A Generic Algorithm for Mid-call Audio Codec Switching

Haytham Assem*, Mohamed Adel†, Brendan Jennings†, David Malone*, Jonathan Dunne‡ and Pat O'Sullivan‡
*Hamilton Institute, National University of Ireland Maynooth, Ireland
Email: {hitham.salama.2012, david.malone}@nuim.ie
†TSSG, Waterford Institute of Technology, Ireland
Email: madel@tssg.org, bjennings@ieee.org
‡ IBM Software Lab, Dublin, Ireland
Email: {jonathan_dunne, patosullivan}@ie.ibm.com

***Abstract*—We present and evaluate an algorithm that performs in-call selection of the most appropriate audio codec given prevailing conditions on the network path between the end-points of a voice call. We have studied the behaviour of different codecs under varying network conditions, in doing so deriving the impairment factors for non-ITU-T codecs so that the E-model can be used to assess voice call quality for them. Moreover, we have studied the drawbacks of codec switching from the end user perception point of view; our switching algorithm seeks to minimise this impact. We have tested our algorithm on different packages that contain a selection of the most commonly used codecs: G.711, SILK, ILBC, GSM and SPEEX. Our results show that in many typical network scenarios, our switching codecs mid-call algorithm results in better Quality of Experience (QoE) than would have been achieved had the initial codec been used throughout the call.**

Keywords*—*VoIP, Audio Codecs, Codec Switching, E-model.

I. Introduction

Voice over Internet Protocol (VoIP) applications have gained wide acceptance by general Internet users and are increasingly important in the enterprise communications sector. However, achieving voice quality levels for VoIP remains a significant challenge, as IP networks typically do not guarantee delay, packet loss, jitter and bandwidth levels. In a VoIP application, voice is digitized and packetized at the sender before its transmission over the IP network to the receiver. At the receiver the packets are decoded and played out to the listener as shown in Figure 1. The process of converting an analogue voice signal to digital is done by an audio "codec". Codecs vary in bandwidth required, latency, sample period, frame size and the maximum achieved end user perceived quality, thus different codecs are better suited to different network conditions.

There are different methods to measure the voice quality accurately in the VoIP networks. ITU-T publish recommendations relating to two test methods: subjective testing and objective testing. Subjective testing, as specified in ITU-T Rec. P.800 [1], involves 12-24 participants individually listening to an audio stream of several seconds and rate the audio quality on the scale of 1 (Poor) to 5 (Excellent), with these ratings being used to form a single Mean Opinion Score (MOS). Subjective testing using MOS is considered as time consuming and expensive and, importantly, it can not be done in realtime. Given this, several techniques for estimating MOS in an objective manner (without human perception) have been proposed; prominent examples are PESQ [2], [3] and the E-Model [4].

PESQ is based on the comparison of two signals to generate the MOS—a reference signal (e.g. captured at the sender) and a degraded signal (e.g. captured at the receiver). The requirement for comparison of both signals makes the approach unsuitable for live monitoring of calls. In contrast, the E-model technique, specified in ITU-T Rec. G.107 [4], is a non-intrusive method that uses network metrics locally monitored at the sender to estimate call quality, so it can be used for live call monitoring. One drawback with the E-model is that it requires knowledge of a so-called "impairment factor" of the codec, which ITU-T provide for codecs they specify, but which is not specified for a range of other commonly used codecs.

Packet loss in the IP network is considered one of the most important factors that cause degradation in the overall voice call quality—packet loss greater than 5% has been shown to have a very detrimental effect on voice quality [5]. The maximum quality that can be achieved differs from codec to codec under different packet loss rates. Given this, we focus in this paper on the use of a generic codec switching algorithm that can respond to changing network conditions during an ongoing call and switch to the most appropriate codec.

The paper is organized as follows: in §II we place our work in the context of the recent published literature on the topic. In §III, we provide a brief description of the E-model and derive impairment coefficients for some common non ITU-T audio codecs. In §IV, we explore the impact of the codec switching on voice call quality. In §V, we propose our adaptive codec switching algorithm. The results of an experiment evaluation

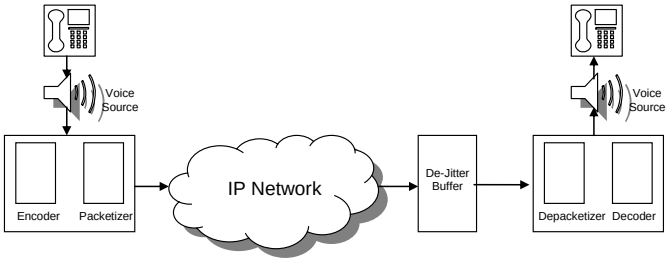

Fig. 1. Basic VoIP System Architecture.

of the algorithm are shown in §VI. §VII concludes the paper.

II. Related Work

In current VoIP applications codec switching is typically achieved via that Session Initiation and Session Description Protocols (SIP/SDP) [6], [7]. The initial session negotiation is achieved by a straightforward handshake protocol interaction wherein each peer exchanges an offer including the list of codecs it supports and a codec is selected. If one peer wishes to switch the code mid-session it initiates a similar handshake procedure is undergone to select a new codec; in this scenario it is important that both peers synchronize with each other in order to avoid data misinterpretation [8].

Aktas et al. [9] compare the speech quality of a set of standard codecs under different network conditions, and propose an adaptive end-to-end based codec switching scheme based on available bandwidth—the codec is chosen accordingly. However, they only evaluate their scheme using two codecs: PCMU and SPEEX. Sulovic et al. [10] propose an algorithm for adaptive adjustment of VoIP sources transmission rate based on voice quality estimated at the receiver. They switched between three codecs in their algorithm: G711, G729A and G723.1 5.3k, showing that their algorithm maintains high MOS values during network congestion.

Costa et al. [11] describe an adaptive codec switching technique embodied in their "NCVoIP" application. NCVoIP starts to monitor and analyze the quality of the voice, changing to a lower or higher codec transmitted rate according to predefined threshold values for each codec. They demonstrated that switching the voice codec when the bandwidth is below the transmission rate of the used codec and using TCP to encapsulate the RTP packets when network congestion exists, results in a significant voice quality improvement. Waltermann et al. [8] introduce a technique for seamless VoIP codec switching in the Next Generation Networks (NGN) based on SIP/SDP session re-negotiation by establishing a parallel media stream and RTP packet filtering. They show that their proposed approach does not cause any annoyance or interruption of the audio stream in 90% of the test cases.

The main difference between our work and those reviewed above is that we perform a detailed analysis of the impact of codec switching on voice quality for a wide range of codecs, deriving some heuristics for when and how often codec switching should be done. These heuristics are incorporated into our codec switching algorithm. In addition and unlike other reviewed approaches, we show that switching codecs based on the packet loss improves call quality but a special care should be taken to avoid the negative impact of switching.

III. Measuring Call Quality

In this section we provide an introduction to the E-model and describe how we estimated the impairment factor used in the E-model for a number of non ITU-T codecs.

A. The E-model

The E-model is used to map network metrics to an estimated Mean Opinion Score (MOS) value. It calculates a rating factor R that ranges from 0 to 100. As shown in Table I, $R = 0$ indicates the worst quality while $R = 100$ indicates the best quality. It is calculated as follows:

TABLE I. Relationship between R and Mean Opinion Score.

R	**Satisfaction Level**	**MOS**
90-100	Very satisfied	4.3+
80-90	Satisfied	4.0-4.3
70-80	Some users dissatisfied	3.6-4.0
60-70	Many users dissatisfied	3.1-3.6
50-60	Nearly all users dissatisfied	2.6-3.1
0-50	Not recommended	1.0-2.6

$$R = R_0 - I_s - I_d - I_{e-eff} + A \tag{1}$$

R_0 is the signal to noise ratio at 0 dBR, I_s is the speech voice impairment factor, I_d indicates the impairments due to the delay, I_{e-eff} is the impairments caused by codecs, the values of R_0 and I_s are defined as 94.77 and 1.41 respectively [4] and A is the advantage factor, assuming our communication system is conventional, then we neglect A value. Consequently, we have:

$$R = 93.2 - I_d - I_{e-eff} \tag{2}$$

I_d is a function of one way delay only; it can be calculated using a 6[th] order polynomial [12]. I_{e-eff} is the packet loss dependent effective equipment impairment factor and can be expressed as:

$$I_{e-eff} = I_e + (95 - I_e)\frac{Ppl}{\frac{Ppl}{BurstR} + Bpl} \tag{3}$$

I_{e-eff} is derived using a codec-specific value (I_e) which represents the impairment factor given by codec compression, and by a packet loss robustness factor (Bpl) that represents the codec robustness against random losses. The values of I_e and Bpl for several codecs are provided by ITU in G.113 recommendation [13]; they were deduced using subjective MOS tests and network experience. Ppl represents the percentage of packet loss and $BurstR$ is the burst ratio when packet loss is bursty ($BurstR > 1$) but it will be equal to 1 if the packet loss is random.

Once calculated, the R value can be used to estimate MOS using the following:

$$MOS = \begin{cases} 1 & \text{if } R < 0 \\ 1 + 0.035R + & \\ \quad R(R-60)(100-R)\times & \\ \quad 7 \times 10^{-6} & \text{if } 0 \leq R \leq 100 \\ 4.5 & \text{if } R > 100 \end{cases} \tag{4}$$

B. Deriving I_e and Bpl for non ITU-T codecs

Although the new objective E-model has been introduced by ITU-T in order to take in its account all the drawbacks of PESQ, it is still restricted to be used only with the codecs provided by ITU-T as neither the impairment factors of all the codecs factors are not provided nor can be calculated easily.

ITU-T recommendation G.113 [13] does not provide codec I_e, Bpl values for the most well know used codecs like ILBC, SILK, GSM and SPEEX. To establish these values we, for each of these codecs, estimate MOS using the PESQ method by directly comparing reference and degraded voice signals.

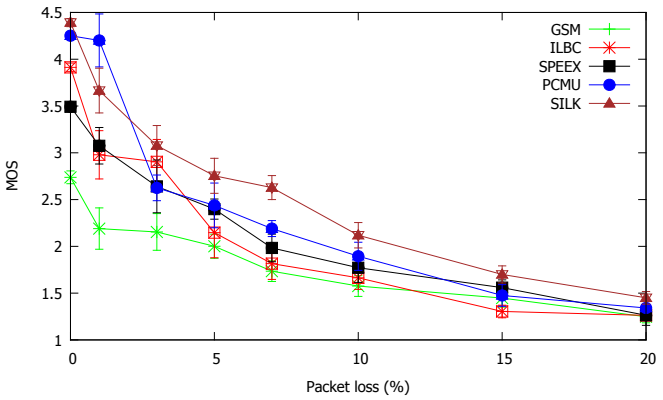

Fig. 2. Non ITU-T codecs' performance.

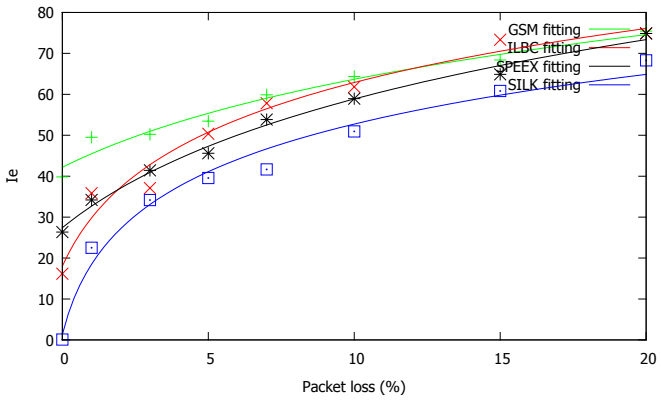

Fig. 3. Deriving I_e factor for four non-ITU codecs.

We then calculate the E-model R value using the following 3rd order polynomial fitting from [14]:

$$R = 3.026MOS^3 - 25.314MOS^2 + 87.06MOS - 57.336 \quad (5)$$

The MOS (PESQ) factor converted to rating factor R does not consider delay impairments (I_d value). Hence, we consider only the equipment impairment, I_{e-eff}, which results from the codec compression rate and packet loss. Therefore, following from (2), R can be converted to I_{e-eff} as

$$I_{e-eff} = 93.2 - R \quad (6)$$

We used PESQ to estimate the MOS for the popular G711, ILBC, SILK, GSM and SPEEX codecs at different packet loss rate ranges from $0 - 20\%$. Each MOS estimation at each percentage of packet loss was measured 5 times and we took the average in order to increase the accuracy of our results. We used Dummynet [15] to embed random packet loss rates during the session. Our results are shown in Figure 2 with the packet loss on the x-axis and the PESQ MOS score on the y-axis.

We observe that the performance of the codecs is different under packet loss rates. For example, SILK out performs the other codecs at 0% packet loss rate. PCMU gives the best performance in the range $0 - 3\%$ packet loss. Starting nearly from 4% packet loss, we found that SILK over performs until 20% packet loss. These observations suggest that switching codecs mid-session in response to increased in detected packet loss rate has the potential to deliver an improved QoE.

In Figure 3, a non linear regression model (similar to the logarithmic function in [12]) can be derived for each codec by the least squares method and curve fitting. The derived I_{e-eff} model has the following form:

$$I_{e-eff} = a \log(1 + b \times Ppl) + c \quad (7)$$

The Ppl in (7) is the packet loss rate in percentage and the parameters (a, b and c) are shown in Table II for the different codecs.

TABLE II. DERIVED LINEAR REGRESSION MODEL PARAMETERS FOR DIFFERENT CODECS.

Parameters	GSM	ILBC	SPEEX	SILK
a	22.931	20.836	28.244	18.3442
b	0.1555	0.762	0.2043	1.54894
c	42.175	18.013	27.423	1.31953

IV. IMPACT OF CODEC SWITCHING

In this section we study the impact of the codec switching process itself. Two factors can lead to degraded quality: the "switch-over gap" when codecs are switched and the overall number of switches during a session.

A. Switch-over Gaps

The switching of codecs during the communication causes a switch-over gap. We define the term switch-over gap as the time taken between sending the RE-INVITE message from the sender side and receiving the ACK from the receiver side indicating the start of transmission with the new codec, in another words, switch-over gap indicates the response time to switch to another codec. Special care should be taken for high switching gap which will lead to decrease the responsiveness time to switch to another codec. Our results show that at high packet loss rates, the RE-INVITE message will be at a higher probability of being lost, which will cause multiple retransmissions till the message reaches the intended receiver, and the same also will happen for the 200 OK and ACK messages, therefore the switch-over gap will increase more.

For guiding the design of a quick responsive codec switching algorithm, we need to minimize the response time as much as possible to make use of the appropriate codec and attain higher call quality. Since the switch-over gap is codec independent, thus we have measured the switch-over time between G711 and ILBC with a packet loss rate ranges from $0 - 40\%$. At each packet loss rate, we have measured 10 values for the switching-over gap measured in $msec$. Figure 4 shows our results indicating the packet loss percentage on the x axis while the switch-over gap on the y axis.

We identify three distinct regions. The first region which is between 0-10% packet loss corresponds to the minimum switch-over gap with an average of $0.5s$—this is the most appropriate range to switch codec. In the second region, the packet loss ranges from $10 - 30\%$ will result in an average of $2s$—in this region special care should be taken when switching because this may affect the responsiveness of the switching algorithm. In the third region between $20 - 40\%$ packet loss, it is not recommended to switch as the switching-over gap

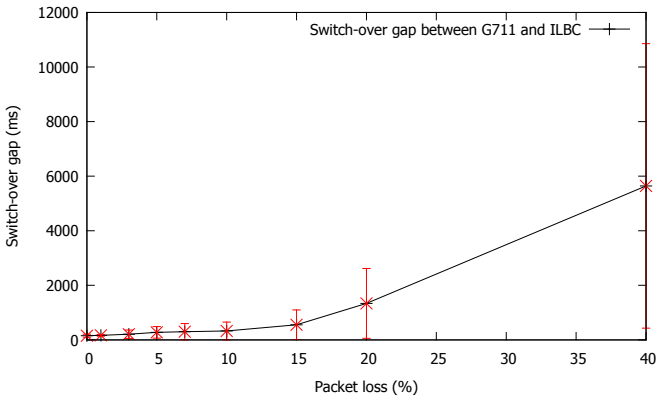

Fig. 4. Switch-over Gap Effect.

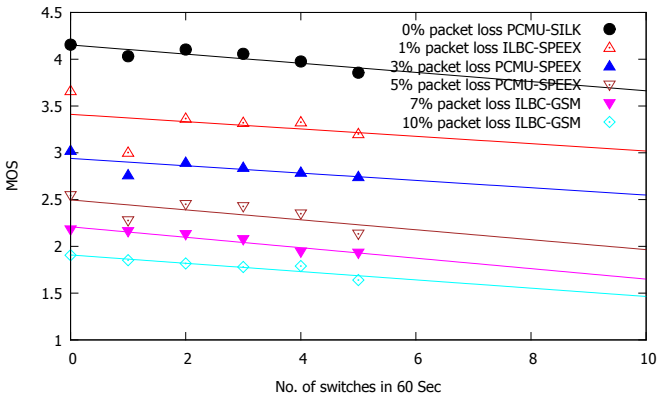

Fig. 5. Effect of Number of Switching on MOS Score.

will dramatically increase, to the extent that might lead to the change of network conditions leading to a false switching decision.

Given these observations, we focus our algorithm on switching codecs in the first region ($0-10\%$ packet loss) in order to minimize the switch-over gap in order to increase the responsiveness of our algorithm.

B. Number of Codec Switches and Silent Gap

Frequent switching of codecs during a session could cause degradation in the overall call quality; in this section we seek to quantify this effect. Restricting ourselves to $0-10\%$ packet loss rate region for minimum switch-over gap, we once again apply the PESQ algorithm to calculate MOS. We use it to quantify the degradation in MOS due to a number of 0-12 codec switches during a $60s$ period—codec switching is done at most every $5s$, which is the RTCP reporting period.

In order to measure the only degradation in the call quality as a result of increasing the number of switches, we selected pairs of codecs which have nearly the same or almost same performance. From Figure 5, we observe that at 0% percent packet loss the performance of PCMU and SILK are nearly the same, at 1% percent packet loss the performance of ILBC and SPEEX are nearly the same and at 3%, as well as at 5% packet loss, the performance of PCMU and SPEEX are nearly identical, additionally, at 7% and 10% iLBC and GSM provide close performance. Thus, we switched several times between these pairs of the stated codecs. The results are shown in Figure 5: we see that the relation between the number of switches and the MOS score is well matched by first order function. Moreover, the slopes of all the lines are nearly the same which means that the rate of degradation is nearly equal under different random packet loss rates that range from $0-10\%$. We can therefore conclude that, in this packet loss range, the degradation is approximately 0.1 in the MOS score for the effect of a single switch.

The switching of the codec during the communication could cause a silent gap in the conversation, due to buffer re-initialization. We define the term silent gap as the length of the non-audible gap that results during codec switching. This can be illustrated as shown in Figure 5 from the degradation in the MOS when there is no switching compared to 1 switch.

V. Codec Switching Algorithm

We now specify a codec switching algorithm that can be used in conjunction with an arbitrary set of codecs available to a VoIP application. The algorithm is based on the use of the E-model to estimate MOS during an ongoing voice session. Thus, it requires knowledge of the impairment factor for different codecs; we have derived these values for a range of common codecs above and ITU-T specifies them for their codecs.

The algorithm, specified in Algorithm 1, operates as follows. It assumes the call starts with a default codec set in the VoIP application. It then waits for every 2 successive RTCP reports (a control period of $10s$), each time calculating the current average packet loss rate. If the packet loss rates is in the $0-10\%$ range, the algorithm estimates the predicted call quality using the E-model for all of the other codecs available to the VoIP application. Once this is done, the MOS scores for the other codecs are compared to the one currently in use and, if the score would be improved by making a switch this is done. This decision takes into account the potential degradation in MOS due to frequent codec switching.

VI. Experimental Analysis

We implemented our codec switching algorithm in Jitsi [16], an open source audio/video Internet phone and instant messenger written in Java. We use Dummynet to emulate a range of typical network conditions. To test the codec switching algorithm, we evaluate it's use with two "packages" of codecs with one default codec, as specified in Table III. For our experiments we played a sample audio file for $3mins$, with the potential for switching a codec being assessed every $10s$.

A. First Package

In this experiment and as shown in Figure 6, we started the call using GSM codec at 0% packet loss; it took the algorithm $10s$ to switch to SILK which has the highest R at this loss rate. For the next $60s$ the MOS for all codecs is degraded by 0.1 as a result of switching. After the end of the previous $60s$, the MOS recovered from the negative effect of the switching and returns back to its value as shown in the time slice between 1:20 and 1:30. At 1:20 we emulated 1% packet loss, so the switching occurred at 1:30 to PCMU. After $40s$ and although

```
begin
    Start the call using the default codec in the VoIP application package;
    while Call is not ended do
        if (! 2 RTCP reports are received) then
            Wait until first 2 RTCP reports are received;
        end
        else
            Calculate average packet loss rate (avgPacketLossRate);
            if (avgPacketLossRate ≤ 10%) then
                Create List codecs <codec, R (codec i)> ;
                R_Current = 93.2 - Ieff; /* Calculate the rating
                factor R (current codec used) */
                MOS_Current= (MOS) R_Current;      /* Convert R
                score to MOS */
                Append the codec name used and its R values in the List;
                Calculate the number of codecs available (nOfCodecs);
                Calculate the number of switches in the previous 60
                seconds (nOfSwitches);
                for (i=0 ; i < nOfCodecs; i++) do
                    switchingEffect = nOfSwitches*0.1;
                    R (codec i) = 93.2 - Ieff;
                    MOS (codec i) = (MOS) R (codec i) -
                    switchingEffect;
                    Append in List<codecName [i], MOS (i)>;
                end
            end
            else
                Do nothing
            end
            SortByDesending (List codecs<codec, MOS (codec i)>);
            highestCodecScore = codecs [0, 1]; /* get the value of
            the codec at the top of the list */
            if (highestCodecScore > MOS_Current ) then
                codecName = codecs [0, 0];
                Switch to codecName;
                nOfSwitches++;
            end
            else
                Do nothing
            end
        end
    end
end
```

Algorithm 1: Codec Switching Algorithm.

TABLE III. CODEC PACKAGES

Package	Codecs Present	Default Codec
First	SILK PCMU GSM	GSM
Second	SPEEX ILBC GSM	GSM

the packet loss was increased to 5%, switching didn't occur at the 2:20 as one switch was already done in the previous $60s$ (-0.1 MOS) and the gain from such switch between PCMU and SILK (+0.1 MOS) was not worthwhile in the context of overall call quality. Finally, at 2:30 the codec was switched to SILK.

B. Second Package

In this experiment, as shown in Figure 7, we started the call using GSM codec at 0% packet loss; it took the algorithm $10s$ to switch to ILBC. At the 1:30, we applied a packet loss of 6%. Thus, the codec was switched to SPEEX in the next slice. At 2:10, the packet loss was decreased to 0%, thus the codec was switched back to ILBC. Although 1 switch occurred before in the previous $60s$ (-0.1MOS) it is worth switching as the total gain expected from such switch will be +0.33 MOS. In slices from 2:20-2:50, the MOS was dropped by 0.2 due to the effect of 2 switches. Consequently, at 2:50 and after the

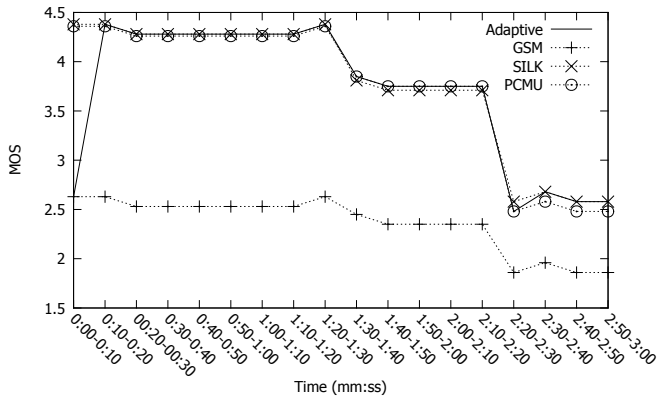

Fig. 6. MOS values for the First Package.

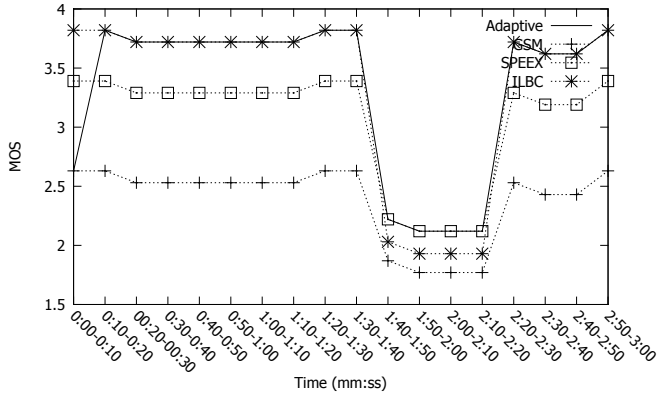

Fig. 7. MOS values for the Second Package.

end of $60s$ from the first switch at 1:40, the MOS returned back to its normal value at current packet loss rate.

VII. CONCLUSION AND FUTURE WORK

Switching codecs during an ongoing voice session can improve user's perceived quality-of-experience due to the fact that different codecs behave differently under different packet loss conditions in the network. In this paper, we empirically studied the impact of codec switching on call quality and specified a codec switching algorithm that takes these impacts into account. We found that switching codecs will result in silent gap and switch-over gaps of different lengths depending on the prevailing pack loss rates. We also found that the number of codec switches within a time interval should be limited so as not to contribute towards degradation in the call quality experienced by users. Our experiments showed that our codec switching algorithm can be applied to a range of different codec packages and that it can produce a significant improvement in voice call quality as compared to the use of a codec selected at the start of a call and maintained for the call duration. We also found that a combination of the PCMU and SILK codecs provides a solution that is more robust to moderate packet loss rates than other commonly used codecs.

For the future work, we intend to extend our algorithm to support wideband audio codecs by applying the newly developed POLQA [17] objective testing method. Furthermore, we are planning to improve our algorithm by studying loss

patterns to assess if the frequency and distribution of losses affect codecs' quality differently.

ACKNOWLEDGMENT

The authors were supported by Science Foundation Ireland (SFI) grants 07/SK/I1216a and 08/SRC/I1403.

REFERENCES

[1] ITU-T, "Methods for Subjective Determination of Transmission Quality," Recommendation P.800, 1996.

[2] ——, "PESQ an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs," Tech. Rep. Recommendation P.862, 2001.

[3] ——, "Wideband extension to Recommendation P.862 for the assessment of wide band telephone networks and speech codecs," Tech. Rep. Recommendation P.862.2, 2001.

[4] ——, "The E-model: a computational model for use in transmission planning," Recommendation G.107, 2009.

[5] S. Agrawal, J. Ramamirtham, and R. Rastogi, "Design of active and passive probes for VoIP service quality monitoring," in *Proc. 12th International Telecommnications Network Strategy and Planning Symposium (NETWORKS 2006)*. IEEE, 2006, pp. 1–6.

[6] IETF, "Session Initiation Protocol, RFC 3261," 2002.

[7] ——, "Session Description Protocol, RFC 4566," Tech. Rep., 2006.

[8] M. Walterman, B. Lewcio, P. Vidales, and S. Moller, "A Technique for Seamless VoIP-codec Switching in Next Generation Networks," in *Proc. 2008 IEEE International Conference on Communications (ICC 2008)*. IEEE, 2008, pp. 1772–1776.

[9] I. Aktas, F. Schmidt, E. Weingrtner, C.-J. Schnelke, and K. Wehrle, "An adaptive codec switching scheme for SIP-Based VoIP," in *Internet of Things, Smart Spaces, and Next Generation Networking*, ser. Lecture Notes in Computer Science, S. Andreev, S. Balandin, and Y. Koucheryavy, Eds. Springer Berlin Heidelberg, Jan. 2012, no. 7469, pp. 347–358. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-32686-8˙32

[10] M. Sulovic, D. Raca, M. Hadzialic, and N. Hadziahmetovic, "Dynamic codec selection algorithm for VoIP," in *Proc. 6th International Conference on Digital Telecommunications (ICDT 2011)*, 2011, pp. 74–79.

[11] N. Costa and M. S. Nunes, "Adaptive Quality of Service in Voice over IP Communications," in *Proc. 5th International Conference on Networking and Services*, ser. ICNS '09. Washington, DC, USA: IEEE Computer Society, 2009, p. 1924. [Online]. Available: http://dx.doi.org/10.1109/ICNS.2009.33

[12] L. Sun and E. C. Ifeachor, "Voice quality prediction models and their application in VoIP networks," *IEEE Transactions on Multimedia*, vol. 8, no. 4, p. 809820, Aug. 2006. [Online]. Available: http://dx.doi.org/10.1109/TMM.2006.876279

[13] ITU-T, "Transmission Impairments due to Speech Processing," Tech. Rep. Recommendation G.113, 2001.

[14] L. Sun, "Speech Quality Prediction for Voice over Internet Protocol Networks," PhD Thesis, University of Plymouth, 2004.

[15] M. Carbone and L. Rizzo, "Dummynet revisited," *SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 2, p. 1220, Apr. 2010. [Online]. Available: http://doi.acm.org/10.1145/1764873.1764876

[16] "JITSI," http://www.jitsi.org, 2012. [Online]. Available: http://www.jitsi.org

[17] ITU-T, "P. 863,perceptual objective listening quality assessment (POLQA)," *Int. Telecomm. Union, Geneva*, 2011.