Study on Short Range Optical Wireless Communications for Reliable and Efficient Data Transmission

高信頼高効率データ伝送用近距離光無線通信方式に関する研究

September 2009

Graduate School of Global Information and Telecommunication Studies
Waseda University

Optical Radio Wireless Application Engineering II

Mohammad Shah ALAM

PREFACE

The work presented in this dissertation was carried out at Matsumoto Optical Radio Wireless Application Lab, Graduate School of Global Information and Telecommunication Studies, Waseda University.

Acknowledgements

First and foremost, I would like to extend my sincere gratitude to my supervisor, Prof. Mitsuji Matsumoto, for providing me with a favorable research environment and immense support during the duration of my studies at GITS, Waseda University. Prof. Matsumoto’s invaluable contribution apart from directing my research work also includes his kind advice and encouragement throughout.

I would also like to convey my special thanks to my doctoral thesis review panel which included Prof. Takuro Sato, Prof. Shigeru Shimamoto and Prof. Hidenori Nakazato of GITS,Waseda University and Prof. Koki Sato of Shonan Institute of Technology. I am very grateful for their kind acceptance to review my doctoral thesis as well as for their helpful and constructive criticism. Also many thanks to all the GITS faculty members and staffs who helped and steered me in the correct path to conduct my research work.

I would like to extend special thanks to Mr. Gontaro Kitazumi, President of IrDA, for his infinite suggestions and expert opinion in assisting and contributing in this research work. His valuable technical input is well appreciated.

I am very grateful to my esteemed past and present colleagues at Matsumoto Lab over the years. I very much appreciate the kind inputs and suggestions I received from Mr. Kamugisha Kazaura, Mr. Liu Jun, Mr. Horacio Sanson, Mr. Bekkali Abdelmoula, Mr. Pham Tien Dat, Mr. Christian Sousa, Mr. Liu Peng and Ms. Chedlia Ben Naila. Special thanks go to Ms. Mariam Salim for her assistance in proofreading part of the document.

I would also like to thank all the members of GigaIR SIG and RoFSO research group including Mr. Katsuhiko Takagaki, Mr. Hideyuki Kamiyama, Mr. Yoshiaki Uoyama, Mr Yomonori Yazaki, Mr. Toshiji Suzuki and Mr. Kazuhiko Wakamori for always being ready to share their extensive knowledge and provide expert opinion.

I would also like to thank the Ministry of Education, Culture, Sports, Science and Technology of Japan for the scholarship which enabled me to pursue graduate studies in GITS, Waseda University.

And finally, my deepest thanks to my wife, Shamim Ara Shawkat, for her constant support and inspiration. Likewise, my sincere thanks would go to my family members who have encouraged and stood behind me throughout my studies.

Table of Contents

ACKNOWLEDGEMENTS .... **v**

TABLE OF CONTENTS .... **vii**

LIST OF FIGURES .... **xi**

LIST OF TABLES .... **xv**

LIST OF ACRONYMS .... **xvii**

SUMMARY .... **xix**

1 INTRODUCTION **1**

1.1 SHORT RANGE WIRELESS DATA CONNECTIVITY 1
  1.1.1 Full-fledged technologies 1
  1.1.2 Other Standards 2
1.2 WHY IRDA LINKS IS LIKELY THE WINNER? 3
1.3 MAIN RESEARCH CONTRIBUTION 5
1.4 ORGANIZATION OF THE THESIS 7

2 IRDA PROTOCOL STACK AND OPERATING PRINCIPLES **9**

2.1 IRDA BASIC PROTOCOL STACK 9
  2.1.1 IrDA Physical Layer (IrPHY) 10
  2.1.2 IrDA Link Access Protocol (IrLAP) 11
    2.1.2.1 Roles within an IrLAP Connection 12
    2.1.2.2 Modes 13
    2.1.2.3 IrLAP Frame Types 13
    2.1.2.4 IrLAP Media Access (MAC) Rules 15
    2.1.2.5 IrLAP Operation during Information Exchange 16
  2.1.3 IrLMP - Link Management Protocol 17
    2.1.3.1 IrLMP Terminology 17
    2.1.3.2 IrLMP Frame Format 18
  2.1.4 TinyTP - the Tiny Transport Protocol 18
    2.1.4.1 Tiny TP Flow Control 18

2.1.4.2 Segmentation and Reassembly 18

2.1.4.3 TinyTP Frame Formats 19

2.2 IRDA HIGHER LAYER PROTOCOLS 19

2.2.1 IrDA Object Exchange (IrOBEX) Protocol 19

2.2.2 IrBurst Protocol 21

2.2.2.1 IrBurst Usage Models 22

2.2.2.2 IrBurst Protocol Stack 22

2.2.2.3 Requirement to Lower Layer Protocols 24

2.2.2.4 IrBurst Data Transmission Procedure 25

2.2.3 IrSimple Protocol 25

2.2.3.1 Overview of IrSimple Protocol 26

2.2.3.2 IrDA Sequence Management Protocol (IrSMP) 28

2.2.3.3 IrSimple Data Transmission Procedure 28

2.2.3.4 Re-transfer procedure 28

3 EFFICIENT ERROR RECOVERY SCHEMES FOR IRDA LINKS 31

3.1 INTRODUCTION 31

3.2 AUTOMATIC REPEAT REQUEST (ARQ) SCHEMES 32

3.2.1 Stop-and-wait (SW) 32

3.2.2 Go-back-N (GBN) 32

3.2.3 Selective repeat (SR) 33

3.3 PROPOSED EFFICIENT ERROR RECOVERY SCHEMES FOR IRDA LINKS 33

3.3.1 Block Based Window Retransmission (BBWR) scheme 34

3.3.1.1 Grouping frames into Blocks 34

3.3.1.2 Retransmission Strategy of the Proposed Scheme 35

3.3.1.3 Performance Analysis 37

3.3.2 Improved Selective Repeat (ISR) Error Recovery Scheme 42

3.3.2.1 Retransmission strategy 42

3.3.2.2 Experiment setup 44

3.3.2.3 Simulation model 45

3.3.2.4 Model validation 45

3.3.2.5 Performance Evaluation 46

3.4 CONCLUSION 47

4 PERFORMANCE EVALUATION OF IRBURST PROTOCOL FOR LARGE DATA BLOCK TRANSMISSION OVER IRDA LINKS 49

4.1 INTRODUCTION 49
4.2 MATHEMATICAL MODELING AND PERFORMANCE EVALUATION 50
    4.2.1 Protocol Mapping of IrBurst and OBEX 50
        4.2.1.1 IrBurst Throughput Efficiency in Error free Transmission........................... 52
        4.2.1.2 IrBurst Throughput Efficiency in presence of transmission errors................ 54
        4.2.1.3 IrBurst Throughput Efficiency considering processor speed......................... 56
        4.2.1.4 OBEX Throughput efficiency for error free transmission 57
        4.2.1.5 OBEX Throughput Efficiency in presence of transmission errors 59
    4.2.2 Performance Evaluation and Results 60
        4.2.2.1 Model Validation 61
        4.2.2.2 IrBurst –OBEX Performance Comparison 61
        4.2.2.3 Performance Evaluation of IrBurst in Presence of Transmission Errors 64
4.3 ENHANCING PERFORMANCE OF IRBURST PROTOCOL 68
    4.3.1 Proposed BBWR ARQ scheme in case of Large Window Size 69
    4.3.2 Proposed ISR ARQ scheme in case of Large Frame Length..................................... 70
4.4 CONCLUSION 71

5 PERFORMANCE EVALUATION OF IRSIMPLE PROTOCOL AND ITS EFFICIENCY ENHANCEMENT WITH EFFECTIVE ERROR CONTROL AND FLOW CONTROL SCHEMES 73

5.1 MATHEMATICAL ANALYSIS 73
    5.1.1 Modeling of IrSimple 74
    5.1.2 Analysis 78
        5.1.2.1 Performance comparison of IrSimple and OBEX at Various Data Rates...... 78
        5.1.2.2 Relationship between IrSMP block size and IrPHY turn around time 79
        5.1.2.3 Relationship between IrSMP block size and data size.................................. 79
5.2 SIMULATION AND ANALYSIS.. 79
    5.2.1 Simulation Parameters.. 81
    5.2.2 Simulation results 81
        5.2.2.1 Model Validation 81

5.2.2.2 Performance Comparison of IrSimple and OBEX at Various Bit Error Rates 82
5.2.2.3 Effect of IrSMP Block Size on IrSimple Performance 84
5.3 IMPROVING IRSIMPLE PERFORMANCE 85
5.3.1 Enhancing Error control scheme 86
5.3.1.1 Existing error control scheme 86
5.3.1.2 Proposed Error Control Scheme 86
5.3.1.3 Effectiveness of the Proposed Error Control Scheme 89
5.3.2 Efficient Flow Control 89
5.3.2.1 Existing Flow Control 90
5.3.2.2 Proposed Efficient Flow Control 91
5.3.2.3 Robustness of Proposed Flow Control Scheme 91
5.4 CONCLUSIONS 95

6 CONCLUSION 97

6.1 SUMMARY OF THE STUDIES 97
6.2 FUTURE WORK 98

BIBLIOGRAPHY 99

APPENDIX A. LIST OF ACHIEVEMENTS 103

List of Figures

Figure 1.1 Wireless communication systems.......... 4
Figure 2.1 Basic IrDA Protocol stack. 10
Figure 2.2 IrLAP frame format (Adapted from [32]).......... 14
Figure 2.3 TinyTP Connect Packet and Data Packet Format 19
Figure 2.4 OBEX Protocol Stack.......... 21
Figure 2.5 OBEX conversation format 21
Figure 2.6 Overview of IrBurst Protocol. 22
Figure 2.7 OBEXTM control type IrBurst protocol stack. 23
Figure 2.8 Independent control type IrBurst protocol stack.......... 23
Figure 2.9 IrSimple and standard IrDA protocol conversation 27
Figure 2.10 IrSimple protocol stack.......... 27
Figure 3.1 Extended control field format of S-frame 35
Figure 3.2 Proposed ARQ scheme for block size of 2 frames and N=15 frames.......... 36
Figure 3.3 Throughput efficiency versus BER for t_{tat}=0.1ms, l=2KB and C=100Mbit/s 38
Figure 3.4 Throughput efficiency versus BER for t_{tat}=0.1ms, l=2KB and C=100Mbit/s 39
Figure 3.5 Throughput efficiency versus BER for BBWR ARQ scheme with t_{tat}=0.1ms, l=2KB and C=100Mbit/s. 40
Figure 3.6 Throughput efficiency versus BER for BBWR with N=31 frames, L=2KB and C=100Mbit/s. 41
Figure 3.7 Performance comparison of BBWR with N=31 frames, t_{tat}=0.1ms and L=2KB. 42
Figure 3.8 Proposed Improved Selective Repeat (ISR) ARQ scheme. 43
Figure 3.9 UFIR Demo boards.......... 44
Figure 3.10 Experiment setup.......... 45
Figure 3.11 Throughput efficiency: experiment versus simulation.......... 46
Figure 3.12 Throughput efficiency versus BER for GBN ARQ scheme with C = 1Gbps, t_{tat} = 0.001ms, N = 127 frames. 47
Figure 3.13 Throughput efficiency versus BER for GBN scheme and ISR ARQ scheme with C = 1Gbps, t_{tat} = 0.001ms, N = 127 frames. 47
Figure 4.1 Mapping IrBurst, TinyTP, IrLMP to IrLAP frames.......... 53
Figure 4.2 OBEX packetization and protocol mapping 58
Figure 4.3 Throughput efficiency: analysis versus simulation.......... 61

Figure 4.4 Performance comparison of IrBurst and OBEX for various data rates. 62

Figure 4.5 Performance comparison of IrBurst and OBEX for various bit error rates at C=100Mb/s. 63

Figure 4.6 Performance comparison of IrBurst and OBEX for various bit error rates at C=16Mb/s. 64

Figure 4.7 Throughput Efficiency versus BER for C=100Mb/s, L=100MB and l=2KB. ... 65

Figure 4.8 Throughput efficiency versus processor speed for different BERs and minimum turnaround time (t_{tat}). 67

Figure 4.9 Effect of window size on IrBurst performance at different BERs for C=100Mb/s, t_{tat}=0.1ms, l=2KB and L=100MB 67

Figure 4.10 Effect of IrLAP frame length on IrBurst performance at different BERs for C=100Mb/s, t_{tat}=0.1ms, N=7 and L=100MB 68

Figure 4.11 IrBurst throughput efficiency versus BER for t_{tat}=0.1ms, *l*=2KB and *C*=100Mbit/s 70

Figure 4.12 Supervisory and Information frame structure for Proposed ISR ARQ scheme 70

Figure 4.13 Throughput Efficiency versus BER for *C*=100Mb/s, t_{tat}=0.1ms, *N*=7 and *L*=100MB 72

Figure 5.1 Mapping IrSimple protocol to IrLAP frames 76

Figure 5.2 Performance comparison of IrSimple and OBEX for various data rates with *l*=2KB, *L*=4MB and *BS*=512KB. 78

Figure 5.3 Relationship between block size and turn around time........ 80

Figure 5.4 Relationship between block size and data size 80

Figure 5.5 Throughput Efficiency: Analysis versus Simulation 82

Figure 5.6 Throughput Efficiency versus BER for IrSimple and OBEX with *C*=100Mb/s, *L*=4MB, *l*=2KB and *BS*=64KB 83

Figure 5.7 Throughput Efficiency versus BER for IrSimple and OBEX with *C*=16Mb/s, *L*=4MB, *l*=2KB and *BS*=64KB. 84

Figure 5.8 IrSimple Throughput efficiency versus bit error rate for various block sizes with *C*=100Mbps, t_{tat}=1.0ms, *l*=2KB and *L*=4MB. 85

Figure 5.9 Existing Error Control Scheme........ 87

Figure 5.10 Proposed Error Control Scheme........ 88

Figure 5.11 Throughput efficiency versus BER for C=100Mbps, t_{tat}=1.0ms, *l*=2KB and *L*=4MB. 89

Figure 5.12 Existing Flow Control Scheme 90
Figure 5.13 Proposed Efficient Flow Control Scheme 92
Figure 5.14 Recovery of Receive Ready (RR-F) frame loss in the proposed flow control scheme of IrSimple protocol 93
Figure 5.15 Recovery of Receive Ready (RR-P) frame loss 94
Figure 5.16 Recovery of Unnumbered Information (UI) frame loss immediately after the release of the flow control 95

List of Tables

Table 4.1 Mathematical Model Variables for IrBurst protocol 51

Table 5.1 Mathematical Model Variables for IrSimple protocol 75

List of Acronyms

ACK	Acknowledgment
ARQ	Automatic Repeat Request
BBWR	Block Based Window Retransmission
BER	Bit Error Rate
BL	Block Last
BOF	Beginning of Frame
CRC	Cyclic Redundancy Checks
DL	Data Last
EOF	End of Frame
FEC	Forward Error Correction
FIR	Fast Infrared
GBN	Go-Back-N
HDLC	High-Level Data Link Control
I-Frame	Information Frame
IR	Infrared
IrDA	Infrared Data Association
IrLAP	IrDA Link Access Protocol
IrLMP	IrDA Link Management Protocol
IrSimple	Infrared Simple
IrSMP	IrDA Sequence Management Protocol
ISR	Improved Selective Repeat
LAN	Local Area Networks
LM-IAS	Link Management Information Access Service
LM-MUX	Link Management Multiplexer
LSAP	Link Service Access Point
LSAP	Link Service Access Point
NDM	Normal Disconnected Mode
NRM	Normal Response Mode
OBEX	Object Exchange

P/F	Poll/Final
P2P	Peer to Peer
PAN	Personal Area Networks
PDA	Personal Digital Assistant
RS	Response
SAR	Segmentation and Reassembly
S-frame	Supervisory Frame
SNR	Signal-to-Noise Ratio
SR	Selective Repeat
SW	Stop and Wait
TE	Throughput Efficiency
UFIR	Ultra Fast Infrared
U-frame	Unnumbered Frame
UWB	Ultra Wideband
VFIR	Very Fast Infrared
VLC	Visible Light Communications
WiFi	Wireless Fidelity

Summary

With the increased demand for short range wireless connectivity, a plethora of radio and optical wireless solutions vie for the resource limited portable devices. While RF wireless technologies, in the form of WiFi (Wireless Fidelity), Bluetooth, UWB (ultra wideband) etc., is getting lots of attention from the researchers, IrDA (Infrared Data Association) links is still an appropriate, cost-effective option with high speed, simple protocol, lack of setup complications, low power, and low cost. Infrared radiation, as a medium for short-range communication, offers several significant advantages over RF transmission, especially if a short-range, low-power, high data-rate connection is the main criterion. Moreover, the infrared spectral region offers virtually unlimited bandwidth that is unregulated worldwide. Another great advantage of IrDA links is it does not require any pairing of devices thus enabling the devices to establish a connection within a short period of time by simply pointing them to each other. This “Point & Shoot” capability makes IrDA superior to other wireless standards. IrDA links also provides a stable connection without being affected by RF interference sources and can be used in areas sensitive to RF interference such as hospitals and aircraft.

In recent years, the environment to enjoy the multimedia content has changed. Most multimedia contents are changed from analog format to digital format which makes those suitable to exchange between consumer electronics and home appliances. As a result, the users need a quick and easy way to transfer digital contents from home servers to portable devices and vice versa to enjoy those anytime, anywhere. It is obvious that IrDA links with two promising protocols, IrBurst protocol and IrSimple protocol, is optimum for this type of point to point communications.

IrBurst is a higher layer protocol designed for high speed exchange of large-scale information. It uses the burst transmission capability of the lower layers of IrDA protocol stack for transmitting large bursts of information without adding any additional overhead. On the other hand, Infrared Simple (IrSimple) is a high-speed infrared communications protocol to provide simple and instant wireless communications between mobile devices and digital home appliances. This simple method of data communications helps shorten time to exchange mobile contents as the mobile contents become larger in size to accommodate the needs of high resolution picture files, music files and video files. IrSimple completes the ‘missing link’ from digital cameras, mobile phones and PDAs to

color photo printer or a television for faster file transfer without any inherent complication and lack of security with other wireless technologies.

The roadmap of Infrared Data Association (IrDA) projects very high speed data connectivity (10Gbps) in near future. Reliability will be a challenging issue for such high speed connectivity as the existing error recovery scheme adopted by IrDA does not fit well to erroneous environment. It is therefore of great importance to investigate the suitability of the two promising IrDA protocols, IrBurst protocol and IrSimple protocol, for high speed data exchange and to enhance the robustness of high speed IrDA links.

Reliability in wireless links can be achieved using Automatic Repeat Request (ARQ) scheme or Forward Error Correction (FEC) scheme. This research work aims to enhance the robustness of high speed IrDA links by proposing ARQ based efficient error recovery schemes. In many indoor environments, there exists intense noise, arising from sunlight, incandescent lighting and fluorescent lighting. With the increase of data transfer rates beyond 100Mbit/s over the half duplex infrared links, a robust automatic repeat request (ARQ) scheme is therefore necessary to cope with this erroneous environment. For its inherent simplicity, Go-Back-N ARQ scheme is deployed for data transmission over IrDA links. However at high data rate, this ARQ scheme requires some lower layer parameters, such as window size and frame size, be adapted to the corresponding optimum values for the correspondent Bit Error Rate (BER). But adaptive approaches always add a significant amount of complexity to the system. Hence, in this work, I propose two error recovery schemes, Block Based Window Retransmission (BBWR) ARQ scheme and Improved Selective Repeat (ISR) scheme, for high speed infrared communication without adapting parameters to the optimum values. The proposed schemes are variants of traditional selective repeat ARQ scheme with controlled buffer management such that buffer overflow never happens. Simulations results for BBWR scheme are presented. Experimental results for ISR scheme, which has very simple algorithms and therefore fitted well in the memory constrained 100Mbps demo boards, are also presented as the proof of concept. Furthermore a simulation model is designed and then verified by comparing the experimental results. Consequently, the proposed ISR scheme is examined by simulation for future GigaIR (1Gbps IrDA links) links which has recently been adopted by IrDA.

Another contribution of this research work is in investigating the suitability of IrBurst protocol for large block data exchange over high-speed IrDA links. The IrBurst issue has been examined in few other research works but the results presented are not sufficient for the complete performance analysis of IrBurst protocol. Furthermore the performance improvement of IrBurst protocol compared to existing IrOBEX protocol is not presented in any of the works. It is therefore of interest to develop a systematic and comprehensive analysis of IrBurst protocol for large data block exchange over high speed IrDA links and to compare the performance with existing protocol. Hence, in this research, I investigate the IrBurst protocol behavior in detail and derive a comprehensive and more realistic model for IrBurst. A complete analytical model is carried out to derive IrBurst throughput efficiency (TE) over the IrDA protocol stacks both in the case of error free transmission and in presence of transmission errors. Results are presented which reveal that IrBurst scales well to handle large data blocks at high data rates compared to the existing OBEX protocol. The impact of different layer parameters on IrBurst throughput efficiency (TE) in presence of transmission errors is examined. Furthermore the effect of proposed BBWR ARQ scheme and ISR ARQ scheme on IrBurst throughput efficiency is also examined. Simulation results show that employment of the proposed ARQ schemes highly improves IrBurst throughput performance at high bit error rates (BERs).

The final contribution of this work is performance evaluation of IrDA high speed transmission protocol, IrSimple Protocol and its efficiency enhancement with effective error control and flow control schemes. A mathematical model is carried out to derive a simple equation for IrSimple throughput efficiency (TE) over the IrDA protocol stacks. Based on this model, the performance of IrSimple protocol is compared with existing IrOBEX protocol at various data rates. It has shown that the IrSimple protocol scales well to handle fast data exchange at high data rates compared to the existing OBEX protocol. In order to evaluate the IrSimple performance in presence of transmission errors, a simulation model is also developed and based on the model a performance comparison of IrSimple and existing IrOBEX protocol is carried out for various bit error rates. Furthermore, the effect of different layer parameters on IrSimple throughput is explored for various BERs. Finally, improvement in flow control for IrSimple protocol is presented by introducing small sized supervisory frame at lower layer to reduce the traffic in the system. A complete examination of the proposed flow control scheme is carried out for all possible cases where frame losses can occur due to transmission error. It shows that the proposed flow control scheme recovers well from possible frame losses without adding any complexity to the system and reduces the redundant data retransmissions.

Chapter 1

Introduction

In recent years, the consumer electronics has seen a rapid drop in price for memory chips which enables all the portable devices to surge with gigantic memory size as a built-in option. At the same time, the environment of enjoying multimedia content has been changed with the alteration of all analog formats to digital formats and users can now carry their favorite data contents such as music files, movie files, and picture files in portable devices to enjoy anytime, anywhere. As a result, users are now opting for efficient data transfer methods to exchange data contents between portable device such as music player, camera, cell phone etc. and home servers such as desktop computer, DVD recorder. The short range wireless connectivity is therefore thriving as one of the best options to support this type of services by replacing the traditional cables thereby giving the users greater degree of mobility.

1.1 Short Range Wireless Data Connectivity

A plethora of short-range wireless data communication solutions now vie for the limited connectivity option of resource restricted consumer electronics. The current crops of contenders mainly include IrDA, Bluetooth and WiFi (802.11a/b/c/d/g/n), and many new technologies such as TransferJet, Visible Light Communications (VLC) are thriving. These technologies cover a wide range of capabilities and constraints. Sometimes they compete and other times they complement.

1.1.1 Full-fledged Technologies

WiFi

The first IEEE 802.11 specification was introduced in 1997 with the primary goal of providing wireless LAN access. At first, component costs were expensive, interoperability was chancy, and security was a major concern. Together, these factors prevented widespread adoption. But, over time, component cost has dropped, many security concerns have been addressed, and new specification versions (such as

802.11b, 802.11a, and 802.11g) have emerged that increase throughput. Because of the large physical range and “always-on” connection model, Wi-Fi technology consumes a lot of power, limiting its use in PDAs, phones, and other lightweight portable devices [1].

Bluetooth

Bluetooth-enabled wireless headsets started to emerge in 2000, but component cost, power usage, and even regulatory barriers prevented widespread adoption. Since then, cost and power usages have improved, making Bluetooth a valuable add-on feature for high-end PDAs and mobile phones. However the maximum data rate supported by Bluetooth 2.0 was still 3Mbps.

Recently *Bluetooth* 3.0 is adopted which is expected to fulfill the consumers’ need for speed while providing the same wireless *Bluetooth* experience. Manufacturers of consumer electronics and home entertainment devices can now build their products to send large amounts of video, music and photos between devices wirelessly at speeds consumers expect. The inclusion of the 802.11 Protocol Adaptation Layer (PAL) in *Bluetooth* 3.0 provides increased throughput of data transfers at the approximate rate of 24 Mbps [2].

IrDA

The IrDA protocol specification was developed by inclusive organizations to provide short range, low-cost, indoor, point-to-point links utilizing the IR spectrum. IrDA also offers the advantage of being easy to implement and simple to use, in addition to the high data rates. IrDA links aim to replace cables between devices such as laptop computers, personal digital assistants (PDAs), digital still and video cameras, mobile phones and printers. The ‘point and shoot’ nature of the IrDA user model requires line of sight link alignment, and as a result short data transfer time is important. IrDA is on the process of adopting its GigaIR specification to support 1Gbps data rates while it is projecting for 10Gbps data links in near future [3].

1.1.2 Other Standards

TransferJet

TransferJet is a wireless technology that allows a pair of devices that want to communicate to do so simply by bringing them close together. While the communication distance is only up to a short 3 cm, it achieves ultrahigh-speed transfers with an effective data transfer rate of 375 Mbits/s [4].

Wireless USB

Wireless USB is based on the Ultra-WideBand (UWB) common radio platform, which is capable of sending 480 Mbit/s at distances up to 3 meters and 110 Mbit/s at up to 10 meters [5].

RFID

RFID is a wireless alternative to barcode scanners, allowing a component costing 25 cents or less to identify itself without a power source [6].

ZigBee

ZigBee, like Bluetooth, uses an unlicensed RF band for data communication, but targets applications that demand lower power, lower throughput, and greater physical range such as home automation, remote control, and device monitoring [7].

Ultra-Wideband (UWB)

Ultra-wideband uses a unique signalling mechanism that allows extremely high throughput (100 Mbps or more) using a simplified component design requiring very little power. Because UWB technology transmits over a wide swath of radio frequencies, it is unimpeded by the interference problems that obstruct traditional RF and infrared signals [8].

Visible Light Communications (VLC)

The data can be transmitted by lighting LED on and off at ultra high speed. The visible light communication has characteristics to be ubiquitous, transmitted at ultra high speed and harmless for human body and electronic devices, compared to those by radio and infrared communications [9].

1.2 Why IrDA Links is likely the Winner?

To date, various wireless technologues (as shown in Figure 1.1) have been developed and are used for personal area networks (PANs) and local area networks (LANs). These technologies have different data rates and coverage distance. Therefore

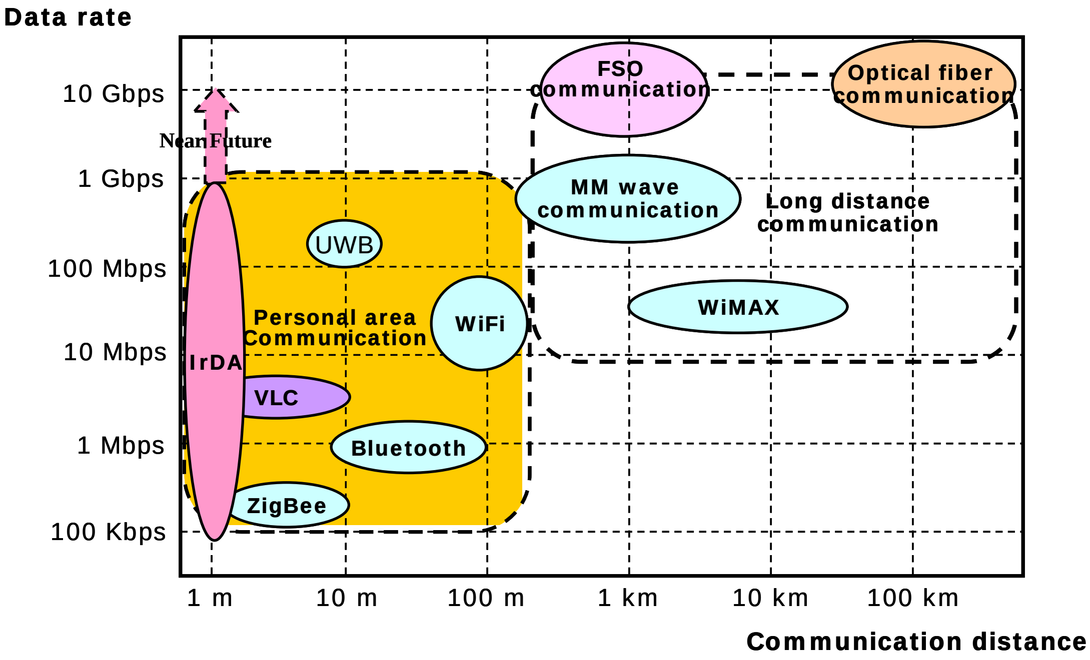


Figure 1.1 Wireless communication systems

depending on the deployment scenario, intended application and the cost benefit analysis, some technologies can be more suitable than others. The rapid emergence of mobile terminals in work and home environments is accelerating the introduction of mobile wireless links. But portable devices are subject to severe limitations on size, weight, power consumption and battery lifetime. The desire for inexpensive, high-speed links satisfying all these requirements has motivated the interest in infrared wireless communication.

Infrared radiation, as a medium for short-range communication, offers several significant advantages over RF transmission, especially if a short-range, low-power, high data-rate connection is the main concern. High-speed infrared transceivers are available at low cost. The infrared spectral region offers virtually unlimited bandwidth that is unregulated worldwide. Another great advantage of IrDA links is it does not require any pairing of devices that is needed for RF technology. This enables the devices to establish a connection within a very short period of time.

A study in [10] finds that IrDA links takes only 1.12sec whereas Bluetooth connectivity takes at least 10.24sec to discover other devices within its range. This key feature gives IrDA a clear advantage over any other competing technologies to meet the user's primary expectation “the faster, the better” for specific types of applications. For example, a user may need to download a movie file, usually in some GB sizes, from home server to his/her portable player. Alternatively the user may want to transfer a small sized picture file from his/her cell phone to a printer or a large display. In both the cases, the user's primary requirement is simple and faster data exchange irrespective of the content size.

It is obvious that an IrDA connection is optimum for this type of services (point to point). It is superior to the other current wireless standards because of its "Point & Shoot" capability. Connection establishment between device and equipment, when compared to other wireless standards, IrDA is far quick and easy. IrDA also provides a stable connection without being affected by RF interference sources. This means that vending machines, printers, displays providing these services can be located in areas sensitive to RF interference such as hospitals and aircraft.

1.3 Main Research Contribution

The roadmap of Infrared Data Association (IrDA) projects very fast data connectivity (10Gbps) and application specific efficient protocols (IrSimple, IrBurst, IrUSB) in near future. Reliability will be a challenging issue for such high speed connectivity as the existing error recovery scheme does not fit well to cope with erroneous environment beyond 16Mbps data rate [11]. Considering all these issues, the main research contribution presented in this work is in performance evaluation of two promising protocols, IrBurst protocol and IrSimple protocol, to investigate their suitability for high speed Infrared communications and in proposing ARQ based efficient error recovery schemes to enhance the robustness of high speed IrDA links.

Reliability in wireless links can be achieved using Automatic Repeat Request (ARQ) or Forward Error Correction (FEC) scheme. In order to enhance the robustness of high speed IrDA links in case of erroneous environment, ARQ based efficient error recovery schemes are proposed in this research work. In many indoor environments, there exists intense noise, arising from sunlight, incandescent lighting and fluorescent lighting. With the increase of data transfer rates beyond 100Mbit/s over the half duplex infrared links, a robust automatic repeat request (ARQ) scheme is therefore necessary to cope with this erroneous environment. For its inherent simplicity, Go-Back-N ARQ scheme is deployed for data transmission over IrDA links. However at high data rate, this ARQ scheme requires some lower layer parameters, such as window size and frame size, be adapted to the corresponding optimum values for the correspondent Bit

Error Rate (BER). But adaptive approaches always add a significant amount of complexity to the system. Hence, in this work, I present two error recovery schemes named as Block Based Window Retransmission (BBWR) ARQ scheme and Improved Selective Repeat (ISR) scheme for high speed infrared communication operating with a fixed size receiver buffer and without adapting parameters to the optimum values. The proposed schemes are variants of traditional selective repeat ARQ scheme with controlled buffer management such that buffer overflow never happens. Simulations results for BBWR scheme are presented. Experimental results for ISR scheme, which has very simple algorithms and therefore fitted well in the memory constrained 100Mbps demo boards, are also presented as the proof of concept. Furthermore a simulation model is designed and then verified by comparing the experimental results. Consequently, the proposed ISR scheme is examined by simulation for future GigaIR (1Gbps IrDA links) links.

Another contribution of this research work is in investigating the suitability of IrBurst protocol for large block data exchange over high-speed IrDA links. The IrBurst issue has been examined in few other research works but the results presented are not sufficient for the complete performance analysis of IrBurst protocol. Furthermore the performance improvement of IrBurst protocol compared to existing IrOBEX protocol is not presented in any of the works. It is therefore of interest to develop a systematic and comprehensive analysis of IrBurst protocol for large data block exchange over high speed IrDA links and to compare the performance with existing protocol. Hence, in this research, I investigate the IrBurst protocol behavior in detail and derive a comprehensive and more realistic model for IrBurst. Furthermore the effect of proposed BBWR ARQ scheme and ISR scheme on IrBurst throughput efficiency is also examined. Simulation results show that employment of the proposed ARQ schemes highly improves IrBurst throughput performance at high BERs.

The final contribution of this thesis is performance evaluation of IrSimple Protocol and its efficiency enhancement with effective error control and flow control schemes. So far no other research work is done on the topic. In order to evaluate the IrSimple performance in presence of transmission errors, a simulation model is developed and based on the model a performance comparison of IrSimple and existing IrOBEX protocol is carried out for various bit error rates. Furthermore, in an effort to improve IrSimple performance, an effective error control scheme is proposed. Since, IrSimple protocol maintains the data flow from upper layer, an improvement in the data flow control for

IrSimple protocol is also proposed at lower layer to reduce the redundant data retransmissions in the system.

Most of the results obtained in this research work were presented at IrDA standardization meetings and were accepted.

1.4 Organization of the Thesis

This thesis is organized in six chapters detailing the design, evaluation and analysis as well as performance enhancements techniques for high speed IrDA links.

Chapter 1 as the introduction presents the background and the objective of this study highlighting its research contribution, as well as the outline of the thesis.

Chapter 2 provides an overview of IrDA protocol stack and its operating principles. A brief description of IrPHY, IrLAP, IrLMP, TinyTP layers which comprise the basic protocol stack are presented. Different service primitives, packet and frame formats are outlined. The chapter also describes higher layer protocols, existing IrOBEX protocol and newly adopted IrBurst and IrSimple protocol, in brief. This chapter would help to understand smoothly the following chapters in this thesis.

In Chapter 3, two ARQ based error recovery schemes that fit well for IrDA links are presented. Proposed Block Based Window Retransmission (BBWR) and Improved Selective Repeat (ISR) ARQ schemes are variants of ideal selective repeat ARQ scheme and require little modification to the existing error recovery scheme of IrDA system. The proposed schemes operate with a finite buffer size and a finite range of sequence number such that buffer overflow never happens. The performance of the proposed schemes are examined for various link parameters, such as bit error rate (BER), minimum turnaround time and data length, and compared with the performance of the existing go-back-*N* (GBN) ARQ scheme. Simulation results are presented which show that the proposed schemes significantly outperforms the existing GBN ARQ scheme, particularly for links with high bit error rate. At BER=10^{-6}, the proposed schemes can provide almost 50% improvement in throughput efficiency. Furthermore, experimental result of proposed ISR scheme is presented as a proof of concept which coincides with simulation result.

Chapter 4 examines the suitability of IrBurst protocol for large data block exchange over high-speed IrDA links. A complete analytical model is carried out to derive IrBurst throughput efficiency (TE) over the IrDA protocol stacks both in the case of error free transmission and in presence of transmission errors. A simulation model is also developed which validates the analytical model. The impact of different layer parameters on IrBurst throughput efficiency (TE) in presence of transmission errors is examined. Results show that IrBurst scales well to handle large data blocks, especially at high data rates, compared to the existing OBEX protocol. Further analysis shows that IrBurst throughput efficiency (TE) always benefits by a small minimum turnaround time. However, in situations where small turnaround time is not possible, using a large window size or frame length can alleviate the negative effect of large turnaround time by increasing the amount of data sent between turnarounds. Although using large window size or frame length significantly improves the throughput efficiency for low BERs, it renders the link vulnerable to BER increase. Simulation results demonstrate that proposed BBWR and ISR ARQ schemes can provide sufficient robustness for large window size or frame length and highly improves IrBurst performance at high BERs.

In chapter 5, a study on the performance of IrSimple protocol for high-speed exchange of digital contents is carried out in detail. A mathematical model has been carried out to derive IrSimple throughput efficiency (TE). Based on the model, the performance of IrSimple protocol is compared with existing OBEX protocol for various data rates. Furthermore, in order to characterize the IrSimple performance in presence of transmission errors, a simulation model for IrSimple has been developed. A performance comparison of IrSimple and OBEX is carried out for various bit error rates based on the model. Results show that IrSimple protocol scales well to handle fast data exchange at high data rates compared to the existing OBEX protocol. The significance of different layer parameters on IrSimple throughput in presence of transmission errors is also explored. Although small turnaround time results in maximum throughput, such a low turnaround time is not always achievable. Therefore, in situations where small turnaround time is not possible, the use of large block size can mitigate the negative effect of large turnaround time but renders the link vulnerable to high BERs. In an effort to improve robustness, an effective error control scheme similar to Improved Selective Repeat (ISR) ARQ scheme is proposed. Simulation result shows that employment of the proposed error recovery scheme highly improves IrSimple protocol performance at high BERs. Furthermore, an improvement in the existing flow control for IrSimple protocol is proposed to reduce the redundant data retransmissions in the system.

Finally, Chapter 6 provides the concluding remarks of this research work. Areas of future research are also outlined.

Chapter 2

IrDA Protocol Stack and Operating Principles

2.1 IrDA Basic Protocol Stack

With the establishment of the Infrared Data Association (IrDA) in 1993 formed by the collaboration between key industries, an open standard for IR data communication was founded [12]. As a result, an IrDA protocol was established whose aim was to provide a simple, cost-effective, dependable means of IR communication using direct point-to-point connectivity between LANs and devices like portable computers and non-portable computers, printers, fax machines, etc. Currently, IrDA connections can provide a baud rate up to 16Mb/s with a future aspiration of 1Gbps high speed, using half-duplex point-to-point connectivity. The protocol stack specifies both hardware and software, with the bulk of the complexity in the software driver, leaving the hardware as simple and low-cost as possible.

Communications protocols deal with many issues, and so are generally broken into layers, each of which deals with a manageable set of responsibilities and supplies needed capabilities to the layers above and below. The IrDA protocol stack is the layered set of protocols aimed at particularly point-to-point infrared communications and the applications needed in that environment. In this section, each of the basic layers of the IrDA protocol stacks (see Figure 2.1) are described, beginning with the physical layer and the layers above it.

Before going into the details description of these protocols, a brief description of the operation of each of the protocols is given below.

The IrPHY [13] layer specification details the physical hardware for the IR link that includes the IR transmitter and receiver, the filters, and the modulation and encoding hardware. This is described in more detail in the next section.

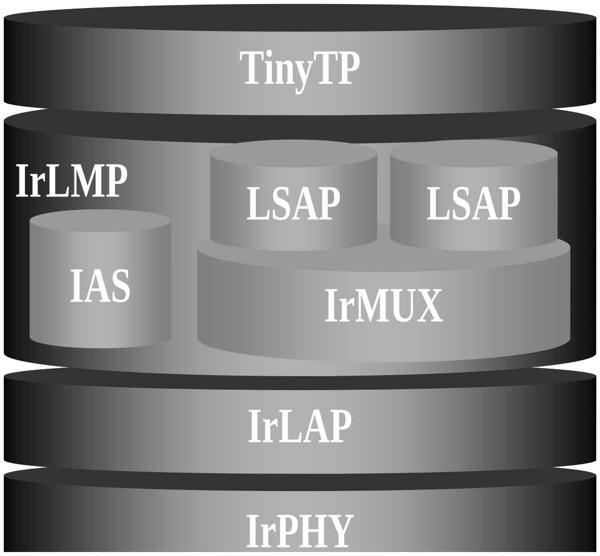


Figure 2.1 Basic IrDA Protocol stack.

The IrLAP [14] layer is an HDLC-based data link layer that provides device discovery, link establishment and shutdown, and reliable data exchange. More details are available in later sections.

The IrLMP [15] layer consists of two distinct elements. The first element is the link management multiplexer (LM-MUX) which provides a way for multiple units on a device to utilize a single established IrLAP link both independently and concurrently. This layer uses link service access points (LSAPs) to collaborate with higher layers of the protocol. This layer also has another element called the information access service (LM-IAS) that provides a database of services given by the host device and allows for the examination of the database by the remote device.

Lastly, TinyTP [16] is a non-compulsory lightweight transport and flow control method that can prevail over any prospective deadlock problems with the IrLAP layer. It insures that IrLAP is not overwhelmed with too much data that can be present at any single time by providing a data segmentation and reassembly service and a credit based flow control procedure.

2.1.1 IrDA Physical Layer (IrPHY)

The mandatory IrPHY (Infrared Physical Layer Specification) [13] is the lowest layer of the IrDA specifications. This layer’s main specifications are:

- Range:

-standard: 1 m
-low power to low power: 0.2 m
-standard to low power: 0.3 m

- Angle: minimum cone ±15°
- Speed: 2.4 kbit/s to 16 Mbit/s
- Modulation: baseband, no carrier

It deals with several traits of the infrared signals such as framing data like begin and end of frame flags (BOFs and EOFs), cyclic redundancy checks (CRCs), and encoding of data bits. It is essential for the physical layer to some extent be implemented in the hardware, but it is entirely possible for the hardware to manage this layer completely.

This layer also includes a software layer, framer, which separates the continuously changing hardware layer from the rest of the stack. Its main task is to receive incoming frames from the hardware and pass them on to the Link Access Protocol Layer (IrLAP). This includes accepting outgoing frames and doing whatever is necessary to send them. The IrDA have minimum and maximum irradiance constraints where the signal must be visible up to a meter away while also maintaining distance so as to not become overwhelmingly close to the irradiance for the receiver to perform properly. IrDA data communications operate in half-duplex mode because while transmitting a device’s receiver is blinded by the light of its own transmitter. Therefore, full-duplex communication is not possible. The two devices that communicate simulate full duplex communication by quickly turning the link around. While the primary device in command of the timing of the link, both sides are required to adhere to specific constraints and are persuaded to turn the link around swiftly.

2.1.2 IrDA Link Access Protocol (IrLAP)

The data link layer of the IrDA protocol stacks is called Infrared Link Access Protocol (IrLAP). IrLAP [14] is based on both the High-Level Data Link Control (HDLC) and the Synchronous Data Link Control (SDLC) added with some distinct characteristics of the infrared communications. IrLAP provides reliable data transfer via the following methods:

- Retransmission.
- Low-level flow control.
- Error detection.

Because the data transfer is handled at a lower level, the upper layers are not burdened with dealing with the task of reliable data deliverance or lack thereof (the upper layers are appraised of any data that is not delivered). Data delivery might fail if the beam path were blocked. Case in point, a person can put a book or a stack of magazines in the path of the infrared beam. So, the IrLAP sends a notification to the upper layer for the higher-level layers to take appropriate actions to solve the problem. For example, an alarm of the interruption in data transmission could be sent to an application of the stack which in turn can alert the user through some type of interface. With user's knowledge of the problem, a solution can be implemented (by removing the book or the stack of magazines from the beam path) without losing the connection and/or the data transmitted to its designated location.

2.1.2.1 Roles within an IrLAP Connection

There are two contributors to a Link Access Protocol (LAP) connection whose relationship can be construed as one in control of the other (master-slave relationship) with each having different responsibilities from the other. The IrDA terms for these are Primary station and Secondary station.

Primary Station

Sends Command frames—initiates connections and transfers. Standard primary items include PCs, PDAs, cameras, and anything that needs to print (printers excluded).

Secondary Station

Sends Response frames—only allowed to respond to contact, not to initiate it. Standard secondary devices are printers and other peripherals, and resource restricted items (secondary stations are smaller and less complex).

In any connection one device must play the primary role. While the other device is changed to the secondary station, its protocol stack has the option of being either that of a secondary or another primary (almost all primaries are able to act as a secondary).After beginning, the primary device heads the talking process, with the two sides taking turns. Both sides are constricted to a time limit of 500 milliseconds at a time to talk to give the other side a chance to talk, even if it responds that it has nothing to deliver at this minute. The topic of primary versus secondary becomes obscure at the higher protocol layers. As soon as the two devices are connected, an application on the side of the secondary can commence an operation as effortlessly as an application on the primary side.

2.1.2.2 Modes

IrLAP is developed with two modes of operation in response to if a connection already exists or not.

Normal Disconnected Mode (NDM)

Also referred to as a contention state, NDM is the default state of disconnected devices. In this mode a device must observe a set of media access rules. It is paramount that a device in NDM mode must verify if other transmissions are happening (also known as media busy) before transmitting. The verification is done by simply listening for activity. The media will only be considered to be available for transmission if no activity is detected for more than the maximum time it takes for a link to turn around or greater than 500 milliseconds. A great ease-of-use feature is provided by the NDM communications rules. A common glitch with link would be not being able to get both sides configured with the same communications parameters, often leading to users getting trapped. This particular problem can be challenging to overcome in embedded devices with no user interface for setting or changing communications parameters. However, this complicated issue is completely eradicated with IrDA solutions since all NMD communications use the following link parameters: ASYNC, 9600 bps, 8 bits, no parity. Throughout the linking process, the two sides exchange capability details, and change to the best parameters supported by both sides.

Normal Response Mode (NRM)

NRM is the mode of operation for connected devices. After both sides are communicating by using the optimal communication parameters, established during NDM, higher stacks are able to then apply normal command and response frames to exchange information.

2.1.2.3 IrLAP Frame Types

There are three frame types used in an IrLAP connection: unnumbered frames (U-frames), supervisory frames (S-frames), and information frames (I-frames). The frame types are dictated by the contents of the control field. Every single frame has a poll/final (P/F) bit in the control field that designates the relocation of transmission control to a different station. When the P bit for the primary station is set, signifying that the polling of the secondary station specified by the address field, the station is given the authorization to transmit. The response of the secondary using the F bit is to restore control to the primary at the end of its data window. The following describes the function of each of the frame types with their format (See Figure 2.2).

Unnumbered Frames (U-frame)

This frame is used for link management. Because these frames do not have any sequencing numbers, they are known as “unnumbered”or U-frames. The establishment of a link is specified by the control field by providing with command and respond codes. Field of their functions are notifying procedural errors that cannot be recovered by retransmissions, detecting and initializing secondary devices, etc. In the present model the link is assumed to already be established. The link establishment parameters or user data are contained within the information field to exchange information outside of an established connection.

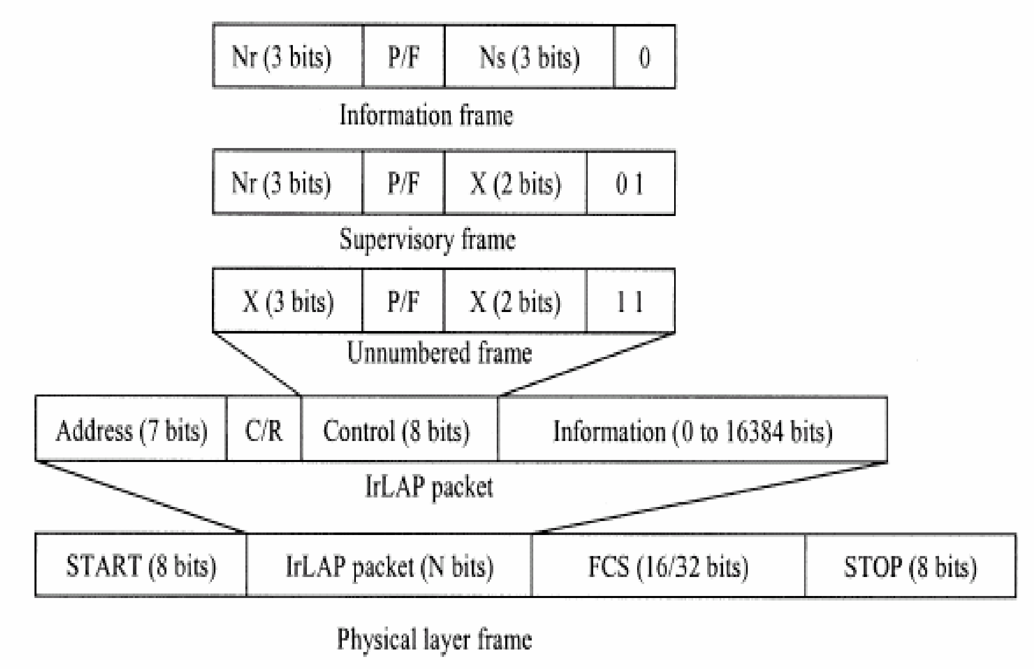


Figure 2.2 IrLAP frame format (Adapted from [32])

Information Frames (I-Frame)

To exchange information in an established connection, the I-frame is employed. In this frame, the control field has sequence numbers of ‘send’ and ‘receive’ Ns and Nr. These numbers are contained in 3-bit fields and thus can have values 0 to 7, and cycle back to zero. To accommodate the VFIR [22] specification, Ns and Nr are allocated with 7bits, the values ranging from 0 to 127, and cycle back to zero using the same method. The ‘send’ sequence numbers indexes the frames in a way that an out-of-sequence Ns. will indicate a lost frame. Therefore the ‘receive’ sequence number is used as an indication for the number of correctly (i.e. in sequence) received frames. A specified Nr indicates that Nr - 1 frames have been successfully received and that Nr is now the next anticipated frame. So, if a received Nr does not match up with its anticipated value, it means that a retransmission of frames beginning at N_r needs to be done.

Supervisory Frame (S-frame)

These frames are used to provide certain control commands and responses to control the flow of information during an established link. The S-frames compose of only the frame overhead (48 bits) in size as thy do not hold any user data within. Their primary function is to deal with positive and negative acknowledgment of frames when there is no user information transmitted by the sender. Like I-frames, they contain the receive sequence number N_r to recognize received frames, however they do not the send sequence numbers N_s. A Receive-Ready, RR, command is used to show that the sending station is prepared to receive new frames, and it also recognized frames specified by N_r. While a Receive Not Ready, RNR, is used to show that the sender station is momentarily not able to receive new frames but does still recognized frames. A reject, REJ, frame means to be a request for retransmission of frames beginning at N_r while a Selective Reject, SREJ, indicates request to retransmit a specific frame indicated by $N_{r.}$

2.1.2.4 IrLAP Media Access (MAC) Rules

The maximum turnaround time of a station that can hold onto a transmission is at most 500 ms. Since this time can supersede every other constraint, it is capable of limiting the upper limitation of frames that can be transmitted at a time to less than seven frames. The number of frames remaining to be transmitted before the station must stop and pass transmission is known as the “window size”.

The primary station contains a F-timer that is set off when the primary sends a frames containing the P-bit set. The F-timer will expire when a frame with the F-bit has not been returned within a particular time limit. This can happen when either the F-bit from the secondary is either lost or not returned or the P-bit from the primary is lost. The primary will in turn transmit a supervisory RR frame (S-RR) to push for an acknowledgement from the secondary.

The primary also contains a P-timer, set to the maximum turnaround time for the station that starts at the launch of its transmission period. When the primary has less than the maximum window value or no existing data frames to send, the station will pause for the P-timer to expire before sending S-frame to deliver transmission to the secondary after any transmission. This is in case further packets subsequently become available for transmission within the maximum transmission time of the primary.

Every time control of a link is passed, a minimum turnaround time delay must be executed by each station before moving forward. Such a time delay is put in place at startup independently for each and every station and utilized to deal with receiver latency. While the most the time delay value can be 10 ms, the value of 1 ms is usually the norm. Unfortunately, a large turnaround delay can have consequences for throughput at high data rates.

2.1.2.5 IrLAP Operation during Information Exchange

The primary holds a send sequence variable V_s a receive sequence variable V_r and a “window” value which represents the number of frames remaining to be sent before the station must stop transmission. If the primary station has data to send, Vs, and Vr are copied to the send number Ns and receive number Nr, respectively, of the outgoing frame. Thus when the frame is being sent, the value of V_s is increased by one while the current window value is decreased by one. The window value of one is an indication of the next frame being the last frame in the sequence which triggers the P-bit of the frame to be set and the station F-timer to start. And if the primary contains no data for transmission, it will send a S-RR frame on the cessation of the P-timer.

Then when the primary takes in reply I-frames from the secondary, the returned value of N_s is assessed against its projected value which should be equivalent to the primary V_r value. And when the send number N_s meets the projected number, then the data from the frame is moved on to the upper layer of the protocol stack in addition to the value of the primary V_r is increased. However, if the secondary N_s does not match the projected value, then the frame is disposed of while no changes are made to the primary $V_{r.}$ If the returned frame is an S-frame or the final I-frame containing the F-bit set, the then value of N_r is tested against its projected value that ought to be the same as the primary V_s value. Assuming that N_r is as projected, the primary can throw away the buffered send packets and carry on sending new frames as was done in the past. Wherever N_r does not match the projected value signifying that the sent packets are lost, the primary has to retransmit the buffered packets starting from the indicated N_r to the end of the sent window.

The secondary station's operation is the same with the exception of that it does not have a F-timer nor a P-timer. Which means that the secondary station will only transmit data when instructed to do so by the primary and will send back transmission if it does contain any data to be transmitted. Furthermore, the last frame in a window of the secondary has a F-bit, not a P-bit.

2.1.3 IrLMP - Link Management Protocol

IrLMP [15] is a required IrDA layer, and provides the following functionality:

- Multiplexing: LMP allows multiple IrLMP clients to run over a single IrLAP link.
- Information Access Service (IAS). A "yellow pages" describing the services available on a device.

2.1.3.1 IrLMP Terminology

In order to have multiple IrLMP connections on a single IrLAP connection, there must be some higher level addressing scheme. The following terminology is used to describe this addressing:

- LSAP (Logical Service Access Point) The point of access to a service or application within IrLMP (for example, a printing service). It is referenced with a simple one byte number, the LSAP-SEL (described next).
- LSAP-SEL (LSAP Selector): A one byte number that corresponds to an LSAP. Think of this as the *address* of a service within the LMP multiplexer. This byte is broken into ranges—0x00 is the IAS server, 0x01 through 0x6F are legal LMP connections, 0x70 are for connectionless services and the rest are reserved for future use.

Given the limited number of LSAP-SEL values, services are not assigned fixed "port addresses" as in TCP/IP. Instead, services have fixed published names, and the LMP IAS (yellow pages) is used to look up the LSAP-SEL for a desired service.

2.1.3.2 IrLMP Frame Format

The IrLMP layer adds the following 2 bytes of information to frames in order to perform its basic operations:

- C: Distinguishes between control and data frames.
- r: Reserved.
- DLSAP-SEL: LSAP-SEL (service address) of the destination of the current frame.
- SLSAP-SEL: LSAP-SEL for the sender of the current frame

2.1.4 TinyTP - the Tiny Transport Protocol

TinyTP [16] provides two functions:

- Flow control on a per-LMP-connection (per-channel) basis.
- SAR (segmentation and reassembly).

TinyTP adds one byte of information to each IrLMP packet to perform its task.

2.1.4.1 Tiny TP Flow Control

Per-channel flow control is currently the most important use of TinyTP. The need of this higher layer flow control is briefly illustrated here. A LAP connection is established, and two LMP connections are made on top of the LAP connection using the LMP multiplexing capability. If one side turns LAP flow control on, the flow of data on the LAP connection (which carries all the LMP connections) is completely cut in that direction and the other side cannot get the data it wants until LAP flow control is turned off. The work of the second side may be seriously disrupted (especially if timers are involved). If flow control is applied on a per LMP connection basis using TinyTP (or other mechanisms), then one side can stop to digest information without negatively affecting the other side.

2.1.4.2 Segmentation and Reassembly

The other TinyTP function is called SAR (segmentation and re-assembly). The basic idea is that TimyTP breaks large data into pieces (segmentation), and puts it back together on the other side (re-assembly).

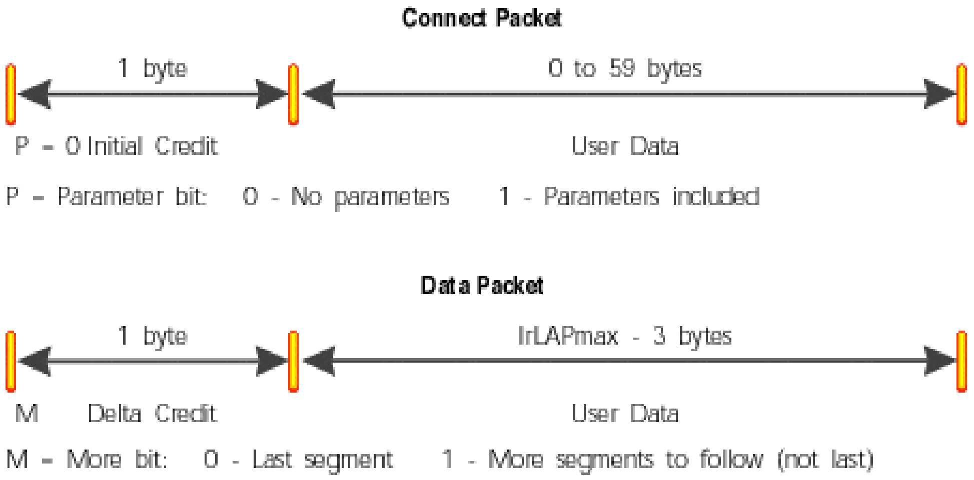


Figure 2.3 TinyTP Connect Packet and Data Packet Format

The entire piece of data being chopped up and re-constituted is called an SDU, or Service Data Unit, and the maximum SDU size is negotiated when the TinyTP/LMP connection is first made.

2.1.4.3 TinyTP Frame Format

The two frame formats used by TinyTP are the connect packet (carried with the IrLMP connect packet, hence the limited data length), and the data packet, carried with IrLMP data packets (See Figure 2.3).

2.2 IrDA Higher Layer Protocols

For the purpose of interconnecting a wide range of devices that support IrDA protocols, IrDA has developed higher layer protocols. In this section, I describe three IrDA higher layer protocols which are primarily aimed at transferring data files efficiently between devices.

2.2.1 IrDA Object Exchange (IrOBEX) Protocol

IrOBEX [17], [18], [19] is a session protocol and can best be described as a binary HTTP protocol, which operates on top of any reliable transport protocol in connection with a simple request and response paradigm. However, OBEX works for many very useful devices that support IrDA or Bluetooth communications but cannot afford the substantial resources required for an HTTP server, and it also targets devices

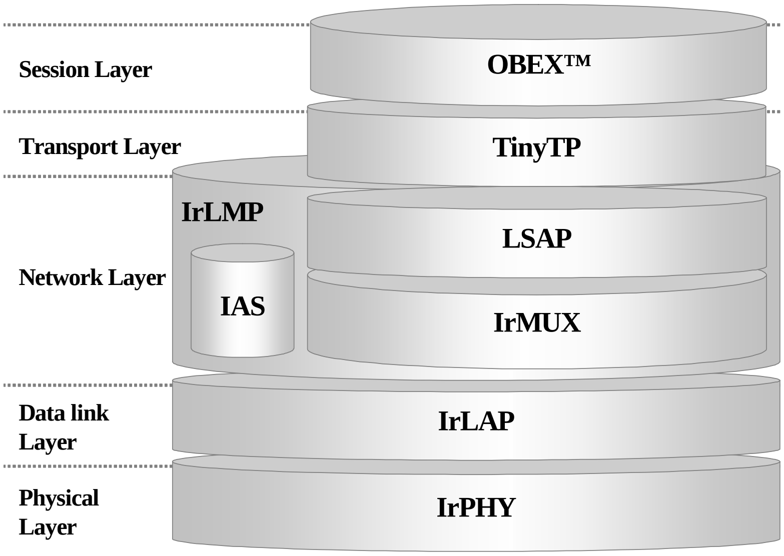


Figure 2.4 OBEX Protocol Stack

with different usage models that require connection to the Web. OBEX is just like HTTP to serve as a compact final hop to a device.

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 its pictures into a film development kiosk, or if lost can be queried (pulled) for the electronic business card of its owner. However, OBEX is not limited to quick connect-transfer-disconnect scenarios. It also allows sessions in which transfers take place over a period of time, maintaining the connection even when it is idle.

OBEX follows a client/server request-response (stop and wait) paradigm for the conversation format. The terms client and server refer to the originator/receiver of the OBEX connection, not necessarily the one who originated the low level IrLAP connection. Requests are issued by the client (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 name indicates, the “PUT” operation sends one object from the client to the server, while the “GET” operation requests that the server return an object to the client. The maximum and minimum length for both request and response packets are 512Kbit and 2048bit

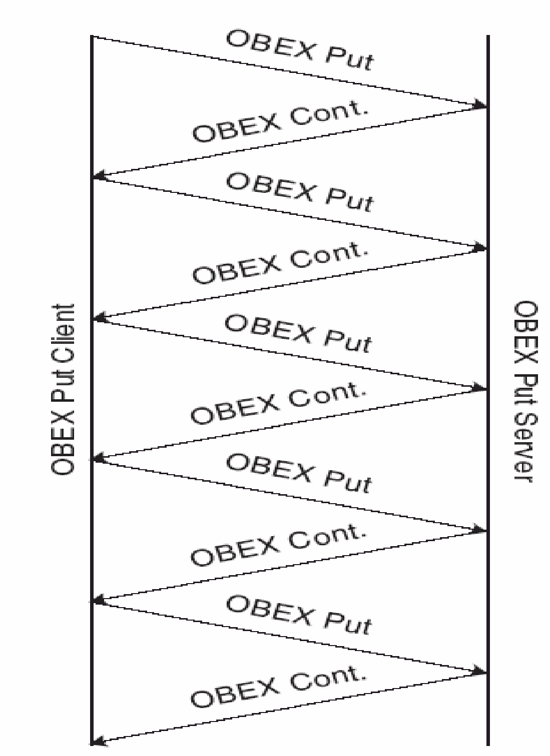


Figure 2.5 OBEX conversation format

respectively [17]. Figure 2.4 shows the OBEX and IrDA Protocol Stack, and Figure 2.5 illustrates the OBEX conversation format in brief.

2.2.2 IrBurst Protocol

IrBurst is a higher layer protocol designed over TinyTP, IrLMP, IrLAP and IrPHY for high speed exchange of large-scale information. It uses the burst transmission capability (maximum window size) of the lower layer protocols of IrDA protocol stacks for transmitting large bursts of information [20].

Usually extra header information is necessary to recognize any protocol. However, in case of IrBurst, no additional overhead is added to the information body for transmission. There is a convenient capability on the lower layer protocols to avoid the extra overhead. The capability is multiple logical channels by Logical Service Access Point (LSAP). Two channels and more are prepared for IrBurst. One channel is used for the control. The other channels are occupied for the large-scale information exchanging. Therefore, the difference of the protocol can be recognized by the identifier of the logical channel and extra header is not necessary on each packet for the transmission (Figure 2.6).

2.2.2.1 IrBurst Usage Models

IrBurst applications are divided into three kinds of usage model. First model is the case of local connection between mobile equipment and server. Second model is the case of the remote connection between mobile equipment and server through the backbone network. Third model is the case of P2P connection between mobile equipments.

2.2.2.2 IrBurst Protocol Stack

In order to analyze the performance of IrBurst protocol [20], it is necessary to understand the underlying layers of the IrDA protocol stacks. The descriptions of basic IrDA protocols are already explained in section 2.1 In case of IrBurst, IrPHY and IrLMP layer add a constant 8byte overhead but do not influence IrBurst operation in any other way. IrBurst is defined two kinds of protocol stack. A type of IrBurst is OBEX™ control type (shown in Fig.2.7); the other type is the independent control type (shown in Fig.2.8).

The OBEX™ control type uses OBEX™ as the carrier of the control messages. IrBurst requires the capability of bi-directional message exchanging in real-time. The capability is not used in the legacy OBEX™ application.

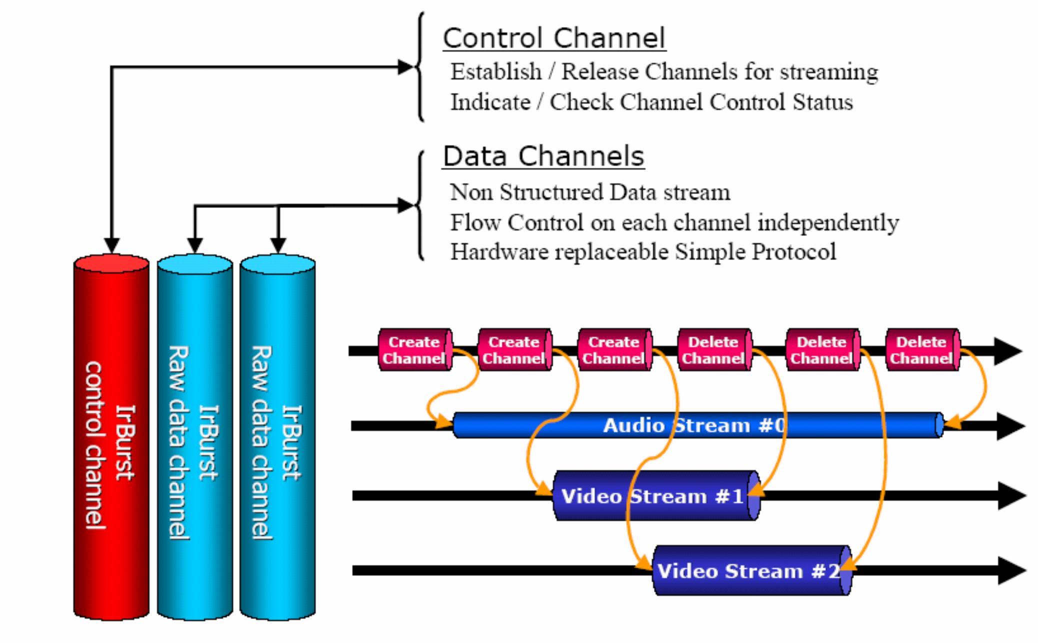


Figure 2.6 Overview of IrBurst Protocol.

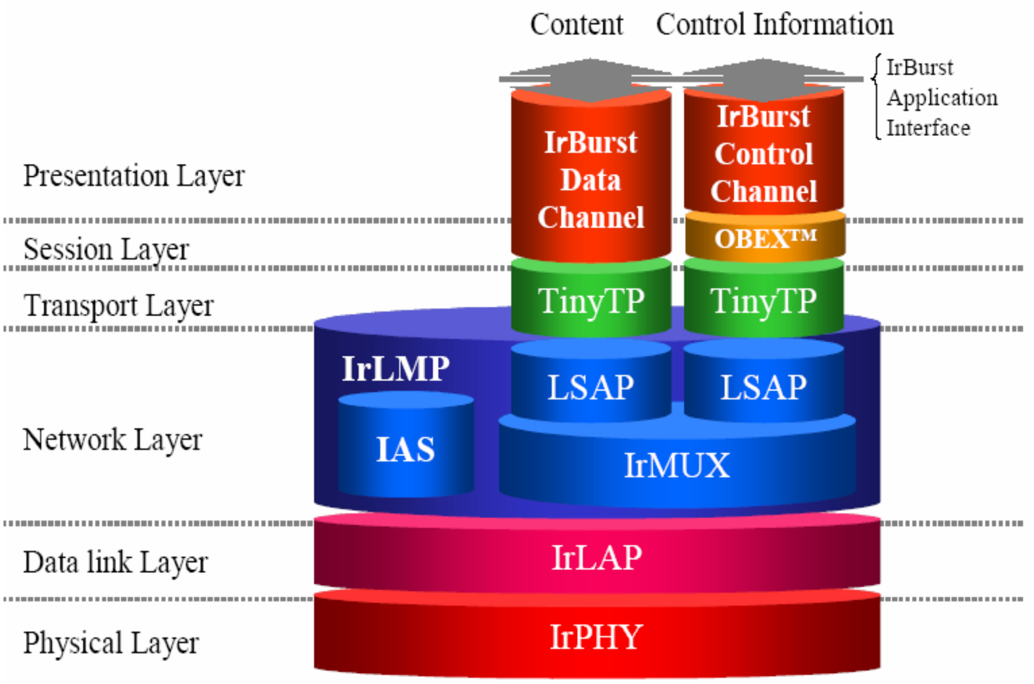


Figure 2.7 OBEXTM control type IrBurst protocol stack.

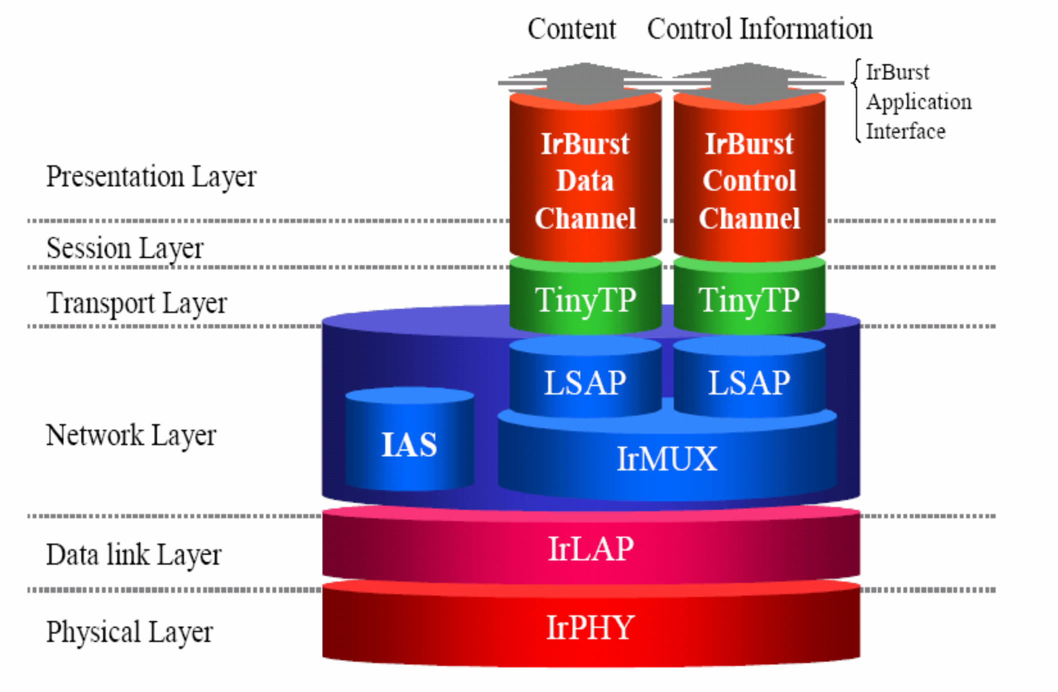


Figure 2.8 Independent control type IrBurst protocol stack.

2.2.2.3 Requirement to Lower Layer Protocols

- IrPHY: In order to transmit multimedia contents, it is necessary to improve the transmission rate. The IrBurst protocol can be applied FIR [21] and VFIR [22]. The transmission rate is close to the physical rate. However, the transmission for the multimedia contents is required higher physical rate than VFIR. If UFIR [23] will be available, it is desirable to use UFIR for large-scale information exchanging.
- IrLAP: For IrBurst protocol, the requirements of IrLAP is as follows

  - Transmission rate: The transmission rate of 4.0 Mbps or more is selected for High-speed Object Transmission.

  - Turnaround time: Since it is the reason of the transmission overhead directly, it is desirable that the turnaround time is as small as possible.

  - Window size: It is better that the window size is as large as possible. Therefore, seven frames are desirable in the case of FIR and 127 frames are desirable in the case of VFIR. If UFIR will be available, maybe some modification is necessary.
- IrLMP: The required conditions to IrLMP for IrBurst protocol are shown below.

  - IAS: LSAP for direct mapping is registered. In direct mapping, a command - message object is directly exchanged using LSAP registered to IAS.

  - LsapSel: At least, two or more LSAP(s) must be able to be used. It is desirable that LSAP can be established dynamically.
- \- TinyTP: The requirements to TinyTP for IrBurst protocol are shown below.

  - Flow control: The flow control on TinyTP is reliable. IrBurst uses this higher layer flow control to ensure the end to end data delivery for the applications. The entire data block from upper layer is split into several TinyTP payloads and this packetization is done in such a way that all packets must fit within a single IrLAP payload.

  - Segmentation and Reassembly: It is not necessary to support Segmentation and Reassembly function in the case of IrBurst protocol.
- OBEX™: The minimum requirements to OBEX™ assumed by this profile are shown below. The IR Equipment supports with both OBEX™ and IrBurst must support these conditions.

  - The conditions of an initiator: The device used as an initiator is operated as an OBEX™ client.

- The conditions of a responder: The device used as a responder is operated as an OBEX™ server.

2.2.2.4 IrBurst Data Transmission Procedure

In case of IrBurst protocol two terms-Initiator and Responder are used for the communication where the terms Initiator and Responder refer to the originator and receiver of the IrBurst connection. Requests are issued by the Initiator (the party that initiates the IrBurst connection). Once a request is issued, the Initiator waits for a response from the Responder before issuing another request. The request/response pair is referred to as an operation. “UPLOAD” and “DOWNLOAD” are the two types of operations used in OBEX. As the name indicates, the “UPLOAD” operation sends data from the Initiator to the Responder, while the “DOWNLOAD” operation requests that the Responder return the data block to the Initiator.

At first, the initiator starts to control high-speed transmission using the control channel. After negotiation, a new data transmission channel is prepared that operates as a stream (See Figure 2.6). The information sent is not peer acknowledged at the IrBurst layer unlike the OBEX protocol.

The initiator starts to control the High-speed transmission via IrBurst. Initiator uses LSAP preregistered into IAS as a control channel, exchanges the Command Message Object with a responder. Initiator orders responder to make and release LSAP as a transmission channel, to download or upload the Transmitted Information.

A responder receives the command message object from an initiator through the control channel. The channel is an LSAP beforehand registered into IAS. The responder performs processing according to the command message, and returns an answer.

2.2.3 IrSimple Protocol

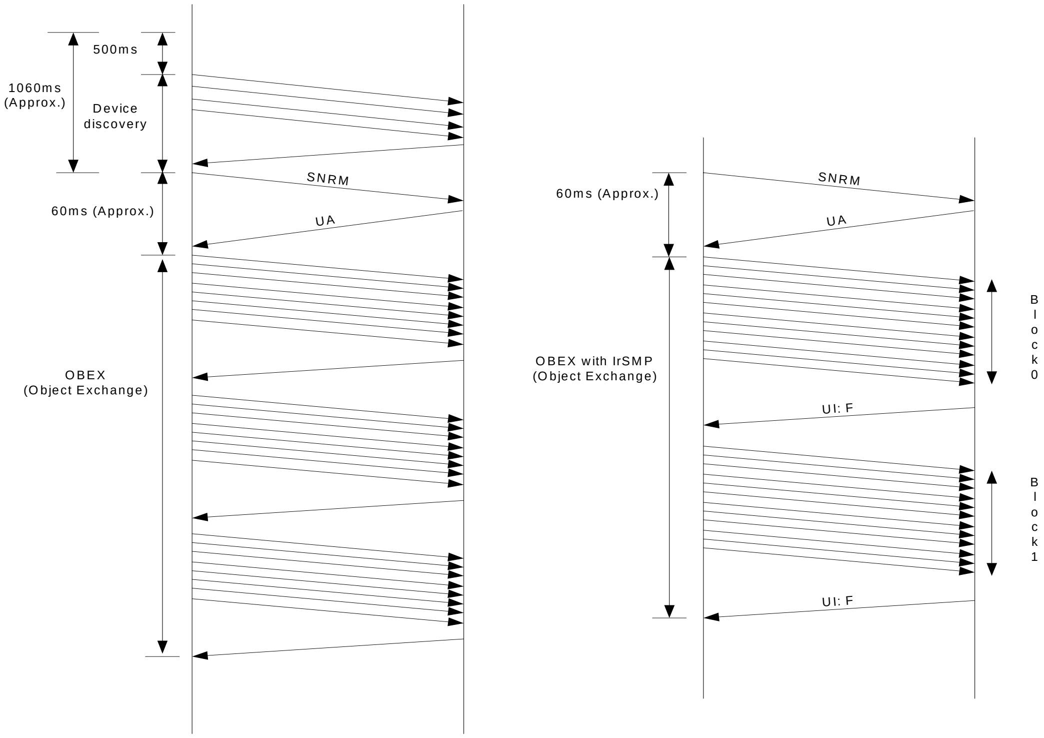
Infrared Simple (IrSimple) [24] is a high-speed infrared communications protocol, recently adopted by Infrared Data Association (IrDA), to provide simple and instant wireless communications between mobile devices and digital home appliances. This simple method of data communications helps shorten time to exchange mobile contents as the mobile contents become larger in size to accommodate the needs of high resolution picture files, music files (mp3, wma) and video files (avi, wmv, 3gp). IrSimple completes the ‘missing link’ from digital cameras, mobile phones and PDAs to color photo printer or a television for faster file transfer without any inherent complication and lack of security with other wireless technologies [24].

2.2.3.1 Overview of IrSimple Protocol

The standard IrDA protocol requires the primary station to listen the channel for 500ms to ensure that there is no other IrDA traffic within range and then broadcasts a message to initiate device discovery. Typical IrDA device discovery procedure consumes approximately 1060ms. After completing the IrDA device discovery phase, the primary can establish a connection by sending a connection request packet (SNRM) to the desired secondary station [14, 25].

In case of IrSimple data transfer, channel listening and device discovery procedures are eliminated to provide a faster method of connection establishment. The primary station sends connection request packet (SNRM command) immediately with negotiation parameters and user data when Upper layer issues connection request primitive. Figure 2.9 illustrates the standard IrDA protocol and IrSimple protocol conversation briefly.

IrSimple protocol stack (See Figure 2.10) is a layered set of protocols particularly aimed at simple and instant point-to-point infrared communications and the applications needed in that environment. Basic layers of IrDA protocol stack i.e. IrPHY, IrLAP and IrLMP add a constant 10 byte overhead but do not influence the IrSimple data transmission in any other way. In addition to these, IrLAP for IrSimple Addition [26] and IrLMP for IrSimple Addition [27] are specified to provide a faster method for discovery a device and establishing an IrLAP connection and a new higher efficiency method of exchanging data. This section describes the IrSMP(Infrared Sequence Management Protocol) [28] layer and the basic procedures of IrSimple protocol briefly.



Standard IrDA Protocol Conversation

IrSimple Protocol Conversation

Figure 2.9 IrSimple and standard IrDA protocol conversation

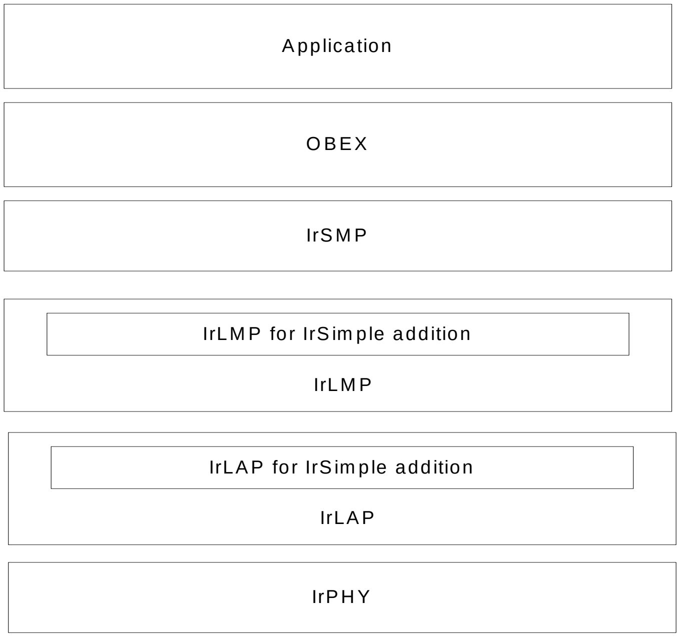


Figure 2.10 IrSimple protocol stack

2.2.3.2 IrDA Sequence Management Protocol (IrSMP)

IrSMP (Infrared Sequence Management Protocol) [28] is the layer at the IrSimple protocol that is between upper layer, OBEX or other protocol layers and lower layer, IrLMP. IrSMP provides the following functions:

- It provides the equivalent function as TinyTP [16] of existing IrDA protocol stack and provides similar SAR (Segmentation and Reassembly) services to upper layer (OBEX) and lower layer (IrLMP). SAR is the process that defines the dividing data from upper layer (OBEX) and puts them to lower layer (IrLMP) and uniting data from IrLMP and puts them to OBEX.
- IrSMP layer adds sequential number at primary station and check sequential number at secondary station for error management. By checking sequence number at the secondary station, it detects this layer any packets fallen down from lower layer, the reasons behind this (e.g. shifted bit synchronization, physical layer unrecognized BOF or STA, or etc.). It also performs error correction at bi-directional transfer mode.

2.2.3.3 IrSimple Data Transmission Procedure

Data Transfer Procedure at IrSimple uses Unnumbered Information (UI) packet. Because of UI packets, window size is ignored and continuous data transfer within Max Turn Around Time (MaxTAT) is enabled. In IrSimple procedure MaxTAT is expand 1 second. When the time that data transfer is over MaxTAT, transfer turn is moved from the primary to the secondary, after then the secondary gives transfer turn immediately.

The primary sends Unnumbered Information (UI) command setting P/F bit to 1 when the direction change parameter is true. However, when there is no data request from the upper layer of primary device and data transfer time is over MaxTAT, it sends RR command for giving transfer turn to the secondary.

The secondary sends UI command setting P/F bit to 1 when the direction change parameter is true. However, when it receives Receiver Ready (RR) command, sends RR response giving transfer turn to the primary [28].

2.2.3.4 Re-transfer Procedure

During the connection procedure, both primary and secondary station exchanges the '*BlockSize*' parameter. '*BlockSize*' of primary station is the maximum data size that it can resend whereas for secondary station it is the maximum size of data which it can receive at once. The value of this parameter for primary station is always set to the lower value than that of the received ones. At data transfer procedure, primary station can send data with lower value of *BlockSize* than that of the secondary station by lump sum. The re-transfer algorithm in data transfer procedure [28] is described below.

- Primary Station

  Primary station determines the start number of sequence number and the last number of sequence number by own *BlockSize* and received *BlockSize*. Primary station increments sequence number and transfer data to the lower layer. When primary station transfers the last packet of the block or the last packet of the data to lower layer, it sets block last (*BL)* bit to 1, and when primary station transfer a packet except it, primary station sets *BL* bit to 0. Primary station can set *BL* bit to 1 at any packets, while total size of data in the Block is equal or less than the BlockSize of secondary station. When primary station transfers the last packet of the data to lower layer, primary station sets data last (*DL)* bit to 1, and when primary station transfer a packet except it, primary station sets *DL* bit to 0. When primary station transfers the packet of the data to lower layer, primary station always sets response (*RS*) bit to 1.

- Secondary station

  Secondary station checks the sequence number of the data packet from lower layer, and when the lost sequence number is detected, secondary station maintains the sequence number for the resending request. When secondary station received the packet which *BL* bit is 1, if there is a packet which secondary station wants primary station to resend, sets *RS* bit to 0, and transfers to lower layer with set the sequence number field to the packet number which secondary station wants primary station to resend. If there is no packet which secondary station wants primary station to resend, if received *DL* bit is 0, sets *DL* bit to 0,*BL* bit to 1,*RS* bit to 1, and transfer to lower layer. At this time it puts nothing in the User Data field. If received *DL* bit is 1, secondary station does not send response packet at once. After receiving upper layer's data request, secondary station sends upper layer's user data with *DL* bit is 1, *BL* bit is 1, *RS* bit is 1, and sequence number is 0.At this time, the sequence number is ignored by primary station. When the *RS* bit of packet from secondary station is 1, if *DL* bit is 1, pass through user data to upper layer and primary station back to 1 and start transfer the next block. If *DL* bit is 0, primary station back to 1 and start transfer the next block. When the *RS* bit of packet from secondary station is 0, primary station resend from the received sequence number to the end of the block. When there is a time-out notice with Status indication from the lower layer at primary station, it resends the last sent packet (*BL* bit is 1). When there is a time-out notice with Disconnect indication from the lower layer at both primary and secondary station, it shifts to the disconnect procedure.

Chapter 3

Efficient Error Recovery Schemes for IrDA Links

3.1 Introduction

Infrared (IR) link provides a secure and a promising alternative to radio for wireless indoor applications, be it for terminals or sensors. However, IR systems also suffer from severe noise and disturbances. Since IR systems are subject to large dynamic variations of signal-to-noise ratio (SNR) at the receiver, the system suffers from unacceptably high error rates or loss of connections [29]. Error control techniques are applied in terms of Automatic Repeat Request (ARQ) or Forward Error Correction (FEC) to recover from these errors. The ARQ scheme is widely used in data communication systems because it is simple and it provides high system reliability. However, ARQ system has one major drawback of its throughput efficiency falling rapidly with channel error rate increase. Since in an FEC scheme the throughput efficiency is set by the code rate, it is constant regardless of the channel conditions. But, FEC systems also have some drawbacks. First, when a received codeword is detected in error, it must be decoded, and the decoded message must be delivered to the user regardless of whether it is correct or incorrect. Since the probability of a decoding error is much greater than the probability of an undetected error, it is hard to achieve high system reliability with FEC. Second, in order to obtain high system reliability, a long powerful code must be used and a large collection of error patterns must be corrected [30]. This makes decoding hard to implement and expensive. For these reasons, ARQ is preferred over FEC for error control in data transmissions over IrDA links.

In this chapter I focus on variant of Automatic Repeat Request (ARQ) schemes to provide reliable communication over the IrDA links. Two ARQ based schemes are proposed to enhance the robustness of next generation high speed IrDA links in case of erroneous environment.

3.2 Automatic Repeat Request (ARQ) Schemes

In an ARQ communication system, a message is first coded with a number of parity-check bits based on a high rate error-detecting code; the coded message, called a codeword, is then transmitted to the receiving end over a channel. At the receiver, parity checking (or syndrome computation) is performed on the received codeword. If the parity checking is successful (or the syndrome of the received codeword is zero), the received codeword is assumed to be error-free and is delivered to the data sink or temporarily stored in a buffer until it is ready to be delivered. At the same time, the receiver notifies the transmitter, via a return (or feedback) channel, that the codeword has been successfully received. If there is a parity failure (i.e., the syndrome of the received codeword is nonzero), errors are detected in the received codeword. Then the transmitter is requested, through the return channel, to resend the same codeword. Retransmission continues until the codeword is successfully received. With this system, erroneous data are delivered to the data sink only if the receiver fails to detect the presence of errors. Using a proper error-detecting code, the probability of an undetected error can be made very small. The three main ARQ protocols are stop-and-wait (SW), go-back-N (GBN) and selective repeat (SR) [31].

3.2.1 Stop-and-Wait (SW)

In a stop and-wait ARQ data transmission system, the transmitter sends a data block to the receiver and waits for an acknowledgment from the receiver. A positive acknowledgment (ACK) from the receiver signals that the data block has been successfully received (i.e., no errors being detected), and the transmitter sends the next data block. A negative acknowledgment (NAK) from the receiver indicates that the data block has been detected in error, and the transmitter resends the data block. Retransmissions continue until an ACK is received by the transmitter. The stop-and-wait ARQ is simple; however, it is inherently inefficient due to the idle time spent waiting for an acknowledgment for each transmitted data block.

3.2.2 Go-back-N (GBN)

In a go-back-N ARQ system, data blocks are transmitted continuously. The transmitter does not wait for an acknowledgment after sending a data block; as soon as it has completed sending one, it begins sending the next data block. The acknowledgment for a data block arrives after a roundtrip delay. The roundtrip delay is defined as the time interval between the transmission of a data block and the receipt of an acknowledgment for that data block. During this interval, *N* - 1 other data blocks have also been transmitted. When a NAK is received, the transmitter stops sending new data blocks. It backs up to the data block that is negatively acknowledged and resends that block and *N* - 1 succeeding blocks. At the receiver, the *N* - 1 received data blocks following an erroneously received data block are discarded regardless of whether or not they are error-free. Due to the continuous transmission and retransmission of data blocks, the go-back-N ARQ is more effective than the stop-and-wait ARQ. Its throughput efficiency depends on the roundtrip delay. It performs effectively on channels where the data rate is low and the roundtrip delay is small. However, it becomes inefficient for channels with high data rate and large roundtrip delay.

3.2.3 Selective repeat (SR)

In an ideal selective-repeat ARQ system, the transmitter only resends those data blocks that are detected in errors. As a result, the throughput efficiency is not affected by the roundtrip delay. This type of ARQ maintains a high throughput over a wide range of bit error rates. However, to achieve this ideal throughput efficiency, extensive buffering is required at the receiver because ordinarily data blocks must be delivered to the user in correct order. If a finite buffer is used at the receiver, buffer overflow may occur which would reduce the throughput of the system. However, if sufficient buffer store is provided at the receiver and if the buffer overflow is handled properly, the selective-repeat ARQ still significantly outperforms the other two types of ARQ in systems where data transmission rate is high and roundtrip delay is large.

3.3 Proposed Efficient Error Recovery Schemes for IrDA Links

For its inherent simplicity, Infrared Link Access Protocol (IrLAP) [14], [32] uses Go-Back-N ARQ scheme as the error control scheme for data transmission over infrared links. As discussed in section 3.2.2, this scheme signifies a waste of transmissions which results in severe deterioration of throughput performance in the case of high data rate transmission.

Hence, in this section two efficient ARQ based error recovery schemes suitable for half duplex IrDA links are proposed. The schemes vary in the way the receiver informs the transmitter about the frames to be retransmitted. The main idea for both these schemes is to store any received error free but out of sequence frames at receiver buffer thus avoiding the retransmission of these frames like pure selective repeat ARQ scheme. After storing the frames, the receiver informs the transmitter about the status of its buffer by sending bitmap information.

In Block Based Window Retransmission (BBWR) scheme, the receiver divides all the frames within a window transmission in blocks. If all the frames within a block are stored in the receiver buffer, it does not retransmit any frame of that block. However even if a single frame within a block is missing because of being received in error and hence is discarded, all the frames of the block is retransmitted. The block status information is sent from secondary station (receiver) to primary station (sender) using a supervisory frame during exchanging the turn.

In Improved Selective Repeat (ISR) scheme, the frames are not grouped into blocks and instead are assigned a corresponding bit position in a bitmap information block. The receiver sends this information using an extended supervisory frame during exchanging turn from secondary station to primary station. Upon receiving this information, the sender retransmits only the frames whose corresponding bit position in the bitmap is negatively acknowledged (by setting the value to ‘0’).

3.3.1 Block Based Window Retransmission (BBWR) scheme

3.3.1.1 Grouping frames into Blocks

All frame positions within the receiver buffer are grouped into a number of blocks that is negotiated during the connection establishment phase and the blocks form the units of acknowledgement and retransmissions. If a block consists of 2^x frames, then the sequence number of a block contains s-x bits, where s is the number of bits in the sequence number of a frame. The i^{th} block contain all frames with sequence numbers in the range $i*t$ to $(i+1)*t-1$ where t is the number of frames in a block. The secondary station always sends a supervisory frame after getting the poll bit in the final information frame. The modified supervisory frame (Figure 3.1) includes the next expected frame number (Nr) and a bit map indicating the status of the blocks in its buffer. Only when a block within the next window has all frames stored in the buffer of secondary station for being error free but out of sequence, the corresponding block is assigned as ‘1’. Based

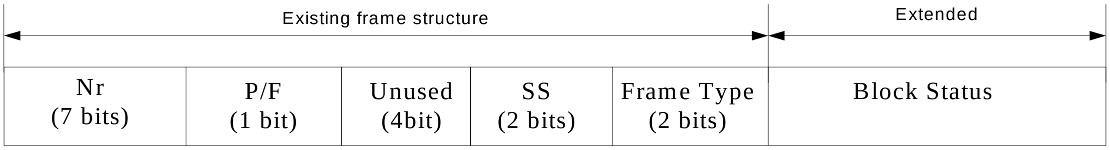


Figure 3.1 Extended control field format of S-frame

on this information, the primary retransmits or transmits only the expected blocks within the next window.

3.3.1.2 Retransmission Strategy of the Proposed Scheme

In BBWR scheme, the primary station sends Information frames (I- frame) until the number of transmitted frames equals to the maximum window size. In the last frame of the current window, the primary sets the P bit to poll the secondary and waits for the acknowledgement as well as the block status information for a certain period of time. If this time expires without any response from the secondary, it sends a Supervisory frame (S-frame) to force the secondary to acknowledge by sending the next expected frame number (E) and the block status information.

At the receiving end, when the secondary station receives an I-frame, it extracts the frame sequence number and compares with the next expected frame number. If the numbers are equal implying that the received frame is in sequence, information data is extracted and passed to the upper layer. At the same time, the secondary station increases the next expected frame number by one (modulo N where N is the total sequence number). It also checks the buffer for the next expected frame whether it is already stored there for being out of sequence in the previous transmissions. If it finds the frame stored there, information data is extracted and sent to the upper layer to release the corresponding buffer position. The next expected frame number E is also increased by 1 and the same procedure continues until it can not find the next expected frame in the buffer. But, if the received I-frame is not in sequence, the system considers two cases. Either one of the previous I-frames in current window transmission was lost due to CRC error or the frame was stored in the buffer and is sent to the upper layer during the buffer check phase for an earlier expected frame as described earlier.

For the first case, the system enters into exception state and the frame is stored in the corresponding buffer position unless it is already occupied by the same frame during earlier window transmissions. E remains unchanged and all the subsequent error free frames within the same window are stored in the similar fashion. During the exception state, no information data is passed to the upper layer and this state exists until the frame sequence number of a received frame N(S) equals E. For the other case, the secondary simply discards the frame.

When the secondary gets the poll bit in the last frame of the current window, it sends a supervisory frame to rotate the turn. The modified supervisory frame (Figure 3.1) includes the next expected frame number (*Nr*) and a bit map indicating the status of the blocks in its buffer. Only when a block within the next window has all frames stored in the buffer of secondary station for being error free but out of sequence, the corresponding block is assigned as ‘1’. Based on this information, the primary retransmits or transmits only the expected blocks within the next window. Otherwise the primary transmits or retransmits all the frames of the current window and performs like the existing GBN scheme.

The operation of my proposed ARQ scheme is briefly explained in Figure 3.2. In this case, the primary station selects the first window from frame number 0 to 14 and sends them accordingly. The secondary station receives frames error free up to frame sequence number 1 and extracts the corresponding information data to send it to the

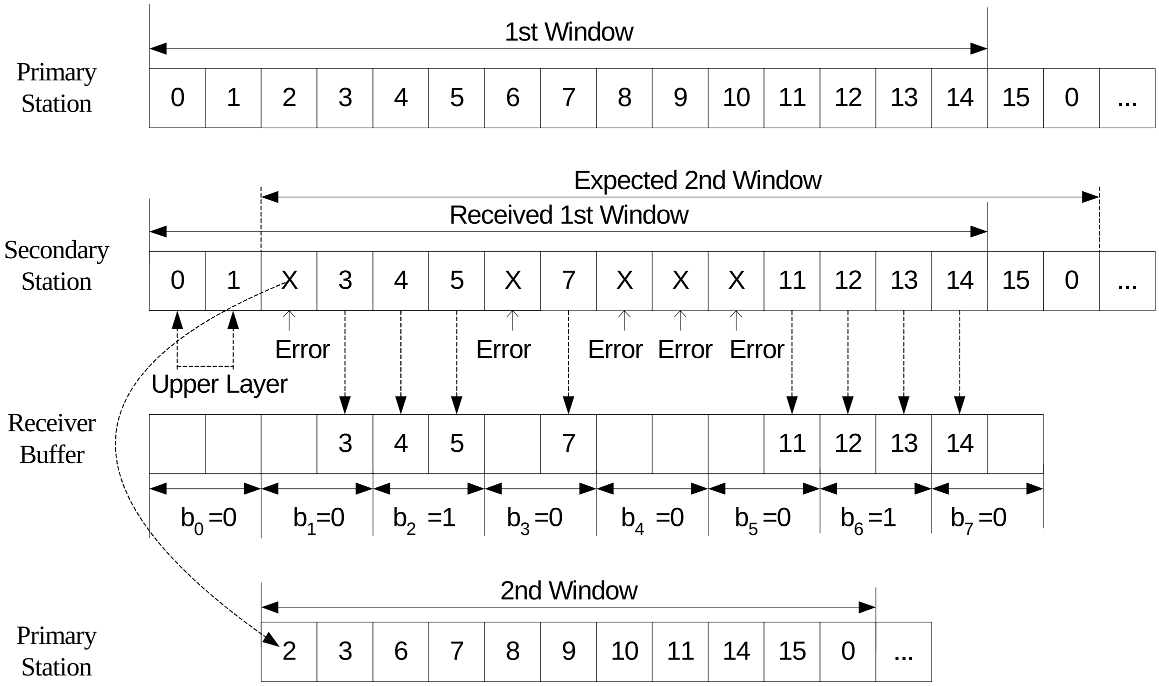


Figure 3.2 Proposed ARQ scheme for block size of 2 frames and N=15 frames.

upper layer. But it receives frame 2 erroneously and therefore discards it. When it receives frame 3 correctly but in an out of sequence manner, it stores the frame in the corresponding position of receiver buffer. Similarly it stores frame 4, 5, 7, 11, 12, 13 and 14 in the buffer but discards the erroneous frames 6, 8, 9 and 10. The buffer positions are already grouped in pre-negotiated blocks during the connection establishment phase between primary and secondary stations. With the algorithm explained earlier of this section, the status of block 2 (b_2) and block 6 (b_6) are set to ‘1’ to indicate primary station that these frames need not to be retransmitted. Since all the frames of block 0 (b_0) are sent to upper layer for being received error free and in sequence, the corresponding block status b_0 is set to ‘0’ to request primary station to send new frames (if available) as the corresponding buffer positions are empty. Moreover, block status b_1, b_3, b_4, b_5, and b_7 are also set to ‘0’ as at least one frame of these blocks was received erroneously and hence was discarded, indicating primary station to retransmit all these frames. Now, the secondary station informs the primary station the next expected frame number (frame 2) and the block status information of all blocks which is 00100010. Upon receiving this information, the primary sets the next window from frame 2 to frame 0 including new frame sequences of 15 and 0. It then only transmits or retransmits the frames whose corresponding buffer status is set to ‘0’. In this way, frames with sequence number 2, 3, 6, 7, 8, 9, 10, 11 and 14 are retransmitted. Consequently, new frames numbered 15 and 0 are also transmitted as the corresponding block b_7 and b_0 equal to ‘0’. However, primary station does not send frames whose block status is set to ‘1’ by secondary station. In this way, frames 4, 5, 12 and 13 are eliminated from being retransmitted.

3.3.1.3 Performance Analysis

In this work, the transmission of a large amount of information data from the primary to the secondary station is considered. It is assumed that the primary station always has information data ready for transmission. Information frame (*I*-frame) carry data from primary to secondary station. The secondary does not transmit information to the primary and responds only with Supervisory frame (*S*-frame). For simplicity, it is assumed that secondary station divides the receiver buffer in 8 blocks to indicate the bitmap information using 1byte unless otherwise stated. Also primary station always receives *S*-frame sent by secondary station error free. To examine the performance of the proposed BBWR scheme and the effect of different link parameters on it, a set of simulation runs was performed using the OPNET™ simulation package [33]. My OPNET simulator simulates IrLAP station behavior for 100Mbit/s half duplex infrared links using BBWR scheme and GBN scheme.

Figure 3.3 plots throughput efficiency versus BER for the proposed BBWR scheme and the existing GBN scheme for 100Mb/s link data rate (C) with minimum turn-around time (t_{tat}) 0.1ms and frame data length (l) 2KB. It shows that the BBWR ARQ scheme with window size (N) 127 frames provides an excellent performance over a wide range of bit error rates for 100 Mbit/s links. However, in the case of GBN scheme, the employment of large window size 127 results in high throughput (98%) for low BER only but renders the link operation very vulnerable to higher BER. But using the proposed BBWR scheme with this large window size of 127 frames results in the same throughput (98%) as using the GBN scheme with N=127 for low bit error rate and provides throughput in usable range for a wide range of bit error rate including high BER. The improvement of the throughput using the BBWR ARQ scheme Compared to the existing GBN scheme with N=127 is significant over a wide range of bit error rate (from10^{-7} to10^{-4}).

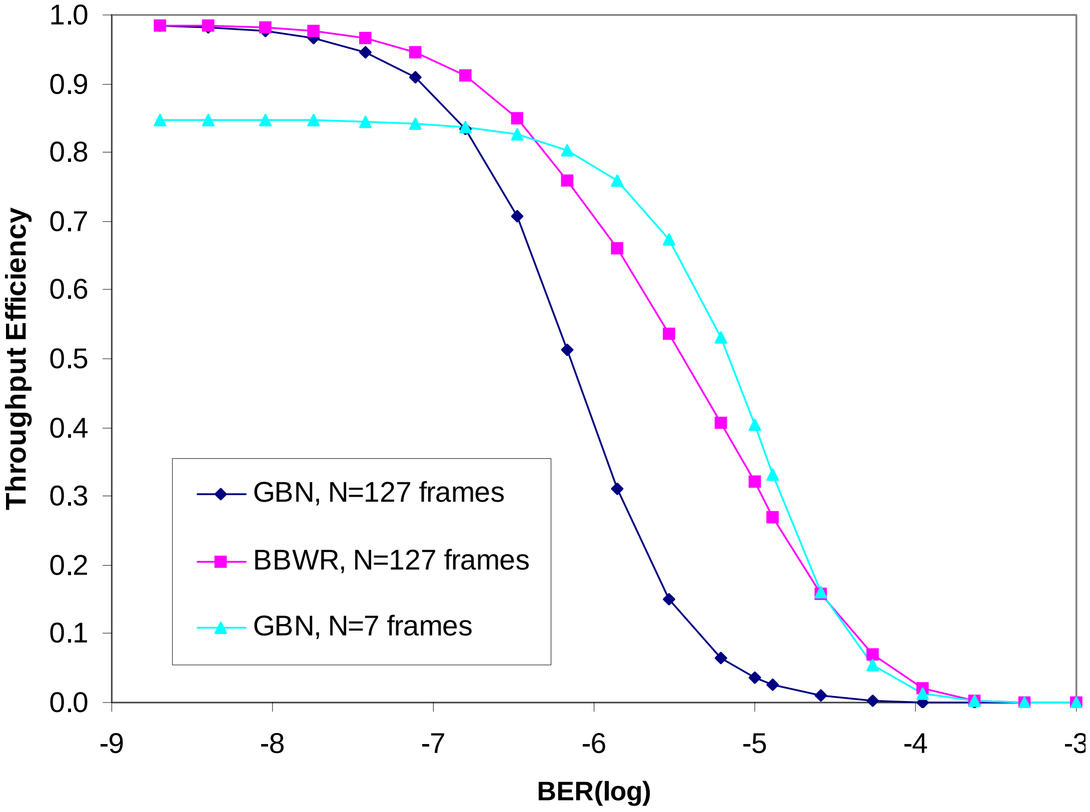


Figure 3.3 Throughput efficiency versus BER for t_{tat}=0.1ms, l=2KB and C=100Mbit/s

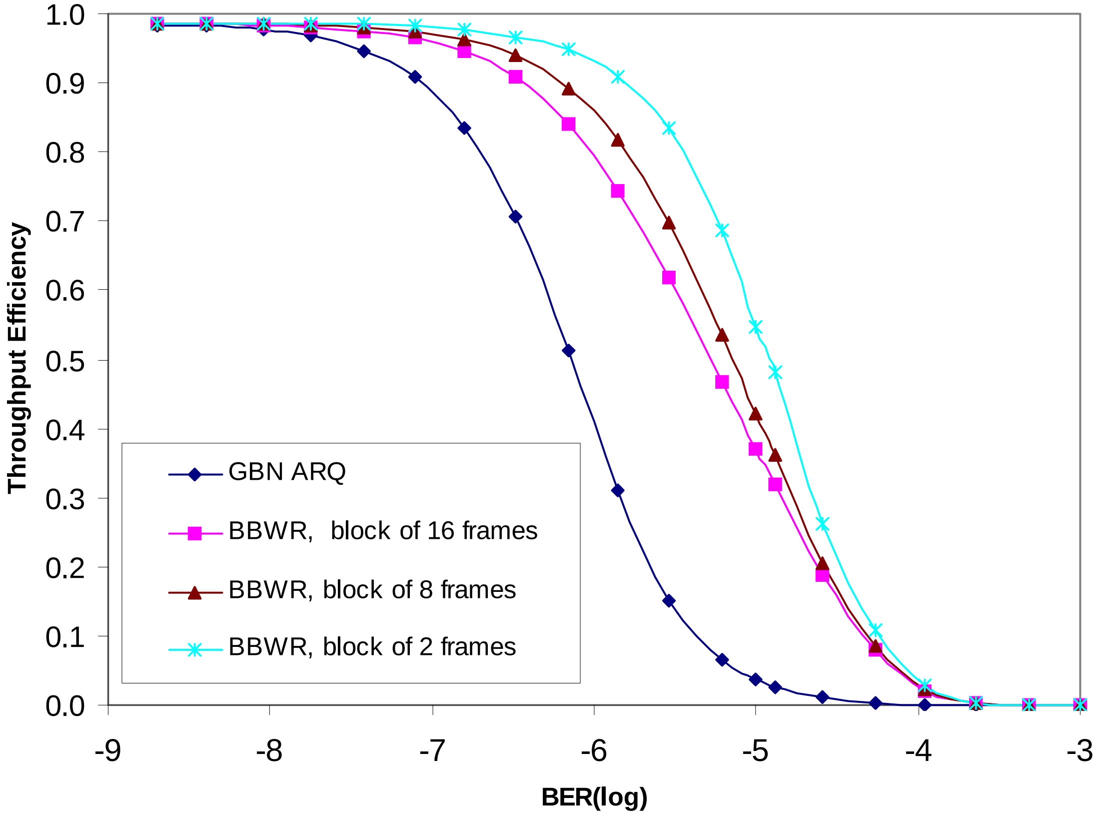


Figure 3.4 Throughput efficiency versus BER for t_{tat}=0.1ms, l=2KB and C=100Mbit/s

Figure 3.4 depicts the effect of block size on the performance of the proposed BBWR ARQ scheme for 100Mb/s link data rate (C) with varying bitmap information, minimum turn-around time (t_{tat}) 0.1ms, window size (N) 127 frames and IrLAP frame length (l) 2KB. If the block size decreases, robustness to the high BERs increases as it reduces more the retransmission of error free but out sequence frames. For the proposed scheme, block size of 2 frames provides the best performance both in terms of throughput efficiency and robustness to BER increase as it provides high throughput (almost 93%) even at high BER (10^{-6}). This throughput reduces to 80% for block size of 16 frames at the same bit error rate (10^{-6}). However, for small block size, secondary station requires more bits to form the bitmap information which may make *S*-frame large enough to get affected by erroneous channel condition.

The effect of window size on the throughput efficiency of proposed BBWR scheme for 100 Mbit/s link with t_{tat}=0.1ms and l=2KB is shown in Figure 3.5.

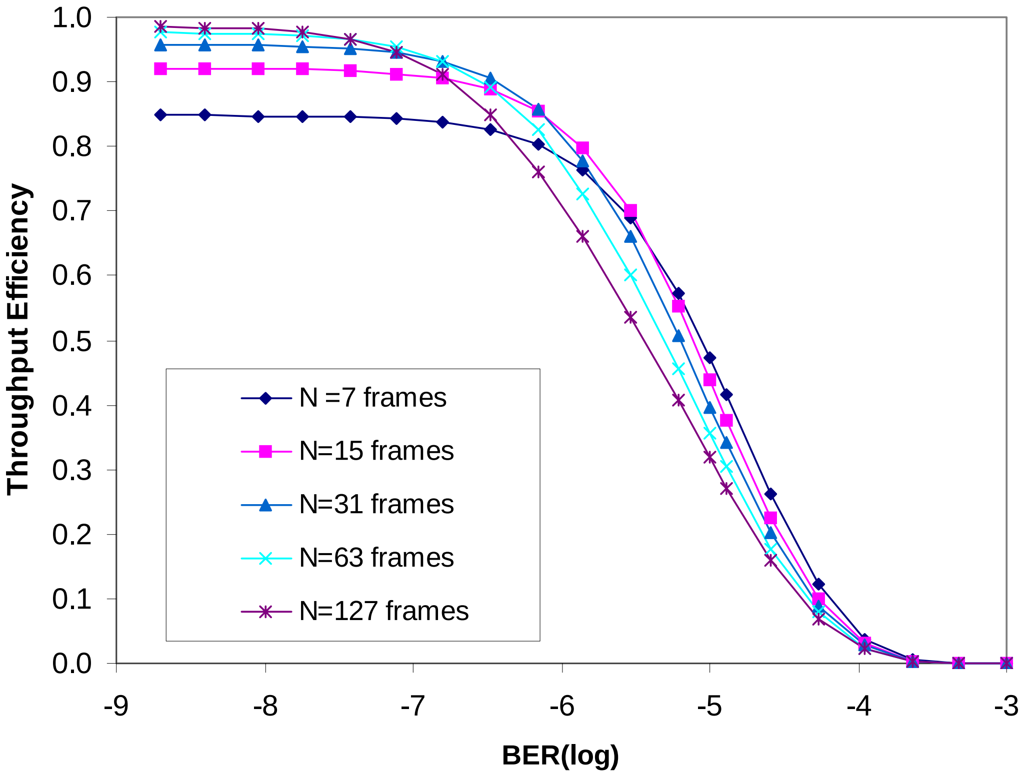


Figure 3.5 Throughput efficiency versus BER for BBWR ARQ scheme with t_{tat}=0.1ms, l=2KB and C=100Mbit/s.

Excellent throughput efficiency at low BERs can be achieved for window size N=127 frames but the throughput is very much vulnerable to the high BER. The figure also depicts that window size N=31 frames can achieve almost the same throughput at low BERs and the performance degradation due to high BER is considerably less. So, for My proposed scheme, I consider the window size as 31 as it provides an excellent performance over a wide range of bit error rate compared to the throughput using other window sizes. Another advantage of this lower window size is that it needs less sequence number, only 31 to be sufficient for maximum sequence number. As a result, the buffer size both at the primary and the secondary station decreases. Henceforth I only consider the window size N=31 frames to analyze the performance my proposed scheme.

Figure 3.6 shows throughput efficiency of BBWR scheme over 100Mbit/s links using N=31 for different minimum turnaround time (t_{tat}) varying from 1.0ms to 0.001ms. The figure shows that turn around time t_{tat} has significant effect on the throughput efficiency at low BERs.

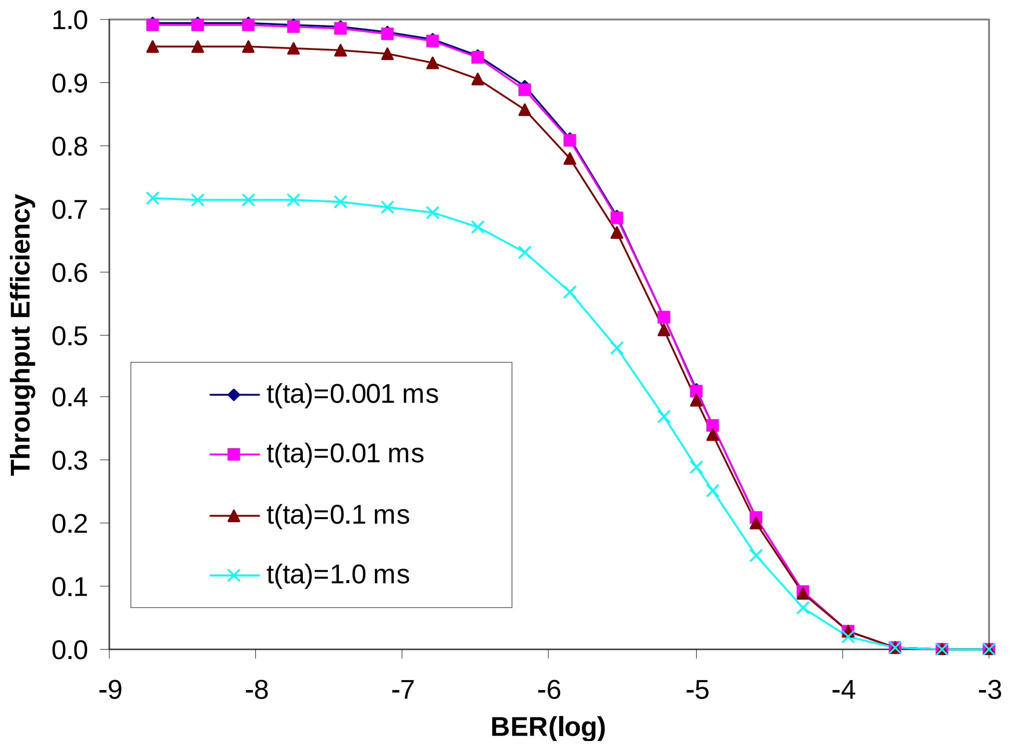


Figure 3.6 Throughput efficiency versus BER for BBWR with *N*=31 frames, *l*=2KB and *C*=100Mbit/s.

There is an immense improvement in throughput efficiency for decrease in t_{tat} from 1.0ms to 0.1ms. More than 30% increase in throughput efficiency is achieved at low BERs. Though the improvement is not at the same ratio for further decrease in t_{tat} from 0.1ms to 0.01ms, the throughput efficiency is achieved over 99% due to this decrement at BER= 10^{-8}. The figure also reveals that the throughput performance is almost the same if t_{tat} is decreased from 0.01ms to 0.001ms. Based on this, I deduce that reducing t_{tat} from 0.01ms to 0.001ms does not improve throughput significantly and is unnecessary.

Finally, Fig. 3.7 compares the throughput performance of BBWR at *N*=31 frames with GBN at *N*=127 frames and at *N*=7 frames. For this comparison I have considered the minimum turn around time (t_{tat}) to be 0.1ms and frame data length (*l*) to be 2 KB. Although the existing GBN scheme shows good throughput performance using *N*=127 frames for low bit error rate, it is very much vulnerable to high BERs. On the other hand, it shows less vulnerability to high BER when it operates at *N*=7 frames but the throughput efficiency remains 10% to 15% lower for low BERs (approximately 10^{-9} to 10^{-7}). However in the case of Block Based Window Retransmission ARQ scheme operating

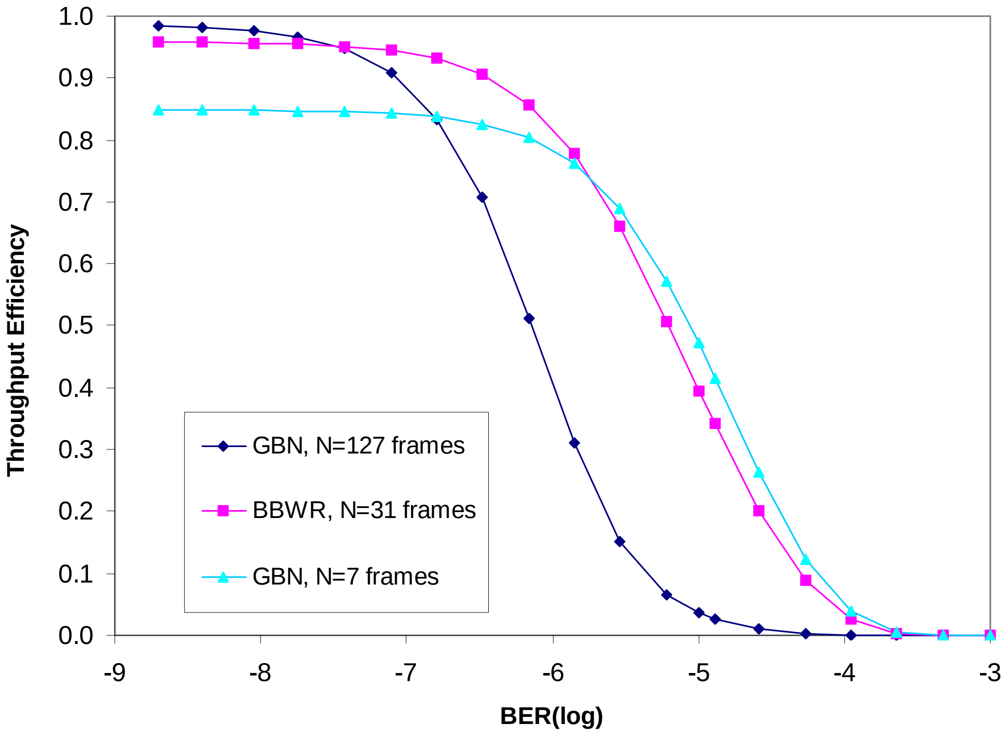


Figure 3.7 Performance comparison of BBWR with N=31 frames, t_{tat}=0.1ms and l=2KB.

with N=31 frames can achieve some great extent of performance in terms of throughput and robustness to vulnerability. For low BER (approximately 10^{-9}), it achieves throughput efficiency at the satisfactory level of 96% which is only 0.02% less than the throughput efficiency of GBN with N=127 at same BER. Moreover the throughput efficiency of BBWR remains in the usable range of over 80% for a wide range of BER including high BER (approximately 10^{-9} to 10^{-6}). At higher BERs (10^{-6} to 10^{-4}), it provides almost the same robustness as using GBN scheme with N=7.

3.3.2 Improved Selective Repeat (ISR) Error Recovery Scheme

3.3.2.1 Retransmission strategy

The new ARQ scheme differs from the previous ARQ scheme (section 3.3.1) in the way of transferring its bit map information and retransmission strategy. Instead of dividing all the frames into groups to form units of acknowledgement and retransmissions, all frames are acknowledged individually. In this scheme, all error free but out of sequence frames received at the secondary station (receiver) are stored in the corresponding positions of a buffer and a bitmap information indicating the status of the buffer is

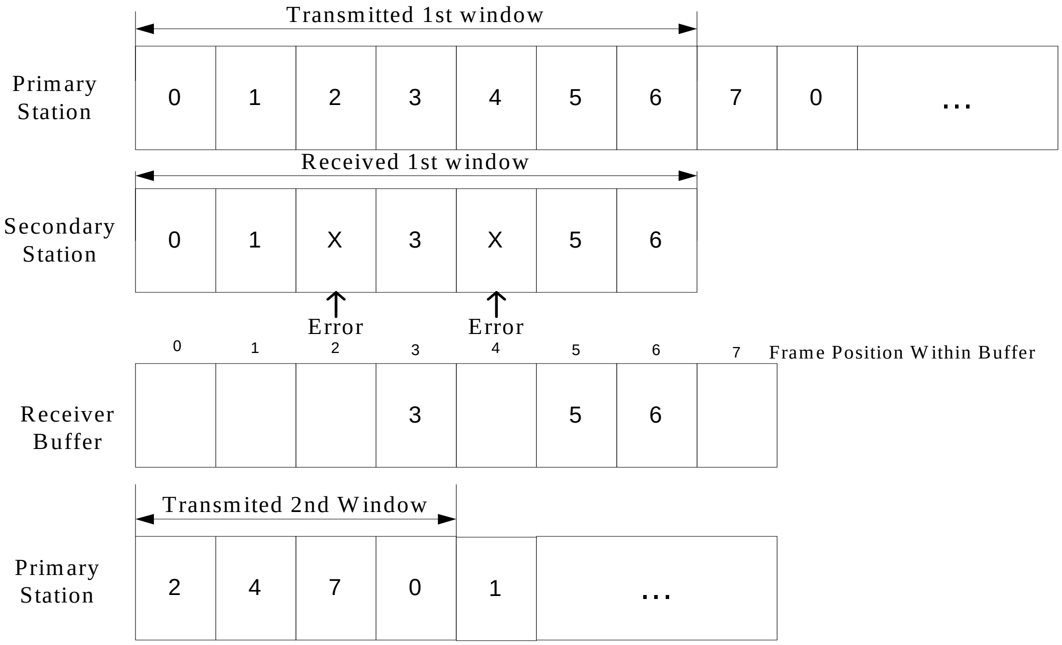


Figure 3.8 Proposed Improved Selective Repeat (ISR) ARQ scheme.

updated accordingly. By default all the corresponding bits are set to ‘0’. When a frame is stored in the buffer, the corresponding bit of the bitmap information is set to ‘1’. After receiving the last Information frame which has the poll bit set as 1, the secondary station (receiver) sends the next expected frame number as well as the bitmap information using a modified supervisory frame. Upon receiving this information, the primary station decodes the bitmap information and adjusts its next window of frames that starts with the next expected frame of secondary station. If any frame within the next window is already stored in receiver buffer, which is indicated by the bit map information, is ignored by the primary station to be retransmitted. As a result, fewer frames are retransmitted in case of error recovery which significantly improves the system throughput efficiency.

The operation of the proposed ISR ARQ scheme is briefly explained in Figure 3.8. Suppose the window size is 7. Now, the primary station selects the first window from frame number 0 to 6 and sends them accordingly. The secondary station receives frames 0 and 1 error free, and extracts the corresponding information data to send it to the upper layer. But frame 2 is received erroneously and therefore is discarded. Next it receives frame 3 correctly but in out of sequence manner, it stores the frame in the corresponding position of receiver buffer. Similarly it stores frame 5 and 6 in the buffer but discards the erroneous frame 4. Now the corresponding bit of frames 3, 5 and 6 are set to ‘1’ to inform primary station that these frames need not be retransmitted. Now, the secondary station informs the primary station the next expected frame number (frame 2) and the bitmap information of all frames which is 00010110. Upon receiving this information, the primary sets the next window from frame 2 to frame 0 including new frames 7 and 0. It then only transmits or retransmits the frames whose corresponding buffer status is set to '0'. In this way, frames with sequence number 2 and 4 are retransmitted. Consequently, new frames numbered 7 and 0 are also transmitted as the corresponding bit of bitmap information equal to '0'. However, primary station does not resend frames 3, 5 and 6.

3.3.2.2 Experiment setup

An experiment environment is set up to investigate the performance of the proposed ISR ARQ scheme for high-speed (100Mb/s) IrDA links. Two Stanley UFIR(100Mb/s) [23] demonstration boards act as the primary and secondary stations. Figure 3.9 shows the UFIR demonstration board. A 32bit soft-core processor, UFIR controller, packet buffers and so on are implemented into one FPGA device to operate as an UFIR demo board. The board has 44MHz system clock. The transmitter buffer size is 28kB and the receiver buffer size is 14kB. A frame tester for IrDA physical layer is placed on the top of the transmitting station to act as a noise generator. The tester connected with a

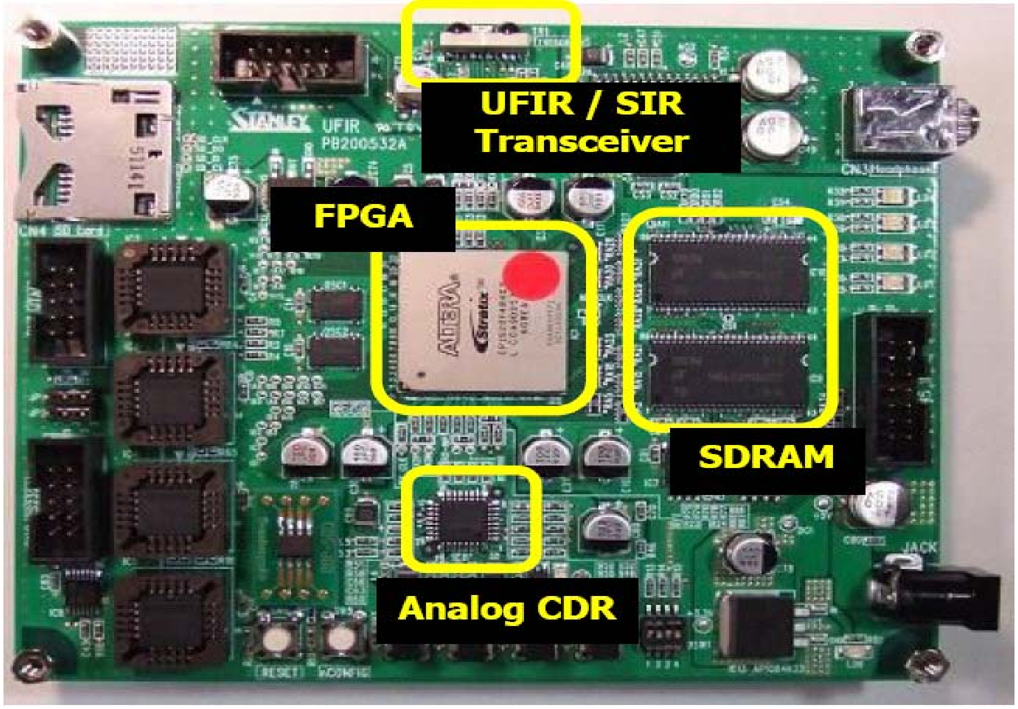


Figure 3.9 UFIR Demo boards

computer generates 1Byte data packet at 115.2kbps baud rate. The interval time for next packet generation can be controlled using a computer. If it generates the packet during data transmission between the stations, it effectively corrupts the transmitted frames. As the result, the frame needs to be retransmitted. The receiving station is connected with a logic analyzer to measure the operational time of every single code. As a result, the total time required to execute the logic of the proposed ISR scheme can be directly measured. Figure 3.10 shows the experiment setup.

3.3.2.3 Simulation model

To examine the performance of the proposed ISR ARQ scheme and the effect of different link parameters on it, a simulation model is developed using OMNeT++ [34]. The simulator simulates IrLAP station behavior for high speed half duplex infrared links using ISR scheme and GBN scheme.

3.3.2.4 Model validation

In order to validate the simulation model, experiment results are compared with that obtained from the simulation. Figure 3.11 plots throughput efficiency versus noise interval time for GBN and proposed ISR ARQ scheme at UFIR (100Mbps) links. It shows that simulation results almost coincide with the experiment results for both the ARQ schemes.

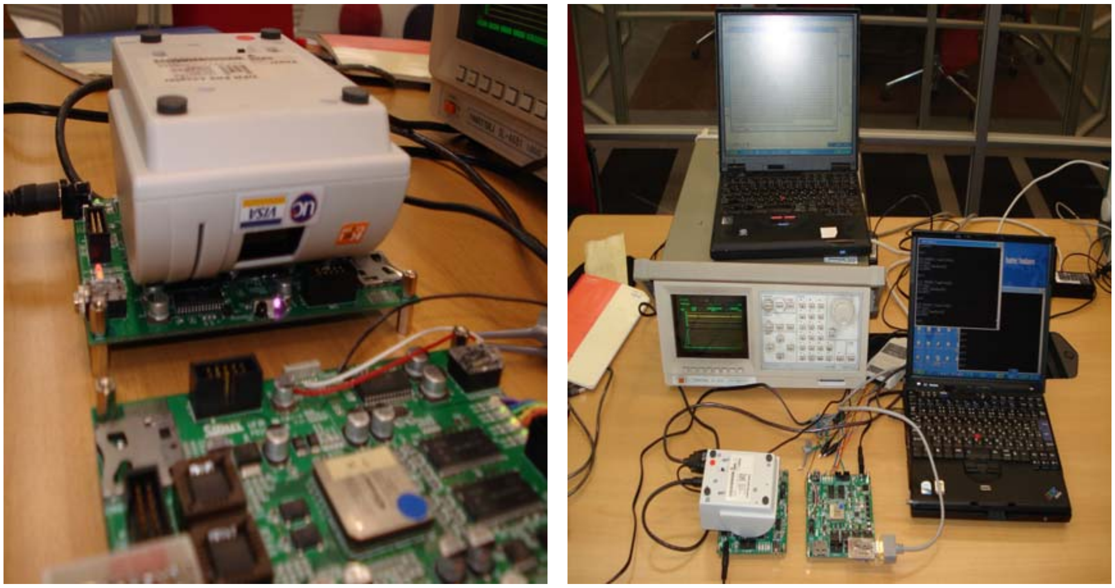


Figure 3.10 Experiment setup

3.3.2.5 Performance Evaluation

The effect of increasing frame length (L) in throughput efficiency for a 1Gbps link with GBN ARQ scheme at t_{tat}=0.001ms and N=127 frames is shown in Figure 3.12. It shows that frame length increase results in 15% throughput increase for low BER but renders the link very vulnerable to BER increase.

Figure 3.13 compares the throughput efficiency of the existing GBN and proposed ISR ARQ scheme for link data rate (C) 1Gbps with minimum turn-around time (t_{tat}) 0.001ms, and data length (l) 32KB. It shows that the throughput is significantly increased for a range of BER values (from10^{-9} to10^{-6}) including high BER if the proposed ARQ scheme is employed. For low BER it provides 5% throughput efficiency increase as that using the GBN ARQ scheme. However, if the BER increases it provides a significant improvement in the throughput compared to the GBN scheme.

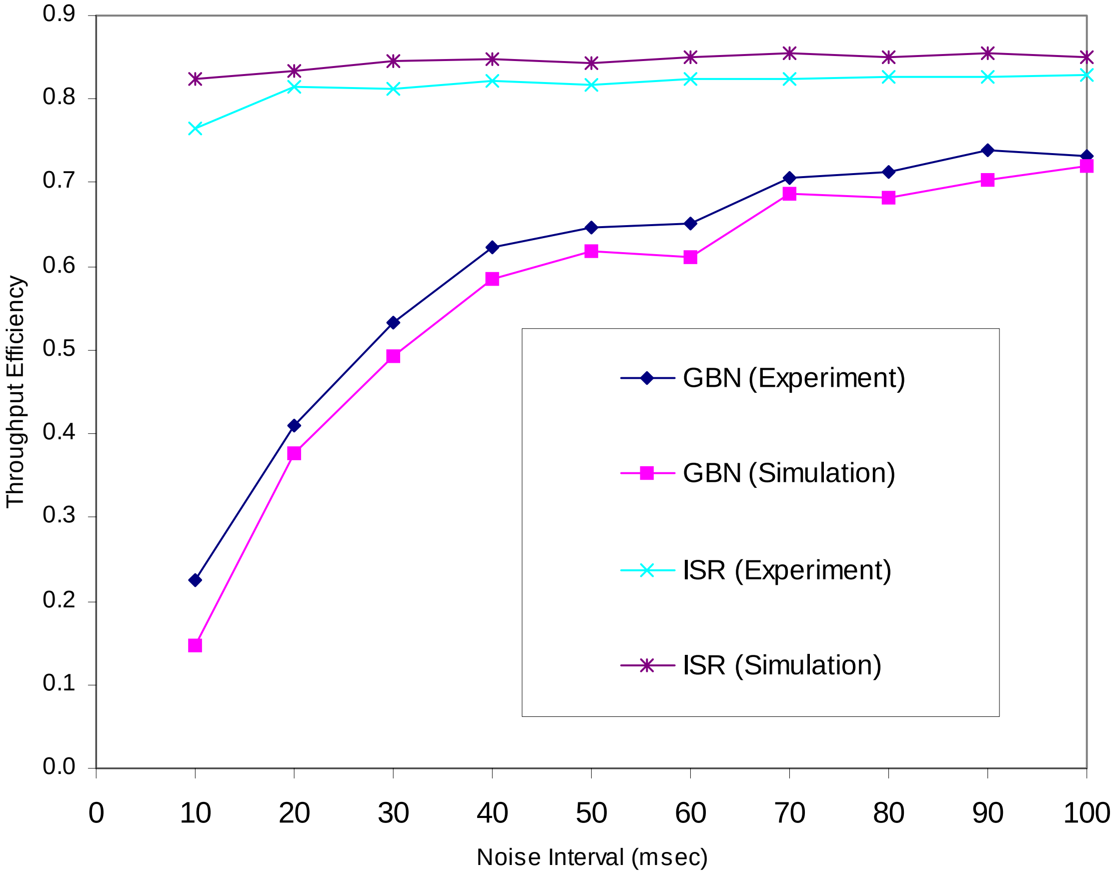


Figure 3.11 Throughput efficiency: experiment versus simulation

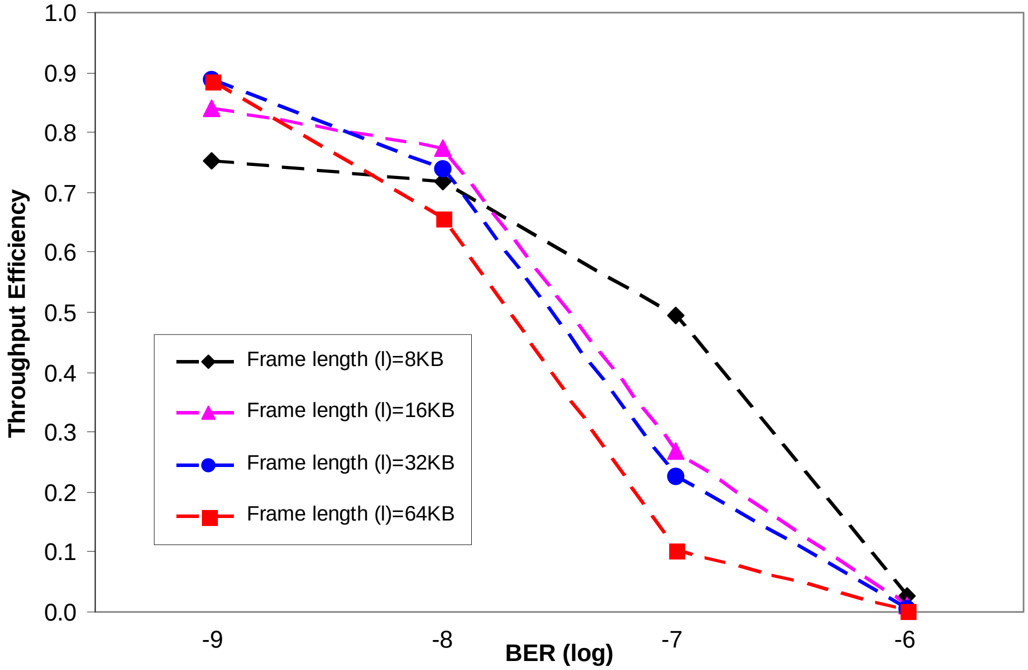


Figure 3.12 Throughput efficiency versus BER for GBN ARQ scheme with C = 1Gbps, t_{tat} = 0.001ms, N = 127 frames.

3.4 Conclusion

Two efficient error recovery schemes are proposed to enhance the robustness of IrDA links in case of erroneous environment. In many indoor environments, there exists intense noise, arising from sunlight, incandescent lighting and fluorescent lighting. With the increase of data transfer rates beyond 100Mbit/s over the half duplex infrared links, a robust automatic repeat request (ARQ) scheme is therefore necessary to cope with this erroneous environment. For its inherent simplicity, Go-Back-N ARQ scheme is deployed for data transmission over IrDA links. However at high data rate, this ARQ scheme requires some lower layer parameters, such as window size and frame size, be adapted to the corresponding optimum values for the correspondent Bit Error Rate (BER). But adaptive approaches always add a significant amount of complexity to the system. Hence, in this work, I present two error recovery schemes named as Block Based Window Retransmission (BBWR) ARQ scheme and Improved Selective Repeat (ISR) scheme for high speed infrared communication operating with a fixed size receiver buffer

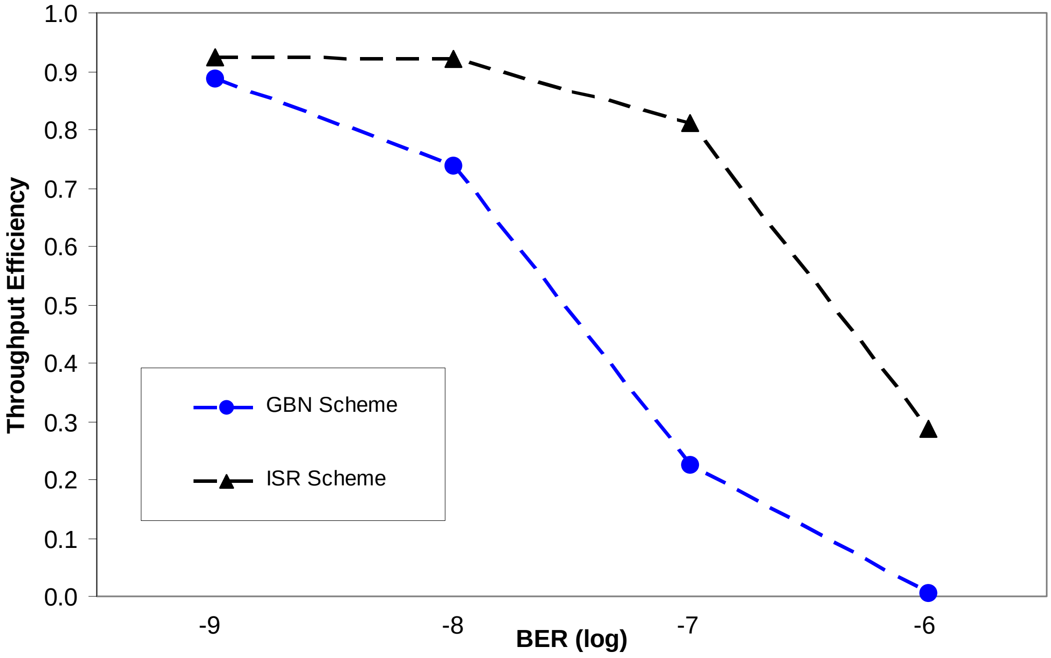


Figure 3.13 Throughput efficiency versus BER for GBN scheme and ISR ARQ scheme with C = 1Gbps, t_{tat} = 0.001ms, N = 127 frames.

without adapting parameters to the optimum values. BBWR scheme is more suitable for large window size while ISR scheme is preferable for system with large frame length thereby keeping the window size small. Simulations results for BBWR scheme are presented which show that the proposed scheme significantly outperforms the existing GBN ARQ scheme and provides almost 50% improvement in throughput efficiency at high BER. Experimental results for ISR scheme are presented as the proof of concept. Furthermore a simulation model is designed and verified by comparing the experimental results for ISR scheme. The proposed ISR scheme is then examined by simulation for future GigaIR (1Gbps) links which also show significant improvement in performance at high BERs.

Chapter 4

Performance Evaluation of IrBurst Protocol for Large Data Block Transmission over IrDA Links

4.1 Introduction

In this chapter, I undertake a detailed study of the performance of IrBurst protocol [20] operating on top of the IrDA protocol stack. The IrBurst issue has been examined in [35], [36] but the result presented in the analysis is not sufficient for the details and complete performance analysis of IrBurst. Also the performance improvement of IrBurst protocol compared to existing IrOBEX protocol [17] is not presented in any of the works. It is therefore of interest to develop a systematic and comprehensive analysis of IrBurst protocol for large data block exchange over high speed IrDA links and to compare the performance with existing protocol. Furthermore, only a few works have been done to investigate the IrDA protocol performance for the future high speed IrDA links (100Mb/s). In [37], the authors have examined the IrDA protocol performance at 100Mb/s and concluded that to maximize throughput at this high-speed, parameters such as window size and frame length values, should be adapted to the corresponding optimum values for the bit error rate (BER). However, this adaptive approach adds a significant amount of complexity to the resource-limited mobile devices. By considering the lower layers in presence of errors and the complexity associated with adaptation, in this chapter, I investigate the IrBurst protocol behavior in detail and derive a comprehensive and more realistic model for IrBurst as well as investigate more effective ARQ schemes rather than optimization of parameters to maximize IrBurst throughput at high data rate [38], [39].

The rest of the chapter is organized as follows: In section 4.2, a mathematical model is derived for IrBurst which allows derivation of throughput taking into account the lower IrDA protocol stack. The section also includes the validation of mathematical model by comparing the results obtained by equation with results derived from simulation and the performance comparison of IrBurst with existing OBEX protocol performance for various data rates at different bit error rates. A detailed study of the effect of processor speed on IrBurst performance and the effect of physical and link layer parameters on the IrBurst throughput efficiency for high-speed IrDA links, especially in presence of transmission errors, is also presented in the same section. In an effort to improve the performance of IrBurst when large window size or large frame length is used, the ARQ schemes proposed in the previous chapter are investigated in section 4.3. Simulation result presented in this section agrees on the effectiveness of the proposed ARQ schemes for improving IrBurst throughput efficiency in case of erroneous environment. Finally section 4.4 presents the conclusions.

4.2 Mathematical Modeling and Performance Evaluation

In order to calculate IrBurst throughput efficiency (TE) over IrDA protocol stacks, a mathematical analysis is carried out. In this model, I calculate IrBurst throughput efficiency by calculating total number of bits required to exchange for transmitting the data block of a given size. IrBurst throughput efficiency is derived both in error free transmission and in presence of transmission errors. Variables used in the mathematical analysis are defined in Table 4.1. To simplify the analysis, I make the following assumptions:

- The data blocks are sent in the IrBurst “UPLOAD” operation mode. However, the derived model can also be modified in a straightforward manner for “DOWNLOAD” operation mode.
- Ony one IrBurst application is active in the sender.
- IrBurst is considered only in connected data transmission mode i.e., no control message is sent during data transmission.

4.2.1 Protocol Mapping of IrBurst and OBEX

In case of sending data using IrBurst [20], the whole data block is sent down to the TinyTP [16] layer as data stream. IrBurst layer does not add any extra overhead to the data. TinyTP layer is the layer that splits the entire data block into several TinyTP payloads and this packetization is done in such a way that all packets must fit within a single IrLAP [14] payload. After adding its fixed overhead of 1byte, TinyTP sends the

Table 4.1 Mathematical Model Variables for IrBurst protocol

Symb.	Parameter Description	Unit
C	Link data rate	bit/s
p	Frame error probability	
p_b	Link bit error rate (BER)	
L	Data block size	bit
N	IrLAP window size	frames
l	Payload size of IrLAP frame(Frame length)	bit
l_{PHY}	Physical layer overhead	48bit
l_{LAP}	S-frame size/ I-frame (IrLAP) header	24bit
l_{LMP}	IrLMP layer header	16bit
l_{TTP}	TinyTP layer header	8bit
l_t	Payload size of TinyTP packet	bit
t_{tat}	IrLAP minimum turnaround time	sec
b_{tat}	Equivalent bits of IrLAP turnaround time: $C \times t_{tat}$	bit
l_{OBEX1}	Overhead of first OBEX packet	bit
l_{OBEXn}	Overhead of all subsequent OBEX packets	48bit
P_{REQ}	OBEX request packet size	bit
P_{RES}	OBEX response packet size	bit
T_{TA}	OBEX turnaround time	sec
b_{TA}	Equivalent bits of OBEX turnaround time: $C \times T_{TA}$	bit

data packet down to the IrLMP [15] layer. In case of IrBurst, IrLMP adds a constant 3byte overhead but do not influence IrBurst operation in any other way. A fixed overhead of 9bytes is added to the data packet when it is passing through the IrLAP layer and IrPHY [13] layer. Since each packet fits within a single IrLAP frame, the data packet is fitted into the IrLAP information frame (I-frame) and the primary station sends a window of I-frames, polls the receiver (secondary station) by setting the Poll/Final (P/F) bit in the last frame of the window and solicit a response.

Because point-to-point infrared communication is inherently half-duplex, the secondary station cannot send the acknowledgement of the received frame immediately rather it waits for the completion of current window transmission. If the frame with P bit set is received and as the present case assumes only the “UPLOAD” operation mode, it awaits a minimum turn- around time (t_{tat}) to allow for the hardware recovery latency and transmits the acknowledgement of received frame using supervisory frame (S-frame) with the Poll/Final (P/F) bit set to pass the transmission control to the primary station. Figure 4.1 illustrates the protocol mapping of a large data block in IrBurst down to the link layer of the stacks (IrLAP).

4.2.1.1 IrBurst Throughput Efficiency in Error free Transmission

After transmitting one full window of frames, primary station turns to receiving mode while secondary station turns to transmitting mode for acknowledging the received frames. Before the secondary switches its mode, it must wait for a minimum turnaround time (t_{tat}) period in order to allow the receiver of primary station to get ready. During this minimum turnaround time, the secondary station processes the TinyTP packet extracted from last IrLAP frame to get TinyTP feedback about the available TinyTP buffer size. For simplicity, I assume that the processing time to get this TinyTP feedback at IrLAP layer of secondary station is always less than t_{tat} throughout this article unless otherwise stated.

After the time span of t_{tat} expires, secondary station transmits the supervisory frame (S-frame) and then again switches to receiving mode to allow the primary station sending the remaining frames. However, the primary also waits a minimum turnaround time (t_{tat}) span in order to allow the receiver of secondary to get ready. Finally, it starts to transmit remaining frames again.

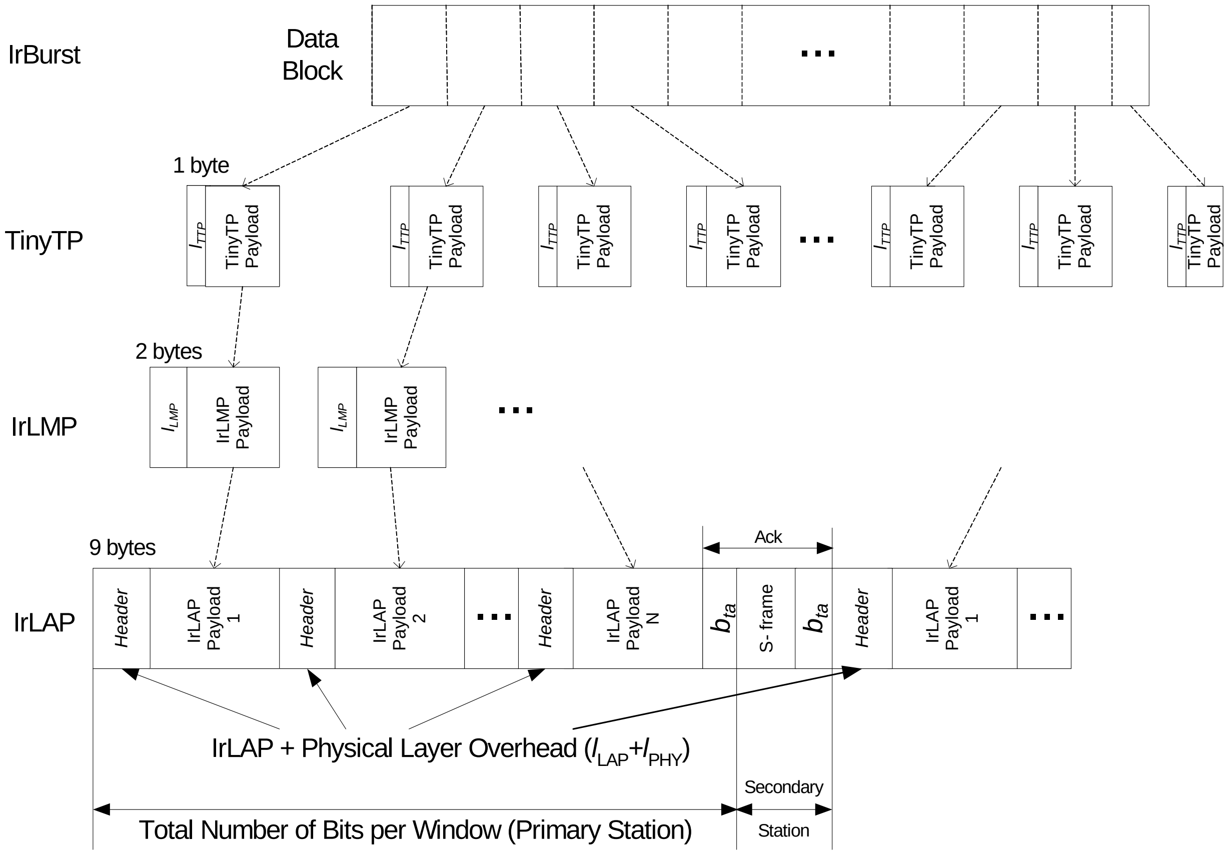


Figure 4.1 Mapping IrBurst, TinyTP, IrLMP to IrLAP frames

As illustrated in Figure 4.1, all TinyTP packets must fit within a single IrLAP payload, therefore the payload of each TinyTP packet (l_t) is:

$$l_t = l - l_{TTP} - l_{LMP} \qquad (4.1)$$

I consider fixed overheads of 2 bytes and 1 byte for IrLMP and TinyTP respectively. Total TinyTP packets required for data block of size L is:

$$n = \left\lceil \frac{L}{l_t} \right\rceil \qquad (4.2)$$

Since each packet fits within a single frame and each window has N frames, I divide n by N to yield the total number of windows (W) required to transmit the block. Because IrLAP turnarounds are dependent upon windows, therefore I multiply by b_{tat} and finally by 2 to consider the total overhead due to turnaround time in terms of bit.

$$O_{tat} = 2 * b_{tat} * \left\lceil \frac{n}{N} \right\rceil \qquad (4.3)$$

To include the overhead due to lower layer, I add the overhead of each layer and multiply it by the total number of packets to yield total overhead for lower layer.

$$H = \left(l_{TTP} + l_{LMP} + l_{LAP} + l_{PHY}\right) * n \qquad (4.4)$$

The secondary station sends Supervisory frame (S-frame) [14] of size $l_{LAP} + l_{PHY}$ to give the acknowledgement (ACK). Therefore the total overhead due to acknowledgement (ACK) considering all the windows is:

$$O_{ACK} = \left(l_{LAP} + l_{PHY}\right) * \left\lceil \frac{n}{N} \right\rceil \qquad (4.5)$$

As illustrated in the figure, for transmitting the content of size L, in addition to this content extra bits are also exchanged which accounts for the overhead of lower layer, acknowledgement and turnaround time effect. By adding up all the overheads and content size, the total number of bits required to transmit the block is:

$$B = O_{tat} + H + O_{ACK} + L \qquad (4.6)$$

The throughput efficiency (TE), which is defined as the ratio of data block size (in bits) to total number of bits required to transmit that block, is therefore given by:

$$TE = \frac{L}{B} \qquad (4.7)$$

4.2.1.2 IrBurst Throughput Efficiency in presence of transmission errors

In case of error, the number of bits required to transmit the same data content is increased due to retransmission of erroneous frame and all the subsequent frames in the window.

For frame error probability p and window size N, the average number of frames correctly transmitted in one full window, N_{corr} is given by [32]:

$$Ncorr = \frac{(1-p)\left(1-(1-p)^N\right)}{p} \qquad (4.8)$$

where,

$$p = 1 - \left(1 - p_b\right)^{l + l_{LAP} + l_{PHY}} \qquad (4.9)$$

Now the total number of windows required to transmit the same data content is also increased due to error. In this case, the total number of windows is:

$$W_{err} = \frac{L}{\left(l - l_{TTP} - l_{LMP}\right) * N_{corr}} \qquad (4.10)$$

Therefore, the overhead due to turnaround time in presence of error is given by:

$$O_{taterr} = 2 * b_{tat} * \lceil W_{err} \rceil \qquad (4.11)$$

The total overhead due to ACK in case of error is:

$$O_{ACKerr} = \left(l_{LAP} + l_{PHY}\right) * \lceil W_{err} \rceil \qquad (4.12)$$

Since the erroneous frame and the subsequent frames of the window are retransmitted in case of error, I subtract total number of windows to transmit the block without any error W (n/N) from W_{err} to yield the extra windows which are retransmitted for error and multiply by N (window size) to find out the total number of retransmitted frames. Finally I multiply this with frame size (l + l_{LAP} + l_{PHY}) to yield the total number of bits which are retransmitted for error:

$$b_R = \left(l + l_{LAP} + l_{PHY}\right) * \left(N * \left(W_{err} - \left(\frac{n}{N}\right)\right)\right) \qquad (4.13)$$

Furthermore, if the last frame of the window is not correctly received, the P bit is also lost and the receiver does not respond as it is unaware of link reversion. The primary waits for an F-timer expiration and sends S-frame forcing the receiver to respond. Assuming S-frames are always received correctly, the P bit loss incorporates an additional frame of size l_{LAP} + l_{PHY} transmission, the associated turnaround time and the delay due to F-timer. As I considered only UPLOAD mode, the delay due to F-timer is assumed ((l+ l_{LAP} + l_{PHY}) +$2b_{tat}$). Finally, since the frame error probability is p, I multiply this by p and therefore the total number of bits required to transmit in case of P bit lost for all windows is:

$$b_{Plost} = \lceil W_{err} \rceil * p * \left(\left(l_{LAP} + l_{PHY}\right) + b_{tat}\right) \qquad (4.14)$$
$$+ \lceil W_{err} \rceil * p * \left(\left(l + l_{PHY} + l_{LAP}\right) + 2b_{tat}\right)$$

Summing (4.4), (4.11), (4.12), (4.13), (4.14) and data block size (L) yields the total number of bits required to transmit the same block in presence of transmission errors:

$$B_{err} = H + O_{taterr} + O_{ACKerr} + b_R + b_{Plost} + L \qquad (4.15)$$

The resulting equation for IrBurst throughput efficiency (TE) in presence of errors is therefore given by:

$$TE = \frac{L}{B_{err}} \qquad (4.16)$$

4.2.1.3 IrBurst Throughput Efficiency considering processor speed

In order to investigate the effect of processing speed on IrBurst throughput efficiency in presence of transmission errors in details, I calculate IrBurst throughput efficiency considering processor speed in this section. I mainly focus on the processing time associated with higher layer (TinyTP layer) rather than processing time at lower layer (IrLAP layer) as in case of IrBurst protocol flow control is performed by TinyTP layer. For this, the processing time of last TinyTP packet which is stripped of last IrLAP frame received at secondary station and the processing time of TinyTP acknowledgement packet sent to IrLAP layer indicating TinyTP buffer size are calculated first.

If the processor speed is defined as P_{speed} kHz which is a b_p bit (16 bit) processor and assuming each b_p bit takes P_{cycle} CPU cycles (2 cycles) in average, I can calculate the processing time of last TinyTP packet (T_{ttp}) as follows:

$$T_{ttp} = \frac{P_{cycle} * l_t}{P_{speed} * b_p * 10^3} \qquad (4.17)$$

Upon processing this packet, TinyTP layer will send TinyTP acknowledgement to IrLAP layer to provide the information of TinyTP buffer size available at secondary station and IrLAP will incorporate this information with its supervisory frame. Time required to process this TinyTP acknowledgement packet (T_{ttpack}) is:

$$T_{ttpack} = \frac{P_{cycle} * (l_{TTP} + l_{LMP})}{P_{speed} * b_p * 10^3} \qquad (4.18)$$

If $t_{tat} > T_{ttp} + T_{ttpack}$ holds true, the secondary waits exactly t_{tat} time before sending the supervisory frame. Otherwise, it has to wait $T_{ttp} + T_{ttpack}$ time and sends the supervisory frame upon receiving the feedback from TinyTP layer. In this case, the overhead due to processing delay (b_{delay}) in terms of bits is as follows:

$$b_{delay} = (T_{ttp} + T_{ttpack}) * C \qquad (4.19)$$

Since primary station also needs to wait t_{tat} time upon receiving supervisory frame from secondary, I add this b_{delay} with b_{tat} and finally multiply with W_{err} that is defined in (4.10), to calculate the total overhead due to turnaround for all windows in presence of transmission errors (O_{delay}).

$$O_{delay} = (b_{delay} + b_{tat}) * \lceil W_{err} \rceil \tag{4.20}$$

By adding up all the overheads from (4.4), (4.12), (4.13), (4.14), (4.20) and content size (L), the total number of bits required to transmit the block in presence of transmission errors considering processing times (B_{ps}) is:

$$B_{ps} = H + O_{delay} + O_{ACKerr} + b_R + b_{Plost} + L \tag{4.21}$$

Therefore IrBurst throughput efficiency (TE) considering processor speed and transmission errors is:

$$TE = \frac{L}{B_{ps}} \tag{4.22}$$

4.2.1.4 OBEX Throughput efficiency for error free transmission

For deriving mathematical model of OBEX [17] throughput efficiency, I consider the OBEX 'PUT' operation mode. Figure 4.2 illustrates the way in which OBEX packetizes a large object of same size (L) for transmission and the protocol mapping of this packet down to the link layer of the stacks based on [40]. The first packet typically contains some extra information (l_{OBEX1} - l_{OBEXn}) of the object (such as name, length, etc.) but all the subsequent packets contains fixed header of length l_{OBEXn} only.

The total number of OBEX packets for an object of size L is given by:

$$n1 = \left\lceil \frac{(L + (l_{OBEX1} - l_{OBEXn}))}{(P_{REQ} - l_{OBEXn})} \right\rceil \tag{4.23}$$

Since TinyTP and IrLMP add their overhead to each OBEX packet when it passes down to the link layer and payload of each IrLAP frame is l, therefore the total IrLAP frames required for each OBEX packet is:

$$n2 = \left\lceil \frac{(P_{REQ} + l_{TTP} + l_{LMP})}{l} \right\rceil \tag{4.24}$$

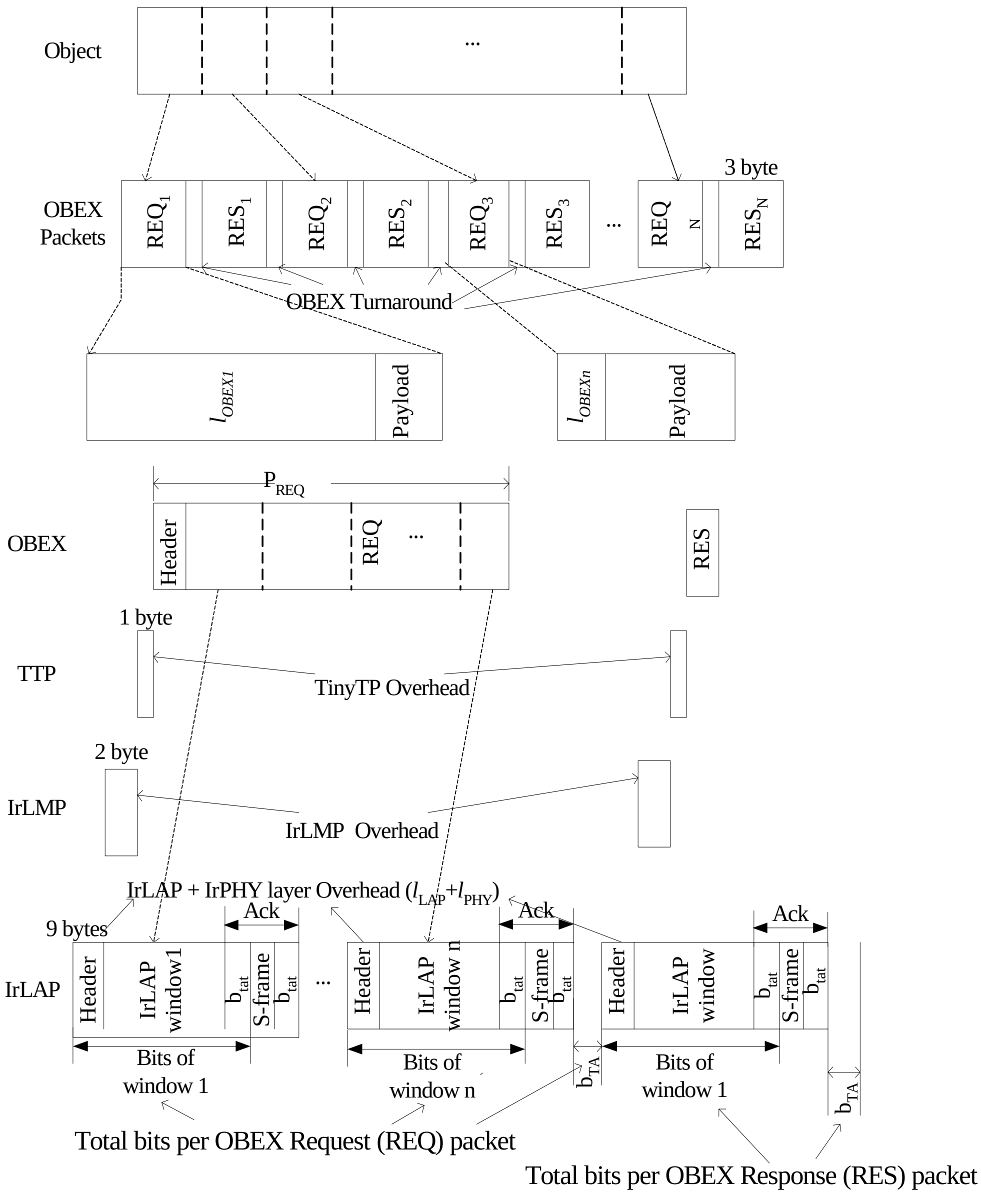


Figure 4.2 OBEX packetization and protocol mapping

Because each window has N frames, the number of windows required to transmit each OBEX packet is:

$$w1 = \left\lceil \frac{n2}{N} \right\rceil \qquad (4.25)$$

The OBEX standard requires that each request (REQ) packet must be acknowledged by a response (RES) packet. Because of the half duplex nature of IrDA, subsequent request packets cannot be transmitted until the corresponding response is received. Therefore, to transmit the object in REQ packets, both the overhead associated with request packet and response packet are considered.

Each request/response pair generates two OBEX turnarounds. Considering this higher layer turnaround effect associated with all the REQ packets, I multiply b_{TA} with n1. To this I add the total overhead due to IrLAP turnaround time ($2*n1*w1*b_{tat}$), total overhead for all layers associated with all OBEX request (REQ) packets ($n1*(n2*(l_{PHY}+l_{LAP})+l_{LMP}+l_{TTP}+l_{OBEXn})$) and overhead due to ACK ($n1*w1*(l_{LAP}+l_{PHY})$). I also include the extra information related to the object in the first packet. This yields the total overhead associated with all the request (REQ) packets (H_{REQ}) for transmitting the object.

$$\begin{aligned} H_{REQ} &= (n1*b_{TA}) + (2*n1*w1*b_{tat}) + (n1*w1*(l_{LAP}+l_{PHY})) \\ &+ (n1*(n2*(l_{PHY}+l_{LAP}) + l_{LMP}+l_{TTP}+l_{OBEXn})) \\ &+ (l_{OBEX1} - l_{OBEXn}) \end{aligned} \tag{4.26}$$

Similarly, the total overhead associated with all the response (RES) packets (H_{RES}) to transmit the object considering no OBEX turnaround time associated with the last response packet is:

$$\begin{aligned} H_{RES} &= ((n1-1)*b_{TA}) + (2*n1*b_{tat}) + (n1*(l_{LAP}+l_{PHY})) \\ &+ (n1*(l_{PHY}+l_{LAP}+l_{LMP}+l_{TTP}+l_{OBEXn})) \end{aligned} \tag{4.27}$$

By adding up all these overheads and the size of the object, the total number of bits required to transmit the object of size L is given by:

$$B_{OBEX} = H_{REQ} + H_{RES} + L \tag{4.28}$$

Therefore OBEX throughput efficiency (TE) is:

$$TE = \frac{L}{B_{OBEX}} \tag{4.29}$$

4.2.1.5 OBEX Throughput Efficiency in presence of transmission errors

Since in case of error, the number of bits required to transmit the same data content is increased due to retransmission of erroneous frame and all the subsequent frames in the window, the number of windows required to transmit each OBEX packet is also increased. Now using (equation 4.8), the total number of windows required to transmit each OBEX packet in case of error is:

$$w1err = \left\lceil \frac{n2}{N_{corr}} \right\rceil \tag{4.30}$$

The total overhead associated with all request packets for transmitting the same object is also increased due to error. Therefore, using (eq. 4.26), the total overhead associated with all Request packets in presence of transmission errors (H_{REQerr}) is:

$$H_{REQerr} = H_{REQ} + \left(\left(\left(\frac{n2}{N_{corr}} \right) - \left(\frac{n2}{N} \right) \right) * N * \left(l + l_{LAP} + l_{PHY} \right) \right) + \lceil w1err \rceil * p\left(\left(l_{LAP} + l_{PHY} \right) + b_{tat} \right) \tag{4.31}$$

The RES packets are small enough to be assumed error free. Therefore, the total overhead associated with all the response (RES) packets (H_{RESerr}) to transmit the object in case of error is same as that in error free transmission (H_{RES}).

$$H_{RESerr} = H_{RES} \tag{4.32}$$

By adding up all these overheads and the size of the object, the total number of bits required to transmit the object of size L is given by:

$$B_{OBEXerr} = H_{REQerr} + H_{RESerr} + L \tag{4.33}$$

Therefore OBEX throughput efficiency (TE) in presence of transmission errors is:

$$TE = \frac{L}{B_{OBEXerr}} \tag{4.34}$$

4.2.2 Performance Evaluation and Results

I have developed a simulation model for IrBurst over IrDA protocol stacks using OPNET^{TM} simulation package [33]. In my model, all the IrDA protocol details are implemented according to the IrDA specifications. I used point to point connection between primary and secondary station. Bit errors are typically the only source of transmission failure on this point-to-point link. All simulations are run for 1000 seconds of simulated time and the first 10% of the data is discarded. The performance measurements are logged at the secondary station.

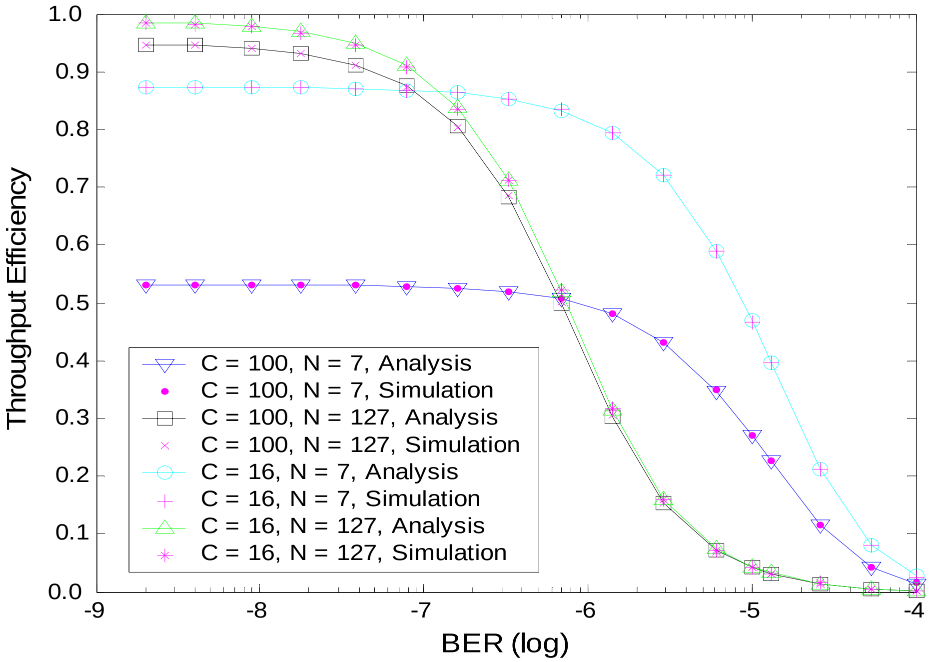


Figure 4.3 Throughput efficiency: analysis versus simulation

4.2.2.1 Model Validation

In order to validate the mathematical model of IrBurst, simulation results are compared with that obtained from the equation. Figure 4.3 plots throughput efficiency versus bit error rate (BER) for different window size (N) values with t_{tat}=1.0ms, l=2KB and L=100MB at two different data rates of 100Mb/s and 16Mb/s. The figure shows that analytical results practically coincide with the simulation results for both the data rates. All simulation results in the plot are obtained with a confidence interval of 98%. In my thesis, only the validation of IrBurst model with transmission error (equation 4.16) is presented.

4.2.2.2 IrBurst –OBEX Performance Comparison

In this section, I apply my model to compare the performance of IrBurst with OBEX for high-speed exchange of a large data block at various data rates and various bit error rates (BERs). For this experiment, the data content size (L) is 100MB, the IrLAP frame length (l) is 2KB and the OBEX turnaround time (T_{TA}) is 1.0ms.

4.2.2.2.1Performance Comparison of IrBurst and OBEX for Various Data Rates

Using equations (4.7) and (4.29) in Figure 4.4, IrBurst and OBEX throughput efficiency (TE) are examined over a range of data rates with window size of 127 frames and for three different minimum turn around time of 0.1ms, 1ms and 10ms. The figure shows that while a longer minimum turnaround time always degrades throughput efficiency both for IrBurst and OBEX; the effect is more pronounced at higher data transfer rates. At low data rate and very low turnaround time, IrBurst and OBEX have almost the same throughput efficiency. However, as the data rate increases or minimum turn around time increases, the improvement in throughput efficiency (TE) for IrBurst over OBEX also increases.

For large value of turnaround time (t_{tat}=10ms), IrBurst provides 10% more TE compared to OBEX for FIR (C=4Mbps) [21] link while it has 50% more TE than OBEX for UFIR (C=100Mbps) link [23]. Even for a very low turnaround time of 0.1ms, IrBurst has almost 30% improvement in throughput efficiency compared to OBEX at C=100Mbps. The figure also shows that, IrBurst with low turnaround time (t_{tat}=0.1ms) provides excellent throughput efficiency over a range of data rates including high data rate (C=100Mbps).

Therefore, IrBurst protocol significantly outperforms existing OBEX protocol for exchange of large data blocks especially at high data rates.

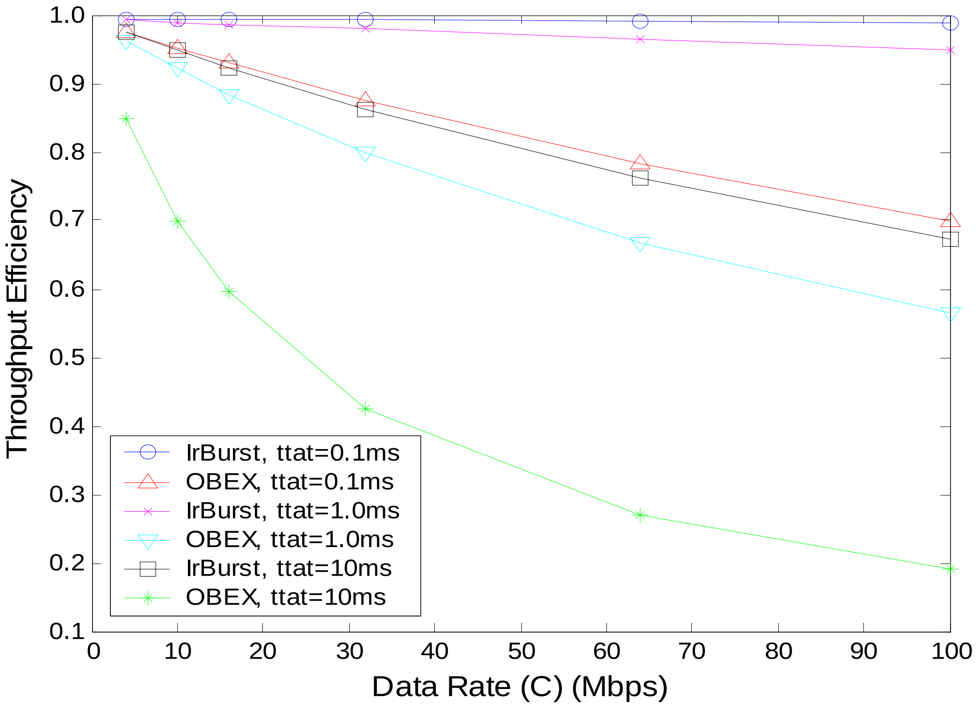


Figure 4.4 Performance comparison of IrBurst and OBEX for various data rates.

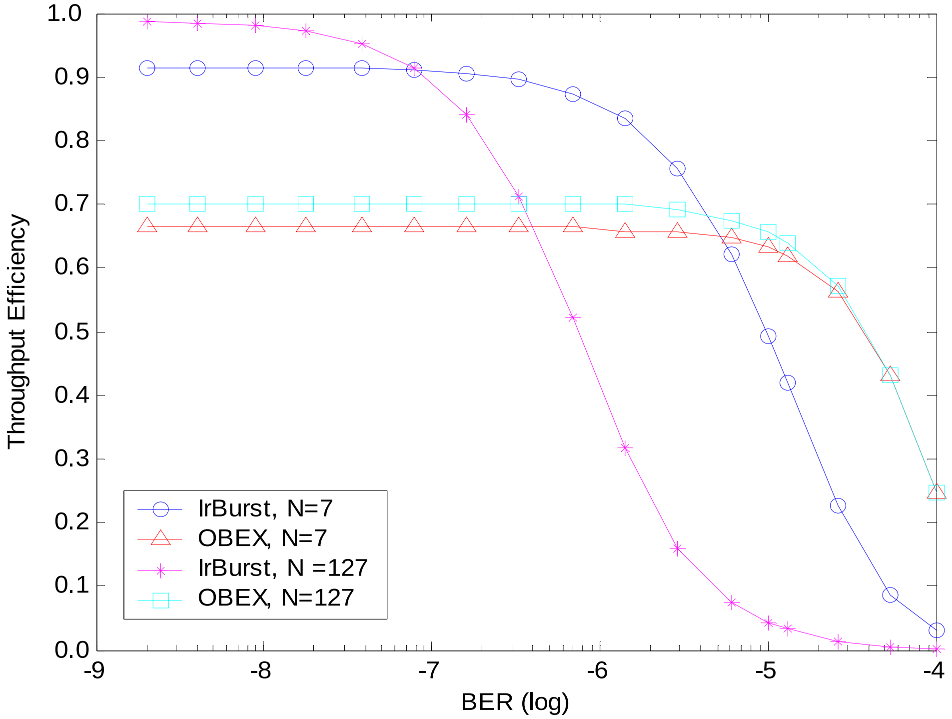


Figure 4.5 Performance comparison of IrBurst and OBEX for various bit error rates at C=100Mb/s.

4.2.2.2.2 Performance Comparison of IrBurst and OBEX for a range of BERs

Figure 4.5 compares IrBurst and OBEX throughput efficiency (TE) over a range of bit error rates (BERs) at C=100Mbps with minimum turn around time of 0.1ms and two different window sizes of 7 and 127 frames using equations (4.16) and (4.34). The figure shows that IrBurst has significant improvement (almost 30%) in throughput efficiency (TE) over OBEX for window size (*N*) of 127 at t_{tat}=0.1ms over a range of bit error rates (10^{-9} to 10^{-6}). For window size (N) of 7 frames, the improvement is nearly 22% over a wide range of bit error rates (10^{-9} to 10^{-5}). However, as the bit error rate (BER) increases, a much different behavior is observed. At high BER (10^{-6}), IrBurst performance falls at the same level of OBEX for window size of 127 whereas for N=7 IrBurst performs better until BER of 10^{-5}. For further increase in BERs, IrBurst performance decreases significantly while OBEX has better robustness to high BERs.

The same comparison of throughput efficiency (TE) between IrBurst and OBEX considering erroneous data transmission for VFIR (C=16Mb/s) [22] link is depicted in Figure 4.6 The figure shows that IrBurst has small improvement in throughput efficiency (7%) over OBEX at C=16Mb/s links with a low turnaround time of 0.1ms for both window size 7 and 127. This figure confirms that OBEX has significant robustness to high BERs for 16Mb/s links also.

Therefore, the performance comparison of IrBurst and OBEX for various BERs also confirms that IrBurst protocol significantly outperforms existing OBEX protocol for exchange of large data blocks especially at high data rates and its throughput efficiency always benefits by a large window size (N=127). However, the OBEX protocol provides high robustness to the BER increase compared to the IrBurst protocol. This is due to the use of Stop and Wait error control scheme at OBEX layer in addition to the lower layer error recovery scheme.

4.2.2.3 Performance Evaluation of IrBurst in Presence of Transmission Errors

In this section I apply my model using (4.16) to a number of scenarios in order to characterize the performance of IrBurst and to examine the effect of the link layer parameters, such as window size and frame length, and physical layer parameters, such as minimum turnaround time, on system throughput for high-speed IrDA links in the

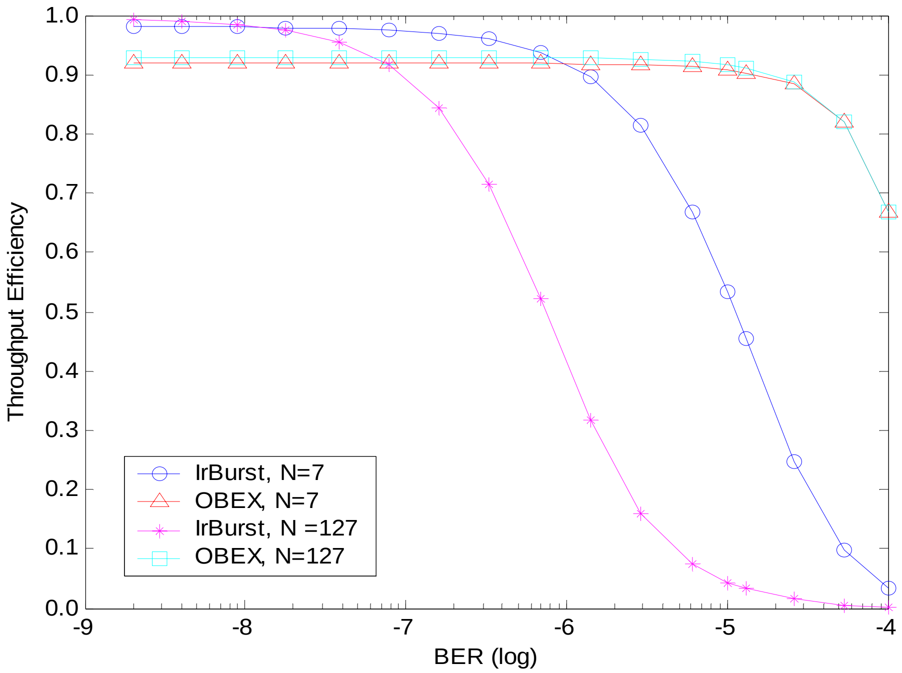


Figure 4.6 Performance comparison of IrBurst and OBEX for various bit error rates at C=16Mb/s.

presence of transmission errors.

4.2.2.3.1 Analysis of IrBurst Throughput Efficiency in Presence of transmission Errors

Figure 4.7 plots throughput efficiency versus BER for C=100Mb/s, l= 2KB, L= 100MB and different values of t_{tat} and N. At t_{tat}=0.01ms, IrBurst has 99.16% efficiency with N=127 but remains very vulnerable to high bit error rates and the efficiency degrades to 50% at BER=10^{-6}.

In contrary, IrBurst with small window size of 7 has excellent throughput performance over a wide range of BERs including high BER (10^{-6}) for this turnaround time. With this window size IrBurst has 98.55% TE for low BER and retains to almost 90% efficiency even at very high BER (10^{-6}). However window size 7 is not sufficient for IrBurst as t_{tat} increases to 0.1ms due to the negative effect of t_{tat}. Although IrBurst with N=7 has the satisfactory level of TE over a wide range of BERs including high BER, it suffers 9% sacrifice of best TE for low BERs. The best TE at low BER is achieved with existing maximum window size of 127.

For further increase in t_{tat} to 1.0ms, a significant difference in throughput efficiency (TE) with N=7 and N=127 is observed. In this case, window size 127 provides

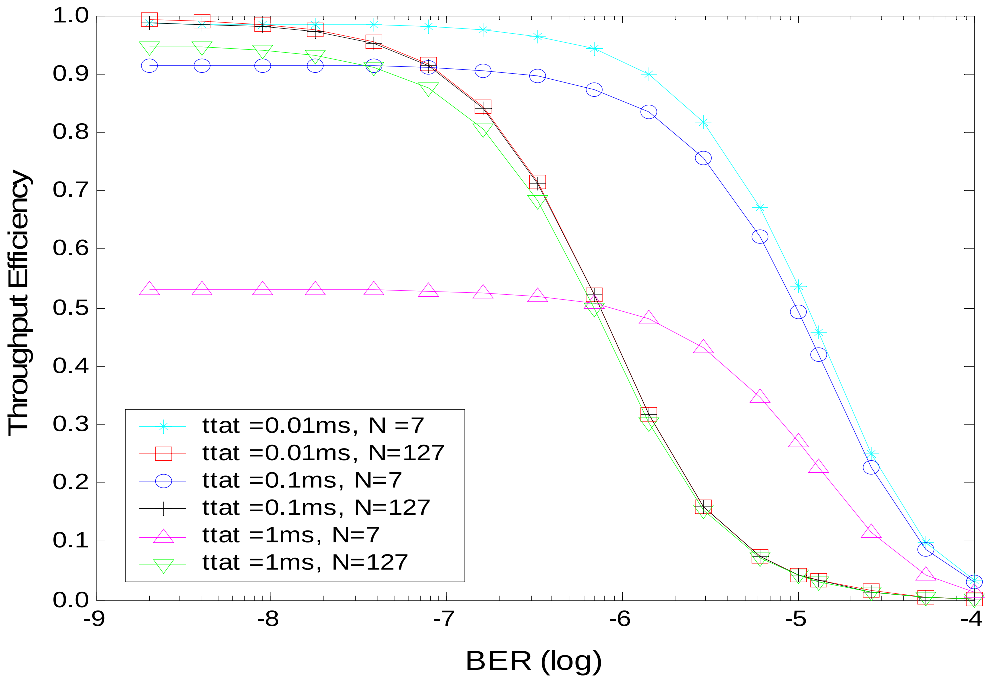


Figure 4.7 Throughput Efficiency versus BER for C=100Mb/s, L=100MB and l=2KB.

best TE which is 40% higher than that of *N*=7 for low BERs but the TE deteriorate considerably at high BERs.

As a conclusion, for very low turnaround time of 0.01ms or less, window size 7 with existing frame length (*l*=2KB) is sufficient to achieve satisfactory performance for IrBurst over a wide range of BERs. However, such a low turnaround time is not always achievable due to physical limitations and backward compatibility with other device. Therefore, in situations where small turnaround time is not possible, using a large window size (*N*=127) or frame length (*l*=16KB) can alleviate the negative effect of a large minimum turnaround time by increasing the amount of data sent between turnarounds.

4.2.2.3.2 Effect of Processor Speed on IrBurst Throughput Efficiency

Figure 4.8 plots IrBurst throughput efficiency against processor speed for various minimum turnaround time (t_{tat}) and BERs with C=100Mb/s, l= 2KB and L=100MB using (4.22). As shown in the figure, IrBurst throughput efficiency (TE) increases significantly as the processor speed increases up to certain limit, for any BER and t_{tat}. However, TE saturates as processor speed exceeds a certain limit which is due to the effect of minimum turnaround time. For t_{tat}=5ms at BER=10^{-8}, IrBurst TE increases with the increase of processor speed from 10 kHz to 500 kHz. However if the processor speed is higher than 500 kHz, throughput does not increase. This is due to the fact that the processing time to get TinyTP feedback upon receiving last frame at secondary station becomes lower than t_{tat} when processor speed reaches 500 kHz. For a lower minimum turnaround time (0.1ms) at the same BER, throughput efficiency increases as processor speed increases up to 30MHz and saturates beyond that processor speed limit. The same characteristics is observed for the same value of t_{tat} at BER=10^{-7}. Hence I can conclude that for a low minimum turnaround time, IrBurst TE is always benefited by higher processor speed. However, for a large turnaround time which is the physical property of IrDA transceivers, a moderate processor speed can also achieve maximum IrBurst TE.

4.2.2.3.3 Effect of IrLAP Window Size

The effect of window size on IrBurst throughput efficiency (TE) for different link BERs with C=100Mb/s, L=100MB, l=2KB and t_{tat} =0.1ms is shown in Figure 4.9. The figure depicts that large window size provides significant throughput increase for low BERs but renders the TE very much vulnerable to BER increase.

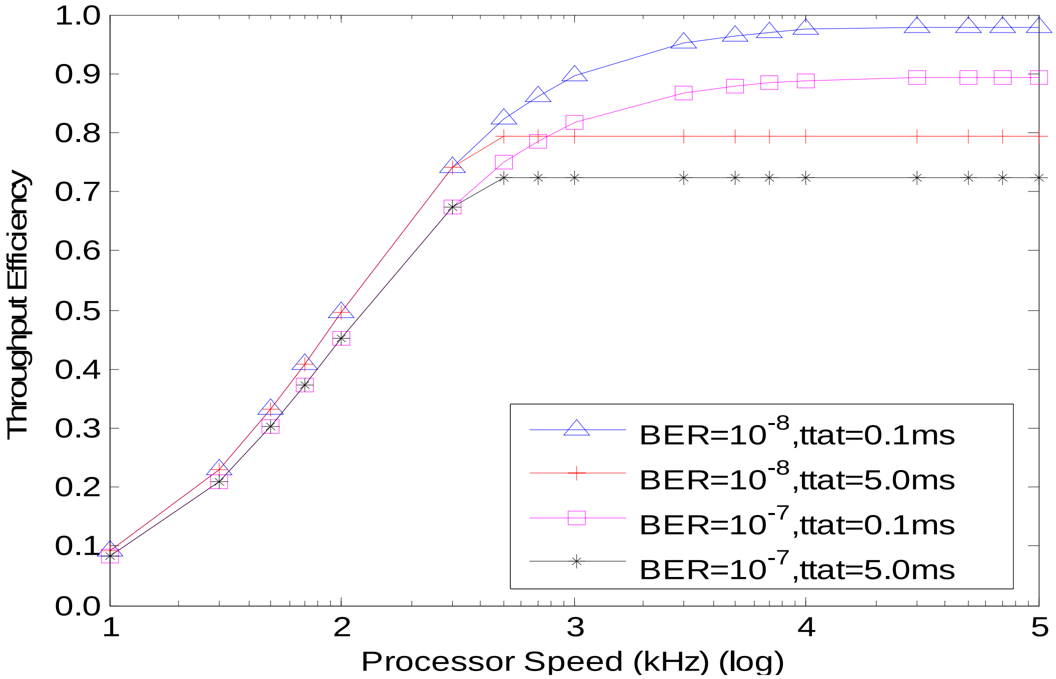


Figure 4.8 Throughput efficiency versus processor speed for different BERs and minimum turnaround time (t_{tat})

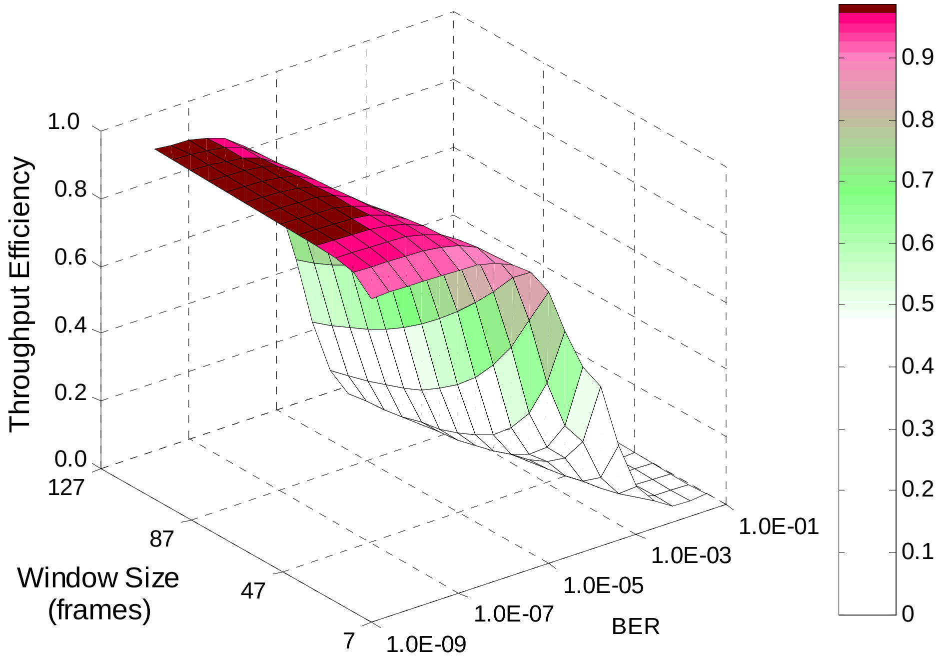


Figure 4.9 Effect of window size on IrBurst performance at different BERs for C=100Mb/s, t_{tat}=0.1ms, I=2KB and L=100MB

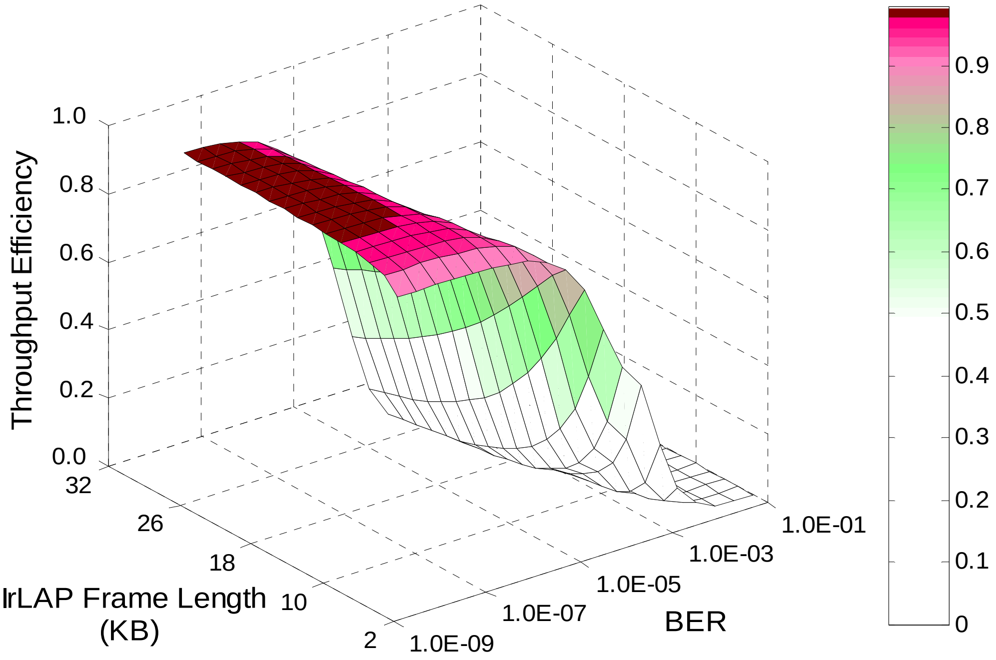


Figure 4.10 Effect of IrLAP frame length on IrBurst performance at different BERs for C=100Mb/s, t_{tat}=0.1ms, N=7 and L=100MB

4.2.2.3.4 Effect of IrLAP Frame Length

Figure 4.10 shows the effect of frame length on the performance of IrBurst at different BERs with C=100Mb/s, L=100MB, N=7 and t_{tat} =0.1ms. As shown in the figure, IrBurst with large frame length has excellent TE at low BERs but suffers severe deterioration in TE for high BERs.

4.3 Enhancing Performance of IrBurst Protocol

The analysis, carried out in the previous chapter, has showed that IrBurst throughput efficiency (TE) always benefits by a small minimum turnaround time. However, such a low turnaround time is not always achievable due to physical limitations and backward compatibility with other device. Therefore, in situations where small turnaround time is not possible, using a large window size (N=127) or frame length (l=16KB) can alleviate the negative effect of a large minimum turnaround time by increasing the amount of data sent between turnarounds. On the other hand, although using large window size or frame length significantly improves the throughput efficiency for low BERs, it renders the link vulnerable to BER increase.

Thus, for large window size or frame length values, significant decrease in IrBurst TE is observed for high BERs caused by the retransmission of correctly received out of sequence frames. This is a limitation of the existing Go-Back-N (GBN) ARQ scheme adopted by IrDA. The basic GBN ARQ error control scheme is the simplest continuous ARQ scheme to implement [29]. For this inherent simplicity, Infrared Link Access Protocol (IrLAP) uses Go-Back-N ARQ scheme as the error control scheme for data transmission over infrared links. The main drawback is that, whenever a received frame is detected in error, the receiver also rejects subsequent N-1 received frames, even though many of them may be error free. As a result, they must be retransmitted and have the chance to be in error in the following retransmission. This signifies a waste of transmissions which results in severe deterioration of throughput performance especially in the case of high data rate transmission. Therefore, an effective ARQ scheme is of great importance for IrBurst protocol throughput as well as high-speed IrDA links throughput, especially at high BERs. To improve IrBurst performance when large window size or large frame length is used, proposed BBWR and ISR ARQ schemes (Section 3.3) at IrLAP layer are examined in the following sections.

4.3.1 Proposed BBWR ARQ scheme in case of Large Window Size

As explained earlier, Block Based Window Retransmission (BBWR) scheme divides all frames stored in receiver buffer in blocks. If all the frames within a block are stored in the receiver buffer, it does not retransmit any frame of that block. In case of large window size, the secondary station needs a large number of bits to form the bitmap information for primary station if all the frames are acknowledged individually. Hence, in this section, I examine the improvement of IrBurst performance by deploying BBWR ARQ scheme, which is discussed in detail in section 3.3.1, at IrLAP layer when large window size is negotiated.

Figure 4.11 plots IrBurst throughput efficiency versus bit error rate (BER) for the proposed ARQ scheme and the existing Go-Back-N (GBN) ARQ scheme for 100Mb/s link data rate (C) with turn-around time (t_{tat}) 0.1ms and IrLAP frame length (l) 2KB. It shows that the proposed BBWR ARQ scheme enables IrBurst to achieve almost 98% throughput efficiency (TE) at low BERs by using large window size (N=127) while it has better robustness than GBN scheme with small window size (N=7) at high BERs.

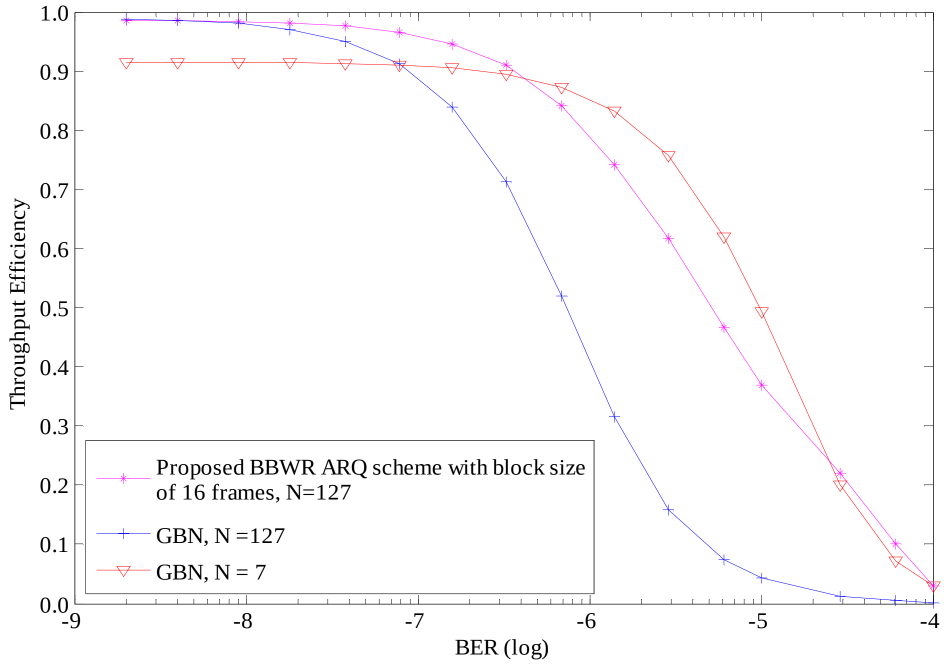


Figure 4.11 IrBurst throughput efficiency versus BER for t_{tat}=0.1ms, *l*=2KB and *C*=100Mbit/s

4.3.2 Proposed ISR ARQ scheme in case of Large Frame Length

Improved Selective Repeat (ISR) ARQ scheme, which is also discussed in chapter 3, may be considered for the case where large frame length instead of large window size is deployed. In case of large frame length, the window size can effectively be kept small (N=7) which allows the proposed scheme to use the existing but rearranged I-frame or S-frame for providing the bit map information (see Fig. 4.12).

Supervisory Frame	Next Expected Frame Number 3 bits	Bit Map Information (for Buffer Status) 8 bits	P/F	Frame Identifier 2 bits	0	1
Information Frame	Next Expected Frame Number 3 bits	Bit Map Information (for Buffer Status) 8 bits	P/F	Frame Sequence Number 3 bits		0

Figure 4.12 Supervisory and Information frame structure for Proposed ISR ARQ scheme

Since only the erroneous frames received at secondary station are required to be retransmitted in this proposed ISR ARQ scheme, the number of frames correctly transmitted by primary station in one full window (N_{corr_ARQ}) is as follows:

$$N_{corr\_ARQ} = N - p * N \qquad (4.35)$$

where variable N and p are defined in Table 4.1 of section 4.2.

Hence, the total number of windows required to transmit the full data content for this ISR ARQ scheme (W_{err_ARQ}) can be defined using (35) as:

$$Werr\_ARQ = \frac{L}{(l - l_{TTP} - l_{LMP}) * N_{corr\_ARQ}} \qquad (4.36)$$

Now replacing W_{err} with this W_{err_ARQ} in (4.11), (4.12), (4.13), (4.14) and finally using the values of these equations in (4.15) and (4.16) of section 4.2.1.2, I can calculate the throughput efficiency of IrBurst considering my proposed ARQ scheme in a straightforward manner.

IrBurst throughput efficiency versus BER for existing Go-Back-N (GBN) ARQ scheme and the proposed ISR scheme is plotted in Figure 4.13 for C=100Mb/s, t_{tat}=0.1ms and L=100MB. The figure shows that analytical result practically coincides with the simulation result for my proposed ARQ scheme. It also depicts that IrBurst, by employing ISR ARQ scheme, achieves 99% TE at low BERs with large frame length of 16KB while it has almost similar robustness at high BERs compared to GBN scheme with small frame length of 2KB.

4.4 Conclusion

In this chapter, I examined the suitability of IrBurst protocol for large data block exchange over high-speed IrDA links, investigated the interaction between IrBurst and the lower layer IrDA protocol stack, and also examined the effect of two proposed ARQ schemes on system throughput. A complete analytical model was carried out to derive IrBurst throughput efficiency (TE) over the IrDA protocol stacks both in the case of error free transmission and in presence of transmission errors. Results are presented which reveal that IrBurst scales well to handle large data blocks and high data rates compared to the existing OBEX protocol. I also examined the impact of IrLAP window size and frame length and IrPHY minimum turnaround time on IrBurst throughput efficiency (TE) in presence of transmission errors. The analysis has showed that IrBurst throughput

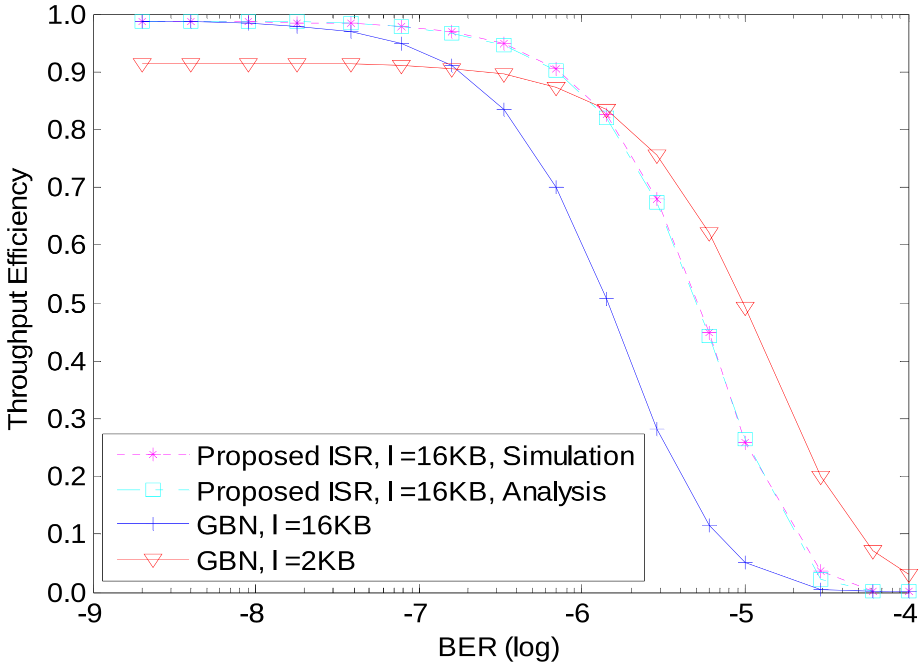


Figure 4.13 Throughput Efficiency versus BER for *C*=100Mb/s, t_{tat}=0.1ms, *N*=7 and *L*=100MB

efficiency (TE) always benefits by a small minimum turnaround time. However, in situations where small turnaround time is not possible, using a large window size or frame length can alleviate the negative effect of a large minimum turnaround time by increasing the amount of data sent between turnarounds. Although using large window size or frame length significantly improves the throughput efficiency for low BERs, it renders the link vulnerable to BER increase. To mitigate the loss of TE at high BER, I deploy BBWR and ISR ARQ schemes at IrLAP layer both in case of large window size and large frame length respectively. Simulation result also shows that employment of the proposed ARQ schemes at IrLAP layer highly improves throughput performance at high BERs.

Chapter 5

Performance Evaluation of IrSimple Protocol and Its efficiency Enhancement with Effective Error Control and Flow Control schemes

This chapter presents an analytical model for IrSimple [24] throughput efficiency over IrDA protocol stacks. Based on this model, the performance of IrSimple protocol is compared with the existing IrDA protocol, OBEX [17], [19] for digital content exchange at high data rates. The relationships of data size, IrSMP block size [28], and link minimum turnaround time on the IrSimple throughput efficiency for high-speed IrDA links are also studied. Furthermore, in order to characterize the IrSimple performance in presence of transmission errors, a simulation model for IrSimple is developed. A performance comparison of IrSimple and OBEX is carried out for various bit error rates based on the model. The model also allows the evaluation of the significance of different layer parameters such as IrSMP block size and IrPHY minimum turnaround time on IrSimple throughput in presence of transmission errors. Consequently, to improve the protocol performance at high BERs; I propose an enhancement of the existing error control scheme. Simulation results indicate that employment of proposed error control scheme results in significant improvement of IrSimple throughput efficiency at high BERs. Finally, an improvement in the flow control scheme that allows the link layer to manage data flow instead of higher layer is presented to reduce the traffic in the processing systems generated by redundant data retransmission considering all possible cases of frame losses [41], [42].

5.1 Mathematical Analysis

In this section, a mathematical model is developed which leads to derivation of the IrSimple throughput efficiency. Based on this IrSimple Model and the OBEX model developed in chapter 4 (Section 4.2.1.4), I compare the performance of IrSimple [24] and existing OBEX protocol [17] for exchanging digital contents at various data rates. The model also allows the evaluation of the relationship of the IrSMP block size, data size and IrPHY minimum turnaround time on IrSimple throughput for high-speed IrDA links. I make use of Table 5.1 for symbol details. For the purpose of developing the mathematical model of IrSimple, the following assumptions are made:

- IrSimple throughput efficiency is calculated considering both connection establishment and data transmission time.
- Bi-directional transfer mode is considered in order to carry out complete performance analysis. However, the derived model can also be modified in a straightforward manner for Uni-directional transfer mode.
- According to the IrSimple specification [24] and IrLAP for IrSimple addition [26], the maximum connection establishment time for IrSimple (t_{CE}) is assumed of 60ms.
- All Objects are pushed to the secondary using OBEX PUT operation.

5.1.1 Modeling of IrSimple

The mathematical model uses Figure 5.1 that illustrates the way in which the IrSimple protocol packetizes an object for transmission and the details protocol mapping of the object throughout the protocol stacks starting from higher layer (OBEX) down to the link layer of the stacks (IrLAP). Since all objects are pushed to the secondary using OBEX PUT operation, pushing an object can take one or more OBEX packets. The first packet typically contains some extra information (l_{OBEX1} - l_{OBEXn}) of the object (such as name, length, etc.) but all the subsequent packets contains fixed header of length l_{OBEXn} only. The total number of OBEX packets for an object of size L is given by:

$$n_{OP} = \left\lceil \frac{\left(L + \left(l_{OBEX1} - l_{OBEXn}\right)\right)}{\left(P_{REQ} - l_{OBEXn}\right)} \right\rceil \qquad (5.1)$$

When the OBEX packets are passed through the IrSMP layer, they are fitted into SMP blocks where the block size is the maximum data size that the primary station of SMP can resend and the secondary station can receive at once. To yield the total number of SMP blocks (n_B), I divide the total amount of data (L + (l_{OBEX1} - l_{OBEXn}) + n_{OP}*l_{OBEXn}) that the SMP has to send for transmitting the object (L) by the block size (BS), which is given by:

Table 5.1 Mathematical Model Variables for IrSimple protocol

Symb.	Parameter Description	Unit
C	Link data rate	bit/s
L	Object size	bit
l	Payload size of IrLAP frame (Frame length)	bit
l_{PHY}	Physical layer overhead	48bit
l_{LAP}	IrLAP layer header or S-frame size	bit
l_{LMP}	IrLMP layer header	16bit
l_{SMP}	IrSMP layer header	24bit
t_{PI}	IrSMP minimum packet interval time	0.1ms
b_{PI}	Equivalent bits of IrSMP minimum packet interval time: $C \times t_{PI}$	bit
t_{CE}	Connection establishment time	sec
b_{CE}	Equivalent bits of Connection establishment time: $C \times t_{CE}$	bit
t_{tat}	IrLAP minimum turnaround time	sec
b_{tat}	Equivalent bits of IrLAP turnaround time: $C \times t_{tat}$	bit

$$n_B = \left(\frac{\left(L + \left(l_{OBEX\,1} - l_{OBEXn}\right) + n_{OP} * l_{OBEXn}\right)}{BS} \right) \quad (5.2)$$

As described in the chapter2, the IrSMP layer packetizes each block in such a way so that all SMP packets must fit within a single IrLAP payload. Therefore, the total number of IrLAP frames required for each SMP block (n_{f1}) is:

$$n_{f\,1} = \left(\frac{BS}{\left(l - l_{SMP} - l_{LMP}\right)} \right) \quad (5.3)$$

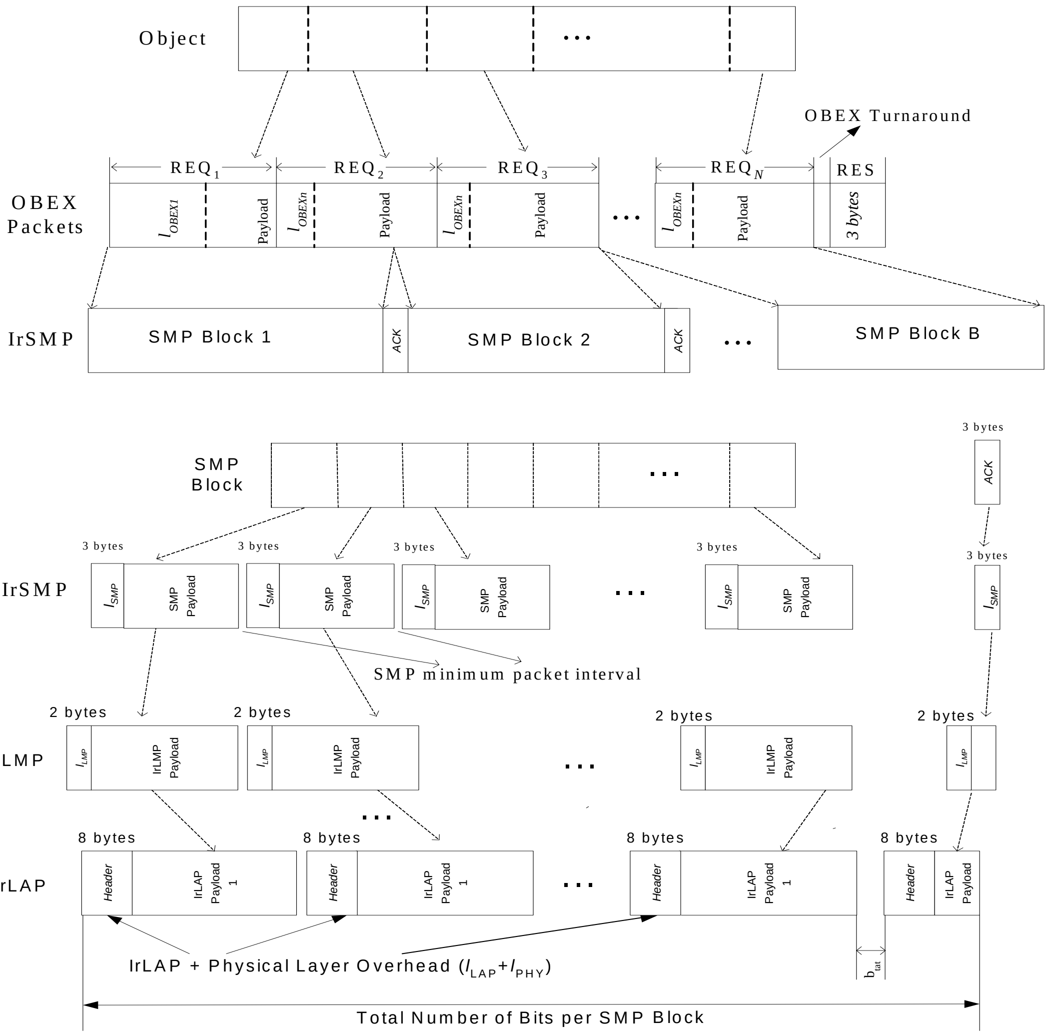


Figure 5.1 Mapping IrSimple protocol to IrLAP frames

Because the total number of blocks is n_B, I multiply n_{f1} by n_B to calculate the total number of IrLAP frames for all blocks (n_{fB}). To this, I multiply each IrLAP frame size (l+l_{LAP}+l_{PHY}) to yield the total number of bits required for all blocks (B_{bits}).

$$B_{bits} = \lceil n_{f1} * n_B \rceil * \left(l + l_{LAP} + l_{PHY} \right) \qquad (5.4)$$

The SMP of primary station maintains a minimum interval between any sending packets which is defined by the secondary station. Since the total number of SMP packets is equal to the total number of IrLAP frames (n_{fB}) and there is no packet interval after the last packet, the overhead due to minimum packet interval for all the SMP packets (O_{PI}) is:

$$O_{PI} = (\lceil n_{fB} \rceil - 1) * b_{PI} \tag{5.5}$$

Since each block requires an acknowledgement (*ACK*) from the SMP of secondary station, therefore the overhead due to acknowledgement (O_{ACK}) considering no ACK associated with last block is:

$$O_{ACK} = (\lceil n_B \rceil - 1) * (2 * b_{tat} + l_{SMP} + l_{LAP} + l_{PHY}) \tag{5.6}$$

OBEX for IrSimple needs only final response (*RES)* for the final PUT command i.e. for the final request packet (*REQ*) only. The overhead due to OBEX Response is:

$$\begin{aligned} O_{RES} &= b_{TA} \\ &+ (2 * b_{tat} + (3 + l_{SMP} + l_{LMP} + l_{LAP} + l_{PHY})) \end{aligned} \tag{5.7}$$

As illustrated in the figure, for transmitting the content of size L, in addition to this content extra bits are also exchanged which accounts for the delay due to connection establishment time (b_{CE}) and overhead of lower layer, acknowledgement and turnaround time effect. By adding up all the delay, overheads, the total number of bits required to transmit the object is:

$$N_{Sbits} = b_{CE} + B_{bits} + O_{PI} + O_{ACK} + O_{RES} \tag{5.8}$$

The throughput efficiency (TE), which is defined as the ratio of object size (in bits) to total number of bits required to transmit that object, is therefore given by:

$$TE = \frac{L}{N_{Sbits}} \tag{5.9}$$

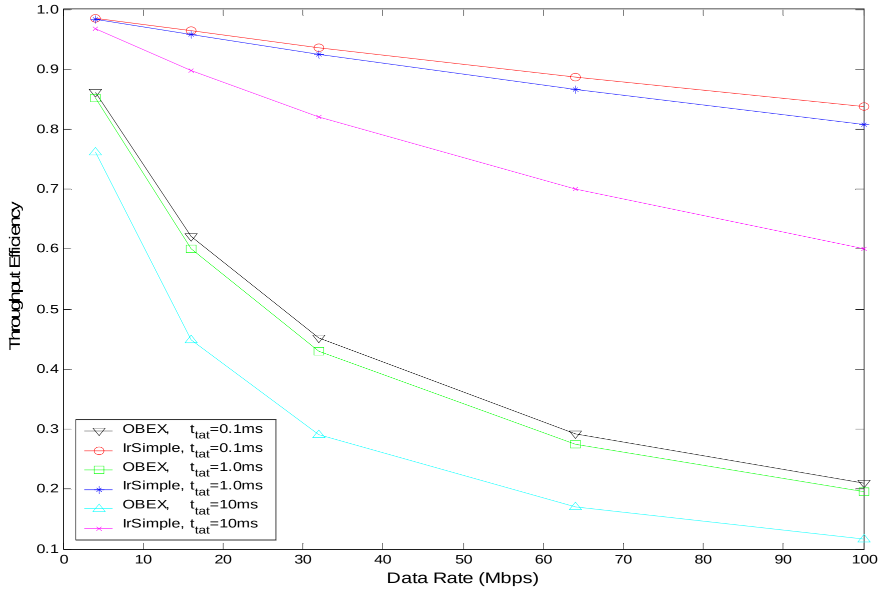


Figure 5.2 Performance comparison of IrSimple and OBEX for various data rates with *l*=2KB, *L*=4MB and *BS*=512KB.

5.1.2 Analysis

Based on the models, the performance of IrSimple is compared with that of standard IrDA OBEX protocol for high-speed exchange of data contents in this section. Furthermore, the relationship between block size and minimum turn around time on IrSimple performance is also explored. For this experiment, the data content size (*L*) is 4MB, the IrLAP frame length (*l*) is 2KB and the OBEX turnaround time (T_{TA}) is 1.0ms.

5.1.2.1 Performance comparison of IrSimple and OBEX at Various Data Rates

Figure 5.2 compares IrSimple and OBEX throughput efficiency (TE) over a range of data rates for three different minimum turn around time (t_{tat}) of 0.1ms, 1.0ms and 10.0ms using equation (5.9) and (4.29). The figure shows that while a longer minimum turnaround time always degrades throughput efficiency for both IrSimple and OBEX; the effect is more pronounced at higher data transfer rates. At a low data rate (4Mbps) and very small turnaround time (0.1ms), both IrSimple and OBEX have more than 85% throughput efficiency and IrSimple has 10% improvement in TE compared to OBEX. As the data rate increases, the improvement in throughput efficiency (TE) for IrSimple over OBEX also increases. For a large turnaround time of 10ms, IrSimple provides 47% more throughput efficiency (TE) compared to OBEX at C=16Mbps while the difference is almost 50% at C=100Mbps. The improvement in TE for IrSimple increases to 62% at C=100Mbps even for a very small turnaround time (0.1ms). Therefore, IrSimple protocol significantly outperforms standard OBEX protocol for instant exchange of data contents between portable appliances, especially at high data rates.

5.1.2.2 Relationship between IrSMP block size and IrPHY turn around time

Using equation (5.9), the effect of IrSMP block size (*BS*) and IrPHY minimum turnaround time on IrSimple throughput efficiency (TE) at C=100Mbps is examined in Figure 5.3. The figure shows that with a combination of small turnaround time and large block size, IrSimple achieves maximum throughput efficiency. Therefore, IrSimple throughput efficiency always benefits by a large block size and a small turnaround time.

5.1.2.3 Relationship between IrSMP block size and data size

Figure 5.4 shows the effect of block size and data size on throughput efficiency (TE) at C=100Mbps and $t_{tat=}$1ms. The figure depicts that with the increase of data size, TE increases slightly for a fixed block size. However, block size increment results in significant TE especially when it fine-tunes with the data size. Hence in this research, I will only consider the data size be 4MB as this is the maximum block size defined by IrDA [28].

So, IrSimple protocol significantly outperforms standard OBEX protocol for instant exchange of data contents between portable appliances, especially at high data rates and its throughput efficiency always benefits by a large block size and a small turnaround time.

5.2 Simulation and Analysis

In order to characterize the performance of IrSimple protocol in presence of transmission errors, I have developed a simulation model for IrSimple considering all the layers of its protocol stack using OPNET^{TM} simulation package [33]. In the model, the IrSimple protocol stack details are implemented according to the IrDA specifications. The main objective of the simulation is to measure the throughput performance of IrSimple protocol under varying bit error rates (BERs).

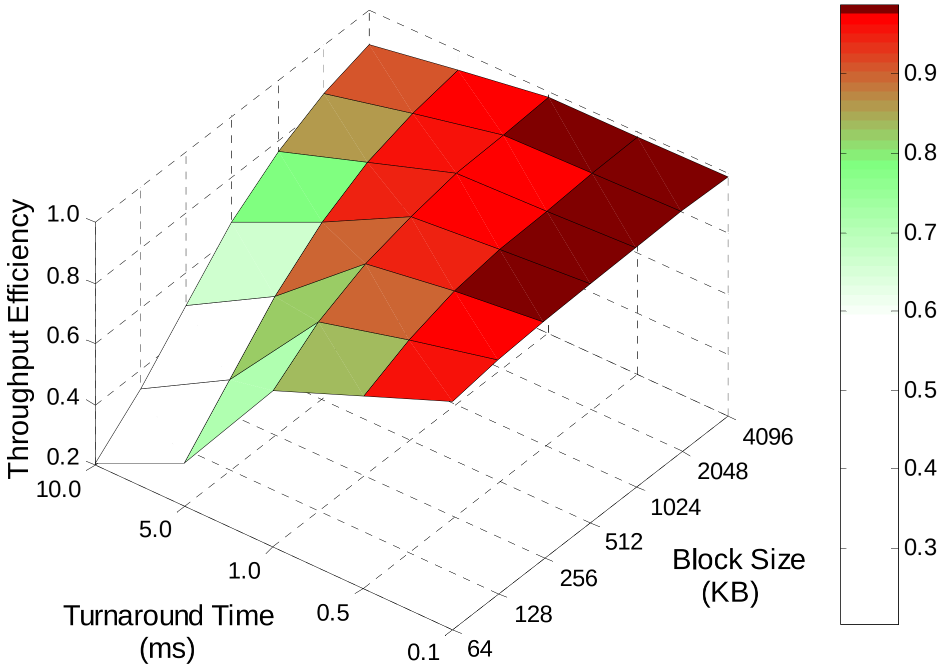


Figure 5.3 Relationship between block size and turn around time

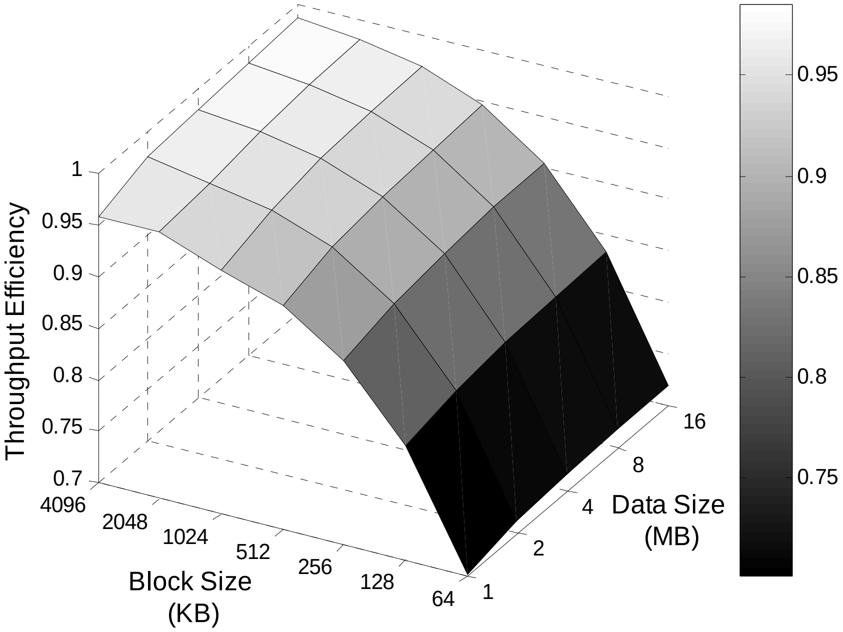


Figure 5.4 Relationship between block size and data size

Based on this model, the performance comparison of IrSimple protocol with the standard IrDA object exchange protocol (OBEX) for various bit error rates is carried out. Furthermore, a study of the importance of different layer parameters such as IrSMP block size and IrPHY minimum turnaround time on IrSimple performance in presence of errors is also presented.

5.2.1 Simulation Parameters

The parameters used throughout the presentation and analysis of simulation results are defined bellow.

- C = Link Data Rate
- L = Object or Data Content Size
- l = Payload size of IrLAP frame (Frame length)
- t_{tat} = IrPHY layer minimum turn around time
- BS = IrSMP Block Size
- T_{TA} = OBEX layer minimum turn around time

5.2.2 Simulation results

This section presents the simulation results. At first, the mathematical model that has been developed in section 5.1 is validated by comparing the results obtained by equation with results derived from simulation. Then, the performance of IrSimple is compared with that of existing OBEX for various data rates in presence of transmission errors. Furthermore, the significance of IrSMP block size and Physical layer minimum turn around time on IrSimple performance for various bit error rates (BERs) is also studied.

5.2.2.1 Model Validation

To validate the mathematical model, the results obtained from equations have been compared with that obtained using simulation in this section. Figure 5.5 plots IrSimple throughput efficiency over a range of data rates for three different minimum turn around time (t_{tat}) of 0.1ms, 1.0ms and 10.0ms.

As shown in the figure, IrSimple protocol provides higher throughput when the data rates increases for instant exchange of data contents between portable appliances.

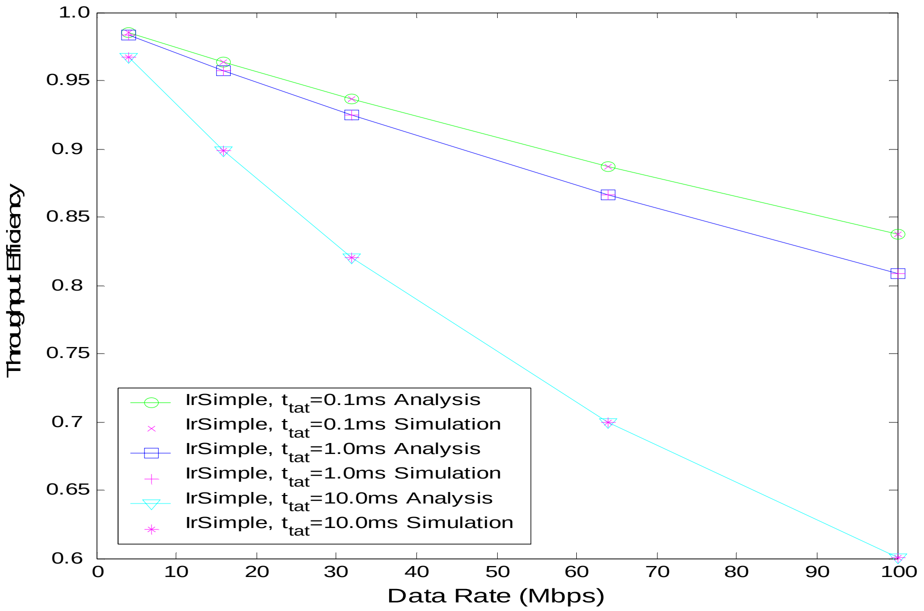


Figure 5.5 Throughput Efficiency: Analysis versus Simulation

The figure shows that analytical results practically coincide with the simulation results for all the data rates.

5.2.2.2 Performance Comparison of IrSimple and OBEX at Various Bit Error Rates

Figure 5.6 plots throughput efficiency versus bit error rate (BER) for UFIR [23] link (C=100Mb/s), l= 2KB, L= 4MB and different values of t_{tat}. The figure shows that IrSimple has almost 64% improvement in throughput efficiency over OBEX at C=100Mb/s links with a low turnaround time of 0.01ms for a range of bit error rates (10^{-9} to 10^{-6}). However, if the BER increases further, IrSimple through efficiency falls sharply although still it provides better throughput compared to the OBEX. And if the BER increases to 10^{-5} then IrSimple throughput efficiency falls at the same level as that of the OBEX. It is also shown form the figure that OBEX has more robustness of BER increase compared to the IrSimple protocol. Whereas, the improvement of IrSimple throughput efficiency is 61% for turnaround time of 1.0ms and 44% for turnaround time of 10.0ms over a range of BERs. However, for high BER, IrSimple throughput efficiency degrades sharply.

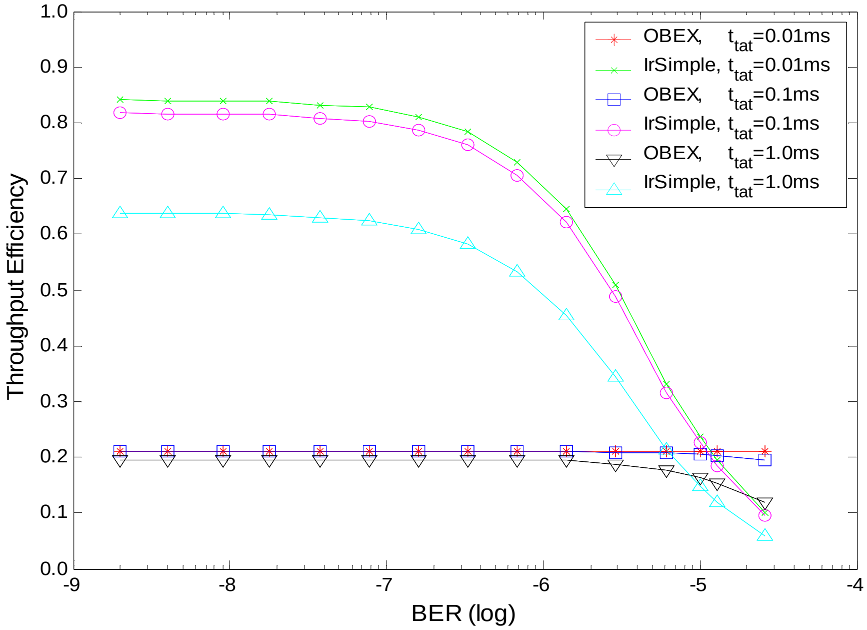


Figure 5.6 Throughput Efficiency versus BER for IrSimple and OBEX with *C*=100Mb/s, *L*=4MB, *l*=2KB and *BS*=64KB

The figure also shows that for large turnaround time of 1.0ms, IrSimple with minimum block size (64KB) can achieve only 63% throughput efficiency (TE). As ttat decreases, IrSimple achieves significant improvement in TE and for t_{tat}=0.1ms, it has almost 82% TE at low BER (10^{-8}). For further decrease in ttat from 0.1ms to 0.01ms, IrSimple performance increases by 2% for low BERs. Therefore, this figure depicts that IrSimple has significant improvement in TE compared to OBEX over a wide range of bit error rates including high BER (10^{-6}) for all turn around time. This is due tothe fact that OBEX consumes significant amount of time for device discovery before connection establishment whereas IrSimple can immediately establish a connection reducing device discovery time. Furthermore, IrSimple throughput efficiency always benefits by small minimum turnaround time.

The same comparison of throughput efficiency (TE) between IrSimple and OBEX considering erroneous data transmission for VFIR (*C*=16Mb/s) link [22] is depicted in Figure 5.7.

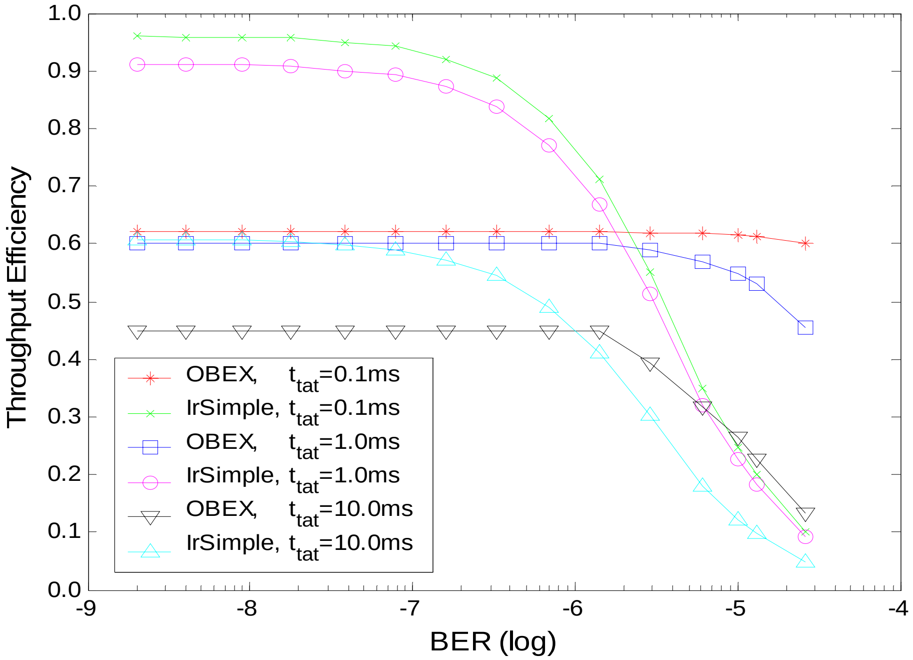


Figure 5.7 Throughput Efficiency versus BER for IrSimple and OBEX with C=16Mb/s, L=4MB, l=2KB and BS=64KB.

The figure shows that IrSimple has almost 35% improvement in throughput efficiency over OBEX at C=16Mb/s links with a low turnaround time of 0.1ms whereas the improvement is 31% for turnaround time of 1.0ms and 15% for turnaround time of 10.0ms low BERs. However, as the BER increases the improvement of throughput efficiency using IrSimple decreases for all turn around time. This figure also confirms that IrSimple provides better throughput efficiency compared to that of OBEX and its throughput efficiency always benefits by small minimum turnaround time. Furthermore, OBEX has significant robustness to high BERs for 16Mb/s links also.

As a conclusion, IrSimple protocol significantly outperforms existing OBEX protocol for exchange of digital contents especially at high data rates (100Mb/s) and its throughput efficiency always benefits by a small minimum turn around time.

5.2.2.3 Effect of IrSMP Block Size on IrSimple Performance

The effect of block size on IrSimple throughput efficiency (TE) for different link BERs with C=100Mb/s, L=4MB, l=2KB and t_{tat} =1.0ms is examined in Figure 5.8. The figure shows that larger block size provides significant throughput increase for low BERs but renders the TE very much vulnerable to BER increase. Thus, for a large block size,

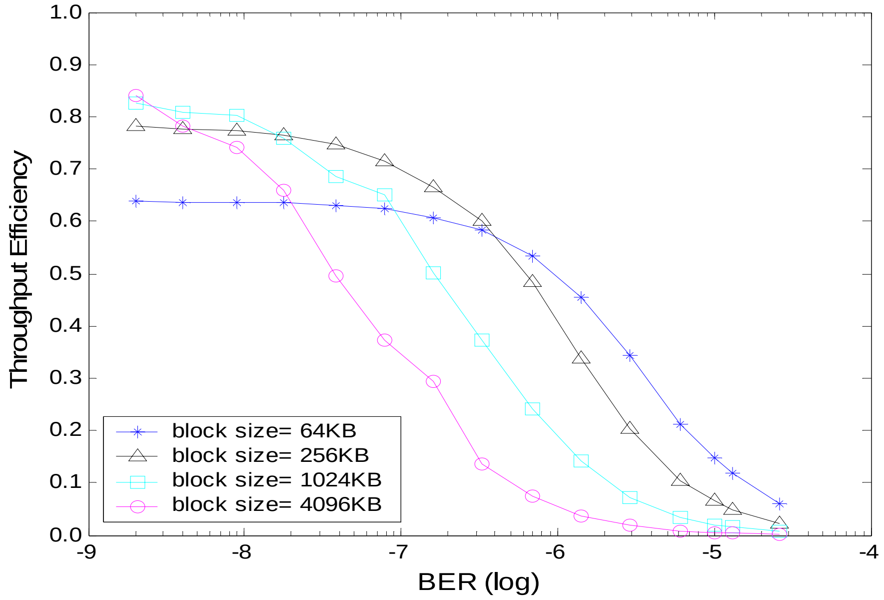


Figure 5.8 IrSimple Throughput efficiency versus bit error rate for various block sizes with *C*=100Mbps, t_{tat}=1.0ms, *l*=2KB and *L*=4MB.

significant decrease in IrSimple TE is observed for high BERs caused by the retransmission of correctly received out of sequence frames.

5.3 Improving IrSimple Performance

The analysis, carried out in the previous section, has showed that IrSimple throughput efficiency (TE) always benefits by a small minimum turnaround time. However, such a low turnaround time is not always achievable due to physical limitations and backward compatibility with other device. Therefore, in situations where small turnaround time is not possible, using a large IrSMP block size can alleviate the negative effect of a large minimum turnaround time. On the other hand, although using large block size significantly improves the throughput efficiency for low bit error rates (BERs); it renders the link vulnerable to BER increase.

Thus, for large block size values, significant decrease in IrSimple TE is observed for high BERs caused by the retransmission of correctly received out of sequence frames. This is a limitation of the existing error control scheme adopted for IrSimple protocol. Therefore, a more effective error control scheme at IrSMP layer is of great importance for increasing IrSimple performance. Hence, in an effort to improve IrSimple performance using large block size; a more effective error control scheme at IrSMP layer is proposed here. Furthermore, a modification of the existing flow control for IrSimple protocol at IrLAP layer is presented to reduce the traffic in the system.

5.3.1 Enhancing Error control scheme

In order to improve IrSimple performance, an effective error control scheme is presented in this section. Before going into the details of the proposed error control scheme, a brief overview as well as the limitation of the existing error control scheme is described. Finally, the performance of the proposed error control scheme is compared with that of existing error control scheme to examine the effectiveness of the proposed error control scheme.

5.3.1.1 Existing error control scheme

In the existing error recovery scheme, IrSMP layer at Secondary station checks the sequence number of the data packet from lower layer [28]. When a lost sequence number is detected, secondary station maintains that sequence number for the retransmission request to primary station during transfer turn and discards all the subsequent frames. When secondary station receives a packet with block last (*BL*) bit set to 1, it sends a response packet to lower layer. Secondary sets response (*RS*) bit to 0 if there is a packet which it wants primary station to resend. Otherwise it sets RS bit to 1. The secondary station also sets the sequence number (*Nr*) field to the packet number which secondary station wants primary station to resend. Upon getting the response packet from secondary with RS field set to 0, primary station resends all the packets from the received sequence number to the end of the block. As a result, many error free packets need to be retransmitted and have the chance to be in error in the following retransmission. This signifies a waste of transmissions which results in severe deterioration of throughput performance especially in the case of high bit error rate. Figure 5.9 explains the operation of existing error control scheme briefly.

5.3.1.2 Proposed Error Control Scheme

In the proposed error control scheme, all error free but out of sequence data packets are stored at the secondary station (receiver) buffer instead of being discarded. The receiver is assumed to have enough buffer storage to hold at most a block size of data packets.

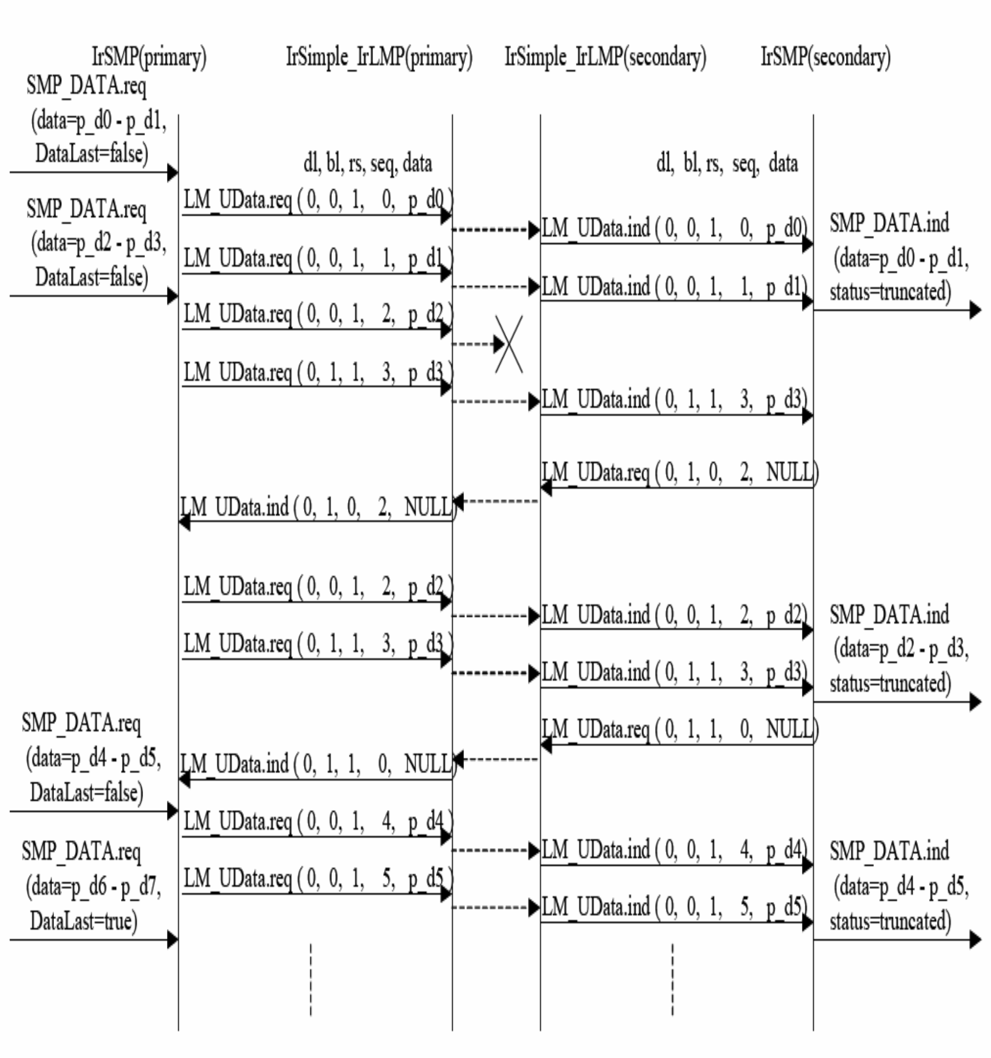


Figure 5.9 Existing Error Control Scheme

The secondary station acknowledges all *UI*-frames received using a new multiple acknowledgement scheme. In the proposed scheme, the response packet sent by secondary station also contains status (*S*) field which is bitmap information to indicate the status of all out of sequence packets stored at receiver buffer. The reserved bit (*R*) is set to ‘1’ to indicate this response packet. A ‘1’ is assigned to the bits in *S* corresponding to packet sequence numbers that have been received correctly. Otherwise, a ‘0’ is assigned to the corresponding sequence numbers. The *S* field reports the reception status of packets with sequence numbers starting from *Nr*+1 to the highest received frame sequence number. The maximum length that *S* field may reach is equal to the 2047. The primary station can easily calculate the length of *S* by subtracting *Nr* from the highest sequence number of packet last sent. The *S* field is optional. It is not needed if no frame has been received with a packet sequence number that is higher than *Nr*. At the Primary station, when a response packet with multiple acknowledgements is received, an automatic retransmission is initiated starting from *Nr* and only the subsequent packets having ‘0’ in the corresponding bit position in status field is retransmitted. The operation of the proposed error control scheme is briefly explained in Figure 5.10.

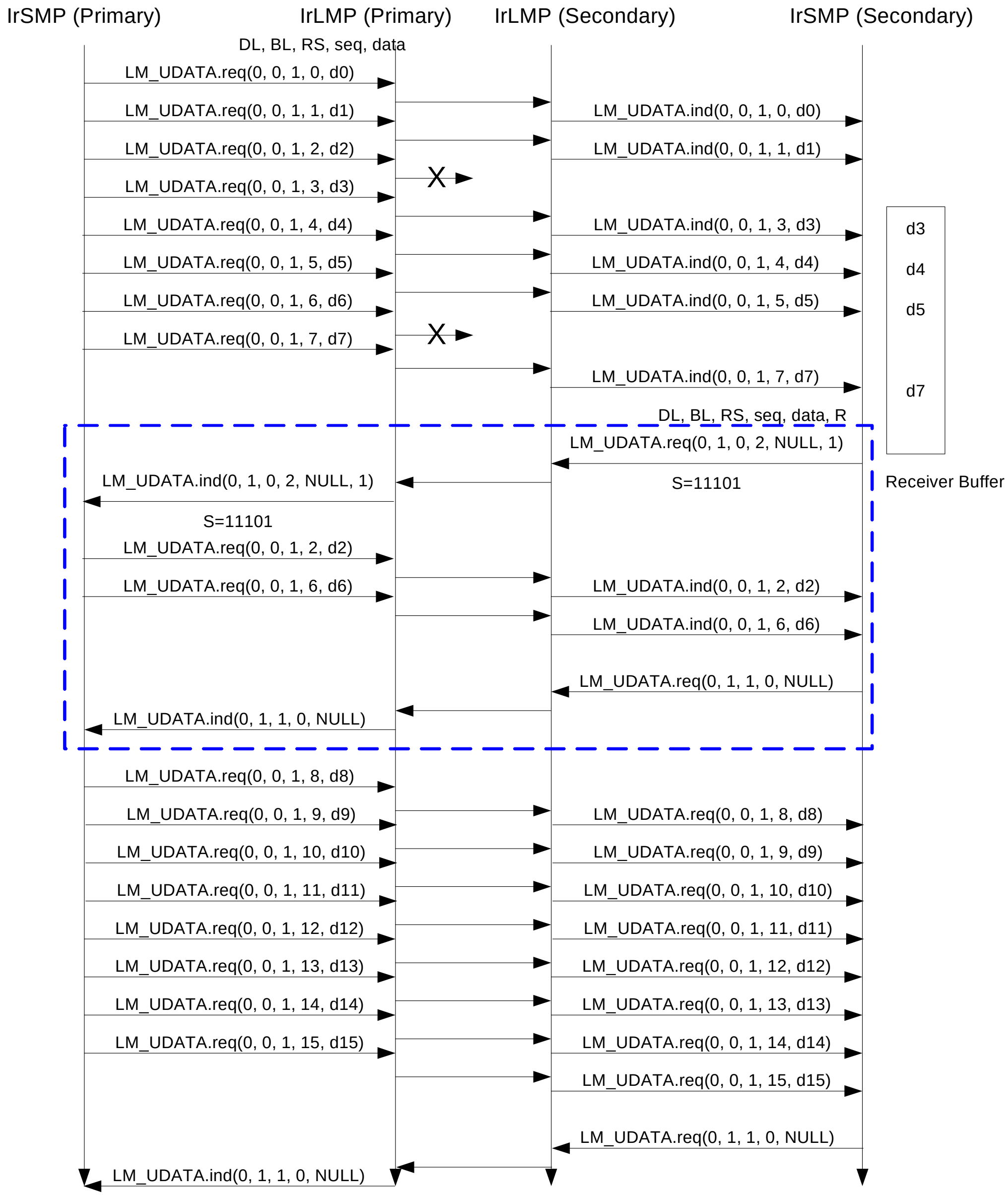


Figure 5.10 Proposed Error Control Scheme

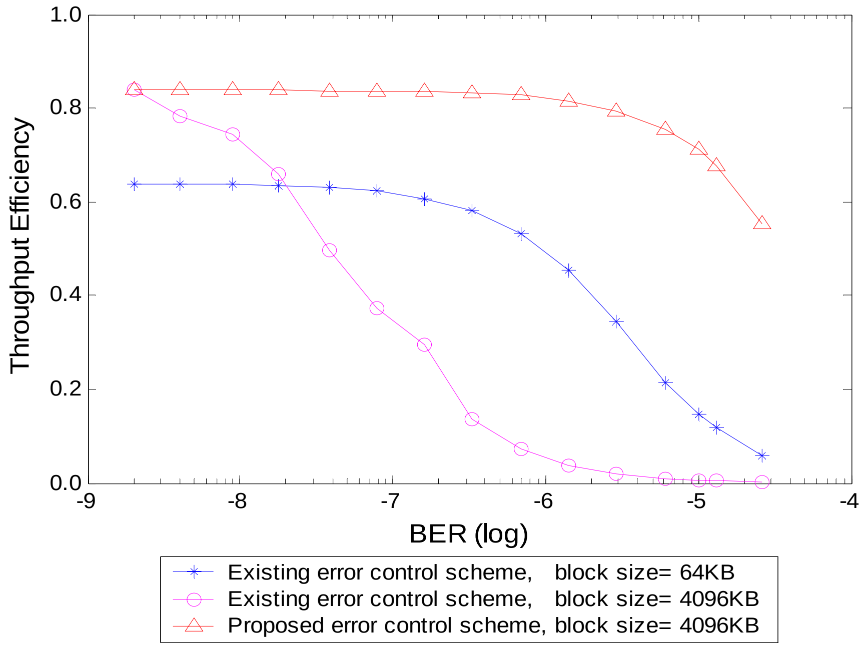


Figure 5.11 Throughput efficiency versus BER for C=100Mbps, t_{tat}=1.0ms, l=2KB and L=4MB.

5.3.1.3 Effectiveness of the Proposed Error Control Scheme

To measure the effectiveness of the proposed error control scheme, I compare the performance of proposed error control scheme and that of the existing error control scheme. This section discusses the result of two error control scheme (the proposed error control scheme and the existing error control scheme).

Figure 5.11 plots throughput efficiency versus bit error rate (BER) for my proposed error control scheme as well as for existing error control scheme with C=100Mbps, t_{tat}= 1.0ms, l=2KB and L=4MB. The figure shows that proposed error recovery scheme enables IrSimple to achieve almost 84% throughput efficiency (TE) at low BERs by using the largest block size (4096KB). Moreover, it provides significant robustness compared to the existing error control scheme with smallest block size (64KB) over a wide range BERs including high BER (10^{-5}).

5.3.2 Efficient Flow Control

This section presents an improvement in the existing flow control for IrSimple protocol at IrLAP layer to provide efficient flow control. At first, a brief description of the existing flow control is carried out.

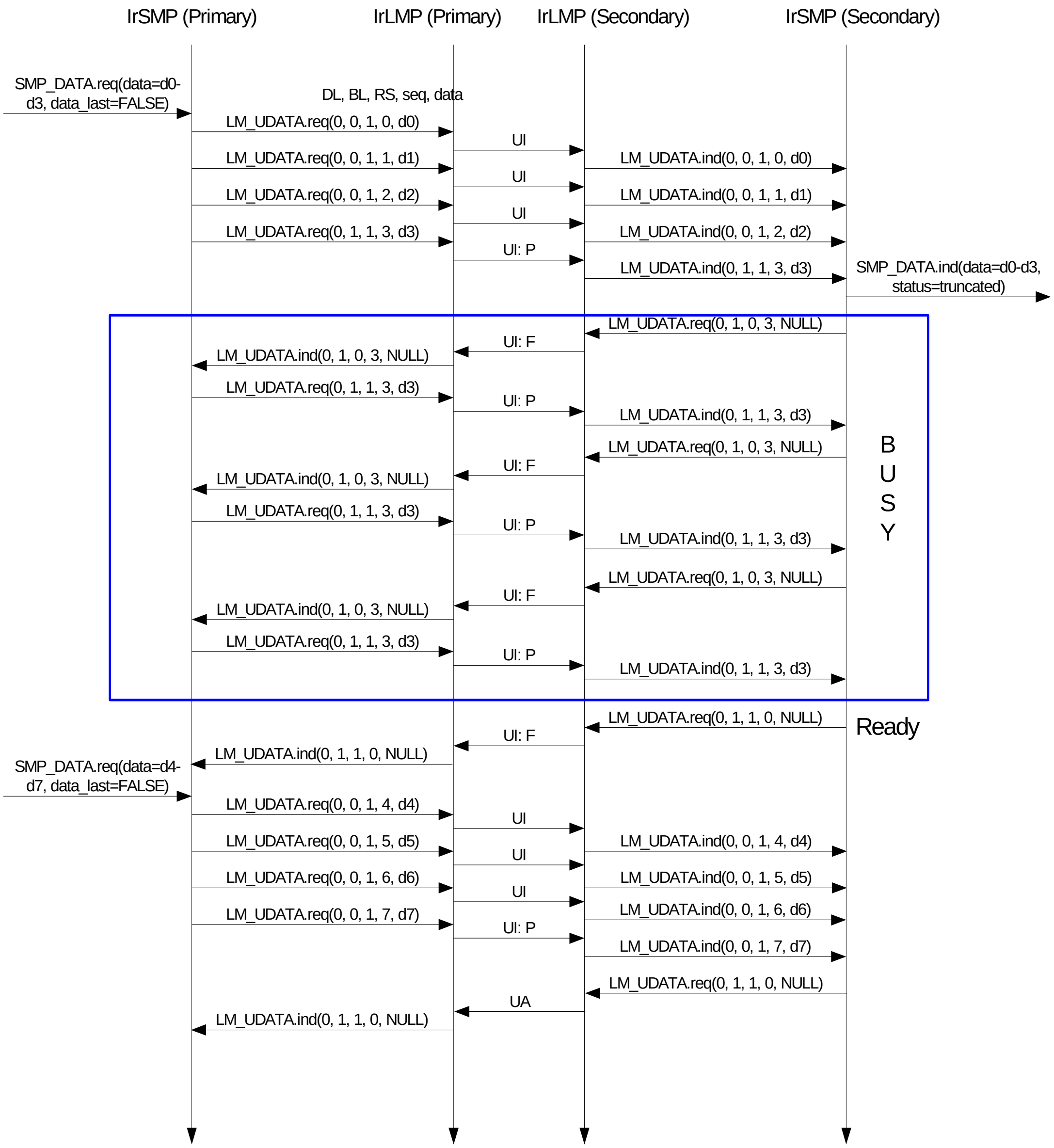


Figure 5.12 Existing Flow Control Scheme

5.3.2.1 Existing Flow Control

In the existing IrSimple protocol, the link flow control is managed by IrSMP layer (higher layer) instead of IrLAP layer (link layer). According to the IrSimple specification [24], when the IrSMP of primary station completes bulk data transfer, the flow control is needed at the IrSMP layer in the secondary station. If the IrSMP of secondary station is busy, the secondary station will not receive the last data and sends the *UI* frame requesting the last data again as there is no data from the upper layer of the secondary station. Receiving the request, primary station resends the last data of the block at IrSMP layer. This process continues until the busy state is over which results in a significant amount of traffic in the processing systems and waste in data transmission caused by the repetition of last data of the block to maintain the flow control. Figure 5.12 shows the existing flow control scehme of IrSimple protocol.

5.3.2.2 Proposed Efficient Flow Control

To reduce the traffic in the processing system generated by the redundant data retransmission, a modification to the existing flow control may be employed. The higher layer, IrSMP should not handle the flow control, instead the link layer, IrLAP layer can manage the flow control using *RR* supervisory frame (*S*-frame) [14] that passes the transmission right alternatively to maintain flow control. As the exchanges of *RR S*-frame are at the link layer instead of resending the last data at higher layer (IrSMP), the modified flow control reduces redundant data retransmissions in the system significantly. Figure 5.13 shows the improved flow control using *RR* supervisory frame. In order to cope with this improvement, no further modification is needed in the existing primary station as the use of *RR* frame with *P* bit set and reception of *RR* frame with *F* bit set is already included. However, as the existing secondary station is not allowed to initiate *RR* frame without any request from its upper layer, a modification to issue the *RR* frame with *F* bit set at the IrLAP layer by the secondary station for maintaining the flow control is suggested.

5.3.2.3 Robustness of Proposed Flow Control Scheme

In this section, the effectiveness of my proposed flow control scheme is investigated for all possible cases where a Receive Ready (RR) frame or Unacknowledged Information (UI) frame may be lost due to transmission error. In this case, it is important that the proposed flow control scheme can recover the error without adding any complexity.

5.3.2.3.1 When Receive Ready (RR) frame from secondary station is lost

The IrLAP layer of secondary station sends RR frame with F bit set to 1 (RR-F) when the IrSMP layer is busy and can not acknowledge the last received frame. If the transmitted RR frame is lost due to transmission error, the maximum turnaround time (MAX TAT) at primary station expires.

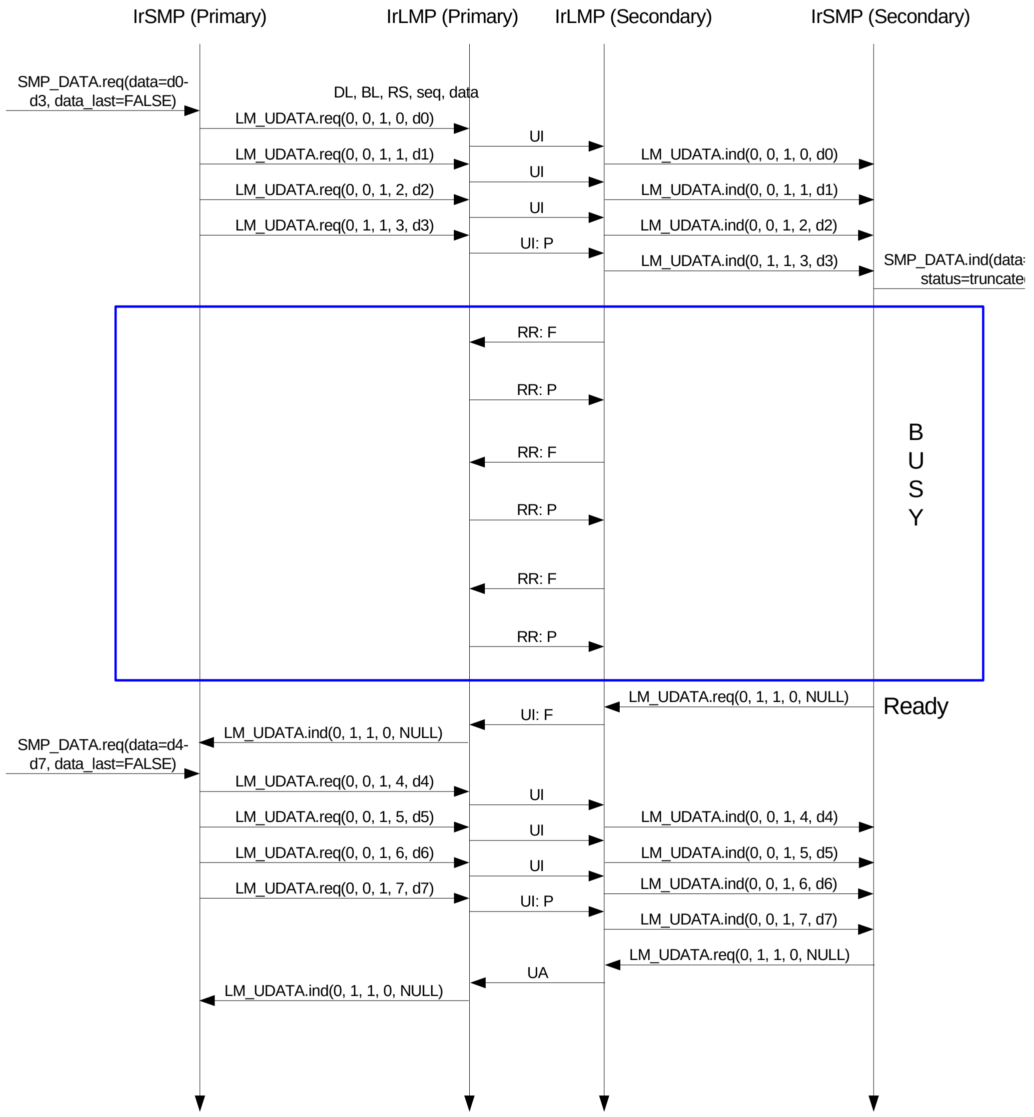


Figure 5.13 Proposed Efficient Flow Control Scheme

Then the IrLAP layer at primary station resends the last transmitted UI frame with P bit set to 1 (UI-P). Upon receiving this frame, IrLAP at secondary station notifies the higher layer about this duplicate data and sends a RR-F frame to return the control. However, IrSMP layer discards the duplicate data. This way, the proposed flow control scheme can recover the previous RR frame loss. Figure 5.14 shows the recovery procedure of the proposed flow control scheme in case of RR frame from secondary station is lost.

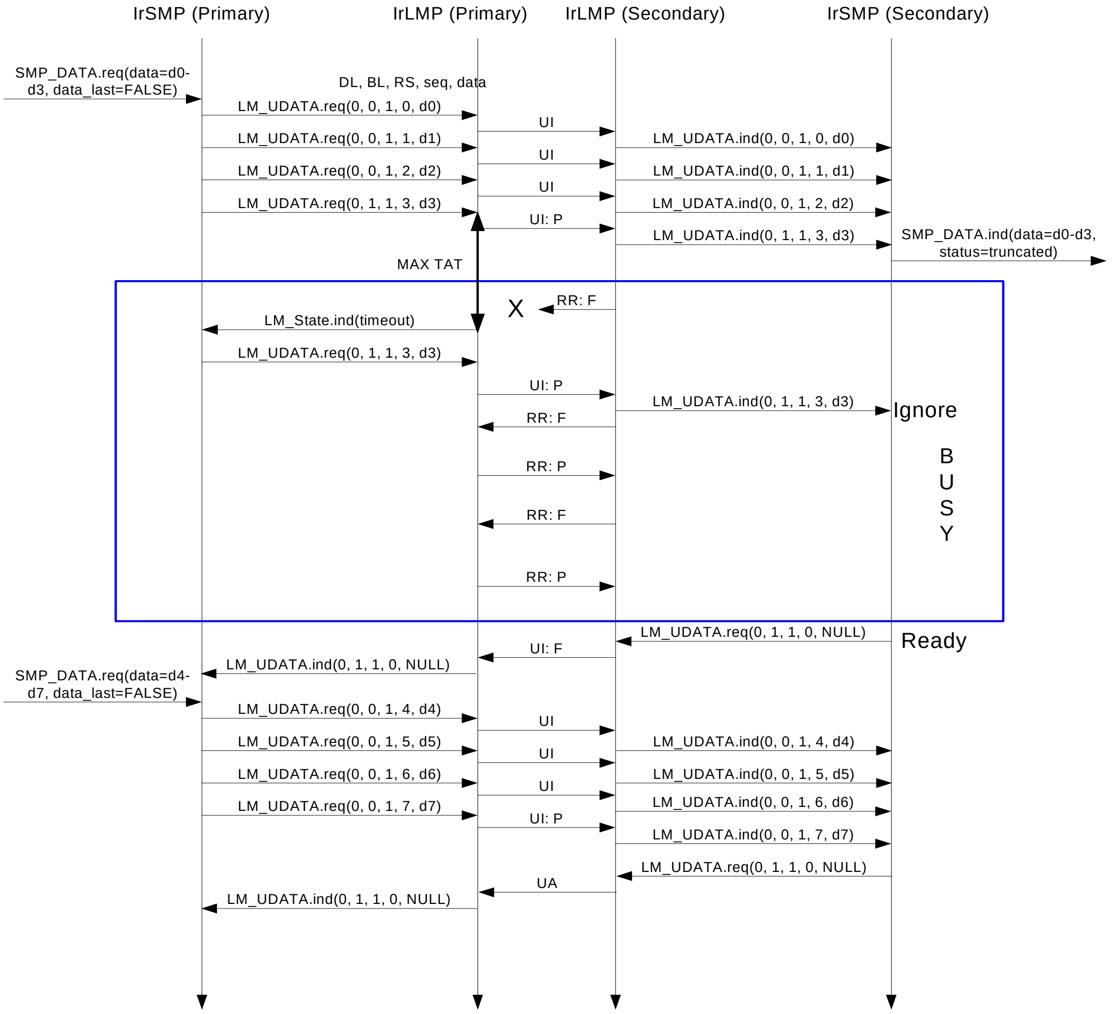


Figure 5.14 Recovery of Receive Ready (RR-F) frame loss in the proposed flow control scheme of IrSimple protocol

5.3.2.3.2 When Receive Ready (RR) frame from primary station is lost

In this case, the maximum turnaround time (MAX TAT) at primary station expires again as the IrLAP layer at secondary station is unable to detect the control rotation due to the loss of primary station RR-F frame and can not respond. IrLAP layer at primary resends the last data and waits for the response from secondary station. When secondary station IrLAP layer receives this frame, it issues a RR-F frame and the system returns to the usual state of the flow control. The operation of this recovery procedure is briefly explained in Figure 5.15

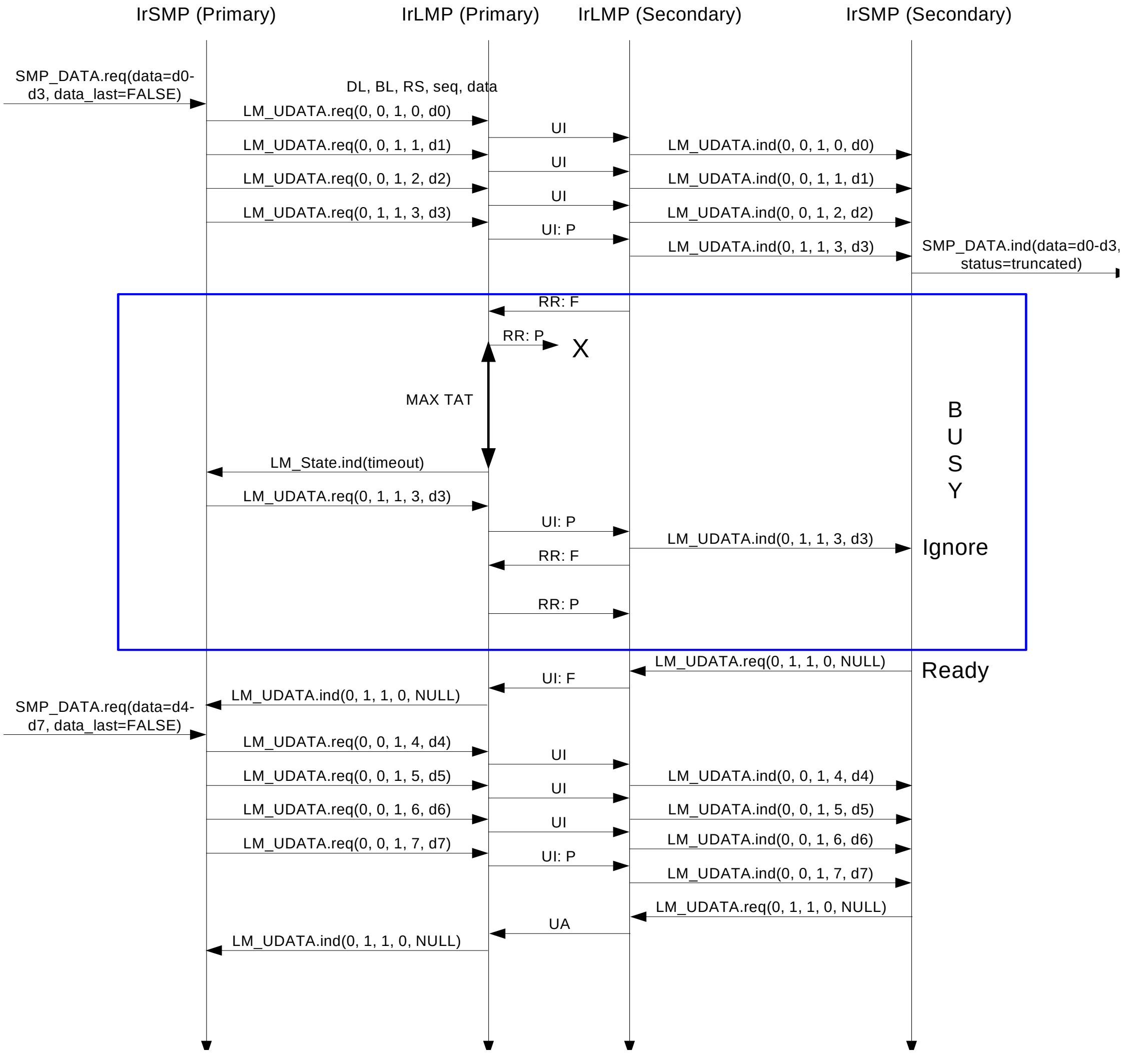


Figure 5.15 Recovery of Receive Ready (RR-P) frame loss

5.3.2.3.3 When Data (UI) frame is lost immediately after the release of the flow control

Figure 5.16 is an example for the loss of the UI frame that the secondary station issues immediately after the release of the flow control. IrLAP layer at primary station notifies a time-out when there is no response from the secondary station as the maximum turnaround time (MAX TAT) is expired. At this point, the system recovers the error the same way explained in previous sections.

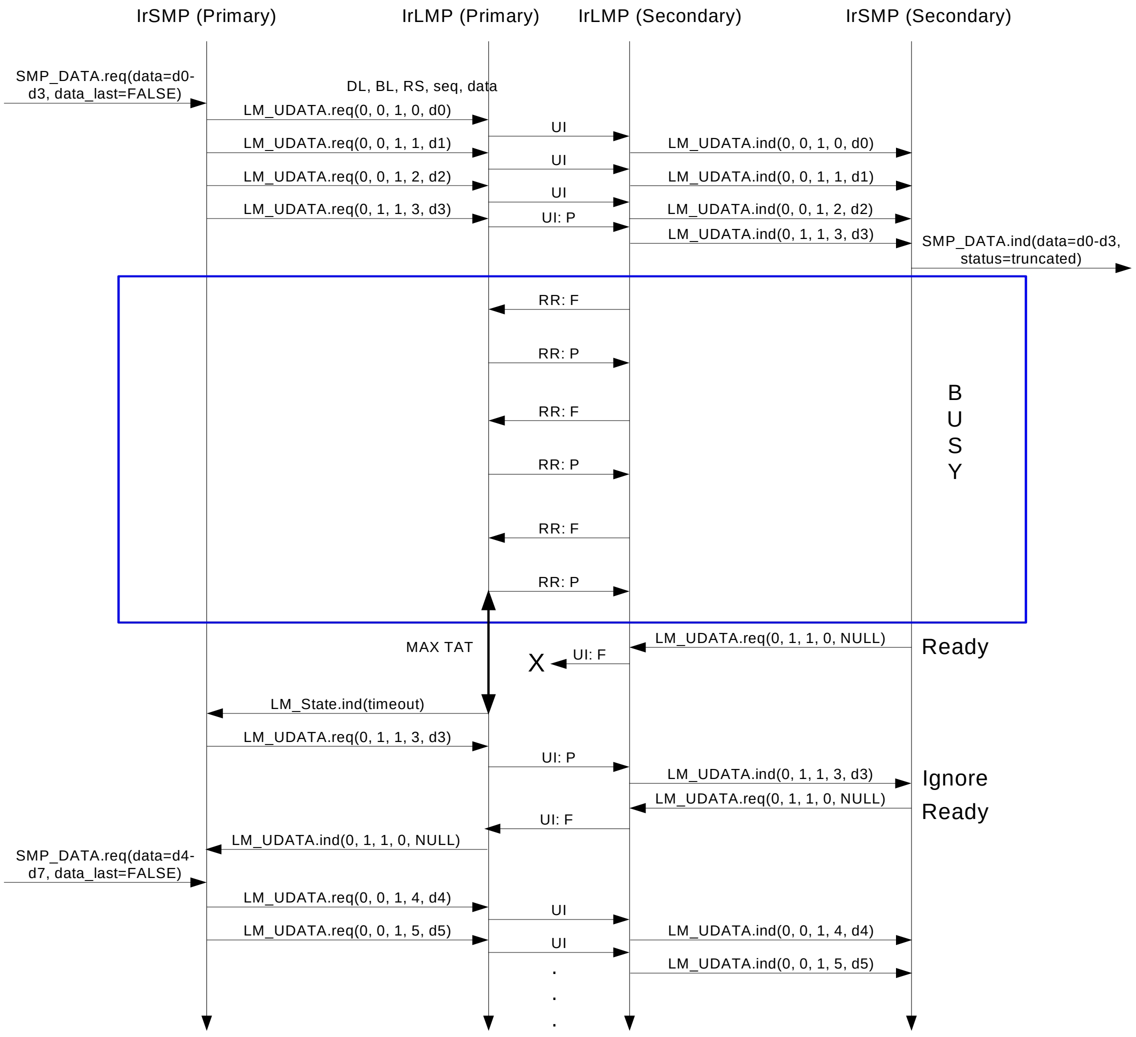


Figure 5.16 Recovery of Unnumbered Information (UI) frame loss immediately after the release of the flow control

5.4 Conclusions

In this chapter, I carried out a mathematical model to derive a simple equation for IrSimple throughput efficiency (TE) over the IrDA protocol stacks. Based on this model, the performance of IrSimple protocol is compared with existing OBEX protocol. It has shown that the IrSimple protocol scales well to handle fast data exchange at high data rates compared to the existing OBEX protocol. Furthermore, the effect of different layer parameters such as IrSMP block size and IrPHY minimum turnaround time on IrSimple throughput is explored for various BERs. Although small turnaround time results in maximum throughput, such a low turnaround time is not always achievable. Therefore, in situations where small turnaround time is not possible, the use of large block size can mitigate the negative effect of large turnaround time. An effective error control scheme at IrSMP layer is also proposed to improve IrSimple throughput performance at high BER when large block size is used. Simulation result also shows that employment of my proposed error recovery scheme highly improves throughput performance at high BERs. Finally, improvement in flow control for IrSimple protocol is presented by introducing RR supervisory frame at IrLAP layer to reduce the traffic in the system. A complete examination of the proposed flow control scheme is carried out for all possible cases where frame losses can occur due to transmission error. It shows that the proposed flow control scheme recovers from any possible frame losses without adding any complexity to the system as well as it reduces the redundant data retransmissions.

Chapter 6

Conclusion

6.1 Summary of the Studies

In this thesis, I propose more effective error recovery schemes, Block Based Window Retransmission (BBWR) scheme and Improved Selective Repeat (ISR) scheme, for future high speed IrDA links to enhance the reliability in case of erroneous environment. The proposed schemes are variants of ideal selective repeat ARQ scheme which fit well for half duplex IrDA links and require little modification to the existing error recovery scheme. BBWR scheme is suitable for system with large window size while ISR scheme is preferable for system with large frame length. The proposed schemes operate with a finite buffer size and a finite range of sequence number such that buffer overflow never happens. Simulation results are presented which shows that the proposed error recovery schemes significantly outperforms the conventional go-back-*N* ARQ scheme, particularly for links with high bit error rate. The significance of the minimum turn-around time and other link parameters on throughput performance are also examined.

In an effort to investigate the suitability of IrBurst protocol for large data block exchange over high-speed IrDA links, a complete analytical model was carried out to derive IrBurst throughput efficiency (TE) both in the case of error free transmission and in presence of transmission errors. Results are presented which reveal that IrBurst scales well to handle large data blocks and high data rates compared to the existing OBEX protocol. I also investigate the impact of IrLAP window size and frame length and IrPHY minimum turnaround time on IrBurst throughput efficiency (TE) in presence of transmission errors. The analysis has showed that IrBurst throughput efficiency (TE) always benefits by a small minimum turnaround time. However, in situations where small turnaround time is not possible, using a large window size or frame length can alleviate the negative effect of a large minimum turnaround time by increasing the amount of data sent between turnarounds. Although using large window size or frame length significantly improves the throughput efficiency for low BERs, it renders the link vulnerable to BER increase. To mitigate the loss of TE at high BER, the proposed BBWR and ISR ARQ schemes at IrLAP layer are employed both in case of large window size and large frame length respectively. Simulation result also shows that employment of the proposed ARQ schemes highly improves throughput performance at high BERs.

Finally, a study on the performance of IrSimple protocol for high-speed exchange of digital contents is carried out in detail. A mathematical model has been carried out to derive IrSimple throughput efficiency (TE). Based on the model, the performance of IrSimple protocol is compared with existing OBEX protocol for various data rates. Furthermore, in order to characterize the IrSimple performance in presence of transmission errors, a simulation model for IrSimple has been developed. A performance comparison of IrSimple and OBEX is carried out for various bit error rates based on the model. The significance of different layer parameters on IrSimple throughput in presence of transmission errors is also explored. In an effort to improve IrSimple performance, an effective error control scheme is proposed. Furthermore, an improvement in the existing flow control for IrSimple protocol is proposed at IrLAP layer to reduce the traffic in the system.

6.2 Future Work

In my future work, I consider the Forward Error Correction (FEC) scheme and hybrid ARQ scheme which is likely to provide further robustness for my proposed BBWR scheme and ISR scheme. This future study will be useful for providing more reliability for real time applications, for example, real time video and audio transmission over IrDA links for home theatre system.

Also I will investigate the performance of IrBurst protocol considering multiple applications and evaluate the performance of proposed flow control scheme for IrSimple protocol.

Bibliography

[1] Henry, P.S. and Hui Luo, “WiFi: what's next?,” IEEE Communications Magazine, vol.40, no.12, pp.66–72, Dec. 2002.

[2] Bluetooth specification, version 3.0 + HS, April 2009.

[3] New standard for Infrared wireless communication of 1 Gigabit/s speed, Press Release, 15 July 2009. (Source: IrDA official website at http://www.irda.org/)

[4] Lau, H.K., ”High speed short range systems for wireless personal area networks,” Wireless Telecommunications Symposium (WTS 2009), April, 2009.

[5] Heng-Te Li, Hao-Yu Chan, Chia-Wei Chao, and Wen-Piao Lin, ” Performance study of wireless universal serial bus transmission system,” 11th International Conference on Advanced Communication Technology, vol.1, pp.193–195, Feb. 2009.

[6] Krishna, P and Husalc, D, “RFID Infrastructure,” IEEE Communications Magazine, vol.45, no.9, pp.4 –10, September 2007.

[7] Baker, N., ”ZigBee and Bluetooth strengths and weaknesses for industrial applications,” Computing & Control Engineering Journal, vol.16, no.2, pp.20–25, May 2005.

[8] Liuqing Yang and Giannakis, G.B., “Ultra-wideband communications: an idea whose time has come,” IEEE Signal Processing Magazine, vol.21, no.6, pp.26–54, Nov. 2004.

[9] Binti Che Wook, H. Komine, T. Haruyama, S. Nakagawa, “Visible light communication with LED-based traffic lights using 2-dimensional image sensor,” IEEE Consumer Communications and Networking Conference, vol.1, pp. 243 – 247, Jan. 2006.

[10] R.W. Woodings, D.D. Joos, T. Clifton, and C.D. Knutson, “Rapid heterogeneous ad hoc connection establishment: accelerating Bluetooth inquiry using IrDA,” IEEE Wireless Communications and Networking Conference 2002 (WCNC 2002), vol.1, pp.342 – 349, March 2002.

[11] Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto, “Block based Window retransmission ARQ scheme for 100Mbit/s infrared links,” IEEE Tencon Conference (Tencon’05), Nov. 2005.

[12] Williams S., “IrDA: Past, Present and Future,” IEEE Personal Communications, vol. 7, No.1, pp.11-19, Feb. 2000.

[13] IrDA, Serial Infrared Physical Layer Link Specification, IrPHY, Version 1.1, 1995.

[14] IrDA, Serial Infrared Link Access Protocol (IrLAP), Version 1.1, 1996.

[15] IrDA, Link Management Protocol (IrLMP), Version 1.1, 1996.

[16] IrDA, Tiny Transport Protocol (IrTinyTP), Version 1.1, 1996.

[17] IrDA, Object Exchange Protocol (IrOBEX), Version 1.3, March, 2003.

[18] A. C. Boucouvalas and Pi Huang, “OBEX over IrDA: performance analysis and optimization by considering multiple applications,” IEEE/ACM Transactions on Networking, vol.14 , no.6, pp.1292 –1301, Dec. 2006.

[19] C.T. Deccio, J. Ekstrom, D.R. Partridge, K.B. Tew, C.D. Knutson, “ A study of the suitability of IrOBEX for high-speed exchange of large data objects,” IEEE Global Telecommunication Conference 2003 (GLOBECOM 2003), vol.5, pp.2664–2668, 2003.

[20] IrBurst High-speed Information Transmission Profile, Version 0.9, February, 2005.

[21] IrDA, Serial Infrared Physical Layer Link Specification, IrPHY, Version 1.2, 1997.

[22] IrDA, Serial Infrared Physical Layer Link Specification, IrPHY, for 16Mb/s Addition (VFIR) - Errata to Version 1.3, 1999.

[23] IrDA, Serial Infrared Physical Layer Link Specification, IrPHY, for 100Mb/s Addition (UFIR) – Proposal version 0.7, Feb. 2005.

[24] IrSimple, Infrared Simple Profile, Version 1.00, October 14, 2005.

[25] Robertson, M.G., Hansen, S.V., Sorenson, F.E. and Knutson, C.D., “Modeling IrDA performance: the effect of IrLAP negotiation parameters on throughput,” Tenth International Conference on Computer Communications and Networks, pp.122–127, Oct. 2001.

[26] IrDA Infrared Serial Link Access Protocol (IrLAP) for IrSimple Addition, IrLAP errata for IrSimple version 1.00, October 14, 2005.

[27] IrDA Serial Infrared Link Management Protocol (IrLMP) for IrSimple Addition, IrLMP errata for IrSimple version 1.00, October 14, 2005.

[28] IrDA, Infrared Sequence Management Protocol (IrSMP) for IrSimple, version 1.00, October 14, 2005.

[29] T. Ozugur, M. Naghshineh and P. Kermani, “Comparison of go-back-N and selective reject ARQ modes of HDLC over half-duplex and fullduplex IR links and the effects of window size and processor speed in utilization,” IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol.2, pp.708-712, Sep.1998.

[30] Shu Lin and Yu P., “A Hybrid ARQ Scheme with Parity Retransmission for Error Control of Satellite Channels,” IEEE Transaction on Communications, vol.30, no.I, pp.1701- 1719, July 1982.

[31] Yu P. and Shu Lin, “An Efficient Selective-Repeat ARQ Scheme for Satellite Channels and Its Throughput Analysis,” IEEE Transaction on Communications, vol. 29, no.3, pp.353-363, March 1981.

[32] V. Vitsas and A. C. Boucouvalas, “Optimization of IrDA IrLAP Link Access Protocol,” IEEE Transaction on Wireless Communications, vol.2, no.5, pp 926-938, Sep. 2003.

[33] OPNET^{TM} modeller, MIL3 Inc., 3400 International Drive NW, Washington DC 20008, USA.

[34] A. Varga. (2004) OMNET++: Discrete event simulation system. Technical University of Budapest. Version 3.1. [Online]. Available: http://www.omnetpp.org/

[35] Pi Huang and A. C. Boucouvalas, “Analysis of the High Speed Infrared Information Transmission Protocol: IrBurst,” in proceedings of Convergence of Telecommunications, Networking and Broadcasting Symposium (PGNET 2004), pp.328-332, June 2004.

[36] Pi Huang and A. C. Boucouvalas, “IrBurst Modelling and Performance Analysis in the Presence of Transmission Errors”, Wireless Personal Communications, vol. 41, no.1, pp.111–125, April 2007.

[37] A. C. Boucouvalas and V. Vitsas, “100 Mb/s IrDA Protocol Performance Evaluation,” IASTED International Conference on wireless and Optical Communications (WOC 2001), pp.49–57, June 2001.

[38] Mohammad Shah Alam, Shamim Ara Shawkat, Gontaro Kitazumi and Mitsuji Matsumoto, “IrBurst Modeling and Performance Evaluation for Large Data Block Exchange over High-Speed IrDA Links,” IEICE Transaction on Communications, Vol.E91-B, No.1, pp.274-285, Jan. 2008.

[39] Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto “Investigation of the Suitability of IrBurst for High-Speed Exchange of Large Data Blocks,” IEEE

International Conference on Communications (ICC 2006), Istanbul, Turkey, June 2006.

[40] Pi Huang and A. C. Boucouvalas, “OBEX Performance Evaluation and Parameter Optimization for High Speed IrDA Links,” IEEE International Conference on Communications (ICC 2004), vol. 7, pp. 3849–3853, June 2004.

[41] Mohammad Shah Alam, Shamim Ara Shawkat, Gontaro Kitazumi and Mitsuji Matsumoto “IrSimple Modeling and Performance Evaluation for High Speed Infrared Communications”, IEEE Global Telecommunication Conference 2006 (GLOBECOM 2006), November, 2006.

[42] Mohammad Shah Alam, Shamim Ara Shawkat, Gontaro Kitazumi and Mitsuji Matsumoto “Performance Evaluation of IrSimple Protocol and Its Efficiency Enhancement with Effective Error Control and Flow Control Schemes,” IIEEJ Journal on Visual Computing, Devices and Communications, vol.38, no.2, pp.145-155, March 2009.

Appendix A. List of Achievements

Category (Subheadings)	
Articles in Refereed Journals	○ Mohammad Shah Alam, Shamim Ara Shawkat, Gontaro Kitazumi and Mitsuji Matsumoto "Performance Evaluation of IrSimple Protocol and Its Efficiency Enhancement with Effective Error Control and Flow Control Schemes", *IIEEJ Journal on Visual Computing, Devices and Communications*, Vol.38, N0.2, pp. 145-155, March 2009. ○ Mohammad Shah Alam, Shamim Ara Shawkat, Gontaro Kitazumi and Mitsuji Matsumoto "IrBurst Modeling and Performance Evaluation for Large Data Block Exchange over High-Speed IrDA Links", *IEICE Transaction on Communications,* Vol.E91-B, No.1, pp.274-285 ,Jan. 2008.
International Conference Proceedings	○ Mohammad Shah Alam, Gontaro Kitazumi and Mitsuji Matsumoto," Future challenges of Irsimple protocol: Efficient flow control scheme and long distance capability", *ITU-T Kaleidoscope Academic Conference 2008*, Geneva, Switzerland, May 2008 ○ Mohammad Shah Alam, Pham Tien Dat, Kamugisha Kazaura, Kazuhiko Wakamori, Toshiji Suzuki, Kazunori Omae, Mitsuji Matsumoto, Yuji Aburakawa, Koichi Takahashi, Takuya Nakamura, Takeshi Higashino, Katsutoshi Tsukamoto, Shozo Komaki, "Characterization of RF signal transmission using FSO link considering atmospheric effects", *SPIE Photonics West 2008*, January 2008, San Jose, USA. ○ Mohammad Shah Alam, Shamim Ara Shawkat, Gontaro Kitazumi and Mitsuji Matsumoto "IrSimple Modeling and Performance Evaluation for High Speed Infrared Communications", published in the proceedings of *IEEE Global Telecommunication Conference 2006 (GLOBECOM 2006)*, San Francisco, USA, November, 2006. ○ Mohammad Shah Alam, Shamim Ara Shawkat and

	Mitsuji Matsumoto “Investigation of the Suitability of IrBurst for High-Speed Exchange of Large Data Blocks”, published in the proceedings of *IEEE International Conference on communications (ICC 2006)*, Istanbul, Turkey, June 2006. ○ Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto, “Performance Analysis of IrBurst for Large Data Contents Exchange over Infrared Links”, published in the proceedings of *First International Conference on Next-Generation Wireless Systems (ICNEWS’06)*, Dhaka, Bangladesh, January, 2006. ○ Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto, “An improved Selective-Repeat ARQ Scheme for IrDA Links at High Bit Error Rate”, published in the proceedings of *Fourth International Conference on Mobile and Ubiquitous Multimedia (MUM 2005)*, Christchurch, New Zealand, December 2005. ○ Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto, “Block based Window retransmission ARQ scheme for 100Mbit/s infrared links”, published in the proceedings of *IEEE Tencon Conference (Tencon’05),* Melbourne, Australia, November 2005.
Presentation at Standardization meetings	Mohammad Shah Alam and Mitsuji Matsumoto, “Performance of High Speed Infrared Communication“, IIEEJ Mobile Image Research Meeting and Workshop, 16 October, 2006, Tokyo, Japan Mohammad Shah Alam and Mitsuji Matsumoto “Proposal of enhancement of error control scheme for UFIR”, *IrDA General Meeting*, 30 May, 2006 San Francisco, USA. Mohammad Shah Alam and Mitsuji Matsumoto “High Speed IrDA QoS Evaluation”, *Asia Technology Summit for Next Generation Short Range Wireless Access Standardization*, 24 October, 2005, Tokyo, Japan.
National Conferences	Mohammad Shah Alam, Pham Tien Dat, Kamugisha Kazaura, Kazuhiko Wakamori, Toshiji Suzuki, Kazunori Omae, Mitsuji Matsumoto, Yuji Aburakawa, Koichi Takahashi, Takuya Nakamura, Takeshi Higashino,

Others	Katsutoshi Tsukamoto, Shozo Komaki, “ACLR Measurement of a FSO Communication System for W-CDMA Wireless Services”, *IEICE Society Conference* 2007, C-14-9 , September 10-14, 2007, Tottori University, Japan
	Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto “Sub-Window (SW) Based Error Control Scheme for Half Duplex Infrared Links with Finite Receiver Buffer”, CINAG 2005, 15 June, 2005, YRP
	Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto “Performance Analysis of IrBurst for High-Speed exchange of Large Data Objects”, *IEICE* Society Conference, 20 September, 2005, Hokkaido
	Mohammad Shah Alam, Shamim Ara Shawkat and Mitsuji Matsumoto “Improved Automatic Repeat Request (ARQ) Scheme for Half Duplex Infrared Links”, *IIEEJ*, The 33rd Media Computing Symposium 2005, 16 June, 2005, Nagano
	P.T. Dat, Mohammad Shah Alam, K. Kazaura, K. Wakamori, T. Suzuki, K. Takahashi, M. Matsumoto, Y. Aburakawa, T. Nakamura, T. Higashino, K. Tsukamoto, and S. Komaki: “A study on transmission of RF signals over a turbulent free space optical link”, International Topical Meeting on Microwave Photonics, Jointly held with the 2008 Asia-Pacific Microwave Photonics Conference (MWP/APMP 2008), pp. 173-176, Gold Coast, Australia, Oct. 3 2008.
	P.T. Dat, Mohammad Shah Alam, K. Kazaura, K. Wakamori, T. Suzuki, K. Omae, M. Matsumoto, Y. Aburakawa, K. Takahashi, T. Nakamura, T. Higashino, K. Tsukamoto and S. Komaki: “Investigation of Suitability of RF signal transmission over FSO links”, 4th International Symposium on High Capacity Optical Networks and Enabling Technologies (HONET 2007), Dubai, UAE, November, 2007.
	Shamim Ara Shawkat, Mohammad Shah Alam and Mitsuji Matsumoto, “An Efficient Selective-Repeat ARQ Scheme for Half-duplex Infrared Links under High Bit Error Rate Conditions”, published in the proceedings of *IEEE*

	Consumer Communications and Networking Conference (CCNC 2006), Las Vegas, USA, January, 2006. Shamim Ara Shawkat, Mohammad Shah Alam and Mitsuji Matsumoto, “A Performance Comparison of IrBurst and IrOBEX Protocol for High-speed Exchange of Large Data Contents”, published in the proceedings of *IEEE Sarnoff Symposium (Sarnoff 2006)*, Princeton, New Jersey, USA, March 2006. Shamim Ara Shawkat, Mohammad Shah Alam and Mitsuji Matsumoto “An Effective Error Control Scheme for No-Packet-Level Acknowledgement Protocol in Infrared Wireless LANs”, *IEICE* Society Conference, 20 September, 2005, Hokkaido.