A "GAP-Model" based Framework for Online VVoIP QoE Measurement

Prasad Calyam, Eylem Ekici, Chang-Gun Lee, Mark Haffner, and Nathan Howes

Abstract: **Increased access to broadband networks has led to a fast-growing demand for voice and video over IP (VVoIP) applications such as Internet telephony (VoIP), videoconferencing, and IP television (IPTV). For pro-active troubleshooting of VVoIP performance bottlenecks that manifest to end-users as performance impairments such as video frame freezing and voice dropouts, network operators cannot rely on actual end-users to report their *subjective* quality of experience (QoE). Hence, automated and *objective* techniques that provide real-time or online VVoIP QoE estimates are vital. Objective techniques developed to-date estimate VVoIP QoE by performing frame-to-frame peak-signal-to-noise ratio (PSNR) comparisons of the original video sequence and the reconstructed video sequence obtained from the sender-side and receiver-side, respectively. Since processing such video sequences is time consuming and computationally intensive, existing objective techniques cannot provide online VVoIP QoE. In this paper, we present a novel framework that can provide online estimates of VVoIP QoE on network paths without end-user involvement and without requiring any video sequences. The framework features the "GAP-model", which is an offline model of QoE expressed as a function of measurable network factors such as bandwidth, delay, jitter, and loss. Using the GAP-model, our online framework can produce VVoIP QoE estimates in terms of "Good", "Acceptable", or "Poor" (GAP) grades of perceptual quality solely from the online measured network conditions.**

Index Terms: **Network management, user quality of experience (QoE), video quality.**

I. INTRODUCTION

Voice and Video over IP (VVoIP) applications such as Internet telephony (VoIP), videoconferencing and IP television (IPTV) are being widely deployed for communication and entertainment purposes in academia, industry, and residential communities. Increased access to broadband networks and significant developments in VVoIP communication protocols viz., H.323 (ITU-T standard) and SIP (IETF standard), have made large-scale VVoIP deployments possible and affordable.

For pro-active identification and troubleshooting of VVoIP performance bottlenecks, network operators need to perform real-time or online monitoring of VVoIP QoE on their operational network paths on the Internet. Network operators cannot rely on actual end-users to report their *subjective* VVoIP QoE on an on-going basis. For this reason, they require measurement tools that use automated and *objective* techniques which do not involve end-users for providing on-going online estimates of VVoIP QoE.

Objective techniques [1], [2] developed to-date estimate VVoIP QoE by performing frame-to-frame peak-signal-to-noise ratio (PSNR) comparisons of the original video sequence and the reconstructed video sequence obtained from the sender-side and receiver-side, respectively. PSNR for a set of video signal frames is given by Equation (1).

$$PSNR(n)_{db} = 20\log_{10}\left(\frac{V_{peak}}{RMSE}\right) \quad (1)$$

where signal $V_{peak} = 2^k - 1$; k = number of bits per pixel (luminance component); and *RMSE* is the mean square error of the Nth column and Nth row of sent and received video signal frame n. Thus obtained PSNR values are expressed in terms of end-user VVoIP QoE that is quantified using the widely-used "mean opinion score" (MOS) ranking method [3]. This method ranks perceptual quality of an end-user on a subjective quality scale of 1 to 5. Fig. 1 shows the linear mapping of PNSR values to MOS rankings. The [1, 3) range corresponds to "Poor" grade where an end-user perceives severe and frequent impairments that make the application unusable. The [3, 4) range corresponds to "Acceptable" grade where an end-user perceives intermittent impairments yet the application is mostly usable. Lastly, the [4, 5] range corresponds to "Good" grade where an end-user perceives none or minimal impairments and the application is always usable. Thus, the aim of network operators is to sustain MOS rankings in Good grade levels [4, 5] for optimal end-user VVoIP QoE.

Such a PSNR-mapped-to-MOS technique can be termed as an *offline* technique because: (a) It requires time and spatial alignment of the original and reconstructed video sequences, which is time consuming to perform, and (b) it is computationally intensive due to its per-pixel processing of the video sequences. Such offline techniques are hence not useful for measuring real-time or online VVoIP QoE on the Internet. In addition, the PSNR-mapped-to-MOS technique does not address impact of the joint degradation of voice and video frames. Hence, impairments that annoy end-users such as "lack of lip synchronization" [4] due to voice trailing or leading video are not considered in the end-user QoE estimation.

To address these problems, we present a novel framework in this paper that can provide *online* objective estimation of VVoIP QoE on network paths: (a) Without end-user involvement for quality rankings, (b) without requiring any video sequences, and (c) considering joint degradation effects of both voice and video. Fig. 2 shows our overall framework to produce

Manuscript received May 15, 2007.

P. Calyam is with the Ohio Supercomputer Center, email: pcalyam@osc.edu.

E. Ekici, M. Haffner, and N. Howes are with the Ohio State University, email: ekici@ece.osu.edu, {haffner.12, howes.16}@osu.edu.

The corresponding author is C.-G. Lee. He is with the Seoul National University, email: cglee@snu.ac.kr.

This work has been supported in part by the Research Settlement Fund for the new faculty of Seoul National University (SNU) and by the Ohio Board of Regents.

1229-2370/07/$10.00 © 2007 KICS

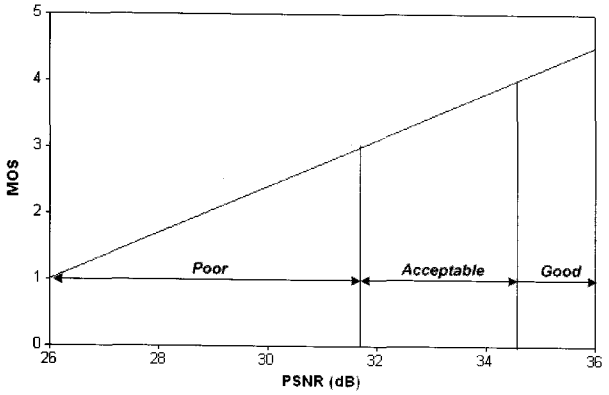


Fig. 1. Mapping of PSNR values to MOS rankings.

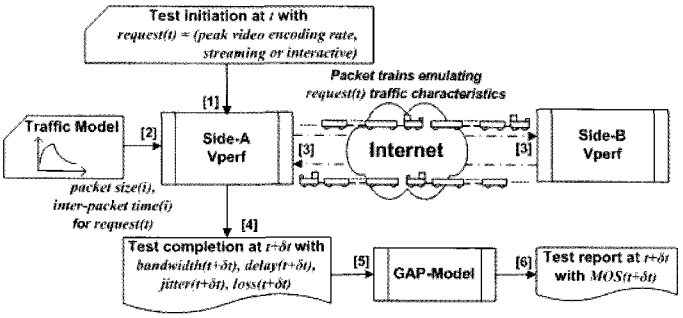


Fig. 2. Online VVoIP QoE Measurement Framework.

online VVoIP QoE estimates. The framework is based on the offline constructed, psycho-acoustic/visual cognitive model of QoE called "GAP-model". For its construction, we use a novel closed-network test methodology that asks human subjects to rank QoE of streaming and interactive audio/video clips for a wide range of network conditions. The collected human subject rankings are fed into the multiple-regression analysis resulting in closed-form expressions of QoE as functions of measurable network factors such as bandwidth, delay, jitter, and loss. Using such an offline constructed GAP-model, our online framework can estimate QoE of an online VVoIP session solely from (i) the continuously measured network conditions, and (ii) the VVoIP session information. Prior to the VVoIP session establishment or while the session is ongoing, our framework can estimate QoE following the flow of Fig. 2. The VVoIP session information *request (t)* specifies the test session's peak video encoding rate and whether the session involves *streaming* or *interactive* VVoIP streams. A streaming session is comprised of one-way streams where an end-user passively receives audiovisual content from a source at the head-end (i.e., IPTV). In comparison, an interactive session is comprised of bi-directional streams where end-users on both ends interact with each other (i.e., videoconferencing). The online network conditions are measured by test initiation at t using a VVoIP-session-traffic emulation tool called "Vperf" [5] that we have developed. After the test duration δt that is required to obtain a statistically stable measurement, the network condition measured in terms of network factors viz., *bandwidth* $(t+\delta t)$, *delay* $(t+\delta t)$, *jitter* $(t+\delta t)$, and *loss* $(t+\delta t)$ are input to the GAP-model. The GAP-model then produces a test report instantly with a VVoIP QoE estimate *MOS* $(t+\delta t)$ in terms of "Good", "Acceptable", or "Poor" (GAP) grades.

The remainder of the paper is as follows: Section II presents related work. Section III describes a VVoIP system and defines related terminology. Section IV explains the closed-network testing for GAP-model formulation. Section V presents the validation of the GAP-model. Section VI concludes the paper.

II. RELATED WORK

Objective techniques that use computational models to approximate subjective QoE (MOS) have been widely studied for VoIP applications [6]–[9]. The E-model [6] is one such technique that has been repeatedly proven to be effective and thus has been widely adopted in measurement tools developed by industry (e.g., Telchemy's VQMon [10]) and open-source community (e.g., OARnet H.323 Beacon [11]). The primary reason for E-model's success is its ability to provide online estimates of VoIP QoE based on instantaneous network health measurements (i.e., packet delay, jitter, and loss) for a given voice encoding scheme. Before E-model, the only available techniques were offline techniques such as PESQ [7] that are not suited for online monitoring of end-user VoIP QoE. The PESQ is an offline technique because it requires the source-side reference audio signal and its corresponding receiver-side audio signal that has experienced degradation due to network conditions. Although the E-model is effective for online VoIP QoE measurements, the considerations used in the E-model are not pertinent for VVoIP QoE measurements due to idiosyncrasies in the video traffic encoding and impairment characteristics. The E-model considers voice traffic as constant bit rate (CBR) encoded with constant packet sizes and fixed data rates that are known for a given voice codec with a set audio sampling frequency. In comparison, the video traffic is variable bit rate (VBR) encoded with variable packet sizes and bursty data rates that depend upon the temporal and spatial nature of the video sequences. Also, E-model considers voice signals to be affected by impairments such as drop-outs, loudness, and echoes, whereas video signals are affected by impairments such as frame freezing, jerky motion, blurriness, and tiling [12].

To estimate video QoE affected by network conditions, the most widely adopted technique is the PSNR-mapped-to-MOS technique which is offline in nature as described in Section I. The traditional PSNR-mapped-to-MOS technique was proven to be inaccurate in terms of correlation with perceived visual quality in many cases due to non-linear behavior of the human visual system for compression impairments [13]–[15]. It is now an established fact that end-user QoE and the pixel-to-pixel-based distances between original and received sequences considered in the PSNR-mapped-to-MOS technique do not always match-up with one another. Hence, several modifications have been made to the traditional PSNR-mapped-to-MOS technique to improve its estimation accuracy. The improved PSNR-mapped-to-MOS technique has been ratified by communities such as the ITU-T in their J.144 Recommendation [2] and the ANSI in their T1.801.03 Standard [16]. It is relevant to note that there are several other objective techniques to measure VVoIP QoE such as ITS, MPQM, and NVFM [33] that are all offline in nature and

are comparable to the PSNR-mapped-to-MOS technique.

Recently, there have been attempts in works such as [17]–[19] to develop a novel technique that can produce online VVoIP QoE estimates. In [17], video distortion due to packet loss is estimated using a loss-distortion model. The loss-distortion model uses online packet loss measurements and takes into account other inputs such as video codec type, coded bit rate, and packetization to estimate online PSNR values. The limitation of this work is that the PSNR degradation was not compared with subjective assessments from actual human subjects and hence the approach effectiveness is questionable. In [18], a Human Visual System (HVS) model is proposed that produces video QoE estimates without requiring reconstructed video sequences. This study validated their estimation accuracy with subjective assessments from actual human subjects, however, the HVS model is primarily targeted for 2.5/3G networks. Consequently, it only accounts for PSNR degradation for online measurements of noisy wireless channels with low video encoding bit rates. In [19], a random neural network (RNN) model is proposed that takes video codec type, coded bit rate, packet loss as well as loss burst size as inputs and produces real-time MOS estimates. All of the above models do not address end-user interaction QoE issues that are affected by excessive network delay and jitter. Further, these studies do not address issues relating to the joint degradation of voice and video frames in the end-user VVoIP QoE estimation. In comparison, our GAP-model addresses these issues as well in the online VVoIP QoE estimation.

III. VVOIP SYSTEM DESCRIPTION

Fig. 3 shows an end-to-end view of a basic VVoIP system. More specifically, it shows the sender-side, network and receiver-side components of a point-to-point videoconferencing session. The combined voice and video traffic streams in a videoconference session are characterized by encoding rate (b_{snd}) originating from the sender-side and can be expressed as

$$\begin{aligned} b_{snd} &= b_{voice} + b_{video} \\ &= tps_{voice}\left(\frac{b_{codec}}{ps}\right)_{voice} + tps_{video}\left(\frac{b_{codec}}{ps}\right)_{video} \end{aligned} \quad (2)$$

where *tps* corresponds to the total packet size of either voice or video packets, whose value equals a sum of the payload size (*ps*) of voice or video packets, the IP/UDP/RTP header size (40 bytes) and the Ethernet header size (14 bytes); and b_{codec} corresponds to the voice or video codec data rate values chosen. For high-quality videoconferences, G.711/G.722 voice codec and H.263 video codec are the commonly used codecs in end-points with peak encoding rates of $\lceil b_{voice} \rceil$ = 64 Kbps and $\lceil b_{video} \rceil$ = 768 Kbps, respectively [20]. The end-users specify the $\lceil b_{video} \rceil$ setting as a "dialing speed" in a videoconference session. The a_{lev} refers to the temporal and spatial nature of the video sequences in a videoconference session.

Following the packetization process at the sender-side, the voice and video traffic streams traverse the intermediate hops on the network path to the receiver-side. While traversing, the streams are affected by the network factors, i.e., end-to-end network bandwidth (b_{net}), delay (d_{net}), jitter (j_{net}), and loss (l_{net})

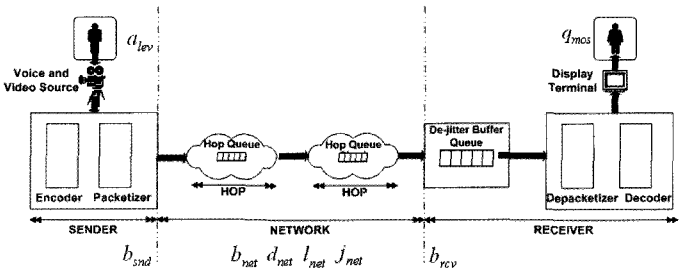


Fig. 3. End-to-end view of a basic VVoIP system.

before they are collected at the receiver-side (b_{rcv}). If there is adequate b_{net} provisioned in a network path to accomodate the b_{snd} traffic, b_{rcv} will be equal to b_{snd}. Otherwise, b_{rcv} is limited to b_{net}, whose value equals the available bandwidth at the bottleneck hop in the network path as shown in the following relations[1]

$$\begin{aligned} b_{net} &= \min_{i=1,\cdots,hops} b_{ithhop}, \\ b_{rcv} &= \min(b_{snd}, b_{net}). \end{aligned} \quad (3)$$

The received audio and video streams are processed using de-jitter buffers to smoothen the jitter effects and are further ameliorated using sophisticated decoder error-concealment schemes that recover lost packets using motion-compensation information obtained from the damaged frame and previously received frames. Finally, the decompressed frames are output to the display terminal for playback with an end-user perceptual QoE (q_{mos}).

From the above description, we can see that q_{mos} for a given set of a_{lev} and b_{snd} can be expressed as

$$q_{mos} = f(b_{net}, d_{net}, l_{net}, j_{net}). \quad (4)$$

Earlier studies have shown that the q_{mos} can be expected to remain within a particular GAP grade when each of the network factors are within certain performance levels shown in Table 1. Specifically, [21] and [22] suggest that for Good grade, b_{net} should be at least 20% more than the dialing speed value, which accommodates additional bandwidth required for the voice payload and protocol overhead in a videoconference session; b_{net} values less than 25% of the dialing speed result in Poor Grade. The ITU-T G.114 [23] recommendation provides the levels for d_{net} and studies including [24] and [25] provide the performance levels for j_{net} and l_{net} on the basis of empirical experiments on the Internet. However, these studies do not provide a comprehensive QoE model that can address the combined effects of b_{net}, d_{net}, j_{net}, and l_{net}.

IV. GAP-MODEL FORMULATION

In this section, we present the GAP-Model that produces online q_{mos} based on online measurements of b_{net}, d_{net}, l_{net}, and

[1]Note that b_{ithhop} is not the total bandwidth but the bandwidth provided to the flow at the i-th hop and hence it can never be larger than the bandwidth requested, i.e., b_{snd}. Thus, b_{net} is the bandwidth measured at the network ends for the flow.

Table 1. q_{mos} GAP grades and performance levels of network factors for $\lceil b_{video} \rceil$ = 768 Kbps.

Network factor	Good	Acceptable	Poor
b_{net}	(>922] Kbps	(576-922) Kbps	[0-576) Kbps
d_{net}	[0-150) ms	(150-300) ms	(>300] ms
l_{net}	[0-0.5)%	(0.5-1.5)%	(>1.5]%
j_{net}	[0-20) ms	(20-50) ms	(>50] ms

j_{net} for a given set of a_{lev} and b_{snd}. A novel closed-network test methodology involving actual human subjects is used to derive the GAP-model's closed-form expressions. In this methodology, human subjects are asked to rank their subjective perceptual QoE (i.e., MOS) of streaming and interactive video clips shown for a wide range of network conditions configured using the NISTnet WAN emulator [26]. Unlike earlier studies relating to QoE degradation which considered isolated effects of individual network factors such as loss [17] and bandwidth [27], we consider the combined effects of the different levels of b_{net}, d_{net}, l_{net}, and j_{net}, each within a GAP performance level.

Although such a consideration reflects the reality of the network conditions seen on the Internet, modeling q_{mos} as a function of the four network factors in three different levels leads to a large number of test cases (i.e., $3^4 = 81$ test cases) per human subject. The test cases can be ordered based on increasing network condition severity and listed as [$< GGGG >$, $< GGGA >, < GGAG >, \cdots, < APPP >, < PPPP >$], where each test case is defined by a particular sequence of the network factor levels $< b_{net}\ d_{net}\ l_{net}\ j_{net} >$. For example, the $< GGGG >$ test case corresponds to the network condition where b_{net}, d_{net}, l_{net}, and j_{net} are in their Good grade levels. Administering all the 81 test cases per human subject is an expensive process and also involves long hours of testing that is burdensome and exhaustive to the human subject. Consequently, the perceptual QoE rankings provided by the human subject may be highly error-prone.

To overcome this, we present a novel closed-network test methodology that significantly reduces the number of test cases and hence the testing time for human subjects for providing rankings without compromising the rankings data required for adequate model coverage. We note that our test methodology can be generalized for any voice (e.g., G.711, G.722, G.723) and video codec (e.g., MPEG-x, H.26x). For simplicity, we focus our testing to only the most commonly used codecs, i.e., G.722 voice codec and the H.263 video codec. These codecs are the most commonly used for business quality videoconferences as observed from our experiences during routine videoconferencing operations at the Ohio Supercomputer Center.[2] They are also the most commonly used codecs on video file sharing sites such as MySpace and Google Video.

In the following subsections, we first explain the test case reduction strategies of our novel closed-network test methodology. Next, we describe our closed-network testing with actual human subjects. Finally, we explain how the q_{mos} rankings obtained from the human subjects are processed to formulate the GAP-model's closed-form expressions.

[2]Majority of today's videoconferencing end-points use the H.263 video codec and a small fraction of the latest end-points support the H.264 video codec, which is an enhanced version of the H.263 codec targeted mainly for improved codec performance at low bit rates.

A. *Test Case Reduction Strategies*

To reduce the number of test cases per human subject for providing rankings without compromising the rankings data required for adequate model coverage, we use two strategies: (i) Reduction based on network condition infeasibility and (ii) reduction based on human subjects' ranking inference.

A.1 Reduction Based on Network Condition Infeasibility

For this strategy, we perform a network emulator qualification study to identify any practically infeasible network conditions i.e., test cases that do not exist in reality. The NISTnet WAN emulator is connected in between two isolated LANs, each having a measurement server with the Iperf tool [28] installed. Different network conditions are emulated with one factor as the control and the other factors as the response. For example, if we use b_{net} as the control, then the responses of the other three factors d_{net}, l_{net}, and j_{net} are measured and so on. All measurements are from Iperf for 768 Kbps UDP traffic streams transferred between the two LANs via NISTnet.

Figs. 4 and 5 show the Iperf measurement results that indicate the infeasible network conditions. The results are averaged over 20 measurement runs of Iperf for each network condition configuration on NISTnet. From Fig. 4 we can see that there cannot be a network condition that has Good j_{net} and Poor b_{net} simultaneously. Hence, $< P{*}{*}G > (= 1 \times 3 \times 3 \times 1 = 9)$ test cases cannot be emulated in reality. Note here that we use our previously defined network condition notation $< b_{net}\ d_{net}\ l_{net}\ j_{net} >$ and we assume '*' can be substituted with either one of the GAP grades. Similarly, from Fig. 5 we can see that there cannot be network conditions that have Good/Acceptable l_{net} and Poor b_{net} simultaneously. Hence, $< P*G* >, < P*A* >, < A*G* >$, and $< A*A* > (9 \times 4 = 36)$ test cases do not exist in reality. By considering all the infeasible network conditions, we can get rid of 39 test cases. Hence, we can reduce the number of test cases to 42 (39 subtracted from 81) per human subject for adequate model coverage.

A.2 Reduction Based on Human Subjects' Ranking Inference

In this subsection,we explain another strategy to further reduce the number of test cases per human subject for providing rankings without compromising the data required for adequate model coverage. The basic idea of this strategy is to eliminate more severe test cases during the testing based on the Poor rankings given by human subjects for relatively less severe test cases. For example, if a human subject ranked test case $< GPPP >$ with an extremely Poor q_{mos} ranking (< 2), it can be inferred that more severe test cases $< APPP >$ and $< PPPP >$ pre-

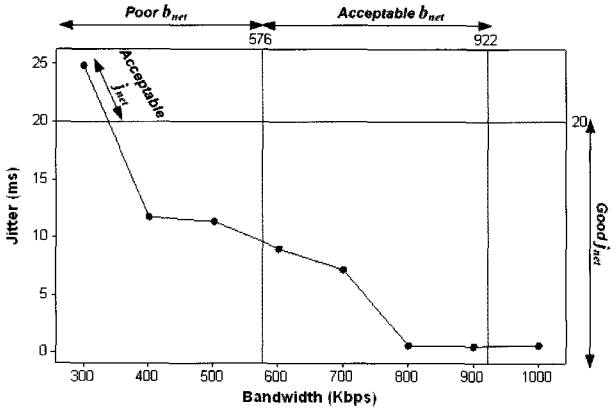


Fig. 4. j_{net} measurements for increasing b_{net}.

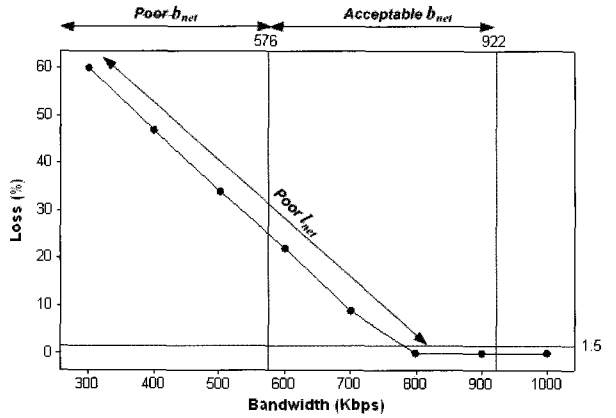


Fig. 5. l_{net} measurements for increasing b_{net}.

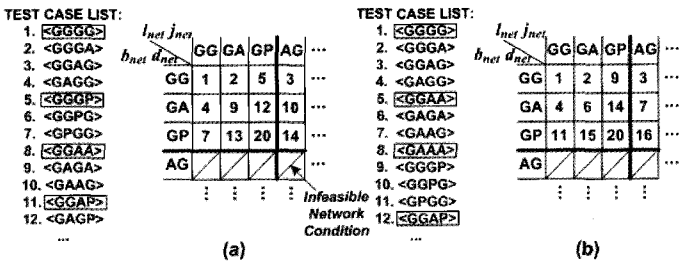


Fig. 6. (a) VF test case ordering and (b) HF test case ordering.

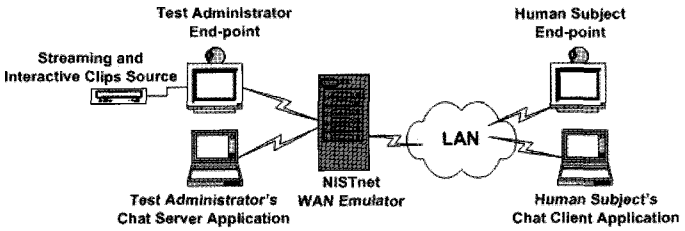


Fig. 7. Test environment setup for the closed-network testing.

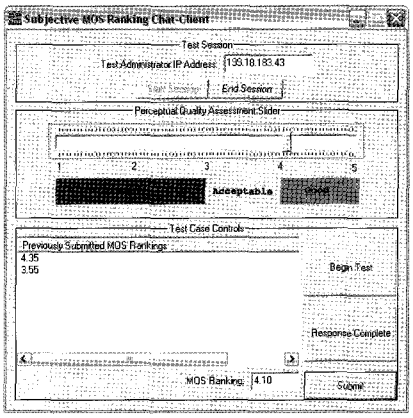


Fig. 8. Screenshot of chat client application with quality assessment slider.

sented to the human subject will also result in extremely Poor q_{mos}. Hence, we do not administer the $< APPP >$ and $< PPPP >$ test cases to the human subject during testing but assign the same Poor q_{mos} ranking obtained for $< GPPP >$ test case to the $< APPP >$ and $< PPPP >$ test cases in the human subject's final testing results. To implement the above test case reduction strategy, we present the test cases with increasing severity ($< GGGG >$ to $< PPPP >$).

Further, the test case reduction strategy can be implemented by increasing the test case severity order in two ways: (i) Vertical-first (VF) or (ii) horizontal-first (HF) — shown in Figs. 6(a) and 6(b), respectively. Using VF ordering, after $< GGGA >$, the next severe condition in the test case list is chosen as $< GGGP >$ where the severity is increased vertically (note that $< GGGA >$, $< GGAG >$, and $< GAGG >$ are equivalent severe conditions); whereas, using HF ordering, the next severe condition is chosen as $< GGAA >$ where the severity is increased horizontally. In the event that $< GGAA >$ test case receives an extremely Poor q_{mos} ranking (< 2) by a human subject, 36 ($= 3 \times 3 \times 2 \times 2$) test cases get eliminated using the inference strategy. Alternately, if $< GGGP >$ test case receives an extremely Poor q_{mos} ranking, only 27 ($= 3 \times 3 \times 3 \times 1$) test cases get eliminated. Hence, using the VF ordering, relatively lesser test cases are eliminated when an extremely Poor q_{mos} ranking occurs. Although HF ordering reduces the testing time compared to the VF ordering, we choose the VF ordering in the human subject testing because it produces more data points and thus relatively better model coverage.

B. Closed-Network Testing

B.1 Test Environment Setup

Fig. 7 shows the test environment we setup that incorporated the key considerations suggested in ITU-T P.911 [29] and ITU-T P.920 [30] for streaming and interactive multimedia QoE assessment tests, respectively. An isolated LAN testbed was used with no network cross-traffic whatsoever. The test station at the human subject end was setup in a quiet room with sufficient light and ventilation. The test station corresponds to a PC that runs a chat client application (shown in Fig. 8) and a videoconferencing end-point connected to a display terminal. The chat client application uses the "quality assessment slider" methodology recommended by [3] for recording human subject rankings. The chat client application allowed the human subject to: (a) communicate his/her test readiness using the "Begin Test" button, (b) indicate completion of his response during interactive

test clips using the "Response Complete" button, and (c) submit subjective rankings using the "MOS Ranking" field at the end of each test case to the test administrator present in a separate room. The videoconferencing end-point was used to view the streaming and interactive test clips.

The test administrator end was equipped with a PC that ran the chat server application. The test administrator end was also equipped with a videoconferencing end-point connected to a display terminal as well as a test clips source that had the streaming and interactive test clips. The test administrator controlled the test clips source and the NISTnet through a control software embedded within the chat server application. The control software guided the test administrator through the different sequential steps involved in the testing and automated core actions to control the clips source and the NISTnet configurations. To show how we addressed the difficult challenges in implementing the interactive tests and made them repeatible through automation, we present the pseudo-code of the control software for the interactive tests in the following:

Pseudo-code of the control software for ***interactive tests***

Input: 42 test cases list for interactive tests
Output: Subjective MOS rankings for the 42 test cases
Begin Procedure
1. **Step-1: Initialize Test**
2. Prepare $j = 1, \cdots, 42$ test cases list with increasing network condition severity
3. Initialize NISTnet WAN emulator by loading the kernel module and flushing inbound/outbound pipes
4. Initialize playlist in interactive clip source
5. Play interactive "baseline clip" for human subject no-impairment stimulus reference
6. **Step-2: Begin Test**
7. Enter ith human subject ID
8. **loop** for j test cases: **if**(Check for receipt of jth "Begin Test" message)
9. Flush NISTnet WAN emulator's inbound/outbound pipes
10. Configure the network condition commands on NISTnet WAN emulator for jth test case
11. Play interactive test clip from clips source
12. Pause interactive test clip at the time point where human subject response is desired
13. **if** (Check for receipt of "Response Complete" message)
14. Continue playing the interactive test clip from the clips source
15. **end if**
16. **if** (Check for receipt of "MOS Ranking" message)
17. Reset interactive test clip in the clips source
18. Save ith human subject's jth interactive MOS ranking to database
19. **if** (ith human subject's jth interactive MOS ranking < 2)
20. Remove k corresponding higher severity test cases from test case list
21. **for** each k
22. Assign ith human subject's jth interactive MOS ranking
23. **end for**
24. **end if**
25. Increment j
26. **end if**
27. **end loop**
28. **Step-3: End Test**
29. Shutdown the NISTnet WAN emulator by unloading the kernel module
30. Close playlist in interactive clip source
End Procedure

B.2 Human Subject Selection

To obtain a broad range of subjective quality rankings from our testing, we selected a total of 21 human subjects evenly distributed across three categories (7 human subjects per category): (i) Expert user, (ii) General user, and (iii) Novice user. An Expert user is one who has considerable business-quality videoconferencing experience due to regular usage and has in-depth system understanding. A General user is one who has moderate experience due to occasional usage and has basic system understanding. A Novice user is one who has little prior business-quality videoconferencing experience but has basic system understanding. Such a categorization allowed collection of subjective quality rankings that reflect the perceptual quality idiosyncrasies dependent on a user's experience level with VVoIP technology.

B.3 Video Clips

For test repeatability, each human subject was exposed to two sets of clips for which, he/she provided q_{mos} rankings. The first set corresponded to a streaming video clip *Streaming-Kelly* and the second set corresponded to an interactive video clip *Interactive-Kelly*, both played for the different network conditions specified in the test case list. These two video clips were encoded at 30 frames per second in CIF format (352 lines $\times$ 288 pixels). The duration of each clip was approximately 120 seconds and hence provided each human subject with enough time to assess perceptual quality. Our human subject training method to rank the video clips is based on the "Double Stimulus Impairment Scale Method" described in the ITU-R BT.500-10 recommendation [31]. In this method, *baseline* clips of the streaming and interactive clips are played to the human subject before commencement of the test cases. These clips do not have any impairment due to network conditions, i.e., q_{mos} ranking for these clips is 5. The human subjects are advised to rank their subjective perceptual quality for the test cases relative to the baseline subjective perceptual quality.

B.4 Test Cases Execution

Before commencement of the testing, the training time per human subject averaged about 15 minutes. Each set of test cases per human subject for the streaming as well as interactive video clips lasted approximately 45 minutes. Such a reasonable testing time was achieved due to: (a) our test case reduction strategy described in Section IV that reduced the 81 possible test cases to a worst case testing of 42 test cases, and (b) our test case reduction strategy described in Section IV that further reduced the number of test cases during the testing based on inference from the subjective rankings.

For emulating the network condition as specified by a test case, the network factors had to be configured on NISTnet to any values within their corresponding GAP performance levels shown in Table 1. We configured values in the performance levels for the network factors as shown in Table 2. For example, for the $< GGGG >$ test case, the NISTnet configuration was $< b_{net} = 960\ Kbps; d_{net} = 80\ ms; l_{net} = 0.25\%; j_{net} = 10\ ms >$. The reason for choosing these values was that the instantaneous values for a particular network condition configuration vary around the configured value (although the average of all the instantaneous values over time is approximately equal to the configured value). Hence, choosing the values shown in Table 2 enabled sustaining the instantaneous network conditions

to be within the desired performance levels for the test case execution duration.

Table 2. Values of network factors within GAP performance levels for NISTnet configuration.

Network factor	Good	Acceptable	Poor
b_{net}	960 Kbps	768 Kbps	400 Kbps
d_{net}	80 ms	280 ms	600 ms
l_{net}	0.25%	1%	2%
j_{net}	10 ms	35 ms	75 ms

C. Closed-Form Expressions

In this subsection, we derive the GAP-model's closed-form expressions using q_{mos} rankings obtained from the human subjects during the closed-network testing. As stated earlier, subjective testing for obtaining q_{mos} rankings from human subjects is expensive and time consuming. Hence, it is infeasible to conduct subjective tests that can provide complete q_{mos} model coverage for all the possible values and combinations of network factors in their GAP performance levels. In our closed-network testing, the q_{mos} rankings were obtained for all the possible network condition combinations with one value of each network factor within each of the GAP performance levels. For this reason, we treat the q_{mos} rankings from the closed-network testing as "training data". On this training data, we use the statistical multiple regression technique to determine the appropriate closed-form expressions.

The average q_{mos} ranking for a network condition j (i.e., q^j_{mos}) is obtained by averaging the q_{mos} rankings of the $N = 21$ human subjects for a network condition j, i.e.,

$$q_{mos} = \frac{1}{N}\sum_{i=1}^{N} q^{ij}_{mos}. \tag{5}$$

The q^j_{mos} ranking is separately calculated for the streaming video clip tests (S-MOS) and the interactive video clip tests (I-MOS). This allows us to quantify the interaction difficulties faced by the human subjects in addition to their QoE when passively viewing impaired audio and video streams. Fig. 9 illustrates the differences in the S-MOS and I-MOS rankings due to the impact of network factors. Specifically, it shows the decreasing trends of the S-MOS and I-MOS rankings for test cases with increasing values of b_{net}, d_{net}, l_{net}, and j_{net} network factors. We can observe that at less severe network conditions ($< GGGG >$, $< GAGG >$, $< GGAG >$, $< GAGA >$, $< GGPG >$, $< GGGP >$), the decrease in the S-MOS and I-MOS rankings is comparable. This suggests that the human subjects' QoE was similar with or without interaction in test cases with these less severe network conditions. However, at relatively more severe network conditions ($< GAPG >$, $< GGPA >$, $< GGAP >$, $< GAPA >$, $< GGPP >$, $< AGPP >$, $< PPGA >$, $< PGPP >$), the I-MOS rankings decrease quicker than the S-MOS rankings. Hence, the I-MOS rankings capture the perceivable interaction difficulties faced by the human subjects during the interactive test cases due to both excessive delays as well as due to impaired audio and video.

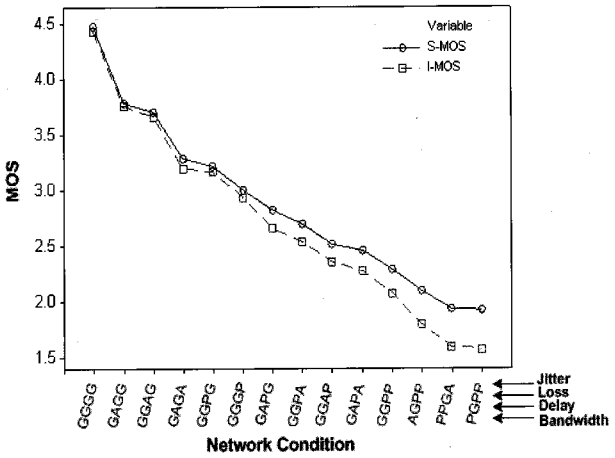


Fig. 9. Comparison of streaming MOS (S-MOS) and interactive MOS (I-MOS).

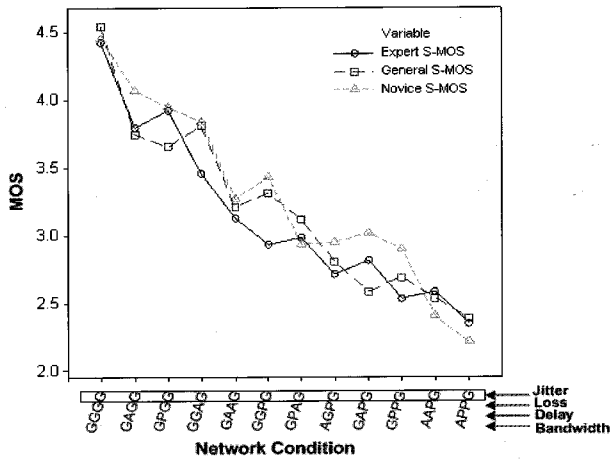


Fig. 10. Comparison of average S-MOS of Expert, General, and Novice human subjects.

As explained in Section IV, the end-user QoE varies based on the users' experience-levels with the VVoIP technology. Fig. 10 quantitatively shows the differences in the average values of S-MOS rankings provided by the Expert, General, and Novice human subjects for Good j_{net} performance level and with increasing network condition severity. Although there are minor differences in the average values for a particular network condition, we can observe that the S-MOS rankings generally decrease with the increase in network condition severity regardless of the human subject category.

To estimate the possible variation range around the average q_{mos} ranking influenced by the human subject category for a given network condition, we determine additional q_{mos} types that correspond to the 25th percentile and 75th percentile of the S-MOS and I-MOS rankings. We refer to these additional q_{mos} types as "lower bound" and "upper bound" S-MOS and I-MOS. Fig. 11 quantitatively shows the differences in the upper bound, lower bound, and average values of S-MOS rankings provided by the human subjects for Good j_{net} performance level and with increasing network condition severity. We observed similar differences in the upper bound, lower bound, and average q_{mos}

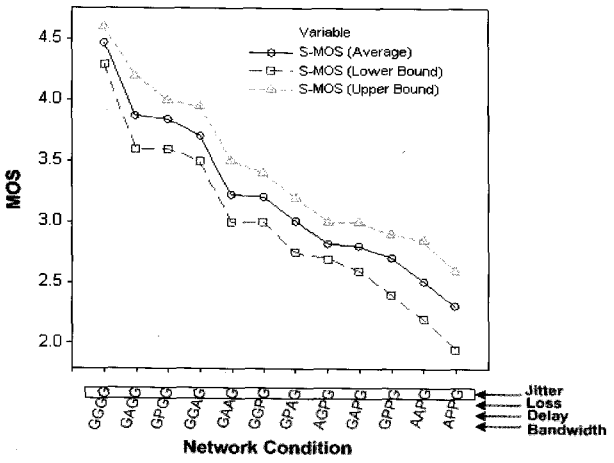


Fig. 11. Comparison of average, lower bound, and upper bound S-MOS.

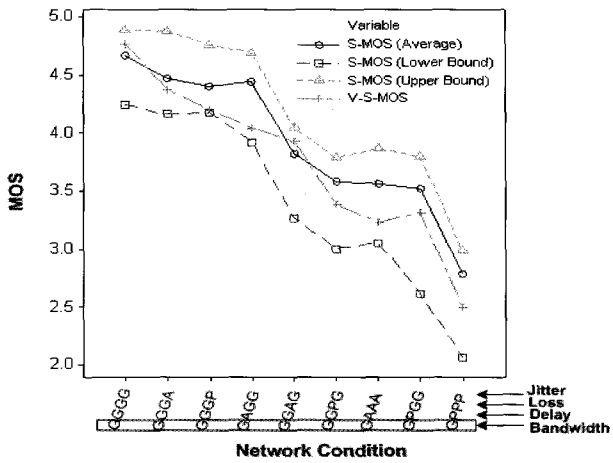


Fig. 12. Comparison of S-MOS with validation-S-MOS (V-S-MOS).

rankings for both S-MOS and I-MOS under other network conditions as well but those observations are not included in this paper due to space constraints.

Based on the above description of average, upper bound and lower bound q_{mos} types for S-MOS and I-MOS rankings, we require six sets of regression surface model parameters for online estimation of GAP-model q_{mos} rankings. To estimate the regression surface model parameters, we observe the diagnostic statistics pertaining to the model fit adequacy obtained by first-order and second-order multiple-regression on the streaming and interactive q_{mos} rankings in the training data. The diagnostic statistics for the first-order multiple-regression show relatively higher residual error compared to the second-order multiple-regression due to lack-of-fit and lower coefficient of determination (R-sq) values. Note that the R-sq parameter indicates how much variation of the response, i.e., q_{mos} is explained by the model. The R-sq values were less than 88% in the first-order multiple-regression and greater than 97% in the second-order multiple-regression. Hence, the diagnostic statistics suggest that a quadratic model better represents the curvature in the I-MOS and S-MOS response surfaces than a linear model. Table 3 shows the significant (non-zero) quadratic regression model parameters for the six GAP-model q_{mos} types, whose general representation is given as follows:

$$q_{mos} = C_0 + C_1 b_{net} + C_2 d_{net} + C_3 l_{net} + C_4 j_{net} + C_5 l_{net}^2 + C_6 j_{net}^2 + C_7 d_{net} l_{net} + C_8 l_{net} j_{net}. \quad (6)$$

V. PERFORMANCE EVALUATION

In this section, we validate the GAP-model q_{mos} rankings using a new set of tests involving human subjects. In the new tests, we use network condition configurations that were not used for obtaining the training q_{mos} rankings and thus evaluate the QoE estimation accuracy of the GAP-model for other network conditions. Finally, we compare the online GAP-model q_{mos} rankings with the q_{mos} rankings obtained offline using the PSNR-mapped-to-MOS technique.

A. GAP-Model Validation

As explained in Section IV, the GAP-model q_{mos} rankings are obtained by extrapolating the corresponding training q_{mos} rankings response surfaces. Given that the training q_{mos} rankings are obtained from human subjects for a limited set of network conditions, it is necessary to validate the performance of the GAP-model q_{mos} rankings for other network conditions that were not used in the closed-network test cases. For the validation, we conduct a new set of tests on the same network testbed and using the same measurement methodology described in Section IV. However, we make modifications in the human subject selection and in the network condition configurations. For the new tests, we randomly select 7 human subjects from the earlier set of 21 human subjects. It is relevant to note that ITU-T suggests a minimum of 4 human subjects as compulsory for statistical soundness in determining q_{mos} rankings for a test case [19]. Also, we configure NISTnet with the randomly chosen values of network factors within the GAP performance levels as shown in Table 4. Note that these network conditions are different from the network conditions used to obtain the training q_{mos} rankings. We refer to the q_{mos} rankings obtained for the new tests involving the *Streaming-Kelly* video sequence as "validation-S-MOS" (V-S-MOS). Further, we refer to the q_{mos} rankings obtained for the new tests involving the *Interactive-Kelly* video sequence as "validation-I-MOS" (V-I-MOS).

Figs. 12 and 13 show the average of the 7 human-subjects' V-S-MOS and V-I-MOS rankings obtained from the new tests for each network condition in Table 4. We can observe that the V-S-MOS and V-I-MOS rankings lie within the upper and lower bounds and are close to the average GAP-model q_{mos} rankings for the different network conditions. Thus, we validate the GAP-model q_{mos} rankings and show that they closely match the end-user VVoIP QoE for other network conditions that were not used in the closed-network test cases.

B. GAP-Model q_{mos} Comparison with PSNR-Mapped-to-MOS q_{mos}

Herein, we compare the GAP-model q_{mos} rankings with the PSNR-mapped-to-MOS q_{mos} (P-MOS) rankings. For estimat-

Table 3. Regression surface model parameters for the six GAP-model q_{mos} types.

Type	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
S-MOS	2.7048	0.0029	-0.0024	-1.4947	-0.0150	0.2918	0.0001	0.0004	0.0055
S-MOS-LB	2.9811	0.0023	-0.0034	-1.8043	-0.0111	0.3746	0.0001	0.0005	0.0069
S-MOS-UB	1.7207	0.0040	-0.0031	-1.4540	-0.0073	0.2746	0.0001	0.0004	0.0043
I-MOS	3.2247	0.0024	-0.0032	-1.3420	-0.0156	0.2461	0.0001	0.0002	0.0058
I-MOS-LB	3.3839	0.0017	-0.0032	-1.3893	-0.0177	0.2677	0.0001	0.0002	0.0055
I-MOS-UB	3.5221	0.0021	-0.0026	-1.3050	-0.0138	0.2614	0.0001	0.0001	0.0053

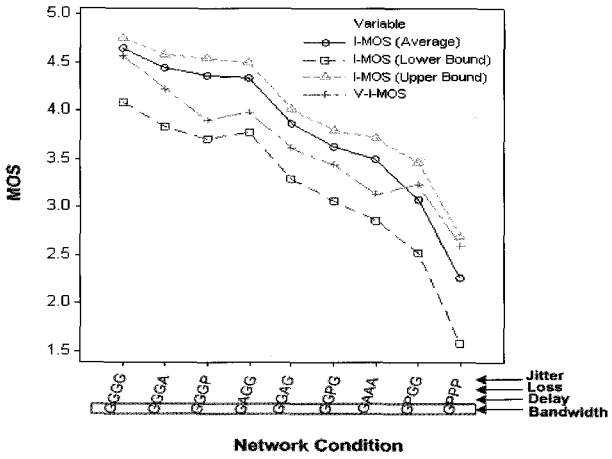


Fig. 13. Comparison of I-MOS with validation-I-MOS (V-I-MOS).

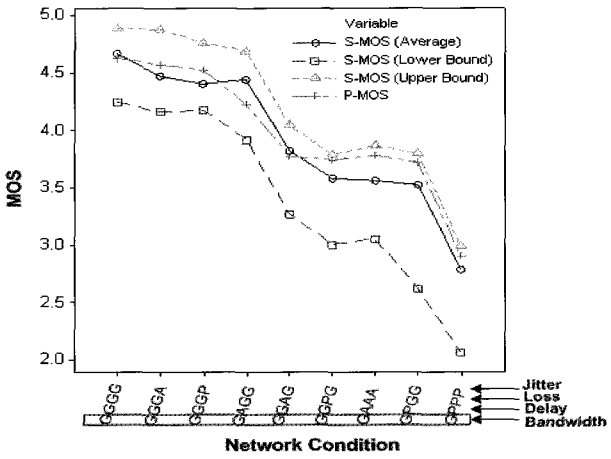


Fig. 14. Comparison of S-MOS with P-MOS.

Table 4. Values of network factors for GAP-model validation experiments.

Network factor	Good	Acceptable	Poor
d_{net}	100 ms	200 ms	600 ms
l_{net}	0.3%	1.2%	1.65%
j_{net}	15 ms	40 ms	60 ms

ing the P-MOS rankings, we use the popular NTIA's VQM software [32] that implements the algorithm ratified by ITU-T in their J.144 Recommendation [2] and the ANSI in their T1.801.03 Standard [16]. The VQM P-MOS rankings only measure the degradation of video pixels caused due to frame freezing, jerky motion, blurriness, and tiling in the reconstructed video sequence and cannot measure interaction degradation. Hence, we only compare the GAP-model S-MOS rankings with the VQM P-MOS rankings for different network conditions. To obtain the P-MOS rankings, we use the same network testbed that was used for the closed-network test cases and configure it with the network conditions shown in Table 4. For each network condition, we obtain 7 reconstructed *Streaming-Kelly* video sequences.

The process of obtaining the reconstructed *Streaming-Kelly* video sequences includes capturing raw video at the receiver-side using a video-capture device, and editing the raw video for time and spatial alignment with the original *Streaming-Kelly* video sequence. The edited raw video sequences further need to be converted into one of the VQM software supported formats: RGB24 (our choice), YUV12, and YUY2. When provided with an edited original and reconstructed video sequence pair, the VQM software performs a computationally intensive per-pixel processing of the video sequences and produces a P-MOS ranking. Note that the above process to obtain a reconstructed video sequence and the subsequent P-MOS ranking using the VQM software consumes several tens of minutes and requires a PC with at least 2 GHz processor, 1.5 GB of RAM, and 4 GB free disk space.

Fig. 14 shows the average of 7 P-MOS rankings obtained from VQM software processing of 7 pairs of original and reconstructed video sequences for each network condition in Table 4. We can observe that the P-MOS rankings lie within the upper and lower bounds and are close to the average GAP-model S-MOS rankings for the different network conditions. Thus, we show that the online GAP-model S-MOS rankings that are obtained almost instantly with minimum computation closely match the offline P-MOS rankings, which are obtained after a time-consuming and computationally intensive process.

VI. CONCLUSION

In this paper, we proposed a novel framework that can provide online objective measurements of VVoIP QoE for both streaming as well as interactive sessions on network paths: (a) Without end-user involvement, (b) without requiring any video sequences, and (c) considering joint degradation effects of both voice and video. The framework primarily is comprised of: (i) A Vperf tool that emulates actual VVoIP-session-traffic to produce online measurements of network conditions in terms of network factors viz., bandwidth, delay, jitter, and loss, and (ii) a psycho-acoustic/visual cognitive model called "GAP-model"

that uses the Vperf measurements to instantly estimate VVoIP QoE in terms of "Good", "Acceptable", or "Poor" (GAP) perceptual quality. We formulated the GAP-model's closed-form expressions based on an offline closed-network test methodology involving 21 human subjects ranking QoE of streaming and interactive video clips in a testbed featuring all possible combinations of the network factors in their GAP performance levels. The offline closed-network test methodology leveraged test case reduction strategies that significantly reduced a human subject's test duration without compromising the rankings data required for adequate model coverage.

The closed-network test methodology proposed in this paper focused on the H.263 video codec at 768 Kbps dialing speed. However, it can be applied to derive additional variants of the GAP-model closed-form expressions for other video codecs such as MPEG-2 and H.264, and higher dialing speeds. Additional variants need to be dervied for accurately estimating end-user VVoIP QoE because the network performance bottlenecks manifest differently at higher dialing speeds and are handled differently by other video codecs. If the additional variants are known, they can be leveraged for "on-the-fly" adaptation of codec bit rates and codec selection in end-points.

REFERENCES

[1] J. Klaue, B. Rathke, and A. Wolisz, "EvalVid - A Framework for Video Transmission and Quality Evaluation," *in Proc. Conf. Modeling Techniques and Tools for Computer Performance Evaluation*, 2003.

[2] *ITU-T Recommendation J.144*, "Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference," 2001.

[3] A. Watson and M. A. Sasse, "Measuring perceived quality of speech and video in multimedia conferencing applications," in *Proc. ACM Multimedia*, Sept. 1998, pp. 55–60.

[4] R. Steinmetz, "Human perception of jitter and media synchronization," *IEEE J. Sel. Areas Commun.*, pp. 61–72, vol. 14, no. 1, pp. 61–72, Jan. 1996.

[5] P. Calyam, M. Haffner, E. Ekici, and C.-G. Lee, "Measuring interaction QoE in internet videoconferencing," in *Proc. IFIP/IEEE MMNS*, Oct. 2007, pp. 14–25.

[6] *ITU-T Recommendation G.107*, "The E-model: A computational model for use in transmission planning," 1998.

[7] *ITU-T Recommendation P.862*, "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs," 2001.

[8] A. Markopoulou, F. Tobagi, and M. Karam, "Assessment of VoIP quality over Internet backbones," in *Proc. IEEE INFOCOM*, June 2002, pp. 150–159.

[9] S. Mohamed, G. Rubino, and M. Varela, "A method for quantitative evaluation of audio quality over packet networks and its comparison with existing techniques," in *Proc. MESAQUIN*, 2004.

[10] Telchemy VQMon. [Online]. Available: http://www.telchemy.com

[11] P. Calyam, W. Mandrawa, M. Sridharan, A. Khan, and P. Schopis, "H.323 beacon: An H.323 application related end-to-end performance troubleshooting tool," in *Proc. ACM SIGCOMM NetTs*, Sept. 2004, pp. 241–246.

[12] S. Winkler, *Digital Video Quality: Vision Models and Metrics*. John Wiley and Sons Publication, 2005.

[13] O. Nemethova, M. Ries, E. Siffel, and M. Rupp, "Quality assessment for H.264 coded low-rate low-resolution video sequences," in *Proc. Conf. Internet and Information Technologies*, 2004.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[15] K. T. Tan and M. Ghanbari, "A Combinational automated MPEG video quality assessment model", in *Proc. Conf. Image Processing and its Application*, July 1999, pp. 188–192.

[16] *ANSI T1.801.03 Standard*, "Digital transport of one-way video signals - Parameters for objective performance assessment," 2003.

[17] S. Tao, J. Apostolopoulos, and R. Guerin, "Real-time monitoring of video quality in IP networks," in *Proc. ACM NOSSDAV*, June 2005, pp. 129–134.

[18] F. Massidda, D. Giusto, and C.Perra, "No reference video quality estimation based on human visual system for 2.5/3G devices," in *Proc. the SPIE*, Mar. 2005, pp. 168–179.

[19] S. Mohamed and G. Rubino, "A study of real-time packet video quality using random neural networks," *IEEE Trans. Circ. Sys. for Video Tech.*, vol. 12, no. 12, pp. 1071–1083, Dec. 2002.

[20] "The video development initiative (ViDe) videoconferencing cookbook," [Online]. Available: http://www.vide.net/cookbook

[21] H. Tang and L. Duan, J. Li, "A performance monitoring architecture for IP videoconferencing," in *Proc. Workshop on IP Operations and Management*, Oct. 2004, pp. 48–54.

[22] "Implementing QoS solutions for H.323 videoconferencing over IP," *Cisco Systems Technical Whitepaper Document Id: 21662*, 2007.

[23] *ITU-T Recommendation G.114*, "One-way transmission time," 1996.

[24] P. Calyam, M. Sridharan, W. Mandrawa, and P. Schopis, "Performance measurement and analysis of H.323 traffic," in *Proc. Passive and Active Measurement Workshop*, Apr. 2004, pp. 137–146.

[25] M. Claypool and J. Tanner, "The effects of jitter on the perceptual quality of video," in *Proc. ACM Multimedia*, Nov. 1999.

[26] NISTnet Network Emulator. [Online]. Available: http://snad.ncsl.nist.gov /itg/nistnet

[27] H. R. Wu, T. Ferguson, and B. Qiu, "Digital video quality evaluation using quantitative quality metrics," in *Proc. Int. Conf. on Signal Processing*, Oct. 1998, pp. 1013–1016.

[28] A. Tirumala, L. Cottrell, and T. Dunigan, "Measuring end-to-end bandwidth with Iperf using Web100," in *Proc. Passive and Active Measurement Workshop*, 2003.

[29] *ITU-T Recommendation P.911*, "Subjective audiovisual quality assessment methods for multimedia applications," 1998.

[30] *ITU-T Recommendation P.920*, "Interactive test methods for audiovisual communications," 2000.

[31] *ITU-T Recommendation BT.500-10*, "Methodology for the subjective assessment of quality of television pictures," 2000.

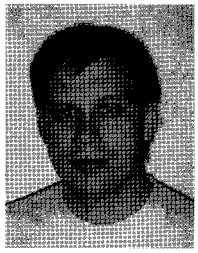
[32] M. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Trans. Broadcasting*, vol. 50, no. 3, pp. 312–322, Sept. 2004.

[33] C. Lambrecht, D. Constantini, G. Sicuranza, and M. Kunt, "Quality assessment of motion rendition in video coding," *IEEE Trans. Circ. Sys. for Video Tech.*, vol. 9, no. 5, pp. 766–782, Aug. 1999.

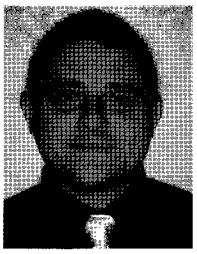
Prasad Calyam received the B.S. degree in Electrical and Electronics Engineering from Bangalore University, India, and the M.S. and Ph.D. degrees in Electrical and Computer Engineering from The Ohio State University, in 1999, 2002, and 2007, respectively. He is currently a Senior Systems Developer/Engineer at the Ohio Supercomputer Center. His current research interests include network management, active/passive network measurements, voice and video over IP, and network security.

Eylem Ekici received his B.S. and M.S. degrees in Computer Engineering from Bogazici University, Istanbul, Turkey, in 1997 and 1998, respectively. He received his Ph.D. degree in Electrical and Computer Engineering from Georgia Institute of Technology, Atlanta, GA, in 2002. Currently, he is an assistant professor in the Department of Electrical and Computer Engineering of The Ohio State University, Columbus, OH. His current research interests include wireless sensor networks, vehicular communication systems, and next generation wireless systems, with a focus on routing and medium access control protocols, resource management, and analysis of network architectures and protocols. He is an associate editor of Computer Networks Journal (Elsevier) and ACM Mobile Computing and Communications Review. He has also served as the TPC co-chair of IFIP/TC6 Networking 2007 Conference.

Chang-Gun Lee received the B.S., M.S. and Ph.D. degrees in Computer Engineering from Seoul National University, Korea, in 1991, 1993 and 1998, respectively. He is currently an assistant professor in the School of Computer Science and Engineering, Seoul National University, Korea. Previously, he was an assistant professor in the Department of Electrical and Computer Engineering, The Ohio State University, Columbus from 2002 to 2006, a research scientist in the Department of Computer Science, University of Illinois at Urbana-Champaign from 2000 to 2002, and a research engineer in the Advanced Telecomm. Research Lab., LG Information and Communications, Ltd. from 1998 to 2000. His current research interests include real-time systems, complex embedded systems, ubiquitous systems, QoS management, wireless ad-hoc networks, and flash memory systems.



Mark Haffner received the B.S. degree in Electrical Engineering from University of Cincinnati in 2006. Currently, he is pursuing an M.S. degree in Electrical and Computer Engineering at The Ohio State University. His current research interests include active/passive network measurements, RF circuit design, and software-defined radios.



Nathan Howes is pursuing a B.S. degree in Computer Science and Engineering at The Ohio State University. His current research interests include active/passive network measurements and network security.