the length of the training sequences (or preamble) d, $d=\{d_i\}$ for i= 0,1,....P-1.

μ_m is the received signal on time m, $\mu_m = \sum_{j=0}^{N-1} h_j d_{m-j}$.

d_{m-j} is the training sequence on time m-j.

$\hat{h}_j$ is the estimated channel transfer function on j^{th} tap.

h_j is the actual channel transfer function on j^{th} tap.

The gradient operator is applied to (3.1)

$$\nabla J(\hat{h}_l) = \frac{\partial J(\hat{h})}{\partial \hat{h}_l} \qquad (l=0, 1, \ldots N-1)$$

$$= \frac{\partial \sum_{m=0}^{P-1} |\mu_m - \sum_{j=0}^{N-1} \hat{h}_j d_{m-j}|^2}{\partial \hat{h}_l}$$

$$= \frac{\partial}{\partial h_l} [\sum_{m=0}^{P-1} (\mu_m \mu_m^* - \mu_m \sum_{j=0}^{N-1} \hat{h}_j^* d_{m-j}^* - \sum_{j=0}^{N-1} \hat{h}_j d_{m-j} \mu_m^* + \sum_{j=0}^{N-1} \hat{h}_j d_{m-j} \sum_{j=0}^{N-1} \hat{h}_j^* d_{m-j}^*)] \qquad (3.2)$$

in (3.2), μ_m^*, d_{m-j}^*, $\hat{h}_j^*$ are the conjugate of μ_m, d_{m-j}, $\hat{h}_j$. * is the operator of the conjugate.

Assuming that $\hat{h}_l = \alpha_l + i\beta_l$, $\hat{h}_j = \alpha_j + i\,\beta_j$ then substituting in (3.2). One finds the gradients of four parts in (3.2) respectively

$$\frac{\partial}{\partial h_l} [\sum_{m=0}^{P-1} (\mu_m \mu_m^*)] = 0 \qquad (3.3)$$

$$\frac{\partial}{\partial h_l} [\sum_{m=0}^{P-1} (\mu_m \sum_{j=0}^{N-1} \bar{h}_j^* d_{m-j}^*)]$$

$$= \sum_{m=0}^{P-1} [\frac{\partial}{\partial \alpha_l} (\mu_m \sum_{j=0}^{N-1} (\alpha_j - i\beta_j) d_{m-j}^*) + i \frac{\partial}{\partial \beta_l} (\mu_m \sum_{j=0}^{N-1} (\alpha_j - i\beta_j) d_{m-j}^*)]$$

$$= 2 \sum_{m=0}^{P-1} \mu_m d_{m-l}^* \qquad (3.4)$$

$$\frac{\partial}{\partial h_l}[\sum_{m=0}^{P-1}(\sum_{j=0}^{N-1}\hat{h}_j d_{m-j}\mu_m^*)]$$

$$=\sum_{m=0}^{P-1}[\frac{\partial}{\partial \alpha_l}(\mu_m^*\sum_{j=0}^{N-1}(\alpha_j+i\beta_j)d_{m-j})+i\frac{\partial}{\partial \beta_l}(\mu_m^*\sum_{j=0}^{N-1}(\alpha_j+i\beta_j)d_{m-j})]=0 \tag{3.5}$$

$$\frac{\partial}{\partial h_l}[\sum_{m=0}^{P-1}(\sum_{j=0}^{N-1}\hat{h}_j d_{m-j}\sum_{j=0}^{N-1}\hat{h}_j^* d_{m-j}^*)]$$

$$=\sum_{m=0}^{P-1}\frac{\partial}{\partial h_l}(\sum_{j=0}^{N-1}\sum_{i=0}^{N-1}\hat{h}_j d_{m-j}\hat{h}_j^* d_{m-i}^*)$$

$$=\sum_{m=0}^{P-1}[\frac{\partial}{\partial \alpha_l}(\sum_{j=0}^{N-1}\sum_{i=0}^{N-1}(\alpha_j+i\beta_j)(\alpha_i-i\beta_i)d_{m-j}d_{m-i}^*)$$

$$+i\frac{\partial}{\partial \beta_l}(\sum_{j=0}^{N-1}\sum_{i=0}^{N-1}(\alpha_j+i\beta_j)(\alpha_i-i\beta_i)d_{m-j}d_{m-i}^*)]$$

$$=2\sum_{m=0}^{P-1}\sum_{j=0}^{N-1}\hat{h}_j d_{m-j}d_{m-l}^* \tag{3.6}$$

Substituting (3.3), (3.4), (3.5), and (3.6) into (3.2), we obtain the gradient of the cost function for $\hat{h}_l$ as follows

$$\nabla J(\hat{h}_l)=-2\sum_{m=0}^{P-1}\mu_m d_{m-l}^*+2\sum_{m=0}^{P-1}\sum_{j=0}^{N-1}\hat{h}_j d_{m-j}d_{m-l}^*$$

$$=2\sum_{m=0}^{P-1}(d_{m-l}^*(\sum_{j=0}^{N-1}\hat{h}_j d_{m-j}-\mu_m)) \tag{3.7}$$

To minimize the cost function (3.1), the following gradient iterative approach applies

$$\hat{h}_{l,k+1}=\hat{h}_{l,k}-\frac{1}{2}\Delta\nabla J(\hat{h}_{l,k}) \tag{3.8}$$

Where, Δ is the convergence factor. The new estimated value $\hat{h}_{l,k+1}$ updates the old one. Substituting (3.7) into (3.8)

$$\hat{h}_{l,k+1} = \hat{h}_{l,k} - \frac{1}{2}\Delta 2 \sum_{m=0}^{P-1} (d^*_{m-l} (\sum_{j=0}^{N-1} \hat{h}_{l,k}\, d_{m-j} - \mu_m))$$

$$= \hat{h}_{l,k} - \Delta \sum_{m=0}^{P-1} (d^*_{m-l} (\sum_{j=0}^{N-1} \hat{h}_{l,k}\, d_{m-j} - \mu_m)) \qquad (3.9)$$

3.1.2 Derivation of Minimum Mean-Square Error (MMSE) Algorithm

An alternative method to minimizing the error in Figure 3.1 is to use ensemble averages. This leads to the Minimum Mean-Square Error solutions. Instead of a finite number of time samples, MMSE algorithm is to get optimal $\hat{h}$ based on minimal error over the ensemble of possible realizations of the stationary environment, [30] [27].

MMSE cost function is defined by the expected value of the square error between the estimated output and the desired version of that signal at time m.

$$J(\hat{h}) = E[\,|\mu_m - \sum_{j=0}^{N-1} \hat{h}_j\, d_{m-j}|^2\,] \qquad (3.10)$$

m : time index

N: the number of taps of the actual channel h, $h = [h_0, h_1 \ldots h_{N-1}]$

d_{m-j} : the transmitted signal on time m-j.

μ_m : the received signal on time m. $\mu_m = \sum_{j=0}^{N-1} h_j\, d_{m-j}$.

Also (3.10) can be written as

$$J(\hat{h}) = E[(\mu_m - \sum_{j=0}^{N-1} \hat{h}_j\, d_{m-j})(\mu^*_m - \sum_{j=0}^{N-1} \hat{h}^*_j\, d^*_{m-j})] \qquad (3.11)$$

Let $\hat{H} = \begin{bmatrix} \hat{h}_0 \\ \hat{h}_1 \\ \cdots \\ \hat{h}_{N-1} \end{bmatrix}$, $\quad D_m = \begin{bmatrix} d_m \\ d_{m-1} \\ \cdots \\ d_{m-N+1} \end{bmatrix}$, then (3.11) is

$$J(\hat{h}) = E[(\mu_m^* - D_m^H \hat{H}^*)(\mu_m - D_m^T \hat{H})]$$

$$= E[(\mu_m^* - \hat{H}^H D_m^*)(\mu_m - D_m^T \hat{H})]$$

$$= E[\mu_m \mu_m^*] - E[\hat{H}^H D_m^* \mu_m] - E[\mu_m^* D_m^T \hat{H}] - E[\hat{H}^H D_m^* D_m^T \hat{H}] \quad (3.12)$$

μ_m is scalar, and μ_m^* is the conjugate of μ_m.

$\hat{H}^*$: the conjugate of vector $\hat{H}$. $\hat{H}$ is a vector.

D_m^H: the conjugate transpose of vector D_m, H is the conjugate transpose operator for the vector.

D_m^T: the transpose of vector D_m, T is the transpose operator for the vector.

If $\hat{h}_j = \alpha_j + i\beta_j$, the gradient function of (3.12) becomes

$$\nabla J(\hat{H}) = \nabla(E[\mu_m \mu_m^*] - E[\hat{H}^H D_m^* \mu_m] - E[\mu_m^* D_m^T \hat{H}] - E[\hat{H}^H D_m^* D_m^T \hat{H}])$$

$$= \nabla E[\mu_m \mu_m^*] - \nabla E[\hat{H}^H D_m^* \mu_m] - \nabla E[\mu_m^* D_m^T \hat{H}] - \nabla E[\hat{H}^H D_m^* D_m^T \hat{H}] \quad (3.13)$$

The four components of (3.13) are calculated respectively below

$$\nabla \mathrm{E}\,[\mu_m \mu_m^*] = \frac{\partial}{\partial \hat{H}}(E[\mu_m \mu_m^*]) = 0 \tag{3.14}$$

$$\nabla \mathrm{E}\,[\hat{H}^H \mathrm{D}_m^* \mu_m] = \nabla \mathrm{E}\,[\mu_m \hat{H}^H \mathrm{D}_m^*]$$

$$= \begin{bmatrix} \frac{\partial}{\partial \alpha_1} E[\mu_m[(\alpha_1 - i\beta_1)(\alpha_2 - i\beta_2)...(\alpha_N - i\beta_N)\mathrm{D}_m^*]] \\ \frac{\partial}{\partial \alpha_2}(E[\mu_m[(\alpha_1 - i\beta_1)(\alpha_2 - i\beta_2)...(\alpha_N - i\beta_N)D_m^*]] \\ ..... \\ \frac{\partial}{\partial \alpha_N}(E[\mu_m[(\alpha_1 - i\beta_1)(\alpha_2 - i\beta_2)...(\alpha_N - i\beta_N)D_m^*]] \end{bmatrix}$$

$$+\mathrm{i}\begin{bmatrix} \frac{\partial}{\partial \beta_1} E[\mu_m[(\alpha_1 - i\beta_1)(\alpha_2 - i\beta_2)...(\alpha_N - i\beta_N)\mathrm{D}_m^*]] \\ \frac{\partial}{\partial \beta_2} E[\mu_m[(\alpha_1 - i\beta_1)(\alpha_2 - i\beta_2)...(\alpha_N - i\beta_N)\mathrm{D}_m^*]] \\ ..... \\ \frac{\partial}{\partial \beta_N} E[\mu_m[(\alpha_1 - i\beta_1)(\alpha_2 - i\beta_2)...(\alpha_N - i\beta_N)\mathrm{D}_m^*]] \end{bmatrix}$$

$$= 2\,\mathrm{E}\,[\mu_m \mathrm{D}_m^*] \tag{3.15}$$

$$\nabla \mathrm{E}\,[\mu_m^* \mathrm{D}_m^T \hat{H}] = \nabla \mathrm{E}\,[\hat{H}^T \mathrm{D}_m \mu_m^*]$$

$$= \begin{bmatrix} \frac{\partial}{\partial \alpha_1} E[[(\alpha_1 + i\beta_1)(\alpha_2 + i\beta_2)...(\alpha_N + i\beta_N)]D_m \mu_m^*] \\ \frac{\partial}{\partial \alpha_2} E[[(\alpha_1 + i\beta_1)(\alpha_2 + i\beta_2)...(\alpha_N + i\beta_N)]D_m \mu_m^*] \\ ..... \\ \frac{\partial}{\partial \alpha_N} E[[(\alpha_1 + i\beta_1)(\alpha_2 + i\beta_2)...(\alpha_N + i\beta_N)]D_m \mu_m^*] \end{bmatrix}$$

$$+\mathrm{i}\begin{bmatrix} \frac{\partial}{\partial\beta_1}E[[(\alpha_1+i\beta_1)(\alpha_2+i\beta_2)...(\alpha_N+i\beta_N)]D_m\mu_m^*] \\ \frac{\partial}{\partial\beta_2}E[[(\alpha_1+i\beta_1)(\alpha_2+i\beta_2)...(\alpha_N+i\beta_N)]D_m\mu_m^*] \\ ..... \\ \frac{\partial}{\partial\beta_N}E[[(\alpha_1+i\beta_1)(\alpha_2+i\beta_2)...(\alpha_N+i\beta_N)]D_m\mu_m^*] \end{bmatrix}$$

$$=0 \qquad (3.16)$$

$$\nabla \mathrm{E}[\hat{H}^H \mathrm{D}_m^* \mathrm{D}_m^T \hat{H}] = \begin{bmatrix} \frac{\partial}{\partial\alpha_1}E[[(\alpha_1-i\beta_1)(\alpha_2-i\beta_2)...(\alpha_N-i\beta_N)]D_m^*D_m^T\begin{bmatrix}(\alpha_1+i\beta_1)\\(\alpha_2+i\beta_2)\\.....\\(\alpha_N+i\beta_N)\end{bmatrix}] \\ \frac{\partial}{\partial\alpha_2}E[[(\alpha_1-i\beta_1)(\alpha_2-i\beta_2)...(\alpha_N-i\beta_N)]D_m^*D_m^T\begin{bmatrix}(\alpha_1+i\beta_1)\\(\alpha_2+i\beta_2)\\.....\\(\alpha_N+i\beta_N)\end{bmatrix}] \\ ..... \\ \frac{\partial}{\partial\alpha_N}E[[(\alpha_1-i\beta_1)(\alpha_2-i\beta_2)...(\alpha_N-i\beta_N)]D_m^*D_m^T\begin{bmatrix}(\alpha_1+i\beta_1)\\(\alpha_2+i\beta_2)\\.....\\(\alpha_N+i\beta_N)\end{bmatrix}] \end{bmatrix}$$

$$+\mathrm{i}\begin{bmatrix} \frac{\partial}{\partial\beta_1}E[[(\alpha_1-i\beta_1)(\alpha_2-i\beta_2)...(\alpha_N-i\beta_N)]D_m^*D_m^T\begin{bmatrix}(\alpha_1+i\beta_1)\\(\alpha_2+i\beta_2)\\.....\\(\alpha_N+i\beta_N)\end{bmatrix}] \\ \frac{\partial}{\partial\beta_2}E[[(\alpha_1-i\beta_1)(\alpha_2-i\beta_2)...(\alpha_N-i\beta_N)]D_m^*D_m^T\begin{bmatrix}(\alpha_1+i\beta_1)\\(\alpha_2+i\beta_2)\\.....\\(\alpha_N+i\beta_N)\end{bmatrix}] \\ ..... \\ \frac{\partial}{\partial\beta_N}E[[(\alpha_1-i\beta_1)(\alpha_2-i\beta_2)...(\alpha_N-i\beta_N)]D_m^*D_m^T\begin{bmatrix}(\alpha_1+i\beta_1)\\(\alpha_2+i\beta_2)\\.....\\(\alpha_N+i\beta_N)\end{bmatrix}] \end{bmatrix}$$

$$= 2\,E\,[D_m^* D_m^T \hat{H}\,] \tag{3.17}$$

Substituting (3.14), (3.15), (3.16) and (3.17) into (3.13)

$$\nabla J(\hat{H}) = 2\,E\,[D_m^* D_m^T \hat{H}\,] - 2\,E\,[\mu_m D_m^*] \tag{3.18}$$

To minimize the cost function, let $\nabla J(\hat{H}) = 0$, then we obtain the following

$$E\,[D_m^* D_m^T \hat{H}\,] = E\,[\mu_m D_m^*]$$

Finally, the solution for the estimated channel transfer function $\hat{H}$ is given by

$$\hat{H} = E\,[D_m^* D_m^T]^{-1}\,E\,[\mu_m D_m^*] \tag{3.19}$$

3.1.3 The Derivation of Least Mean Square(LMS) Algorithm

The Least Mean Square algorithm minimizes the error in Figure 3.1 based on ensemble average. However, the LMS algorithm adopts an iterative approach to minimizing the cost function (3.10) rather than directly solving equation (3.19). An iterative solution to (3.10) is defined instead as follows [28].

$$\hat{H}_{m,k+1} = \hat{H}_{m,k} - \frac{1}{2}\Delta\nabla J(\hat{H}_{m,k}) \tag{3.20}$$

Substituting (3.18) into (3.20), we obtain

$$\hat{H}_{m,k+1} = \hat{H}_{m,k} - \frac{1}{2}\Delta\,(2\,E\,[D_m^* D_m^T \hat{H}_{m,k}\,] - 2\,E\,[\mu_m D_m^*])$$

$$= \hat{H}_{m,k} - \Delta\,(E\,[D_m^* D_m^T \hat{H}_{m,k}\,] - E\,[\mu_m D_m^*]) \tag{3.21}$$

Dropping the expectation operator from (3.21), means assuming expected values are well approximately the instantaneous values, we obtain,

$$\hat{H}_{m,k+1} = \hat{H}_{m,k} - \Delta\,(D_m^* D_m^T \hat{H}_{m,k} - \mu_m D_m^*)$$

$$= \hat{H}_{m,k} - \Delta D_m^* (D_m^T \hat{H}_{m,k} - \mu_m) \quad (3.22)$$

(3.22) is applied iteratively till convergence.

3. 2 Blind Channel Estimation Algorithm

All algorithms obtained above are derived on the basis of realization of training sequences. In other words, they assume the pre-knowledge of some sequences by both transmitter and receiver so as to retrieve the channel characteristics. These kinds of algorithms are called as non-blind algorithm. As an alternative, blind algorithms do not need training sequences to estimate the channel. In one of the many available the techniques in [15] uses superimposed periodic pilot over transmitted signal for channel estimation as in Figure 3.2.a

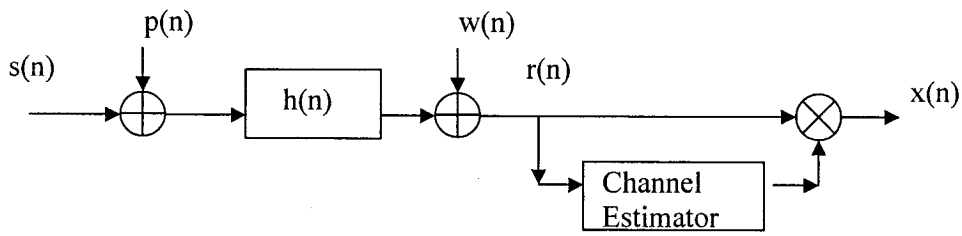


Figure 3.2.a Blind channel estimation using periodic pilots

In Figure 3.2

h (n): Actual channel transfer function with finite taps , n = 1, 2, 3,....L-1.

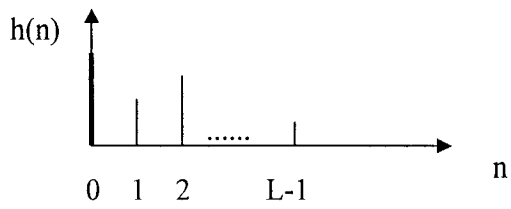


Figure 3.2.b. Assumed channel impulse response

s (n): transmitted signals. Assuming that algorithm works on N bits sequence, $0 \leq n \leq N-1$, where, $N \geq L$.

p (n): pilot signal. $p(n) = \sum_i \delta (n- iL)$

It is periodically added to the s (n). Its period is L, which is also the length of h (n).

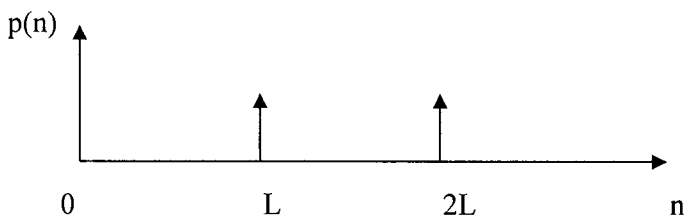


Figure 3.2.c Pilot signal

w (n): In the receiver, additive white Gaussian noise.

r (n): received signal contained pilot and channel distortion.

x (n): received signal without channel distortion

The received signal r(n) can be written as

$r(n) = [s(n) + p(n)] * h(n) + w(n)$

$= p (n) * h (n) + s (n) * h (n) + w (n)$

(3.23)

If p (n) is a periodic sequence with period L , p (n) * h (n) is also periodic with the same period. Define the *i*th length L sub-record of r(n) as follows

$$r_i(n) = r(iL+n) \qquad 0 \le i \le R-1, \quad 0 \le n \le L-1 \tag{3.24}$$

where $R=\frac{N}{L}$, it is the number of sub-record. Assuming it is integer.

The synchronized average of $r_i(n)$ over N bits r(n) is obtained by

$$\begin{aligned}\hat{m}_r(n) &= \frac{1}{R}\sum_{i=0}^{R-1} r_i(n) \\ &= \frac{1}{R}\sum_{i=0}^{R-1} [s(iL+n) * h(n) + p(iL+n) * h(n) + w(iL+n)] \\ &= \frac{1}{R}\sum_{i=0}^{R-1} [p(iL+n) * h(n)] + \frac{1}{R}\sum_{i=0}^{R-1} [s(iL+n) * h(n) + w(iL+n)]\end{aligned} \tag{3.25}$$

In (3.25), $0 \le n \le L-1$. For channel estimation purpose, s(n) and w(n) are considered as random. It is reasonable to assume their mean values are zeros.

$$E[s(iL+n) * h(n) + w(iL+n)] = \lim_{R\to\infty} \left(\frac{1}{R}\sum_{i=0}^{R-1} [s(iL+n) * h(n) + w(iL+n)]\right) = 0 \tag{3.26}$$

when working on long enough sequences, that is R is large enough, we use $\hat{m}_r(n)$ as an estimated values for $\frac{1}{R}\sum_{i=0}^{R-1} [p(iL+n) * h(n)]$

$$\hat{m}_r(n) = \frac{1}{R}\sum_{i=0}^{R-1} [p(iL+n) * h(n)] \qquad 0 \le n \le L-1$$

Since p (n) is a periodic sequence with period L

$$p(n) = p(L+n) = p(2L+n) = \ldots\ldots = p(iL+n) \qquad i = 0,1,2, \ldots\ldots R-1$$

Then $\frac{1}{R}\sum_{i=0}^{R-1} [p(iL+n)] = p(n)$ in the period of $0 \le n \le L-1$

$\hat{m}_r(n)$ can be rewritten as

$$\hat{m}_r(n) = p(n) * h(n) \qquad 0 \le n \le L-1 \qquad (3.27)$$

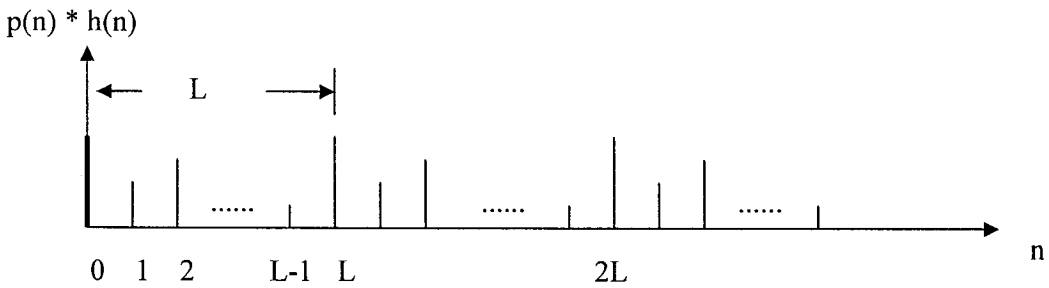
Previously, we define p (n) as impulse function in whole time axis

$$p(n) = \sum_{i=0}^{R-1} \delta(n - iL) \qquad (3.28)$$

Convoluting p(n) with h(n)

$$p(n) * h(n) = \sum_{i=0}^{R-1} h(n - iL) \qquad (3.29)$$

Since h(n) is a finite impulse response for n =0, 1, 2 ...L-1, whereas $\sum_{i=0}^{R-1} h(n - iL)$ represents a periodic function in whole time axis.



If only considered the first period, let i = 0, also suppose that $h(n) \neq 0$ for $0 \le n \le L-1$, we can obtain

$$p(n) * h(n) = h(n) \qquad 0 \le n \le L-1 \qquad (3.30)$$

Substituting (3.30) into (3.27) for the period of $0 \le n \le L-1$, we obtain

$$\hat{m}_r(n) = h(n) \qquad 0 \le n \le L-1 \qquad (3.31)$$

Substituting $\hat{m}_r(n) = \frac{1}{R} \sum_{i=0}^{R-1} r_i(n)$ into (3.31), the actual channel h can be estimated as

$$\hat{h}(n) = \frac{1}{R} \sum_{i=0}^{R-1} r_i(n) \qquad 0 \leq n \leq L\text{-}1 \qquad (3.32)$$

Chapter 4

Simulations of Blind and Non-Blind Channel Estimation Techniques

4.1. Description of Simulation System for Channel Estimation

4.1.1. The Radio Channel Model Setting

In the previous chapter, we have derived four algorithms for channel estimation. The simulations of these algorithms in this chapter will investigate their merits as they apply to Bluetooth application in different scenarios such as slow fading, fast fading, and different noise backgrounds. It is noted that since this simulation is running on Bluetooth application, Bluetooth access code is the best candidate as preamble for non-blind channel estimations.

Assuming the channel filter function that models all multipath fading is given by [7] [9]

$$h(t) = \sum_i a_i e^{-j\varphi_i} \delta(t-\tau_i) \qquad (4.1)$$

α_i represents the amplitude of *i*th multipath at time t. φ_i indicates the phase shift due to free propagation of the *i*th multipath component. τ_i is the time delays of kth multipath.

Consider the four paths case, $\tau_i = iT$, i=0, 1, 2, 3. T is the bit period. Under these assumptions, in the simulations, h (t) is specifically set to be

$$h(t) = (0.5+0.4i)\,\delta(t-1T) + (0.6+0.2i)\,\delta(t-2T) + (0.7+0.3i)\,\delta(t-3T) + (0.1+0.5i)\,\delta(t-4T) \qquad (4.2)$$

If one includes the channel time varying represented by Rayleigh or Rice distribution, the channel transfer function can be represented by

$$H(t) = \alpha(t) \cdot h(t) \qquad (4.3)$$

$\alpha(t)$ is Rayleigh or Rice random variable.

It is obvious that equation (4.3) represents a time varying wireless channel impulse response. However, if the devices or objects around the environment move slowly compared to the FH hop rate, H (t) can be seen as a time-invariant during one hop interval (one time slot) and denoted by

$$H(t) = \alpha \cdot h(t) \qquad (4.3)$$

In the simulations, we simulate these two cases described as slow fading case for time-invariant and fast fading case for time varying channels. Note that the definition of slow fading and fast fading in the simulations is a bit different from that of chapter 3. In chapter 3, we define the slow fading in the sense that the channel does not change within a symbol period, and here slow fading channel indicates that the channel does not change within one hop.

Also, under the consideration of channel models in the simulations, whether $\alpha(t)$ changes fast or slow, the multipath channel h(t) is assumed to vary very slowly such that we can represent it by one snap shot as (4.2) and estimate this value in face of slow or fast $\alpha(t)$.

In the case of Bluetooth FHS system, channels are different on different hops due to the different hopping frequencies [8] [10]; therefore, simulations are implemented on

each hop. The results of the simulations are described by the discrepancy between H and $\hat{H}$ relatively. H denotes the actual desired channel, and $\hat{H}$ expresses estimated channel. The channel estimation error is defined by:

$$pErr = (\|H - \hat{H}\|^2 / \|H\|^2) \cdot 100 \qquad (4.5)$$

This serves as a measure of the quality of the channel estimation techniques in the thesis.

4.1.2. Non-Blind Approach and Blind Approach of Channel Estimation

- **Non-blind approach**

Non-blind approaches need some training sequence for channel estimations. In other words, during the estimation procedure, the receiver knows the training sequence it will receive prior to actual information transmission. From the earlier algorithms' derivation, LMS, LS, and MMSE should group under non-blind approaches. According to the definition of the Bluetooth access code in Bluetooth specifications mentioned in chapter 1, the access code is fixed and known by the devices. Hence, we can utilize it as a training sequence or preamble. Therefore, in the simulations, we are using 64 bits Bluetooth access code as a preamble sequence. The loss of bandwidth is 9.6% for one slot packets, 2.6% for five slot packets.

- **Blind approach**

Blind approach does not require training sequence, and then it can function

anywhere including during information transmission. In the simulations, unlike non-blind approach, we do not set the restriction to the sequence length. Here, a blind channel estimation algorithm is introduced by superimposing pilots on the signals to implement channel estimation [15]. The advantage of this technique is saving precious bandwidth. The pilots just add on top of the signals instead of replacing them. In the simulations, the pilot is an impulse function, and it adds to signals every four bits.

4.1.3. System Flow Charts and Parameters Explanations

In the thesis, non-blind channel estimation algorithms such as LMS, LS, MMSE, and blind channel estimation algorithm are respectively simulated for purposes of estimating channels. Bluetooth access code is utilized as a preamble for non-blind channel estimation techniques. The blind channel estimation technique works on data without any pre-knowledge in the simulations.

The LMS, LS channel estimation algorithms in the simulations adopt gradient techniques instead of solving the matrix equation directly. The optimal estimated channel is achieved by many iterative channel updates. The LMS algorithm uses the instantaneous estimator. In other words, the channel updates are computed for each input sample, and the channel modified after each sample [21]. In the simulations, we modified the procedure to increase the iterative channel updates for each sample. For each sample, we iteratively update the estimated channel with new one till it reaches the sub-optimum. In the simulations, we set the parameter *iterations* for this step in the flow chart of LMS channel estimation. Then we move to next sample to continue the

LMS channel updates procedure. The LS iterative algorithm is to collect all preamble bits, and then minimize the summation of the errors [32]. In the simulations, the LS iterative algorithm uses the gradient technique to obtain the optimal estimated channel. In other words, each time the estimator computes the gradient estimate; it uses the previous estimated channel. So, we set the parameter *iteration* for this step in the flow chart of LS channel estimation.

Presently, we will introduce the implementations of all such simulations via their flow charts.

- Flow chart of the LMS channel estimation algorithm

It is shown in Figure 4.1 and 4.2. First of all, we should set the channel model which will be estimated. In the case of slow fading, the channel model will include two parts: Rayleigh fading and multipath distortion. In the case of fast fading, only multipath distortion will be estimated. So, α is not included in the channel model H. The explanation is given in 4.2.2. Since the outcome of the simulation is the probability how many chances the channel estimation errors are less than 5%, we should run the simulation many times. In the simulations, we set parameter *simulations* for this purpose in the flow charts of all four algorithms. The value of the *simulations* is set to 10000 times so as to output statistic value. The actual desired received signals are obtained by convolving the transmitted preambles with the actual channel transfer function h, and the estimated received signals are obtained by convolving the transmitted preambles with the estimated channel transfer function $\hat{h}$. When multiplying the estimated received signals by Rayleigh or Rice fading for the fast fading

case, α (t) is varying with channels. The LMS algorithm estimates the channel sample by sample instead of collecting all preambles. The channel is modified for each new sample. So, there is a decision to find the end of the preamble in the flow chart. In the simulations, the parameter *iterations* we mentioned before is set to 100 times. The final outcome in the simulations is the statistic of the channel estimation error less than 5%.

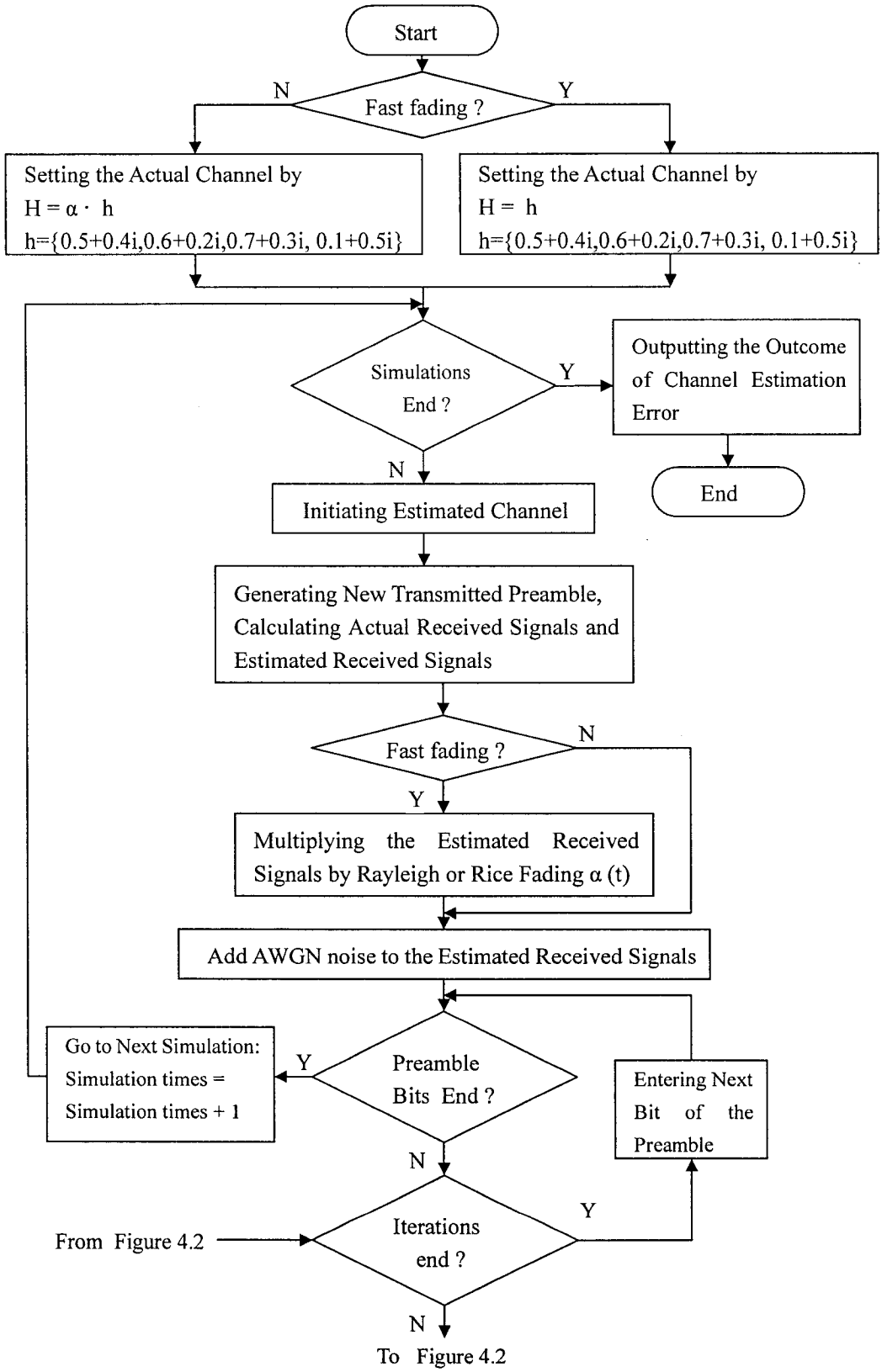


Figure 4.1 Flow Chart of the LMS Channel Estimation Technique

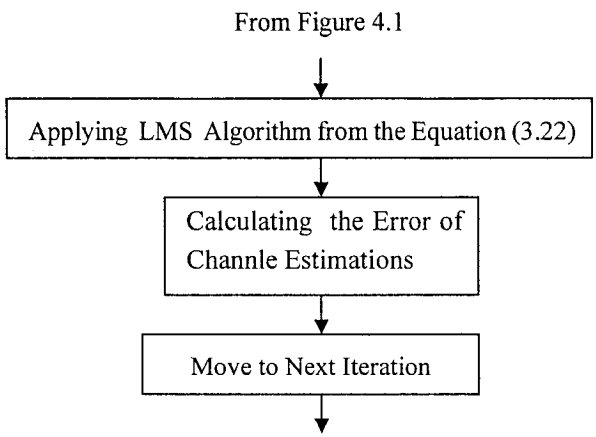


Figure 4.2 Continuation of the Flow Chart of the LMS Channel Estimation Technique

- Flow chart of the LS channel estimation algorithm

Figures 4.3 and 4.4 flow charts describe the implementation procedure for the simulation of LS channel estimation algorithm. In the beginning of the flow, some steps are same to the LMS channel estimation algorithm. However, the LS algorithm processes its channel estimation only after collecting all preambles. There is no decision to find the end of the preamble in the flow. The values of simulation times and iterative times are same to those of LMS.

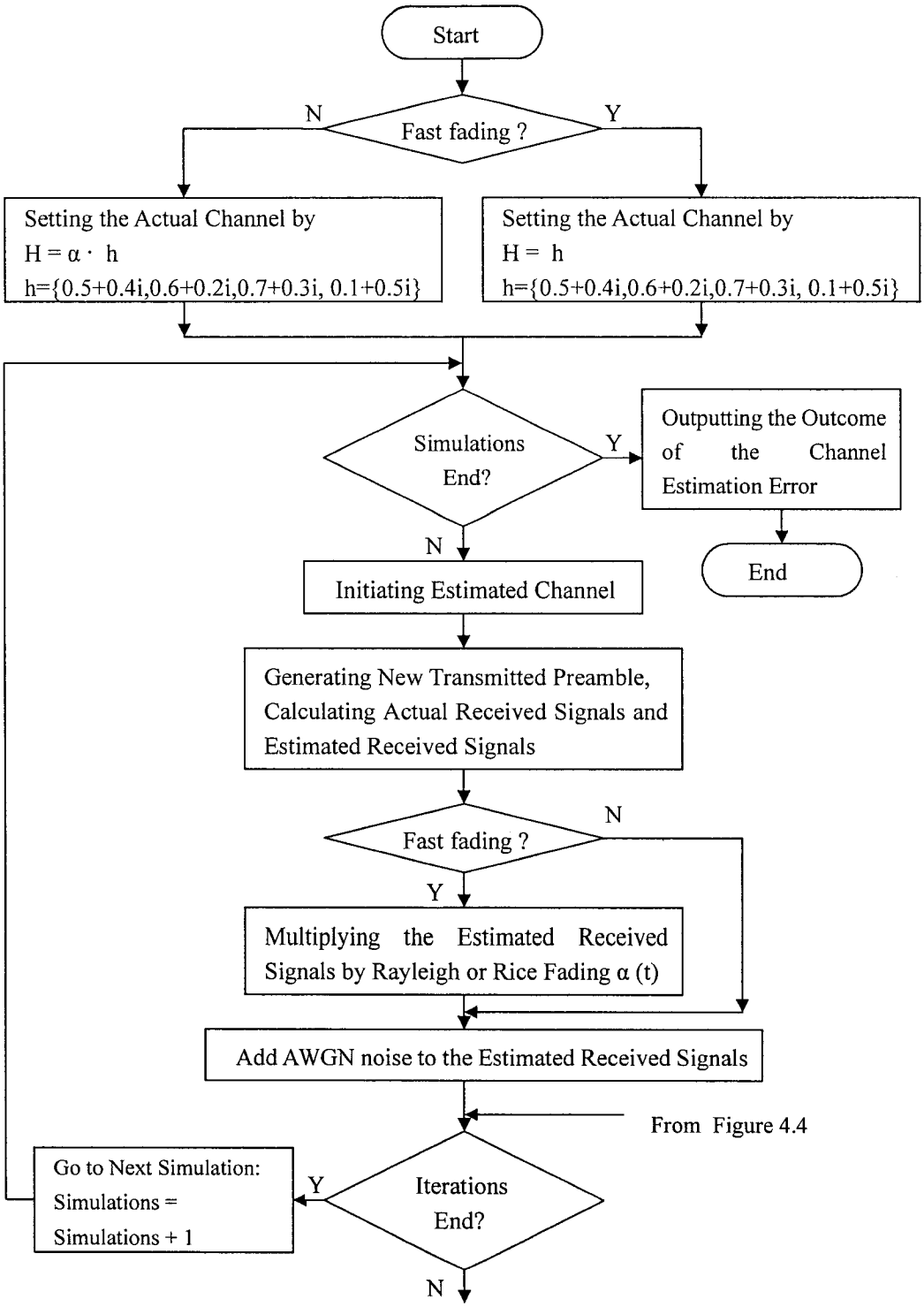


Figure 4.3 Flow Chart of the LS Channel Estimation Technique

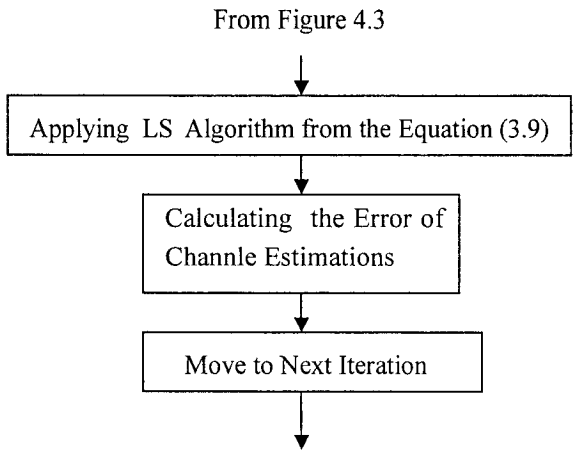


Figure 4.4 Continuation of the Flow Chart of the LS Channel Estimation Technique

- Flow chart of the MMSE channel estimation algorithm technique

In Figure 4.5 and 4.6, the first several steps in the flow are similar to those in LMS channel estimation. In the thesis, MMSE channel estimation algorithm does not follow iterative procedures. No decision on iterations is involved in the flow. But, matrix or vector calculation is involved in the flow. The average of the preamble autocorrelation in (3.19) and the cross-correlation of preamble and estimated received signals in (3.19) are both time averages on the period of transmitting preambles. The value of simulation times is same as that of LMS.

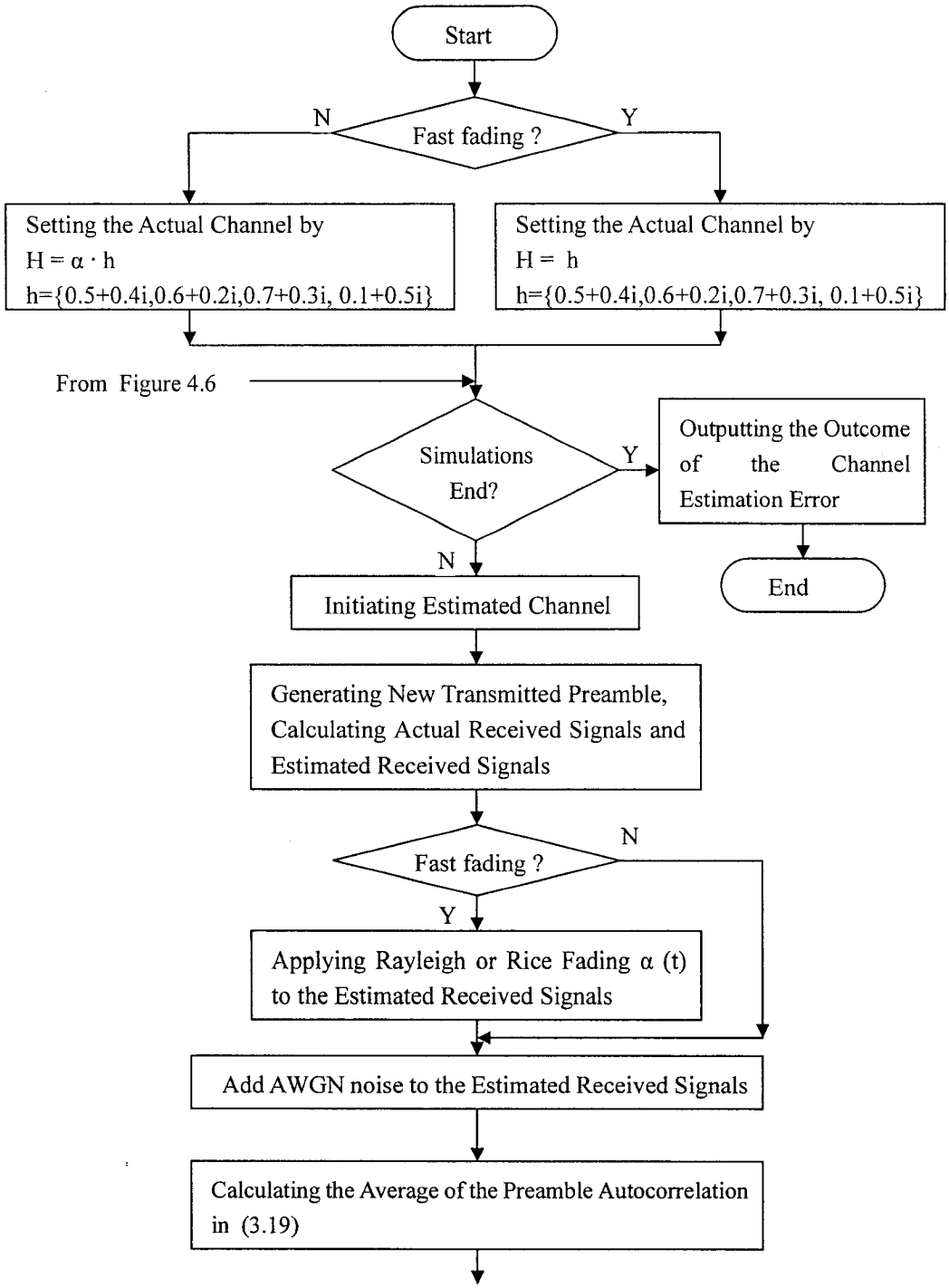


Figure 4.5 Flow Chart of the MMSE Channel Estimation Technique

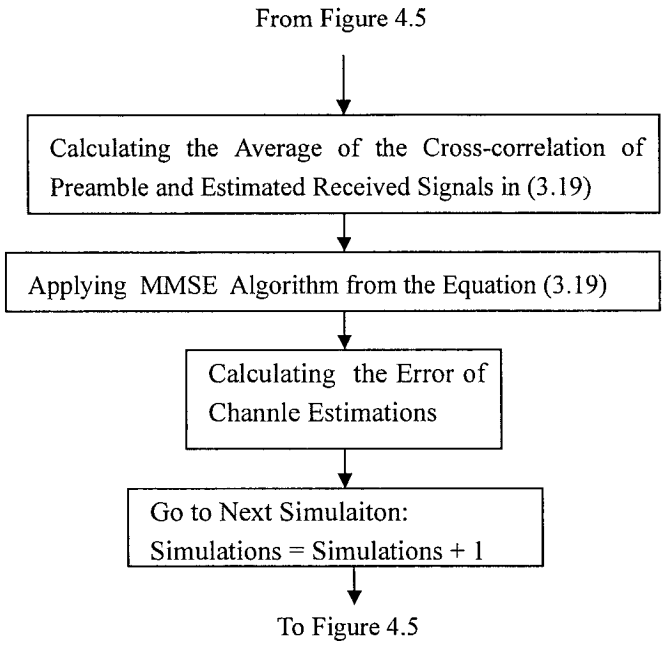


Figure 4.6 Continuation of the Flow Chart of the MMSE Channel Estimation Technique

- Flow chart of the blind channel estimation technique

The flow chart in Figure 4.7 and 4.8 describes the implementation of the blind channel estimation algorithm. The value of *simulations* is same as that of LMS. On fast fading channel, Rayleigh fading α (t) will be different on each channel. In the simulations, the periodic length of the pilot is equal to the channel length. It is an impulse function occurring at the beginning of each pilot period.

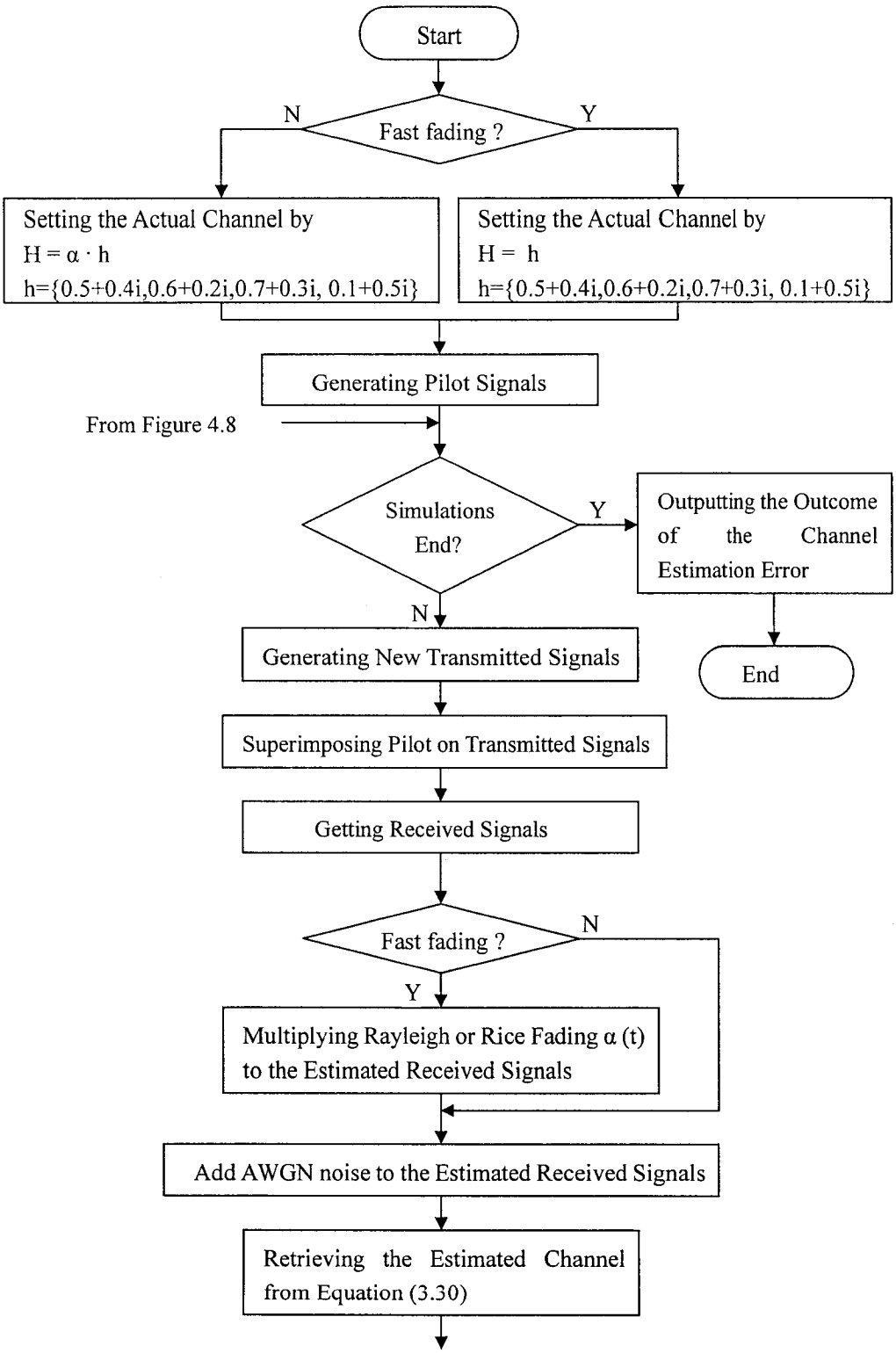


Figure 4.7 Flow Chart of the Blind Channel Estimation Technique

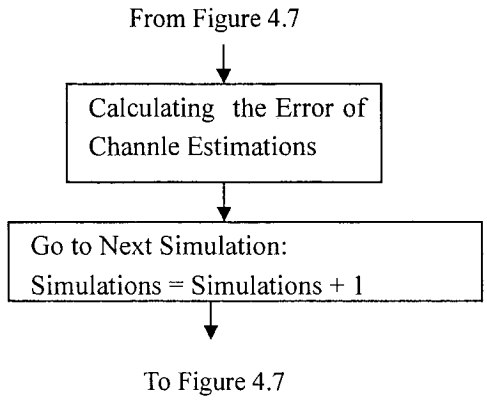


Figure 4.8 Continuation of the Flow Chart of the Blind Channel Estimation Technique

4. 2 Indoor Radio Channel Scenarios

4. 2. 1 Slow Fading Scenario

In the simulation of slow fading scenario, assumptions are made as follows: The channel fading does not change within a transmission time slot, so the Rayleigh fading or Rice fading is a constant during the transmission time slot allocated to one packet. In this case, the channel impulse function we are considering in our simulation is assumed as follows:

$$H(t) = \alpha \{(0.5+0.4i)\,\delta(t-1T) + (0.6+0.2i)\,\delta(t-2T) + (0.7+0.3i)\,\delta(t-3T) + (0.1+0.5i)\,\delta(t-4T)\} \quad (4.6)$$

α is a constant coefficient during a time slot to model Rayleigh fading or Rice fading. Here, the purpose of channel estimation is to find out $\hat{H}$ that is close to H. Blind and non-blind estimation approaches are employed in this case.

4. 2. 2 Fast Fading Scenario

If the channel changes within a transmission time slot, it will be regarded as a fast fading channel in our simulation. In this case, we make the assumption in our simulation that only Rayleigh fading changes fast, the multipath channel h(t) in equation 4.2 varies very slow such that it can be seen as invariant part within a transmission slot. Under this assumption, the coefficient α in the channel expression (4.6) will be randomly changed within a transmission slot. In this case, it is useless to try to estimate the coefficient α. However, the multipath distortion h (t) of the channel is fixed, and if it bears more weight in the channel, it is worthwhile to estimate it. Then, the channel we are trying to estimate in fast fading case is given by

$$H(t) = (0.5+0.4i)\,\delta(t-1T) + (0.6+0.2i)\,\delta(t-2T) + (0.7+0.3i)\,\delta(t-3T) + (0.1+0.5i)\,\delta(t-4T) \quad (4.7)$$

Note that compared to the slow fading case (4.6), H(t) in fast fading case does not include the Rayleigh fading coefficient α . Rayleigh fading is not estimated in fast fading case.

If a corresponding transmitted signal is s(t), the received signal can be obtained by

$$r(t) = \alpha(t) \cdot H(t) * s(t) + n \quad (4.8)$$

Obviously, in equation 4.8, $\alpha(t)$ is put outside of H(t) in the fast fading case.

During the simulation, two rates of fast fading channel are considered. Respectively, the channel changes every 10 bits and 20bits. With respect to fast fading

channel with 10 bits rate, $\alpha(t)$ will not change within the 10 bits and each channel of 10 bits has different $\alpha(t)$ having a Rayleigh or Rice random value. When α is a Rayleigh random variable, the channel is regarded as Rayleigh fast fading channel; if α is a Rice random variable, the channel is Rice fast fading channel.

4.3 Results and Discussion

The following Figures demonstrate the performances of the Bluetooth channel estimation techniques with both non-blind and blind approaches. In the simulations, we assume to investigate the probability of channel estimation errors less than 5% on both slow fading and fast fading channel. If channel estimation errors become larger, the quality of compensation for the distorted channel will degrade. Some of channel distortion still exists to cause bit error. The larger the channel estimation error is, the larger the bit errors.

- Slow fading case

The results of non-blind channel estimation techniques i.e. LMS, LS, MMSE for slow fading case are presented in Figures 4.9, 4.10, 4.11.

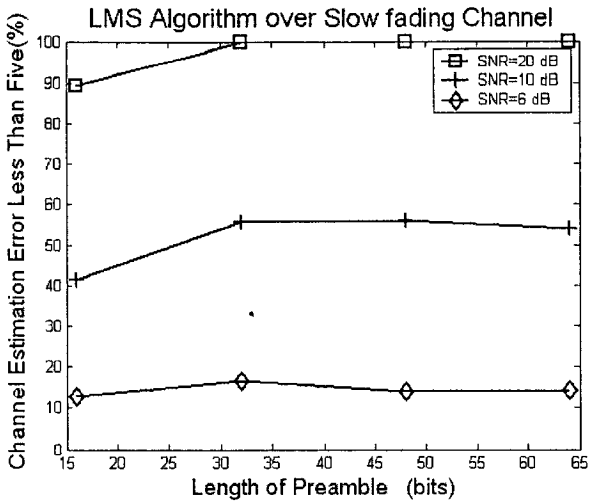


Figure 4.9 LMS algorithms over slow fading channel, α is estimated

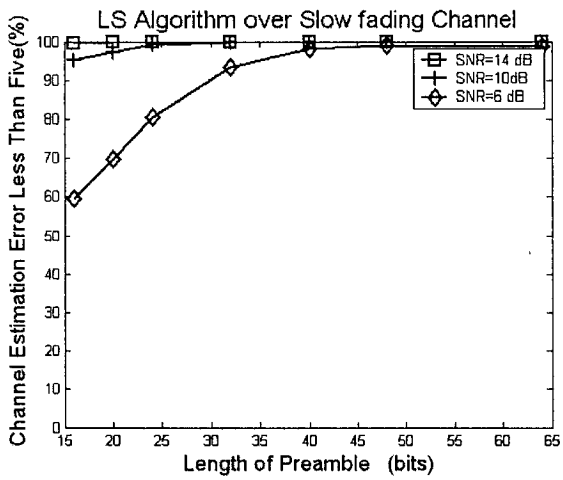


Figure 4.10 LS algorithms over slow fading channel, α is estimated

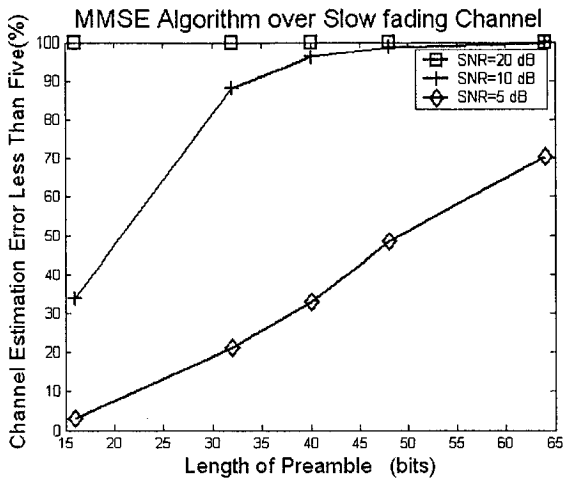


Figure 4.11 MMSE algorithms over slow fading channel, α is estimated

Comparing Figure 4.9, 4.10, 4.11, demonstrates that the LS algorithm has the best performance for Signal to Noise Ratio (SNR) 5, 10, 20 compared to the LMS and MMSE techniques. If SNR is 20 dB or a higher value, all of three algorithms can achieve perfect quality within 64 bits. When SNR decreases to SNR=10dB or 5 dB, LMS quality degrades and assumes an approximately flat curve while preamble bits increase; LS slightly drops for small preambles (<35 bits) and still achieves a good quality as the length of preamble goes to 64 bits; MMSE is more sensitive to noise than LS. If SNR=10 dB, it can achieve good quality within 64 bits. However, if SNR=5 dB, it can not. MMSE has an ascending curve like LS. From the trend of the curves of MMSE and LS, It can be expected that if the preamble is large enough the estimated errors will converge to zero. This quite complies with the performance analysis in the appendix A.

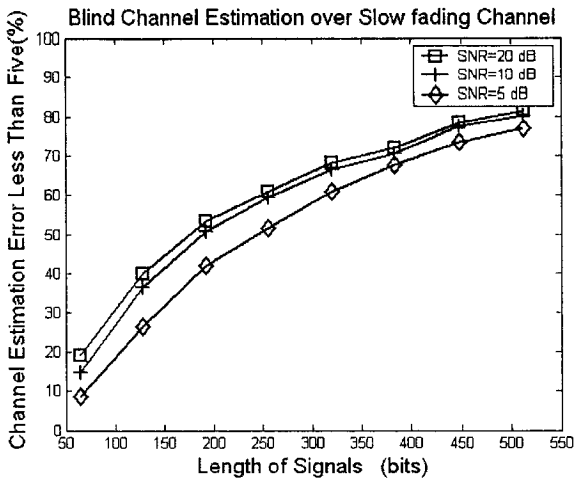


Figure 4.12 Blind channel estimation over slow fading channel, α is estimated, pilots add to sequences every 4 bits

Blind channel estimation algorithm adopted in the work superimposes pilots on information signals, then acquire the channel information from received signals. The estimation results of such blind technique on slow fading channel are shown in Figure 4.12. Obviously, the results of blind technique are worse than those of the non-blind techniques. This conclusion is obtained under the assumption that the pilot has the same power level as the information signal. If the pilot's power increases, the performance of blind channel estimation algorithm will be improved.

Comparing Figure 4.12 of blind approach and Figures 4.9, 4.10, 4.11 of the non-blind approach, it is clear that blind algorithm needs more bits of sequences to obtain the same quality as non-blind algorithm. It does not work on the small sequence, however still no loss of BW is encountered in the blind case.

With respect to the slow fading case channel estimation, because the fading factor α is included into the estimated function and considered as a constant during estimation

procedure, the estimation performance is not affected by the fading factor any more, that is the channel estimation error result is the same whether fading is deep or not.

- Fast fading case

Figure 4.13 - Figure 4.16 present performances for fast fading case, while the Rayleigh fading channel changes every 10 bits within 64 bits. $\overline{\alpha^2}$ is the second moment of Rayleigh stochastic variable and satisfies equation (2.7)

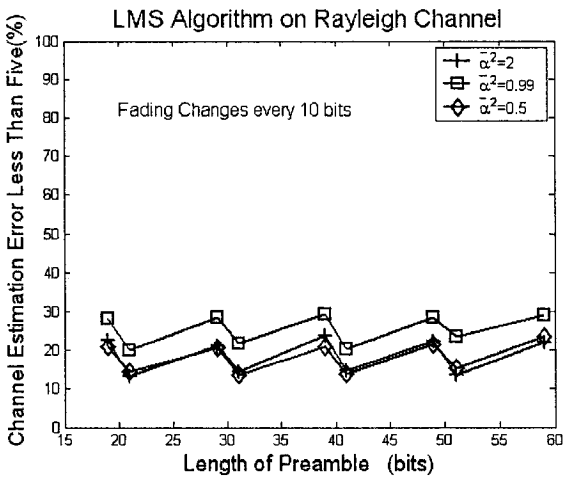


Figure 4.13 LMS algorithms over Rayleigh fast fading channel,

The channel changes every 10 bits, $\overline{\alpha^2}$ is the second moment of Rayleigh stochastic variable(2.7), α is not estimated.

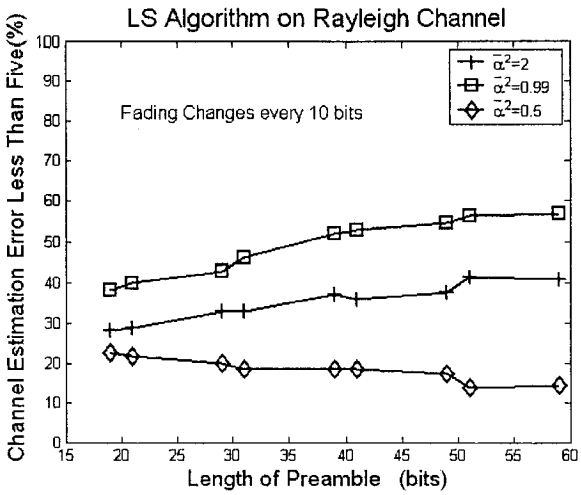


Figure 4.14 LS algorithms over Rayleigh fast fading channel,

The channel changes every 10 bits, $\overline{\alpha^2}$ is the second moment of Rayleigh stochastic variable(2.7), α is not estimated

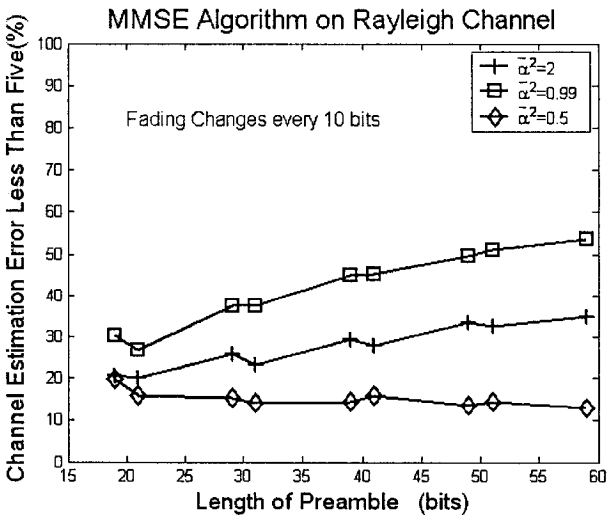


Figure 4.15 MMSE algorithms over Rayleigh fast fading channel,

The channel changes every 10 bits, $\overline{\alpha^2}$ is the second moment of Rayleigh stochastic variable(2.7), α is not estimated

When the Rayleigh fading channel changes several times within a transmission slot, the performance of the channel estimation will greatly degrade for all of these algorithms compared with the slow fading case. The deeper the fading is, the worse the quality of the channel estimation in fast fading case, which corresponds to the performance analysis in the appendix A. Of Figures 4.13-4.15, the LS algorithm has the best performance among them. However, it is still much worse than the case in Figure 4.10. Even if Rayleigh fading is small ($\overline{\alpha^2} = 0.99$), there is less than 80% chance in the LS that channel estimation error is bigger than ten. So, we can come to this conclusion if Rayleigh fading channel is time varying in Bluetooth application, non-blind channel estimation algorithm does not work well.

Figure 4.16 shows the results of the blind channel estimation technique in Rayleigh fast fading case. Comparing the channel estimation errors on the 64 bits preambles or signal in Figure 4.13-4.16, the blind channel estimation on Rayleigh fast fading has worse performance than the non-blind channel estimation techniques. This conclusion is obtained under the assumption that the pilot has the same power level as the information signal. If the pilot's power increases, the performance of blind channel estimation algorithm will be improved.

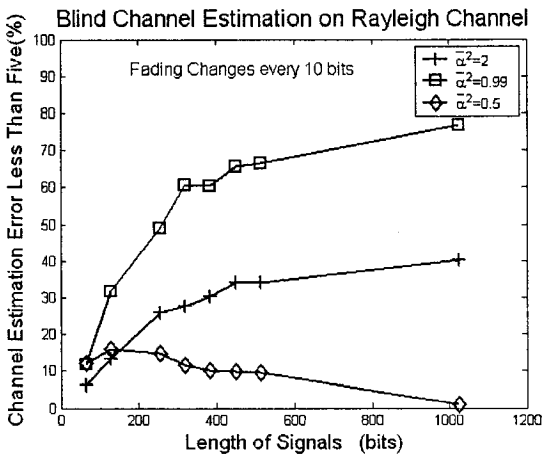


Figure 4.16 Blind channel estimation algorithms over Rayleigh fast fading channel, The channel changes every 10 bits, and pilots add to signals every 4 bits, $\overline{\alpha^2}$ is the second moment of Rayleigh stochastic variable(2.7), α is not estimated

Also, from Figure 4.13-4.16, one can find the similar curve trend to the slow fading case that LS, MMSE and the blind channel estimation algorithms have ascending curves, whereas LMS almost stays flat. That is because LS, MMSE and the blind channel estimation technique tend to receive all preamble bits then average them to estimate the channel. The random fading effect may be canceled by the average operation. So, more preamble bits will smooth the fading to achieve better quality for LS, MMSE, and blind algorithms. However, in the LMS case, from Figure 4.13, we find that the gradient algorithm is fast such that the channel estimation adapts to the fast changing channels. The curve of LMS in Figure 4.13 periodically changes while the channel changes every ten bits. So it does not matter if preamble is large or small. In other words, increasing the bits of preamble does not offer improvement in the channel estimation process.

Figures 4.17-4.20 represent the performance of the non-blind and blind channel estimation algorithms on the fast Rice fading channel.

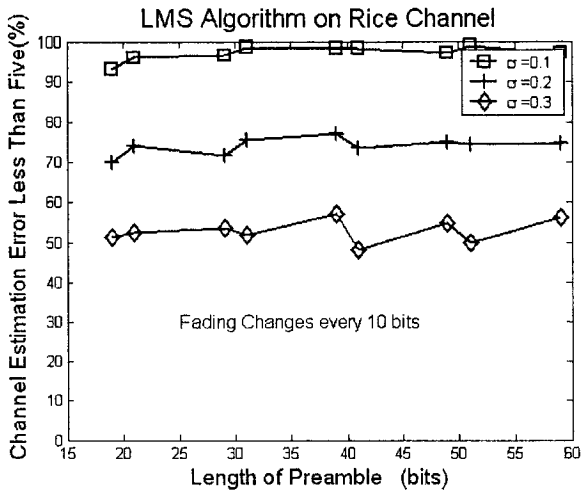


Figure 4.17 LMS algorithms over Rice fast fading channel,
The channel changes every 10 bits, σ is a parameter
related to Rice (2.10), α is not estimated

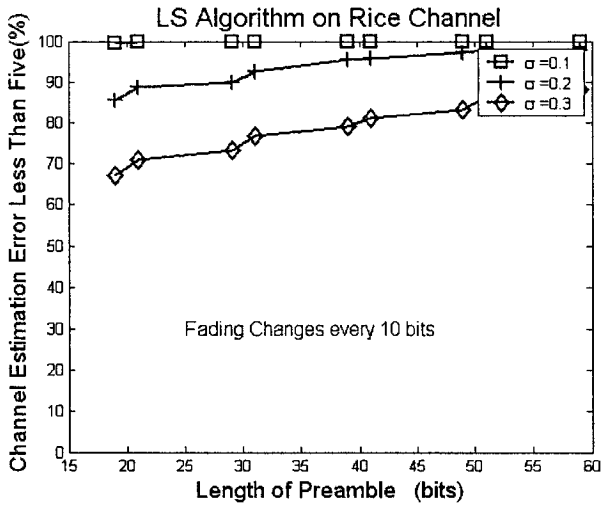


Figure 4.18 LS algorithms over Rice fast fading channel,
The channel changes every 10 bits, σ is a parameter
related to Rice (2.10), α is not estimated

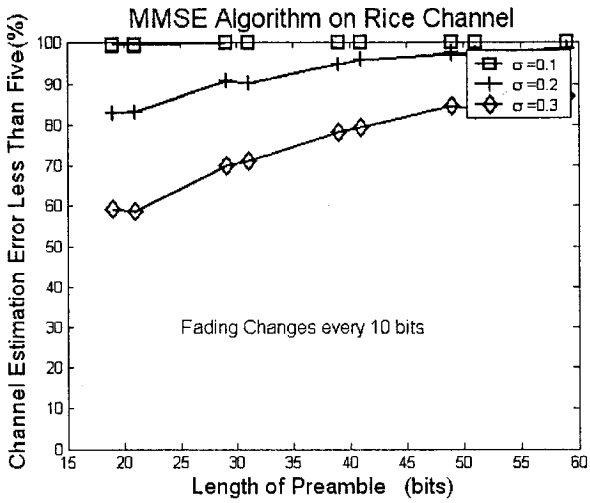


Figure 4.19 MMSE algorithms over Rice fast fading channel,
The channel changes every 10 bits, σ is a parameter
related to Rice (2.10), α is not estimated

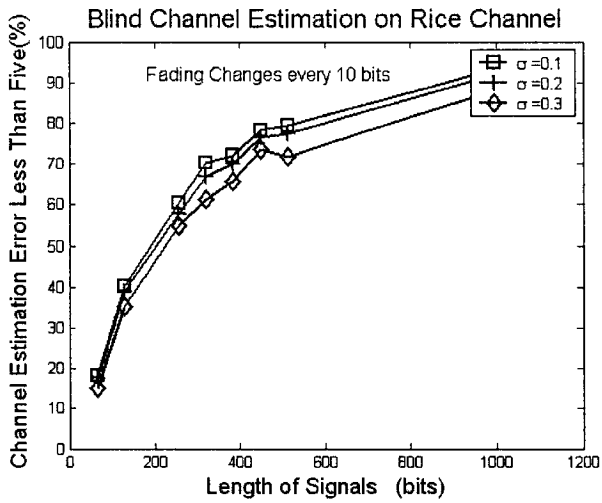


Figure 4.20 Blind channel estimation algorithms over Rice fast fading channel,
The channel changes every 10 bits, and pilots add to signals every 4 bits
σ is a parameter related to Rice (2.10), α is not estimated

Figures 4.17-4.20 and Figures 4.13-4.16 show that the channel estimations in Rice fading case have better performance than that in Rayleigh fading case for both

non-blind and blind channel estimation. This conclusion quite corresponds to theory. In Rice fading case, since there is dominant path direct to receiver from transmitter, the fading components are relatively weak compared to LOS component. As a result, the fading is shallower than that of the Rayleigh case. When the LOS component decreases, fading component becomes dominant and fading will deepen, thus becoming the Rayleigh case.

Blind channel estimation technique still shows a worse performance than the non-blind approach in Rice fast fading case. In Figure 4.20, there is less than 40% chance that channel estimation error is less than ten for the blind approach, whereas more than 70% chance in Figure 4.17 for the LMS non-blind technique.

Figures 4.21, 4.22 present the convergence property for the iterative LS and LMS algorithms. The estimation error in Figure 4.21 and 4.22 is denoted by err=$|H-\hat{H}|^2$, H is desired the channel transfer function, $\hat{H}$ is estimated channel transfer function. From Figures 4.21, 4.22, we find that the estimation error converges to a constant when the number of iterations is more than 10 bits. Based on these results, we choose iterations value in the simulation as 100 times so as to guarantee convergence.

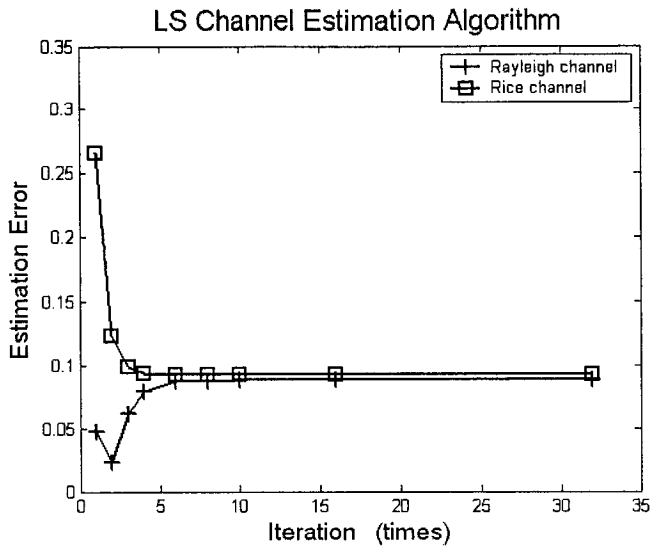


Figure 4.21 Estimation error vs. iteration for LS channel estimation (α^2=0.98) and Rice channel (σ=0.2), length of preamble equals 48 bits, AWGN is 20 dB

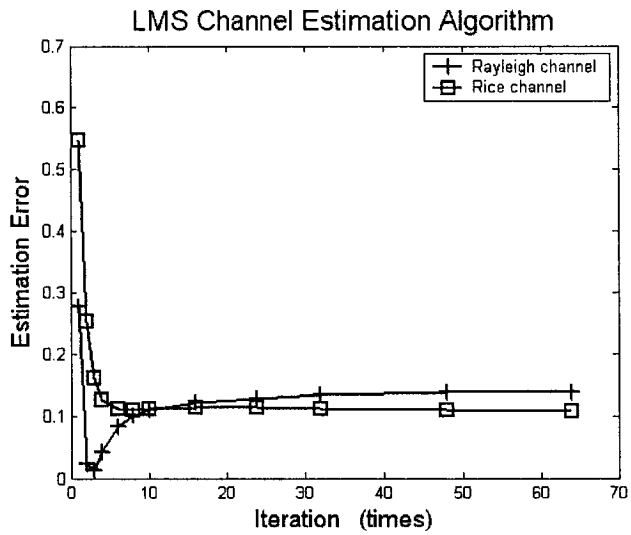


Figure 4.22 Estimation error vs. iteration for LMS channel estimation (α^2=0.98) and Rice channel (σ=0.2), length of preamble equals 48 bits, AWGN is 20 dB

Chapter 5

Conclusions and Future Work

5.1. Conclusions

In order to mitigate the impairments to the transmitted signals on wireless channel, the channel compensation technique is a common way based on the channel estimation results. In this thesis we have attempted to investigate the performances of some channel estimation techniques in several multipath fading scenarios for Bluetooth systems.

We have derived channel estimation expressions and algorithms such as Least Mean Square (LMS), Least Square (LS), Minimum Mean Square Error (MMSE) and blind channel estimation approach. In addition, we modeled the slow multipath fading and the fast multipath fading simulation system. From the simulation results and the performance analysis of these algorithms, we find that for slow fading case the channel estimation algorithms of LMS, LS, MMSE, and blind approach theoretically can achieve a perfect performance when the preamble or signal is large enough. However, for fast fading case, the results are a bit different. The channel estimation error highly depends on the fading effect for both of Rayleigh and Rice fading cases. It assumes a large value related to the mean of the Rayleigh or Rice stochastic variables even if the preamble or signal is large enough.

In chapter 4, we performed the simulations of LMS, LS, MMSE, and blind algorithms derived above for Rayleigh slow fading, Rice slow fading, fast Rayleigh fading and fast Rice fading channels. Several observations can made from the simulation results.

1. Of the non-blind algorithms, LS has a least channel estimation error for the same length of preambles among the three non-blind algorithms. LS and MMSE will get a smaller error when the preamble becomes larger, but LMS will exhibit a flat curve with more bits of preamble. LMS is more sensitive to AWGN than the other two non-blind algorithms.
2. Blind channel estimation has a worse quality than the non-blind techniques. In the simulation, it needs a much larger sequences to achieve same performance as non-blind approach. But it does not need a preamble at all. No data rate and bandwidth is wasted in the blind channel estimation.
3. On slow fading channels, and for Bluetooth application it is possible to estimate the correct channel transfer function using 64 bits access code as training sequences by adopting LMS , LS and MMSE algorithms. Blind channel estimation algorithm should be performed on larger numbers of bits instead of within 64 bits access code.
4. On fast fading channels, for the Rayleigh case the channel estimation error is too large to estimate correctly the channel transfer function for all four algorithms. The deeper the fading is, the worse the quality of the channel estimation on the fast fading channel. For Rice case the results are much better than that of Rayleigh case. It is possible to get a correct channel transfer function within 64 bits of Bluetooth access code using non-blind approaches.
5. The observation on the results on fast fading channel shows that LS or blind channel estimation algorithm tends to receive all preambles then works on them to estimate the channel; it will average all fast fading on the channels and more

preambles will smooth the fading to achieve better quality. However, in LMS the gradient algorithm is fast such that the channel estimation adapts to the fast changing channels. So it does not matter if the preamble is large or small. In other words, increasing the bits of preamble does not offer improvement in channel estimation.

5. 2. Future Work

As mentioned in chapter 1, Bluetooth technology is a low-cost and low-power application. Therefore, the channel estimator is not allowed a large number of computations. Iterative techniques may be better candidates than matrix computation. Of the iterative techniques, the investigation of complexity versus iterative channel estimation algorithms is necessary. Besides, it also needs to optimize the iterative algorithms by finding optimal convergence factor as well as algorithm improvement. On the other side, some mechanisms may be considered to trigger the channel estimation procedure only for necessary conditions. To do so, it may save power for the Bluetooth system.

The simulations in the thesis analyze the performances of two kinds of channel estimation techniques. Next, we need to test the improvement of transmission performance between devices when the channel estimation techniques are applied to compensate the Bluetooth transmission channel.

The results of the simulations reveal that for fast Rayleigh fading case the performances of the channel estimation are not satisfied using any of the techniques of blind and non-blind channel estimations in the thesis. We should turn to other techniques

to mitigate the time varying multipath fading in wireless channel, such as adaptive equalization techniques [31].

The Bluetooth channels resides in a busy 2.4G ISM band shared with many other applications i.e. WLAN. The interferences from other applications or other Bluetooth piconets poses big problems. It also greatly degrades the quality of the channel estimations when it occurs during the channel estimation procedure. Detection of this kind of interferences should be considered so as to improve Bluetooth data transmission performance. Many researches on this issue have been reported, but still more efforts are needed.

References

[1] Nathan J. Muller, "Bluetooth Demystified", McGraw-Hill , 2000

[2] Robert Morrow, "Bluetooth Operation and use", McGraw-Hill, 2002

[3] Brent A.Miller , Chatschik Bisdikian, "Bluetooth Revealed", Prentice Hall, 2000

[4] Jennifer Bray, Charles Sturman, "Connect without cable", Prentice Hall PTR, 2001

[5] Pravin Bhagwat, "Bluetooth: Technology for Short-Range Wireless Apps", IEEE internet computing, June 2001

[6] Theodore S. Rappaport, "Wireless communications: principles and practice", Prentice Hall, second edition, 2002

[7] Lin, J.; Priakis, J.G., "An enhanced optimal windowed RLS algorithm for fading multipath channel estimation", Military Communications Conference, 1993., IEEE, p1003-p1007, vol.3

[8] A.A. Saleh, R.A. Valenzuela "A statistical model for indoor multipath propagation", IEEE J. Select. Areas Commun., 5:128-137, Feb. 1987.

[9] Hong Zhao; Qiang Wang; "On frequency hop synchronization in multipath Rayleigh fading", Vehicular Technology, IEEE Transactions on, Volume: 47 , Issue: 3, Aug. 1998 Pages: 1049 - 1065

[10] Zepernick, H.-J.; Wysocki, T.A.; "Multipath channel parameters for the indoor radio at 2.4 GHz ISM band", Vehicular Technology Conference, 1999 IEEE 49th , May 1999, Pages: 190 - 193 vol.1

[11] Bluetooth SIG groups, "Specification of the Bluetooth system", ver 1.0 drsf foundation, July 1999

[12] Cheol-Hee Park; Jong-Ho Paik; Young-Hwan You; Min-Chul Ju; Jin-Woong Cho;

"Techniques for channel estimation, DC-offset compensation, and link quality control in Bluetooth system", Consumer Electronics, IEEE Transactions on , Volume: 46, Issue: 3, Aug. 2000 Pages: 682 - 689

[13] Janssen, G.J.M.; Stigter, P.A.; Prasad, R.; "Wideband indoor channel measurements and BER analysis of frequency selective multipath channels at 2.4, 4.75, and 11.5 GHz", Communications, IEEE Transactions on, Volume: 44 , Issue: 10, Oct. 1996 Pages: 1272 - 1288

[14] Seong-Cheol Kim; Bertoni, H.L.; Stern, M.; "Pulse propagation characteristics at 2.4 GHz inside buildings", Vehicular Technology, IEEE Transactions on , Volume: 45 , Issue: 3 , Aug. 1996 Pages:579 - 592

[15] Tong Zhou, G.; Viberg, M.; McKelvey, T.; "Superimposed periodic pilots for blind channel estimation", Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on , Nov. 2001 Pages:653 - 657 vol.1

[16] Matthias Patzold, "Mobile fading cannels", Wiley, 2002

[17] "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)", IEEE 802.15.1

[18] Per Jahansson; Manthos Kazantzidis; Rohit Kapoor; Mario Gerla "Bluetooth: An Enabler for Personal Area Networking", IEEE Network Sep./Oct. 2001

[19] Haartsen, J.C.; "The Bluetooth Radio System", Personal Communications, IEEE Volume: 7 , Issue: 1 , Feb. 2000 Pages:28 - 36

[20] Karnik, A.; Kumar, *A.*; "Performance Analysis of the Bluetooth Physical Layer", Personal Wireless Communications, 2000 IEEE International Conference on , 17-20 Dec. 2000 Pages:70 – 74

[21] José C. Principe, Neil R. Euliano, W. Curt Lefebvre "Neural and Adaptive Systems: Fundamentals through Simulations", 1999, Wiley.

[22] Lok, T.M.; Wei, V.K.-W.; "Channel Estimation with Quantized Observations", Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on , 16-21 Aug. 1998 Pages:333

[23] Chatschik, B.; "An Overview of the Bluetooth Wireless Technology", Communications Magazine, IEEE , Volume: 39 , Issue: 12 , Dec. 2001 Pages:86 - 94

[24] Sairam, K.V.S.S.S.S.; Gunasekaran, N.; Redd, S.R.; "Bluetooth in Wireless Communication", Communications Magazine, IEEE , Volume: 40 , Issue: 6 , June 2002 Pages:90 – 96

[25] J.D.Parsons, "Mobile Radio Propagation Channel ", Wiley, 2000

[26] Ozen, S.; Zoltowski, M.D.; Fimoff, M.; "A Novel Channel Estimation Method: Blending Correlation and Least-Square Based Approches", Acoustics, Speech, and Signal Processing, 2002. IEEE International Conference on, May 2002 Pages:III-2281 - III-2284 vol.3

[27] Galli, S.; "A New Family of Soft-Output Adaptive Receivers Exploiting Nonlinear MMSE Estimates for TDMA-Based Wireless Links", Communications, IEEE Transactions on, Dec. 2002 Pages:1935 - 1945

[28] Ging-Shing Liu; Che-Ho Wei; "Analysis of adaptive LMS estimator with cyclic sequences in complex frequency domain" Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on, Volume: 42, Issue: 12, Dec. 1995 Pages:829 - 832

[29] Seunghyun Min; Kwang Bok Lee; "Channel estimation based on pilot and data traffic channel for DS/CDMA systems", Global Telecommunications Conference, 1998. GLOBECOM 98. The Bridge to Global Integration IEEE, Volume: 3, 8-12 Nov. 1998 Pages: 1384 - 1389

[30] Mohamad, H.; Weiss, S.; Rupp, M.; "MMSE limitations for subband adaptive equalisers", Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on, Volume: 2, 3-6 Nov. 2002 Pages: 1233 – 1237

[31] Otnes, R.; Tuchler, M.; "Soft iterative channel estimation for turbo equalization: comparison of channel estimation algorithms" Communication Systems, 2002. The 8th International Conference on, 25-28 Nov. 2002 Pages: 72 - 76 vol.1

[32] J. C. Liberti, Jr., and T. S. Rappaport, "Smart Antennas for Wireless Communications, IS-95 and Third Generation CDMA Applications", Prentice-Hall 1999.

Appendix A

Performance Analysis of Channel Estimation Techniques

A.1 Slow Fading Channel

The definition of slow fading channel in this thesis implies that the channel does not change within one transmission slot or frequency hop. H_f denotes multipath channel components. H_f in the thesis is always assumed to vary so slow that it is considered as to be fixed. H_c denotes the actual channel transfer function to be estimated. $\hat{H}$ denotes the estimated channel transfer function. For slow fading channel, H_c is defined as

$$H_c = \alpha H_f \qquad (A.1)$$

α is Rayleigh fading or Rice fading coefficient. It is a constant within a hop on slow fading channel. The channel estimation error in the thesis is defined as

$$\text{error} = \frac{|\hat{H} - H_c|^2}{|H_c|^2} \qquad (A.2)$$

- **MMSE algorithm under slow fading channels**

From (3.19), the estimated channel function is $\hat{H} = E\,[D_m^* D_m^T]^{-1}\ E\,[\mu_m D_m^*]$. According to (3.10) definition, we have $\mu_m = D_m^T\ H_c$. Therefore, in ideal operating conditions (no Additive White Gaussian Noise (AWGN)), $\hat{H}$ can perfectly describe H_c as follow

$$\hat{H} = E\,[D_m^* D_m^T]^{-1}\ E\,[D_m^*\ D_m^T\ H_c] = H_c \qquad (A.3)$$

If AWGN n_m occurs on the channel, the estimated channel should be

$$\overline{H}_{n_0} = E\,[D_m^* D_m^T]^{-1}\ E\,[(\mu_m + n_m)\ D_m^*]$$

$$= E[D_m^* D_m^T]^{-1} E[\mu_m D_m^* + n_m D_m^*]$$

$$= E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*] + E[D_m^* D_m^T]^{-1} E[n_m D_m^*]$$

Note that μ_m and n_m are scalar while D_m is a vector.

In the case of $\hat{H}_{n_0}$, the channel estimation error is

$$\text{error} = \frac{|\hat{H}_{n_0} - H_c|^2}{|H_c|^2}$$

$$= \frac{|E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*] + E[D_m^* D_m^T]^{-1} E[n_m D_m^*] - E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*]|^2}{|E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*]|^2}$$

$$= \frac{|E[D_m^* D_m^T]^{-1} E[n_m D_m^*]|^2}{|E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*]|^2}$$

Assuming D_m^* and n_m are independent of each other

$$\text{error} = |E[n_m]|^2 \frac{|E[D_m^* D_m^T]^{-1} E[D_m^*]|^2}{|E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*]|^2} \qquad \text{(A.4)}$$

Theoretically, the mean of AWGN equals zero, $E[n_m] = 0$

So, in the simulations, as the number of samples becomes large enough, error goes to zero.

- **LS algorithm under slow fading channels**

(3.7) gives is the l^{th} gradient of the cost function. If we express the gradient in matrix form instead of h_l, (3.7) can be written in the alternative form

$$\nabla J(\hat{H}) = 2 \sum_{m=0}^{P-1} D_m^* D_m^T \hat{H} - 2 \sum_{m=0}^{P-1} D_m^* \mu_m$$

Letting $\nabla J(\hat{H}) = 0$, we get $\hat{H}$

$$\hat{H} = (\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1} (\sum_{m=0}^{P-1} D_m^* \mu_m) \qquad (A.5)$$

For the AWGN free case, we can also derive $\hat{H} = H_c$. If AWGN adds to the channel, the estimation error becomes

$$\text{error} = \frac{|\hat{H}_{n_0} - H_c|^2}{|H_c|^2}$$

$$= \frac{|(\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1} (\sum_{m=0}^{P-1} D_m^* (\mu_m + n_m) - (\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1} (\sum_{m=0}^{P-1} D_m^* \mu_m)|^2}{|(\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1} (\sum_{m=0}^{P-1} D_m^* \mu_m)|^2}$$

$$= \frac{|(\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1} (\sum_{m=0}^{P-1} D_m^* n_m)|^2}{|(\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1} (\sum_{m=0}^{P-1} D_m^* \mu_m)|^2} \qquad (A.6)$$

Where one can not see the same converge to zero as P$\rightarrow \infty$, however, simulation results will prove the same.

- **Blind estimation algorithm under slow fading channels**

From (3.25) and (3.29), estimation error becomes

$$\hat{H} - H_c = \frac{1}{R} \sum_{i=0}^{R-1} [\, s(iL+n) * h(n) + w(iL+n)]$$

$$= \frac{1}{R} \sum_{i=0}^{R-1} [\, s(iL+n) * h(n)\,] + \frac{1}{R} \sum_{i=0}^{R-1} [\, w(iL+n)] \qquad (A.7)$$

For AWGN, if R is large enough, $\sum_{i=0}^{R-1} [\, w(iL+n)]$ converges to zero. If we assume that s(iL+n) is random with zero mean, then for large enough R, the term $\sum_{i=0}^{R-1} [\, s(iL+n) * h(n)\,]$ converges to zero.

A.2 Fast Fading Channel

For the fast fading cases, we make the assumption that the fading channels consist of two parts - multipath distortion H_f and Rayleigh fading α. The Multipath distortion varies so slowly that it is assumed to be fixed within a transmission slot, whereas the Rayleigh fading changes fast. So, on the fast fading channel we only try to estimate multipath distortion H_f in the simulation. Under the assumption above, we define the H_c as

$$H_c = H_f \tag{A.8}$$

- **MMSE algorithm under fast fading channels**

On fast fading channel, in the presence of AWGN, the estimated channel function from (3.19) is

$$\hat{H}_{n_0} = E[D_m^* D_m^T]^{-1} E[(\alpha \mu_m + n_m) D_m^*]$$

$$= E[D_m^* D_m^T]^{-1} (E[\alpha \mu_m D_m^*] + E[n_m D_m^*])$$

Using (A.2), the channel estimation error is

$$\text{error} = \frac{|\hat{H}_{n_0} - H_c|^2}{|H_c|^2}$$

$$= \frac{|E[D_m^* D_m^T]^{-1} E[\alpha \mu_m D_m^*] + E[D_m^* D_m^T]^{-1} E[n_m D_m^*] - E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*]|^2}{|E[D_m^* D_m^T]^{-1} E[\mu_m D_m^*]|^2}$$

Assume that α is independent of μ_m and D_m^*. AWGN n_m is also independent of signals D_m^*. α, μ_m and n_m are scalars.

$$\text{error} = \frac{|(E[\alpha]-1)\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[\mu_m \mathrm{D}_{\mathrm{m}}^{*}]+\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[n_m]\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}]|^2}{|\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[\mu_m \mathrm{D}_{\mathrm{m}}^{*}]|^2} \qquad \text{(A.9)}$$

Because the mean of AWGN equals zero, we have $E[n_m]=0$. The channel estimation error can be rewritten as

$$\text{error} = \frac{|(E[\alpha]-1)\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[\mu_m \mathrm{D}_{\mathrm{m}}^{*}]|^2}{|\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[\mu_m \mathrm{D}_{\mathrm{m}}^{*}]|^2}$$

where α, μ_m are scalar, D_m is a vector.

$$\text{error} = |\mathrm{E}[\alpha]-1|^2 \frac{|\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[\mu_m \mathrm{D}_{\mathrm{m}}^{*}]|^2}{|\mathrm{E}[\mathrm{D}_{\mathrm{m}}^{*}\mathrm{D}_{\mathrm{m}}^{\mathrm{T}}]^{-1}E[\mu_m \mathrm{D}_{\mathrm{m}}^{*}]|^2}$$

$$= |\mathrm{E}[\alpha]-1|^2 \qquad \text{(A.10)}$$

In the simulations, if the preamble is large enough, the channel estimation error does not go to zero. It converges to a constant of $|\mathrm{E}[\alpha]-1|^2$.

- **LS algorithm under fast fading channels**

According to (A.5) and the (A.2), on fast fading channel in the presence of AWGN , the channel estimation error is given by

$$\text{error} = \frac{|\hat{H}_{n_0} - H_c|^2}{|H_c|^2}$$

$$= \frac{|(\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1}(\sum_{m=0}^{P-1} D_m^*(\alpha\mu_m + n_m) - (\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1}(\sum_{m=0}^{P-1} D_m^*\mu_m)|^2}{|\sum_{m=0}^{P-1} D_m^* D_m^T)^{-1}(\sum_{m=0}^{P-1} D_m^*\mu_m)|^2}$$

$$= \frac{|\sum_{m=0}^{P-1} (D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^*\alpha\mu_m - \sum_{m=0}^{P-1} (D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^*\mu_m + \sum_{m=0}^{P-1}(D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^* n_m|^2}{|\sum_{m=0}^{P-1} (D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^*\mu_m|^2}$$

Where α , μ_m and n_m are scalars, D_m is a vector. error can be rewritten as

$$\text{error} = \frac{|(\alpha-1)\sum_{m=0}^{P-1}(D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^*\mu_m + \sum_{m=0}^{P-1}(D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^* n_m|^2}{|\sum_{m=0}^{P-1}(D_m^* D_m^T)^{-1}\sum_{m=0}^{P-1} D_m^*\mu_m|^2}$$

(A.11)

- **Blind estimation algorithm under fast fading channels**

Since Rayleigh fading α is not estimated in the fast fading case in the simulations, the (3.25) should be rewritten as

$$\hat{m}(n) = \frac{1}{R}\sum_{i=0}^{R-1} [\, s(iL+n) * \alpha\, h(n) + p(iL+n) * \alpha\, h(n) + w(iL+n)]$$

$$= \frac{1}{R}\sum_{i=0}^{R-1} \alpha\, [\, p(iL+n) * h(n)] + \frac{1}{R}\sum_{i=0}^{R-1} [\alpha\, s(iL+n) * h(n) + w(iL+n)] \qquad (A.12)$$

(A.12) minus (3.27), estimation error should be

$$\hat{H}_{n_0} - H_c = \frac{(\alpha-1)}{R}\sum_{i=0}^{R-1} [\, p(iL+n) * h(n)] + \frac{1}{R}\sum_{i=0}^{R-1} [\alpha\, s\,(iL+n) * h\,(n) + w\,(iL+n)]$$

$$= \frac{(\alpha-1)}{R}\sum_{i=0}^{R-1} [\, p(iL+n) * h(n)] + \frac{\alpha}{R}\sum_{i=0}^{R-1} s\,(iL+n) * h\,(n) + \frac{1}{R}\sum_{i=0}^{R-1} w\,(iL+n)$$

(A.13)

(A.13) is the channel estimation error for the blind channel estimation algorithm. Here, s(iL+n) and w(iL+n) are assumed as random with the mean of zero, then if R is large enough , we have

$$\frac{\alpha}{R}\sum_{i=0}^{R-1} s\,(iL+n) * h\,(n) = 0$$

$$\frac{1}{R}\sum_{i=0}^{R-1} w\,(iL+n) = 0$$

Then (A.13) becomes

$$\hat{H}_{n_0} - H_c = \frac{(\alpha - 1)}{R} \sum_{i=0}^{R-1} [\, p(iL+n) * h(n)] \qquad (A.14)$$

Under the definition of (3.30), also according to (3.28), (3.31),

$$h(n) = p(n)*h(n) = \frac{1}{R} \sum_{i=0}^{R-1} [\, p(iL+n) * h(n)] = H_c \qquad (A.15)$$

(A.15) is under the assumption of the chapter 3 i.e. $0 \leq n \leq L-1$

From (A.15),(A.14), we obtain

$$\overline{H}_{n_0} - H_c = (\alpha - 1) H_c$$

Since α is a constant scalar, then

$$\text{error} = \frac{|\hat{H}_{n_0} - H_c|^2}{|H_c|^2} = \frac{|(\alpha - 1)H_c|^2}{|H_c|^2} = |\alpha - 1|^2 \qquad (3.47)$$

Therefore, even if we work on a large number of sequences, the channel estiomation error of blind channel estimation on fast fading channel converge to a constant instead of zero.