Real-Time VoIP Quality Measurement for Mobile Devices

Whai-En Chen, *Member, IEEE,* Pin-Jen Lin, and Yi-Bing Lin, *Fellow, IEEE*

***Abstract*—The quality of voice over Internet protocol (VoIP) is typically measured by mean opinion score (MOS) that significantly varies in wireless environments. Several approaches have been proposed to evaluate the MOS values of voice paths. However, these solutions require extra servers/gateways to conduct heavy computation for MOS measurement of VoIP calls. We propose two MOS measurement procedures for mobile devices. In our approach, a simple program is installed in each of the mobile devices (through an app-store like mechanism). In the perceptual evaluation of speech quality (PESQ) MOS measurement procedure, a lightweight real-time table lookup solution significantly reduces the computation time of PESQ MOS measurement from 315.4 s to 3 s. In the E-model MOS measurement procedure, the E-model MOS value can be accurately computed in 5.35 s.**

***Index Terms*—All-IP, E-model, mean opinion score (MOS), perceptual evaluation of speech quality (PESQ), session initiation protocol (SIP), voice over Internet protocol (VoIP).**

I. Introduction

VOICE over Internet protocol (VoIP) services have been widely deployed in recent years. Due to the packet-switching nature of IP networks, VoIP's quality is typically not as stable as that of voice carried by public switched telephone network. This statement is particularly true for a mobile device that connects a VoIP call through a wireless link. Therefore, it is desirable to provide an efficient mechanism for a mobile device to quickly and accurately check the voice quality before the user actually places a call.

Mean opinion score (MOS) utilizes a digit to indicate the average voice quality and the digit ranges from 1 to 5, where 1 is bad, 2 is poor, 3 is fair, 4 is good, and 5 is excellent. The MOS measurement for voice quality is recommended by ITU-T P.800 [1]. The MOS value is produced by averaging the results of a subjective test, where the listeners (i.e., the testers) under a controlled condition rate the voice quality [2]. The major problems of the subjective tests are that the evaluation procedure consumes lots of time and the implementation of the controlled condition costs lots of money. Therefore, the subjective tests are not widely used for measuring the voice quality. The objective test is an alternative way for voice quality measurement.

Perceptual evaluation of speech quality (PESQ) specified by ITU-T P.862 [3] and E-model specified by ITU-T G.107 [4] are two well-known objective methods. PESQ compares the degraded voice signals and the reference voice signals to generate the MOS value for a voice transmission path. The degraded voice signals are generated by using real-time packet loss histogram. E-model considers different kinds of transmission impairments to rate the R factor. E-model first derives the R factor from one-way delay, packet loss rate, and the burst length of packet loss [5]. Then E-model converts the R factors into the MOS value. We will elaborate more on PESQ and E-model in Sections II and III, respectively.

A typical VoIP measurement tool [6], [7] measures the voice quality between two VoIP user agents (UAs) [IP phone devices; see Fig. 1(a) and (b)] installed in hard-wired devices such as personal computers or notebooks. These devices are connected to VoIP gateways [Fig. 1(c) and (d)] that are responsible for computing the VoIP quality. Specifically, to measure the VoIP quality for the voice path from UA1 to UA2, VoIP gateway 1 will send voice packets to VoIP gateway 2, and VoIP gateway 2 computes the VoIP quality through the packet transmission histogram. Besides, an extra VoIP server [Fig. 1(e)] is required for the network administrator to initiate the measurement procedure and to access the VoIP quality statistics. The measurement tool can use either PESQ or E-model to estimate the voice quality.

In universal mobile telecommunications system (UMTS) IP multimedia subsystem (IMS) network [Fig. 2(h) and (i)] [8], [9], the UAs [Fig. 2(c) and (g)] are implemented in the mobile devices [Fig. 2(a) and (b)] using the session initiation protocol (SIP) [10]. In this environment, the traditional MOS measurement environment in Fig. 1 does not work because a mobile device may move from one radio base station [Fig. 2(j)] to another, and it is not effective to attach the VoIP gateways to all base stations. In [11], Chan proposed an approach that measures the MOS value for the voice path between a mobile device and a SIP server. This approach eliminates the VoIP gateway in Fig. 1. On the other hand, the SIP server is required to compute the MOS value. To evaluate the VoIP quality between two mobile devices, it is more appropriate to conduct MOS measurement within these mobile devices.

Manuscript received November 14, 2010; revised March 13, 2011; accepted May 12, 2011. Date of publication September 23, 2011; date of current version November 23, 2011. This work was supported in part by the National Science Council, under Grants 97-2221-E-009-143-MY3, 98-2219-E-009-016, 98-2221-E-009-059-MY2, and 99-2628-E-197-001, and in part by Chunghwa Telecom, IBM, ITRI, III, ITRI-NCTU Joint Project, and MoE ATU Plan.

W.-E. Chen is with the National Ilan University, Ilan 260, Taiwan (e-mail: wechen@niu.edu.tw).

P.-J. Lin is with National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: pjlin@csie.nctu.edu.tw).

Y.-B. Lin is with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan, and with King Saud University, Riyadh 11451, Saudi Arabia (e-mail: liny@cs.nctu.edu.tw).

Digital Object Identifier 10.1109/JSYST.2011.2165604

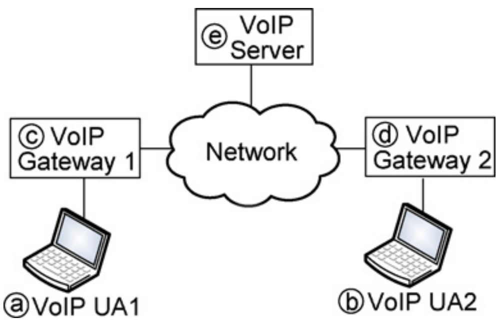

Fig. 1. Traditional MOS measurement environment.

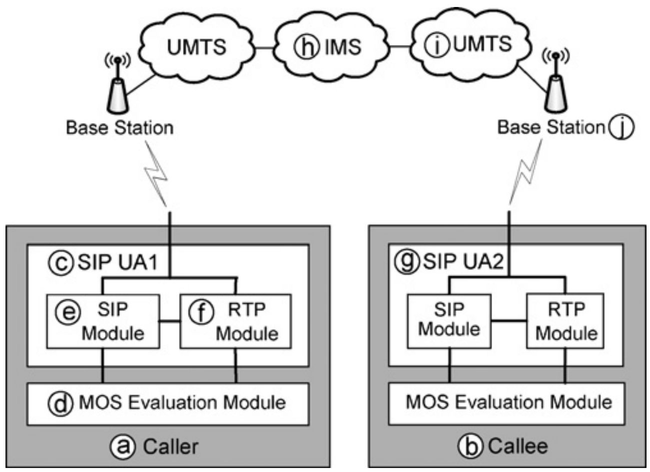

Fig. 2. Mobile device-based MOS measurement environment.

A straightforward solution is to push the VoIP gateway/server functions [Fig. 1(c)–(e)] into the mobile devices [Fig. 2(a) and (b)] [12]. A major problem of this approach is that MOS computation may consume significant amounts of CPU power. Therefore, such execution may not be affordable in a mobile device with limited computing and storage resources.

This paper proposes two efficient mobile device-based MOS measurement approaches. The first approach utilizes a table lookup method to speed up the PESQ-based MOS measurement. The second approach directly computes the E-model-based MOS within a limited (and short) period of time.

II. PESQ-Based Table Lookup MOS Measurement

As we mentioned above, PESQ compares the degraded signals and the reference signals to evaluate the quality of a voice path. In the PESQ measurement procedure [13], 20 voice files retrieved from P.862 are adopted as the input reference signals and the lengths of these voice files range from 7.1 to 8.6 s. The degraded signals are generated according to real-time packet loss histogram. For a device (i.e., Nokia N900) with the ARM Cortex-A8 600 MHz CPU, the computation for PESQ comparisons takes 306.8 s. Therefore, using traditional PESQ on the mobile devices cannot determine the MOS value during call setup and it is important to reduce the computation latency. To speed up the PESQ-based MOS measurement in a mobile device, we propose a table lookup approach. Note that

TABLE I
MOS Table for G.729 Codec

P_L \ L_B	1	2	3	4	5
1%	3.75	3.72	3.72	3.72	3.72
2%	3.67	3.62	3.60	3.57	3.56
3%	3.60	3.51	3.50	3.49	3.47
4%	3.52	3.43	3.41	3.39	3.36
5%	3.45	3.33	3.33	3.31	3.29
6%	3.40	3.26	3.24	3.22	3.21
7%	3.34	3.19	3.16	3.15	3.14
8%	3.30	3.12	3.09	3.07	3.05
9%	3.25	3.06	3.03	3.01	3.00
10%	3.21	3.02	2.99	2.97	2.95

although people typically relate PESQ MOS directly to packet loss rate, different packet loss distributions with the same rates may result in different MOS values. In our approach, packet histograms are characterized by the packet loss rate P_L and the burst packet loss length L_B, where L_B is defined as the expected number of consecutive lost packets.

The precomputed table (i.e., Table I) is generated as follows. We perform PESQ under different loss patterns. These loss patterns are generated by using the Gilbert model [14] with various P_L and L_B. We will elaborate more on the Gilbert model in Section III-B. For each combination of P_L and L_B, we execute PESQ for 4000 times and get the average MOS value. The size of the table is reasonably small because we only need to list the MOS values for $P_L < 10\%$ and $L_B < 5$. When P_L and L_B are larger than these thresholds, the MOS values are always smaller than 3 no matter what the packet loss histograms are, and the mobile device can immediately report to the user that the voice quality is poor without table lookup. Table I shows the MOS values for the G.729 codec [15] under various P_L and L_B. The MOS table for other codecs can be constructed by using the same approach and will not be elaborated further.

The MOS table is preloaded into a mobile device. Before the mobile device makes a VoIP call, it measures the voice path quality by collecting P_L and L_B statistics. These statistics are used to look up the table to obtain the MOS value for the voice path. It is clear that table lookup can significantly reduce the computing overhead and storage (because the mobile device does not need to maintain and compare 20 voice files specified in ITU-T P.862). We capture the system time before and after the MOS computation to measure the computing time. With the proposed table lookup approach, the computation latency for PESQ comparisons is reduced from 306.8 s to 55 μs.

However, the table lookup approach uses P_L and L_B to estimate the MOS values instead of the complete packet loss histograms, which may result in inaccurate MOS values. Therefore, it is desirable to evaluate the accuracy of table lookup by comparing it with the standard PESQ measurement procedure [13].

This section elaborates on how a mobile device measures the PESQ-based VoIP quality by using the table lookup approach, and then conducts experiments to evaluate the accuracy of this approach.

A. Message Flow for PESQ-Based Table Lookup MOS Measurement

This section describes the message flow for MOS table lookup before a call session is established. We assume that the calls are set up by using the SIP, and the voice packets are delivered by real-time transport protocol (RTP) [16], [17]. Our solution can be easily ported for other VoIP protocols.

In the table lookup procedure, test packets are generated and transmitted between the calling mobile device [Fig. 2(a)] and the called mobile device [Fig. 2(b)], and the mobile devices compute the MOS values of the voice path in UMTS/IMS [Fig. 2(h) and (i)]. Each of the mobile devices is installed a typical SIP UA [Fig. 2(c)] [13] and a MOS evaluation module [Fig. 2(d); this module is developed in this paper]. The SIP UA consists of the SIP module [Fig. 2(e)] for call control and the RTP module [Fig. 2(f)] for voice packet transmission. The MOS evaluation module contains a table (i.e., Table I) that maps P_L and L_B to the MOS values. The size of the MOS evaluation module is within 26 kb, which is small as compared with that of the SIP UA (with size of 3090 kb).

Assume that both SIP UA1 (140.113.1.1) and UA2 (140.113.2.2) have installed the MOS evaluation modules. When UA1 makes a call to UA2 directly (this scenario can be easily generalized for the case where a SIP server is involved), the following procedure is executed (Fig. 3).

1) *Step A.1*: UA1s SIP module sends the SIP INVITE message to UA2s SIP module. In this message, the *c* field of session description protocol (SDP) [18] indicates that the IP address of UA1 is 140.113.1.1. The first *m* field indicates that UA1 will receive the RTP voice packets (for voice conversation) at port 9000, and the second field *m* specifies that UA1 will receive the test packets (for evaluating voice quality) at port 9002. In this field, "p-mos" indicates that the PESQ MOS value will be computed.
2) *Step A.2*: Upon receipt of the INVITE message, UA2s SIP module alerts the called user that an incoming call has arrived, and sends a SIP 180 Ringing message to UA1s SIP module.
3) *Step A.3*: UA2s SIP module retrieves the IP/port information of UA1 from the SDP of the INVITE message, and sends this information to UA2s RTP module.
4) *Step A.4*: According to the *m* field "p-mos 9002 RTP/AVP 18," UA2s RTP module sends the RTP test packets to UA1s RTP module (140.113.1.1:9002). Note that a G.729 test packet is an RTP packet with 10-byte null payload. (Alternatively, the exact sending time of the packet may be stored in the payload as described in Section IV.)
5) *Step A.5*: From the first RTP test packet sent by UA2, UA1s RTP module retrieves the IP/port information of UA2 (i.e., 140.113.2.2:9002).
6) *Step A.6*: UA1s RTP module sends the RTP test packets to UA2s RTP module according to the IP/port information obtained at Step A.5.
7) *Step A.7*: The two-way test-packet transmission is conducted for a period T_P, and then each of the RTP modules computes P_L and L_B through the received packets.

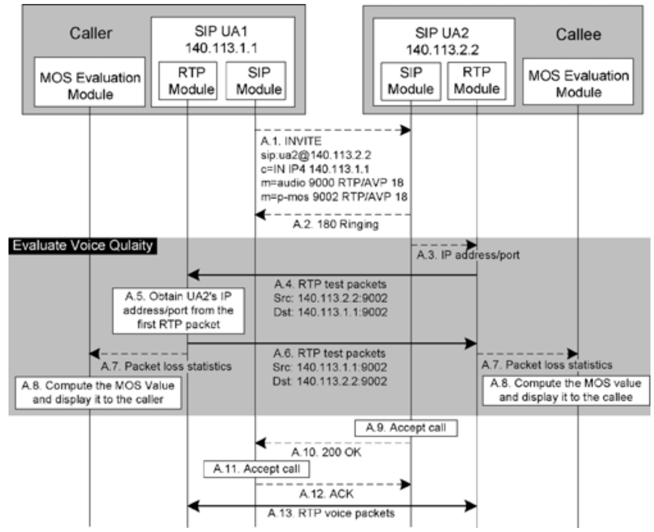

Fig. 3. Message flow of PESQ-based table lookup MOS measurement (dashed: SIP and UA internal control messages; solid: RTP packets).

8) *Step A.8*: The MOS evaluation module uses the measured P_L and L_B to look up Table I to obtain the mapped MOS value. This MOS value is displayed to the user so that he/she can decide if the voice quality is acceptable.
9) *Steps A.9 and A.10*: Suppose that the MOS value is acceptable, UA2s SIP module replies with a 200 OK message.
10) *Steps A.11 and A.12*: Upon receipt of the 200 OK message, UA1s SIP module replies with an ACK message to UA2s SIP module.
11) *Step A.13*: The voice RTP session is established between the RTP modules of UA1 and UA2.

If the called user does not accept the call at Step A.9, UA2s SIP module will not reply the 200 OK message and the call will not be established. At Step A.11, if the calling user does not accept the call, UA1s SIP module cancels the call by sending UA2 a SIP CANCEL message (if it has not received the 200 OK message) or a BYE message (if it has received the 200 OK message). If either one of the mobile devices does not install the MOS evaluation module, the call will be set up following standard SIP procedure without MOS measurement (i.e., Steps A.3–A.9 and Step A.11 will not be executed).

At Step A.7, test-packet transmission is conducted for a period T_P. It is clear that the longer the period T_P, the more accurate the MOS value. However, to provide VoIP quality to a user in real time, T_P should not be set too long. Therefore, it is important to select an appropriate T_P value such that accurate MOS value can be measured within a short period.

B. Experimental Results

This section conducts experiments to select an appropriate T_P value. We first generate 860 RTP packets (note that in G.729, the voice packets are delivered in every 10 ms and the longest length of voice file is 8.6 s) from UA1 to UA2

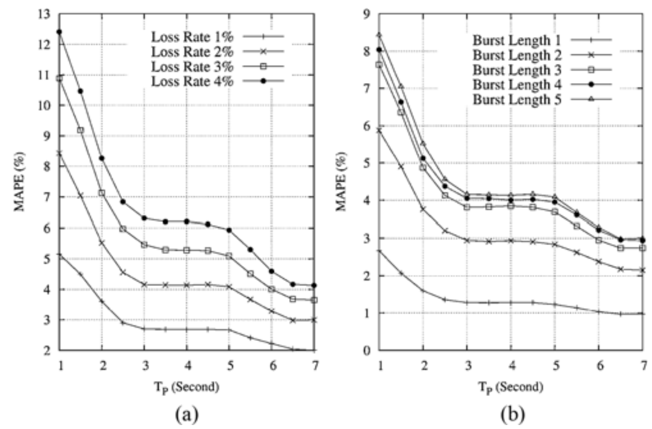

Fig. 4. MAPE of the PESQ-based table lookup MOS measurement (N = 2000). (a) L_B = 5. (b) P_L = 2%.

through a network with controlled packet loss conditions. At UA2, we execute the PESQ algorithm in [13] to obtain the accurate MOS value M_P. To measure M_P, test-packet transmission is conducted for 8.6 s, and then the voice files are used for comparison at the receiving mobile device (UA2). Such comparison takes 306.8 s.

Then we conduct the table lookup procedure to measure the packet loss rate and the burst packet loss length for T_P s, where T_P = 1, 2, ..., 7. We use the measured statistics to retrieve the corresponding MOS value $M_P(T_P)$ from the table. The elapsed time for MOS table lookup is a very short delay of 55 μs. The error $\delta(T_P)$ between M_P and $M_P(T_P)$ is defined as

$$\delta(T_P) = \left| \frac{M_P - M_P(T_P)}{M_P} \right| \times 100\%, \quad T_P = 1, 2, \ldots, 7.$$

We repeat each experiment for N times, and the error of ith run is denoted as $\delta_i(T_P)$. Then the mean average percentage error (MAPE) for T_P is calculated as

$$\mathrm{MAPE}(T_P) = \left(\frac{1}{N}\right) \sum_{i=1}^{N} \delta_{\mathrm{i}}(T_P), \quad T_P = 1, 2, \ldots, 7.$$

In our experiments, the MAPE converges when $N \geqq 2000$.

Fig. 4(a) plots the MAPE against T_P under P_L = 1%, 2%, 3%, 4%, and N = 2000. The packet loss patterns are generated from the Gilbert model [13] with L_B = 5. Fig. 4(b) plots the MAPE against T_P curves for various burst loss lengths L_B = 1, 2, 3, 4, and 5, where the packets loss rate is fixed to P_L = 2%. Fig. 4 shows that the MAPE decreases as T_P increases. Specifically, the MAPE value significantly drops when T_P increases from 1 s to 3 s. When $3 < T_P < 5$ s, MAPE is not improved by increasing T_P. According to [19], an estimation approach is considered accurate if the MAPE is within 10%. Therefore, with T_P = 3, the table lookup approach provides reasonably accurate MOS measurement (an MAPE less than 7%), while the MOS measurement time is reduced from 306.8 s to 55 μs. Note that in the standard PESQ procedure, to produce M_P, the packet transmission period is longer than 8.6 s, and the file comparison time is longer than 306.8 s [13]. Therefore, the table lookup approach can reduce the total MOS computation time from 315.4 s to 3 s.

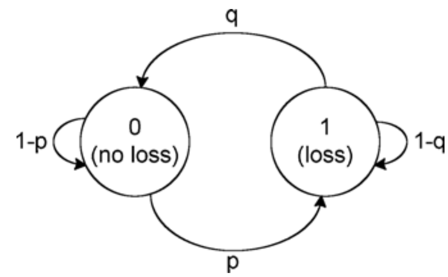

Fig. 5. State diagram of the Gilbert model.

III. E-Model-Based MOS Measurement

In this section, we propose an approach to directly compute E-model-based MOS in mobile devices. The E-model [4] computes a rating factor R, and then converts the R factor into the MOS value. The R factor can be obtained through the following:

$$R = Ro - Is - Id - Ie{-}eff. \tag{1}$$

In (1), Ro represents the basic signal-to-noise ratio. The simultaneous impairment factor Is is a combination of all impairments which occur more or less simultaneously with the voice signal. The delay impairment factor Id is calculated from one-way delay T_D through a series of equations. To reduce the computation cost in a mobile device, this tedious series of equations can be converted into a simplified sixth order polynomial function [20] as follows:

$$\begin{aligned} Id = & -2.468 \cdot 10^{-14} \cdot T_D^6 + 5.062 \cdot 10^{-11} \cdot T_D^5 - 3.903 \cdot 10^{-8} \\ & \cdot T_D^4 + 1.344 \cdot 10^{-5} \cdot T_D^3 - 0.001802 \cdot T_D^2 + 0.103 \cdot T_D \\ & -0.1698. \end{aligned} \tag{2}$$

The effective equipment impairment factor Ie-eff represents the impairments caused by packet loss and low bit-rate codecs, which is expressed as

$$Ie{-}eff = Ie + (95{-}Ie) \cdot \left(\frac{P_L}{\frac{P_L}{BurstR} + Bpl} \right). \tag{3}$$

In (3), the values of equipment impairment factor Ie and packet-loss robustness factor Bpl are dependent on the codec. For the G.729 codec, Ie = 10 and Bpl = 18. The burst ratio $BurstR$ is defined as [4]

$$BurstR = \frac{\text{Average length of observed bursts in an arrival sequence}}{\text{Average length of bursts expected for the network under random loss}}$$

which is derived from the Gilbert model described by a two-state Markov process. The state diagram of the Gilbert model is illustrated in Fig. 5, where p is the transition probability from the "no loss" state to the "loss state," and q is the transition probability from the "loss" state to the "no loss" state.

For packet loss patterns generated from the Gilbert model

$$P_L = \frac{p}{p+q} \quad \text{and} \quad L_B = \frac{1}{q} \tag{4}$$

and $BurstR$ can be expressed as

$$BurstR = \frac{1}{p+q}. \tag{5}$$

TABLE II
DEFAULT VALUES OF THE PARAMETERS IN THE E-MODEL

Parameter	Default Value
Ro	94.769
Is	1.414
Ie	10 (for G.729)
Bpl	18 (for G.729)

From (4)

$$q = \frac{1}{L_B},\ p = \frac{P_L}{L_B(1-P_L)} \tag{6}$$

and from (5) and (6), we have

$$BurstR = L_B(1-P_L). \tag{7}$$

The E-model MOS value is converted from the R factor as follows:

$$\text{MOS} = \begin{cases} 1, & \text{for } R \le 0 \\ 1+0.035R+R(R-60)(100-R)7\times10^{-6}, & \text{for } 0<R<100 \\ 4.5, & \text{for } R \ge 100. \end{cases} \tag{8}$$

Since the E-model computes the MOS value without comparing voice files, the computation overhead is relatively low as compared with that of PESQ. On the other hand, the E-model uses one-way delay as an input, which is not considered in PESQ. The measurement of one-way delay requires accurate time synchronization between the call parties. To perform time synchronization in our approach, a mobile device accesses a network time protocol (NTP) [21] server to obtain accurate time.

In this section, we elaborate on how a mobile device measures the E-model-based VoIP quality. Then we conduct experiments to determine how many test packets should be transmitted before the E-model MOS value can be obtained with a reasonable accuracy.

A. Message Flow for E-Model-Based MOS Measurement

In the E-model, default values are used for parameters *Ro*, *Is*, *Ie*, and *Bpl* [4] as listed in Table II. In our approach, P_L, L_B, and T_D are measured. Then *BurstR* is computed from P_L and L_B by using (7), and P_L and *BurstR* are used in (3) to compute *Ie-eff*. Parameter *Id* is computed from T_D by using (2). Then *Ie-eff* and *Id* are used in (1) and (8) to compute the MOS value.

Through the same network architecture illustrated in Fig. 2 (except that an NTP server in the Internet is accessed), the E-model-based MOS measurement procedure is executed as follows (Fig. 6).

1) *Step B.1*: UA1s SIP module utilizes NTP to synchronize the system time with the NTP server (e.g., time.stdtime.gov.tw in Taiwan).
2) *Step B.2*: UA1s SIP module sends the SIP INVITE message to UA2s SIP module to specify that port 9002 is used for delivering test packets. This step is the same as Step A.1 in Fig. 3, except that in the second *m* field, "e-mos" indicates that the MOS measurement will follow the E-model-based procedure.
3) *Step B.3*: Upon receipt of the INVITE message, UA2s SIP module synchronizes the system time with the NTP server.
4) *Step B.4*: UA2s SIP module alerts the called user that an incoming call has arrived, and sends a SIP 180 Ringing message to UA1s SIP module.
5) *Step B.5*: UA2s SIP module retrieves the IP/port information of UA1 from the SDP of the INVITE message, and sends this information to UA2s RTP module.
6) *Step B.6*: Transmission of test packets is similar to Steps A.4–A.6 in Fig. 3 except that the system time information (8 bytes) of the sending devices is encapsulated in the 10-byte payload of the G.729 test packets, which is used by the receiving devices to compute the one-way delay.
7) *Step B.7*: After the test-packet transmission has conducted for a time period T_E, the MOS evaluation module uses the packet loss statistics and one-way delay to compute the MOS value based on (1)–(3), (7), and (8), and displays the MOS value to the user.
8) *Step B.8*: The call parties determine if the call should be connected as in Steps A.9–A.12 in Fig. 3.

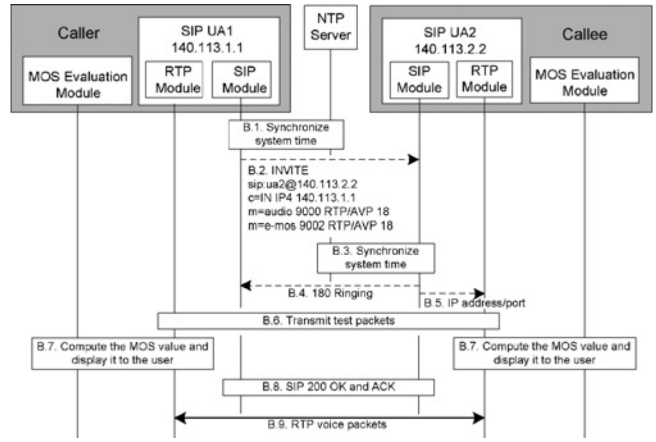

Fig. 6. Message flow of E-model-based MOS measurement (dashed: SIP and UA internal control messages; solid: RTP packets).

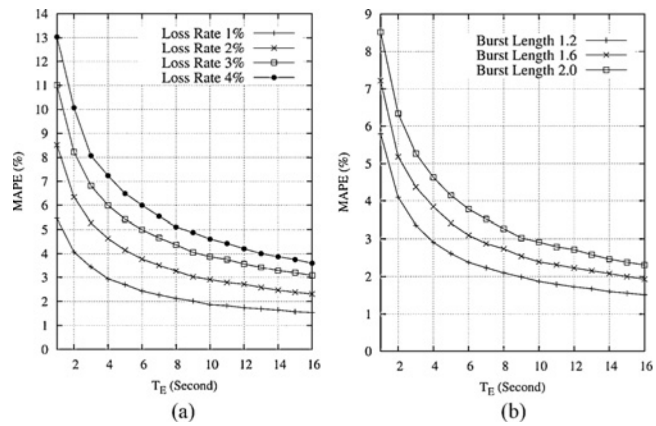

Fig. 7. MAPE of the E-model-based MOS measurement (N = 2000). (a) L_B = 2. (b) P_L = 2%.

9) *Step B.9*: If the call is connected, the voice RTP session is established between the RTP modules of UA1 and UA2.

B. Experimental Results

In the above procedure, it is necessary to select an appropriate T_E value such that accurate MOS value is obtained within a short period of time. Similar to the definition of $M_P(T_P)$, $M_E(T_E)$ is the MOS value computed by using the E-model-based MOS measurement procedure, where the test packets are transmitted for T_E s. The "accurate" E-model MOS value is defined as

$$M_E = \lim_{T_E \to \infty} M_E(T_E).$$

In our experiment, $M_E \approx M_E(10000)$. Although the definition of M_E is different from that for M_P, both of them are the "exact" values computed in the traditional approaches [12]. The definitions of $\delta(T_E)$, MAPE(T_E), and N are the same as those described in Section II-B. Fig. 7 shows that MAPE(T_E) significantly drops when T_E increases from 1 s to 10 s. For $T_E > 10$ s, increasing the number of test packets, MAPE(T_E) is only insignificantly improved. Fig. 7 also indicates that for $L_B = 2$ with various P_L values, or $P_L = 2\%$ with various L_B values, MAPE(T_E) $< 7\%$ for $T_E > 5$ s.

At Steps B.1 and B.3 in Fig. 6, the NTP synchronization takes 0.35 s for a device with ARM Cortex-A8 600 MHz in our configuration. Therefore, if $T_E = 5$ s, the total execution time of the E-model-based MOS measurement is 5.35 s, which is a reasonable cost to provide real-time VoIP quality measurement.

IV. Conclusion

Several approaches have been proposed to measure the VoIP quality (the MOS value) of the voice path between two call parties. These approaches require involvement of extra servers/gateways to conduct heavy MOS computation, which may not be appropriate for mobile all-IP network where mobile devices with limited computation resources move around the network. This paper proposed two MOS measurement procedures for mobile devices. In the PESQ MOS measurement procedure, a lightweight real-time table lookup solution significantly reduced the computation time of PESQ MOS from 315.4 s to 3 s. In the E-model MOS measurement procedure, MOS value can be accurately computed in 5.35 s. These results indicated that the proposed approaches can effectively provide the real-time VoIP quality measurement.

The two MOS measurement approaches proposed in this paper can be integrated. If the mobile devices in the voice path exercise different MOS measurement procedures (i.e., one for PESQ and another for E-model), our approach still worked if the mobile device running PESQ algorithm also included the sending time in the payload of each test packet delivered to the other device (running E-model algorithm). The test packets were transmitted for max(T_P, T_E) = 5 s. As a final remark, the proposed approaches are pending Taiwan and U.S. patents.

References

[1] ITU-T. *Methods for Subjective Determination of Transmission Quality*, document Rec. P.800, International Telecommunication Union-Telecommunication Standardization Sector, Jun. 1996.

[2] Y.-F. Chen, T. Li, D.-N. Gao, X. Hu, X.-P. Zhang, and J. Liu, "A secure mobile communication approach based on information hiding," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 2, no. 3, pp. 134–139, 2007.

[3] ITU-T. *Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-end Speech Quality Assessment of Narrow-band Telephone Networks and Speech Codecs*, document Rec. P.862, International Telecommunication Union-Telecommunication Standardization Sector, Feb. 2001.

[4] ITU-T. *The E-Model: A Computational Model for Use in Transmission Planning*, document Rec. G.107, International Telecommunication Union-Telecommunication Standardization Sector, Mar. 2005.

[5] Y.-B. Lin, Y.-C. Sung, H. F. Rashvand, C.-L. Liu, and Y.-J. Lia, "Mobile-Taiwan experience in voice over IP-worldwide interoperability for microwave access trial," *IET Commun.*, vol. 4, no. 9, pp. 1130–1141, 2010.

[6] A. D. Berstein, M. Segal, W. Wang, and I. Yuval, "System for real time voice quality in voice over packet network," U.S. Patent 7 197 010, Mar. 27, 2007.

[7] L. Foodman, "Embedding sample voice files in voice over IP (VoIP) gateways for voice quality measurements," U.S. Patent 7 280 487, Oct. 9, 2007.

[8] Y.-B. Lin and A.-C. Pang, *Wireless and Mobile All-IP Networks*. New York: Wiley, 2005.

[9] *Technical Specification Group Services and Systems Aspects: IP Multimedia Subsystem Stage 2. Technical Specification 3G TS 23.228 Version 10.3.1 (2011-1)*, 3GPP Partnership Project, 2011.

[10] *SIP: Session Initiation Protocol*, document IETF RFC 3261, Internet Engineering Task Force, Jun. 2002.

[11] Y.-H. Chan, "VoIP quality measurement for multiple codecs," M.S. thesis, Nat. Chiao Tung Univ., Hsinchu, Taiwan, Jul. 2007.

[12] IxChariot [Online]. Available: http://www.ixchariot.com/products/datasheets/ixchariot.html

[13] Y.-C. Sung, Y.-B. Lin, R.-H. Liou, and L.-F. Shieh. (2010 May). NCTU-VT: A freeware for wireless VoIP performance measurement. *Wireless Commun. Mobile Comput.* [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/wcm.963

[14] E. N. Gilbert, "Capacity of a burst noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1266, Sep. 1960.

[15] *Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP)*, document Rec. G.729, International Telecommunication Union-Telecommunication Standardization Sector, Jan. 2007.

[16] *RTP: A Transport Protocol for Real-Time Applications*, document IETF RFC 3550, Internet Engineering Task Force, Jul. 2003.

[17] J.-R. Lin, A.-C. Pang, and Y.-C. Wang, "iPTT: Peer-to-peer push-to-talk for VoIP," *J. Wireless Commun. Mobile Comput.*, vol. 8, no. 10, pp. 1331–1343, 2008.

[18] *SDP: Session Description Protocol*, document IETF RFC 4566, Internet Engineering Task Force, Jun. 2006.

[19] C. D. Lewis, *Industrial and Business Forecasting Method*. London, U.K.: Butterworth Scientific, 1982.

[20] L. Sun and E. C. Ifeachor, "Voice quality prediction models and their application in VoIP networks," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 809–820, Aug. 2006.

[21] *Network Time Protocol (Version 3) Specification: Implementation and Analysis*, document IETF RFC 1305, Internet Engineering Task Force, 1992.

Whai-En Chen (M'99) was a Research Assistant Professor with National Chiao Tung University, Hsinchu, Taiwan, from 2002 to 2007. From August 2007 to 2009, he was an Assistant Professor with the Institute of Computer Science and Information Engineering (CSIE) and the Chief of the Network Division of the Computer and IT Center, National Ilan University, Ilan, Taiwan. Since February 2010, he has been an Associate Professor and the Director of the Institute of CSIE. His current research interests include Internet protocol (IP) multimedia subsystems, voice over IP, IPv6, mobility management, and IEEE 802.16.

He is a member of ACM and a Voting Member for IEEE 802.16.

Pin-Jen Lin received the B.S. and M.S. degrees from National Chiao Tung University (NCTU), Hsinchu, Taiwan, in 2006 and 2008, respectively. He is currently pursuing the Ph.D. degree from NCTU.

His current research interests include 3G Internet protocol multimedia subsystems, SIP-based VoIP services, and IPv6 translation mechanisms.

Yi-Bing Lin (M'96–SM'96–F'03) is the Vice President and Chair Professor with the College of Computer Science, National Chiao Tung University (NCTU), Hsinchu, Taiwan, and a Visiting Professor with King Saud University, Riyadh, Saudi Arabia. He is the author of the books *Wireless and Mobile Network Architecture* (New York: Wiley, 2001), *Wireless and Mobile All-IP Networks* (New York: Wiley, 2005), and *Charging for Mobile All-IP Telecommunications* (New York: Wiley, 2008).

He received numerous research awards including the National Science Council Distinguished Researcher Award in 2005 and the Academic Award of the Ministry of Education in 2006. He is a fellow of ACM, AAAS, and IET.