An Adaptive Codec Switching Scheme for SIP-based VoIP

Ismet Aktas, Florian Schmidt, Elias Weingärtner,
Cai-Julian Schnelke, and Klaus Wehrle
{lastname}@comsys.rwth-aachen.de

Chair of Communication and Distributed Systems,
RWTH Aachen University
Ahornstr. 55, 52074 Aachen, Germany

Abstract. Contemporary Voice-Over-IP (VoIP) systems typically negotiate only one codec for the entire VoIP session life time. However, as different codecs perform differently well under certain network conditions like delay, jitter or packet loss, this can lead to a reduction of quality if those conditions change during the call. This paper makes two core contributions: First, we compare the speech quality of a set of standard VoIP codecs given different network conditions. Second, we propose an adaptive end-to-end based codec switching scheme that fully conforms to the SIP standard. Our evaluation with a real-world prototype based on Linphone shows that our codec switching scheme adapts well to changing network conditions, improving overall speech quality.

Keywords: VoIP communication, Codec Switching, SIP

1 Introduction

The constantly changing dynamics of wireless and mobile environments are a great challenge for Voice-over-IP (VoIP) communications. Current VoIP software has only limited capabilities to deal with these dynamics. They typically support a number of codecs that differ in the optimal speech quality and required network parameters (jitter, packet loss rate, bandwidth). However, those VoIP clients typically negotiate one single codec for the entire duration of the call. While these codecs might be able to adapt themselves to a limited degree to changing network conditions such as available bandwidth, network delay or packet loss rate change in the meantime, the VoIP clients abide with their initial codec choice. Hence, they often apply a codec that is not well suited for the present network situation although better codec choices would be available.

To our knowledge, none of these VoIP clients implement an adaptive strategy to switch the session's speech codec upon changing network conditions.

In this paper we investigate how we can improve the speech quality of SIP-based VoIP calls using codec switching when the network conditions change. Specifically, this paper makes the following contributions:

- We first compare the speech quality of four standard speech codecs (GSM, PCMU, PCMA, Speex) given different network conditions using the MOS-LQO [7, 2] metric (Section 2).
- We propose an adaptive codec switching scheme that is fully compliant with the Session Initiation Protocol (SIP). It allows specialized SIP VoIP clients to dynamically adjust their codec choice to the current network conditions (Section 3).
- We evaluate our codec switching scheme using a real-world prototype build around the Linphone [3] client software. Our evaluation shows that our scheme is well able to deliver an increased speech quality if the network conditions change during a VoIP session (Section 4).

We discuss important related work in Section 5. Section 6 concludes the paper with final remarks.

2 Sensitivity of speech codecs to network conditions

In this section we first discuss which networking conditions influence the speech quality of VoIP sessions in general. After a brief description of our evaluation environment, we analyze the influence of these different network conditions on the speech quality of four different free codecs: Speex (in its $8kHz$ version), GSM-FR and G.711, also known as Pulse Code Modulation (PCM) with its two variants PCMU and PCMA. In order to objectively evaluate perceived speech quality, we rely on the PESQ tool [7, 2] that rates perceived quality with a MOS-LQO (mean opinion score – listening quality objective) score ranging from 1 (bad quality) to 5 (excellent quality).

2.1 Influencing network conditions

We now discuss the interplay of speech quality and network conditions. In this paper we regard (1) packet loss, (2) jitter, and (3) available bandwidth as the factors that define the current network condition. We opted for these parameters as we believe that they have the strongest effect on VoIP communications.

Packet loss strongly influences VoIP service quality because of the real-time streaming nature of VoIP traffic; lost VoIP packets are typically not retransmitted. Hence, knowing about how each codec can cope with packet loss is crucial for picking an adequate codec under unstable network conditions.

Similarly, the prevalence of **jitter**, that is, the variation of the end-to-end delay from packet to packet, also affects VoIP communication. VoIP's real-time properties require packets to arrive steadily. If the delay variation is too high, packets arrive too late to be of any use, and are discarded. Typically, streaming implementations include a jitter buffer that helps reducing the impact of this effect by buffering the packets for a short while before playing them back, effectively trading a reduction in packet loss for a higher delay. However, since delay has to be kept low for VoIP, this technique is limited, and if jitter increases beyond the bounds of the buffer, it negatively influences speech quality. We regard jitter as an important network condition factor, as the abilities of the considered codecs to handle packet delay variations differ.

Finally, **bandwidth** is important as different codecs have different bandwidth requirements due to the fact that they employ different amounts of data compression. High bit rate codecs have larger bandwidth requirements, but usually provide higher quality.

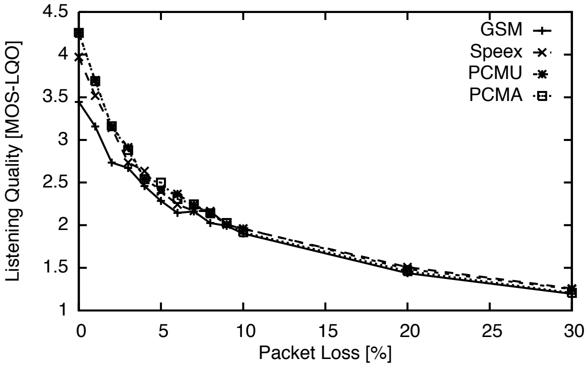

Fig. 1. Influence of packet loss on perceived speech quality for several voice codecs.

2.2 Testing Environment

In order to analyze the influencing parameters we conducted several experiments. To assess speech quality, we used the PESQ test standardized by the ITU-T in the full-reference version. As reference file, we used the ITU-T test file u_af1s03.wav (female voice speaking two sentences with a short pause between them) [10]. For VoIP communication, we employed the Linphone client as it is open source and already offers a fair set of commonly used speech codecs.

Our test setup consisted of two notebooks. Both notebooks were running Ubuntu 10.04 (LTS) and were connected to a router via 100 $MBit/s$ Ethernet. We chose a wired connection for these tests as we wanted to have full control over channel effects, and avoid additional uncontrollable environmental effects that would be witnessed in a WLAN connection.

All tests followed the same order. One notebook, the sender, initiated the call and transmitted the ITU-T test file via Linphone. The other notebook, the receiver, answered the call and recorded the audio output. We used netem [5] to insert jitter or packet loss into the connection in a controlled fashion, and employed traffic shaping via the token bucket filter [9] to reduce the available bandwidth. For each combination of codec and a certain packet loss/jitter/bandwidth, we repeated the experiment 100 times.

2.3 Codec performance under different network conditions

Figures 1, 2, and 3 show our experimentation results for the codecs GSM, Speex, PCMU (PCM with μ-law encoding), and PCMA (PCM with A-law encoding) under different network conditions.

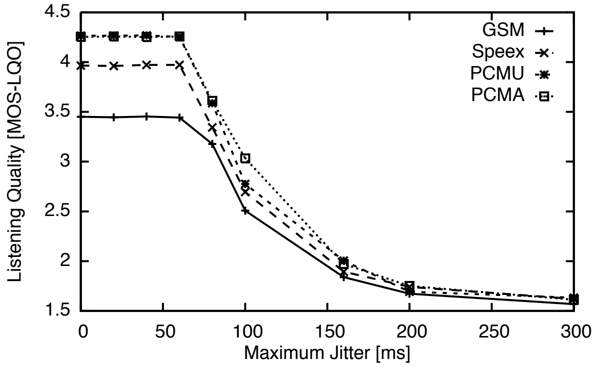

Fig. 2. Influence of delay jitter on perceived speech quality for several voice codecs.

In Figure 1 the experimentation results for varying packet loss rates are shown that are manipulated with the help of netem [5]. The results for the different codecs are very close to each other with no codec having a real advantage over another. As the values above 10% packet loss are already too bad for a real communication, codec switching does not make any sense.

Jitter was also manipulated with netem. The performance was tested with a packet delay following a normal distribution and a maximum variation from $\pm 20\,ms$ to $\pm 100\,ms$ in $20\,ms$ steps and additionally for $\pm 160\,ms$, $\pm 200\,ms$ and $\pm 300\,ms$. The chosen delay variations also ultimately led to packet reordering. Figure 2 shows that the limit of the jitter buffer of all codecs is reached roughly between $60\,ms$ and $80\,ms$. Beyond this point, the speech quality degrades sharply for all codecs all codecs, and their MOS-LQO ratings converge.

In order to limit the bandwidth, we used the token bucket filter [9] traffic shaper.

Each audio codec was tested with upstream bandwidths from $10\,kbit/s$ up to $100\,kbit/s$ in $10\,kbit/s$ steps. The results can be seen in Figure 3. The divergence between the value reached in $10\,kbit/s$and the $20\,kbit/s$ test cases of the PCMU and the PCMA codecs may be due to the extremely low quality, at which output is so garbled due to data loss that quality assessment fails to properly evaluate the speech. From $20\,kbit/s$ to $30\,kbit/s$ the GSM codec performs slightly better than the Speex codec. Up from $40\,kbit/s$ until $90\,kbit/s$ the Speex codec outperforms all other investigated codecs. After $90\,kbit/s$ the PCMU and the PCMA codecs perform better than the other. Further tests are not necessary as neither

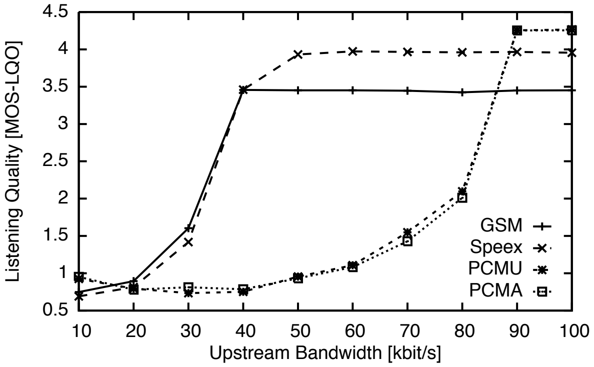

Fig. 3. Influence of available bandwidth on perceived speech quality for several voice codecs.

the Speex codec nor the GSM codec are expected to outperform the PCMU or the PCMA codecs at higher bandwidths.

All in all the tests demonstrates that with the available set of codecs, bandwidth is the best characteristic to select a codec since it shows the most difference in quality. In all remaining cases the codecs behave similarly well, resulting in no need for a codec change. The experiments also show that the GSM codec can be dropped as a useful audio codec because it never outperforms the other codecs in any of the experimented network conditions.

Even though PCMA and PCMU always perform marginally better than Speex in the packet loss and jitter tests, we have to consider the low MOS-LQO value of the two PCM codecs at upstream bandwidths smaller than 90 kbit/s. Here the selection of Speex as a codec provides a better MOS-LQO value.

As a result, we draw the conclusion that especially switching between Speex and PCM[1] on available bandwidth is reasonable. In particular, we should use Speex for every bidirectional bandwidth between 0 kbit/s and 180 kbit/s while PCM above 180 kbit/s bandwidth. Note that, since we only investigated a limited set of codecs, this proposed switching decision is rather simple. The switching scheme proposed in the following will, however, also work with more sophisticated switching decisions and more codecs to switch between.

3 Adaptive Codec Switching

Based on the previous findings we now propose an adaptive codec switching scheme that dynamically switches the codecs of a SIP session in order to improve the speech quality. We first discuss the overall system design before we describe the switching scheme in further detail.

[1] PCMA or the PCMU codec curves are very similar in all cases, therefore from now on we use only the term PCM to indicate both

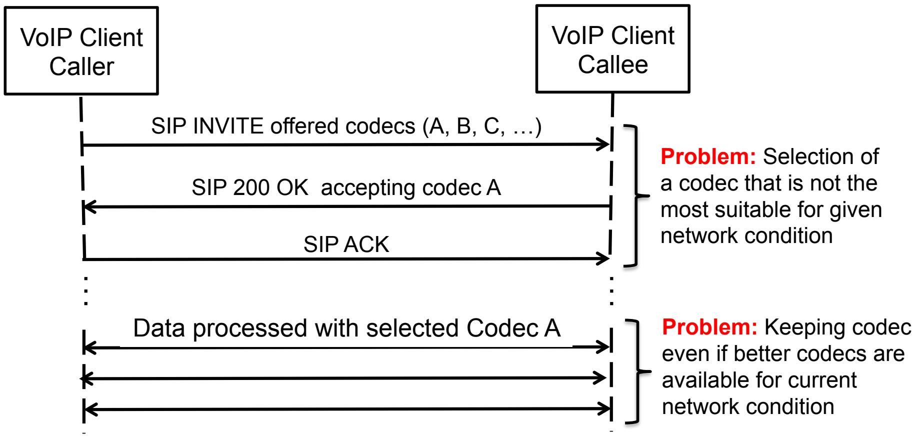
VoIP Client
Caller
VoIP Client
Callee
SIP INVITE offered codecs (A, B, C, …)
SIP 200 OK accepting codec A
SIP ACK
Data processed with selected Codec A
Problem: Selection of a codec that is not the most suitable for given network condition
Problem: Keeping codec even if better codecs are available for current network condition

Fig. 4. A static codec selection scheme and its inherent problems.

3.1 System Design

In todays VoIP clients typically the Session Initiation Protocol (SIP) is used to establish, modify and terminate sessions where two or several more participants are involved [8]. After a session is established, the Real-Time Transport Protocol is used to transport VoIP data to the participant. RTP is usually used in conjunction with its helper protocol Real-Time Control Protocol (RTCP) that provides periodic feedback about the the reception quality. A typical VoIP session creation and communication is depicted in Figure 4. The caller sends an INVITE message with a set of usable codecs. If the callee agrees to the conditions, it sends the SIP 200 OK accepting code with the preferred codec in the message body. In a third step the caller sends out an ACK message to confirm the the start of call. After the call is established the whole conversation is encoded with the selected codec. There are two problems in a wireless and mobile environment where network conditions change rapidly and unpredictable. First, the selected codec at the beginning of the call may not be well suited to the network conditions at the beginning of the call. Second, network conditions may change, so there is also a need during a call to detect such changes.

In Figure 5 we present the design idea of our approach. Before a call, we measure the bandwidth and determine the adequate codec and send an INVITE message with the determined codec to the callee. The answer of the callee and the ACK of the caller remains same.

During a call we monitor the network conditions. If a high bandwidth consuming high quality codec is used, we check if the current packet loss given by RTCP reports raises above a certain threshold to decide whether we change to a low bandwidth consuming low quality codec. If we are currently using a low bandwidth consuming low quality codec, we measure the currently available bandwidth regularly. If it is above a certain threshold, we change to a high bandwidth consuming high quality codec. To coordinate the change with the callee,

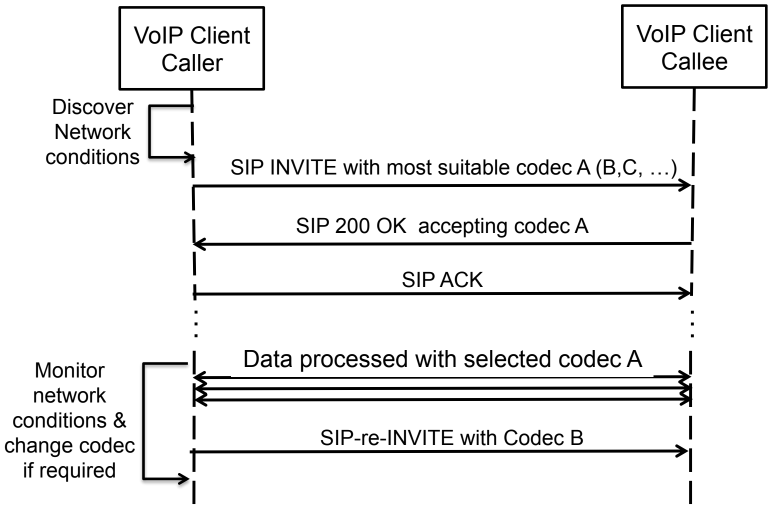
VoIP Client
Caller
VoIP Client
Callee
Discover
Network
conditions
SIP INVITE with most suitable codec A (B,C, …)
SIP 200 OK accepting codec A
SIP ACK
Data processed with selected codec A
Monitor
network
conditions &
change codec
if required
SIP-re-INVITE with Codec B

Fig. 5. Proposed codec selection scheme. Before the call is sent out, the available bandwidth is measured, and the optimal codec is chosen. During the call, network conditions are measured and codecs switched when beneficial.

we used the re-INVITE message defined in the SIP standard. This message facilitates adding, removing, or modifying a session. Our codec change constitutes such a modification. The detailed description on how we selected these conditions for the available set of codecs that we have is explained in the following sections.

3.2 Switching Scheme

As discussed, bandwidth is the most relevant network condition for our set of available codecs. In Figure 6 we graphically show the decision graph for our codec selection scheme which runs fully automatic and without user interaction.

In a first step we have to ensure that we use the best performing codec for the current bandwidth when we start the call. Since Linphone does not offer a bandwidth measuring possibility, we used WBest [11]. Although this causes longer initiation time (around 1 second) for a call, we believe that such a short time is not annoying for a user if it is at the beginning of a call. If the available bandwidth exceeds our threshold ($180\,kbit/s$), we send out a SIP INVITE message with PCM as a codec. If it is smaller than our threshold, we offer only the low bandwidth consuming codec Speex.

Recognizing a decrease in the bandwidth is fairly easy. If the codec needs more bandwidth than we have available, this ultimately leads to packet loss. RTCP reports already provide information about packet loss. If the reported packet loss increases above our threshold of 10%, we switch from the high bandwidth consuming codec to the low bandwidth consuming.

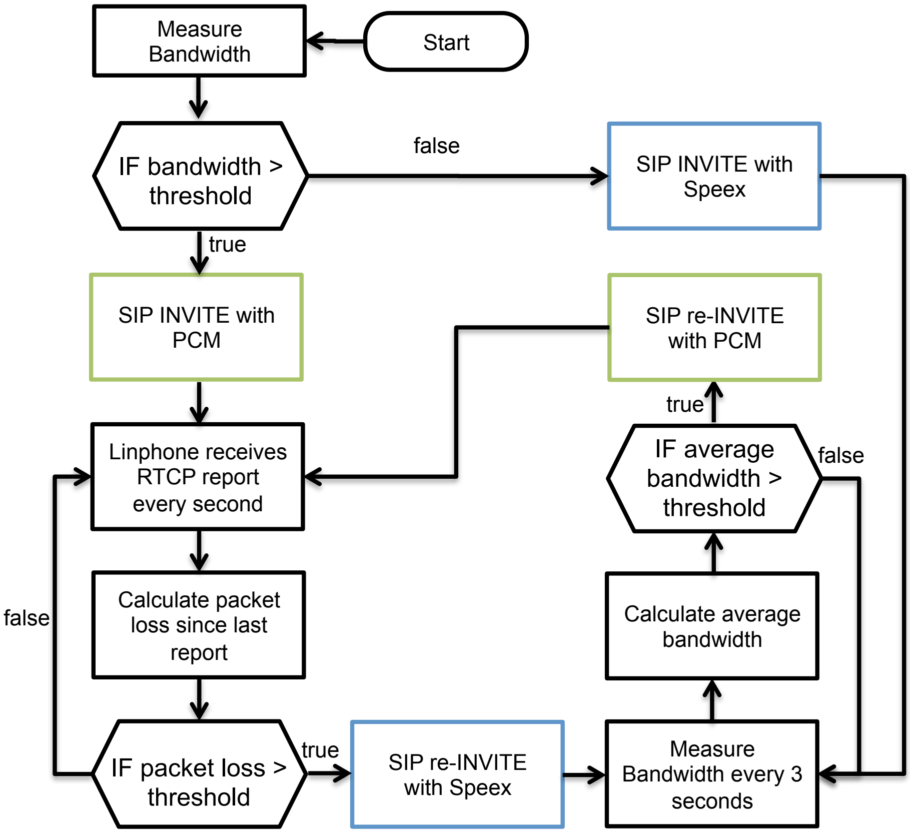
Measure Bandwidth
Start
IF bandwidth > threshold
false
SIP INVITE with Speex
true
SIP INVITE with PCM
SIP re-INVITE with PCM
true
Linphone receives RTCP report every second
IF average bandwidth > threshold
false
Calculate packet loss since last report
Calculate average bandwidth
false
IF packet loss > threshold
true
SIP re-INVITE with Speex
Measure Bandwidth every 3 seconds

Fig. 6. Flowchart of the adaptive codec switching scheme.

An increase of available bandwidth is harder to discover. Zero percent packet loss does not necessarily mean that we have enough bandwidth to spare to switch from a codec with low bandwidth consumption to one with a high bandwidth consumption. So we once again have to use WBest to get the currently available bandwidth. We measure the available bandwidth every 3 seconds as every measurement with WBest creates an overhead to our network channel. This is a good trade-off between reacting to an increase fast enough and overhead. When using PCM we do not run WBest at all because there is no benefit to finding out whether even more bandwidth is available. To ensure that a bursty, short-term increase in bandwidth does not lead to a premature codec change, we calculate the bandwidth over a sliding window history of three measurements. We switch from Speex to PCM conservatively because switching to PCM at a bandwidth below 180 $kbit/s$ leads to considerable quality degradation and should be avoided.

Note that the codec switching intelligence lies in the hands of the caller's modified Linphone client. This has three main reasons. (1) There is no need

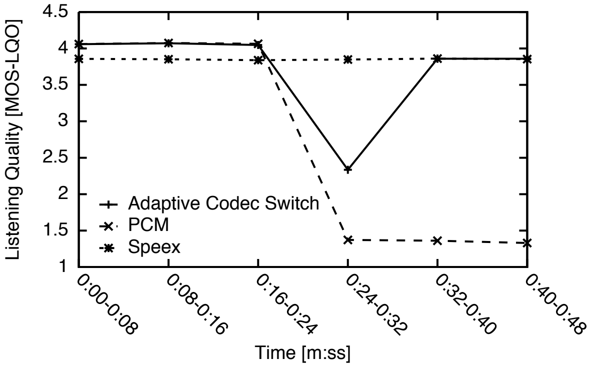

Fig. 7. MOS-LQO values for decreasing bandwidth. At 24s, bandwidth was reduced from 200 kbit/s to 65 kbit/s, and our scheme switched from PCM to Speex.

for both sides to create overhead by running bandwidth measurements with WBest. (2) If one of the parties has a too low bandwidth the packet loss will be recognized by the caller via the RTCP reports no matter whether the caller or the callee has a bad connection. (3) A caller-only implementation means that only one side of the communication has to be changed, the changes are backwards-compatible, and deployment can be done incrementally because it does not require any support from the callee. However, one optimization that could be done at the callee's side is to increase the frequency at which RTCP reports are sent. That way, the caller is informed earlier about packet loss rates that suggest a codec switch, which increases responsiveness of our scheme and increases overall speech quality.

4 Evaluation

In our evaluation we investigate the speech quality achieved by our codec selection scheme. We compare our scheme with a static use of either Speex or PCM. To highlight the effects of the switch in either direction, we present two evaluation cases: a bandwidth increase and a bandwidth decrease situation.

In the decreasing case, the bandwidth is limited to 200 $kbit/s$ initially, and is further reduced to 65 $kbit/s$ after 24 seconds. Conversely, in the increasing case, the available bandwidth is increased from 65 $kbit/s$ to 200 $kbit/s$ 24 seconds into the experiment. We use the same ITU-T test file as in Section 2.3. We loop that test file six times to create a 48-second test file out of the 8-second sample.

Note that each case was tested 20 times with each approach (Speex, PCM and our codec switch scheme). The total MOS-LQO value for the whole test period of 48 seconds measured by the PESQ tool is not fair to compare, since the longer the test file is the lesser influence is of the switching gap from one codec to another. Therefore, we decided to divide each record to 8 second chunks

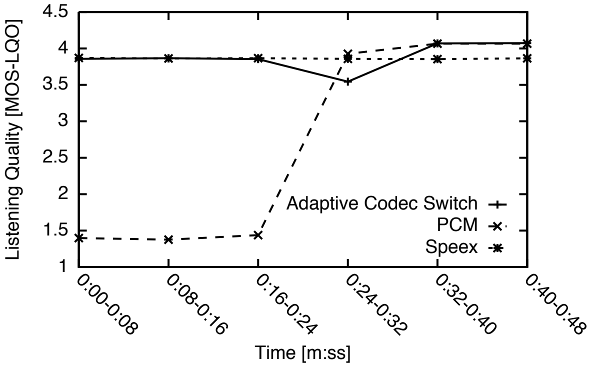

Fig. 8. MOS-LQO values for increasing bandwidth. At 24s, bandwidth was increased from 65 kbit/s to 200 kbit/s, and our scheme switched from Speex to PCM.

and compare those chunks with our original 8 second test file. This has the further advantage that it also shows the perceived quality over the course of the experiment, instead of one aggregated result.

The results from the decreasing case are shown in Figure 7. As expected, the speech quality of the Speex codec stay constant throughout the test, because the bandwidth limitation to $65\,kbit/s$ is still above Speex's requirements. On the other hand, PCM shows a strong degradation of quality after the bandwidth reduction. The effect of our codec switching scheme can clearly be seen. Note also how it correctly chooses PCM as initial codec. The temporary degradation between 24 and 32 seconds can be attributed to two factors. (1) The bandwidth decrease is detected due to frame losses, which reduces the listening quality in the time before the switch takes place. (2) Linphone's current implementation reacts to a re-INVITE codec switch with a small playback gap of about $200\,ms$, which also decreases the perceived quality.

The results from the increasing case are shown in Figure 8. Again, Speex's speech quality stays constant. PCM benefits from the increased available bandwidth, which leads to a strong quality improvements after the 24-second mark. Our codec-switching scheme correctly decides on the better codec to use at any given point in time by choosing Speex as initial codec, and switching to PCM after the bandwidth change at 24 seconds. The slight degradation between 24 and 32 seconds can be attributed to Linphone's playback gap, which temporarily decreases perceived quality.

To conclude our evaluation, our test show that our codec switching scheme selects the specified codec properly at the beginning and during the communication. In addition, we can see how it improves listening quality compared to a static codec choice, except for the short time of the switch itself. Note that we do not expect to change codecs very frequently, so these evaluation results overemphasize the temporary quality loss during the switch; in a real setup with long conversations and only occasional codec switches, the overall quality improvement will strongly outweigh the short degradations.

5 Related Work

Related work on adaptive codec switching typically focuses on adaptation to degrading network conditions, and only discusses adaptation to improving conditions briefly or not at all. Furthermore, to the best of our knowledge, none of these approaches provides a solution for choosing the optimal codec at the beginning of the call.

In [1] the authors suggest to use packet delay as an indicator to select codecs. Their reasoning is to detect network congestion that way, and to preemptively switch codecs before prohibitively high packet loss occurs. However, their purely analytical approach does not focus on speech quality as a metric; therefore it is not clear whether such a switching approach actually would improve quality.

In [6] packet loss is used as an indicator to switch the codec in degrading as well as improving conditions. However, how and when to infer from low packet loss that additional bandwidth is available is not discussed. The adaptation to improving conditions is not evaluated either, so this question stays open. Similarly, the authors of [12] use packet loss as an indicator for both degrading and improving channel conditions. Furthermore, they propose a handover scheme between different types of network (e.g., WiFi and WiMAX) that also takes signal strength into account. Again, no evaluation for adaptation to improving network conditions is presented. Both [6] and [12] employ a SIP re-invite technique similar to ours.

The authors of [4] propose an adaptation that combines the goals of quality and security. They continuously monitor the MOS via a no-reference scoring algorithm and then decide on which codec to use, whether to introduce additional forward error correction, and how much security overhead they can introduce without compromising quality. However, the authors do not fully address the increasing bandwidth case. Moreover, their design has neither been implemented nor tested, so it is unclear how well their approach would work in reality.

6 Conclusion and Future Work

In this paper, we analyzed the speech quality of several standard VoIP codecs for different network conditions. The results of this analysis showed that bandwidth is a defining metric for quality.

Based on that, we designed an adaptive coded switching scheme that fully conforms to the SIP standard and integrated it into the Linphone [3] VoIP client. Depending on available bandwidth, our adaptive codec switching scheme performs three tasks: (1) choosing the currently best performing codec before the actual communication starts, (2) changing to a low bandwidth consuming low quality codec when the packet loss increases, and (3) changing to a high bandwidth consuming high quality codec when the bandwidth increases.

Our evaluation shows that our solution produces improvements in the perceived listening quality compared to a static codec choice at the beginning of the call.

In the future, we plan to investigate the integration of further open codecs such as G.726 or Internet Bit Rate Codec (iLBC). These codecs have low bandwidth consumption which makes them a good alternative to Speex. In addition, we aim at evaluating the effects of further network metrics such as packet or bit error rate on codec performance. More codec choices together with more metrics may lead to an extension of our switching scheme with more sophisticated strategies that further improve listening quality.

References

1. C. Casetti, J. C. De Martin, and M. Meo. Framework for the analysis of adaptive voice over ip. In *2000 IEEE International Conference on Communications*, pages 821 – 826, June 2000.
2. ITU-T P.862 : Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, 2001.
3. Linphone. http://www.linphone.org.
4. W. Mazurczyk and Z. Kotulski. Adaptive voip with audio watermarking for improved call quality and security. *Journal of Information Assurance and Security*, 2(3):226–234, 2007.
5. netem. http://www.linuxfoundation.org/collaborate/workgroups/networking/netem.
6. S. L. Ng, S. Hoh, and D. Singh. Effectiveness of adaptive codec switching voip application over heterogeneous networks. In *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pages 7 pp. –7, Nov. 2005.
7. Pesq. http://www.pesq.org/.
8. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard), June 2002. Updated by RFCs 3265, 3853, 4320, 4916, 5393, 5621, 5626, 5630, 5922, 5954, 6026, 6141.
9. tbf. http://linux.die.net/man/8/tc-tbf.
10. Testfile. http://http://www.itu.int/rec/T-REC-P.862-200511-I!Amd2.
11. Wbest: a bandwidth estimation tool for ieee 802.11 wireless networks. http://web.cs.wpi.edu/ claypool/papers/wbest/.
12. Y. C. Yee, K. N. Choong, A. L. Y. Low, and S. W. Tan. Sip-based proactive and adaptive mobility management framework for heterogeneous networks. *J. Netw. Comput. Appl.*, 31:771–792, November 2008.