Performance Improvement of UDP and TCP Traffic by CJM Algorithm in Voice Transmission

GJCST Computing Classification
C 2.5,C 2.2, C 4

IKram Ud Din[1] Saeed Mahfooz[2] Muhammad Adnan[3]

***Abstract*-Different protocols are used at transport layer of the OSI model to transmit real time applications (i.e., video and voice). In this paper, two networks were analyzed in which the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP) were used for transportation. The purpose of this study is to evaluate the performance of TCP and UDP in real time transmissions for the parameters: Traffic Received, End-to-End Delay, and variation in delay (Jitter). At the receiver side, an algorithm called Chunk-based Jitter Management (CJM) [19] is applied on the buffer, which divides the packets into small size chunks and then plays them out. The evaluation of these transmission protocols has been done in OPNET. The graphs show the results in which the CJM algorithm performs well in end-to-end delay, jitter in voice packets, and voice packets receiving.**

Keywords: TCP, UDP, Transmission Protocol, Delay, Jitter

I. INTRODUCTION AND RELATED WORK

The most vital transport protocol used nowadays is TCP. However, the vigorous transportation protocol for the streaming media is UDP. Two prime reasons that UDP is not being used commonly are: i). some organizations are blocking this protocol and ii). it is not friendly to other flows. In the meantime, TCP is obviously reliable and friendly to other flows, but with so many basic controls in the protocol such as flow control, congestion control, and others with the heavy acknowledgement mechanism, resulting jitters and delays. Thus it is unsurprisingly not friendly to the real time application [1]. Recent years have witnessed increasing demand for multimedia information services and explosive growth of the Internet. Transmission of the real time applications via the Internet has received magnificent attention. In term of transport protocol, the hereditary problems are lacking of variability in bit rates, throughput guarantees, jitters or delays, and packet loss [2], [3]. Those characteristics are not ―friendly" to the real time data. Real time applications are able to compromise packet losses but sensitive to packet delays. Conventional perception holds that UDP is a better protocol than TCP for the transmission of critical data [4], [5]. This perception is straightforward to be understood because UDP is a best-effort delivery service. Theoretically, there will be less delay and provides better throughput. Unfortunately, this best-effort transport protocol potentially hinders the performance of other applications that employ TCP, or worse, jeopardize the stability of the Internet [6]. On the other hand, TCP employs congestion control schemes that vary dynamically to network conditions, and thus, it often yields jitters and packet delays [7]. Its reliability is naturally unsuitable for critical data (i.e., video and voice) [3]. Although the above-mentioned grounds seem to indicate that the present transport protocols, namely UDP and TCP, are not appropriate for real time applications. Many researchers also found that TCP is a more fashionable transport protocol than UDP [8, 9]. The researchers have justified their views by giving a reason on congestion control mechanism used by TCP regulates rate as a function of the number of packets sent by the application [1]. Other studies have found that video clips on the Internet nowadays are encoded at bit rates of 89-300 Kbps [10] ,[11]. The other interesting studies have also found that TCP lost packet recovery mechanism or retransmission is not too severe for real time applications [12].In recent years, Voice over IP (VoIP) has gained a lot of popularity. Session Initiation Protocol (SIP) developed by IETF for VoIP signaling is a communication control protocol used with transport layer's protocols, e.g., TCP and UDP. Today‗s SIP applications are mostly operating over the unreliable and connectionless transport protocol, UDP.

Because SIP establishes connection and TCP is also a connection-oriented protocol which brings delay and jitter in real time transmission, therefore, UDP is a suitable protocol for video conferencing [13].

The remaining paper is organized as: Section II describes the structure of the network. Section III illustrates architecture of the proposed system. The OPNET simulations and results are given in section IV. Section V concludes our work. The future work is described in section VI, and references are given in section VII.

II. STRUCTURE OF THE NETWORK

In the given scenario (Fig. 1), a network is established between the two cities of Pakistan, i.e., Karachi and Lahore. The scenario consists of servers and clients. One server and a client are located at each site. The servers are named as VoIP_Karachi and VoIP_Lahore. The simulator used in this work is OPNET Modeler 14.0. The packets were sent and received for 10 minutes from one place to another. Once the network was tested for the normal flow, and second time the CJM algorithm was applied on the receiver‗s buffer. The networks were named as Normal_Flow for normal transmission of the data, and Chunk_based for the network on which the CJM algorithm was applied as in [14]. Routing Information Protocol (RIP) is the protocol implemented for routing on both side routers.

About-[*]Department of Computer Science, University of Peshawar, Pakistan (e-mail-ikramuddin205@yahoo.com[1],saeedmahfooz@yahoo.com[2], adnan5283@yahoo.com[3])

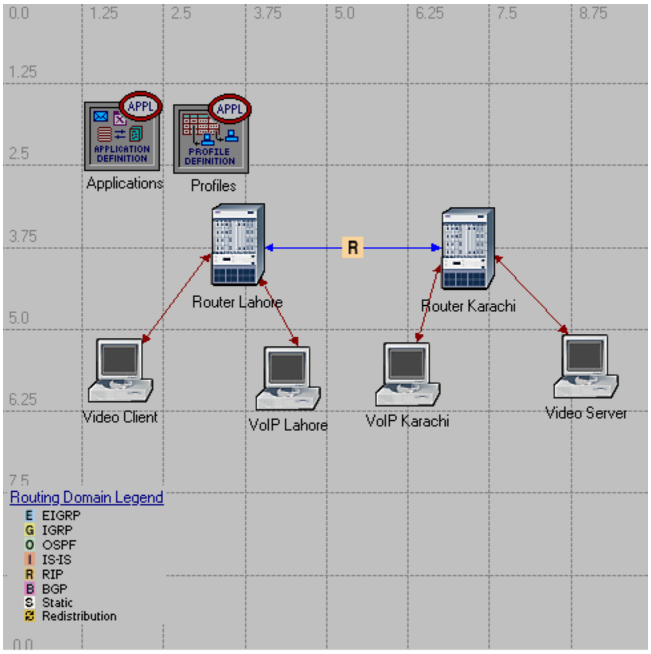

Fig. 1: Structure of the Network

Traffic Receive

Video or voice traffic is the total number of audio and video packets received during video conferencing or other type of real time communication (e.g., IP telephony) [14].

End-to-end Delay

End-to-end delay refers to the time taken by a packet to be reached across a network from source to the final destination. End-to-end delay depends on the end-to-end data paths/signal paths, the CODEC, and the payload size of the packets. Delay is the latency, one-way or round-trip, encounters when data packets are transmitted from one place to another. In order to maintain the expected voice quality for VoIP, the roundtrip delay must remain within approximately 120 milliseconds. [15], [14].

Jitter

In the context of voice over IP, jitter is the variation in delay of packets received, caused by network congestion or route changes. Jitter is a vital quality of service (QoS) factor in evaluation of network performance. It is one of the significant issues in packet based network for real time applications [16]. The variation of interpacket delay or jitter is one of the primary factors that agitates voice quality [17]. Jitter plays a vital role for the measurement of the Quality of Service of real time applications. The effect of end-to-end delay, packet loss, and jitter can be heard as: The calling party says “Good morning, everybody!” With end-to-end delay, the called party hears “…..Good morning, everybody!” With packet loss, the called party hears “Go….od… mor… ng ery body!” With jitter, the called party hears “Good…morning, eve…....ry… body!” [18], [14].

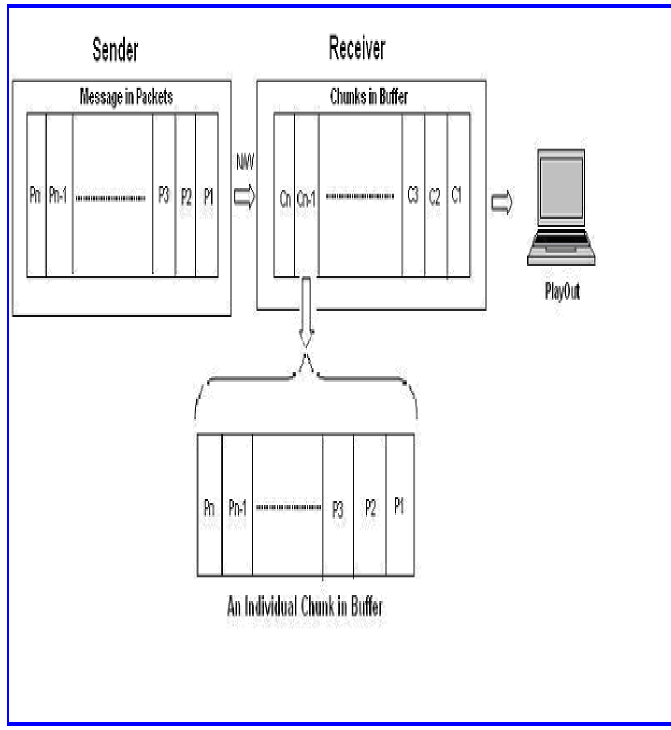

Fig. 2: Architecture of the Chunk-based Jitter Management Technique

III. SYSTEM'S ARCHITECTURE

Architecture of the CJM system comprises of three major components, i.e., sender, receiver, and network that are shown in fig 2.

A. Chunking of Packets

A voice message is mostly tightly coupled within a group of consecutive packets. Therefore, the proposed algorithm suggests chunking of the packets at the destination in the buffer. Thus, playing of a chunk, mostly, conveys a complete message without something missing. Hence, the QoS is improved while playing it out. All chunks are of the same size.

B. Algorithmic Outlines

The informal description of the proposed chunk-based jitter management algorithm in its pseudocodal form is given below [19]:

[Reading packets from network]
Read packets from the network and store in the buffer at the receiver.
[Chunking]
    Group the packets into same size chunks.
[PlayOut Chunks]
    Read chunks until buffer is empty i.e.
    do
    [Read Chunk]
    Read chunk from the buffer.

For (PacketCounter=1; PacketCounter<= ChunkSize; PacketCounter++)
[PlayOut Packets]
StreamOut Packets
End of For
ChunkCounter++;
While (ChunkCounter <= BufferSize)
[Store new Packets in the buffer from network]
Repeat step 1 to 3
[Stop]
Exit

IV. OPNET SIMULATION AND RESULTS

In this part, a scenario was examined in which jitter, delay, and packet receive rate were observed. The number of UDP and TCP packets received in Normal_Flow network is shown in figure 3. Figure 4 illustrates the voice traffic received when the CJM algorithm is applied on the receiver buffer. Jitter in the Normal_Flow network is given in figure 5, while it is shown in figure 6 when the CJM algorithm is applied on the buffer. The end-to-end delay in the Normal_Flow and Chunk-based networks is shown in figure 7 and 8, respectively.

A. Performance Evaluation

The number of voice traffic received in the Normal_Flow and Chunk-based networks are shown in figure 3 and 4, respectively. In the Normal_Flow network, there is a slight difference between the packets received through UDP and TCP, but a huge difference was observed when the CJM algorithm was applied on the buffer. The voice packet delay variation or jitter was also minimized through CJM algorithm (figure 6) as compared to the normal flow of data (figure 5). The end-to-end delay in TCP and UDP is almost the same in the Normal_Flow network as shown in figure 7, but it is quite improved in the chunk-based network.figer 8, In the given diagrams, the X-axis shows the amount of simulation time and the Y-axis shows the number of packets per second in figure 3 and 4, while the value of jitter in seconds in figure 5 and 6, and the value of delay in seconds in figure 7 and 8, respectively.

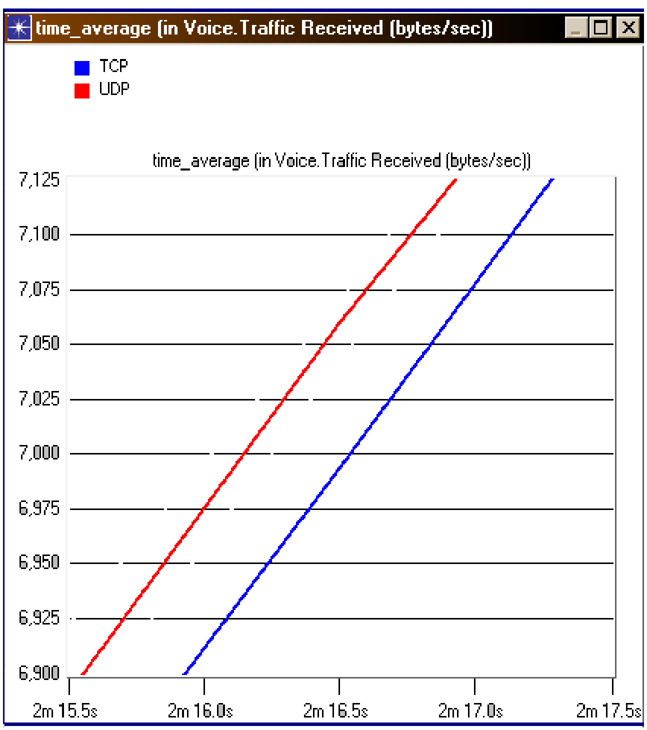

Fig. 3: Voice traffic received in Normal Flow

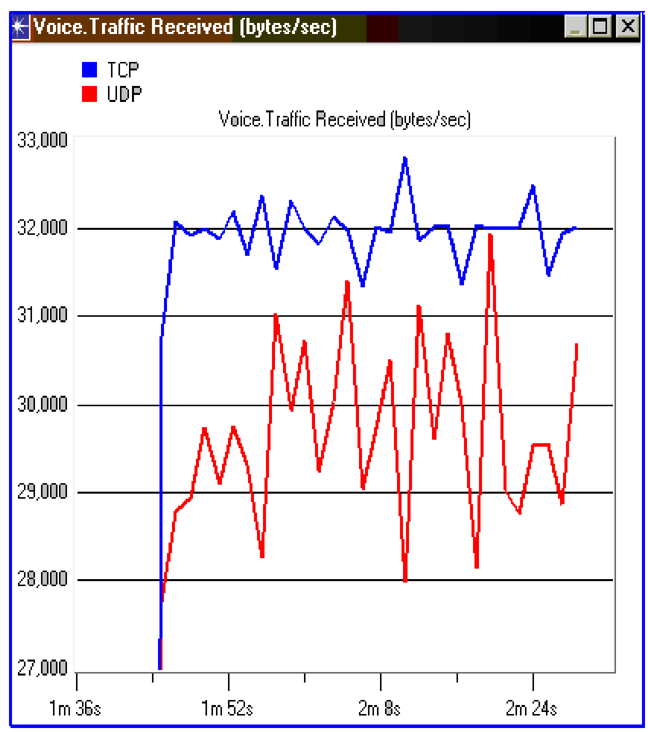

Fig. 4: Voice traffic received when CJM algorithm is applied on the buffer

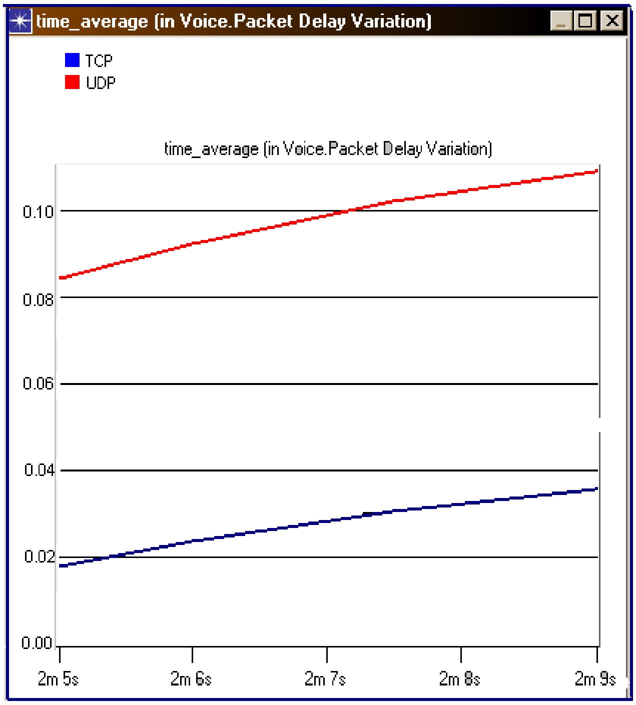

Fig. 5: Voice Packet delay variation in Normal Flow

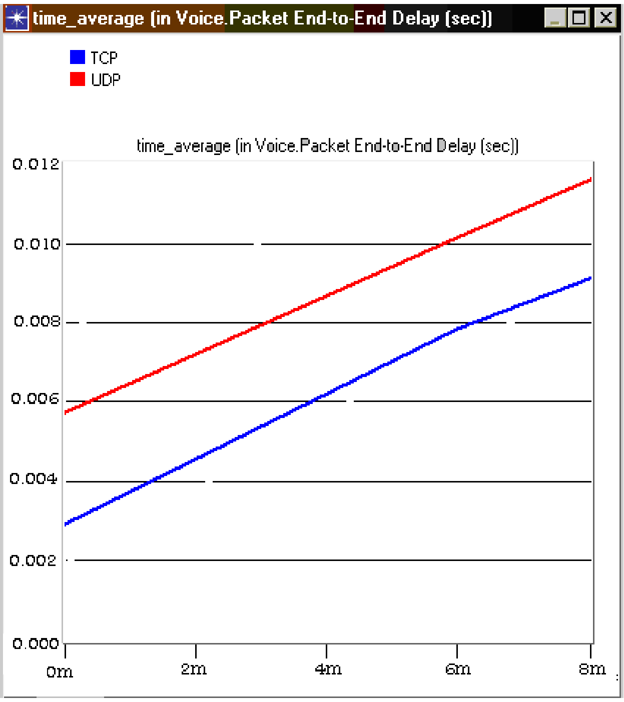

Fig. 7: Voice Packet end-to-end delay in Normal Flow

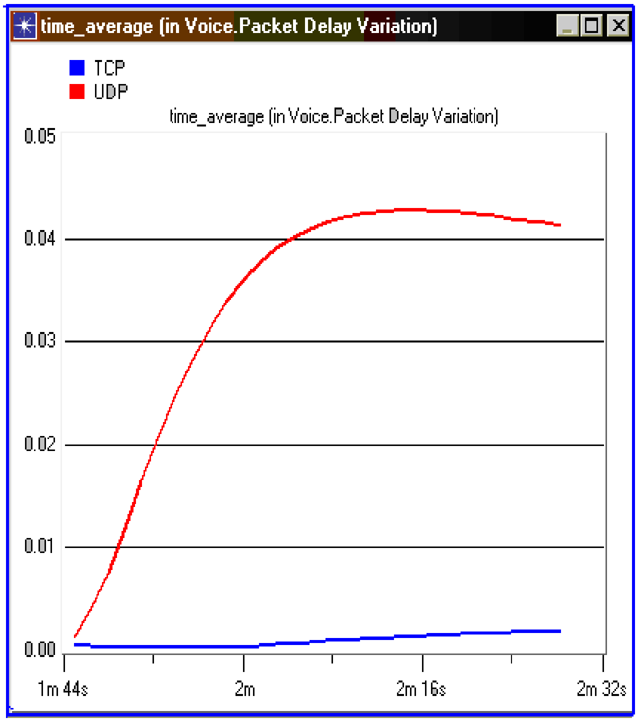

Fig. 6: Voice Packet delay variation after applying CJM algorithm on the buffer

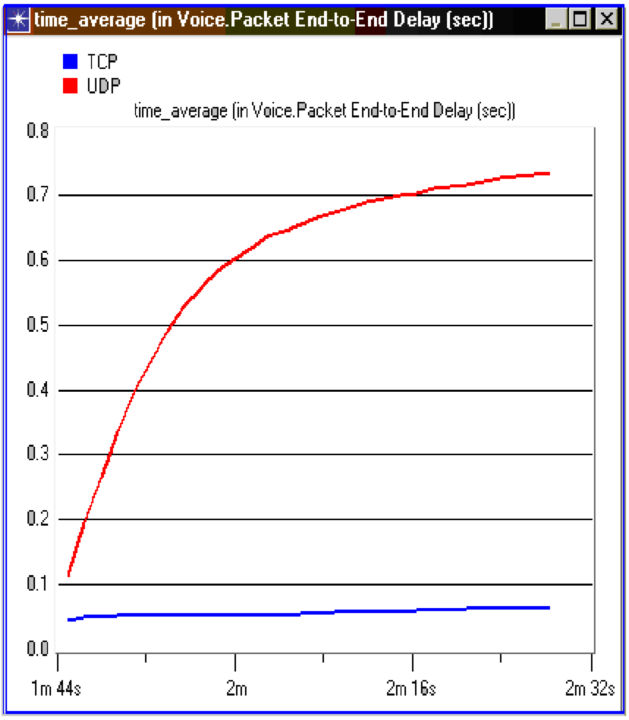

Fig. 8: Voice Packet end-to-end delay after CJM algorithm is applied on the buffer

V. CONCLUDING REMARKS

Today's networks size has been growing rapidly and support complicated applications, e.g., voice messages and video conferencing. Quality transmission is demand of the time. This needs some good results during transmission produced by transport protocol. The work done in this paper evaluates the available transport protocols: UDP and TCP for traffic receiving, jitter, and end-to-end delay. Our work for each of these parameters is based on OPNET simulation. The study presents a comprehensive result both for TCP and UDP against the parameters traffic received, jitter, and end-to-end delay one by one. After arriving packets at the destination, the Chunk-based Jitter Management (CJM) algorithm is applied on the buffer. The simulation results show that CJM algorithm performs better as compared to the normal flow of data

VI. FUTURE WORK

As for feasible future work, we are planning to carry on with the implementation of Stream Control Transmission Protocol (SCTP) which has both the qualities of UDP and TCP, and combine it with the existing cooperative transport protocols. In this way, we will implement accurate cooperative mechanisms that will further improve network performance.

VII. REFERENCES

1) Suki M. Arif, O.G., Suhaidi Hassan, A Survey on Buffer and Rate Adaptation Optimization in TCP-Based Streaming Media Studies, in International Conference on Network Applications, Protocols and Services. 2008, Executive Development Center: Malaysia.
2) M. Prangl, I.K. and H. Hellwagner, Towards QoS Improvements of TCP-Based Media Delivery, in Networking and Services (ICNS 2008). 2008. p. 188-193.
3) H. Sun, A. Vetro. and J. Xin, An Overview of Scalable Video Streaming: Research Articles. Wireless Communications & Mobile Computing, 2007. 7: p. 159-172.
4) E . Brosh, S. A. Baset, D. Rubenstein, and H. Schulzrinne, The Delay-Friendliness of TCP, in SIGMETRICS. 2008.
5) S. Boyden, A.M. and C. Williamson, TCP Vegas Performance with Streaming Media, in Performance, Computing, and Communications Conference (IPCCC 2007). 2007. p. 35-44.
6) S. Floyd and K. Fall, Promoting the use of end-to-end congestion control in the Internet. IEEE/ACM Trans. Netw, 1999. 7: p. 458-472.
7) Wang, W. Wei., et al, Application-layer multipath data transfer via TCP: Schemes and performance tradeoffs. Perform. Eval, 2007. 64: p. 965-977.
8) L. Guo, E. Tan, S. Chen, Z. Xiao, O. Spatscheck, and X. Zhang, Delving into internet streaming media delivery: a quality and resource utilization perspective, in 6th ACM SIGCOMM conference on Internet measurement. 2006, ACM: Rio de Janeriro, Brazil. p. 217-230.
9) Wang, J. Kurose, P. Shenoy, and D. Towsley, Multimedia streaming via TCP: an analytic performance study, in SIGMETRICS: Perform. Eval. Rev. 2004. p. 406-407.
10) M. Li, M. Claypool., R. Kinicki, and J. Nichols, Characteristics of streaming media stored on the Web. ACM Trans. Inter. Tech, 2005. 5: p. 601-626.
11) L. Guo, S. Chen., Z. Xiao, and X. Zhang, Analysis of multimedia workloads with implications for internet streaming, in 14th international conference on World Wide Web. 2005, ACM: Chiba, Japan. p. 519-528.
12) Loguinov and H. Radha, in INFOCOM. 2002: Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies.
13) Rosenberg, J., et al, SIP Session Initiation Protocol, in KFC3261, IETF, Editor. 2002.
14) IU. Din, S. Mahfooz, and M. Adnan, Analaysis of the Routing Protocols in Real Time Transmission: A Comparative Study. Global Journal of Computer Science and Technology, 2010, Accepted for Publication.
15) Paul J. Fong, E. Knipp, et al, Configuring Cisco Voice Over IP, Second Edition.
16) 1C. Demichelis and P. Chimento, IP Packet Delay Variation Metric for IP Performance Metrics (IPPM). Request for Comments: 3393, 2002.
17) Pedrasa, J.R.I. and C.A.M. Festin, An Enhanced Framing Strategy for Jitter Management, in TENCON 2005 IEEE Region 10. 2005: Melbourne, Qld. p. 1-6.
18) Cisco Systems, Inc, Cisco Voice Over IP. Student Guide, Edition V. 4.2. 2004.
19) IU. Din, S. Mahfooz, and M. Adnan, ―AChunk-based Jitter Management Algorithm for Voice Transmission", Global Journal of Computer Science and Technology, 2010. Accepted for Publication.