Could Skype Be More Satisfying?
A QoE-Centric Study of the FEC Mechanism in an Internet-Scale VoIP System

Te-Yuan Huang, Stanford University
Polly Huang, National Taiwan University
Kuan-Ta Chen, Academia Sinica
Po-Jung Wang, National Taiwan University

Abstract

The phenomenal growth of Skype in recent years has surpassed all expectations. Much of the application's success is attributed to its FEC mechanism, which adds redundancy to voice streams to sustain audio quality under network impairments. Adopting the quality of experience approach (i.e., measuring the mean opinion scores), we examine how much redundancy Skype adds to its voice streams and systematically explore the optimal level of redundancy for different network and codec settings. This study reveals that Skype's FEC mechanism, not so surprisingly, falls in the ballpark, but there is surprisingly a significant margin for improvement to ensure consistent user satisfaction.

There is no doubt that Skype is the most popular VoIP service. At the end of 2009, there were 500 million users registered with Skype, and the number of concurrent online users regularly exceeds 20 million. According to TeleGeography, in 2008 8 percent of international long-distance calls were made via Skype, making Skype the largest international voice carrier in the world. Skype's success is attributed largely to the quality of the voice calls. Users everywhere seem to be *happy* with the service, no matter:

• How great the distance between the two parties in the conversation
• How limited the bandwidth between the parties
• How lossy the path the voice data traverses

Skype must be doing something right while sending voice bits over the Internet.

This study is motivated by a simple question: How does Skype adapt its voice data to heterogeneous connection quality and still keep users happy? Let us flash back to our experience designing multimedia systems over the years. The general tricks are twofold:

• In the presence of persistent data loss (i.e., the bandwidth along the path the data traverses is limited), a multimedia system adapts the sampling and compression rate at the codec level to reduce the amount of bits for delivery, while ensuring that the content is still perceptible.
• In the presence of sporadic data loss (i.e., links along the path the data traverses are lossy, or the background traffic contains short-term bursts that occupy a router's buffer space temporarily), a multimedia system adapts the amount

of redundant data, as well as the scheme for encoding such data, at the error concealment level.

Hereafter, we refer to the redundancy-based error concealment function of the system as the forward error correction (FEC) mechanism. There are two components in a general FEC mechanism: a *redundancy control algorithm* and a *redundancy coding scheme*. The control algorithm decides how much redundant FEC data should be added to the voice stream, and the coding scheme determines how redundant FEC data should be multiplexed and embedded.

For its PC-to-PC voice calls, Skype employs a voice codec that generates variable bit rate (VBR) voice streams depending on the available network bandwidth [1]. However, this is not the case for the PC-to-public switched telephone network (PSTN) service, which actually creates revenue for Skype. The PC-to-PSTN service, also called *SkypeOut*, enables users to make calls to traditional PSTN telephones, including fixed landline and mobile telephones. In some countries Skype also allows users to have a local phone number to receive calls from traditional telephone users; the service is known as *SkypeIn*. In 2008 Skype launched a monthly subscription program, which attracts more users to switch to Internet telephony completely. The program boosted Skype's quarterly revenue to US$170 million in the second quarter of 2009. For much of its commercial success (i.e., SkypeIn and SkypeOut calls), Skype falls back to the International Telecommunication Union (ITU)-defined fixed rate codec (constant bit rate [CBR]), G.729. The immediate implication is that Skype's FEC mechanism is in fact the major design flaw that is holding up the revenue. On March 31, 2009, Skype released an official version on iPhone and iPod Touch. Within 10 days, there were two million downloads from the iTunes App Store. The codec used in the handheld version is also G.729. As Skype expands

---

services for the wireless and mobile market, the role of the FEC mechanism becomes increasingly important.

Focusing on Skype's FEC mechanism, we investigate the relationship between the redundancy ratio and the network loss rate. Our major findings are:

- Skype increases the redundancy ratio as the network loss rate increases.
- Skype's control algorithm does not take the individual codec and packet loss patterns (burstiness) into consideration.

These findings indicate that, although Skype's FEC mechanism addresses the need to tune up the redundancy ratio when the network loss increases, there are still discrepancies in consistent user satisfaction. To address the problem, we adopt implementations of public domain codecs and systematically investigate how user satisfaction (i.e., mean opinion score [MOS]), with voice calls is influenced by the levels of packet loss and burstiness, as well as the type of redundancy coding scheme. The results suggest that the FEC mechanism should be codec-, loss-pattern-, and redundancy-coding-scheme-specific. Given the desired MOS level, codec, and redundancy coding scheme in use, our work leads to an FEC mechanism that ensures consistent quality of experience (QoE) based on the measured loss rate and loss burstiness.

Skype Overview

Launched in August 2003 and subsequently acquired by eBay in 2005, Skype was the brainchild of Estonia-based Ahti Heinla, Priit Kasesalu, and Jaan Tallinn, who are also the founders of KaZaA, a well-known file sharing software. Similar to general VoIP services, the system consists of a control plane and a data plane. The directory service and signaling protocol are implemented in the control plane, and the voice data transmission is implemented in the data plane. In the following subsections we discuss Skype's control and data planes, as well as recent studies of Skype's mechanism in the data plane.

Skype's Control Plane

Like its file sharing predecessor KaZaA, Skype's network architecture is a multitiered peer-to-peer system [2]. There are two types of Skype nodes in the overlay network: ordinary nodes and super nodes. Super nodes maintain an overlay network among themselves, while ordinary nodes pick one or some super nodes with which to associate. All Skype users are by default ordinary nodes, but it has been shown that a publicly reachable ordinary node with sufficient resources has higher probability of being selected as a super node [3]. The main responsibilities of super nodes are call relaying and directory services. They serve as proxies between firewalled Skype clients. This allows Skype nodes behind most firewalls and gateways to establish peer-to-peer calls without special configurations.

Super nodes also help perform directory services. According to [4], Skype implements a global index algorithm on its overlay network, so that every node in the network has full knowledge of all available users and resources through its associated super node. The only centralized component in Skype is the login server, which authenticates users with public key mechanisms. After authentication, all further signaling is performed in the peer-to-peer network.

Skype's Data Plane

The major reason for Skype's popularity is its voice quality, which is the result of Skype's codec selection and application-level FEC mechanism. A list of the codecs used by Skype and their characteristics is provided in [5]. Like its proprietary protocol, how Skype selects a particular codec for a voice session is not publicly known. Even so, through observations, we have found that G.729 is always used in SkypeOut as well as in voice sessions involving Skype clients on handheld devices, such as iPhones. For voice calls between PCs, different versions of Skype use different codecs. The default audio codec for version 3.1[1] is iSAC, a product of Global IP Solutions; however, since version 3.2, Skype has adopted SVOPC [6], an in-house developed codec, as the default codec. Note that both iSAC and SVOPC are variable bit rate (VBR) codecs. In early 2009 Skype announced another in-house developed codec called SILK for its latest version (version 4.0). Skype claims that SILK achieves the same quality as SVOPC, while requiring only half of the bit rate. Despite this claim, a recent poll of Skype users showed that 46 percent of the respondents would rather revert to an earlier version.[2]

Recent Studies

Bonfiglio et al. [1] analyzed how Skype adapts its traffic to different network conditions. They found that when the available bandwidth decreases, the bit rate and payload size of Skype traffic are also decreased. On the other hand, when packet loss is detected, Skype mitigates its impact by sending voice packets with redundancy. The authors also proposed a source traffic model of Skype. Based on the model, Skype's traffic is decided by three parameters:

- The bit rate used by the codec
- ΔT, the framing time of human speech
- The redundancy factor, which is the number of previous blocks Skype retransmits with the current encoded block

Following Bonfiglio et al. in [7], we conducted experiments with the iSAC codec and found that the first two parameters, the encoding bit rate and speech framing time, are controlled by the codec, while the amount of redundancy is controlled by the Skype program.

In this work we provide a more complete understanding of Skype's FEC mechanism and present a methodology for deriving the optimal FEC mechanism for general VoIP system design toward *consistent QoE*.

Skype's FEC Mechanism

To observe how Skype copes with lossy networks, we collected Skype VoIP traces in a controlled network environment. We set up a FreeBSD box as a layer 2 bridge to connect a Skype caller and a callee, and controlled the VoIP traffic passing through with the *dummynet* [8] kernel module. Although Skype can transmit its voice packets through UDP or TCP, we focus on UDP flows in this article and leave the study of the effect of TCP/UDP protocols for future work.

Observations on Skype Traces

We collected Skype VoIP traces for both PC-to-PSTN (G.729) and PC-to-PC (iSAC and SVOPC) voice calls. In each experiment we increased the network loss rate from 0 to 10 percent in 1 percent increments every 180 s.

G.729 — Figure 1a shows the scatter plot of the payload sizes of the packets in a SkypeOut (G.729) session. Since the bit rate and framing time of the G.729 codec are constants, the size of all G.729 voice frames should be the same. However, from the figure we can observe that the payload size
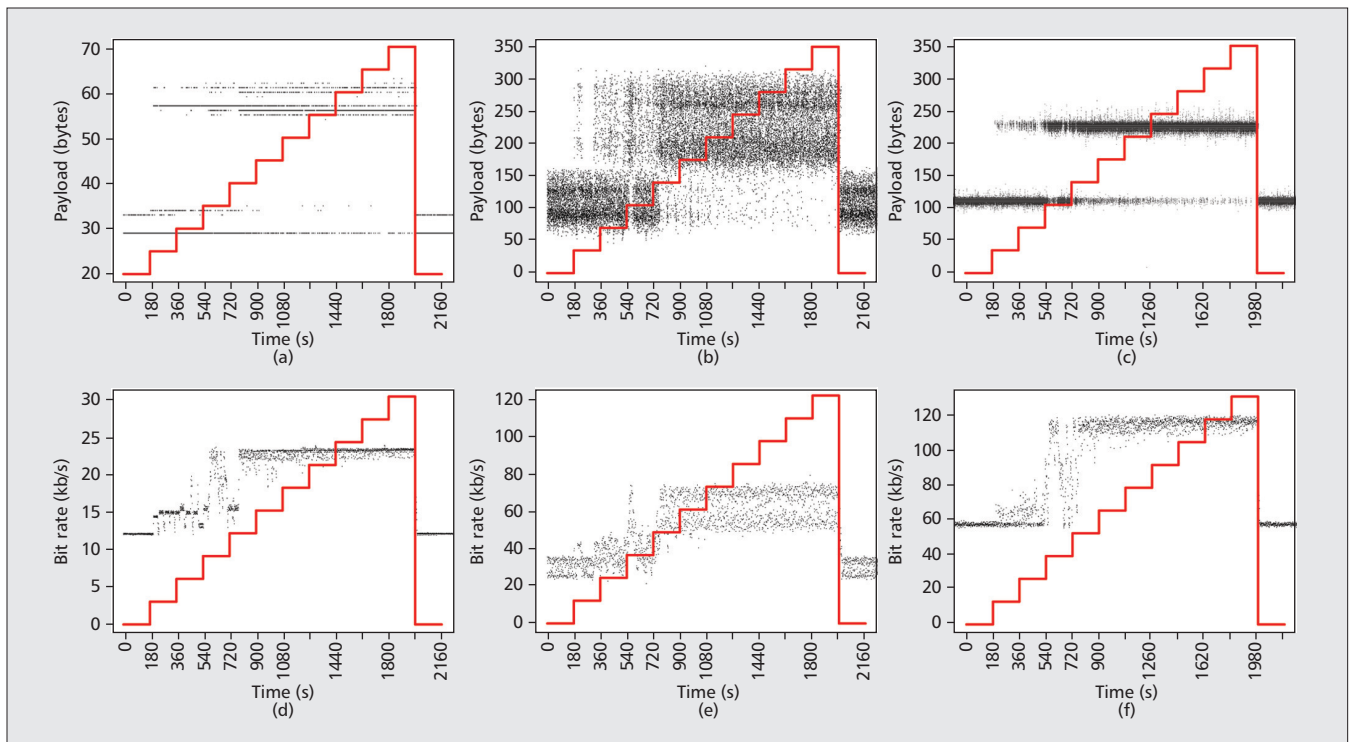

Figure 1. *The impact of the network loss rate on the payload size and bit rate of Skype packets for three different codecs: a) payload, G.729; b) payload, iSAC; c) payload, SVOPC; d) bit rate, G.729; e) bit rate, iSAC; f) bit rate, SVOPC.*

changes as the network loss rate increases. When the loss rate is 0 percent (i.e., between 0 and 180 s), the payload size remains around 30 bytes; however, as the loss rate increases, we find there are more packets with a payload of about 60 bytes. Furthermore, when the loss rate reaches 10 percent (i.e., between 1800 and 1980 s), the majority of packets have a payload of approximately 60 bytes. Interestingly, when the loss rate returns to 0 percent after 1980 s, the payload size drops back to around 30 bytes. Meanwhile, the bit rate of this voice session almost doubles when the payload of the majority of the packets is 60 bytes, as shown in Fig. 1d. Thus, the increment in the payload size is not the result of Skype adjusting the interpacket gap and including a higher number of voice frames in each packet. Instead, it indicates that Skype changes the proportion of packets with redundant FEC data based on the network loss rate.

iSAC — The iSAC trace exhibits similar behavior. However, the iSAC codec's bit rate is variable, so the size of its voice frames fluctuates within a range. When the loss rate is 0 percent, the payload size is within the range of (0, 160) bytes, as shown in Fig. 1b. As the loss rate increases, we find more packets with a payload in the range of (160, 320) bytes; and when the loss rate reaches 10 percent, the majority of the packets have a double-size payload. Moreover, the bit rate almost doubles when the traffic encounters a significant amount of network loss, as shown in Fig. 1e.

SVOPC — The effect of the network loss rate on the payload size and bit rate of the SVOPC trace are shown in Fig. 1c and 1f, respectively. Again, we find that the trace pattern is similar to those of G.729 and iSAC. Like iSAC, SVOPC has a variable bit rate; thus, the size of its voice frame also fluctuates within a range. When there is no network loss, the payload size is within the range of (90, 140) bytes; however, as the network loss rate increases, the number of packets with a double-size payload also increases along with the bit rate.

In summary, to mitigate the impact of packet loss on PC-to-PSTN (G.729) and PC-to-PC (iSAC and SVOPC) calls, Skype seems to piggyback redundant FEC data to a number of packets based on the observed network loss rate.

Understanding Skype's FEC Mechanism

To understand Skype's FEC mechanism, we attempt to quantify the amount of redundant FEC data added to Skype voice traffic. We define the *redundancy ratio* as the percentage of packets that carry redundant voice data (i.e., packets that have a double-size payload). If all packets carry redundant data, the redundancy ratio would equal 1. Conversely, if none of the packets carry redundant data, the redundancy ratio would be 0.

Deriving Redundancy Ratio — In the following we present our method for inferring the redundancy ratio used by Skype based on the traces collected in the above experiments. We take the three codecs as examples, although the method can be generalized to other codecs supported by Skype.

It is easier to deal with the SkypeOut (G.729) traces since the size of G.729 voice frames is constant. As shown in Fig. 1a, there is a clean cut, in payload size, between packets with and without redundant FEC data. To do so, we use a simple threshold method with the threshold set at 40 bytes to derive the redundancy ratio. Thus, if 30 percent of packets have a payload larger than 40 bytes, the redundancy ratio will be 0.3.

Dealing with iSAC and SVOPC traces is more difficult because they have variable bit rates and variable framing times. First, we need to determine the framing time of a packet based on the observed interpacket gap. Then the payload size must be normalized to the determined framing time before we can apply the same threshold method. The steps are described in detail in [7]. In this study the thresholds for iSAC and SVOPC traces are set at 160 bytes and 150 bytes, respectively. We repeated the above experiment five times for each codec. Figure 2a shows the average redundancy ratios and their 95 percent confidence intervals for the three compared codecs under various network loss settings.
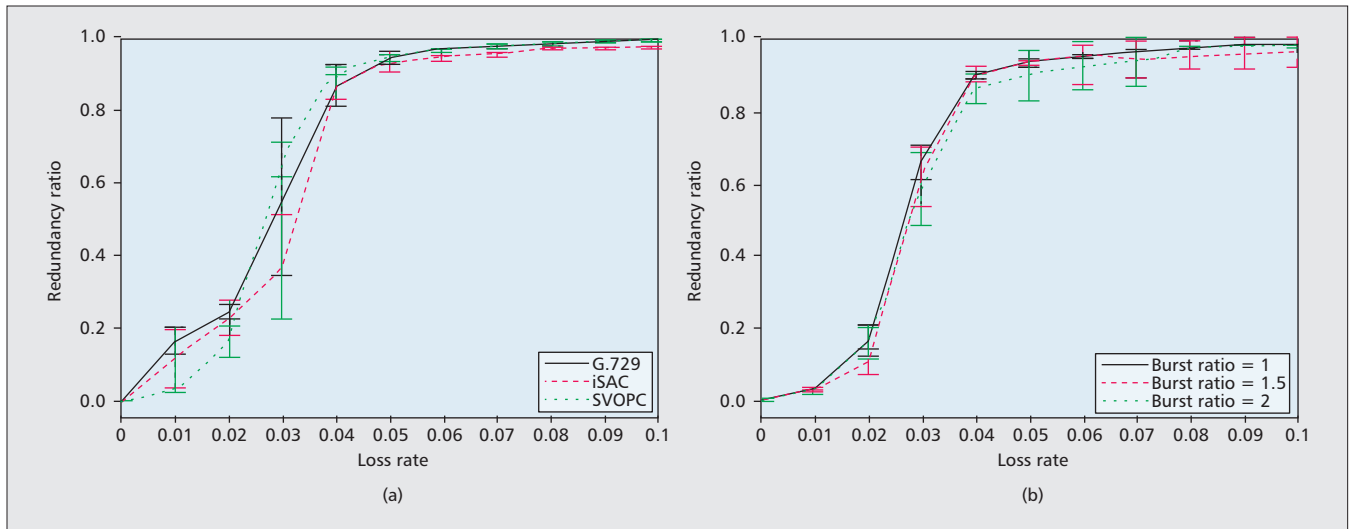
Figure 2. *Comparison of Skype's FEC mechanism for the G.729, iSAC, and SVOPC codecs, and under different levels of network loss burstiness for SVOPC: a) redundancy ratios under random loss; b) redundancy ratios for SVOPC under bursty loss.*

Skype's FEC Mechanism for Different Codecs — The results in Fig. 2a demonstrate that for all three codecs, the redundancy ratio increases gradually when the loss rate is between 1 and 2 percent, and it increases dramatically between 2 and 4 percent. The redundancy ratio remains above 0.9 when the loss rate is higher than 4 percent. In other words, more than 90 percent of packets carry redundant FEC data when there is a large amount of packet loss. Furthermore, in the figure the 95 percent confidence intervals of the three curves largely overlap. This suggests that Skype applies the same FEC mechanism for different codecs, even though it leads to different levels of user perception, as we show in the next section.

Skype's FEC Mechanism under Bursty Loss — As the network loss over the Internet is often bursty, it would be interesting to learn how Skype's FEC mechanism works under different levels of network loss burstiness. To quantify the burstiness of network loss, we adopt the *burst ratio* metric, defined in ITU-T G.107 [9] as the ratio of the average length of consecutive losses under bursty loss to that under random loss. By definition, the burst ratio is equal to 1 when packet loss is purely random, and it is greater than 1 when packet loss is bursty. Specifically, a burst ratio equal to 2 indicates that the average length of consecutive losses is twice as long as that of purely random losses. Again, we conduct the experiments by increasing the network loss rate from 0 to 10 percent in 1 percent increments every 180 s. However, this time, the packet loss is bursty rather than uniformly distributed. We implemented the Gilbert model [10] to emulate the burst ratio in dummynet. Details of the implementation can be found in [7].

Figure 2b shows the observed redundancy ratio of SVOPC when its traffic experiences packet loss with different burst ratios. In the figure each curve corresponds to a burst ratio setting, and the curves' 95 percent confidence intervals overlap each other. We have the same observation on the redundancy ratios of iSAC and G.729 under different levels of burstiness.

In summary, our experiment results suggest that Skype only adjusts redundancy ratios based on the network loss rate. In other words, Skype does not consider the difference of each codec or network loss burstiness in its FEC mechanism.

QoE Analysis of FEC Mechanisms

Having shown that Skype adjusts its redundancy ratio based on the observed network loss rate only, we now consider the question: Does the applied redundancy ratio provide good QoE? To address this question, we evaluate the performance of FEC mechanisms in terms of QoE in order to derive an optimal FEC design for general VoIP services.

In the following subsections we first present our methodology for deriving the optimal redundancy ratio given a desired voice quality. Then we systematically investigate how the ratio might vary depending on the audio codec, the loss pattern, and the FEC coding scheme.

Emulation Methodology

Figure 3a illustrates the flow chart of our emulation methodology. First, we encode an audio clip into voice frames, and then emulate network loss by dropping voice frames based on the Gilbert model. Depending on the applied redundancy ratio, some of the dropped voice frames would be recoverable from the redundant data at the receiver side. We then decode the resulting voice stream into a degraded audio clip and evaluate the clip's voice quality by PESQ [11], which compares the degraded audio clip with the original version and outputs an MOS [12]. The MOS is a commonly used indicator of users' perceived quality. The MOS value ranges from 1 to 5, where 1 is the lowest perceived audio quality and 5 the highest.

We repeat the above emulation for a range of redundancy ratios and a range of network loss rates. The redundancy ratio with the desired MOS score is considered the optimal redundancy ratio. For example, if the desired MOS score is 3.5, the optimal redundancy ratios are those that are rated exactly 3.5 under each network loss rate.

Optimal Redundancy Ratios for Different Codecs

First, we consider whether the optimal redundancy ratios would be the same for different audio codecs. To address this issue, we conduct emulations using two publicly available codecs:

• G.711, the most widely used codec in digital speech systems
• G.729, the codec used by SkypeOut

Unfortunately, having no access to the encoders and decoders, we are not able to derive the optimal redundancy ratios for proprietary codecs such as iSAC and SVOPC.

The optimal redundancy ratios derived by our methodology for G.711 and G.729 are shown in Figs. 4a and 4b respectively. On each graph, the contour curve labeled with a number, say 3.5, represents the combinations of loss rates and redundancy ratios that yield the same MOS score, 3.5. Thus, each contour curve represents the optimal redundancy ratios for
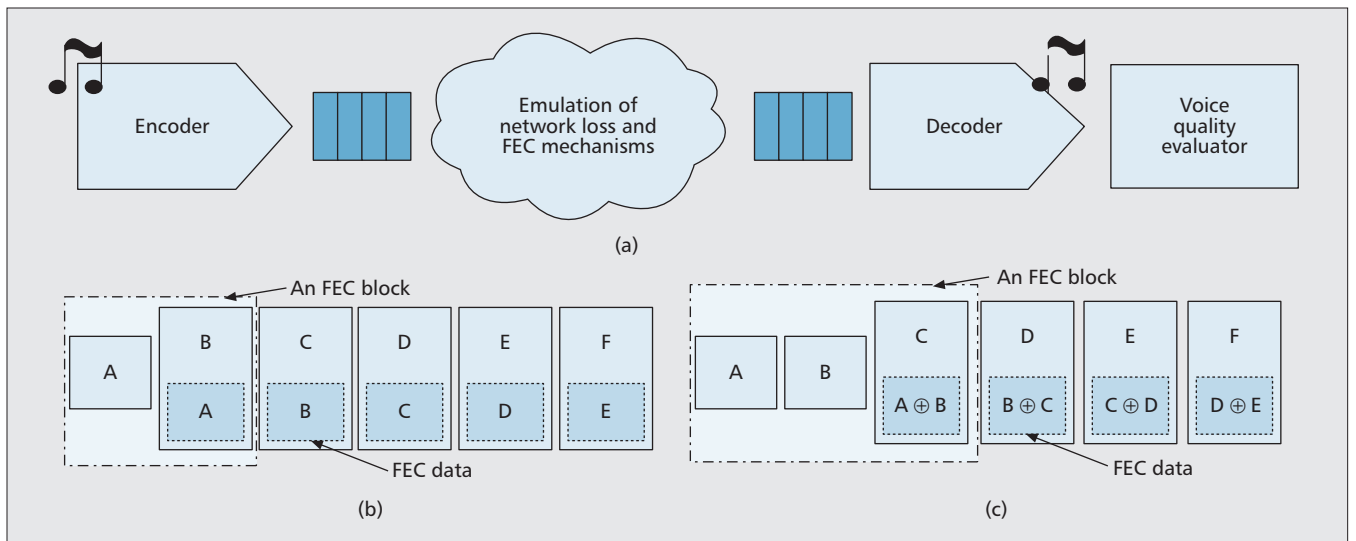
Figure 3. *The flow chart of our emulation methodology and two examples of the Reed Solomon FEC coding schemes: a) the flow of information in our emulation for computing audio quality under a given network condition; b) RS-(2, 1) code; c) RS-(3, 2) code.*

the associated MOS value. We observe that for a certain loss rate, higher redundancy ratios yield higher MOS scores. On the other hand, for a fixed redundancy ratio, higher loss rates lead to lower MOS scores. Comparing the contour plots of the two codecs, we find that the optimal redundancy ratios required to maintain a certain MOS score for each codec are very different. For example, assuming the network loss rate is 4 percent and the desired MOS score is 3.5, the redundancy ratio should be set at 0.2 for G.711. In contrast, we can achieve the same audio quality by setting the redundancy ratio at 0.8 if G.729 is used.

Optimal Redundancy Ratio for Different Loss Patterns

We repeat the emulations, except that we now infer the optimal redundancy ratios for different burst ratios. In Figs. 4a and 4b the optimal redundancy ratios for three burst ratio settings, 1.0, 1.5, and 2.0, are plotted with black solid lines, red dashed lines, and green dotted lines, respectively. The results show that the redundancy ratio should be increased more aggressively if we wish to maintain the same audio quality under higher burst ratios. For example, to maintain a consistent level of user perception at MOS 3.5 with G.711 under network loss rate of 4 percent, the redundancy ratio should be set to 0.2 when the burst ratio is 1; however, it should be set to 0.45 and 0.75 when the burst ratios are 1.5 and 2, respectively.

Optimal Redundancy Ratio for FEC Coding Schemes

Since the payloads of Skype packets are encrypted, we cannot determine which FEC coding scheme Skype employs. Nevertheless, based on the observation that the payload size doubles when FEC data is added, it is highly likely that Skype piggybacks the coding of previous voice packets onto the current packet. This FEC coding scheme, which is also known as Reed-Solomon (n,k) code, or RS-(n,k) code, is widely used in VoIP systems [13]. An RS-(n,k) code can recover all losses in the same FEC block if and only if at least k out of n packets are received. Figure 3b shows an example of RS-(2,1), where each packet piggybacks a copy of the previous packet. Since each packet carries redundant FEC data, the redundancy ratio in this example is 1. Under RS-(2,1) coding, if a voice packet is dropped during transmission, it can be restored if the subsequent packet is not dropped and carries redundant voice data.

Other Reed-Solomon codes, such as RS-(3,2) or RS-(4,3),

can also have the double-size payloads we observed in Skype's voice traces. An example of RS-(3,2) with a redundancy ratio of 1 is shown in Fig. 3c. In this case each packet carries the multiplex of its two previous packets. Under RS-(3,2) coding, a packet can be recovered if the other two packets in the same FEC block have not been dropped and at least one of them carries FEC data. For example, when packet B is lost, it is recoverable from $C \oplus (B \oplus C)$ if both packets C and D are received successfully.

Figure 5 shows the QoE analysis of the RS-(2,1) and RS-(3,2) codes under different levels of loss burstiness. The codec in these experiments is G.729, which is used by SkypeOut. Compared to the RS-(2,1) code, RS-(3,2) is more resilient to network loss. For example, under the RS-(3,2) code, if packet B is lost, it can be recovered either from $C \oplus (B \oplus C)$ if packets C and D are received correctly, or from $A \oplus (A \oplus B)$ if packets A and C are received. On the other hand, under the RS-(2,1) code, packet B can only be recovered from the piggybacked copy in packet C. Figure 5a compares the RS-(2,1) and RS-(3,2) codes under uniform random loss (i.e., the burst ratio equals 1). As shown in the figure, to maintain the voice quality at MOS 3.5 when the network loss rate is 4 percent, the redundancy ratio needed for RS-(3,2) must be set at 0.6, compared to 0.8 for RS-(2,1).

Moreover, RS-(3,2) is also more resilient to bursty loss. For example, if both packet A and B are lost, A is not recoverable under RS-(2,1) code. However, under RS-(3,2), if packets C and D are both received successfully, packet B can be recovered. Once B is recovered, packet A could also be restored from $B \oplus (A \oplus B)$. When the burst ratio is 1.5, to maintain a consistent level of user perception at MOS 3.3 under a network loss rate of 4 percent, the redundancy ratio should be set at 0.6 for RS-(3,2), as shown in Fig. 5b; however, it would need to be at 0.85 for RS-(2,1).

Under RS-(3,2) coding, it might be necessary to wait for three subsequent packets to arrive before the receiver can recover a lost packet. With iSAC and SVOPC, whose framing times could be as high as 60 ms, waiting for three more packets is equivalent to adding an additional 180 ms of FEC delay in the worst case. This would not be acceptable for real-time interactive voice communication. The FEC delay for other Reed-Solomon codes, such as RS-(4,3), may be worse, but they are more resilient to network loss. The choice of FEC coding scheme is a trade-off between responsiveness and loss resilience.
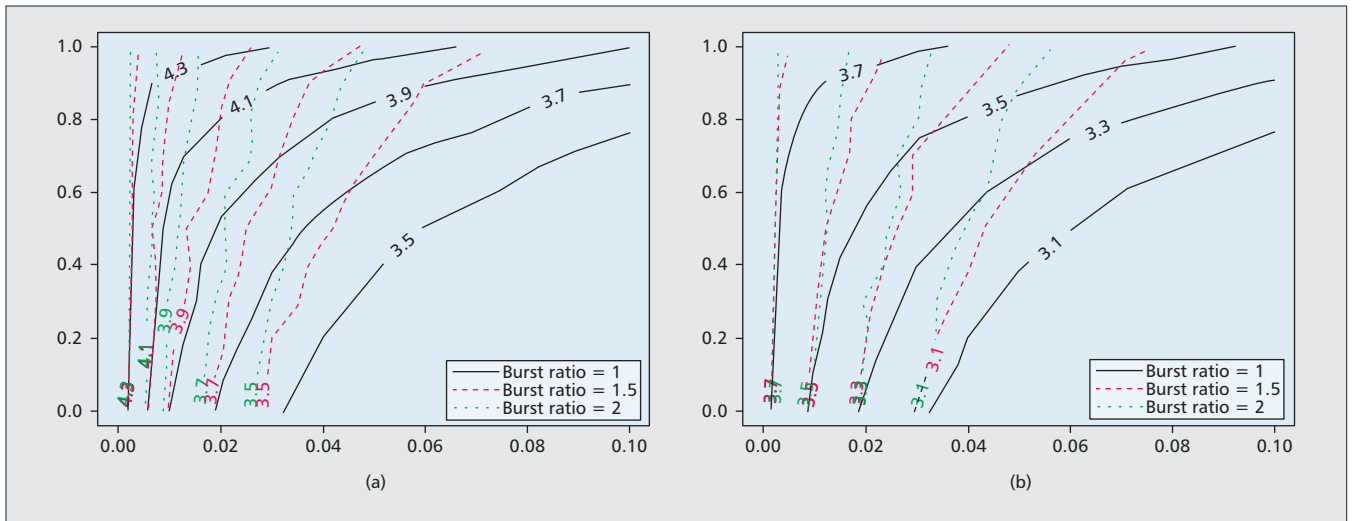
Figure 4. *The contour plots of audio quality scores under three burst ratio settings: a) G.711; b) G.729.*
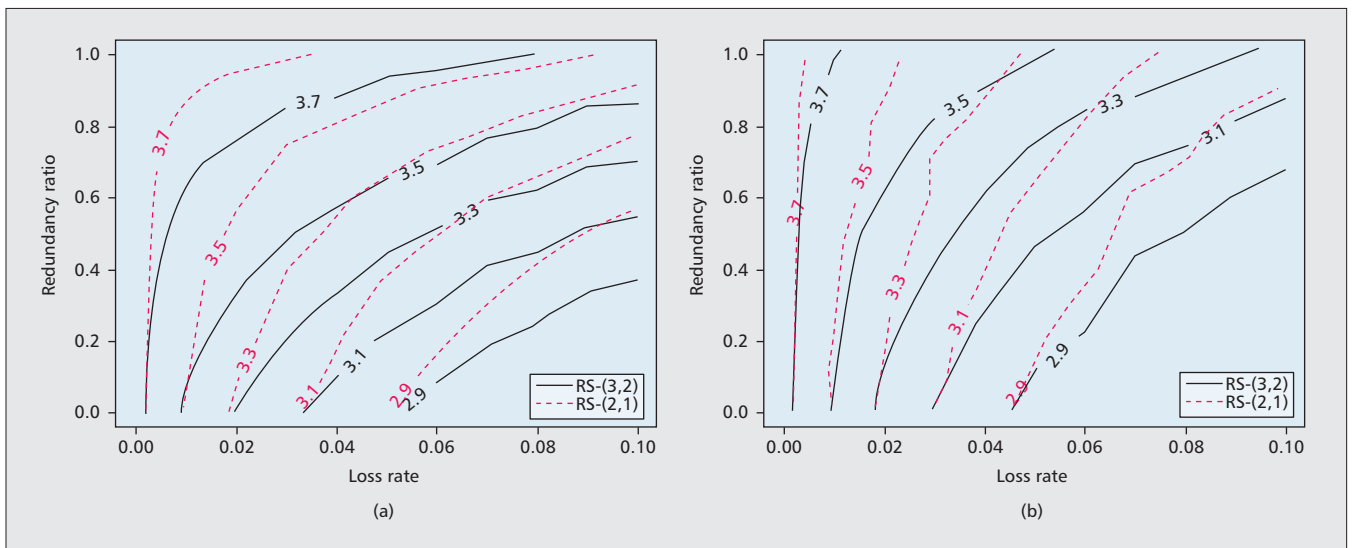

Figure 5. *The contour plots of audio quality scores under two Reed-Solomon coding schemes: a) burst ratio = 1; b) burst ratio = 1.5.*

Skype vs. Optimal

Having determined Skype's FEC mechanism in the previous section and derived the optimal algorithm above, we can now assess whether Skype's redundancy ratio settings are optimal. In Fig. 6 we overlap Skype's observed redundancy ratios and the optimal redundancy ratios for G.729; each graph corresponds to a certain burst ratio. By comparing the contour curves and Skype's redundancy ratio curve, we find that *Skype fails to maintain consistent voice quality*.

For example, as shown in Fig. 6a, Skype's audio quality is better than that of an MOS score of 3.5 when the loss rate is higher than 4 percent or lower than 1 percent, but its quality level is much lower than 3.5 when the loss rate is between 2 and 4 percent. The inconsistency in voice quality could be frustrating for users. On the other hand, assuming that the desired MOS score is 3.3, this phenomenon indicates that Skype may inject more redundant traffic than necessary into the network by over-adjusting the redundancy ratio. That is, Skype injects extra traffic into the network. This would increase the probability of congestion and in turn result in lower quality of experience for users. In contrast, by adjusting the redundancy ratio to the optimal redundancy ratio derived

by our methodology, we can ensure a balance between bandwidth utilization and voice quality.

In Fig. 6b we quantify how much Skype's FEC mechanism deviates from the policy that achieves consistent audio quality under various network conditions. The graphs are computed based on the assumption that the desired MOS score is 3.4, as Skype's audio quality is mostly around that level in our emulation scenarios. For each network setting, we plot the bandwidth Skype uses and the MOS score Skype provides on the respective normalized scales. The desired MOS score and the bandwidth required to achieve the desired audio quality are both set at 100. In this figure we observe that the bandwidth utilization and audio quality of Skype fluctuate over different network settings. As in the scenario with a 2 percent loss rate in Fig. 6b, Skype uses too little bandwidth, which results in a quality level lower than desired. Conversely, as in the scenarios with a higher than 4 percent loss rate in Fig. 6b, Skype injects too much redundant data and thus achieves a quality level higher than necessary.

Could Skype Be More Satisfying?

Yes, Skype could be more satisfying. Our results show that Skype's audio quality is not consistent as it adjusts the redun-
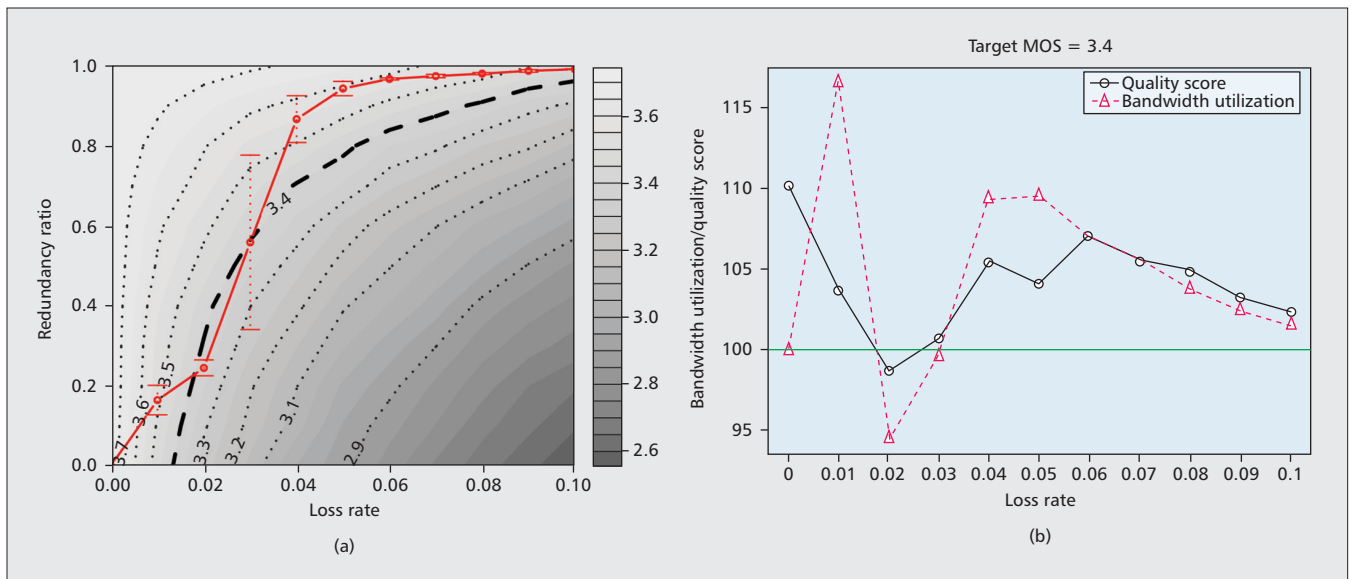
Figure 6. *a) Comparison of Skype's FEC mechanism vs. the optimal redundancy ratios (burst ratio = 1); b) quantification of how Skype's FEC mechanism deviates from optimal (burst ratio = 1).*

dancy ratio independent of the codec, redundancy coding scheme, and network loss burstiness. The inconsistency in voice quality may result in user frustration or overutilization of bandwidth. Therefore, to balance the needs of users and ensure network efficiency, a more sophisticated FEC mechanism is required.

Acknowledgment

References

[1] D. Bonfiglio *et al.*, "Tracking Down Skype Traffic," *Proc. IEEE INFOCOM '08*, Phoenix, AZ, Apr. 2008.
[2] S. Baset and H. Schulzrinne, "An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol," *Proc. IEEE INFOCOM '06*, Barcelona, Spain, Apr. 2006.
[3] S. Guha, N. Daswani, and R. Jain, "An Experimental Study of the Skype Peer-to-Peer VoIP System," *Proc. 5th Int'l. Wksp. Peer-to-Peer Sys.*, Santa Barbara, CA, Feb. 2006, pp. 1–6.
[4] Skype, "Skype P2P Telephony Explained"; http://www.skype.com/help/guides/p2pexplained/
[5] D. Bonfiglio *et al.*, "Detailed Analysis of Skype Traffic," *IEEE Trans. Multimedia*, vol. 11, no. 1, Jan. 2009, pp. 117–26.
[6] J. Lindblom, "A Sinusoidal Voice Over Packet Coder Tailored for the Frame-Erasure Channel," *IEEE Trans. Speech Audio Process.*, 2004.
[7] T.-Y. Huang, K.-T. Chen, and P. Huang, "Tuning Skype's Redundancy Control Algorithm for User Satisfaction," *Proc. IEEE INFOCOM '09*, Rio de Janeiro, Brazil, Apr. 2009.
[8] L. Rizzo, "Dummynet and Forward Error Correction," *Proc. USENIX Annual Tech. Conf.*, 1998.
[9] ITU-T Rec. G.107, "The E-model, A Computational Model for Use in Transmission Planning," Mar. 2005.
[10] E. Gilbert, "Capacity of a Burst-Noise Channel," *Bell Sys. Tech. J.*, vol. 39, Sept. 1960, pp. 1253–65.
[11] ITU-T Rec. P.862, "Perceptual Evaluation of Speech Quality (PESQ), an Objective Method for End-to-end Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs," Feb. 2001.
[12] ITU-T Rec. P.800, "Methods for Subjective Determination of Transmission Quality," 1996.
[13] W. Jiang and H. Schulzrinne, "Comparison and Optimization of Packet Loss Repair Methods on VoIP Perceived Quality under Bursty Loss," *Proc. NOSSDAV '02*, Miami Beach, FL, May 2002.

Biographies

TE-YUAN HUANG (huangty@stanford.edu) is a Ph.D. student in the Computer Science Department at Stanford University. She received her B.S. in computer science from National Chiao Tung University in 2006 and her M.S. in electrical engineering from National Taiwan University in 2008. She is interested in multimedia networking, wireless networking, mobile computing, and mobile social networking.

POLLY HUANG [M] (phuang@cc.ee.ntu.edu.tw) is an associate professor at the Department of Electrical Engineering, the Graduate Institute of Communication Engineering (joint appointment), and the Graduate Institute of Networking and Multimedia (courtesy) of National Taiwan University. She received her Ph.D. in computer science from the University of Southern California, and her B.S. in mathematics from National Taiwan University. Prior to joining National Taiwan University, she spent time in the early stage of her career at AT&T Labs-Research, Swiss Federal Institute of Technology Zurich, and the University of California at Los Angeles. Her research interest spans wireless sensor networking and multimedia networking. She is a member of ACM.

KUAN-TA (SHENG-WEI) CHEN (ktchen@iis.sinica.edu.tw) is an assistant research fellow at the Institute of Information Science and the Research Center for Information Technology Innovation (joint appointment) of Academia Sinica. He received his Ph.D. in electrical engineering from National Taiwan University in 2006, and his B.S. and M.S. in computer science from National Tsing Hua University in 1998 and 2000, respectively. His research interests include Internet and multimedia QoE management, Internet measurement, VoIP, network security, and online games.

PO-JUNG WANG (r97942139@ntu.edu.tw) received his B.S. in communication engineering from National Central University in 2009. He is currently a Master's student in the Graduate Institute of Communication Engineering, National Taiwan University. His research interests include network traffic analysis and measurement of VoIP, quality of service, and performance evaluation. He is also interested in network security and reverse engineering.