An Experimental Improvement Analysis of Loss Tolerant TCP (LT-TCP) For Wireless Network

Md. Abdullah Al Mamun
Dept. of CSE, DUET
Gazipur – 1700, Bangladesh

Sumaya kazary
Asst. Prof., Dept. of CSE, DUET, Gazipur – 1700 Bangladesh

Momotaz Begum
Lecturer, Dept. of CSE, DUET, Gazipur – 1700 Bangladesh

Md. Rubel
Dept. of CSE, DUET
Gazipur – 1700, Bangladesh

Abstract—Now-a-days TCP is a famous protocol used in Internet but the main problem is packet losses due to congestion. In this thesis we proposed a new Loss Tolerant TCP (LT-TCP), an enhancement of TCP which makes it robust and applicable for extreme wireless environment. In the proposed LT-TCP two additional term, data and data header compression are added in existing LT-TCP. We reduce the total volume of data and packet size in our adaptive method which is able to minimize the congestion and increase the reliability of wireless communication. The ECN respond about random data packet loss and disruption process. The overhead of Forward Error Control FEC is imposed just-in-time process and target to maximize the performance even if the path characteristics are uncertain. This proposal show that it will perform better over regular TCP and it is possible to reduce packet losses up to 40-50%.

Keywords- LT-TCP; Mix reliability; Timeouts; Congestion avoidance; end-to-end Algorithms; Compression technique; Packet erasure rate; TCP SACK; RFEC; PFEC; RAR; ECN; throughput; RTT; Goodput.

I. INTRODUCTION

TCP is the most popular protocol to deliver data reliably regardless of the form and construction of the network [1]. When a data packet traverse a wireless link, a major fraction of packet losses due to transmission error [2]. TCP is robust in that it can adapt to disparate network conditions [4, 7]. LT-TCP uses an adaptive, end-to-end hybrid Automatic Repeat request/Forward Error Correction ARQ/ FEC [3, 4] reliability strategy and exploits Explicit Congestion Notification ECN for incipient congestion detection [5]. Congestion control is the major problem of managing network traffic where the total demand of resources such as bandwidth among the computing users exceeds the available capacity [8]. In terms of packet loss is occurred more often due to high Bit Error Rates (BERs) than due to congestion [9]. When using TCP over wireless network, it considers each packet loss as a sign of congestion and invokes congestion control measures at the source [6]. This results in severe performance degradation. To improve the TCP performance such as WLAN using congestion response, mix of reliability mechanism, time out avoidance and handle a large volume of data [10]. It studies as extensive literature on the performance of TCP and emphasis the ways distinguish the effects due to congestion. It address the TCP congestion avoidance and control issue over the wireless links from the end-to-end communication, thereby leading to efficient network resource utilization and improving the application response time. It also analyze widely used TCP end-to-end algorithm and presents a new technique that enables the TCP to better adapt to the wireless environment [11].

We provision a data compression technique which reduce the volume of data [12], hence reduce the total number of packets and proactive FEC in the original window as a function of the estimate of the actual packet erasure rate. Subsequently reactive FEC is used to mitigate the effects of erasures, during the transmission phase. An adaptive maximum segment size (MSS) provides a minimum number of packets in the TCP window, again seeking to risk of timeouts [10]. We seek to adaptively balance the FEC and packet overhead. While reducing the risk timeouts and also rapidly erased packets [3].

II. PROPOSED LT-TCP

A. Overview of Flow Graph

In accordance to Fig: 1 application data are providing compression technique which yield compressed data. This compressed data broken into TCP segments where the MSS is chosen to accommodate proactive FEC (PFEC) packets in the window. Reactive FEC (RFEC) packets are computed at the same time and held in reserve. Feedback from the receiver provides not only the loss estimate but also information (e.g. SACK blocks) that can be used to compute the number of RFEC packets to send for each block [10]. When the sender receives ACK, it determines the type of packets to send (Data/PFEC/RFEC) and transmits them. This provides self-clocking and follows the semantics of TCP behavior. LT-TCP comprises the following building blocks that complement each other and extend SACK to provide resilience.PFEC and RFEC help to data recovery. PFEC operates in conjunction with adaptive MSS and determined by the current estimates of loss rate. RFEC is computed based on feedback from the receiver and the loss rate estimate. The receiver can reconstruct the data packets as soon as any k out of n packets arrives at the receiver.

B. Data and Header Compression

Compression is a technique which reduces the volume of original information [7, 12]. This is done either to reduce the volume of size of text and images to be transmitted or to reduce the bandwidth that is required for its transmission audio and

video. It required less bandwidth during transmission it is also secured because of its encrypted form.

C. RAR Compression Technique

RAR is a powerful allowing you to manage and control archive files.

Data Type	Original Data Size	Compressed Data Size	Number Of Segment (MSS 1500 Bytes)	
			Before	After
Text	10 KB	4.3 KB	7	4
	10 MB	3.5 MB	6990	2567
	100 MB	47.9 MB	69905	33745
Image	100 KB	69.5 KB	65	46
	1MB	789 KB	687	535
	10 MB	8 MB	6989	5800
Audio	10 MB	8.9 MB	6991	5786
	100 MB	89.6 MB	69905	62633
	1 GB	985 MB	715800	688565
Video	10 MB	8.9 MB	6990	6222
	100 MB	91.9 MB	69905	64733
	1 GB	999 MB	715698	697444

Figure 1.   Compression comparison in terms of data size

Compressed output for different data type [12] Console RAR [12] supports archives only in RAR format, the names of which usually have a ".rar" extension. ZIP and other format are not supported.
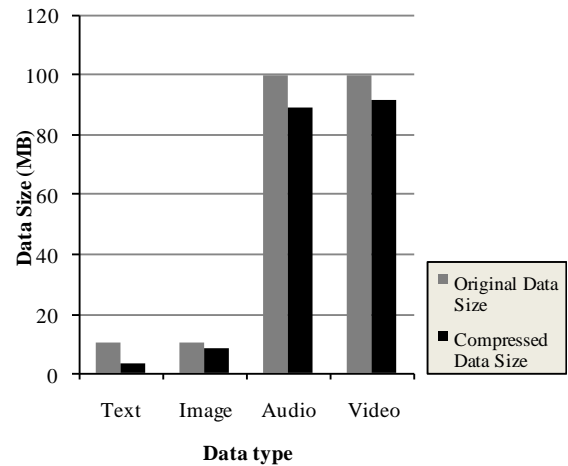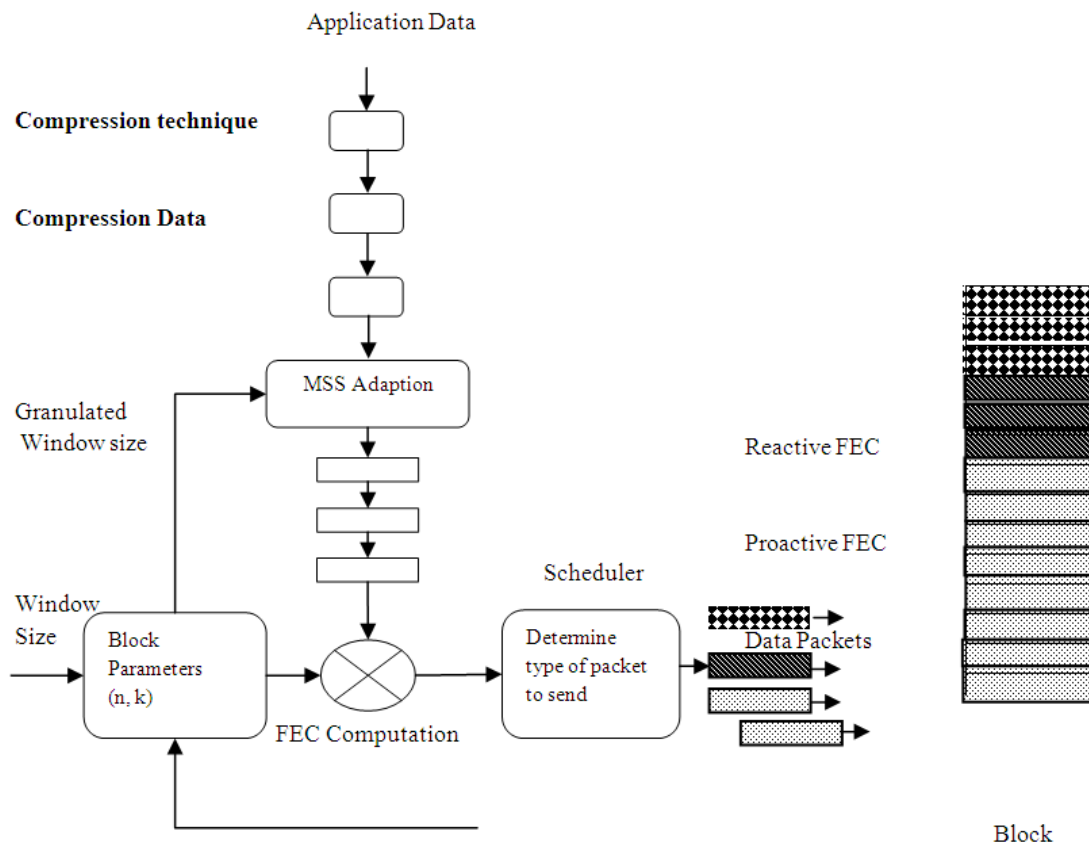

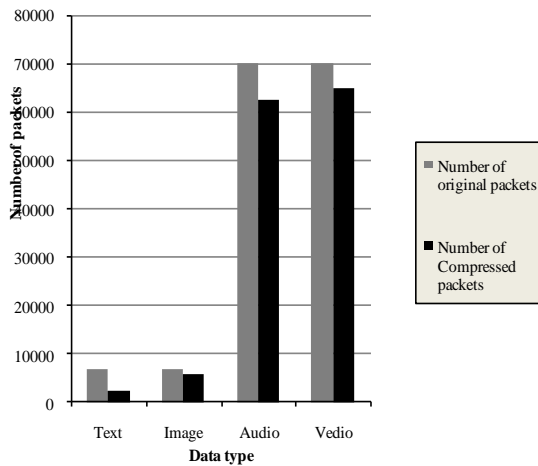Figure 2.   Proposed LT-TCP Flow graph

Figure 3.   Compression comparison in terms of number of packets.

D. Van Jacobson TCP/IP Header Compression

The IPv4 header is 20 bytes and when carrying UDP (8 bytes) and RTP (12 bytes) at least the packet header become 40 bytes. A compression scheme usually compresses such header to 2-4 bytes. On and average, considering a few uncompressed packets and few relatively large packets, more than 80% savings [7] can be observed. When the compared with the payload being carried, in such cases as voice type where payload size is usually static in range of 20-60 bytes, the header size presents a huge overhead. In header compression in such cases results is major bandwidth savings. The IPv6 with a header size of 40 bytes is gaining wide acceptance and has been included in Release 5 and onward version of 3G wireless network. In this case, header compressions yield in even more savings.

Figure 4.   Abstract view of header compression [13].

TABLE I.   THE HEADER COMPRESSION GAINS [13].

Protocol Headers	Total header size (bytes)	Min. compressed header size (bytes)	Compression gain (%)
IPV4/TCP	40	4	90
IPV4/UDP	28	1	96.4
IPv4/UDP/RTP	40	1	97.5
IPv6/TCP	60	4	93.3
IPv6/UDP	48	3	93.75
IPv6/UDP/RTP	60	3	95

These benefits lead to improved *QoS* in the network and the possibility for operators to improve their average revenue per user. The operators will be able to retain and attract customers with better *QoS* on the network and more services and content the links.

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

III.   PERFORMANCE ANALYSIS

A. Data & Header Compression Comparison

We use single bottleneck test case fig 5 with 10 flows and erasure rates varying from 0-50%. Host is ECN enabled; bottleneck implements RED/ECN on a 250 KB buffer (i.e. up to 500 packets of size of 500 bytes). *minthres* and *maxthres* values are as shown. According to table the 1, 2 and 3 we observed the performance for uncompressed data and compressed data & header both are given below

TABLE II.   PERFORMANCE OF UNCOMPRESSED DATA

Parameter	LT-TCP		
RTT	*10 ms*	*80 ms*	*200 ms*
Goodputs (MB/s)	5.4	4.5	3.9
Number of timeout	17	10	5
Throughput (MB/s)	7.4	6.3	5.1

TABLE III.   PERFORMANCE OF COMPRESSED DATA

Parameter	LT-TCP		
RTT	*10 ms*	*80 ms*	*200 ms*
Goodputs (MB/s)	7.3	6.5	5.1
Number of timeout	12	7	3
Throughput (MB/s)	9.5	7.3	6.4

From the above tables we observed that we compressed the data and header it mitigate the congestion and packet losses. The flow congestion is lasting for 100s for each operation. To assess the contribution of LT-TCP components, we use a 30% PER test case. Metrics include aggregate throughput, goodputs, number of timeouts and congestion window dynamics. We account for all packet header overheads.

B. Proposed LT-TCP vs. Traditional LT-TCP

In terms of packet erasure both perform well. However, the performance of traditional LT-TCP drops to 10% sometimes even more. In addition an error rate (40%) is sufficient to break a single LT-TCP connection due to repeated timeouts but proposed LT-TCP can reduce it significantly. Fig 6 shows different error rates for a number of RTT that means the degradation of performance is linear. Proposed LT-TCP manages packet error rates by avoiding timeouts and able to recover lost packets using proactive and reactive FEC even if RTT is too high.

C. Performance Evaluation

Finally we observe the Goodputs of both LT-TCP are shown in table IV. As we can see the proposed LT-TCP's Goodputs is higher than the traditional LT-TCP and performance is increasing with respect in time. It perform maximum at 80ms.

TABLE IV.   PERFORMANCE OF PROPOSED LT-TCP

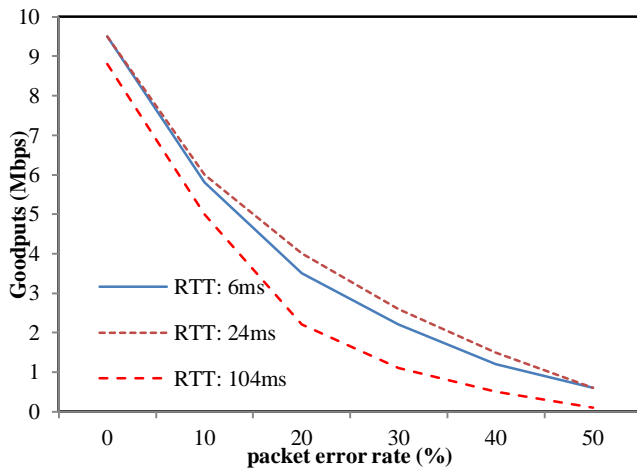| Parameter	LT-TCP		
RTT	*10 ms*	*80 ms*	*200 ms*
LT-TCP	5.4	4.5	3.9
Proposed LT-TCP	7.3	6.5	5.1
Increased performance (%)	19	20	12

Figure 5.   LT-TCP performance with increase erasure rate and RTT
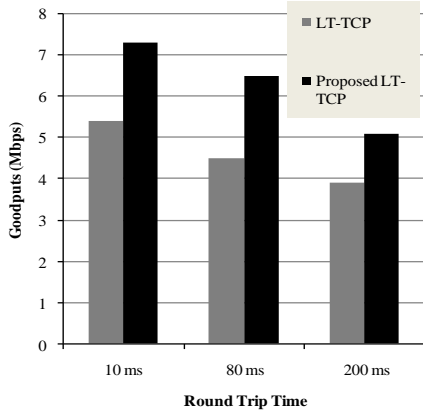

Figure 6.   Performance graph of proposed LT-TCP

IV.   CONCLUSION

Our TCP is the dominant reliable protocol used in internet; we have proposed a Loss-Tolerant TCP (LT-TCP) which introduces additional mechanisms as data & header compression in an adaptive manner. Our enhancement allow good performance even under demanding conditions through reduce the total volume of data that made less number of packet and packet size which mitigate congestion and perform better than traditional LT-TCP.

REFERENCES

[1]   Computer Networks by Andrew S. Tanenbum.

[2]   "TCP PERFORMANCE ENHANCEMENT OVER WIRELESS NETWORKS" By Aiyathurai ir.canterbury.ac.nz/bitstream/10092/1229/1/thesis_fulltext.pdf.

[3]   S. Biswas, G.Judd, D. Aguayo, J. Bicket and Morris: Link-level measurements from an 802.11b mesh network. In SIGCOMM, Aug'02

[4]   C. Ladas, R. M. Edwards AMIEE, M. Mahdavi, and G. A. Manson (2002): Class based selective ARQ Scheme for high performance TCP and UDP over wireless links. Paper presented at the mobile and wireless network, 2002. 4th international workshop on it.

[5]   K. K. Ramkrishnan S. Floyed and D. Black: The addition of Explicite Congestion Notification (ECN) to IP, Sept 2001.

[6]   C. Barakat and E. Altman: Bandwidth tradeoff between TCP and link level FEC. Computer Networks 39(2): 133-150, June 2002.

[7]   S. Casner and V. Jacobson: Compressing IP/UDP/RTP Headers for low speed serial links. RFC 2508, Feb 1999. IETF Network working group.

[8]   RFC 793 (1981). Transmission control protocol. From http:/www.ietf.org/rfc/rfc793.txt.

[9]   Data Communication and Networking by behrouz A. Forouzan.

[10] Omesh Tickoo, Vijaynarayanan Subramanian, Shivkumar Kalyanaraman and K. K. Ramakrishnan: LT-TCP: End-to-End Framework to Improve TCP Performance over Networks with Lossy Channels. In thirteen international workshop on Quality of Service (IWQoS 2005), Passau, Germany 2005.

[11] CHIU D. M and JAIN R.: analysis of increase and decrease algorithms for congestion avoidance in computer networks. Computer Networks and ISDN system, Vol 17 June, pp. 1-14

[12] Compression principles techniques and tools RAR.

[13] "White paper: An introduction to IP header compression" from EFFNET AB.

AUTHORS PROFILE

Md. Abdullah al mamun obtained his Bachelor of Science in Engineering degree from Department of Computer Science and Engineering (CSE) of Dhaka University of Engineering & Technology (DUET), Gazipur-1700, Bangladesh. At present he is performing extensive research on Network and Web Security, Wireless networking, Software Architecture, Machine Vision, Artificial Intelligence, Protocol cryptography. His key research interest includes Cryptography, analysis and Algorithm designs, E-mail: aamcse@gmail.com.

Momotaz Begum, Lecturer, Department of CSE, DUET and she achieved her B. Sc. in Engineering degree   from Department of Computer Science and Engineering (CSE), DUET, Bangladesh. E-mail: momotaz03_duet@yahoo.com

Sumaya kazary, Asst. Prof., Department of CSE, DUET and she achieved her B. Sc. in Engineering degree   from Department of Computer Science and Engineering (CSE), DUET, Bangladesh. E-mail: kazal_duet@yahoo.com