# Adaptive Rate Voice over IP Quality Management Algorithm

Eugene S. Myakotnykh
Centre for Quantifiable Quality of Service in Communication Systems (Q2S) [1],
Norwegian University of Science and Technology,
Trondheim, Norway
E-mail: esm@q2s.ntnu.no

Richard A. Thompson
Telecommunications and Networking Program,
Department of Information Sciences and Telecommunications, University of Pittsburgh,
Pittsburgh, USA
E-mail: thompson@mail.sis.pitt.edu

*Abstract*—**The quality of voice-over-IP communication relies significantly on the network that transports voice packets because this network does not usually guarantee available bandwidth, delay, and loss that are critical for real-time voice traffic. The solution proposed here is to manage a voice-over-IP stream dynamically, changing encoding parameters as needed to assure quality. The paper proposes an adaptive-rate control algorithm that establishes interaction between a VoIP sender and a receiver, and manages voice quality in real-time. Simulations demonstrate that the system provides better average communications quality than traditional fixed-rate VoIP.**

***Keywords-adaptive VoIP; E-model; Packetization; Speech quality; Voice-over-IP (VoIP)***

## I. INTRODUCTION

A packet-switched network does not provide reliable transport of real-time data: it does not guarantee available bandwidth, end-to-end delay and packet loss parameters, which are critical for real-time voice traffic. Most of the previous research in the VoIP quality has concentrated on networking issues of QoS management. Many different algorithms were developed to improve the transport of packetized voice traffic, including traffic classification (Differentiated Services technology [1, 2]), bandwidth reservation (Integrated Services architecture [3], Resource Reservation Protocol [4]), congestion avoidance, Multi-Protocol Label Switching technology [5], and others. These approaches use different techniques to decrease transmission delay and/or probability of voice packet congestion in the network and make the Internet more suitable for real-time traffic transmission. However, these methods often do not provide acceptable results nor do they solve the problem completely because (1) not all equipment and service providers support same QoS protocols and quality standards, and (2) the Internet is a dynamic media; the technologies often cannot react to changing network conditions and manage the quality of every communication session in real-time.

The alternative approach is to adaptively manage voice encoding parameters on the sender side depending on the network conditions. A proper choice of a voice payload size or compression may enhance the quality of VoIP because it can dynamically change a configuration of a VoIP system, so that the system matches a current state of the network. This approach proposes to adjust a voice stream to the network and change parameters of the stream in real-time, depending on the network state.

This paper proposes an adaptive quality management mechanism that changes speech encoding parameters on the sender side in real-time depending on network impairments. This area is in the early stage of its development and investigation of many questions related to VoIP quality measurement and management, dependencies between multiple parameters and the resulting quality, is necessary to develop intelligent and efficient adaptive VoIP codecs.

When designing an adaptive quality management algorithm, several questions must be answered:

(1) Which factor (or factors) should be used to make a decision that a change of certain speech encoding parameters is required or not required at a given moment of time?

(2) How the end-user speech encoding parameters (packet duration and compression) affect VoIP quality under different network conditions. What encoding parameters should be changed and how to do it? How often should such a decision be made (per talkspurt, periodically, etc.)?

(3) How should feedback from the receiver be sent to the sender side?

Although this paper speaks about adaptive VoIP quality management and uses a set of narrowband voice codecs for analysis and as an example, the result and the approach can potentially be extended to a wider set of narrowband codecs, to wideband encoding and to IP-based audio in general.

This article is an extended version of paper [6] and it is organized as follows: the next Section provides an overview of related research in the area of adaptive VoIP. Section III describes network scheme and assumptions used in our simulation studies. Section IV investigates the effect of speech encoding parameters (packet size and compression/encoding variation) on quality of VoIP communications; Section V describes decision-making parameters for the proposed algorithm; Section VI shows the actual adaptive voice quality management mechanism. The results of the simulation study are presented in Section VII. Conclusion is drawn in Section VIII.

[1] "Centre for Quantifiable Quality of Service in Communication Systems, Centre of Excellence" appointed by The Research Council of Norway, funded by the Research Council, NTNU and UNINETT. http://www.q2s.ntnu.no

## II. Related Work

### A. Adaptive Quality Management

A number of studies investigating the idea of adaptive voice quality management are available. Multiple papers (for example, Qiao et al. [7], Seo et al. [8], Matta et al. [9], and others) adopt the GSM/UMTS Adaptive Multi-Rate codec (AMR) [10] for the IP-network. The AMR codec was originally developed for wireless networks and the decision about adaptation of its encoding parameters is based on channel interference. The philosophy behind AMR is to lower a codec rate as the channel interference increases and thus enabling more error correction to be applied. Evidently, the process of adaptive quality management in the IP-network is different than that in wireless communications: there is no channel interference, there are IP packets instead of radio signals, and the threshold choice and management process will be different. The papers above present ideas of how to use the existing encoding scheme in the IP network.

A real-time change of speech encoding parameters can be achieved through variation of voice packet size or compression (encoding scheme). As the voice packet travels through the Internet, an overhead with control information is added to the voice payload. The size of overhead added to a voice packet is 40 bytes (the application layer Real-Time Transport Protocol (RTP) [11] header is 12 bytes; the transport layer UDP header is 8 bytes; and the IP header is 20 bytes), and it is significant compared to a typical voice payload size. If the G.711 codec [12] is taken as an example, the length of one 10-ms packet is 80 bytes. So, the RTP/UDP/IP overhead is 50% of the payload size and the total bandwidth required for the voice stream transmission is 96 kbps.

Changing end-user parameters may significantly affect bandwidth requirements per call and, as a result, its quality. For example, increasing a voice stream IP-rate will lead to increased quality, but the probability of quality degradation due to potential congestion may also increase because of higher channel capacity requirements. Several papers (for example, [13], [14], [15]) studied how changing of the encoding parameters affects VoIP communication quality.

Various parameters can be used to detect congestion in the network and make a decision about encoding parameters adaptation. For example, Bolot et al. [16] and Mohamed et al. [17] perform adaptive rate control based on packet loss statistics. The computational quality model called the E-model [18] is used in [9], [19] and [20]. Ngamwonwattana, [21] makes decision about codec rate adaptation based on moving average thresholds of delay and packet loss, and proposes sending control messages from the receiver "on demand".

The recent paper of F. Sabrina and J. Valin [22] describes an adaptive mechanism using Speex codec [23]. The authors propose a novel criterion to get feedback about the network condition and a mechanism for adjusting the encoding bit rate based on the feedback and on instantaneous speech properties.

### B. Voice-over-IP Quality Assessment

This paper uses several metrics to estimate quality of voice-over-IP communications. Voice quality measurement methodologies include subjective testing (involves human subjects and considered as the ultimate way of quality evaluation), and objective techniques (signals comparison or computational methods). The leading subjective criterion of voice quality measurements is the Mean Opinion Score (MOS), which is defined in the ITU-T Recommendation P.800 [24]. MOS is a score of voice quality as perceived by a large number of people listening to speech over a communication system. This Recommendation uses the scale from one to five and the MOS of some session of voice transmission is the average estimate of voice quality rates assigned by individual listeners (1 – bad, 2 – poor, 3 – fair, 4 – good, 5 – excellent). Subjective tests are usually complex and time-consuming and cannot be used for real-time quality assessment. Objective mechanisms are needed for this purpose.

Here in the project, the real-time decision about parameters adaptation is based on two computed metrics: an instantaneous quality level, which is measured per talkspurt using the computational E-model [18], and, in addition to this, a change of integral perceptual quality level, which is estimated based on a model developed by AT&T [25]. The original version of the E-model is relatively complex. It includes about 20 input parameters representing various terminal, network and environmental quality impairment factors. In the narrowband voice-over-IP area, the simplified version of this model is often used with default values for all but a few parameters - delay and packet loss. The model computes speech quality rating on a 100-point scale as:

$$R = R_0 - I_d - I_{e\text{-}eff} \qquad (1)$$

R is the resulting indicator of voice quality; $R_0$ is the maximum score, achievable by codecs in the absence of loss and significant delay; $I_d$ is the impairment factor caused by end-to-end delay (a function of delay); $I_{e\text{-}eff}$ is the effective equipment impairment factor, which depends on used codec, and also on packet loss rate and effectiveness of packet loss concealment algorithms. The E-Model is based on the concept that "psychological factors on the psychological scale are additive" [18]. It does not imply that the factors are uncorrelated, but only that their contributions to the estimated impairments are independent and each impairment factor can be computed separately. Numerical characteristics for different codecs and more details about the model can be found in [18], [26], [27]. A similar model for wideband telephony is proposed in [28].

The E-model uses a special mapping function to establish a relationship between the 100-point R-scale and the traditional MOS scale (Equation 2).

$$MOS = 1 + 0.035R + R(R-60)(100-R) \cdot 7 \cdot 10^{-6} \qquad (2)$$

This model is not a perfect tool to calculate absolute quality level, but it is acceptable for measuring variations in quality.

Detailed investigations of this question by NTT Lab (Japan) [29] concluded that correlation coefficient of results provided by the E-model with subjective human testing results is about 80%. Since, goal of these algorithms is to achieve noticeable relative quality improvement, the E-model can be used to track changes in quality.

The E-model estimates average quality during a certain period of time. But quality, as perceived by the end-user, depends not only on significance of a quality distortion, but also when this distortion happened during a communication session. The effect, which reflects the way that a listener remembers call quality, is called "recency" effect. This effect implies that periods of low or high quality positioned at the end of a speech sample have a stronger influence on the overall session quality then when such periods are positioned in the beginning of the sample. In tests conducted by AT&T [25], a burst of noise was created and moved from the beginning to the end of a 60-second call. When the noise was at the start of the call, users reported a higher MOS score than when the noise was at the end of the call. Tests reported by France Telecom [30] showed a similar effect. The effect is believed to be due to the tendency for people to remember the most recent events or possibly due to auditory memory, which typically decays over a 15-30 second interval [30].

Further discussion about these parameters will be provided below. Using these metrics, the novel mechanism to adaptively manage speech encoding parameters is proposed.

## III. Simulation Design

### A. Network Model

The proposed adaptive speech quality management algorithm is tested using a simulation implemented in Matlab. The network topology for the simulation is shown in Fig. 1. A simplified scenario with a single place of potential congestion is analyzed. The congestion may be caused by bursty background traffic through the router. The link capacity is 5 Mbps, but, actually, this number is not very important. But, a portion of voice and data traffic in the network is important: a significant difference in VoIP quality is seen over the same network, with the same total (voice plus data) average traffic load, but with different voice-to-data traffic ratios. The presence of large data packets with bursty behavior creates "instability" in the voice transmission process, which causes additional delay variation (jitter) and, as a result, higher delay and/or packet loss. The propagation and network processing delay is assumed fixed at 50 ms (it may be noticeably higher in real networks). The bottleneck router uses FIFO queuing and the drop-tail mechanism in case of overflow. The router has a finite queue size (64 Kbyte), enough to keep packets in queue for about 100 milliseconds.

### B. Voice Encoding

This project uses parameters of three narrowband codecs for developing and simulating the proposed adaptive quality management scheme: the G.711 [12] (PCM encoding with no compression), the G.726 [31] (ADPCM encoding with 2:1 compression), and the G.729a [32] (CS-ACELP encoding with 8:1 compression). While codec may have different voice payload sizes, the discrete values 10 ms, 20 ms, 30 ms will be used. So, nine different sets of encoding parameters are used. These codecs and parameters are chosen for analysis because: 1) they provide a relatively high level of quality (higher than or close to the toll-grade); 2) their quantitative characteristics are known in terms of maximum encoded speech quality in the absence of packet loss and significant delay [18]; and 3) the difference in channel capacity consumption in these codecs is significant: for example, the G.711 codec with 10 ms voice payload size requires 96 kbps per stream (64 kbps of audio bit rate and 32 kbps to send the RTP/UDP/IP overhead); the 30 ms G.729 codec needs just 18.7 kbps channel (8 kbps audio rate and 10.7 kbps overhead bit rate). Selecting one of the nine sets of encoding parameters will be based on metrics calculated on the receiver side. These codecs are chosen as examples; similar adaptive mechanisms can be used with a different set of narrowband codecs, with some set of wideband codecs or with a combination of narrowband and wideband codecs.

### C. Call Characteristics

It is assumed that speech codecs with variable parameters (packet size, compression) are used. Delay and packet loss statistics is calculated on the receiver side. The E-model [18] is applied to get a quality metrics based on these parameters. The E-model parameters for the selected codecs are defined in [18] and [27]. Simulated call duration is 120 seconds. Silence suppression is not used (for simplicity). Voice stream may include a single call or a group of calls (aggregated voice traffic). All calls in the group use the same speech encoding algorithm; calls can be managed simultaneously; and all have the same behavior (the quality of all calls degrades equally).

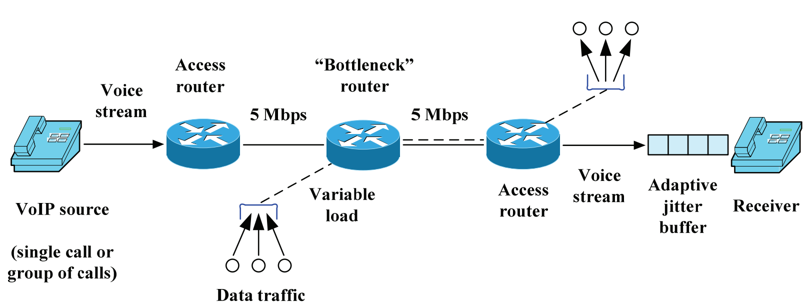



Figure 1: Simulated network structure

### D. Background Traffic Modeling

It is desirable to generate background traffic with characteristics similar to traffic patterns in the Internet. Studies revealed that Internet traffic exhibits properties of (1) self-similarity, (2) burstiness and (3) long-range dependency (LRD) [33]. The self-similar process shows that short-time traffic behavior patterns are close to long-time patterns. LRD means that there is a statistically significant correlation across large time-scales [34]. In [35, 36], it is suggested to use multiple Pareto aggregation sources with a Pareto index parameter $a < 2$. Our study uses this approach by generating Pareto On/Off traffic from 10 different sources. The traffic from each source consists of sending packets at a fixed rate only during the On periods, whereas the Off periods are idle. The packet sizes are: 64 bytes (60% of packets), 550 bytes (25% of packets) and 1500 bytes (15% of packets). The aggregated traffic will have all the required characteristics. The Network Simulator NS-2 [37] also uses this model of background traffic generation.

It is important that the model does not separate the generated traffic into TCP and UDP flows. But the approach is good to model Internet traffic behavior, even in congested networks, ignoring nonlinearities arising from the interaction of multiple traffic sources because of network resource limitations and TCP's feedback congestion control algorithm [38]. One possible reason is that more than 90 percent of TCP sessions in the Internet are very short (1-2 seconds) and exchange less than 10 Kbytes of data [39].

According to recent research [40, 41], the nature of traffic in the Internet changes because of a significant increase of peer-to-peer traffic and assumptions used in this section may not be true in future. The proportion of peer-to-peer traffic in the network increased significantly during the last several years and achieves 50% of the total traffic. This fact may change two assumptions: a) the packet-size distribution of the Internet traffic will change; b) TCP session will exchange more data, will be longer and the TCP back-off mechanism should be simulated. This study uses the On/Off Pareto model and the TCP-based model will be analyzed in future.

### E. Jitter Buffer Management

Jitter is a variation in packet transit delay caused by queuing and serialization effects on the path through the network. It is eliminated by jitter buffers, which temporarily store arriving packets and send them to a receiver in equal intervals. The buffer may have a fixed size, but if is too small, a lot of packets may be discarded because of a significant delay variation. This will negatively affect speech quality. Increasing the jitter buffer allows waiting longer for delayed packets but increases the overall end-to-end delay, which also negatively affects speech quality. A lot of research focuses on adaptive jitter buffer strategies to find some optimal point in the tradeoff between the end-to-end delay and packet loss and to optimize speech quality dynamically.

The basic adaptive playout algorithm of Ramjee *et al.* [42] is used in the simulation. It calculates two statistics to make a jitter buffer adaptation decision. For each arriving packet, it computes the expected mean and variation in the end-to-end delay ($d_i$ and $\upsilon_i$. respectively). Specifically, the end-to-end delay estimate for packet *i* is computed as

$$d_i = \alpha \cdot d_{i-1} + (1-\alpha) \cdot n_i \quad (3)$$

where $n_i$ is the *i*-th packet delay. The variation is: as:

$$\upsilon_i = \alpha \cdot \upsilon_{i-1} + (1-\alpha) \cdot | d_i - n_i | \quad (4)$$

Packet playout time in this algorithm is calculated as:

$$p_i = t_i + d_i + \beta \upsilon_i \quad (5)$$

where $t_i$ is the time the packet was sent. $\alpha = 0.875$ and $\beta=4$.

While these equations estimate $d_i$ and $\upsilon_i$ for each packet, playout time $p_i$ is adjusted only in the intervals between talkspurts (periods of speech). Different papers use periods between 200 and 700 ms to describe talkspurt durations and there is no agreement about the "best" number. The ITU P.59 [43] recommendation specifies an artificial on/off model for generating human speech with the talkspurts and silence intervals of 227 ms and 596 ms. Jiang and Schulzrinne [44] reported mean spurts and gaps of 293 ms and 306 ms in experiments with the G.729 codec. The durations of 300 ms both for silence and active speech periods are used in the simulation.

## IV. The Effect of End-user Parameters on VoIP Quality

### A. Effect of Background Traffic on VoIP Quality

Before investigating the effect of the parameters on speech quality, it is important to demonstrate that the quality is affected by not only high link utilization, but also by the proportion of voice and data traffic in the network. This hypothesis is rather evident: the data traffic is bursty and it may cause a sudden congestion in the network, which results in higher delays and/or losses of voice packets.

The simulation design is based on the network scheme and the assumptions described in the previous section. The narrowband G.711 codec with 64 kbps of audio bit rate is used in the example. Speech quality is measured in MOS depending on the total link utilization U (voice and data traffic; average during the call) and D (ratio of data traffic and total mixed traffic). Note that U is not an average ISP network utilization; this is a utilization of the bottleneck link during the considered 120-second simulation. The results and the standard deviations are presented in Fig. 2.

The results show that the behavior of voice traffic significantly depends on the presence of, and load of, the data traffic in the network. Even a relatively small volume of data traffic can cause a significant degradation of voice traffic. While this conclusion is intuitively clear, the simulation study gives numerical estimates. If the average link utilization exceeds 70%, adaptive quality management mechanisms can potentially be used and change of packet size and/or speech compression may decrease the quality degradation effect. But, this number (70%) depends on the assumptions and will be slightly different with other adaptive jitter buffer algorithms, different approaches for background traffic generation, and assumptions about talkspurt duration

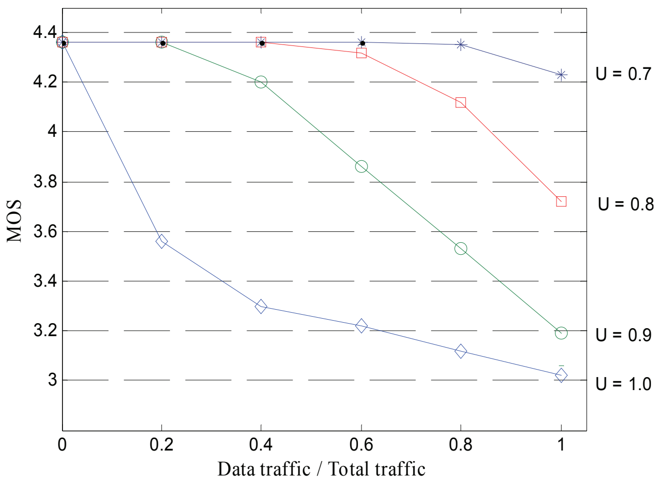



Figure 2: Speech quality depending on traffic structure

It is necessary to understand that, in most cases, the structure of traffic in the network (in the congested router) is not known and not possible to measure (unless a user or a provider controls all hardware devices along a path of a given call). But the results of the study presented below still helpful in understanding of how variations of voice stream parameters affect its quality and in which scenarios the end-user codec adaptation may be beneficial.

### B. *Effect of Packet Size Variation on VoIP Quality*

The effect of packet size variation is difficult to describe theoretically because many of the parameters affecting quality (delay, loss, jitter) are not independent and improving one parameter may cause a decline in another. Some effects of packet size on speech quality are very clear, others are less evident. Four main relationships are identified:

(1) Increasing packet size leads to an increase of end-to-end delay. If the delay is not too large, the direct impact of packet size increase is very small and not perceptually noticeable [18]. But, if the delay is significant, an additional increase of packet duration may be noticeable.

(2) Increasing packet size leads to decrease of the IP-rate per call. This may decrease congestion in the network and improve the quality of communication. The dependency is also evident and the question about the effectiveness of voice transmission was briefly discussed in Section 2.

There are two less evident, but also important effects of packet size variation on VoIP quality:

(3) A loss of one “long” packet has more significant negative effect on speech quality than a random loss of several “small” packets. Mathematical representation of this effect in the E-model can be found in [18].

(4) In presence of data traffic in the network, increase of voice packet size decreases link utilization, but increases the data-to-voice traffic ratio. As demonstrated in the previous section, this may cause additional “instability” in the network, which may result in higher jitter, delay or loss. This factor may affect the resulting speech quality, but it is not clear how significant the effect might be.

It is seen that increasing packet size causes different effects on the speech quality. Since, the result is difficult to predict theoretically, it is investigated using simulation. The portions of voice (number of calls) and data traffic in the network were changed, and the average call quality was measured for different packet sizes. Speech quality is measured in MOS depending on the portion of voice traffic in the network $V$, the total link utilization $U$ (average during the call) and the packet size $PS$ (changes from 10 to 50 ms). For each combination of $\{V, U, PS\}$, 100 experiments were run. The standard deviation of MOS scores does not exceed 0.15 MOS. The simulation results for several specific values of $V$ (portion of voice traffic) are shown in Fig. 3.

a) V = 0 (single call)

b) V = 0.4 (40% voice traffic)

c) V = 0.8 (80% voice traffic)

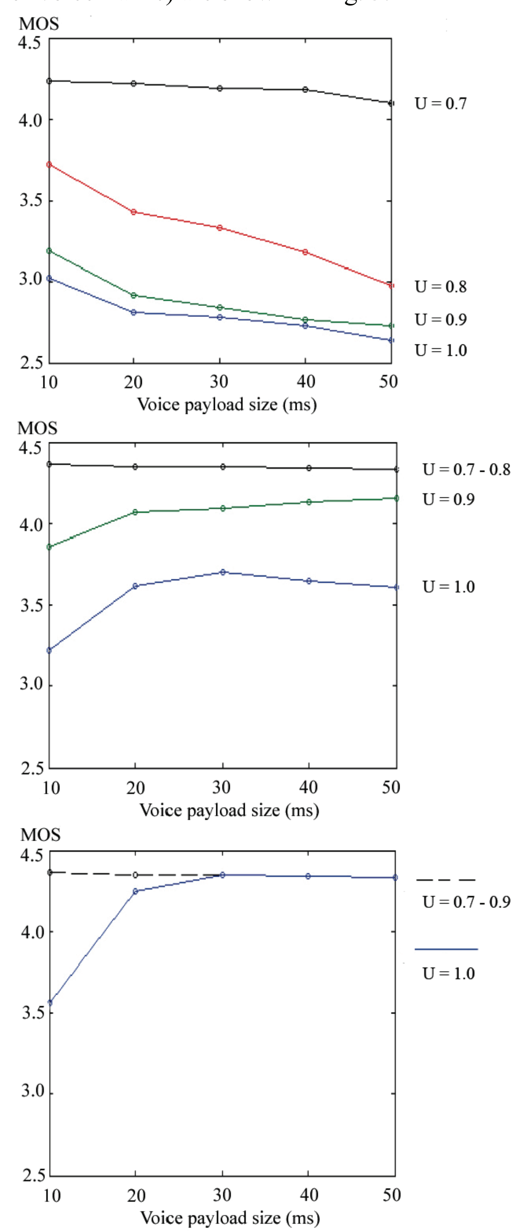



Figure 3: Effect of packet size on VoIP quality

In the case of a single call or a relatively small number of calls (Figure 3a), change of packet size does not provide any quality improvement because it does not influence the situation in the network. Increase of packet size in this scenario makes the situation even worse; it is recommended to keep it small (10 ms).

Congested links with dominated voice traffic are more stable (Figures 3b, 3c). They can handle higher utilization (for example, 80% or even 90%) without noticeable quality degradation (again, these numbers depend on the assumptions about the network structure, traffic generation, etc). In these scenarios, quality can be improved even further using higher rate codecs.

Based on the study, the “lower bound” of managed voice traffic in a congested link when change of packet size provides noticeable improvement in quality is about 30%. With higher voice load, packet size variation improves quality despite the multiple negative effects discussed in the previous section. In these cases, a 20-ms packet size is enough to increase average call quality; 30 ms can provide even better quality under some scenarios.

### C. *Effect of Compression on VoIP Quality*

In addition to the packet size variation, multiple encoding algorithms with compression can be used. Compressed speech often has even smaller IP-rate, but there is a more significant loss in codec quality due to compression. In addition to the G.711 codec, this study considers the G.726 codec with 2:1 compression and the G.729a codec with 8:1 compression. The goal of this section is to answer the question: if a number of simultaneously managed calls does not change, would it be better to use the codec with higher compression under certain network conditions or packet size variation would be more effective.

Similar to the previous section, change of compression causes opposing effects on VoIP quality:

(1) Increasing compression generally leads to a decreased codec quality. For example, the maximum quality, which can be achieved by the G.729a codec under ideal conditions, is noticeably less than that of the G.711 codec.

(2) Increasing compression leads to decreased IP-rate per call. This may decrease congestion in the network (especially if a group of calls is managed) and improve the quality of communication.

(3) Increasing compression not only decreases codec quality, but also the effectiveness of packet loss concealment algorithms. Concealment of compressed packets is less effective. For example, the loss of 1% of the G.729a packets has more significant negative impact on speech quality than the same loss of the G.711 packets.

(4) As in the previous section, if there is data traffic in the network, voice compression increases the data-to-voice traffic ratio, which may negatively result on speech quality.

Mean VoIP communications qualities in several scenarios: 1) the G.711 codec is used (no compression, 20 ms packet size); 2) the G.726 codec (2:1 compression; 20 ms packet size); 3) the G.729a codec (8:1 compression; 20 ms packet size). Notations: V - portion of voice traffic in the network, U - total bottleneck link utilization (voice and data traffic; average during a call). U and V are measured based on the G.711 codec, 10-ms packet size. Fig. 4. shows the simulation results for several specific values of *V* (portion of voice traffic).

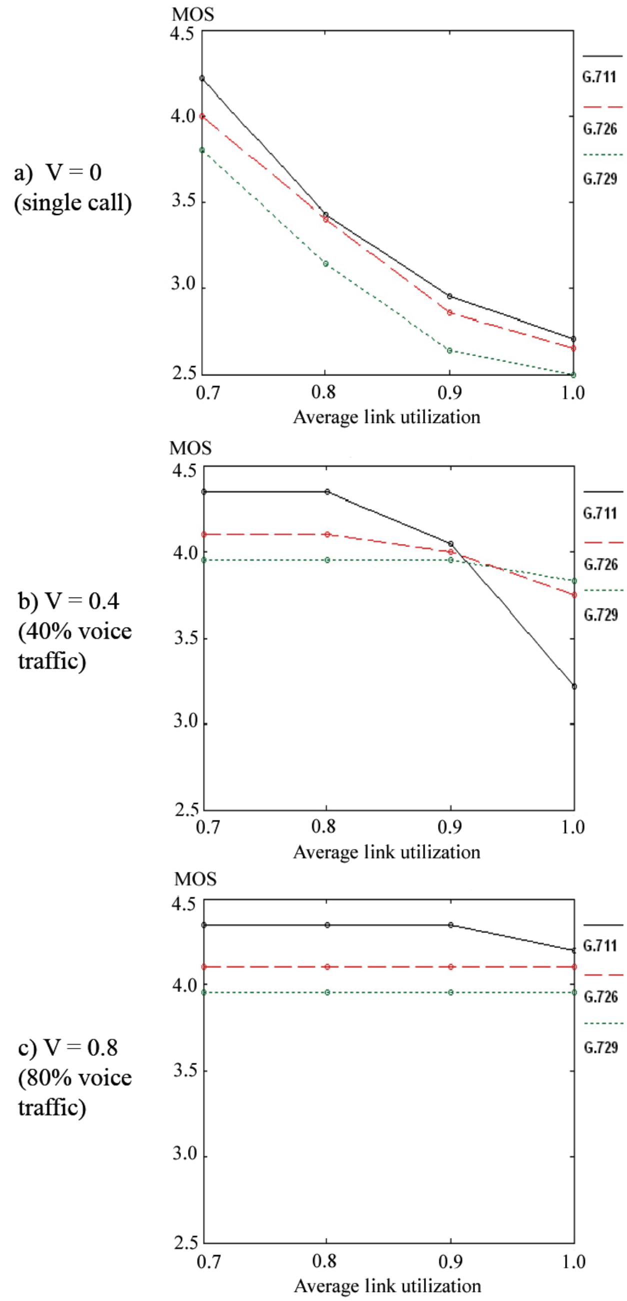



Figure 4: Effect of compression on VoIP quality

The results demonstrate that, when a portion of managed voice traffic in the network is relatively small (Fig. 4a), neither the packet size variation nor the increase of compression (or even both of them together) provide improvement in quality. For best achievable quality under the given scenario, it is necessary to use the best available codec (the G.711 or even wideband codecs) and small voice payload size (10 ms; smaller packet sizes make the communication process very inefficient). If portion of managed voice traffic is more significant, changing packetization provides better resulting speech quality than compression variation (until a certain level of the bottleneck link utilization) despite the fact that the compressed speech uses less channel capacity (Fig. 4c). Only when the bottleneck link is very heavily congested (more than 90%), the compressed encoding provides better quality (Fig. 4b).

### *D. Summary*

The presented results are based on assumptions about network structure, background traffic pattern, speech characteristics (talkspurt duration), which are described in the previous section. Other jitter buffer algorithms or, for example, a different model of speech representation (talkspurt and silence periods) will change numerical results, but the general conclusion will remain the same: in congested networks try to manage voice payload size first, because it may provide higher resulting quality than codecs with compression; if this does not help – change a voice stream bit rate using both higher compression and packet size variation. The results are consistent with other studies mentioned in Section 2, but provide approximate quantitative criteria when end-user variation of encoding parameters may be effective for average call quality improvement.

This study had assumed that average background traffic load in the network is known. In real networks this information is not available to the end-points, so mechanisms would be needed to estimate the effect of the network on communication quality. Monitoring a call or a group of calls on the receiver side, we do not know average link utilization and average volume of background traffic in the network. Number of simultaneously managed calls is known, but this information is not too important: the quality depends on the proportion of data traffic in the network and, even if there are many calls, often it cannot be said with high confidence whether the network is voice-only or even if voice traffic dominates in the network. For this reason, the conclusions from this part cannot be used directly.

It is important to remember that two adaptive mechanisms are going to work simultaneously: a) the proposed variable sender-based encoding mechanisms and b) the receiver's adaptive jitter buffer. The adaptive jitter buffer mechanism is used to improve short-term quality (its fast reaction does not change the encoding characteristics, but manages the delay-loss tradeoff). Sender-based management is designed to improve long-term voice flow characteristics (to choose the encoding scheme that best matches the given network conditions).

## V. Decision-making Parameters for Adaptive Speech Quality Management

Which parameters should be taken into account to make a decision about quality adaptation? One variant is to use, for example, the mean delay, the moving average delay, or loss statistics. This approach has already been used in several papers mentioned above. It would not be acceptable to analyze these parameters separately: high packet loss definitely means a significant degradation in quality but low (or absence of) packet loss does not mean an absence of degradation because the adaptive jitter buffer size can be very significant and we can get high end-to-end delay instead of the loss. So, it would be better to use these parameters together. In other words, one must measure quality, which depends on end-to-end delay, loss, and codec characteristics. This project does not analyze some "less evident" parameters affecting speech quality like echo, attenuation, noise in a channel, etc. The quality can be measured using the computational E-model [18].

How can this model be used? The adaptive algorithms proposed here will measure and manage quality-per-talkspurt. Human communication consists of periods of active speech and periods of silence. The adaptive jitter buffer algorithm changes its buffer size in periods between talkspurts (during silence periods). So, it would be logical to analyze speech quality behavior, and to make decisions about adaptive quality management at the end of a talkspurt (at the end of an active speech period). The E-model can be used to calculate the quality of each talkspurt (referred to as "instantaneous quality"). Packets within a talkspurt have the same end-to-end delay; network loss and jitter buffer loss can easily be counted. The difference between instantaneous quality levels in two consecutive talkspurts can be very significant because of the bursty nature of the background traffic.

But, knowledge of instantaneous call quality is not enough to make a decision about changing the speech encoding parameters. Measurable voice quality can change significantly and immediately but it takes some time for users to understand that the quality has changed. Perceptual (real) speech quality is different from an instantaneous computational quality. Perceptual quality takes into account factors not only during the last short period of time (last second or several seconds), but all quality values and quality variation history starting at the beginning of a call [45]. So, in addition to the computational quality model, it would be useful to estimate (1) instantaneous perceptual speech quality at any moment of time during a call, and (2) integral quality at any moment of time during a call.

The E-model can also be used to measure the average call quality at a given moment of time during a conversation. This metric would probably be acceptable, but this project tries a different approach, using a metric, that describes integral (total) speech quality. Integral quality that is calculated as a mean of instantaneous qualities is not a very good metric. This model does not take into account the history of previous quality variations (frequent variations may result in a relatively high average quality but noticeably smaller real perceptual quality).

Instead of using just a mean MOS metric, integral quality is calculated from the beginning of a call using the perceptual model of Rosenbluth [25]. This model presents a call as a sequence of 8-second intervals. Quality within each interval is calculated as a mean of instantaneous qualities. Integral call quality is calculated as a weighed sum of the qualities of the longer intervals and reflects an opinion that quality levels at the end of a conversation have higher weights on the overall perceived call quality than quality levels at the beginning of a session. Perceptual quality is usually lower than the average computational quality if there are frequent variations of instantaneous quality levels. This model was justified by subjective experiments performed by AT&T. It is proposed to use the weighting average with weights

$$W_i = \max\left[1,\ 1+(0.038+1.3\cdot L_i^{0.68})\cdot(4.3-MOS_i)^{\{0.96+0.61\cdot L_i^{1.2}\}}\right] \quad (6)$$

$$MOS_I = \frac{\sum_i W_i \cdot MOS_i}{\sum_i W_i} \quad (7)$$

$MOS_I$ is the integral perceptual call quality; $MOS_i$ is the MOS during the shorter measurement period; $L_i$ is a location of a degradation period (measured on 0-to-1 scale; 0 indicates the beginning of a conversation, 1 is the end of a conversation; the parameter changes proportionally to time starting from the beginning of a call).

This perceptual call quality metrics will be used as one of decision parameters for adaptive speech quality management. If there are concerns about this model, it is possible to calculate the integral quality using weighted average with exponentially distributed weights described in [45]. The idea in Equations 6 and 7 is similar; the representation is a little bit more complex than just the exponentially distributed weighting.

## VI. Adaptive VoIP quality management algorithm

This section describes the proposed adaptive quality management mechanism. The following section provides and example and results of a simulation study.

Step 1: Collect statistics of packet delays before the jitter buffer. If there are multiple calls, is assumed that the quality degradation pattern of all calls to be similar (this assumption is confirmed by our preliminary simulations). So, just one call from aggregated voice traffic may be chosen for analysis.

Step 2: Calculate packet playout time (Section 3E). This parameter is calculated continuously for each arrived packet but, since the jitter buffer is adjusted only in pauses between talkspurts (between active speech periods), the end-to-end delay is constant within a talkspurt.

Step 3: Calculate the quality of a talkspurt based on the E-model. This calculation includes: (1) calculating packet loss within a talkspurt (network and jitter buffer loss); (2) measuring end-to-end delay, which is constant within a talkspurt. This "quality per talkspurt" is referred as "instantaneous quality" and denoted it as $Q_I$. The difference between instantaneous quality levels in two consecutive talkspurts can be very significant because of the bursty nature of the background traffic. But, knowledge of this instantaneous call quality is not enough to make a decision about adaptation of speech encoding parameters.

Step 4: Calculate the maximum achievable quality level for a given codec under the given network conditions. This calculation is based on zero packet loss and minimum network delay (the sum of transmission delay, propagation delay and minimum queuing delay). This delay is taken from the analysis of incoming packets delays. Using this minimum network delay, calculate the maximum achievable quality under the given set of encoding characteristics:

$$R = R_0 - I_{e\text{-}eff} - I_d$$

$$I_d = \text{min. network delay} + \text{min. jitter buffer delay}$$

$$MOS = 1 + 0.035R + R(R-60)(100-R)\cdot 7\cdot 10^{-6} \quad (8)$$

MOS is a Mean Opinion Score on the 1-to-5 scale; R is the indicator of voice quality (the 100-point scale) from the computational E-model, $R_0$ = 93.2 is the maximum achievable narrowband quality, $I_d$ is a function of delay; $I_{e\text{-}eff}$ describes impairments related with encoding (codec quality) and packet loss. See the E-model standard [18] for more details about voice quality calculation. The result looks something like in Fig. 5:

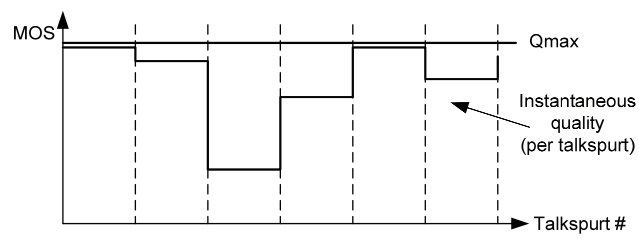



Figure 5: Instantaneous speech quality measurement

Step 5: Continuously calculate the integral voice quality level based on the model expressed by Equations 6 and 7.

Step 6: Decision. The proposed quality management scheme uses three parameters: 1) instantaneous quality level – $Q_I$; 2) integral perceptual quality – $Q_T$; and 3) maximum quality level achievable under the given set of speech encoding parameters – $Q_M$. If the number of managed calls (which is assumed to be known) is not significant (for example, one or two), voice flow is concluded to have an insignificant effect on the network, and it would be better to use the best available codec from the very beginning and not to use any quality adaptation strategies.

Similar to [7], two thresholds are used in the algorithm: 0.25 MOS and 0.5 MOS. These thresholds are used only to describe integral quality variation and these numbers are not chosen arbitrarily. A change in quality of 0.2-0.25 MOS is not too significant, but is noticeable by some people; smaller changes in quality are noticeable only by a relatively small percentage of listeners. A change in quality of about 0.5 MOS is very significant and is noticeable by almost everybody. If the best narrowband G.711 codec is used and its quality decreases by 0.5 MOS, the resulting quality will be lower than the toll-grade quality level. If the G.729 codec is used and its quality degrades by 0.5 MOS, the resulting level of speech quality (about 3.4-3.5 MOS) is considered to be low by most people.

The situation with thresholds for instantaneous quality level is a little more complex. Talkspurt quality can decrease for two reasons: 1) high jitter buffer size, which results in high end-to-end delay and, usually, not significant loss, or 2) packet loss (usually not in the network but on a receiver size caused by significant delay variation and insufficient jitter buffer size). The effect of delay is generally lower: for example, with 150 ms of network and packetization delay and an additional 80 or 100 ms of jitter buffer size, the decrease in quality is about 0.3-0.4 MOS. But, if a bursty loss of packets causes a loss of, for example, only 3 out of 30 packets in a given talkspurt, the decrease in the instantaneous quality equals to 0.9 MOS. If 5 packets out of 30 are discarded, the resulting quality (for the G.711 codec) will be only 2.65 MOS (with a maximum level of 4.3-4.4 MOS). As in the case of integral quality, two thresholds are used: 0.3 and 1.0. If the difference between maximum and instantaneous quality levels does not exceed 1.0 on the MOS scale, the observed packet loss is considered reasonable.

This model does not use quality adaptation mechanisms during the first several seconds of conversation because the perceptual quality model expressed by Equations 7 and 8 is very sensitive to quality variations in the beginning of a call. In the simulation, this period is set to 8 seconds.

The details of the algorithm follow. Consider the differences between two parameters: 1) between the maximum and integral qualities ($Q_M - Q_T$), and 2) between the maximum and instantaneous quality levels ($Q_M - Q_I$). The first difference quantifies total quality variation; the second one describes instantaneous quality changes.

Condition 1:

- if $Q_M - Q_T > 0.5$ *// Low or unacceptable level of quality. Something has to be done immediately*
  - if $Q_M - Q_I > 1.0$ *// Instantaneous quality level is also very low. Not too much can be done in this situation. Switch to the G.729 codec with 30 ms packet size (the worst codec using the minimum IP-rate)*
    - ⇨ *Action: switch to the G.729 codec, 30 ms packet size*
  - if $0.3 < Q_M - Q_I < 1.0$
    - ⇨ *Action: keep current settings expecting that adaptive buffer will compensate the degradation*
  - if $Q_M - Q_I < 0.3$ *// Total quality is very low, but instantaneous quality level is close to maximum. Quality degradation is not seen, so start slowly to improve the codec quality by deceasing packet size*
    - ⇨ *Action: decrease packet size by 10 ms if a current size is higher*

Condition 2:

- if $0.2 < Q_M - Q_T < 0.5$ *// Degradation of quality is noticeable, try to improve the situation*
  - if $Q_M - Q_I > 1.0$ *// Have a long bursty loss of packets. The network is significantly congested.*
    - ⇨ *Action: use codec with higher compression: if current codec is G.711, switch to G.726; if current codec is G.726, switch to G.729*
  - if $0.3 < Q_M - Q_I < 1.0$ *// Also, the situation is not good and bursty packet loss is observed*
    - ⇨ *Action: increase packet size by 10 ms or change codec if current packet size is 30 ms (maximum)*
  - if $Q_M - Q_I < 0.3$ *// Instantaneous quality is good; expect integral quality improvement*
    - ⇨ *Action: decrease packet size by 10 ms if a current size is higher*

Condition 3:

- if $Q_M - Q_T < 0.2$ *// Degradation of quality is not significant but might be noticeable; try to improve the situation*
  - if $Q_M - Q_I > 1.0$ *// Significant quality degradation. Total quality is good but one more bursty loss can significantly drop overall quality. Try to avoid.*
    - ⇨ *Action: increase packet size by 10 ms up to 30 ms*
  - if $0.3 < Q_M - Q_I < 1.0$ *// Assume that this decrease of quality is temporal and due to single loss or increase of end-to-end delay*
    - ⇨ *Action: keep current settings*
  - if $Q_M - Q_I < 0.3$ *// Everything is fine: both total and instantaneous qualities are high.*
    - ⇨ *Action: decrease packet size up to minimum or switch to a better codec*

The algorithm is summarized in Table 1.

<u>Step 7:</u> To change current encoding algorithm according to the decision.

The actions defined above cannot be executed immediately. The collected information about instantaneous and integral quality levels has to be transmitted to the sender. The transmission delay can be significant in congested networks. Assume that three consecutive talkspurts on the sender side (TS1, TS2, TS3) are separated by periods of silence (S1, S2). According to assumptions in Section 3, the mean durations of the active speech and silence period are

TABLE I: THE ALGORITHM SUMMARY

| Q(M)-Q(I) | Q(M)–Q(T) | | |
|---|---|---|---|
| | ≤ 0.2 | 0.2 < … < 0.5 | ≥ 0.5 |
| ≤ 0.3 | - if current packet size is higher than 10 ms, decrease it - if current size is 10 ms but used codec is not the G.711, switch to a better codec | - decrease packet size up to 10 ms | - if current packet size is lower than 30 ms, increase it |
| 0.3 < … < 1.0 | - keep current settings | - if current packet size is lower than 30 ms, increase it | - if used codec is not G.729, switch to a codec with higher compression |
| ≥ 1.0 | - if current packet size is lower than 30 ms, increase it | - if used codec is not G.729, switch to a codec with higher compression | - switch to the G.729 codec with 30 ms packet size |

300 milliseconds. Assume that the receiver gets TS1 and makes a decision to send some control information to the sender. The period of time between the departure of the TS1 talkspurt and the arrival of the feedback from the receiver is equal to a round-trip delay (RTT). In congested networks this RTT might be significant and longer than the period of silence between the TS1 and TS2 talkspurts. In this case, the decision about quality adaptation will not be applied to the second talkspurt (TS2); it would be applied to TS2 only if the RTT is less than the S1 duration (300 ms). So, the receiver would not see the result of the requested changes of speech encoding parameters until the TS3 talkspurt, about one second later. The minimum reaction time of the algorithm is 300 ms (when RTT ≤ 300 ms). If the assumptions about speech/silence duration are different, these numbers will change respectively.

This fact has to be taken into account in the adaptation scheme. So, a restriction is added that, if a receiver analyzes a talkspurt (for example, TS1) and sends a control message to the sender to change speech encoding parameters, the next control message cannot be sent after the next consecutive talkspurt.(TS2); but only after analyzing of TS3, if it is required. This restriction provides more stability to the algorithm.

One more restriction is added. If a decision is made about several consecutive improvements of speech encoding parameters (for example, to decrease the voice payload size from 30 ms to 20 ms and then to 10 ms or to replace a given codec by a better codec) these changes should not be made too quickly because each change causes noticeable increase of IP-rate per call and thus, a higher probability of degradation due to congestion. The preliminary experiments showed that the system is more stable if the receiver waits for four talkspurts (about 2 seconds) between such decisions.

In sender-based control, observations about the network and resulting speech quality must be reported back to the sender. Utilizing RTCP is a common approach: packet loss and delay variation statistics are included in RTCP reports. These packets are sent periodically, usually every 5 seconds. But, obviously, this type of control is very slow to respond to the network. If the control mechanism must make decisions more frequently, it is necessary to use a different scheme rather than RTCP. But more frequent periodic call control may introduce additional traffic in the network. The paper assumes that the adaptive quality management mechanism sends control messages, not periodically, but on demand, that is, when a change of sender parameters is required. As the simulation results will demonstrate, a decision to vary the encoding parameters is made less frequently than per talkspurt. Also, if a group of calls is managed, just one message can be sent to deliver feedback from the receiver, but not to control every call independently. This approach will not create a significant amount of additional control traffic in the network

## VII. SIMULATION RESULTS

### A. *Example*

Fig. 6 demonstrates an example of the statistics collected in one of the simulations. The first picture shows the packet delay and the jitter buffer size calculated in Step 2. If packet delay (blue line) is higher than jitter buffer size (red line), the packet is lost (discarded) and restored using a current codec's packet loss concealment mechanism. The second figure demonstrates instantaneous (per talkspurt) speech quality (blue line, Step 3) and integral (perceptual) quality (red line, Step 5). The third picture shows the background traffic rate averaged over 10 ms intervals, and also the average long-term traffic rate.

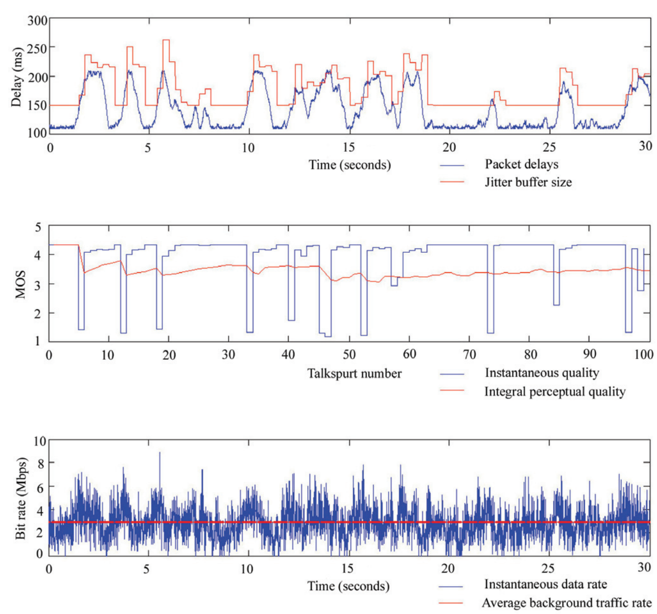
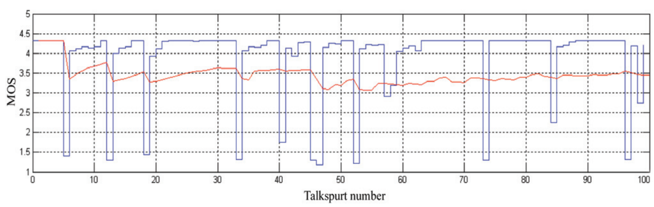



Figure 6: Measured speech quality statistics

Figures 7 shows the quality of the VoIP stream when exactly the same background traffic pattern exists and the stream is managed by the proposed adaptive quality management mechanism. This example considers a rather congested network with 40% of voice and 50% of data traffic. Blue (upper) line is the instantaneous quality level; red (lower) line is the integral quality level. Because of the real-time adaptation, the algorithm helped to choose encoding settings and change them dynamically to improve average communication quality.

Without adaptive encoding




With the adaptive scheme

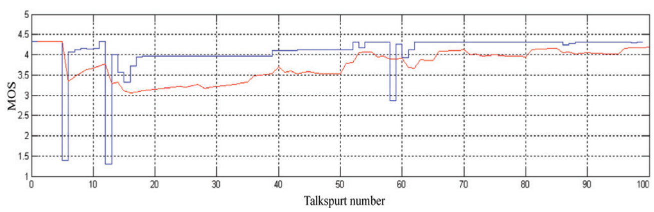


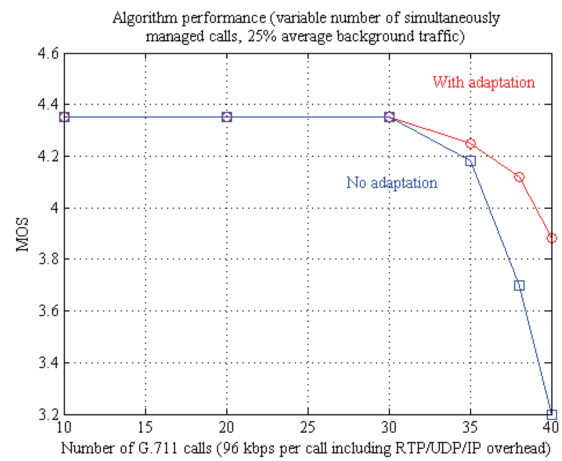

Figure 7: Speech quality comparison

### B. *Simulation study*

The simulation described in Section 3 was used to analyze effectiveness of the proposed adaptive speech quality management algorithm. As it was mentioned in Section 4, the algorithm (and the approach of adaptive encoding in general) is not effective if a portion of managed real-time voice traffic in the network (in the bottleneck) is less than approximately 30%. The simulation was performed for 25% and 50% of average background data traffic load in the network and for different number of voice calls. The graphs in Fig. 4 show average qualities with and without adaptive encoding calculated using the E-model.

The results compare average MOS scores in the two scenarios; the simulation was executed 20 times for each set of parameters (more extensive simulation study will be preformed in future). Fig. 8 demonstrates that adaptive encoding allows decreasing of quality degradation in case of network congestion. For example, having 50% average background traffic load and 20 simultaneous fixed-rate G.711 calls, average quality of these calls was around 3.4 MOS. Adaptive algorithm detects degradation in quality caused by traffic burstiness and high network utilization, adaptively changes packetization or encoding and results in better average quality (around 3.9). The average increase in quality is quite significant.

Although the increase in average quality is seen with the proposed algorithm, two important things should be emphasized. 1) In heavily congested networks, individual MOS scores for a call can be noticeably different. Running the simulation multiple times and having the same volume of traffic and the same number of VoIP calls in the network, significant difference in call quality may be seen. This happens because of different background traffic patterns (burstiness of generated traffic in a given simulation). 2) Increase in average quality does not mean that call quality is improved in all individual simulations. For example, having 50% average background traffic load and 18 G.711 calls in the network, out of 20 runs, 2 experiments showed slight decrease in quality (less than 0.2 MOS) and two had even more significant drop in quality (about 0.4-0.5 MOS). This also happens because of background traffic burstiness: in certain cases, the algorithm expects a significant congestion in the network and future quality degradation, and switches to compressed lower quality codec. But, if situation in the network suddenly improves, the adaptation decision becomes not effective. The "intelligence" of the algorithm should be improved to decrease the number of such failures.



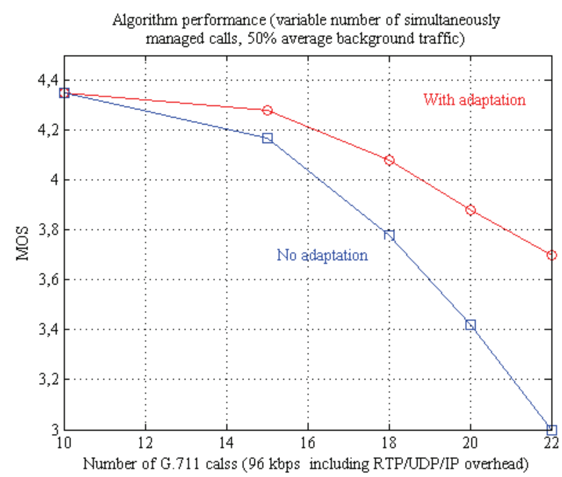




Figure 8: Simulation results

## VIII. CONCLUSION AND FUTURE WORK

In this paper, an adaptive control mechanism was designed to dynamically manage and improve the average quality of VoIP communication. In this scheme, the receiver makes a control decision based on two parameters: 1) the computational instantaneous quality level, which is calculated per talkspurt using the E-model and 2) the perceptual metric, which estimate the integral speech quality by taking into account the fact that a decrease of communication quality depends, not only on the presence of packet delay or loss in the network, but also on the position of a quality degradation period in the call.

The algorithm works together with the adaptive jitter buffer mechanism. The adaptive jitter buffer is used to manage short-term quality; the sender-based adaptation technique tries to choose encoding parameters to improve a long-term quality by decreasing network congestion and, as a result, significant instantaneous changes in quality.

The paper uses three narrowband codecs with different packet sizes for analysis but the approach can be extended to a wider set of narrowband codecs, to wideband encoding schemes and, potentially, to IP-based audio in general.

Several questions will be addresses in the future work. First, it is necessary to work more on the algorithm and to decrease a number of cases when adaptive encoding makes a situation with quality worse. Second, according to recent research, the nature of traffic in the Internet changes because of significant increase of peer-to-peer traffic and assumptions about background traffic modeling used in Section 3 may not be true in future. Other traffic models simulating TCP backoff mechanism and/or mix of TCP and UDP traffic should be studied. Third, an adaptive change of speech encoding parameters affects perceptual quality. The proposed algorithm tried to avoid rapid changes of codecs and make adaptation as smooth as possible, but the question about what users hear when encoding parameters are changed and how this variation affects user feeling about a call, should also be investigated in more detail.

## REFERENCES


[1] RFC 2474, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", 1998

[2] RFC 2475, "An Architecture for Differentiated Service", 1998

[3] RFC 2205, "Resource ReSerVation Protocol (RSVP)", 1997

[4] RFC 1633, "Integrated Services in the Internet Architecture: an Overview", 1994

[5] RFC 3031, "Multiprotocol Label Switching Architecture", 2002

[6] E. Myakotnykh, R. Thompson, "Adaptive Speech Quality Management in Voice-over-IP Communications", Fifth Advanced International Conference on Telecommunications AICT'09, Venice, Italy, May 2009

[7] Z. Qiao, L. Sun, N. Heilemann and E. Ifeachor, "A New Method for VoIP Quality of Service Control Use Combined Adaptive Sender Rate and Priority Marking", IEEE International Conference on Communications, 2004

[8] J. Seo, S. Woo, K. Bae, "Study on the application of an AMR speech codec to VoIP", Acoustics, Speech, and Signal Processing, 2001 Proceedings

[9] J. Matta, C.Pepin, K. Lashkari, R. Jain, "A source and channel arte adaptation algorithm for AMR in VoIP using the E-model", Proceedings of NOSSDAV 2003

[10] 3G TS 26.071, "Mandatory Speech Codec speech processing functions; AMR Speech Codec; General Description", 1999

[11] RFC 3550: "RTP: A Transport Protocol for Real-Time Applications", July 2003

[12] ITU-T G.711, "Pulse code modulation (PCM) of voice frequencies", Geneva, Switzerland, 1988

[13] H. Oouchi, T. Takenaga, H. Sugawara and M. Masugi, "Study on Appropriate Voice Data Length of IP Packets for VoIP Network Adjustment", NTT Network Service Systems Laboratories, 2002

[14] L. Yamamoto, J. Beerends, "Impact of network performance parameters on the end-to-end perceived speech quality", Expert ATM Traffic Symposium, Greece, 1997

[15] B. Ngamwongwattana, "Effect of packetization on VoIP performance", in Proc. ECTI-CON, 2008

[16] J. Bolot and A. Vega-Garcia, "Control Mechanisms for Packet Audio in the Internet", Proceedings IEEE Infocom, San Francisco, CA, pp 232-239, 1996

[17] S. Mohamed, F. Cervantes-Perez and H. Afifi, "Integrating Network Measurements and Speech Quality Subjective Scores for Control Purposes." IEEE Infocom, 2001

[18] ITU-T G.107, "The E-model, a computational model for use in transmission planning", Geneva, Switzerland, 2000

[19] Y. Huang, J. Korhonen, Y. Wang, "Optimization of source and channel coding for voice over IP", IEEE International Conference on Multimedia and Expo, 2005

[20] S. Huang, P. Chang, E. Wu, "Adaptive voice smoothing with optimal E-model method for VoIP services ", IEICE transactions on communications, vol. 89, 2006

[21] B. Ngamwongwattana, "Sync & Sense Enabled Adaptive Packetization VoIP", PhD Dissertation, University of Pittsburgh, 2007

[22] F. Sabrina, J. Valin, "Adaptive Rate Control for Aggregated VoIP Traffic", Globecom, 2008

[23] J. Valin, "The speex codec manual", http://www.speex.org/docs

[24] ITU-T P.800, "Methods for subjective determination of transmission quality", Geneva, Switzerland, 1996

[25] J. H. Rosenbluth, "Testing the Quality of Connections having Time Varying Impairment", Committee contribution T1A1.7/98-031, 1998

[26] ITU-T Rec. G.107 Amendment 1, "Provisional impairment factor framework for wideband speech transmission", 2006

[27] ITU-T Rec. G.113, "Transmission impairments due to speech processing", Geneva, Switzerland, 2001

[28] S. Möller, A. Raake, N. Kitawaki, A. Takahashi, M.Wältermann, "Impairment Factor Framework for Wide-Band Speech Codecs", IEEE Transactions on Audio, Speech, and Language Processsing, Vol. 14, No. 6, November 2006

[29] A. Takahashi, H. Yoshino, "Perceptual QoS assessment technologies for VoIP", IEEE Communications Magazine, July 2004

[30] ITU-T SG12 D.139: "France Telecom study of the relationship between instantaneous and overall subjective speech quality for time-varying quality speech sequences", Geneva, Switzerland, 2000

[31] ITU-T G.726, "40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)", Geneva, Switzerland, 1990

[32] ITU-T G.729, "Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)", Geneva, Switzerland, 1996

[33] V. J. Ribeiro, M. Coates, R. H. Riedi, S. Sarvotham, B. Hendricks, and R. Baraniuk, "Multifractal cross-traffic estimation", in Proc. of ITC Specialist Seminar on IP Traffic Measurement, September 2000

[34] T. Karagiannis, M. Molle, and M. Faloutsos, "Long-range dependence: Ten years of Internet traffic modeling", IEEE Internet Computing, 2004

[35] M. S. Taqqu, W. Willinger, and R. Sherman, "Proof of a Fundamental Result in Self-Similar Traffic Modeling", ACM Computer Communications Review, pp. 5 – 23, April 1997

[36] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through High-Variability: Statistical Analysis of Ethernet

LAN Traffic at the Source Level", Proceedings of the ACM/SIGCOMM'95, Cambridge, MA, 1995

[37] The Network Simulator – ns-2, www.isi.edu/nsnam/ns/

[38] K. Park, G. Kim, M. Crovella, "On the relationship between file sizes, transport protocols, and self-similar network traffic", Proc. IEEE International Conference on Network Protocols, 1996

[39] C. Williamson, "Internet traffic measurement", Internet Computing, IEEE, 2001

[40] N. Basher, Aniket Mahanti, Anirban Mahanti, C. Williamson, and M. Arlitt, "A Comparative Analysis of Web and P2P Traffic", WWW2008. Beijing

[41] Laird Popkin (Pando Network), Doug Pasko (Verizon), "P4P: ISPs and P2P", 2006

[42] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive playout mechanisms for packetized audio applications in wide-area networks", in Proc. INFOCOM, 1994

[43] ITU-T P.59, "Telephone Transmission Quality Objective Measuring Apparatus: Artificial Conversational Speech", Geneva, Switzerland, 1996

[44] W. Jiang and H. Schulzrinne, "Analysis of On-Off Patterns in VoIP and their Effect on Voice Traffic Aggregation", IEEE International Conference on Computer Communications and Networks, 2000

[45] A. D. Clark, "Extensions to the E Model to incorporate the effects of time varying packet loss and recency" ,Telecommunication standards contribution, Document T1A1.1/2001-037, April 2001