AN EMBEDDED PACKET TRAIN AND ADAPTIVE FEC SCHEME FOR VOIP OVER WIRED/WIRELESS IP NETWORKS

Chih-Wei Huang, Somsak Sukittanon, James A. Ritcey, Aik Chindapol[†], and Jenq-Neng Hwang Dept. of Electrical Engr., Box 352500, University of Washington, Seattle, WA 98195, USA

[†]Siemens Corporate Research, Princeton, NJ 08540, USA

ABSTRACT

Voice over IP (VoIP) has become the fastest growing wireless alternative to conventional telephony service by way of ongoing deployment of WLAN hotspots and even powerful WiMAX coverage. Resulting from the wired/wireless combined best-effort based heterogeneous IP networks which provide more fluctuation in available bandwidth and end-to-end delay, the performance of VoIP quality, especially using the handheld wireless devices, has been greatly degraded due to frequent packet loss and longer delays. This paper proposes a real-time embedded packet train probing scheme for estimating end-to-end available bandwidth so as to accomplish effective congestion control. By trading acceptable delays with adaptive packetization of voice bitstreams, as well as adaptive insertion of forward error correction (FEC) packets, an optimized system driven QoS approach for VoIP can thus be achieved.

1. INTRODUCTION

With the emerging of wireless technology, a variety of multimedia services, e.g. audio and/or video conferencing, are available today through portable devices and even more increasingly accessible in the near future. Voice over IP (VoIP) becomes an alternative to conventional telephony service in many locations by way of ongoing deployment of WLAN hotspots and even powerful WiMAX coverage. However, present Internet is neither providing adequate quality of service (QoS) for users nor ready to become the universal network satisfying all our communication needs [1].

The quality of VoIP applications is mainly affected by packet loss and delay. Generally, it can tolerate packet loss to some extent but requires the delay to be in a relatively rigorous range. It is considered one way delay (OWD) of duration smaller than 150 ms as good quality for voice conversation applications, while more than 400 ms is intolerable. For jitter, variance of the packet inter-arrival time has to be smaller than 50 ms. Conclusively, ~1% of packet loss is requirement for quality VoIP. In today's heterogeneous Internet, the problem of VoIP quality can be even worse. A variety of wireless access technologies such as wireless local area network, cellular network, and bluetooth, may co-exist with wired backbone introducing different link layer control mechanisms and high bit error rates. The consequence of the wired/wireless combined best-effort based IP networks can provide more fluctuation in available bandwidth and end-to-end delay, causing more jitters.

To overcome these deficiencies, there are mainly two approaches to achieving end-to-end QoS support: network centric and end-system centric [2]. Network centric approach provides prioritized services to different traffic classes in networking equipments between senders and receivers. Representative proposals are DiffServe networks on top of conventional IP as well as 802.11e and WiMAX for wireless environments. Endsystem centric method instead depends solely on senders and receivers. It performs congestion and error control to maintain most media quality. To meet the quality of VoIP service requirements and application properties (unicast, peer-to-peer), a new end-system driven solution is proposed in this paper. Through effective integration of end-to-end available bandwidth estimation, voice bitstream packetization, congestion control, and packet level forward error correction (FEC), our proposed method is specially designed for wired/wireless VoIP applications. The most important advantage is the ability to share bandwidth with other traffics and sessions while minimizing packet loss and delay to maintain consistent quality.

The paper is organized as follows: Sections 2 and 3 address related works in congestion and error control. We then introduce our proposed architecture in Section 4. Section 5 discusses the simulations and results, followed by the conclusion in Section 6.

2. CONGESTION CONTROL AND INTERNET TRAFFIC MODEL

For end-to-end QoS, congestion control is one of the important tasks happening at end-systems. Basically, end-systems adjust data rates according to observed network conditions, i.e., packet loss and delay statistics. However, the long delay created by retransmission used in TCP is not practical in multimedia applications including VoIP. Unlike TCP, many UDP control schemes try not to grab more channel shares than a TCP session under the same environment. TFRC [4] is a widely known equation-based protocol which manages to get smooth and accurate round trip time (RTT), though its RTT-biased result is arguably representing the actual data rate.

Available bandwidth estimation tools, e.g., BIC [5], also play an important role for layered congestion control. Receivers perform bandwidth detection before subscribing to a proper layer according to its estimated available bandwidth. This method requires modifications when it is applied to single layer voice codec.

More recently, a traffic model is proposed in [3] under firstin-first-out and fluid traffic assumptions. In this method, if a stream, with transmitted data rate lower than or equal to the endto-end available bandwidth, is sent, the received data, at the same rate, can be received at the receiver end. That implied if we compare the interarrival time of received packet and the time gap at the sender, these two values should be identical. Although the fluctuating Internet traffic makes observed events not ideal, we have designed an approach to tolerate the traffic diversity and show that the new model is suitable for VoIP application.

3. ERROR CONTROL AND WIRELESS CHANNEL

Due to the bursty nature of multichannel wireless fading, packet can be lost at a much higher rate than in wired channel. To provide error control, the packet level FEC Reed-Solomon code is applied in this paper. We dynamically choose (n,k) parameters based on desired protection level and affordable redundancy. Thus, as long as k packets in an n packet block are successfully received, the voice stream can be decoded without error.

Gilbert/Elliot's two-state Markov model is used to simulate the bit error rate resulting in the packet loss. This method can be approximated as a two-state Markov chain with transition matrix

 $\begin{bmatrix} p & 1-q \\ 1-p & q \end{bmatrix}$, where p and q are the probabilities the i^{th}

packet that is good or bad given the $(i-1)^{\text{th}}$ packet is also good or bad. The FEC decoding error rate, defined as the error rate of protected data, for a specific pair of p and q can be calculated using this model [7]. Conclusively, the right amount of FEC packets can thus be added assuming p and q estimated at receivers.

Along with error control, some actions need to be taken to maintain the performance of congestion control protocol when it comes to wireless. Congestion control can only deal with congestion loss while error control is for wireless loss. Coupling packet loss classification (PLC) to the control loop is one of the promising ways [6] to improve the efficiency and is used in our paper to discriminate two loss sources.

4. PROPOSED ARCHITECTURE

4.1. System overview

A VoIP system with proposed end-system QoS support is illustrated in Fig. 1. It is only shown in one-way direction to make our discussion clearer. The system is built on top of UDP/RTP protocol for real-time support and RTCP for periodic feedback.

The process starts from voice encoding at the sender side. Encoded frames then pass through the packet formation and scheduling controller in charge of transmission jobs in application layer. This controller determines 1) how many frames in a packet, 2) how much packet loss protection to provide, and 3) the time to send out the next packet depending on RTCP feedback from the receiver. There are several framing choices and two modes, normal and probing, for scheduling. More details of decision strategies will be discussed in Section 4.3.

At the receiver side, received packets first go through the PLC unit to categorize packet losses from congestion and wireless separately. The receiver is responsible for making a conclusion toward congestion status and sending the message back to the sender based on embedded probing mechanisms. The feedback includes a (n,k) combination from adaptive FEC unit.

4.2. Embedded probing

In most literature, specially designed packet sequences are transmitted to probe the Internet traffic for bandwidth estimation by analyzing packets at the receiver side. In a layered multicast system [5], the server can continuously send out all layers of data for receivers to subscribe appropriate amount of contents with affordable bandwidth. The probing needs, usually a stream in certain data rate, are fulfilled by enabling receivers to temporarily subscribe to higher layers. If the resulting outcome from analyzing relative OWD is positive, it means a receiver is open to join a higher layer (group) with better quality or protection.



Figure 1. The system diagram (one-way).

In wireless VoIP, however, the network resource is under a tighter budget. We prefer not to transmit any extra data while performing probing by packet scheduling. Fig. 2 shows examples of different packet scheduling and packetizing. Assuming that the receiver observes wireless loss percentage above some preset quality threshold, it has to decide whether adding extra protection (i.e., the color filled FEC packets as shown in Figure 2(c)), which increases the data rate from r1 to r2, will cause congestion or not. At this moment, source packets transmission can be rescheduled as Fig. 2(b) from regular scheduling as Fig. 2(a). Note that, during this short packet train rescheduling, the rate is temporarily set as r^2 , while maintaining the overall transmission rate to be r^1 . More rate options are available resting on packetizing and unavoidable internet overhead with 40 bytes/packet under IP/UPD/RTP structure. For example, packetizing G.729 frames in 3 frames/packet yield 18.6 Kbps. To get even lower transmission rate, 4 frames/packet can further reduce the rate to 16 Kbps as shown in Figure 2(d). Therefore we are flexible in trading longer delay (more frames per packet) with lower packet loss (adding more FEC packets). More information about G.729 will be discussed in Section 5.1. Note that any packet formation is required to meet overall delay constraint, < 400ms. For instance, by introducing at most extra 100 ms in frame delay to the first frame in the first packet of a train, it can accommodate 9 packets/train in a r1 = 18.6 Kbps, r2 = 31 Kbps, (n,k) = (3,5)probing under 3 frames/packet.



The next challenge is to determine whether there is enough bandwidth for target probing rate (r2). According to the traffic model, we define *equal* event for sum of gaps (Fig. 2(b)) as below:

$$Eq = I\left(\frac{\left|G_{sum,sender} - G_{sum,receiver}\right|}{\max(G_{sum,sender}, G_{sum,receiver})} < \delta\right)$$
(1)

where *I* is an indicator with value 1 if the equation inside the bracket is true, or 0 on vice versa. *G* is the sum of gaps within a train of sender or receiver. δ is a threshold suggested by [3]. We take two *Eq* s for each packet train in its *total length* and *total length-1* respectively. The sending gap and receiving gap are regarded to be equal, if any of the two *Eq* s is 1.

To avoid bad conclusion drawn by these relatively short packet trains, a supplement index S_{eq} is also continuously calculated to facilitate the decision. The index at the *n*th time is

$$S_{eq,n} = (1 - \alpha) \times S_{eq,n-1} + \alpha \times (Eq_n + Eq_{n-1})$$
(2)

To make the decision whether the probing bandwidth is lower than the available bandwidth, we not only need to make sure that the sending gap and receiving gap are more or less equal, but to ensure two additional conditions have to be satisfied. First, S_{ea} rises to a high enough value quickly within period 1, and, second, it stays above a reasonably high value for more than period 2. More specifically, S_{eq} stays at high values implies there exists a number of consecutive Eq = 1 and the channel is less likely to be congested. Obviously, two thresholds, Th_{Sea} , Th_{Seq_high} and two time intervals need to be specified. We set period 1 = period 2 as the time interval of receiving 150 packets, since generally 50 packets/train is required to get a good cross sample [5] Three times of this value compensates our short packet trains well in accuracy (see 5.2). Taking 9 packets/train for example, it has at most 16 Eq and S_{eq} samples per period. A positive decision can be resulted from reaching Th_{Seq_high} within 150 packets and not to drop below Th_{Seq} for another 150 packets.

4.3. Quality optimization

To monitor the quality, the receiver always keeps four observation values: congestion loss rate, wireless loss rate, and the estimated p, q transition rates. For the delay, the overall effect when choosing (n,k) can be approximate as below:

$$Total_Delay \approx t_f \times F \times k + Internet_OWD$$
(3)

where the fixed t_f represents per frame interval of voice codec, F is the number of frames per packet, and k is one of FEC parameters. Internet OWD is around 150 ms in real world ISP environments [9]. That shows if we want to improve transmission efficiency (higher F) or achieve particular error protection level with lower redundancy (higher k), we also increase the overall delay. In our algorithm, we check (3) every time performing a rate change. A total delay constraint of 250 ms is commonly set but a hard limit of 400 ms is adopted to assure the delay quality even under heavily congested condition based on packing more frames per packet. The algorithm is demonstrated in Fig. 3.

For the case with both loss rates lower than thresholds, no action should be taken. When there is no congestion at the current rate, we try to probe first then add FEC. When the channel is congested, the system will reduce its rate to relieve the congestion by removing FEC or packetizing more frames in a packet (i.e., relaxing the delay constraint) at the same time. To support the same quality at different packetizing conditions, we increase the number of frame/packet one-by-one and search for every (n,k) for suitable FEC. Also, under longer delay tolerance, system will keep looking for a chance to provide services with less delay. An explicit example can be referenced in 5.3. It also needs to be noticed that the congestion loss threshold is set slightly higher than wireless loss because FEC should be more aggressively added since it is the main mechanism to recover lost packets.

| start |
|--|
| If total delay constrain = 250 ms |
| If wireless loss $> 1\%$ and congestion loss $< 5\%$ |
| Find (n, k) with least redundancy |
| Start probing at higher rate |
| If probing result is positive |
| Add $FEC(n, k)$ |
| Else |
| Go to start |
| Elseif congestion_loss > 5% |
| total_delay_constraint = 400 ms |
| If FEC added |
| Remove FEC |
| Else |
| Pack one more frame per packet |
| Go to start |
| Else |
| Find FEC at desired protection level at similar rate |
| Apply new FEC |
| total_delay_constraint = 250 ms |
| Go to start |
| |

Figure 3. Psudo-code for quality optimization algorithm.

5. SIMULATIONS

5.1. Configuration setup

The system was implemented on ns2 network simulator. Simulation topology, as shown in Fig. 4, contained congestionprone wired (bold lines) and wireless (dotted lines) part as the last hop to users. VoIP senders (S1, S2 and S3) were connected to the bottleneck link N1-N2 by wire and route packet to corresponding receivers (R1, R2 and R3) with pre-specified wireless BER. Competing traffic existed from N0 to N3.



Figure 4. Network topology with specified capacity and delay where "S" and "R" represent VoIP senders and receivers with session number. N1-N2 was the bottleneck link.

To setup cross traffic, we assume that the Internet traffic becomes smoother over a time scale of one-tenth to several seconds [8]. In this paper, cross traffic is configured in constant bitrate with randomizing dither to achieve desired average available bandwidth in a time interval of tens of seconds.

In respect of voice codec, we simulated G.729 which results in 10 ms/frame, 100 frames/s, and constant rate at 8 Kbps. Its loss concealment function can maintain good listening quality under ~1% frame loss.

5.2. Probing accuracy

We first evaluated the performance of embedded probing scheme for available bandwidth. There was no wireless loss introduced.



Figure 5. Probing results of (a) potentially congested channel (b) non-congested channel.

There were in total 100 simulations conducted with varying parameter values used in the cross traffic streams. We empirically found that $\alpha = 0.4$, $Th_{Seq} = 0.9$, and $Th_{Seq_high} = 1.8$ created the best outcome. In all the 100 simulations conducted, 50 at ~20 Kbps and 50 at ~ 60 Kbps of available bandwidth, this parameter set achieve 89% accuracy. The α value contributed a tradeoff between convergence speed and smoothness for moving average of S_{eq} . A probing accuracy performance of node S3 was illustrated in Fig. 5. The true available bandwidth measured at the input queue of bottleneck link is shown in fluctuating lines, and the target probing bandwidth shown in horizontal lines. The decisions were made correctly by our algorithm, as indicated in * or \Box for congested or non-congested. From Fig. 5(a), the available bandwidth was below the target rate for more than 70% of time. Intuitively, they should be judged as potentially congested. From Fig. 5(b), it was similar to the previous plot expect the available bandwidth was higher. This case, available bandwidth was enough in average sense.

5.3. Overall performance

We further constructed several 100-second scenarios of experiments with consistent wireless loss parameters. The sender status were the same as above, one in low rate, one high rate, and one probing with low rate set on 18.6 Kbps (3 frames/second). From 0 to 30 seconds, there was plenty of available bandwidth averaged ~60 Kbps. Starting from the 30th second, it suddenly dropped to ~10 Kbps than bounced back to ~50 Kbps at the 60th second. Wireless links were modeled as p = 0.9 and q = 0.1 respectively resulting average packet loss rate at 10%.

An overall experiment result is illustrated in Fig. 6, which includes fluctuating available bandwidth, receiving rate with three major rate changes, decoding error spots due to uncovered packet loss, and three probing outcomes. Starting with 18.6 Kbps (3 frames/packet) sending rate, no congestion loss was found while high wireless loss was detected, this led to (n,k) = (5,3) FEC protection with new rate 31 Kbps to achieve lower than 1% actual packet loss. Then 9 packets/train, generating 100 ms delay, was used to perform probing. The system started to apply FEC around the 8th second and the decoding errors are significantly reduced. The next scenario, more than 5% of congestion loss was noticed at the 45th second, so the sender removed FEC to decrease the rate back to 18.6 Kbps. However, that was not enough due to the existing high wireless error, which calls for the (5,3) FEC

protection with 5 frames/packet repacketization to maintain the same rate. This releases the delay limitation to 400 ms. Afterward, the system kept probing at 31 Kbps for a chance to be back with 250 ms delay and (5,3) FEC at the 67th second.



Figure 6. Overall performance: (a) available bandwidth, receiving rate and probing decisions; (b) decoding error spots for FEC only and proposed scheme.

During the whole process, decoding error rate was low at 0.73% except the startup time before the first positive probing comparing to 1.67% for sending 31 Kbps FEC stream all the way where high congestion loss at 4.54% was suffered between the 40^{th} and 60^{th} second. The low packet loss using proposed scheme was achieved through the sacrifice of overall delay to the limit under extreme network conditions.

6. CONCLUSIONS

This paper proposes an innovative strategy to meet the quality of VoIP service requirements under noisy and bandwidth deficient wired/wireless environment. Through effective integration of embedded packet train bandwidth probing, voice bitstream packetization, congestion control, and packet level FEC, our method can greatly reduce the wireless packet loss with the VoIP service requirement strictly satisfied.

7. REFERENCES

[1] F.A. Tobagi, "Voice over IP: the challenges behind the vision," Signals, Systems and Computers, Asilomar Conference on, Nov. 2004.

[2] Q. Zhang, W. Zhu, Y. Zhang, "End-to-end QoS for video delivery over wireless Internet," *Proceedings of the IEEE*, Jan. 2005.

[3] N. Hu, P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques," *JSAC, IEEE*, Aug. 2003.

[4] S. Floyd, M. Handley, J. Padhye, J. Widrner, "Equation-based congestion control for unicast applications," *SIGCOMM*, 2000.

[5] Q. Liu, J. Yoo, B.-T. Jang, K.-H. Choi, and J.-N. Hwang, "A scalable VideoGIS system for GPS-guided vehicles," *Signal Processing: Image Communication*, March 2005.

[6] H.-F. Hsiao; A. Chindapol, J.A. Ritcey, Y.-C. Chen, J.-N. Hwang, "A new multimedia packet loss classification algorithm for congestion control over wired/wireless channels," *ICASSP*, 2005.

[7] Q. Zhang, G. Wang, Z. Xiong, J. Zhou, W. Zhu, "Error robust scalable audio streaming over wireless IP networks," *IEEE Transactions on Multimedia*, Dec. 2004.

[8] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A nonstationary Poisson view of Internet traffic," *INFOCOM, IEEE*, 2004.
[9] January 2004 Report of the ICFA-SCIC Monitoring Working Group