Wireless Personal Area Networks and Free-Space Optical Links

Pi Huang

A thesis submitted in partial fulfilment of the requirements of Bournemouth University for the degree of Doctor of Philosophy

13 March 2006

Bournemouth University

Wireless Personal Area Networks and Free-Space Optical Links

Abstract

This thesis is concerned with the link layer design of indoor (IrDA) and outdoor infrared links, as well as the performance of the higher layers of two major Wireless Personal Area Network (WPAN) technologies: IrDA and Bluetooth.

Recent advances in wireless technology have made it possible to put networking technology into small portable devices. During the past few years, WPAN technologies have been the subject of a tremendous growth both in research and development. Although many studies have been conducted on wireless links to address different issues on physical and link layers, wireless communications are still characterised by high error rates because of the frequently changing medium. On the other hand, performance studies of the higher layers are also very important. In this thesis, for the first time, a comprehensive study of the interactions between the higher and the lower protocol layers of IrDA and Bluetooth has been carried out to improve the overall system performance.

Mathematical models for the link layers are introduced for the infrared systems: infrared data association (IrDA) and free space optics (FSO). A model for the IrDA (indoor infrared) link layer is developed by considering the presence of bit errors. Based on this model, the effect of propagation delay on the link throughput is investigated. An optimization study is also carried out to maximize the link throughput. FSO (outdoor infrared) links are often characterized by high speed and long link distance. A mathematical model for the FSO link layer is also developed. Significant improvement of the link throughput is achieved by optimizing the link parameters.

Based on the link layer model, the performance of the IrDA higher layers (transport, session and application layers) is investigated. First, a mathematical model of TinyTP (transport protocol) is elaborated and subsequently verified by simulations. The effects of multiple connections and available buffer size are investigated. The throughput at the TinyTP level is optimized for different buffer sizes. Subsequently, the session layer, including Object Exchange (OBEX) and IrDA Burst (IrBurst) protocols, is studied and modelled. The derived mathematical model is verified by simulation results. A set of protocol parameters and hardware selection guidelines is proposed to optimize the overall system performance while also keeping the hardware

requirement to a minimum. Finally, two rapidly developing IrDA applications, IrDA financial messaging (IrFM) and IrDA simple connection (IrSC), are studied. IrFM is investigated by comparison to other digital payment technologies, while the performance of IrSC is compared in two different technical approaches.

In order to improve the throughput and minimize the transmission delay for the Bluetooth data applications, a systematic analysis is carried out for the Bluetooth Logical Link Control and Adaptation Layer Protocol (L2CAP). L2CAP is layered above the Bluetooth link layer (Baseband) and is essential to Bluetooth data applications. A simple and intuitive mathematical model is developed to derive simple equations for the L2CAP throughput and the average packet delay. The derived throughput equation, which is validated by simulations, takes into account bit errors as well as packet retry limits. Finally, a number of easy-to-implement performance enhancement schemes are proposed, including the optimum use of the protocol parameters.

Dedication

献给我的父母亲，黄泰林，梁秀琼

感谢他们对我无止境的爱和支持

献给我的未婚妻金婷

感谢她一直以来对我的爱和关怀

Publications resulting from thesis

Journals Publications:

[1] Pi Huang and A. C. Boucouvalas, "Bluetooth L2CAP Modelling and performance Analysis", accepted for publication in *Mediterranean Journal of Computers and Networks*, Feb 2006.

[2] Pi Huang and A. C. Boucouvalas, "Future Personal 'e-Payment': IrFM", *IEEE Magazine on Wireless Communications*, volumn 13, Issue 1, page(s): 60-66, Feb 2006.

[3] Pi Huang and A. C. Boucouvalas, "OBEX over IrDA: Performance Analysis and Optimization by Considering Multiple Applications", accepted for publication in *IEEE/ACM Transaction on Networking*, (to appear) Feb, 2007.

[4] Pi Huang and A. C. Boucouvalas, "Modelling IrSC: Proposal Options and Performance Analysis", accepted for publication in *IEE Proceedings Circuits, Devices & Systems*, Dec, 2005

[5] Pi Huang and A. C. Boucouvalas, "Modelling OBEX over IrDA Protocol Stack", accepted for publication in *International Journal of Communication Systems (Wiley)*, Sept, 2005.

[6] Pi Huang, P. Chatzimisios and A. C. Boucouvalas, "Optimising IrDA throughput by including processing time with physical layer consideration", *Optical Society of America (OSA) Journal of Optical Networking (JON)*, Issue: June2005, Vol. 4, No. 6, Jun, 2005

[7] Pi Huang and A. C. Boucouvalas, "Enhancing the Performance of High Speed Free-Space Optical Links through An Adaptive Link Layer", submitted to *IEE Proceedings Communications*, 2005.

[8] A. C. Boucouvalas and Pi Huang, "Gbit/s Data Rate IrDA Protocol Performance Evaluation", the *Mediterranean Journal of Computers and Networks*, Issue: October2005, Vol. 1, No. 2, Oct. 2005.

[9] A. C. Boucouvalas and Pi Huang, "Modelling and Optimizing TinyTP over IrDA Stacks", *EURASIP Journal of wireless communication network*, volume 2005:1, pages: 45-56. DOI: 10.1155/WCN.2005.45

[10] Pi Huang and A. C. Boucouvalas, "IrBurst Modelling and Performance Analysis in the Presence of Transmission Errors", submitted to *Journal of Wireless Personal Communications (Kluwer)*, Sept, 2004.

Conferences Publications:

[11] Pi Huang, A. C. Boucouvalas and P. Chatzimisios, "Modelling the Bluetooth Logical Link Control and Adaptation Layer", in the *proceeding of PREP2005*, Page: 58-59, Lancaster, Mar 2005.

[12] Pi Huang and A. C. Boucouvalas, "Delay Analysis for Bluetooth Baseband ACL Packets", *Convergence of Telecommunications, Networking & Broadcasting Symposium 2005 (PGNET 2005)*, Liverpool, Jun 2005.

[13] Pi Huang and A. C. Boucouvalas, "Modelling IrDA Transport Layer TinyTP", *Communication Systems, Networks and Digital Signal Processing 2004 (CSNDSP2004)*, New Castle, Jul 2004. ISBN: 0-7017-0177-3, Page: 374-377.

[14] Pi Huang and A. C. Boucouvalas, "Analysis of the IrFM Digital Payment System", *Communication Systems, Networks and Digital Signal Processing 2004 (CSNDSP2004)*, New Castle, Jul 2004. ISBN: 0-7017-0177-3, Page: 478-482.

[15] Pi Huang and A. C. Boucouvalas, "OBEX Performance Evaluation and Parameter Optimization for High Speed IrDA Links", *IEEE International Conference on Communications (ICC 2004)*, Paris, Jun 2004. ISBN: 0-7803-8534-9, Volume: 7, Page: 3849 - 3853.

[16] Pi Huang and A. C. Boucouvalas, "Analysis of the High Speed Infrared Information Transmission Protocol: IrBurst", *Convergence of Telecommunications, Networking & Broadcasting Symposium 2004 (PGNET 2004)*, Liverpool, Jun 2004. ISBN: 1-9025-6010-8, Page: 328-332.

[17] Pi Huang and A. C. Boucouvalas, "A Link Layer Analytical Model for High Speed Full-Duplex Free Space Optical Links", *International Optical Wireless Communications Symposium (IOWCS)*, Warwick, Sept 2003. ISBN: 0-902683-68-3, Volume: 1, Page: 36-45.

[18] Pi Huang, P. Chatzimisios and A. C. Boucouvalas, "A study of link propagation delay and processing speed for IrDA links at Gbit/s data rate", *International Optical Wireless Communications Symposium (IOWCS)*, Warwick, Sept 2003. ISBN: 0-902683-68-3, Volume: 1, Page: 21-30.

[19] Pi Huang and A. C. Boucouvalas, "OBEX and High Speed IrDA Links", *International Optical Wireless Communications Symposium (IOWCS)*, Warwick, Sept 2003. ISBN: 0-902683-68-3, Volume: 1, Page: 73-78.

NOTE: Most of the above publications are available at:

http://decweb.bournemouth.ac.uk/staff/phuang/

Acknowledgements

This work was carried out during my stay as a PhD student at Bournemouth University from September 2002 to September 2005. It would not have been possible without the help and friendship of several people.

First of all, I would like to thank my supervisor, Prof. Tony Boucouvalas, for his guidance and assistance, throughout my research project. Being my mentor, he provided me not only with insightful discussions, but also with the necessary degree of freedom. I am truly grateful for all his help and support.

Special thanks to Jim Roach and the rest of the management team of the School of Design, Engineering and Computing who have provided generous support throughout the studies. I am grateful to my past and present colleagues, Petr Voles, Periklis Chatzimisios, Piotr Stepien, Adam Polus, Vassilis Vitsas, Rahul Premraj, John Kanyaru, Tom Teng and many others for their friendship, help and excellent working environment they have provided over the years. I would also like to acknowledge many of my friends, ZhiDA Li, MingHua Liu, Johana Urbanova, Fan Yang and YuLei Gong for the wonderful supports.

I am gifted with my parents who have been believing and supporting me throughout my entire life. I am really grateful for their continuous encouragement and limitless love in the pursuit of my academic degrees.

Contents

List of Figures

List of Tables

Abbreviations

ABM	Asynchronous Balanced Mode
ACL	Asynchronous Connectionless
ACK	Acknowledgement
AP	Access Point
APD	Avalanche Photodiode
ARM	Asynchronous Response Mode
ARQ	Automatic Repeat request
BER	Bit Error Rate
BOF	Beginning of Frame
BPSK	Binary Phase Shift Keying
BSS	Basic Service Set
CA	Collision Avoidance
CDMA	Code Division Multiple Access
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
DD	Direct Detection
DSSS	Direct Sequence Spread Spectrum
EOF	End Of Frame
ETSI	European Telecommunication Standards Institute
F-bit	Final bit
FCC	Federal Communications Commission
FCS	Frame Check Sequence
FEC	Forward Error Correction
FHSS	Frequency Hopping Spread Spectrum
FIR	Fast Infrared
FSO	Free Space Optics
GBN	Go-Back-N
GSM	Global System for Mobile communication
HDLC	High-level Data Link Control
HCI	Host Controller Interface
HEC	Header Error Check
I-frame	Information frame
IAS	Information Access Service
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
LM-MUX	IrLMP Multiplexer
IT	Information Technology
IrBurst	IrDA Burst Protocol
IrDA	Infrared Data Association
IrFM	Infrared Financial Messaging
IrLAP	IrDA Link Access Protocol
IrLMP	Infrared Link Management Protocol
IrSC	IrDA Simple Connection
IM	Intensity Modulation
IR	Infrared
ISM	Industrial, Scientific and Medical

ISO	International Organization for Standardization
LAN	Local Area Network
L2CAP	Logical Link Control and Adaptation Layer Protocol
LED	Light Emitting Diode
LMP	Link Manager Protocol
LSAP	Link Service Access Point
LOS	Line Of Sight
MAC	Media Access Control
MIMO	Multiple Input Multiple Output
NACK	Negative Acknowledgement
NRM	Normal Response Mode
OBEX	Object Exchange Protocol
OSI	Open Systems Interconnection
P-bit	Poll bit
PDA	Personal Digital Assistant
PDU	Packet Data Unit
PHY	Physical layer
PIN	Positive Intrinsic Negative photodiode
POS	Point of Sale
PTD	Personal Trust Device
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
REJ S-frame	Reject Supervisory frame
RF	Radio Frequency
RFCOMM	Radio Frequency Communications protocol
RR S-frame	Receive Ready Supervisory frame
S-frame	Supervisory frame
SCO	Synchronous Connection Oriented
SDP	Service Discovery Protocol
SDU	Service Data Unit
SIG	Special Interest Group
SIR	Serial Infrared
SNR	Signal-to-Noise Ratio
TAT	Turn Around Time
TCP/IP	Transmission Control Protocol/Internet Protocol
TCS	Telephony Control Protocol Specification
TDMA	Time Division Multiple Access
TinyTP	IrDA Tiny Transport Protocol
TPE	Throughput Efficiency
TTPSAP	TinyTP Service Access Point
U-frame	Unnumbered frame
UWB	Ultrawideband
VFIR	Very Fast Infrared
WLAN	Wireless LAN
WMAN	Wireless Metro Area Network
WPAN	Wireless Personal Area Network
WTT	Window Transmission Time
WWAN	Wireless Wide Area Network

Chapter 1. Introduction

1.1 Motivation

In recent years there has been a rapid increase in the use of mobile and portable computing devices. The progress in technology has addressed the issues of high unit price and low data rate, thus driving significant growth in the popularity of wireless devices. Wireless personal area network (WPAN) communications, including IrDA and Bluetooth, are increasingly used to reduce the inconvenience of wired connections and to provide local area connectivity for portable computing devices. Laptops, personal digital assistants (PDA's) and mobile phones are becoming ever more popular and ever more powerful with multimedia capabilities. This increase in mobile computing has led to a greater demand for wireless data connectivity with comparable service level to that of wired networks [1][2].

In the field of wireless communications, radio frequencies (RF) and infrared (IR) optics are competing transmission technologies that are being considered as complementary transmission media [3][4]. Radio communications have a long transmission range and are typically transmitted omnidirectionally. Because of this, radio is preferred when user mobility is of prime importance. However, most of the radio frequency bands need operation licenses. The limited license free bands (i.e. 2.4 GHz, 5 GHz) are used by many different technologies (i.e. Bluetooth, Wi-Fi). Thus, the interferences between devices using different radio technologies can be severe [108][109][110]. Infrared is preferred when point-to-point links of high capacity are necessary and when international compatibility is required [3][5]. Infrared communications have the advantages of high available bandwidth, low component power consumption, worldwide license-free operation and inherently high security. However, unlike radio, most of the infrared communications require a line of sight (LOS) transmission path which restricts certain applications. In addition, the transmission power is limited by the eye safety regulation [6]. Thus, the infrared transmission range is relatively short compared to that of radio.

Infrared links are widely implemented as the solutions for both indoor short distance and outdoor long distance wireless communications [9][10]. In order to develop a universal standard for indoor infrared communication, the Infrared Data Association (IrDA) was formed in 1993 by major IT companies. IrDA has worked steadily to establish specifications for a low cost, interoperable, and easy-to-use infrared communication technology. The resultant IrDA 1.0 protocol standard [39], first released in 1995, specifies point-to-point directed half-duplex links with a data rate up to 115.2 Kbit/s and a range up to 2 metres using the standard serial port interface. Version 1.1 [40] of the protocol supports data rates up to 4 Mbit/s using additional hardware with a recent extension to 16 Mbit/s. Nowadays, infrared

data communication technologies as defined by IrDA ship in millions of devices each year [11]. A large amount of work has been carried out to address different issues of the IrDA physical layer. In order to achieve high overall system performance, it is however essential to optimize the other layers of the IrDA protocol stacks and ensure that the lower layer is efficiently utilized by the higher layers.

Besides indoor infrared communications, outdoor infrared communications are also under rapid development. High available bandwidth, world wide license-free operation and deployment simplicity ensure that free space optical (FSO) systems (outdoor infrared) are one of the most promising approaches for addressing the emerging broadband access market and its "last mile" bottleneck [12]. Such robust systems, which establish communication links by transmitting laser beams directly through the atmosphere, have matured to the point that off the shelf products are now available [13]. In recent years, numerous FSO products have been set up. Available systems offer capacities in the range of 100Mbps to 2.5Gbps, and demonstration systems report data rates as high as 160Gbps [14]. Since the signal to noise (STN) ratio of the FSO links is significantly affected by the frequent changing weather conditions [3][5], one of the biggest challenges to FSO is the relatively low link operational availability compared to that of the radio links. The FSO systems however can be designed to be more resilient to low STN radio and achieve higher throughput by optimizing link layer parameters according to the real time link parameters.

Bluetooth is a standard for short range, low power, and low cost radio communication. Although originally envisioned as a cable-replacement technology by Ericsson in 1994 [15], embedded Bluetooth chips are now widespread in numerous types of devices. They include portable devices (PC's, PDA's, mobile phones), data peripherals (mice, keyboards, joysticks, printers), and audio peripherals (headsets, speakers, stereo receivers). In 1998, Ericsson joined forces with other international companies to form the Bluetooth Special Interest Group (SIG). Joint work by the SIG members allowed the Bluetooth vision to evolve into open standards to ensure rapid acceptance and compatibility in the marketplace. Bluetooth technology is already supported by over 2100 companies around the world [106]. The resulting Bluetooth specification, developed by the Bluetooth SIG, is open and freely available for developers worldwide. Recently, based on the Bluetooth Specification, IEEE WPAN standard 802.15 has been developed [7][8]. Besides the voice applications (i.e. headsets), Bluetooth technology is increasingly used as a method for data exchange (i.e. file exchange, Internet). The Bluetooth Logical Link Control and Adaptation Layer Protocol (L2CAP) bridges between the Bluetooth link layer (Baseband) and data applications. In order to improve the performance of the Bluetooth data applications, it is very important to study and optimize L2CAP layer.

The thesis is primarily concerned with the design issues on the data link, transport, session and application layers of the International Organization for Standardization (ISO) Open System Interconnect (OSI) seven-layer reference model.

Data link layer design is very important as it can minimize physical and link layer delays and increase the efficiency for the information transfer that will utilize the considered infrared and radio links. The performance link layer may be measured by the link throughput efficiency (also known as link utilization) and the packet delay. Throughput efficiency expresses the time portion of the total period during which the medium successfully transfers information between stations. It takes into account all significant factors that affect performance, including (a) the packet transmission time, (b) the medium access mechanism, (c) the transmission control passing scheme, (d) the transmission errors introduced by the wireless medium, (e) the acknowledgement delays and (f) the retransmission delays. Packet delay expresses the average time for a packet to be transmitted from the source to the destination. It takes into account various delays, including (a) the medium access delay, (b) the transmission delay, (c) the propagration delay, (d) queue delay and (e) retransmission delay.

Besides optimizing the physical and link layers, having efficient high level protocols is equally important to communication systems. The design of higher layers must make sure that they are compatible with and interact effectively with lower layers. Additionally, the higher layers should have minimum protocol overhead, as well as keeping the hardware requirement to a minimum. In this thesis, the higher layers of IrDA and Bluetooth are studied by examining (a) the high level buffer sizes, (b) the packet processing delays, (c) the high level throughput efficiency, (d) the optimum data packet sizes, (e) the packet retransmission timeouts and (f) security.

1.2 Statement of the problems

Link layer design must minimize physical hardware latency delays and minimize the link protocol delays such as medium access, link turnaround and packet retransmission. An efficient link layer should have high link utilization while minimizing the transmission error at the same time. Increasing the amount of data which can be transmitted before reversing link direction will reduce the relative link layer overhead and reduce the frequency of link turnaround, thus tending to increase throughput. However, increasing the data size also increases the packet error probability, thus increasing the number of retransmissions and decreasing throughput. In order to efficiently utilize the IrDA physical link, it is essential to model and optimize the IrDA link layer. By using a similar approach in modelling the IrDA

link layer, a robust FSO link layer can be designed to increase the link operational avalaiblity and improve the link throughput.

In contrast to the studies devoted to the lower layers (physical and link) of IrDA and Bluetooth, the number of studies on higher layers is limited. In order to achieve high overall system throughput and low transmission delay, it is important to examine the interaction between lower and higher layers.

Most of the published IrDA and Bluetooth link layer evaluations focus on link layer performance by assuming that data packets of infinite size are always ready to transmit and that there is a single application operating. In reality, higher layers offer finite size packets to lower layers and may manage multiple applications simultaneously. For instance, TinyTP (IrDA transport layer) allows multiple applications to operate the IrDA link concurrently [44]. OBEX (IrDA session layer) operates in a similar way to a 'stop and wait' protocol and has a finite maximum packet size [45]. In order to provide a higher level of transmission reliability, higher layers may also implement separate error retransmission schemes and packet timeout timers, such as the Blutooth L2CAP layer [104]. These features of higher layers restrict the use of lower layer. Thus, the analysis of the link layers cannot be directly applied to model the behaviours of the higher layers. In order to have accurate analyses on the overall IrDA and Bluetooth system behaviours, it is important to develop new mathematical models to include various characteristics of the higher layers as well as taking the lower layers into account.

1.3 Outline of research work

This work focuses in particular on the efficient link layer design of the indoor (IrDA) and outdoor (FSO) infrared communications, as well as the higher layer performance of two major WPAN technologies: IrDA and Bluetooth. The following summarises the main components of the research in the thesis:

(a) Link layers of IrDA and FSO

- In order to examine the performance of IrDA IrLAP protocol (link layer) operating at Gbit/s data rates, a mathematical model is developed by including the effect of propagation time. Based on the model, optimization equations are derived for the window and frame size for any given bit error rate (BER). The effect of propagation time and minimum turnaround time on throughput at different data rates has been studied.
- By applying a similar approach in deriving the IrLAP throughput equation, a mathematical model for the full-duplex FSO links is developed using the

Asynchronous Response Mode (ARM) of High-level Data Link Control (HDLC) as the link layer protocol. Subsequently, optimum window and frame size are obtained to maximize the link throughput by using a numerical method. In order to apply the optimum values, a simple adaptive real time algorithm is developed.

(b) Higher layers of IrDA

- A comprehensive mathematical model for IrDA Tiny Transport Protocol (TinyTP) is derived by considering multiple IrDA application connections and taking the underlying IrDA protocol stacks into account. The accuracy of the model is verified by simulations. An optimization study is carried out for the system. Several optimization equations for IrLAP window and frame sizes are derived and later validated.
- The IrDA session protocol Object Exchange (OBEX) is analysed and a mathematical model is developed to derive the throughput equation for OBEX. In order to improve the system performance, optimum OBEX packet size and IrLAP link parameters are studied. The optimum equations for window and frame are derived for both individual and simultaneous optimizations and then verified by the exact numerical approach.
- To tackle the OBEX inefficiency in transmitting large size objects, a new session protocol IrDA Burst (IrBurst) is proposed and developed by IrDA. The performance of IrBurst and OBEX is compared.
- A comprehensive analysis of IrDA Financial Massaging (IrFM) is carried out. Important issues in designing the digital payment system are examined for the IrFM system. Special attention is given to the IrDA connection procedure. Several design guidelines are provided on link data rate, IrDA turnaround time and the Exchange Station Identification (XID) sending interval. The guidelines aim to achieve fast IrDA connections while keeping the hardware requirements to the minimum.
- IrDA Simple Connection (IrSC) is a protocol dedicated for fast connection and instant data transmission between IrDA devices. In this work, two technical approaches (IrSC1 and IrSC2) for IrSC are investigated. Simple equations for calculating the transmission delay for the unchanged IrDA standard, IrSC1 and IrSC2 are derived respectively. The transmission delay of different approaches is examined and compared. The advantages and disadvantages of IrSC1 and IrSC2 are discussed.

(c) Higher layer of Bluetooth

- A systematic analysis for the Bluetooth L2CAP protocol is carried out. The throughput equation of L2CAP is derived by considering the effect of the Baseband layer and taking into account the presence of bit errors. The mathematical model is then verified by the simulation. Bluetooth protocol parameters are optimized including the effects of the Flush Timeout and L2CAP Packet data unit (PDU) size.
- An inclusive analysis is carried out for the average packet delay for Bluetooth Baseband ACL packets and L2CAP PDU's. Precise yet simple delay equations are derived by including all the possible network activities. Based on the developed delay model, the delay of six different Baseband packets is examined. The average delay of DH packets (without FEC protection) and DM packets (with FEC protection) is investigated and compared.

1.4 Thesis outline

The thesis consists of the following parts: Chapter 2 discusses infrared and Bluetooth connectivity, as well as giving an overview of the relevant body of literature. Chapter 3 studies the link layer of both indoor and outdoor infrared. Chapters 4, 5, 6 study the IrDA higher layers including transportation, session and application layers. Chapter 7 studies the throughput and the average packet delay of the Bluetooth L2CAP layer. Finally, Chapter 8 concludes the thesis and gives suggestions for future research.

In particular, Chapter 2 provides background information to the thesis including an overview of indoor wireless communication technologies, free space optics, IrDA and Bluetooth data communication protocols. It presents the special characteristics of the infrared and the radio medium and discusses the design challenges of the higher layer protocols. Finally, this chapter reviews the research that has been carried out in the area of infrared and Bluetooth communications.

Chapter 3 presents link layer mathematical models for high speed infrared links (both indoor and outdoor) by using the "window transmission time" model. The link layer throughput equations are derived by including the presence of bit errors and propagation delay. Optimization study is carried out for both link layer models. The optimum equations are derived for the window and frame size. The effect of propagation delay is investigated for the high speed links. Finally, a real time adaptive algorithm is developed and presented for the adaptation of optimum window and frame size.

Chapter 4 presents the IrDA transport layer TinyTP and describes the details of TinyTP functionality. It examines the TinyTP performance by developing a mathematical model which allows derivation of the TinyTP throughput taking into account the lower IrDA

protocol stacks. Simulations are carried out to examine the accuracy of the model. The TinyTP receiver window size and the IrLAP window and frame sizes are optimized for the maximum throughput for any given BER. The effects of IrLAP turnaround time and the receiver buffer size are investigated for the 16 Mbit/s IrDA links.

Chapter 5 investigates the IrDA session layer: OBEX and IrBurst. An analytical model is developed for OBEX. Based on the model, the OBEX and link layer parameters are optimized which maximizes system throughput for any BER. The optimization analysis is verified by comparing the results derived from the exact algorithm. The effects of maximum OBEX packet size and OBEX turnaround time on throughput are examined for different link data rates. Subsequently, the detailed operation of IrBurst is analysed. Finally, the performance of OBEX and IrBurst is compared.

Chapter 6 studies the IrLAP connection procedure and derives the link connection delay for different IrDA applications. It then examines the performance of IrFM by considering the major performance factors of a digital payment system such as connection speed, security, reliability, simplicity and interoperability. The IrFM is also compared with other competing or potential technologies. To compare the performance of two different technical approaches to the new application IrSC, simple equations are derived for calculating the transmission delay for the unchanged IrDA standard, IrSC1 and IrSC2 respectively. Finally, the disadvantages and advantages of IrSC1 and IrSC2 are discussed based on the analytical results.

Chapter 7 presents the detailed properties of the Bluetooth Baseband and L2CAP layers. A comprehensive mathematical analysis is then carried out. The L2CAP throughput equation and the delay equations to calculate the Baseband packets and L2CAP PDU are derived from the analysis. The mathematical analysis is verified by simulations. To examine the performance of the Bluetooth system, numerical results are presented based on the derived equations. Bluetooth device design guidelines for choosing appropriate Flush Timeout and L2CAP PDU size are provided.

Chapter 8 concludes the thesis and proposes directions for future research in the field of infrared and Bluetooth connectivity.

Chapter 2. Background

2.1 Introduction

This chapter provides background information to the research presented in the thesis. The outline of this chapter is as follows: Section 2.1 provides an overview of wireless communications and compares IR and RF transmission media. Section 2.2 describes indoor wireless connectivity and presents current standards for WLAN and WPAN. Section 2.3 provides an overview of the IrDA and Bluetooth protocol stacks. Section 2.4 gives an introduction to the FSO systems. Section 2.5 discusses the design challenge of infrared link layers, while section 2.6 discusses the design challenges of the higher layers of WPAN. Section 2.7 presents the advantages and disadvantages of mathematical modelling and computer simulation techniques that evaluate the performance of communication systems. Section 2.8 presents the performance metrics used to evaluate the system performance. Finally, the related researches carried out by other researchers are reviewed in section 2.9.

2.2 An overview of wireless communications

Ubiquitous access to information characterizes whole new kinds of information systems in the 21st Century. This is enabled by rapidly emerging wireless communications systems, based on radio and infrared transmissions, and utilizing such technologies as wireless area networks, personal communications systems, cellular telephony, etc. Wireless communications are transforming the way people use computers and other personal electronic devices at work, home, or when travelling.

Wireless networks are divided into four main categories: wireless personal area networks (WPAN's), wireless local area networks (WLAN's), wireless metro area networks (WMAN's) and wireless wide area networks (WWAN's). Based on the transmission range, wireless networks can be divided into two broad segments: short-range and long-range. Short-range wireless networks are confined to limited areas, i.e. buildings, manufacturing plants or family houses. These networks are often operated over either the infrared spectrum or the unlicensed spectrum reserved for industrial, scientific and medical (ISM) use. For short-range wireless networks, the data rates are often relatively low [18]. WPAN's and WLAN's belong to this category. Long-range networks continue where WPAN's and WLAN's end. Connectivity is typically provided by companies that sell the wireless connectivity as a service, i.e. cell phone services, cellular data services. These networks often cover large geographical areas such as a metropolitan area, a state or province, or even an entire country. The WMAN's and WWAN's are in this category. As given in [33], the transmission ranges and functionalities of different

wireless technologies are illustrated in Figure 2.1.

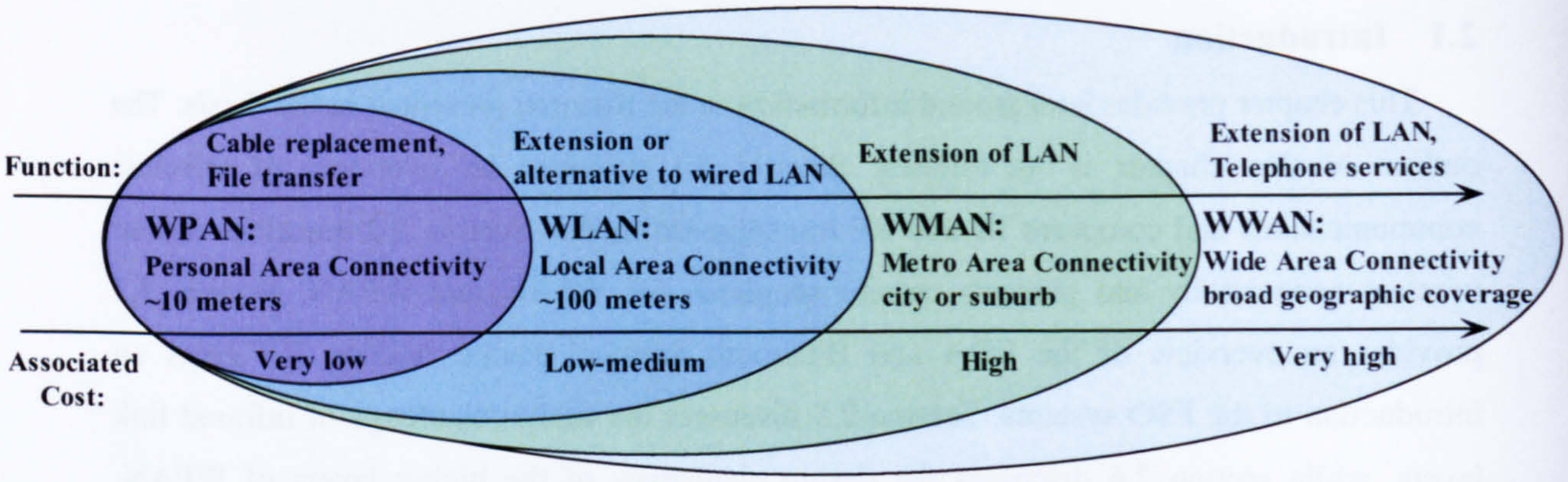


Figure 2.1 Wireless technologies categorized by range

Wireless communications can be carried out through two different media: infrared (IR) and radio (RF). Both IR and RF have certain strengths and weaknesses which make them more suitable for certain wireless environments and applications. Table 2-1 compares the strengths and weaknesses of IR and RF media for wireless communications [57][59].

	Strength	Weakness
IR	1. Very high data rates possible 2. Low power consumption 3. No regulation on infrared spectrum 3. Low cost 4. Inherent high security as LOS path is required 5. Good immunity to electrical interference 6. No multipath fading	1. Low mobility (LOS path required) 2. Power output limited by eye safety regulation 3. Susceptible to noise from ambient light sources 4. Half duplex for IrDA (indoor infrared) links 5. Multipath dispersion can limit data rate
RF	1. Long transmission range 2. High level of mobility 3. No LOS path needed 4. Full duplex communications capability 5. Frequency division multiplexing, and spread spectrum modulation techniques possible	1. Frequency spectrum use is highly regulated 2. Frequency spectrum has high level of congestion 3. Susceptible to electrical interference 4. Security concerns due to signal leakage 5. Low maximum data rates

Table 2-1 Infrared versus radio

Infrared—The IR spectrum offers virtually unlimited bandwidth capable of accommodating high data rates [19][20]. As a line of sight (LOS) path is required by the directed IR systems (both IrDA and FSO links), good security is inherently provided and the security issue is much simplified. There is also no operation license requirement for the infrared systems.

After years of development and standardisation, indoor infrared components are cheap, easy to build and universally compatible. The free space optics (FSO) systems (outdoor infrared) exploit the high available infrared bandwidth and provide solutions to the 'last mile' bottleneck issue with very high capacity links [21]-[23]. The power output of the IR transmitters is limited by the eye safety regulations [24], thus, the IR systems have a relatively short transmission range. For diffuse IR links, multipath dispersion from wall and ceiling reflections can limit the maximum data rate. IR wireless is suitable for low cost, low power, short-range and low mobility systems with high data rates and inherent high security [9][19].

Radio—RF has a long transmission range and can be received omnidirectionally. RF communication systems can have powerful transmitters with very sensitive receivers, thus providing a large transmission range with signal radiating in all directions and passing through walls and objects. Additionally, it is easy to design full-duplex communications for RF systems. Many different techniques can be used for channel multiplexing including frequency division multiplexing (FDM), time division multiplexing (TDM) and code division multiplexing (CDM). However, RF systems are restricted by the highly congested and regulated RF spectrum. Radio transmissions are regulated worldwide and require government licensing. The ISM radio bands are an exception to the licensing rule. United States and Canada allocate ISM bands at 902-928 MHz, 2,400-2,484 MHz and 5,725-5,850 MHz [4][16]. They are called the 900 MHz, the 2.4 GHz and the 5 GHz ISM bands respectively. The 2.4 GHz ISM band is allocated worldwide but some countries allocate slightly different ISM bands of 900 MHz and 5 GHz [17]. Higher frequency ISM bands can achieve higher data rates. They however require more elaborate electronics and are subject to higher interference from other radio sources [108][109][110]. To avoid licensing, indoor wireless communications often utilize the ISM bands. RF is preferred for applications where high user mobility over long ranges in varying environments is required.

2.3 Indoor wireless communications: WPAN and WLAN

WPAN and WLAN are the two network technologies for indoor wireless communications. WPAN's are small wireless networks that typically extend to 10 metres or less. Because of their limited range, WPAN's are used mainly as cable replacement technologies for data synchronization and connectivity for devices that are close to each other.

In contrast to WPAN's, WLAN's provide wireless connectivity over a local area of approximately 100 metres between the access point and the associated clients. Today's WLAN's are based on the IEEE 802.11 standards and are referred to as Wi-Fi networks. This section provides an overview of the indoor wireless network technologies. Two basic types of

wireless topologies are introduced.

2.3.1 Topologies

There are two types of indoor wireless communication topology: ad hoc networks and infrastructure networks. The most basic wireless topology is consisted of a set of stations, which recognize each other and are connected via the wireless media in a peer-to-peer fashion. This form of network topology is referred to as an ad hoc network as shown in Figure 2.2. In an ad hoc network, the mobile stations communicate directly with each other. Not every mobile station is able to communicate directly with every other station due to the range limitations.

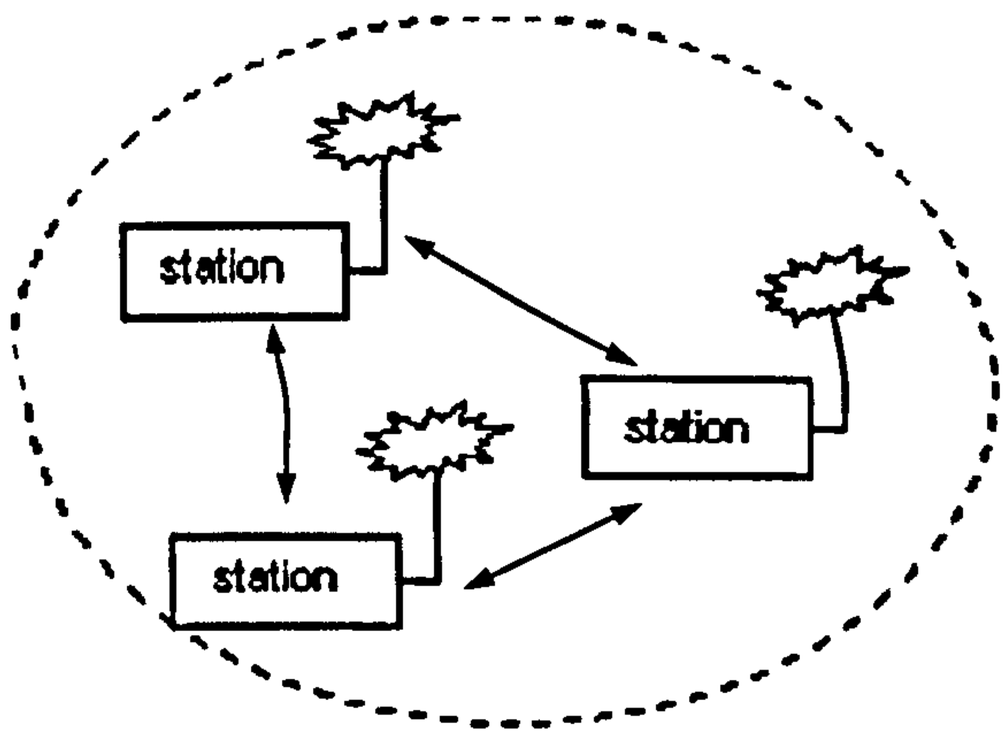


Figure 2.2 Ad hoc network

The infrastructure network topology is shown in Figure 2.3. It is a basic ad hoc environment with a component called an access point. The access point provides a local relay function for the stations. All stations have to communicate through the access point and do not communicate directly since all frames are relayed between stations by the access point. The access point may also provide connection to other networks. With the help of the access point, it is possible to provide a wider range of services (e.g. wireless Internet access).

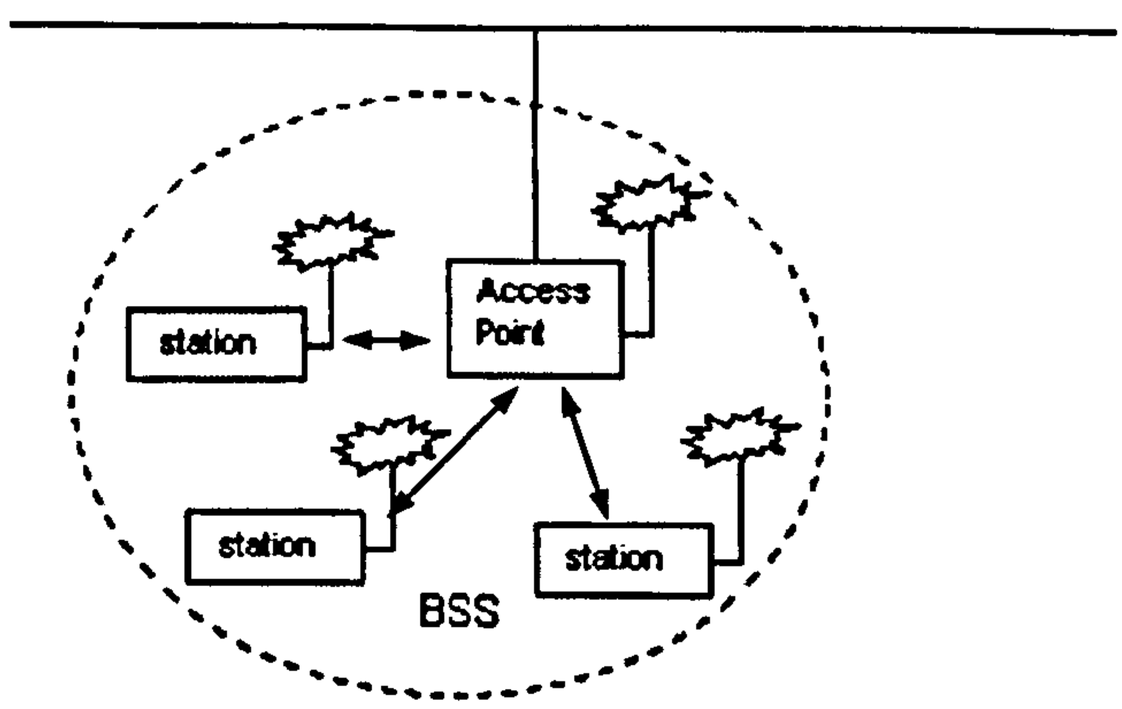


Figure 2.3 Infrastructure network

2.3.2 WPAN technologies

IrDA and Bluetooth technologies are the prevalent WPAN technologies today (Figure 2.4). IrDA technology utilizes the license free infrared spectrum to provide directed wireless connectivity up to 2 metres. The IrDA standards support half-duplex links with a maximum data rate of 16 Mbit/s [37]. A new IrDA physical layer standard that supports data rates up to 100 Mbit/s is currently under development. Bluetooth technology operates in the 2.4 GHz unlicensed frequency band and has a typical transmission radius of 10 metres [38]. Currently, Bluetooth products are based on the standard version 1.2 at a data rate of 1 Mbit/s. The 3 Mbit/s Bluetooth 2.0+ Enhanced Data Rate (EDR) standard has been proposed and products using the new standard are expected to appear on the market soon.

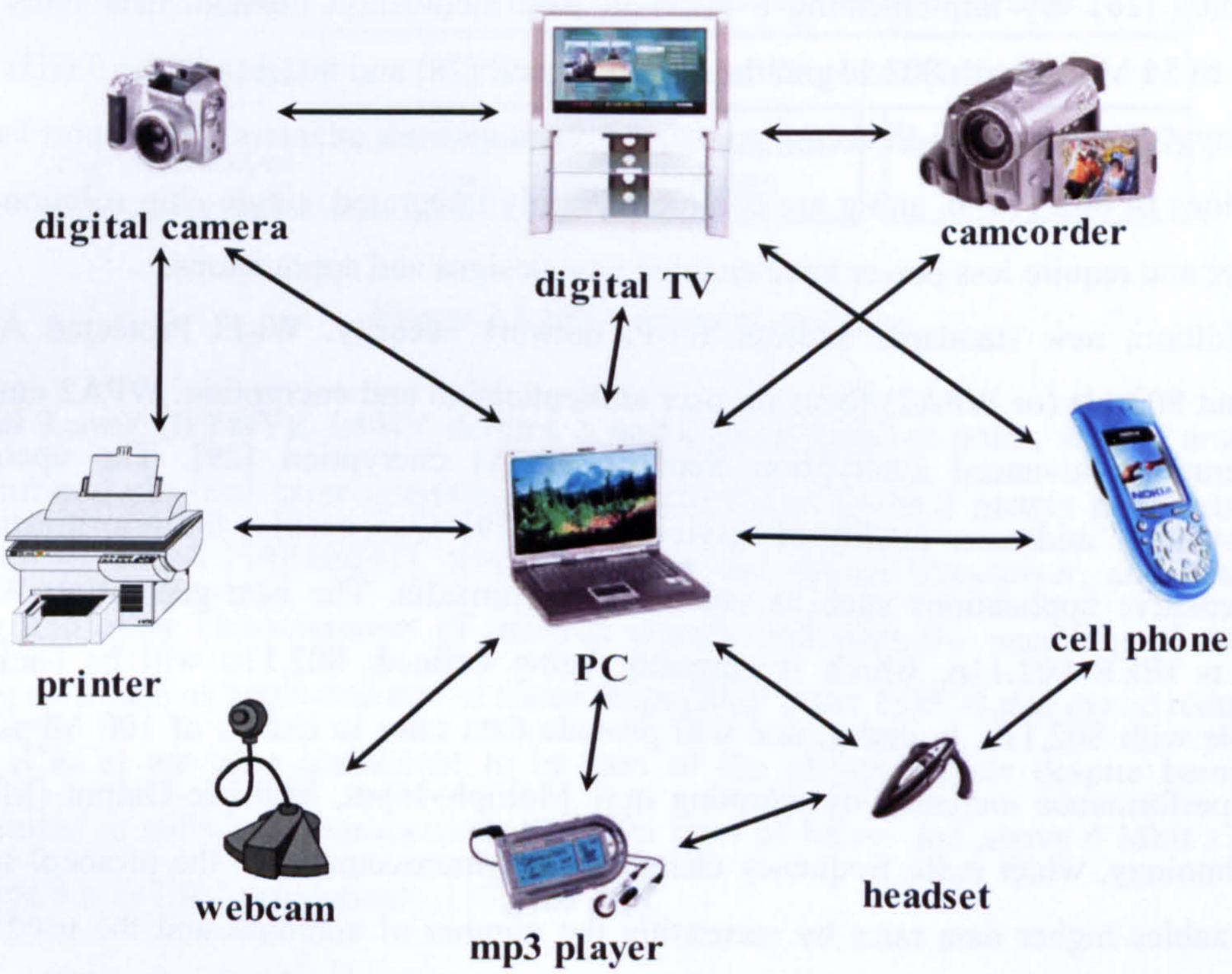


Figure 2.4 The typical WPAN applications. WPAN provides wireless connection between devices within a short distance (up to 10 metres).

In the future, WPAN applications that require higher data rates may adopt the emerging high bandwidth Ultrawideband (UWB) technology. UWB provides high bandwidth by transmitting at very low power across a broad frequency spectrum. The UWB physical interface specification, IEEE 802.15.3a, is under development by the IEEE, and a competing specification is under development by an industry working group called the multi-band Orthogonal Frequency Division Multiplexing (OFDM) Alliance (MBOA). Initial UWB products with data rates of 100-480 Mbps are anticipated in early 2006 [33]. Future versions

are expected to have data rates of up to 1 Gbit/s. However, although the U.S. Federal Communications Commission (FCC) has approved a large amount of spectrum for UWB in the U.S., there are regulatory and regional policy issues outside the U.S.

ZigBee (IEEE 802.15.4) [25] wireless technology also fits in the WPAN category. It is optimized for low-bandwidth niche applications such as instrumentation and home automation.

2.3.3 WLAN technologies

The IEEE 802.11 family of protocols, also called Wi-Fi, is the dominant WLAN technology being widely implemented into various WLAN products. IEEE 802.11b was the first commercially successful WLAN technology. It operates in the 2.4 GHz frequency band at 11 Mbit/s [26]. By implementing a different data modelation method, data rates were increased to 54 Mbit/s with 802.11g in the 2.4 GHz band [28] and 802.11a in the 5 GHz band [27]. Today, "dual-band" Wi-Fi access points and client network adapters that support various combinations of 802.11a, b, and g are common. Highly integrated, single-chip solutions that are smaller and require less power have enabled new designs and applications.

In addition, new standards address Wi-Fi network security. Wi-Fi Protected Access (WPA) and 802.11i (or WPA2) focus on user authentication and encryption. WPA2 employs next-generation Advanced Encryption Security (AES) encryption [29]. The upcoming 802.11e standard addresses quality of service (QoS) [29]. QoS enables the prioritization of latency-sensitive applications such as voice and multimedia. The next-generation WLAN standard is IEEE 802.11n, which is currently being defined. 802.11n will be backward compatible with 802.11a, b, and g, and will provide data rates in excess of 100 Mbps. The 802.11n performance increases by adopting new Multiple-Input, Multiple-Output (MIMO) radio technology, wider radio frequency channels, and improvements to the protocol stacks. MIMO enables higher data rates by increasing the number of antennas and the used radio channels in a wireless device. 802.11n is scheduled for IEEE ratification in mid-2006.

2.4 IrDA and Bluetooth standards

The following sections present background information on the IrDA and the Bluetooth protocol stacks.

2.4.1 IrDA protocol stacks

The IrDA protocol stacks are a layered set of protocols running on devices using short distance wireless indoor point-to-point infrared communications. By extending the stack

model presented in [57], the IrDA protocol stacks are illustrated in Figure 2.5. A brief description of the IrDA protocol stacks is as follows:

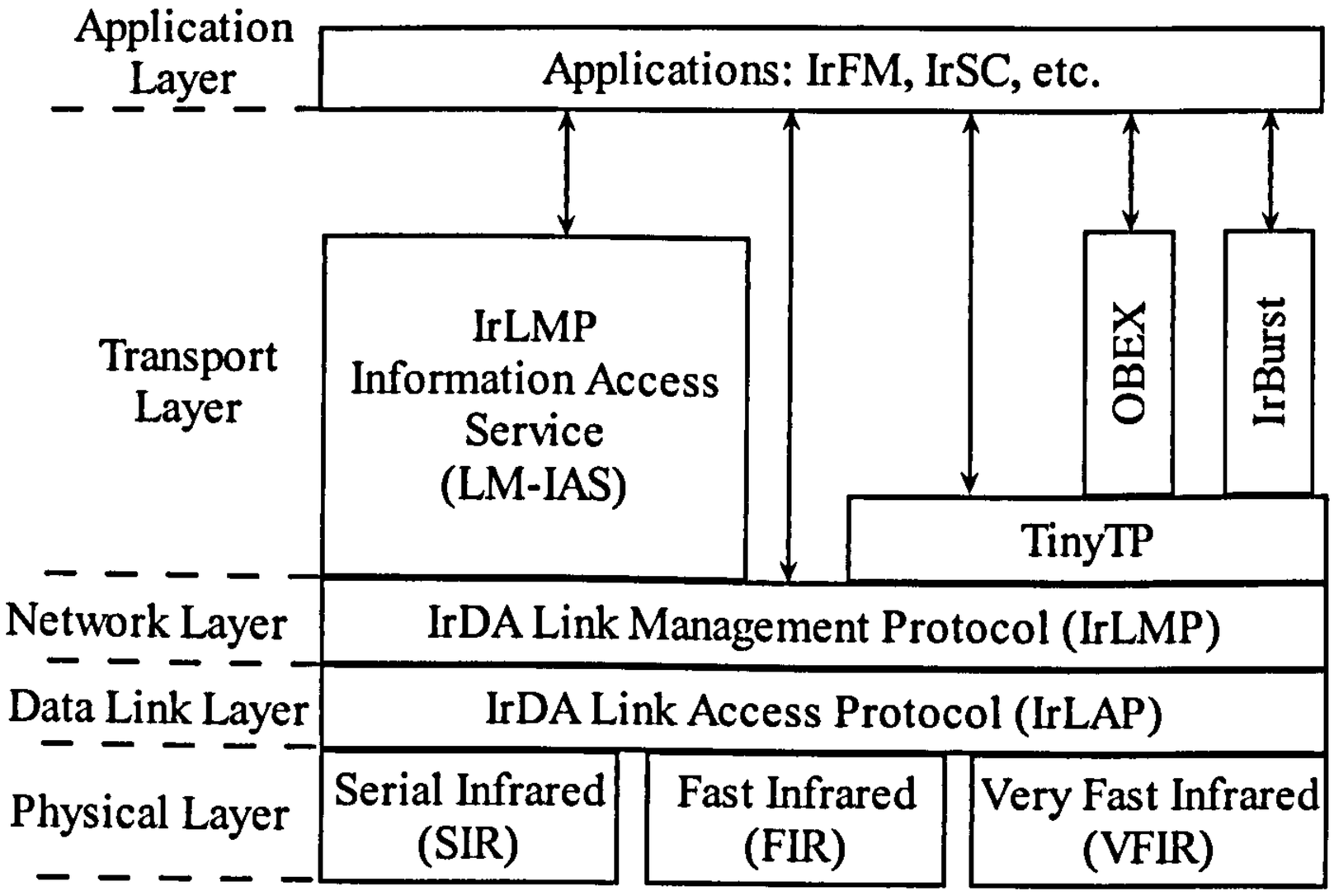


Figure 2.5 IrDA protocol stacks

Physical Layer (IrPHY): IrPHY defines a half-duplex, point-to-point, narrow angle (30° cone) infrared physical layer operating over a distance of up to 2 metres at speeds of 9.6 Kbit/s to 16 Mbit/s [39][40][41]. IrPHY includes the optical transceiver, and deals with shaping and other characteristics of infrared signals including the encoding of data bits. Framing data such as begin and end of frame flags (BOF's and EOF's) and cyclic redundancy checks (CRCs) are also considered to be part of the physical layer despite being often implemented in software. Transceivers with data rates of below and above 4 Mbit/s employ 16-bit and 32-bit CRC respectively, [39] and [40].

IrLAP (IrDA Link Access Protocol): IrLAP is the IrDA link layer protocol. It is based on High-Level Data Link Control (HDLC) and Synchronous Data Link Control (SDLC) with extensions for some unique characteristics of infrared communications [42]. By using mechanisms including error detection, retransmission and low-level flow control, IrLAP provides reliable data transfer.

IrLMP (IrDA Link Management Protocol): IrLMP provides support for multiple software applications or entities to operate independently and concurrently, sharing the single communication channel provided by IrLAP between the transceivers [43].

IAS (Information Access Service): IAS is normally considered as a part of IrLMP. It provides "yellow pages" of services on a device [43]. For instance, when a mobile phone wants to launch an IrFM application, it will first consult the IAS to obtain the IrFM information from the Point of Sale (POS).

TinyTP (Tiny Transport Protocol): TinyTP is a simpler version of TCP which serves as the transport layer for the stack [44]. It provides flow control on a per-IrLMP-connection basis to ensure end to end data transmission fidelity.

OBEX (Object Exchange): Running on top of TinyTP, OBEX is a session protocol designed to transmit data objects between resource-limited devices [45]. It has been adopted as the object exchange framework for wireless transports including IrDA and Bluetooth.

IrBurst (IrDA Burst protocol): IrBurst is an alternative session layer protocol dedicated for high speed large volume information transfer over IrDA links [47]. OBEX is designed for small object exchange, while IrBurst is optimized for sending large volumes of data.

IrFM (IrDA Financial Messaging): The IrFM Point & Pay Profile contains detailed consumer usage models, terminal and mobile client implementation guidelines, architectural definitions [46] for sending and receiving payment and transaction record information between Personal Trust Devices (PTD) and a financial terminal such as a POS. The IrFM standard defines the core protocol commands, as well as the services used in different payment scenarios.

IrSC (IrDA Simple Connection): Proposed by the members of IrDA, IrSC is a protocol dedicated for fast connection and instant data transmission over IrDA links. A typical application is to fast download pictures from digital cameras or mobile phones to TV set-top boxes or digital TV's, and display the pictures on TV screens.

2.4.2 Bluetooth protocol stacks

The Bluetooth Special Interest Group (SIG) developed and published the Bluetooth Specification [104]. By deriving from the Bluetooth standard [104], Figure 2.6 illustrates the Bluetooth protocol stacks. The Bluetooth wireless specification includes radio (physical) layer, link layer and application layer definitions for product developers.

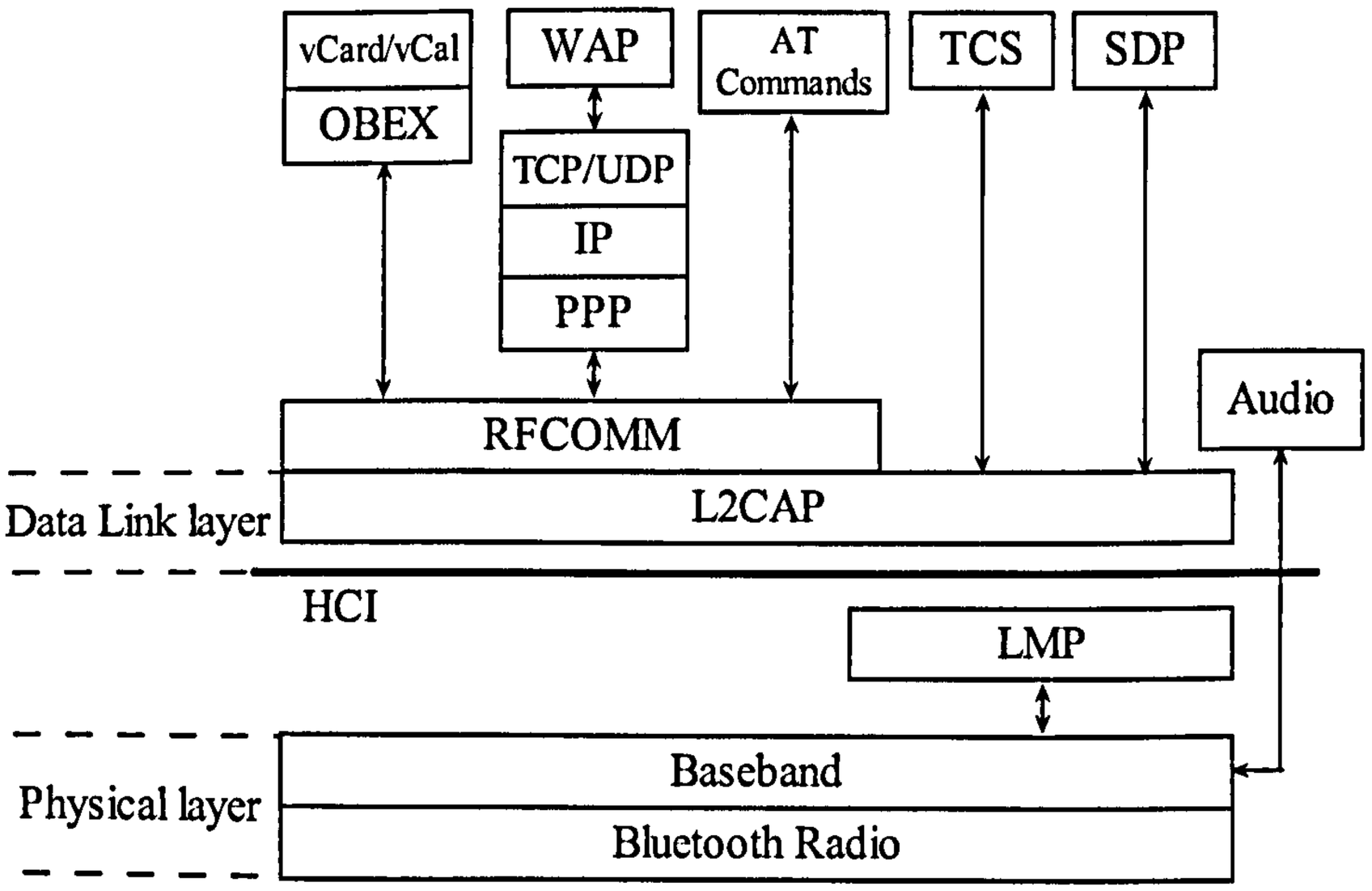


Figure 2.6 Bluetooth protocol stacks

Bluetooth Radio: By using spread spectrum technology, the Bluetooth Radio layer defines the requirements for a Bluetooth transceiver operating in the 2.4 GHz ISM band. The Bluetooth radio accomplishes spectrum spreading by frequency hopping in 79 hops displaced by 1 MHz, starting at 2.402 GHz and finishing at 2.480 GHz.

Baseband: The Baseband layer lies on top of the Bluetooth radio layer in the Bluetooth stacks. It manages physical channels and links apart from other services such as error correction, data whitening, hop selection and Bluetooth security. The Baseband protocol is implemented as a Link Controller, which works with the Link Manager for carrying out link level routines such as link connection and power control. The baseband also manages asynchronous and synchronous links, handles packets, as well as carrying out paging and inquiry to access the Bluetooth devices in the area. The baseband transceiver applies a time-division duplex (TDD) scheme.

LMP (Link Manager Protocol): LMP is used by the Link Managers for link set-up and control. LMP is responsible of managing connection states, enforcing fairness among slaves, power management and other management tasks.

HCI (Host Controller Interface): HCI provides a command interface to the Baseband Link Controller and Link Manager, and access to hardware status and control registers.

L2CAP (Logical Link Control and Adaptation Protocol): L2CAP supports higher level protocol multiplexing and the conveying of quality of service information. It also provides optional error retransmission, segmentation and reassembly, and per application flow control.

RFCOMM (Radio Frequency Communications protocol): RFCOMM provides emulation of serial ports over the L2CAP protocol. The protocol is based on the ETSI standard TS 07.10.

SDP (Service Discovery Protocol): SDP provides a means for applications to discover which services are provided by or available through a Bluetooth device. It also allows applications to determine the characteristics of those services available.

TCS (Telephony Control Protocol Specification): TCS provides telephony services.

AT (Attention) commands: AT commands are used to control mobile phones and modems in the multiple usage models. In Bluetooth, AT commands are based on ITU-T Recommendation V.250 [102] and ETS 300 916 (GSM 07.07) [103].

2.5 Free space optics

Free Space Optics (FSO), also known as outdoor infrared, is an optical technology and a simple concept involving the transmission of voice, video and data through the air using laser beams [49]. FSO represents one of the most promising approaches for addressing the emerging broadband access market.

FSO requires light, which can be focused by using either light emitting diodes (LED's) or laser diodes. The use of lasers is a simple concept similar to optical transmissions using fibre optic cables; the only difference is the medium. For FSO, the medium is air instead of fibre/glass. FSO offers many advantages when compared to fibre. It is a zero sunk costs solution. As opposed to fibre, FSO can be redeployed if a customer moves or cancels service. The installation of FSO is done in a fraction of the cost and time compared to that of fibre. Therefore, FSO links allow service providers to accelerate deployment of network connectivity as well as extending the reach of the high optical capacity to anyone who needs it. FSO delivers high bandwidth optically without the need to purchase expensive spectrum licenses, distinguishing itself clearly from the RF wireless technologies. Currently, FSO is capable of offering data rates up to 2.5 Gbit/s [21][22].

Depending on the type of the network which the FSO link belongs to, many different link layer protocols can be put on the FSO links including Ethernet data link or High-Level Data

Link Control (HDLC) protocols. One of the most popular link layer protocols used on FSO is HDLC [48], which has been widely implemented. The question is how well can HDLC perform at high speed considering the presence of bit errors and accounting for the propagation delay between the end users. In this thesis, the HDLC performance over FSO is to be studied by considering the physical characteristics of FSO links. The basics of the FSO systems are discussed in the following sections.

2.5.1 Principle of infrared transceivers

The principle of the FSO technology is relatively simple. It is based on connectivity between IR transceiver units, each consisting of an optical transceiver with a laser transmitter and a photon-sensitive receiver to provide full-duplex and bi-directional transmission capability, as shown in Figure 2.7. Each FSO unit uses a high-power optical source, plus a lens that focuses the transmitted light through the atmosphere to the receiving lens. The receiving lens focuses the light to a high-sensitivity receiver via optical fibre. Positive-Intrinsic-Negative (PIN) photodiodes or Avalanche Photodiodes (APD's) are commonly used in the receiver to produce an electrical current proportional to the incident optical power.

Optical wireless systems share several characteristics with fibre optics. Optical wireless solutions commonly use the same optical transmission wavelengths as fibre optics—850nm and 1550nm. Optical wireless and fibre systems use the same components such as lasers, receivers and amplifiers. Some systems include fibre connections inside the transmission link heads, to separate electronics and optics.

Similar to fibre optics, FSO systems also target the high bandwidth telecommunication market. While fibre optics can be used over longer distances, optical wireless targets shorter distances (up to a few kilometres) due to the variability of the terrestrial atmosphere as a transmission medium.

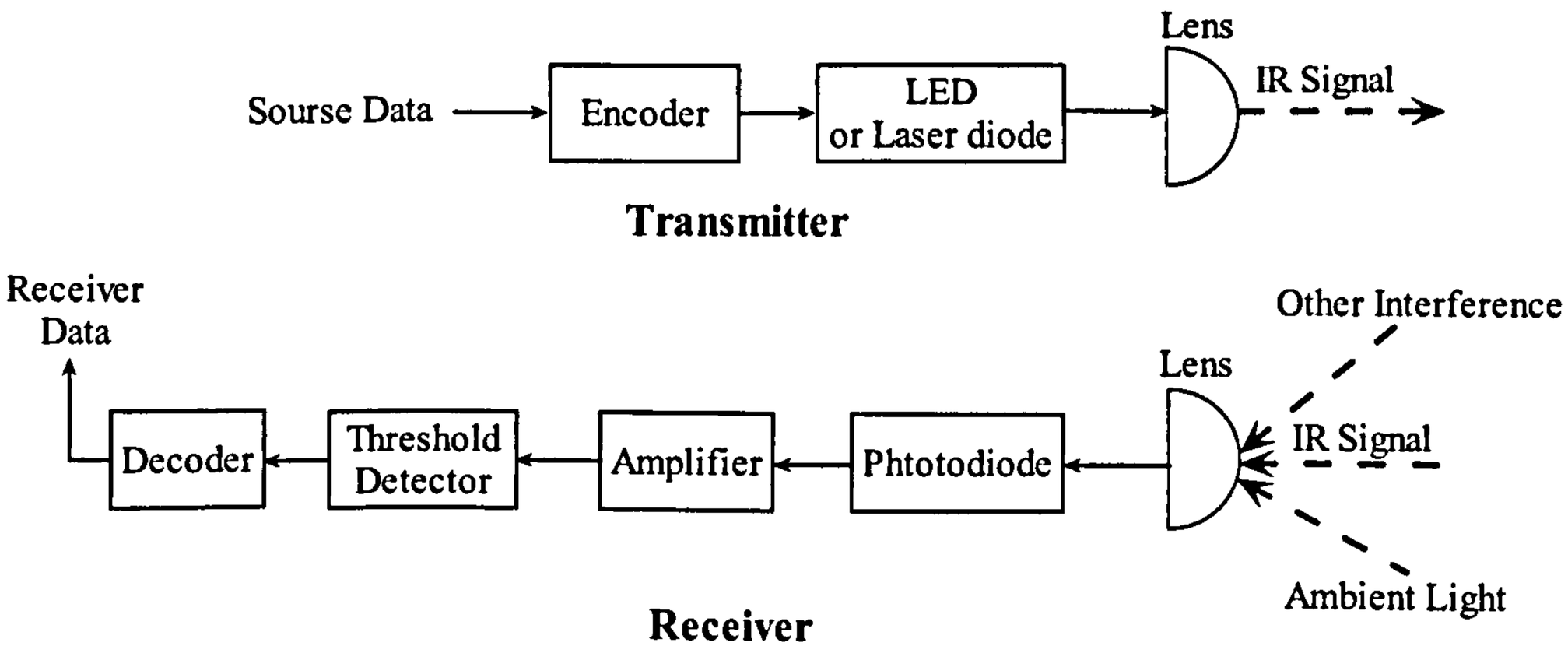


Figure 2.7 Basic wireless infrared link

2.5.2 Laser safety and classifications

FSO systems are typically designed to be eye-safe, which means that they pose no danger to people who might happen to encounter the laser beam. Laser eye safety is classified by the International Electrotechnical Commission (IEC), which is the international standards body for all fields of electrotechnology. While the IEC is an advisory agency, its guidelines are adopted by the regulatory agencies in most of the countries. A laser transmitter that is completely safe when viewed by the unaided eye is designated IEC Class 1.

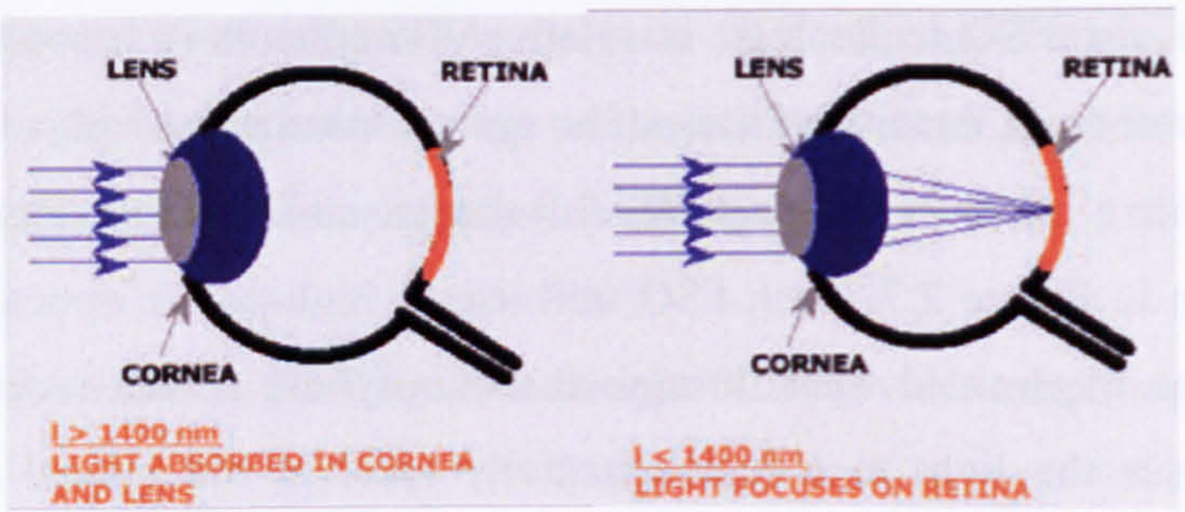


Figure 2.8 The light absorption of human eyes

The eye-safe limits vary with laser wavelength. The FSO hardware currently on the market can be classified into two categories – systems that operate at a wavelength near 800 nm and those that operate near 1550 nm [14]. Laser beams at 800 nm are near-infrared and therefore invisible, yet like visible wavelengths, the light passes through the cornea and lens and is focused on a tiny spot on the retina. This is illustrated in the diagram on the right side of Figure 2.8, which applies for visible and near-infrared wavelengths in the range of 400 to 1400 nm. The collimated light beam entering the eye in this retinal-hazard wavelength region is concentrated by a factor of 100,000 times when it strikes the retina. Thus, at 800 nm the retina could be permanently damaged by some commercially available optical-wireless products before the victim is aware that hazardous illumination has occurred. In contrast, the diagram on the left schematically shows that laser beams at wavelengths greater than 1400 nm are absorbed by the cornea and lens, and do not focus onto the retina. Because of these biophysical properties of the eye, wavelengths > 1400 nm are allowed approximately 50 times greater intensities than wavelengths near 800 nm [50].

2.5.3 Challenges of FSO at the physical layer

A number of factors can significantly degrade the FSO performance [10][13], such as various weather conditions, temporary blockage of the LOS path, etc. The major challenges for the design of the FSO physical layer are presented below:

Fog: Fog substantially attenuates visible radiation, and it has a similar effect on the near-

infrared wavelengths that are employed in laser communications. Similar to the case of rain attenuation with RF wireless, fog attenuation only partially impedes the transmission for optical wireless, because the optical link can be engineered in such a way that an acceptable signal power will be received for a large fraction of the time even in the presence of heavy fog. Laser communication systems can be enhanced to yield even greater availabilities by combining them with RF systems.

Physical Obstructions: Laser communication systems that employ large receiver optics will eliminate temporarily obstruction concerns from objects such as birds.

Aligning Stability: Aligning stability in commercial laser communication systems is achieved by two methods. The simpler, less costly method is to widen the beam divergence so that if either ends of the link moves, the receiver will still be within the beam. The second method is to employ a beam tracking system. While more costly, such systems allow for a tighter beam to be transmitted allowing for higher security and longer distance transmissions.

Scintillation: Performance of many laser communication systems is adversely affected by scintillation on bright sunny days. Through the use of a large aperture receiver, widely spaced transmitters, and finely tuned receiver filtering, downtime due to scintillation can be avoided.

2.6 Challenges in the design of the infrared link layers

The properties of the infrared medium make the design of wireless protocols more challenging than cable networks. Several issues in the protocol stack design have to be addressed differently if the infrared medium is utilized at the physical layer. The IR medium exhibits the following challenges:

Half-duplex operation: For an IrDA communication port, IR transmitter and receiver are installed next to each other to minimize equipment cost. When a station transmits an IR signal, a large fraction of the signal leaks into the reception circuit. Usually, the power of the transmitted signal is orders of magnitude higher than the power of the received signal. As a result, the leakage signal has higher power than the received signal, making remote signal detection very difficult while transmitting. Thus, it is impossible for a station to receive data when it transmits. The nature of the half-duplex operation degrades the performance of IrDA links.

Minimum turnaround time: Due to the leakage of the IR signal, the receiver circuitry needs a minimum Turn Around Time (TAT) to recover after the transmission ends. Thus, a transmitting station is able to receive data after a TAT time period when its transmission ends. As a result, all participating stations must wait a TAT delay after the transmission finishes before initiating a new frame transmission to ensure that all stations are able to receive the new frame. The effect of TAT delay can be significant for high speed links. TAT should be taken into account in the design of IrDA link layer protocols.

Full-duplex operation: In contrast to IrDA, FSO systems often provide full-duplex capability to have high link efficiency. However, depending on the type of link layer protocol used in the system, the advantage of having full-duplex capability may not be utilized effectively. This issue is highlighted when a link layer protocol with error detection/retransmission function (i.e. HDLC) is employed. When one station detects transmission errors in a frame, it should inform the sender (i.e. by sending a NACK) immediately to stop the sender transmitting any new data frames and retransmit the erroneous frame. However, the station may be currently engaged in sending other data, which can cause a long delay in sending the NACK. Additionally, other delays, including turnaround and propagation time, can further reduce the efficiency of full-duplex links.

Channel errors: In wired networks, the probability of errors is very small. A low bit error rate (BER) is expected and an immediate frame acknowledgment is meaningless. In contrast, IR wireless channels may have high BER, resulting in a much higher frame error probability. To cope with frame errors, IR and radio wireless link layer protocols may utilize an immediate acknowledgement (ACK) frame, which follows every data frame transmission. If the ACK frame is not received, the transmitter reschedules the data frame for retransmission. ACK frames may result in significant overhead, especially when followed by considerable TAT delays, as in IrDA systems. In order to minimize the ACK frame overhead, infrared wireless link layer protocols may choose to acknowledge a number of data frames using a single ACK frame. They may also employ smaller frame sizes to decrease the frame error probability [52]. Another alternative is the implementation of Forward Error Correcting (FEC) codes. Infrared link layer protocols should be efficiently designed to minimize the total delay of data frame retransmissions, ACK frames, frame overheads, TAT delays.

2.7 Challenges in the design of the WPAN higher layers

In order to maximize the system performance, designing efficient higher layers is as important as optimizing the physical and link layers. Many issues need to be addressed for the

WPAN higher layers:

Compatibility: In order to efficiently utilize the link bandwidth, the higher layers must be perfectly compatible with the lower layers. The design of high layers with necessary compatibility involves:

(a) Free and quick communication with the lower layers;

(b) Acquisition of all the information available on the lower layers;

(c) Understanding of the capability of the lower layers so that the higher layer will not feed more data and try to accomplish more tasks than the lower layer can cope;

(d) Ablility to shut down, pause or save current applications at anytime since the WPAN ad hoc connections could have link breakage without any warning.

Security: The transmission medium in a wired network can be physically secured leading to manageable access control of the network. Wireless networks, especially RF networks, are more difficult to secure since the transmission medium is open to anyone within the geographical range of a transmitter. Thus, they are prone to malicious interceptions. Wireless access should include encryption and authentication in order to accomplish data privacy. Efficient and simple-to-use security schemes must be incorporated in wireless network designs to minimize the chances of unauthorized access. While encryption of wireless traffic can be achieved, it is usually at the expense of increased cost and decreased performance. The vulnerability of the wireless links has been identified in several studies [53][54] and a number of solutions have been proposed [55][56].

Efficiency: In order to achieve high throughput efficiency, the overhead of higher layers should be kept at a minimum while ensuring that the overhead carries all the necessary information. Higher layers should also effectively manage the channel multiplication for various applications. With regard to the channel multiplication, separating the signal and data transmission into different logical channels could be an excellent design for large content transmission [47]. Additionally, the size of higher layer packets should be optimized according to the available buffer size and the maximum frame size of the link layer.

Connection speed: The transmission delay includes connection time and data transfer time. In reality, some applications of WPAN do not need to transmit large data packets and demand a very short connection time (i.e. IrFM applications). In such cases, fast connection establishment is likely to play a more important role than fast data transfer. In order to speed up the connection establishment, the traditional connection procedures of WPAN may be

simplified on specific occasions. For example, the fast IrDA connection procedure [42] can be used when one IrDA station is at the fixed location.

2.8 Performance modelling of communication systems

Data communication systems, including different protocol layers, can be very complex with numerous factors and system parameters affecting the performance of the whole system [57][58]. Modelling and analysis of data communications protocols is essential in determining the factors that affect the system performance and in optimizing parameters to maximise performance.

The benefits of modelling are:

- Performance evaluation of a particular aspect of a system can be made without physically implementing the real system
- Intuitive understanding of the system operation and the dominant factors that affect performance can be obtained
- Evaluation of different parameter values can be obtained in order to provide optimum performance under specific conditions
- Issues of protocol design that affect performance can be highlighted and possible protocol design improvements can be tested and evaluated
- Recommendations can be made to system designers for obtaining optimum system performance

There are two principal methods for performance modelling of communications systems: mathematical modelling and computer simulation [51].

Mathematical modelling: A mathematical model consists of one or more equations that express system performance, such as throughput, as a function of physical parameters, such as link distance or bit error rate, protocol parameters, such as frame size, window size, buffer size, and system parameters, such as number of devices. Mathematical techniques including probability theory, statistics, queuing theory and stochastic process modelling are often used to develop a mathematical model of a system or a protocol layer. Based on the mathematical model, analytical graphs can be produced to show how the system performance changes with the modifications of different system parameters. These graphs are very useful for protocol designers since analytical results can be easily reproduced once the mathematical model is developed. The benefits of using mathematical modelling are: (a) analytical formulae are developed to model the behaviour of a specific feature of a system (b) an intuitive understanding of the dominant factor and relationships that affect performance are obtained.

In order to develop a mathematical model, it is necessary to make assumptions and approximations obtain manageable mathematical derivations. Since computer simulation modelling accurately predicts system performance as explained in the next section, mathematical models are usually validated by comparing analytical results and simulation results.

Computer simulation modelling: Computer simulation requires the development of software programs that accurately mimic the behaviour of a communication system under different conditions. In simulation, one can artificially model any part of the network, such as access points, different protocol layers, amount of traffic, etc. The computer program emulates the behaviour of every station independently and produces very accurate results because it replicates the behaviour of the real system. Simulation models usually involve few or no constraints. The software model is used to monitor the performance results of throughput, delay, collisions, or any other performance metrics while changing different parameters. This enables designers to determine the results of various configuration settings. Computer simulations are generally event driven where an event is a time dependent occurrence such as a packet arrival or a timer expiration. Each event in simulation processes has a particular simulation time 'tag' association. This allows events of different devices in simultaneous processes to have the same simulation time although executed sequentially in the computer program. The main advantage of simulation models is that detailed information and output statistics about the performance can be obtained even for very complex communication systems with few or no constraints. The computer simulation processes can however take a considerable amount of computing time depending on the system complexity and the type of output statistics required. In addition, the output results may not give the same level of intuitive view of performance as mathematical models since the dominant factors affecting performance are difficult to determine.

The major simulation tools, including OPNET [34] and NS2 [35], do not provide simulation modules for the IrDA protocol stacks and only provide the simulation modules for Bluetooth Baseband through a software patch.

In this thesis, two sets of MATLAB simulation program are developed and presented for the IrDA and the Bluetooth protocol stacks respectively. The dedicated simulation programs are separated into different modules for each of the protocol stacks. The programs emulate the environment of the physical layers and the protocol stacks operating on top of them. The details of the programs are provided in Chapters 4 and 7.

2.9 Performance metrics

The adequate metrics to evaluate the performance of an information exchange system depend on the user applications as well as the characteristics of the traffic that the system is expected to carry [59]. The traffic presented to the system is typically called the offered load. If the offered load contains time insensitive data, such as file transfer, email and web browsing, the communication system should focus on maximizing the rate at which data can be sent through the network. If the offered load contains time sensitive data, such as encoded speech and video, the communication system should focus on minimizing the delay of delivering the data to the destination since significant delay variations for the time sensitive packets are often not acceptable. The following is a brief discussion of the widely accepted performance metrics [57][58][59], which are utilized in this work:

Throughput: Throughput is defined as the number of information bits that can be transmitted through the network per second (bit/s). For the time insensitive data, network designers as well as implementers aim to maximize system throughput in order to achieve a better performance. Delays in delivering specific data are however of secondary importance. Throughput usually expresses the performance of a particular information exchange system and is also referred to as utilization. It specifies the actual performance of the system. Throughput is typically compared to the data rate to express the performance degradation introduced by different communication protocol behaviours, such as packet headers, retransmission delays and transmission errors.

This work examines the performance of the WPAN communication systems at different layers by evaluating the throughput efficiency, also called normalized throughput, which expresses the percentige of the total time that the system is utilized to deliver the offered load. For instance, if the average packet size is l bits, the average time to transfer a single packet is T seconds, and C bit/s is the data rate of the channel, then the throughput efficiency is given by $\frac{l}{TC}$.

Delay: The delay of a system specifies the time needed for information data to travel from the source to the destination station. Furthermore, the average packet delay is defined as the average time spent by a packet from the instant the packet enters the transmission buffer until the time it is correctly received. Users are particularly interested in the delay which the system delivers their information data to the destination. Delay is more important where time sensitive data is involved. Types of delays in communication systems are [60]:

- **Access delay** arises when a transmitted packet is not correctly received at the destination due to a packet collision, or when the shared wireless medium is currently used by other

stations or applications. Packet collisions take place when several stations access the shared medium. Since the transmission is often controlled by the master for WPAN technologies, the probability of having packet collisions is very small. In reality, waiting for other stations or applications before accessing the shared medium is the major access delay and it is studied in Chapter 7 for Bluetooth protocols.

- **Retransmission delay** is due to packet retransmissions that are triggered by transmission errors. A number of protocols carry out packet retransmission if the transmitted packet is not correctly received at the destination. Transmission errors are more likely when a wireless medium is utilized by other sources and may significantly degrade the system performance. This work considers retransmission delays due to transmission errors for the IrDA and the Bluetooth protocols in Chapters 3, 4, 5 and 7.
- **Propagation delay** is the time needed for the signal to travel between the source and the destination stations. In Chapter 3, propagation delay is included in the link layer mathematical modelling for the IrDA and the FSO high speed links.
- **Queuing delay** often occurs in the packet switching WAN's. When a packet reaches a packet switching device, it may have to wait on a queue if more packets wait for the intended destination. Queuing delay accounts for the time a packet spends on a queue in a packet switching device. Since the thesis only concerns simple network topologies and the queuing delay is very short in such networks, this work therefore does not consider queuing delays.

Robustness against channel transmission errors: The wireless network is time-varying and error-prone compared to the wired network. Ambient noise can significantly degrade performance and make the link between two stations unusable for short periods of time. It is important to have a protocol that is robust against transmission errors [57][59].

Additional metrics can be used to evaluate the performance of wireless communication protocols. They are however out of the scope of the work presented here. These are:

Fairness: A communication system is fair if it does not exhibit preference to any single station when multiple stations are trying to access the channel. This results in fair sharing of the bandwidth. This definition can be biased when traffic with different priorities is handled. When multimedia traffic is supported, fairness is defined as being able to distribute bandwidth in proportion to the time allocation of each station.

Power Consumption: Most portable devices have limited battery power, hence, it is important to conserve power and provide some power saving features. For a large scale ad

hoc network (i.e. ZigBee network), it is also important to ensure the fairness of power consumption for each station.

2.10 Related research

2.10.1 Research in infrared wireless system

Numerous studies have been carried out to address different problems of IR systems, especially the design of physical and link layers. The major challenges of designing the IR physical layers are different for the indoor and the outdoor systems. In many indoor infrared transmission environments, there often exists intense ambient infrared noise, arising from sunlight, incandescent lighting and fluorescent lighting. Fluorescent lighting, especially from the fluorescent electronic ballasts [61][62], is considered to be the biggest noise source among the three noise sources. The rapid fluctuations of the optical power produced by fluorescent lights manifest itself as interference [63]. This interference has different intensity and bandwidth characteristics depending on the type of lamp that produces it [64]. However, for the outdoor IR systems (FSO), the frequent change of atmosphere is the biggest challenge for the design of the FSO physical layer. A FSO signal propagating through the Earth's atmosphere is subject to attenuation and distortion due to absorption by aerosols, scattering by particles such as fog and atmospheric turbulence. All three conditions can significantly reduce the amount of light energy that is received [83][84][85]. However, the effects of various noise on the IR systems, either from other light sources or from the change of atmosphere, can be alleviated by choosing the appropriate modulation scheme, increasing the signal strength, or applying the forward error correction scheme [61][62][64][83][84][85]. Additionally, other issues of designing the IR physical layers including different modulation schemes for high speed links [88][89][90] and eye safety [24][86][87] have also been addressed.

Besides physical layers, the studies of link layer performance improvements and evaluations have also been undertaken to address different infrared link layer issues. The performance of infrared point-to-point and point-to-multipoint connectivity can be measured by the link utilization. IrDA link layer IrLAP is based on the widely used HDLC protocol and utilizes the Go Back N (GBN) Automatic Error Request (ARQ) scheme. The performance of the GBN ARQ scheme is studied in [66][67]. In [68][69], Bux derived an analytical model that evaluates HDLC performance using a concept named window transmission time (WTT) that derives the time to transmit a data 'window'. Subsequently, Boucouvalas [70][73][82], Barker [76][77], Vitsas [71][72] and Ozugur [74][75] extended the work by conducting their analysis on the specific area of IrDA links. Based on the concept of WTT, an analytical model for the IrLAP performance was developed in [76][77]. The WTT IrLAP model was employed to study the effect of minimum turn around time on the link throughput in [78]. An IrLAP

simulator was developed by using the OPNET simulation tool. It was used to validate the WTT IrLAP analytical model derived in [79]. A C++ IrLAP simulator was developed in [80]. This simulator was employed to study the effect of minimum turn around time when the link data rate increases to 4 Mbit/s and the IrLAP window size increases to 7 frames.

The performance improvement of replacing the IrLAP GBN ARQ scheme with a Selective Reject (SREJ) ARQ scheme is presented in [81]. This paper concludes that the GBN ARQ scheme is good enough for IrLAP due to the significant turn around delays arising from switching the IR transceivers between the transmitter and the receiver. The effects of extending window size to 127 frames for the 16 Mbit/s data rate is studied in [82] using the WTT IrLAP analytical model. This work questioned the effectiveness of increasing window size to 127 frames and advised the implementation of optimum IrLAP window and frame size for the high bit error rate (BER) environments. Based on [78][82], by taking the first derivative of the throughput equation, Vitsas derived equations which allow easy calculation of the optimum values for the IrLAP window and frame size parameters in [71][72]. Results indicated that significant improvement on the link throughput can be achieved if the optimum values are implemented in the high BER environments. The optimum equations for the IrLAP window and frame size were also confirmed by simulations.

The IrDA link layer research carried out so far focused on relatively low link data rates and it assumed the propagation delay is small enough to be ignored. However, IrDA links are expected to evolve to higher data rates since the infrared spectrum offers virtually unlimited bandwidth. In fact, IrDA has already set up a SIG dedicated to drafting a new physical layer supporting the link data rate as high as 100 Mbit/s. Thus, it is worth examining how well IrLAP performs at higher data rates for backward compatibility and also what the optimum link layer parameters for any BER should be. For high speed IrDA links, the important IrLAP parameters expected to affect the system performance are the minimum turnaround time, the average bit error rate and the propagation time. This thesis extends the WTT mathematical model of IrLAP presented in [78] by including the effect of link propagation time, and examines the performance of IrLAP operating at Gbit/s data rates.

Although many studies have been conducted on the physical layer of FSO, so far no systematic evaluation has been carried out for the full-duplex FSO links at the link layer level. HDLC is a widely implemented link layer protocol, which operates on top of the physical layer of FSO links. Bux provided an analytical model that evaluates HDLC performance in [68][69]. However, the study was carried out by assuming wired networks are the transmission medium. How well can HDLC perform for the high speed FSO links? In this work, based on the mathematical model of IrLAP, the special characteristics of the FSO links (i.e. full-duplex, long distance) are added to the model and the performance of HDLC over FSO is investigated. Subsequently, the HDLC parameters are optimized for the FSO links. A

real time adaptation algorithm is proposed to apply the optimum parameters to infrared systems according to the estimated instant error rate.

In contrast to the studies devoted to the IR physical and link layers, only a limited number of studies on the higher layers have been carried out. In [91], Tourrilhes presented a novel approach to the use of TCP/IP applications over IrDA links. The paper proposed a new scheme that enables the use of the IrDA communication stacks by various applications (i.e. Internet browsers) in a transparent fashion with minimum overhead. In [92], a study of OBEX (IrDA session layer) was carried out to search for a suitable OBEX packet size to optimize the OBEX performance. However, the paper focused on the OBEX protocol by assuming all the other IrDA layers add only overheads to the OBEX packet. In fact, the effects of other layers can be significant and the behaviour of other layers should be considered in the analysis. In order to improve the performance of the whole system, it is important to examine the interaction between the lower and higher layers. Additionally, most of the IrDA system evaluations focus on link layer performance by assuming that data packets of infinite size are always ready to transmit and that there is a single application operating [73][76][77][82]. In reality, higher layers offer only finite size packets to lower layers and may manage multiple applications simultaneously. For instance, OBEX operates in a similar way to a 'stop and wait' protocol and has a finite maximum packet size [45]. TinyTP (IrDA transport layer) allows multiple applications to operate the IrDA link concurrently [44]. In this thesis, the IrDA higher layers (transport, session and application) are investigated. Protocol parameters and hardware selection guidelines are proposed to maximize the overall system performance while also keeping the hardware requirement to a minimum.

2.10.2 Research in Bluetooth system

Soon after the release of the Bluetooth specification, Bluetooth attracted a high level of attention from both manufacturers and researchers. In recent years, a large number of digital devices have had been equipped with Bluetooth chipsets for wireless communication needs in an ad hoc fashion [104] and [106]. Motivated by the demands of 'hand-free' and 'cable-free' conversation from cell phone users, the first major Bluetooth application was the short range wireless headset. After the successful introduction and development of the Bluetooth headset (voice application), applications for information exchange between Bluetooth devices (data application) have been gradually accepted and used. These applications are considered to be a complement to the predominant technology—IrDA. Nevertheless, as a relatively new technology, Bluetooth faces many challenges and problems. Numerous studies have been carried out to address various Bluetooth issues including radio frequency (RF) interference, security, Baseband packet size adaptation, Baseband throughput analysis and transmission error evaluation.

Since WLAN utilizes radio frequency technology and also targets short range wireless communication, Bluetooth is often compared to WLAN technology and studied in the context of the WLAN and Bluetooth network coexistence. The most widely implemented version of WLAN – IEEE 802.11b operates in the same frequency band (2.4GHz) as Bluetooth. IEEE 802.11b uses Direct Sequence spread spectrum technique as the physical layer, while Bluetooth employs Frequency Hopping. Although 802.11b and Bluetooth use different spread spectrum techniques, the RF interference is significant in the heavily loaded coexistence environment [107][108][109][110]. Despite critiques of the relative low capacity compared to WLAN, Bluetooth has its own unique applications and market due to the low chipset price and power consumption. Bluetooth is now widely recognised and has been recently standardized by the IEEE organization with the name of IEEE 802.15.1 WPAN.

Because Bluetooth uses RF technology and operates in the unlicensed ISM frequency band, Bluetooth communication is potentially vulnerable to interception and thus needs extra attention to ensure security. Rise [111] claims that all computer attacks are a combination of one or more of the following improper conditions: validation, exposure, randomness, and deallocation. Rise's resulting taxonomy is the Validation Exposure Randomness Deallocation Improper Conditions Taxonomy (VERDICT). In [112] and [113], the authors analysed the security scheme used by the Bluetooth system and demonstrated the security vulnerabilities of Bluetooth by using VERDICT. Additionally, a Bluetooth configuration management was proposed to mitigate the vulnerabilities of the Bluetooth piconets.

Two types of logical transports are supported by the Bluetooth Baseband layer: asynchronous connectionless (ACL) and synchronous connection oriented (SCO). The Baseband uses different packets to transmit information for different types of transports. Different Baseband packet types have different payload lengths, error resistance abilities and use different numbers of time slots, as presented in [114][115]. A real time packet adaptation according to the instant error rate can improve the link throughput considerably [116].

A few Baseband analytical models are presented in [117][118][119][120][121][122]. Valenti, in [117], derived an analytical expression for the Baseband throughput as a function of channel error rate for the 6 different types of Baseband packets. In [118] and [119], the authors employ a signal capture model to derive the Baseband throughput. The throughput is evaluated by not only considering the transmission error rate but also including the effect of different Bluetooth network topologies. Zurbes examined the Bluetooth throughput performance in the existence of a large number of Bluetooth networks—piconets in [120].

In [123][124][125][126], additional Bluetooth issues are investigated. The power fairness problem for Bluetooth piconets is addressed in [123]. In [124] and [125], the performance of the transport protocols including UDP and TCP over Bluetooth piconets is evaluated. To

solve the disconnect issue when the Bluetooth devices move between piconets, a handoff scheme is presented in [126]. These issues are however out of scope of this thesis.

The situation of the research conducted on Bluetooth is similar to that for IrDA. There are only limited studies of the higher layers of Bluetooth. A crucial issue is to examine the important interaction between the lower layer (Baseband) and the higher layer (L2CAP). L2CAP protocol is layered over the Baseband [104]. L2CAP bridges the data applications and the Baseband by providing services including channel multiplexing, error retransmission, packet segmentation and reassembly. In this thesis, the Bluetooth L2CAP layer is examined in detail. The throughput equation of the L2CAP layer is to be derived by considering the presence of bit error.

Since short transmission delay is crucial to many applications (i.e. video, audio), it is important to examine the packet delay for the Bluetooth networks. In [129][130][131][132][133], authors studied Baseband ACL packet delay by applying different traffic scheduling algorithms to the Bluetooth network. These papers addressed the problem of designing efficient and fair polling schemes that would achieve very low end-to-end packet delay. A number of polling schemes was proposed and their performance was compared. However, in these studies, ACL applications are considered as the only network activity. In practice, a Bluetooth network may be used by other applications (i.e. headphone) and may support other activities (i.e. sending beacon trains). In Chapter 7, a more realistic and comprehensive analysis is given for the Baseband ACL packet and L2CAP Packet Data Unit (PDU) delay.

2.11 Summary of the Chapter

This chapter presents an overview for various wireless technologies. Two major WPAN technologies, IrDA and Bluetooth, are discussed in detail. The ISO OSI reference model and the protocol stacks of IrDA and Bluetooth are briefly introduced. An introduction to the FSO technologies is also provided.

This chapter also provides a methodology for the research work carried out in the thesis. The challenges and unsolved problems of designing the infrared link layers and the higher layers of WPAN are discussed. In order to evaluate the performance of a communication system, two approaches can be used: mathematical modelling and computer simulation. The advantages and disadvantages of both approaches are compared. The performance metrics, which are used in the analysis to evaluate the system performance, are presented. Finally, the chapter reviews the relevant body of literatures that are carried out in the area of infrared and Bluetooth systems.

Chapter 3. Enhancing Infrared Data Link Layer by Including the Effect of Propagation Time

3.1 Introduction

The performance of the IrDA links is affected by a number of factors: the half-duplex operation, the time required to reverse link direction, transmission errors introduced by the wireless medium and the propagation delay especially when high speed data rates are implemented. In contrast to the IrDA links, the FSO links often provide full-duplex capability. Depending on the type of link layer protocol used in the FSO system, the advantage of having full-duplex capability may not however be utilized effectively. This issue is highlighted when a link layer protocol with an error detection/retransmission function (i.e. HDLC) is employed. Additionally, the FSO links are also affected by transmission errors and, more significantly, by the propagation delay since the link distance of a FSO system is much longer than that of IrDA.

IrLAP drives the IrDA hardware and implements a Go-Back-N retransmission scheme to cope with transmission errors at the link layer. For the FSO systems, different link protocols can be implemented depending on the network configuration. One of the most popular link layer protocols used on FSO is HDLC, which has been widely implemented in products. This Chapter addresses different design issues on infrared link layers for efficient operation. The IrLAP protocol over the IrDA links and the HDCL protocol over the FSO links are studied and optimized respectively.

The outline of this Chapter is as follows: Section 3.1 describes the IrLAP layer, the parameters it negotiates for efficient link operation and the IrLAP frame structure. In section 3.2, a mathematical model that evaluates the IrLAP throughput by calculating the average window transmission time is developed. Section 3.3 derives and validates the optimization equations for IrLAP window and frame size to maximize the throughput of the IrDA link layer. Section 3.4 presents a number of result graphs to evaluate the IrLAP performance at the Gbit/s data rates. Section 3.5 describes the properties of the Asynchronous Response Mode of HDLC as the FSO link layer. The throughput formula of the FSO link layer is derived in section 3.6. In section 3.7, an optimization and evaluation study of the FSO link layer is carried out. Finally, section 3.8 proposes an adaptive system for the real time adaptation of the optimized link parameters.

3.2 IrLAP layer

The IrLAP design is based on the High level Data Link Control (HDLC) and Simple Data Link Control (SDLC) protocols [11]. The functions of IrLAP include device discovery, link establishment, data exchange and error recovery [42]. IrLAP stations operate in two modes: in the Normal Disconnect Mode (NDM) during the contention state and in the Normal Response Mode (NRM) during the connection state. During the contention state, a station advertises its existence along with the physical and link layer parameters it supports during the information exchange procedure to all stations within its transmission range. IrLAP assigns primary and secondary roles in NDM. One of the participating stations is assigned the primary role and all the other stations are assigned the secondary roles. Any station may claim to become the primary station but only one wins the contention. In NRM mode (connection state), only the transmissions to or from the primary station are permitted. If a secondary station wishes to communicate with another secondary station, it does so through the primary station. In NDM mode, communicating stations determine the best connection capability that can be supported by both stations. The following parameters are negotiated during NDM mode:

a) Data rate (*C*). This parameter specifies the transmission rate of the IrDA stations.

b) Maximum turn around time (T_{max}). This parameter specifies the maximum time period a station can hold the transmission control. IrLAP specifies that T_{max}=500ms for data rates less than 115.2 Kbit/s but smaller values may be agreed for the 115.2 Kbit/s or higher data rates.

c) Frame size (l_{LAP}). This parameter specifies the maximum size of the data field of any received Information frame (I-frame). IrLAP supports a maximum data size value of 2048 bytes (16 Kbits).

d) Window size (W_{max}). This is the maximum number of unacknowledged I-frames a station can receive before it has to transmit an acknowledgement. The transmitting station may request an acknowledgement before the window size is reached. IrLAP specifies that W_{max} has an upper limit of 7 for the data rates of less than 4 Mbit/s and 127 for the data rates of 4 Mbit/s and 16 Mbit/s.

e) Minimum turn around time (t_{ta}). This is the minimum time required to reverse link direction.

Parameters (b), (c), (d) and (e) are negotiated and agreed independently for each station. However, both stations must agree and use the same data rate (parameter (a)).

Figure 3.1 presents the IrLAP frame structure. A frame consists of the START flag, the IrLAP packet, the frame check sequence (FCS) field and the STOP flag. The FCS field contains a 16-bit CRC for data rates lower than 4 Mbit/s and a 32-bit CRC for the 4 Mbit/s and higher data rates.

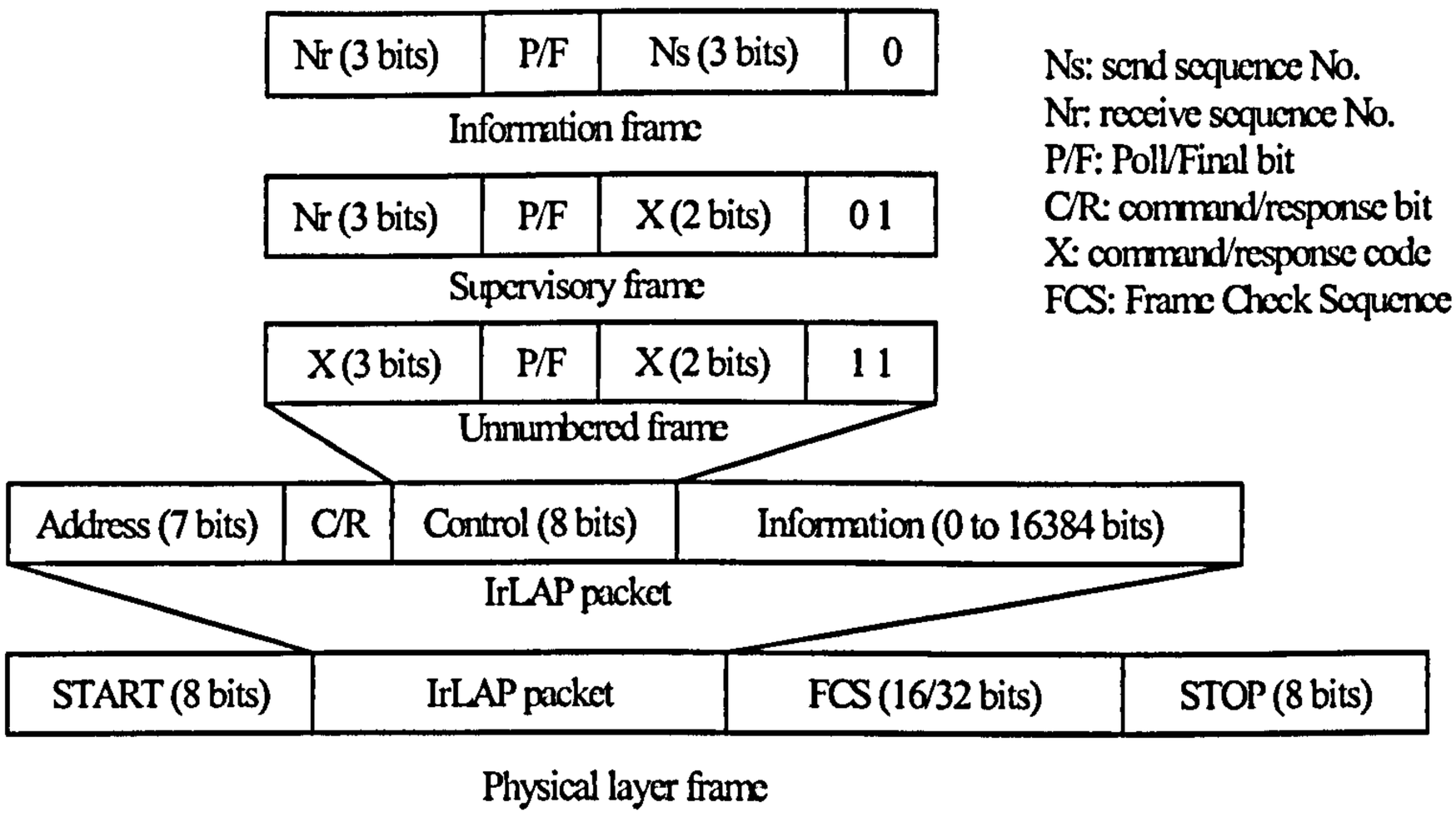


Figure 3.1 IrDA frame structure

IrLAP specification defines the following frame types:

a) Unnumbered frames (U-frames): U-frames are used to transmit the link management information. They are used to exchange connection data, to discover and initialize secondary stations and to report procedural errors that can not be recovered by retransmissions. The IrSC SIG proposed a technical approach to use the U-frames for data transmission (to be discussed in Chapter 6).

b) Supervisory frames (S-frames): During the information exchange procedure (NRM mode), S-frames are used to report frame sequencing errors, to acknowledge correctly received frames and to request an acknowledgement from the transmission peer.

c) Information frames (I-frames): I-frames carry information data to the remote station during the connection state. I-frames are sequenced to ensure that they are received in the correct order.

Every frame has a control field. The control field contains a frame identifier, which determines the frame type and includes the P/F bit, which is used to pass the transmission control. The control field of I-frames (Figure 3.1) contains a send sequence number (*Ns*), which is used to number the I-frames to be transmitted. The control field of I-frames and S-frames also contains a receive sequence number (*Nr*), which is used to indicate the expected sequence number of the next I-frame. The control field of S-frames and U-frames contains the command/response code (X) of the frame. SIR and FIR specifications define an 8-bit control field. *Ns* and *Nr* occupy 3 bits each and cycle through values from 0 to 7 which means the maximum window size is 7. VFIR specification extended the length of the control field to 16 bits for the 4 Mbit/s and 16 Mbit/s data rates. In the 16-bit control field, *Ns* and *Nr* are 7 bits each and cycle through values 0 to 127 which means a maximum window size of 127 is supported.

The control field always contains the P/F bit, which serves as a token passing between the communication peers. When it is set by the primary station, it is the poll (P) bit. When it is set by the secondary station, it is the final (F) bit. When the P/F bit is set (value of 1), the link direction is reversed. Primary sets the P-bit to solicit a response or sequence of responses from the secondary. When the secondary receives a frame with the P-bit set, it responds by transmitting one or more frames. The secondary sets the F-bit of the last frame it transmits to reverse link direction and returns transmission control to the primary station. The secondary stations therefore have transmission control only when they are transmitting frames to the primary station [42].

IrLAP also employs a P-timer to force the link turnaround. The P-timer is assigned with the maximum turn around time (T_{max}) agreed between the two stations during the contention state (NDM). As T_{max} stands for the maximum time the station can hold transmission control, the station starts the P-timer when it receives a frame with the P/F bit set and stops the P-timer when it transmits a frame with the P/F bit set. Expiration of P-timer indicates that the station has already held transmission control more than T_{max} and has to immediately transmit a Receive Ready (RR) S-frame with the P/F bit set to pass the transmission control.

The primary station also employs an F-timer to limit the time that a secondary station can hold the transmission control. The primary starts the F-timer once it sends a frame with the P-bit set and stops the F-timer upon the reception of a frame with the F-bit set. The F-timer expiration means that the secondary failed to return transmission control within the agreed time period. Since the P-timer operation in the secondary station guarantees that this never happens, F-timer expiration can only be explained by the loss of either the frame that contains the P-bit or the frame that contained the F-bit. The primary resolves this situation by transmitting a RR S-frame with the P-bit set when the F-timer expires.

3.3 Analytical model of IrLAP by taking propagation time into consideration

Since IrDA links are commonly employed for information transfer from one device to another, this work considers the transmission of a large amount of information data between two stations. The saturation case is studied, in which the transmitter always has information data ready for transmission. In the considered scenario, only two IrDA stations form the infrared WPAN, the primary station and a single secondary station. It is assumed that the transmitting station claims and is granted the primary role during the contention state. It is also assumed that transmission errors follow the uniform distribution. The parameters used in the current model are shown in Table 3.1.

Symbol	Parameter Description	Unit
c	Speed of light	m/sec
C	Link data rate	bits/sec
d	Distance between the transmitter and the receiver	m
p_b	Link bit error rate	-
p	Frame error rate	-
W_{max}	Maximum window size	-
N_{LAP}	Number of frames in one window	-
l_{LAP}	I-frame data length	bits
l'_{LAP}	S-frame length/ I-frame header	bits
t_I	Transmission time of an Information (I)-frame	sec
t_{Imax}	Transmission time of a 16 Kbits I-frame	sec
t_s	Transmission time of a Supervision (s)-frame	sec
t_{ta}	Minimum turnaround time	sec
t_{ack}	Acknowledgement time	sec
t_{Fout}	F-timer timeout period	sec
t_p	Propagation time	sec

Table 3-1 Parameters Used in Modelling IrLAP Throughput

In the contention state (NDM), the primary station determines the window size N_{LAP} it will employ. N_{LAP} represents the maximum number of I-frames the primary can transmit before soliciting an acknowledgement. Maximum window size parameter W_{max} is negotiated and agreed between the two stations in the contention state. However, the maximum time a station can hold transmission control (T_{max}) must always be obeyed and, according to IrLAP specification [42], T_{max} has a higher priority than W_{max}. T_{max} combined with the implemented frame size and link data rate may limit the window size that can be applied. In other words, if the time needed for transmitting W_{max} I-frames exceeds T_{max}, a smaller window size has to be implemented. Thus, N_{LAP} is given by:

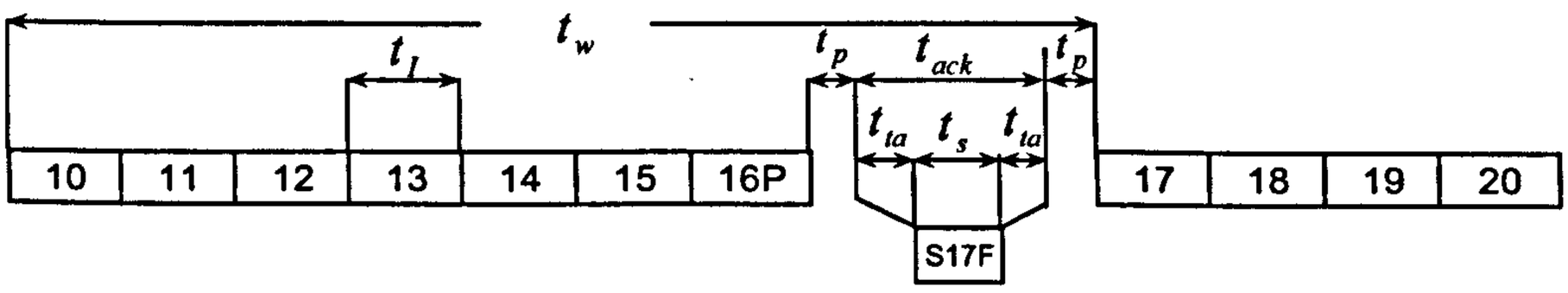
$$N_{LAP} = \min\left\{W_{\max}, \left\lfloor \frac{T_{\max}}{t_I} \right\rfloor\right\} \tag{3.1}$$

In reality, W_{max} is often smaller than $\left\lfloor \frac{T_{\max}}{t_I} \right\rfloor$ since T_{max} is relatively large (500 ms). In this work, N_{LAP} is assumed to have the same value as W_{max} ($N_{LAP} = W_{max}$).

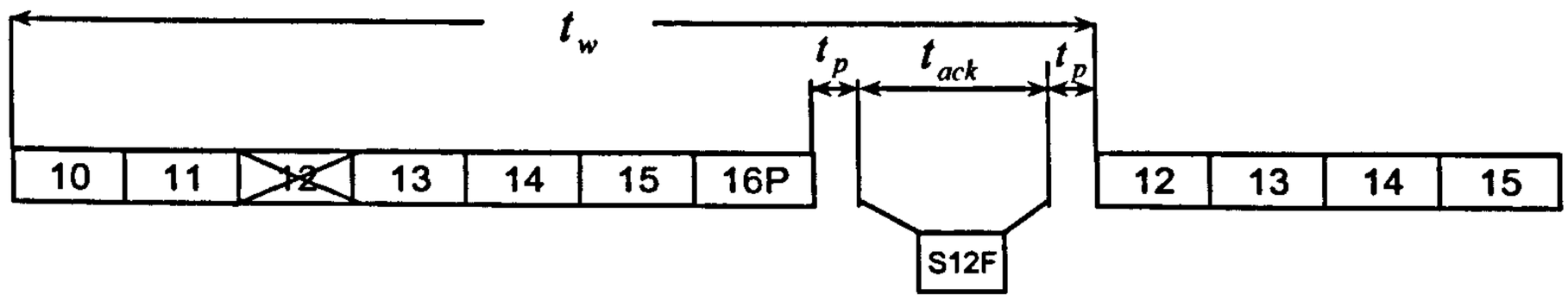
The saturation model considered in this work can be summarized as follows. The

transmitting station (primary) always has information ready for transmission. As a result, it transmits a window of N_{LAP} consecutive I-frames and reverses the link direction by setting the P-bit in the last I-frame. The receiver (secondary) awaits a minimum turnaround time (t_{ta}) and responds with an S-frame (acknowledgement) indicating the next I-frame expected. The RR S-frame always has the F-bit set, thus the receiver hands the link control back to the transmitter immediately. If no error is indicated in the RR frame, the transmitter waits t_{ta} and transmits the next window which consists of N_{LAP} new I-frames.

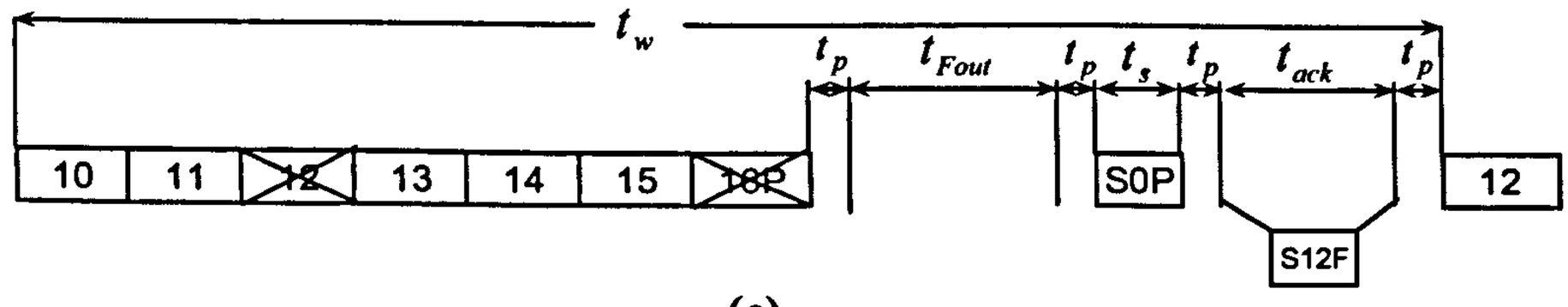
If there are any transmission errors, the transmitter determines the number of frames correctly received before any errors occurred and retransmits the erroneous frame as well as the frames following it, in the next window, followed by new I-frames to form a complete N_{LAP} frame window. If the last I-frame in a window transmission is lost, the receiver fails to respond since the P-bit is lost. When the F-timer expires, the transmitter sends an RR S-frame with the P-bit set forcing the receiver to acknowledge the correctly received I-frames.



(a)



(b)



(c)

S stands for Supervision frame, P/F stands for Poll/Final bit

Figure 3.2 IrLAP transmission model

The IrLAP transmission model for the situation where the primary has a very large file to send is illustrated in Figure 3.2. Figure 3.2 (a) shows a 7-frame window and data transmission with no errors. Figure 3.2 (b) indicates that there is an error in frame 12 and Figure 3.2 (c) shows an error in frame 12 but also an error in frame 16 where the P bit is set and lost. Since this work studies the link performance at very high data rates, the effect of propagation delay

is included in the analysis. It should be noted that it always takes the stations (both transmitter and receiver) t_p seconds (propagation delay) before responding to any requests or transmitting a new window.

In the following section, the IrLAP mathematical model presented in [78] is extended and modified by including the propagation delay. The relevant parameters and symbols used in the analysis are shown in Table 3-1. Values for t_s, t_I, t_{ack}, t_p, p and t_{Fout} are given by:

$t_s = \frac{l'_{LAP}}{C}$, $t_I = \frac{l_{LAP} + l'_{LAP}}{C}$, $t_{ack} = 2t_{ta} + t_s$, $t_p = \frac{d}{c}$, $p = 1-(1-p_b)^{l_{LAP}+l'_{LAP}}$ and $t_{Fout} = t_{\mathrm{Im}ax} + 2t_{ta}$ according to the IrDA standard [42].

In the following analysis, the concept of "window transmission time" (WTT) [69] is applied to develop the mathematical model for IrLAP in [78]. WTT is used to denote the average time needed for a complete window transmission. It denotes the average time taken from the beginning of the window's first frame transmission to the beginning of the next window's first frame transmission. In this work, WTT incorporates time needed for I-frame transmissions and acknowledgements, delays due to the link reverse, propagation and timer timeouts [140][151].

From [78], the average number of correctly transmitted frames in an N_{LAP}-frame window N_{cor} is given by:

$$N_{cor} = \sum_{i=1}^{N_{LAP}-1} i(1-p)^i p + N_{LAP}(1-p)^{N_{LAP}} = \frac{(1-p)(1-(1-p)^{N_{LAP}})}{p} \qquad (3.2)$$

where p is the frame error rate.

According to Figure 3.2, the average transmission time for one IrLAP window t_w is:

$$t_w = t_1 + t_2 + t_3 \qquad (3.3)$$

Where $t_1 = N_{LAP}t_I + t_p$, $t_2 = t_{ack} + t_p$, and $t_3 = p(t_{Fout} + t_s + 2t_p)$.

From (3.2) and (3.3), the IrLAP throughput F_{LAP} defined as the number of information bits correctly received per second is derived as:

$$F_{LAP} = \frac{l_{LAP}N_{cor}}{t_w} = l_{LAP}\frac{1-p}{p}\frac{(1-(1-p)^{N_{LAP}})}{N_{LAP}t_I + t_{ack} + 2t_p + p(t_{Fout} + t_s + 2t_p)} \qquad (3.4)$$

In the special case of an error free channel $p_b = 0$, one can easily get the frame error rate $p = 1-(1-p_b)^{l_{LAP}+l'_{LAP}} = 0$. In this case F_{LAP} becomes:

$$F_{LAP} = \frac{l_{LAP}N_{LAP}}{N_{LAP}t_I + t_{ack} + 2t_p} \qquad (3.5)$$

The IrLAP throughput efficiency (TPE) is given in (3.6), where C is the link data rate:

$$TPE = F_{LAP} / C \tag{3.6}$$

The relative time spent on each of the protocol tasks affecting IrLAP performance is given in (3.7)—(3.12).

The time intervals taken by the acknowledgements (T_{ack}), P bit frame loss (T_{Fout}) and propagation time (T_p), are given by:

$$T_{ack} = \frac{t_{ack}}{t_w} \tag{3.7}$$

$$T_{Fout} = \frac{p(t_{Fout} + t_s)}{t_w} \tag{3.8}$$

$$T_p = \frac{2t_p(1+p)}{t_w} \tag{3.9}$$

The time intervals spent on transmitting frame overheads ($T_{l'}$), transmitting error frames in a window transmission (T_{err}) and transmitting correctly received but out of sequence frames (T_{cor}) are given in (3.10), (3.11) and (3.12):

$$T_{l'} = \frac{N_{LAP} l'_{LAP} / C}{t_w} \tag{3.10}$$

$$T_{err} = \frac{pN_{LAP} l_{LAP} / C}{t_w} \tag{3.11}$$

$$T_{cor} = \frac{(N_{LAP} - pN_{LAP} - \frac{1-p}{p}(1-(1-p)^{N_{LAP}}))\frac{l_{LAP}}{C}}{t_w} \tag{3.12}$$

3.4 IrLAP window and frame sizes optimization and verification

F_{LAP} in equation (3.4) is a function of both window size N_{LAP} and frame size l_{LAP}. The IrLAP throughput can be optimized by implementing appropriate N_{LAP} and l_{LAP} for different BER [71]. If it is convenient to optimize only one variable, either N_{LAP} or l_{LAP}, the IrLAP throughput can be optimized by fixing either N_{LAP} or l_{LAP} and optimizing the other. Nevertheless, the best IrLAP throughput is achieved when both N_{LAP} and l_{LAP} are simultaneously optimized with BER. Based on the work carried out by Vitsas and Boucouvalas [72][73], the optimum window and frame sizes are derived in this section by using a similar approach.

3.4.1 Optimum window or frame size for maximum throughput

The optimum value of N_{LAP} for any fixed l_{LAP} is derived by taking $\frac{\partial F_{LAP}}{\partial N_{LAP}} = 0$ of equation (3.4). For any fixed N_{LAP}, optimum l_{LAP} is derived by taking $\frac{\partial F_{LAP}}{\partial l_{LAP}} = 0$. After some calculations and approximations (see Appendix I for the detailed derivation), optimum window (N_{opt}) and frame (l_{opt}) sizes are derived:

$$N_{opt} = \sqrt{\frac{2(t_{ack} + 2t_p)C}{{l_{LAP}}^2 p_b}} \tag{3.13}$$

and

$$l_{opt} = \sqrt{\frac{2(N_{LAP} l'_{LAP} + (t_{ack} + 2t_p)C)}{{N_{LAP}}^2 p_b}} \tag{3.14}$$

3.4.2 Simultaneous optimization of window and frame size for maximum throughput

In this case, both window and frame size are simultaneously adjusted. The maximum possible IrLAP throughput can be achieved. To optimize both N_{LAP} and l_{LAP}, the first step is to take $\frac{\partial F_{LAP}}{\partial N_{LAP}} = 0$ for equation (3.4) to derive the optimum N_{LAP} for fixed l_{LAP}. The optimum N_{LAP} equation for fixed l_{LAP} is then substituted into (3.4). Throughput F_{LAP} becomes a function of frame size l_{LAP} for optimum N_{LAP}. Next, the derivative of $\frac{\partial F_{LAP}}{\partial l_{LAP}} = 0$ is taken and the optimum l_{LAP} equation for optimum N_{LAP} is derived. Subsequently, the derived optimum l_{LAP} equation for optimum N_{LAP} is substituted into (3.4). At this stage, the throughput F_{LAP} becomes a function of N_{LAP} for optimum l_{LAP}. Finally, $\frac{\partial F_{LAP}}{\partial N_{LAP}} = 0$ is taken and the equation of optimum N_{LAP} for optimum l_{LAP} is derived. This essentially calculates $\frac{\partial F_{LAP}}{\partial N_{LAP}} = \frac{\partial F_{LAP}}{\partial l_{LAP}} = 0$. The derivation is given in Appendix I. The simultaneous optimum window and frame sizes are given in (3.15) and (3.16):

$$l_{opt} = \sqrt{\frac{l'_{LAP}}{p_b}} \tag{3.15}$$

and

$$N_{opt} = \sqrt{\frac{2(t_{ack} + 2t_p)C}{l'_{LAP}}} \tag{3.16}$$

Equations (3.15) and (3.16) reveal that N_{opt} is essentially independent of BER, and l_{opt} becomes very large and takes values larger than 16 Kbits (the maximum allowed by the IrLAP specification [42]) for low bit error rates (BER $< 2.6\times10^{-7}$ from equation (3.15) using $l'_{LAP} = 72$, and $l_{opt} = 16$Kbit).

In order to comply with the IrLAP specification, for the simultaneous N_{LAP} and l_{LAP} optimization, when l_{opt} is required to be greater than 16 Kbits (BER $< 2.6\times10^{-7}$), l_{opt} is fixed at 16 Kbits and equation (3.13) is used to calculate optimum N_{LAP}. When BER $> 2.6\times10^{-7}$, equations (3.15) and (3.16) are used to calculate the optimum N_{LAP} and l_{LAP} values.

3.4.3 Verification of the optimization equations

A number of assumptions are made in deriving the optimum window and frame size. In order to examine the accuracy of equations (3.13)--(3.16), verification is carried out based on the comparison of the exact optimum results and the analytical optimum results. The exact optimum results are obtained by a numerical algorithm which locates maximum throughput, given by equation (3.4), by numerically cycling the integer values of N_{LAP} or l_{LAP} in the range 1-127 and 512-16384 bits respectively for different BER's. Analytical optimum results are calculated based on optimum equations (3.13)--(3.16). The IrDA standard recommends that the bit error rate (BER) should be less than 10^{-8}. However, due to the ad hoc nature of infrared links, many factors can increase the BER to over 10^{-8}, such as careless alignment, high ambient noise (close to a fluorescent light source), and partial blockage. Given the fact that the link protocol faces much greater challenge under higher bit error rate, we therefore also examine BER>10^{-8}.

In order to have an accurate and intuitive comparison between the analytical and the exact results, the mean of the difference between the two results and the standard deviation of the difference are examined.

The difference between the analytical and the exact results is given by:

$$x(i) = \left| y_{analytical}(i) - y_{exact}(i) \right| \tag{3.17}$$

The equation for calculating the mean of the difference between the analytical and the exact results is:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x(i) \tag{3.18}$$

The equation for calculating the standard deviation of the difference between the analytical and the exact results is:

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(x(i)-\bar{x}\right)^2} \tag{3.19}$$

In Figure 3.3, Figure 3.4 and Figure 3.5, the throughput efficiency, simultaneous optimum frame and window sizes are plotted against bit error rate in the range of 10^{-7} to 10^{-5} for two different data rates, 4 Mbit/s and 16 Mbit/s. The values of the other parameters are given as follows: t_{ta}=10^{-4} s, $d = 2\text{m}$.

Since a very good match between the throughput efficiency calculated from the analytical equations and the throughput efficiency derived from numerical methods is observed from Figure 3.3, as well as the value of $\bar{x}$ and σ, approximations made to derive (3.13) to (3.16) are validated. Figure 3.3 also shows that the IrLAP throughput is highly dependent upon BER and in fact degrades significantly with the high BER values.

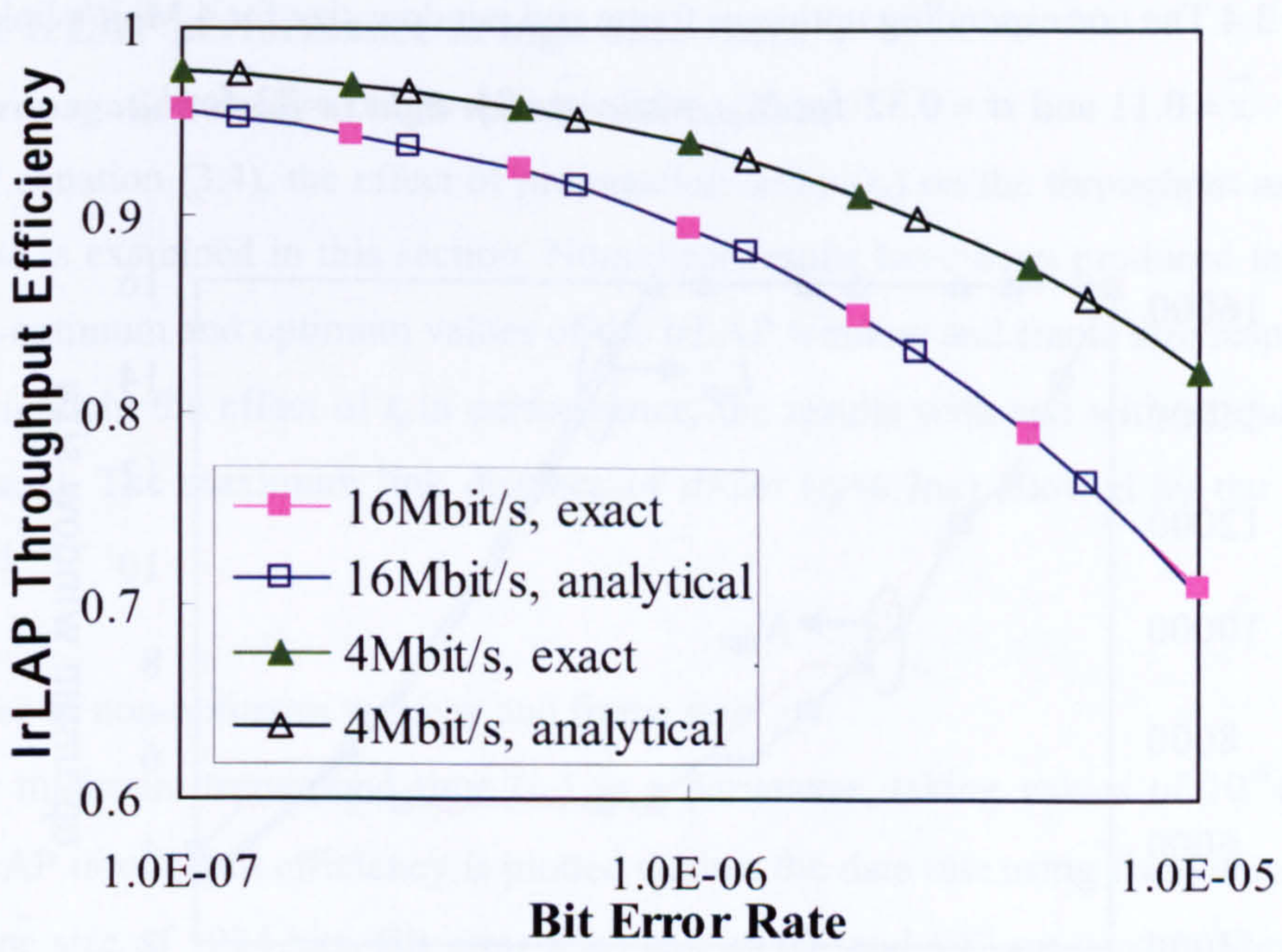


Figure 3.3 Comparison of TPE by using exact algorithm and analytical formula for 4 and 16 Mbit/s links

$\bar{x} = 0.0007$ and $\sigma = 0.0006$ for the 4 Mbit/s link, while $\bar{x} = 0.001$ and $\sigma = 0.0008$ for the 16 Mbit/s link.

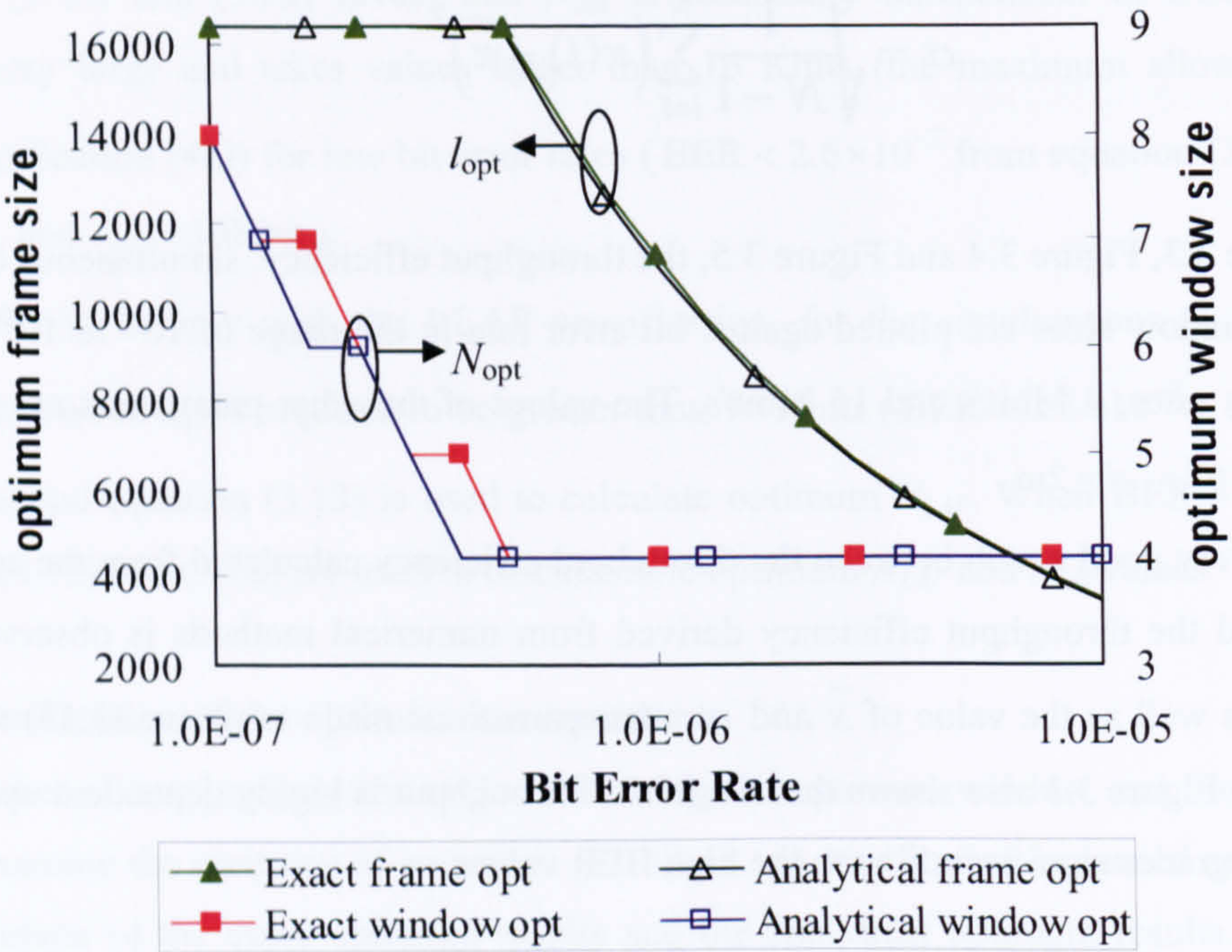


Figure 3.4 The corresponding optimum frame and window size for 4 Mbit/s links

$\bar{x} = 0.11$ and $\sigma = 0.32$ for N_{opt}, while $\bar{x} = 84$ and $\sigma = 72$ for l_{opt}.

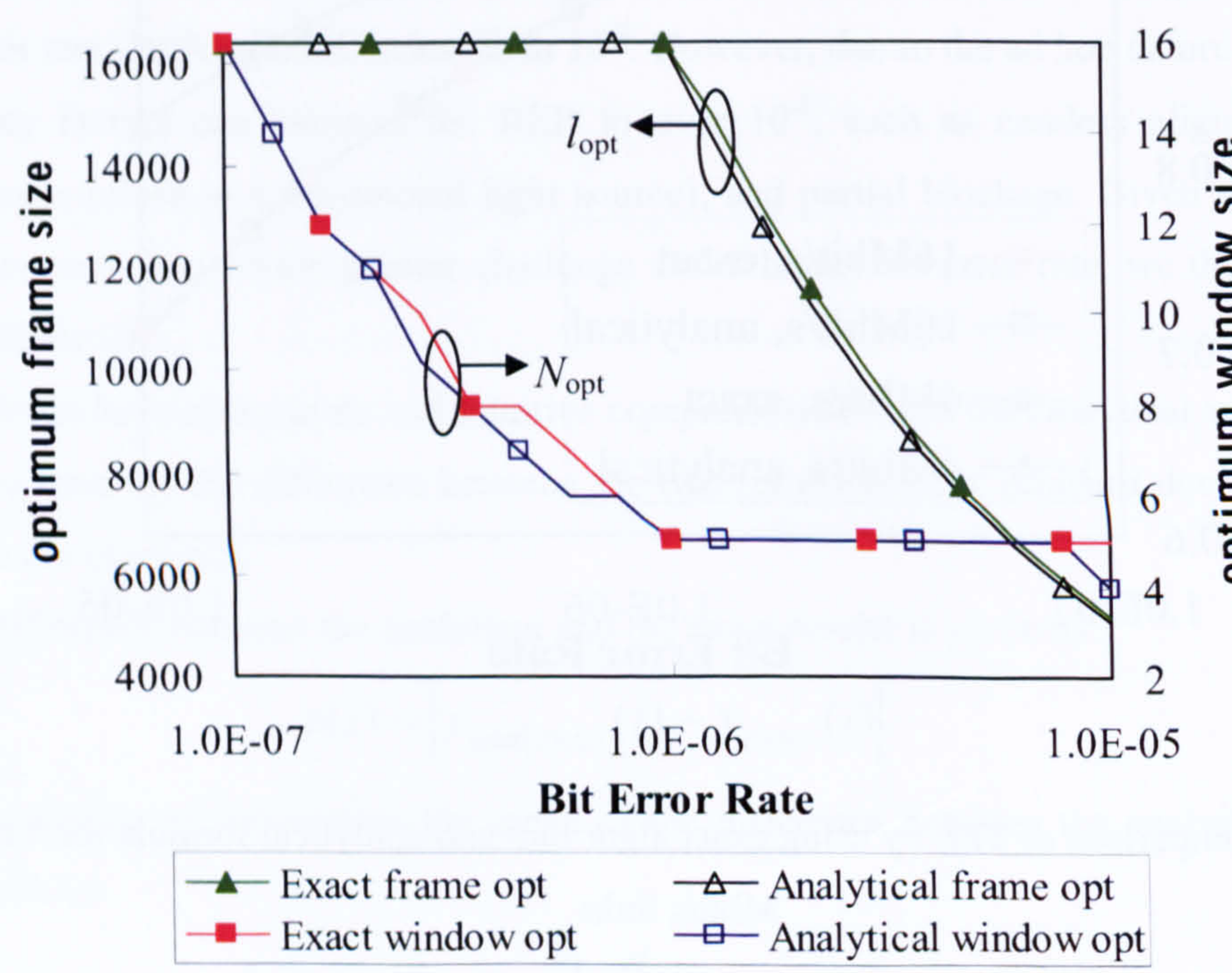


Figure 3.5 The corresponding optimum frame and window size for 16 Mbit/s links

$\bar{x} = 0.21$ and $\sigma = 0.42$ for N_{opt}, while $\bar{x} = 118$ and $\sigma = 152$ for l_{opt}.

The simultaneous optimum frame and window size values are plotted against BER in Figure 3.4 and Figure 3.5, for 4 Mbit/s and 16 Mbit/s respectively. It should be noted that at the low BER, the optimum frame size values are required to be greater than the maximum frame size and are fixed at 16 Kbits. The values of $\bar{x}$ and σ are also calculated for the optimum window and frame sizes. As shown in both figures, the optimum frame size decreases with the increasing BER. The value of the optimum window size increases significantly with the decrease of BER when the optimum frame size is fixed. For both optimum frame size and window size, the analytical and numerical result curves are very close to each other and the corrsponding $\bar{x}$ and σ are also small. The difference between the numerical and analytical optimum values is negligible since the corresponding throughput efficiencies shown in Figure 3.3 have a very good match.

3.5 The IrLAP performance at high data rates

3.5.1 Propagation delay at high speed transmissions

Using equation (3.4), the effect of propagation delay (t_p) on the throughput as a function of data rate is examined in this section. Numerical results have been produced in two cases, using non-optimum and optimum values of the IrLAP window and frame size respectively. In order to highlight the effect of t_p in performance, the results with and without considering t_p are compared. The maximum link distance of d=2m (t_p=6.7ns) allowed by the standard is considered.

3.5.1.1 Use of non-optimum window and frame size

Using minimum turnaround time (t_{ta}) as a parameter, taking values of 10^{-6}s, 10^{-7}s and $10^{-8}s$, IrLAP throughput efficiency is plotted against the data rate using fixed window size of 7 and frame size of 1024 bits. Bit error rates (p_b) of 10^{-4} and 10^{-8} are used in Figure 3.6 and Figure 3.7 respectively.

For the non-optimum N_{LAP} and I_{LAP}, the effect of propagation time becomes significant for t_{ta} of 10^{-8}s and data rate greater than 10 Gbit/s. The results also show that the effect of the propagation delay is independent of the BER (by comparing respective curves in Figure 3.6 and Figure 3.7).

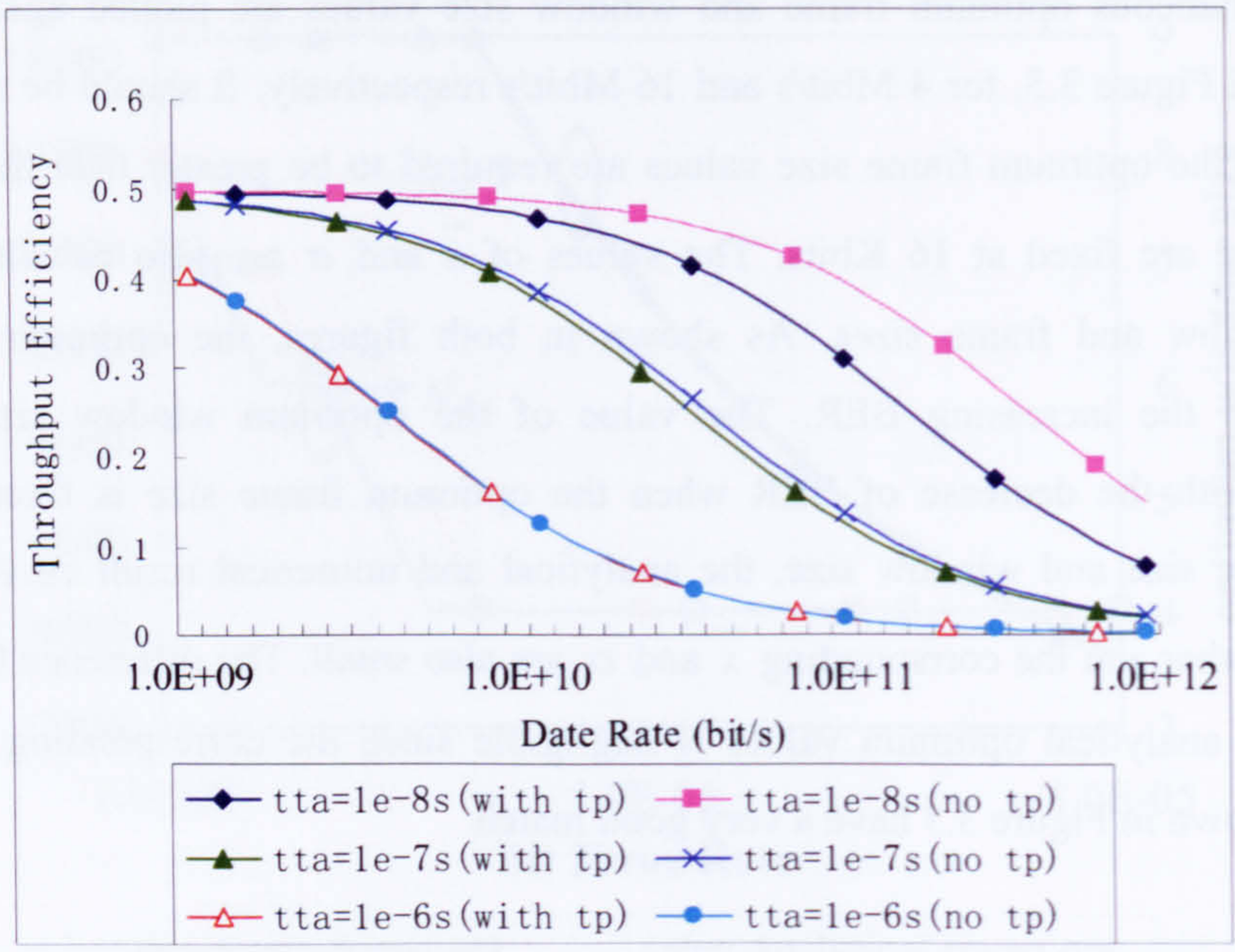


Figure 3.6 Effects of propagation delay with N_{LAP} =7, l_{LAP}=1024 bits and p_b=10^{-4} $\bar{x} = 0.047$ and $\sigma = 0.039$ for t_{ta} of 10^{-8}s, $\bar{x} = 0.008$ and $\sigma = 0.005$ for t_{ta} of 10^{-7}s, $\bar{x} = 0.0008$ and $\sigma = 0.0006$ for t_{ta} of 10^{-6}s. $\bar{x}$ is the mean of the throughput efficiency difference with and without considering t_p and σ is the standard deviation of the throughput efficiency difference. $\bar{x}$ and σ stand the same meaning throughout section 3.4.

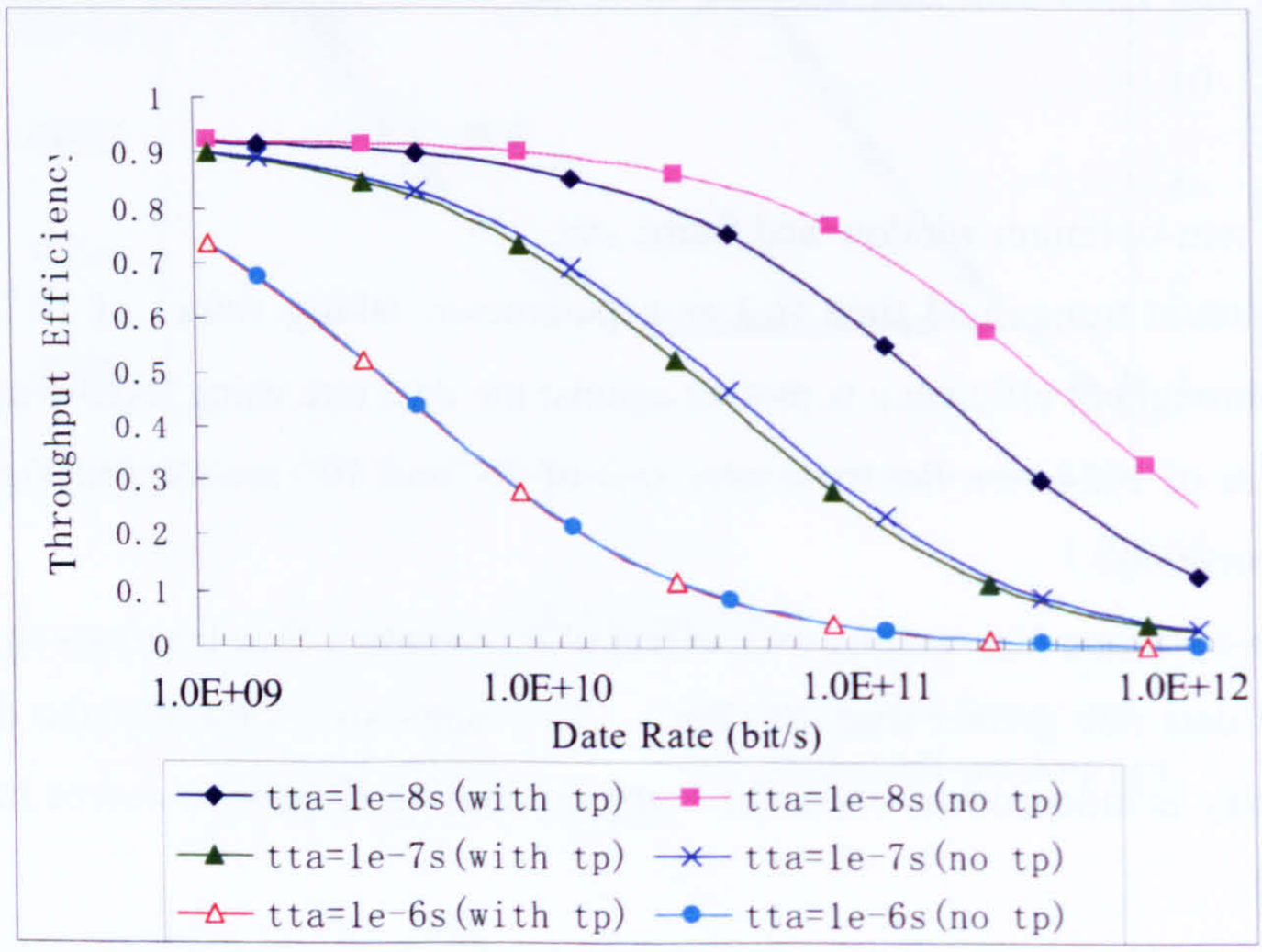


Figure 3.7 Effects of propagation delay with N_{LAP} =7, l_{LAP}=1024 bits and p_b=10^{-8} $\bar{x} = 0.089$ and $\sigma = 0.071$ for t_{ta} of 10^{-8}s, $\bar{x} = 0.015$ and $\sigma = 0.009$ for t_{ta} of 10^{-7}s, $\bar{x} = 0.0014$ and $\sigma = 0.0011$ for t_{ta} of 10^{-6}s

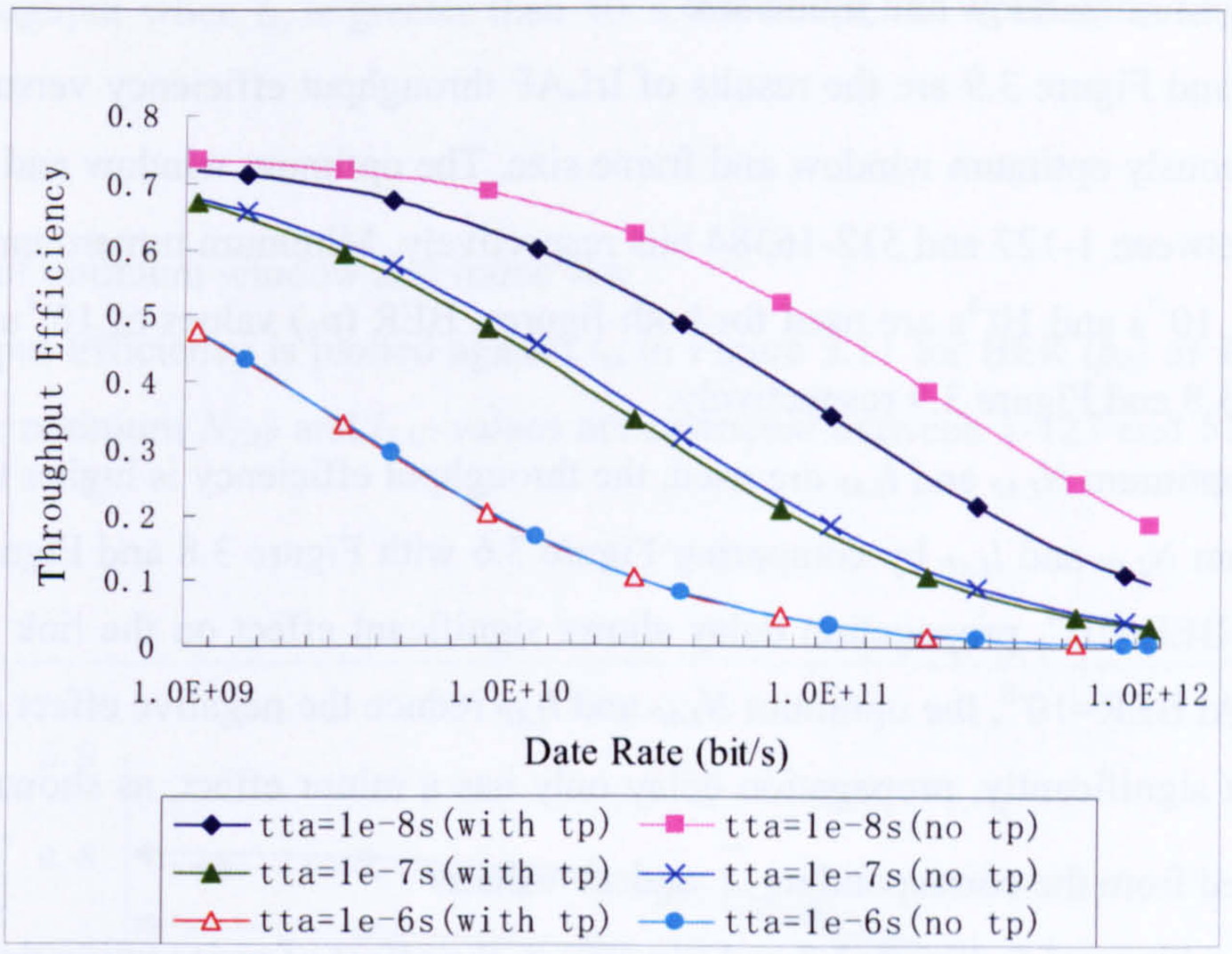


Figure 3.8 Effects of propagation delay with optimum N_{LAP} & l_{LAP} and p_b=10^{-4}

$\bar{x} = 0.083$ and $\sigma = 0.040$ for t_{ta} of 10^{-8}s, $\bar{x} = 0.014$ and $\sigma = 0.005$ for t_{ta} of 10^{-7}s, $\bar{x} = 0.0011$ and $\sigma = 0.0008$ for t_{ta} of 10^{-6}s

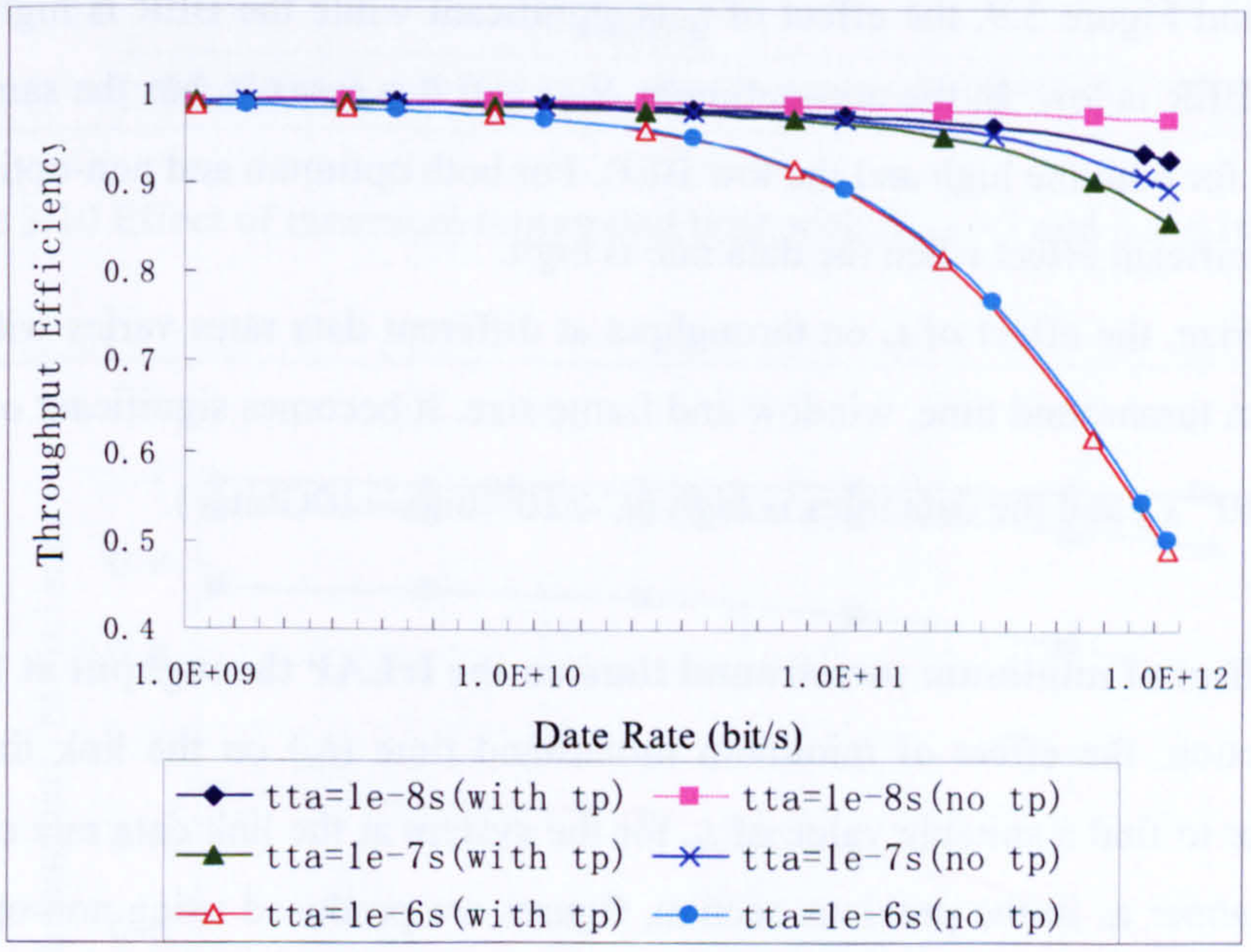


Figure 3.9 Effects of propagation delay with optimum N_{LAP} & l_{LAP} and p_b=10^{-8}

$\bar{x} = 0.009$ and $\sigma = 0.011$ for t_{ta} of 10^{-8}s, $\bar{x} = 0.007$ and $\sigma = 0.010$ for t_{ta} of 10^{-7}s, $\bar{x} = 0.004$ and $\sigma = 0.004$ for t_{ta} of 10^{-6}s

3.5.1.2 Use optimum window and frame size

Figure 3.8 and Figure 3.9 are the results of IrLAP throughput efficiency versus data rate using simultaneously optimum window and frame size. The optimum window and frame size are restricted between 1-127 and 512-16384 bits respectively. Minimum turnaround time with values of 10^{-6}s, 10^{-7}s and 10^{-8}s are used for both figures. BER (p_b) values of 10^{-4} and 10^{-8} are used in Figure 3.8 and Figure 3.9 respectively.

When the optimum N_{LAP} and I_{LAP} are used, the throughput efficiency is higher than that of the non-optimum N_{LAP} and I_{LAP} by comparing Figure 3.6 with Figure 3.8 and Figure 3.7 with Figure 3.9. At BER=10^{-4}, propagation delay shows significant effect on the link throughput when t_{ta}=10^{-8}. At BER=10^{-8}, the optimum N_{LAP} and I_{LAP} reduce the negative effect of t_p on the link throughput significantly, propagation delay only has a minor effect, as shown in Figure 3.9 and observed from the corresponding $\bar{x}$ and σ values.

As shown in Figure 3.6, Figure 3.7 and Figure 3.8, the effect of propagation delay is more pronounced when the value of t_{ta} is less than 10^{-7}s. As the value of t_{ta} decreases and approaches the value of t_p (t_p=$6.7\times10^{-9}s = 6.7ns$), the propagation delay degrades the link throughput considerably. The effect of propagation delay on throughput is overwhelmed by the effect of t_{ta} when large t_{ta} values are used.

The results also show that when the optimum values of window and frame size are used, in Figure 3.8 and Figure 3.9, the effect of t_p is significant while the BER is high, but much less while the BER is low. In the non-optimum N_{LAP} and I_{LAP} cases, t_p has the same effect on the throughput for both the high and the low BER. For both optimum and non-optimum cases, t_p has more significant effect when the data rate is high.

To summarize, the effect of t_p on throughput at different data rates varies with the value of the minimum turnaround time, window and frame size. It becomes significant only when t_{ta} is small ($t_{ta} \le 10^{-7}s$) and the data rates is high ($C \ge 10^{10}$bit/s = 10Gbit/s).

3.5.2 The effect of minimum turnaround time on the IrLAP throughput at 10 Gbit/s

In this section, the effect of minimum turnaround time (t_{ta}) on the link throughput is studied in order to find a suitable value of t_{ta} for the system at the link data rate of 10 Gbit/s. In a similar manner as in the previous section, figures are produced using non-optimum and optimum N_{LAP} and I_{LAP} values respectively.

3.5.2.1 Use of non-optimum window and frame size

The IrLAP throughput Efficiency is plotted against t_{ta} in the range from 10^{-7}s to 10^{-9}s in Figure 3.10. BER (p_b) of 10^{-7}, 10^{-6}, 10^{-5} and 10^{-4}, window size N_{LAP}=7, frame size I_{LAP} =16 Kbits are used in the calculation. In Figure 3.10, it can be seen that t_{ta} has significant effect on

the link throughput when t_{ta} is greater than 10^{-8}s. However, the t_{ta} effect is negligible when $t_{ta} < 10^{-8} s$.

3.5.2.2 Use of optimum window and frame size

Throughput Efficiency is plotted against t_{ta} in Figure 3.11 for BER (p_b) of 10^{-7}, 10^{-6}, 10^{-5} and 10^{-4}. The optimum N_{LAP} and l_{LAP} values are restricted between 1-127 and 512-16384 bits respectively.

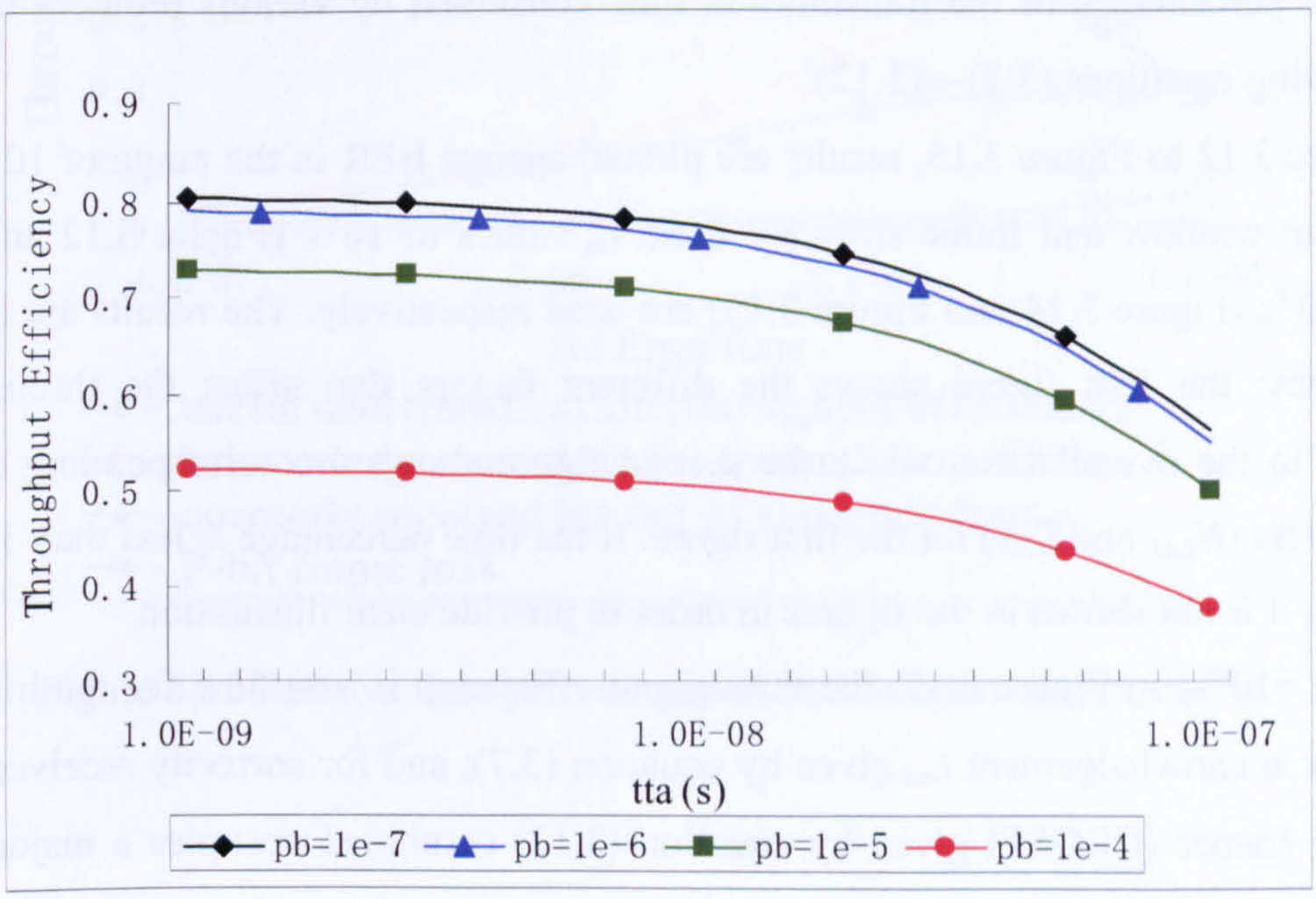


Figure 3.10 Effect of minimum turnaround time with N_{LAP} =7 and l_{LAP} =16Kbit

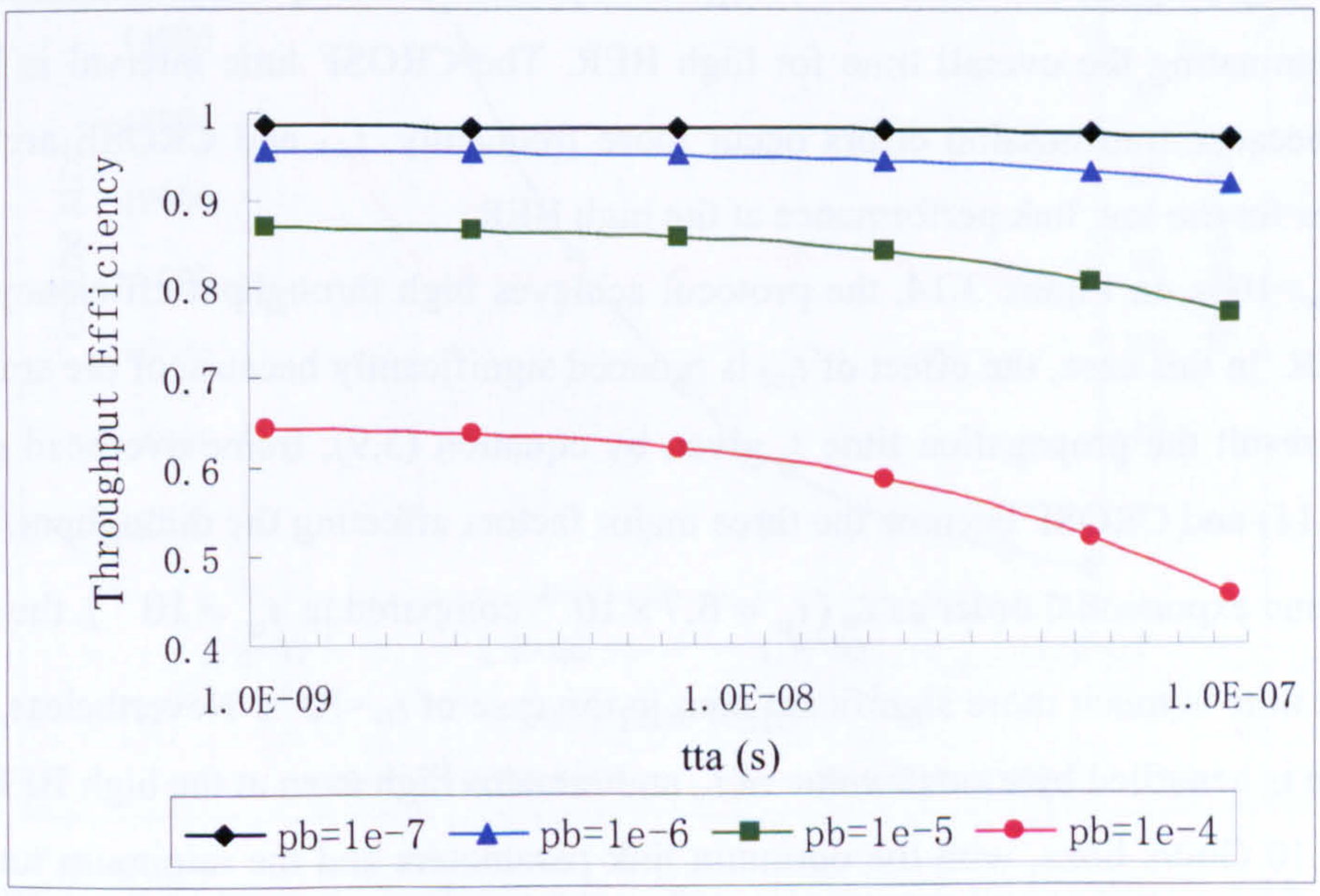


Figure 3.11 Effect of minimum turnaround time with optimum N_{LAP} and l_{LAP}

By comparing Figure 3.10 and Figure 3.11, much higher throughput efficiency is achieved by using the optimum N_{LAP} and l_{LAP} values. The results from Figure 3.10 and Figure 3.11 reveal that the minimum turnaround time has significant effect only when it takes the value $t_{ta} > 10^{-8} s$. Therefore, t_{ta} of 10^{-8}s should be used when the link data rate is 10 Gbit/s.

3.5.3 The effects of different protocol tasks on the 10 Gbit/s link throughput by using the optimum window and frame values

The significance of different factors that affect the IrLAP throughput is studied in this section. The percentages of the transmission time consumed by various protocol tasks, are examined using equations (3.7)—(3.12).

In Figure 3.12 to Figure 3.15, results are plotted against BER in the range of 10^{-4} to 10^{-7}. The optimum window and frame sizes are used. t_{ta} values of 10^{-6}s (Figure 3.12 and Figure 3.13) and 10^{-8}s (Figure 3.14 and Figure 3.15) are used respectively. The results are shown in paired figures: the first figure shows the different factors that affect the throughput in comparison to the overall time, while the second figure shows the corresponding optimum link parameters (N_{LAP} and l_{LAP}) for the first figure. If the time percentage is less than 1% of the overall time, it is not shown in the figures in order to provide clear illustration.

When t_{ta}=10^{-6}s, in Figure 3.12, the throughput efficiency is very low for the high BER. The time for acknowledgement t_{ack} given by equation (3.7), and for correctly received but out of sequence frames (CROSF) given by equation (3.12) combined occupies a major portion (>0.5) of the overall time when BER>10^{-5}. The t_{ack} time interval increases further for $\mathrm{BER} > 0.5 \times 10^{-4}$. The reason is that small optimum window size (N_{LAP}) is used for high BER, as shown in Figure 3.13, and the acknowledgement packets are frequently transmitted. t_{ack} is therefore dominating the overall time for high BER. The CROSF time interval is large for high BER because transmission errors occur more frequently. t_{ack} and CROSF are the two main reasons for the low link performance at the high BER.

When t_{ta}=10^{-8}s, in Figure 3.14, the protocol achieves high throughput efficiency even at the high BER. In this case, the effect of t_{ack} is reduced significantly because of the small value of t_{ta}. As a result the propagation time t_p given by equation (3.9), frame overhead given by equation (3.11) and CROSF become the three major factors affecting the throughput. Because t_p has the same exponential order as t_{ta} ($t_p = 6.7 \times 10^{-9}$ compared to $t_{ta} = 10^{-8}$), the effect of propagation time is much more significant than in the case of t_{ta}=10^{-6}s. Nevertheless, the link performance is benefited by a small value of t_{ta} and remains high even at the high BER.

For the 10 Gbit/s links, with the optimum link parameters and the minimum turnaround time as small as 10^{-8}s, a throughput efficiency of 0.65 can still be achieved even at BER as high as 10^{-4}, as shown in Figure 3.14.

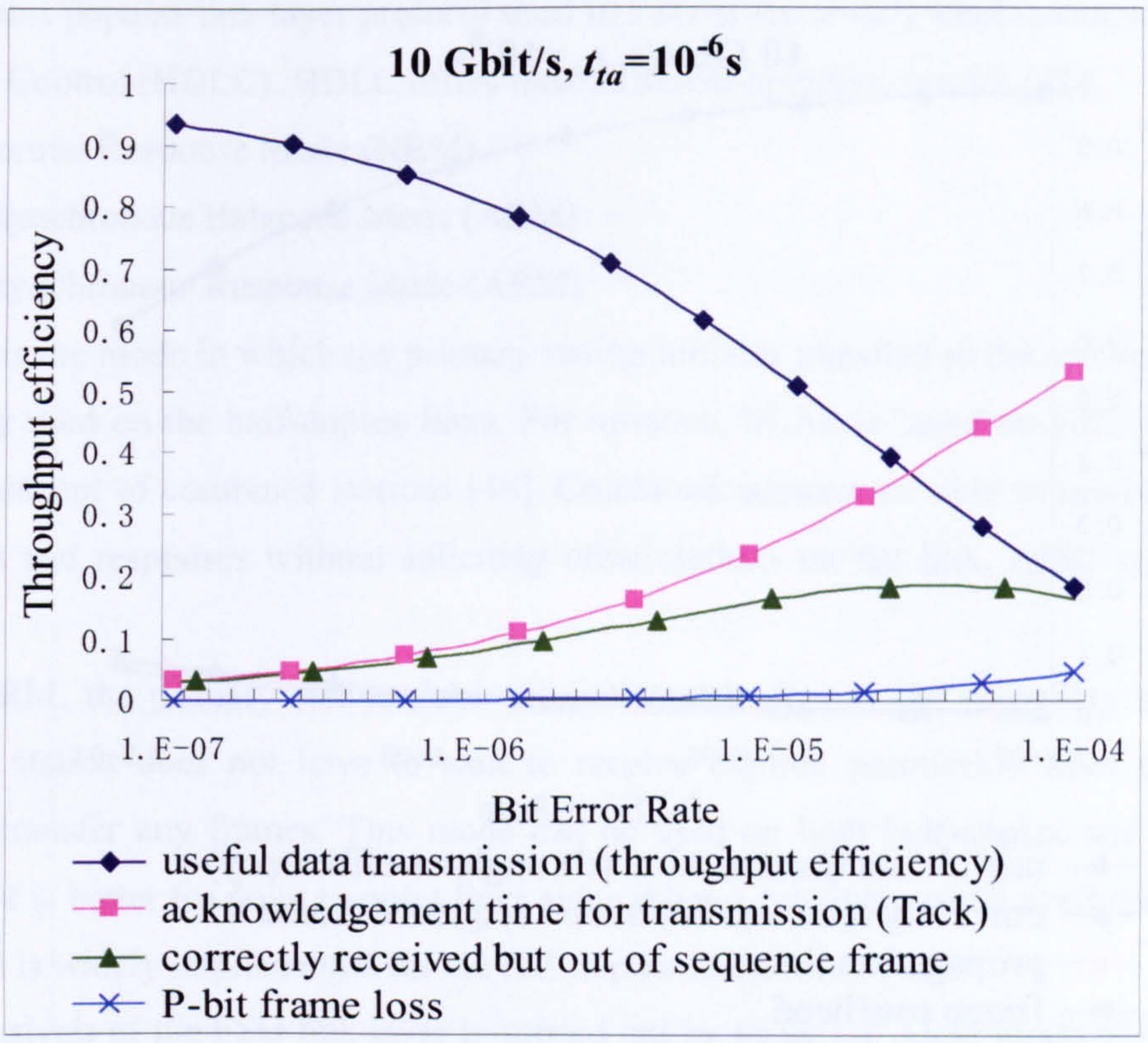


Figure 3.12 Relative time percentages of transmission time, t_{ta}=1e^{-6}s

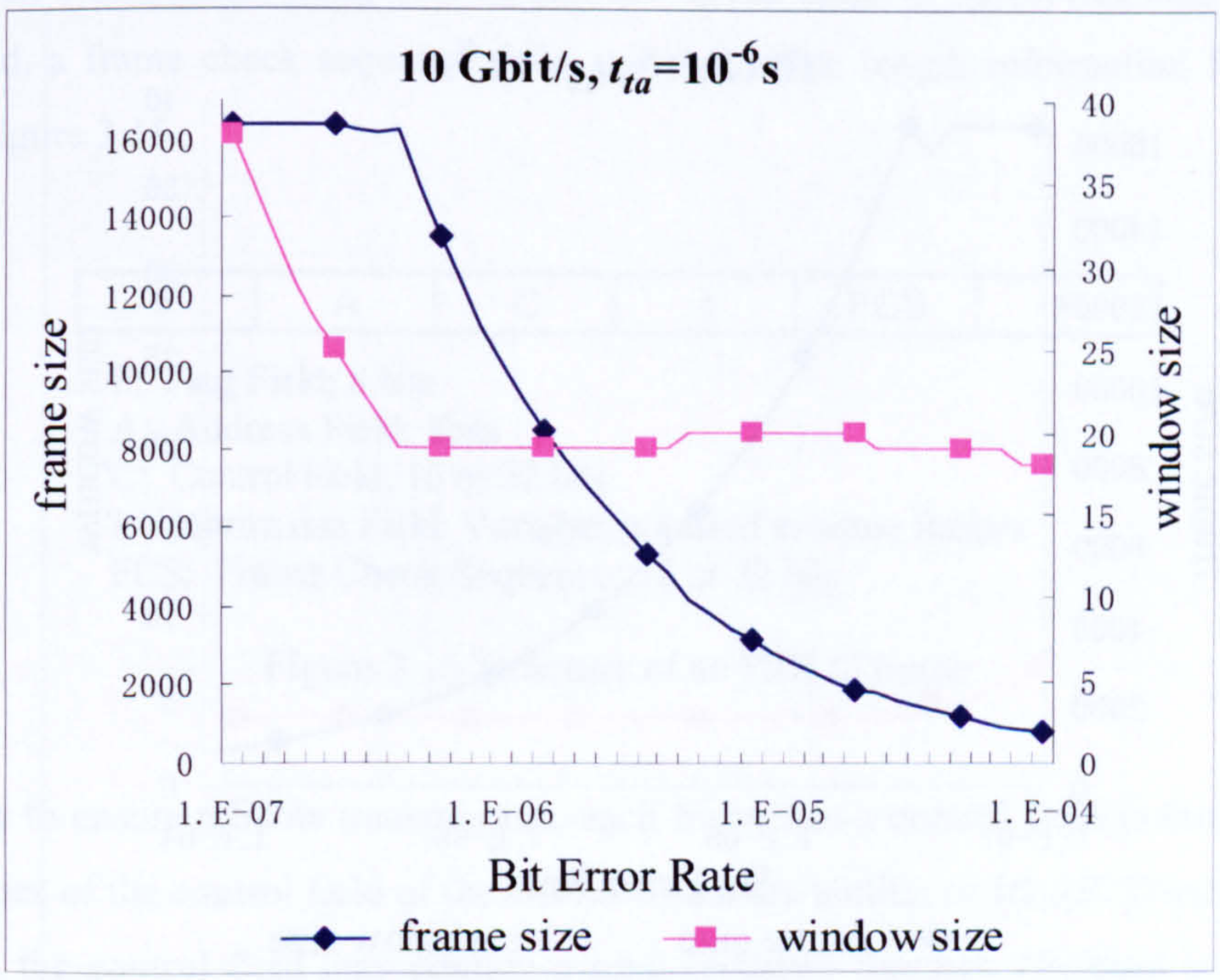


Figure 3.13 Optimum values of N_{LAP} & l_{LAP} corresponding to Figure 3.12

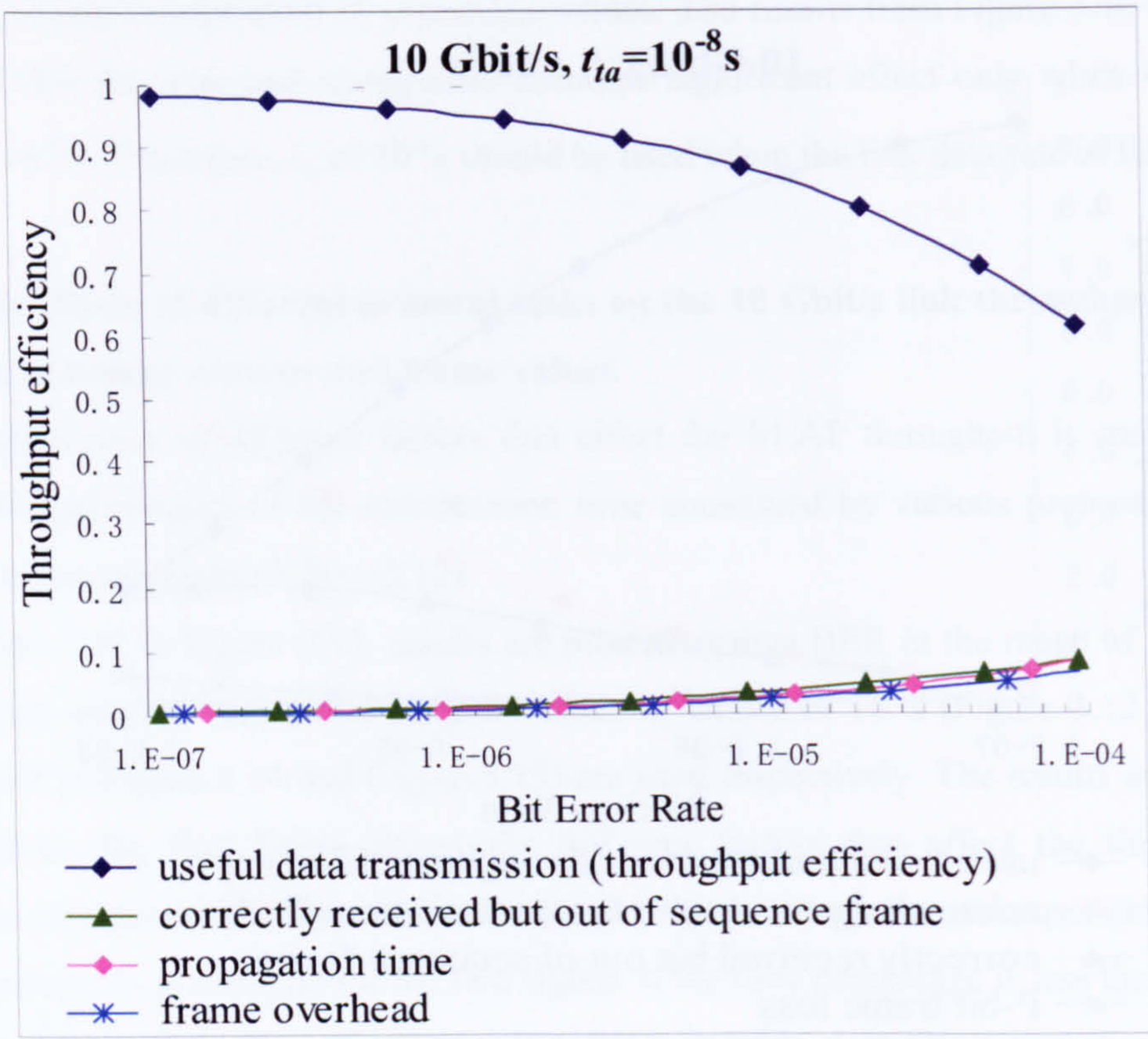


Figure 3.14 Relative percentages of transmission time, t_{ta}=1e^{-8}s

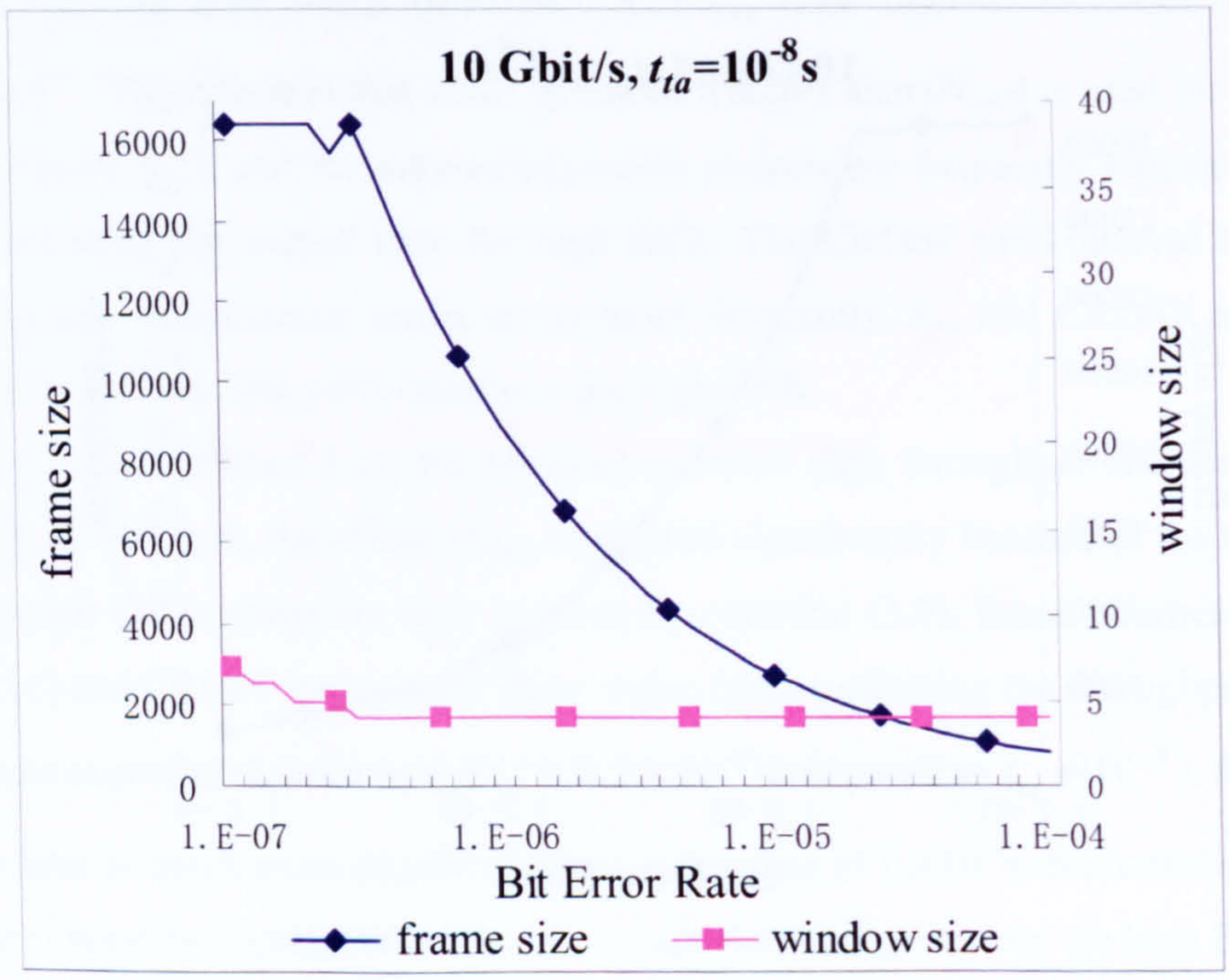


Figure 3.15 Optimum values of N_{LAP} & l_{LAP} corresponding to Figure 3.14

3.6 Full-duplex link layer for FSO links

The most popular link layer protocol used in FSO is the widely implemented High-Level Data Link Control (HDLC). HDLC offers three different operation modes [48]:

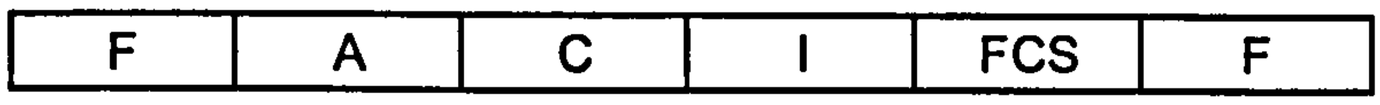
- Normal Response Mode (NRM)
- Asynchronous Balanced Mode (ABM)
- Asynchronous Response Mode (ARM)

NRM is the mode in which the primary station initiates transfers to the secondary station, and is only used on the half-duplex links. For instance, IrLAP is based on NRM [42]. ABM uses the concept of combined stations [48]. Combined stations are able to send and receive commands and responses without soliciting other stations on the link. ABM is not widely adopted.

For ARM, the primary station does not initiate transfers to the secondary station. The secondary station does not have to wait to receive explicit permission from the primary station to transfer any frames. This mode can be used on both half-duplex and full-duplex links. ARM is better for point to point links since it has a relatively small overhead compared to ABM. It is widely implemented for the full-duplex links.

The analysis of the FSO link layer is carried out by using the ARM mode of HDLC. The Go-Back-N (GBN) scheme is selected as the automatic repeat request (ARQ) mechanism of the link [93]. The analysis is limited to the interconnection of two stations.

Similarly to the structure of the IrLAP frame Figure 3.1, a HDLC frame consists of two flag fields, an 8-bit address field, a 16-bit (for low speed links) or 32-bit (for high speed links) control field, a frame check sequence field and a variable length information field [48], as shown in Figure 3.16.



F: Flag Field; 8 bits
A: Address Field; 8bits
C: Control Field; 16 or 32 bits
I: Information Field; Variable, not used in some frames
FCS: Frame Check Sequence; 16 or 32 bits

Figure 3.16 Structure of an HDLC frame

In order to ensure reliable transmission, each frame has a control field in the header. The functionalities of the control field of the HDLC frame are similar to IrLAP. Depending on the frame type, the control field may contain a send sequence number, *Ns*, used to number the frames to be sent or a receive sequence number, *Nr*, used to number frames correctly received. For the purpose of requesting an acknowledgement from the receiver, the control

field also contains the poll/final bit (P/F), which provides signalling between the primary station and secondary station. The primary station sets P-bit to acquire a status response from the secondary station. The secondary station responds to the P-bit by sending the Supervisory frame to the primary station with the F-bit set. P/F bit is set in the final frame of the transmission to indicate the end of a full window and request acknowledgement from the receiver. A full-duplex data transmission model is considered in the following analysis. Thus, one end of the link has to send Information (I-) frames, as well as the Supervisory (S-) frames to acknowledge error or successful window transmission from the other end.

The transmission procedure consists of the following steps: a sender transmits data in the form of I-frames with a length of l_{IIDLC} and can transmit N_{IIDLC} (window size) frames without receiving acknowledgement from the receiver. I-frame contains a send sequence number (*Ns*) denoting the order of the frame. At the end of each window, the sender requests an acknowledgement from the receiver. If a full window is successfully received, the receiver sends a Receive Ready S-frame (RR-S). RR-S contains a receive sequence number (*Nr*) which acknowledges the correct reception of frames up to *Nr*-1, thus indicating that *Nr* is the next frame expected. If an error is found during the transmission, the receiver will send a Reject S-frame (REJ-S) immediately to request a retransmission. REJ-S contains an *Nr* which rejects frame *Nr* and indicates that *Nr* is the next (retransmit) frame expected. The link propagation delay is considered and added to both responses. In the GBN scheme, all the frames will be discarded after any corrupted frame is detected.

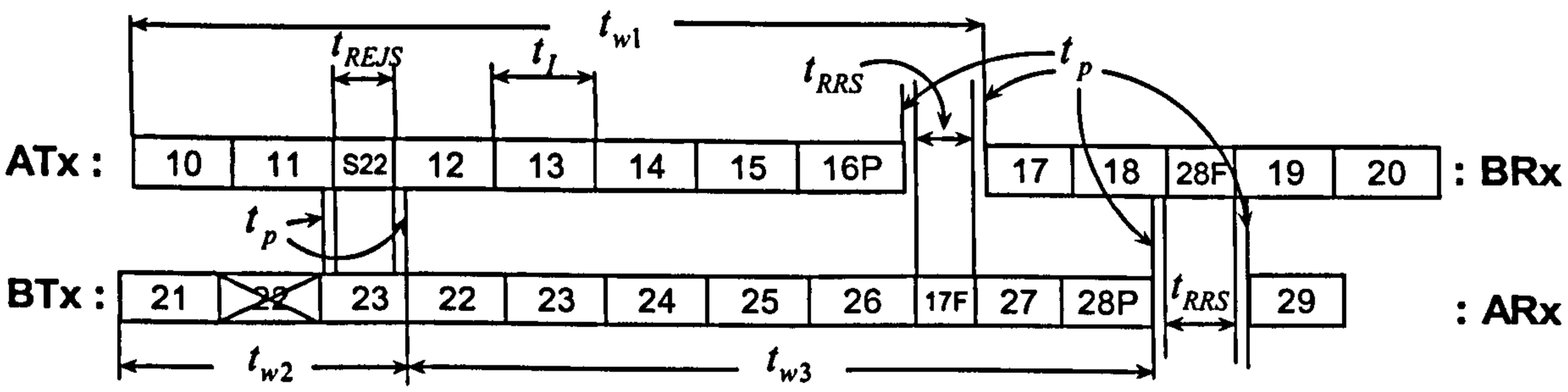


Figure 3.17 Window transmission model of a duplex link with two channels of ATx-BRx and BTx-ARx. Where Tx stands for Transmitter and Rx stands for Receiver

The analysis in this section extends the mathematical model presented in [81]. From [81], the link transmission model is illustrated in Figure 3.17 and Figure 3.18. Figure 3.17 demonstrates a 7 frames window transmission on a duplex link with two channels. It shows an error free transmission for ATx-BRx channel, while there is an error in frame (#) 22 for

BTx-ARx channel. It should be noted that propagation time (t_p) is not necessarily smaller than the S-frame transmission time (t_S).

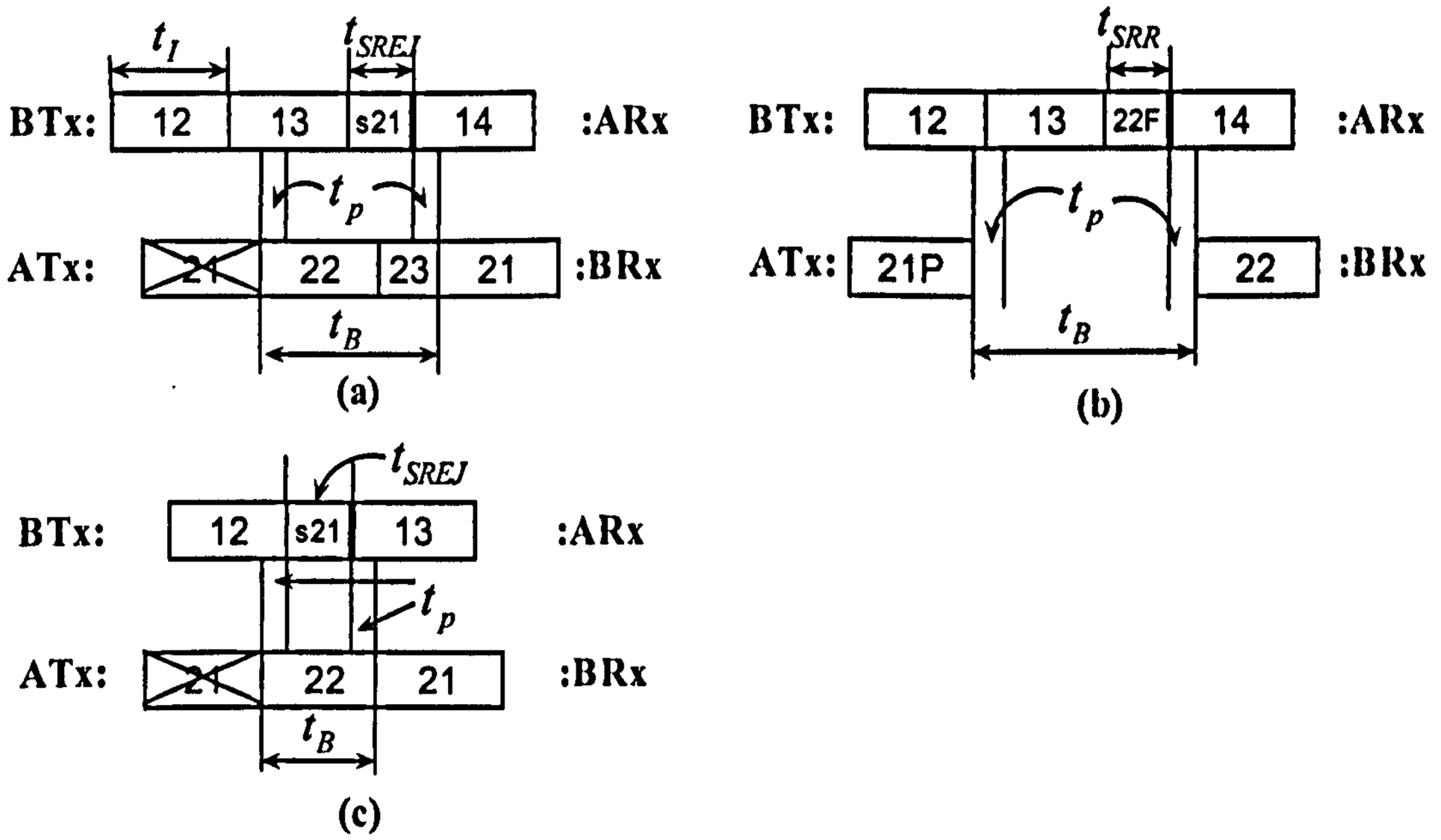


Figure 3.18 Acknowledgements for an error or a successful window transmission from BTx to ATx-BRx channel

Figure 3.18 illustrates three possible scenarios that BTx acknowledges for an error or a successful window transmission from ATx. In Figure 3.18 (a), BRx receives a corrupted #21 from ATx while BTx is transmitting #13 at the same time. BTx sends the REJ-S #21 after #13. In Figure 3.18 (b), BRx receives #21 with Poll bit set (last I-frame of the window) while BTx has just started to transmit #13. BTx sends the RR-S #22 after #13. In Figure 3.18 (c), BRx notices an error in #21 from ATx before the transmission of #13 starts. BTx then sends the REJ-S #21 before sending #13.

Symbol	Parameter Description	Unit
N_{IIDLC}	Number of frames in one window (window size)	-
L_{IIDLC}	Information (I)-frame message data length (frame size)	bit
l'_{IIDLC}	Supervision (S)-frame length/ I-frame overhead	bit
t_I	Transmission time of an I-frame	sec
t_S	Transmission time of an S-frame	sec
t_{RRS}	Transmission time of an RR-S	sec
t_{REJS}	Transmission time of an REJ-S	sec

Table 3-2 Parameters used in the FSO link throughput modelling

The relevant parameters and symbols used in the analysis are shown in Table 3-1 and Table 3-2. t_S, t_{RRS}, t_{REJS}, t_I, t_p and p are given by:

$$t_S = t_{RRS} = t_{REJS} = \frac{l'_{HDLC}}{C},\ t_I = \frac{l_{HDLC} + l'_{HDLC}}{C},\ t_p = \frac{d}{c} \text{ and } p = 1-(1-p_b)^{l_{HDLC}+l'_{HDLC}}$$

The frame overhead is 72 bits (l'_{IIDLC}=72). Using [42] as a guide, the value ranges of window and frame sizes are restricted to 1-512 and 128-16384 bits respectively. Therefore, *Ns* and *Nr* cycle from 0 to 511. The maximum value for l_{IIDLC} is 16384 bits.

3.7 Analytical model of the full-duplex FSO link protocol and the effect of propagation delay using non-optimum link parameters

3.7.1 HDLC modelling for the FSO links

In [81], Ozugur analysed the performance of HDLC and the effect of window size and packet error rate on the link utilization. This thesis extends and consummates the work presented in [81]. In this section, a mathematical model is developed which allows derivation of link throughput for the FSO full-duplex link in the presence of bit errors [139] [150]. Similarly to the derivation of the IrDA throughput equation, the concept of 'window transmission time' (WTT) is used in the analysis.

A saturation scenario is considered: there is always a frame ready to transmit in both directions. The S-frame is assumed to be error free due to the small size. Analysis of one link direction is sufficient since the duplex links are symmetrical. Throughput of the channel consisting of transmitter A and receiver B (ATx-BRx), as shown in Figure 3.17, is considered.

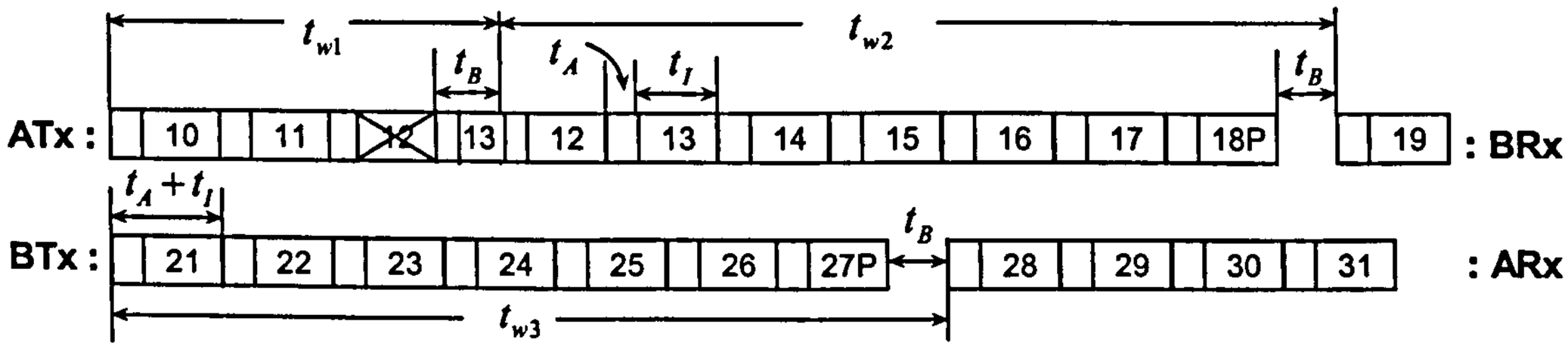


Figure 3.19 illustrates the transmission model. t_A represents the average time per frame for sending S-frames and t_B represents the average waiting time for receiving S-frames. ATx-BRx channel: Retransmit frame 12 due to error; BTx-ARx channel: Error free transmission of a window.

Three different tasks of the duplex transmission are studied individually: sending S-frames to acknowledge BTx, waiting for the S-frame from BTx and I-frame transmission. By considering these three protocol tasks, the transmission model is illustrated in Figure 3.19 which shows the average time of sending S-frame t_A at the beginning of each frame transmission and the average time waiting for an S-frame t_B. First, t_A is to be identified in the following section.

As described in [81], the error free reception of the last frame of a window indicates a successful window transmission. Therefore, the probability of error free reception of the last frame of a window (p_{full}) is also the probability for ATx to send an RR-S acknowledging the correct reception of a full window. By considering the probabilities of error free reception of the first, second, . . . last frame of a window being $1-p$, $(1-p)^2$, . . . , $(1-p)^{N_{HDLC}}$ respectively, p_{full} can be derived as:

$$p_{full} = \frac{(1-p)^{N_{HDLC}}}{\sum_{i=1}^{N_{HDLC}} (1-p)^i} = \frac{p(1-p)^{N_{HDLC}}}{(1-p)(1-(1-p)^{N_{HDLC}})} \tag{3.20}$$

The probability of an I-frame being in error (p) is also the probability of sending an REJ-S to request a retransmission from BTx. By combining the probability of sending an RR-S (p_{full}), given in (3.20), the average time t_A consumed on sending S-frames (both RR-S and REJ-S) before each I-frame transmission is:

$$t_A = t_S(p + p_{full}) \tag{3.21}$$

The average time consumed on waiting for S-frames (t_B), is given next. When an error or a successful transmission of a window is detected, an S-frame (either RR-S or REJ-S) must be sent. Due to the stochastic nature of such events, BTx has to wait for a time slot to finish the current transmission before sending an S-frame to acknowledge ATx. The analysis assumes that this waiting time slot is half of an I-frame transmission time $t_I/2$. In addition, t_B should also include propagation delay t_p and S-frame transmission time t_S. Thus, the average time needed to wait for an acknowledgement from BTx is:

$$t_B = \frac{1}{2}t_I + 2t_p + t_S \tag{3.22}$$

From Figure 3.18 (b) and Figure 3.19, the transmission time of an error free window is:

$$t_N = N_{HDLC}(t_I + t_A) + t_B \tag{3.23}$$

When an error has occurred in the $(i+1)^{th}$ frame, where i is the correctly received frames by BRx, the transmission time is given as:

$$t(i) = (i+1)(t_I + t_A) + t_B \tag{3.24}$$

where $0 \le i \le N_{HDLC} - 1$. It should be noted that the window length is i+1 (not N_{HDLC}) since ATx will stop transmitting new I-frames and retransmit the erroneous frame once an REJ-S is received.

The probability of successive i correct I-frame transmissions followed by an error is given by:

$$p_s(i) = (1-p)^i p \tag{3.25}$$

From (3.23)-(3.25), the HDLC throughput F_{HDLC1} for the ATx-BRx channel defined as the correctly received information bits per second is derived as:

$$F_{HDLC1} = \sum_{i=1}^{N_{HDLC}-1} \left(\frac{i \times l_{HDLC}}{t(i)} \times p_s(i) \right) + \frac{N_{HDLC} \times l_{HDLC}}{t_N} (1-p)^{N_{HDLC}} \tag{3.26}$$

The throughput efficiency for the ATx-BRx channel is:

$$TPE = F_{HDLC1} / C \tag{3.27}$$

The average transmission time for one window is given by:

$$t_w = \sum_{i=0}^{N_{HDLC}-1} \left(t(i)(1-p)^i p \right) + (1-p)^{N_{HDLC}} t_N \tag{3.28}$$

The relative time spent on each protocol task affecting the link throughput is given next.

The time interval T_{SS} taken in sending S-frames to BTx-ARx is:

$$T_{SS} = \frac{t_A \sum_{i=0}^{N_{HDLC}-1} \left((i+1)(1-p)^i p \right) + t_A N_{HDLC} (1-p)^{N_{HDLC}}}{t_w} = \frac{t_A (1-(1-p)^{N_{HDLC}}) / p}{t_w} \tag{3.29}$$

The time interval T_{WS} due to waiting for S-frames from BTx is:

$$T_{WS} = \frac{t_B \sum_{i=0}^{N_{HDLC}-1} \left((1-p)^i p \right) + t_B (1-p)^{N_{HDLC}}}{t_w} = \frac{t_B}{t_w} = \frac{\frac{1}{2} t_I + 2t_p + t_S}{t_w} \tag{3.30}$$

The time interval $T_{I'}$ to send the I-frame overheads is:

$$T_{I'} = \frac{\left(\sum_{i=0}^{N_{HDLC}-1} \left((i+1)(1-p)^i p \right) + N_{HDLC} (1-p)^{N_{HDLC}} \right) \frac{l'_{HDLC}}{C}}{t_w} = \frac{(1-(1-p)^{N_{HDLC}}) l'_{HDLC} / pC}{t_w} \tag{3.31}$$

Considering all the time intervals for the protocol tasks, the following equation always holds true:

$$TPE + T_{SS} + T_{WS} + T_{I'} = 1 \tag{3.32}$$

Due to symmetry of the link, the above analysis is also valid for the reverse channel BTx-ARx. The throughput of BTx-ARx is therefore $F_{HDLC2} = F_{HDLC1}$.

3.7.2 Effect of propagation delay using non-optimum window and frame size

The effect of propagation delay on the link throughput is studied in this section. Figure 3.20 shows the throughput efficiency (TPE) against the link distances in the range of 100m to 10km. The bit error rate is set to 10^{-6}, 10^{-5} and 10^{-4} with data rate C equal to 2.5 Gbit/s. Two different combinations of window and frame sizes are used: N_{HDLC}=512, l_{HDLC}=16Kbit and N_{HDLC}=7, l_{HDLC}=5Kbit.

The link has higher throughput when operating at the low BER's. The TPE decreases as the link distance increases. The propagation time t_p which is proportional to the link distance has a significant effect on the TPE for long distance links especially when the BER is low ($p_b = 10^{-6}$). Additionally, t_p has different effects on throughput for different N_{HDLC} and l_{HDLC} combinations. For instance, TPE is only 0.2 at the distance of 10km for N_{HDLC}=7 and l_{HDLC}=5kbit when $p_b = 10^{-6}$, while TPE is 0.7 for N_{HDLC}=512 and l_{HDLC}=16kbit at the same BER. Because RR-S's have to be sent frequently while small N_{HDLC} and l_{HDLC} values are used, the time waiting for RR-S's becomes large when the link distance is large. However, errors are more likely to occur when l_{HDLC} is large and REJ-S's have to be sent frequently. A better throughput could be achieved by implementing the appropriate window and frame size for different BER's as shown in the following.

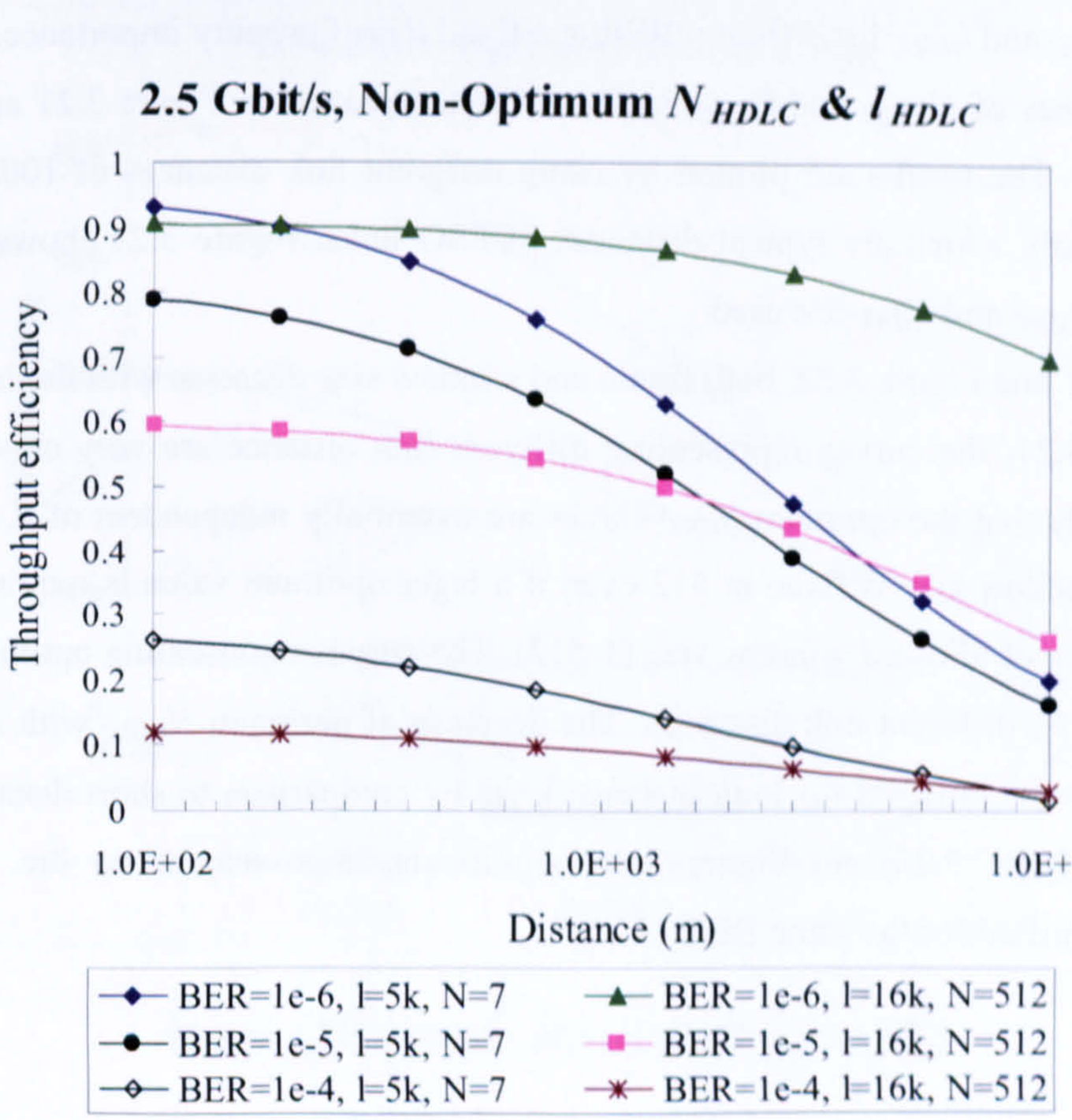


Figure 3.20 TPE against the link distance using non-optimum N_{HDLC} and l_{HDLC} values. l and N are the acronyms for l_{HDLC} and N_{HDLC} respectively

3.8 Optimization of the FSO link layer

3.8.1 Optimizing the frame and window size for different BER and link distance

In order to maximize link throughput, N_{HDLC} and l_{HDLC} values should be optimized for the given BER. The optimum values can be typically obtained by undertaking derivative of F_{HDLC} in equation (3.26). Optimum link throughput is achieved for N_{HDLC} and l_{HDLC} values that satisfy $\frac{\partial F_{HDLC}}{\partial N_{HDLC}} = \frac{\partial F_{HDLC}}{\partial l_{HDLC}} = 0$. Due to the complexity of equation (3.26), an optimum value searching routine is employed to calculate the optimum N_{HDLC} and l_{HDLC} instead of obtaining analytical equations. BER and link distance are used as parameters.

Firstly, the link distance d is kept fixed. The algorithm finds the maximum throughput numerically by cycling N_{HDLC} and l_{HDLC} values in the range allowed by the protocol for a specific BER. It then locates the corresponding optimum N_{HDLC} and l_{HDLC} values for the maximum throughput at the given BER. By repeating this process for different BER, optimum N_{HDLC} and l_{HDLC} values can be obtained for different BER at a fixed d.

Secondly, BER is fixed and the optimum N_{HDLC} and l_{HDLC} values are calculated by using the same process as described above by varying d.

Finally, the optimum values can be obtained by varying BER and d simultaneously. Given the fact that the link distance is normally fixed for the field FSO installations, obtaining the optimum N_{HDLC} and l_{HDLC} for different BER at a fixed d is of primary importance.

Optimum values of N_{HDLC} and l_{HDLC} against BER are plotted in Figure 3.21 and Figure 3.22 respectively. The results are plotted by using different link distances of 100m, 500m, 1km, 2km and 3km, which are typical distances of FSO links. Figure 3.23 shows the TPE when optimum N_{HDLC} and l_{HDLC} are used.

In Figure 3.21 and Figure 3.22, both frame and window size decrease with the increase of BER. In Figure 3.21, the curves representing different link distance are very close to each other, which imply that the optimum l_{HDLC} values are essentially independent of d. In Figure 3.22, optimum window size is fixed at 512 even if a lager optimum value is required due to the restricted range of allowed window size (1-512). The curves representing optimum N_{HDLC} are distinguished by different link distances. The decrease of optimum N_{HDLC} with increasing BER is much more significant for long distance links by comparison to short distance links. By comparing Figure 3.23 and Figure 3.20, significant improvement on the FSO link throughput is identified for the same BER.

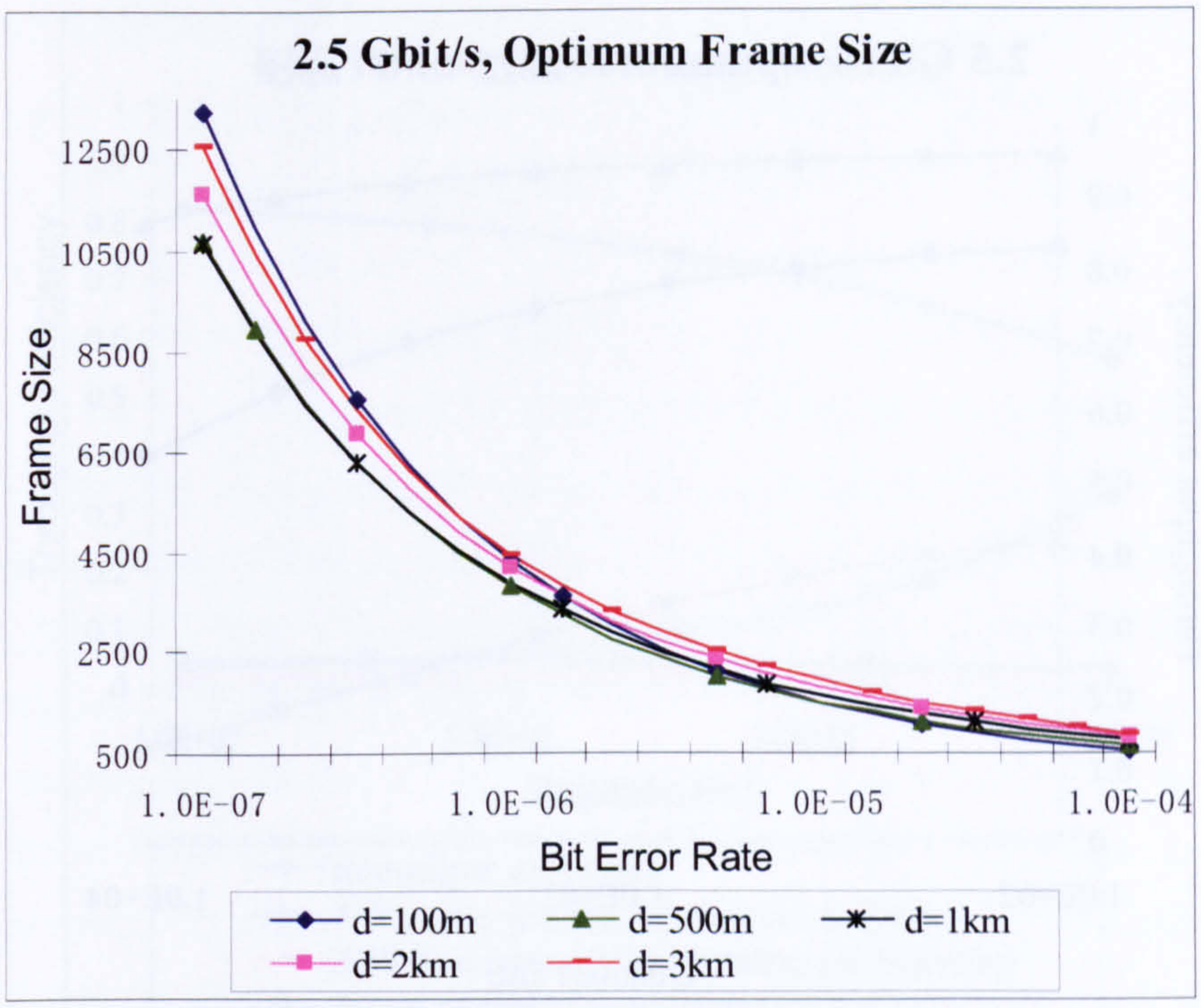


Figure 3.21 Optimum frame sizes against BER

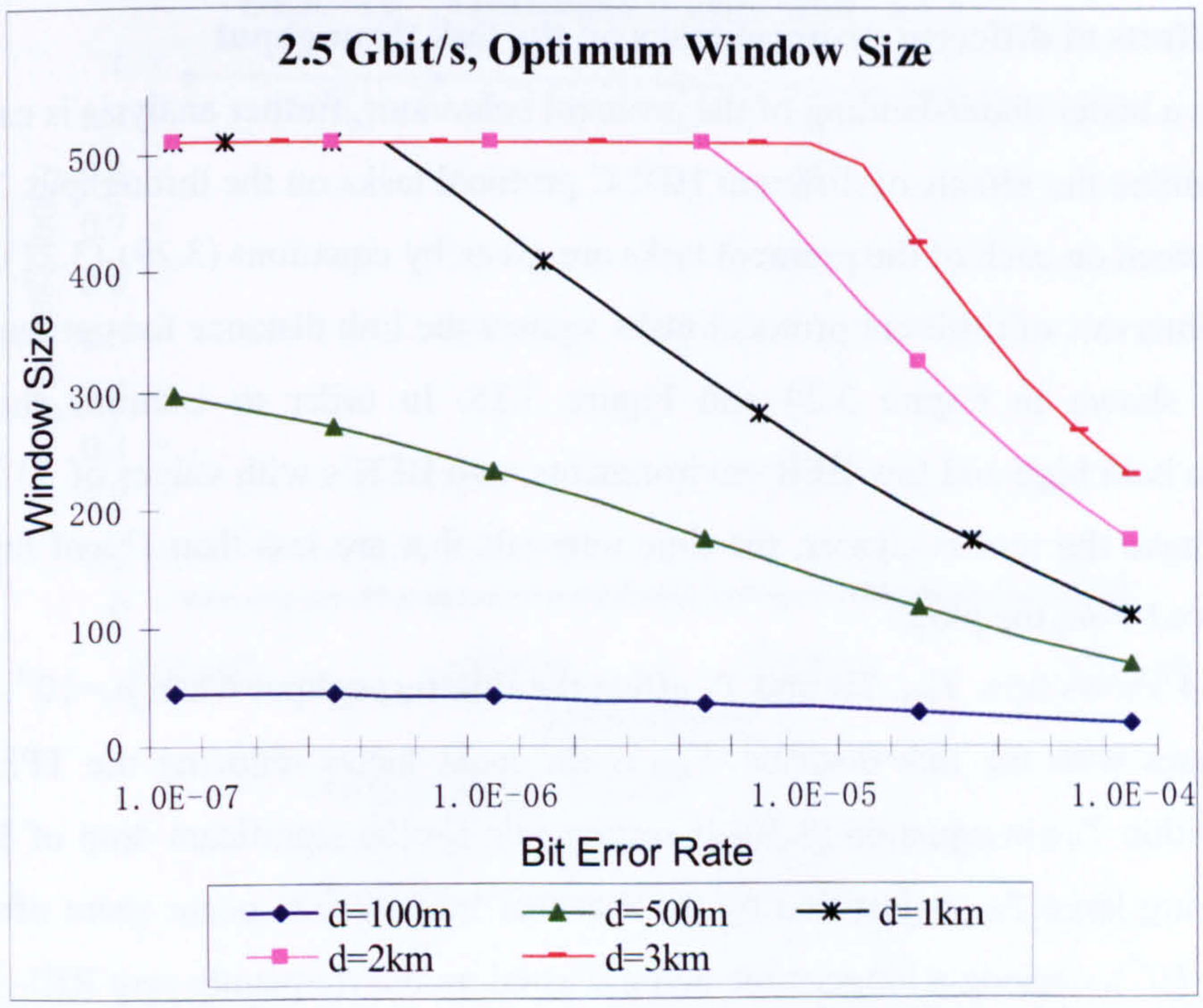


Figure 3.22 Optimum window sizes against BER

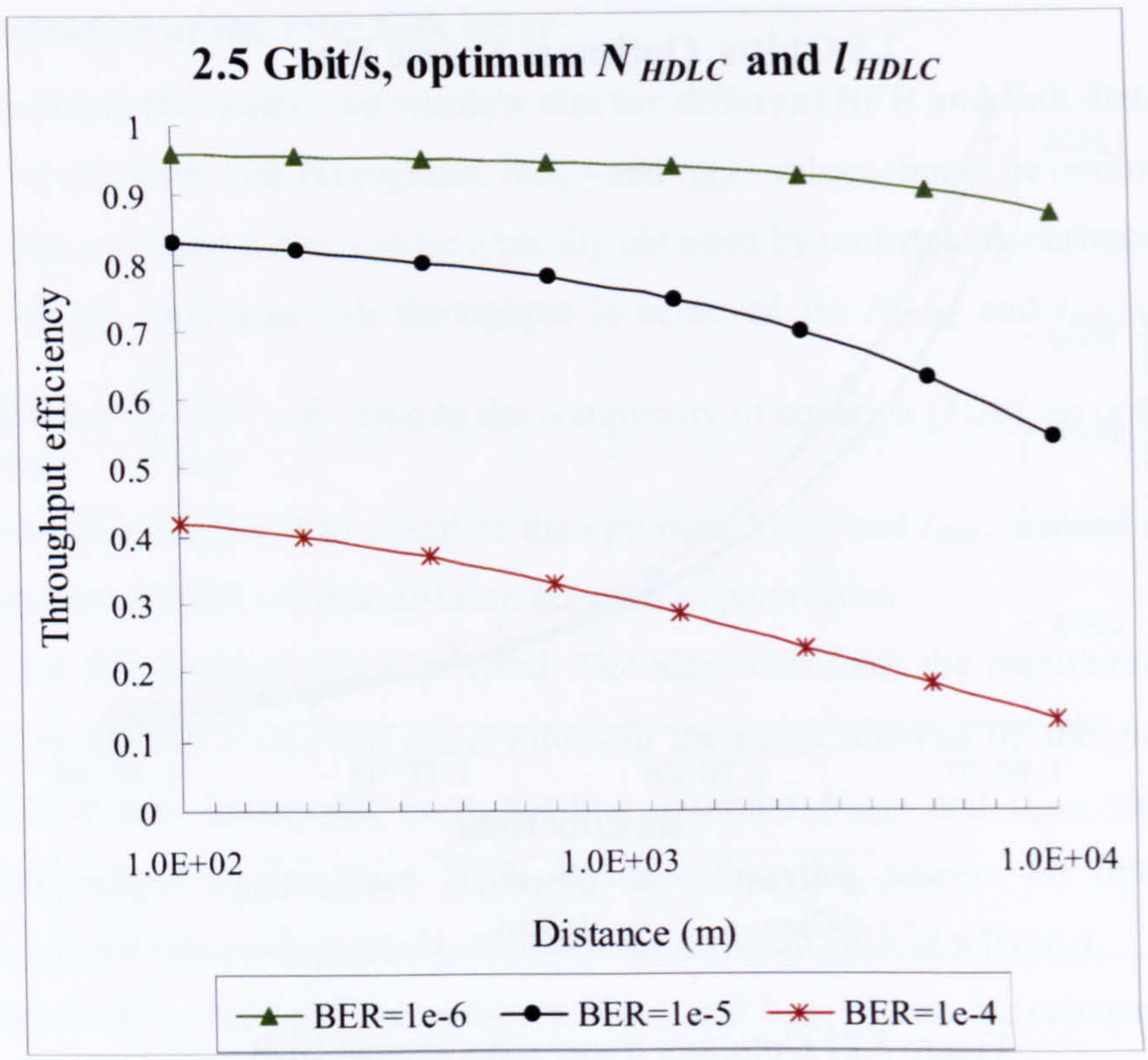


Figure 3.23 TPE against link distance using the optimum window and frame size

3.8.2 The effects of different protocol tasks on the link throughput

To acquire a better understanding of the protocol behaviour, further analysis is carried out in order to examine the effects of different HDLC protocol tasks on the throughput. The time intervals consumed on each of the protocol tasks are given by equations (3.29)-(3.31).

The time intervals of different protocol tasks against the link distance for optimum N_{HDLC} and l_{HDLC} are shown in Figure 3.24 and Figure 3.25. In order to examine the system performance in both high and low BER environments, two BER's with values of 10^{-5} and 10^{-7} are used. To make the results clearer, the time intervals that are less than 1% of the overall time are removed from the plots.

Figure 3.24 shows how T_{WS}, T_{SS} and $T_{I'}$ affect the link throughput when $p_b=10^{-5}$. The T_{WS} interval increases with the link distance. T_{WS} is the main factor reducing the TPE at long distances. t_p within T_{WS} in equation (3.30) is responsible for the significant drop of TPE. The reason for having large T_{WS} is justified by the fact that frame errors occur more often at the high BER ($p_b=10^{-5}$), causing a longer time being wasted on the frequently sent REJ-S's. Both T_{SS} and $T_{I'}$ constantly consume a small portion of the overall time at all link distances and have minor effects on the throughput.

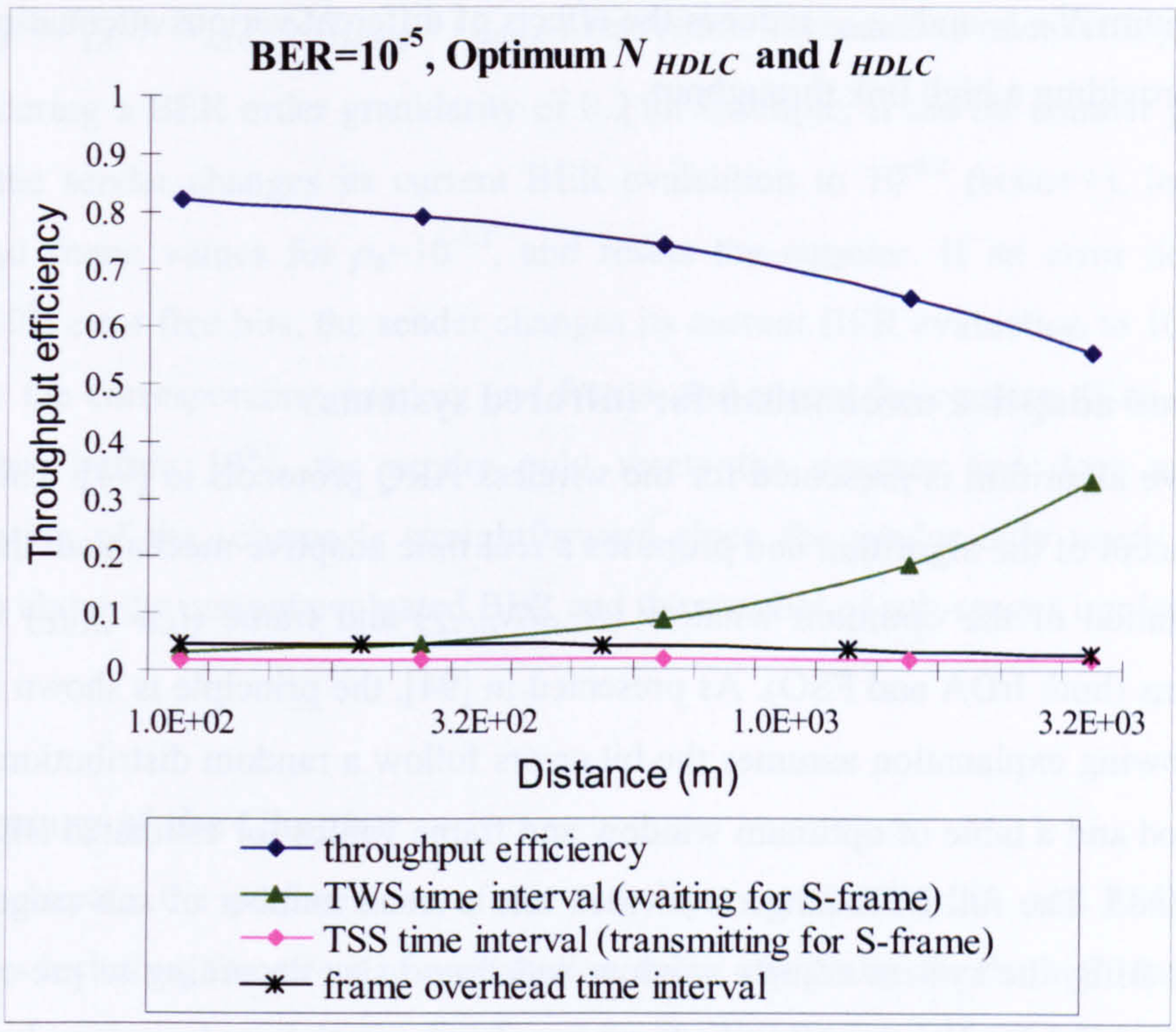


Figure 3.24 Time intervals spent on different protocol tasks when $p_b=10^{-5}$

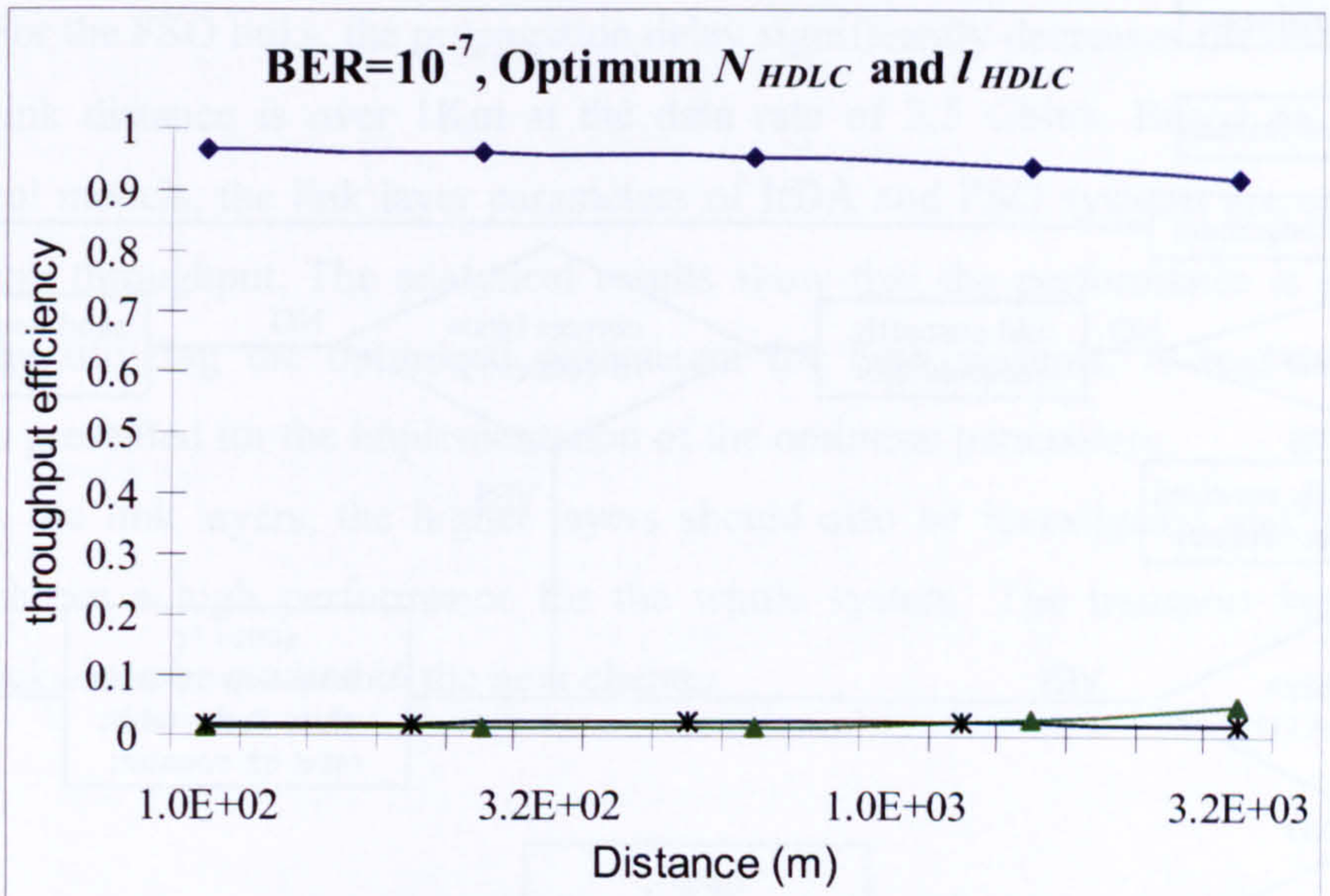


Figure 3.25 Time intervals spent on different protocol tasks when $p_b=10^{-7}$

When BER=10^{-7}, throughput efficiency of over 0.9 is achieved even at d=3km as shown in Figure 3.25. T_{WS} has a much smaller effect on the throughput compared to that in Figure 3.24. Frame overhead also takes a small time interval for all distances. The result shows that

using the optimum N_{IIDLC} and l_{IIDLC} reduces the effects of different various attenuating factors significantly providing a high link throughput.

3.9 Real time adaptive mechanism for infrared systems

An adaptive algorithm is presented for the wireless ARQ protocols in [94]. This section adopts the concept of the algorithm and proposes a real time adaptive mechanism that allows the implementation of the optimum window (N_{LAP}/N_{HDLC}) and frame (l_{LAP}/l_{HDLC}) values to infrared systems (both IrDA and FSO). As presented in [94], the principle is shown in Figure 3.26. The following explanation assumes the bit errors follow a random distribution, the link distance is fixed and a table of optimum window and frame values for estimated BER ranges has been obtained. The full BER range is divided into a small number of sub-ranges. Based on the current BER, the system adjusts window and frame size according to pre-calculated optimum window and frame value tables.

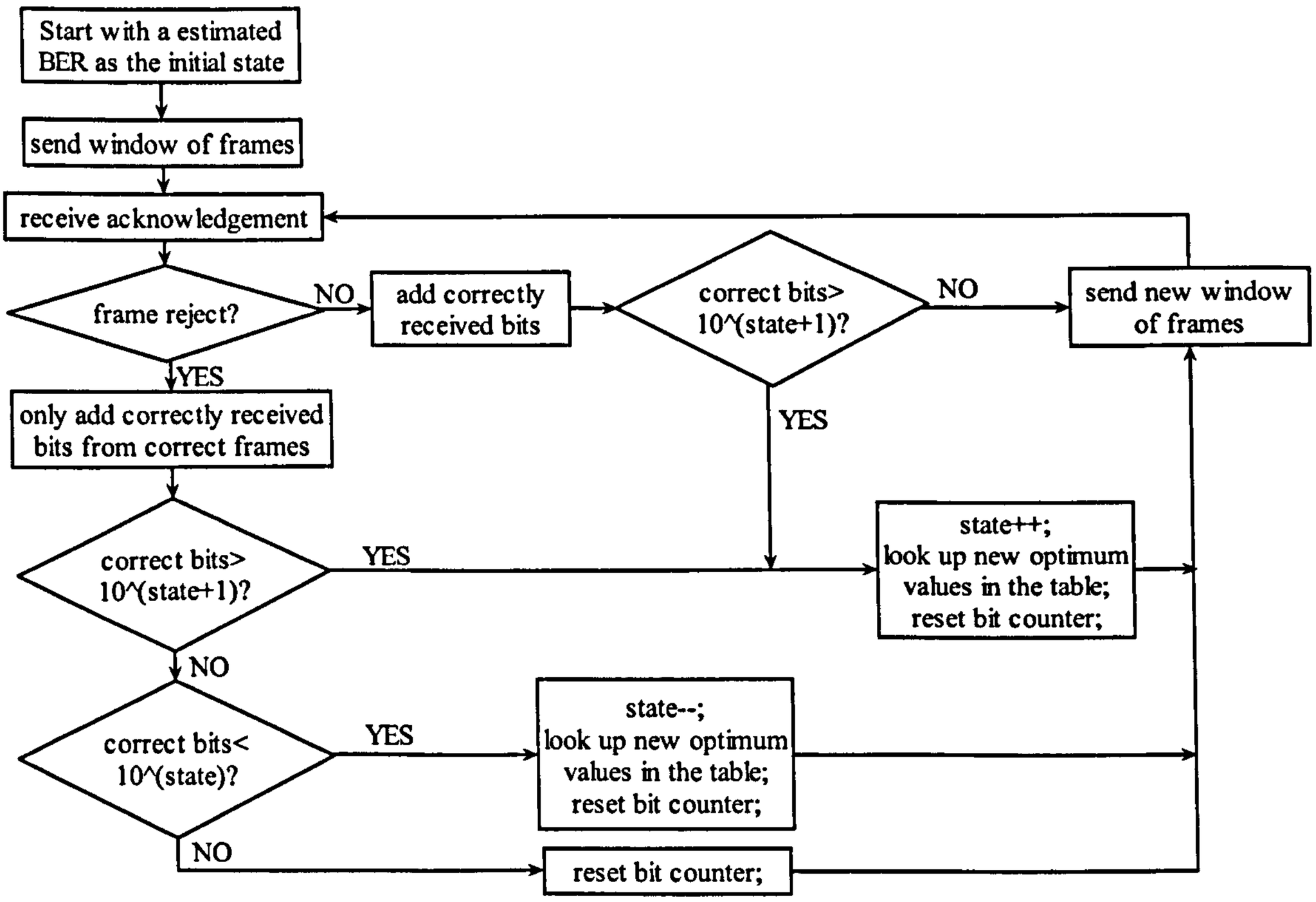


Figure 3.26 Real time optimum window and frame values adaptation scheme

To estimate the BER, the sender counts the frames acknowledged by the receiver and multiplies the number of frames correctly received with the frame size. The product

$N_{LAP} \times (l_{LAP} + l'_{LAP}) / N_{HDLC} \times (l_{HDLC} + l'_{HDLC})$ represents the number of correctly transmitted bits. Considering a BER order granularity of 0.2 for example, if the bit counter goes beyond $10^{6.2}$ bits, the sender changes its current BER evaluation to $10^{-6.2}$ (state++), looks up new window and frame values for p_b=$10^{-6.2}$, and resets the counter. If an error occurs before 1000000 (10^6) error free bits, the sender changes its current BER evaluation to $10^{-5.8}$ (state--), implements the corresponding window and frame, and resets the counter. If an error occurs after 10^6 but before $10^{6.2}$, the sender only resets the counter and does nothing else. Implementation of the scheme is straightforward since the sender only needs to hold the information about the current evaluated BER and the number of sub-ranges implemented.

3.10 Summary of the Chapter

In this chapter, the performance of the link layers of both IrDA and FSO systems is examined by including the effect of propagation delay especially for the high data rate links. Mathematical analytical models are developed for the link layers of both IrDA and FSO. For the IrDA links, the propagation delay does not have significant effect on the link throughput until the IrLAP minimum turn around time is as small as 10^{-7}s and the data rate is as high as 10 Gbit/s. For the FSO links, the propagation delay significantly decreases the link throughput when the link distance is over 1Km at the data rate of 2.5 Gbit/s. Based on the derived mathematical models, the link layer parameters of IrDA and FSO systems are optimized for the maximum throughput. The analytical results show that the performance is significantly improved by utilizing the optimized parameters for both systems. A real-time adaptive algorithm is presented for the implementation of the optimum parameters.

Besides the link layers, the higher layers should also be investigated and optimized in order to achieve a high performance for the whole system. The transport layer of IrDA protocol stacks is to be studied in the next chapter.

Chapter 4. Modelling and Optimizing IrDA Transport layer: TinyTP

4.1 Introduction

All the previous publications that focus on link layer performance assume that data of infinite size is always available and only one application utilizes the link. In practise, upper layers (e.g. TinyTP) offer finite size packets to the link layer at specific time periods due to protocol behaviour and limited buffer size. TinyTP also allows multiple applications to operate the IrDA link concurrently. It is therefore of interest to examine the system throughput at the TinyTP level.

The rest of this chapter is organized as follows: In section 4.1, the detailed functionalities of TinyTP protocol are described. Then, a mathematical model is developed for TinyTP which allows derivation of throughput taking into account the lower IrDA protocol stacks in section 4.2. Section 4.3 elaborates the simulation programs developed for the IrDA protocol stacks. Section 4.4 validates the throughput equations by simulations, as well as carrying out a throughput analysis when different IrLAP parameters and processor speeds are used. Finally, section 4.5 carries out an optimization study to maximize the TinyTP throughput for any given bit error rates by optimizing the major parameters. The optimized parameters include the TinyTP receiver window size, the IrLAP window and frame sizes, as well as the IrLAP minimum turnaround time.

4.2 Tiny Transport Protocol (TinyTP)

TinyTP is a simplified transport protocol serving as a flow control mechanism to work with the IrDA Link Management Protocol (IrLMP) [44]. Even though IrLAP provides reliable data transfer by implementing the GBN error retransmission scheme, TinyTP is still important to ensure the end to end data delivery for the application [146]. This is due to the possible deadlock problem of multiplexed channels introduced by IrLMP Multiplexer (LM-MUX). Reliance on IrLAP to provide flow-control for a multiplexed channel can result in deadlocks if the consumption of data from one multiplexed channel is dependent on data flowing in an adjacent multiplexed channel. Conversely, if inbound data on a multiplexed channel cannot be consumed and the underlying IrLAP connection cannot be flow-controlled due to the possibility of deadlock, inbound data (freshly arrived or buffered) must be discarded in the event of buffer overflow. This reduces the reliable delivery service provided by IrLAP to a best effort delivery service provided by LM-MUX. To overcome this problem TinyTP provides two functions:

- Segmentation and reassembly (SAR).
- Flow control on a per-IrLMP-connection (per-channel) basis.

Since TinyTP runs on top of the IrLMP protocol, it is necessary to briefly introduce IrLMP before looking into the details of TinyTP. IrLMP provides support for multiple software applications or entities to operate independently and concurrently, sharing the single link provided by IrLAP between the transceivers [43]. To realise the link multiplexing, IrLMP provides the LM-MUX service by assigning each application a unique Link Service Access Point (LSAP) address. Figure 4.1 illustrates the relationships between Stations, LSAP's, LSAP Connections, LSAP Connection Endpoints, IrLAP connections and IrLAP connection endpoints. There may be at most one LSAP connection between the same pair of LSAP's. However, a single LSAP may contain several LSAP connection endpoints connected by different LSAP's. LSAP Selector (SEL) is used to distinguish between LSAP's within one IrDA station. An LSAP connection is uniquely identified by the LSAP addresses of the sender and receiver at each end of the connection. IrLMP transfers upper layer data segments based on First In First Out (FIFO) queuing [43].

The following analysis assumes the multiple application channels equally share the link. After the connection initialisation, IrLMP adds a 2-byte header to the upper layer packet providing the LSAP addresses of the sender and the receiver.

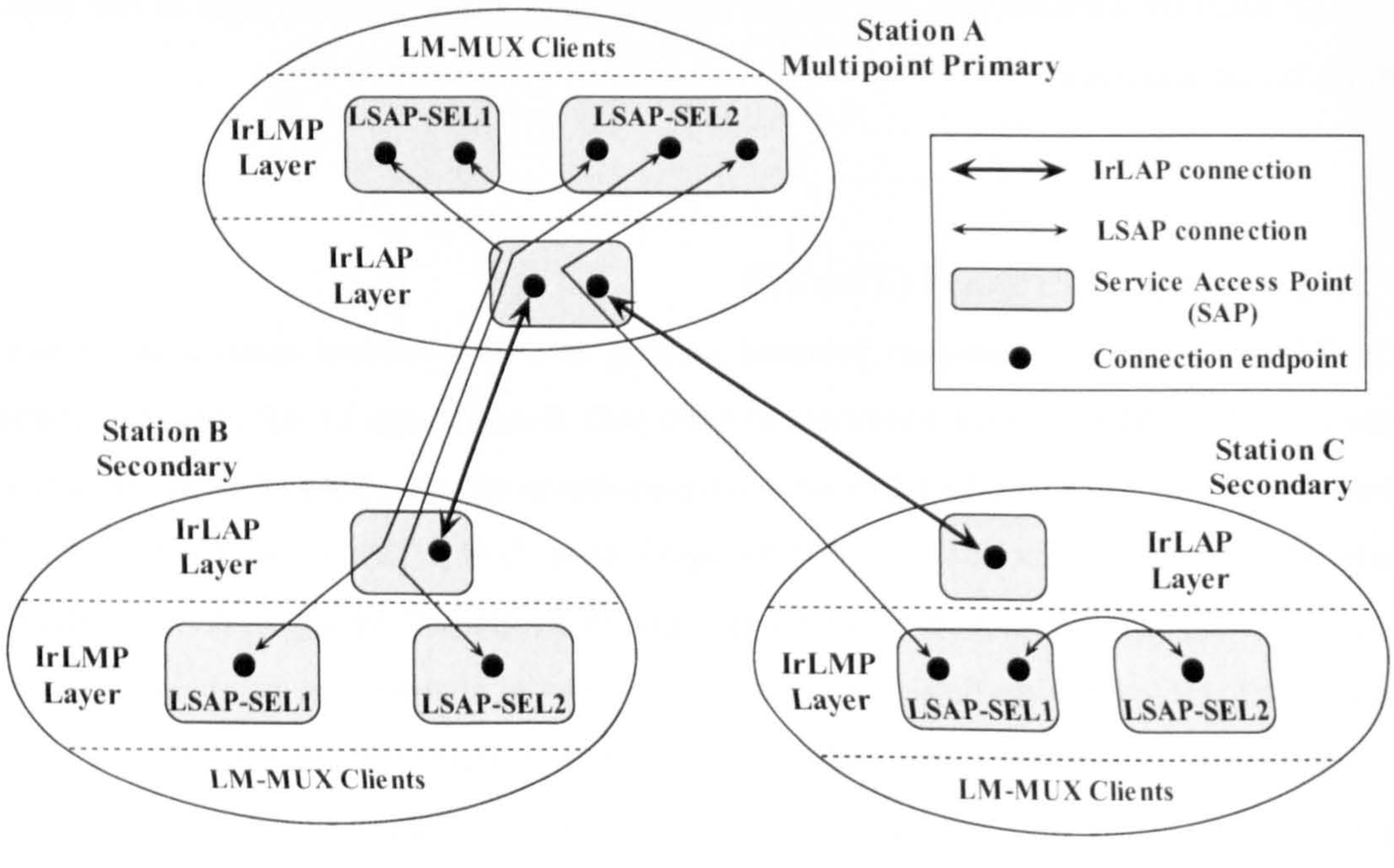


Figure 4.1 Connectivity of LM-MUX and IrLAP

The first the two functions of TinyTP is segmentation and reassembly (SAR). The entire data packet from upper layers, so called Service Data Unit (SDU), can be segmented and

reassembled in TinyTP Packet Data Units (PDU). The maximum PDU size is negotiated at the TinyTP/IrLMP connection establishment. One TinyTP PDU has to fit within one IrLAP frame. Maximum TinyTP PDU size (l_{TTP}) has to satisfy the condition $l_{TTP} \le l_{LAP} - l'_{LMP} - l'_{TTP}$, where l'_{LMP} and l'_{TTP} stand for the headers of IrLMP and TinyTP. In this work, the challenge of having large application files to transmit is considered. To make TinyTP efficient, it is assumed that l_{TTP} is set to its maximum value $l_{TTP} = l_{LAP} - l'_{LMP} - l'_{TTP}$.

The other function of TinyTP is to perform flow control. TinyTP maintains a value of receiver window (w) for each TinyTP channel. The value of w is decided by the TinyTP buffer size of the communication peer. The sender will send PDU if $w>0$ and subtract 1 from w. Therefore, each TinyTP application can send a maximum of w PDU's without receiving acknowledgement but it has to stop when $w=0$. w is updated by the TinyTP acknowledgement from its transmission peer. Following analysis assumes every TinyTP application has the same value w for receiver window.

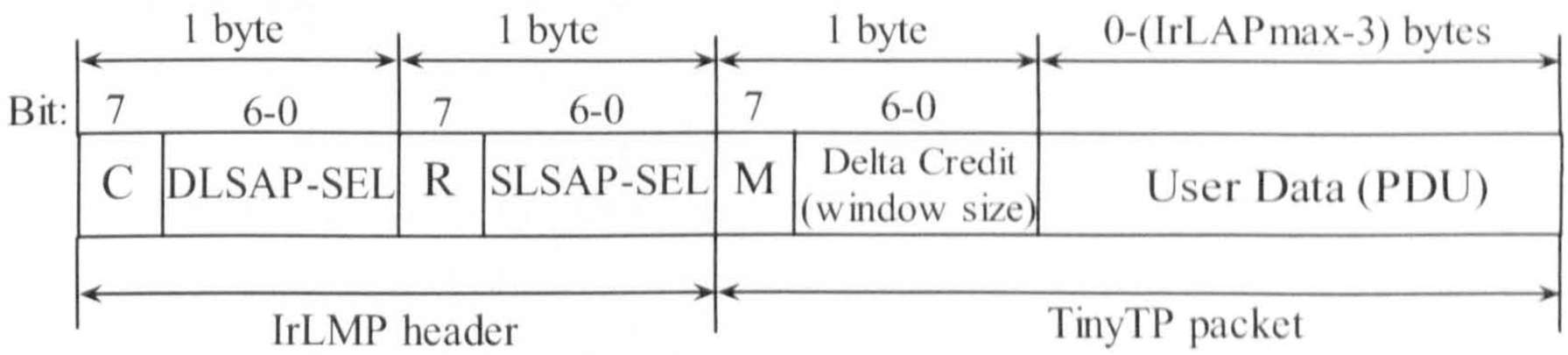


Figure 4.2 IrLMP and TinyTP packet format

The packet format of IrLMP and TinyTP is given in Figure 4.2. The first two bytes are the IrLMP header. C is the control bit. When C is set to 1, it indicates that the packet is a command packet. When C is set to 0, the packet is treated as data. The R bit is reserved for future use and should be set to 0. The DLSAP-SEL and SLSAP-SEL fields contain the LSAP addresses of destination and source respectively. For the TinyTP packet, the first byte is the TinyTP header. M is the More bit and is only significant if SAR has been specified. When M is 0, it indicates that the User Data field contains the final segment of a segmented SDU. Delta Credit field specifies the number (up to 127) of additional data TinyTP PDU's that may be sent in the reverse direction, which is the value of the TinyTP receiver window size. A data TinyTP PDU can carry 1 or more octets of data. A dataless TinyTP PDU has a zero length User Data field. It is always possible to send a dataless TinyTP PDU in order to update the receiver window size of the peer TinyTP entity.

Applications communicate with TinyTP by quoting their TinyTP Service Access Point (TTPSAP). Each TinyTP Service Access Point (TTPSAP) is accessible through single IrLMP LSAP of LM-MUX. A TTPSAP is identified by the address of the LSAP (provided in the IrLMP header) through which it is accessed. After TinyTP connection initialisation, TinyTP adds 1-byte of header carrying information including its window (buffer) size and segmentation status.

A flow chart of the data transmission with multiple TinyTP connections is illustrated in Figure 4.3. At the sender, TinyTP first segments the application packet (i.e. OBEX packet) into TinyTP PDU's. The PDU's are then multiplexed by IrLMP and forwarded to IrLAP for transmission. At the receiver, IrLAP delivers the received frames to IrLMP. IrLMP demultiplexes the IrLAP frames and forwards them to TinyTP. TinyTP reassembles the received PDU's and places them into buffer for the application to process. The acknowledgements of TinyTP and IrLAP are then sent back to the sender.

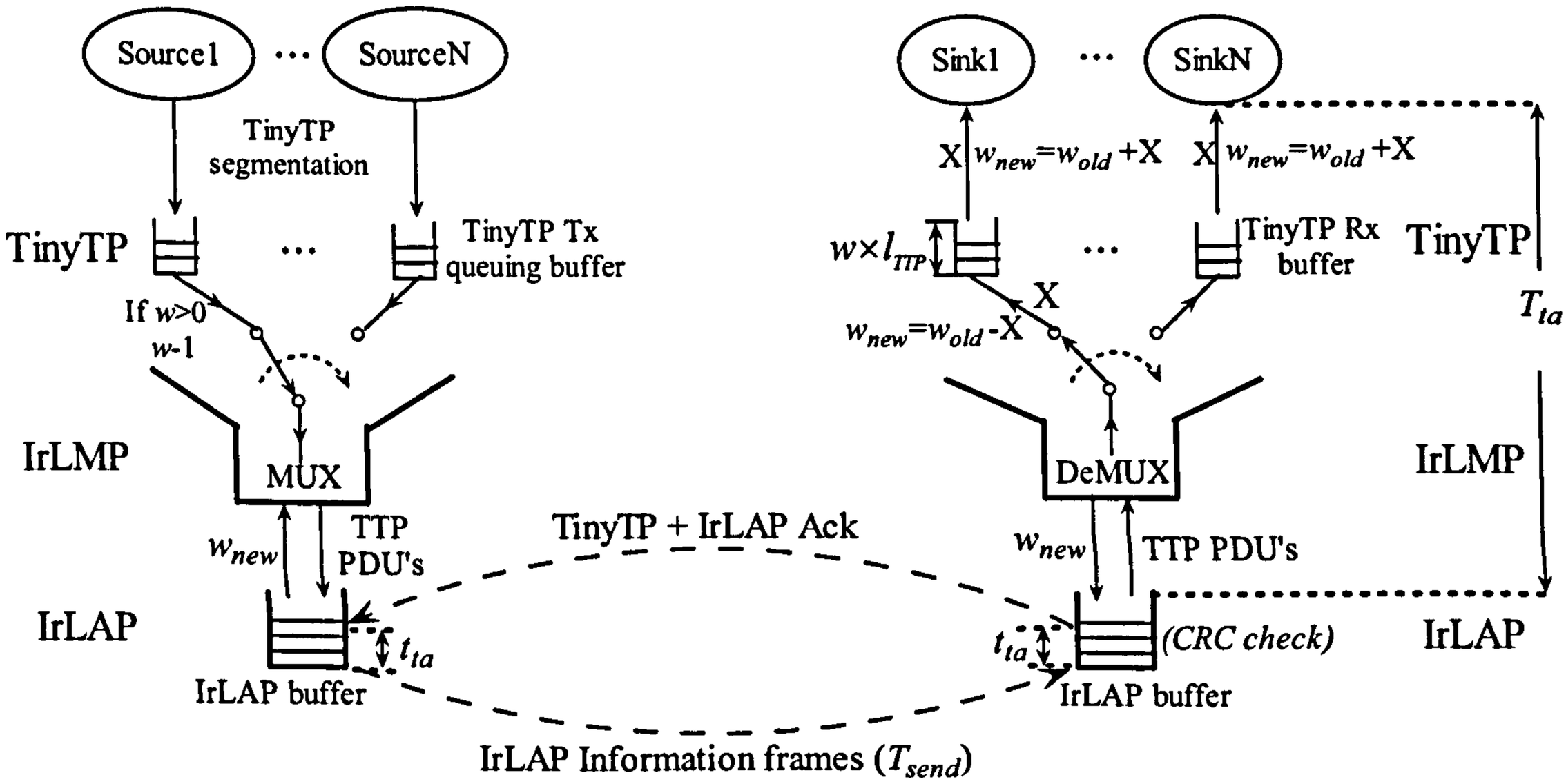


Figure 4.3 TinyTP data transmission

4.3 Analytical model of TinyTP

The TinyTP mathematical model assumes large application files (e.g. mp3, movie clip) to be sent from the primary to the secondary [143]. TinyTP PDU's therefore are always at the maximum size ($l_{TTP} = l_{LAP} - l'_{LMP} - l'_{TTP}$) to accommodate the application data. The 'connected' TinyTP PDU's (excluding the connection establishment and termination PDU's) are

considered in the derivation. Therefore, IrLMP and TinyTP have fixed headers of 2 bytes and 1 byte respectively. Table 3-1 and Table 4-1 list the details of the symbols that are used in the analysis.

Symbol	Parameter Description	Unit
B	Number of TinyTP connections	-
w	TinyTP receiver window size/ buffer size	-
l_{TTP}	Maximum TinyTP PDU size, $l_{TTP} = l_{LAP} - l'_{LMP} - l'_{TTP}$	bit
l'_{LMP}	IrLMP header	bit
l'_{TTP}	TinyTP header	bit
T_{ack}	Time for transmitting the TinyTP acknowledgement	sec
T_{ta}	Time for TinyTP to process the received PDU's and prepare the acknowledgement	sec

Table 4-1 Parameters used in the modelling

4.3.1 IrLAP Modelling

In order to derive the throughput at the TinyTP layer, the average time to successfully transmit an IrLAP window is to be identified first. As demonstrated in Chapter 3, the effect of propagation time for the IrDA link on throughput is minor unless the link data rate reaches the order of Gbit/s. Since only the data rates that are currently supported by the IrDA standard (up to 16 Mbit/s) will be examined in this Chapter, the propagation time is not included in the mathematical modelling in order to simplify the analysis. According to Figure 3.2, the average time to successfully transmit one IrLAP window consists of the time for I-frame transmissions, acknowledgement transmissions and retransmissions, as well as delays for reversing the link t_{ta} and timer time outs t_{Fout}. The average time to transmit one IrLAP window with a length of A I-frames is given as:

$$t_A = At_I + p(t_{Fout} + t_s) + t_{ack} + 2t_{ta} \tag{4.1}$$

where A is the length of an IrLAP window, the IrLAP link parameters t_s, t_I, t_{ack}, p and t_{Fout} are derived in section 3.2

The probability of having errors in an IrLAP window with A frames is:

$$p_1 = 1 - (1-p)^A \tag{4.2}$$

Due to the small value of p, p_1 can be approximated as:

$$p_1 = 1 - (1-p)^A \approx 1 - (1 - Ap) = Ap \tag{4.3}$$

When errors occur in transmitting an IrLAP window with probability p_1, due to the stochastic nature of error occurrence, it is sufficient to assume that on average the error occurs in the

middle of the window, and retransmission with window length of $0.5A$ will trigger to recover the error. If further error/errors occur in the retransmission (with probability of $p_2 = p_1(1-(1-p)^{0.5A}) \approx 0.5A^2p^2$), another retransmission window is needed with window length of half the previous, i.e. $0.25A$, and this process continues. When the retransmission window is less than 1, it is considered that the whole window has been successfully transmitted. By including the first window transmission and all the retransmissions, the average time to successfully transmit the IrLAP window is given as:

$$\begin{aligned} T_{send}(A) &= t_A + p_1(\frac{1}{2}At_I + p(t_{Fout}+t_s)+t_{ack}+2t_{ta}) + \cdots + p_X(\frac{1}{2^X}At_I + p(t_{Fout}+t_s)+t_{ack}+2t_{ta}) \\ &= \left(1+\frac{1}{2}Ap+\cdots+\left(\frac{1}{2}\right)^{\frac{1}{2}X(X+1)}A^Xp^X\right)At_I + \left(1+Ap+\cdots+\left(\frac{1}{2}\right)^{\frac{1}{2}X(X-1)}A^Xp^X\right)(p(t_{Fout}+t_s)+t_{ack}+2t_{ta}) \\ &= \left(1+\sum_{i=1}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i+1)}(Ap)^i\right)\right)At_I + \left(1+Ap+\sum_{i=2}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i-1)}(Ap)^i\right)\right)(p(t_{Fout}+t_s)+t_{ack}+2t_{ta}) \end{aligned} \quad (4.4)$$

where X is an integer representing the number of possible retransmissions ($X = \lfloor \log_2 A \rfloor$). X satisfies the length of the retransmission window to be larger than or equal to 1 I-frame ($\frac{1}{2^X} \cdot A \geq 1$).

4.3.2 Derivation of the TinyTP throughput

Before deriving the TinyTP throughput, we first discuss two TinyTP parameters: the time to transmit a TinyTP acknowledgement (T_{ack}) and the time to hold the incoming TinyTP PDU's in the buffer (T_{ta}).

As described in section 4.1, TinyTP acknowledgement only provides the information about the updated secondary receiver window size. Therefore, the secondary simply sends the TinyTP header l'_{TTP} as the TinyTP acknowledgement. By including the headers of the other layers, the transmission time of the TinyTP acknowledgement is given by:

$$T_{ack} = \frac{l'_{LAP}+l'_{LMP}+l'_{TTP}}{C} \quad (4.5)$$

The time to hold the incoming TinyTP PDU's in the buffer (T_{ta}) is the time interval from passing the IrLAP frames to IrLMP to getting the TinyTP acknowledgement ready at the IrLAP layer. As shown in Figure 4.3, T_{ta} includes the time to process and strip the headers all the way up to TinyTP, the time to process the TinyTP PDU's and empty the TinyTP buffer, as well as preparing TinyTP acknowledgement and adding the headers of other layers. The time to process the TinyTP PDU's (T_p) varies for different applications and processors. The

analysis assumes that the IrDA device uses an 8-bit processor and each 8-bit data requires 2 CPU cycles in average. Since T_p is the major factor of T_{ta}, we assume $T_{ta} \approx T_p$:

$$T_{ta} \approx T_p = \frac{2Al_{TTP}}{8v} = \frac{Al_{TTP}}{4v} \tag{4.6}$$

where A is the incoming IrLAP window size and v is the processor speed in Hz.

When a TinyTP receiver window size w is allocated for each of the B TinyTP connections, the IrDA receiver has to assign a TinyTP buffer with size of $B \times w \times l_{TTP}$. Due to the fact that many of the IrDA enabled devices are resource-limited, such devices often can not provide large memory size for TinyTP. For a given maximum IrLAP window size N_{LAP}, three possible scenarios with different receiver window sizes are considered. B denotes the number of TinyTP connections. If it is assumed that the TinyTP connections equally share the IrDA link as IrLMP delivers data based on FIFO queuing, the following three scenarios are possible:

A. $Bw \leq N_{LAP}$

The TinyTP transmission model is illustrated in Figure 4.4 by mapping TinyTP PDU's into IrLAP frames. In Figure 4.4, parameters w=2, B=2 and $N_{LAP} \geq 4$ are employed to satisfy the condition of $Bw \leq N_{LAP}$. The IrLAP window will be always less than 4 due to the w constraint. Since the Delta Credit is 0 after transmitting the first IrLAP window, the primary can not transmit more TinyTP PDU's until the secondary finishes processing the received PDU's and informs the primary of the updated receiver window size. Since the time to process the received PDU's and prepare the TinyTP acknowledgement packet (T_{ta}) depends on the CPU speed of the receiver/secondary, it is typically much longer than IrDA link turnaround t_{ta} and the time to transmit the IrLAP acknowledgement. Thus, it is sufficient to assume $T_{ta} > t_{ta} + t_{ack}$. After IrLAP successfully delivers the IrLAP frames, the secondary has to wait T_{ta} before the TinyTP acknowledgement gets ready. Since there are two TinyTP connections in Figure 4.4, the secondary needs to send two TinyTP acknowledgements. Another window will be sent from the primary following the same routine. It is sufficient to analyse one window transmission for the TinyTP throughput derivation.

As shown in Figure 4.4, and from equation (4.4), the average time for one TinyTP window transmission T_1 is given by:

$$T_1 = T_{send}(Bw) + T_{ta} + BT_{ack} + t_{ta} \tag{4.7}$$

where w is the receiver window size.

The TinyTP throughput which is defined as information bits per second is:

$$F_{TTP} = \frac{Bw \times l_{TTP}}{T_1} \tag{4.8}$$

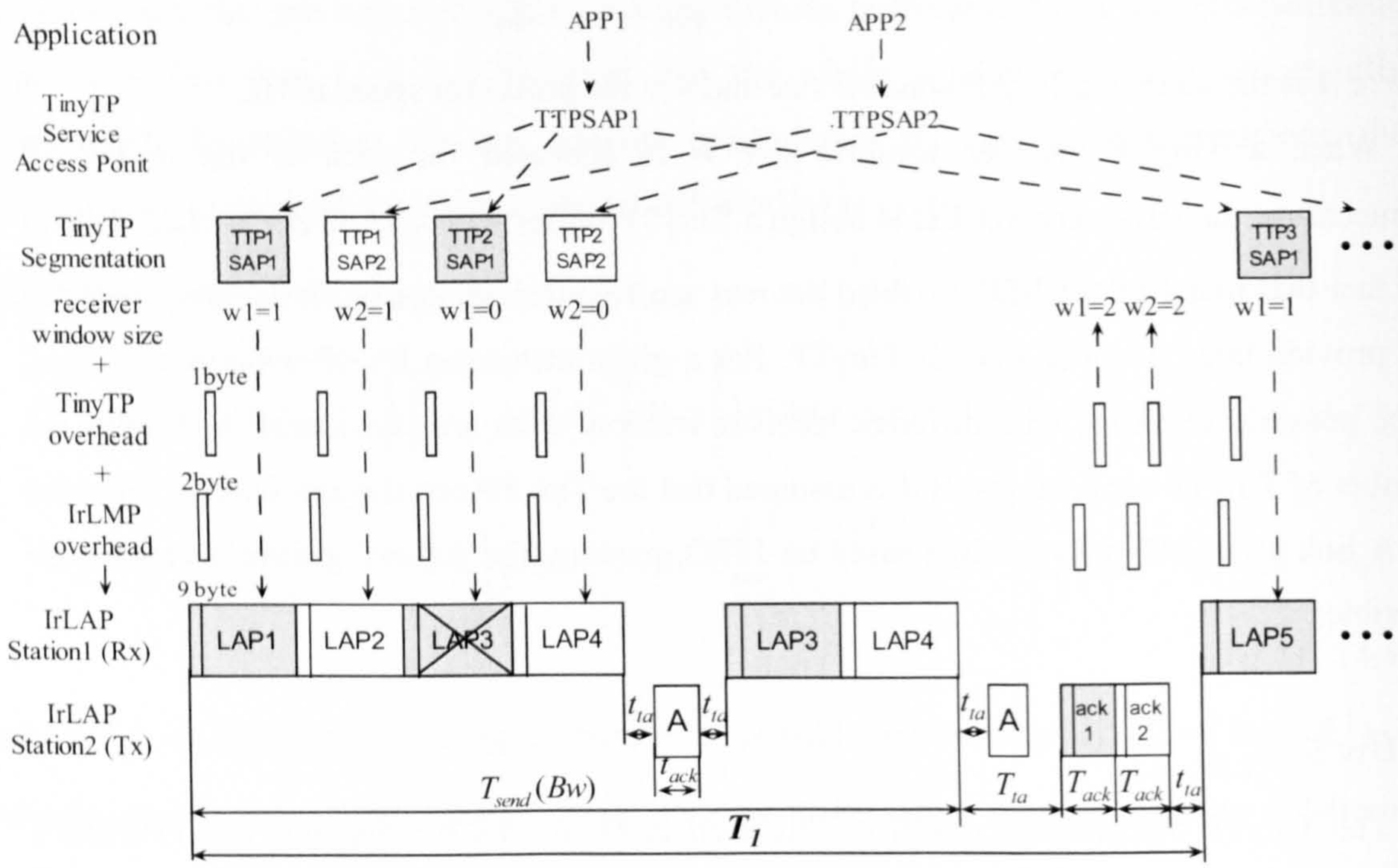


Figure 4.4 TinyTP transmission model with $Bw \leq N_{LAP}$; initial state: $w1$=$w2$=2. A is the IrLAP acknowledgement and ack is the TinyTP acknowledgement. IrLAP frame LAP3 is corrupted in the transmission. LAP3 and LAP4 are retransmitted in the next IrLAP window.

B. $N_{LAP} < Bw < 2N_{LAP}$

The TinyTP transmission model with w=3, B=2 and N_{LAP}=4 satisfying $N_{LAP} < Bw < 2N_{LAP}$ is illustrated in Figure 4.5. The first TinyTP window has 4 PDU's and makes use of maximum IrLAP window length. Since the secondary is fed by 4 TinyTP PDU's and has no time to process, the secondary will send 2 TinyTP acknowledgements to give the information of available buffer size for each application. In this case, the secondary acknowledges the primary with $w1$=$w2$=1 (available buffer size subtracted from the incoming PDU's, 3-2=1). The primary is then able to send 2 PDU's in the second window. Assuming the 4 TinyTP PDU's of the last window have been processed and consumed, each of the receiver windows equals 2 (3-1=2). The secondary then acknowledges with $w1$=$w2$=2. As the same process will be repeated, analysing two window transmissions is sufficient for deriving the TinyTP throughput.

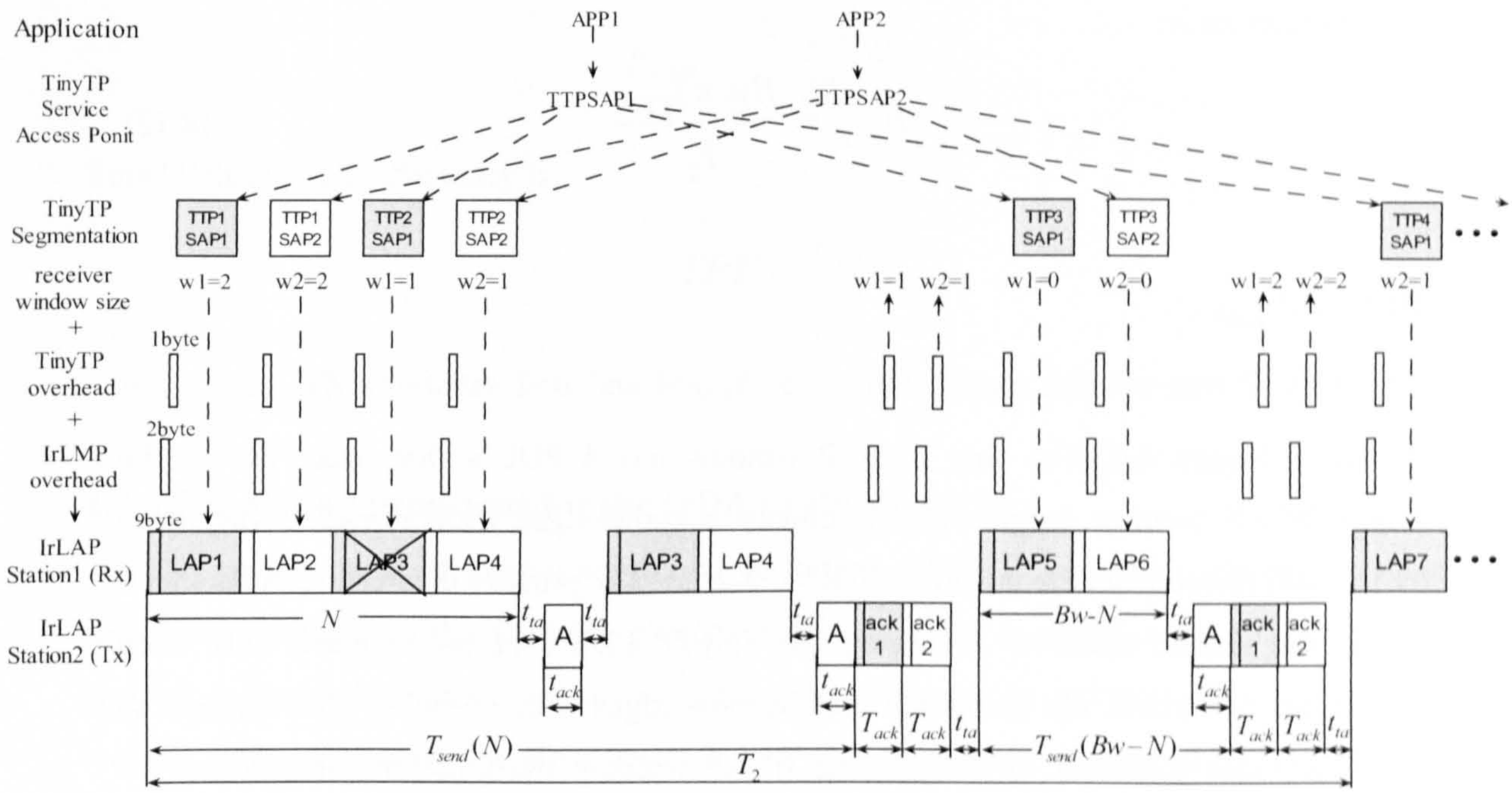


Figure 4.5 TinyTP transmission model when $N_{LAP} < Bw < 2N_{LAP}$, initial state: $w1=w2=3$

By using equation (4.4), the average transmission time for the first and the second IrLAP window is given in (4.9) and (4.10):

$$T_{send}(N_{LAP}) = \left(1+\sum_{i=1}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i+1)}(N_{LAP}p)^i\right)\right)N_{LAP}t_1 + \left(1+N_{LAP}p+\sum_{i=2}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i-1)}(N_{LAP}p)^i\right)\right)(p(t_{Fout}+t_s)+t_{ack}+2t_{ta}) \tag{4.9}$$

$$T_{send}(w-N_{LAP}) = \left(1+\sum_{i=1}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i+1)}((w-N_{LAP})p)^i\right)\right)(w-N_{LAP})t_1 + \left(1+(w-N_{LAP})p+\sum_{i=2}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i-1)}((w-N_{LAP})p)^i\right)\right)(p(t_{Fout}+t_s)+t_{ack}+2t_{ta}) \tag{4.10}$$

From Figure 4.5 and equation (4.4), the average time for one TinyTP window transmission T_2 is given as:

$$\begin{aligned} T_2 &= T_{send}(N_{LAP}) + BT_{ack} + t_{ta} + T_{send}(Bw-N_{LAP}) + BT_{ack} + t_{ta} \\ &= T_{send}(N_{LAP}) + T_{send}(Bw-N_{LAP}) + 2BT_{ack} + 2t_{ta} \end{aligned} \tag{4.11}$$

TinyTP throughput is:

$$F_{TTP} = \frac{Bw \times l_{TTP}}{T_2} \tag{4.12}$$

c. $Bw \geq 2N_{LAP}$

The TinyTP transmission model with w=5, N_{LAP}=4 and B=2 satisfying $Bw \geq 2N_{LAP}$ is illustrated in Figure 4.6. The first TinyTP window has 4 PDU's that make use of the maximum IrLAP window length. The secondary acknowledges with $w1$=$w2$=3 (available buffer size subtracted from the incoming PDU's, 5-2=3). The primary then can send another 4 PDU's in the second window. Assuming the TinyTP PDU's of last window have been processed and consumed, the secondary then acknowledges with $w1$=$w2$=3 (5-2=3), and so on. It is therefore sufficient to analyse one IrLAP window transmission for deriving the TinyTP throughput.

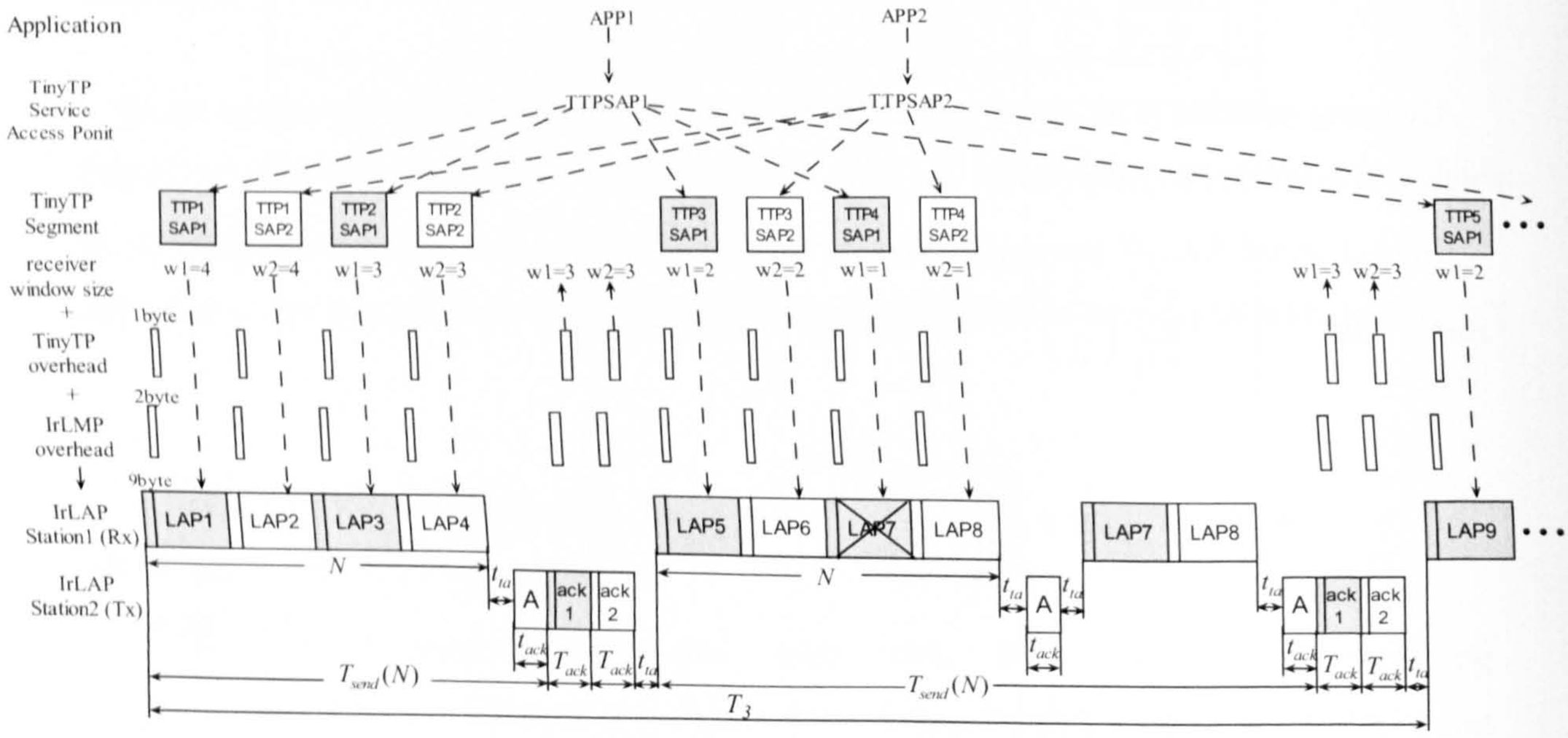


Figure 4.6 TinyTP transmission model when $Bw \geq 2N_{LAP}$, initial state: $w1$=$w2$=5

From Figure 4.6, each of the IrLAP windows has a maximum length of N_{LAP} frame. The average transmission time for one TinyTP window is given as:

$$T_3 = T_{send}(N_{LAP}) + BT_{ack} + t_{ta} \tag{4.13}$$

The TinyTP throughput is given as:

$$F_{TTP} = \frac{N_{LAP} l_{TTP}}{T_3} \tag{4.14}$$

TinyTP throughput efficiency is:

$$TPE = \frac{F_{TTP}}{C} \tag{4.15}$$

4.4 Simulation programs for the IrDA protocol stacks

The major network simulation tools, including OPNET and NS2, do not provide simulation modules for the full IrDA protocol stacks. In order to examine the accuracy of the derived mathematical model, a set of simulation programs is developed to emulate the realistic infrared communication channel and the behaviour of the IrDA stacks. The simulation programs are written in the MATLAB™ programming language [36]. The source code of the programs is listed in Appendix II. In [79], Barker developed and subsequently verified an OPNET model for the IrDA physical and data link layers. In order to validate the accuracy of the developed MATLAB™ simulation program presented in this section, the results from both simulation programs are compared. Since these two simulation programs yield identical results, the developed MATLAB™ simulation program is validated.

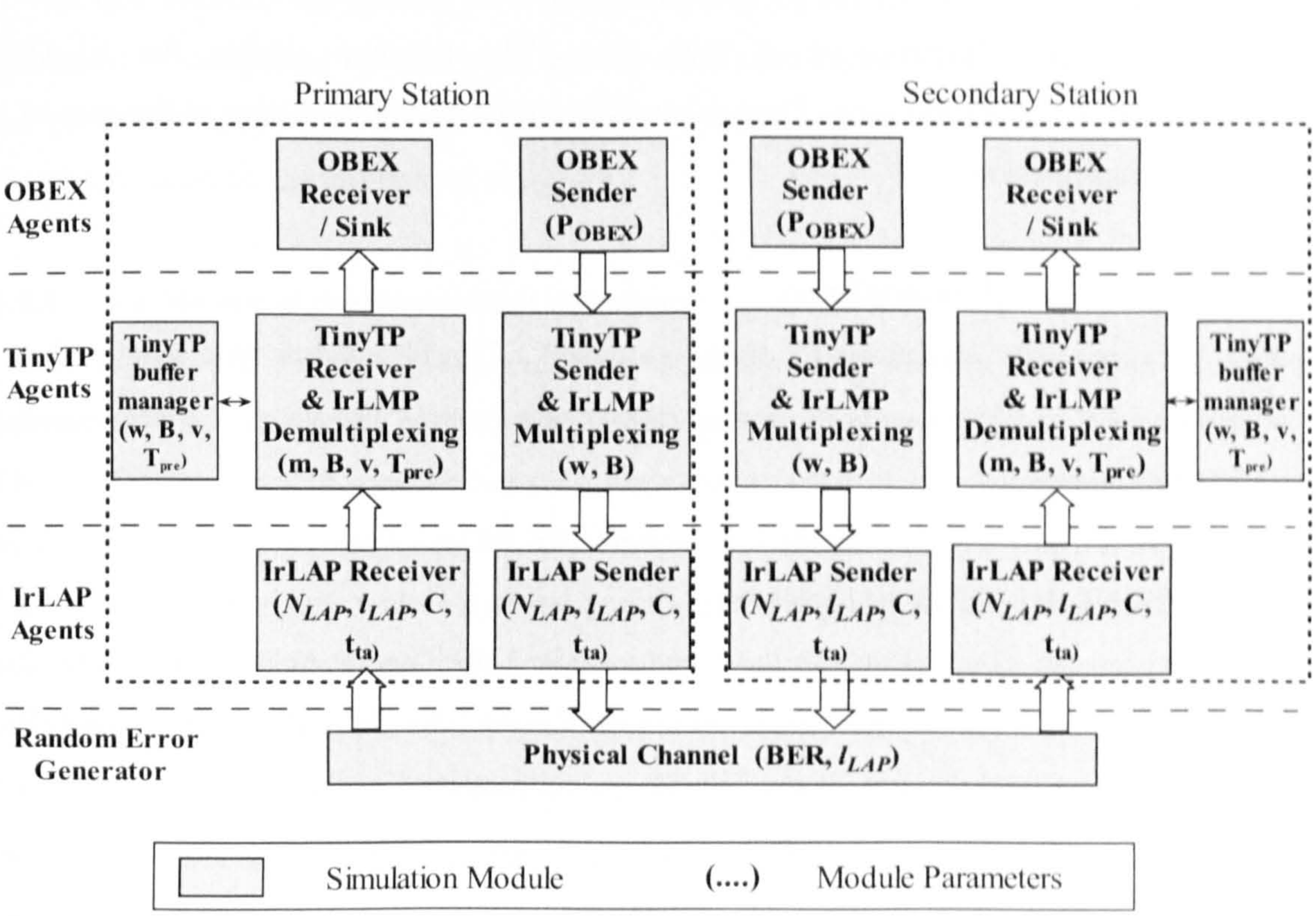


Figure 4.7 Structure of the simulation program for IrDA protocol stacks

As shown in Figure 4.7, the simulation programs consist of 3 different 'agents' for the IrLAP, TinyTP and OBEX, as well as a random error generator acting as the physical channel. Each agent has its sub-agents/simulation modules. A brief description of different simulation modules follows.

Physical channel:

Function `'physical'` is the IrDA Physical layer. Based on the given average bit error probability and the length of the IrLAP frame, it randomly generates error to emulate the nature of infrared communication channel. The function returns a Boolean value. 0 indicates the error free frame transmission, while 1 indicates the frame corruption.

IrLAP sender:

Function `'irlapsender'` calculates the time to transmit an IrLAP window. By using the function `'physical'`, `'irlapsender'` considers the channel error, the I-frame transmission and the possible supervision frame transmission. The function returns the IrLAP window transmission time (Tlapsend).

IrLAP receiver:

Function `'irlapreceiver'` calculates the time to transmit the IrLAP ack and erroneous frame number if the transmission is with error. The function returns the IrLAP acknowledgement transmission time (Tlapack) and the corrupted frame number in the case of the erroneous transmission.

TinyTP sender & IrLMP multiplexer:

Function `'ttpsender'` calculates the time to transmit a TinyTP window with length of m by calling function `'irlapsender'` and `'irlapreceiver'`. The window transmission time (Tttpsend) is returned.

TinyTP receiver & IrLMP demultiplexer:

Function `'ttpreceiver'` calculates the time to transmit the TinyTP Acks and the TinyTP buffer size at the receiver side. The TinyTP Ack transmission time (Tttpack), the new TinyTP window size (Wnew) and the size of the TinyTP packet remained in the buffer (numberbN) are returned.

TinyTP buffer manager:

Function `'ttpbuffer'` calculates the new TinyTP buffer size (Wnew) and the length of TinyTP packets in the buffer (numberb) for each TinyTP connection. The function returns the new TinyTP window size (Wnew), the bits remain in buffer (numbcrb) and the proccssing time (Tprocess).

OBEX sender/Sink:

Function `'obexsender'` uses 'ttpsender' and 'ttpreceiver' to calculate the time to transmit an OBEX packet window with length of l_{obex} (Tobexsend). The function returns Tobcxsend.

OBEX receiver:

Function `'obexreceiver'` calculates and returns the time to transmit the OBEX acknowledgements (Tobexack) by using `'ttppsender'` and `'ttpreceiver'`.

4.5 Validation of the throughput equations and analysis of the TinyTP throughput

In this section, the accuracy of the throughout equations (4.8), (4.12) and (4.14) is examined by comparing the analytical and the simulation results. Subsequently, the effects of the processor speed and IrLAP window and frame sizes on the TinyTP throughput are examined based on the throughput equations.

4.5.1 Validation of the throughput equations

In Figure 4.8, TinyTP throughput efficiencies are compared by using the throughput equations of the analytical method and the computer simulation discussed in section 4.3. Three different receiver window (w) sizes are chosen in the plot representing three different buffer situations. The maximum IrLAP window and frame size are fixed at N_{LAP}=20 and l_{LAP}=16Kbit respectively with additional parameters set to C=16Mbit/s, v=10MHz, t_{ta}=10^{-4}s and B=2. Unless otherwise specified, the same values of C, v, t_{ta} and B are used throughout the analysis. The throughput efficiencies are plotted against the bit error rate in the range of 10^{-5} to 10^{-8}. The mean ($\bar{x}$) of the throughput efficiency difference between the analytical and the simulation results and the standard deviation (σ) of the throughput efficiency difference are also calculated by using equations (3.17)-(3.19).

Figure 4.8 shows that the analytical results match the simulation results very well and the corresponding values of the mean difference and the standard deviation are very small. This

verifies the mathematical model derived in section 4.2. All of the three throughput efficiencies (TPE) decrease with the increase of the bit error rate (BER). In the case of w=15 and 30, the system obtains much better TPE's than when w=5 especially for the low BER ($TPE > 0.8$). The TPE for w=30 is slightly better than for w=15. The figure shows that the system achieves the best possible throughput for any bit error rate by using a receiver window size at least twice as the maximum IrLAP window size ($Bw \geq 2N_{LAP}$). A good TinyTP throughput level is also reached by using w=15 ($N_{LAP} < Bw < 2N_{LAP}$). A receiver window size in the range of $N_{LAP} < Bw < 2N_{LAP}$ therefore provides good system performance for a relatively smaller buffer size.

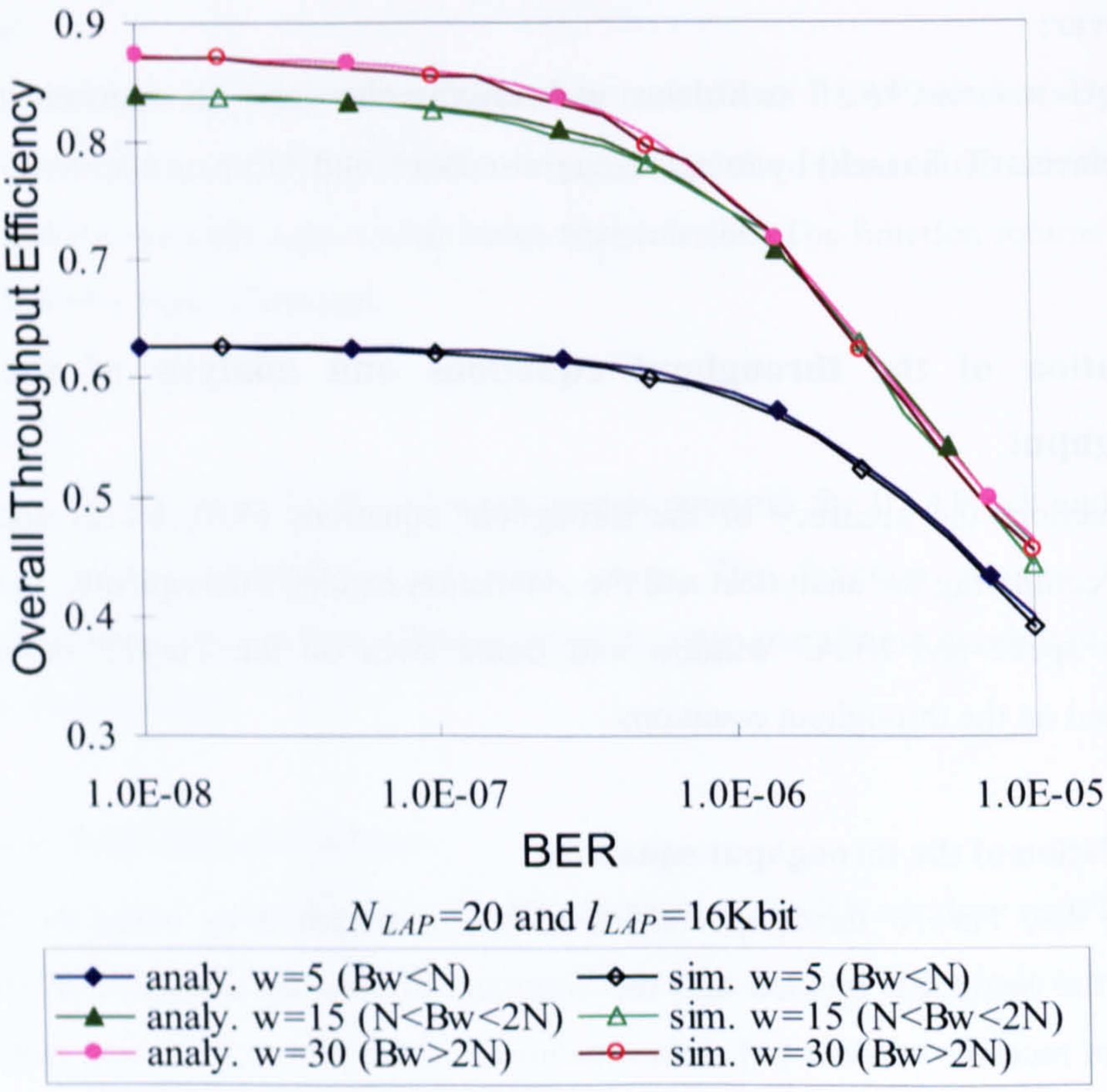


Figure 4.8 Overall TinyTP throughput efficiency comparison for various receiver window size w; 'analy.' stands for the analytical results while 'sim.' stands for the simulation results. N is the acronym for N_{LAP}. Overall TinyTP throughput is the aggregate throughput of B channels.

$\bar{x} = 0.004$ and $\sigma = 0.002$ for w=5, $\bar{x} = 0.005$ and $\sigma = 0.005$ for w=15, $\bar{x} = 0.005$ and $\sigma = 0.003$ for w=30.

4.5.2 Effects of the IrLAP window size (N_{LAP}) and frame size (l_{LAP})

TinyTP throughput efficiencies for different IrLAP window and frame size are compared in Figure 4.9. The TinyTP receiver window size is set to w=20. Throughput efficiency is plotted against the bit error rate in the range of 10^{-5} to 10^{-8}. All the TPE curves decrease with the increasing BER. The system achieves better TPE by using large frame size (l_{LAP}=16Kbit) at the low BER. At the high BER, the system obtains better TPE by implementing small frame size (l_{LAP}=2Kbit). For the same IrLAP window size, the crossing points of the two curves that represent different frame sizes imply that a better throughput can be achieved by appropriately adjusting the window and frame sizes.

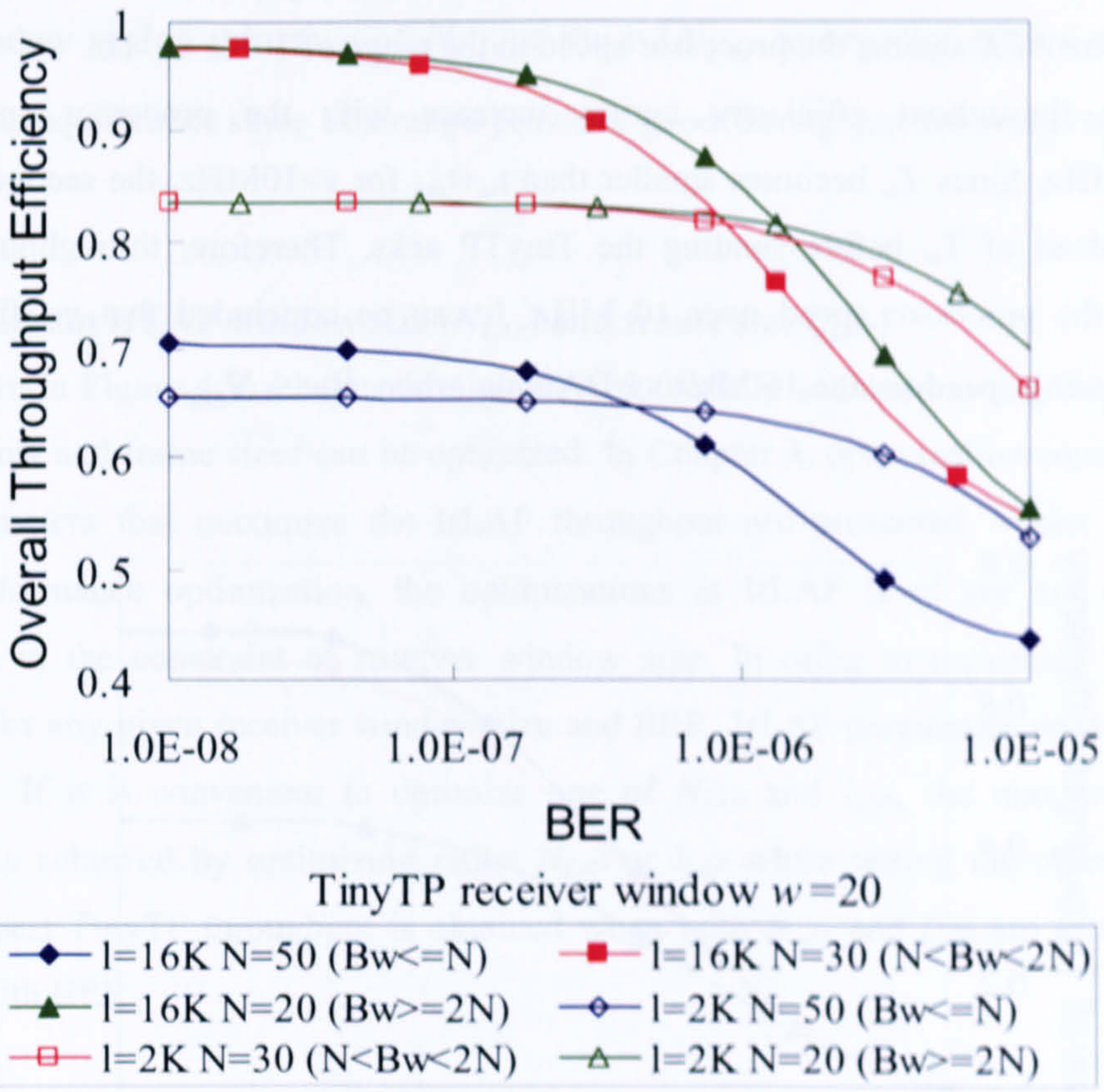


Figure 4.9 TinyTP throughput efficiency comparison using different receiver window size w. l and N are the acronyms for l_{LAP} and N_{LAP} respectively.

4.5.3 Effect of the processor speed (v)

In the TinyTP mathematical analysis, it is assumed that the TinyTP PDU's in the previous window can always be processed and consumed for the case of $N_{LAP} < Bw < 2N_{LAP}$ and $Bw \geq 2N_{LAP}$. The processor speed does not have any effect on the TinyTP throughput when

$Bw > N_{LAP}$. This assumption holds true only when the extreme condition $T_{ta} \leq 3t_{ta} + 2t_{ack} + BT_{ack} + t_I$ is satisfied. To fulfil this condition, processor speed has to be at least as high as $\frac{N_{LAP} l_{TTP}}{4(3t_{ta} + 2t_{ack} + BT_{ack} + t_I)}$. For instance, for the 1 Mbit/s IrDA link with N_{LAP}=4, l_{LAP}=16Kbit, t_{ta}=10^{-3}s and B=2, the processor speed v has to be at least 0.8 MHz. If $v < \frac{CN_{LAP} l_{TTP}}{4(3Ct_{ta} + 2Ct_{ack} + CBT_{ack} + l_{LAP})}$, TinyTP throughput will deteriorate due to the extra time needed for processing the TinyTP PDU's. Based on the mathematical model, Figure 4.10 shows the effect of processing speed when $Bw \leq N_{LAP}$. The result is obtained by using the following parameters: N_{LAP}=20, l_{LAP}=16Kbit and w=5. The throughput efficiencies are plotted in three different *BER* against the processor speed in the range of 10^5 to 10^8 Hz.

All three throughout efficiency curves increase with the processor speed until v>10^7Hz=10MHz. Since T_{ta} becomes smaller than t_{ta}+t_{ack} for v>10MHz, the secondary waits for t_{ta}+t_{ack} instead of T_{ta} before sending the TinyTP acks. Therefore, throughput will not benefit from the processor speed over 10 MHz. It can be concluded that v=10MHz is a suitable processing speed for the 16 Mbit/s IrDA links when $Bw \leq N_{LAP}$.

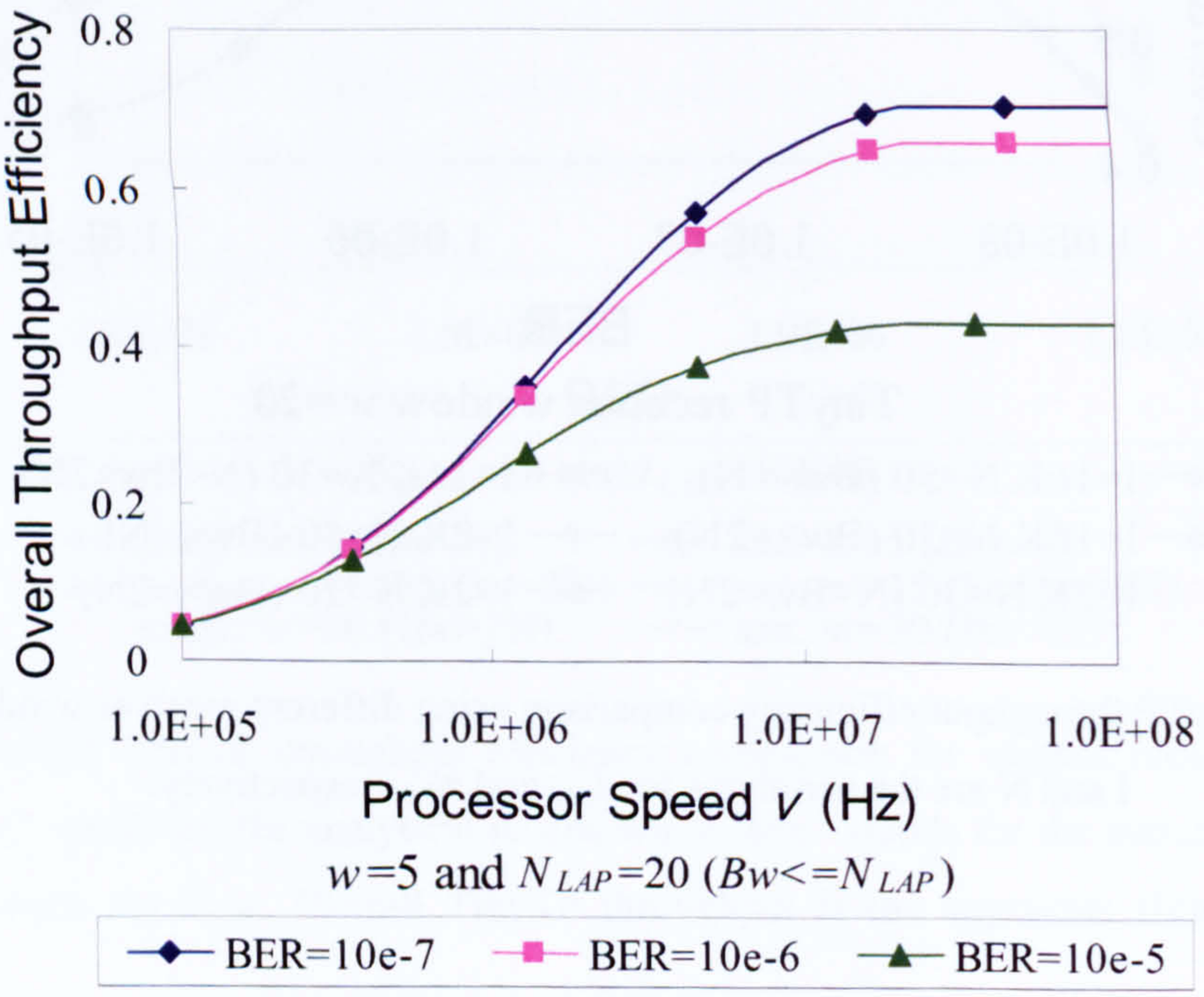


Figure 4.10 Effect of processor speed on throughput efficiency for different *BER* when $Bw \leq N_{LAP}$

4.6 Optimization of the major parameters

4.6.1 Optimum TinyTP receiver window size (w)

As shown in Figure 4.8 and Figure 4.9, the system achieves its best performance when $Bw \geq 2N_{LAP}$ because it takes full advantage of the IrLAP maximum window size. Since the throughput is also constrained by the IrLAP window size N_{LAP}, it will not be improved by a receiver window size larger than $2N_{LAP}$. This means that the system will reach the same throughput as $Bw=2N_{LAP}$ even when $Bw>2N_{LAP}$. Therefore, a receiver window size of $w=2N_{LAP}$ can always achieve the best throughput even if only one TinyTP connection is running. Good TinyTP throughput is also obtained by using a receiver window size of $N_{LAP} < Bw < 2N_{LAP}$. For the devices with limited memory resources, TinyTP can use a receiver window size in the range of $N_{LAP} < Bw < 2N_{LAP}$ to improve system performance and resource requirement since this range provides good throughput for relatively a smaller buffer size.

4.6.2 Optimum IrLAP window size (N_{LAP}) and frame size (l_{LAP})

As shown in Figure 4.9, a better throughput is potentially achieved at the TinyTP level if IrLAP window and frame sizes can be optimized. In Chapter 3, optimization equations of the IrLAP parameters that maximize the IrLAP throughput are presented. When considering TinyTP performance optimization, the optimizations at IrLAP level are not suitable for TinyTP due to the constraint of receiver window size. In order to maximize the TinyTP throughput for any given receiver window size and BER, IrLAP parameters are optimized in this section. If it is convenient to optimize one of N_{LAP} and l_{LAP}, the maximum TinyTP throughput is achieved by optimizing either N_{LAP} or l_{LAP} while setting the other to a fixed value. The best TinyTP throughput is obtained when both N_{LAP} and l_{LAP} are simultaneously optimized with BER.

4.6.2.1 Optimum window or frame size for maximum TinyTP throughput

A. $Bw \leq N_{LAP}$

Since one TinyTP connection cannot send more than w segments before receiving acknowledgement, IrLAP window size always equals to Bw, as shown in Figure 4.4. In this case, only IrLAP frame size l_{LAP} can be optimized with N_{LAP} set to the value of Bw.

The optimum value of l_{LAP} for any fixed N_{LAP} is derived by calculating $\frac{\partial F_{TTP}}{\partial l_{LAP}} = 0$ of F_{TTP} in the equation (4.8) which is a function of l_{LAP}. After some calculations and approximations, the optimum equation for l_{LAP} is derived:

$$l_{opt} = \frac{1}{Bw}\sqrt{\frac{2(t_{ack} + BT_{ack} + 3t_{ta})C}{p_b}} \tag{4.16}$$

B. $N_{LAP} < Bw < 2N_{LAP}$

In this case, both N_{LAP} and l_{LAP} are adjustable. Bw is limited to the range from N_{LAP} to $2N_{LAP}$. It is possible to maximize throughput by fixing either N_{LAP} or l_{LAP} and optimizing the other variable. The optimum value of N_{LAP} for any fixed l_{LAP} is derived by taking $\frac{\partial F_{TTP}}{\partial N_{LAP}} = 0$ of F_{TTP} in the equation (4.8). For fixed N_{LAP}, optimum l_{LAP} value is derived by taking the derivative $\frac{\partial F_{TTP}}{\partial l_{LAP}} = 0$. After some calculations and approximations, the optimum equations for N_{LAP} and l_{LAP} are given by:

$$N_{opt} = \frac{Bw}{2} \tag{4.17}$$

$$l_{opt} = \sqrt{\frac{2(t_{ack} + 3t_{ta} + BT_{ack})C}{(2N_{LAP}^{\ 2} - 2BwN_{LAP} + B^2w^2)p_b}} \tag{4.18}$$

C. $Bw \geq 2N_{LAP}$

By using the same approach as in the case above, optimum equations for N_{LAP} and l_{LAP} are given as:

$$N_{opt} = \sqrt{\frac{2(t_{ack} + 3t_{ta} + BT_{ack})C}{(l_{LAP} + l'_{LAP})^2 p_b}} \tag{4.19}$$

$$l_{opt} = \sqrt{\frac{2(N_{LAP}l'_{LAP} + (t_{ack} + 3t_{ta} + BT_{ack})C)}{N_{LAP}^{\ 2} p_b}} \tag{4.20}$$

4.6.2.2 Simultaneous optimum window and frame size for maximum TinyTP throughput

In this case, both IrLAP window and frame size are simultaneously adjusted. The maximum possible TinyTP throughput can be achieved. In order to derive optimum N_{LAP} and l_{LAP} values, first l_{LAP} is fixed to derive optimum N_{LAP} by calculating $\frac{\partial F_{TTP}}{\partial N_{LAP}} = 0$. Then, the derived optimum N_{LAP} as a function of l_{LAP} is substituted to the throughput equation. Throughput F_{TTP} becomes a function of frame size l_{LAP} for optimum N_{LAP} values. Optimum

l_{LAP} equation is derived by calculating $\frac{\partial F_{TTP}}{\partial l_{LAP}} = 0$. Finally, the optimum equation N_{LAP} is derived by substituting the optimum l_{LAP} to the throughput equation and calculating $\frac{\partial F_{TTP}}{\partial N_{LAP}} = 0$.

A. $Bw \leq N_{LAP}$

As described in section 4.5.2.1.A, IrLAP window size N_{LAP} is fixed at Bw. Only IrLAP frame size l_{LAP} is required to be optimized for throughput. The optimum l_{LAP} is given by equation (4.16).

B. $N_{LAP} < Bw < 2N_{LAP}$

In 4.5.2.1.B, optimum N_{LAP} equation (4.17) is already l_{LAP} independent. To derive optimum l_{LAP}, we only need to substitute (4.17) into the throughput equation (4.12). Optimum N_{LAP} and l_{LAP} in this case are given as:

$$N_{opt} = \frac{Bw}{2} \tag{4.21}$$

$$l_{opt} = \frac{2}{Bw}\sqrt{\frac{(t_{ack} + 3t_{ta} + BT_{ack})C}{p_b}} \tag{4.22}$$

C. $Bw \geq 2N_{LAP}$

After some calculations and approximations, simultaneous optimum N_{LAP} and l_{LAP} are given as:

$$N_{opt} = \sqrt{\frac{2(t_{ack} + 3t_{ta} + BT_{ack})CYp_b}{4l' + 4l'p_bY + p_b{}^2Yl'^2 - p_b{}^2Y^2l'}} \tag{4.23}$$

$$l_{opt} = \sqrt{\frac{4l_{LAP}' + 4l_{LAP}'p_bY + p_b{}^2Yl_{LAP}'^2 - p_b{}^2Y^2l_{LAP}'}{Yp_b{}^2}} \tag{4.24}$$

where $Y = \sqrt{\frac{2(t_{ack} + 3t_{ta} + BT_{ack})C}{p_b}}$

For the low BER, l_{opt} takes values larger than 16 Kbits, which is not allowed by the IrLAP specification. In practice, both approaches for optimum IrLAP parameters are used. Approach 4.5.2.1.C, equation (4.20) is used to obtain N_{opt} in low BER for fixed maximum value l_{LAP}=16Kbits, until the calculated l_{opt} is less than 16 Kbits ($\sim$BER $= 5.7 \times 10^{6}$ from (4.24)

using B=2 and t_{ta}=10^{-4}s). Approach 4.5.2.2.C (equations (4.23) and (4.24)) is applied to obtain N_{opt} and l_{opt} values for higher BER.

In order to examine the accuracy of the optimum equations derived in this section, the results obtained from the equations are compared with the results obtained by the exact numerical methods. The exact optimum results are obtained using a numerical algorithm which locates maximum throughput from equations (4.8), (4.12) and (4.15) by numerically cycling the integer values of N_{LAP} or l_{LAP} for different BER's. In Figure 4.11, the overall throughput efficiency is plotted against the BER in the range of 10^{-5} to 10^{-8} for the simultaneous optimum N_{LAP} and l_{LAP}. The corresponding optimized IrLAP window and frame size are plotted in Figure 4.12.

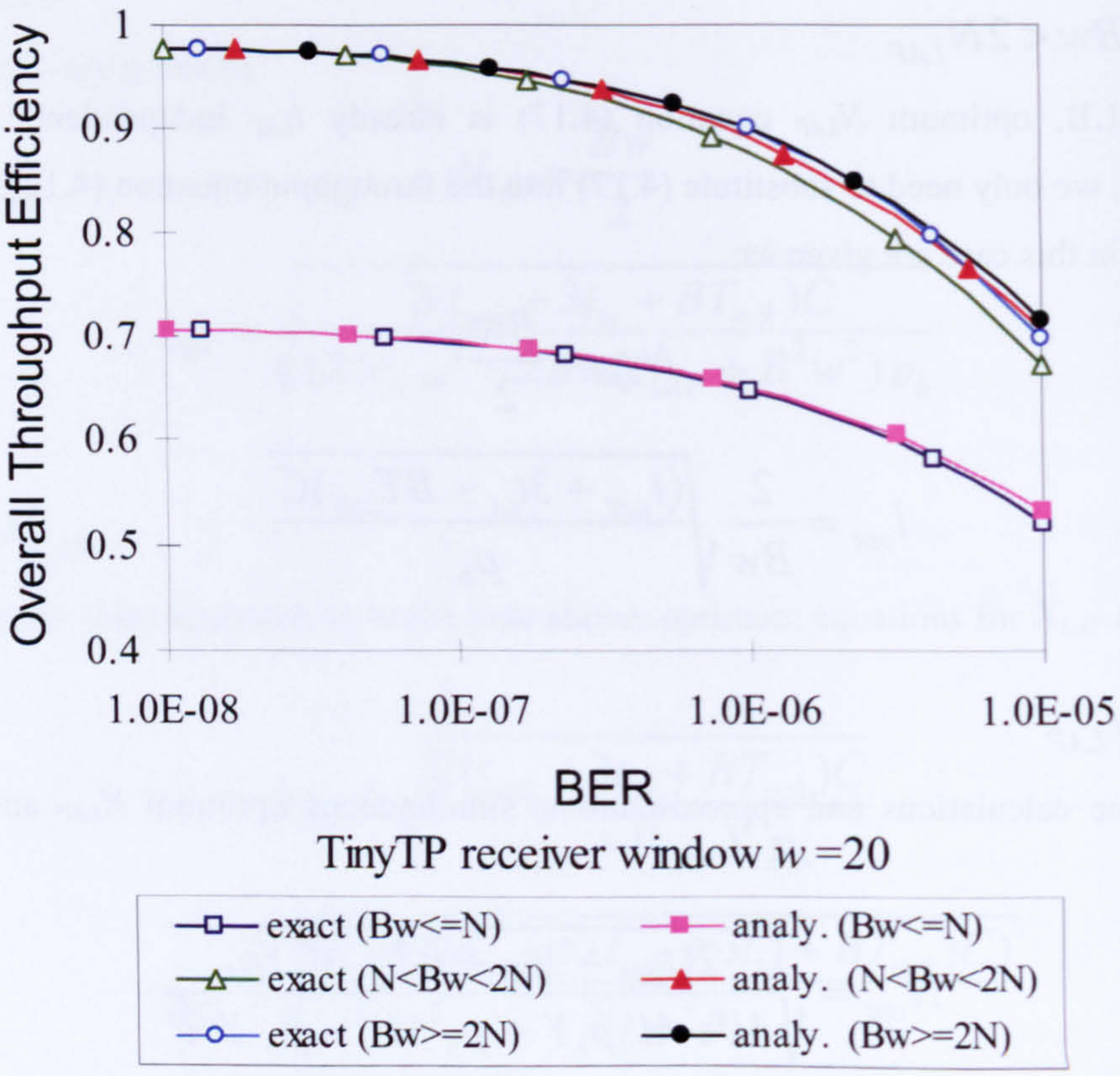


Figure 4.11 TinyTP throughput efficiency using optimum IrLAP window and frame size. l and N are the acronyms for l_{LAP} and N_{LAP} respectively.

$\bar{x}=0.003$ and $\sigma=0.003$ for $Bw \le N$, $\bar{x}=0.011$ and $\sigma=0.010$ for $N<Bw<2N$, $\bar{x}=0.002$ and $\sigma=0.005$ for $Bw \ge 2N$. $\bar{x}$ is the mean of the difference between the exact and the analytical results. σ is the standard deviation of the difference. $\bar{x}$ and σ have the same meaning in section 4.5.2.

In Figure 4.11, the throughput efficiencies gradually decrease as the BER increases. Comparing the optimum TPE results obtained from the equations with the results obtained from the exact numerical method, they show very good agreement for all three cases. In the case of $Bw>2N_{LAP}$, the system always performs at its optimum throughput. For a given size of TinyTP receiver window, N_{opt} should always satisfy the condition $Bw>2N_{LAP}$ and be calculated from equations (4.19) and (4.23) for the corresponding BER. A comparison between Figure 4.9 and Figure 4.11 shows that the optimizations of IrLAP window and frame size are necessary since the performance is significantly improved at the TinyTP level when the optimum values are implemented.

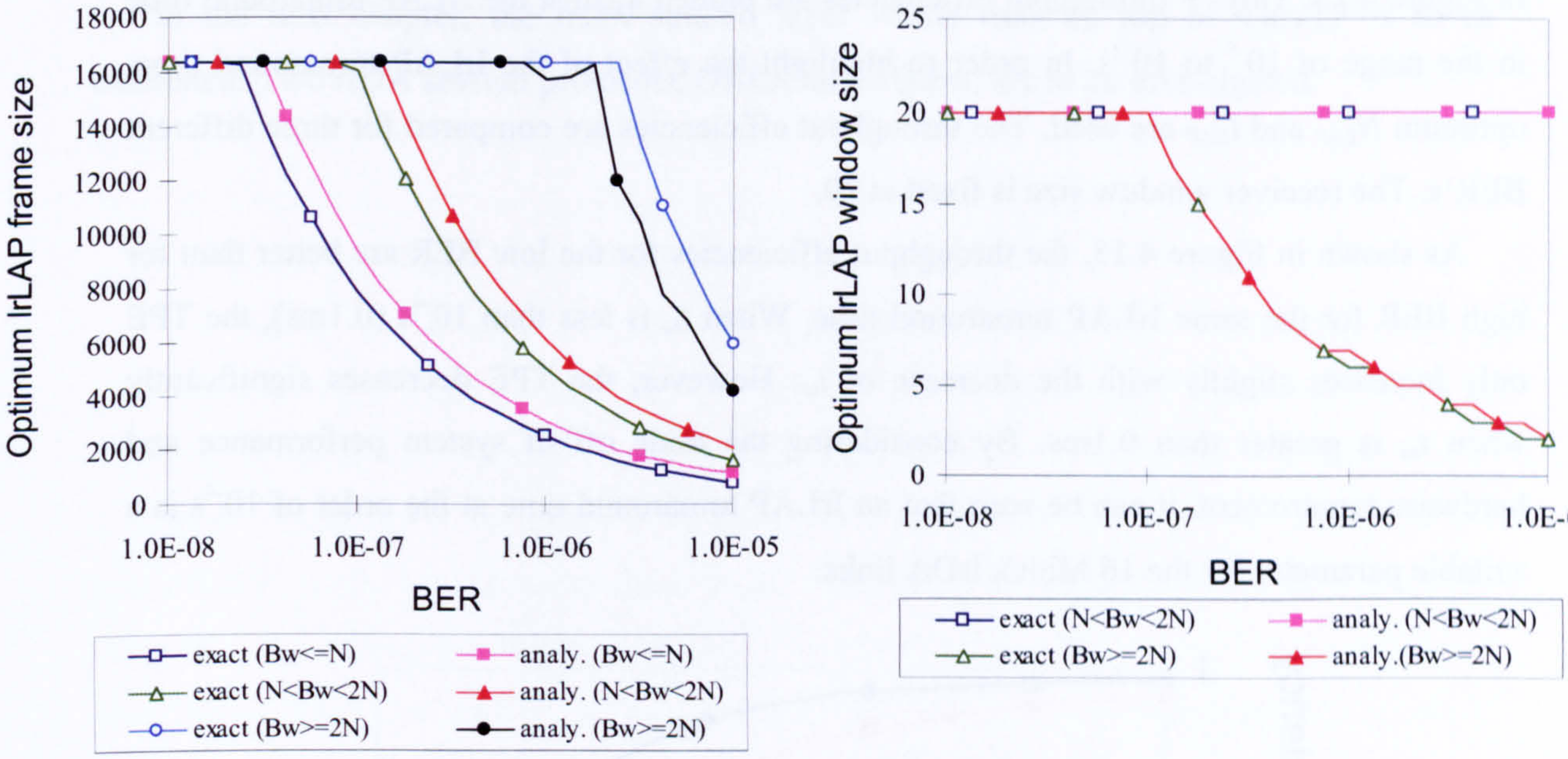


Figure 4.12 The corresponding optimum IrLAP window and frame size to Figure 4.11. l and N are the acronyms for l_{LAP} and N_{LAP} respectively.

For the optimum IrLAP frame size: $\bar{x}=783$ and $\sigma=666$ for $Bw \le N$, $\bar{x}=755$ and $\sigma=732$ for $N<Bw<2N$, $\bar{x}=724$ and $\sigma=1284$ for $Bw \ge 2N$. The unit is bit.

For the optimum IrLAP window size: $\bar{x}=0$ and $\sigma=0$ for $N<Bw<2N$, $\bar{x}=0.12$ and $\sigma=0.33$ for $Bw \ge 2N$.

As shown in Figure 4.12, the optimum frame sizes l_{opt} are fixed at 16 Kbits in the low BER and then drop down significantly with the increasing BER. Although there are some differences between the exact and equation approaches, the curves representing two different approaches have the same shape. Since the corresponding analytical and exact optimum

throughput efficiencies match very well (as shown in Figure 4.11), the optimum equation for l_{LAP} is adequate.

The optimum window sizes N_{opt} have exactly the same values of 20 for either the exact or the equation results when $N_{LAP}<Bw<2N_{LAP}$. For the case of $Bw \geq 2N_{LAP}$, N_{opt} also has a good agreement between the two approaches especially at the low BER. Thus, the optimum equation for N_{LAP} is verified.

4.6.3 Optimum IrLAP turnaround time

As shown in the previous section, N_{opt} should always satisfy the condition $Bw>2N_{LAP}$ in order to optimize the TinyTP throughput. Only the case of $Bw>2N_{LAP}$ is therefore considered. In Figure 4.13, TinyTP throughput efficiencies are plotted against the IrLAP turnaround time in the range of 10^{-5} to 10^{-2}s. In order to highlight the effect of the IrLAP turnaround time, optimum N_{LAP} and l_{LAP} are used. The throughput efficiencies are compared for three different BER's. The receiver window size is fixed at 20.

As shown in Figure 4.13, the throughput efficiencies for the low BER are better than for high BER for the same IrLAP turnaround time. When t_{ta} is less than 10^{-4}s (0.1ms), the TPE only increases slightly with the decrease of t_{ta}. However, the TPE decreases significantly when t_{ta} is greater than 0.1ms. By considering the trade off of system performance and hardware requirement, it can be seen that an IrLAP turnaround time at the order of 10^{-4}s is a suitable parameter for the 16 Mbit/s IrDA links.

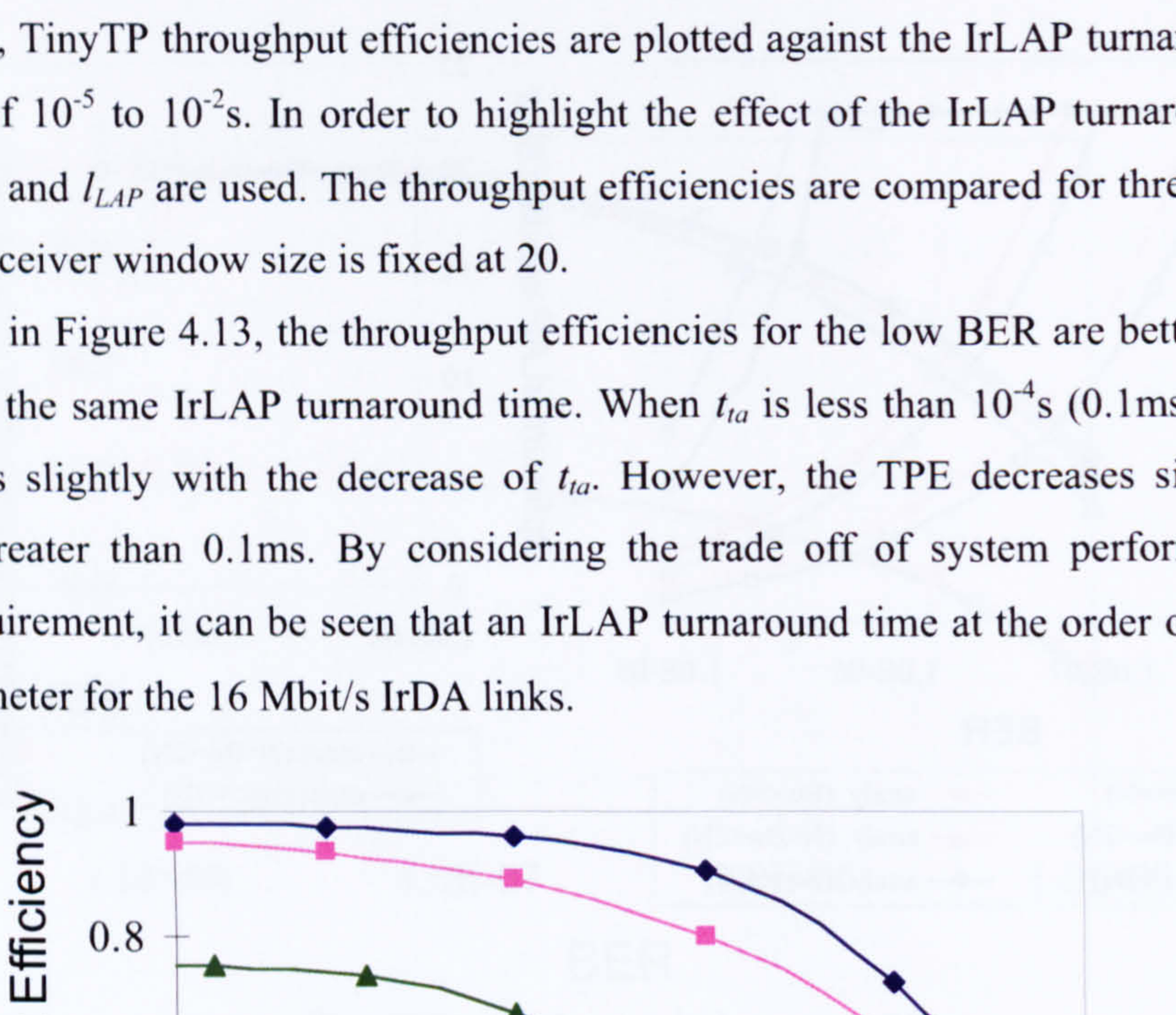


Figure 4.13 Effect of IrLAP turnaround time on TinyTP throughput efficiency

4.7 Summary of the Chapter

This chapter presents a mathematical model for TinyTP over the IrDA protocol stacks. Based on this model, a comprehensive optimization study is carried out to improve system performance at the transport layer. Four major parameters are optimized including the TinyTP receiver window, the IrLAP window and frame size, as well as the IrLAP turnaround time. The analysis shows that the system always achieves its best performance when the receiver window size is at least twice as large as the IrLAP window size ($Bw \geq 2N_{LAP}$). Good TinyTP throughput is also obtained when the receiver window size is larger than the IrLAP window size but smaller than two times of the IrLAP window size ($N_{LAP} < Bw < 2N_{LAP}$). An IrLAP turnaround time in the order of 10^{-4}s is suitable for the 16 Mbit/s IrDA links.

In the next chapter, the IrDA session layer which runs on top of TinyTP is to be examined. Two IrDA session protocols, OBEX and IrBurst, are to be investigated.

Chapter 5. Modelling IrDA Session Layer: OBEX and IrBurst

5.1 Introduction

Object Exchange protocol (OBEX) is a compact, efficient session layer protocol that enables a wide range of devices to exchange data in a simple and spontaneous manner. OBEX was developed by the members of IrDA for the purpose of interconnecting a wide range of devices that support IrDA protocols. It is not, however, limited to use only in the IrDA environment. OBEX has been adopted by other wireless technology transports, including Bluetooth, as the framework for wireless object exchange.

IrDA Burst protocol (IrBurst) is another session layer protocol that is designed for high speed large volume information transfer over IrDA links. IrBurst has been recently developed and published by IrDA in order to complement OBEX which is best suited for transmitting small files. Both OBEX and IrBurst are layered above TinyTP, as shown in Figure 2.5. In this chapter, a detailed study of the performance of OBEX and IrBurst protocol operating on top of the IrDA protocol stacks is carried out.

This chapter is organized as follows: In section 5.1, the detailed properties of OBEX are elaborated. Then, a mathematical model is developed for OBEX which allows derivation of throughput taking into account the lower IrDA protocol stacks in section 5.2. The OBEX throughput equation is subsequently verified by simulations. Section 5.3 carries out an optimization study to maximize the OBEX throughput for any given bit error rates by optimizing the related protocol parameters. The effects of the maximum OBEX packet size and OBEX turnaround time are examined in section 5.4. In section 5.5, the protocol details of IrBurst are discussed. Finally, the performance of IrBurst and OBEX are compared when the transmission of large size objects is considered.

5.2 Object Exchange (OBEX) protocol

IrDA is a short-range, point-to-point, high throughput, low-power technology. This makes it suitable for wireless point-to-point object exchange—sometimes referred to as "point and shoot". OBEX is the mechanism that IrDA employs as a minimum baseline for the point and shoot usage model.

The fundamental goal of OBEX is interoperability. If two users each have an IrDA enabled device and want to exchange an object, it should not matter what type of object it is, or what the relative capabilities of the two devices are—they should be able to exchange the object. The approach taken by OBEX is modelled after the HyperText Transport Protocol (HTTP). The goal of HTTP is to exchange objects of all types as flexible as possible,

independent of data types. That includes being able to reject objects that it does not understand without compromising those portions that it does understand. OBEX can be viewed as a stripped-down, binary version of HTTP, offering a very similar model of flexible data exchange. OBEX works for many devices that support IrDA or Bluetooth communications but cannot afford the substantial resources required by an HTTP server. It also targets devices with different usage models that require connection to the internet.

A major use of OBEX is as a "Push" or "Pull" application, allowing rapid and ubiquitous communications among portable devices in dynamic environments. For instance, a laptop user pushes a file to another laptop or PDA; a digital camera pushes pictures into a film development kiosk, or queries (pulled) for the electronic business card of its owner. OBEX is not limited to quick connect-transfer-disconnect scenarios as it also allows sessions in which transfers take place over a period of time, maintaining the connection even when the session is idle.

Similarly to the HTTP, OBEX employs a Request-Response paradigm often referred to as the Client-Server architecture as the conversation format [45]. The terms client and server refer to originator and receiver of the OBEX connection, not necessarily the one who originated the low level IrLAP connection. Requests are issued by the client representing the party that initiates the OBEX connection. Once a request is issued, the client waits for a response from the server before issuing another request. The request/response pair is referred to as an operation. "PUT" and "GET" are the two types of operations used in OBEX. As the names suggest, the "PUT" operation sends one object from the client to the server, while the "GET" operation requests that the server returns an object to the client. The maximum and minimum lengths for both request and response packets are 512 Kbits and 2 Kbits respectively [45].

Figure 5.1 illustrates OBEX in the process of packetizing a large object for transmission initiated by the 'PUT' operation. An object is any complete logical unit of data. It may be an individual data structure, a data packet, or a file. Every object has two basic parts. First, there is header information, which carries information about the object. This might include the object's size, name, creation time and date stamp, type, etc. Secondly, there are the actual data. The initial OBEX request packet (first packet) will typically, although not strictly required to do so, have certain header information of the object. Although no specific requirements are imposed to the first packet, following analysis assumes that first packet conveys all the necessary header information of the object, as shown in Figure 5.1. The connection-oriented session allows capabilities and information to be exchanged just once at the start of the connection, and allows state information to be kept. The subsequent OBEX packets therefore only have to include the Operation code (Opcode), the packet length field and the body header. The Opcode field is a 1-byte value that specifies the type of the OBEX

packet ('put' packet, 'get' packet, etc). The packet length field provides the information of the OBEX packet length. The body header consists of the body header ID field identifying it as an object body, and a 2-byte body length field indicating the length of the payload.

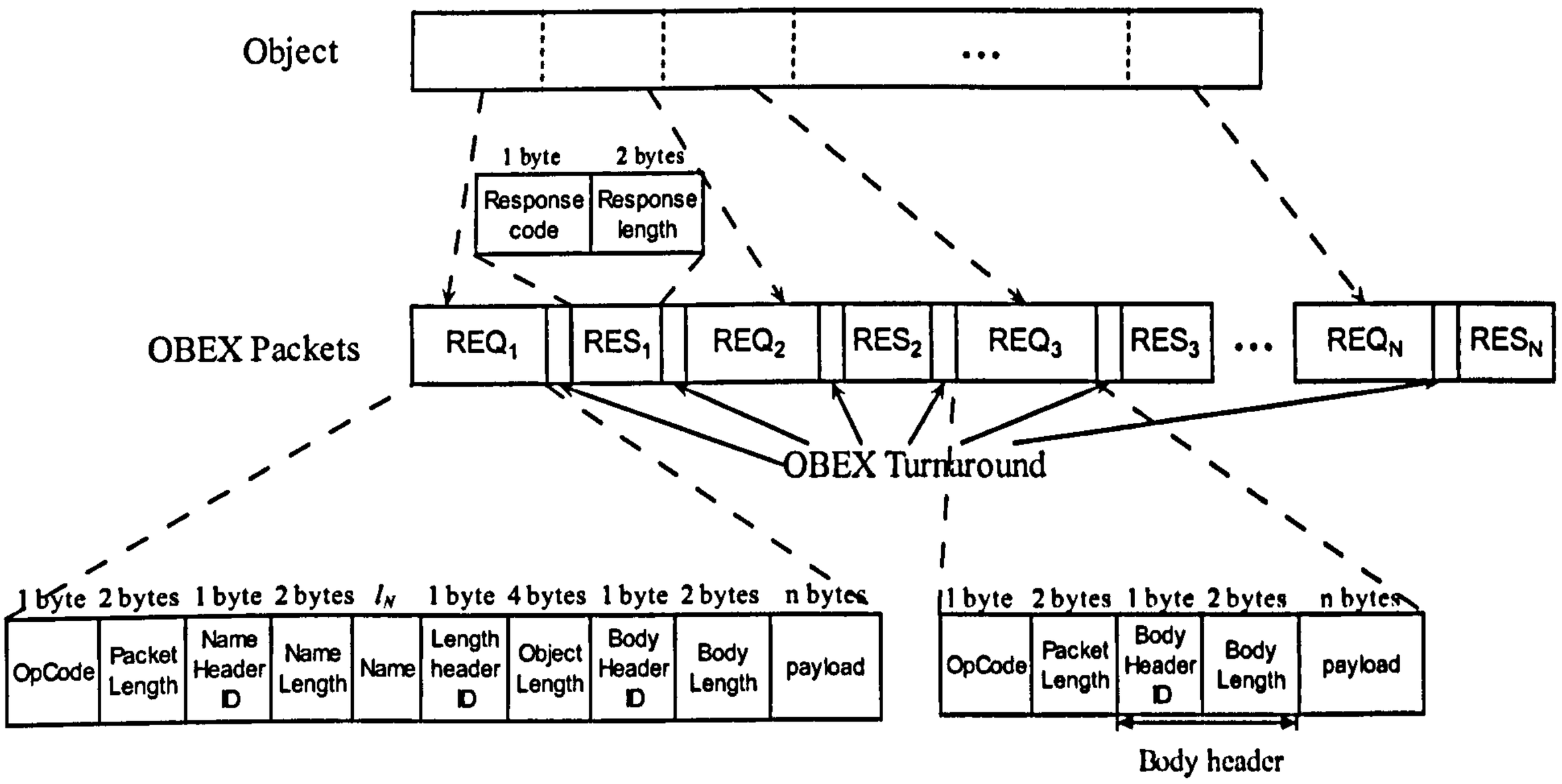


Figure 5.1 OBEX object packetization

After the connection establishment of the OBEX session, the OBEX session communicates with the TinyTP layer by quoting a unique Link Service Access Point Selector (LSAP-SEL) assigned by IrLMP. Details of the IrDA connection establishment procedures are to be discussed in Chapter 6. It should be noted that more than one OBEX sessions can operate consecutively at the same time. As discussed in Chapter 4, multiple IrLMP connections are created when multiple OBEX sessions are implemented. Since the effects of multiple connections and different receiver buffer size are considered and examined in Chapter 4, only one OBEX session is considered in the throughput derivation in order to simplify the analysis in this Chapter. It is also assumed that the receiver buffer is always sufficient to accommodate all the incoming TinyTP PDU's. In such case, TinyTP only requires to add one byte header information to each OBEX packet and it does not need to perform flow control.

5.3 Analytical model of OBEX

For the purpose of developing the mathematical model, the following assumptions are made:

- The OBEX packets are sent in the OBEX 'PUT' operation mode.
- Only one OBEX session is active in the sender, thus the IrDA link is exclusively used by one IrLMP connection assigned for the OBEX session.
- Only the 'connected' OBEX packets (not the first packet) are considered.
- The buffer of the peer is large enough to accommodate the incoming TinyTP PDU's, therefore, no TinyTP flow control is performed and TinyTP maximum segment size plus TinyTP and IrLMP headers are equal to the IrLAP frame size ($l_{TTP} + l'_{LMP} + l'_{TTP} = l_{LAP}$).

The details of the parameters used in the OBEX modelling are given in Table 3-1, Table 4-1 and Table 5-1. The headers of different layers are illustrated in Figure 5.2. The mathematical model uses Figure 5.3 for the derivation of OBEX throughput by considering the underlying IrDA protocol stacks.

Symbol	Parameter Description	Unit
P_{REQ}	OBEX request packet size	bit
P_{RES}	OBEX response packet size	bit
l'_{OBEX}	OBEX request packet overhead	bit
l_{OBEX}	OBEX request packet payload $l_{OBEX} = P_{REQ} - l'_{OBEX}$	bit
O	Object size	bit
T_{REQ}	Time to transmit an OBEX response packet	sec
T_{OBEX}	OBEX turnaround time	sec
T_{full}	Time to transmit the full IrLAP windows	sec
T_{rem}	Time to transmit the incomplete IrLAP window	sec

Table 5-1 OBEX parameters used in modelling

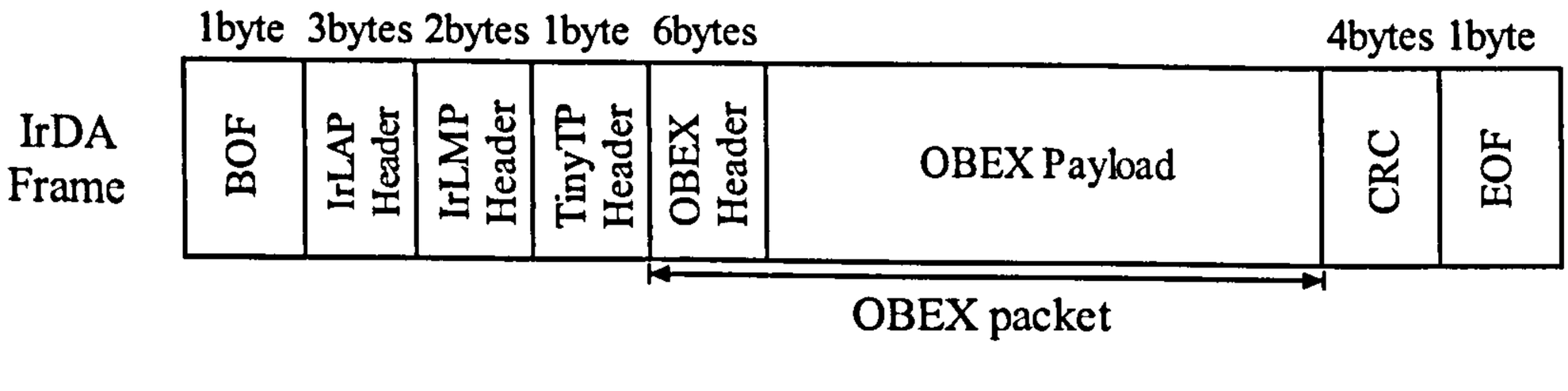
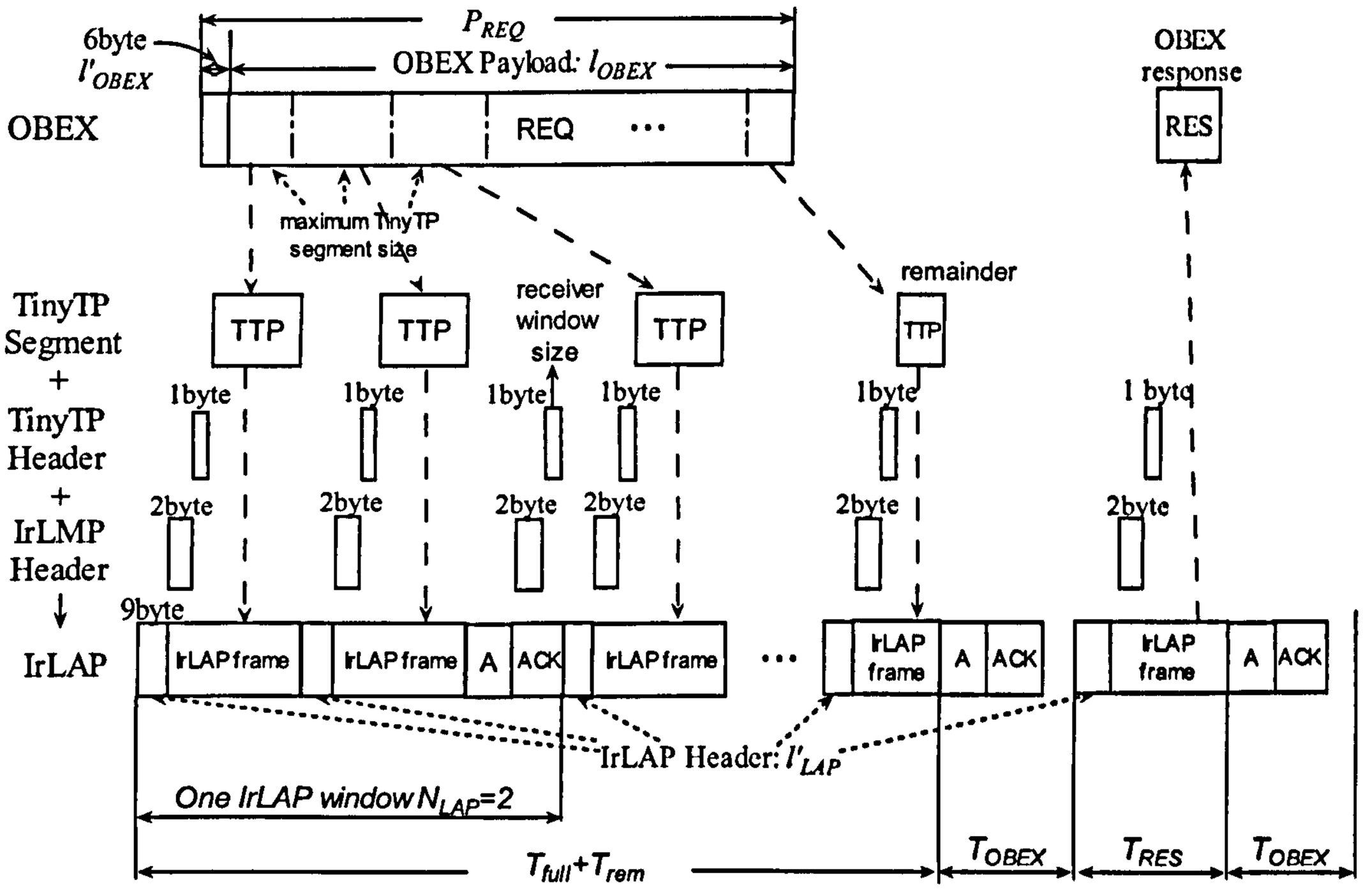


Figure 5.2 VFIR IrDA frame structure



A stands for the IrLAP acknowledgement, ACK stands for the TinyTP acknowledgement

Figure 5.3 Mapping OBEX, TinyTP, IrLMP to IrLAP frames

For each OBEX request packet, TinyTP has to transmit a packet with the length of P_{REQ}, as illustrated in Figure 5.3. One OBEX request packet is often segmented into a number of TinyTP PDU's and sent by IrLAP. The segmented OBEX packet is likely to be sent in multiple *full* IrLAP windows and a single *incomplete* IrLAP window at the end, Figure 5.3. When $l_{TTP} \cdot N_{LAP} < P_{REQ}$, the OBEX packet requires more than one IrLAP window for its transmission. Thus, to transmit a large OBEX packet ($l_{TTP} \cdot N_{LAP} < P_{REQ}$), several complete IrLAP windows and probably one incomplete IrLAP window are needed.

When $l_{TTP} \cdot N_{LAP} \geq P_{REQ}$, the OBEX packet can be sent within a single IrLAP window. If the OBEX packet does not require a full IrLAP window for the transmission, the transmitter simply sets the Poll bit to 1 (P=1) in the last IrLAP frame requesting an acknowledgement from the receiver.

The average time required for transmitting all the complete IrLAP windows for one OBEX request packet is:

$$T_{full} = \left\lfloor \frac{P_{REQ} / l_{TTP}}{N_{cor}} \right\rfloor \cdot t_w = \left\lfloor \frac{N_{OBEX}}{N_{cor}} \right\rfloor \cdot t_w \qquad (5.1)$$

When N_{OBEX} is the number of IrLAP windows required to transmit one OBEX packet. N_{cor} is the average number of IrLAP frames correctly transmitted in a full IrLAP window and t_w is the average time to transmit a full IrLAP window excluding the propagation delay. It should be noted that $T_{full} = 0$ if $l_{TTP} \cdot N_{LAP} \geq P_{REQ}$. As discussed in section 3.2, N_{cor} and t_w are given as:

$$N_{cor} = \sum_{i=1}^{N_{LAP}-1} i(1-p)^i p + N_{LAP}(1-p)^{N_{LAP}} = \frac{(1-p)(1-(1-p)^{N_{LAP}})}{p} \qquad (5.2)$$

$$t_w = N_{LAP}t_I + t_{ack} + p(t_{Fout} + t_s) + T_{ack} \qquad (5.3)$$

The length of the incomplete IrLAP window is given as:

$$L_{rem} = (P_{REQ})\text{modulus}(l_{TTP} \cdot N_{cor}) \qquad (5.4)$$

The number of IrLAP frames in the incomplete IrLAP window is:

$$N_{in} = \lceil L_{rem} / l_{TTP} \rceil \qquad (5.5)$$

The probability of having error/errors in the incomplete IrLAP window is:

$$p_{in1} = 1-(1-p)^{N_{in}} \qquad (5.6)$$

Because of the small value of p, p_{in1} can be approximated as:

$$p_{in1} = 1-(1-p)^{N_{in}} \approx 1-(1-N_{in}p) = N_{in}p \qquad (5.7)$$

Similarly to the derivation in section 4.2, if errors occur in transmitting the incomplete IrLAP window with probability of p_{in1}, it is sufficient to assume that on average the error occurs in the middle of the window due to the randomness of error occurrence. Retransmission will occur to recover the error with window length of $0.5N_{in}$. If additional errors occur during retransmission with probability of $p_{in2} = p_{in1}(1-(1-p)^{0.5N_{in}}) \approx 0.5{N_{in}}^2 p^2$, another retransmission window is required with window length half of the previous value, i.e. $0.25N_{in}$, and so on. When the length of retransmission window becomes less than 1, the IrLAP window is considered to be successfully transmitted. The average time for transmitting the incomplete window therefore can be derived as:

$$\begin{aligned}
T_{rem} &= N_{in}t_I + p(t_{Fout} + t_s) + t_{ack} + T_{ack} + p_{in1}(\frac{1}{2}N_{in}t_I + p(t_{Fout} + t_s) + t_{ack} + T_{ack}) + \\
&\quad \cdots + p_{inN}(\frac{1}{2X}N_{in}t_I + p(t_{Fout} + t_s) + t_{ack} + T_{ack}) \\
&= \left((1 + \frac{1}{2}N_{in}p + \cdots + \left(\frac{1}{2}\right)^{\frac{1}{2}X(X+1)} {N_{in}}^X p^X)\right) N_{in}t_I + (1 + p_{in1} + \cdots + p_{inX})(p(t_{Fout} + t_s) + t_{ack} + T_{ack}) \\
&= \left(1 + \sum_{i=1}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i+1)} (N_{in}p)^i\right)\right) N_{in}t_I + \left(1 + N_{in}p + \sum_{i=2}^{X}\left(\left(\frac{1}{2}\right)^{\frac{1}{2}i(i-1)} (N_{in}p)^i\right)\right)(p(t_{Fout} + t_s) + t_{ack} + T_{ack})
\end{aligned} \qquad (5.8)$$

where X is an integer with value of $X = \lfloor \log_2 N_{in} \rfloor$ that satisfies $1/2^X \cdot N_{in} \geq 1$.

Since OBEX employs Request-Response as the transmission scheme, the transmitter has to wait for the acknowledgement from the receiver before sending the next OBEX packet. OBEX response packet from the receiver is only sent after the successful transmission of an OBEX request packet. Since the response packets are used only for the acknowledgement purpose (no payload), the packet length is equal to the OBEX header l'_{OBEX} ($P_{RES}=l'_{OBEX}$). Due to the small size of l'_{OBEX}, it is sent in a single IrLAP frame and it is assumed to be transmission error free. The time required to transmit a response packet is:

$$T_{RES} = \frac{P_{RES} + l'_{TTP} + l'_{LMP} + l'_{LAP}}{C} = \frac{l'_{OBEX} + l'_{TTP} + l'_{LMP} + l'_{LAP}}{C} \qquad (5.9)$$

Due to the small size of the IrLAP and TinyTP acknowledgements and the fast hardware link turnaround (IrLAP turnaround t_{ta}), OBEX turnaround time T_{OBEX} is assumed to be much longer than the IrLAP and TinyTP acknowledgement time ($T_{OBEX} >> t_{ack} + T_{ack} + 2t_{ta}$). From Figure 5.3, the average time to transmit one OBEX packet is:

$$T = T_{full} + T_{rem} + T_{RES} + 2T_{OBEX} \qquad (5.10)$$

The OBEX throughput, which is defined as the useful data bit per second, is given by:

$$F_{OBEX} = \frac{P_{REQ} - l'_{OBEX}}{T} = \frac{l_{OBEX}}{T} \qquad (5.11)$$

The throughput efficiency is:

$$TPE = \frac{F_{OBEX}}{C} \qquad (5.12)$$

By using the simulation programs as discussed in section 4.3, the accuracy of the derived OBEX throughput equation is examined by comparing the analytical results and the simulation results. In Figure 5.4, the throughput efficiency is plotted against the bit error rate in the range from 10^{-4} to 10^{-8} by using different OBEX packet sizes, IrLAP window and frame sizes. The other parameters are set as follows: C=16Mbit/s, T_{OBEX}=2ms, t_{ta}=0.1ms. These parameter values will be used throughout this chapter unless otherwise specified. The mean ($\bar{x}$) of the difference between the analytical and the simulation results and the standard deviation (σ) of the difference are also calculated by using equations (3.17)-(3.19).

As shown in Figure 5.4, the analytical results match very well with the simulation results for all three different parameter combination cases. The difference between the analytical and the simulation results is marginal. This verifies the mathematical model derived above. When comparing the three different parameter combination cases, Par.1 provides lower throughput

at the low BER but achieves higher throughput at the high BER compared to Par.2 and Par.3. The results indicate that different value of OBEX packet size P_{REQ}, and IrLAP window N_{LAP} and frame l_{LAP} size have different effects on the OBEX throughput. The OBEX throughput therefore may be improved by optimizing P_{REQ}, N_{LAP} and l_{LAP}.

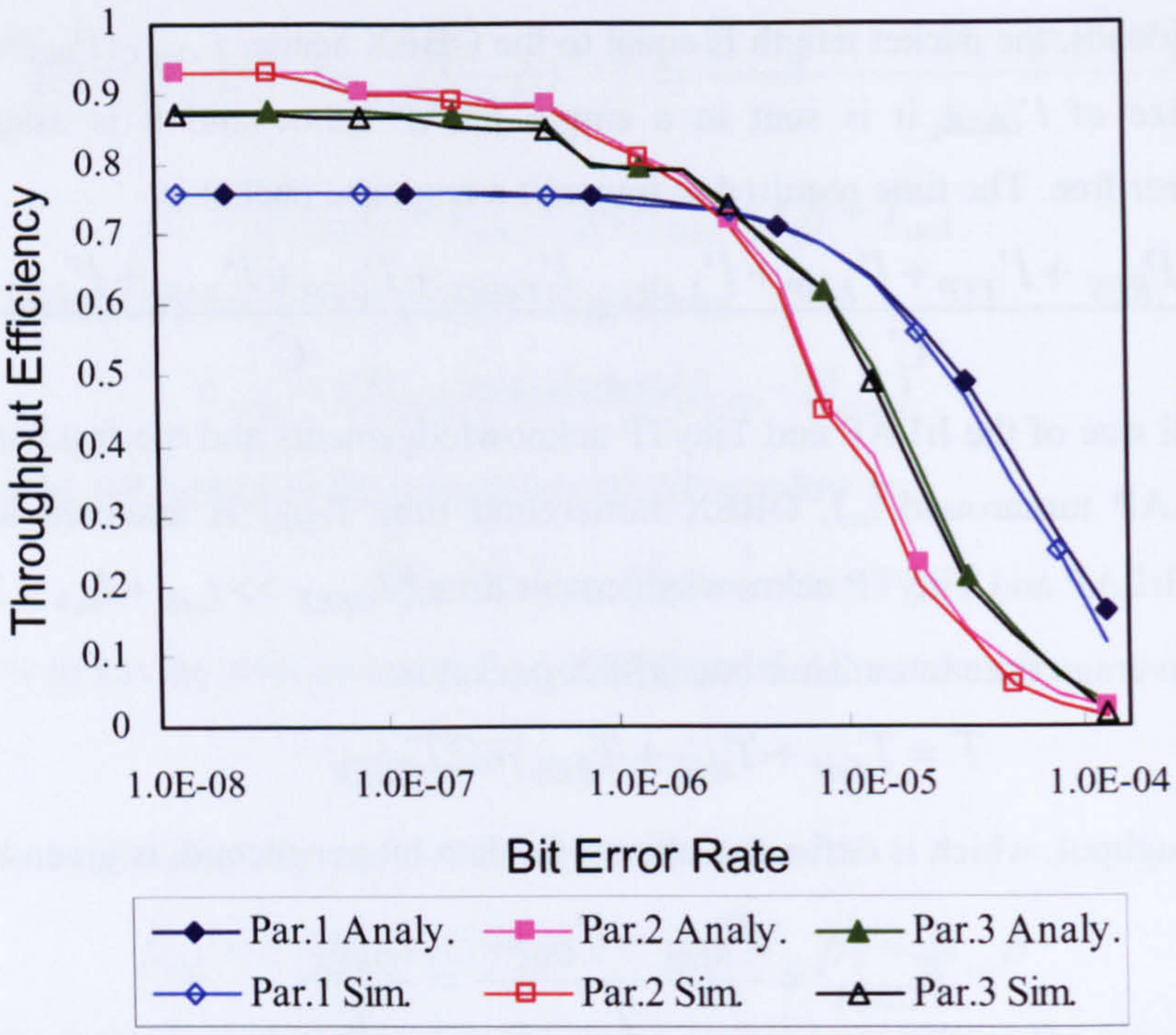


Figure 5.4 OBEX throughput efficiency against BER using non-optimum parameters. Analy. stands for analytical results and sim. stands for simulation results

Par.1: N_{LAP}=30, l_{LAP} =1024bit, P_{REQ} =512Kbit;

Par.2: N_{LAP}=60, l_{LAP} =6Kbit, P_{REQ} =256Kbit;

Par.3: N_{LAP}=80, l_{LAP}= 12Kbit, P_{REQ}=128Kbit;

$\bar{x} = 0.009$ and $\sigma = 0.012$ for Par.1, $\bar{x} = 0.012$ and $\sigma = 0.009$ for Par.2, $\bar{x} = 0.010$ and $\sigma = 0.006$ for Par.3.

5.4 Parameters Optimization

As shown in the previous section, OBEX performance can be improved by choosing the appropriate OBEX and IrLAP parameters for a given BER. Equations for optimizing the parameters are derived in this section. Several evaluation results are presented in order to verify the mathematical analysis.

5.4.1 Optimum OBEX packet size

Maximum OBEX packet size P_{REQ} is one of the negotiable parameters during the connection establishment stage. Its value is between from 2 Kbits to 512 Kbits as defined in [45]. By assuming the time to transmit the IrLAP windows is equally distributed for each IrLAP frame, the following equation holds true:

$$\left\lfloor \frac{P_{REQ}}{l_{TTP}\cdot(1-p)\cdot(1-(1-p)^{N_{LAP}})/p} \right\rfloor \cdot t_w + T_{rem} = \frac{P_{REQ}\cdot t_w}{l_{TTP}\cdot(1-p)\cdot(1-(1-p)^{N_{LAP}})/p} \tag{5.13}$$

P_{REQ} is very large compared to the OBEX header l'_{OBEX}, hence, it is sufficient to assume $P_{REQ} - l'_{OBEX} \approx P_{REQ}$. By substituting this assumption and the assumption given by (5.13) to equation (5.11), the OBEX throughput becomes:

$$\begin{aligned} F_{OBEX} &= \frac{P_{REQ} - l'_{OBEX}}{\left\lfloor \frac{P_{REQ}}{l_{TTP}\cdot(1-p)\cdot(1-(1-p)^{N_{LAP}})/p} \right\rfloor \cdot t_w + T_{RES} + T_{rem} + 2T_{OBEX}} \\ &\approx \frac{P_{REQ}}{\frac{P_{REQ}\cdot t_w}{l_{TTP}\cdot(1-p)\cdot(1-(1-p)^{N_{LAP}})/p} + T_{RES} + 2T_{OBEX}} \end{aligned} \tag{5.14}$$

Due to the small value of p, it is assumed that $1-(1-p)^{N_{LAP}} \approx 1-(1-N_{LAP}p) = N_{LAP}p$. Let $(T_{RES} + 2T_{OBEX})\cdot N_{LAP}\cdot l_{TTP}\cdot(1-p) = A$, then (5.14) is further simplified to:

$$F_{OBEX} = \frac{P_{REQ}\cdot N_{LAP}\cdot l_{TTP}\cdot(1-p)}{P_{REQ}\cdot t_w + A} \tag{5.15}$$

where A is independent of P_{REQ}. Since $t_w < 1$ and $N_{LAP}\cdot l_{TTP}\cdot(1-p) > 1$, then $t_w < N_{LAP}\cdot l_{TTP}\cdot(1-p)$. F_{OBEX} therefore should increase when the value of P_{REQ} increases. One can conclude from equation (5.15) that the OBEX throughput always benefits from a bigger P_{REQ} for any bit error rate.

Besides the OBEX packet size, IrLAP window N_{LAP} and frame sizes l_{LAP} are the other two major factors affecting the OBEX throughput. IrLAP window and frame sizes are to be optimized in the following sections by using a similar approach that has been used in section 4.5.

5.4.2 Optimum IrLAP window or frame size for maximum OBEX throughput

Due to the half duplex nature of the IrDA link, window size is an important and easily adjustable parameter. If a large window size N_{LAP} is applied, the possibility of sending frames following an erroneous frame is higher. All the frames that are received after an erroneous

frame are considered out of sequence and discarded by the receiver. These frames essentially delay the reversing of link direction. Time taken for such frame transmissions reduces throughput.

In contrast, for a small window size N_{LAP}, the link transmission direction changes more often and the IrLAP acknowledgements have to be sent more frequently. It is possible to optimize N_{LAP} for various bit error rates. By calculating the derivative of N_{LAP} for equation (5.14) $\frac{\partial F_{OBEX}}{\partial N_{LAP}} = 0$, the optimum value of N_{LAP} for any fixed l_{LAP} and P_{REQ} is derived:

$$N_{opt} = \sqrt{\frac{2t_{ack}C}{{l_{LAP}}^2 p_b}} \tag{5.16}$$

Since the header of the IrLAP frame is very small compared to the payload of the IrLAP frame, two approximations have been made in the derivation above: $p = 1-(1-p_b)^{l_{LAP}+l'_{LAP}} \approx l_{LAP}p_b$ and $t_I \approx \frac{l_{LAP}}{C}$.

Besides optimizing the window size, the frame size can also be optimized to improve the OBEX throughput. A smaller frame size reduces frame error probability and the necessary for error retransmissions. Implementing a small frame size however results in relative increase of IrLAP overheads since each frame transmission requires the transmission of flags, address field, control field and FCS. An optimum frame size l_{LAP} can be derived for calculating $\frac{\partial F_{OBEX}}{\partial l_{LAP}} = 0$ for fixed N_{LAP} and P_{REQ}. After some calculations, the equation for optimum l_{LAP} is derived:

$$l_{opt} = \frac{-t_{ack}Cp_b + \sqrt{2t_{ack}Cp_b}}{N_{LAP}p_b} \tag{5.17}$$

In order to examine the accuracy of equations (5.16) and (5.17), the analytical optimum results are compared with the exact optimum results. Similarly to the method used in section 4.5.2, the exact optimum results are obtained using a numerical algorithm which locates maximum throughput from equation (5.11) by numerically cycling the integer values of N_{LAP} or l_{LAP} in the range of 1-127 and 512bit-16Kbit respectively at different bit error rates.

The optimum values of window N_{LAP} or frame size l_{LAP} using the exact and the analytical approaches are compared in Figure 5.5 and Figure 5.7. The corresponding throughput efficiencies are presented in Figure 5.6 and Figure 5.8. The results are plotted against the BER by using a P_{REQ} value of 512 Kbits. The range of the results is restricted by the IrLAP specification [42]. Thus, optimum N_{LAP} is fixed at 127 if a value larger than 127 is required, while the optimum l_{LAP} is fixed at 16 Kbits if a value larger than 16 Kbits is required.

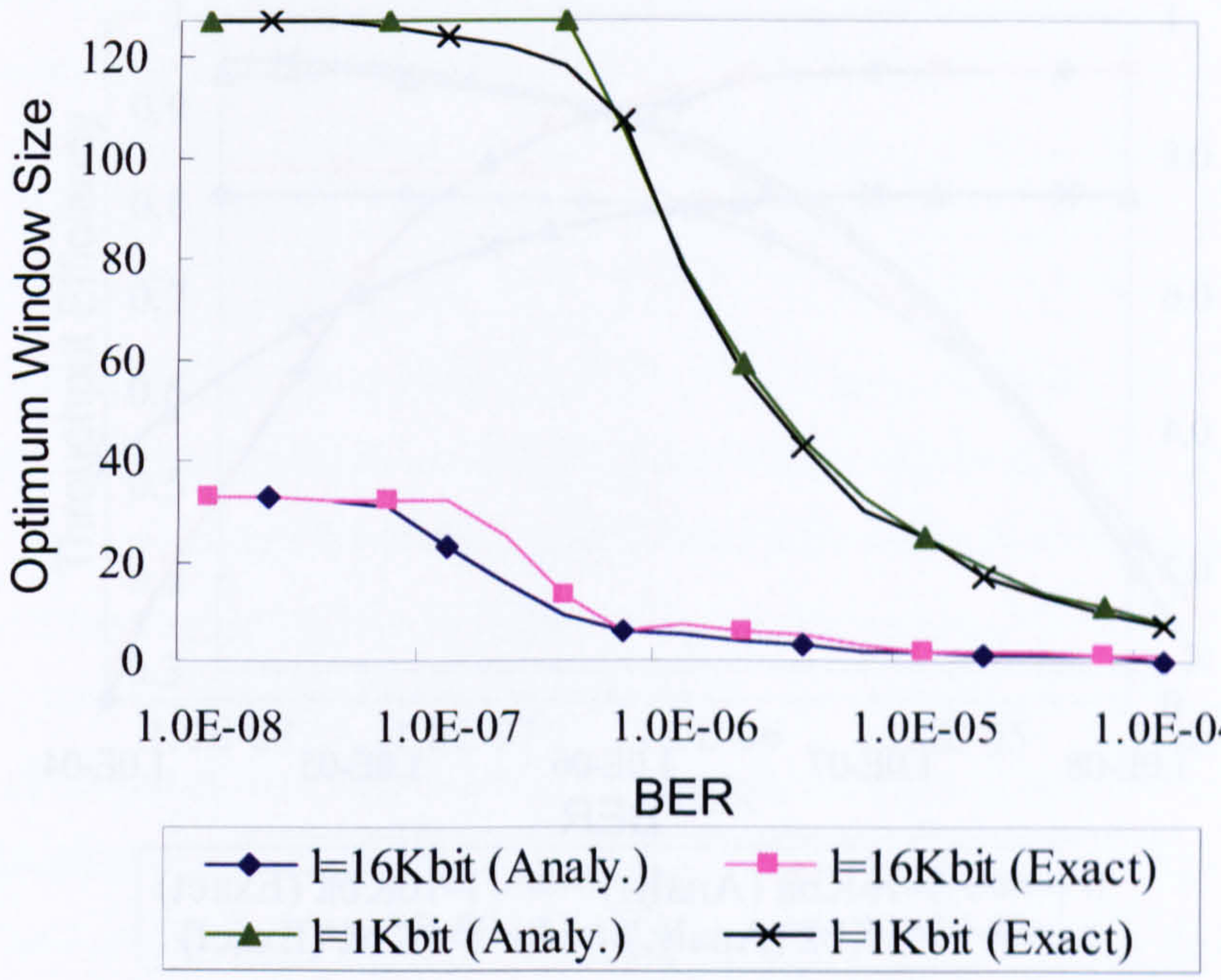


Figure 5.5 Comparison of optimum window sizes when using exact algorithm and analytical equations for two fixed frame sizes of 16 Kbits and 1 Kbits. l is the acronym of l_{LAP}. $\bar{x} = 1.88$ and $\sigma = 2.26$ for l=1Kbit, $\bar{x} = 1.94$ and $\sigma = 2.86$ for l=16Kbit. where $\bar{x}$ is the mean of the difference between the analytical and the exact results. σ is the standard deviation of the difference. $\bar{x}$ and σ have the same meaning throughout section 5.3.

As shown in Figure 5.5, the optimum N_{LAP} decreases rapidly when the BER increases. By comparing the exact and the analytical optimum N_{LAP} values, the biggest difference is observed at the low BER. For the high BER, these two approaches match very well. The differences between the exact and analytical results are mainly due to the approximation applied to equation (5.13). The approximation reduces the effect of the incomplete window in the throughput equation. Since the optimum N_{LAP} values are large for the low BER, the incomplete window may also be large, and hence has a bigger effect on the throughput as discussed later. As the OBEX throughput is already very high at the low BER, accurate optimum N_{LAP} values for low BER do not significantly improve the throughput. Since there is more to be gained on the throughput at the high BER, it is more important to obtain accurate optimum N_{LAP} values. As shown in Figure 5.6, only a small difference is observed on the corresponding throughput results despite the difference between the exact and the analytical approaches for optimum N_{LAP} at the low BER. This shows that the optimum IrLAP window equation (5.16) is adequate. This also verifies the validity of the approximations made in the optimum equations derivation.

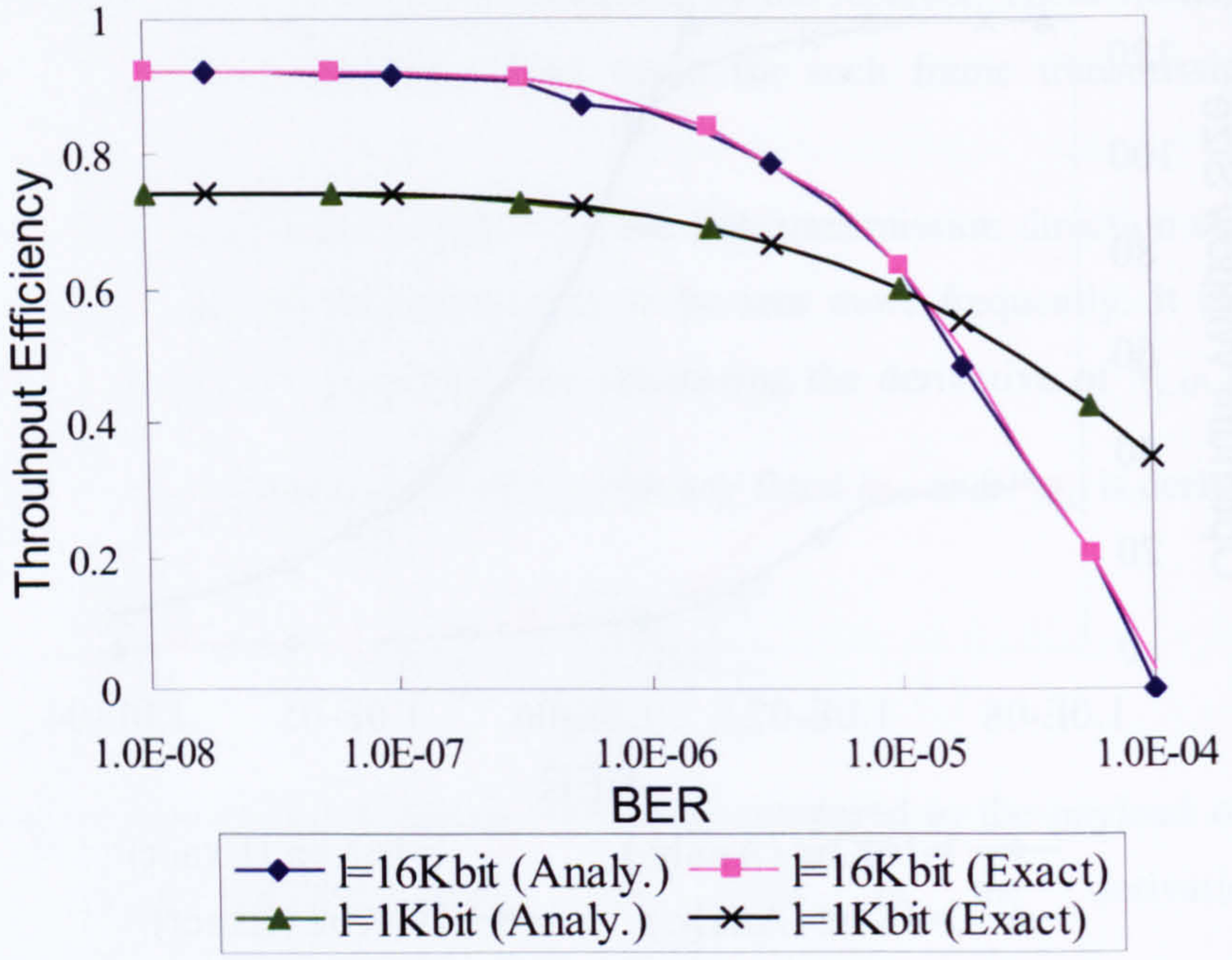


Figure 5.6 The corresponding throughput efficiency of Figure 5.5

$\bar{x} = 0.001$ and $\sigma = 0.001$ for l=1Kbit, $\bar{x} = 0.008$ and $\sigma = 0.009$ for l=16Kbit.

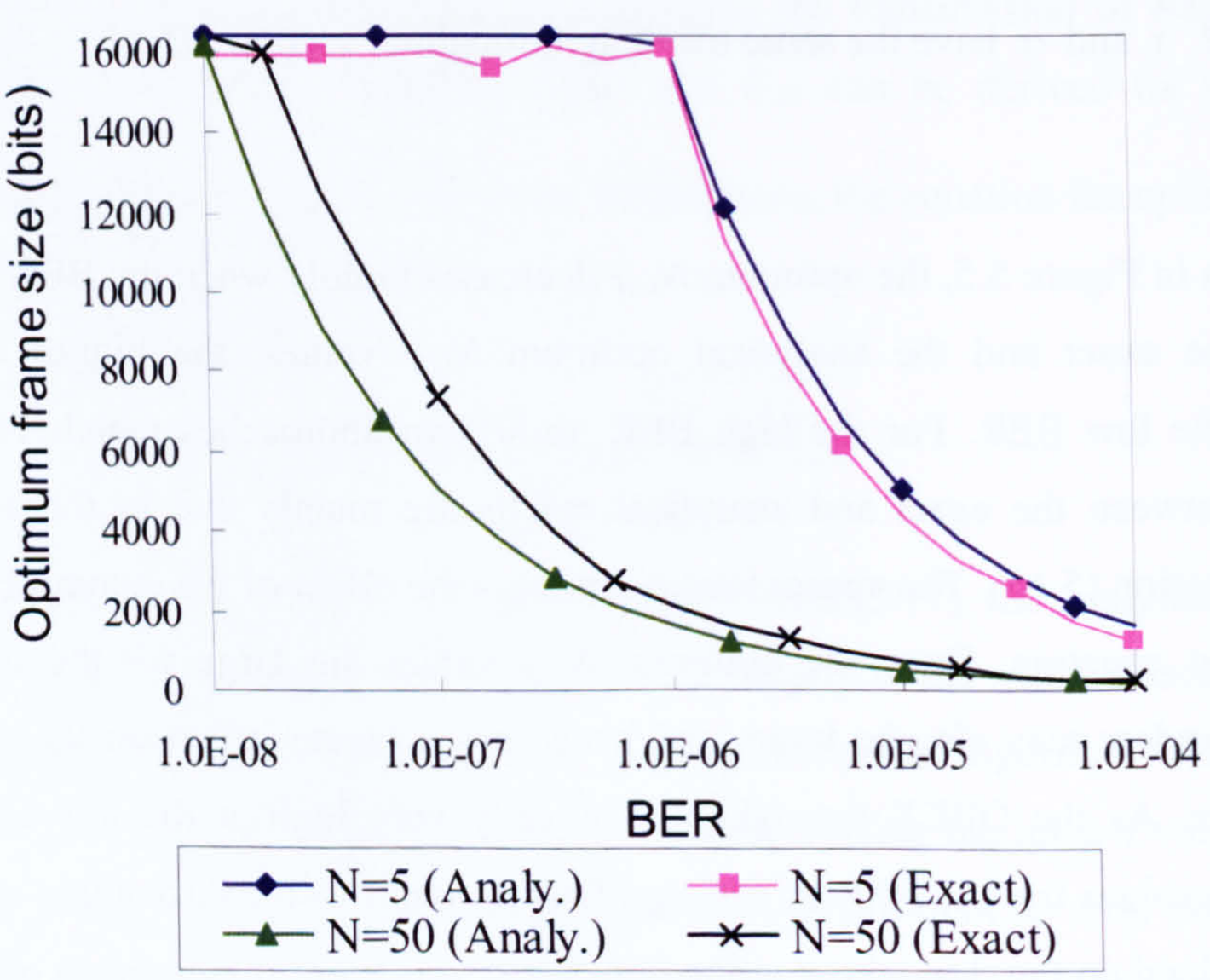


Figure 5.7 Comparison of optimum frame sizes when using exact algorithm and analytical equations for two fixed window sizes of 5 and 50. N is the acronym of N_{LAP}

$\bar{x} = 527$ bits and $\sigma = 212$ bits for N=5, $\bar{x} = 1072$ bits and $\sigma = 1280$ bits for N=50.

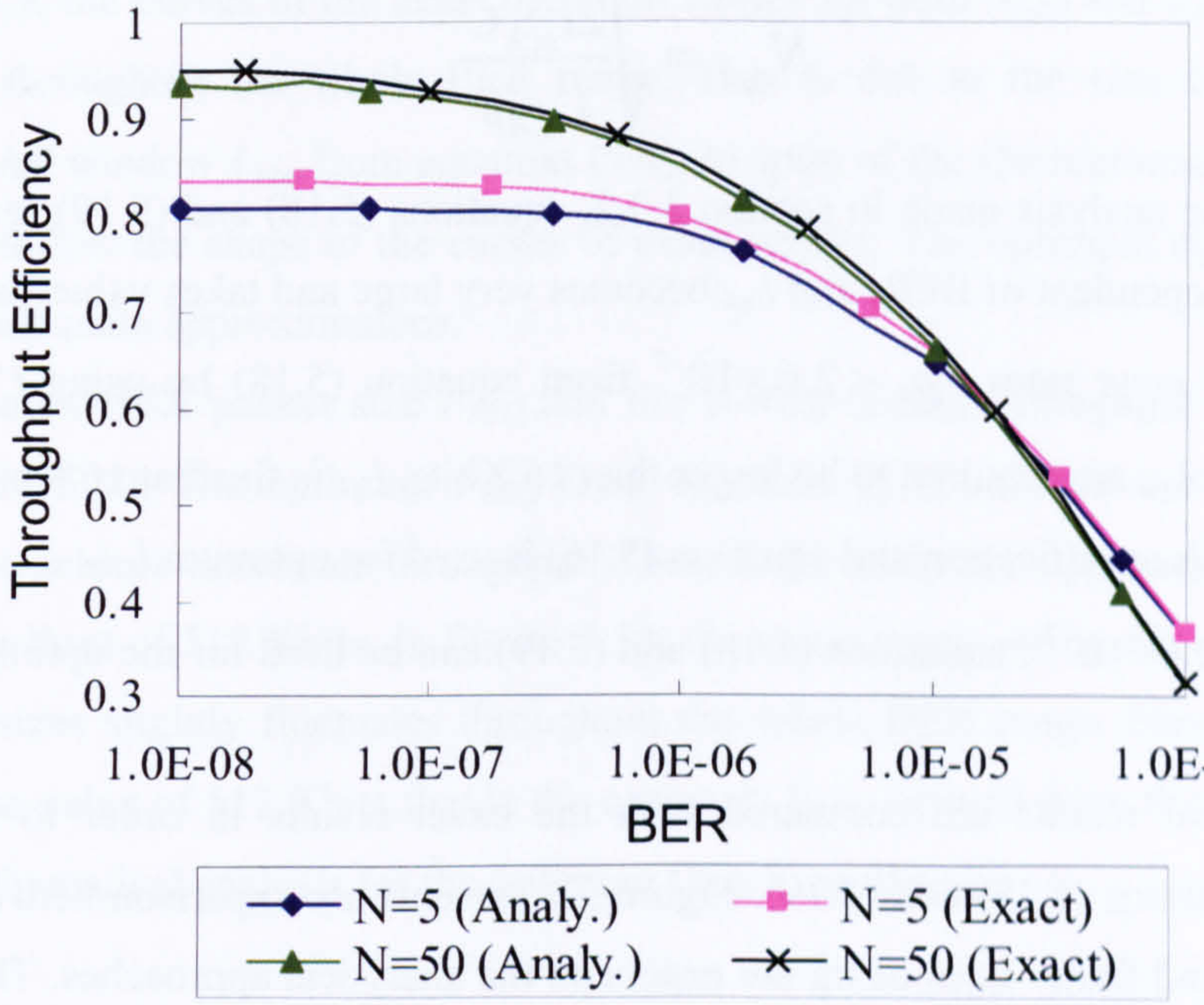


Figure 5.8 The corresponding throughput efficiency of Figure 5.7

$\bar{x} = 0.018$ and $\sigma = 0.008$ for N=5, $\bar{x} = 0.010$ and $\sigma = 0.003$ for N=50.

In Figure 5.7, the optimum l_{LAP} has large values at the low BER and then decreases rapidly as the BER increases. There are some differences between the analytical and the exact results especially for the low BER. This is also due to the approximation made in (5.13) as discussed later. The difference for optimum l_{LAP} between the two approaches at low BER does not however affect the OBEX throughput significantly as shown in Figure 5.8. Thus, using equation (5.17) for optimum l_{LAP} is adequate as well.

5.4.3 Simultaneous optimization of IrLAP window and frame sizes for maximum OBEX throughput

The best possible OBEX throughput can be achieved when N_{LAP} and l_{LAP} are simultaneously optimized for a given bit error rate. The optimum N_{LAP} and l_{LAP} are obtained by calculating first derivative of throughput equation (5.15) with respect to N_{LAP} and l_{LAP}, $\frac{\partial F_{OBEX}}{\partial N_{LAP}} = \frac{\partial F_{OBEX}}{\partial l_{LAP}} = 0$. After some calculations and approximations, the simultaneous optimum values of N_{LAP} and l_{LAP} are derived:

$$l_{opt} = \sqrt{\frac{l'_{LAP}}{p_b}} \tag{5.18}$$

and

$$N_{opt} = \sqrt{\frac{2t_{ack}C}{l'_{LAP}}} \tag{5.19}$$

Similar to the analysis made in section 3.3.2, equations (5.18) and (5.19) reveal that N_{opt} is essentially independent of BER, and l_{opt} becomes very large and takes values larger than 16 Kbits at low bit error rates ($p_b < 2.6\times10^{-7}$ from equation (5.18) by using $l'_{LAP} = 72\text{bit}$). When the results l_{opt} are required to be larger than 16 Kbits, l_{opt} is fixed at 16 Kbits in order to comply with IrDA specification and equation (5.16) is used for optimum l_{LAP}.

For BER $> 2.6\times10^{-7}$, equations (5.18) and (5.19) can be used for the optimum N_{LAP} and l_{LAP} values.

The analytical results are compared with the exact results in order to examine the accuracy of equations (5.18) and (5.19). Figure 5.9 presents a comparison between optimum IrLAP window and frame sizes using the exact and the analytical approaches. The results are plotted against the BER in the range of 10^{-8}~10^{-4}.

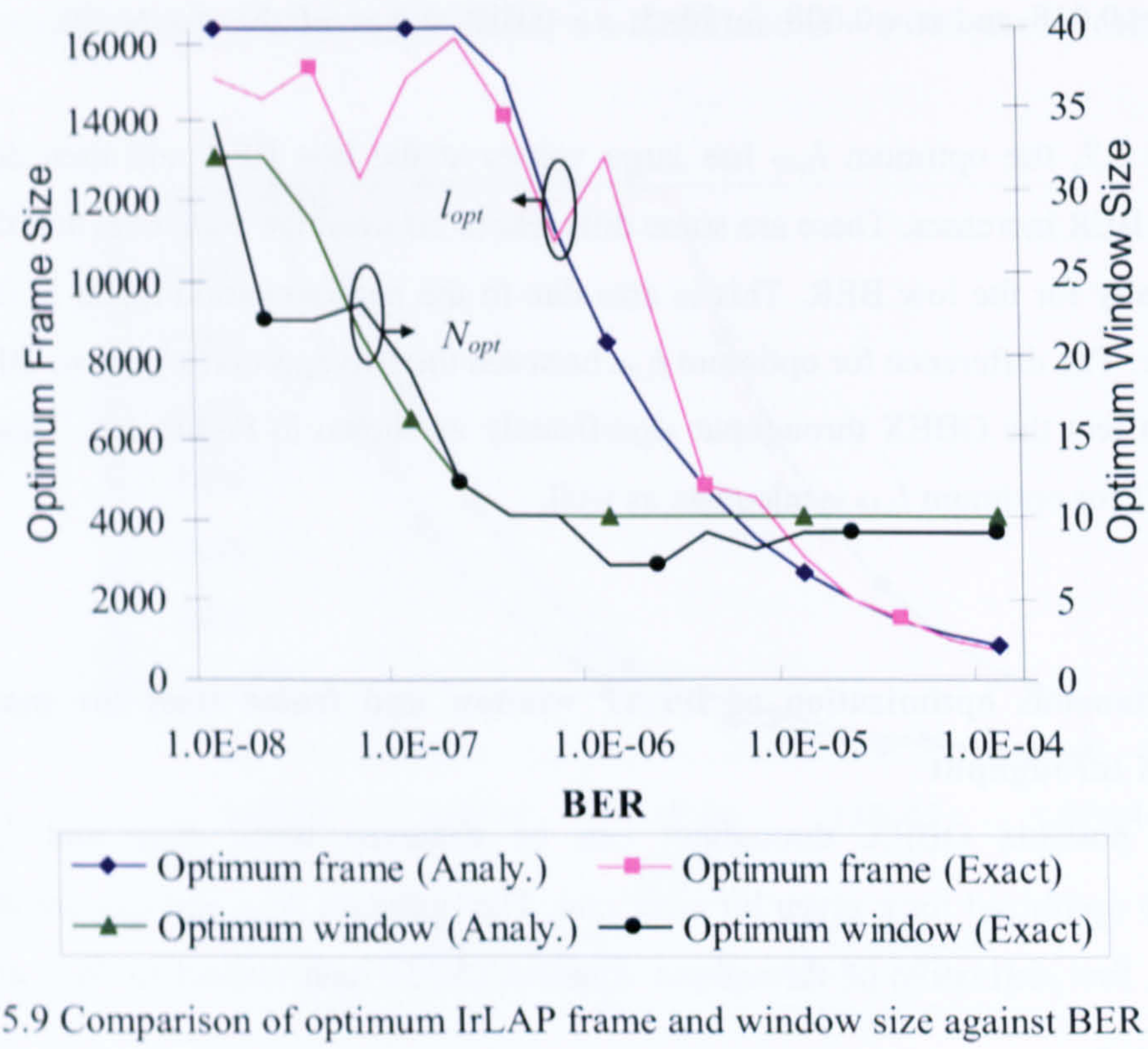


Figure 5.9 Comparison of optimum IrLAP frame and window size against BER using the exact and the analytical results

$\bar{x} = 1160\,\text{bit}$ and $\sigma = 1357\,\text{bit}$ for the optimum IrLAP frame size, $\bar{x} = 2.18$ and $\sigma = 2.51$ for the optimum IrLAP window size

In Figure 5.9, the curves of the exact optimum values for both N_{LAP} and I_{LAP} show a non-smooth shape throughout the whole BER range. This is due to the size changes of the incomplete IrLAP window L_{rem} from equation (5.4). In spite of the fluctuations, the analytical results closely follow the shape of the curves of exact results. The optimum equations (5.18) and (5.19) are adequate approximations.

The optimum OBEX packet size P_{REQ} and the corresponding throughput efficiency are shown in Figure 5.10. The optimum P_{REQ} from equation (5.15) should use its maximum allowed value to achieve maximum throughput. Thus, in the analytical approach, the optimum P_{REQ} uses upper limit of 512 Kbits. In Figure 5.10, the curve representing the exact optimum OBEX packet sizes slightly fluctuates throughout the whole BER range. Nevertheless, it is very close to the value of 512 Kbits that is the optimum P_{REQ} derived from the analysis. This verifies the mathematical analysis for the optimum OBEX packet size.

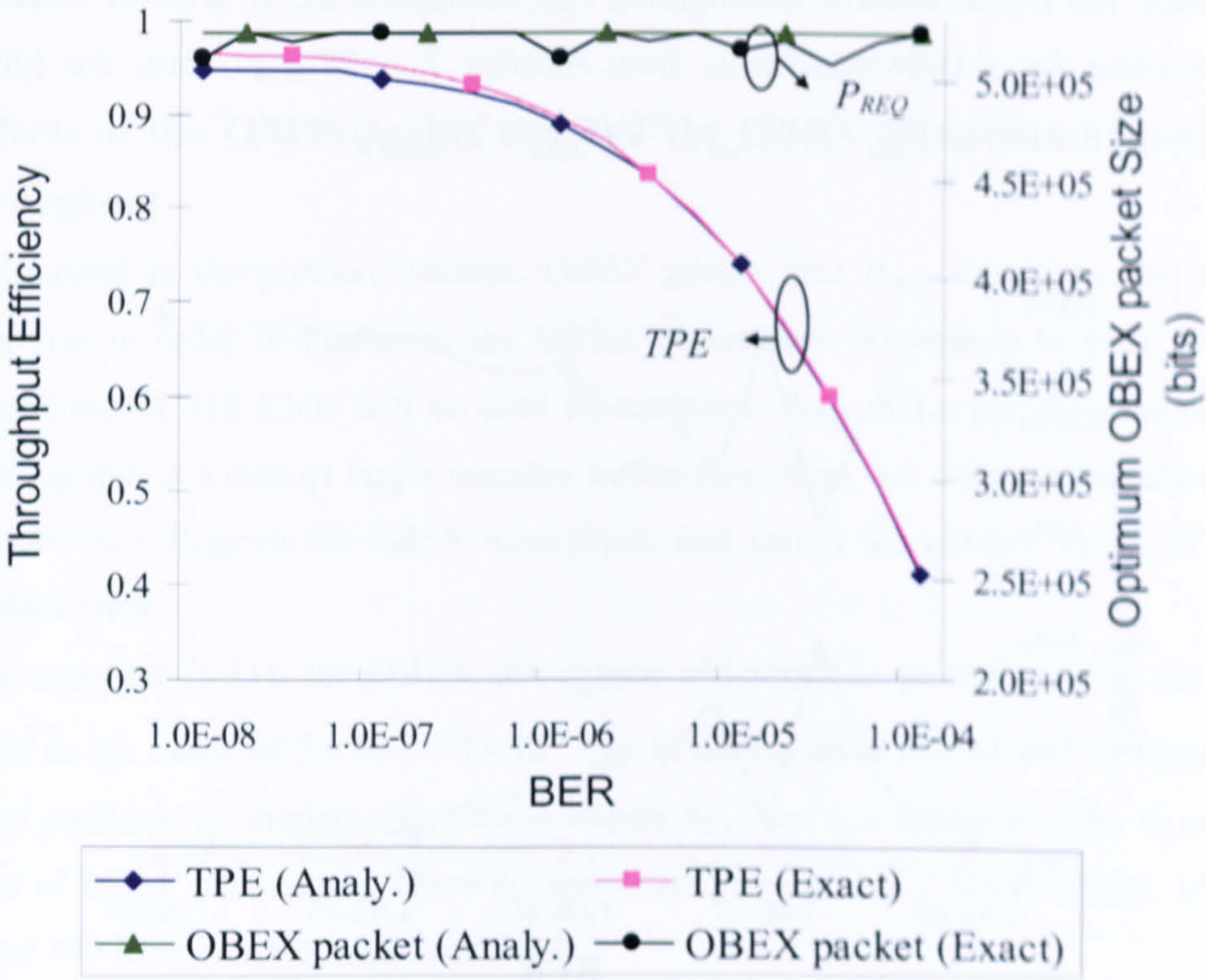


Figure 5.10 Comparison of the optimum OBEX packet size against BER using the exact and the analytical results; the corresponding OBEX TPE using the optimum parameters $\bar{x} = 3855$ bit and $\sigma = 5315$ bit for the optimum OBEX packet size, $\bar{x} = 0.008$ and $\sigma = 0.006$ for the throughput efficiency

The corresponding OBEX throughput efficiencies using optimum P_{REQ}, N_{LAP} and I_{LAP} are shown in Figure 5.10. Only a small difference is observed on the OBEX throughput between

the exact results and the analytical results. The two throughput efficiency curves match well especially at the high BER. Additionally, the OBEX system using optimum parameters shows considerable performance improvements when compared it to the non-optimized cases shown in Figure 5.4. Significant improvement is achieved at the high BER when the optimum parameters are used. For instance, OBEX throughput efficiency of over 0.4 is achieved despite the high BER (BER=10^{-4}), while throughput efficiency of 0.1 or less is observed for the non-optimised cases at the same BER.

5.4.4 Effects of different protocol tasks on the OBEX throughput

The relative time consumed on each of the protocol tasks is examined in this section. According to equation (5.10), the average time T to transmit one OBEX packet consists of the following components: T_{full}, T_{rem}, T_{RES} and T_{OBEX}. The transmission is separated into three protocol tasks: full IrLAP window transmission T_{full}, incomplete IrLAP window transmission T_{rem} and waiting for acknowledgements from receiver $T_{RES}+2T_{OBEX}$. Thus, the following equation always holds true $T_{full}/T + T_{rem}/T + (T_{RES} + 2T_{OBEX})/T = 1$.

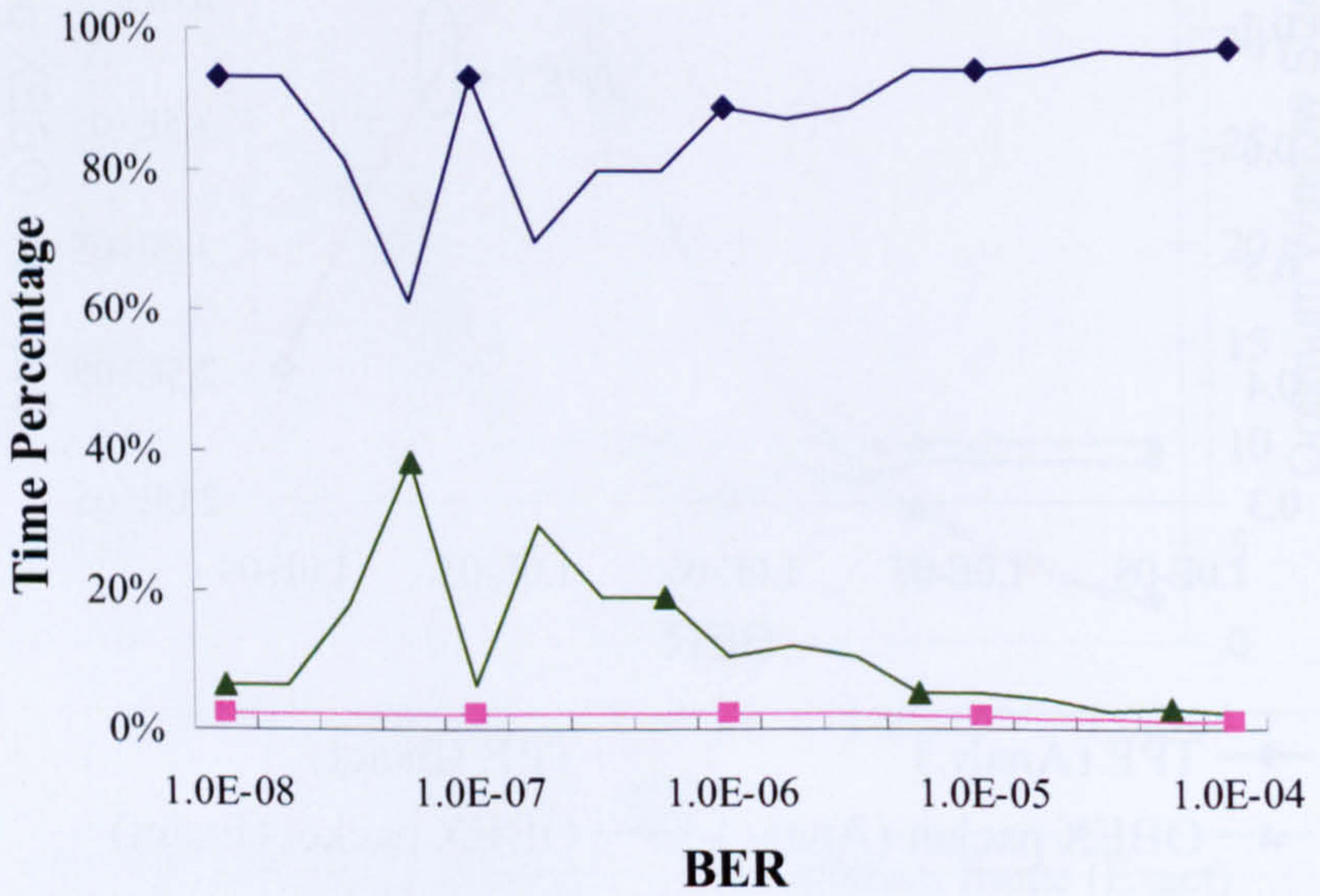


Figure 5.11 Time percentage against the BER for three different OBEX protocol tasks

The time percentage of the three different protocol tasks is plotted against the BER in Figure 5.11. Optimum protocol parameters (P_{REQ}, N_{opt} and l_{opt}) are used. As shown in Figure

5.11, full window transmission that carries the main body of an OBEX packet is the dominant protocol task for the whole BER range. Incomplete window transmission represents the last part of an OBEX packet which can not fill up one full IrLAP window. It fluctuates in a complementary manner with the full window transmission. Finally, the acknowledgement time portion that represents the time spent on waiting for acknowledgements from the receiver, is very small compared to the other two time portions. The fluctuation of the full and the incomplete window transmission is due to the changing BER and the corresponding changing optimum N_{LAP} and I_{LAP} values. Although incomplete window transmission takes a large time percentage at the low BER, it only consumes a negligible time percentage compared to the full window transmission at the high BER. This justifies the assumption made for equation (5.13) holds true in the case of high BER. Recall that OBEX throughput gains significantly with the optimum parameters only for the high BER, the approximation in (5.15) is therefore sufficient to derive the optimum parameters in sections 5.3.2 and 5.3.3.

5.5 Effects of the OBEX packet size and the OBEX turnaround time on the throughput

As discussed in the previous section, OBEX packet size P_{REQ} should be set to in the maximum size in order to maximize the OBEX throughput. According to [45], the upper packet size limit of 512 Kbits will be used for optimum P_{REQ}. A Larger P_{REQ} improves the OBEX throughput at a cost of larger memory buffer size. Next, we examine the effect of the OBEX packet size P_{REQ} on the OBEX throughput, and search for suitable values of P_{REQ} at different data rates.

Using equation (5.11), the OBEX throughput efficiency is plotted against the OBEX packet size in the range of $5\times10^4 \sim 1\times10^6$ bits at the bit error rate of 10^{-6} in Figure 5.12. Results are produced by implementing the optimum N_{LAP} and I_{LAP} values in order to minimize the effects of IrLAP parameters. Three different data rates of 115.2 Kbit/s (SIR), 16 Mbit/s (VFIR) and 100 Mbit/s (UFIR) are considered.

For all three data rates, the corresponding OBEX throughput efficiencies increase with the OBEX packet sizes P_{REQ}. The OBEX throughput of the 100Mbit/s link benefits most as P_{REQ} increases compared to the throughput of the other two lower data rate links. For the 100 Mbit/s links, the throughput efficiency of over 0.75 is achieved when the OBEX packet size P_{REQ} increases to 1 Mbits in combination with the use of optimum N_{LAP} and I_{LAP}. However, P_{REQ} size of 500 Kbits and 100 Kbits are sufficient for 16 Mbit/s and 115.2 Kbit/s respectively with the use of optimum N_{LAP} and I_{LAP}. Larger P_{REQ} does not necessary since it leads to only marginal improvement on the throughput.

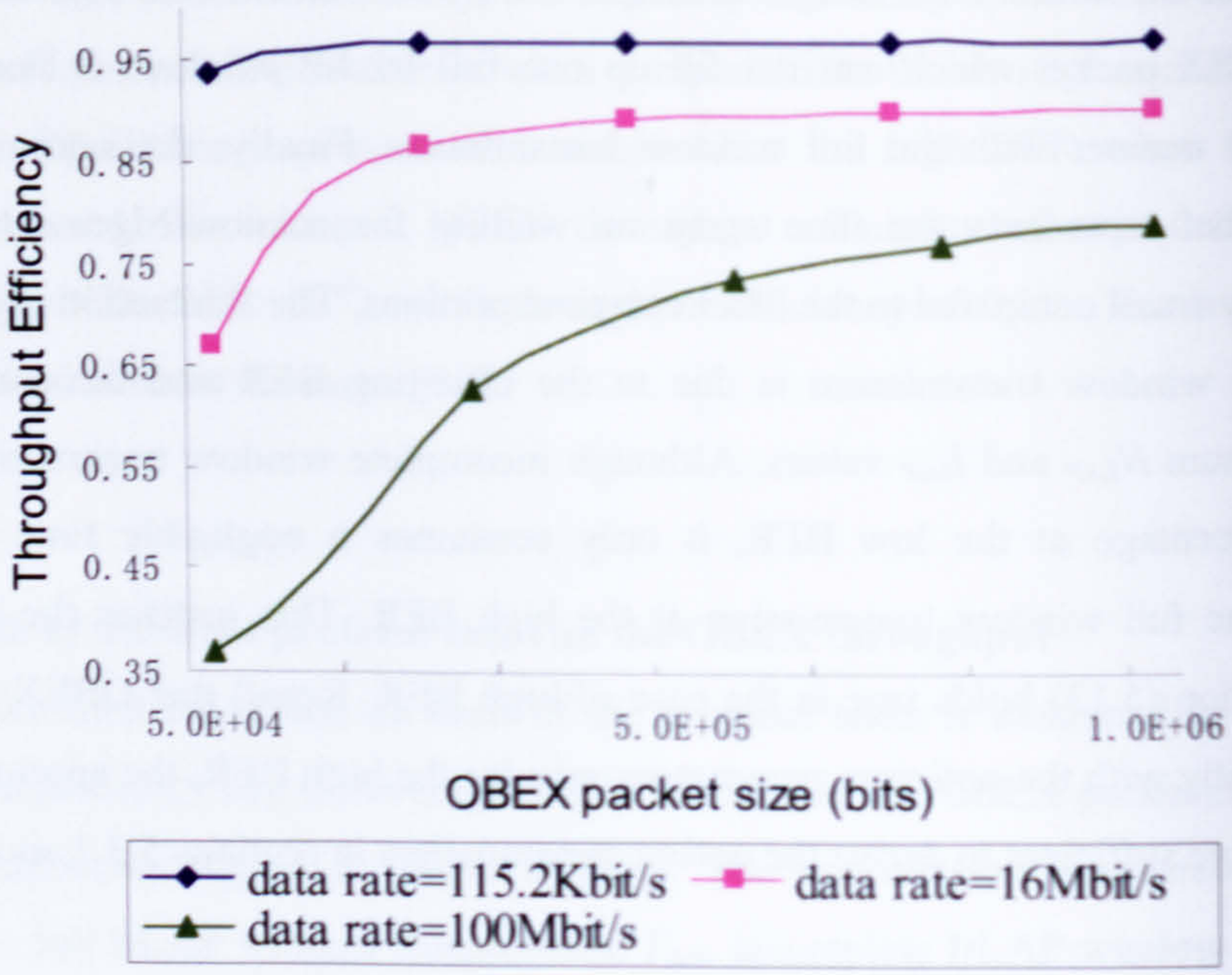


Figure 5.12 Throughput efficiency against OBEX packet size at different data rates

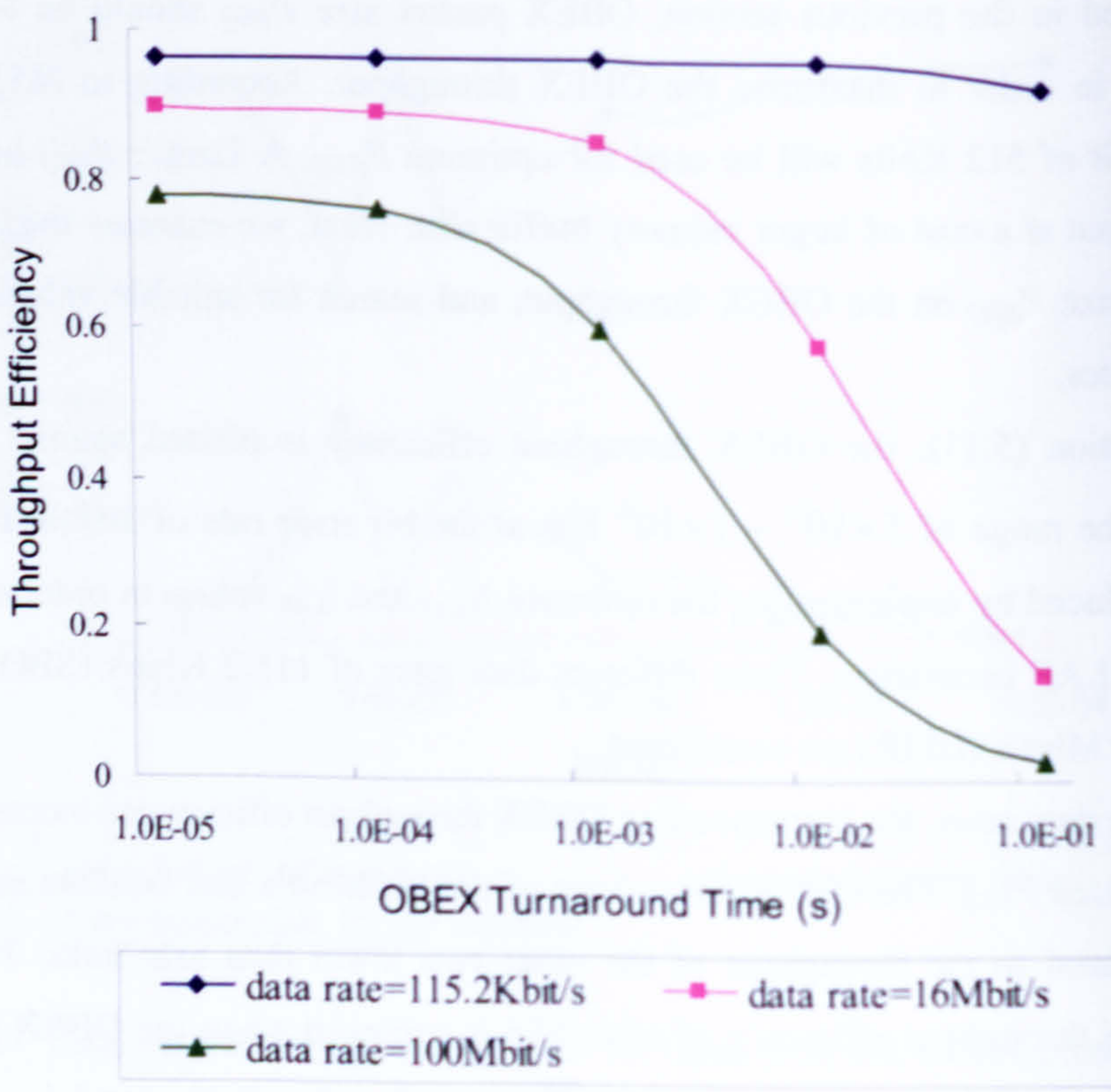


Figure 5.13 Effect of OBEX turnaround time on OBEX throughput at different data rates

OBEX turnaround time T_{OBEX} of 2ms is used in all the previous result figures. This high layer turnaround time depends on the CPU speed of communication peers rather than IrDA transceivers. The study of the effect of OBEX turnaround time on the OBEX throughput and recommend suitable value of T_{OBEX} for OBEX at various data rates is provided next.

In Figure 5.13, The OBEX throughput efficiency is plotted against the OBEX turnaround time in the range of $1\times10^{-5} - 1\times10^{-1} s$ at the bit error rate of 10^{-6}. In order to highlight the effect of T_{OBEX} and to minimize the effects of other factors, optimum P_{REQ}, N_{LAP} and I_{LAP} values are used. Three different data rates of 115.2 Kbit/s, 16 Mbit/s and 100 Mbit/s are considered.

The OBEX throughput efficiency decreases with the increase of the OBEX turnaround time. The high data rate links are more sensitive to a large T_{OBEX} than the low data rate links. At BER=10^{-6}, a very poor OBEX throughput efficiency is observed for 100 Mbit/s when T_{OBEX}=0.1s, while throughput efficiency of over 0.9 for 115.2 Kbit/s is possible for the same T_{OBEX}. For the 100 Mbit/s links, T_{OBEX} of less than 10^{-4} s is necessary, while for 16 Mbit/s or less, T_{OBEX} at the order of 10^{-3} s is sufficient.

5.6 IrDA Burst (IrBurst) protocol

IrBurst is an infrared high speed transmission session protocol targeting the large size information transmission [47]. Currently, OBEX is the most popular protocol for managing the object exchange and has been widely implemented into many IrDA enable devices. However, OBEX protocol is designed to mainly transmit small data objects and is inefficient to transmit a large size object. The reason is that OBEX manages the transmission in the manner of 'Request and Response' as described in section 5.1. For each OBEX packet the sender sends, the sender has to wait for the acknowledgement from the receiver before sending the next OBEX packet. The OBEX standard limits the maximum OBEX packet size to 512 Kbits, which means that large objects have to be split into packets. Since a large size object will be sent in a number of OBEX packets, the 'Request and Response' scheme means that the sender has to spend a long time waiting for the unnecessary turnaround time and therefore is clearly not suitable for the transmission of a large size object.

In the near future, the data rate of the IrDA connections will inevitably increase. The standard for Ultra Fast infrared (UFIR) that supports the data rate up to 100 Mbit/s, is already under development. The technology of outdoor infrared network has already reached Gbit/s order. It can be envisioned that the outdoor infrared technology can be imported to the IrDA technology once the cost of the components that enable fast data rates become more

acceptable. The fast improvement of the data rates initiates the development of new IrDA applications and services that often require the exchanging of large size objects.

One of such applications being developed is the 'content' vending machine proposed by the Japanese mobile service provider NTT Do-Co-Mo. The concept of the 'content' vending machine is very similar to the traditional vending machine. However, instead of selling snacks and drinks, the 'content' vending machine sells digital content, including MP3 songs, mobile ring tones, animated screen savers or games, as well as enabling the uploading and printing of digital pictures. The 'content' vending machine, equipped with an IrDA port, sells the digital content through the IrDA connection to the customer's mobile devices. Because of the increasing speed of the IrDA links and the increasing size of digital information content, the short distance transmission issue—known as the 'last meter problem' [8], becomes more and more evident. In order to solve the issue, a system that enables high speed transmission of large size objects between mobile devices becomes essential.

In order to provide a compatible environment for future requirements, IrDA recently proposed IrBurst, a protocol that provides a complementary capability to OBEX. In contrast to OBEX, IrBurst is designed to fulfil the requirement for transmitting bursts of data (e.g. MP3 songs, video clips). When a user points his/her IrDA mobile device to another IrDA equipment, the user can send or receive one or more large size objects in a very shot period. This feature of IrBurst is called "Point and Exchange". More specifically, the feature is called "Point and Receive" in the case of downloading and "Point and Send" in the case of uploading. There are three usage models defined in the IrBurst standard: a) local connections between mobile devices and servers (i.e. home or office network); b) connection between mobile devices; c) remote connection between mobile devices and servers through backbone network (i.e. 'content' vending machine).

IrBurst provides a fast and efficient connection that allows for high-speed object exchange. In order to eliminate the effect of the slow OBEX turnaround and increase the efficiency in transmitting large size objects, IrBurst transmits information by operating control and data channels separately [47], as shown in Figure 5.14. The data channels are dedicated to information transmission for different applications. The control channel is dedicated to sending the control commands (i.e. 'connect', 'pause', 'disconnect') of all the data channels that operate as data streams. Each channel is assigned a unique LSAP-SEL. The reliability of IrBurst is based on the connection oriented services provided by TinyTP, as well as the IrLAP error retransmission mechanism. Unlike in the case of OBEX, the information sent is not peer acknowledged at the IrBurst layer. This feature which is clearly suitable for low delay bursts of streams, avoids the following:

- Discarding extra packet overheads which are required when OBEX is used.
- Being able to send burst data information.

- No need to wait for the acknowledgement packets

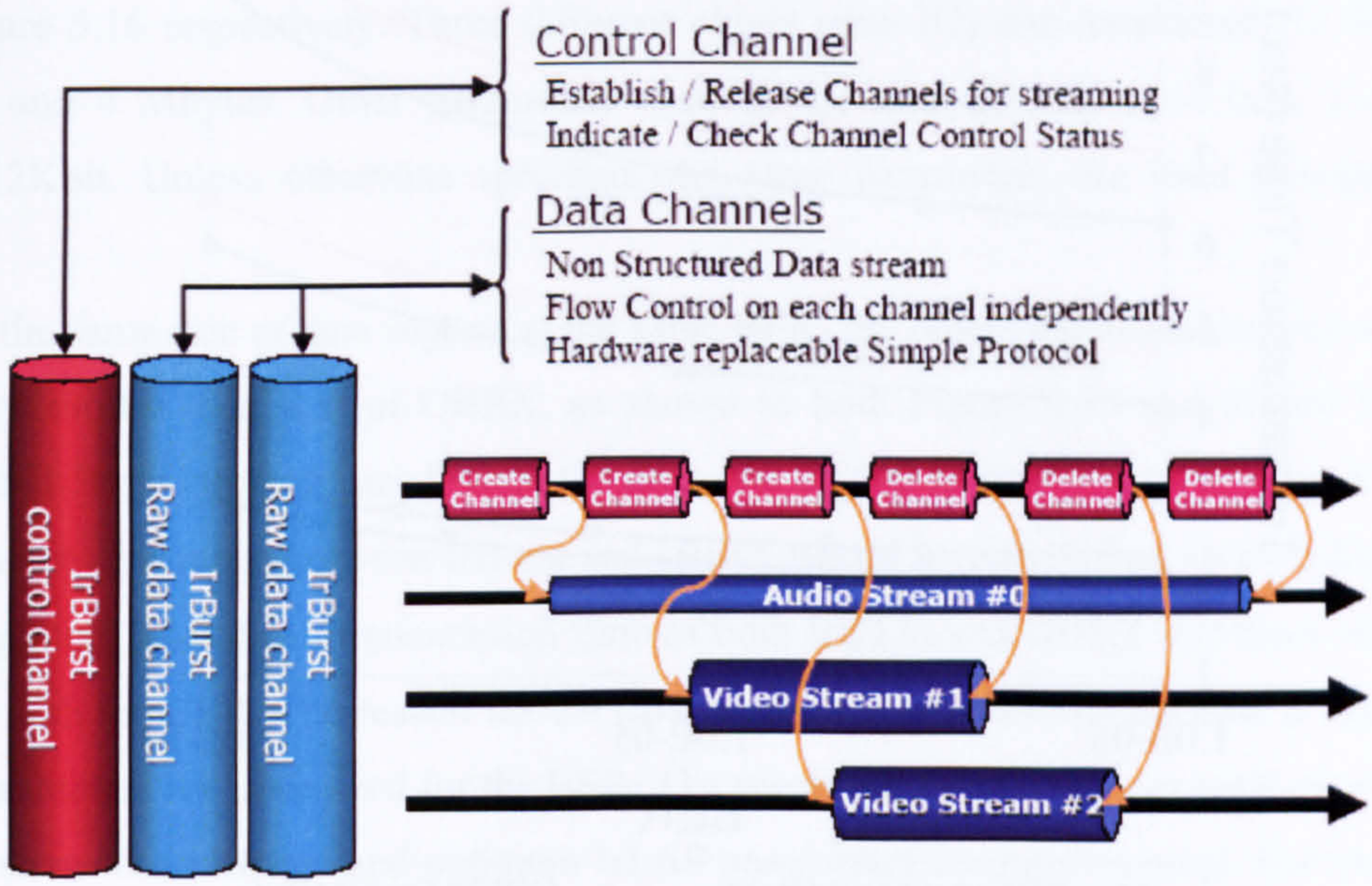


Figure 5.14 IrBurst channels organization

The following analysis assumes that the control channel does not have data to send while the data channels are transmitting. Thus, the control channel does not interrupt the data transmission, although it remains possible in the real situations depending on circumstances. The control channel buffer in Figure 5.14, shown next to the data channels, is therefore assumed to be empty in this study.

Similarly to OBEX, IrBurst is also designed to run on top of TinyTP. Since IrBurst does not send any extra overhead information in the data channel and does not transmit data in the form of packets, the transmission model of IrBurst is the same as the one of TinyTP as given in Figure 4.3. Therefore, the TinyTP mathematical model derived in section 4.2 and 4.5 is also applicable for the analysis of IrBurst.

5.7 Performance comparison of IrBurst and OBEX

In this section, the performance of IrBurst and OBEX is compared. In order to highlight the performance difference of these two protocols, large size objects are considered for the transmission. It is assumed that the receiver buffer size is sufficient to accommodate all the incoming packets and only one IrLMP connection is active. The optimum IrLAP window and frames sizes are used.

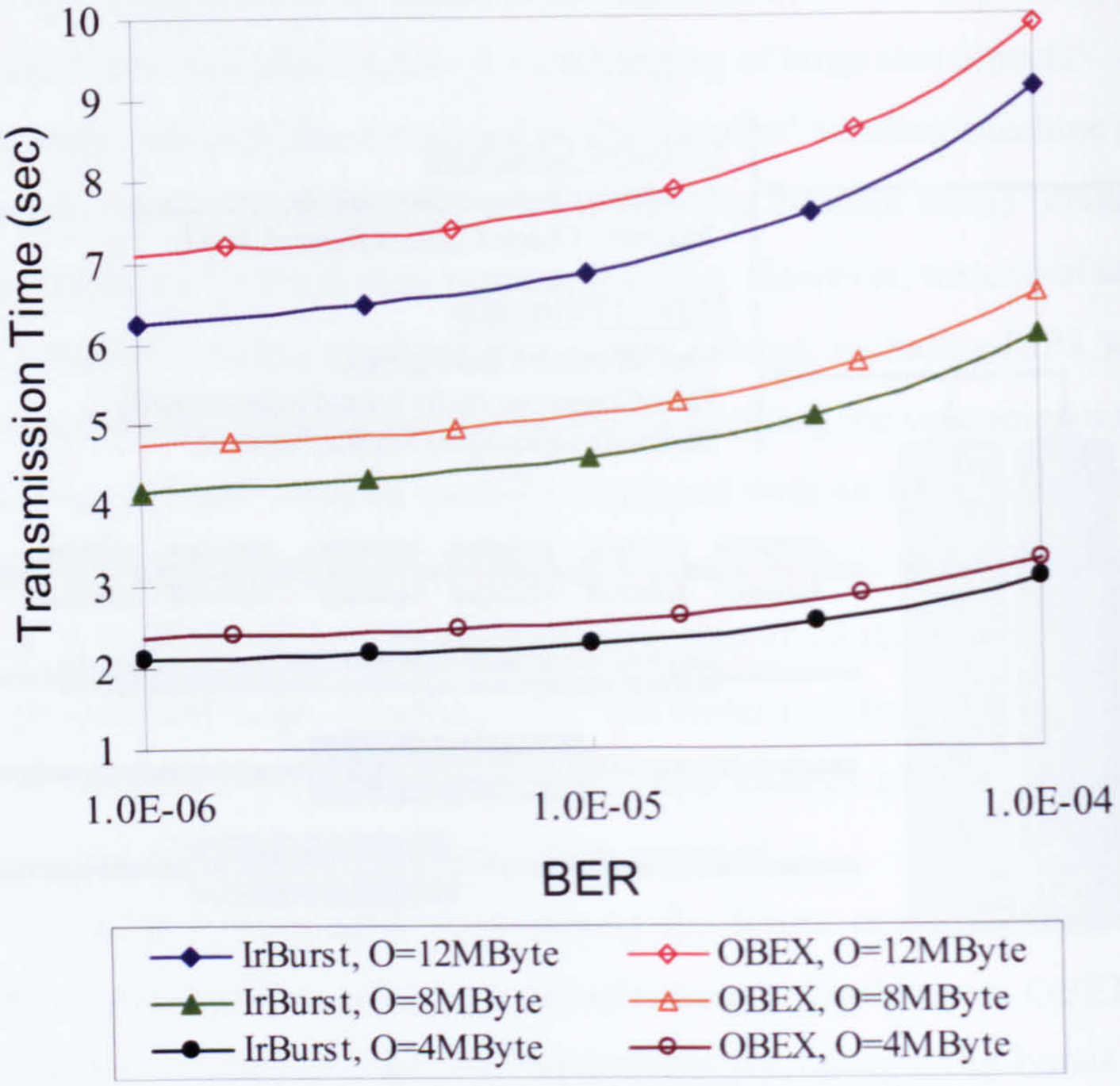


Figure 5.15 Transmission time against bit error rate, *C*=16Mbit/s

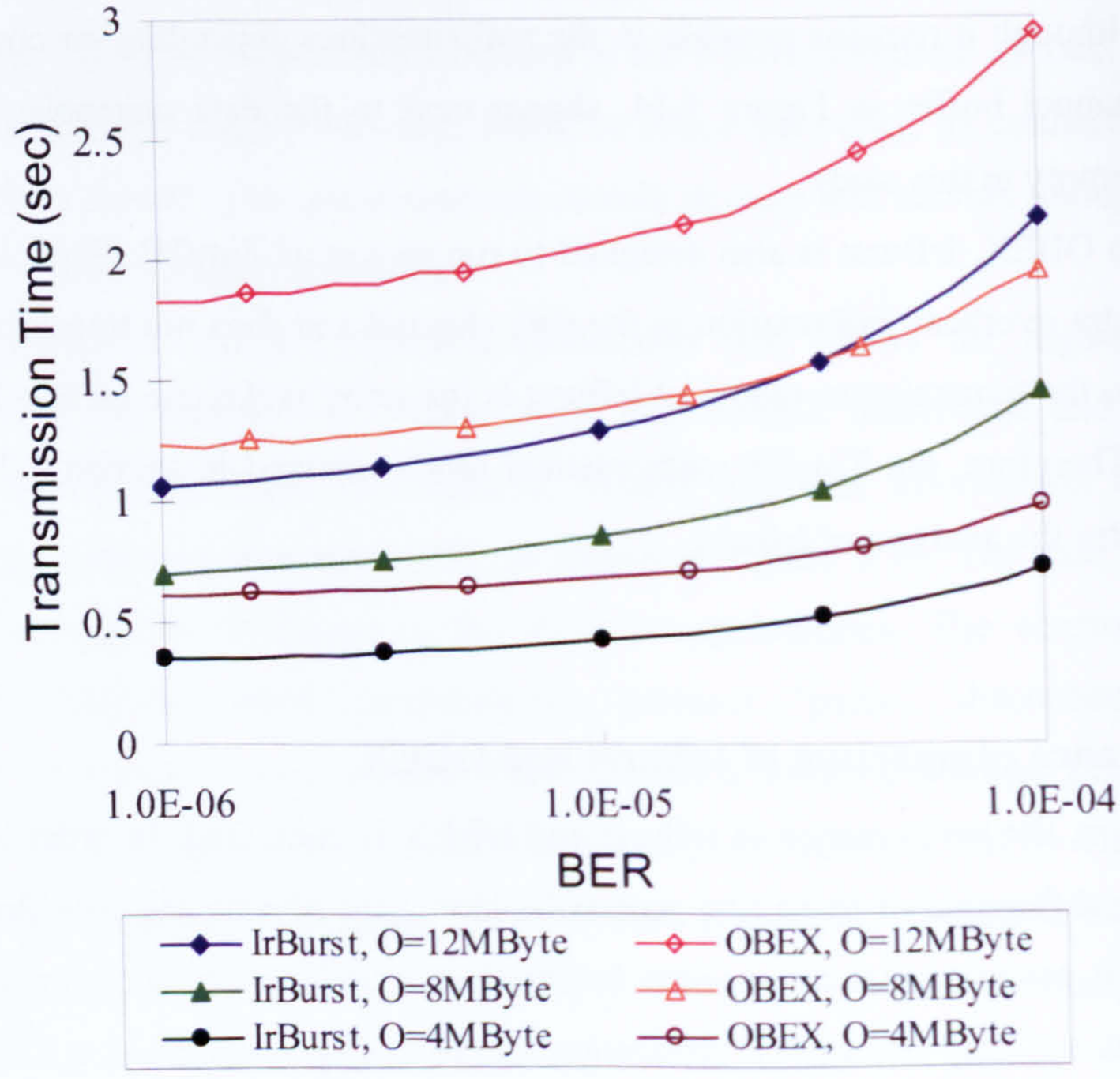


Figure 5.16 Transmission time against bit error rate, *C*=100Mbit/s

In Figure 5.15 and Figure 5.16, the object transmission time is plotted against the bit error rate in the range of 10^{-6} to 10^{-4}. Date rates of 16 Mbit/s and 100 Mbit/s are used in Figure 5.15 and Figure 5.16 respectively. Three different object sizes (O) are considered: 12 Mbytes, 8 Mbytes and 4 Mbytes. Other parameters used in the analysis are: t_{ta}=0.1ms, T_{OBEX}=2ms, P_{REQ}=512Kbit. Unless otherwise specified, the same parameters are used throughout this section.

For the same size of data objects at the same BER, the object transmission time of IrBurst is always shorter than that of OBEX, as shown in both Figure 5.15 and Figure 5.16. The benefit of IrBurst is emphasized when the data rate of 100 Mbit/s is used (Figure 5.16). The performance difference between IrBurst and OBEX would increase when an even higher data rate is applied. The object transmission time of both IrBurst and OBEX increases slowly and steadily with the BER. The reason for the slow increment of transmission time is because the IrDA parameters are optimized for the BER. The result shows an excellent performance of the IrDA system when IrBurst and optimum IrLAP parameters are implemented. For instance, at the data rate of 16Mbit/s and 100 Mbit/s, it takes IrBurst less than 0.5 second and 2 seconds to transmit a 4 MByte object, which is the typical size of an MP3 song. Such a short transmission time is very desirable for the 'point and shoot' IrDA applications.

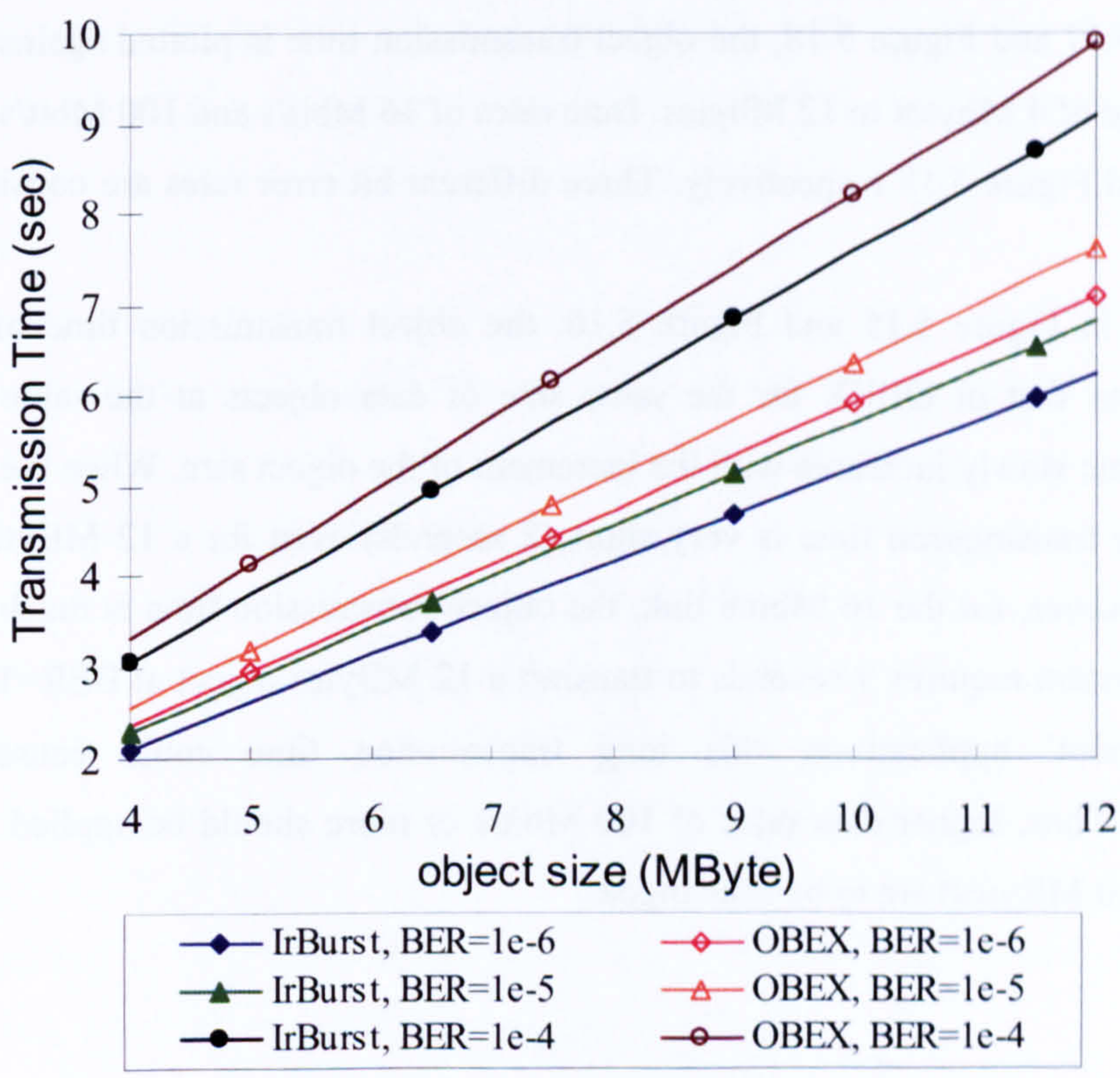


Figure 5.17 Transmission time against object size, C=16Mbit/s

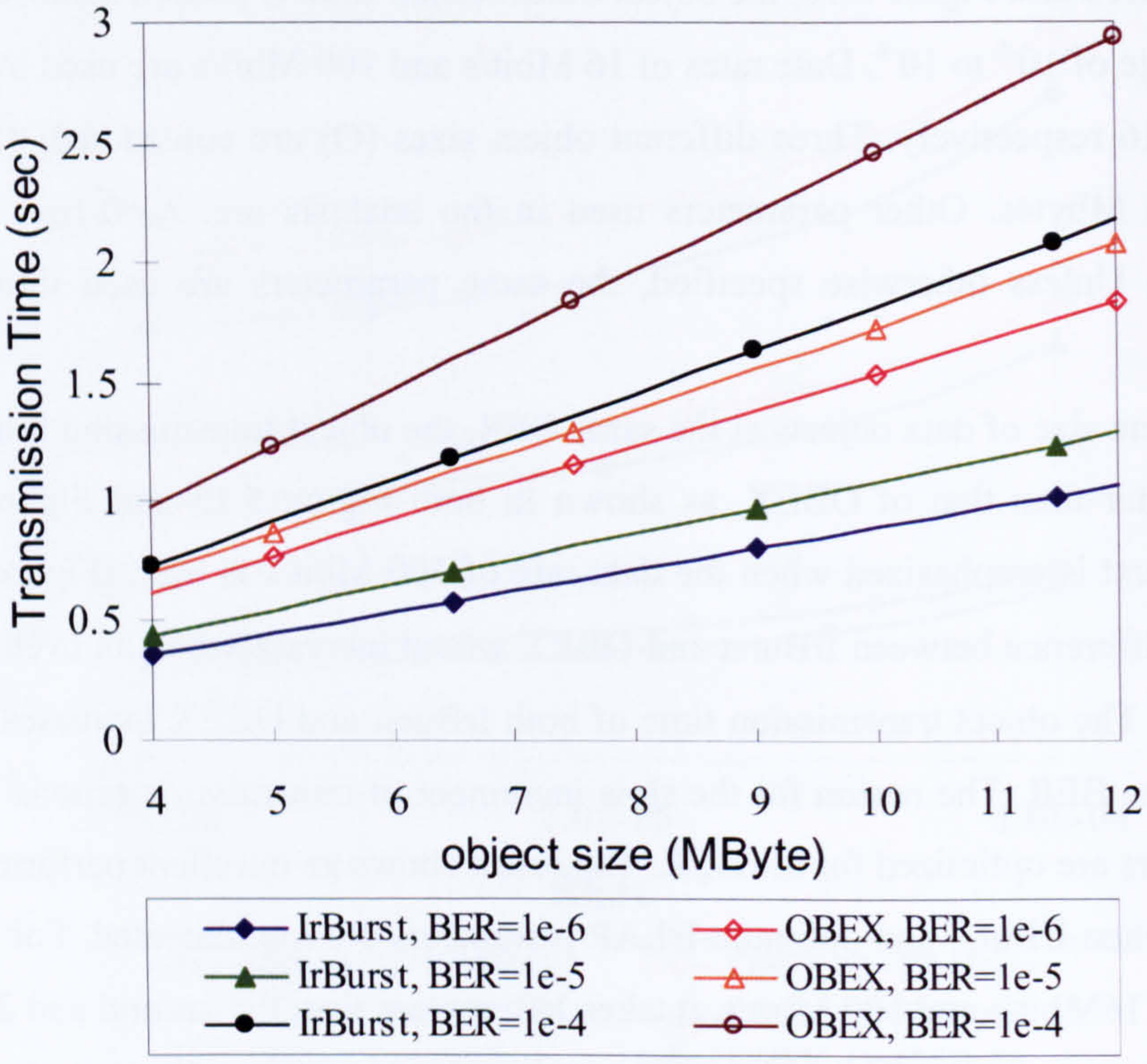


Figure 5.18 Transmission time against object size, C=100Mbit/s

In Figure 5.17 and Figure 5.18, the object transmission time is plotted against the object size in the range of 4 Mbytes to 12 Mbytes. Date rates of 16 Mbit/s and 100 Mbit/s are used in Figure 5.17 and Figure 5.18 respectively. Three different bit error rates are considered: 10^{-6}, 10^{-5} and 10^{-4}.

As shown in Figure 5.15 and Figure 5.16, the object transmission time of IrBurst is always less than that of OBEX for the same size of data objects at the same BER. The transmission time slowly increases with the increment of the object size. When the data rate is 100 Mbit/s, the transmission time is very short (2 seconds) even for a 12 MBytes object at BER=10^{-4}. However, for the 16 Mbit/s link, the object transmission time is much longer, for instance, the system requires 9 seconds to transmit a 12 MBytes object at BER=10^{-4}. For the 'point and shoot' applications, this long transmission time could cause customer dissatisfaction. Thus, higher data rates of 100 Mbit/s or more should be applied when large objects (up to 10 MBytes) are to be exchanged.

5.8 Summary of the Chapter

In this chapter, two of the IrDA session layer protocols, OBEX and IrBurst are investigated. The OBEX throughput equation is derived and subsequently verified by

simulations. OBEX packet size, IrLAP window and frame size are optimized to achieve the maximum OBEX throughput. A study of the effects of the maximum OBEX packet size (P_{REQ}) and the OBEX turnaround time (T_{OBEX}) is also carried out. For the 100 Mbit/s links, P_{REQ} of 1 Mbits is required to achieve good throughput efficiency. P_{REQ} of 500 Kbits and 100 Kbits are sufficient for 16 Mbit/s and 115.2 Kbit/s respectively. For the 100 Mbit/s links, T_{OBEX} of less than 10^{-4} s is necessary, while for 16 Mbit/s or less, an OBEX turnaround time of 10^{-3} s is sufficient to achieve good throughput efficiency. The performance of IrBurst and OBEX is compared. IrBurst shows an excellent performance compared to that of OBEX when transmitting large size files.

In the next chapter, two rapidly developing IrDA applications, IrDA financial messaging (IrFM) and IrDA simple connection (IrSC), are to be investigated.

Chapter 6. Study of Two IrDA Applications: IrFM and IrSC

6.1 Introduction

Infrared Financial Messaging (IrFM) and IrDA Simple connection (IrSC) are the two rapid developing IrDA applications that generate the most public interests and have the most promising future.

The rapid advancement of both device technology and users' capabilities, is the foundations of a new digital payment system which promises to change the traditional way of payments. By utilizing the IrDA technology, IrFM is a financial messaging specification with great prospects as the future digital payment system. In this chapter, a comprehensive analysis of IrFM is carried out. The analysis clarifies the benefits of using such a system and develops a model which identifies the significance of major performance factors of the digital payment systems. Issues concerning connection delay, security, reliability, simplicity and interoperability of IrFM are investigated. The performance of the competing digital payment technologies such as Radio Frequency Identification (RFID) is also compared with that of IrFM.

Proposed by IrDA, IrSC protocol is a protocol designed for fast connection and instant data transmission of small objects over point-to-point infrared links [134]. Although the protocol is still under development, it has attracted both manufacturers and researchers. The IrDA special interest group proposed two different technical approaches to address the requirements of IrSC. In this chapter, the technical approach 1 (IrSC1) and approach 2 (IrSC2) are examined. The analysis carried out in this chapter clarifies the issues of the design of IrSC, compares the performance of the standard IrDA connection, IrSC1 and IrSC2, as well as providing suitable design guidelines for IrDA devices for high performance of IrSC.

The outline of this chapter is as follows: Section 6.1 provides overviews of IrFM and IrSC specifications. Since both IrFM and IrSC require a short device connection delay, the IrDA device discovery and connection procedures are investigated in section 6.2. In section 6.3, a comprehensive analysis is carried out for IrFM covering a wide range of important issues of the digital payment systems including security, reliability, interoperability, simplicity and transaction delay. Additionally, the current market status of IrFM is presented and the competing digital payment technologies are compared with IrFM. Finally, in section 6.4, the performance of the standard IrDA connection, IrSC1 and IrSC2 is compared. A set of design guidelines is also proposed for the design of IrSC.

6.2 Overviews of IrFM and IrSC

6.2.1 IrFM

Smart cards, credit cards and ATM cards have all been available to consumers for years and have been competing as means of payment. As a result, consumers often need to carry many plastic cards in their bulky wallets and have to manage several, incompatible, payment systems. Additionally, after purchases, consumers are given paper receipts and are required to keep them for merchandise purchase proof in case of return/exchange. Unfortunately, paper receipts can be lost causing inconvenience and customer dissatisfaction. Consumers are not the only ones who suffer. Vendors as well are experiencing significant charge-back loss (over $500 million annually [46]) because they are unable to locate the paper record receipt which must be produced as proof of purchase. Therefore there is a great desire to create a digital payment system which can integrate different payment systems, as well as reducing the costs and the risks of financial transactions.

Today, mobile phones are so pervasive that most of us carry them everywhere. They are multimedia appliances for voice communication, text and more recently multimedia messaging. Mobile phones are often equipped with other technologies (e.g. IrDA and Bluetooth) that can facilitate the exploitation of new data services. In an effort to make use of IrDA devices in digital payment applications, IrDA published the IrFM Point and Pay Profile specification in 2003 after three years development. As discussed in previous chapters, IrDA is well-suited for the "point and shoot" applications, in which two devices exchange objects of various types. IrFM profile takes this idea and extends it to a "point and pay" profile, in which customers can perform transactions at Point of Sale (POS) terminals using a short-range IrDA connection.

The basic notion of IrFM is straightforward: replace the physical financial instruments such as credit and debit cards with digital content in an eWallet. An eWallet can be any handheld device that people commonly carry, such as mobile phones or personal digital assistances (PDA). At the POS, a user would simply "point and pay" by aiming his/her mobile phone at the POS terminal rather than swiping a plastic card through a magnetic reader. Not only the user does not have to worry about the notorious problem of magnetic stripes getting worn out and difficult to read, but after beaming pertinent financial information to the POS terminal, the user now receivers a digital receipt that can be synchronized with financial applications on his/her mobile.

In addition, a number of interesting applications can be easily implemented on the IrFM platform, such as electronically archiving in-store coupons or other incentives for the user and integrating applications that would enable the user to take advantage of such incentives. For example, a customer uses a digital shopping list stored in his hand-held device, and the device alerts him/her of the fact that there is a special offer on a certain product and that he/she has

coupons for that product. At the check-out counter, the user would just "point and pay". The coupons would automatically be credited to the purchase and debited from his coupon supply. The potential of IrFM application is extensive, and IrDA is an optimum platform for digital payment systems because of its simple protocol stacks, short range, and directional medium.

As shown in Figure 6.1, IrFM provides a quick and seamless way for consumers to use the IrDA enabled Personal Trust Device (PTD) to pay for services and merchandises. It is done by beaming their "soft" credit cards, debit cards, e-cash, or other financial instruments to POS's, ATM's, vending machines or other compatible payment terminals.

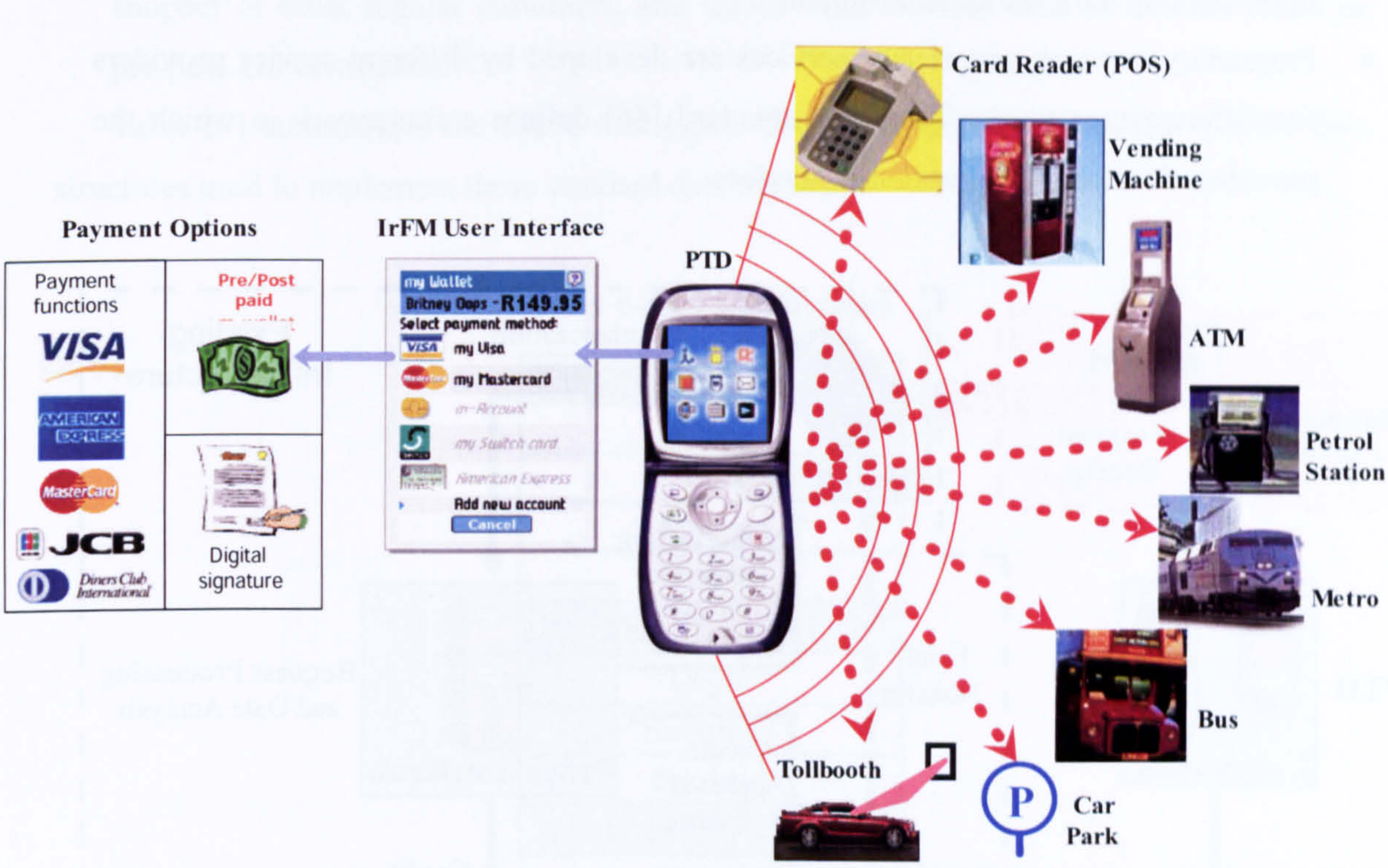


Figure 6.1 Mobiles equipped with IrDA ports would be the perfect platforms for digital payment system

Despite of its straight-forward concept, a digital payment system will not be successful unless it addresses the consumers' needs and provides clear benefits. Consumers will continue to use the proven traditional methods until an easier, more secure and faster solution to their financial management becomes available. In addition to meeting the consumers' needs, the new system should not pose additional burdens to the financial infrastructure.

The IrFM Point & Pay Profile contains detailed consumer usage models, terminal and mobile client implementation guidelines, architectural definitions [46] for sending and receiving payment and transaction record information between PTD and a financial terminal such as POS. Figure 6.2 illustrates a typical financial transaction flow using IrFM. The

manual card-swiping/card-reading interaction between the physical card and the POS terminal is replaced by the PTD and an IR-enabled POS terminal. After the transaction has been "beamed", the back-end processing of the transaction is treated as if the process had been carried out by card-swiping/card-reading interaction. The IrFM standard defines the core protocol commands, as well as the services used in different payment scenarios:

- Core IrFM protocol: it is the collection of the common commands needed for the IrFM applications including the connection establishment and termination.
- Core IrFM services: it supports the basic 'IrFM operations' used in all scenarios by implementing the Core IrFM Protocol.
- Proprietary services: proprietary services are developed by different service providers for different user needs. The IrFM standard [46] defines a framework in which the providers develop their proprietary services.

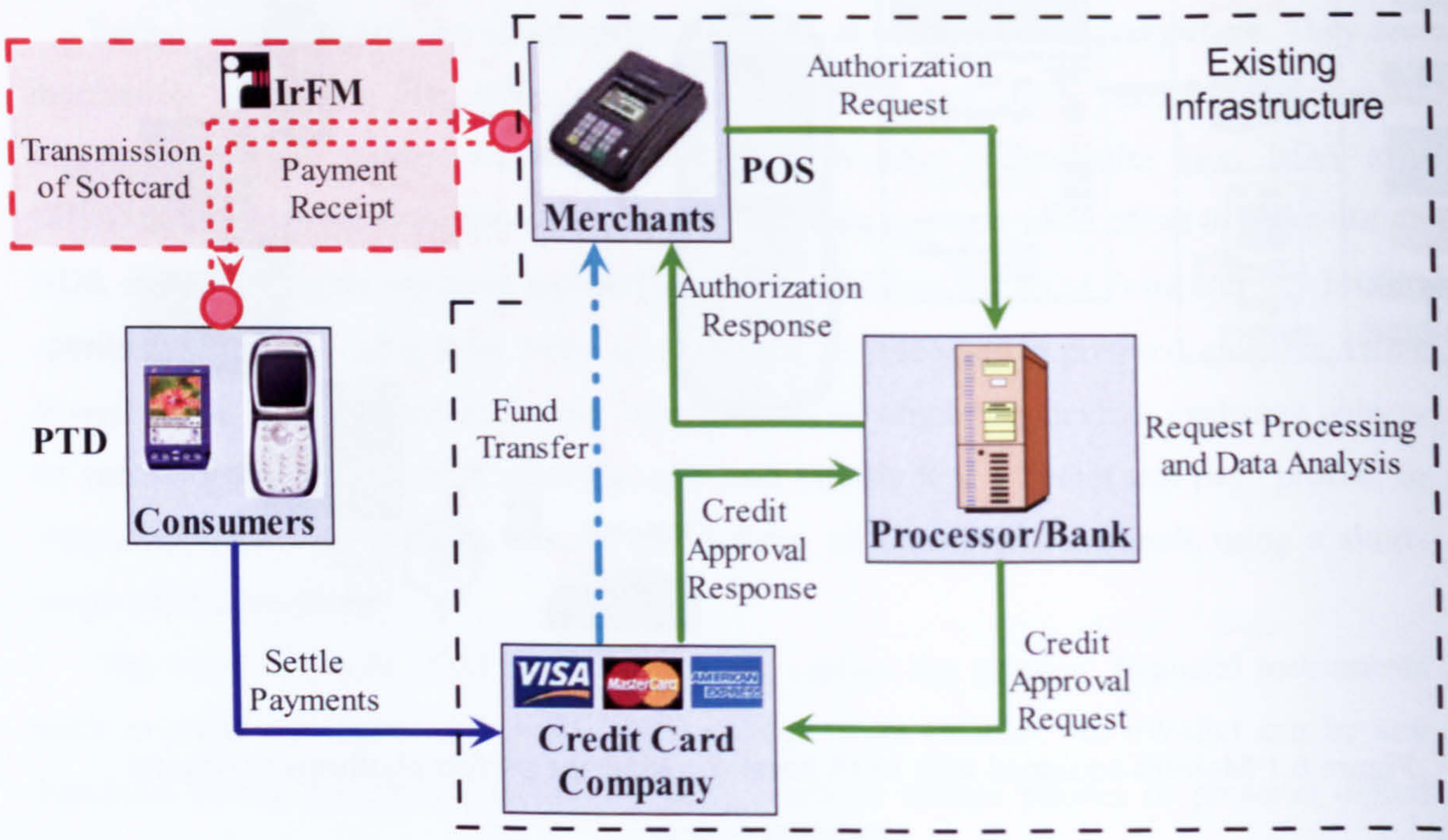


Figure 6.2 Transaction flows of the IrFM digital payment system

In order to increase the connection speed, IrFM implements the fast IrDA device discovery procedure [42]. Apart from this, IrFM does not have special requirements on the underlying IrDA protocol stacks. The IrFM protocol operates as an OBEX service, with the PTD functioning as the OBEX server and the POS as the OBEX client. IrFM transactions consist of Put and Get OBEX operations between the POS and PTD. Three categories of data exchange are defined by the IrFM profile as the core IrFM services: 1) Financial instruments; 2) Receipts; and 3) Vouchers.

- Financial instruments are mechanisms for paying for purchases. These include credit and debit cards, checks and cash.
- Receipts are records of transactions. These include digital receipts that are valid for legal purpose, receipts that indicate summary information such as a grand total, and itemized receipts that contain detailed information about a transaction.
- Vouchers are vendor specific mechanism to reward shoppers and create incentives for purchases. These include coupons that grant a discount to a shopper for a specific purchase, tickets that grant entrance or admission, loyalty programs to reward frequent shopper or other regular customers, and stored value devices such as in-store credit or pre-paid gift certificates.

Table 6-1 summarizes the above data types. The IrFM profile defines the specific data structures used to implement these standard data objects.

Table 6-1 IrFM data types

Financial Instrument
Credit
Debit
Check
Cash
Receipt
Legal
Summary
Itemized
Voucher
Coupon
Ticket
Loyalty
Stored value (prepaid)

6.2.2 IrSC

Although a majority of portable digital devices are equipped with IrDA ports, there are only a limited number of applications available for IrDA communications [134]. One potential IrDA application is fast digital image transmission from mobiles and cameras to TV's or PC monitors for display. Because the "point and shoot" nature of the IrDA usage model requires line-of-sight link alignment, short data transfer time is essential. Currently, IrDA endeavours to develop the Infrared Simple Connection (IrSC) protocol that allows instant transfer of digital data such as images from IrDA enabled handheld devices to

projectors, TV's, Set-Top Boxes and PC monitors, as shown in Figure 6.3. IrSC is optimized for ad hoc wireless networks and a simplified protocol targeting the instant transfer of data from handheld devices to fixed location devices. In comparison to IrBurst, as described in Chapter 5, IrSC deals with the transmission of relatively smaller size objects at even shorter transmission times.

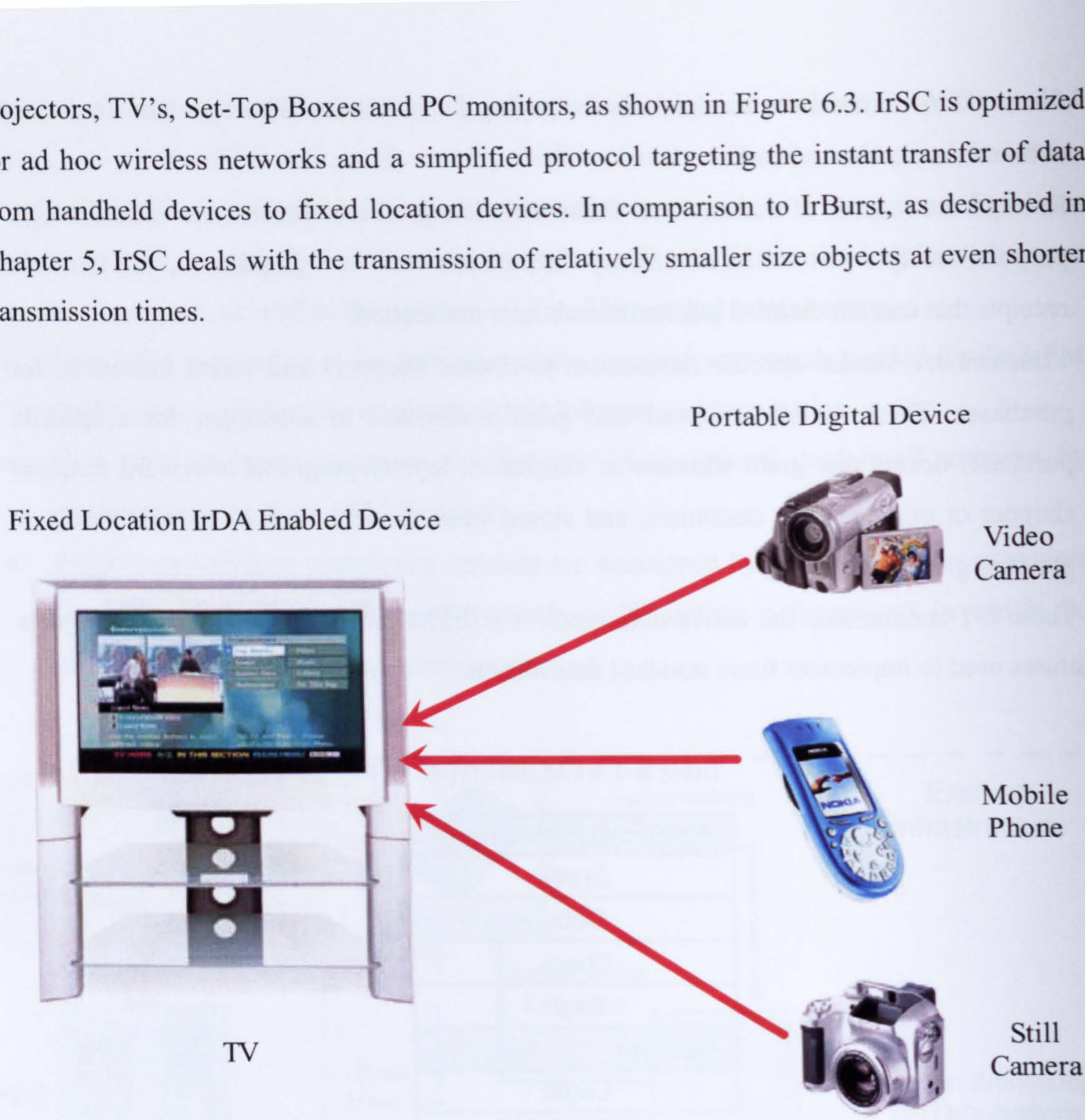


Figure 6.3 Typical IrSC applications

Several IrDA industry members are involved in the development of IrSC and have proposed a number of different technical approaches to IrSC. Among all the proposals, two technical approaches called IrSC1 and IrSC2 are the most promising and are to be voted by the IrDA special interest group for adoption to the IrSC standard. In order to reduce the IrDA connection time, both IrSC1 and IrSC2 implement the fast IrDA device discovery procedure [42]. In addition, IrSC1 and IrSC2 remove high level error corrections (OBEX acknowledgement) and flow control (TinyTP acknowledgement) to reduce the transmission time.

IrSC2 introduces a radical change to the current IrDA standard. It uses the IrLAP Unnumbered frame for the transmission and removes the IrLAP error correction scheme. IrSC2 thus further reduces the data transmission time at the cost of becoming susceptible to errors. It therefore requires manual retransmission if any errors occur (i.e. 'press the button again').

IrSC1 maintains the IrLAP error correction scheme and requires IrLAP level acknowledgement. The sender has to wait for the IrLAP turnarounds due to the half-duplex nature of IrDA links. As a result, the data transmission time of IrSC1 should be longer compared to that of IrSC2. However, the transmission is error free by implementing the error correction scheme. The properties of the IrDA standard connection (IrDAS), IrSC1 and IrSC2 are compared in Table 6-2.

	IrDAS	IrSC1	IrSC2
IrDA device discovery	Normal procedure	Fast procedure	Fast procedure
IrLAP error correction	Yes	Yes	No
TinyTP flow control	Yes	No	No
OBEX acknowledgement	Yes	No	No
Manual retransmission require?	No	No	Yes

Table 6-2 Comparison of IrDAS, IrSC1 and IrSC2

6.3 IrDA device discovery and connection procedures

As mentioned in the previous section, both IrFM and IrSC implement the IrLAP fast connection procedure. The details of the IrDA device discovery and the consequent connection procedures are discussed in this section.

6.3.1 Standard IrDA Discovery procedure

For the IrDA connection procedures, the device discovery procedure is carried out first in order to locate all the nearby IrDA stations. The device discovery procedure takes the following steps to accomplish device discovery: one station (initiator) listens to the medium for a time period greater than 500ms. If no infrared traffic is detected, the initiator broadcasts command Exchange Station Identification (XID) frames initiating a discovery procedure using *n* time slots. All nodes that receive the command XID frame become responders and each respond with a response XID frame in a random slot between 1 and *n*. The command and response XID frames exchange the IrDA device information including the IrLAP version number, the source and destination device addresses [42].

In practice, 8 continuous discovery slots are implemented to reduce the chance of response collision. 8 XID frames are sent in each discovery slot. The time for one discovery slot (t_{sl}) should not exceed 85 ms and must always be at least 25 ms. An example of four IrDA stations carrying out standard device discovery procedure is presented in Figure 6.4.

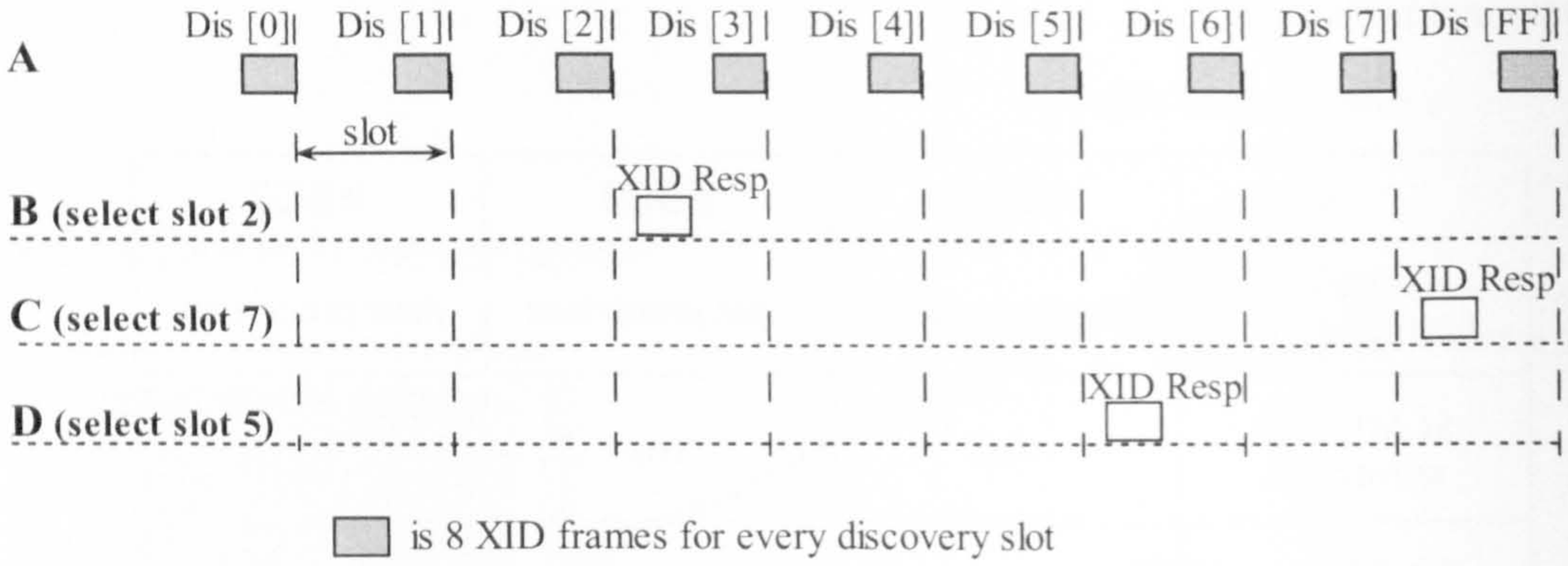


Figure 6.4 Example of the standard IrDA device discovery procedure. Stations A, B, C and D are all in Normal Disconnect Mode (NDM), which means the medium is not in use. Node A initiates discovery to locate all the nearby IrDA stations. Discovered station respond in randomly selected slots.

6.3.2 IrDA Fast Discovery procedure

If the initiator occupies a fixed location, the fast IrDA device discovery can be implemented to speed up the connection procedures by assuming only one other IrDA station in the transmission range. Since one of the IrFM or IrSC peers is considered to be a fixed device (i.e. POS for IrFM applications, TV for IrSC applications), the fast IrDA device discovery procedure can be used for IrFM and IrSC. There are three differences between the standard and the fast IrDA device discovery procedure:

- The listening period before sending out XID frames is not bounded by the 500 ms media access constraint. In the analysis, a listening period of 100ms is considered.
- One XID discovery slot is implemented since only one IrDA station is in the connection range.
- The time for one discovery slot (t_{sl}) is set to the minimum: 25 ms.

6.3.3 Connection procedures following the device discovery

The detailed connection procedures are presented in Figure 6.5. After a successful device discovery, the initiator (assigned as the primary station) selects one station from all the stations found by the discovery procedure. It then initiates the IrLAP connection by sending a Set Normal Response Mode (SNRM) frame. The other station (assigned as the secondary

station) responds to the SNRM by either an Unnumbered Acknowledgement (UA) to accept the parameters, or a Frame Reject (FRMR) to reject and renegotiate the parameters. Seven connection parameters [42] are negotiated in this process. The parameters include baud rate, maximum turn around time, IrLAP frame size, IrLAP window size, additional BOF's, minimum turn around time and link disconnection/threshold time. Both the SNRM and UA frames are 22 Bytes long. The IrLAP connection is now established. As the primary station has no knowledge of the data rates supported by the secondary, the IrLAP connection frames are sent at the default rate of 9.6 Kbit/s.

After exchanging link parameters during the IrLAP connection, the following commands are sent at the fastest data rate supported by both stations. All the higher layer packets are sent in the IrLAP frame and thus need to include the IrLAP header. The primary (i.e. PTD, TV) queries the Information Access Service (IAS) of the secondary (i.e. POS, digital camera) once the IrLAP connection is established. The query exchanges the IrLMP Link Access Service Points (LASP) of the queried OBEX server (i.e. eWallet, picture album server). To access an OBEX server, the IAS request and response packets are 37 Bytes and 15 Bytes long respectively [43].

The stations continue with the connection procedures by setting up the IrLMP/TinyTP connection. TinyTP connection parameters have the maximum length of 60 Bytes and are encapsulated in the IrLMP connection packets. The information of maximum TinyTP PDU size, maximum SDU size, TinyTP Service Access Point (TTP SAP) and QoS parameters [44] is exchanged during the TinyTP connection. The IrLMP connection request/response packet requires 5 Bytes to exchange the LASP address of the stations [43]. This represents 65 Bytes in total for each IrLMP request/response packet.

The final step for the connection procedures is to create the OBEX session. The analysis assumes that the OBEX session request/response packets have the maximum allowed length of 260 Bytes [45]. Depending on the application, the OBEX session request/response may exchange various information of the object to be sent, as discussed in section 5.1.

The total connection time including both the discovery and connection procedures is shown in Figure 6.6. The total connection time T_{con} is given as:

$$T_{con} = T_{interval} + T_{discov} + t_{LAP} + t_{IAS} + t_{LMP} + t_{OBEX} \qquad (6.1)$$

where $T_{interval}$ is the listening period before carrying out discovery procedure and T_{discov} is the IrDA device discovery time. $T_{interval}$=500ms and $T_{discov}=8t_{st}$ for the standard device discovery, while $T_{interval}$=100ms and $T_{discov}=t_{st}$(min)=25ms for the fast device discovery. t_{LAP}, t_{IAS}, t_{LMP}, t_{OBEX} are the time for the IrLAP, IAS, IrLMP&TinyTP and OBEX connections respectively.

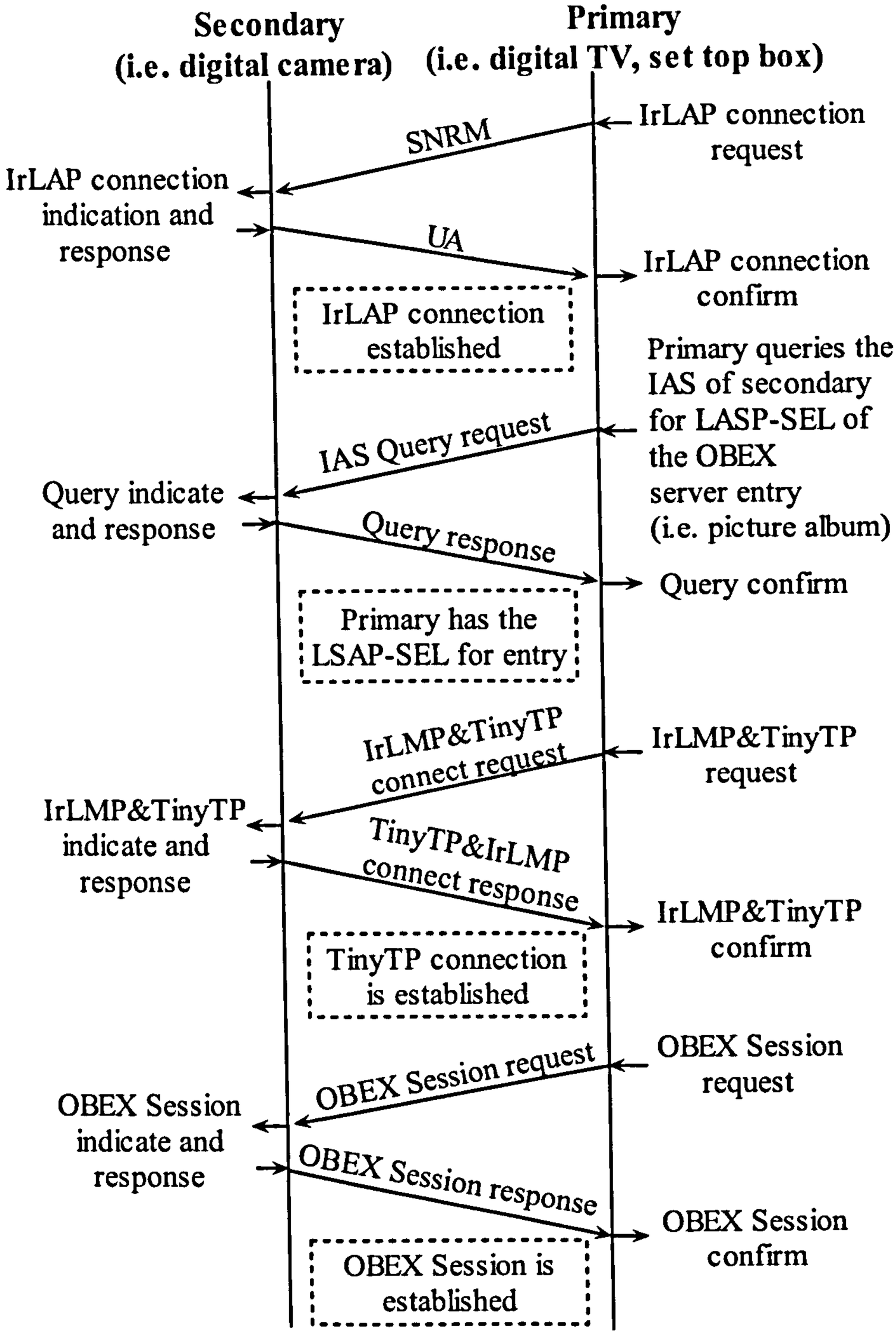
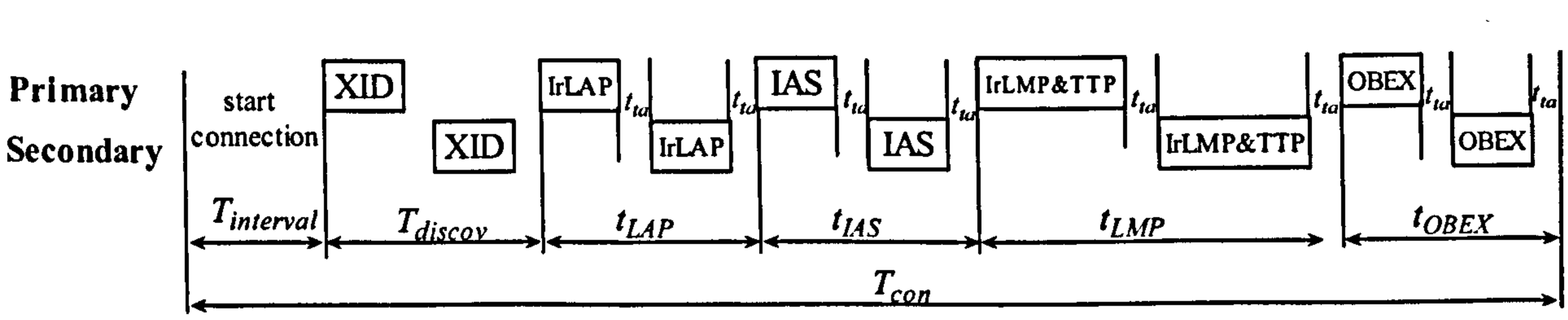


Figure 6.5 Detailed connection procedure for all the IrDA layers



t_{ta}: IrLAP turnaround time

Figure 6.6 The total connection time. Only one XID slot is shown in the figure

6.4 IrFM—the ePayment system

IrFM was designed with a number of objectives in mind, namely short transaction delay, high level of security, reliability, simplicity and interoperability. The following sessions provide a detailed study of IrFM.

6.4.1 Transition delay

One of the essential criteria for the success of IrFM is the short transaction delay. Consumers will not accept the digital system if they need to wait for a long time to complete their transactions. Various studies have suggested that the acceptable delay should be no longer than 1 second [95] from the time the user points his/her IrDA device to the POS until the purchase price is displayed on the screen of the device. After the price is shown, the user will be asked to choose the payment method and authenticate the payment. The system is mainly driven by the user thereafter.

The IrFM transaction delay includes connection time and data transfer time. Numerous studies have been carried out in the aim of improving the IrDA throughput [70]-[79]. However, nearly all the studies focused on reducing the data transfer time by considering an infinite size packet for transmission. In contrast, IrFM transactions only need to exchange relatively small size packets. The fast connection establishment is therefore likely to play a more important role than fast data transfer for IrFM applications. In this section, the total IrFM connection time, which is defined as the time delay from pointing the PTD to POS to displaying the merchandise price on the PTD, is investigated.

To describe IrFM in detail, we are going to study an IrFM scenario: "paying for groceries". After the groceries are being scanned, the customer approaches the counter, powers up the IrDA enabled PTD, and points it at the infrared sensor on the counter. The eWallet application automatically launches with a reliable IrFM connection to the POS. The groceries' price is then displayed on the PTD. The customer chooses a payment method, authenticates himself (i.e. enters password) and authorizes the payment. After authenticating the payment with the bank, the POS sends the itemized bill back to the PTD.

According to the IrFM standard, fast IrLAP connect procedure is used for the IrFM applications. As discussed in section 6.2, the connection time T_{con}, which is the time from starting the medium listening period to finishing the OBEX session establishment, is given by equation (6.1).

In order to bring the grocery purchase onto the screen of the PTD, IrFM has to perform at least two IrFM operations after the OBEX session establishment: 'exchange IrFM information' and 'put price'. These two operations need to set up and terminate OBEX

connection twice, as well as transmitting several OBEX operations. In Figure 6.7, the detailed packet exchange between the POS and the PTD for the IrFM application is illustrated.

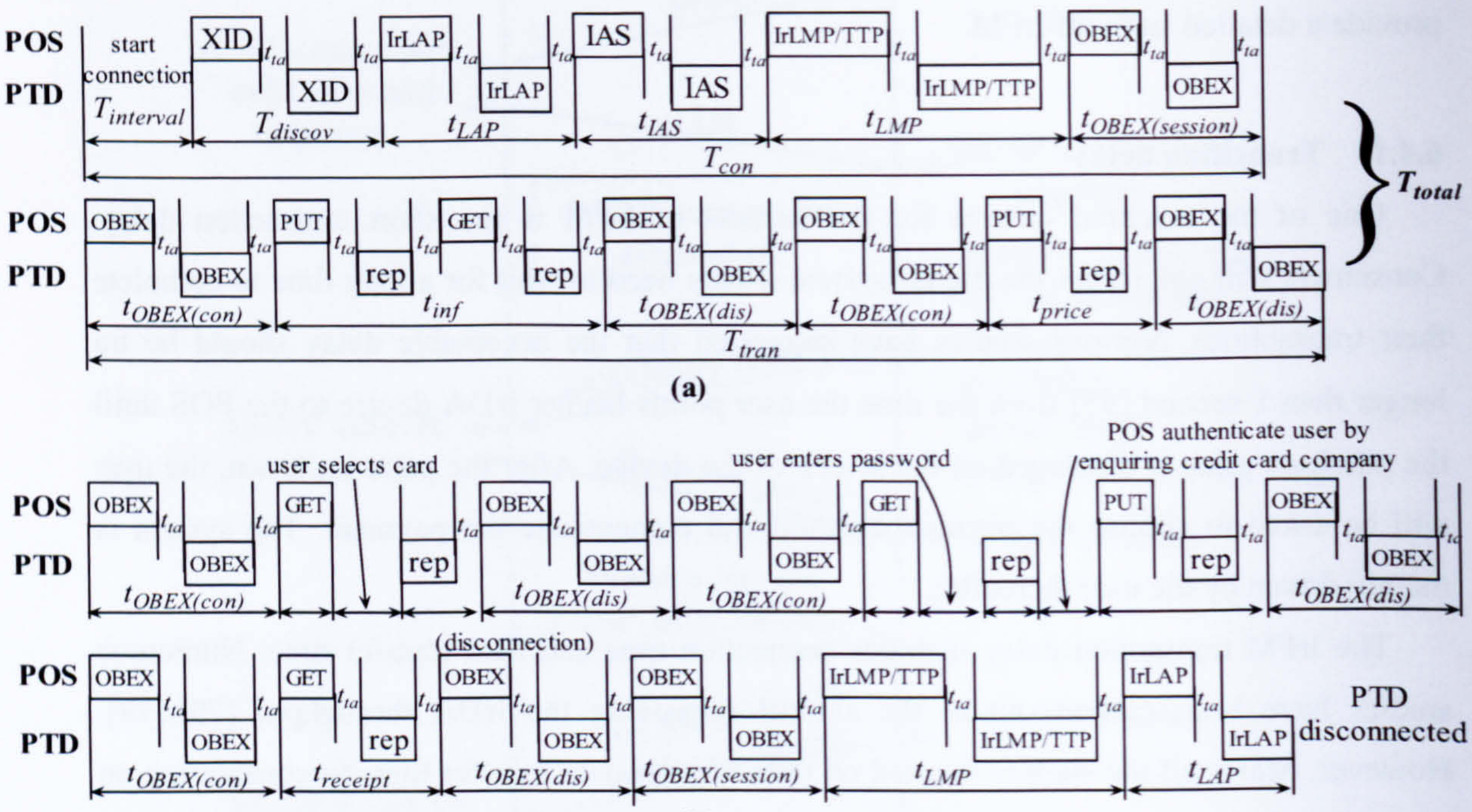


Figure 6.7 Example of an IrFM application "paying for groceries": (a) From the start of the IrFM connection to displaying the merchandise price on the PTD; (b) From the payment method selection, payment authentication and receipt delivery to the connection termination.

From Figure 6.7, the transmission time for the two necessary IrFM services before displaying the price on the PTD is given by:

$$T_{tran} = 2t_{OBEX(con)} + t_{inf} + 2t_{OBEX(dis)} + t_{price} \tag{6.2}$$

where $t_{OBEX(con)}$ and $t_{OBEX(dis)}$ are the time for the OBEX connection and disconnection procedures, while t_{inf} and t_{price} are the time for exchanging IrFM and the merchandise price information.

The total connection time (T_{total}) to display the groceries' price on the PTD consists of the connection time T_{con} and the transmission time for the two IrFM services T_{tran}:

$$T_{total} = T_{con} + T_{tran} \tag{6.3}$$

The calculated total connection time for different data rate (C) and IrDA turnaround time (t_{ta}) are shown in Table 6-3. The table shows that T_{total} reduces significantly with the

increasing data rate and also benefits from a smaller value of t_{ta}. T_{total} becomes less than 1s when $C \geq 38.4\text{Kbit/s}$ and $t_{ta} \leq 1ms$.

C (bit/s) \ t_{ta}	10ms	1ms	0
9600	2.40 s	2.19 s	2.17 s
19200	1.53 s	1.32 s	1.29 s
38400	1.10 s	0.89 s	0.87 s
57600	0.95 s	0.75 s	0.72 s
115200	0.81 s	0.60 s	0.58 s

Table 6-3 Total connection time for various data rates and minimum turnaround times

The time allocation in percentage of the components $T_{interval}$, $t_{ta}(total)$, T_{con} excluding t_{ta} and T_{tran} excluding t_{ta} for data rate from 1Kbit/s to 1Mbit/s is shown in Figure 6.8. Since all the time portions for the IrFM connection are considered, the sum of the four percentages always equals to 100%. T_{con} and T_{tran} are the two dominant factors at low data rates. When combined, they occupy more than 90% of the overall time at C=1Kbit/s. $T_{interval}$ and $t_{ta}(total)$ gradually increase to over 90% of T_{total} at C=1Mbit/s.

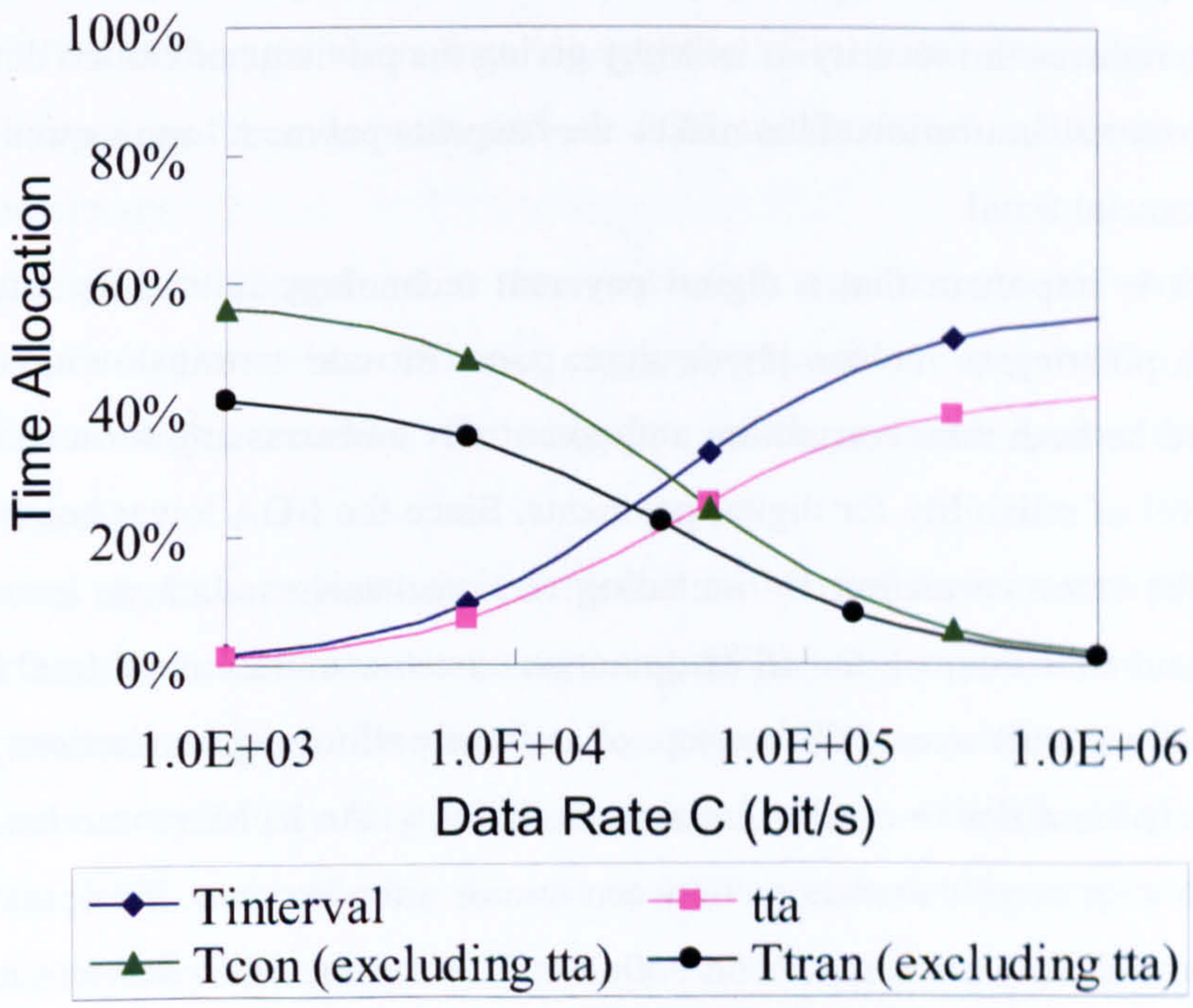


Figure 6.8 Percentages of time allocation for the four different components of IrFM total connection time

As shown in Table 6-3 and Figure 6.8, implementing high data rates can reduce the connection time but only leads to insignificant improvement when C>100Kbit/s. The fastest speed of 115.2 Kbit/s specified by Serial Infrared (SIR and t_{ta} of 1ms are adequate for the IrFM applications. To further reduce the connection time, a smaller value of $T_{interval}$ should be applied.

6.4.2 Security and Reliability

Security is one of the prime concerns in any financial exchange instrument. IrFM has a very high level of security compared to other wireless technologies (e.g. Bluetooth, Wi-Fi). The condition of a clear 'Line of sight' path has been long considered as the major disadvantage of infrared communication. However, short distance 'Line of sight' becomes a prominent advantage for the high security conscious financial transaction. Since infrared IrDA communication is directed by user's 'point and shoot' behavior and is a strictly 'peer to peer' communication, it is difficult to intercept undetected which makes it an ideal platform to exchange confidential information. Digital signatures, password or PIN authentication, and session encryption for the e-payments can be implemented over IrFM transactions to further increase the security.

The biggest security concern for IrFM is the 'express payment' application. IrFM enables 'express payment' [46] in order to reduce the payment time in small transactions (i.e. vending machines, tollbooth). In these cases, the consumer does not need to enter a password to authenticate the payment. The 'express payment' reduces the transaction time significantly. However, it also reduces the security of IrFM by giving the privilege of certain devices to bill consumers without authentication. This makes the 'express payment' application potentially vulnerable to financial fraud.

Obviously it is important that a digital payment technology is reliable. Standing at a checkout stand, pointing a mobile phone at a point of sale terminal without anything happening would be both time consuming and potentially embarrassing for a shopper. IrFM offers a high level of reliability for digital payments. Since the IrDA lower layer has already taken care of the error correction by including the mechanisms such as error detection, retransmission and flow control, the IrFM operations are transmitted error free. In addition, IrFM applications are built around the concept of services performing transactions where each service is uniquely identified to ensure transaction reliability. An IrFM transaction is achieved by performing one or more transactions between one or more services. To operate a service properly, the same service must appear on both the PTD and the POS. Services are uniquely identified by a 128bits Universally Unique Identifier (UUID). A brief disconnection during the IrFM transaction is possible due to natural user interactions, as the user may want to remove the device from infrared range to read the display or input a PIN. Such a

disconnection should not cause a transaction to fail, starting the process all over again. By utilizing the service UUID, IrFM accommodates potentially several disconnects during the financial transaction either at logical break points where the user is prompted for information or during the financial transaction itself. If the disconnect exceeds a preset period of time, the transaction process will start over again from the beginning.

6.4.3 Simplicity and Interoperability

The goal of IrFM is to replace traditional financial transaction instruments with electronic exchange over the point-to-point infrared link. If doing so introduces greater complexity to the user than what is currently experienced with traditional instruments such as checks and credit cards, the venture is destined to fail. In order to attract more users (both consumers and merchants) in accepting the concept of digital payment and for fast system adoption, the payment system needs to be of low cost, utilize the existing infrastructure, as well as being simple to use. Since IrFM requires no additional hardware support and is simple to use, it meets the above requirements. To use IrFM, a consumer only needs to have an IrDA enabled PTD, install the IrFM software suit in the PTD and add the credit card. For merchants, it is also straight-forward to set up IrFM. They are required to install an IrDA dongle and the IrFM software on the traditional POS. The financial transaction request is sent to the payment point where it is used to authenticate the physical credit card payments. As depicted in Figure 6.2, no additional network infrastructure is needed for the IrFM system.

The goal of interoperability is essential to the success of IrFM. Interoperability exits when devices from different manufacturers function together. Given that consumers use handheld devices of all types and brands with various POS terminals, all of these systems must interoperate seamlessly.

Interoperability is achieved when two essential requirements are met: First, the communication protocol must be consistent, employing the same layers of the same protocol stacks in the same way; secondly, the endpoints of the communication must exchange standard data objects.

IrDA protocol stacks have been implemented in millions of devices over years. Time has proven that IrDA devices have outstanding compatibility among different manufacturers. Thus, IrDA protocol stacks fulfils the first interoperability requirement. Being the only infrared standard for digital payment, IrFM is expected to have a high level of compatibility. According to the IrFM standard [46], the use of IrDA standard object types will guarantee that objects are correctly understood by the other device. Furthermore, IrFM alerts the user and cancels the transaction if the other device failed to understand the IrFM object being sent. Therefore, the second requirement is also supported by IrFM.

6.4.4 Current market status

Besides IrDA, other technologies like Radio Frequency Identification (RFID) [96], Bluetooth [104] and Cellular network [97] are all among the potential technologies for digital payment systems. In this section, IrFM is compared with other potential digital payment technologies. The basic comparison of these technologies is summarized in Table 6-4. The current status of IrFM is also investigated.

RFID is the technology that uses radio waves to automatically identify objects and is the major competitor for IrFM. For more than a decade, RFID has been used in manufacturing plants for tracking parts and work in process. Over last few years, RFID has also been adopted as a convenient payment mechanism. One of the most prominent applications today is the prepaid RFID cards used in public transportation systems and fast food restaurants. More recently, a few companies have proposed and are developing RFID embedded credit/debit cards (i.e. MasterCard's Paypass, ExxonMobil's Speedpass). Compared to IrFM, the biggest advantage of RFID payment systems is the familiarity to consumers and merchants since RFID credit/debit cards are essentially the plastic banking cards with RFID chipset installed. However, RFID cards currently do not have any integration capabilities. Thus, RFID cannot address the problem of having different cards for different banks, shops or services. Additionally, RFID cards can not store any digital receipt (paper receipt is still used) and do not support any value-added services (i.e. e-vouchers, loyalty programs).

Bluetooth is radio frequency based technology operating in the 2.4GHz licence free band. It has numerous security/privacy concerns as a payment technology [112]. Although there are many encryption techniques that can be applied to Bluetooth system, none of them can completely meet the security demands [113]. Furthermore, encryption techniques complicate the whole system, resulting in additional cost and performance drop. Another issue for Bluetooth being a digital payment system is the long connection delay. The conclusion is that Bluetooth is still not a mature enough technology for digital payment systems.

Several cellular mobile operators have proposed mobile payment systems. NTT Do-Co-Mo has successfully launched a mobile payment trial [97] in 2003. There are however three major drawbacks limiting a large scale adoption of such payment systems: (a) a significant load to the base station is added; (b) a long transaction time is required as three RF exchanges are required between the mobile phone and the base station for each transaction; (c) mobile providers have to add additional gateways to interpret and re-route the payment messages.

	IrFM	RFID	Bluetooth	Cellular Network
Connection delay	medium	short	long	long
Security	high	medium	low	low
Usage simplicity	medium	easy	medium	medium
Modification/Addition to infrastructure	medium	medium	medium	high
Investments from providers	low	medium	low	high
Integrability (more than 1 cards can be put in the device)	Yes	No	Yes	Yes
Digital receipt	Yes	No	Yes	Yes
Value-added services (e.g. e-vouchers)	Yes	No	Yes	Yes

Table 6-4 Comparison of different technologies for digital payment systems

IrFM effectively addresses all the issues of payment systems. Soon after the IrFM standard published, both hardware and software companies unveiled a number of products that support IrFM [98]. For instance, Agilent Technologies offers the smart IrDA dongle (HSDL-300) and the IrFM Software (HSDL-S500) for Mobile Phone and PDA; Sony Ericsson Mobile launched the "Mova SO506iC" mobile phone for NTT Do-Co-Mo, which is the third smart-card handset that is compatible with i-mode® FeliCa® service for IrFM applications. Sony Ericsson also unveiled a 3G CDMA 1x EV-DO phone for the KDDI corporation, the "W21S", which supports IrFM applications; EMBEDnet Inc. announced an IrDA protocol evaluation test harness (IrTESTER) for IrFM and IrDA stacks.

IrFM has been introduced to the financial market by several companies in a number of countries including U.S.A., Japan and South Korea [99]. Among all the banks and credit card companies, Visa is the leading adopter of IrFM. Visa partnered with NTT Do-Co-Mo in Japan, KDDI, SK Telecom and Harex InfoTech in South Korea [99]. The Visa IrFM payment system, named 'Proximity Payment', was launched by the end of 2003 and is now in service. South Korea has the biggest IrFM user group [99]: by the end of 2003, 500,000 IrFM-payment enabled mobile phones had been sold by LG TeleCom [100] and KTF [101]. Additionally, 2,000 ATM machines, 1,000 department stores, 1,500 vending machines, 100 parking lots, 80,000 POS, and Hwang Song Tunnel tollbooths have been equipped with IrFM payment readers.

6.5 Performance comparison of IrDAS, IrSC1 and IrSC2

In order to compare the performance of the IrDA standard connection (IrDAS), the IrSC technical approach 1 (IrSC1) and the IrSC technical approach 2 (IrSC2), the total transmission time for transmitting a digital picture is investigated in this section.

6.5.1 Mathematical modelling

The following scenario is considered in the analysis: display a picture from a digital camera on a TV. The user powers up the digital camera, and points it at the infrared port on the TV set top box. The 'easy-picture' application automatically launches an OBEX connection to the set top box. The user selects the picture which he/she wants to display on the TV screen and transfer it by pressing the send button on the camera.

As discussed in section 6.2, the connection time T_{con} is given in equation (6.1). Next, the transmission time for the digital picture is to be derived.

IrSC2 requires manual retransmission if any error occurs (i.e. picture does not show properly on the TV screen), as pointed out in section 6.1. Since the error retransmission of IrSC2 is carried out manually, the transmission time is various. In order to model the IrSC2 protocol behaviours, the analysis assumes that the picture file is transmitted without error. It is also assumed that the buffer is large enough to accommodate all the incoming TinyTP PDU's. The symbols used in the derivation are detailed in Table 3-1, Table 4-1 and Table 5-1.

IrDAS and IrSC1 use the same mechanism at the IrLAP layer (packets are transmitted by means of frames and windows). However, IrDAS also implements the high level OBEX error retransmission scheme which requires sending OBEX acknowledge for each OBEX packet. The data transmission time of IrDAS is given as:

$$T_{tra} = \frac{l_{data}}{l_{OBEX}}\left(\frac{l_{OBEX} + l'_{OBEX}}{l_{TTP}N_{LAP}}(N_{LAP}t_I + t_{ack} + T_{ack}) + T_{RES} + 2T_{OBEX}\right) \quad (6.4)$$

where l_{data} is the size of the picture file (i.e. a 500-KByte picture), l_{TTP} is the TinyTP PDU size, N_{LAP} is the IrLAP window size, t_I is the IrLAP information frame transmission time, t_{ack} is the IrLAP acknowledge time, T_{ack} is the TinyTP acknowledge time, T_{RES} is the OBEX acknowledgement transmission time, T_{OBEX} is the OBEX turnaround time. l_{OBEX} and l'_{OBEX} are the size of the OBEX packet and the OBEX header respectively.

Without sending any OBEX and TinyTP acknowledgements, the data transmission time of IrSC1 is:

$$T_{tra} = \frac{l_{data}}{l_{OBEX}}\left(\frac{l_{OBEX} + l'_{OBEX}}{l_{TTP}N_{LAP}}(N_{LAP}t_I + t_{ack})\right) \quad (6.5)$$

No error correction is implemented for IrSC2. Since Unnumbered frames are used to carry OBEX packets, IrLAP frames are sent continuously within the Maximum IrLAP turn around time (maximum value of 500ms as defined in [42]). In order to finish the OBEX packet transmission before the forced IrLAP turnaround, we recommend the IrSC2 to support the maximum turnaround time up to 5s. The data transmission time of IrSC2 is given by:

$$T_{tra} = \frac{l_{data}}{l_{OBEX}}\left(\frac{l_{OBEX} + l'_{OBEX}}{l_{TTP}} t_1 + 2t_{ta}\right) \tag{6.6}$$

The total time from the user initiating the transfer to display a picture on TV consists of the connection time and the data transmission time:

$$T_{total} = T_{con} + T_{tra} \tag{6.7}$$

6.5.2 Analysis of results

In Figure 6.9, the connection time and the total time to display a picture on the TV is plotted against the data rate in the range of 100 Kbit/s to 100 Mbit/s. Frame size of 16 Kbits, windows size of 7 and minimum turnaround time of 10ms are implemented. A typical digital picture size (l_{data}) of 500 KBytes and an OBEX packet size (l_{OBEX}) of 512 Kbits (maximum size defined in [45]) are considered. The discovery slot time t_{sl}=50ms and t_{sl}=25ms are used in the figure respectively for the standard device discovery and the fast device discovery.

As shown in Figure 6.9, more than 40 seconds is needed to send the picture at the data rate level of 10^5 bit/s for all of the three systems. It is unrealistic to ask a user to hold a digital camera for more than 40s. Thus, the data rates in the level of 10^5 bit/s (115.2 Kbit/s and 576 Kbit/s) are not adequate for this application and our study will focus on the data rates over 1 Mbit/s.

In Figure 6.10, the connection time and the total time is re-plotted against the data rate in the range of 1 Mbit/s to 100 Mbit/s by using the same parameters in Figure 6.9. Since no error correction scheme is employed for IrSC2, it always requires less time to transmit the file at the same data rate. By implementing the fast connection procedure, both IrSC approaches yield better performance than IrDAS. The connection time is not significantly affected by the data rate and the values are around 1.1s and 0.25s respectively for the standard IrDA connection and the IrDA fast connection. It should be noted that, IrSC2 is susceptible to errors although the effect of errors is not included in this study. The user is required to retransmit the file manually ('pressing the button again') if the picture transfer failed. IrSC1 is therefore suitable for the high BER environments. In order to have the T_{total} under 3s which is suitable for the IrSC applications, data rates higher than 4Mbit/s (FIR) should be implemented.

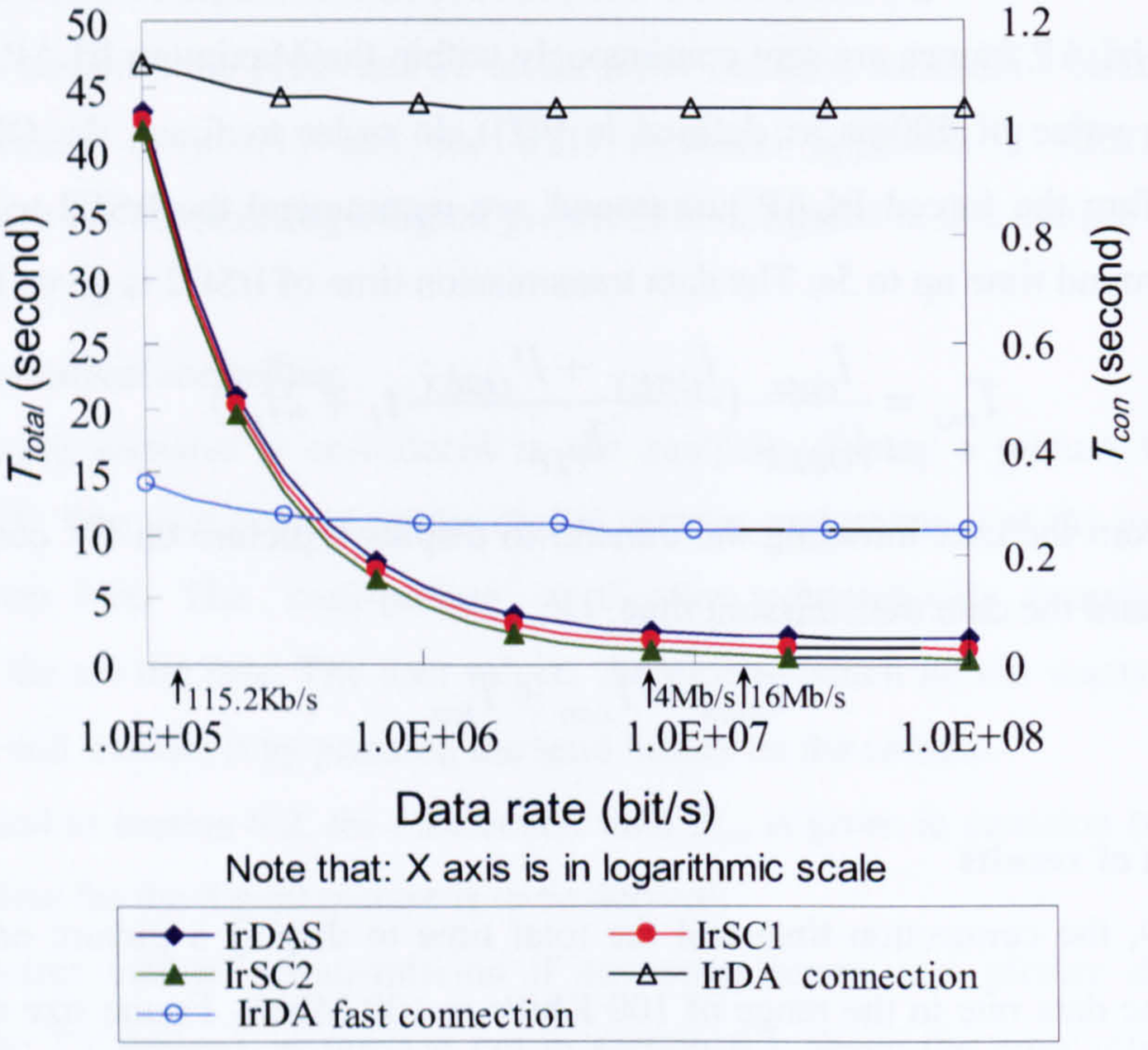


Figure 6.9 Total time to display a 500-KBytes picture on a TV against the data rate from 100Kbit/s to 100Mbit/s

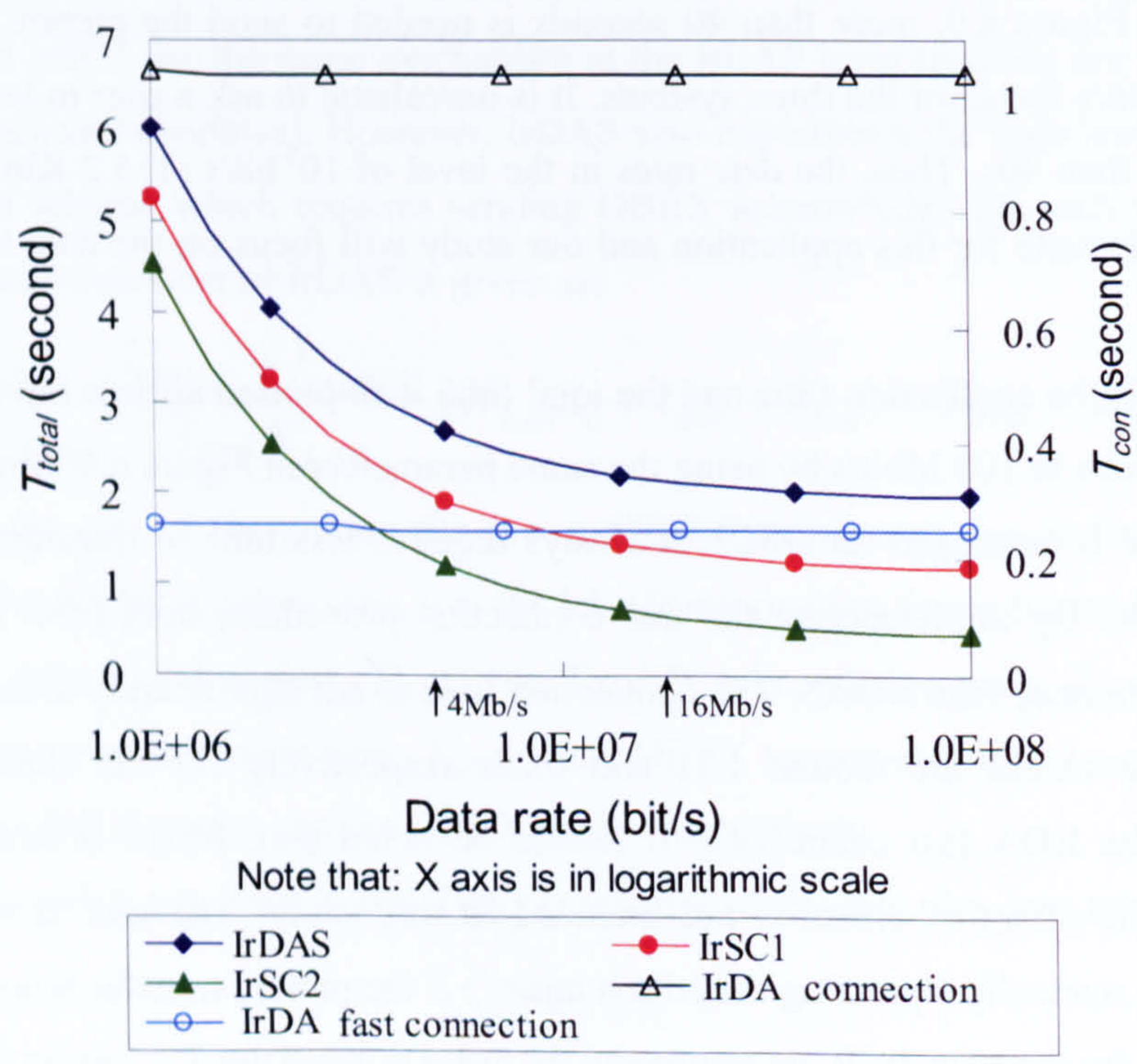


Figure 6.10 Total time to display a 500-KByte picture on a TV against the data rate from 1Mbit/s to 100Mbit/s

In practice, it is possible to have faster minimum turnaround times (t_{ta}) for faster links [57]. In Figure 6.11, the total time and the connection time are plotted against the data rate by implementing a t_{ta} decreasing with the data rate (C), which is more realistic than having a fixed and large minimum turnaround time. The value of t_{ta} is calculated as follows:

$$t_{ta} = \frac{10ms \times 10^6}{C}.$$

Figure 6.11 shows that at 4Mbit/s (FIR) the system can achieve about 1s total transfer time for both IrSC1 and IrSC2. Connection time is less than 0.2s when $C \geq 4$Mbit/s . IrDAS requires 2s of total transfer time, which is acceptable at C=4Mbit/s. For the IrLAP frame size of 16 Kbits and window size 7, there is no significant difference between IrSC1 and IrSC2, with IrSC2 being marginally faster.

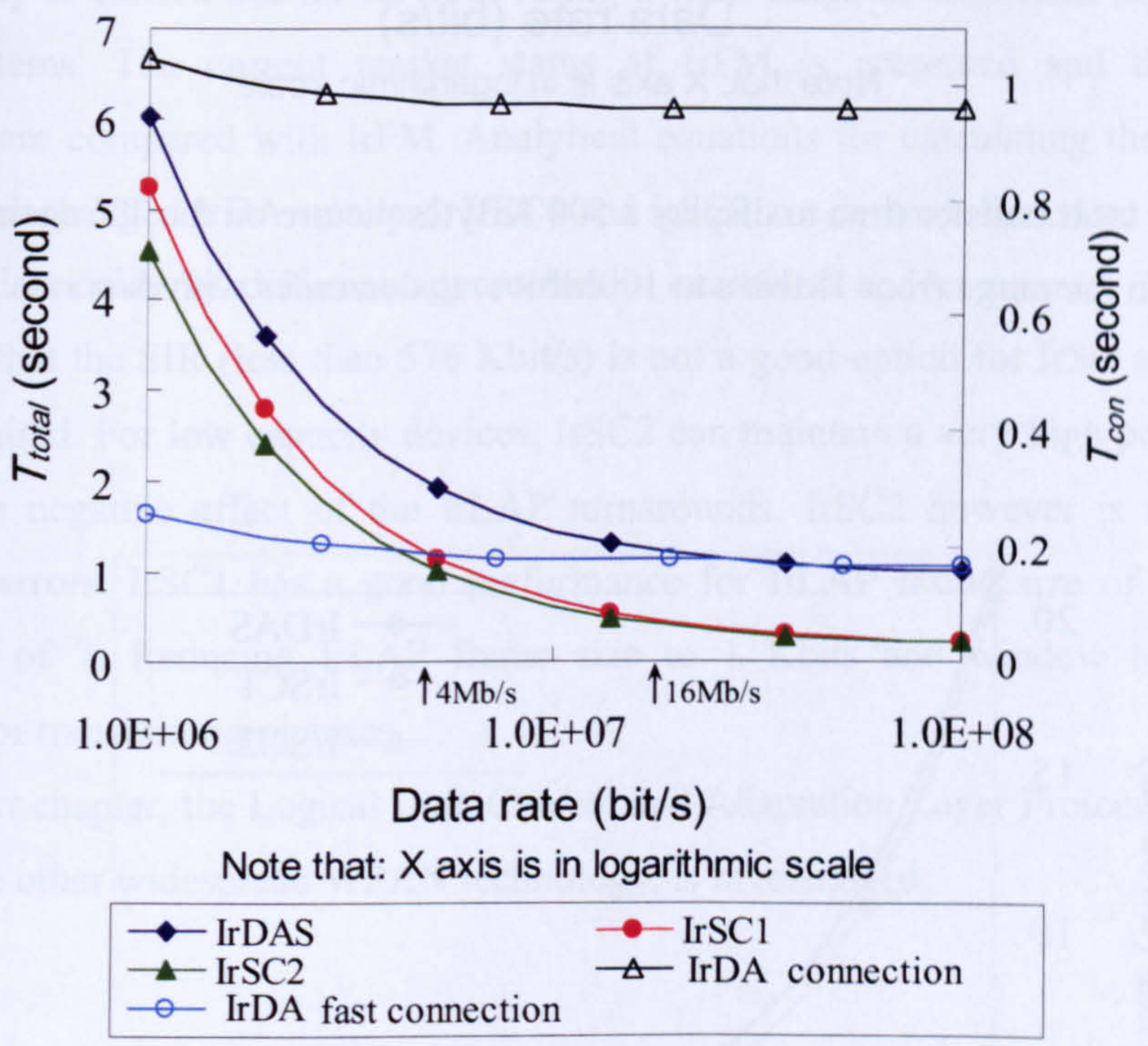


Figure 6.11 The total time to display a 500 KBytes picture on the TV against the data rate in the range from 1Mbit/s to 100Mbit/s. t_{ta} decreases with data rate.

In the above figures, the IrLAP frame of 16Kbits and window size of 7 are used requiring large buffer.

Figure 6.12 demonstrates the effect of reducing the IrLAP frame size from 16Kbit to 1Kbit, and reducing the window size from 7 to 1 (low cost approach). The connection time is not plotted since it has the same value as in Figure 6.11.

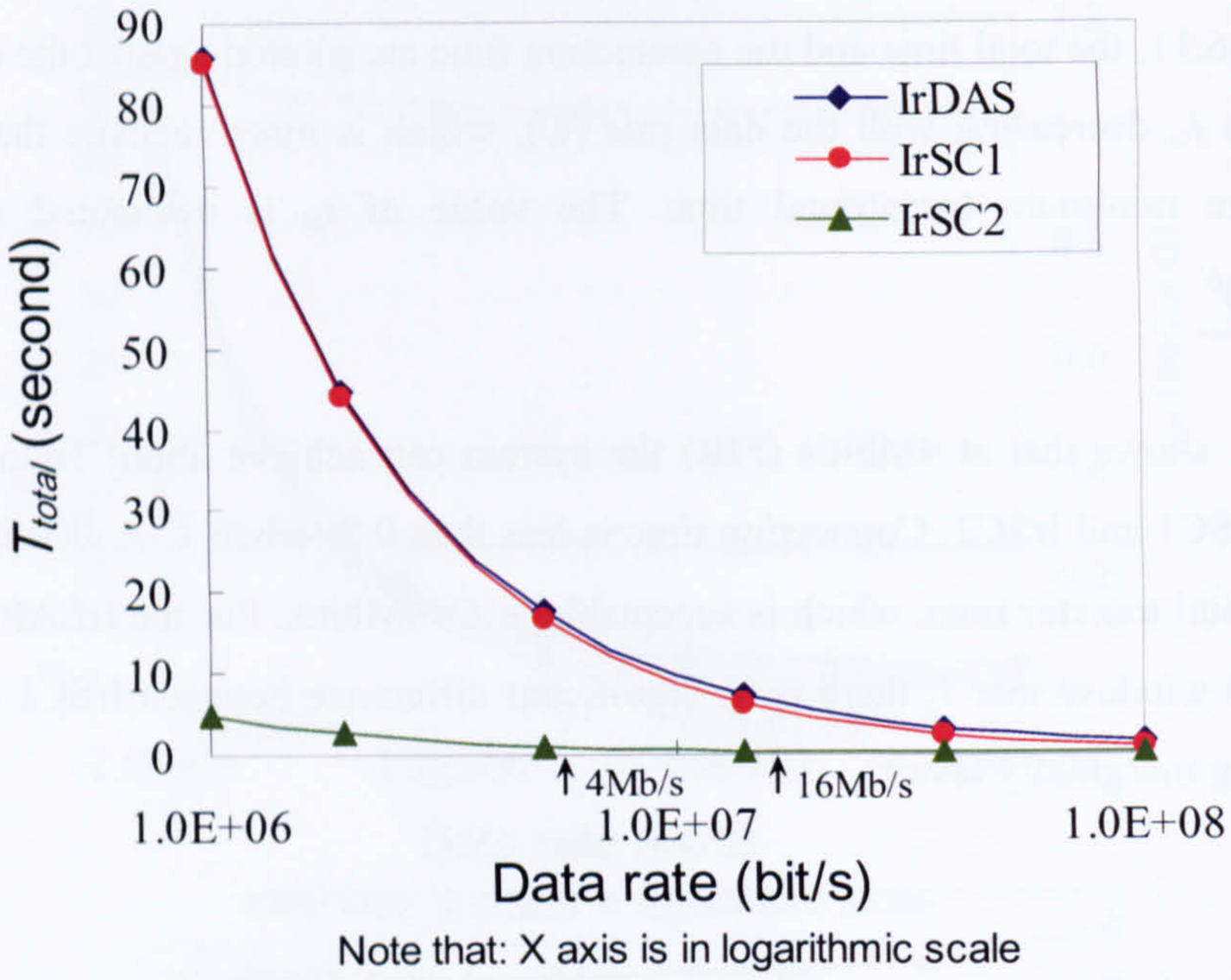


Figure 6.12 The total transfer time to display a 500 KBytes picture on the TV against the data rate in the range from 1Mbit/s to 100Mbit/s. t_{ta} decreases with data rate.

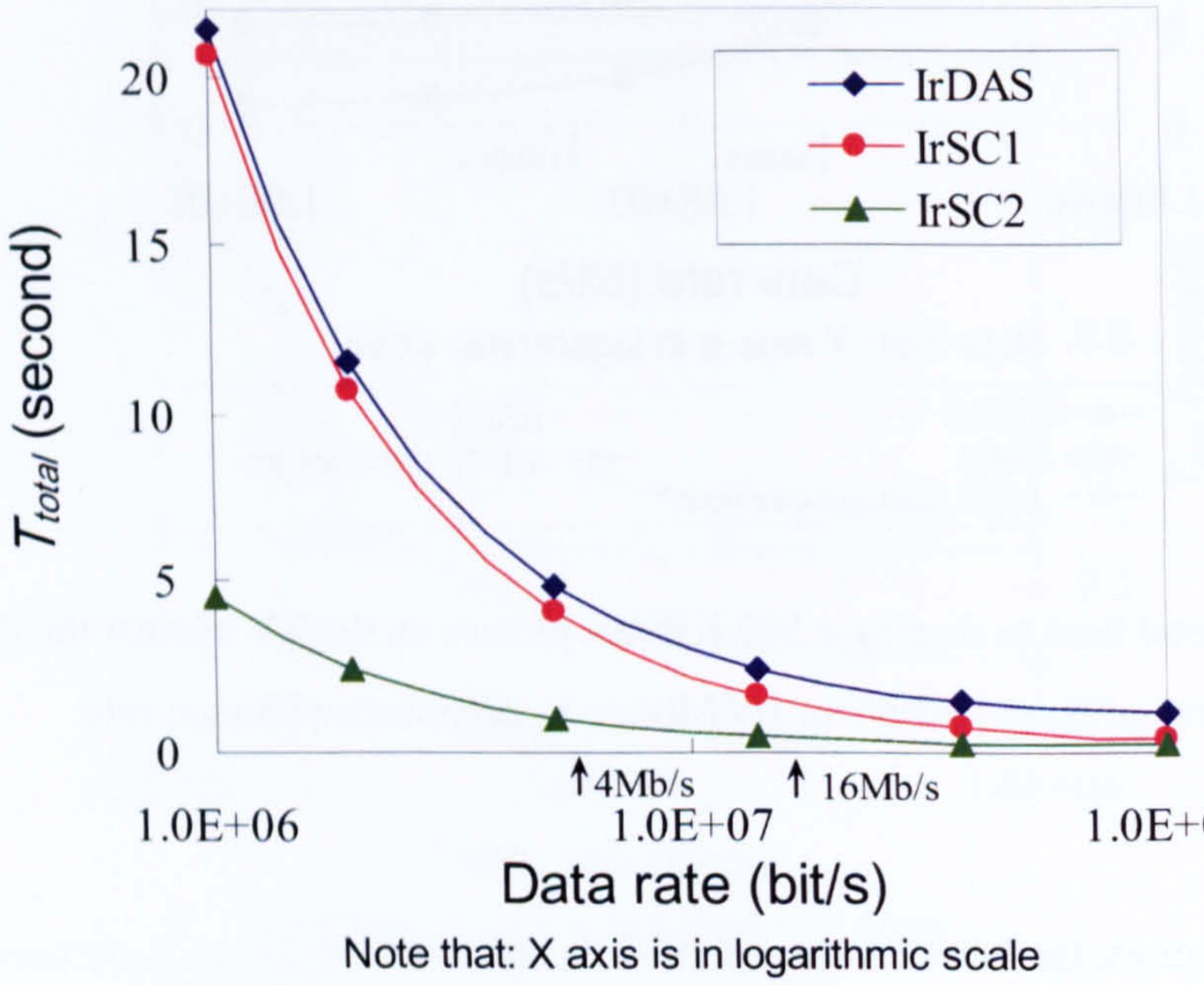


Figure 6.13 The total transfer time to display a 500 KBytes picture on the TV for the data rate in the range from 1Mbit/s to 100Mbit/s. t_{ta} decreases with data rate.

Figure 6.12 shows that even at 4Mbit/s IrSC1 requires the total transfer time of almost 20s! This is due to more frequent IrLAP turnarounds since small IrLAP frame and window sizes are chosen. IrSC2 is not strongly affected by the change of the IrLAP frame and window size. Only IrSC2 offers a realistic solution.

In Figure 6.13, the IrLAP frame size of 5 Kbits and the window size of 1 (medium cost approach) are used. The total transfer time is about 4s at the data rate of 4Mbit/s for IrSC1. IrSC2 offers the same performance despite the change of the IrLAP parameters.

6.6 Summary of the Chapter

In this chapter, two rapidly developing IrDA applications, IrFM and IrSC, are studied. An inclusive study is carried out for IrFM covering a wide range of important issues of digital payment systems. The current market status of IrFM is presented and the competing technologies are compared with IrFM. Analytical equations for calculating the transmission delay for the unchanged IrDA standard, IrSC1 and IrSC2 are derived. Based on the equations, the transmission delay of different approaches is examined and compared. The analytical results show that the SIR (less than 576 Kbit/s) is not a good option for IrSC and 4Mbit/s or higher is required. For low capacity devices, IrSC2 can maintain a very high performance by bypassing the negative effect of the IrLAP turnarounds. IrSC2 however is susceptible to transmission errors. IrSC1 has a good performance for IrLAP frame size of 16 Kbits and window size of 7. Reducing IrLAP frame size to 1 Kbits and window to 1 makes it unworkable for transmitting pictures.

In the next chapter, the Logical Link Control and Adaptation Layer Protocol (L2CAP) of Bluetooth, the other widespread WPAN technology, is investigated.

Chapter 7. Throughput and Delay Analysis of Bluetooth L2CAP Layer

7.1 Introduction

Currently, there are two dominant technologies used in WPAN: IrDA and Bluetooth. From Chapter 3 to Chapter 6, the performance of the IrDA system is examined and optimized at different communication layers including the data link, the transport, the session and the application layers. In order to provide a complete picture of the WPAN technologies, it is also important to examine the performance of Bluetooth.

After the successful introduction and development of voice applications (Bluetooth headsets), the Bluetooth data applications have been gradually accepted and used. The Bluetooth data applications use the Asynchronous Connection-Less (ACL) transport provided by the Bluetooth Baseband layer. To improve the Bluetooth ACL performance, several studies have been carried out to address different issues at the Baseband layer including ACL throughput analysis [117]-[123], packet adaptation [114][116] and handoff schemes between piconets [126]. In contrast to the studies focused on the Baseband, there are only a limited number of studies to examine the important interaction between the lower layer (Baseband) and the higher layer (L2CAP). The Logical Link Control and Adaptation Layer Protocol (L2CAP) is layered over the Baseband [104]. L2CAP bridges the data applications and the Baseband by providing services including channel multiplexing, packet segmentation and reassembly, error retransmission. In this chapter, the Bluetooth L2CAP layer is examined in detail. The interaction between Baseband and L2CAP is investigated. Subsequently, the throughput equation of the L2CAP layer is derived by also considering the presence of bit errors.

Since short transmission delay is crucial to many applications and to user experience, it is important to examine the packet delay for the Bluetooth networks. In [129]-[133], authors studied Baseband ACL packet delay by applying different traffic scheduling algorithms to the Bluetooth network. There studies however assume the ACL applications as the only network activity in the analysis. In practice, a Bluetooth network can be used by other applications (i.e. voice) and may support other activities (i.e. sending beacon trains). In this chapter, a more realistic and comprehensive analysis is given for the Baseband ACL packet and L2CAP Packet Data Unit (PDU) delay.

The outline of the chapter is as follows: Section 7.1 provides detailed descriptions of the Baseband and L2CAP layer. Section 7.2 discusses the Baseband ACL packet error rate. A mathematical model that evaluates the overall L2CAP throughput is developed in section 7.3. Subsequently, section 7.4 validates the analytical model by simulations and examines the

effects of the flush timeout and L2CAP PDU size on throughput. In section 7.5, a comprehensive delay model is developed leading to the derivation of the average Baseband packet and L2CAP PDU delays. Finally, based on the mathematical analysis, numerical results of packet average delay are presented in section 7.6.

7.2 Detailed Baseband Layer and L2CAP Layer Properties

7.2.1 Bluetooth Piconet

A Bluetooth link is formed within the context of a piconet. A piconet consists of two or more devices that occupy the same physical channel. The physical channels are identified by different synchronization clocks and hopping sequences. The synchronization clock is identical with the Bluetooth clock of one of the devices in the piconet, known as the master of the piconet. The hopping sequence is derived from the master's clock and the master's Bluetooth device address. All other synchronized devices in the piconet are referred to as slaves.

A number of independent piconets may coexist within a common location. Different piconets are separated by different frequency hopping sequences and have independent piconet clocks. A Bluetooth device may participate concurrently in two or more piconets on a time-division multiplexing basis. Since the piconet is formed by synchronizing to the master's Bluetooth clock, a Bluetooth device can never be a master of more than one piconet. A Bluetooth device may however be a slave in many independent piconets.

Within a piconet, time is divided into slots with $\delta = 625\mu s$. All transmissions are synchronized to this slot grid and are completely controlled by the piconet master. Transmissions from the master and to the master are separated by time slots. As shown in Figure 7.1, the master starts packet transmissions in even slot numbers, while the slaves start in odd slot numbers but only after being polled by the master.

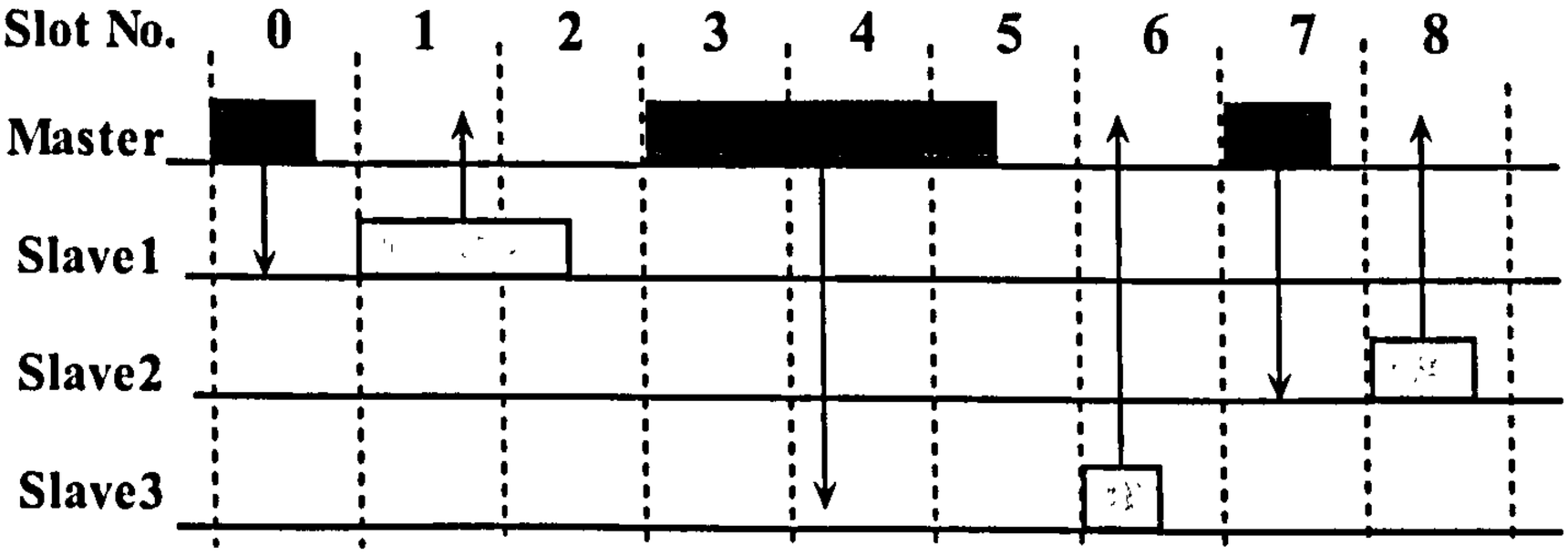


Figure 7.1 Bluetooth Baseband transmission example

7.2.2 Connection modes

A Bluetooth station in the connection state can be in any of the four following modes: Active, Hold, Sniff and Park mode.

- **Active mode:** The Bluetooth unit actively participates in the channel traffic. The master schedules the transmission based on traffic demands to and from different slaves. Active slaves listen in the master-to-slave slots for packets. If an active slave is not addressed, it may sleep until the next master transmission.
- **Sniff mode:** Devices synchronized to a piconet can enter power-saving modes in which device activity is lowered. In the Sniff mode, a slave device listens to the piconet at a reduced rate, thus reducing its duty cycle. The Sniff interval is programmable and depends on the application. It has the highest duty cycle (least power efficient) of all 3 power saving modes (Sniff, Hold & Park).
- **Hold mode:** The master unit can put slave units into Hold mode, where only an internal timer is running. Slave units can also demand to be put into Hold mode. Data transfer restarts instantly when units transition out of Hold mode. This mode has an intermediate duty cycle and medium power efficiency.
- **Park mode:** The Bluetooth device only remains synchronized to the piconet. Parked devices give up their MAC address and periodically listen to the traffic to re-synchronize and check on broadcast messages. It has the lowest duty cycle and the highest power efficiency of all 3 power saving modes.

For the Active, Sniff and Hold stations, a maximum of 8 units can participate in one piconet. Besides having slower duty cycle, the behaviour of Sniff and Hold stations is the same as the Active stations. There is no limit for the number of Park stations in a piconet. In this work, the Bluetooth stations are considered to be either in Active or Park mode.

7.2.3 Logical transports and Baseband packets

Two logical transports are supported by the Bluetooth Baseband [104]: Asynchronous Connectionless (ACL) that is used mainly for data applications and Synchronous Connection Oriented (SCO) that is used mainly for voice applications.

The ACL link is a point-to-multipoint link between the master and all the slaves participating in the piconet. In the slots that are not reserved for the SCO links, the master can establish an ACL link on a per-slot basis to any slave, including the slave already engaged in an SCO link. Packet retransmission is applied for ACL packets.

The SCO link is a symmetric point-to-point link between a master and a single slave in the piconet. The master maintains the SCO link by using reserved slots at regular intervals. The SCO link mainly carries voice information. The SCO packets are used for the 64 KByte/s speech transmissions and are never retransmitted.

The Baseband defines 13 different packet types [104]. Among all the packet types, the Data High rate (DH) packets including DH1, DH3, DH5, as well as the Data Medium rate (DM) packets including DM1, DM3 and DM5 are commonly used for the ACL transport. They are referred to as the ACL packets in the analysis. A rate of 2/3 Hamming Forward Error Correction (FEC) coding is applied to the payload of DM packets, while the payload of DH packets is not protected by any error correction scheme. DM1/DH1, DM3/DH3 and DM5/DH5 packets have a maximum duration of one, three and five time slots respectively. Each data packet is acknowledged by a fast 1-slot acknowledgement (without payload). The packet will be retransmitted until a positive acknowledgment is received or a timeout is exceeded.

Type	Payload Header (bit)	Payload l_{PL} (bit)	CRC (bit)	FEC	Total size of the data field l_{BS} (bit)
DM1	8	0-136	16	2/3	240
DH1	8	0-216	16	no	240
DM3	16	0-968	16	2/3	1500
DH3	16	0-1464	16	no	1496
DM5	16	0-1792	16	2/3	2745
DH5	16	0-2712	16	no	2744

Table 7-1 Bluetooth ACL packets

For the SCO transport, High quality Voice (HV) packets are commonly used. There are three HV packet types: HV1, HV2, and HV3. The master will send SCO packets to the slave at regular intervals (T_{SCO}) in the reserved master-to-slave slots. An HV1 packet can carry a 1.25 ms of speech and is protected by 1/3 FEC, thus an HV1 packet is sent every two time slots ($T_{SCO} = 2\delta = 1.25ms$). An HV2 packet can carry 2.5ms of speech and is protected by 1/3 FEC. An HV2 is sent every four time slots ($T_{SCO} = 4\delta = 2.5ms$). An HV3 (no FEC used) packet can carry 3.75 ms of speech and is sent every six time slots ($T_{SCO} = 6\delta = 3.75ms$). The slave is allowed to transmit in the slave-to-master slot reserved for its SCO logical transport.

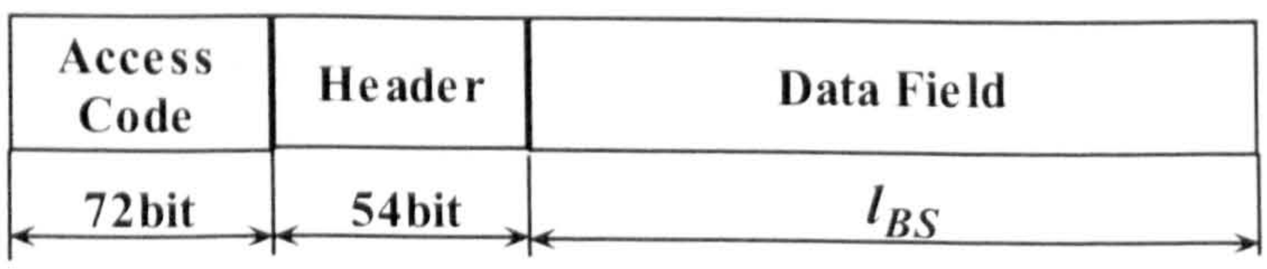


Figure 7.2 Standard Baseband packet

The standard Baseband packet format is illustrated in Figure 7.2. Each packet consists of 3 entities: the access code, the header, and the payload. The access code identifies all packets exchanged on a physical channel: all packets sent in the same physical channel are preceded by the same access code. The header contains link control (LC) information and consists of 6 fields: logical transport address, type code, 1-bit flow control information, 1-bit acknowledge indication, packet sequence number and Header Error Check (HEC). Data Field is the packet payload which carries the data information of the higher layer packet data unit.

7.2.4 Baseband transmission routine

The Baseband transmission (TX) routine is carried out separately for each asynchronous and synchronous link. Figure 7.3 shows the usage of asynchronous and synchronous buffers in the TX routine. In this figure, only one TX asynchronous buffer and one TX synchronous buffer are shown. In the master, there is a separate TX asynchronous buffer for each slave and there are one or more TX synchronous buffers for each synchronous slave. Each TX buffer consists of two First In First Out (FIFO) registers: one current register which can be accessed and read by the Link Controller in order to compose the packets, and the other register that can be accessed by the Baseband Resource Manager to load new information.

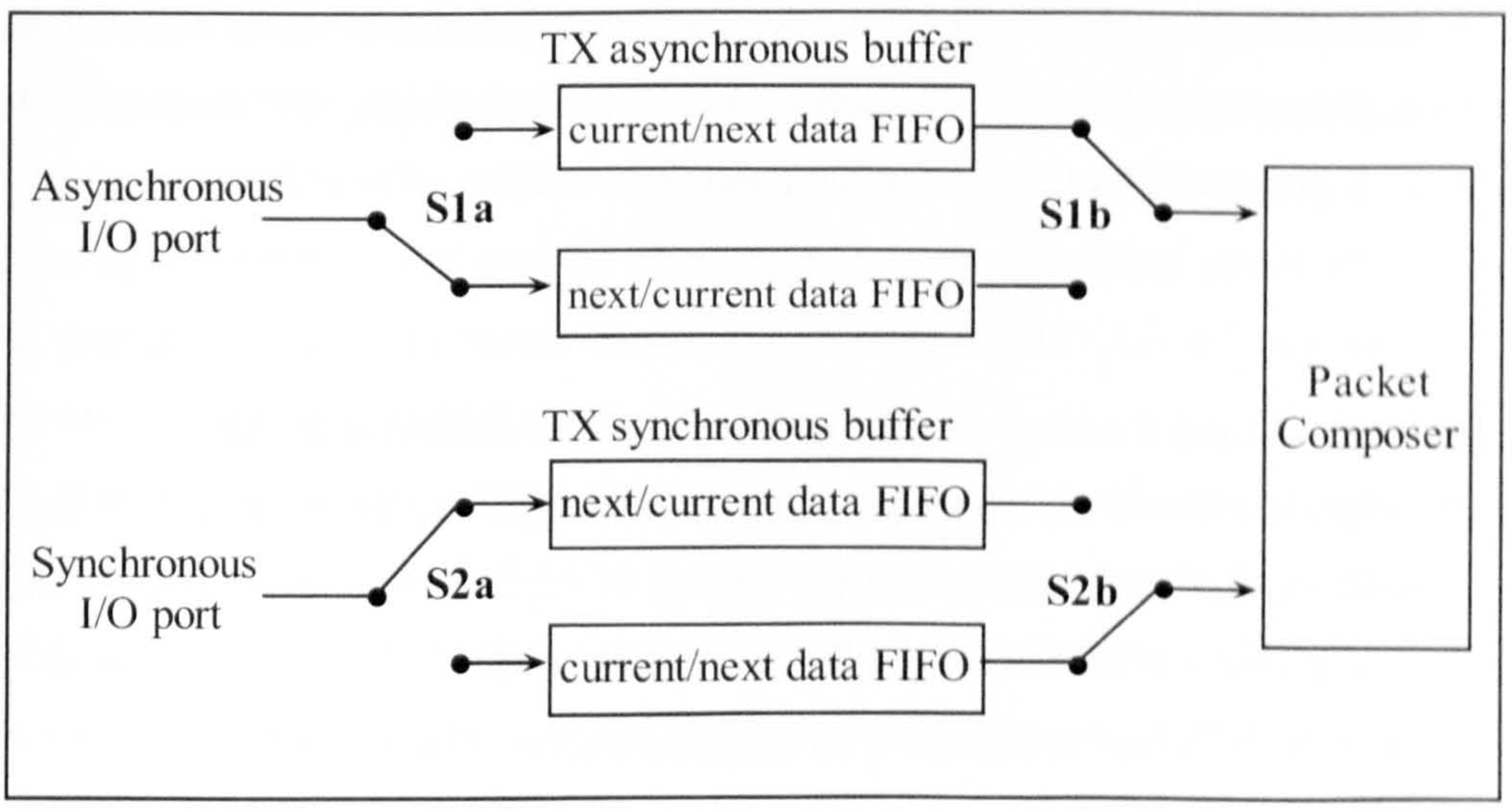


Figure 7.3 Baseband transmission buffer

The TX routine works as follows. For ACL links, the Baseband Resource Manager (BRM) loads new data in the register to which the switch S1a points. Next, BRM gives a command to the Link Controller, which forces the switch S1 to change (both S1a and S1b switch synchronously). When the payload needs to be sent, the packet composer reads the current register and, depending on the packet type, builds a payload which is appended to the

channel access code and the header. The packet is subsequently transmitted. The response packet arrives in the following RX slot if it concerned a master transmission, or may be postponed until some later RX slot if it concerned a slave transmission. The result of the transmission is reported back in the response packet. In the case of an ACK, the switch S1 changes position and loads the payload in the next register; if a NACK (explicit or implicit) is received instead, the switch S1 will not change position. In this case, the same payload is retransmitted at the next TX occasion.

For SCO links, the BRM loads new data into the register to which the switch S2a points. Next, it forces the switch S2 to change. During the SCO reserved slots, the packet composer reads the current register, builds the appropriate HV packet and sends the packet off. In contrast to ACL links, the BRM does not wait for any response and switches S2 to load the next packet since there is no retransmission for the HV packets.

7.2.5 Baseband packet Flush Timeout

For each Baseband packet, a Flush Timeout (t_{TO}) [104] is used to limit the retransmission. It is the maximum transmission time for any Baseband packet. When Flush Timeout expires, all segments of the L2CAP packet data unit (PDU), which the Baseband packet belongs to, are flushed from the Baseband buffer. Although Flush Timeout can be disabled, in practice it is often implemented. With Flush Timeout, the Baseband is more efficient and fair as no packet can be retransmitted forever and each packet has a limited longest transmission time. Flush Timeout is negotiated at the connection procedure and the value is between $625\mu s$ and 1.28 s.

7.2.6 The Logical Link Control and Adaptation Layer Protocol (L2CAP)

L2CAP is layered above the Baseband Protocol and regarded as the data link layer. It supports channel multiplexing and carries the quality of service information [104]. L2CAP also provides optional error retransmission and flow control. All the Bluetooth data applications use L2CAP to communicate with the Baseband as shown in Figure 2.6.

L2CAP supports 3 different operation modes, namely Basic, Flow Control and Retransmission. Basic mode is the simplest one. It provides three functionalities: protocol multiplexing, PDU segmentation and reassembly, quality of service. Besides the functionalities of Basic mode, PDU's sent to a peer entity are numbered and acknowledged in Flow Control and Retransmission modes. In Flow Control mode no retransmissions take place, missing PDU's however can be detected and reported as lost. In Retransmission mode, PDU is retransmitted when needed to ensure that all PDU's are delivered to the peer.

In the analysis, L2CAP is assumed to operate in the 'retransmission' mode. This mode is widely implemented since it provides a reliable and error-free communication link for the data applications. In this mode, L2CAP employs an independent error check scheme Frame Check Sequence (FCS) on L2CAP PDU protecting it against any possible failures of the Cyclic Redundancy Check (CRC) of Baseband packets. Retransmission of L2CAP PDU is triggered when the FCS indicates error or Baseband discards the PDU due to the expiration of Flush timeout. L2CAP permits higher level protocols and applications to transmit and receive L2CAP data packets up to 65531 Bytes long. The structure of the L2CAP PDU is given in Figure 7.4.

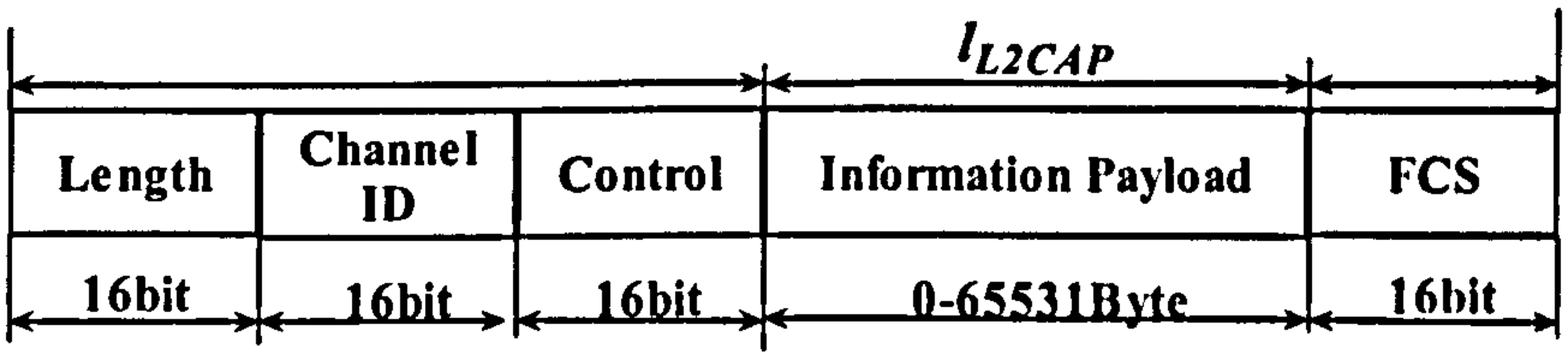


Figure 7.4 L2CAP PDU format

7.2.7 Coding schemes for error detection and correction

Since Bluetooth is a wireless communication system, the system is considered inherently unreliable in poor RF environments. To counteract this, the system provides levels of protection at each layer. The header of a baseband packet uses FEC coding to allow error correction by the receiver and an 8-bit HEC to detect any remaining errors after correction. For the payload, certain Baseband packet types include FEC for the payload (e.g. DM packets) and a 16-bit CRC is applied for error detection.

The resulting reliability gained by this ARQ error retransmission scheme is only as dependable as the ability of the HEC and CRC codes to detect errors. In most cases this is sufficient, however it has been shown that for longer packets, the probability of an undetected error is too high to support data applications [104], especially those with a large amount of data being transferred.

The L2CAP layer provides an additional level of error control that is designed to detect the occasional undetected errors in the baseband layer and request retransmission of the affected data. Another 16-bit CRC code named FCS is employed by L2CAP to check for errors in each L2CAP PDU. This provides the level of reliability required by most data applications.

7.3 Baseband ACL Packet Error Probability

As discussed in [117], the probability of retransmission can be determined for any instantaneous bit error rate (p_b) according to the real-time signal-to-noise ratio. For any given ACL packet, error occurs when any of the following five events occur [117]:

A: the destination radio fails to synchronize with the access code of the forward packet;

B: the header of the forward packet is corrupted;

C: the payload of the forward packet is corrupted;

D: the source is unable to synchronize with the access code of the return packet (1-slot Baseband acknowledgement)

E: the header of the return packet is corrupted.

An ACL packet is in error and has to be retransmitted if any of the events A-E occur. For any given bit error rate of p_b, the packet error rate is:

$$p = A(p_b) + B(p_b) + C(p_b) + D(p_b) + E(p_b) \tag{7.1}$$

where $A(p_b), \ldots, E(p_b)$ are the possibility of events A, . . . , E respectively.

Synchronization between master and slave is achieved by correlating the 72 bits access code with a stored copy of the access code. Since the demodulation circuit needs time to initialise, the output of the first few bits is often lower than the signal detection level. In the analysis, the circuit is assumed to require 10 bits to initialise. Thus, the effective access code is 62 bits. The packet is synchronized only if the output of the correlator exceeds the synchronization threshold (S_{syn}). S_{syn} is chosen according to the desired sensitivity level of the false alarm. A packet can be successfully synchronized when no more than (62-S_{syn}) errors are in the received access code. The probabilities of event A and D are:

$$A(p_b) = D(p_b) = 1 - \sum_{i=0}^{62-S_{syn}} \binom{62}{i} \left({p_b}^i (1-p_b)^{62-i} \right) \tag{7.2}$$

where i stands for the number of error bits.

The header payload is 18 bits and encoded by the (3, 1) Hamming coding resulting in a 54 bits header [104]. Since the (3, 1) code can correct one error for every 3-bit encoded block (triplet), a triplet is erroneous only if 2 bits or more are incorrect. The header of a packet is corrupted if any of the 18 encoded triplets are decoded incorrectly:

$$B(p_b) = E(p_b) = 1 - \left(\binom{3}{2} p_b (1-p_b)^2 + (1-p_b)^3 \right)^{18} = 1 - (3p_b(1-p_b)^2 + (1-p_b)^3)^{18} \tag{7.3}$$

The most likely error event is type C. For the DH packets, event C occurs if any of the payload bits is received incorrect and thus:

$$C(p_b) = 1 - (1-p_b)^{l_{BS}} \tag{7.4}$$

where l_{BS} is the data field size of a Baseband packet, as shown in Table 7-1, l_{BS}=240bits for DH1, l_{BS} =1496bits for DH3, and l_{BS} =2744bits for DH5.

The payload of DM packets is protected by the (15, 10) Hamming coding, which is capable of correcting one error bit per 15-bit encoded block. The payload is corrupted if any of the code blocks contain more than one error:

$$C(p_b)=1-\left(\binom{15}{14}p_b(1-p_b)^{14}+(1-p_b)^{15}\right)^{Q}=1-\left(15p_b(1-p_b)^{14}+(1-p_b)^{15}\right)^{Q} \quad (7.5)$$

where $Q= l_{BS}$ /15. Q=16 for DM1, Q=100 for DM3, and Q=183 for DH5.

7.4 L2CAP throughput modelling

The challenge of transmitting a large higher layer packet is considered in the analysis. A L2CAP PDU is to be fragmented and transmitted in many Baseband packets. To derive the L2CAP throughput for a Bluetooth station, a saturation case is considered. In this case, there is always a L2CAP PDU ready to be sent from the master to a slave, and the slave only acknowledges the incoming packet and has no data to send back to the master. The Bluetooth transmission model is illustrated in Figure 7.5.

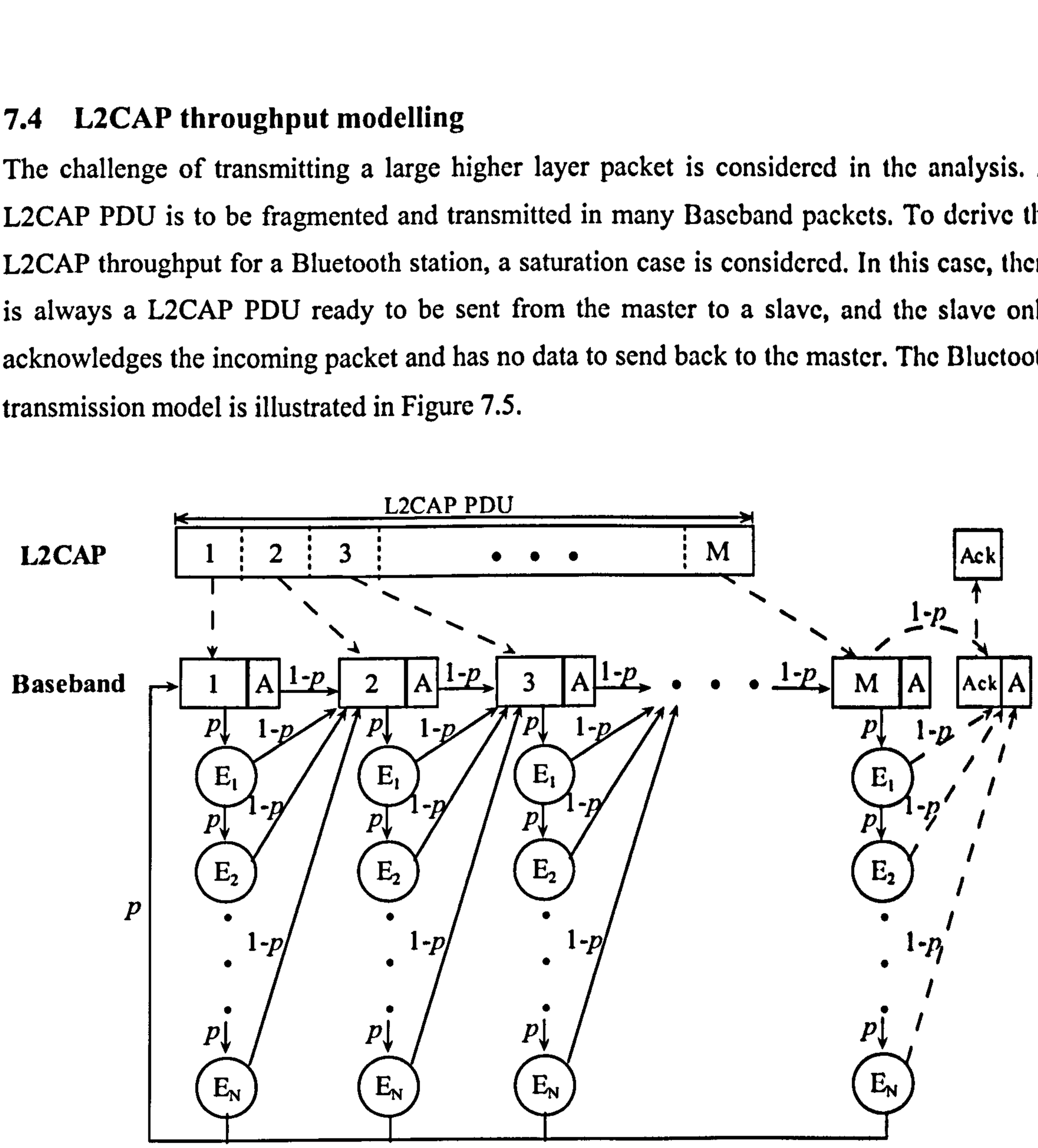


Figure 7.5 Bluetooth transmission model

A Baseband packet will be retransmitted until it is successfully delivered or the Flush timeout has expired. The possibility (p) of having an erroneous Baseband packet in the transmission has been derived in section 7.2. If the Flush timeout of the Baseband packet expires, the Baseband packet will be discarded. Subsequently the L2CAP PDU to which the discarded Baseband packet belongs will also be discarded. The Bluetooth Baseband therefore has to retransmit the whole L2CAP PDU if any of its Baseband packets are discarded.

As shown in Figure 7.5, the average time to transmit one Baseband packet should take into consideration all the possible retransmissions. By including a 1-slot acknowledgement, the average time to transmit one Baseband packet is given by:

$$T_{BS} = a\delta + a\delta\sum_{i=1}^{N} p^i = a\delta\left(1+\frac{p(1-p^N)}{1-p}\right) \tag{7.6}$$

where a is the number time slots for transmitting different Baseband packets: a=2 for DH1&DM1, a=4 for DH3&DM3 and a=6 for DH5&DM5, δ is a single time slot ($\delta = 625\times10^{-6}s$), N is the number of retransmissions before expiration of the Flush Timeout (t_{TO}), thus N has the value of $N = \left\lfloor \frac{t_{TO}}{a\delta} \right\rfloor - 1$.

The error detection scheme of L2CAP identifies errors due to the failures of the HEC or CRC error checks on the baseband packets. L2CAP retransmits the whole PDU if any error occurs.

As discussed in [127] [128], CRC is not infallible. For an n-bit checksum, 1 of 2^n random blocks will have the same checksum for non-equivalent data blocks. Thus, for the n-bit error checking coding, 1 in 2^n errors cannot be detected. L2CAP employs another 16-bit FCS for the error checking. The probability of having undetected errors in HEC (8 bits) and CRC (16 bits) of a Baseband packet is given as follows:

$$p_{un} = {(B(p_b)+E(p_b))}/{2^8} + {C(p_b)}/{2^{16}} \tag{7.7}$$

When a Baseband packet is retransmitted more than N times, t_{TO} expires. It causes the whole L2CAP PDU, which the Baseband packet belongs to, to be discarded and L2CAP will retransmit the PDU. By considering the probability of L2CAP retransmission, Figure 7.5, the time to successfully transmit a L2CAP PDU with 1, 2, 3, . . . , n Baseband packets is derived:

$$T(1) = T_{BS} + p^{N+1}T(1) = \frac{T_{BS}}{1-p^{N+1}} \tag{7.8}$$

$$T(2) = T(1) + T_{BS} + p^{N+1}T(2) = \frac{T(1) + T_{BS}}{1 - p^{N+1}} \tag{7.9}$$

$$T(3) = T(2) + T_{BS} + p^{N+1}T(3) = \frac{T(2) + T_{BS}}{1 - p^{N+1}} \tag{7.10}$$

...

$$T(n) = T(n-1) + T_{BS} + p^{N+1}T(n) = \frac{T(n-1) + T_{BS}}{1 - p^{N+1}} \tag{7.11}$$

where n is a positive integer larger than 1 (n=2, 3, ...).

Since the acknowledgement of L2CAP is only the L2CAP header, a 1-slot Baseband packet is enough to carry the L2CAP ack. Due to the small size of the L2CAP acknowledgement packet, its transmission is assumed to be error free. By including the Baseband acknowledgement, as shown in Figure 7.5, two time slots are needed for a L2CAP ack ($T_{ack} = 2\delta = 1.25ms$). Taking into account the probability of undetected errors (p_{un}) in a Baseband packet, for a PDU that is fragmented into M packets, the average time to successfully transmit the L2CAP PDU and receive the L2CAP ack is given by:

$$T_{L2CAP}(M) = (T(M) + T_{ack})(1 + Mp_{un}) = \left(\frac{T(M-1) + T_{BS}}{1 - p^{N+1}} + T_{ack} \right)(1 + Mp_{un}) \tag{7.12}$$

where $M = \left\lceil \frac{l_{L2CAP} + l'_{L2CAP}}{l_{PL}} \right\rceil$, l_{L2CAP} and l'_{L2CAP} are the payload and the 8 Bytes header of the L2CAP PDU respectively, l_{PL} is the payload size of a Baseband packet as shown in Table 7-1.

Finally, the Bluetooth throughput at the L2CAP layer can be derived as:

$$F = \frac{l_{L2CAP}}{T_{L2CAP}(M)} \tag{7.13}$$

As a data application saturation case is considered in the above analysis (only L2CAP PDU is transmitted), the derived throughput F is the maximum throughput of the Bluetooth data application for one Bluetooth station (either master or slave).

7.5 Model validation and effects of the flush timeout and L2CAP size

7.5.1 Simulation program for Bluetooth protocol stacks

In order to examine the accuracy of the derived mathematical modelling at the Bluetooth L2CAP layer level, a set of simulation programs is developed to emulate the Bluetooth

protocol stacks (up to the L2CAP layer) by using MATLAB™. The source code of the programs is presented in Appendix III. As shown in Figure 7.6, the simulation programs consist of 2 different 'agents' for the L2CAP and Baseband respectively, as well as a random error generator acting as the Bluetooth physical channel. A brief description of different simulation modules is provided in the following sections.

Bluetooth Physical channel:

Function `physical` realises the Bluetooth physical layer. Based on the average bit error probability, the synchronization threshold, the length and the type of the Bluetooth packet, it randomly generates errors in order to emulate the nature of radio communication channel. Three of the possible error events are considered in the function. The function returns a Boolean value where 0 indicates the packet transmission error free and 1 indicates the packet is corrupted.

Bluetooth Baseband:

Function `Baseband` is the Bluetooth Baseband layer imitating the behaviour of both Bluetooth masters and slaves. It calculates the time to transmit one Baseband packet and keeps track of the status of the flush timeout. By using the function `physical`, `Baseband` considers the channel error and includes all the possible retransmission. The function returns the time needed to transmit the given baseband packet.

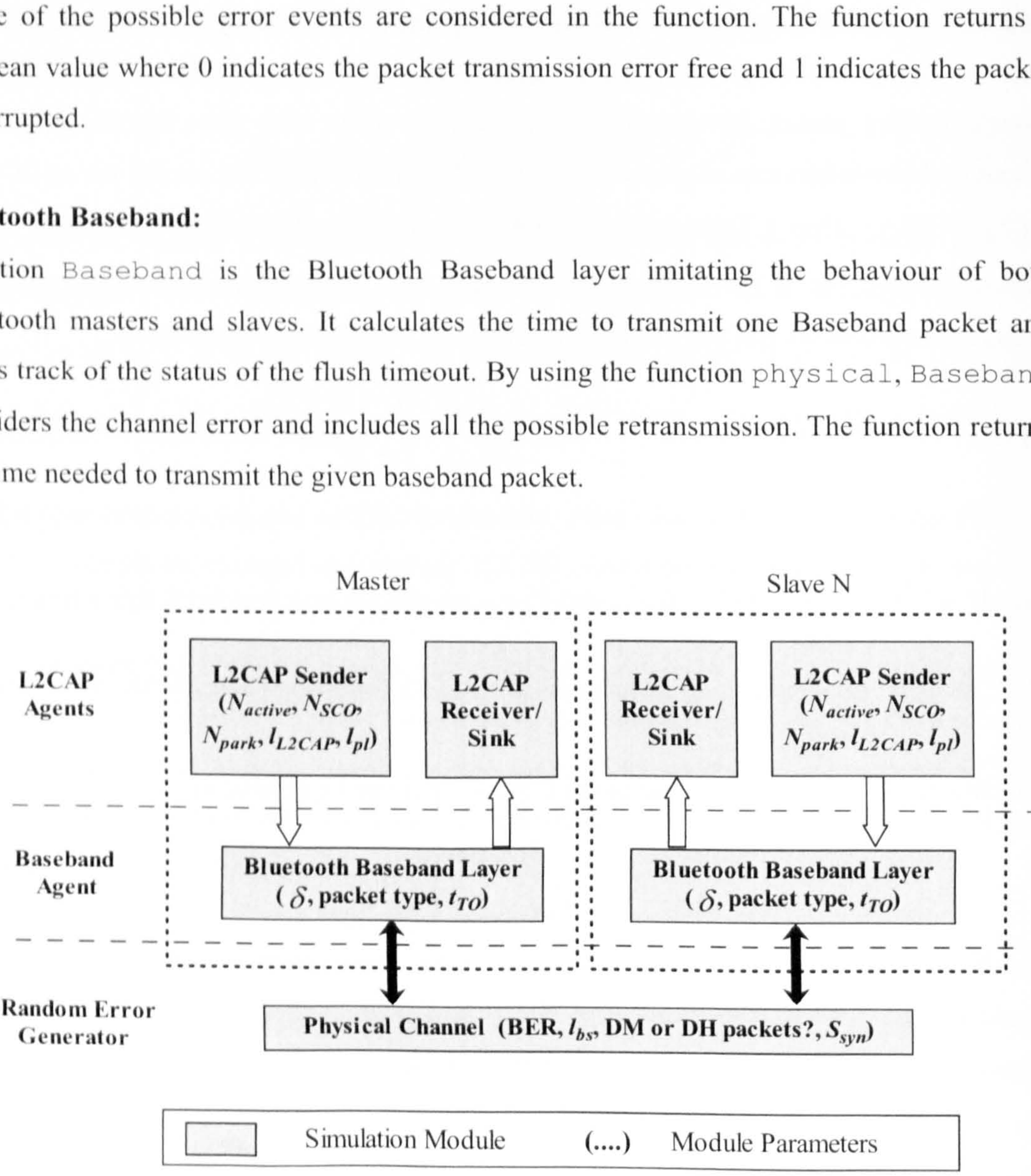


Figure 7.6 Structure of the simulation program for Bluetooth protocol stacks

Bluetooth sender:

Function `l2capsender` is the L2CAP PDU sender. It calculates the time to transmit a L2CAP PDU with size of l_{L2CAP}. By using the function `baseband` and `l2capreceiver`, `l2capsender` includes all the possible retransmissions by considering the possible Baseband timeout and the undetected errors in Baseband packets. The function returns the transmission time of the given L2CAP PDU (Tl2capsender).

Bluetooth receiver:

Function `l2capreceiver` is the L2CAP receiver. It calculates the time to transmit a L2CAP ack and also decides if there are any undetected Baseband packet errors. The function returns the transmission time of a L2CAP acknowledgement.

7.5.2 Comparison with simulation

In Figure 7.7, the L2CAP throughput of six different type ACL packets is plotted against bit error rate in the range of 10^{-6} to 10^{-1}. L2CAP PDU size l_{L2CAP} of 65531 Bytes (maximum PDU size [104]), synchronization threshold (S_{syn}) of 60-bit and Flush Timeout t_{TO} of 0.1 s are used. The simulation results are also plotted in Figure 7.7 to compare with the analytical results. The mean ($\bar{x}$) of the difference between the analytical and the simulation results, as well as the standard deviation (σ) of the difference are also calculated for each type of Baseband packets by using equations (3.17)-(3.19).

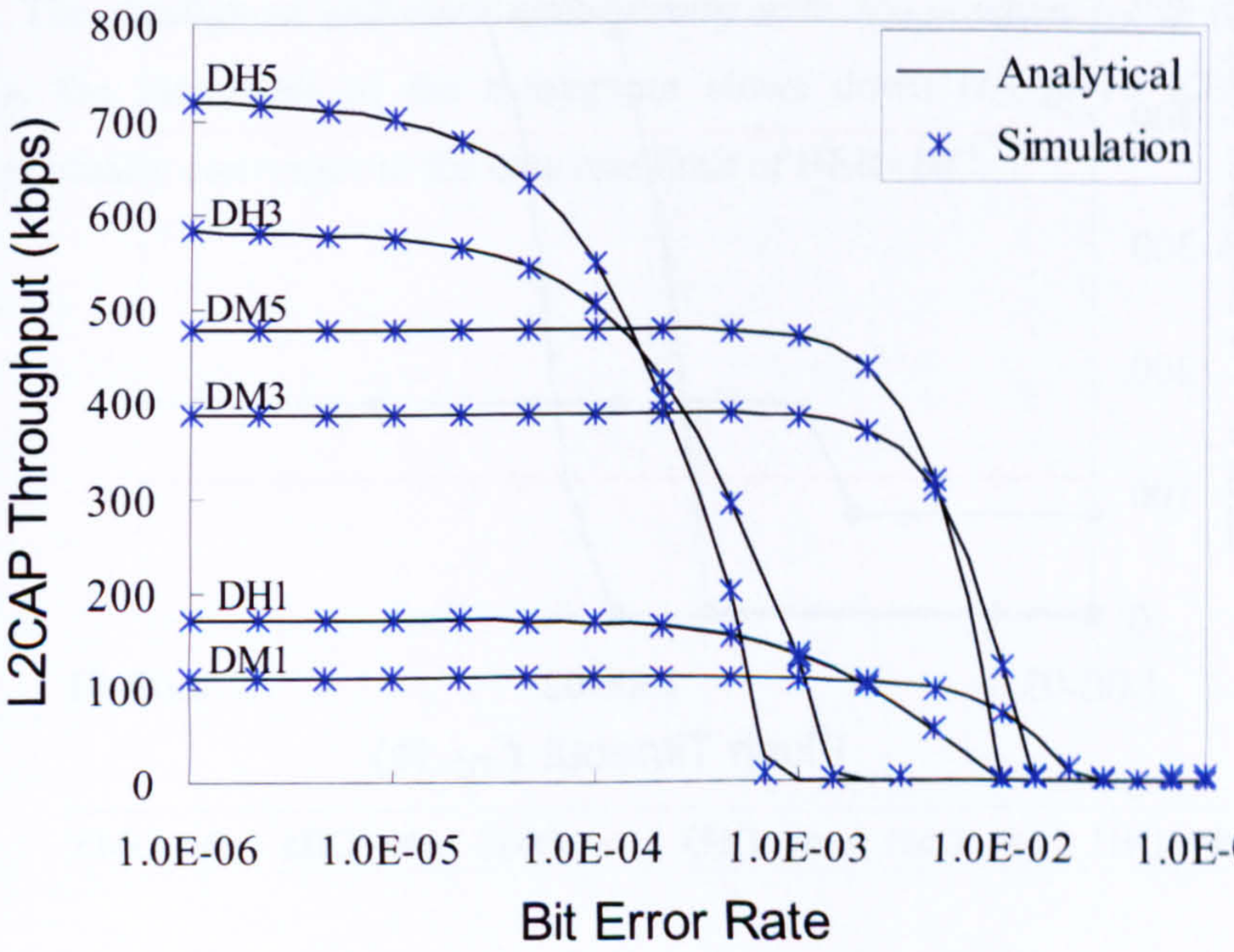


Figure 7.7 L2CAP throughput for 6 types of Baseband packet in the BER range of 10^{-6}-10^{-1}

$\bar{x}=0.30$ and $\sigma=0.32$ for DH1, $\bar{x}=0.74$ and $\sigma=0.86$ for DH3, $\bar{x}=1.46$ and $\sigma=2.66$ for DH5, $\bar{x}=0.55$ and $\sigma=1.29$ for DM1, $\bar{x}=0.59$ and $\sigma=0.77$ for DM3, $\bar{x}=1.04$ and $\sigma=1.49$ for DM5. The unit is kbps.

The simulation results show very good agreement with the analytical results by observing the values of $\bar{x}$ and σ, as well as the 'simulation-analytical' curve pairs. This verifies the accuracy of our mathematical model. At the low BER, the L2CAP throughput for each corresponding packet type converges to the maximum data rates (shown in Table 7-1). All the throughputs drop significantly with the increasing BER and approach 0 when BER>10^{-2}. Without the FEC mechanism, the DH packets are more susceptible to error. Although the performance of DH packets is poor at the high BER, they yield much better throughput in the low BER than the DM packets without the FEC overhead.

7.5.3 Effect of the Flush timeout on L2CAP throughput

By varying t_{TO} from 10^{-3}s to 0.11s, the throughput of 6 different types of Baseband packet is plotted in Figure 7.8. The following parameters are used: l_{L2CAP}=65531Bytes, S_{syn}=60bits, BER=10^{-4}.

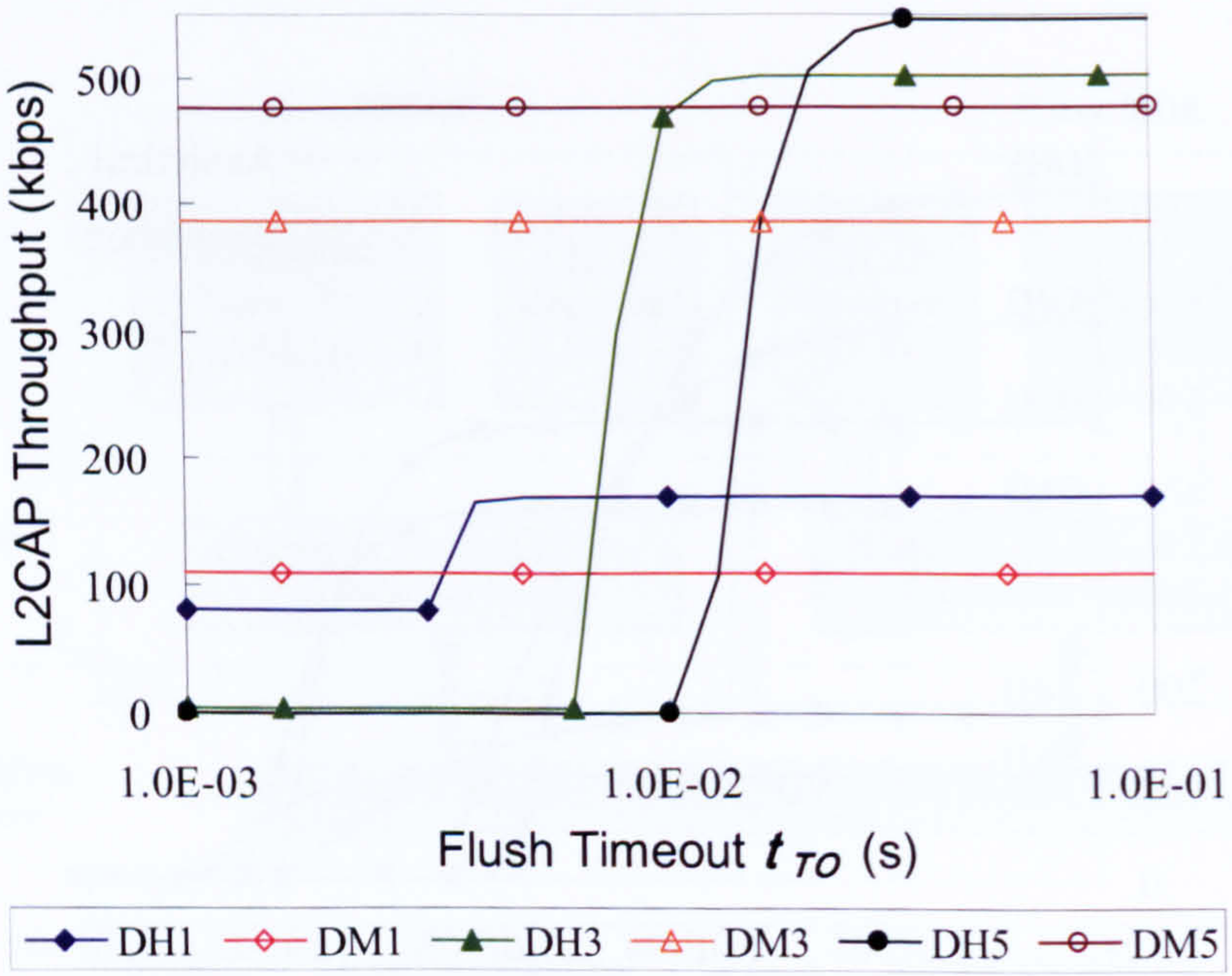


Figure 7.8 L2CAP throughput for 6 types of Baseband packet when t_{TO} is in the range of 10^{-3}s-0.1s

If t_{TO} expires, the Baseband packet is discarded and the related L2CAP PDU is flushed from the buffer. Consequently, L2CAP has to retransmit the PDU. When t_{TO} is small, the number of possible Baseband packet retransmissions is limited to $N = \left\lfloor \frac{t_{TO}}{a\delta} \right\rfloor - 1$ and L2CAP PDU's are dropped and retransmitted more frequently. As shown in Figure 7.8, the DH packets have a poor performance when t_{TO} is small. The throughput of the DM packets is however not significantly affected by t_{TO}. Because the DM packets are more resistant to errors due to the FEC protection, they do not need to be retransmitted frequently. On the contrary, the DH packets are susceptible to error and thus rely more on the Baseband packet retransmission. In summary, a large t_{TO} (t_{TO}>0.1 s) is required for the DH packets to achieve good throughput. t_{TO} has only a minor effect on the DM packets compared to the effect on the DH packets.

7.5.4 Effect of the L2CAP PDU size on L2CAP throughput

The effect of L2CAP PDU size l_{L2CAP} on the L2CAP throughput is illustrated in Figure 7.9. The following parameter values are used: t_{TO} =0.1s, S_{syn}=60bits, BER=10^{-4}. When l_{L2CAP} is small, the PDU is divided into a small number of Baseband packets, especially for the large packets (DH/DM3, DH/DM5). The last Baseband packet, which may not be filled, has a significant effect on the throughput for a small PDU. The throughput therefore fluctuates when l_{L2CAP} is small, Figure 7.9. Despite the fluctuation, the trend of the throughput increases with l_{L2CAP}. The throughput increases significantly with l_{L2CAP} when l_{L2CAP} is small. For a larger l_{L2CAP}, the increment of the throughput slows down (l_{L2CAP}>10 KBytes) and the throughput gradually converges to the data rate limit of BER=10^{-4}.

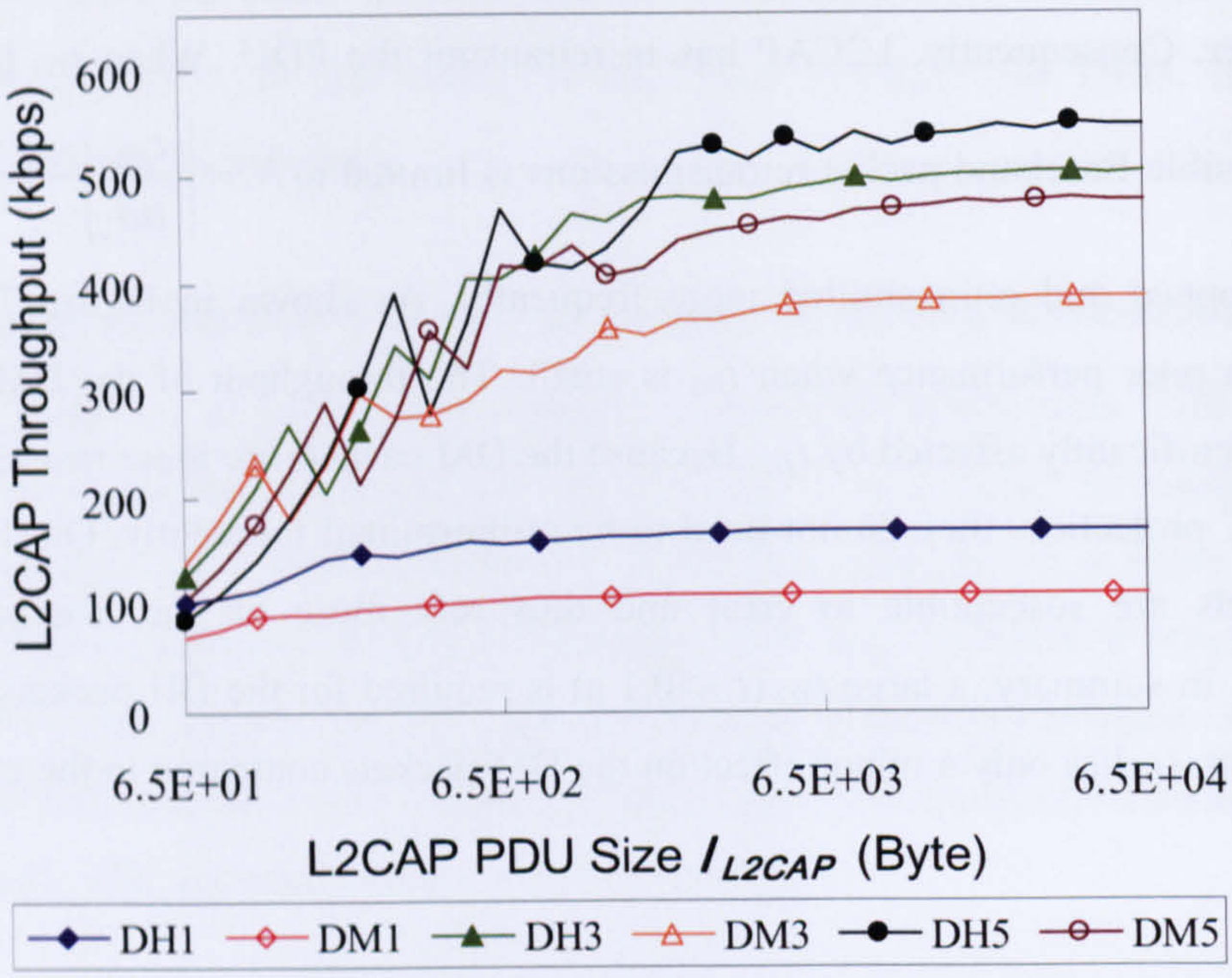


Figure 7.9 L2CAP throughput for 6 types of Baseband packet when l_{L2CAP} in the range of 65-65000 Bytes

7.6 Analysis of the Baseband ACL packet and L2CAP PDU Delay

7.6.1 Average ACL Baseband packet delay

In this analysis, it is assumed that all the slaves have finished the connection procedures and are connected to the piconet master. Since the master takes full control of and schedule the network traffic, there is no access delay for any stations. The average delay D_{BS} for transmitting an ACL Baseband packet is defined as the time interval from the time an ACL packet is loaded into the Baseband buffer to the time when a positive ack for this packet is received. D_{BS} is given by:

$$D_{BS} = \delta A_n S_t \tag{7.14}$$

where A_n is the average number of transmissions (including retransmissions) required to transmit the ACL Baseband packet and S_t is the average number of time slots required for each ACL Baseband packet transmission.

By adding the first packet transmission and N possible retransmissions, A_n is given by:

$$A_n = 1 + \sum_{i=1}^{N} p^i = \sum_{i=0}^{N} p^i = \frac{1 - p^{N+1}}{1 - p} \tag{7.15}$$

where N has the value of $\left\lfloor \frac{t_{TO}}{\delta S_t} \right\rfloor - 1$.

The master schedules the transmission based on traffic demands to and from different active slaves. It also reserves time slots for SCO links and supports regular transmissions to keep parked slaves synchronized to the channel. Thus, S_t should include the time slots needed to transmit the ACL packet, the delay of ACL traffic from other active slaves, and other network activities (to be discussed later). S_t depends on the number of active slaves (N_{active}), the probability of starting ACL packet transmission for each time slot (P_{start}), and the number of time slots (a) needed for the ACL packet transmission.

A saturation case is considered, where there are always ACL packets ready to send between the master and each of the active slaves. Since no traffic scheduling policy is prescribed in the Bluetooth specification [104], masters can manage the traffic by using any scheduling algorithm. The performance of different scheduling algorithms has been analyzed through discrete event simulation techniques in [129] and [131]. In the following analysis, we assume the master manages the ACL traffic by polling the active slaves sequentially and evenly, thus, the piconet is equally shared by all the active slaves. S_t is given by:

$$S_t = a\frac{1}{P_{start}/N_{active}} = \frac{aN_{active}}{P_{start}} \tag{7.16}$$

where a=2 for DH1&DM1, a=4 for DH3&DM3 and a=6 for DH5&DM5. Up to 7 active slaves are supported in one piconet, thus $N_{active} \le 7$. As described in section 7.1, Bluetooth masters use odd time slots for transmission, while the slaves use even time slots. Thus, for each time slot, a master sends an ACL packet with probability of $P_{start}/2$. Hence, the probability of a master sending an ACL packet to a specific active slave is $P_{start}/2N_{active}$ for a piconet with N_{active} active slaves. Similarly, an active slave has the probability of $P_{start}/2N_{active}$ of sending an ACL packet to the master. Therefore, for each time slot, the possibility of starting an ACL Baseband packet transmission, either from the master to a specific active slave or from an active slave to the master, is P_{start}/N_{active} as given in equation (7.16).

In order to derive S_t, P_{start} is to be identified first. P_{start} is related to the status of piconet which is affected by two other network activities: A) SCO traffic; B) the transmission of beacon train and access windows for parked slaves. For each piconet time slot, there are four possible events: 1) starts an ACL packet transmission with probability P_{start}; 2) being in the duration of an ACL packet transmission with probability P_{dur}; 3) starts or being in the duration of a SCO transmission with probability P_{SCO}; 4) starts or being in the duration of a transmission of beacon trains and access windows with probability of P_{Park}. The following equation always holds true:

$$P_{start} + P_{dur} + P_{SCO} + P_{Park} = 1 \tag{7.17}$$

P_{dur}, P_{SCO} and P_{Park} have to be identified in order to derive P_{start}. First, the relationship of P_{start} and P_{dur} is to be examined.

P_{start} and P_{dur} can be subdivided into:

$$P_{start} = P_{start1} + P_{start3} + P_{start5} \tag{7.18}$$

$$P_{dur} = P_{dur1} + P_{dur3} + P_{dur5} \tag{7.19}$$

P_{start1}, P_{start3}, P_{start5} are the probabilities of starting a 1, 3, 5-slot packet transmission respectively. P_{dur1}, P_{dur3}, P_{dur5} are the probabilities of being in the duration (excluding the 'starting' slot) of 1, 3, 5-slot packet transmission respectively.

1-slot packets always start and finish transmission within one time slot, thus, $P_{dur1} = 0.3$ and 5-slot packets last more than one time slot. Therefore, for any given time slot, only 3 and 5-slot packets can be in the duration of a packet transmission. By excluding the 'starting' slot (P_{start3} and P_{start5}), there are two and four 'duration' slots for 3 and 5-slot packets respectively, thus:

$$P_{dur3} = \begin{cases} \dfrac{2P_{start3}}{2P_{start3} + 4P_{start5}} P_{dur} \\ 2P_{start3} \end{cases} \tag{7.20}$$

$$P_{dur5} = \begin{cases} \dfrac{4P_{start5}}{2P_{start3} + 4P_{start5}} P_{dur} \\ 4P_{start5} \end{cases} \tag{7.21}$$

From (7.19), (7.20) and (7.21), P_{dur} is given as:

$$P_{dur} = P_{dur3} + P_{dur5} = 2P_{start3} + 4P_{start5} \tag{7.22}$$

To determine the relationship between P_{dur} and P_{start}, we need to identify P_{start3} and P_{start5} respectively. In this analysis, the simplest scenario is considered: the probabilities of transmitting 1, 3 or 5-slot packets are the same, thus the probabilities of starting the transmission of 1, 3 or 5-slot packets are the same:

$$P_{start1} = P_{start3} = P_{start5} = \frac{1}{3} P_{start} \tag{7.23}$$

It should be noted that, the delay model is not limited by this assumption. P_{start1}, P_{start3} and P_{start5} can be set to any value according to the percentages of 1, 3 and 5- slot ACL packets being used in the real situation.

By substituting (7.23) into (7.22), P_{dur} becomes:

$$P_{dur} = 2P_{start} \tag{7.24}$$

By substituting (7.24) into (7.17), P_{start} is:

$$\begin{aligned} P_{start} &= 1-(P_{dur}+P_{SCO}+P_{Park}) \\ &= 1-(2P_{start}+P_{SCO}+P_{Park}) \\ &= \frac{1-(P_{SCO}+P_{Park})}{3} \end{aligned} \tag{7.25}$$

In the following section, P_{SCO} and P_{Park} are to be identified.

7.6.1.1 The probability of transmitting SCO traffic (P_{SCO})

When a piconet is used by a voice application, an SCO link is established between the master and the slave. The master will send SCO packets to the slave at regular intervals (T_{SCO}) in the reserved master-to-slave slots. The SCO packets HV1, HV2 or HV3 will be used. By considering duplex voice applications (i.e. telephone) for the SCO transports, the same HV packet type is assumed to be used by both the master and the slave.

When HV1 packets are used for one SCO transport, all the time slots are reserved and no other traffic can be carried by the piconet. When HV2 packets are used for one SCO transport, the master reserves half of the time slots for the SCO and uses the rest for ACL traffic. When HV3 packets are used for one SCO transport, the master also supports ACL links after reserving one third of the time slots for the SCO. To summarize, a piconet can simultaneously support the maximum of one, two and three SCO transports respectively for HV1, HV2 and HV3. When a SCO link is established, the piconet can carry ACL traffic only if HV2 and HV3 are used. Therefore, the probability for a time slot that is used by SCO links is given by:

$$P_{SCO}\begin{cases} =1 & \text{for HV1} \\ =\dfrac{N_{SCO}}{2} & \text{for HV2} \\ =\dfrac{N_{SCO}}{3} & \text{for HV3} \end{cases} \tag{7.26}$$

where N_{SCO} is the number of the established SCO links. $N_{SCO} \le 2$ for HV2, $N_{SCO} \le 3$ for HV3.

7.6.1.2 The probability of transmitting beacon train and access windows (P_{Park})

When a slave does not need to participate in the piconet channel, but still wants to remain synchronized to the channel, it enters Park mode. Park mode is a state with little activity. There is no limit on the number of parked slaves in a piconet. To support parked slaves, the master establishes a beacon train when one or more slaves are parked. The beacon train

consists of one beacon slot or a train of equidistant beacon slots which is transmitted periodically with a constant time interval. The beacon train is illustrated in Figure 7.10. A train of N_B ($N_B \geq 1$) beacon slots is defined with an interval of T_{Beacon}. The beacon slots in the train are separated by ΔB slots. N_B and T_{Beacon} are chosen according to the channel characteristics such that there are sufficient beacon slots for parked slaves to synchronize.

Following the beacon train, an access window is sent where the parked slaves can send requests to be unparked, Figure 7.10. To increase reliability, the access window can be repeated N_A times ($N_A \geq 1$). In the access window, master polls for each parked slave in the master-to-slave slot. The slave-to-master slot is divided into two half time slots of $312.5\mu s$. In each half slot, a parked slave is allowed to respond according to its access request address (AR_ADDR). The slave shall only send an access request in the proper slave-to-master half slot if it has been polled in the preceding master-to-slave slot. Thus, there are $2\lceil N_{Park}/2 \rceil$ time slots in the access window for a piconet that has N_{Park} parked slaves.

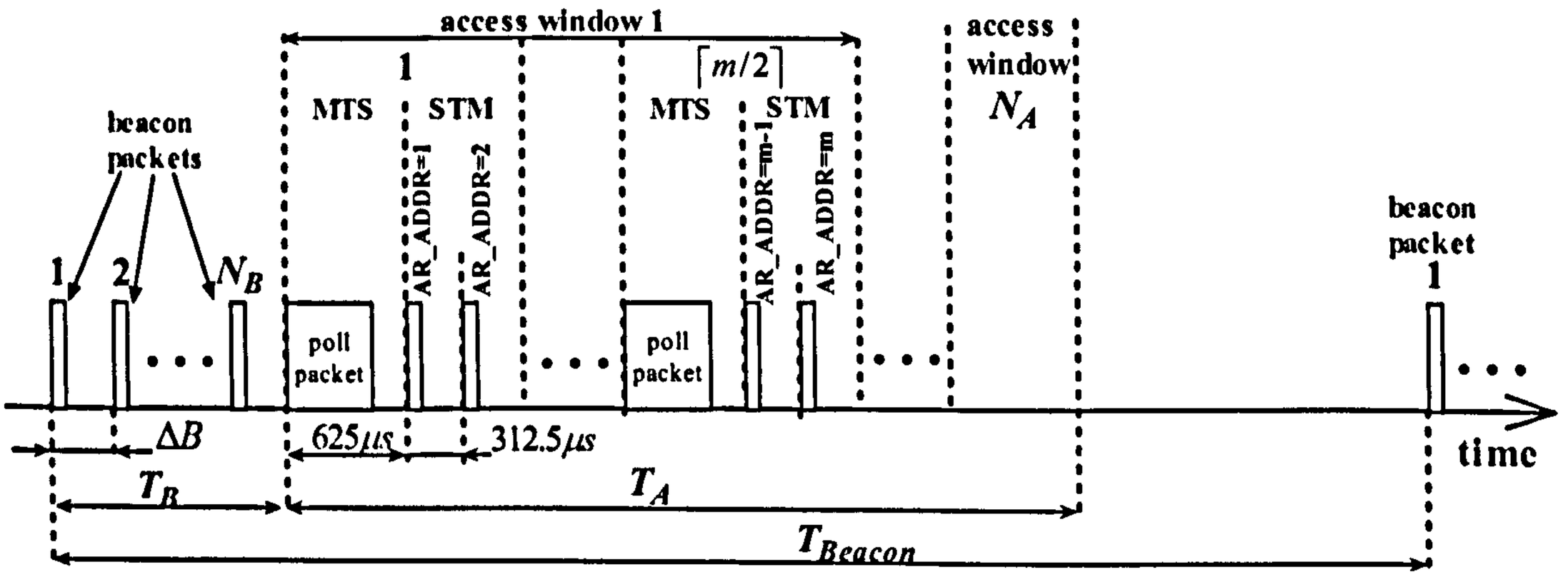


Figure 7.10 beacon train and access window

As shown in Figure 7.10, the time to transmit a beacon train is given by:

$$T_B = \delta \times \Delta B \times N_B \tag{7.27}$$

The time spent on the access windows for a time span of T_{Beacon} is:

$$T_A = 2\delta \times \left\lceil \frac{N_{park}}{2} \right\rceil \times N_A \tag{7.28}$$

The probability for a time slot that is either transmitting Beacon trains or in the duration of access windows is:

$$P_{Park} = \frac{T_A + T_B}{T_{Beacon}} \tag{7.29}$$

P_{start} is identified by substituting (7.26) and (7.29) into (7.25). The average number of required time slots is also derived for each ACL Baseband packet transmission S_t in equation (7.16). Finally, the average ACL Baseband packet delay D_{BS} is identified by substituting (7.16) into (7.14):

$$D_{BS} = \frac{3\delta a N_{active}(1-p^{N+1})}{(1-p)(1-P_{SCO}-\frac{T_A+T_B}{T_{Beacon}})} \tag{7.30}$$

7.6.2 Average L2CAP PDU delay

By replacing T_{BS} as D_{BS} in equations (7.8)-(7.11), with consideration of all the possible piconet activities, the times to successfully transmit a L2CAP PDU with 1, 2, 3, . . . , n Baseband packets are:

$$D(1) = D_{BS} + p^{N+1}D(1) = \frac{D_{BS}}{1-p^{N+1}} \tag{7.31}$$

$$D(2) = D(1) + D_{BS} + p^{N+1}D(2) = \frac{D(1)+D_{BS}}{1-p^{N+1}} \tag{7.32}$$

$$D(3) = D(2) + D_{BS} + p^{N+1}D(3) = \frac{D(2)+D_{BS}}{1-p^{N+1}} \tag{7.33}$$

...

$$D(n) = D(n-1) + D_{BS} + p^{N+1}T(n) = \frac{D(n-1)+D_{BS}}{1-p^{N+1}} \tag{7.34}$$

Similar to equation (7.12), for a L2CAP PDU that is fragmented into M Baseband packets, the average delay to successfully transmit the L2CAP PDU is:

$$D_{L2CAP}(M) = D(M) + T_{ack} + (D(M)+T_{ack})Mp_{un} = (D(M)+T_{ack})(1+Mp_{un}) \tag{7.35}$$

7.7 Delay Result Analysis

In this section, the delays of both Baseband packets and L2CAP PDU are examined by the analytical results. In Figure 7.11 and Figure 7.12, the delay of six types of ACL baseband packets is plotted against the BER in the range of 10^{-6} to 10^{-2}. $N_{active}=2$ and $N_{park}=2$ are used in Figure 7.11, while $N_{active}=7$ and $N_{park}=15$ are used in Figure 7.12. t_{TO}=1s and S_{syn}=60bits are used in both figures. In the piconet physical channel, a slave may lose synchronization if it does not receive a packet from the master at least every 250 ms. In this

work the value of T_{Beacon} is set to 250 ms. The following parameters are used to calculate P_{park}: $\Delta B = 2$, $N_B = 3$, $N_A = 2$. To calculate P_{SCO}, one SCO link $N_{SCO} = 1$ and the packet type of HV3 are considered. Unless otherwise specified, t_{TO}=1s, S_{syn}=60bits and the same parameters for calculating P_{park} and P_{SCO} are used for all the results in this section.

As shown in Figure 7.11, the delay of Baseband packets stays the same for the low BER and increases significantly with the BER when BER>10^{-4}. Due to the frequent retransmission, the increment of the DH packets' delay is more significant than for the DM packets. DH3 and DH5 reach the Flush Timeout (1s) constantly when BER>10^{-3}. In Figure 7.12, a congested piconet scenario is examined (7 active and 15 parked slaves). The delay for Baseband packets is much longer compared to the less congested scenario in Figure 7.11 at the same BER. Similarly to Figure 7.11, the delay increases significantly at the high BER especially for the DH packets.

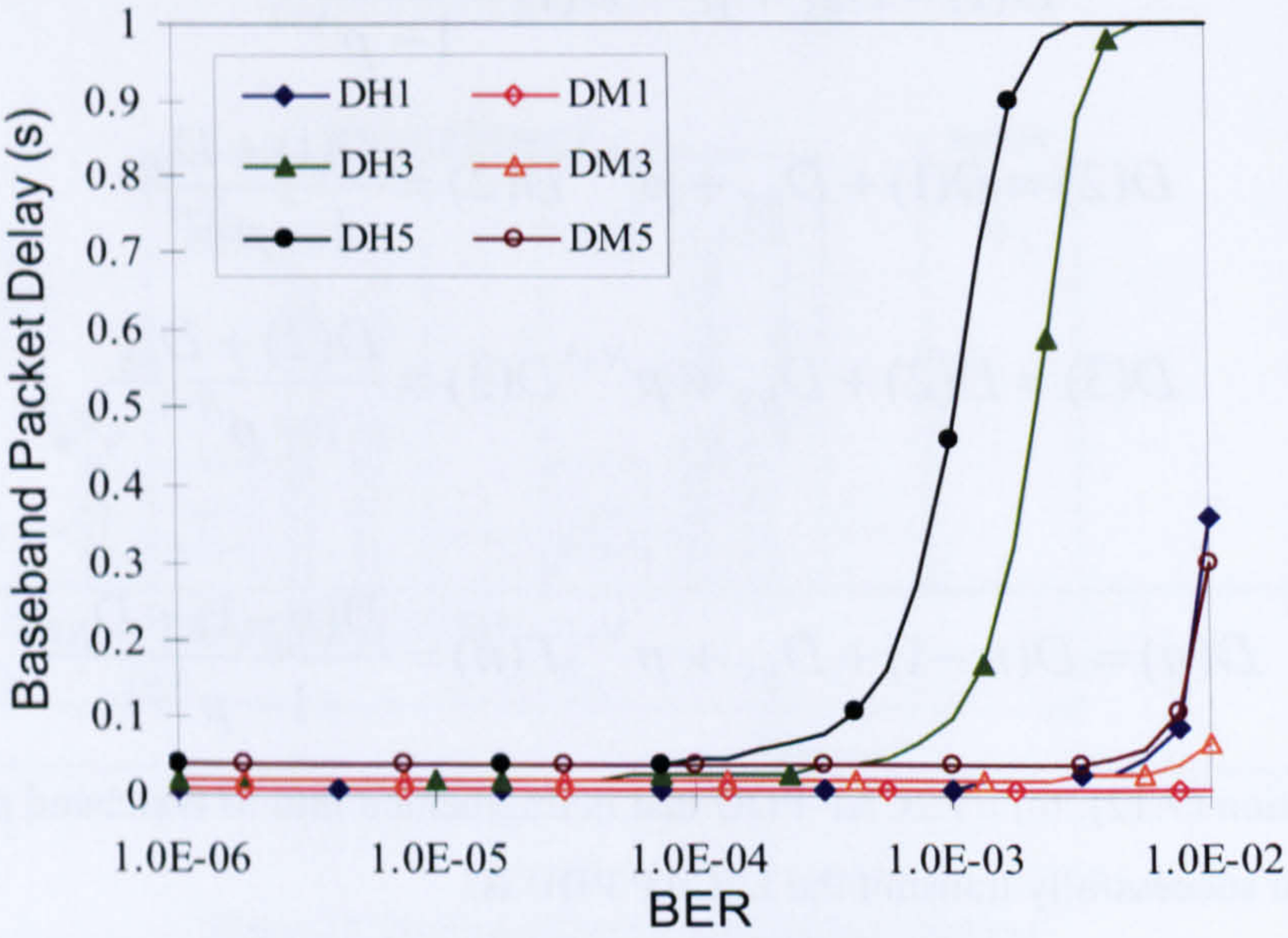


Figure 7.11 average ACL baseband packet delay against BER. N_{active}=2 and N_{park}=2

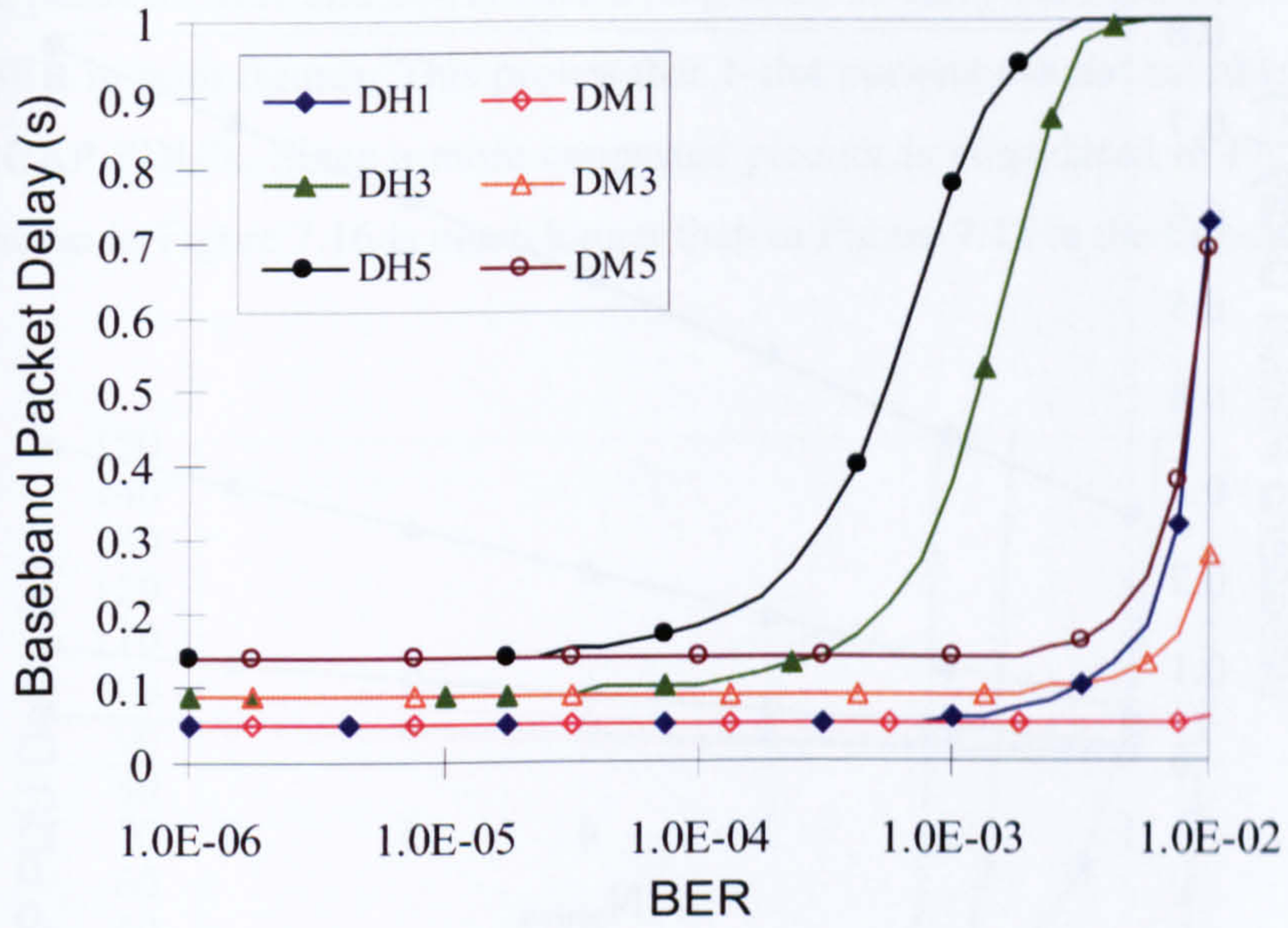


Figure 7.12 average ACL baseband packet delay against BER. N_{active}=7 and N_{park}=10

The delay of Baseband packets is plotted against the number of active slaves (up to 7). Different BER values of 10^{-6} and 10^{-2} are used in Figure 7.13 and Figure 7.14 respectively. Two parked slaves are considered ($N_{park} = 2$).

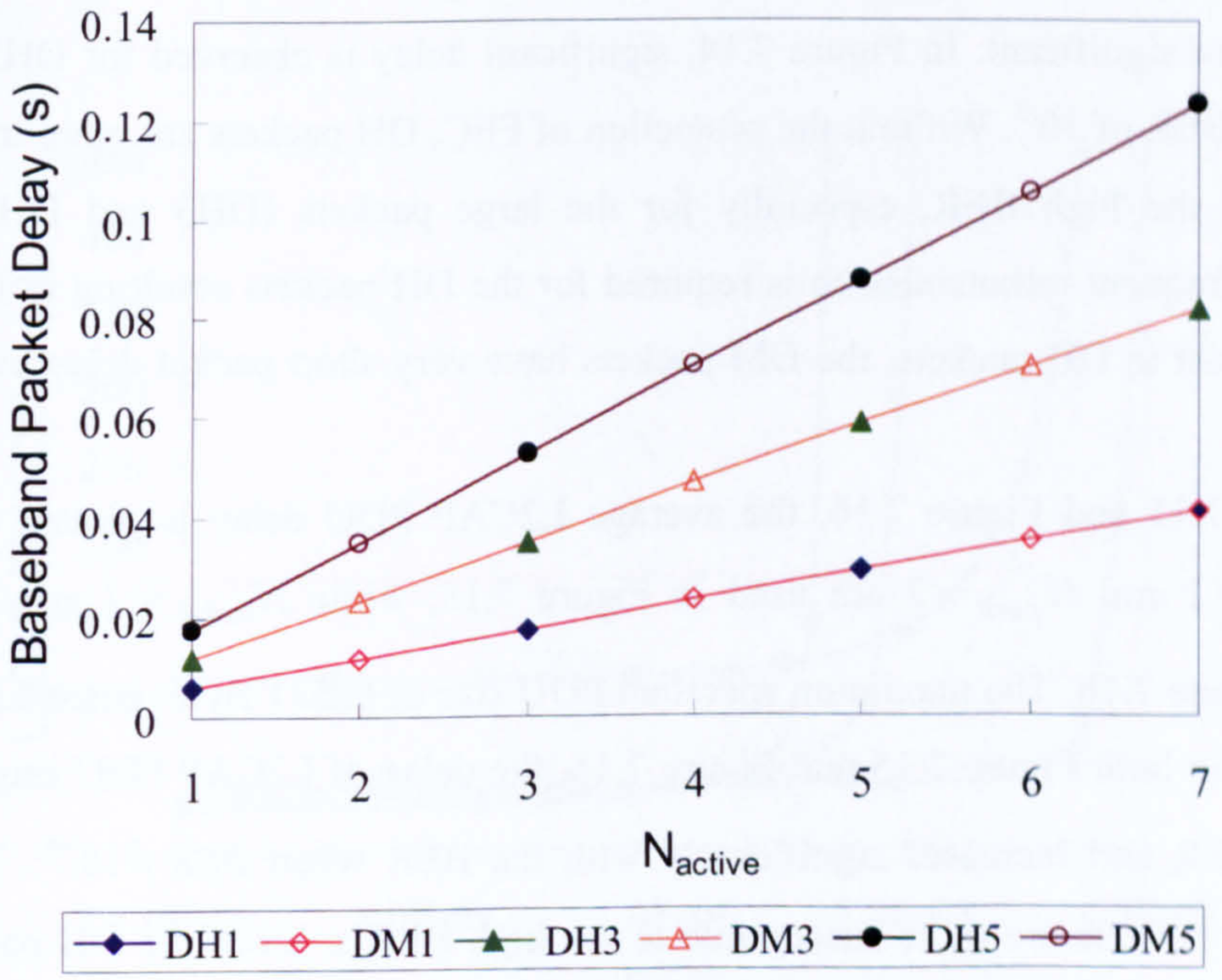


Figure 7.13 average ACL baseband packet delay against N_{active}. BER=10^{-6} and N_{park}=2

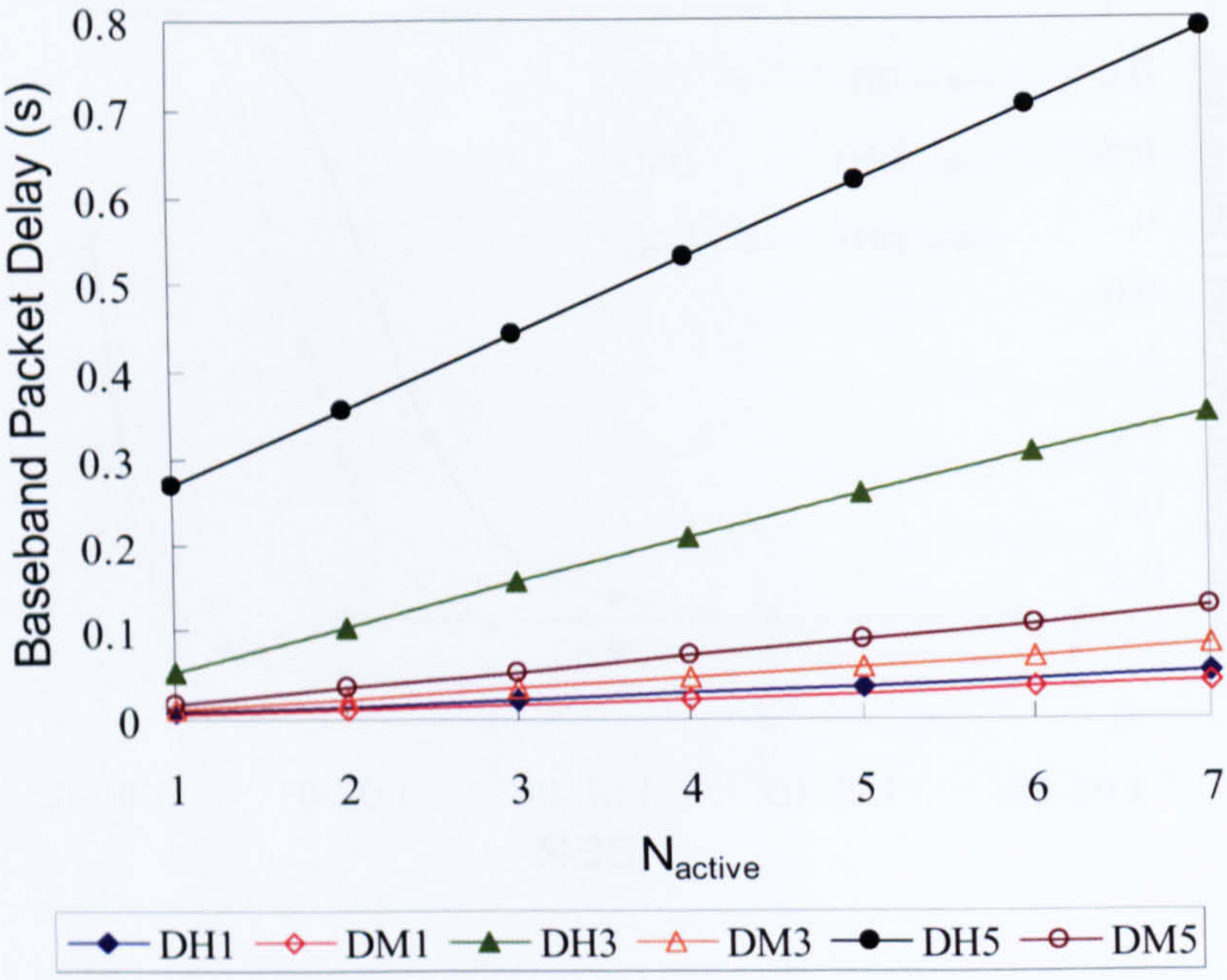


Figure 7.14 average ACL baseband packet delay against N_{active}. BER=10^{-3} and N_{park}=2

As shown in the result figures, the delay of Baseband packets increases linearly with the number of active slaves. In Figure 7.13, there is no difference in delay between the DH and DM packets for the same slot packets. Since low BER (BER=10^{-6}) is used, the Baseband packet error rate is low even for the unprotected DH packets and the benefit of the FEC protection is not significant. In Figure 7.14, significant delay is observed for DH3 and DH5 packets at the BER of 10^{-3}. Without the protection of FEC, DH packets are more transmission error prone at the high BER, especially for the large packets (DH3 and DH5). As the consequence, frequent retransmission is required for the DH packets resulting in long packet delay. In contrast to DH packets, the DM packets have very short packet delay even at BER as high as 10^{-3}.

In Figure 7.15 and Figure 7.16, the average L2CAP PDU delay is plotted against the BER. $N_{active} = 2$ and $N_{park} = 2$ are used in Figure 7.15, while $N_{active} = 7$ and $N_{park} = 15$ are used in Figure 7.16. The maximum specified PDU size of 65531 Bytes is used in all cases.

As shown in both Figure 7.15 and Figure 7.16, the delay of L2CAP PDU stays the same for the low BER and increases significantly with the BER when $BER > 10^{-4}$. The PDU's carried by DH packets are more susceptible to the high BER's. Since the DH packets reach the Flush timeout at high BER's, as shown in Figure 7.11 and Figure 7.12, the DH packets are frequently dropped by Baseband and L2CAP has to retransmit the PDU repeatedly. Therefore, significant delay is observed for the DH packets at high BER's. Moreover, the 1-

slot Baseband packets (DH1 and DM1) have a long delay to carry the PDU of 64 Kbytes even for the low BER in both figures. This proves that 1-slot packets are not suitable for carrying large size L2CAP PDU's. Since a more congested piconet is considered in Figure 7.16, the PDU delay shown in Figure 7.16 is much longer than in Figure 7.15 at the same BER.

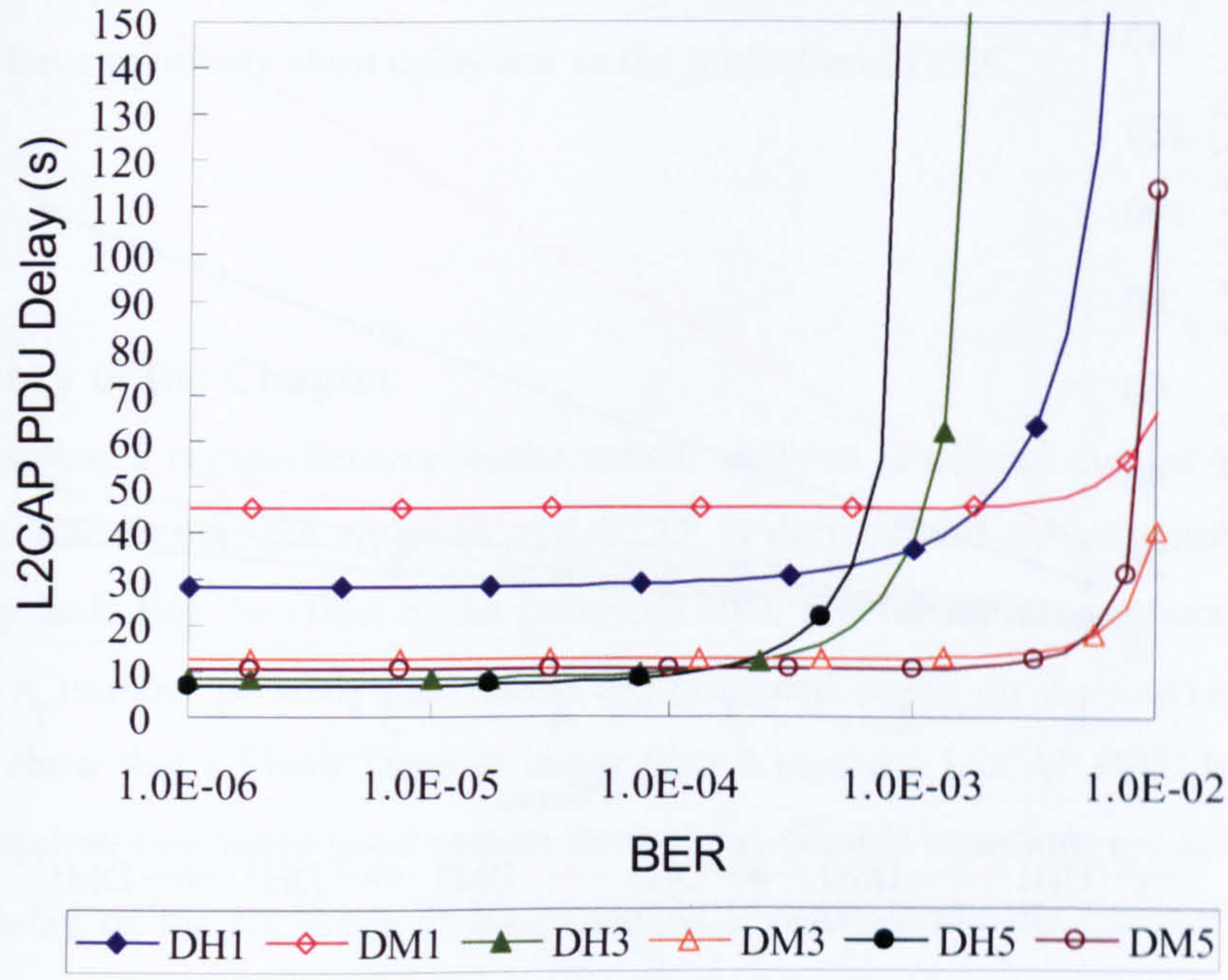


Figure 7.15 average L2CAP PDU delay against BER. N_{active}=2 and N_{park}=2.

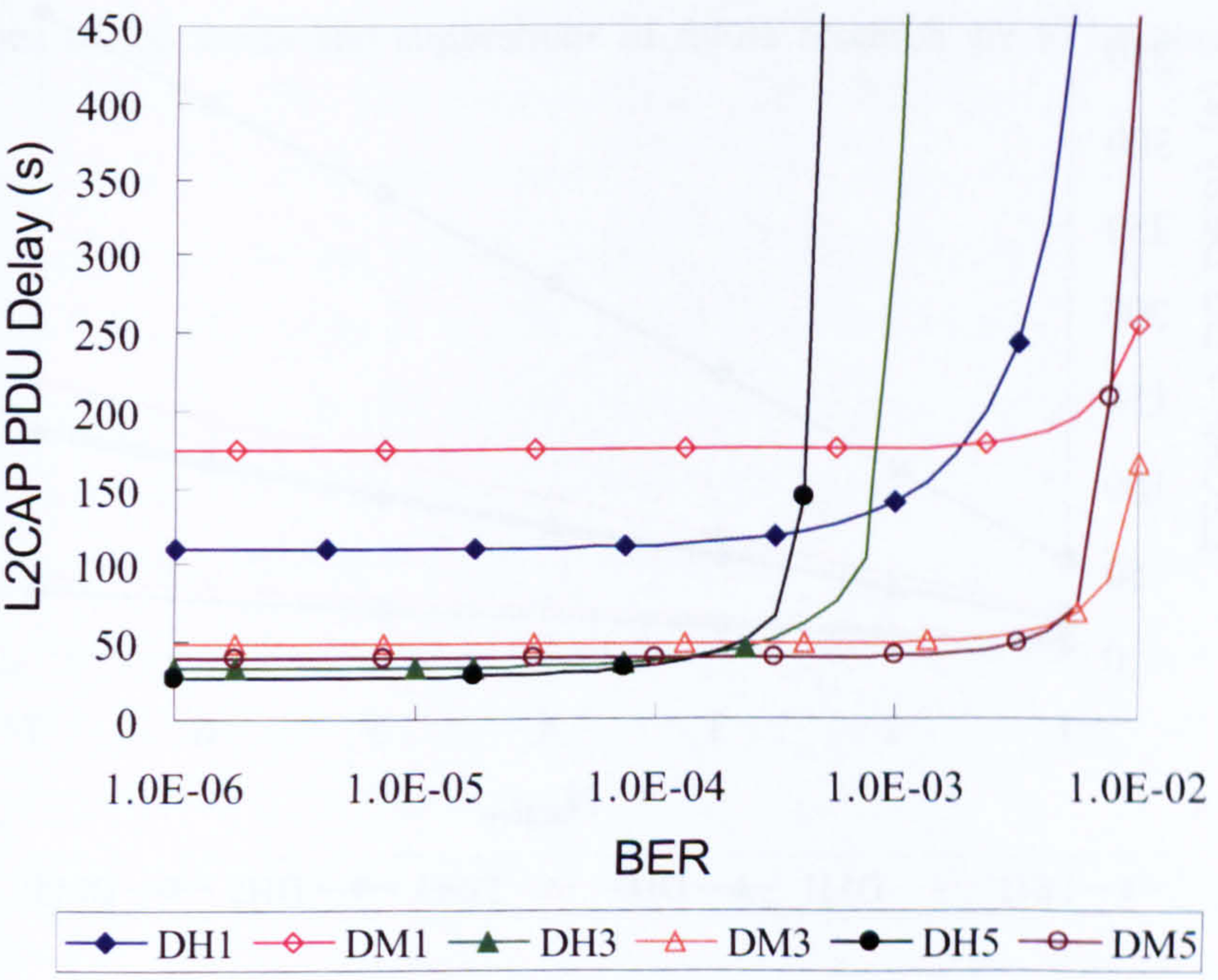


Figure 7.16 average L2CAP PDU delay against BER. N_{active}=7 and N_{park}=15.

The average delay of L2CAP PDU is plotted against the number of active slaves. Different BER values of 10^{-6} and 10^{-2} are used in Figure 7.17 and Figure 7.18 respectively. PDU size of 65531 Bytes and two parked slaves are considered ($N_{park} = 2$).

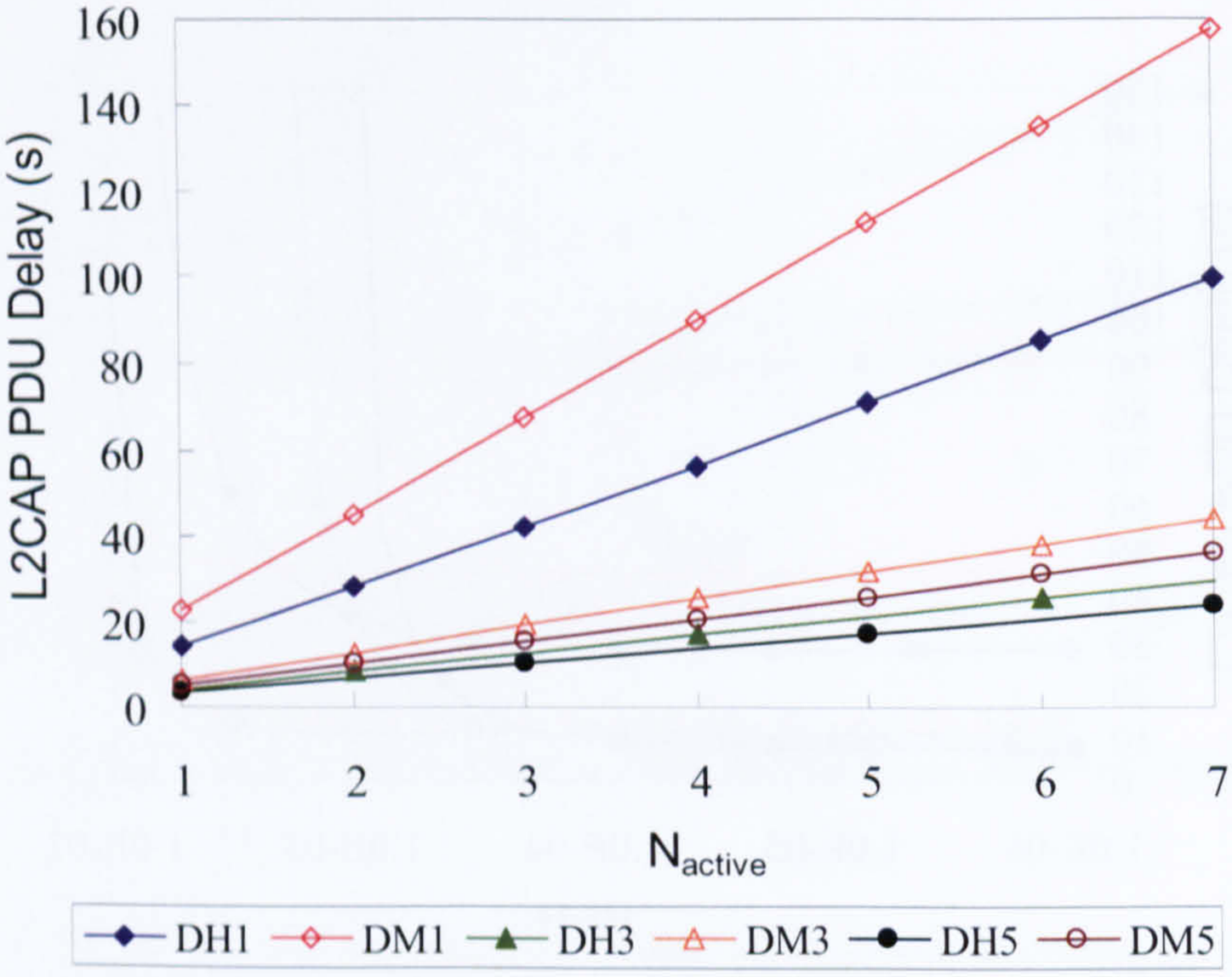


Figure 7.17 average L2CAP PDU delay against N_{active}. BER=10^{-6} and N_{park}=2.

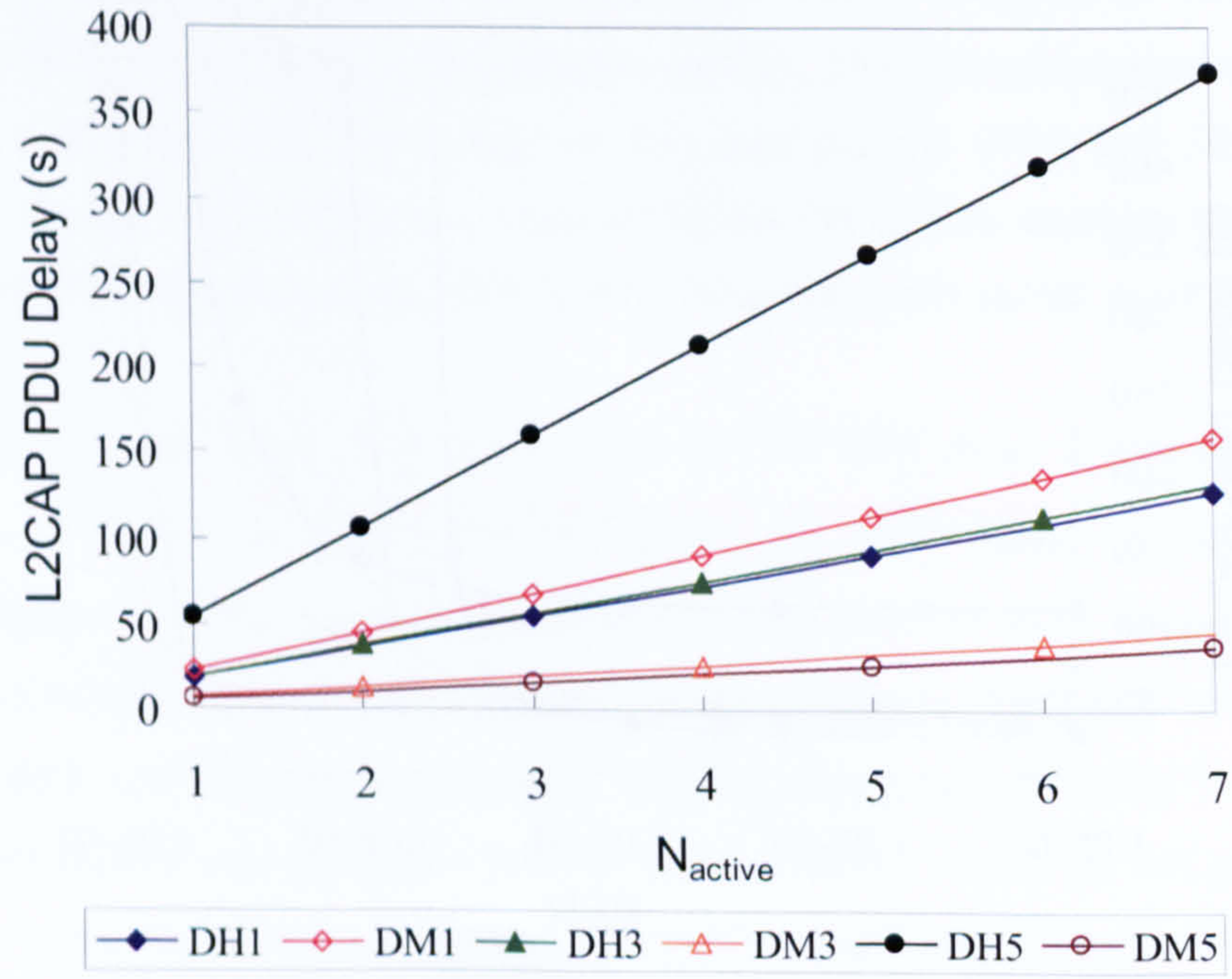


Figure 7.18 average L2CAP PDU delay against N_{active}. BER=10^{-3} and N_{park}=2.

The delay of L2CAP PDU increases linearly with the number of active slaves in both figures. In Figure 7.17, DH packets have longer delay than DM packets. Since the Baseband packet error probability is low when BER=10^{-6}, the benefit of having FEC for the DM packet is not significant. Compared to DM packets, without the FEC overhead, DH packets can carry more data per packet with better efficiency. As shown in Figure 7.18, the DH packets have a much longer delay compared to Figure 7.17 since higher BER of 10^{-3} is used. The DM packets however still have relatively short delay due to the protection of FEC.

7.8 Summary of the Chapter

In this chapter, a comprehensive mathematical analysis is carried out for the Bluetooth L2CAP layer. The throughput equation of L2CAP is derived and subsequently verified by simulations by including the effect of the Baseband layer and taking into account the presence of bit errors. A number of analytical results are produced based on the derived throughput equation and show that a Flush Timeout larger than 0.1s and a L2CAP PDU larger than 10 KBytes are required to achieve good system throughput. Simple equations are also derived for the average delay of the six types of Baseband ACL packets and the L2CAP PDU's. The effects of different network parameters on the delays of the Baseband packets and the L2CAP PDU are examined. The results show that the DH Baseband packets have relatively short delay in the low BER compared to the DM Baseband packets, while the DM packets have a much shorter delay than the DH packets in the high BER environments.

Conclusions of the thesis and suggestions of future research are to be presented in the next chapter.

Chapter 8. Conclusions and Suggestions for Future Research

8.1 Summary of the thesis

This work focuses on the link layer design of the outdoor (FSO) and the indoor (IrDA) infrared systems, as well as the performance of the higher layers of two major WPAN technologies: IrDA and Bluetooth. A number of mathematical models are developed and verified by simulations for different communication layers of FSO, IrDA and Bluetooth. Based on the derived mathematical models, the system performance at different layers is examined and optimized for different bit error rate environments. Several implementation issues for the higher layers of IrDA and Bluetooth have been discussed. In addition, a number of design guidelines are proposed. The following protocol stack layers have been studied in the thesis:

- ARM mode of HDLC as the full-duplex FSO link layer protocol
- IrDA Link Access Protocol (IrLAP)
- IrDA Tiny Transport protocol (TinyTP)
- Object Exchange protocol (OBEX)
- IrDA Burst protocol (IrBurst)
- IrDA Financial Messaging profile (IrFM)
- IrDA Simple Connection profile (IrSC)
- Bluetooth Logical Link Control and Adaptation Layer Protocol (L2CAP)

The main contributions of the thesis are:

- The derivation of the mathematical equation which allows calculation of the FSO link layer throughput. The performance of the FSO link layer is also improved by optimizing the link layer parameters.
- The derivation of the throughput equations for different IrDA protocol stack layers (from the data link layer to the session layer) by considering a variety of different performance metrics, as well as the derivation of the optimum equations for various protocol parameters to maximize the system performance.
- A number of implementation issues on the higher layers (OBEX, IrBurst, IrFM and IrSC) are studied and analysed. Based on the analysis, a set of design guidelines is proposed to maximize the overall system performance as well as keeping the hardware requirement to the minimum.
- Derivation of equations to calculate the L2CAP throughput, the Baseband packet delay and the L2CAP PDU delay.

The thesis provides a comprehensive analysis to the protocol stacks of IrDA and Bluetooth. As shown in the analytical results, the system performance, including throughput and packet

delay, can be improved significantly by adopting the optimum protocol parameters and choosing the appropriate hardware. This work is very benefitial for both researchers and manufacturers who work in the field of WPAN. The developed and verified the mathematical models for different layers of IrDA and Bluetooth can serve as the mathematical framework. Thus, other researchers who are interested in carrying out research in the area of WPAN can make use of and extend the mathematical models to examine a perticular system aspect from their own angles. As for the manufacturer, the product developers can design the IrDA and Bluetooth chipsets according to the design guidelines provided in this thesis and optimize the performance for different applications.

8.2 Conclusions for the FSO link layer analysis

A mathematical model for the full-duplex FSO links is presented by using the ARM mode of HDLC as the link layer protocol. Based on the model, optimum window and frame sizes are obtained. The analytical results show that significant improvement on link throughput can be achieved by utilizing the optimized parameters. Furthermore, a simple and efficient adaptive algorithm has been developed to implement the optimum parameters in the real time.

The protocol component tasks that affect link performance are examined. The analytical results show that the time interval (T_{WS}) spent on waiting for S-frames from the receiver has a significant effect on the throughput for the long distance links due to long propagation delays. The results also show that the propagation delay has a significant effect on the link performance, and the effect is likely to be even more severe if higher data rates are applied in the future. However, by adopting the optimum values, the FSO system can still achieve high link throughput even if it operates in a high bit error rate environment.

8.3 Conclusions for the IrDA protocol stacks analysis

8.3.1 Data link layer—IrLAP

The performance of IrDA IrLAP protocol operating at Gbit/s data rates is examined. A systematic analysis of IrLAP protocol has been carried out by including the effect of link propagation time. Bit error rate (BER) dependent optimization equations of the IrLAP window and frame sizes are derived and also verified for the maximum link layer throughput.

The effects of propagation delay and minimum turnaround time on the link throughput at different data rate has been studied by using both optimum and non-optimum IrLAP window and frame size for comparison. The analytical results show that the throughput efficiency is much higher when the optimum link parameters are implemented. The effect of propagation delay (t_p) becomes significant when the IrLAP minimum turn around time (t_{ta}) has a small

value ($t_{ta} \leq 10^{7} s$) at very high data rates ($C \geq 10 Gbit/s$). To achieve good throughput efficiency, t_{ta} of 10^{-8}s should be used when the link data rate is 10 Gbit/s. Furthermore, different protocol tasks of IrLAP have been studied at the data rate of 10 Gbit/s. The results indicate that propagation delay, even for the distance of 2m, does not significantly reduce the throughput until the data rate reaches 10 Gbit/s for the indoor short distance IrDA links.

From the analytical results, one can conclude that the IrLAP protocol is not a limitation even for the data rates as high as 10 Gbit/s if a small enough minimum turnaround time ($t_{ta} = 10^{-8} s$) is used and the optimum IrLAP window and frame sizes are implemented.

8.3.2 Transport layer—TinyTP

A comprehensive mathematical model for the IrDA TinyTP protocol is developed by including the effects of the multiple IrLMP connections and the underlying IrDA protocol stacks. In order to examine the accuracy of the derived mathematical model, a set of simulation programs based on MATLAB™ is developed. By using the simulation programs, the mathematical model is subsequently verified. Analytical results are produced to compare the throughput efficiency by implementing different receiver window size, IrLAP window and frame sizes. The results show that the system always achieves its best performance when the receiver window size is at least twice as large as the IrLAP window size ($Bw \geq 2N_{LAP}$). Good TinyTP throughput is also obtained when the receiver window size is larger than the IrLAP window size but smaller than two times the IrLAP window size ($N_{LAP} < Bw < 2N_{LAP}$). For the resource limited devices, in order to improve system performance and resource requirement, TinyTP can use a receiver window size in the range of $N_{LAP} < Bw < 2N_{LAP}$ since this range achieves good throughput as well as requiring relatively small buffer size.

In order to optimize the system performance, several optimization equations for the major protocol parameters, including the receiver window size, IrLAP window and frame sizes, are derived as a function of bit error rate and later validated by the exact numerical method. By comparing the analytical results, one can see that significant improvements in the throughput can be achieved by implementing the optimized protocol parameters. Finally, the effect of the IrLAP minimum turnaround time on the TinyTP throughput is examined. An IrLAP turnaround time in the order of 10^{-4}s is suitable for the 16 Mbit/s IrDA links.

8.3.3 Session layer—OBEX and IrBurst

In order to optimize the overall performance of a system, it is also important to make sure the upper layer (i.e. session layer) is efficient and compatible with the lower layer. In this work, the performance of the two session protocols of IrDA, OBEX and IrBurst, is examined.

OBEX has been widely implemented in the wireless object exchange applications for both IrDA and Bluetooth communications. The thesis examines the performance of OBEX protocol, investigates the interaction between OBEX and the lower IrDA protocol stacks, and also optimizes different parameters to achieve the maximum system throughput.

An analytical model is developed to derive the OBEX throughput equation. Based on the model, the effects of the OBEX packet size, and the IrLAP window and frame sizes on OBEX throughput is investigated. The system has a very poor performance at the high BER with the non-optimized protocol parameters.

In order to maximize the throughout and improve the system performance at the high BER, an optimization study is carried out for the OBEX size and the IrLAP parameters. The study concludes that the OBEX throughput always benefits by using a large OBEX size, IrLAP frame size should be reduced gradually with the increasing BER, and the optimum IrLAP window size is independent of BER until $BER > 2.6 \times 10^{-7}$. The optimization equations for the IrLAP window and frame sizes are derived and then verified by the exact numerical method. The analytical results show that the significant improvements on the throughput can be achieved by applying the optimized parameters.

To further improve the OBEX performance, a study of the effects of the maximum OBEX packet size (P_{REQ}) and the OBEX turnaround time (T_{OBEX}) is carried out. Based on the analytical results, the conclusion is given as follows: for the 100 Mbit/s links, P_{REQ} of 1 Mbits is required to achieve good throughput efficiency. P_{REQ} of 500 Kbits and 100 Kbits are sufficient for 16 Mbit/s and 115.2 Kbit/s respectively. For the 100 Mbit/s links, T_{OBEX} of less than 10^{-4} s is necessary, while for 16 Mbit/s or less, an OBEX turnaround time of 10^{-3} s is sufficient to achieve good throughput efficiency.

The other IrDA session protocol, IrBurst, is also investigated. IrBurst recently has been proposed by IrDA in order to provide a complementary capability to OBEX. It is designed to fulfil the requirements for transmitting bursts (low delay) of information data. The analysis of IrBurst shows that the derived mathematical model for TinyTP is also applicable for IrBurst. Subsequently, the performance of IrBurst and OBEX are compared when the transmission of large size objects is considered. The analytical results are produced for both 16 Mbit/s and 100 Mbit/s links. IrBurst shows an excellent performance compared to that of OBEX when transmitting large size files. For instance, when the data rate is 100 Mbit/s, IrBurst takes less than 0.5 second to transmit a 4 MBytes file, which is the typical size of an MP3 song.

8.3.4 IrDA applications—IrFM and IrSC

Two rapidly developing IrDA applications, IrFM and IrSC, are studied in this work.

In order to improve efficiency of payment process, to integrate various incompatible financial transaction schemes and eliminate cumbersome safekeeping of paper receipts, a novel approach to replace the traditional payment is essential. Two factors, the maturity of personal digital devices and the awareness and capability of consumers using such devices, establish the environment for a digital payment system.

IrFM, proposed by IrDA, is one of the most promising candidates for the digital payment system. The thesis carries out an inclusive analysis of IrFM. Several design guidelines are proposed for the link data rate, the IrLAP minimum turnaround time and the XID sending interval to achieve fast IrDA connections while keeping the hardware requirements at the minimum. Other important issues in designing the digital payment system are carefully examined for the IrFM system including security, reliability, simplicity and interoperability. Additionally, the current market status of IrFM is presented and the competing digital payment technologies are compared with IrFM. The study shows that IrFM is an excellent approach that addresses almost every aspect of the requirements of a desirable digital payment system.

IrSC is a protocol dedicated for fast connection and instant data transmission between IrDA devices and is still under development by an IrDA SIG. A typical IrSC application is to download a picture from a mobile phone and display it on a TV screen. In the analysis, two technical approaches for IrSC (IrSC1 and IrSC2) are investigated.

Analytical equations for calculating the transmission delay for the unchanged IrDA standard, IrSC1 and IrSC2 are derived. Based on the equations, the transmission delay of different approaches is examined and compared. The analytical results show that the SIR (less than 576 Kbit/s) is not a good option for IrSC and 4Mbit/s or higher is required. IrSC2 is the fastest and cheapest option but it is likely to be low quality due to lack of error correction, and could cause annoyance to users. However, for low capacity devices (where only a small buffer is available), IrSC2 can still maintain a very high performance by bypassing the negative effect of the IrLAP turnarounds. IrSC1 has a good performance for IrLAP frame size of 16 Kbits and window size of 7. Reducing IrLAP frame size to 1 Kbits and window to 1 (cheaper to implement) makes it unworkable for transmitting pictures.

8.4 Conclusions for the Bluetooth L2CAP layer analysis

Bluetooth L2CAP layer bridges the data applications and the Bluetooth Baseband layer by providing services including channel multiplexing, packet segmentation and reassembly, and error retransmission. In order to have a good overall performance for the Bluetooth system, it is important to optimize the performance of L2CAP. A systematic analysis for L2CAP is carried out in the analysis. The throughput equation of L2CAP is derived by

considering the effect of the Baseband layer and taking into account the presence of bit errors. The mathematical model is then verified by the developed MATLAB™ simulation programs. The effects of the Baseband Flush Timeout and the L2CAP PDU size are then investigated. The analytical results show that a Flush Timeout larger than 0.1s and a L2CAP PDU larger than 10 KBytes are required to achieve good system throughput.

This work also carries out a delay analysis for the Baseband ACL packets and the L2CAP PDU's by considering all the possible Bluetooth network activities. By including the presence of bit errors, simple equations are derived for calculating the average ACL packet and L2CAP PDU delays. Based on the delay equations, the effects of different network parameters on the delays of the six Baseband packets and the L2CAP PDU are examined. The results show that the DH Baseband packets have relatively short delay at low BER compared to the DM Baseband packets. However, the DH packets are very susceptible to BER and the delay of the DH packets increases significantly with BER when $BER > 10^{-3}$. With the protection of FEC, the DM packets have a much shorter delay than the DH packets in the high BER environments.

8.5 Suggestions for future research

To investigate the 100 Mbit/s IrDA standard: IrDA recently started the development of a new IrDA physical layer specification (UFIR) that can support data rate up to 100 Mbit/s. For such a high speed infrared link, a new modulation scheme and possibly a new error detection scheme are required. In order to develop a stable and easy to implement high speed IrDA physical layer, it is essential to have a suitable modulation and error detection scheme. Therefore, to investigate the performance and the stability of the 100 Mbit/s IrDA standard should be one of the future research topics in the area of IrDA communication.

To provide the "zero delay" service: With the upcoming UFIR standard, the IrDA data rate is to be increased significantly in the near future. Small size objects, such as IrFM instructions and business cards, can be transmitted almost instantly through a connected high speed IrDA link. It is advantageous to be able to provide near instant data delivery, known as "zero delay", for time sensitive applications. The bottleneck for providing "zero delay" would be the connection delay. In order to reduce the connection delay, a modification to the existing IrLAP standard is required.

To investigate the OBEX performance on the Bluetooth protocol stacks: OBEX is adopted as the object exchange framework for not only IrDA but also Bluetooth

communications. For the Bluetooth protocol stacks, OBEX is layered above the Radio Frequency Communications protocol (RFCOMM), which emulates the serial ports over the L2CAP protocol. It is of interest to examine the performance of OBEX over the Bluetooth protocol stacks. The analysis of OBEX can be carried out based on the derived and verified mathematical model of L2CAP presented in the thesis.

References

[1] Ramiro J. and Abdallah C.T., "Wireless Communications and Networking: An Overview", *IEEE Antenna's and Propagation Magazine*, 2002, 44(1), pp. 185-193.

[2] Desimone A. and Nanda S., "Wireless Data: Systems, Standards, Services", *ACM-Baltzer Journal of Wireless Networks*, 1995, 1(3), pp. 241-253.

[3] Kahn J. M. and Barry J. R., "Wireless infrared communications", *Proceedings of IEEE*, 1997, vol. 85, pp. 265-298.

[4] Bantz D. F. and Bauchot F. J., "Wireless LAN design alternatives", *IEEE Network*, 1994, vol. 8, no. 2, pp. 43-53.

[5] Barry J. R., "Wireless Infrared Communications", *Kluwer Academic Publishers*, 1994.

[6] Boucouvalas A. C., "IEC 825-1 Eye Safety Classification of some consumer electronic products", *Proceedings of IEE Colloquium on Free-Space Communication Links*, 1996, no. 13, pp.1-6.

[7] IEEE Standard 802.15.1, 2003.

[8] Heile R.F., "Solutions for the last 10 meters: an overview of IEEE 802.15 working group on WPANs", *International Symposium on Wearable Computers*, 1999, pp. 10-11.

[9] Heatley D., Wisely D., Neild I. and Cochrane P., "Optical wireless: the story so far", *IEEE communications magazine*, 1998, Vol. 36, No.12, pp.72-82.

[10] Dettmer R.A., "ray of light [free space optical transmission]", *IEE Review*, Volume: 47 Issue: 2, 2001 pp. 32 -33.

[11] Williams S., "IrDA: Past, Present and Future", *IEEE Personal Communications*, 2000, 7(1), pp. 11-19.

[12] Willebrand H. A. and Ghuman B. S., "Fiber optics without fiber", *IEEE Spectrum*, 2001, pp.40.

[13] Davis, C.C., Smolyaninov, I.I., Milner, S.D. "Flexible optical wireless links and networks", *IEEE Communications Magazine*, 2003, Volume: 41 Issue: 3, pp. 51 -57.

[14] http://www.free-space-optics.org/

[15] McDermott-Wells, P., "What is Bluetooth?", *IEEE Potentials*, Volume 23, Issue 5, Dec. 2004-Jan. 2005 pp. 33 – 35.

[16] Tanenbaum A.S., "Computer Networks", Prentice Hall, 1996.

[17] Pahlavan K., "Trends in Local Wireless Networks", *IEEE Communications Magazine*, 1995, vol. 33, no. 3, pp. 88-95.

[18] Varshney U. and Vetter R., "Emerging mobile and wireless networks", *Communications of ACM*, June 2000, vol. 43, no. 6, pp. 73-81.

[19] Kahn J.M. and Barry J.R., "Wireless infrared communications", *Proceedings of IEEE*,

1997, vol. 85, pp. 265-298.

[20] Carruthers J. B., "Wireless Infrared Communications", Encyclopaedia of Telecommunications, chapter, New York: Wiley, 2002.

[21] Davis, C.C.; Smolyaninov, I.I.; Milner, S.D. "Flexible optical wireless links and networks", *IEEE Communications Magazine*, 2003, Volume: 41 Issue: 3, pp. 51 -57.

[22] Acampora, A.S. and Krishnamurthy, S.V."A broadband wireless access network based on mesh-connected free-space optical links" *IEEE Personal Communications*, 1999, Volume: 6 Issue: 5, pp. 62 –65.

[23] Chinlon Lin, Kung-Li Deng and Chun-Kit Chan "Broadband optical access networks", *The 4th Pacific Rim Conference on Lasers and Electro-Optics, 2001*. CLEO/Pacific Rim 2001, Volume: 2, pp: II-576 -II-577 vol.2

[24] Boucouvalas A. C. "IEC 825-1 Eye Safety Classification of some consumer electronic products", *Proceedings of IEE Colloquium on Free-Space Communication Links*, 1996, London, no. 13, pp.1-6.

[25] IEEE Standard 802.15.4 (ZigBee), 2003.

[26] IEEE Standard 802.11b, 1999.

[27] IEEE Standard 802.11a, 1999.

[28] IEEE Standard 802.11g, 2003.

[29] Qiunn, L. "Wireless Communications Technology Landscape", *Dell's technology white paper*, Feb 2005.

[30] P. Chatzimisios, A. C. Boucouvalas and V. Vitsas, "IEEE 802.11 Wireless LANs: Performance Analysis and Protocol Refinement", *EURASIP Journal on Wireless Communications and Networking* (WCN), volume: 2005:1, pages: 67-78. DOI: 10.1155/WCN.2005.45

[31] Stallings, *W. Data & Computer Communications*, 6th edition. New Jersey: Prentice Hall.

[32] Tanenbaum, A. S., 2002. *Computer Networks*, 4th edition. New Jersey: Prentice Hall.

[33] "Ultrawideband: High-speed, short-range technology with far-reaching effects", *MBOA-SIG White Paper*, Sept 2004

[34] OPNET network simulator, http://www.opnet.com/

[35] NS2 network simulator, http://www.isi.edu/nsnam/ns/

[36] Matlab, http://www.mathworks.com/

[37] IrDA official website, http://www.irda.org/

[38] Bluetooth official website, http://www.bluetooth.org/

[39] IrDA, Serial Infrared Physical Layer Specification (IrPHY), version 1.0, 1994

[40] IrDA, Serial Infrared Physical Layer Specification (IrPHY), version 1.1, 1995

[41] IrDA, Serial Infrared Physical Layer Specification (IrPHY), version 1.4, IrDA, 2001

[42] IrDA, Serial Infrared Link Access Protocol (IrLAP), version 1.1, 1996

[43] IrDA, Link Management Protocol (IrLMP), version 1.1, 1996

[44] IrDA, Tiny Transport Protocol (TinyTP), version 1.1, 1996

[45] IrDA, Object Exchange Protocol (OBEX), version 1.3, March, 2003

[46] IrDA, Infrared Financial Messaging Point and Pay Profile (IrFM), version 1.0, 2003

[47] IrDA, Burst data protocol (IrBurst), version 1.0, 2004

[48] ISO 3309 High Level Data Link Control (HDLC) Procedures, Frame Structure 1991-06-01

[49] "Optical Wireless Solutions for Service Provider Applications", *LightPointe Communications white paper*, 2004.

[50] IEC 825-1 publication, 1993.

[51] Higginbottom G.N., *Performance Evaluation of Communication Networks*, Artech House, 1998.

[52] Gummalla A.C.V. and Limb J.O., "Wireless medium access control protocols", *IEEE Communications Surveys*, Second Quarter 2000.

[53] Haas Z. J., Deng J., Liang B., Papadimitratos P., and Sajama S., "Wireless Ad Hoc Networks", *Encyclopedia of Telecommunications*, John Wiley, 2002.

[54] Stajano F. and Anderson R., "The Resurrecting Duckling: Security Issues for Ad Hoc Wireless Networks", *in Proceedings of 7th International Workshop on Security Protocols, LNCS*, Springer-Verlag, pp. 172 – 194, 1999.

[55] Papadimitratos P. and Haas Z. J., "Secure Data Transmission in Mobile Ad Hoc Networks", *in Proceedings of the Second ACM Workshop on Wireless Security (WiSe'03), in conjunction with ACM MobiCom 2003*, Atlanta, USA, pp. 41-50, 2003.

[56] Zhou L. and Haas Z. J., "Securing Ad Hoc Networks", *IEEE Network Magazine*, vol,13, no.6, pp. 24 – 30, Nov./Dec. 1999.

[57] Vitsas V., "Link Layer Protocol Performance of Indoor Infrared Wireless Communications", *PhD Thesis*, Bournemouth University, 2002.

[58] Barker P., "Analytical and Simulation Performance Modelling of Indoor Infrared Wireless Data Communications Protocols", *PhD Thesis*, Bournemouth University, 2003.

[59] Chatzimisios P., "Performance Modelling and Enhancement of Wireless Communication Protocols", *PhD Thesis*, Bournemouth University, 2004.

[60] Comer D. E., *Computer Networks and Internets with internet applications*, Prentice Hall, 2001.

[61] Moreira A.J.C., Valadas R.T. and Oliveira Duarte A.M., "Performance of infrared transmission systems under ambient light interference", *IEE proceedings of Optoelectronics*, vol. 143, 1996, pp.339-346.

[62] Narashimhan R., Audeh M.D. and Kahn J.M., "Effect of electronic-ballast fluorescent

lighting on wireless infrared links", *IEE proceedings Optoelectronics*, vol.143, 1996, pp.347-354.

[63] Gfeller F.R. and Bapst U., "Wireless in-house data communication via difuse infrared radiation", *Proceedings of IEEE*, 1979, vol.67, pp. 1474-1486.

[64] Wong K.K. and O'Farrell T., "Spread Spectrum techniques for indoor wireless IR communications", *IEEE wireless communications magazine*, Apr. 2003, vol.10 No.2, pp.54-63.

[65] Timothy L. and Grotzinger, P.E., "Atmospheric Effects on FSO Systems", *white paper, 2005*, LaserWireless, Inc. http:\\www.laserwireless.net.

[66] Benelli G., "A Go-Back-N protocol for mobile communications", *IEEE Transactions on Vehicular Technology*, vol. 40, no. 4, pp. 714-720, Nov. 1991.

[67] Lin S., Costello D.J.J. and Miller M.J., "Automatic-repeat-request error-control schemes", *IEEE Communications Magazine*, vol. 22, no. 12, Dec. 1984.

[68] Bux W., Kummerle K. and Truong H.L., "Balanced HDLC procedures: A performance analysis", *IEEE Transactions on Communications*, 1980, vol. 28, no. 11, pp.1889-1898.

[69] Bux W. and Truong H.L., "High level data link control traffic considerations", *9th International. Teletraffic Congress*, Session 17, Torremolinos, Spain, Oct. 18-24, 1979.

[70] Boucouvalas A.C., Barker P., "Asymmetry in optical wireless links", *IEE Proceedings of Optoelectronics*, Vol.147, Aug. 2000, pp.315-321

[71] Vitsas V. and Boucouvalas A. C., "Window and frame size adaptivity for maximum throughput in IrDA links", *Proceedings of The 3rd Electronic Circuits and Systems Conference*, Slovak University of Technology in Bratislava, Bratislava, Slovakia on September 5-7, 2001, pp.147-152.

[72] Vitsas V. and Boucouvalas A.C., "Optimisation of IrDA IrLAP Link Access protocol", *IEEE Transactions on Wireless Communications*, Vol. 2, No. 5, pp.926-938, Sept 2003

[73] Boucouvalas A. C., Vitsas V., "Optimum Window and Frame Size for IrDA Links", *IEE Electronic Letters*, Vol.37, No.3, pp194-196, 1st Feb. 2001.

[74] Ozugur T., Naghshineh M., Kermani P., Copeland J.A., "On the performance of ARQ protocols in infrared networks", *International Journal of Communication Systems 2000*, 13:617–638.

[75] Ozugur T., Naghshineh M., Kermani P., OLsen CM., Rezvani B., Copeland J.A., "ARQ protocol for infrared wireless LANs: packet-level ACK or no-packet-level ACK?", *Proceedings of IEEE ICUPC'98*, Florence, Italy, October 1998; 1235–1239.

[76] Barker P. and Boucouvalas A. C., "Performance analysis of the IrDA protocol in wireless communications", *1st International Symposium on Communication Systems and Digital Signal Processing*, Sheffield, UK, 6-8 April 1998, pp. 6-9.

[77] Barker P. and Boucouvalas A. C., "Performance modeling of the IrDA protocol for

infrared wireless communications", *IEEE Communications magazine*, vol. 36, no. 12, pp. 113-117, Dec. 1998.

[78] Barker P., Boucouvalas A. C. and Vitsas V., "Performance modelling of the IrDA infrared wireless communications protocol", *International Journal of Communication Systems*, vol. 13, pp. 589-604, 2000.

[79] Barker P. and Boucouvalas A.C., "A simulation modeling of IRDA infrared communication protocol", *1st International Symposium on Communication Systems and Digital Signal Processing*, Sheffield, UK, 6-8 April 1998, pp. 2-5.

[80] Robertson M.G., Hansen S.V., Sorenson F.E. and Knutson C.D., "Modeling IrDA performance: the effect of IrLAP negotiation parameters on throughput", *Proc. of the 10th IEEE International Conference on Computer Communications and Networks*, 2001, pp. 122-127, 2001

[81] Ozugur T., Naghshineh M. and Kermani P., "Comparison of go-back-N and selective reject ARQ modes of HDLC over half-duplex and full-duplex IR links and the effects of window size and processor speed in utilization", *Proc of the 9th IEEE International Symposium on Personal, Indoor and Mobile Radio 155 Communications (PIMRC)*, 1998, vol. 2, pp. 708-712.

[82] Boucouvalas A.C. and Barker P., "IrLAP protocol performance analysis of IrDA wireless communications", *IEE Electronics Letters*, vol. 34, no. 25, pp. 2380-2381, 1998.

[83] Mitani, Y., Kagawa, N., Tsutsumoto, K., Takakuwa, Y., "Correlation between scintillation of laser beam and wind noise", *Proceedings of the 41st SICE Annual Conference, SICE 2002.* Volume: 3 , 5-7 Aug. 2002 Page(s): 1881 -1884 vol.3

[84] Bushuev, D., Arnon, S., "Enhanced detector array receiver for optical wireless communication network", *The 22nd Convention of Electrical and Electronics Engineers in Israel*, 2002, Page(s): 178 -180

[85] Zhu, X., Kahn, J.M., "Pairwise codeword error probability for coded free-space optical communication through atmospheric turbulence channels", *IEEE International Conference on Communications, 2001 (ICC 2001).* Volume: 1, 11-14 June 2001 Page(s): 161 -164 vol.1

[86] Donhowe, M.N., Garber, P.L., Theorin, C.R., Casarino, D.A., "Eye-safety optimized launch for parallel fiber optic data links", in *Proceedings of Electronic Components and Technology Conference*, 1999. Page(s): 1122 -1126

[87] Khoo, S.H., Zyambo, E.B., Faulkner, G., O'Brien, D.C., Edwards, D.J., Ghisoni, M., Bengtsson, J., "Eyesafe optical link using a holographic diffuser", *IEE Colloquium on Optical Wireless Communications*, June 1999 Page(s): 3/1 -3/6

[88] Spirou, G., Yavin, I., Weel, M., Mikaelian, T., Vorozcovs, A., Andreyuk, A.,

Kumarakrishnan, A., Battle, P.R., Swanson, R.C., "A high-speed modulated retro-reflector for lasers", in *IEEE Proceedings of Aerospace Conference*, 2002, Volume: 3, Page(s): 3-1481 -3-1486 vol.3

[89] Jinlong Zhang, "Modulation analysis for outdoors applications of optical wireless communications", in *Proceedings of International Conference of Communication Technology*, 2000, (WCC - ICCT 2000). Volume: 2, 2000 Page(s): 1483 -1487 vol.2

[90] Martini, R., Gmachl, C., Falciglia, J., Curti, F.G., Bethea, C.G., Capasso, F., Whittaker, E.A., Paiella, R., Tredicucci, A., Hutchinson, A.L., Sivco, D.L., Cho, A.Y., "High-speed modulation and free-space optical audio/video transmission using quantum cascade lasers", *IEE Electronics Letters*, Volume: 37 Issue: 3, 1 Feb 2001 Page(s): 191 –193

[91] Tourrilhes J., Magalhaes L. and Carter C., "On-Demand TCP: Transparent peer to peer TCP/IP over IrDA", in the *Proceedings of IEEE international conference on communications, 2002, (ICC 2002)*, vol. 5, Pages:3250 - 3258

[92] Deccio, C.T.; Ekstrom, J.; Partridge, D.R.; Tew, K.B.; Knutson, C.D.; "A study of the suitability of IrOBEX for high-speed exchange of large data objects" *Global Telecommunications Conference, 2003*. GLOBECOM '03. IEEE ,Volume: 5 , 1-5 Dec. 2003, Pages:2664 - 2668

[93] V. Vitsas and A.C. Boucouvalas, "Automatic Repeat Request Schemes for Infrared Wireless Communications", *IEE Electronics Letters*, Feb 2002, Vol. 38, No. 5 pp.254-246

[94] Modiano E., "An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols", *Wireless Networks,* 1999, **5**(4), pages 279–286

[95] H.R. Damon, R.J. Brown, L. Faulkner, white paper "Creating an End-To-End Digital Payment System" *IrDA press*, Oct, 1999.

[96] RFID Journal, http://www.rfidjournal.com/

[97] Japanese mobile operator 'NTT DoCoMo', http://www.nttdocomo.com/

[98] IrDA Newsletters, "IrDA insider, December 2004 Edition", IrDA, Dec, 2004

[99] IrFM market report www.actisys.com/Documents/IrFmBackgroundMarket_030213.pdf

[100] LG TeleCom, http://www.lgtelecom.com/

[101] Korea mobile operator KTF, http://www.ktf.com/

[102] International Telecommunication Union, "ITU-T Recommendation V.250"

[103] ETSI, TS 101 369 (GSM 07.10) version 6.1.0

[104] Bluetooth Core Specification, Version 1.2, *Bluetooth SIG*, Nov, 2003.

[105] Knutson, C.D.; Hall, E.; Vawdrey, D.; "Bluetooth [wireless connectivity]", *IEEE Potentials*, Oct.-Nov. 2002, Volume: 21 , Issue: 4 , Pages: 28 – 31

[106] Sairam, K.V.S.S.S.S.; Gunasekaran, N.; Redd, S.R.; "Bluetooth in wireless

communication", *IEEE Communications Magazine*, Jun 2002, Volume: 40 Issue: 6, Pages: 90-96

[107] Pasolini, G.; "Analytical investigation on the coexistence of Bluetooth piconets", *IEEE Communications Letters*, March 2004, Volume: 8, Issue: 3, Pages: 144 – 146.

[108] Howitt, I.; "Bluetooth performance in the presence of 802.11b WLAN", *IEEE Transactions on Vehicular Technology*, Nov. 2002. Volume: 51, Issue: 6, Pages: 1640-1651.

[109] Arumugam, A.K.; Doufexi, A.; Nix, A.R.; Fletcher, P.N.; "An investigation of the coexistence of 802.11g WLAN and high data rate Bluetooth enabled consumer electronic devices in indoor home and office environments", *IEEE Transactions on Consumer Electronics*, Aug. 2003. Volume: 49, Issue: 3, Pages: 587-596.

[110] Wang Feng; Arumugam, N.; Krishna, G.H.; "Performance of a Bluetooth piconet in the presence of IEEE 802.11 WLANs", *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2002.* 15-18 Sept. 2002. Pages: 1742-1746 vol.4

[111] Rice, J. "Collaborative Production Strategies for Technological Innovation and Leadership in Network Industries: The Case of Bluetooth", *Danish Research Unit on Industrial Dynamics (DRUID) Winter Conference*, Klarskovgaard, 2001

[112] Hager, C.T.; Midkiff, S.F.; "An analysis of Bluetooth security vulnerabilities", *IEEE Wireless Communications and Networking, 2003. WCNC 2003.* Pages: 1825 - 1831 vol.3

[113] Hager, C.T.; Midkiff, S.F.; "Demonstrating vulnerabilities in Bluetooth security", *IEEE Global Telecommunications Conference, 2003. GLOBECOM '03*, 1-5 Dec. 2003, Pages: 1420 - 1424 vol.3

[114] Pasolini, G.; De Troia, M.; Verdone, R.; "Throughput evaluation for a Bluetooth piconet with link adaptation", *IEEE Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003*. Sept. 2003, Pages: 1727 - 1731 vol.2

[115] Das, A.; Ghose, A.; Gupta, V.; Razdan, A.; Saran, H.; Shorey, R.; "Adaptive link-level error recovery mechanisms in Bluetooth", *IEEE International Conference on Personal Wireless Communications 2000*. Page(s): 85 -89, Hyderabad, India, December, 2000.

[116] Y. Lim, Y. Kim, and J. S. Ma.; "An adaptive segmentation scheme for the Bluetooth-based wireless channel", *IEEE International Conference on Computer Communications and Networks (ICCCN 2001)*, Scottsdale, AZ, 2001

[117] Valenti, M.C.; Robert, M.; Reed, J.H.; "On the throughput of Bluetooth data transmissions", *IEEE Wireless Communications and Networking Conference, 2002. WCNC2002.* 17-21 March 2002 Pages: 119 - 123 vol.1

[118] De Morais Cordeiro, C.; Agrawal, D.P.; Sadok, D.H.; "Interference modeling and

performance of Bluetooth MAC protocol", *IEEE Transactions on Wireless Communications*, Nov. 2003, Volume: 2, Issue: 6, Pages: 1240 – 1246.

[119] De Morais Cordeiro, C.; Sadok, D.; Agrawal, D.P.; "Modeling and evaluation of Bluetooth MAC protocol", *International Conference on Computer Communications and Networks 2001*. 15-17 Oct. 2001 Pages: 518 - 522

[120] Zurbes, S.; "Considerations on link and system throughput of Bluetooth networks", *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2000.* 18-21 Sept. 2000 Pages: 1315 - 1319 vol.2

[121] Misic, J.; Misic, V.B.; "Modeling Bluetooth piconet performance", IEEE Communications Letters, Jan 2003, Volume: 7, Issue: 1, Pages: 18–20

[122] Chui, T.Y.; Thaler, F.; Scanlon, W.G.; "A novel channel modeling technique for performance analysis of Bluetooth baseband packets", *IEEE International Conference on Communications*, 2002. ICC 2002, 28 April-2 May 2002, Volume: 1, Pages: 308 – 312.

[123] Al-Harthi, S.; Ramesh Rao; "A switch model for improving throughput and power fairness in Bluetooth piconets", *IEEE Global Telecommunications Conference, 2003. GLOBECOM '03*. 1-5 Dec. 2003 Pages: 1279 - 1283 vol.3

[124] M. Connolly, C.J. Sreenan; "Analysis of UDP Performance over Bluetooth", *Information Technology & Telecommunications Conference (IT&T)*, October 2003.

[125] Ling-Jyh Chen; Kapoor, R.; Sanadidi, M.Y.; Gerla, M.; "Enhancing bluetooth TCP throughput via link layer packet adaptation", *IEEE International Conference on Communications 2004*, June 2004, Vol: 7, pp: 4012 – 4016.

[126] Won Hee Lee; Yang-Ick Joo; Kyun Hyon Tchah; Yongsuk Kim; DooSeop Eom; "Handoff provisioning in Bluetooth wireless personal area networks", *IEEE Transactions on Consumer Electronics*, Nov. 2003. Volume: 49, Issue: 4, Pages: 1004 – 1012.

[127] A.C. Boucouvalas "Probability of undetected errors in Optical wireless links", *SPIE Proceedings on Wireless Data Transmission*, 23-25 October 1995, vol. 2601, pp.247-255., Philadelphia, USA.

[128] Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. "Cyclic Redundancy and Other Checksums", *Ch. 20.3 in Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed.* Cambridge, England: Cambridge University Press, pp. 888-895, 1992.

[129] A. Capone, R. Kapoor, and M. Gerla, "Efficient polling schemes for Bluetooth picocells", *Proc. ICC 2001*, vol. 7, Helsinki, Finland, June 2001, pp. 1990–1994.

[130] J. Misic and V. B. Misic, "Modeling Bluetooth piconet performance", *IEEE Communications Letter*, vol. 7, no. 1, pp. 18–20, Jan 2003.

[131] A. Das, A. Ghose, A. Razdan, H. Saran, and R. Shorey, "Enhancing performance of asynchronous data traffic over the Bluetooth wireless adhoc network", *Proc. IEEE INFOCOM 2001*, vol. 1, Anchorage, Apr. 2001, pp. 591–600.

[132] D. Miorandi, A. Zanella and G. Pierobon, "Performance Evaluation of Bluetooth Polling Schemes: an Analytical Approach", *ACM Mobile Networks and Applications*, vol. 9, n. 1, page 63--72, Feb.2004.

[133] G. Zussman, A. Segall, and U. Yechiali, "Bluetooth Time Division Duplex - Analysis as a Polling System", *Proc. 1st IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON'04)*, Oct. 2004.

[134] Pi Huang and A. C. Boucouvalas, "Modelling IrSC: Proposal Options and Performance Analysis", submitted to *IEE Proceedings Circuits, Devices & Systems*, Jun, 2005

[135] Pi Huang and A. C. Boucouvalas, "Bluetooth L2CAP Modelling and performance Analysis", submitted to *IEE Electronic Letters*, 2005.

[136] Pi Huang and A. C. Boucouvalas, "Future Personal 'e-Payment': IrFM", accepted for publication in *IEEE Magazine on Wireless Communications*, 2005.

[137] Pi Huang, P. Chatzimisios and A. C. Boucouvalas, "Optimising IrDA throughput by including processing time with physical layer consideration", submitted to *the OSA Journal of Optical Networking (JON)*, 2005.

[138] Pi Huang and A. C. Boucouvalas, "Modelling OBEX over IrDA Protocol Stack", accepted for publication in *International Journal of Communication Systems (Wiley)*, Nov, 2004.

[139] Pi Huang and A. C. Boucouvalas, "Enhancing the Performance of High Speed Free-Space Optical Links through An Adaptive Link Layer", submitted to *IEE Proceedings Communications*, 2005.

[140] A. C. Boucouvalas and Pi Huang, "Gbit/s Data Rate IrDA Protocol Performance Evaluation", submitted to *the Mediterranean Journal of Computers and Networks*, 2005.

[141] Pi Huang and A. C. Boucouvalas, "OBEX over IrDA: Performance Analysis and Optimization by Considering Multiple Applications", submitted to *IEEE/ACM Transaction on Networking*, Sept, 2004.

[142] Pi Huang and A. C. Boucouvalas, "IrBurst Modelling and Performance Analysis in the Presence of Transmission Errors", submitted to *the Journal of Wireless Personal Communications (Kluwer)*, Sept, 2004.

[143] A. C. Boucouvalas and Pi Huang, "Modelling and Optimizing TinyTP over IrDA Stacks", *EURASIP Journal of wireless communication network*, volume 2005:1, pages: 45-56. DOI: 10.1155/WCN.2005.45

[144] Pi Huang and A. C. Boucouvalas, "Modelling the Bluetooth Logical Link Control and

Adaptation Layer", in the *proceeding of PREP2005*, Page: 58-59, Lancaster, Mar 2005.

[145] Pi Huang and A. C. Boucouvalas, "Delay Analysis for Bluetooth Baseband ACL Packets", *Convergence of Telecommunications, Networking & Broadcasting Symposium 2005 (PGNET 2005)*, Liverpool, Jun 2005.

[146] Pi Huang and A. C. Boucouvalas, "Modelling IrDA Transport Layer TinyTP", *Communication Systems, Networks and Digital Signal Processing 2004 (CSNDSP2004)*, New Castle, Jul 2004. ISBN: 0-7017-0177-3, Page: 374-377.

[147] Pi Huang and A. C. Boucouvalas, "Analysis of the IrFM Digital Payment System", *Communication Systems, Networks and Digital Signal Processing 2004 (CSNDSP2004)*, New Castle, Jul 2004. ISBN: 0-7017-0177-3, Page: 478-482.

[148] Pi Huang and A. C. Boucouvalas, "OBEX Performance Evaluation and Parameter Optimization for High Speed IrDA Links", *IEEE International Conference on Communications (ICC 2004)*, Paris, Jun 2004. ISBN: 0-7803-8534-9, Volume: 7, Page: 3849 - 3853.

[149] Pi Huang and A. C. Boucouvalas, "Analysis of the High Speed Infrared Information Transmission Protocol: IrBurst", *Convergence of Telecommunications, Networking & Broadcasting Symposium 2004 (PGNET 2004)*, Liverpool, Jun 2004. ISBN: 1-9025-6010-8, Page: 328-332.

[150] Pi Huang and A. C. Boucouvalas, "A Link Layer Analytical Model for High Speed Full-Duplex Free Space Optical Links", *International Optical Wireless Communications Symposium (IOWCS)*, Warwick, Sept 2003. ISBN: 0-902683-68-3, Volume:1, Page:36-45.

[151] Pi Huang, P. Chatzimisios and A. C. Boucouvalas, "A study of link propagation delay and processing speed for IrDA links at Gbit/s data rate", *International Optical Wireless Communications Symposium (IOWCS),* Warwick, Sept 2003. ISBN: 0-902683-68-3, Volume:1, Page:21-30.

[152] Pi Huang and A. C. Boucouvalas, "OBEX and High Speed IrDA Links", *International Optical Wireless Communications Symposium (IOWCS)*, Warwick, Sept 2003. ISBN: 0-902683-68-3, Volume:1, Page:73-78.

Appendix I

Common equations used in the derivation:

$$t_s = \frac{l'_{LAP}}{C},\ t_l = \frac{l_{LAP} + l'_{LAP}}{C},\ p = 1-(1-p_b)^{l_{LAP}+l'_{LAP}},\ t_{Fout} = t_l + 2t_{ta},$$

$$F_{LAP} = l_{LAP}\frac{1-p}{p}\frac{(1-(1-p)^{N_{LAP}})}{N_{LAP}t_l + t_{ack} + 2t_p + p(t_{Fout} + t_s + 2t_p)}$$

1. Derive optimal value for l_{LAP} when N_{LAP} is fixed

Approximations:

App0 $p = 1-(1-p_b)^{l_{LAP}+l'_{LAP}} \approx 1-(1-(l_{LAP}+l'_{LAP})p_b) = (l_{LAP}+l'_{LAP})p_b$

App1 $\frac{1-p}{p} = \frac{1}{p} - 1 \approx \frac{1}{p}$

F_{LAP} becomes:

$$F_{LAP} = l\frac{1}{(l_{LAP}+l'_{LAP})p_b}\frac{1-(1-(l_{LAP}+l'_{LAP})p_b)^{N_{LAP}}}{N_{LAP}\frac{l_{LAP}+l'_{LAP}}{C} + (l_{LAP}+l'_{LAP})p_b t_{Fout} + (l_{LAP}+l'_{LAP})p_b t_s + 2(l_{LAP}+l'_{LAP})p_b t_p + t_{ack} + 2t_p}$$

App2 $(l_{LAP}+l'_{LAP})p_b t_{Fout} = 0$

App3 $(l_{LAP}+l'_{LAP})p_b t_s = 0$

App4 $2(l_{LAP}+l'_{LAP})p_b t_p = 0$

App5 $1-(1-(l_{LAP}+l'_{LAP})p_b)^{N_{LAP}} \approx N_{LAP}(l_{LAP}+l'_{LAP})p_b - \frac{N_{LAP}(N_{LAP}-1)}{2}(l_{LAP}+l'_{LAP})^2 p_b^2$

Thus, we have rearranged F_{LAP} as

$$F_{LAP} = \frac{C}{2N_{LAP}p_b}l_{LAP}\frac{2N_{LAP}p_b - N_{LAP}(N_{LAP}-1)(l_{LAP}+l'_{LAP})p_b^2}{(l_{LAP}+l'_{LAP}) + \frac{t_{ack}C}{N_{LAP}} + \frac{2t_pC}{N_{LAP}}}$$

Do the derivative of F_{LAP}, $\frac{\partial F_{LAP}}{\partial l_{LAP}}$ and set it to 0. After some calculations, we get the positive root of l_{LAP},

$$l_{LAP} = \frac{2(N_{LAP}-1)p_b(N_{LAP}l'_{LAP} + (t_{ack}+2t_p)C) - \sqrt{8N_{LAP}(N_{LAP}-1)p_b(N_{LAP}l'_{LAP} + (t_{ack}+2t_p)C)}}{-2N_{LAP}(N_{LAP}-1)p_b}$$

Considering

$$2(N_{LAP}-1)p_b(N_{LAP}l'_{LAP} + (t_{ack}+2t_p)C) << \sqrt{8N_{LAP}(N_{LAP}-1)p_b(N_{LAP}l'_{LAP} + (t_{ack}+2t_p)C)}$$

$$l_{LAP} = \frac{\sqrt{8N_{LAP}(N_{LAP}-1)p_b(N_{LAP}l'_{LAP}+(t_{ack}+2t_p)C}}{2N_{LAP}(N_{LAP}-1)p_b} = \sqrt{\frac{2(N_{LAP}l'_{LAP}+(t_{ack}+2t_p)C)}{N_{LAP}(N_{LAP}-1)p_b}}$$

Considering $N_{LAP} \approx N_{LAP}-1$

$$l_{opt} = \sqrt{\frac{2(N_{LAP}l'_{LAP}+(t_{ack}+2t_p)C)}{N_{LAP}^2 p_b}}$$

2. Derive optimal value for N_{LAP} when l_{LAP} is fixed

By taking $A = p(t_{Fout}+t_s+2t_p)+t_{ack}+2t_p$

APP1 $(1-p)^{N_{LAP}} \approx 1-N_{LAP}p+\frac{N_{LAP}(N_{LAP}-1)}{2}p^2$

F_{LAP} becomes: $F_{LAP} = \frac{l_{LAP}(1-p)}{2p}\frac{2N_{LAP}p-N_{LAP}^2p^2+N_{LAP}p^2}{N_{LAP}t_l+A}$

Do the derivative of F_{LAP}, $\frac{\partial F_{LAP}}{\partial N_{LAP}}$ and set it to 0. After some calculations, we get a simplified equation:

$(-pt_l)N_{LAP}^2-2pAN_{LAP}+(2+p)A=0$

Considering $p<<2$, $-2pA<<-pt_l$ and $A \approx t_{ack}+2t_p$

Equation becomes: $(-pt_l)N_{LAP}^2+2(t_{ack}+2t_p)=0$

APP2 $p=(l_{LAP}+l'_{LAP})p_b$

And given $t_l = \frac{l_{LAP}+l'_{LAP}}{C}$

Finally $N_{opt} = \sqrt{\frac{2(t_{ack}+2t_p)C}{(l'_{LAP}+l_{LAP})^2 p_b}} \approx \sqrt{\frac{2(t_{ack}+2t_p)C}{l_{LAP}^2 p_b}}$

3. Derive optimal values of N_{LAP} and l_{LAP} for maximum throughput

$p=1-(1-p_b)^{l_{LAP}+l'_{LAP}} <=> 1-p=(1-p_b)^{l_{LAP}+l'_{LAP}} <=> (1-p)^{N_{LAP}}=(1-p_b)^{N_{LAP}(l_{LAP}+l'_{LAP})}$

According to the previous section, optimal N is given by

$$N_{opt} = \sqrt{\frac{2(t_{ack}+2t_p)C}{(l'_{LAP}+l_{LAP})^2 p_b}} <=> N_{LAP}(l_{LAP}+l'_{LAP}) = \sqrt{\frac{2(t_{ack}+2t_p)C}{p_b}} = A_1 \text{ (constant)}$$

Taking $A_2 = t_{Fout}+t_s+2t_p$ (constant)

APP1 $p \approx (l_{LAP}+l'_{LAP})p_b$

F_{LAP} becomes:

$$F_{LAP} = \frac{(1-(1-p_b)^{A_1}}{\frac{A_1}{C}} \frac{l_{LAP}(1-(l_{LAP}+l'_{LAP})p_b)}{(l_{LAP}+l'_{LAP})p_b(1+\frac{(l_{LAP}+l'_{LAP})p_b A_2 C}{A_1}+\frac{t_{ack}C}{A_1}+\frac{2t_p C}{A_1})}$$

Taking $H = \frac{(1-(1-p_b)^{A_1})}{\frac{A_1}{C}}$ (constant), $a = \frac{A_2 C}{A_1}$ and $b = t_{ack} + 2t_p$

Thus, $F_{LAP} = H \frac{-l_{LAP}{}^2 p_b - l_{LAP} l'_{LAP} p_b + l_{LAP}}{(l_{LAP}+l'_{LAP})^2 p_b{}^2 a + (l_{LAP}+l'_{LAP})p_b(1+\frac{bC}{A_1})}$

Do the derivative of F_{LAP}, $\frac{\partial F_{LAP}}{\partial l_{LAP}}$ and set it to 0. After some calculations, we get a simplified equation:

$$-l_{LAP}{}^2 p_b(1+2l'_{LAP} p_b a+(1+\frac{bC}{A_1}))+l_{LAP}p_b(-2l'_{LAP}{}^2 p_b a-2l'_{LAP}(1+\frac{bc}{A_1}))+l'_{LAP}{}^2 p_b a+l'_{LAP}(1+\frac{bC}{A_1})=0$$

APP2 $-2l'_{LAP}{}^2 p_b a-2l'_{LAP}(1+\frac{bc}{A_1})=0$

$$l_{opt}{}^2 = \frac{l'_{LAP}(l'_{LAP} p_b a+(1+\frac{bC}{A_1}))}{p_b(1+2l'_{LAP} p_b a+(1+\frac{bC}{A_1}))}$$

APP3 $\frac{bC}{A_1} \approx 0$

APP4 $2l'_{LAP} p_b \approx 0$

So $l_{opt} = \sqrt{\frac{l'_{LAP}}{p_b}}$

Substitute l_{opt} in $N_{LAP} = \sqrt{\frac{2(t_{ack}+2t_p)C}{l_{LAP}{}^2 p_b}}$, we get

$$N_{opt} = \sqrt{\frac{2(t_{ack}+2t_p)C}{l'_{LAP}}}$$

Appendix II

The source code of the developed MATLAB™ IrDA protocol simulation modules are presented in this section. It should be noted that the source code is annotated. The line starts with a percentage mark (%) is the comments.

IrDA Physical Layer:

```
% Function 'physical' is the IrDA physical layer. It is a random error generator based on the given error
% probability and the length of the packet.
%
% Two parameters are expected: error probability (pb) and IrLAP packet length (L). It returns a
%boolean value. 0 indicates error free, while 1 means error is occurred.

function err=physical(pb,L)

x=round(rand(1,L)/pb);
if find(x==2)
   err=1;
else
   err=0;
end
```

IrLAP Sender:

```
% Function 'irlapsender' is the IrLAP sender. It calculates the time to transmit the IrLAP window. By
%using the function 'physical', 'irlapsender' considers the channel error and includes the possible
%supervision frame transmission.
%
% Parameters N, lLAP, C, tta and pb are expected. The function returns the IrLAP window
%transmission time (Tlapsend)

function Tlapsend=irlapsender(N, lLAP, C, tta, pb)
lLAPmax=16384;
LLAP=72;

% time to transmit the IrLAP window with length of N
frametime=(lLAP+LLAP)/C;
Tlapsend=frametime*N+tta;

% the time to transmission the supervision frame when error occurs in the last IrLAP frame
if physical(pb,lLAP+LLAP)
   Tlapsend=Tlapsend+(lLAPmax+LLAP)/C+2*tta;
   while physical(pb, LLAP)
      % when the supervision frame is in error, resend the S-packet
      Tlapsend=Tlapsend+(lLAPmax+LLAP)/C+2*tta;
   end
end
```

IrLAP Receiver:

```
% Function 'irlapreceiver' is the IrLAP receiver. It calculates the time to transmit the IrLAP ack and
%erroneous frame number if the transmission is in error.
%
% Parameters N, lLAP, C, tta and pb are expected. The function returns the IrLAP acknowledgement
```

```
%transmission time (Tlapsend) and the corrupted frame number if the transmission is erroneous.

function [Tlapack, Nerr]=irlapreceiver(N, lLAP, C, tta, pb)
lLAPmax=16384;
LLAP=72;

% Time to transmit the IrLAP ack including any possible retransmission
Tlapack=LLAP/C+tta;
while physical(pb,LLAP)
   Tlapack=Tlapack+(lLAPmax+LLAP)/C+2*tta;
end

% The number (Nerr) of the first corrupted frame. Nerr=0 if error free
Nerr=0;
for Ns=1:N
   if physical(pb,lLAP+LLAP)
      Nerr=Ns;
      break
   end
end
```

TinyTP Buffer Calculator:

```
% Function 'ttpbuffer' is the TinyTP buffer calculator. It calculates the new TinyTP buffer size (Wnew)
%and the length of TinyTP packets in the buffer (numberb) on each TinyTP connection base.
%
% Parameters W, m, lLAP, numberb, B and Tpre are expected. The function returns new TinyTP
%window size (Wnew), bits remain in buffer (numberb) and processing time (Tprocess)

function [Wnew, numberbN, Tprocess]=ttpbuffer(W, m, lLAP, numberb, B, Tpre)

% Headers of TinyTP and IrLMP (bit)
LTTP=8;
LLMP=16;
L=LTTP+LLMP;

% Processor speed (HZ)
v=1e7;

% Calculate the bits that have been processed during Tpre
% assuming each TinyTP connection is euqally devided the CPU processing power
q=round(Tpre*4*v/B);
m=m/B;
if q>=numberb
   Wnew=W-m;
   numberbN=m*(lLAP-L);
else
   Nb=(numberb-q)/(lLAP-L);
   Wnew=W-Nb-m;
   numberbN=numberb-q+m*(lLAP-L);
end

% Time to processthe TinyTP packet if the TinyTP buffer is empty (the primary can not send more
%TinyTP segment)
if Wnew
   Tprocess=0;
else
   Tprocess=numberbN/(4*v);
   Wnew=W;
```

```
    numberbN=0;
end
```

TinyTP Sender:

```
% Function 'ttpsender' is the TinyTP sender module. It calculates the time to transmit a TinyTP
%window with length of m
%
% Parameters m, lLAP, C, tta and pb are expected. The window transmission time (Tttpsend) is
%returned.

function Tttpsend=ttpsender(m, lLAP, C, tta, pb)
Tttpsend=0;
while m>0
   Tlapsend=irlapsender(m, lLAP, C, tta, pb);
   [Tlapack Nerr]=irlapreceiver(m, lLAP, C, tta, pb);
   if Nerr
      m=m-Nerr+1;
   else
      m=0;
   end
   Tttpsend=Tttpsend+Tlapsend+Tlapack;
end
```

TinyTP Receiver:

```
% Function 'ttpreceiver' is the TinyTP receiver module. It calculates the time to transmit the TinyTP
%acks and the TinyTP buffer size at the receiver side.
%
% Parameter W, m, numberb, B, C, tta, pb, lLAP and Tpre is expected. The TinyTP ack transmission
%time (Tttpack), the new TinyTP window size (Wnew) and the size of the TinyTP packet remained in
%the buffer (numberbN) are returned.

function [Tttpack, WnewN, numberbN]=ttpreceiver(W, m, numberb, B, C, tta, pb, lLAP, Tpre)
% Headers of TinyTP and IrLMP (bit)
LTTP=8;
LLMP=16;
L=LTTP+LLMP;

% Time to transmit the TinyTP ack
Tttpack=0;
x=B;
while x
   Tlapsend=irlapsender(x, L, C, tta, pb);
   Tttpack=Tttpack+Tlapsend;
   [Tlapack Nerr]=irlapreceiver(x, L, C, tta, pb);
   x=0;
   if Nerr
      Tttpack=Tttpack+Tlapack;
      x=x-Nerr+1;
   end
end

% Calculate the new TinyTP buffer size (Wnew) and the length of TinyTP packets in the buffer
%(numberb) in the function 'buffer' If the TinyTP buffer is empty, the secondary will take some time
%(Tprocess) to process the incoming TinyTP packet

for y=1:B
```

```
   [WnewN(y) numberbN(y) Tprocess(y)]=ttpbuffer(W, m, lLAP, numberb(y), B, Tpre);
end

Tpro=sum(Tprocess);
Tttpack=Tttpack+Tpro;
```

OBEX Sender:

```
% Function 'obexsender' is the OBEX sender module. It calculates the time to transmit an OBEX
%packet window with length of lobex
%
% Parameters m, lLAP, C, tta, pb and lobex are expected. The packet transmission time (Tobexsend) is
%returned.

function Tobexsend=obexsender(W, B, N, lLAP, C, tta, pb, lobex)
%LOBEX=48;
LLMP=16;
LTTP=8;
L=LTTP+LLMP;
a=B*lobex/(lLAP-LLMP-LTTP);
%[TTPTPE Tobexsend]=ttp(W, B, N, lLAP, C, tta, pb, a);

% Initialisation for the first TinyTP window transmission
Tpre=0;
numberb(1:B)=0;
Ttotal=0;

while a>0
   if a<=B*W
   Wnew=a;
else
   Wnew=B*W;
end
% Length of the current TinyTP window (m)
if N>Wnew
   m=Wnew;
else
   m=N;
end
Tttpsend=ttpsender(m, lLAP, C, tta, pb);
[Tttpack WnewN numberbN]=ttpreceiver(W, m, numberb, B, C, tta, pb, lLAP, Tpre);
Wnew=sum(WnewN);
numberb=numberbN;
if sum(numberb)
    Tpre=Tttpsend+Tttpack;
else
    Tpre=0;
  end
  a=a-m;
  if a>0
    Ttotal=Ttotal+Tttpsend+Tttpack;
  else
    Ttotal=Ttotal+Tttpsend;
  end
end
Tobexsend=Ttotal;
```

OBEX Receiver:

```
% Function 'obexreceiver' is the OBEX receiver module. It calculates the time to transmit the OBEX
%Acks.
%
% Parameter W, m, numberb, B, C, tta and pb is expected. The OBEX ack transmission time
%(Tobexack) is returned.

function Tobexack=obexreceiver(W, B, C, tta, pb)

% Headers of different layers (bit)
LTTP=8;
LLMP=16;
LOBEX=48;
L=LTTP+LLMP+LOBEX;

% Time to transmit the OBEX ack
Tobexack=0;
x=B;
while x
   Tlapsend=irlapsender(x, L, C, tta, pb);
   Tobexack=Tobexack+Tlapsend;
   [Tlapack Nerr]=irlapreceiver(x, L, C, tta, pb);
   x=0;
   if Nerr
      Tobexack=Tobexack+Tlapack;
      x=x-Nerr+1;
   end
end
```

Appendix III

The source code of the developed MATLAB™ Bluetooth protocol simulation modules are presented in this section. It should be noted that the source code is annotated. The line starts with a percentage mark (%) is the comments.

Bluetooth Physical Layer:

```
% Function 'physical' is the Bluetooth physical layer. It is a random error generator based on the given
%error probability and the length of the Baseband packet.
%
% Four parameters are expected: error probability (pb) and Baseband packet length (lbs), Baseband
%packet type (type), Synchronization threshold (Ssyn). It returns a boolean value. 0 indicates error
%free, while 1 means error is occurred.

function err=physical(pb, lbs, type, Ssyn)

% event A
p1=1;
for i=0:(62-Ssyn)
   p1=p1-factorial(62)/(factorial(i)*factorial(62-i))*pb^i*(1-pb)^(62-i);
end

% event B
p2=1-(3*pb*(1-pb)^2+(1-pb)^3)^18;

% event C
Q=lbs/15;
if type==1
   p3=1-(1-pb)^lbs;
else
   p3=1-(15*pb*(1-pb)^14+(1-pb)^15)^Q;
end

% baseband packet error rate
p=2*p1+2*p2+p3;

% Decision
x=ceil(rand(1)/p);
if x==1
   err=1;
else
   err=0;
end
```

Bluetooth Baseband Layer:

```
% Function 'Baseband' is the Bluetooth Baseband layer imitating the behaviour of both master and
%slave. It calculates the time to transmit one Baseband packet and signify any timeout. By using the
%function 'physical', 'Baseband' considers the channel error and includes the all the possible
%retransmission.
%
% Parameters pb, lbs, type, t, Ssyn and tTO are expected. The function returns the timeout value and
%the Baseband packet transmission time

function base=baseband(pb, lbs, type, t, Ssyn, tTO)
```

```
t1=625*1e-6;
T=t+t1;

% the time to transmission the supervision frame when error occurs in the last IrLAP frame
while physical(pb, lbs, type, Ssyn)&T<tTO
   T=T+t+t1;
end

% if the time out expires, notify the L2CAP layer and set the packet transmission time to tTO
base(1)=0;
if T>=tTO
   T=tTO;
   base(1)=1;
end

base(2)=T;
```

L2CAP Sender:

```
% Function 'l2capsender' is the L2CAP PDU sender. It calculates the time to transmit a PDU. By using
%the function 'baseband', 'l2capsender' includes all the possible retransmissions by considering the
%possible Baseband timeout and the undetected errors in Baseband packets.The transmission time for
%L2CAP ack is also included.
%
% Parameters pb, lbs, type, t, Ssyn, tTO, l2cap, lpl, Nactive, Nsco and Npark are expected. The
%function returns the L2CAP PDU transmission time (Tl2capsender)

function Tl2capsender=l2capsender(pb, lbs, lpl, type, t, Ssyn, tTO, l2cap, lpl, Nactive, Nsco, Npark)
L2cap=16*4;
T=0;
receiver(1)=1;
base(1)=1;
a=0;
M=floor((l2cap+L2cap)/lpl);
l=l2cap+L2cap-lpl*M;
while (receiver(1)|base(1))&a<5
   % M+1 is the number of Baseband packets to transmit for one L2CAP PDU
   for n=1:M
      base=baseband(pb, lbs, type, t, Ssyn, tTO);
      T=T+base(2);
      if base(1)
         a=a+1;
         break
      end
   end
   if base(1)
      continue
   end
   base=baseband(pb, l, type, t, Ssyn, tTO);
   T=T+base(2);
   receiver=l2capreceiver(pb, lbs, type, Ssyn, tTO, M+1);
   T=T+receiver(2);
end

if a==5
   T=(M+1)*tTO*5;
end

Tl2capsender=T;
```

L2CAP Receiver:

```
% Function 'l2capreceiver' is the L2CAP receiver. It calculates the time to transmit a L2CAP ack and
%also decides if there're any undetected Baseband errors
%
% Parameters pb, lbs, type, Ssyn, tTO and M are expected.

function receiver=l2capreceiver(pb, lbs, type, Ssyn, tTO, M)

%event B
p2=1-(3*pb*(1-pb)^2+(1-pb)^3)^18;
% event C
Q=lbs/15;
if type==1
   p3=1-(1-pb)^lbs;
else
   p3=1-(15*pb*(1-pb)^14+(1-pb)^15)^Q;
end

p=M*((2*p2)/(2^8)+p3/(2^16));

% to see if there's any undetected error in Baseband packets
x=ceil(rand(1)/p);
if x==2
   receiver(1)=1;
else
   receiver(1)=0;
end

% work out the time to transmit the L2CAP ack
l=16*4;
base(1)=1;
a=0;
while (base(1))
   base=baseband(pb, l, type, 625*1e-6, Ssyn, tTO);
   a=a+base(2);
end
receiver(2)=a;
```