Design and Traffic Engineering of VoIP for Enterprise and Carrier Networks

James Yu and Imad Al Ajarmeh
DePaul University, Chicago, Illinois, USA
jyu@cdm.depaul.edu   iajarmeh@cdm.depaul.edu

Abstract

The paper presents an extension of the Erlnag-B model for traffic engineering of Voice over IP (VoIP). The Erlang-B model uses traffic intensity and Grade of Service (GoS) to determine the number of trunks in circuit-switched networks. VoIP, however, is carried over packet-switched networks, and network capacity is measured in bits per second instead of the number of trunks. We study different network designs for VoIP, and propose a Call Admission Control (CAC) scheme based on network capacity. We then propose a new measurement scheme to translate network bandwidth into the maximum call load. With this new metric, the Erlang-B model is applicable to VoIP. We conducted experiments to measure the maximum call loads based on various voice codec schemes, including G.711, G.729A, and G.723.1. Our results show that call capacity is most likely constrained by network devices rather than physical connections. Therefore, we recommend considering both packet throughput (pps) and bit throughput (bps) in determining the max call load. If network capacity is constrained by packet throughput, codec schemes would have almost no effect on the maximum call load.

Keywords: VoIP, Erlang B, Call Admission Control, Traffic Engineering, Packet Throughput

1. Introduction

The growing popularity of Voice over IP (VoIP) is evident on the residential, enterprise, and carrier networks. The traditional IP-based networks are designed for data traffic, and there is no engineering consideration for voice traffic which is sensitive to packet delay and loss. To meet the new challenges of network convergence of both voice and data services on the same network, traffic engineering is important to network design as well as to the continual operation of the services. This paper provides an in-depth study of the VoIP traffic engineering and presents an enhanced traffic engineering model for VoIP. Among the various available traffic engineering models, the Erlang-B model has been widely used to engineer the voice traffic of circuit-switched networks for many years [1]. The purpose of the Erlang-B model is to calculate the resources (outgoing trunks) based on the Grade of Service (GoS) and traffic intensity. An example of traditional circuit-switched network is illustrated in Figure 1.

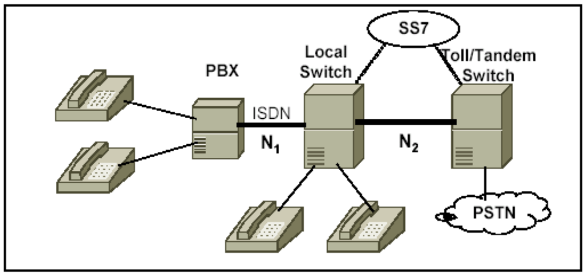

Figure 1. Legacy Telephone Network

The limiting resource in this network is the number of trunks between switches. For enterprise users, this resource is the number of trunks (N1) between their PBX and the local switch. If an enterprise subscribes too few trunks, the end-user would experience a high probability of blocking, for both incoming and outgoing calls. If the enterprise subscribes too many trunks, many of them will not be used, resulting in poor resource utilization and waste of money. On the carrier side of the network, the limiting resource is the number of trunks (N2) between a local switch and a tandem/toll switch. N2 is determined by network engineers to satisfy the traffic demand on the carrier core network. Traffic engineering is to calculate the required network resources (N1 or N2) based on the traffic demand and service requirements.

In packet-switched networks, there are no circuits or trunks. These networks accept any incoming packets. If the arrival rate of incoming packets is higher than the service rate of the network, constrained by network devices or outgoing links, packets will be buffered for later delivery. The effect of packet buffering is longer delay. If the buffer is full, new packets are discarded, which result in packet loss.

When packets are lost, an upper layer protocol between the sender and the receiver (not in the intermediate node) may retransmit the packet, which would result in even longer delay. Of course, some protocols, such as UDP, may ignore the lost packets and take no actions. This operation of packet-switching is not appropriate for voice communication which is sensitive to delay and packet loss.

This paper is an extension of our earlier publication [2] with expanded work on the design of an overlay network for VoIP, more detailed coverage on traffic measurement, and additional VoIP experiments. This paper is organized as follows: Section 2 provides a brief overview of how others are addressing the traffic engineering issue of VoIP. Section 3 explains the traditional Erlang-B model, and Section 4 presents the architecture and design of VoIP networks for the enterprise and carrier environment. A detailed analysis of VoIP traffic and its applicability to the Erlang-B model is given in Section 5. We present a comprehensive experimental design to emulate the VoIP traffic, and the results are given in Section 6. The last section, Section 7, presents the conclusion and some open issues for future work

2. Call Admission Control

The purpose of Call Admission Control (CAC) is to determine if the network has sufficient resource to route an incoming call. In the circuit-switched networks, the Call Admission Control algorithm is simply to check if there are circuits (or trunks) available between the origination switch and the termination switch. VoIP traffic is carried over packet-switched networks, and the concept of circuits (trunks) is not applicable. However, the need for Call Admission Control (CAC) of VoIP calls is the same. Packet switched networks, by nature, accepts any packet, regardless of voice or data packets. When the incoming traffic exceeds the network capacity, congestion occurs. Control mechanism is needed to address the issue of congestion by traffic shaping, queuing, buffering, and packet dropping. As a result of this procedure, packets could be delayed or dropped. Delay is usually not an issue for data-only applications. Packet loss can also be recovered by retransmission, which is supported by many protocols, such as TCP or TFTP. However, retransmission would cause longer delay which is not acceptable to time-sensitive applications. For voice traffic, delay and packet loss would degrade the voice quality, which is not acceptable to end-users. It should be noted that that CAC is different from Quality of Service (QoS) as frequently referenced in the literature. The main difference is that QoS is a priority scheme to differentiate the traffic already on the network, while CAC is to police the traffic from coming to the network when the network is congested [3].

CAC for circuit-switched network is implemented in the Q.931 and SS7 signaling [1]. Q.931 is to determine if there is a free B channel in the ISDN trunk and reserve the B channel for an incoming call. SS7 signaling is to identify a free DS0 channel between central office switches and reserve that DS0 channel for an incoming call. Although VoIP is on a packet-switch network, voice communications still require *circuits* (an end-to-end connection) to guarantee its voice quality.

There are many publications about ensuing voice quality over IP networks, and the general approach of Call Admission Control is to reject a VoIP call request if the network could not ensure the voice quality. CAC mechanisms are classified as measurement-based control and resource-based control.

Measurement-based Control: For measurement-based control, monitoring and probing tools are required to gauge the network conditions and load status in order to determine whether to accept new calls or not [4]. A protocol, such as RSVP, is required to reserve the required bandwidth before a call is admitted into the network.

Resource-based Control: In the case of resource-based control, resources are provisioned and dedicated for VoIP traffic. The resource for VoIP is usually calculated in network bandwidth [5]. The CAC approach in this paper is resource-based control, but our approach to calculating traffic demand is different from others.

Those two mechanisms are also referenced as link-utilization-based CAC and site-utilization-based CAC [6]. Another reference of these two methods is measurement-based CAC and parameter-based CAC [7]. In both CAC methods, the voice quality of a new call and other existing calls shall be assured after a call admission is granted.

3. The Erlang-B Model

The Erlang-B model is the standard to model the network traffic of circuit-switched networks. It is known as the blocked-calls-cleared model [8], where a

[1] SS7 signaling is for North America, and it is known as Common Channel Signaling (CCS) 7 or C7 internationally. Their functions are the same, but implementations are different.

blocked call is removed from the system. In this case, the user will receive an announcement of circuit busy. Note that a busy announcement is not the same as busy signal, which is the case when the callee is already on the phone. From the perspective of the Erlang-B model, not-answered-calls and busy calls are all considered successful calls. This section provides a brief overview of the Erlang-B model and its application to the circuit-switched network. Our goal is to enhance the model and apply it to the IP network.

3.1. Traffic Measurement

In a circuit-switched network, the limiting resource is the number of circuits which is also known as trunks (N). The traffic load on the network is measured by Traffic Intensity which is defined as

$$\textit{Traffic Intensity (A)} = \textit{Call Rate} \times \textit{Call Holding Time}$$

where call rate is the number of incoming calls during a certain period of time. Call Rate is randomly distributed and follows the Poisson distribution. Call Holding Time is the summation of (a) call duration which is the conversation time, (b) waiting time for agents at call center, and (c) ringing time [9]. The measurement unit of Traffic Intensity is *Erlang* which is the traffic load of one circuit over an hour. For example if a circuit is observed for 45-minute of use in a 60-minute interval, the traffic intensity is *45÷60=0.75 Erlang.*

The third parameter of the Erlang-B model is Grade of Service (GoS) which is *probability* of an incoming call being blocked. For a typical circuit-switched network, the reason for a call being blocked is that all trunks are busy. A GoS of 0.01 shows that there is 1% probability of getting a busy announcement. GoS is a critical factor for calculating the required number of trunks since it represents the trade off between service and cost. For a local telephone switch, if we set the number of trunks (to the tandem office) equal to the number of subscriber lines, the switch would have GoS=0 (100% non-blocking), regardless of the traffic load. Of course, this is a hypothetical example as no carriers would have this engineering practice.

3.2. The Model

The Erlang B model is commonly used to determine the mathematical relationship of the traffic measurements defined in Section 3.1. The assumptions of the Erlang B model are

Infinite number of sources: The model implies that an infinite number of users who could make a call through the network. In practice, if the number of users is much larger than the number of trunks, this assumption is considered valid.

Random call arrival: Since we have a large number of users, each user may initiate or receive a call at any time. The call arrival is random and follows the Poisson distribution, which also implies that the inter arrival time follows the exponential distribution. The randomness also implies that call events are independent of each other, where $\mathrm{Call}_{[i]}$ and $\mathrm{call}_{[i+1]}$ are two independent calls.

Blocked calls are cleared: When a call is blocked due to insufficient resources (trunks), the user will get a recording or a fast busy tone. The call request is discarded (cleared) by the network and the user must hang up and try again at a later time.

Random holding time: The holding time (call duration and waiting time) also follows the exponential distribution.

It should be noted that the assumptions of the Erlang-B model are transparent to the underlying networks, regardless of whether it is a circuit-switched network carrying traditional phone calls, or a packet-switched network carrying voice calls in the form of VoIP. Another important note is that the Erlang B model has been proved to be fairly *robust* where minor violation of model assumptions would still yield useful and practical results for traffic engineering. For example, one could argue that incoming calls are not totally *independent* of each other, especially during a special occasion. To address this concern, the standard practice is to take a conservative approach in measuring traffic intensity on the Busiest Hour of the Busiest Week/Season (BSBH) in a year. In other words, one should never engineer the network based on the *average* demand; instead, it should be based on *quasi-peak* demand. Based on the above assumptions, we can derive the mathematical formula for the Erlang B model:

$$GoS = (A^N \div N!) \div [\textstyle\sum_{\kappa=0}{}^{N} (A^k \div k!),\ k=0,N]$$

where A is Traffic Intensity in Erlangs and N is number or trunks.

Due to the popularity of the Erlang B model among network engineers, an on-line calculator is available to calculate the model parameters [10].

4. Voice over IP (VoIP) Networks

This paper studies three VoIP architectures: (1) enterprise network, (2) access network of Internet service provider, and (3) VoIP carrier network.

4.1. VoIP network for Enterprise

The VoIP network for enterprise is illustrated in Figure 2.

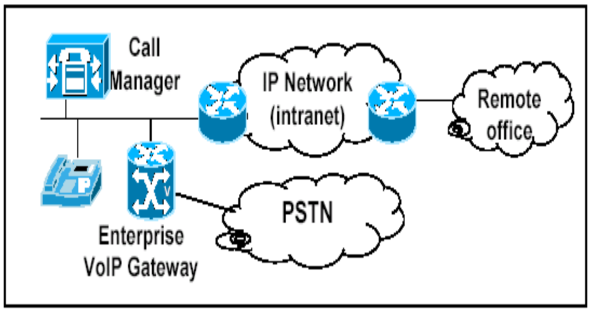

Figure 2. VoIP for Enterprise Networks

In the enterprise network, voice calls are carried over the packet-switched IP network within the enterprise. The VoIP network has an interface to the PSTN network, usually a T1 link. At the perimeter, the VoIP gateway provides the signaling interworking between Session Initiation Protocol (SIP) and Q.931/ISDN. The signaling function is to establish a duplex end-to-end connection between the caller and the callee, and it could be initiated from either direction. After the call setup, the VoIP gateway extracts the voice payload from the IP packets (for outgoing calls) or encapsulates the voice payload onto the IP packets (for incoming calls).

In some implementations, the enterprise phone network consists of IP phones, and a Call Manager. In other cases, the enterprise local phone system has both IP and analog phones. In the latter case, the call control process requires a hybrid PBX supporting both IP and analog calls [11].

Traffic engineering for the enterprise network has two elements. The first one is the engineering of the trunk capacity (number of DS0's) to the PSTN, and the Erlang-B model is applicable for this element. The second element is the network capacity (in bps) on the enterprise network which carries both voice and data traffic as illustrated in Figure 3.

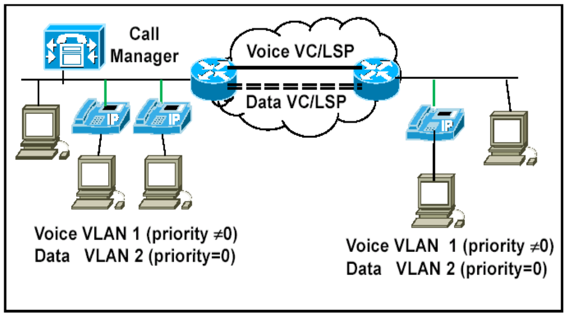

Figure 3. Enterprise Voice and Data Network

In general, Local Area Network (LAN) is either 100BaseTX or Gigabit Ethernet with capacity up to 1000Mbps. Although it is unlikely to see network congestion on LAN, we need to consider the bursty nature of data traffic. Therefore, our recommendation is to enable VLAN-tagging (802.1Q) with priority (802.1p). 802.1p supports a 3-bit priority scheme, with up to eight priority queues. Most Ethernet switches and IP phones support 802.1Q/p, but many support only two priority queues: priority≠0 for priority (voice) traffic and priority=0 for best effort (data) traffic. Frames with priority≠0 have priority over frames with priority=0 and will be processed first. With this priority scheme, we could consider 100% of the LAN bandwidth is reserved for voice traffic. If there is no voice traffic, Ethernet switches will then forward data frames. Because of the high capacity bandwidth of Ethernet and the use of 802.1p, traffic is unlikely to encounter congestion on the LAN.

The Wide Area Network (WAN), however, has relatively low bandwidth, usually from 1.5M (T1) to 45M (DS3). In rare cases of large enterprises, it could go up to 155M (OC-3). Figure 3 illustrates an example of a single connection between two locations, and this connection needs to carry both voice and data traffic. As discussed in Section 2, we propose to use the resource-based control mechanism where we provision a dedicated connection for voice traffic. The dedicated connection could be a physical link, an ATM or Frame-Relay Virtual circuit (VC), or an MPLS-based Label Switch Path (LSP). The dedicated connection has guaranteed bandwidth for voice traffic, and the traffic engineering model will be based on this bandwidth. This network design does not need to consider the bursty nature of data traffic and would never experience network congestion (for voice traffic) if Call Admission Control (CAC) is implemented. The Call Manager decides whether to accept or reject an incoming call request based on provisioned bandwidth and available bandwidth.

4.2. Access Network

The second VoIP architecture is the access network, where an enterprise subscribes to the VoIP service through an Internet Service Provider (ISP). The network architecture is illustrated in Figure 4. Because the VoIP traffic is carried over the public Internet which is a best-effort network and does not support QoS, we cannot apply Call Admission Control in this architecture. The engineering of trunks between the ISP voice gateway and the PSTN follows the Erlang-B model as described in Section 3.2.

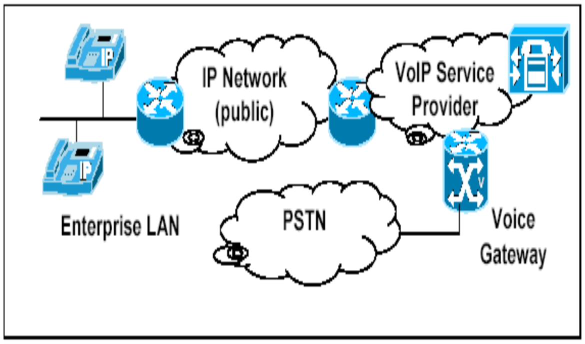

Figure 4. VoIP for Access Networks

4.3. Tandem Service over a Carrier Network

The third VoIP architecture is tandem service over the carrier network as illustrated in Figure 5. The two major network elements are Voice Trunking Gateway and Softswitch. Voice Trunking Gateway receives Voice Time Division Multiplexing (TDM) traffic from legacy voice switches and converts it to IP packets and forwards the packets to the IP backbone for transport. Softswitch uses the Signaling System 7 (SS7) to interface with the legacy voice switches and also to interface with other softswitches. The purpose of the SS7 is to establish an end-to-end connection between the caller and callee. It should be noted that the edge router may also accept VoIP traffic from another VoIP carrier.

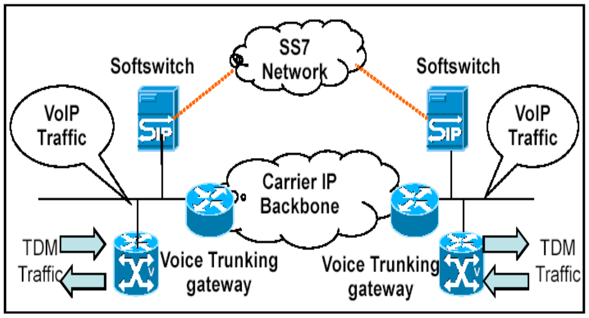

Figure 5. VoIP for IP-based Carrier Networks

Figure 5 shows only the voice traffic, and the IP backbone carries both voice and data traffic as illustrated in Figure 6.

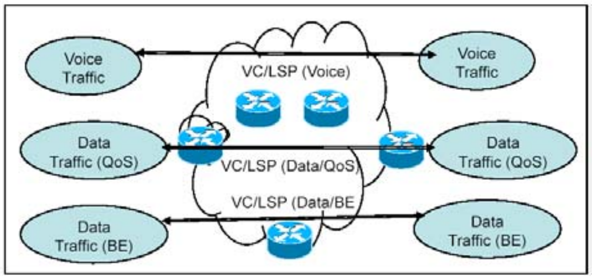

Figure 6. Carrier IP Backbone

In our network design of resource-based control, we propose three over-lay networks on the IP-backbone: voice network, QoS data network, and Best Effort (BE) data network. As discussed earlier, we could use either Virtual Circuit (VC) or Label Switch Path (LSP) to provision virtual connections and create the overlay network among physical nodes. Because voice network is a dedicated network, we could avoid the network congestion issue by implementing Call Admission Control (CAC) on softswitches. If the voice network has capacity to ensure voice quality for a new call, the call is accepted and the softswitch uses the SS7 signaling protocol to establish a connection over the IP backbone. Otherwise, the call request is rejected. Traffic engineering is to calculate the demand and determine the bandwidth required on the voice overlay network to ensure Grade of Service (GoS).

5. VoIP Traffic Analysis

VoIP packets are transported over Real-time Transport Protocol (RTP) which in turn uses UDP. RTP provides sequencing and time-stamp to synchronize the media payload. Real-time Transport Control Protocol (RTCP) is used in conjunction with RTP for media control and traffic reporting. Our experiment shows that RTCP is only about 1% of the VoIP traffic, so RTCP traffic is excluded in our analysis for traffic engineering.

5.1. VoIP Packet Overhead

VoIP encapsulates digitized voice in IP packets. The standard Pulse Code Modulation (PCM) uses 256 quantization level and 8,000 samples per seconds. As a result, we have a digitized voice channel of 64 kbps

(DS0). If we use 20ms sampling interval, each sample will be

64,000 bps × 20 ms = 1,280 bits = 160 bytes

This digitized voice is then encapsulated in an RTP/UDP/IP packet as illustrated in Figure 7 [12].

Layer-2 header	IP header 20 bytes	UDP header 8 bytes	RTP header 12 bytes	Payload 160 bytes

Figure 7. VoIP Frame

If the layer-2 is Ethernet, the 802.3 frame header, Frame Check Sequence (FCS), preamble, and Inter-Frame Gap (IFG) add additional 38 bytes. If the layer-2 is Point-to-Point Protocol (PPP), its header and FCS are 7 bytes.

PCM is the standard codec scheme for G.711, which does not use any voice compression algorithm. If a codec compression algorithm is used, the bandwidth for a voice channel is reduced to 8 kbps for G.729A and 5.3-6.3 kbps for G.723.1. Some codec schemes employ a silence compression mechanism where the bit rate is significantly reduced if no voice activity is detected. Furthermore, look-ahead algorithms are used in order to anticipate the difference between the current frame and the next one. In this paper we do not address those enhancements. A summary of voice codec schemes is shown in Table 1 [13].

Table 1. Vocoding and VoIP Overhead

	G.711 (10 ms sampling interval)	**G.711** (20 ms sampling interval)	**G.729A** (20 ms sampling interval)	**G.723.1** (30 ms sampling interval)
Raw BW in bps [1]	64,000	64,000	8,000	5,300
VoIP Payload (bytes)	80	160	20	20
VoIP overhead (802.3)	78	78	78	78
VoIP overhead (PPP)	47	47	47	47
BW in bps (802.3)[2]	126,400	95,200	39,200	26,133
BW in bps (PPP) [2]	101,600	82,800	26,800	17,867

5.2. VoIP Traffic Characteristics

VoIP Systems use two types of messages on the IP networks: (a) Control Traffic, and (b) IP Voice Payload Traffic. The control traffic is generated by the call setup and management protocols and is used to initiate, maintain, manage, and terminate connections between users. VoIP Control traffic consumes little bandwidth and does not require to be included in the traffic engineering modeling. It is possible to provision another overlay network for signaling messages which have more stringent requirements than the payload traffic.

IP voice payload traffic consists of the messages that carry the encoded voice conversations in the form of IP packets. This type of traffic is what concerns network engineers as it requires relatively high bandwidth and has strict latency requirements. IP Voice payload Traffic is referred to as VoIP traffic and has some unique characteristics that require special handling and support by the underlying IP networks. The traffic characteristics that should be considered for VoIP networks are:

Real Time Traffic: Voice conversations are real time events. Therefore, transmitting voice data over IP networks should be performed as close to real time as possible, maintaining packet sequence and within a certain latency and latency variation (jitter) limits.

Small Packet Size: In order to minimize the sampling delay and hence maintain the latency constrains, VoIP data is carried in relatively small IP packets.

Symmetric Traffic: VoIP calls always generate symmetric traffic, same bandwidth from caller to calee and from callee to caller. This characteristic of VoIP traffic combined with the small packet size will have impact on the network devices as we will see later in this article.

Any-to-any Traffic: any user might call any other user on the VoIP network which limits the ability of network engineers to predict the path of traffic flow. VoIP traffic might be initiated or terminated at any terminal point of the network, unlike many of the IP data networks where the majority of the traffic flows are known (e.g., clients to servers).

5.3. VoIP Call Requirements

Although human ear can tolerate some degradation in the voice quality and still be able to understand the

[2] The bandwidth (BW) is for one voice channel. Required Bandwidth includes the overhead based on the codec packet sampling rate.

conversation; however, there are certain requirements that should be met so that a VoIP call is acceptable. Transporting a Voice Call over the packet switched network has many challenges posed by the nature of the IP-based network which was originally designed for the data traffic. On the VoIP network, the major factors that determine voice quality are given as follows:

Delay: Represents the one-way end-to-end delay which is measured from speaker's mouth to listener's ear (mouth-to-ear). Delay includes coding/decoding, packetization, processing, queuing, and propagation delay. The ITU-T G.114 [14] recommends for the one-way delay to be less than 150 ms in order to maintain a quality conversation and transparent interactivity. If VoIP packets are delayed more than this limit, collisions might happen when the call participants talk at the same time.

Jitter: This is a measure of the variation in time of arrival (TOA) for consecutive packets. The original voice stream has fixed time intervals between frames; however, it is impossible to maintain this fixed interval on the IP network. The variation is caused by the queuing, serialization and contention effect of the IP networks. VoIP endpoints provide jitter buffers to compensate for the variation in TOA and to support the re-sequencing process. Packets enter the jitter buffer at a variable rate (as soon as they are received from the network) and are taken out at a constant rate for proper decoding. Buffering increases the overall latency and the jitter buffer size should be carefully chosen in a way to keep the overall latency (one-way delay) within the acceptable range. Packets arriving outside the jitter buffer boundaries will be discarded. Jitter calculations should also consider voice activity detection, out of order packets, and lost packets.

Packet Loss: Unlike data connections, VoIP has some tolerance to packet loss; however, if packet loss ratio exceeds a certain limit the quality of the call will be negatively affected. Several reasons might lead to packet loss in a network such as network congestion, transmission interference, attenuation, rejection of corrupted frames, and physical link errors. Different voice codec schemes have different tolerance to packet loss; however, it is recommends that packet loss be kept bellow 1%. It should also be noted that some packets might reach the intended destination and yet be dropped because they are late by more than the jitter buffer value. Therefore, measuring packet loss must also include the jitter buffer loss which is a factor of jitter buffer size and packet delay variation.

Vocoding (voice codec): the vocoding scheme is another important factor in determining voice quality. A codec scheme could implement compression algorithm, redundancy and lost packet hiding techniques. Different vocoding schemes also generate different digitally encoded voice frames in terms of frame size, bit rate, and the number of frames per second.

5.4. Measurement of Voice Quality

Based on the above requirements for VoIP calls, the ITU-T standard provides the following guideline for the voice quality measurement [15]:

Table 2. VoIP Quality Measurement

Network Parameter	Good	Acceptable	Poor
Delay (ms)	0-150	150-300	> 300
Jitter (ms)	0-20	20-50	> 50
Packet Loss	0-0.5 %	0.5-1.5%	> 1.5%

A common voice quality measurement scheme is the Mean Opinion Score (MOS) where different voice samples are collected and played back to a group of people who rank the voice quality between 1 and 5 (1 is the worst and 5 is the best). An MOS of 4 or better is considered toll quality. The objective of Call Admission Control is to prevent network congestion so that all calls could achieve toll quality or better.

5.5. Erlang B Model for VoIP

In the previous sections, we studied different VoIP architectures, network design, VoIP call requirements and traffic engineering using Erlang-B model. This section presents how to use the Erlang-B model to engineer the VoIP traffic so that we can provide the optimum solution to balance between service quality and cost. The goal is to provide adequate bandwidth and network devices capable of supporting the call demand. In VoIP networks, the concepts of Grade of Service (GoS), and traffic intensity (call arrival rate and call holding time) are the same as in circuit-switched networks. However, the number of trunks in the Erlang-B model is not applicable to a packet-switched network. Therefore, we propose to use the maximum number of simultaneous calls with toll quality. This parameter is also referenced as *maximum call load* in this paper. We will provide an experimental framework to measure this parameter in Section 6. This parameter is comparable to the number of trunks used in the Erlang-B model. With the proposed revision, the Erlang-B model has the same three parameters:

A: Traffic Intensity
GoS: Probability of blocking calls
N: Max Call Load

6. Experimental Design and Analysis

We developed an empirical framework to emulate the VoIP traffic in the lab environment. The emulated VoIP traffic is the UDP traffic with the payload size equal to the RTP header and vocoding data.

6.1 VoIP Traffic Emulation

Our experiments were performed using different network links and architectures. The lab configuration is illustrated as follows:

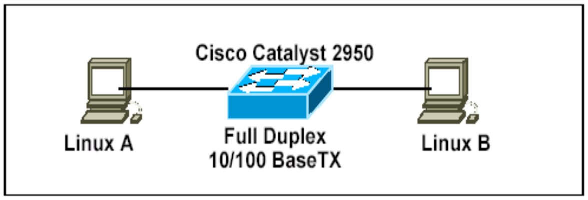

Figure 8a. VoIP Test over Switched Ethernet

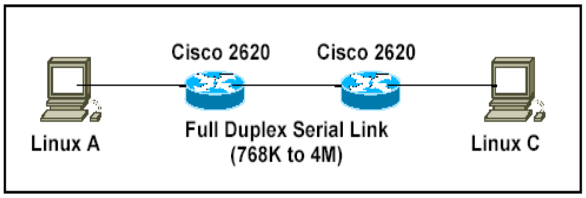

Figure 8b. VoIP Test over Serial Interface

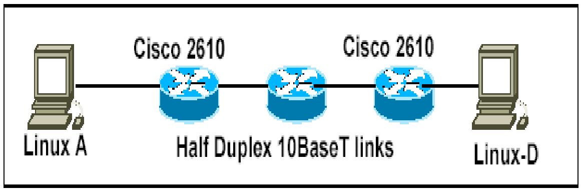

Figure 8c. VoIP Test over Routed Ethernet

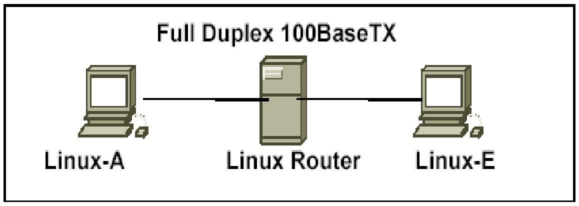

Figure 8d. VoIP Test over Routed Fast Ethernet

The switched Ethernet environment is for the baseline measurement which is to ensure the validity of our measurement tool and the measurement process. The low speed link (serial interface up to 2Mbps) is to emulate the enterprise intranet, and the high speed links (4Mbps and up) are to emulate a potential carrier IP backbone.

In each experiment run, the sender sends a batch of UDP messages (with a sequence number and a time stamp on each message) to the receiver. When the receiver receives messages, it echoes them back immediately. The symmetric traffic is to emulate a voice call. When the sender receives the echoed message, it computes the delay and then sends the message with a new time stamp and a new sequence number. The number of messages in the batch is similar to the TCP window for flow control and congestion control. Our objective is to achieve the maximum link utilization by having the maximum number of messages in the batch without causing any congestion or packet loss. When network congestion or packet loss happens, it implies poor voice quality.

During the experiment, we also monitor the CPU utilization of the sender and receiver machines. If the CPU utilization is above 60%, we consider the experiment invalid as the bottleneck is on the CPU and not on the network. We also conducted a baseline measurement in which we use the message size close to the MTU of 1,500 bytes. The purpose of the baseline measurement is to demonstrate that the experiment is able to achieve the wire speed performance. The expected results (theoretical limit) are calculated based on the overall bandwidth requirements for each codec shown in Table 1. Table 4 shows a summary of the theoretical maximum call load for different codec schemes on different links.

Table 4. Theoretical Call Capacity

Links	**G.711** (20ms)	**G.711** (10ms)	**G.729A** (20ms)	**G.723.1** (30ms)
FD FT1 (768k)	9.3	7.6	28.7	43
FD E1 (2.0M)	24.2	19.7	74.6	111.9
FD 2×E1 (4.0M)	48.3	39.4	149.3 [1]	223.9[3]
10BaseT (HD)	52.5	39.6	127.6 [1]	191.3[3]
10BaseT (FD)	105		255.1	382.7
100BaseTX (FD)	1,050	791.1	2,551	3,827

[3] Note that a Full Duplex Serial link of 4.0M carries more calls than a half duplex 10BaseT link because PPP has less overhead than Ethernet. (See Table 1

The following section presents the experimental results. We compare the experimental results with the theoretical limits presented in Table 4 as follows:

Utilization = experimental result ÷ theoretical limit

This new metric is to measure the efficiency of a link for voice calls, and it is different from the traditional measure of data throughput and link utilization.

6.2. Experiment Results

The first experiment is a VoIP traffic test over a full duplex 10/100BaseTX link. The key measurement is the maximum number of simultaneous calls with toll quality (max call load). The results of this experiment are presented in Table 5. The column labeled "utilization" is the comparison to the theoretical limit presented in Table 4. Figure 9 shows a graphical comparison between the theoretical and experimental max call limit on a 10BaseT full duplex link.

Table 5. 10BaseT Full Duplex Switched Link

Message Size (bytes)	Codec	Max call Load	Utilization (%)
1450	(baseline)	---	96%
160	G.711	105	100%
20	G.729A	251	98%
20	G.723.1	376	98%

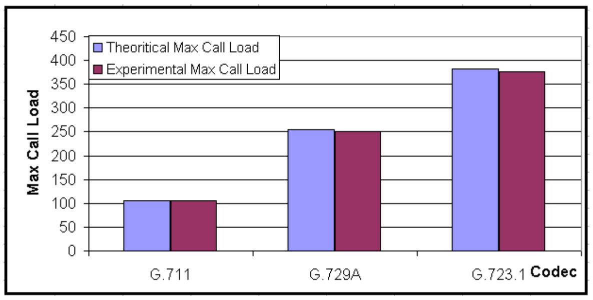

Figure 9. 10BaseT FD Switched Link

When we tried to run this experiment over the 100BaseTX link, the CPU utilization of the Linux machine reached 98%. Therefore, the experiment of 100M is considered not applicable for measuring the max call load.

The second experiment is to test the VoIP traffic over a serial link with two routers; we configured the link speeds to 768Kbps, 2Mbps, and 4Mbps. The results are given in Table 6. Figure 10a, Figure 10b, and Figure 10c show the graphical comparison between the theoretical and experimental max call limit on a 768Kbps, 2Mbps, 4Mbps serial links respectively.

Table 6. Full Duplex Serial Links (2 routers)

Codec	Serial Link (768K)		Serial Link (2M)		Serial Link (4M)	
	Max Load	Util.	Max Load	Util.	Max Load	Util.
Baseline	---	98%	---	98%	---	98%
G.711	9.2	99%	24.2	100%	40.0	83%
G.729A	28.0	98%	61.5	82%	70.0	47%
G.723.1	42	98%	92.3	82%	105.0	47%

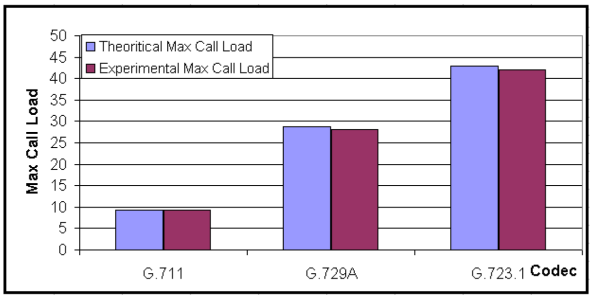

Figure 10a. Serial Link (768Kbps)

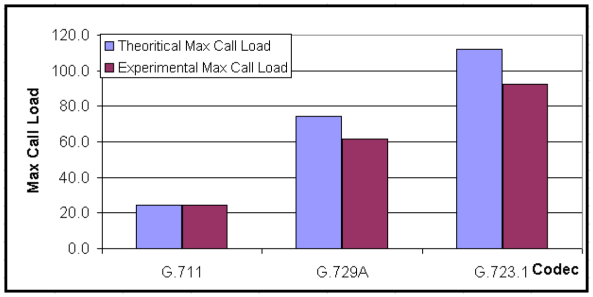

Figure 10b. Serial Link (2Mbps)

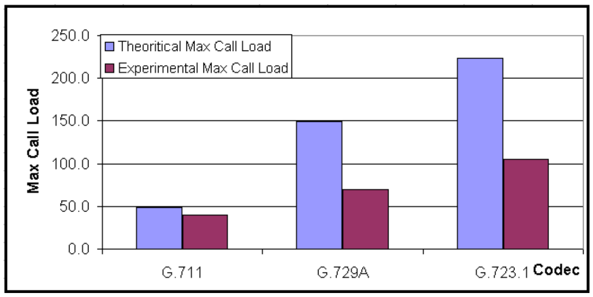

Figure 10c. Serial Link (4Mbps)

The third experiment is to emulate VoIP over three routers with 10BaseT link (half duplex), and the results are presented in Table 7 and Figure 11. During the experiment run, we also monitor the CPU utilization of traffic transmitter and receiver. The CPU utilization on the transmission side is 40% for G.723.1 and G.729A and 20% for G.711. The utilization is much lower on the receiver side, less than 10% in all cases.

Table 7. 10BaseTX Routed Link

Codec	Half Duplex (10BaseT)	
	Max Call Load	Utilization (%)
Baseline	---	97%
G.711	41	78%
G.729A	73	57%
G.723.1	109.5	57%

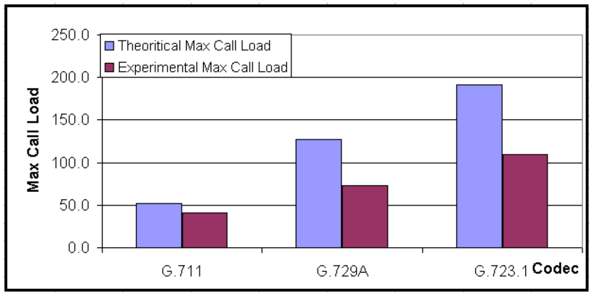

Figure 11. 10BaseTX HD Routed Link

The fourth experiment is to emulate VoIP over a routed full duplex 100BaseTX link. In this experiment, we used a Linux-Based router on a Pentium 4 machine, and the CPU utilization for sender and receiver is less than 40% in all cases. The results of this experiment are shown in Table 8 and Figure 12 bellow.

Table 8. 100BaseTX Routed Links

Codec	Full Duplex (100BaseTX)	
	Max Call Load	Utilization (%)
Baseline	---	97%
G.711	390	37.1%
G.729A	465	18.3%
G.723.1	897	18.2%

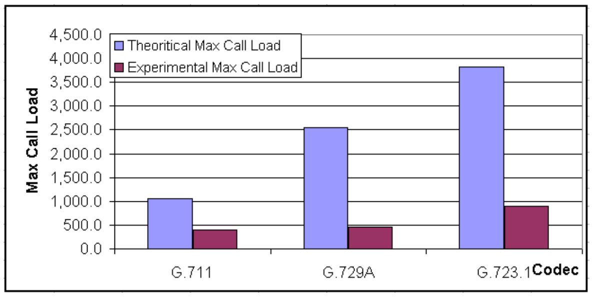

Figure 12. 100BaseTX FD Routed Link

A summary of the observed maximum call loads versus expected (theoretical) maximum call loads is shown in Figure 13.

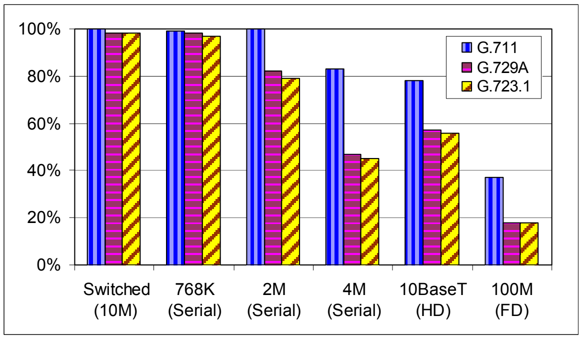

Figure 13. Call Utilization for Various Links

The fifth experiment is to study the effect of the sampling interval on the maximum call load. In this experiment we changed the sampling interval for G.711 to 10ms, and the payload size was also changed to 80 bytes. We ran the experiment over 10BaseTX full duplex switched link and 10BaseT routed link. Table 9 and Figures 14a and 14b show the comparison between Max Call Load and link utilization for different packet sampling rates.

Table 9. Call Load and Packet Sampling Rate

Codec	10BaseT Switched Link		10BaseT Routed Link	
	Max Call Load	Util.	Max Call Load	Util.
G.711 (10ms)	77	98%	26	67%
G.711 (20ms)	105	100%	41	78%

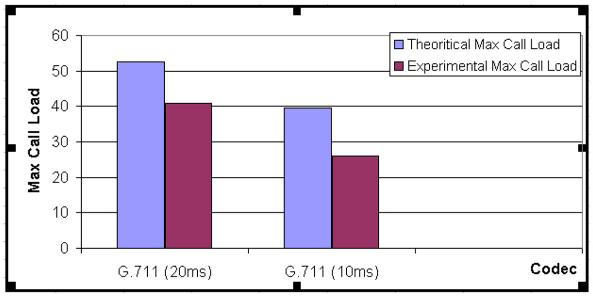

Figure 14a. Packet Sampling Rates and Codec on 10BaseT Half Duplex Link

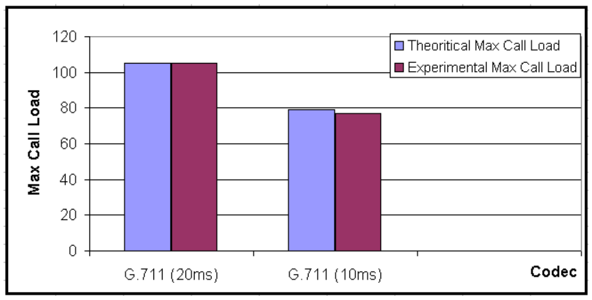

Figure 14b. Packet Sampling Rate and Codec on 10BaseT Full Duplex Link

The observations from these experiments are summarized as follows:

1. We are able to achieve wire speed performance (96% or better) using the max message size in all experiments. This result confirms the validity of the measurement tool and the experiment process.
2. The data shows close to 100% utilization on 10BaseT switched Ethernet (Table 5.) It shows that we could achieve the max call load as calculated from the available bandwidth.
3. In the cases of routed networks, we observed close to 100% utilization only on low speed links, but poor utilization on high speed links. It shows that the max call load cannot be achieved on the high speed links.
4. G.711 always yields better utilization than G.729A which is comparable to G.723.1. It shows that the smaller size for a codec scheme would yield lower utilization on the link. This is an interesting result, and we will investigate further later.
5. Although G.729A and G.723.1 compress the voice payload by a factor of 8-10, their improvement to the max call load is less than 10% on high speed links.
6. When using larger packet sampling rates (from 10ms to 20ms), we notice significant increase in the Max Call Load.

In summary, the experimental results raise a question about how to measure call loads for VoIP. Many other studies calculate the call load based on the bit throughput (bps), and our experiment shows that bps alone could not explain the results observed in the experiment as there is a large discrepancy between observed data and calculated data.

6.3. Packet Throughput and Max Call Load

Our lab experiments show that in the case of low utilization, it always involves routers. This observation leads to the study of packet throughput (number of packets processed per second) of network devices. The routers used in this experiment are Cisco 2610 and Cisco 2620. According to the product specifications [16], these routers are able to carry 1,500 packets per second (pps). If Cisco Express Forwarding (CEF) is enabled and the traffic pattern is applicable, the router could achieve 15,000 pps. Each VoIP call requires two connections (one in each direction) and this is the symmetric characteristic of VoIP traffic we discussed in Section 5.2.

The way pps is calculated for router is that each packet is counted twice as it goes through the incoming port and the outgoing port. If we use 20ms sampling interval and 64-byte frames, the calculated max call load of a router would be

15,000 pps ÷ (1000 sec ÷ 20 ms) ÷ 4 = ***75 calls/sec***

And for 30ms sampling interval (G723.1) we have

15,000 ÷ (1000 ÷ 30) ÷ 4 = ***112 calls/sec***

These numbers are consistent with all the experimental results of the routers. In other words, the max call load is bounded by the router "capacity" rather than the link capacity.

We also noticed that we were able to achieve maximum utilization on the physical links for the baseline tests (using MTU as the packet size). The inconsistency in utilization leads to the question about the root cause of difference between the baseline tests and emulated VoIP tests. To answer this question, we need to study the VoIP traffic characteristics in 5.1 and compare with the processing of packets by network devices. We find that VoIP uses small packet size to transfer calls. In order to achieve higher link utilization using small packet size, we need to send more packets per second. Pushing more small packets into the network would not cause congestion on the link itself; instead, the routers on the network may not be able to process the demand and become the congesting point.

For example if we use G.729A codec on a half duplex 10BaseT link:

Frame Size = 98 bytes (or 784 bits)
20 byte (payload) +
8 byte (UDP) +
12 byte (RTP) +
20 byte (IP) +
38 byte (Ethernet, preamble, and IFG)

If we want to achieve full link utilization (10M bps) using G.729 codec, we need packet throughput of

10,000,000 bps ÷ 2 ÷ 784 bit/packet = 6,377pps

Since VoIP traffic is symmetric in both directions, we need the network to handle twice this amount. According to the product specification, each packet is counted twice as it goes through the router (coming and leaving). Therefore, the required packet throughput for the router is:

$$6{,}377 \times 2 \times 2 = 25{,}508 \; pps$$

As discussed earlier, our router (Cisco-2600) is capable of processing only 15,000 pps. Because of this constraint, we observe a lower link utilization which is

15,000 ÷ 25,508 = 58.8%

This calculated utilization is almost identical to our experimental results of 57% as presented in Table 7

This example of calculation is applicable to all the results we obtained in this research. It proves our point that the limiting factor (bottleneck) is on the router's capability to process packets rather than the network itself. Therefore, to provide sound traffic engineering for VoIP we need to consider *pps* as well as *bps*.

When we use a Linux machine as a router, we are able to achieve a much higher call load, close to 470 calls/sec (Table 8). However, this number is still far below the link capacity of 100BaseTX. In our experiments, each router has only two interfaces. If the call load is constrained by the router capability, then adding more interfaces to the router would further lower the utilization for each link.

If a carrier has a high-end router, such as Cisco 12000 series with the capability of 4,000,000 pps, this router could handle up to:

4M ÷ (1000 ÷ 20) ÷ 4 = ***20,000 calls/sec***
(Based on the 20ms sampling interval)

This capacity would be sufficient to achieve the theoretical limit of G.711 on a gigabit link, but still fall short for G.729A on the same link. If we choose a more aggressive packet sampling rate, such as 10ms, this capacity would not meet the demand of G.711 for a single gigabit link while most routers have multiple gigabit links and OC-3/OC-12 links.

If the bottleneck is on a network device (as we observed in our experiments), using a compression scheme would not solve the congestion problem. This is because most commonly used codec schemes require the same packet throughput. In other words, compression will not reduce the number of packets generated. The choice of the packet sampling interval, 10ms vs. 20ms, would significantly change the Maximum Call Load as it directly affects the transmitted number of packets per second.

The theoretical Maximum Call Load, if calculated based on bandwidth consumption, increases with the increase of the packet sampling rate. The reason is that higher packet sampling rate is associated with larger packet size and less overhead.

It should also be noted that Robust Header Compression (ROHC defined in RFC 3409) for RTP/UDP/IP does not improve max call load if the limiting factor is on pps instead of bps. ROHC reduces the header overhead but does not reduce the number of packets.

7. Conclusion

The Erlang-B model has been used by the telecom industry to determine the call capacity of circuit-switched networks for many years. We are proposing to use the max call load for VoIP networks as a comparable measure to network trunks. With this modification, the Erlang-B model is applicable to determine the call capacity of VoIP networks.

Packet-switched networks, by nature, do not have the concept of blocking, and all incoming packets are accepted even if the new packets will add more loads on the network which could result in delay and packet loss. In the case of VoIP, this will cause quality degradation to the new calls as well as to the existing ones. The solution to this problem is to use a Call Admission Control (CAC) where call manager or softswitch can apply the Erlang-B model to implement a CAC algorithm to accept or reject an incoming call request.

The traditional approach of calculating the maximum call load is based on network bandwidth, and our experiments show that this approach fails to work on some routed networks with high speed links. Our experiments show that packet throughput (pps) of network devices could be the constraint for VoIP traffic engineering. Based on our findings, network engineers should calculate not only the physical bandwidth of network interfaces but also the capacity of network devices. If the device capacity is the limiting factor, codec schemes would have no effect on the call capacity; instead, packet sampling interval could significantly change the maximum call load. For example, one of our experiments shows that increasing the packet sampling rate from 10ms to 20ms would increase the max call load by 37%. Of course, a higher packet sampling rate introduces longer delay which will adversely affect voice quality. Therefore, this is a trade-off between call capacity and call quality in traffic engineering.

We also acknowledge one deficiency in applying the Erlang-B for VoIP traffic. Many VoIP implementations support silence suppression. During the silence time, the VoIP end-device (an IP phone or a VoIP gateway) may transfer small number of packets while the Erlang-B model assumes the same packet transmission rate as the talking state. This issue could be addressed by applying a new model for traffic intensity as presented in [17], and such a model is a direction of our future research.

Acknowledgement

This research project is partially supported by the Quality Instruction Council (QIC) grant of DePaul University. The authors would like to thank ISP, Inc. at British Columbia, Canada for its generous donation of a high capability Linux server for the experiment.

REFERENCES

[1] Cisco, “Voice Design and Implementation Guide” http://www.cisco.com/en/US/tech/tk1077/technologies_tech_note09186a0080094a8b.shtml

[2] James Yu and Imad Al Ajarmeh, “Call Admission Control and Traffic Engineering of VoIP,” Second International Conference on Digital Communications, ICDT 2007, San Jose, CA, July 2007

[3] Cisco, “VoIP Call Admission Control” http://www.cisco.com/en/US/docs/ios/solutions_docs/voip_solutions/CAC.html

[4] Solange R. Lima, Paulo Carvalho, and Vasco Freitas. “Admission Control in Multiservice IP Networks: Architectural Issues and Trend,” *IEEE Communications,* Vol. 45 No. 4, April 2007, 114-121

[5] Erlang and VoIP Bandwidth Calculator, http://www.voip-calculator.com/calculator/eipb/

[6] Shenquan Wang, et. al. “Design and Implementation of QoS Provisioning System for Voice over IP,” *IEEE Transactions on Parallel and Distributed Systems*, Vol 17 No. 3, March 2006

[7] Xiuzhong Chen, et. al. “Survey on QoS Management of VoIP,” International Conference on Computer Networks and Mobile Computing, IEEE 20-23 October 2003, 68-77.

[8] R. F. Rey (editor) “Engineering and Operations in the Bell System,” AT&T Bell Laboratories, 1983. pp. 158-160

[9] Richard Parkinson, “Traffic Engineering Techniques in Telecommunications”, Infotel Systems Corporation, April 2002

[10] Erlang on-line Calculator, http://www.erlang.com/calculator/

[11] Karen Van Blarcum, “VoIP Call Recording – Understanding The Technical Challenges of VoIP Recording”, AudioCode Inc. White Paper, December 2004

[12] Bruce Thompson and Xiaomei Liu, “Bandwidth Management for the University Edge,” Cisco, NCTA 2005

[13] John Downey, “Understanding VoIP Packet Sizing and Traffic Engineering,” SCRE Cable-Tec Expo White Paper (June 2005) http://www.recursosvoip.com/docs/english/cdccont_0900aecd802c52e5.pdf

[14] One way Transmission time, ITU-T Recommendation G.114, May 2003

[15] A. Markopoulou, F. Tobagi, and M. Karam, "Assessing the Quality of Voice Communications over Internet Backbones", in IEEE/ACM Transactions on Networking, Vol.11, Issue 5, October 2003, pp.747-760.

[16] Cisco Portable Product Sheet – Router Performance http://www.cisco.com/web/partners/downloads/765/tools/quickreference/routerperformance.pdf

[17] Jorn Seger, “Modeling Approach for VoIP Traffic Aggregations for Transferring Tele-traffic Trunks in a QoS enabled IP-Backbone Environment”, International Workshop on Inter-Domain Performance and Simulation, Austria, February 2003.