A No-reference Voice Quality Estimation Method for Opus-based VoIP Services

Péter Orosz, Tamás Skopkó, Zoltán Nagy, and Tamás Lukovics
Faculty of Informatics
University of Debrecen
Debrecen, Hungary
e-mail: oroszp@unideb.hu

Abstract— **By the emergence of real time media applications, correlating users' satisfaction with measured service quality is a constant challenge. Accordingly, this area was under intensive research in the last decade. Finding the correlation among Quality of Experience (QoE) for voice, measured Quality of Service (QoS) parameters in the network, and objective voice performance metrics is a key task. Most of the voice metrics use the reference content to compare the quality of the received stream. In contrast, this paper introduces a mathematical low-complexity, no-reference method that performs real-time estimation of QoE for Opus-based voice services. To determine the estimator function, we performed combined (subjective and objective) assessments to build a reference data set of 3-tuples of MOS, jitter and loss values. Applying polynomial regression, we used the reference data set to search a low-degree two-variable polynomial to hash objective QoS metrics (jitter and loss) to the subjective MOS score of the service quality. In the final phase of our investigation, we were evaluated the performance of the polynomials with a set of four audio clips.**

Keywords- Opus codec; real-time voice; QoE; QoE estimation; QoS-QoE correlation.

I. Introduction

Real-time audio services are sensitive to the timing and transmission performance of the network infrastructure. Since voice communication is interactive, low latency (≤150 ms) is a crucial requirement for an acceptable level of user experience. While monitoring of these performance metrics is a common solution, especially in dedicated infrastructures (like mobile networks), none of these parameters alone shows direct correlation with the perceptual quality (QoE). Therefore, service providers also apply subjective evaluation methods occasionally. Mean Opinion Score (MOS) and similar simplified scale, such as Absolute Category Rating (ACR) are generally used. Since subjective evaluation is time consuming and circumstantial, service providers mainly use these tools for a short but intensive period. A further drawback of an evaluation based on user feedback is the static evaluation of a whole user session (i.e., at the end of the conversation). Though the method can be extended by localizing the time moments of errors that affect QoE dramatically (e.g., by pressing a specific key straight after a disturbing glitch or noise), still the most efficient way for a service provider to assess perceptual quality of a service would be a dynamic, real-time analysis method, which estimates user experience. To accomplish this aim, the relationship between objective metrics (QoS) and perceptual quality should be investigated. Since the measurements should be run on network nodes aggregating intensive traffic (e.g., routers and switches), the method has to enable a hardware-accelerated implementation. Therefore, the complexity of the algorithm should be kept to a relatively low level.

We already investigated the performance of the evolved Opus audio codec in real network conditions [1]. In the related research we subjectively evaluated the voice content using the original, source audio content as reference. During the transmission we used pre-determined QoS parameters in a full-controlled laboratory network. We pointed out that the investigated Opus codec provides very good voice reproduction with a wide range of network parameters. We also found that there is a close-to-linear relation between MOS and packet loss rate. The experiments motivated us to unfold deeper relations between QoS and QoE. We set up a subsequent investigation consisting of three phases. The first phase is built upon the subjective evaluation technique discussed in our previous paper. In the second phase we continue to focus on VoIP services provided on managed networks instead of Over-the-Top (OTT) and we use statistical analysis methods, i.e., polynomial regression and correlation analysis on the reference data set of the first phase to describe the relation between the QoS parameters and QoE. In the third phase, we introduce the optimal coefficients for the polynomials that can be used to estimate the QoE. We are also evaluating the generality of the method by performing subjective as well as objective assessments with four independent voice clips.

In Section II, we summarize researches and developments relevant to the quality of the Opus audio codec. Section III shows some important details of voice transmission that have to be considered for the quality assessment of a voice codec. In Section IV, we present our evaluation method and its associated measurement configuration among some important details about VoIP transmission. In Section V, the investigation of the suitable statistical description is introduced as well as its mathematical background. We also validate our evaluation method by further subjective assessments detailed in Section IV. In the closing Section VII, we summarize our observations, the introduced evaluation method and sketch further research aspects.

II. Related Works

A. Raake summarized the elements of VoIP speech quality evaluation and assessments [2].

A. Ramö and H. Toukomaa evaluated Opus' MDCT and LP modes by subjective listening tests and compared them with 3GPP AMR, AMR-WB and ITU-T G.718B, and they stated Opus to be a good alternative for the aforementioned codecs [3]. The paper of C. Hoene et al. includes different listening tests and compares the codec to Speex (both NB and WB), iLBC, G.722.1, G.722.1C, AMR-NB and AMR-WB [4]. They conclude that Opus performs better, though at lower rates, however AMRNB and AMR-WB still outperform the new Opus codec. Valin *et al.* present further improvements in the Opus encoder that help to minimize the impact of coding artifacts [5].

The International Telecommunication Union (ITU) has its own recommendation for standardized evaluation of speech quality: after many years of development (superseding PEAQ, PSQM, PESQ and PESQ-WB algorithms), POLQA (ITU-T P.863) is able to evaluate speech sampled up to 14 kHz using the MOS metrics [6].

Neves *et al* proposed a No Reference model for monitoring VoIP QoE based on the E-Model [7]. The method was proven to be a class C2 conformance in terms of subjective MOS scoring.

S. Cardeal *et al* introduced ArQoS, a probing system that integrates network performance monitoring methods as well as QoE assessment methods in a telecommunication infrastructure [8].

W. Cherif *et al* presented a non-intrusive QoE prediction method called A_PSQA based on a Random Neural Network (RNN) approach [9].

L. Fei *et al* discuss packet delay and bandwidth as main factors affecting QoE and introduce a carrier scheduling scheme for LTE based on their research [10]. They have significant QoE improvements in their simulation results.

Jelassi S. *et al* made an extensive survey on objective and subjective QoE assessment methods [11].

V. Aggarwal *et al* employs machine learning technique to passive QoS measurement information to predict VoIP QoE with at least 80% accuracy [12].

In our previous paper, we summarized objective voice quality evaluation methods, as well as subjective approaches. We examined the correlation of QoS metrics packet loss and jitter with subjective evaluations for VoIP audio transmission [1]. We used pre-defined QoS: loss and jitter values were iterated through a selected range. We streamed real human voice in emulated WAN environment based on the specific QoS parameters. The results were evaluated by a number of volunteers. We ran the experiment with both Opus and its predecessor, the Speex codec. Opus performed more uniformly on a wide range of QoS parameters than the Speex codec. Opus also showed a close-to-linear correlation with packet loss rate. These results opened the way for the construction of an estimator function.

III. Background

Since interactive real-time audio streaming is very sensitive to timing parameters, it is very common to use a specific protocol for media transmission. UDP-based Real-Time Transport Protocol provides the necessary parameters for time-sensitive data exchange [13]. Its most important metadata are the sequence number and the timestamp. The former provides a way of determining packet losses, reorders and duplications. The protocol assigns a per-application play-out buffer, where the packets are sorted using the timestamps and sequence numbers and accordingly duplicated packets can also be filtered out. To handle timing, the time-related information is also used. Since the goal is to ensure a continuously decoded media stream, this layer avoids both buffer over and underruns. A moving time window will specify what packets are received on time. Too early and too late packets will be dropped.

For optimal operation, it is necessary to have a constantly available guaranteed bandwidth. This can be assured most easily by using a constant bitrate (CBR), when packets of roughly the same size are transmitted with fixed rate. Most real-time media services still use CBR operation mode to effectively manage allocation of resources, and also, scheduling mechanisms can better handle a constant packet rate.

During an RTP/UDP real-time audio transmission using the Opus audio codec, the audio frames are all the same type, in contrast to the video frames of the H.264 codec. There are no key-frames that store data for a full video frame as a reference for several subsequent B-type and P-type frames. An audio frame stores audio samples for a fixed period of time (10-40 ms, typically). Thus, the effect of a lost packet always produces a gap in the voice stream. In contrast, jitter itself does not necessarily lead to data loss with appropriate buffering at the receiver side. We are not focusing on packet reordering and duplication since RTP handles these anomalies transparently. As far as sequence numbering and playout buffering allows these metrics do not cause degradation of quality for the end user.

Research of quality evaluation methods is not a novel field. However, creating objective methods that correlate well with subjective assessments is still a challenge. Most objective methods are full reference (FR), as the original media content is required for them to work. Since the source material is rarely available in the real world, there is a demand for solutions operating without the original content or only with some trace of it. They are no-reference (NR) or reduced reference (RR) methods. If we want to develop such a method for real-time QoE prediction, low calculation overhead is a further requirement, since embedded systems and mobile devices have limited computing power and resources.

IV. First phase of the investigation: reference assessments

As a basis of our research, we planned an environment providing laboratory conditions for the evaluation that can be repeated many times and reproduced by other groups. Since network anomalies such as packet loss and jitter occur in real networks accidentally, we decided to use a network emulation tool that can introduce these types of error in a statistical way.

A. Source media content

We selected one high quality, pre-recorded voice clip containing an easy to understand male voice in the native language of the volunteers who participated in the evaluation. The clip was taken from an audio book, stored in standard CDDA resolution (44.1 kHz, 16 bit, stereo) and was resampled to 48 kHz, 16 bit, mono at the source of the stream. Its length was cut to 60 seconds. This length allows the listener to pick up the pace, and also enables a statistics-based network emulator to reach the steady state.

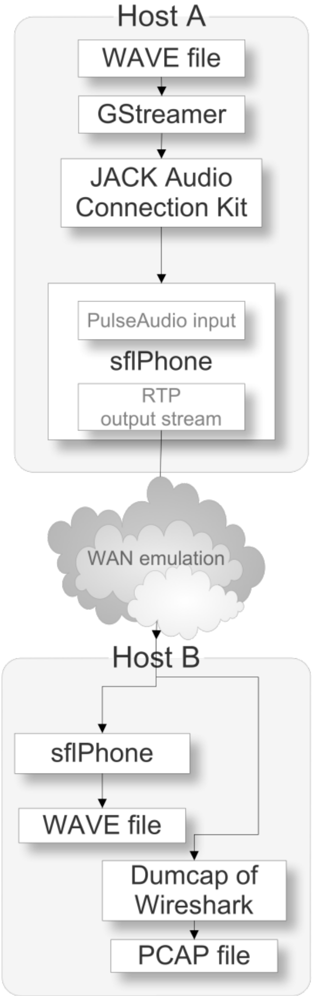


Figure 1. The measurement setup: audio is fed into the VoIP client on Host A and is transported through RTP to the other client on Host B

B. Measurement setup

An emulated WAN connection including two communication endpoints was constructed for the assessments (see Fig. 1).

Endpoints feature generic multi-core x64-based architectures and they were equipped with Intel PRO/1000 NICs. Fedora Core 18 was installed to both hosts with unmodified Linux 3.8.1-x kernel (with a jiffy setting of 1000 Hz). We have chosen version 1.2.2 of the *sflPhone* VoIP application, since it natively supports the Opus codec and our initial traffic measurements confirmed the expected QoS performance (i.e., packet rate, uniform distribution of inter-arrival times and packet sizes) [14].

The source voice clip was injected into the input of the softphone on Host A. *JACK Audio Connection Kit* is a general audio tool and is able to connect audio inputs and outputs of different applications and audio devices [15]. The current version of sflPhone can accept *ALSA* and *PulseAudio* datastream at its input. *PulseAudio* was selected since it can be connected with JACK. Since it has a native output plugin (sink) for *JACK*, the audio clip was fed into *JACK* from an uncompressed PCM WAVE file with the *GStreamer* application [16]. We carefully configured the applications not to perform unnecessary audio sample rate conversion throughout the digital audio path. The sflPhone application on Host B was configured to save the audio data into an uncompressed PCM WAVE file for further QoE assessments. Noise reduction and echo cancellation features were turned off. During the measurement, we used the *Netem* Linux kernel module, which was configured symmetrically on both directly connected interfaces to emulate a WAN connection and produced various network anomalies that affect QoS (i.e., packet loss and variation of network delay) [17].

Since we don't have to distinguish between different types of media packets, like in the case of a video stream, where packets contain different type of video frames (and key-frames can be transmitted in multiple consecutive packets), Netem’s Layer-2 emulation technique is suitable for our measurement goals. During the iterated measurements, we stored both the WAVE file from the receiving softphone and the network traffic trace containing the received RTP stream [18]. ITU recommends delay to be kept under 150 ms for an acceptable interactive service and we wanted to emulate a generic WAN connection therefore we have chosen 100 ms of delay in each direction. The codec was measured with a series of parameters (Table 1). Netem network parameters were iterated using the following scheme:

TABLE I. PRE-DEFINED QOS VALUES FOR NETWORK EMULATION

Netem parameters	Set of values
jitter	0-20 ms in 1 ms steps
packet loss	0-40% in 1% steps

The softphone client generated 100 RTP packets per second. Packet size varied between 40 to 159 bytes (see Fig. 2).The codec generated an audio frame in each 8 ms. Its operation mode was constrained VBR (CVBR), which forces the encoder to operate at an average nominal bitrate. In our case this was 64 kbps and the variation of the inter-arrival time was in the range of ±500 µs.

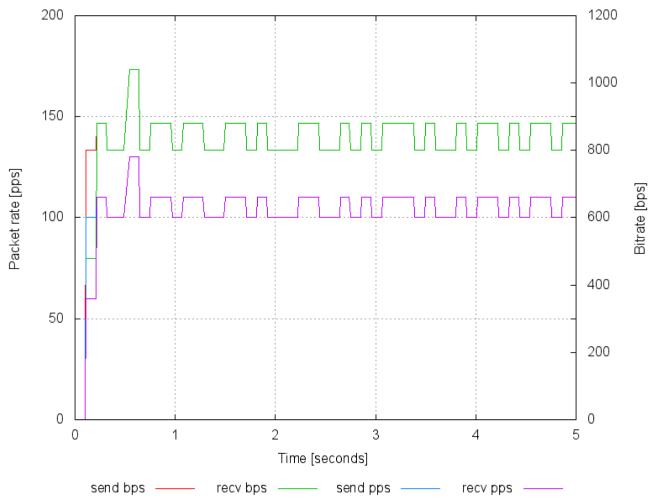


Figure 2. Packet rate and bandwidth during the voice transfer

C. *Evaluating the content*

Since we worked with a wide range of parameters, the iterated measurements resulted in more than 100 audio clips. Choosing finer resolution of these parameters would increase the number of clips even further, that could make the the evaluation significantly complicated by the monotony of the huge set of samples. As a reference for the evaluation, an initial measurement with zero jitter and no packet loss were run. The one-minute audio files got subjective QoE values by 15 volunteers who were in a calm and peaceful home environment. The evaluators were not VoIP professionals, had no deep knowledge of VoIP services and audio codecs.. Everyone did the tests at her own pace, with a comfortable timing, to minimize monotony. Each assessment session was started by listening to the reference voice clip. In the QoE analysis, the average of the MOS ratings was assigned to all audio files.

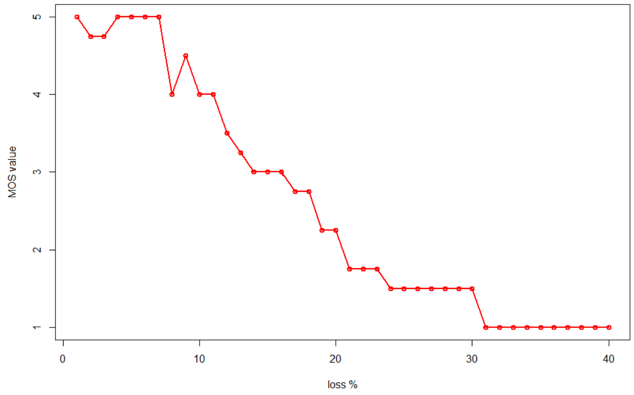


Figure 3. Correlation between packet loss ratio and subjective quality of experience expressed in MOS

We experienced near-linear relationship between quality degradation and the amount of lost packets (see Fig. 3).

Opus over RTP tolerates packet losses well even up to 30%. It can be a significant advantage when used with wireless access. Lower subjective quality right after the reference clip was a side-effect of the subjective assessment: evaluators got used to the good quality and even a small increment in the packet loss rate caused lower MOS ratings.

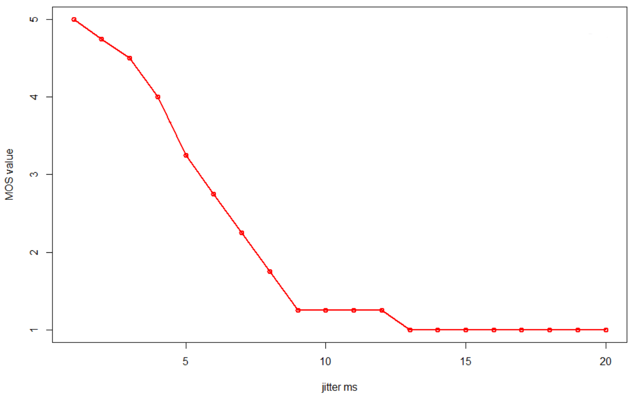


Figure 4. Correlation between jitter and subjective quality of experience expressed in MOS

The jitter-sensitivity of the transmission was also consistent but was further from the linear relation (see Fig. 4). The reason is that jitter can cause packet loss in the RTP layer since too early (generally caused by bursty packet forwarding) and too late packets were discarded by the transport protocol. Opus tolerates well a small amount of jitter, typically below 4 ms. However, the result of jitter elimination depends on the size of the receiver buffer. Greater jitter may cause significant degradation of perceived quality. In our investigation, this quality drop was experienced up to 9 ms. In contrast, in the range of 10 to 12 ms of jitter, an acceptable steady state was perceptible.

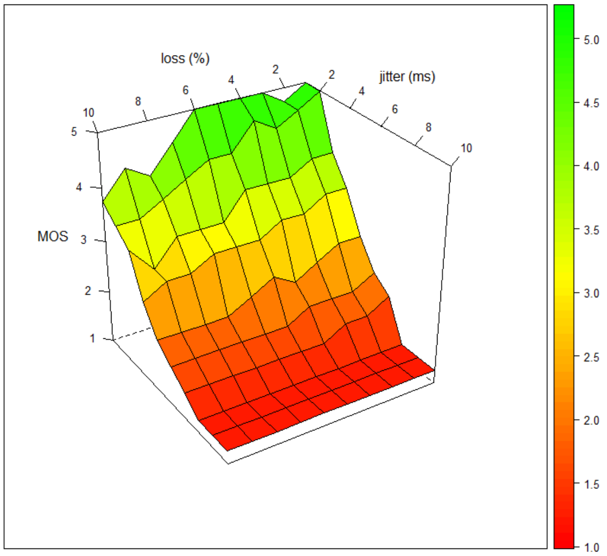


Figure 5. MOS values under mixed network conditions

Sensitivity to combined effect of the aforementioned network anomalies confirmed the experiences of the simple scenarios. Opus was performing uniformly in term of packet loss. Contrarily, the effect of jitter made the relationship more complex as seen on Fig 5. However, the surface mapped to the MOS values of the combined measurements did not result in a very complex pattern and therefore it made us motivated to undertake a deeper investigation of the correlation between QoS and QoE.

V. Second phase of the investigation: statistical analysis

A. *Polynomial regression*

The result of the first phase, i.e., assessed packet loss, jitter and MOS values, are presented in a 3-axis graph. We used polynomial regression to determine a two-variable polynomial of a matched surface. This is calculated using the class of functions (1).

$$F = \{p_n(x) = a_0 + a_1 x + \cdots + a_n x^n\} \qquad (1)$$

A matched polynomial regression is the result class of polynomials of (2).

$$M(\eta - f^*(\xi))^2 = \min_{\forall f \in F} M(\eta - f(\xi))^2 \qquad (2)$$

Coefficients are determined by solving the linear simultaneous equations in (3).

$$\begin{pmatrix} 1 & M\xi & \cdots & M\xi^n \\ M\xi & M\xi^2 & \cdots & M\xi^{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ M\xi^i & M\xi^{i+1} & \cdots & M\xi^{i+n} \\ \vdots & \vdots & \ddots & \vdots \\ M\xi^n & M\xi^{n+1} & \cdots & M\xi^{2n} \end{pmatrix} * \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_i \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} M\eta \\ M\eta\xi \\ \vdots \\ M\eta\xi^i \\ \vdots \\ M\eta\xi^n \end{pmatrix} \qquad (3)$$

The polynomial regression was calculated using the *Matlab* function *poly*{ij}. The surface model can be tuned by the degree of input parameters. In a polynomial regression using second degree for the first variable, and first degree for the second one (poly21) can be described as (4).

$$Z = p_{00} + p_{10}x + p_{01}y + p_{20}x^2 + p_{11}xy \qquad (4)$$

where *x* denotes the amount of packet loss and *y* specifies the amount of jitter.

B. *Surface inspection using SSE*

Sum of Squares due to Error (SSE) determines the standard deviation between a set of points and the matched surface and is calculated using (5).

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \qquad (5)$$

The surface is less accurate match for the specified set of points if the SSE value falls further from zero.

C. *Matched surfaces – poly11*

Figure 6 plots the matched surface for this purely linear model, where the surface is determined by (6) and its coefficients are summarized in Table II.

$$f(x,y) = p_{00} + p_{10}x + p_{01}y \qquad (6)$$

TABLE II. Coefficients for (6)

p_{00}	5.151
p_{10}	-0.07513
p_{01}	-0.4111

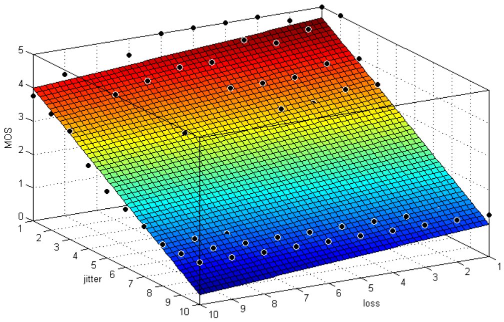


Figure 6. The surface determined by (6) and the MOS values from subjective evaluation in Phase 1

Goodness of fit with SSE is 16.24, and thus the MOS scores fall far from the surface displayed on Fig. 6.

D. *Matched surfaces – poly12*

If we use a second degree polynomial for the jitter (y) variable, the describing function will be (7). The coefficients are specified in Table III.

$$f(x,y) = p_{00} + p_{10}x + p_{01}y + p_{02}y^2 + p_{11}xy \qquad (7)$$

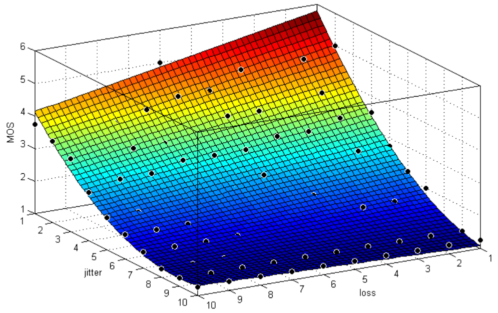


Figure 7. The surface determined by (7) and the MOS values from subjective evaluation in Phase 1

TABLE III. COEFFICIENTS FOR (7)

p_{00}	6.985
p_{10}	-0.2052
p_{01}	-1.063
p_{11}	0.02292
p_{02}	0.04696

Goodness of fit using SSE is 2.336. Introducing a second degree jitter variable resulted in a significant improvement as shown on Fig. 7.

E. *Matched surfaces – poly22*

In this case we are using a second degree function for the loss parameter also. The corresponding polynomial is (8).

$$f(x,y) = p_{00} + p_{10}x + p_{01}y + p_{20}x^2 + p_{11}xy + p_{02}y^2 \qquad (8)$$

TABLE IV. COEFFICIENTS FOR (8)

p_{00}	7.067
p_{10}	-0.2382
p_{01}	-1.07
p_{20}	0.003474
p_{11}	0.02217
p_{02}	0.04787

The SSE goodness of fit is 2.289 and means only a slightly closer match was achieved by introducing a more complex calculation. However, increasing the degree of the loss variable, which shows close-to-linear relationship to the MOS values, did not imply a significantly lower SSE value.

F. *Matched surfaces – poly13*

We also inspected the effect of a higher degree polynomial for jitter. In this case the describing function is (9) and its coefficients are specified in Table V.

$$f(x,y) = p_{00} + p_{10}x + p_{01}y + p_{11}xy + p_{02}y^2 + p_{12}xy^2 + p_{13}y^3 \qquad (9)$$

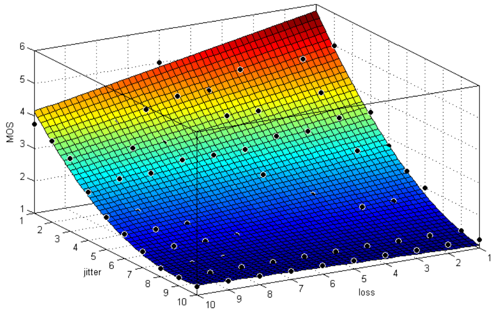


Figure 8. The surface determined by (9) and the MOS values from subjective evaluation in Phase 1

TABLE V. COEFFICIENTS FOR (9)

p_{00}	6.728
p_{10}	-0.241
p_{01}	-0.7999
p_{11}	0.04121
p_{02}	-0.01849
p_{12}	-0.001633
p_{03}	0.004354

For the surface demonstrated on Fig. 8, the SSE goodness of fit value is 1.69. It means a better matching surface but also adds a significant amount of complexity to the calculation.

G. *More matched surfaces*

We calculated a number of matched surfaces of higher degree functions. Fig. 9 shows their matching efficiency using SSE goodness of fit versus the degree of functions. The list of calculated polynomials, their number of coefficients and their error values are listed in Table VI and Table VII. Some questions are raised at this point raised: Do higher degree polynomials lead to better correlation results? What is the optimal complexity-error trade-off for a real-time NR method?

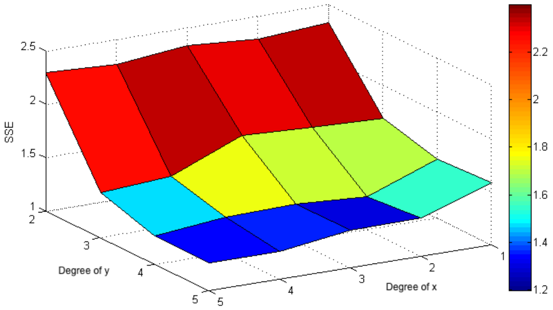


Figure 9. SSE goodness of fit for surface matching using polynomial regression and second and higher degree functions for packet loss and jitter parameters

TABLE VI. NUMBER OF COEFFICIENTS FOR DIFFERENT DEGREES OF LOSS AND JITTER VARIABLES

Degree of x	**Degree of y**				
	1	**2**	**3**	**4**	**5**
1	3	5	7	9	11
2	5	6	9	12	15
3	7	9	10	14	18
4	9	12	14	14	20
5	11	15	18	20	21

TABLE VII. SSE GOODNESS OF FIT VALUES FOR DIFFERENT DEGREES OF LOSS AND JITTER VARIABLES

Degree of x	Degree of y				
	1	2	3	4	5
1	16.2389	2.3363	1.6905	1.5559	1.5829
2	14.1052	2.2895	1.7138	1.3088	1.362
3	13.8641	2.3346	1.7445	1.353	1.3542
4	13.9947	2.2665	1.4665	1.3375	1.2664
5	13.7948	2.3008	1.4275	1.2668	1.2674

VI. THIRD PHASE OF THE INVESTIGATION: VALIDATION OF THE METHOD

The primary aim of this last phase is to verify the generality of the correlation calculation based on the coefficients and polynomials determined in Phase 2. The measurement environment described in Phase 1 remained unchanged.

The final goal of investigation is to construct a low degree evaluation function that inputs measured objective values and outputs the estimated value of QoE on the MOS scale.

A. *Source media contents*

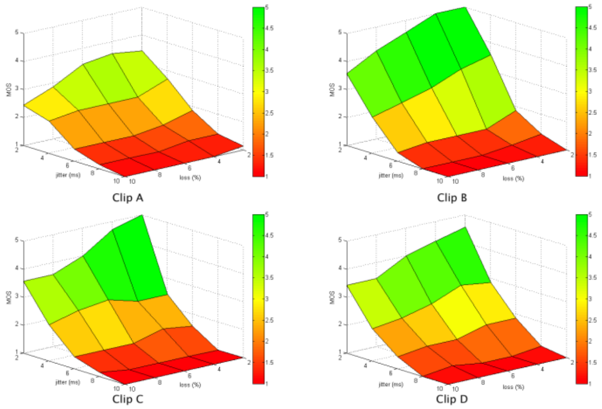
We selected four voice clips from different sources. A female voice was present in two of the clips (clips A and B), and a male voice in the other two (clips C and D). Only native language was used. All clips were taken from audio books, being stored in standard CDDA resolution (44.1 kHz, 16 bit, stereo), they were resampled to 48 kHz, 16 bit, mono at the source of the stream. All the audio clips were 60 seconds long.

B. *Evaluating the goodness of QoS-QoE mapping*

The evaluations were executed in the same way as presented in Phase 1. In this phase, 10 volunteers were involved to the subjective assessment. Clip A containing a female voice got lower scores on the MOS scale than expected. This may be caused by the intonation of the person's voice. We compared the uncompressed source material and the 64 kbps Opus-encoded audio without network errors. While no disturbing artifacts were experienced in the source material, the Opus codec made sometimes the woman's voice raised in volume and this may be the root cause of the lower MOS scores.

We made 3D surfaces of MOS scores in the inspected QoS parameter range for all of the audio clips. The surfaces are presented on Fig. 10.

There is a quasi-linear correlation between MOS value ranging from 2 to 4 for Clip A. Over 4 ms of jitter, the experienced quality started to degrade. However, increasing packet loss did not degrade the scores dramatically.



Clip A

Clip B

Clip C

Clip D

Figure 10. MOS values for different packet loss and jitter parameters

In Clip B also containing a female voice, the coding artifacts were not so significant. This resulted in higher MOS scores in general. The close-to-linear relation can be observed in the case of packet loss. Above the MOS value of 2, this correlation with the jitter is also present.

The correlation between MOS and jitter is still first-degree above MOS value 2 in Clip C with the male voice, while correlation with packet loss got closer to a second-degree function.

The surface of Clip D is similar to Clip C but got lower scores in general. This may also be caused by the source material.

Based on the surface analysis, we can conclude that there is a quasi-linear correlation between packet loss and MOS scores in the range of 2 to 4. This may enable good estimation of perceived quality. We assume that the most interesting range of the MOS scale is from 3 to 4. This range is where the customer will decide between continuing using the service and giving up. Around the MOS value of 4, there is a long-term acceptance-level of the service quality, while below 3, perceptive quality may become poor with annoying artifacts.

We also experienced the Opus codec being very sensitive to the voice intonation in some cases. All of the source materials were high quality, but the encoding process at this bitrate introduced some artifacts that affected the experienced quality very dramatically when packet losses or higher jitter occurs.

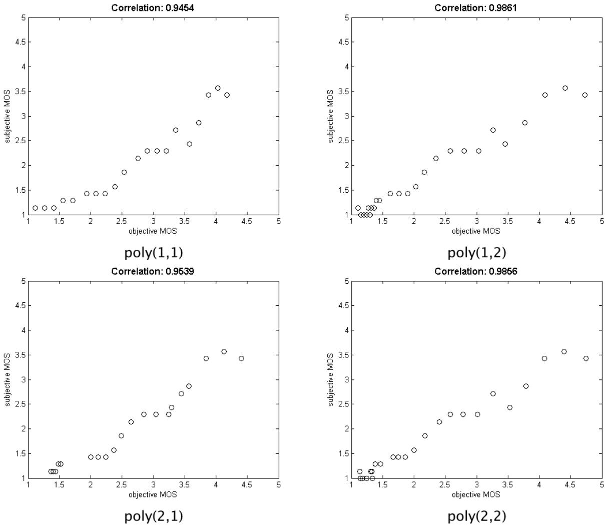
C. *The results – Polynomial regression*

Based on the conclusion of Phase 2, we assume that constructing an estimator function using first or second degree function is possible. It would be beneficial for a hardware implementation of such an estimator function to use low calculation complexity

The correlations of the objective and subjective MOS series, which is calculated from (1) are presented on Fig.11 to 14.

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_x \sigma_y} = \frac{E[(X-\mu_x)(Y-\mu_Y)]}{\sigma_x \sigma_y} \qquad (10)$$

where μ is the expected value of the random variable and a σ is its standard deviation.

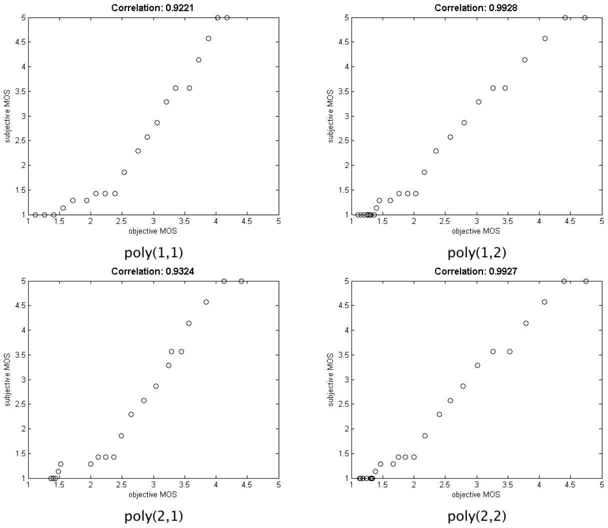


poly(1,1) poly(1,2)

poly(2,1) poly(2,2)

Figure 11. Correlation using different degree polynomials for Clip A

Poly(x,y) is the regression function, where packet loss is represented by *x*, and jitter is described by *y*.



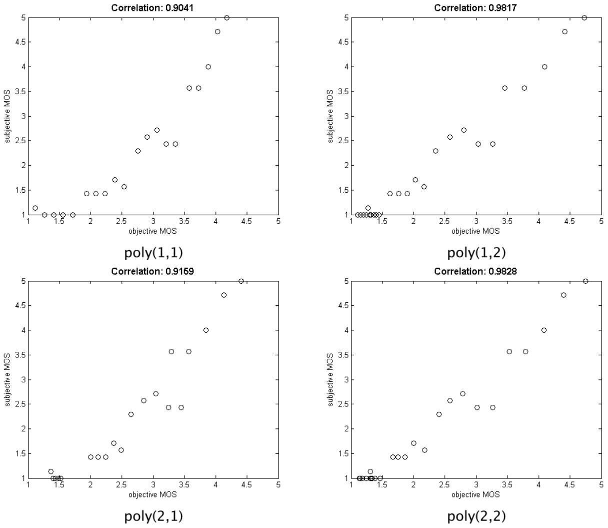
poly(1,1) poly(1,2)

poly(2,1) poly(2,2)

Figure 12. Correlation using different degree polynomials for Clip B

In the case of Clip A depicted on Fig. 11, the purely linear functions for both QoS parameters result in good correlation only within the MOS score range of 2 to 2.5. When a second degree function is used for packet loss, correlation only increased a small amount, with mainly the bottom range of the MOS scale showing better correlation. Using a second degree function for jitter, the correlation is better not only for lower MOS scores but in the whole range. If both functions are second degree, the level of correlation is not changed significantly.

Fig. 12 presents the analysis for Clip B. Linear functions only provide 92.21% correlation, and using a second degree packet loss function does not improve its accuracy perceptively. Second degree variable for jitter shows significant improvement in correlation, reaching 99%. In the MOS range from 2 to 4 it shows quasi-linear behavior in each case.

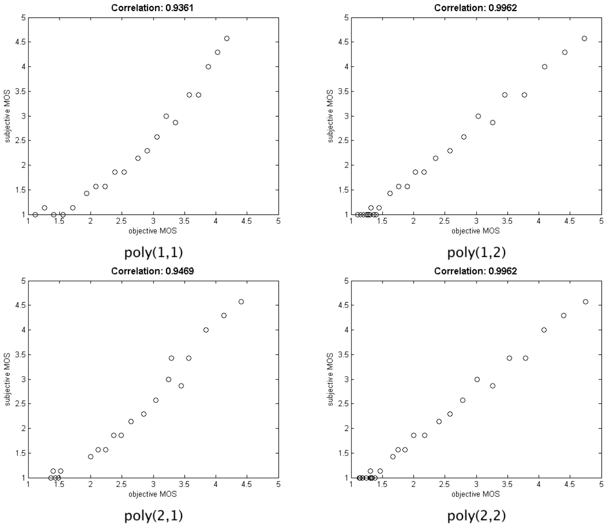


poly(1,1) poly(1,2)

poly(2,1) poly(2,2)

Figure 13. Correlation using different degree polynomials for Clip C

The male voice in Clip C led to similar results in general, but the uniformity in the aforementioned range is not as strong. The benefit of a second degree jitter variable in the function was also confirmed by this evaluation. This is demonstrated in Fig. 13.



poly(1,1) poly(1,2)

poly(2,1) poly(2,2)

Figure 14. Correlation using different degree polynomials for Clip D

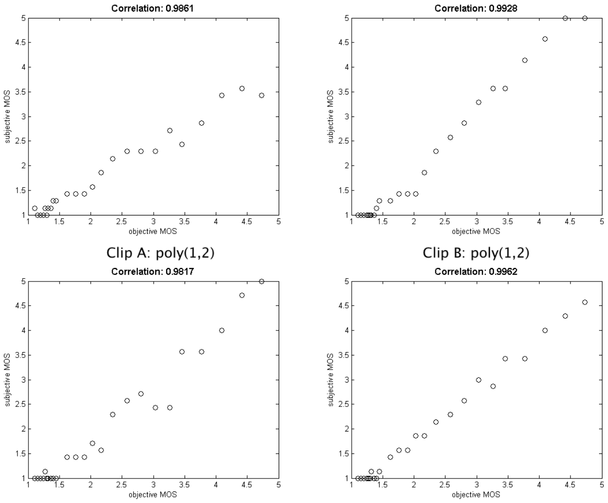
Assessment of Clip D plotted in Fig. 14 resulted in rather uniform results almost through the whole MOS range.

We calculated the overall correlation of first and second degree polynomials. The correlation values are summarized in Table VIII.

TABLE VIII. CORRELATION VALUES FOR FIRST AND SECOND DEGREE POLYNOMIAL REGRESSION ANALYSIS. FIRST PARAMETER: DEGREE FOR PACKET LOSS, SECOND PARAMETER: DEGREE FOR JITTER

	Clip A	Clip B	Clip C	Clip D
poly(1,1)	0.9454	0.9221	0.9041	0.9361
poly(1,2)	0.9861	0.9928	0.9817	0.9962
poly(2,1)	0.9539	0.9324	0.9159	0.9469
poly(2,2)	0.9856	0.9927	0.9828	0.9962

Based on these calculations we decided to use *poly*(1,2) as an optimal choice for the prediction, since increasing the degree of the function would not improve the performance significantly. Fig. 15 plots *poly*(1,2) points for all the audio clips.



Clip A: poly(1,2)

Clip B: poly(1,2)

Clip C: poly(1,2)

Clip D: poly(1,2)

Figure 15. Correlation between subjective and estimated MOS values using first degree variable for packet loss, second degree variable for jitter in the estimator function

VII. CONCLUSION AND FUTURE WORK

We presented a three-phase investigation with the aim of constructing an NR-type objective method for estimating VoIP QoE based on the Opus audio codec. In the first phase of the investigation, we performed subjective QoE evaluations of voice content streamed through an emulated network environment with pre-defined QoS behavior. In the second phase, we determined the degree of relation between QoS parameters and QoE using polynomial regression. We also determined the coefficients for the specific degree of polynomials in the correlation analysis. In the third phase, we repeated the evaluation technique using four independent contents. We calculated the correlation for first and second degree functions in the polynomial regression. We concluded that using first degree functions a correlation of 90% is achievable in view of QoS parameters packet loss and jitter. If a second degree variable is used for the jitter, a correlation above 98% was observed.

Using low degree polynomials, our new method can be the basis of a hardware-accelerated QoE estimation method for predicting perceptual quality in real-time by measuring QoS parameters (packet loss and jitter) and estimating the MOS score on-the-fly without using the original content. We have chosen the Opus audio codec as a reference but our model can also be applied to other audio codecs with a behavior similar to Opus.

ACKNOWLEDGMENT

The research of Tamás Skopkó, Zoltán Nagy and Tamás Lukovics was supported by the TÁMOP 4.2.2.C-11/1/KONV-2012-0001 project. The project was implemented through the New Széchenyi Plan, co-financed by the European Social Fund.

The research of Péter Orosz was supported by the European Union and the State of Hungary, co-financed by the European Social Fund in the framework of TÁMOP 4.2.4.A/2-11-1-2012-0001 'National Excellence Program'.

REFERENCES

[1] P. Orosz, T. Skopkó, Z. Nagy, and T. Lukovics, "Performance analysis of the Opus codec in VoIP environment using QoE evaluation," ICSNC 2013, The Eighth International Conference on Systems and Networks Communications, IARIA, pp. 89-93, 2013.

[2] A. Raake, "Speech quality of VoIP: Assessment and prediction," John Wiley & Sons, 2007.

[3] A. Ramö, and H. Toukomaa, "Voice quality characterization of IETF Opus codec," INTERSPEECH, ISCA, pp. 2541-2544, 2011.

[4] C. Hoene, J. M. Valin, K. Vos, and J. Skoglund, "Summary of Opus listening test results draft-valin-codec-results-03," November 2013.

[5] J. M. Valin, G. Maxwell, TB. Terriberry, and K. Vos, "High-Quality, low-delay music coding in the Opus codec," 135th AES Convention, AES p. 8942, October 2013.

[6] ITU-T P.863: Perceptual objective listening quality assessment, January 2011.

[7] F. Neves, S. Cardeal, S. Soares, P. Assunção, F. Tavares, "Quality model for monitoring QoE in VoIP services," International Conference on Computer as a Tool (EUROCON), IEEE, pp.1-4, April 2011, doi:10.1109/EUROCON.2011.5929300

[8] S. Cardeal, F. Neves, S. Soares, F. Tavares, and P. Assunção, "ArQoS®: System to monitor QoS/QoE in VoIP," International Conference on Computer as a Tool (EUROCON), IEEE, pp. 1-2, April 2011, doi:10.1109/EUROCON.2011.5929310

[9] W. Cherif, A. Ksentini, D. Négru, and M. Sidibe, "A_PSQA: PESQ-like non-intrusive tool for QoE prediction in VoIP services," International Conference on Communications (ICC), IEEE, pp. 2124-2128, 2012, doi: 10.1109/ICC.2012.6364004

[10] F. Liu, W. Xiang, Y. Zhang, K. Zheng, and H. Zhao, "A novel QoE-based carrier scheduling scheme in LTE-Advanced

networks with multi-service," Vehicular Technology Conference (VTC Fall), IEEE, pp. 1-5, 2012, doi: 10.1109/VTCFall.2012.6398912

[11] S. Jelassi, G. Rubino, H. Melvin, H. Youssef, and G. Pujolle, "Quality of experience of VoIP service: A survey of assessment approaches and open issues," Communications Surveys & Tutorials, IEEE, pp. 491-513, 2012, doi: 10.1109/SURV.2011.120811.00063

[12] V. Aggarwal, E. Halepovic, J. Pang, S. Venkataraman, and H. Yan, "Prometheus: Toward Quality-of-Experience estimation for mobile apps from passive network measurements," ACM HotMobile'14, ACM, p. 18, February 2014, doi: 10.1145/2565585.2565600

[13] V. Jacobson, R. Frederick, S. Casner, and H. Schulzrinne, "RTP: A transport protocol for real-time applications," RFC 3550, July 2003.

[14] sflPhone, [Online]. Available from: http://sflphone.org/, 20.06.2014

[15] JACK Audio Connaction Kit, [Online]. Available from: http://jackaudio.org/, 20.05.2014

[16] GStreamer open source multimedia framework, plugin list, [Online]. Available from: http://gstreamer.freedesktop.org/documentation/plugins.html, 20.06.2014

[17] netem: Linux Networking Emulator, [Online]. Available from: http://www.linuxfoundation.org/collaborate/workgroups/networking/netem, 20.06.2014

[18] J. Spittka, K. Vos, and J. M. Valin, "RTP Payload for Opus Speech and Audio Codec draft-ietf-payload-rtp-opus-01," August 2013.