Performance metrics and configuration strategies for group network communication

Tom Z. J. Fu   Dah Ming Chiu
Department of Information Engineering
The Chinese University of Hong Kong
Email: {zjfu6, dmchiu}@ie.cuhk.edu.hk

John C. S. Lui
Computer Science & Engineering Department
The Chinese University of Hong Kong
Email: cslui@cse.cuhk.edu.hk

***Abstract*—There is an increasing number of group-based multimedia applications over the Internet, for example, voice conference or multi-player games. For these applications, it is often necessary to select a strategy to distribute the multimedia streams or mixing the multimedia stream data so as to provide better quality of service (QoS) guarantees. However, there is no appropriate metrics to evaluate the QoS of a group multimedia session, despite abundant literature on how to evaluate the QoS for two-party communication (e.g. MOS, E-Model). In this paper, we propose a new measure which is called the *group mean opinion score* (GMOS). To leverage on existing work, our definition of GMOS is based on two-party MOS, hence, it can be estimated via measurement of network parameters and fitting these data into the E-Model. We conduct large scale experiments using the latest SKYPE conference software. We first calibrate the GMOS based on the subjective scores of our experiments, then for individual conference sessions, we check whether our approach can pick a server configuration strategy to achieve the best GMOS. The study shows our proposed methodology is very promising and the potential of applying to other group-based applications.**

I. Introduction

For two-party multimedia applications, there is extensive literature on how to characterize, measure and model the performance of such applications [1]–[9]. In particular, the performance of two-party multimedia applications can be evaluated at a subjective level, using *mean opinion score* (MOS [1]). Various factors that affect performance are identified, such as the type of codecs used and other measurable network conditions like loss rate and round-trip time. Through extensive measurement studies and comparison with MOS data, a model is developed to predict MOS based on measurable parameters. One well establish model is the E-Model [2].

More recently, multi-party multimedia applications have also become popular. One example is the support for conferencing in the recently released version of SKYPE [29]. However, there is no standard metric to characterize the performance of a conferencing application (or multi-party application). The motivation for us to seek a performance metric is due to the consideration of how to configure a communication and mixing strategy for a conferencing application. In a peer-to-peer implementation of group communication [10], a strategy is needed so as to select a particular peer as the server. Unless there is a performance metric to compare the difference of using different peers as servers, it is not clear how the choice can be systematically made.

Motivated by this problem, we explore how to define a *group-based* MOS metric for all parties - we call this the *group mean opinion score* (GMOS). Following the same framework of the two-party paradigm, one needs to have both a subjective measure as well as a connection to physically measurable parameters, such as network delay and loss rates between different peers. In the end, this new framework should allow us to make configuration decisions that is expected to yield the best subjective evaluation. Parallel to the E-model case, we also develop a mapping between the subjective GMOS and the measurable parameters as well as the configuration decisions.

In this paper, we first propose a GMOS equation based on MOS between bilateral communications. This GMOS model includes a parameter that can be used to calibrate the model for specific applications and users. We conduct SKYPE conferencing experiments to see whether the model can be consistently applied to multiple experiments with the same application and user population. The result gives us some idea of how to calibrate the model parameter, α.

Secondly, we develop a *two-step mapping method*(TSMM) to predict GMOS based on measurable network parameters and the server selection decision. The first step is to measure network parameters (delay and loss) and apply E-Model to find out MOSes for each bilateral session. The second step is to use our calibrated GMOS model to predict the subjective evaluation for different leader (server) selections. Finally, we compare our predication to actual scores given by users. Our conclusion is that our GMOS model and the leader selection approach is able to produce good decisions.

The outline of the paper is as follows. In Section II, we present the definition of GMOS and results from our experiments to support this measure. In Section III, we illustrate various topologies used to support the voice conference and propose the *leader selection strategy*(LSS) based on the end system mixing topology. We then describe our experimental settings and measurements in Section IV. In Section V, we present the *two-step mapping method*(TSMM) and analyze the effectiveness of this proposal. We discuss the applications of our proposed methodology in section VI. Related work is given in Section VII, Section VIII concludes.

II. Performance metrics of network Voice Conference: $GMOS$

When users participate in a voice conference session, they will hear more than one speaker's voice. The voice quality of different speakers will vary depending on the heterogeneous network conditions between the listener and the speakers. Assume that the session has N participants where participant i is denoted as $\mathcal{P}_i$, $i \in \{1, \ldots, N\}$. $\mathcal{P}_i$ will provide $N-1$ MOS scores [1] for other participants and these MOS scores represent the audio quality of these participants from $\mathcal{P}_i$'s perspective. Additionally, $\mathcal{P}_i$ has a score to reflect the overall quality of the conference session, and this is our proposed group mean opinion score (GMOS). We propose to use GMOS to relate the MOS scores of other participants as well as a subjective measure on the group audio quality. Formally, the GMOS of $\mathcal{P}_i$ is:

$$GMOS_i\left(MOS_i(1), \ldots, MOS_i(N), \alpha\right) = AVE + \alpha(AVE - MIN)U(-\alpha) + \alpha(MAX - AVE)U(\alpha), \quad (1)$$

where $MOS_i(k)$ is the MOS score set by $\mathcal{P}_i$ for $\mathcal{P}_k$, and

$$\begin{aligned} AVE &= \frac{\sum_{k=1}^{N-1} MOS_i(k)}{N-1}, \\ MAX &= \max\left\{MOS_i(1), \ldots, MOS_i(N)\right\}, \\ MIN &= \min\left\{MOS_i(1), \ldots, MOS_i(N)\right\}, \\ \alpha &\in [-1, 1], \\ U(x) &= \begin{cases} 1 & x > 0, \\ 0 & x \leq 0. \end{cases} \end{aligned}$$

Note that participant $\mathcal{P}_i$ can use the parameter α to control his subjectivity on the quality of the group communication. For example, when $\mathcal{P}_i$ sets $\alpha = -1$ (or $\alpha = 1$), the $GMOS_i$ will be equal to MIN (or MAX). When $\mathcal{P}_i$ sets $\alpha = 0$, $GMOS_i$ will be equal to AVE. In other words, if $\mathcal{P}_i$ feels that the conference quality is defined by the minimum (or maximum) MOS of other participants, $\mathcal{P}_i$ will set α to -1 (or 1). If $\mathcal{P}_i$ feels that the conference quality is defined by the average of $N-1$ MOS scores, he will set $\alpha = 0$. In practice, different values of α represent different subjective view of $\mathcal{P}_i$ on the overall quality of the group conference.

TABLE I
MOS, GMOS and Meanings

MOS	GMOS	Meaning
5	5.0	Excellent
4	4.0	Good
3	3.0	Fair
2	2.0	Poor
1	1.0	Bad

Based on the ITU-T P.800 recommendation, the MOS is an integer between 1 to 5. Table I provides the physical meaning of each value of MOS [1], [11]. For GMOS, we believe it is appropriate to represent it also by a number between 1.0 and 5.0, but we relax the integer constraint. By making it a real number with one decimal place has enough resolution to reflect QoS measure. Note that we need more resolution that GMOS is affected by more parameters than MOS. Other than the resolution, the physical meaning of GMOS is very similar to MOS (refer to Table I). To evaluate the effectiveness of the proposed GMOS, we first test the GMOS formula using some real life experiments. The detail of the experiment settings will be described in a later section. Here we just show the figures and discuss the meaning of the results. To carry out the experiment, we invited some people to have audio conferences via SKYPE [29]. All the voice contents were recorded using standard recording software. A total of 18 sets of experiments were carried out during January of 2007. Out of the 18 conferences, three were 3-person conferences, ten were 4-person conferences and five were 5-person conferences. Subsequently, we invited 25 subjects to listen to and give subjective scores to these 18 records. Every time they finished one record, they first gave MOSes to all the speakers who appeared in that experiment (one MOS for each speaker), and then they gave a subjective GMOS (between $1.0-5.0$) to express their satisfaction on the overall quality of the voice conference record. Consider a 4-person conference as an example. Each subject would have given five scores, four of which are integer MOSes towards the quality of individual participants and one for the subjective GMOS score, which reflects his opinion on the overall quality of the voice conference.

On collecting all these scores, we calculate the α value through the MOSes and GMOS according to Eq (1). The total number of the MOSes and GMOS pairs are 437 including all the three types of conference: 3-person, 4-person and 5-person. Based on Eq. (1), we can determine 392 α values. Figure 1 is the frequency distribution of the computed α.

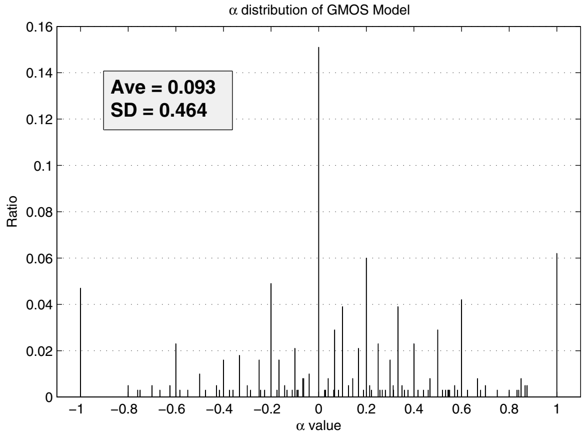

Fig. 1. α distribution

We make the following observation.

a) In our GMOS model, we implicitly assume that GMOS should be a number between the minimum and the maximum of the MOSes. As we have explained before, when $\alpha \in [-1, 0)$, this implies that the user concerns more about the worst case performance, or they are "pessimistic" about the overall conference quality. Consequently, the GMOS is below the average and it is between the AVE and MIN value of the MOSes. When $\alpha \in (0, 1]$, it implies that the user concerns more about the best case performance, or they are "optimistic" about the overall conference quality. The results of the experiment reveal that in reality, a certain percentage of people are "very pessimistic" or "very optimistic" , which means that their GMOS scores will either be smaller than the minimum or larger than the maximum value of the MOSes. In our experiment, this proportion is $(437 - 392)/437$, or $\approx 10\%$.

b) From Figure 1, one can observe that a large proportion of subjects think that GMOS should be the average (15%), or very close to the average of MOSes (summing up the ratio of $\alpha \in [-0.2, 0.2] \approx 50\%$). As a result, we set the default value of α to be 0.

c) The average value of all the 392 α samples is 0.093, and it is larger than 0 (the default value). Statistically, it indicates that these subjects are more "optimistic" on average. Another possible interpretation is that the value of α is application dependent. The average value of $\alpha = 0.093$, obtained by an experiment using SKYPE, might be specifically applicable to SKYPE, or audio conference only, but not necessarily for other group-based multimedia applications.

III. Conference Leader Selection strategies

In this section, we make use of the GMOS model and take a closer look at the QoS issues of voice conference application. In particular, we seek to answer the following question: is it possible to improve the overall quality of a voice conference session via some configuration strategies?

Several types of topologies supporting multi-party voice conference are discussed in [15], e.g., end system mixing, conference server mixing, full mesh and combination of conference servers and full mesh. Figure 2 illustrates four types of the topologies [15].

We will mainly focus on the "*end system mixing*" topology illustrated in Figure 2. Following are some justifications of choosing this type of topology:

- It is the simplest and easiest to implement among the four topologies.
- It does not require a dedicated server to hold the conference.
- It requires far less bandwidth than the full mesh topology.
- In an overlay peer-to-peer network scenario, the end system mixing topology is more effective and suitable as compared with the other three topologies.

Note that the main disadvantage of the end system topology is the heavy loading on the leader or the media stream mixer (the "A" node in the top-right of Figure 2). Based on the end system mixing topology, one of the conference participants should be the conference leader who is in charge of establishing and starting the conference, inviting and adding participants to the conference and also ending it. Also, the computation of mixing and forwarding media streams is carried out by this leader. It implies that the leader performs many essential functions and it will greatly affect the overall quality of voice conference.

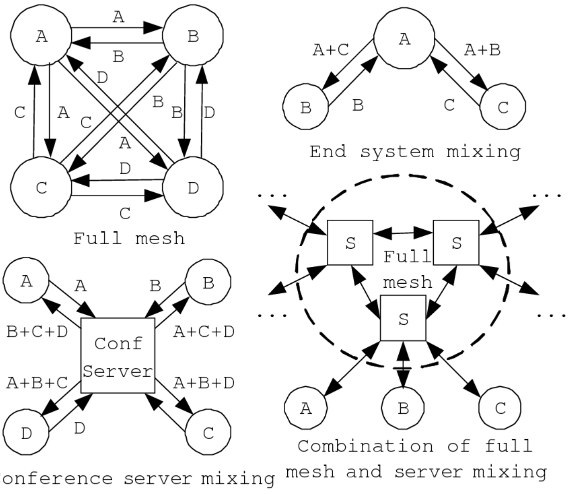

Fig. 2. Illustration of four types of voice conference topologies [15]

In Section II, we mentioned about the voice conference experiments using SKYPE [29] and the three types of experiments: 3-person, 4-person and 5-person conference call. We use Ethereal [30] to collect all packets from computers of all participants including the leader. The traffic from measurement indicates that the topology of SKYPE conference is indeed an end system mixing topology. Figure 3 shows three types of topologies of our conference experiments. The leader is the one that determines the overall QoS and there is no traffic between non-leader participants, i.e., their traffic has to be relayed via the leader.

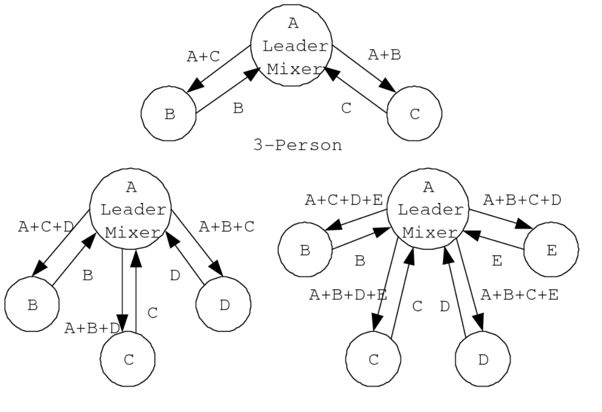

Fig. 3. Topology of 3-person, 4-person and 5-person SKYPE conference

Since every participant will have a GMOS to show their subjective view on the overall quality of the voice conference, we need to categorize GMOS into conference leader's GMOS (denoted as GMOS_L) and non-leader participants' GMOS (denoted as GMOS_M). Furthermore, we assume that the leader's GMOS_L, which is the subjective assessment towards each non-leader participant from the leader's perspective, is the representation of the subjective evaluation of all non-leader participants towards the overall quality of the voice conference. In other words, this implies that the leader's opinion can represent other non-leader participants' opinion on evaluating the overall conference quality.

Based on the assumption and discussions above, we propose the *leader selection strategy*(LSS) of properly selecting the conference leader to improve the overall quality of the voice conference. Suppose there are N participants in a voice conference. Each of them being the leader once in turn, they will get a GMOS$_{L_i}, (i = 1, 2, \ldots, N)$, and finally a total of N GMOS$_L$. Each of the GMOS$_{L_i}, (i = 1, 2, \ldots, N)$ represents the overall quality of the N voice conference under the condition that participant i being the conference leader. The *leader selection strategy*(LSS) is to select participant i whose GMOS$_{L_i}$ is the largest among all the GMOS$_{L_i}, (i = 1, 2, \ldots, N)$ to be the conference leader. Also, participant i, after being selected as the conference leader, his/her GMOS$_{L_i}$ should satisfy the following equation:

$$GMOS_{L_i} = \arg \max_{k=1,2,\ldots,N} GMOS_{L_k}. \quad (2)$$

Asking participants to provide GMOS on the overall quality of voice conference is a subjective test and it is difficult to obtain the GMOS before or during the conference. Instead, we first estimate the MOS from the network traffics (e.g., packet loss rate, jitter, codec,..etc) during a voice session. We also need to estimate GMOS. We propose the *two-step mapping method*(TSMM) to estimate the leader and participants' GMOSes. This will be describe in Section V.

IV. Experiment Description

Our experiments are separated into two parts. In the first part, the aim is to validate the GMOS(MOSes, α) and to determine the default and the average values of α. These experimental results were shown in Section II. In the second part of the experiment, it is to verify the effectiveness of the *leader selection strategy*(LSS) proposed in Section III.

In both parts of the experiments, the network setting is represented in Figure 4. All computers in the experiments are equipped with Intel P4 CPUs and at least 512M Memory with 10M/100M Ethernet network card. Three of these computers are located in Shanghai China, accessing the Internet via ADSL of Shanghai Telecomm while the other computer is in Shanghai assessing the Internet through the campus LAN in the Shanghai Jiao Tong University. The computer in Hong Kong is connected through the campus LAN in the Chinese University of Hong Kong. These computers are installed with SKYPE [29] (Version 2.5), professional audio recording and processing software Audition [31] (Version 1.5), and measurement tool Ehtereal [30] (Version 0.99).

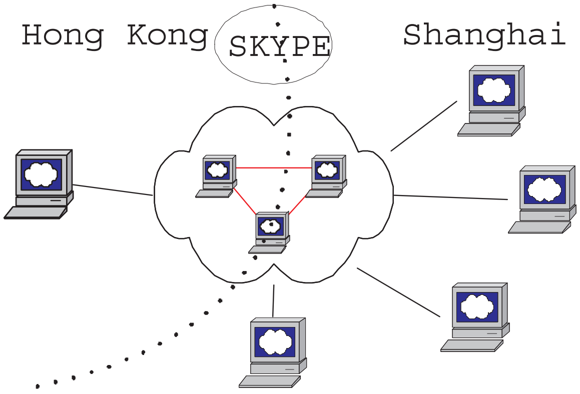

Fig. 4. Network setting of SKYPE conference

In the first part of the experiment, we asked people to have voice conference using SKYPE [29] and every participants used Audition [31] to record voice of the communication session. The duration of each conference was around 30 seconds per person, e.g., if there were four persons in the conference, it would last for about two minutes. The reason why we set the length of recording longer than that of the voice session described in [1] is that we consider the subjects who are invited to listen to the records and give the MOSes (to each individual speaker) and GMOS (to the whole conference) need more time to distinguish different speakers' voices.

In the second part of the experiments, we not only asked participants to have voice conference and performing the recording work, but also organized them to change the conference leader in turn. For instance, when we are going to have a 3-person conference, e.g., Alice, Bob and Cathy. We will ask them to do three experiments in which each of them being the conference leader once in turn. Lastly, the group of N experiments for our second part were carried out in short duration so that these experiments could be considered as operating under the same traffic condition and the results could be more accurate.

In order to evaluate the *two-step mapping method*(TSMM) and further validate the *leader selection strategy*(LSS) proposed in section III, we need to measure some network parameters during the voice conference. Through packet trace by Ethereal [30], one can obtain the statistics such as *bit rate*, *jitter*, *loss rate* (under random loss model) and *loss rate* and *state transfer probability* (under 2-state Markov loss model [13]). We perform the Ping-like measurements between the leader and each non-leader participant separately during the conference to obtain the *Round Trip Time*(RTT) so that we can estimate the one way delay from the measured RTT. These measured and calculated statistics are inputs to the E-Model [2]. The frequency of the Ping-like measurement is set to 1 Hz, which is low enough not to affect the normal traffic generated by SKYPE and this is performed at the leader's computer to each non-leader participant's computer.

V. Data analysis and results

The results of the first part of the experiments are for GMOS model and parameter α, and we described them in Section II already. Here, we concentrate on the second part of the experiments: to evaluate the *two-step mapping method*(TSMM) and validate the *leader selection strategy*(LSS) that we presented in previous section.

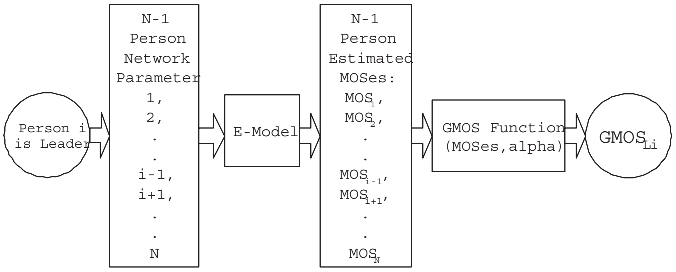
Person i is Leader
N-1 Person Network Parameter 1, 2, . . i-1, i+1, . . N
E-Model
N-1 Person Estimated MOSes: MOS_1, MOS_2, . . MOS_{i-1}, MOS_{i+1}, . . MOS_N
GMOS Function (MOSes,alpha)
$GMOS_{Li}$

Fig. 5. Architecture of the *two-step mapping method*(TSMM)

The *two-step mapping method*(TSMM) is used to calculate the estimated GMOS with inputs of network parameters between the leader and non-leader participants. Figure 5 depicts the architecture of the *two-step mapping method*(TSMM).

The first step of the *two-step mapping method*(TSMM) is to map the measured network parameters to estimate MOS. For this step, we apply the E-Model [2] since it is widely used for two-party communication [12], [13], [20]–[24].

For the E-model, the inputs are different classes of impairments which will affect the output R-value. The R-value is a real number between 1 to 100, which represents the quality of the 2-party communication. Some classes of the impairments are related to network condition such as the *effective equipment impairment factor* ($I_{e,eff}$), which comprises the effect of voice codec and packet loss, and the *delayed impairment factor* (I_d). Other impairments are not related to network conditions, for example, R_0, the initial value representing the *signal-to-noise ratio*(SNR) quality and I_s, which accounts for the effect of simultaneous problem etc. Based on the proposal in [16], we take default values for all impairments parameters defined in [2] except for $I_{e,eff}$ and I_d since they are related to network conditions. The E-Model is then reduced to:

$$R = 93.2 - I_{e,eff} - I_d. \tag{3}$$

Once we know the value of R, we can find the MOS value via:

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

The remaining issue is to estimate $I_{e,eff}$ and I_d. As $I_{e,eff}$ is defined to comprise the effect of voice codec and packet loss [2], [13], we need to find the codec SKYPE uses before estimating $I_{e,eff}$. From [17]–[19], one will find that SKYPE uses iLBC [25], [26] or iSAC [27] codec, both of which are the products of GlobalIPSound [28]. Authors in [12] propose a method to calculate $I_{e,eff}$, which is suitable for us because it proposes how to set the constants in the formula when the speech codec is iLBC [25], [26]. In summary, $I_{e,eff}$ is

$$I_{e,eff} = I_e + (95 - I_e)\frac{Ppl}{Ppl + Bpl}. \tag{5}$$

TABLE II
SETTINGS OF I_e AND Bpl FOR DIFFERENT CODECS [13]

Codec	Tp[ms]	PLC	I_e	$Bpl[\%]$
G.711	10	Sil.Ins	0	4.3
G.711	10	App.I	0	25.1
G.729A	20	Native	11	19.0
iLBC	30	Native	11	32.0

In Equation (5), I_e and Bpl are two constants derived by authors in [12]. Table II shows part of the settings of the constants of different codec [13]. Here, Bpl is the packet loss robustness factor, i.e. the larger the Bpl, the smaller the effect of packet loss on the audio session. Ppl is the packet loss percentage which is expressed as

$$Ppl = ppl * 100\% \tag{6}$$

and ppl is the packet loss probability. For the parameter ppl, one can estimate it as an independent loss probability under the assumption that packet loss is a "random loss". Alternatively, one can use a 2-state Markov loss model (or Gilbert model) [13] to model the bursty loss scenario. For the 2-state Markov loss model, it has two parameters p and q, with p being the transition probability from good state to loss state, while q is the transition probability from loss state to good state, and $pc = 1 - q$ is the conditional loss probability.

The model to calculate $I_{e,eff}$ under the 2-state Markov loss model is also proposed in [13]:

$$\begin{cases} I_{e,eff} &= I_e + (95 - I_e)\frac{Ppl}{\frac{Ppl}{BurstR} + Bpl}, \\ BurstR &= \frac{1-pc}{1-ppl}. \end{cases} \tag{7}$$

The $BurstR$ is the burst ratio of loss. If $BurstR < 1$, it means that when the previous packet was lost, it has a lower probability of losing the current packet. When $BurstR = 1$, it represents "random loss" and when $BurstR > 1$, it implies bursty loss.

We obtain the values of ppl, pc and $BrustR$ by analyzing the traffic data. We observe that the cases that $BurstR > 1$ occur with low frequency, and the measured $BrustR$ is slightly larger than 1, which implies that the packet loss scenarios during our experiments were mainly from random losses. Therefore, we apply Equation (5) to obtain the $I_{e,eff}$.

To estimate the impairment I_d, we use two approaches. One is letting I_d be the default value defined by [2] and ignoring end-to-end delay. The other one is to apply the formula with inputs RTT, which was obtained by our ping-like measurements. We calculate the impairment I_d via:

$$\begin{aligned} I_d &= 0.024d + 0.11(d - 177.3)V(d - 177.3), \\ d &\approx RTT/2, \\ V(x) &= \begin{cases} 0 & x < 0, \\ 1 & x \geq 0. \end{cases} \end{aligned} \tag{8}$$

Since the delay condition will have different effects on systems implemented by different buffer strategies, and the mechanism that SKYPE uses to deal with end-to-end packet delay is unknown, we use both approaches to estimate I_d:

$$\begin{cases} R = 93.2 - [I_e + (95 - I_e)\frac{Ppl}{Ppl+Bpl}] - I_d(\text{default,}) & \text{(a)} \\ \\ R = 93.2 - [I_e + (95 - I_e)\frac{Ppl}{Ppl+Bpl}] & \\ \qquad -[0.024d + 0.11(d - 177.3)V(d - 177.3)]. & \text{(b)} \end{cases}$$

After obtaining the R-value using function (a) and (b) listed above, one can derive two different MOS values of the same pair of network parameters by applying Equation (4) with two different R values from (a) and (b). We call the process of mapping network parameters to MOS value as the first-step mapping function.

The second step of the *two-step mapping method*(TSMM) is to apply the estimated MOSes obtained from first-step mapping function into Equation (1) so as to get the estimated GMOS. Since $\alpha = 0$ is the default value and $\alpha = 0.093$ is the average of all 392 samples obtained from our experiments. We experiment with these two values in the second-step mapping function. As a result, we have four types of the combination of applying the *two-step mapping method*(TSMM). Table III summarizes these four types. Note that these four types are actually objective methods to estimate the MOSes and GMOS given by various participants. So in order to evaluate and analyze the performance of the *two-step mapping method*(TSMM) and the *leader selection strategies*(LSS), we invite participants to listen to the audio clips which were recorded from the conference leader's computer and to provide MOSes and GMOS scores just like what have been done in the first part of the experiments. The results are shown in Figure 6 to Figure 8.

TABLE III
SUMMARY OF 4 TYPES OF THE *two-step mapping method*(TSMM)

Method	R fitting model	α
M1	(a)	0.093
M2	(b)	0.093
M3	(a)	0
M4	(b)	0

Figure 6 shows the estimated MOS by the first-step mapping function (a) with the MOS which is actually the average value of all the integer MOSes (1 to 5) scored by subjects. Figure 7 shows the estimated MOSes by the first-step mapping function (b) with subjective MOSes. Here, we use the following measure to analyze the mapping results:

$$\text{Average}(\Delta y) = \frac{\sum_{i=1}^{N} |\hat{y}_i - y_i|}{N} \qquad (9)$$

In Equation (9), y_i is the value of mapping results and $\hat{y}_i$ is the expected value with the same x_i. For results in Figure 6 and Figure 7, the expected values are on the line of $y = x$ because the mapping is from MOS (E-Model [2], objective) to MOS (Subjective). Through Equation (9), we derive the average(Δy) of fitting function (a) is 0.4282 and average(Δy) of fitting function (b) is 0.4755.

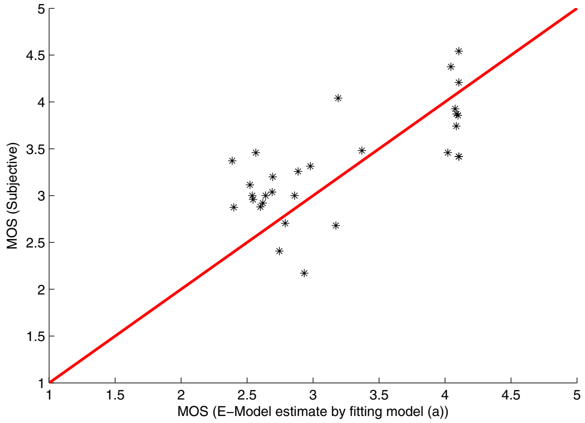

Fig. 6. Mapping from MOS (E-Model) of first-step mapping model (a) to MOS (Subjective)

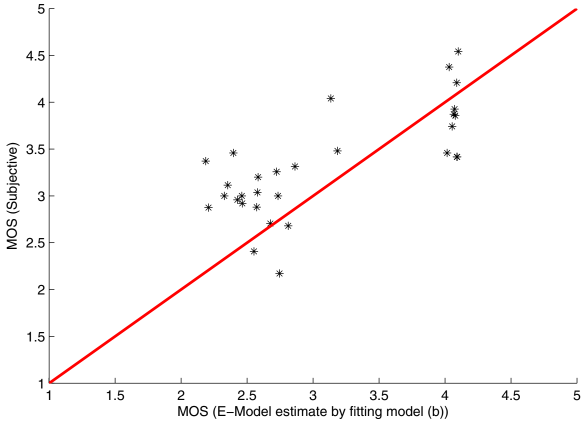

Fig. 7. Mapping from MOS (E-Model) of first-step mapping model (b) to MOS (Subjective)

Figure 8 illustrates the performance of these four types of *two-state mapping method*(TSMM). We apply the average(Δy) in Equation (9) again. Here, y_i is the result obtained by applying the four approaches of the *two-step mapping method*(TSMM) and $\hat{y}_i$ is the subjective GMOS. One can observe from Table IV that the difference between GMOS scored by subjects and that obtained by four approaches of *two-step mapping method*(TSMM) is very small. It means the objective method we proposed (TSMM) to estimate the subjective GMOS works quite well.

TABLE IV
ILLUSTRATION OF AVERAGE(Δy)

	M1	M2	M3	M4
Average(Δy)	0.1948	0.1486	0.1712	0.1417

Next, we check the correctness of the *leader selection strategy*(LSS) proposed in Section III. Table V illustrates the GMOS given by subjects and the estimated GMOS derived

TABLE V
ILLUSTRATION OF LEADER SELECTION RESULTS OF GMOS (SUBJECTIVE) AND ESTIMATED GMOS BY 4 TYPES OF THE *two-step mapping method*(TSMM)

Index	N	Person	Leader	GMOS(Sub)	GMOS(M1)	GMOS(M2)	GMOS(M3)	GMOS(M4)
1	3	'A', 'B', 'C'	'A'	2.637	2.945	2.853	2.917	2.815
2	3	'A', 'B', 'C'	'B'	3.541*	3.597*	3.541*	3.548*	3.486*
3	3	'A', 'B', 'C'	'C'	3.293	3.507	3.471	3.450	3.411
4	4	'A','B','C','D'	'A'	3.065	3.258	3.106	3.231	3.064
5	4	'A','B','C','D'	'B'	3.900*	3.866*	3.824*	3.806*	3.760*
6	4	'A','B','C','D'	'C'	3.295	3.647	3.614	3.600	3.565
7	4	'A','B','C','D'	'D'	3.523	3.403	3.351	3.331	3.275
8	4	'J','K','L','M'	'J'	3.152	3.333	3.227	3.246	3.129
9	4	'J','K','L','M'	'K'	3.162	3.699	3.628	3.657	3.581
10	4	'J','K','L','M'	'L'	3.600*	3.714*	3.665*	3.674*	3.622*
11	4	'J','K','L','M'	'M'	3.536	3.570	3.509	3.517	3.451
		Correct / Total		–	3/3	3/3	3/3	3/3

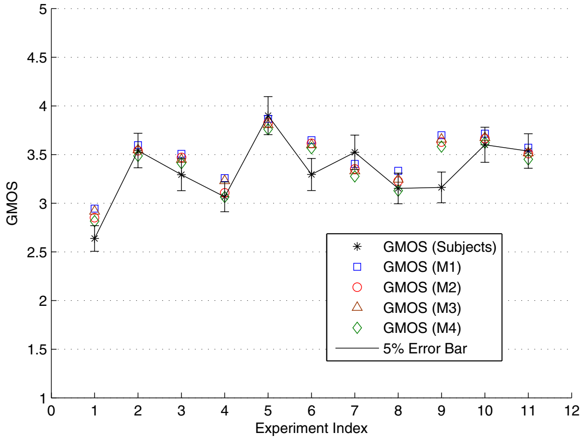

Fig. 8. 4 types of Two-State Mapping Method with GMOS (Subjective)

from these four types of *two-step mapping method*(TSMM). These results are arranged into experiment groups. In order to check whether the selection of the leader by the *leader selection strategy*(LSS) is efficient or not, we perform an N-person conference experiment through SKYPE for N times so that each participants has the chance to be a leader for one time. This N times experiment with the same participants are called the experiment group. We consider the participant who receives the largest GMOS$_{L_i}$ given by all subjects in condition that he is the leader, is the correct server to be the leader. If any of these four types selects the same person as the leader, this shows a correct selection, otherwise an incorrect selection was made.

In summary, we have carried out three groups of conference experiments, two of which are 4-person conference and one is a 3-person conference. The GMOS$_L$ value in Table V with a star "$*$" is the largest value of GMOS$_L$ in each group. The person with the largest GMOS$_L$ should be the leader according to the *leader selection strategy*(LSS). The leader selected by the "GMOS (Sub)" column in which the GMOS$_L$ is given by subjects is considered as the correct one because it comes from subjective tests. "GMOS$_L$" in the last four columns comes from the four types of applying the *two-step mapping method*(TSMM) to estimate the GMOS$_L$ and selects the leader by the *leader selection strategy*(LSS). The results show that all the four types of applying the two-step mapping method have selected the *correct leader* in each group of conference experiments.

VI. APPLICATIONS OF PROPOSALS TO VOICE CONFERENCE

As we have discussed in the previous sections, our GMOS metric can be implemented to evaluate the overall quality of voice conference so that we can know how good the providing service is, and whether it can be improved from the provider view as well. The GMOS we propose can be used to evaluate many voice quality applications, e.g., USI [33].

The *leader selection strategy*(LSS) we propose can be applied in several ways:

a) Before N persons start a voice conference, the software can measure traffic between any two participants so as to estimate the network parameters. Then it can utilize the *two-step mapping method*(TSMM) to estimate $GMOS_{L_i}$ and then apply the *leader selection strategy*(LSS) to select the proper leader.
b) During a voice conference, the $GMOS_{L_i}$ could be an indicator to reveal the overall quality of the whole conference. And the software can maintain a light-weight testing traffic to select a leader candidate by the *leader selection strategy*(LSS) among the $N-1$ non-leader participants so that when the current conference gets disconnected due to some unforeseen network condition, it can restart with a new and proper conference leader.

VII. RELATED WORK

E-Model (ITU-T Rec. G.107 [2]) is designed to be a non-intrusive parametric model to estimate the subjective MOS (ITU-T P.800 [1]). Number of works have focused on the implementation and extension of E-Model. COLE et al. [14] propose to simplify the E-Model base on only two network related impairments, e.g., I_e, effects of packet loss and I_d, end-to-end delay. Alexander [12] proposes the $I_{e,eff}$, the effective equipment impairment factor quantifying the impairment of a codec under both random loss and bursty loss. There are more detailed description and discussions in [13], which covers all the related topics on assessment and prediction on speech quality of VoIP. The results in [13] estimating the $I_{e,eff}$ impairment with iLBC [25], [26] Internet low bit rate codec under random loss are important and related to our results. Samir et al. [32] propose other non-intrusive parametric model to estimate the subjective MOS. The proposal is a random neural networks-based (RNN) approach, which could map to the subjective MOS very well, but the model needs a large sample space to train the coefficients of RNN. Kuan-Ta Chen et al. [33] introduce an innovative way to quantify the user's satisfaction on voice quality. In their work, they define the *user satisfaction index*(USI) and provide the method of deriving USI from network parameters via SKYPE [29] measurements, e.g. bit rate, jitter and RTT. However, the authors did not find the relationship between USI and subjective MOS, which so far has been considered as the standard way of measuring the quality of speech.

There are only a few articles focusing on quality of service for voice conference. Jonathan and Henning [15] demonstrate some existing voice conference topologies and also propose a protocol for decentralized conference. SKYPE [29] is becoming the most popular software due to its voice quality, robustness and free distribution. Experiments validating our models and proposals are using SKYPE because it supports concurrent voice conference very well. Early measurements on SKYPE, [17]–[19] reveal the basic properties of the software. Especially, in [17], the authors observe that the codecs used by SKYPE are iLBC [25], [26] and iSAC [27] and the network topology of SKYPE conference is the end system mixing topology [15].

VIII. Conclusion

In this paper, we propose *GMOS*, which is a a group-based MOS metric to evaluate the overall quality of voice conference. Note that this performance measure is important, not only because it lacks in this area but with this measure, it provides designers a systematic means to design group-based communication services as well. To leverage an existing work on MOS, we let GMOS be composed of MOS values plus a calibration parameter α. The parameter α can be calibrated in two ways. One is to indicate the user's perception on the conference quality, the second way is to consider it as an application dependent parameter.

Further, we propose the *two-step mapping method*(TSMM) to estimate the leader's GMOS$_L$ from the network parameters between the leader and the non-leader participants. We also propose the *leader selection strategy*(LSS) to improve the overall quality of voice conference by properly selecting the conference leader.

To validate our proposals, we have invited 25 subjects to listen to and give the scores to the records of our conference experiments by SKYPE. These subjects are asked to provide the MOS to each speakers in the records, and also an overall score GMOS (a subjective test) to the whole conference record as well. The results of our experiments indicate that both of the *two-step mapping method*(TSMM) and the *leader selection strategy*(LSS) perform very well.

Acknowledgment

This work is affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

References

[1] "Methods for subjective determination of transmission quality," 1996, ITU-T Recommendation P.800.

[2] "The E-Model, a computational model for use in transmission planing," 2005, ITU-T Recommendation G.107.

[3] "Objective quality measurement of telephone-band (300 − 3400 Hz) speech codecs," 1998, ITU-T P.861.

[4] "Method for objective measurements of perceived audio quality," 1999, ITU-R BS. 1387.

[5] "Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," 2001, ITU-T P.862.

[6] S. R. Quackenbush, T. P. Barnwell, III, and M. A. Clements, *Objective Measures of Speech Quality*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[7] J.Liang and R. Kubichek, "Output-based objective speech quality," in *Proceedings of IEEE Vehicular Technol. Conf.*, Stockholm, Sweden, 1994, pp. 1719 − 1723.

[8] D.-S. Kim, "ANIQUE: an auditory model for single-ended speech quality estimation," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 821 − 831, Sep. 2005.

[9] "Single-ended method for objective speech quality assessment in narrow-band telephony applications," 2004, ITU-T P.563.

[10] Y. H. Chu, S. G. Rao, S. Seshan and H. Zhang, "Enabling conferencing applications on the Internet using an overlay multicast architecture," In *Proceedings of ACM SIGCOMM'*01, August, 2001.

[11] "Definition of Catergories of speech transmission quality," 1999, ITU-T Recommendation G.109.

[12] A.Raake, "Short- and long-term packet loss behavior: towards speech quality prediction for arbitrary loss distributions," *IEEE Trans. Audio, Speech Lang. Process.*,vol. 14, no.6, pp. 1957 − 1968, Nov. 2006.

[13] A. Raake, *Speech Quality of VoIP - Assessment and Prediction.* Chichester, U.K.: Wiley, 2006.

[14] R. G. Cole and J. Rosenbluth, "Voice over ip performance monitoring," *Computer Communication Review*, vol. 31, no. 2, pp. 9 − 24, April 2001.

[15] J. Lennox and H. Schulzrinne, "A protocol for reliable decentralized conferecing," *ACM International Workshop on Network and Operating Systems Support for Diginal Audio and Video (NOSSDAV'03)*, June, 2003.

[16] T. Bu, Y. Liu and D. Towsley, "On the TCP-Friendliness of VoIP Traffic," in the *Proceedings of IEEE Conference on Computer and Communications(INFOCOM)* 2006.

[17] S. A. Baset and H. Schulzrinne, "An analysis of the Skype peer-to-peer internet telephony protocol," In *Proceedings of IEEE INFOCOM'*06, Apr. 2006.

[18] K. Suh, D. R. Figueiredo, J. Kurose and D. Towsley, "Characterizing and detecting relayed traffic: A case study using Skype," In *proceedings of IEEE INFOCOM'*06, Apr. 2006.

[19] S. Ehlert and S. Petgang, "Analysis and signature of Skype VoIP session traffic," Fraunhofer FOKUS Technical Report, NGNI-SKYPE-06b.

[20] A. W. Rix, J. G. Beerends, D. S. Kim, P. Kroon and O. Ghitza, "Objective assessment of speech and audio quality - technology and applications," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 14, no.6, pp. 1890 − 1901, Nov. 2006.

[21] S. Möller, A. Raake, N. Kitawaki, A. Takahashi and M. Wältermann, "Impairment factor framework for wide-band speech codecs," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 14, no.6, pp. 1969 − 1976, Nov. 2006.

[22] T. A. Hall, "Objective speech quality measures for Internet telephony," Proc. SPIE vol. 4522, p. $128 - 136$, Voice over IP (VoIP) Technology, 2001.

[23] J. Linden, "Achieving the highest voice quality for VoIP solutions," GSPx *The International Embedded Solutions Event, Santa Clara*, 2004.

[24] "Assessing VoIP call quality using the E-Model," white paper of IXIA. http://www.ixiacom.com/library/white_papers/

[25] iLBC codec. http://www.gipscorp.com/files/english/datasheets/iLBC.pdf

[26] S. V. Andersen, W. B. Kleijn, R. Hagen, J. Linden, M. N. Murthi and J. Skoglund, "iLBC - A linear predictive coder with robustness to packet losses," in the *Proceedings of IEEE Workshop of Speech Coding* 2002.

[27] iSAC codec. http://www.gipscorp.com/files/english/datasheets/iSAC.pdf

[28] Global IP Solutions. http://www.gipscorp.com/default/customers.html

[29] http://www.skype.com/download/

[30] http://www.ethereal.com/download.html

[31] http://www.adobe.com/downloads/

[32] S. Mohamed, G. Rubino and M. Varela, "Performance evaluation of real-time speech through a packet network: a random neural networks-based approach," Performance Evaluation 57, pp. $141 - 161$, 2004.

[33] K. T. Chen, C. Y. Huang, P. Huang and C. L. Lei, "Quantifying Skype user satisfaction," In *Proceedings of ACM SIGCOMM'*06, Sept. 2006.