

# The Design of a Multi-Party VoIP Conferencing System over the Internet\*

Batu Sat, Zixia Huang, Benjamin W. Wah  
 Department of Electrical and Computer Engineering  
 and the Coordinated Science Laboratory  
 University of Illinois at Urbana-Champaign  
 Urbana, IL 61801, USA  
 {batusat,zhuang21,wah}@uiuc.edu

## Abstract

In this paper, we present the design of a VoIP conferencing system that enables the voice communication of multiple users in the Internet. After studying the conversational dynamics in multi-party conferencing, we identify user-observable metrics that affect the perception of conversational quality and their trade-offs. Based on the dynamics and the behavior on delays, jitters, and losses of Internet traces collected in the PlanetLab, we design the transmission topology and schemes for loss concealments and play-out scheduling. Last, we compare the performance of our system and Skype (Version 3.5.0.214) using repeatable experiments that simulate human participants and network conditions in a multi-party conferencing scenario.

## 1 Introduction

Voice conversation is the most natural form of interpersonal communication. In a conversation of two or more participants, each person takes turns in uttering his/her thoughts and listens to others. In rare cases, multiple participants may speak simultaneously or one of them may interrupt another, causing double-talk. A face-to-face conversation is one in which all the participants reside in the same physical location, such as a meeting room. However, with the globalization of activities, there is an increasing need for people to communicate across geographic locations. This leads to the development of systems that enable face-to-face like communications with high speech quality and high perception of presence. In this paper, we present the design of a VoIP system that uses the public Internet for multi-party conferencing among users located around the world.

When VoIP is implemented using the public Internet, users may experience quality degradations due to non-stationary and dynamic delays and losses in the Inter-

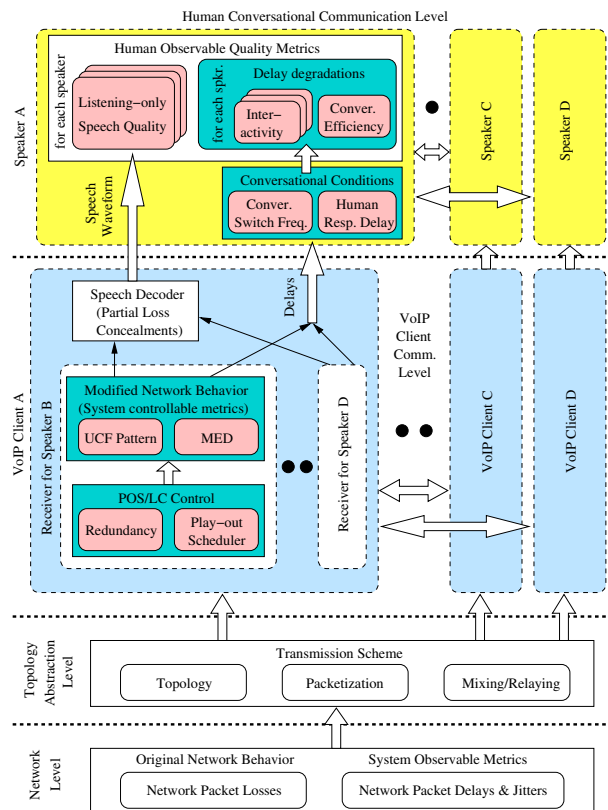


Figure 1. A VoIP conferencing system.

net [11]. Packets may be lost, either in isolation or in batches, and may experience sudden delay increases. This behavior cause packets to be unavailable at the receiver at their scheduled play-out times. To smooth the irregular arrival of packets, receivers commonly employ jitter buffers for storing received packets and play-out schedulers for playing the speech signals. We define the *mouth-to-ear delay* (MED) as the delay a speech frame incurs at the sender, the network, and the receiver jitter buffer before it is played.

Figure 1 depicts the interactions among the components of a VoIP conferencing system, the network, and the human

\*Proceedings of IEEE International Symposium on Multimedia, 2007

participants. It shows the dependencies among the system-observable, system-controllable, and user-observable metrics. (Some of the terms in the figure are explained later.)

A multi-party conversation consists of a series of alternating speech segments with *talk-spurts* and *silence periods*. The quality of the conversation from each participant’s perspective can, therefore, be evaluated by examining the quality of the one-way speech from each speaker and that of the interactive conversation among the participants.

In a one-way transmission of speech, the *listening-only speech quality* (LOSQ) improves as MED increases [12]. As more time is allowed for packets to arrive, even packets with long delays can be received in time, leading to high LOSQ at the receiver. Those packets lost in the network can be recovered by *loss-concealment* schemes (LC) that send redundant copies of these packets in subsequent ones. Hence, a perfect LOSQ can be achieved by a sufficiently long MED and a large number of redundant copies. The fraction of those frames that cannot be recovered is captured by the *unconcealed frame rate* (UCFR) [10]). Note that degradations in LOSQ as a function of MED also depend on the codec used: low bit-rate codecs tend to be less robust to losses, especially when consecutive frames are lost.

A user’s perception of LOSQ mainly depends on the intelligibility of the speech heard, since the user lacks a reference to the original sequence. LOSQ can be measured by either subjective (MOS: ITU P.800 [7]) or objective (PESQ: ITU P.862 [8]) methods.

The quality of an interactive conversation, however, does not depend on LOSQ alone. The *G.114 guidelines* [5] prescribe a one-way MED of less than 150 ms to be desirable for a voice-communication system and more than 400 ms to be unacceptable. However, they do not specify a metric for measuring the effect of delays, nor do they give trade-offs that lead to conversations of high perceptual quality.

As observed in our previous studies [12], different network paths exhibit different delay, jitter, and loss behavior. Therefore, an utterance spoken by a speaker can arrive at different listeners with different LOSQ and delays. This behavior leads to three potential problems. First, each listener will have a slightly different perception of the same conversation in a conference call. This may cause double-talks when more than one persons start speaking at the same time and the listeners perceive the double-talk at different points in time. Second, from a listener’s perspective, there is asymmetry in the silence durations in between different speaker’s speeches. This means that some speakers may appear to be more distant than others, or some respond slower than others. Last, when the same speech of different quality is delivered to different listeners, it is possible that one listener cannot understand an utterance and request the speaker to repeat it. This leads to significant inefficiency to all participants. For these reasons, it is not trivial to choose a play-

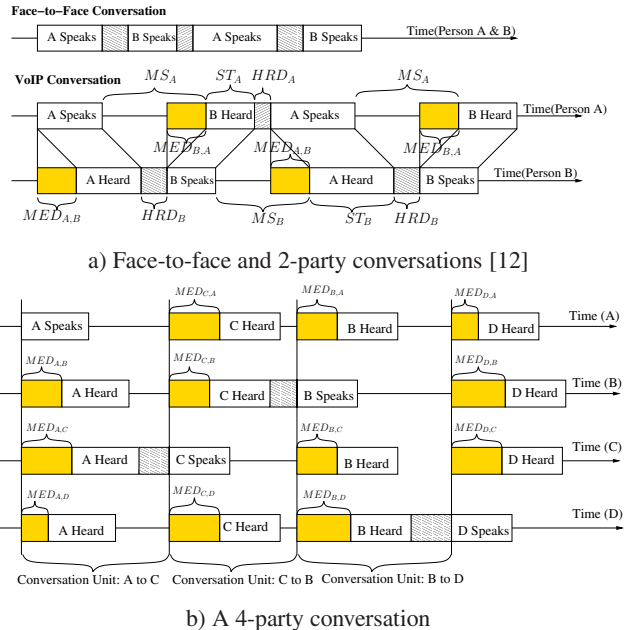


Figure 2. The dynamics of VoIP conversations.

out schedule for each client in order to maximize the overall perception of conversational quality for every participant.

**Problem Statement.** In this paper, we study the design of a VoIP conferencing system with high conversational quality that is consistent across time and participants. Based on the conversational dynamics and the Internet behavior, we investigate its transmission topology and develop effective schemes for loss concealment and play-out.

The rest of the paper is organized as follows. Section 2 presents the dynamics of a multi-party conversation. Based on experiments conducted in the PlanetLab, Section 3 describes the Internet behavior. Section 4 presents our VoIP system and relates its design to that of Skype and other studies. Last, Section 5 presents our experimental results.

## 2 Multi-Party Conversational Model

**Conversational Dynamics.** The dynamics of a face-to-face conversation is different from that over a channel with delay [12]. In a two-party VoIP conversation, there are two realities, each observed by one user, where the silence periods in between speech segments are of different durations. As a result, in a multi-party VoIP conversation, there are as many realities as there are participants.

Figure 2 depicts the dynamics of three types of conversations. Due to different delays from the speaker to the listeners, an utterance spoken can arrive at different listeners with different quality and delays. Hence, to a listener, not all speakers are symmetric: some appear to be more distant than others, or some respond slower than others.

We define  $HRD_Y$  (*human-response delay* perceived by Y) to be the duration after Y perceives that X has stopped

talking and before Y starts talking, during which Y thinks about how to respond to X [12]. However, the same delay is perceived to be longer by X, which we call  $MS_X$  (*mutual silence* perceived by X). Further, Z, who is simply listening to X and Y, perceives the silence duration yet another way. The relation among  $MS_X$ ,  $MS_Z$ , and  $HRD_Y$ , where  $MED_{X,Y}$  is the MED from X to Y and  $X \rightarrow Y$  is the conversational switch from speaker X to speaker Y, is:

$$\begin{aligned} MS_X^{X \rightarrow Y} &= MED_{X,Y} + HRD_Y + MED_{Y,X} \\ MS_Y^{X \rightarrow Y} &= HRD_Y \\ MS_Z^{X \rightarrow Y} &= MED_{X,Y} + HRD_Y - MED_{X,Z} + MED_{Y,Z}. \end{aligned} \quad (1)$$

For the purpose of analysis, a conversation can be divided into segments called *conversational units* (CU) that is identified by the start and the end times of the segment in absolute time (Figure 2b). For example, a CU from X to Y is denoted by the start of X's speech until the start of the next speaker Y's speech. Its duration is represented as:

$$CU^{X \rightarrow Y} = MED_{X,Y} + ST_X + HRD_Y. \quad (2)$$

**Quality Metrics.** The quality of a VoIP conversation depends on LOSQ as well as its naturalness and rhythm. We have identified three important metrics.

a) The *conversational interactivity* (CI) is the ratio between the silence period experienced by A (the current speaker) before hearing the response of C (the next speaker) and the duration A previously waited after hearing B (the speaker before A) [12]. Since the MEDs from a speaker to his/her listeners can vary significantly, each listener can perceive different CI when different clients respond to the speaker. For this reason, we extend our previous CI metric [12] in order to allow for different perception of CI with respect to each speaker interacting with the user analyzed.

Based on user-observable metrics, we define the *interactivity factor* ( $CI_i^{i \rightarrow j}(t)$ ) of a single-talk speech segment  $t$  ( $ST_t$ ) from person  $i$ 's perspective to be the ratio of  $MS_i$  experienced by  $i$  after speaking  $ST_t$  and the  $HRD_i$  waited by  $i$  before  $ST_t$  is spoken, where  $j$  is the next speaker:

$$CI_A^{A \rightarrow B}(t) = \frac{MS_A^{A \rightarrow B}}{HRD_A^{t-1}}, \quad CI_A^{A \rightarrow C}(t) = \frac{MS_A^{A \rightarrow C}}{HRD_A^{t-1}}. \quad (3)$$

In a face-to-face conversation, CI is approximately 1. However, CI increases as the round-trip delay increases. If the asymmetry in the response times increases, humans tend to have a degraded perception of interactivity that will result in the degradation of the conversational quality. One possible effect is that, if A perceives that B is responding slowly, then A tends to respond slowly as well.

b) The *conversational efficiency* (CE) measures the extension in time to accomplish a VoIP conversation when there are communication delays (Figure 2).

$$CE = \frac{\text{Speaking Time} + \text{Listening Time}}{\text{Total Time of Call}}. \quad (4)$$

Since a conversation over a network is charged according to its duration, the same conversation may cost more for a network with longer MEDs. This effect is especially pronounced in international and mobile calls, when both the network delay and the per-minute price are higher. Each participant perceives the same CE during the conversation.

c) *Conversational symmetry* (CS). As each participant perceives different CI with respect to others, he/she tends to perceive a degradation in the naturalness of the conversation because it does not resemble a face-to-face conversation with small and uniform delays. To capture the symmetry perceived by A, we define CS to be the ratio of the maximum MS experienced by A and the minimum MS experienced by A recently (say in the last minute):

$$CS_A = \frac{\max\{MS_A^{A \rightarrow X}\}}{\min\{MS_A^{A \rightarrow X}\}}. \quad (5)$$

The degradations due to delays may also depend on the conversational condition, such as the type of the conversation being carried out and the conversational switching frequency [12]. For example, in a conversation with less frequent switches between the parties, the degradations due to long MEDs will be perceived less severely. In contrast, in a conversation with higher switching frequency, there is an increased need for face-to-face like interactivity.

Note that during a VoIP session, a user does not have an absolute perception of MEDs because he/she does not know who will speak next and when that person will start talking. However, by perceiving the indirect effects of MED, such as MS and CE, the participant can deduce the existence of MED. For this reason, a participant cannot estimate exactly the duration of a CU but knows that it is closely related to CE. In short, MS, CI, CE, and CS are user-perceptible metrics that are intimately affected by MED.

Currently, there is no standard that relates MEDs to user-perceptible conversational-quality metrics. There are also no objective or subjective metrics that evaluate the quality of a multi-party conversation over a network with delay.

**Trade-offs.** There are trade-offs between MED and LOSQ, similar to those in a two-party conversation. These trade-offs exist for each speaker-listener pair, making it a multi-dimensional problem. When individually solved for a given speaker, the trade-offs can lead to different MEDs for different listeners. However, improving CE by minimizing the MED for each speaker-listener pair may increase the risk of unconcealed losses due to delay spikes. Moreover, it can degrade the perception of symmetry in CI. Hence, a proper balance must be made between MED and LOSQ.

There are also trade-offs between the consistencies of the conversational dynamics across participants and the delivery of high and consistent quality to all. If one participant cannot understand the spoken utterance, than he/she, depending on the importance of the topic, will interrupt the

**Table 1.** Internet traces collected in July and August, 2007, from one source to 7 destinations (duration 10 min; packet period 30 ms; DL: delay; JT: jitter; JT30: jitters larger than 30 ms wrt mean delay; JT60: jitters larger than 60 ms wrt mean delay; and LR: loss rate). Delays are classified into low (less than 100 ms), high (larger than 100 ms), and mixed (a combination of both). Similarly, jitters are classified into low (less than 5% in JT60), high (greater than 5% in JT60), and mixed; and losses into low (less than 5%), high (greater than 5%) and mixed. The delay, jitter and loss behavior of the different receivers is characterized by Type into uniform and non-uniform. The destination nodes are listed using a triplet of three numbers (# in aSia,# in America,# in eUrope).

Set	Type	DL (L/H/M)	JT (L/H/M)	LR (L/H/M)	Hour (CST)	Source		Dest. (S,A,U)	Mean DL (ms)		JT30 (%)		JT60 (%)		LR (%)	
						Loc	IP Addr		Min	Max	Min	Max	Min	Max	Min	Max
1	Uniform	L	L	L	20:00	CA,USA	169.229.50.14	(1,2,4)	42.2	94.6	0.00	0.23	0.00	0.15	0.00	0.00
2	Uniform	H	L	L	18:00	China	219.243.201.77	(0,3,4)	107.3	190.4	0.03	4.2	0.00	3.5	0.00	0.01
3	Uniform	H	L	H	23:00	Hong Kong	137.189.97.18	(0,3,4)	101.2	204.3	0.02	1.8	0.00	1.64	14.7	22.7
4	Uniform	H	H	L	22:00	Taiwan	140.112.107.80	(1,3,3)	198.0	280.4	74.7	76.5	68.3	72.2	0.14	0.22
5	Non-unif	M	L	L	20:00	Czech	195.113.161.82	(2,3,2)	56.0	158.4	1.8	2.3	0.45	0.97	0.00	3.39
6	Non-unif	M	H	L	17:00	CA,USA	171.66.3.181	(2,2,3)	74.9	170.9	27.8	48.2	5.2	6.2	0.00	4.33
7	Non-unif	M	L	H	1:00	Hong Kong	137.189.97.18	(1,3,3)	85.4	195.9	0.01	1.9	0.00	1.6	15.3	22.8
8	Non-unif	M	L	M	11:00	Canada	198.163.152.229	(2,2,3)	52.4	147.3	0.00	0.86	0.00	0.83	0.00	16.9
9	Non-unif	M	M	L	5:00	UK	128.232.103.203	(2,3,2)	26.5	139.9	0.01	8.11	0.00	8.10	0.00	3.2
10	Non-unif	H	M	M	1:00	China	211.94.143.61	(0,4,3)	103.7	198.9	2.7	12.6	1.2	6.6	1.9	8.6
11	Non-unif	M	M	M	8:00	Hungary	152.66.244.49	(3,2,2)	22.6	190.6	0.02	79.8	0.00	79.0	0.00	25.1

current speaker to ask him/her to repeat the last sentence. This will lead to significant inefficiency in the communication. Hence, an increased importance should be placed on consistent and high quality when the number of participants is large and the network conditions have disparities.

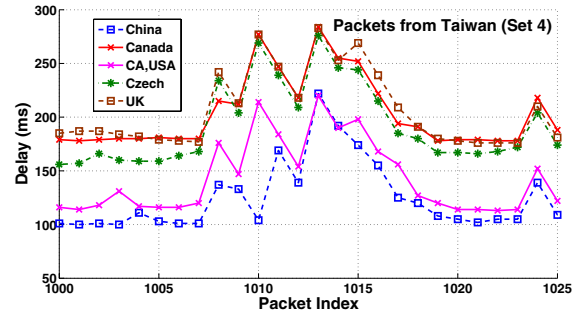
### 3 Internet Traffic Behavior

The Internet is a best-effort public network with path-dependent, non-stationary and dynamic behavior. In our previous studies [12], we have collected traffic traces for a two-party VoIP conversation in the Internet. In this section, we present our observations on Internet transmissions that are related to multi-party VoIP conversations.

In VoIP conferencing, network traffic exhibits more disparities because there may be multiple participants that are scattered around the world. In general, the traffic patterns need to be characterized in a multi-dimensional fashion.

In our experiments, we have collected real-time Internet traffic traces in the PlanetLab. Packets were sent from one node to several other nodes simultaneously using point-to-point UDP packets every hour over a 24-hour period. We used a 30-ms packet period in order to match the sending rate in VoIP transmissions. As it is important to measure the delay each packet took to travel from the sender to the destinations, each packet carried in its payload a local timestamp that was synchronized every 10 minutes by a nearby NTP time server. We used three local NTP servers, one in each continent, in our experiments: `time.nist.gov` (Americas), `ntp.time.ac.cn` (Asia), and `ntp2.npl.co.uk` (Europe and Middle East). Let  $\Delta t_1$  be the offset of the sender from its nearby NTP server, and  $\Delta t_2$  be that of the receiver. The one-way delay between these two nodes is:

$$DL = (t_2 - \Delta t_2) - (t_1 - \Delta t_1). \quad (6)$$



**Figure 3.** Delay behavior of packets collected from Taiwan to Xian (China), Canada, California (USA) and Czech at 1:00 CST in August 2007 (Trace 4).

Our scheme assumes that the various NTP servers are synchronized to within some small tolerance and that each client has compensated for round-trip delays between itself and the nearby NTP server. Although it does not guarantee that all local clocks are perfectly synchronized, the errors incurred are small enough when compared to the one-way delay between two clients. The errors are also expected to be smaller than a simple scheme that computes the one-way delay as half of the round-trip delay between two nodes.

Table 1 shows the statistics of 11 sample traces collected from one source to 7 destinations. There are three observations on the data collected.

First, the traces have large variations in their delays, jitters, and losses that depend on the time they were collected.

Second, there may be large disparities in delays, jitters, and losses across the destinations for packets sent from a source. The behavior tends to be more uniform across destinations in the same continent but have larger disparities across continents. For example, packets in Trace 11 from Hungary to nodes in Europe have less than 100 ms average

**Table 2.** Delay and jitter behavior of packets collected from Hungary to Hong Kong, China, Finland and Berkeley in Trace 11 at 1:00 CST in July, 2007. See keys in Table 1.

Destination	Min DL	Avg DL	Max DL	JT30	JT60
Hong Kong	133 ms	190.6 ms	1529 ms	79.8%	79.0%
China	121 ms	150.3 ms	1495 ms	77.4%	76.1%
Finland	24 ms	25.7 ms	64 ms	0.03%	0.00%
Berkeley	90 ms	90.8 ms	126 ms	0.02%	0.00%

delay and little jitters. However, the same stream to Asia has over 120 ms average delay and has jitters and losses.

Third, the behavior of packets from one source to multiple destinations may be correlated. Figure 3 shows that the delays of packets sent from Taiwan to five destinations in Asia, America, and Europe are strongly correlated. Such correlations are likely caused by congestion in the vicinity of the source node. In contrast, Table 2 illustrates that packets sent from Hungary experience high jitters to destinations in Asia. Such correlations are likely caused by congestion in links between Europe and Asia.

#### 4. Design of a VoIP Conferencing System

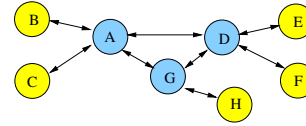
The VoIP conferencing system in Figure 1 can be viewed in two parts. The component managing the transmission of voice packets is common to all clients and requires their coordination and cooperation. The remaining component is implemented as the play-out scheduling (POS) and lost-concealment (LC) schemes in individual clients, similar to those in a two-party system. In addition, a client may need to manage the speech stream from each participant.

In this section, we discuss these components and relate them to the designs in Skype (Version 3.5.0.214). There are several VoIP software available for use in the Internet, including Skype, Google-Talk, Windows Live Messenger, Yahoo Messenger, and Gizmo Project. To the best of our knowledge, only Skype supports multi-party conferencing.

**Transmission scheme.** This is characterized by the connection topology and the location where audio mixing is done [13]. Its design depends on the trade-offs between  $P$ , the maximum number of packets transmitted or relayed by any node in one period, and  $ME2ED$ , the maximum end-to-end delay observed by any speaker-listener pair. Perceptual quality is affected by  $P$  because sending packets too frequently may lead to congestion and loss. It is affected by  $ME2ED$  that captures the worst-case one-way delay.

Assuming  $M$  clients in the call,  $N(t)$  of them speaking simultaneously at time  $t$ , and the simple case in which clients do not join or leave during a call, there are three possible connection topologies for a VoIP conferencing system.

a) A *decentralized* scheme requires each client to send packets to every listener, either directly via unicasts or via multicasts if available. The most common architecture is a



**Figure 4.** An overlay topology with  $M = 8$  and 3 parents.

full mesh, where each of the  $N$  speaking clients sends its data to each of the  $M - 1$  listening clients via unicasts. Although  $ME2ED$  is the shortest in this topology, the scheme may be bottlenecked at a client, especially when the number of clients is large. Each client maintains  $M - 1$  jitter buffers and decoders,  $N(t)$  of which are active at  $t$ .

b) A *centralized* scheme requires all clients to communicate with either a dedicated server or one of the VoIP clients (called a host or a bridge) that operates in one of two ways.

The host can decode the incoming speech streams, mix the waveforms (if multiple clients are speaking), and re-encode the waveform to be sent to the  $M$  clients. In this case,  $ME2ED$  is large because it involves the transmission of signals across two hops, as well as the de-jittering delay in mixing the signals. Further, tandem coding (repetitive encoding and decoding) causes significant degradations in quality, especially for low-bit-rate codecs [13]. The benefits, however, are that each speaking client sends packets to a single host, and that each client only maintains a single jitter buffer and decoder, independent of  $M$ .

Alternately, the host can simply select a subset of the speakers and relay their signals to all clients. If the host chooses to limit the number of simultaneous streams, it can do so by using either the loudest-talker or the first-come-first-serve algorithm [13]. The scheme may increase  $ME2ED$  if it chooses to first decode the signals.

Skype adopts the first alternative [3] in which the node that initiates the call (called central host) invites a maximum of nine clients to join and acts as their router. The central host also mixes the streams received before forwarding them to the clients. This is evidenced by our observation that, under no loss and jitter, the packet size and packet rate to each client is not increased when the number of simultaneous speakers is increased. Another evidence is that the central host is generally more loaded than the other clients.

c) A *hybrid* scheme uses an *overlay network* to relay the speech packets sent by a client to all listeners. Each speaking client communicates with the nearest node in the overlay network, whereas nodes in the overlay network relay packets to each other using either a centralized or decentralized scheme (Figure 4). Although a listening client still needs to maintain a receiver for each speaker, its transmission burden is significantly reduced. There have been several studies on overlay-network designs, both in general and in the context of VoIP conferencing applications [13, 1], using different optimization criteria.

In this paper, we study a commonly used overlay topology constructed by a subset of the clients in the call (called

parent nodes). All the parent nodes are fully connected, and each remaining node in the call (called child node) is connected to only one parent node. Our goal is to design a topology with a proper trade-off between  $ME2ED$  and  $P$ . To avoid the delay and degradations in tandem coding, each parent node simply forwards all the packets received to other parent nodes as well as the connected children.

In the initialization phase when the call is set up, the client that initiated the call collects the network delay and loss information among the clients. Due to the prohibitive nature of enumerating all overlay topologies, we use a greedy algorithm. The heuristic first determines the client pair with  $ME2ED$  (called bottleneck pair) in the full-mesh topology. It then finds a single-parent topology that minimizes  $ME2ED$  among the  $M$  single-parent topologies. If the difference between  $ME2ED$  in the full-mesh topology and that in the optimal single-parent topology is small (say less than 50 ms), then it uses the best single-parent topology as the overlay network. Otherwise, it adds a second parent node in order to reduce  $ME2ED$  to a lower level as well as decreasing  $P$ . The heuristic iteratively increases the number of parents until either the difference between  $ME2ED$  of the current topology and that of the previous topology is small enough or the bottleneck pair in the full-mesh topology is directly connected in the current topology candidate. The process can be repeated during the conversation if there is a significant change in the network condition.

**Coding and packetization.** In our conferencing system we use the ITU G722.2 codec [6] (23.85 kbps bit-rate option), with each packet at 40-ms period and containing two 20-ms frames. We use this codec because it is a wide-band codec with high-quality outputs, and its source code is readily available. The bit rate used allows packets from multiple speakers destined to the same listener at the same time to be combined into one packet, without exceeding the MTU, even when redundant piggybacking is used. Note that wide-band speech is overwhelmingly preferred to narrow-band speech, and the use of a wide-band codec is a necessity when comparing to a commercial system like Skype.

In comparison, Skype [2] uses the proprietary iSAC [4] codec in its two-way calling feature. iSAC is an adaptive codec with a framing option between 30-60 ms and a bit rate between 10-32 kbps. However, we do not have any information whether iSAC is also used in its multi-party conferencing implementation. Our experiments show that Skype adopts four framing options in multi-party conferencing: 60 msec, 45 msec, 30 msec and 15 msec, with a payload ranging from 246-255 bytes, 196-205 bytes, 136-170 bytes and 96-110 bytes. Our measurements indicate that, when the network has low loss and low jitters (regardless of delays), all nodes progressively increase from an initial period of around 60 ms and 32 kbps to around 15-ms period and 50 kbps. Further, each node adaptively adjusts

its rate according to the network condition. For instance, if one of the links has higher jitters, then its packet period may stay at 30 ms. Our measurements also indicate that clients in Skype employ silence suppression and send silence packets of around 16-21 bytes every 50 ms.

**Loss concealments (LC).** In our previous studies [10, 11] we have designed several end-to-end LC schemes for concealing packet losses in two-party VoIP applications. To reduce the network overhead and improve the response of LC adaptations, we use in this paper a link-based LC scheme, instead of an end-to-end scheme. We use threshold-based redundant piggybacking to conceal network losses by resending previously transmitted packets in the current packet. In case when multiple speakers are talking, we combine their packets that are destined to the same client into one packet (without exceeding the MTU) to prevent increasing the packet rate. This scheme requires each parent node to maintain retransmission buffers for storing recently received speech frames. By limiting the degree of redundant piggybacking to 4, the overhead is small because the scheme needs  $P$  buffers, each limited to 4 packets.

In comparison, our measurements show that Skype, in response to high losses in one direction of a link, doubles its payload (regardless of delay) for that direction of the link, without changing the packet rate. Moreover, the packet size and rate for the other links remain the same. This change applies to both voice packets as well as silence packets. Based on these observations, we anticipate that Skype carries out two-way piggybacking in response to network losses.

**Play-out scheduling (POS).** There have been numerous studies on the design of POS algorithms for two-party VoIP applications [9, 14]. The general goal of these algorithms is to optimize a time-varying cost-function, based on either system-observable or user-observable metrics [12]. Under moderate delays ( $< 300$  ms), the algorithm tries to hug the network-delay curve in order to minimize MED, without incurring significant packet losses due to lateness.

In a multi-party conversation, the goal of minimizing the MED for each individual path may not be crucial because the overall MED is governed by the bottleneck path. In this paper, we propose a new POS algorithm that adapts according to the bottleneck path. Assuming that the end-to-end-delay statistics between the current speaker and all clients is periodically broadcast to all participants,  $node_{BN}(t)$  (the listening client that experiences the highest delay from the current speaker at time  $t$ ) as well as the bottleneck path and its estimated MED are known to each client. The bottleneck node then adapts its MED according to this delay statistics, while the non-bottleneck nodes adapt its MED based on both this statistics as well as the most recent MED estimate of the bottleneck node:

$$\begin{aligned} MED_{BN} &= F(0.99) \\ MED_{non-BN} &= \alpha F(0.99) + (1 - \alpha) \hat{MED}_{BN}, \end{aligned} \quad (7)$$

**Table 3.** The locations of the PlanetLab nodes in the network traces used in our experiments.

Person	Trace Set 1	Trace Set 2	Trace Set 3
A	CA, USA	UK	NH, USA
B	Canada	Hong Kong	UK
C	NH, USA	Finland	Canada
D	Hefei, China	NH, USA	Hungary
E	Hong Kong	Hungary	Hefei, China

where  $F$  is the CDF of the network delay between a speaker-listener pair in the past 10 seconds. Here,  $\alpha$  adjusts how symmetric the MEDs would be for different clients listening to the same speaker. For  $\alpha = 0$ , all listening nodes use the recent estimate of the bottleneck MED, which improves CS but degrades CI and CE. In contrast,  $\alpha = 1$  reduces the scheme to a non-cooperating scheme by choosing the optimal MED for each speaker-listener pair, which improves CI and CE but degrades CS. In this paper, we use  $\alpha = 0.3$  for simplicity. In the future we plan to conduct human-subject tests in order to verify the optimality of  $\alpha$ , possibly as a function of the conversational condition.

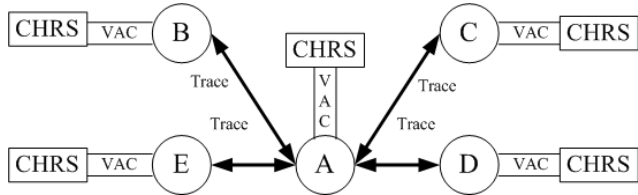
We are not able to identify the POS algorithm used in Skype because its voice packets are encrypted and the source code of the clients is not available.

## 5. Experimental Results

In this section, we compare the performance of our VoIP system with Version 3.5.0.214 of Skype, using experiments that simulate human participants and network conditions in a 5-party conferencing scenario. To facilitate fair and repeatable comparisons, we used the same network and conversational conditions in each evaluation. Based on 10 pre-recorded wide-band speech segments of 1-3 sec in duration, 2 from each of the 5 participants, we generated a random sequence on the order in which participants would speak. Table 3 lists the three sets of network traces collected in the PlanetLab used in our experiments.

In our system, we first chose the overlay topology in the initialization phase. We then evaluated our system in a computer simulation implemented in MATLAB that carried out G722.2 coding/decoding and PESQ executables via system calls. We also recorded all the spoken as well as heard wave files for later analysis.

In Skype, we conducted our experiments using five computers that were configured in a conference call (Figure 5). We used the computer corresponding to person A in Table 3 to initiate the call and to act as the host. In each computer, we implemented a Conference Human Response Simulator (CHRS) that communicated with Skype via the Virtual Audio Cable (VAC) software, which behaved like a virtual pipe for audio transmissions. The goal of CHRS is to simulate a multi-party conversation with smooth turn-taking between



**Figure 5.** The configuration of Skype's experiments.

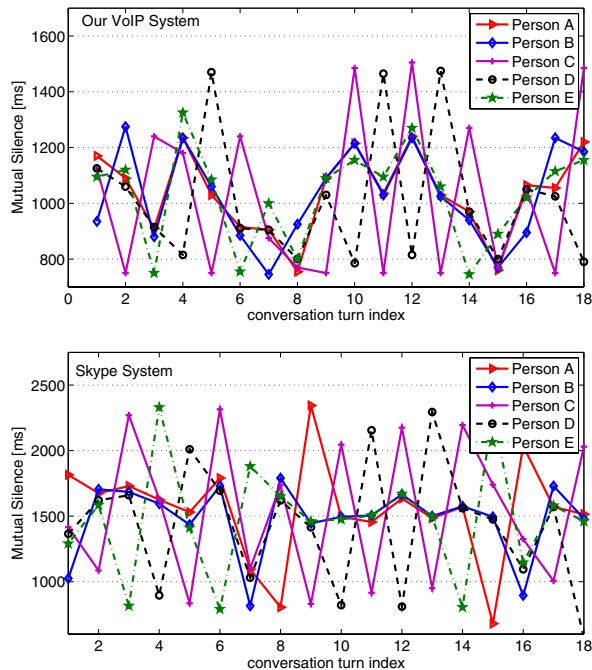
**Table 4.** The performance of our VoIP conferencing system and Skype evaluated under the three sets of traces in Table 3. MS: mutual-silence duration; Rsp: respondent; PrSpk: prior speaker; Lst: listener; CI: conversational interactivity; CS: conversational symmetry; CE: conversational efficiency; CMOS: comparative MOS rating between our system and Skype (ours is better for positive numbers).

Set	System	MS [ms]			CI	CS	CE	PESQ	CMOS
		Rsp.	PrSpk.	Lst.					
1	Ours	1256	780	1029	1.62	1.68	70	3.477	+0.87
	Skype	2078	853	1510	2.44	1.80	62	2.754	
2	Ours	1072	780	925	1.35	1.40	73	3.741	+0.80
	Skype	1975	866	1462	2.32	2.11	63	2.916	
3	Ours	1071	780	928	1.36	1.35	72	3.735	+1.13
	Skype	1983	898	1463	2.29	2.40	62	2.995	

participants and without double-talks. By using a predefined order in which the participants conversed, when a particular participant's turn is up for conversation, its CHRS waited for 750 ms after detecting the end of the previous speech, before sending some prerecorded speech waveforms to Skype. To allow the analysis of quality, CHRS also recorded the spoken waveforms as well as the waveforms heard from other participants. To simulate Internet traffic, UDP packets between any of the 5 computers were routed through a modified Linux computer that trapped and released UDP packets with delay and loss patterns driven by traces collected in the PlanetLab. We also used Ethereal in each node to monitor incoming and outgoing packets.

We processed the waveforms in each of the conversations from both systems. Based on the boundaries extracted from the spoken and heard waveforms, we computed the Mutual Silence (MS) perceived by each client between two segments, as well as CI, CS, and CE. For each segment, we also evaluated its LOSQ using PESQ. Last, we conducted unofficial CMOS (Comparative MOS) tests that compared each of the conversations generated by our system and the corresponding conversations of Skype using the methodology defined in ITU P.800 [7]. In our tests, each test subject was presented two conversations and was asked to compare the quality of one relative to another. The discrete scores recorded are from the set  $\{-3, -2, -1, 0, 1, 2, 3\}$  that correspond to, respectively, *much worse*, *worse*, *slightly worse*, *about the same*, *slightly better*, *better*, and *much better*.

Table 4 summarizes the results of the two systems evalu-



**Figure 6.** Durations of mutual silence experienced by each of the five persons during the conversation using our system (top) and Skype (bottom) using trace set 1.

ated under the three sets of network traces. We observe that listeners in our system experience MS that is about 480 ms (or 30%) on average shorter than that in Skype. Similarly, a participant waiting for a response to his/her speech waits an average 2,078 ms in Skype as opposed to 1,256 ms in ours.

Figure 6 further depicts the MS durations in each conversation unit (CU) perceived by each of the 5 participants in Skype and in our system. Note that in each CU, one person is the respondent, one is the prior speaker, and the remaining are listeners. For both systems, the respondent waits about 750 ms in HRD, which is the smallest amongst the MS durations for that CU. In the next CU, the previous respondent (now the prior speaker) usually experiences the highest MS. In the following CU, the same user, possibly becoming a listener and experiencing MS similar to that of other listeners, has MS that is larger than that of a respondent but smaller than that of a prior speaker. Our scheme has lower MS on average for both the prior speakers and the listeners, and consequently, a person’s perception of MS has less variance than that experienced when using Skype.

Table 4 also captures the difference in MS in terms of CI, in which our system is shown to provide a more interactive conversation. Further, CS experienced in our system is closer to 1 as compared to that of Skype, which indicates a more balanced MS that is experienced by different users in the same conversation or by the same user at different

times. The overall effect of reduced silence durations in our system is captured by CE that shows, for each of the three network traces, the channel is idle about 30% of the time as compared around 40% in Skype. The PESQ evaluations of wide-band speech segments in our system show a clear improvement in LOSQ with respect to that of Skype.

The unofficial CMOS tests conducted between the conversations generated by our system and the corresponding conversations by Skype indicate that, for trace sets 1 and 3, our system is slightly preferred as compared to Skype, and there is no significant difference between two systems for trace set 2. Subjects indicated that our system had significantly shorter delays but was almost in par with Skype in terms of LOSQ, despite the better PESQ values.

## References

- [1] Y. Amir, C. Danilov, S. Goose, D. Hedqvist, and A. Terzis. An overlay architecture for high quality VoIP streams. *IEEE Trans. on Multimedia*, 8(6):1250–1262, Dec. 2006.
- [2] S. Baset and H. Schulzrinne. An analysis of the Skype peer-to-peer Internet telephony protocol. In *Proc. IEEE Int’l Conf. on Computer Communications*, Apr. 2006.
- [3] T. Fu, D. Chiu, and J. Lui. Performance metrics and configuration strategies for group network communication. In *Proc. IEEE Int’l Workshop on Quality of Service*, June 2007.
- [4] iSAC Codec. Datasheet of Internet speech audio codec.
- [5] ITU-G.114. One-way transmission time.
- [6] ITU-G.722.2. Wideband coding of speech at around 16 kbit/s using adaptive multi-rate wideband (AMR-WB).
- [7] ITU-P.800. Methods for subjective determination of transmission quality.
- [8] ITU-P.862. Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs.
- [9] S. B. Moon, J. Kurose, and D. Towsley. Packet audio playout delay adjustment: performance bounds and algorithms. *Multimedia Systems*, 6(1):17–28, Jan. 1998.
- [10] B. Sat and B. W. Wah. Analysis and evaluation of the Skype and Google-Talk VoIP systems. In *Proc. IEEE Int’l Conf. on Multimedia and Expo*, July 2006.
- [11] B. Sat and B. W. Wah. Evaluation of conversational voice quality of the Skype, Google-Talk, Windows Live, and Yahoo Messenger VoIP systems. In *IEEE Int’l Workshop on Multimedia Signal Processing*, (accepted to appear) Oct. 2007.
- [12] B. Sat and B. W. Wah. Playout scheduling and loss-concealments in VoIP for optimizing conversational voice communication quality. In *Proc. ACM Multimedia*, Augsburg, Germany, (accepted to appear) Sept. 2007.
- [13] P. J. Smith, P. Kabal, M. L. Blostein, and R. Rabipour. Tandem-free VoIP conferencing: A bridge to next-generation networks. *IEEE Communications Magazine*, pages 136–145, May 2003.
- [14] L. Sun and E. Ifeachor. New models for perceived voice quality prediction and their applications in playout buffer optimization for VoIP networks. In *Proc. IEEE Communication*, volume 3, pages 1478–1483, 2004.