A Holistic Study of VoIP Session Quality—The Knobs that Control

R. Venkatesha Prasad*, Vijay S.*, H. N. Shankar‡, Przemysław Pawełczak*, R. Muralishankar‡, Ignas Niemegeers*
*Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, Mekelweg 4, 2600 GA Delft, The Netherlands
Email: {vprasad, p.pawelczak, i.niemegeers}@tudelft.nl, vijaysathyanarayanarao@student.tudelft.nl
‡Faculty of Telecom Engineering
PES Institute of Technology, Bangalore - 560085, India
Email: {hnshankar, muralishankar}@pes.edu

***Abstract*— VoIP packets, when transported over the Internet, experience loss and variable delay. The effect of the network not only depends on the background flows but also on the parameters of VoIP packets itself, such as VoIP packet size and the packet generation intervals. While higher sized packets experience more losses, they experience less delay jitter and handling them is thus easy at the playout buffer. To investigate the effect of various network conditions on VoIP session holistically, we present a complete end to end study considering various states of the underlying network. We present as a case study of G.711 coded packets generated at 20 and 40 ms intervals for comparison. While packets carrying 20 ms data are better when the network is loaded, 40 ms packetization is favored when the network is not saturated. This affects the jitter and loss thus affecting the quality. We explain this trade-off using Mean Opinion Scores.**

I. Introduction

Long distance calls are expensive when transported over Public Switched Telephone Network (PSTN). The current trend is to provide this service on data networks, especially in the light of popular applications like Skype or Google Talk. The integration of voice and data communication enriches the user experience and enhances the interactivity by allowing sharing of multiple types of media. IP suite, originally built for data traffic, works on the best effort delivery principle. Since resources are shared through statistical multiplexing in the IP networks, the total number of calls supported on such networks may be enhanced using speech compression techniques with codecs like G.723 or G.729, and with the use of silence suppression. Even after compression, codecs like iLBC [1], assuming sufficient error correction, can provide almost toll quality speech. While service providers are interested in number of calls supported, users are concerned about the quality of experience of such calls.

Data networks do not guarantee a faithful voice transmission and reproduction as in PSTN. The voice packets are usually transported using connectionless protocol such as UDP. Datagram packets are not guaranteed to reach destination, and also the network characteristics do not offer any predictability of the voice traffic. Many aspects affects the mouth to ear voice quality and with important ones being *packet loss*, *delay* and *delay jitter*. Packet loss can cause cuts in voice reception and delay can make a session less interactive. The delay jitter can cause havoc for the design of the playout buffer, which is needed to buffer a few initial packets to ensure a continuous playout. Higher playout buffer size offers increased tolerance towards jitter but increases mouth to ear delay. One simple way to reduce the delay at the playout buffer is to detect the talk spurts [2] and transmit only those segments. This scheme, while reducing the bandwidth, avoids building up of playout buffer. Unfortunately this is not sufficient, and thus the effects of packet delay, loss and delay jitter need to be kept under control. Techniques discussed until now can be implemented at the application layer assuming that the network layer provides the necessary bandwidth. However, understanding and analyzing the effects of the network on the voice packets comprise an important ingredient for the design of a successful VoIP application. Applications should be able to adapt to the conditions at the lower layers. Measurements at the application layer enable fine tuning of the application without interacting with the lower layers. This work brings out the nexus between the packetization interval, playout buffer and the background network traffic. If the network layer provides QoS support, it can be additionally leveraged for a good quality of speech by appropriately adapting at the higher layer. Issues of concern in this study are not only delay, delay jitter and packet loss at various stages, but also the size of the packets and in turn packetization interval. Packetization interval is the time between the generation of two consecutive packets; thus if packetization interval is higher then the packet size would also be higher. The above issues have been analyzed in depth by various researchers and umpteen number of studies are available. However, they are commonly seen in isolation. We take a view that these characteristics are dependent on each other and thus their effects are to be considered holistically.

Authors would like to thank EU-funded Magnet Beyond project and Adaptive Ad-Hoc Free Band Wireless Communications (AAF) project funded by the Freeband program of the Dutch Ministry of Economic Affairs. The work presented here does not necessarily reflect the views of Magnet Beyond and AAF.

1-4244-1457-1/08/$25.00 © IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2008 proceedings.

Further, since the network carries many traffic streams, the voice packets experience many variations vis-á-vis packets belonging to other sessions. We also attempt to discover some of the effects of other traffic streams in the Internet on the VoIP packets by measurements using application similar to [3]. Later these measurements are used to characterize and drive a simple simulation model to gain insights into the behavior of packets, which may be useful for working with real networks. Simulation model is mandated to keep the background network traffic same across experiments while other parameters are changed. With only real measurements we will not be able to achieve this.

Interrelated parameters of VoIP packets with relation to the network are: (a) larger size packets experience higher packet drop but lesser delay jitter; (b) larger size packets inherently induces higher delay and thus less interactivity; (c) smaller sized packets provide better interactivity due to lesser delay but decrease the throughput of the network, since RTP header of 12 B and UDP and IP headers of 28 B are added to every VoIP packet; (d) smaller size packets experience higher delay jitter. This study throws light on the intricate relationships of these parameters and their effects on the quality of VoIP sessions.

This paper is organized as follows. First in Section II we discuss some of the parameters of realtime traffic in the Internet. We bring out clearly their influence and their importance for session quality. Then the VoIP traffic characteristics found from measurements across the Internet are discussed in Section III. Later in Section IV a simple model is developed to find the complex relations between various parameters. Also in this section a case study for computation of Mean Opinion Score (MOS) is presented with respect to packet loss, delay and jitter. Finally we discuss the observations and conclude in Section V.

II. Issues in Realtime Traffic over IP Networks

The key problems for multimedia delivery over the Internet, especially VoIP packets, are the average end to end transit times, its variations, out of sequence and duplicated packet deliveries and packet loss. Some of these aspects are discussed below in detail.

A. Delay

Delay, in the context of this paper, is the time taken by a packet to transit from sender's mouth to receiver's ear. For a good quality interactive conference, the end-to-end delay should be less than 200 ms. However, for international calls it may be relaxed to 400 ms [4], [5]. End to end delay is composed of: the time to sample audio, the time to compress and decompress audio, the network propagation time (including network access), the time spent in router queues, operating system imposed delays in the process scheduling, and receiver buffering (playout delay). Delay is affected by number of hops between sender and receiver and also the type of connectivity.

A method for calculating transit time is to synchronize the sender's and receiver's clock. Accuracy in this scheme is difficult because synchronization between source and destination in the range of few millisecond is difficult to achieve. Network Time Protocol (NTP) [6] is used for synchronization in the Internet. Although packet structure of NTP is rich and supports precision upto less a microsecond, it is not supported by all operating systems to meet this bound thus NTP is not widely deployed for time synchronization. To overcome this problem, end to end delay is measured using echo-based techniques. Echo-based techniques suffer because of different properties of the network path in the forward and reverse direction. Studies have found that paths through the Internet are often asymmetric, i.e. delays measured are not the same when router sequence is not identical in different directions.

Reducing coding delay at the application layer and playout delay at the receivers can reduce the end to end delay. The playout buffer introduces more delay if the packetization interval is higher. Therefore, it is recommended to use smaller packetization interval (thus smaller packets). However, this causes higher bandwidth requirement (see Section I) and also relatively higher jitter (see Section II-B). Significant reduction in delay can be achieved only if the network provides enough QoS support.

B. Jitter

Jitter is the variation in delay that successive packets experience. It is due to the variable time that each packet spends in the service queue of routers. Waiting time in the router queue depends mainly on the scheduling algorithm and the load on routers at that instant. In hypothetical case where delays are same for all the packets, a playout buffer at the receiver is not required. Packets can be played out as and when they are received. In practice, jitter is a serious problem as it can result in gaps in the played out audio. To avoid these gaps, enough data is buffered in the jitter buffer at the receiver before starting the playout. Thus deferred playback evens out the breaks at the cost of higher delay seen by a few packets.

Jitter is directly concerned with the inter arrival time of packets and hence it is referred to as inter arrival jitter in [7]. It is defined as the mean deviation (smoothed absolute value) of the difference, $D_{i,j}$, in packet spacing at the receiver compared to that at the sender for a pair of packets. It is equal to the difference in the "relative transit time" for the two packets. Relative transit time is the difference between a packet's RTP timestamp and the receiver's clock at the time of arrival. For packet i, if S_i is the RTP timestamp and R_i the time of arrival, then for two packets i and j, D_{ij} may be expressed as $D_{i,j} = (R_j - R_i) - (S_j - S_i) = (R_j - S_j) - (R_i - S_i)$.

To even out delay jitter, there can be a fixed length or adaptive playout buffer. Fixed length jitter buffer is not advisable as the network conditions dynamically change. Though adaptive playout buffer can be easily implemented for a receiver handling a single stream, it is difficult for a conference bridge/server receiving multiple streams from many clients, since maintaining an adaptive buffer for each stream introduces computational overhead. Since the focus of this paper is to study the effect of delay jitter with respect to the packet size and packet formation interval, we assume a constant playout

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2008 proceedings.

buffer at the receiver for finding the effect of the network, packet loss, delay, delay jitter and packet size simultaneously.

Assuming the network characteristics almost constant, the variation in the delay seen by two consecutive packets would be the same for any packetization interval. However, the effect of delay jitter is higher, if the deviation over the mean is higher compared to the packetization interval. For example, suppose that the delay jitter is 60 ms. Under the ideal conditions the packets carrying 20 ms and 40 ms of data experience the same delay jitter. To cancel a delay jitter of 60ms when each packet carries 20ms data, 3 buffers are needed. However, for 40 ms packets only 2 buffers are required. Therefore, a smaller playout buffer would induce packet loss at the playout buffer. *Pari passu* the higher sized packets experience more delay at the buffer. This is an important observation we make in this study and we present its effects under various scenarios later in Section IV.

C. Packet Loss

Realtime multimedia applications usually react to network congestion with less flexibility due to their more stringent timing constraints. Packets from a realtime multimedia application must arrive at the receiver before the scheduled playout time, otherwise they are considered lost. Packet loss may occur due to the congestion in the network or at the playout buffer due to buffer overflow. Many studies consider only the packets dropped in the network while providing solutions to recover the loss. However, we also take the packet drop at the receiver also into account since the net effect for the user is the same, whether the packet is dropped in the network or at the client (receiver). Realtime traffic can tolerate packet loss to a certain extent at the cost of degraded quality. However it is a serious problem in multimedia applications because many codecs rely on the continuity of the codec states between consecutive frames. Packet loss disrupt these states, leading to mistracking at the decoder (some refined packet loss tolerant codecs are available now, such as iSAC or iLBC). In fact, higher compression codecs relay more on the redundant data in the speech or on the model of the vocal tract, and thus they have more dependency on the previous frames. As the congestion is unavoidable, interactive multimedia applications need to survive the congestion period and minimize its impact on the quality.

Applications can choose to ignore lost packet or recover from them - by retransmission or by receiver based error recovery or masking. Considerable study has gone into understanding UDP packet loss in the Internet since UDP is used to transfer data in realtime multimedia applications [8]–[13]. Packet loss and delay are correlated [8]. It is also observed that packet loss increases with the size of packets [9] in the Internet. A recent study of packet size distribution and its effect on packet losses with respect to loss concealment can be found in [13]. A number of alternative approaches to solve this are available for example, using codecs with loss concealment. In the next section the packet loss observed at the receivers for real VoIP traces across the Internet is presented in detail.

III. Packet Loss Measurements with Real Traces

The packet loss in the network is not in user control. The network characteristics changes with time and background traffic load. We take into account the packet loss at the receiver due to playout buffer (the packet loss due to network is considered in the next section). As we have already seen (in Section II-B) that due to the various background traffic flows the realtime media packets undergoes variable delay in the network. Assuming that the network characteristics are unchanged, we want to know the effect of delay jitter at the playout buffer at the receiver. As discussed earlier, we also consider the packetization interval with respect to delay jitter.

The experiment were conducted on the VoIP testbed using VQube VoIP system [3]. Buffer size at the receivers were fixed. Two end points were chosen so as to make audio stream pass through many hops to capture the Internet characteristics. Traces of audio traffic were captured by running *tcpdump* on a third computer running Linux operating system on the same LAN. Experiments with different packet sizes were performed. Each end terminal logged the delay, predicted delay and sequence numbers of the packets. We initially considered 20 ms packetization interval since many codecs use this as frame duration. For getting a better insight into the effects of packetization interval, we also considered multiples of 20 ms, i.e., 40 ms and 60 ms. However, we present here a comparison of 20 ms and 40 ms packetization intervals.

To record traces, calls were established between Bangalore, India and Zurich, Switzerland and Iowa in the USA. *traceroute* performed showed that the hosts were 13 and 8 hops away. These traces are used to find inter-arrival times of packets at the receiver. Packets with inter-arrival times of these traces are sent to playout buffer of different sizes.

Histogram of inter-arrival time of one such trace with packetization interval of 40 ms is shown in Fig. 1(a). It shows that packets are densely populated around 40 ms in this case with the peak number of packets at approximately 39.8 ms. Average percentage of packet loss for different buffer sizes and for different packetization intervals are shown Fig. 1(b). We used 20 traces for both packetization intervals in these measurements. Packets are dropped at the playout buffer because some packets arrive with inter-arrival time less than the inter-arrival time at the sender (bars to the left of peak shown in Fig. 1(a)). If packetization delay is more, then the drop at the playout buffer is less compared to the case of smaller intervals for the same playout buffer size. The network induced delay distribution is more or less same for all the packets. If the packets are spaced at larger intervals, then the deviation caused by network does not affect them much compared to packets generated at smaller packetization intervals. For smaller packetization the delay jitter induced by the network is higher when compared with its mean inter-arrival times. This causes the packets to form bunches and therefore some of them are dropped at the playout buffer. This is an indirect way of inferring about the delay jitter, which is higher for packets transmitted at smaller intervals. Therefore

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2008 proceedings.

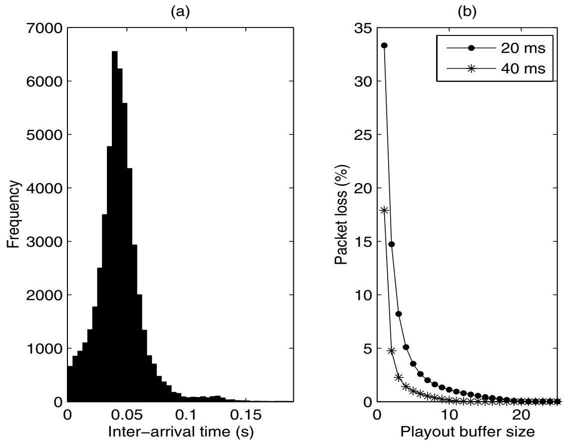

Fig. 1. VoIP session measurement results extracted from traces collected on the testbed: (a) histogram of inter-arrival delay, (b) percentage of packet loss at the playout buffer.

for packets with smaller packetization, tuning has to be more robust to accommodate more jitter. The adaptive playout buffer management is one way to circumvent this effect [14] but it requires some computing power at the receiver which is a premium if hand held devices are used.

Here, packet loss at a fixed size playout buffer is used to indirectly infer the delay jitter. One might wonder if the observed lower packet loss for larger packetization intervals is due to less number of packets arriving at the receiver. Earlier studies [9] have shown that large packets experience more drop in the network and hence less arrivals at the receiver. When packetization interval increases, the packet size also increases. Therefore the buffer would have been often starving in the case of larger packetization due to more loss experienced by packets of larger size in the network. It is thus crucial to find the end to end loss and the delay.

We recall the characteristics observed so far. The larger size packets undergo higher loss in the network; smaller size packets experience more jitter. To find the influence of the network exhaustively, we need to conduct the experiments with different packetization intervals under different background traffic load. We can not keep the network characteristics across the experiments same. This leads to the study with a simulation model of a bottleneck link approximately representing network and a playout buffer together. Percentage of packet loss with respect to various background traffic loads are found in this case.

IV. Modeling and Simulation

More than 90% of the Internet traffic is through TCP [15]. The TCP sliding window flow control always tries to maximize the bandwidth usage. Congestion occurs due to a bottleneck router in the path because of the higher offered load at that router. If the arrival rate of packets exceeds serving capacity of the router, an arriving packet will be queued in the input queue. If the queue is full then usually the newly arrived packet is dropped. Thus the TCP window reduces due to loss of packets

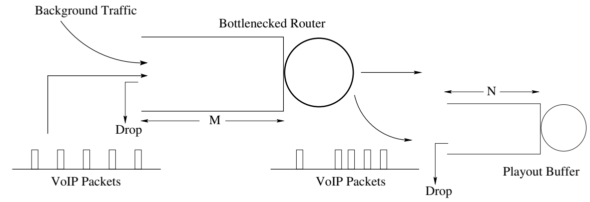

Fig. 2. VoIP session simulation model; M, N - queue size of bottleneck router and playout buffer, respectively.

and in turn reduces the load at the bottleneck router. Because of this the router gets a chance to serve the queued up packets. This sort of dynamic changes at the routers in the network influences the traffic characteristics of the UDP packets. To compare the net effect of packetization interval on packet loss including the network induced loss, we approximate the entire path of the media packets through a single bottlenecked router. The simulation model proposed is as shown in Fig. 2. This model gives the flexibility for varying the link occupancy. A simplistic model of the bottleneck router is considered with a fixed buffer size $M = 256$ kB to simulate network drop at this router due to buffer overflow. In simulation the background traffic (BT) trace is kept constant for 20 and 40 ms packetization intervals to enable a fair comparison. Playout buffer size N is one of the simulation parameters. Packet inter-arrivals for BT have been studied extensively and are approximated as exponentially distributed [11]. We took packet size distribution from the real traces; seven traces totalling up to 21 million packets were collected on different days and times. Number of packets versus packet size is shown in Fig. 3(a). Few packets exceeded 1500 Bytes, which we neglected. Cumulative distribution of packet size is shown in Fig. 3(b). It matches with National Laboratory for Applied Network Research and other studies [16]. This empirical CDF is used to generate packet sizes for BT to mimic the real Internet traffic.

A. Simulation Results

The link speed is constant and by varying the arrival rate with mean packet size derived from the above CDF, one can find total rate of BT. Various BT with respect to arrival rate are identified through the factor γ, that represents the total input bytes compared with the link speed in bytes. It is given by

$$\gamma = \lambda P L^{-1},$$

where λ denotes packet arrival rate (packets/s), P denotes mean packet size (Bytes) and L denotes link speed (Bytes/s). The system allows $\gamma \geq 1$ since the queue size is fixed. We have varied γ from 0.94 to 1.2 to catch all the effects that occur around $\gamma = 1$. For $\gamma < 1$ the network is not loaded. With $\gamma \geq 1$ the packet loss in the network is high and hence supporting VoIP applications may be difficult without QoS guarantees from the network. Therefore γ around 1 is of interest since the packet drop, and jitter severely affects the quality of the session.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2008 proceedings.

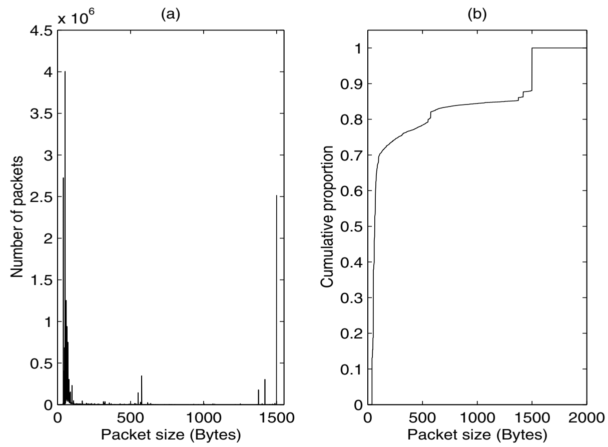

Fig. 3. Collected VoIP packet size statistics: (a) histogram, (b) CDF.

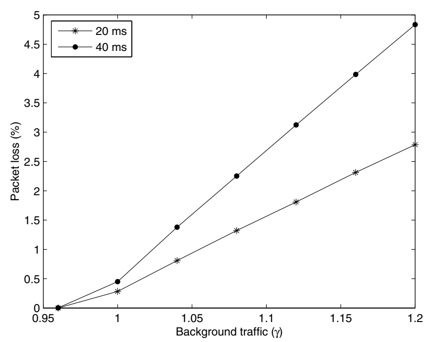

Fig. 4. Simulation of percentage of packet loss in the VoIP network testbed.

Average percentage of packet loss for 20 ms and 40 ms packetization intervals, with respective packet sizes, is found for 20 simulation runs, each run lasting for 10^7 packets. At higher γ, 20 ms packets experience less drop in the network compared to 40 ms packets, see Fig. 4. As we have already seen in Section III, the percentage of packet loss is higher for 20 ms packetization interval at the playout buffer. Therefore, it is of interest to find when the packet loss for 20 ms exceeds that of packets with 40 ms packetization intervals. With the help of this simulation model, the difference between total percentage of packet loss for 20 and 40 ms packetization intervals are tabulated for different queue sizes in Table I. In the table, 0 means 20 ms packetization interval is better (less overall drop, both at network and at playout buffer) than 40 ms. A positive value shows the amount by which packet loss for 20 ms packetization interval exceeds its 40 ms counterpart. For lower γ (less background traffic) and with lower buffer size, 40 ms packetization interval is better. For higher playout buffer sizes, 20 ms is always better since packet loss in the network becomes significantly less. This difference increases with the difference in packetization intervals.

It is observed that large packets, and in turn larger packetization interval, may be compressed more to avoid loss in the network since the packet size becomes lesser with

TABLE I

COMPARISON OF TOTAL PACKET LOSS FOR 20 AND 40 MS PACKETIZATION INTERVALS; N - PLAYOUT BUFFER SIZE

N	γ				
	0.96	1.0	1.04	1.08	1.12
2	3.1737	2.8265	0.2327	0	0
3	1.8905	1.4942	0.6160	0.0133	0
4	1.1615	0.8147	0	0	0
5	0.8014	0.4865	0	0	0
6	0.5726	0.2896	0	0	0
7	0.4177	0.1604	0	0	0
8	0.3095	0.0688	0	0	0
9	0.2302	0.0052	0	0	0
10	0.1701	0	0	0	0

compression. At the same time it causes less drop at playout buffer since packetization interval is more. It is worthwhile to note that compressing beyond a level will not help since the media RTP packets have at least 40 B of header, as discussed in Section I. Further, with higher compression the quality of the decoded speech is poor and even a lower percentage of packet loss may cause severe quality degradations. Moreover, as we have discussed earlier, with higher packetization intervals the total mouth to ear delay increases reducing the interactivity of the session. Therefore, there is no single technique to address all these aspects. Next we give a case study which takes into consideration all the parameters for finding the quality of a session. This can be used dynamically to fix an operating point for a session.

B. MOS versus Delay and Packet Loss: a Case Study

Usually, the total delay, though correlated with packet loss, remains with in a certain bound. In fact, one of the major causes for delay is due to the queuing at the playout buffer. This section gives a case study of MOS for the end to end session, which is dependent on delay and packet loss. We use G.711 codec here in this study for which impairment factors are standardized. The E-Model defined in [17] is an analytic model for voice quality assessment. The basic result of E-model is the calculation of R-factor [18] which is used to find MOS as

$$\text{MOS} = \begin{cases} 1, & \text{for } R < 0, \\ 4.5, & \text{for } R > 100, \\ \Sigma, & \text{else,} \end{cases}$$

where $\Sigma = 1 + 0.035R + 7 \cdot 10^{-6}(R^2 - 60R)(100 - R)$. For G.711 codec R-factor is computed as

$$R \approx 94.2 - I_{d_{me}} - I_{ef},$$

where impairment associated with mouth to ear delay is

$$I_{d_{me}} = 0.024 d_{me} + 0.11(d_{me} - 177.3)\ H(d_{me} - 177.3),$$

and impairment associated with packet loss is

$$I_{ef} = 11 + 40 \ln(1 + 10P_d),$$

where d_{me} is mouth to ear delay, P_d is the total packet loss and

$$H(x) = \begin{cases} 1 & \text{if } x > 0, \\ 0, & \text{else} \end{cases}$$

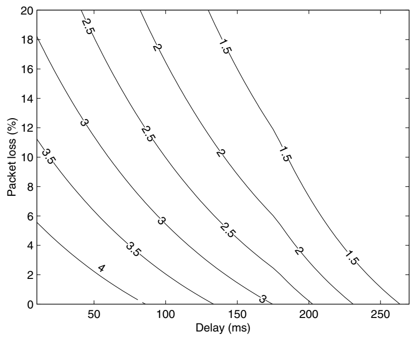

Fig. 5. Delay versus packet loss for different MOS.

The above expression uses the total delay and total packet loss unlike explicit division of delay in the network, playout buffer and due to codecs as in [18]. For different codecs Appendix I of [19] gives the values of I_{ef} to formulate R-factor for different packet loss. Fig. 5 represents the MOS values for different mouth to ear delays and packet losses. Lower packet loss and higher delay also result in lower MOS. For a particular MOS requirement, given the network delay and loss, the balancing act will be with respect to managing the packet loss and delay at the buffer. Packet loss in network can also be reduced using smaller size packets. In this sense the packetization interval becomes very important to reduce the loss in the network and delay at the playout buffer. As observed earlier, coding might reduce packet size and in turn packet loss in the network but I_{ef} also changes accordingly. Therefore concealment of packet loss and redundancy become very important aspects. These observations lead to many schemes to reduce packet drop percentage and the delay. One such method, called as Adaptive Packetization with Packet Interleaving, implemented on the VoIP testbed developed in-house is discussed in [20].

V. Conclusions

We presented a mix of measurements and simulation study to get the insights into the end to end packet loss and the delay of a VoIP session. Not only that the increase in the packet size results in higher loss is confirmed with this study, we also found that the effective delay jitter for higher packetization interval is lower. Thus there is always a chance that depending on the traffic on the network, lower packet sizes do not necessarily yield lower packet loss and hence better quality, since the jitter is higher. Further, the reverse is also true that higher packetization not necessarily mean that better quality since it increases the delay and confronts higher loss.

We explained all the intricate relation between the knobs that control the quality—the loss, delay and jitter, and packetization interval. We also deliberated on the effect of background traffic. We envisage that the quality can be assured by constantly forming trade-off between many of these varying parameters. We implemented a primitive version where the operating point for a given MOS is selected, see Fig. 5, based on the observed packet loss and tuning the playout buffer.

The effect of different packetization intervals and with different coding schemes (including FEC) on the packet loss and delay and, thereby, on MOS require further study. Though indirect, bringing out the effect of packetization interval affecting the delay jitter is a novel feature of this study. Moreover, we considered mouth to ear MOS calculation with all the parameters that affect the MOS.

References

[1] S. Andersen, A. Duric, H. Astrom, R. Hagen, W. Kleijn, and J. Linden, "Internet Low Bit Rate Codec (iLBC)," Internet Engineering Task Force, RFC 3951, 2001.

[2] R. Venkatesha Prasad, R. Muralishankar, Vijay S., H. N. Shankar, P. Pawełczak, and I. Niemegeers, "Voice Activity Detection for VoIP—An Information Theoretic Approach," in *proc. 49th IEEE Global Telecommunications Conference (IEEE GLOBECOM 2006)*, San Francisco, CA, USA, Nov. 2006.

[3] (2006) VQube Internet Telephony Application. [Online]. Available: http://www.vqube.com/

[4] ITU-T G.114, "One-way Transmission Time," International Telecommunication Union, Recommendation G.114, Feb. 1993.

[5] P. T. Brady, "Effects of transmission delay on conversational behaviour on echo-free telephone circuits," *Bell System Technical Journal*, vol. 50, no. 1, pp. 115–134, Jan. 1971.

[6] D. Mills, "NTP: Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI ," Internet Engineering Task Force, RFC 2030, Oct. 1996.

[7] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force, RFC 3550, July 2003.

[8] S. B. Moon, J. Kurose, P. Skelly, and D. Towsley, "Correlation of packet delay and loss in the internet," University of Massachusetts, Amherst, Department of Computer Science, Tech. Rep. 98-11, Jan. 1998.

[9] H. Sawashima, Y. Hori, H. Sunahara, and Y. Oie, "Characteristics of UDP Packet Loss: Effect of TCP Traffic," in *Proc. 7th International Conference of Internet Society (INET 97)*, Kuala Lumpur, Malaysia, June 1997.

[10] J. C. Bolot, H. Crepin, and A. V. Garcia, "Analysis and control of audio packet loss in the internet," in *proc. NOSSDAV 95*, Durham, New Hampshire, 1995.

[11] J. C. Bolot, "Characterizing end-to-end packet delay and loss in the internet," *Journal of High Speed Networks*, vol. 2, no. 3, pp. 289–298, Sept. 1993.

[12] J. Pointek, F. Shull, R. Tesoriero, and A. Agrawala, "NetDyn revisited: a replicated study of network dynamics," *Comput. Netw. and ISDN Syst.*, vol. 29, no. 7, pp. 831–840, 1997. [Online]. Available: citeseer.ist.psu.edu/pointek96netdyn.html

[13] G. Dn, V. Fodor, and G. Karlsson, "On the Effects of the Packet Size Distribution on the Packet Loss Process," *Telecommunication Systems Journal*, vol. 32, no. 1, pp. 31–53, May 2006.

[14] K. Fujimoto, S. Ata, and M. Murata, "Adaptive Playout Buffer Algorithm for Enhancing Perceived Quality of Streaming Applications," *Telecomm. Syst.*, vol. 25, no. 3-4, pp. 259–271, 2004.

[15] M.-S. Kim, Y. J. Won, and J. W. Hong, "Characteristic analysis of internet traffic from the perspective of flows," *Comput. Commun.*, vol. 29, pp. 1639–1652, June 2006.

[16] K. Thompson, G. Miller, and R. Wilder, "Wide area internet traffic: Patterns and characteristics," *IEEE Network*, vol. 11, no. 6, pp. 10–23, Nov. 1997.

[17] ITU-T G.107, "The E-model, a Computational Model for use in Transmission Planning," International Telecommunication Union, Recommendation G.107, Feb. 2003.

[18] R. Cole and J. Rosenbluth, "Voice over IP performance monitoring," *ACM Comput. Commun. Rev.*, vol. 31, no. 2, pp. 9–24, 2001.

[19] ITU-T G.113, "Transmission Impairments due to Speech Processing," International Telecommunication Union, ITU-T, Recommendation G.113, Feb. 2001.

[20] H. Dagale, "Audio conferencing tool for Intranet," Master's thesis, Indian Institute of Science, Bangalore, India, 2001.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE CCNC 2008 proceedings.